

**ÍNDICE GENERAL**

<b>I</b>	<b>INTRODUCCIÓN .....</b>	<b>5</b>
<b>I.1°</b>	<b>OBJETIVOS.....</b>	<b>7</b>
<b>I.2°</b>	<b>FASES DEL DESARROLLO DEL PROYECTO.....</b>	<b>7</b>
<b>I.3°</b>	<b>MEDIOS EMPLEADOS .....</b>	<b>8</b>
<b>I.4°</b>	<b>COMPONENTES EMPLEADOS .....</b>	<b>8</b>
<b>1</b>	<b>CONCEPTOS GENERALES DE ONDAS ULTRASÓNICAS.....</b>	<b>9</b>
<b>1.1</b>	<b>ULTRASONIDOS.....</b>	<b>11</b>
1.1.1	Teoría física de los ultrasonidos .....	11
1.1.2	Generadores de ultrasonidos.....	12
1.1.3	Aplicaciones de los ultrasonidos .....	13
<b>1.2</b>	<b>PIEZOELECTRICIDAD .....</b>	<b>15</b>
1.2.1	Transductores piezoeléctricos .....	16
<b>1.3</b>	<b>EFEECTO DEL AIRE SOBRE LAS ONDAS ULTRASÓNICAS.....</b>	<b>17</b>
1.3.1	Velocidad de propagación de los ultrasonidos .....	17
1.3.2	Atenuación de la señal de ultrasonidos.....	19
<b>2</b>	<b>SENSORES DE ULTRASONIDOS .....</b>	<b>21</b>
<b>2.1</b>	<b>CARACTERÍSTICAS DE LOS SENSORES .....</b>	<b>23</b>
<b>2.2</b>	<b>ANTENAS PARABÓLICAS.....</b>	<b>24</b>
<b>3</b>	<b>ETAPAS DE POTENCIA.....</b>	<b>25</b>
<b>3.1</b>	<b>INTRODUCCIÓN.....</b>	<b>27</b>
<b>3.2</b>	<b>ETAPA DE TRANSMISIÓN .....</b>	<b>27</b>
<b>3.3</b>	<b>ETAPA DE RECEPCIÓN.....</b>	<b>30</b>
3.3.1	Grupo amplificador .....	30
3.3.2	Grupo seleccionador.....	32
<b>3.4</b>	<b>ALIMENTACIÓN DEL CIRCUITO.....</b>	<b>35</b>
<b>4</b>	<b>CIRCUITO DE CONTROL .....</b>	<b>37</b>
<b>4.1</b>	<b>INTRODUCCIÓN.....</b>	<b>39</b>
<b>4.2</b>	<b>GENERACIÓN DE LA SEÑAL DE 40 KHZ.....</b>	<b>39</b>
<b>4.3</b>	<b>CONTAJE DEL TIEMPO TRANSCURRIDO EN EL VIAJE DE LA ONDA.....</b>	<b>41</b>
<b>4.4</b>	<b>MEDIDA DE LOS FACTORES AMBIENTALES.....</b>	<b>41</b>

4.4.1	Medida de la temperatura .....	42
4.4.2	Medida de la humedad relativa .....	43
<b>4.5</b>	<b>CONTROL SOBRE EL “LCD” Y RECEPCIÓN DE LOS PULSADORES.....</b>	<b>45</b>
4.5.1	Control sobre el LCD .....	45
4.5.2	Recepción de los pulsadores .....	46
<b>4.6</b>	<b>ALIMENTACIÓN DEL CIRCUITO DE CONTROL .....</b>	<b>47</b>
<b>4.7</b>	<b>ESQUEMA ELÉCTRICO DEL CIRCUITO DE CONTROL.....</b>	<b>48</b>
<b>5</b>	<b>PROGRAMA DEL MICROCONTROLADOR .....</b>	<b>49</b>
<b>5.1</b>	<b>INTRODUCCIÓN .....</b>	<b>51</b>
<b>5.2</b>	<b>ARCHIVO PRINCIPAL .....</b>	<b>51</b>
5.2.1	Etapa de inicializado .....	54
5.2.2	Etapa indicadora de memoria vacía (MemoriaVacía) .....	54
5.2.3	Etapa de bifurcación (Bifurca) .....	56
5.2.4	Etapa de identificación del origen .....	57
5.2.5	Etapa de medida (MEDIR).....	58
5.2.6	Etapa de almacenado en memoria (Memorizar).....	60
5.2.7	Etapa de lectura (LEER) .....	61
5.2.8	Etapa de calibración del medidor (Calibrado).....	62
5.2.9	Subrutina que direcciona la posición inmediata superior (Incremen) .....	63
5.2.10	Subrutina que direcciona la posición inmediata inferior (Decremen).....	64
5.2.11	Subrutina de espera a la desactivación de los pulsadores (TeclasActivas) .....	64
5.2.12	Subrutina de espera a las pulsaciones sin rebotes (Antirrebotes).....	65
5.2.13	Subrutina de representación en el LCD “Posición” (UbicaPos) .....	65
<b>5.3</b>	<b>ARCHIVO VERMEN.....</b>	<b>66</b>
<b>5.4</b>	<b>ARCHIVO CALCULO .....</b>	<b>66</b>
<b>5.5</b>	<b>ARCHIVO LINELCD .....</b>	<b>67</b>
<b>5.6</b>	<b>ARCHIVO ACCESMEN.....</b>	<b>67</b>
<b>5.7</b>	<b>ARCHIVO PARAMETR .....</b>	<b>67</b>
<b>5.8</b>	<b>ARCHIVO SENULT .....</b>	<b>68</b>
<b>5.9</b>	<b>ARCHIVO TABLAS .....</b>	<b>69</b>
<b>6</b>	<b>DISEÑO DE CIRCUITOS .....</b>	<b>73</b>
<b>6.1</b>	<b>INTRODUCCIÓN .....</b>	<b>75</b>
<b>6.2</b>	<b>DIAGRAMA DE BLOQUES .....</b>	<b>75</b>
<b>6.3</b>	<b>CIRCUITOS ELÉCTRICOS.....</b>	<b>75</b>
6.3.1	Etapa de transmisión .....	75
6.3.2	Circuito de control.....	76
6.3.3	Etapa de recepción .....	76
<b>6.4</b>	<b>PLACAS DE CIRCUITO IMPRESO .....</b>	<b>77</b>
6.4.1	Cara de componentes .....	77
6.4.2	Cara de pistas .....	77

6.4.3	Placa para los pulsadores.....	78
<b>6.5</b>	<b>FOTOGRAFÍAS DE LA PLACA CONSEGUIDA.....</b>	<b>78</b>
6.5.1	Cara de pistas .....	78
6.5.2	Cara de componentes.....	78
6.5.3	Pantalla LCD y pulsadores .....	79
6.5.4	Sensores de ultrasonidos .....	80
6.5.5	Montaje completo.....	80
<b>6.6</b>	<b>COSIDERACIONES EN LA REALIZACIÓN DE LA PLACA .....</b>	<b>80</b>
<b>7</b>	<b>GUÍA DE USUARIO .....</b>	<b>83</b>
7.1	INTRODUCCIÓN.....	85
7.2	TABLAS DE CARACTERISTICAS.....	85
7.3	REALIZACIÓN DE MEDIDAS.....	86
7.4	LECTURA DE MEDIDAS .....	86
7.5	BORRADO DE LA MEMORIA.....	86
7.6	INFORMACIÓN ADICIONAL <b>SOBRE EL lcd.....</b>	<b>86</b>
<b>8</b>	<b>PRESUPUESTO.....</b>	<b>87</b>
8.1	EVALUACIÓN DE COSTE DEL PROTOTIPO.....	89
8.1.1	Coste del material.....	89
8.1.2	Coste del diseño .....	90
8.2	EVALUACIÓN DE COSTE DEL PRODUCCIÓN.....	90
8.2.1	Coste del material.....	90
8.2.2	Coste de la mano de obra .....	91
8.3	PRECIO DE SALIDA AL MERCADO .....	91
<b>9</b>	<b>CONCLUSIONES .....</b>	<b>93</b>
9.1	CONCLUSIONES <b>TÉCNICAS.....</b>	<b>95</b>
9.2	CONCLUSIONES PERSONALES .....	95
<b>A</b>	<b>APÉNDICE RECOPIULATORIO DE IMÁGENES VARIAS .....</b>	<b>97</b>
A.I	SEÑALES CAPTURADAS CON EL OSCILOSCOPIO .....	99
A.II	FOTOGRAFÍAS DEL MEDIDOR EN SUS DISTINTAS FUNCIONES .....	100
A.III	FOTOGRAFÍAS DEL MODO EN REALIZAR LOS ENSALLOS.....	101
<b>B</b>	<b>APÉNDICE DEL CÓDIGO DE PROGRAMA.....</b>	<b>103</b>
B.I	ARCHIVO PRINCIPAL .....	105
B.II	ARCHIVO VERMEN.....	116

<b>B.III</b>	<b>ARCHIVO CÁLCULO .....</b>	<b>123</b>
<b>B.IV</b>	<b>ARCHIVO LINELCD .....</b>	<b>126</b>
<b>B.V</b>	<b>ARCHIVO ACCESMEM.....</b>	<b>130</b>
<b>B.VI</b>	<b>ARCHIVO PARAMETR .....</b>	<b>136</b>
<b>B.VII</b>	<b>ARCHIVO SENULT .....</b>	<b>143</b>
<b>B.VIII</b>	<b>ARCHIVO TABLAS .....</b>	<b>147</b>
<b>C</b>	<b>APÉNDICE DE HOJAS DE ESPECIFICACIONES.....</b>	<b>159</b>

# I INTRODUCCIÓN

## ÍNDICE:

I.1º	OBJETIVOS .....	7
I.2º	FASES DEL DESARROLLO DEL PROYECTO.....	7
I.3º	MEDIOS EMPLEADOS .....	8
I.4º	COMPONENTES EMPLEADOS .....	8



La gran cantidad de utensilios, aparatos y maquinas, que aportan una mejor calidad de vida a la humanidad, deben su existencia al avance de la ciencia en el amplio abanico de materias que abarca, siendo la electrónica, la materia que más se desarrolla en la actualidad, gracias a la investigación.

Con este Proyecto Fin de Carrera se pone de manifiesto, como gracias a los avances de la electrónica, se consigue formar y ampliar las utilidades de un **Medidor de Distancia por Ultrasonidos**, gracias al uso del microcontrolador. El microcontrolador empleado, es el PIC16F876A el cual lo fabrica **al** casa Microchip Technology.

### I.1° OBJETIVOS

La principal utilidad que se persigue dar a este medidor, es visualizar la medida de distancia, que previamente ha sido calculada con los parámetros capturados, por el propio microcontrolador. Como utilidades secundarias se encuentran, la visualización de la temperatura y la humedad relativa del ambiente y la posibilidad de guardar la medida que desee el usuario, dentro de la memoria de 60 posiciones, reservadas en la “EEPROM” del microcontrolador.

### I.2° FASES DEL DESARROLLO DEL PROYECTO

- ◆ Diseño del hardware, con el que se busca la relación de componentes, que dan las mejores prestaciones respecto las etapas amplificadoras de los transductores ultrasónicos y que permiten la mejor configuración para los cálculos que lleva a cabo el microcontrolador.
- ◆ Desarrollo del software, con el que se realiza, desde la medida de los parámetros necesarios para esta aplicación, hasta la comunicación con el usuario, pasando por todos los cálculos y otras operaciones que dan lugar a los resultados buscados.
- ◆ Realización de ensayos, para la detección de posibles errores sobre el funcionamiento del medidor, con el fin de eliminarlo o compensarlo.
- ◆ Creación del prototipo, con el que se consigue un eslabón entre el diseño realizado, y una posible comercialización de este producto.

### I.3° MEDIOS EMPLEADOS

- ◆ Dentro de los programas informáticos, se encuentran:
  - + MPLAB IDE versión 5.70, con él que se realiza la programación del microcontrolador (PIC16F876A).
  - + IC-Prog versión 1.05C, con él que se realiza la grabación del microcontrolador (PIC16F876A).
  - + Proteus 6, con el que se realiza la simulación de circuitos eléctricos y el diseño de placas de circuito impreso.
  - + ISO.CAM y DrillPro-Route-Pro, con los que se dirige la maquina fresadora.
  - + Microsoft Word 97, con el que se ha realizado este informe.
- ◆ Placas Board, sobre las que se realizan los distintos circuitos de pruebas.
- ◆ Un osciloscopio digital, fuente de alimentación y polímetro, con los que se han realizado las pruebas de texteo de señales necesarias, sobre el circuito.
- ◆ Una sala amplia (Laboratorio 1019), donde se han realizado los ensayos de medida de distintas distancias.
- ◆ Maquina fresadora, con la que se realizan las placas de circuito impreso.

### I.4° COMPONENTES EMPLEADOS

- ◆ Un microcontrolador PIC16F876A, el cual ejecutar el programa que tiene cargado.
- ◆ Una pareja de transductores de ultrasónicos, los encargados de realizar la transformación de la señal eléctrica en una onda acústica y viceversa, por medio del transmisor y el receptor respectivamente.
- ◆ Circuitos integrados para la amplificación de la señal, tanto en la transmisión como en la recepción.
- ◆ Un LCD y pulsadores, que sirven de útiles comunicativos entre el usuario y el microcontrolador.
- ◆ Elementos discretos, como son resistencias y condensadores.

# 1 CONCEPTOS GENERALES DE ONDAS ULTRASÓNICAS

## ÍNDICE:

<b>1.1</b>	<b>ULTRASONIDOS .....</b>	<b>11</b>
1.1.1	Teoría física de los ultrasonidos .....	11
1.1.2	Generadores de ultrasonidos .....	12
1.1.3	Aplicaciones de los ultrasonidos.....	13
<b>1.2</b>	<b>PIEZOELECTRICIDAD.....</b>	<b>15</b>
1.2.1	Transductores piezoeléctricos .....	16
<b>1.3</b>	<b>EFECTO DEL AIRE SOBRE LAS ONDAS ULTRASÓNICAS.....</b>	<b>17</b>
1.3.1	Velocidad de propagación de los ultrasonidos .....	17
1.3.2	Atenuación de la señal de ultrasonidos.....	19



## 1.1 ULTRASONIDOS

### 1.1.1 Teoría física de los ultrasonidos

Antes de entrar en un pequeño estudio de las propiedades y características de los ultrasonidos, conviene dar una definición de los mismos. Por ultrasonidos se entiende que son ondas propagadas a través de un material en cualquiera de sus tres estados (sólido, líquido o gaseoso), cuyo rango de frecuencias comienza en el umbral superior de la audición humana (20 kHz), y termina alrededor de  $10^9$  Hz. Siendo por lo tanto imposibles de detectar por el oído humano.

El descubrimiento de los ultrasonidos data del año 1883, por los trabajos realizados por “Galton”, quien con un resonador de alta frecuencia, pudo medir el límite superior de frecuencia al que respondía el oído humano.

El modo de propagación de los ultrasonidos, sobre los diferentes medios materiales, es análogo a la forma en que se propagan las ondas sonoras, con la salvedad de que la absorción para los ultrasonidos es mayor, reduciéndose por lo tanto el alcance de la señal. En oposición con esta limitación, del corto alcance, aparece la ventaja de que las ondas ultrasónicas presentan una longitud de onda de pequeño tamaño, lo cual es primordial en un amplio abanico de aplicaciones, además de facilitar el estudio de su propagación, por poderlas considerar como ondas planas. Es por este motivo que la transmisión se realiza en línea recta y la energía no se puede desplazar a través de discontinuidades.

Para comprobar que las ondas de ultrasonidos poseen una longitud de onda pequeña, es conveniente saber, que la longitud de onda se define como la distancia, en el espacio, que abarca una oscilación completa de la onda a estudiar. De este modo, se relaciona la longitud de onda de cualquier señal ondulatoria, con la velocidad de propagación de dicha onda, de forma directa y con la frecuencia de la misma onda, de forma inversa, como se representa de forma matemática en la siguiente expresión.

$$\left. \begin{array}{l} e = v * t \\ t \equiv T = \frac{1}{f} \end{array} \right\} \Rightarrow e \equiv \lambda = v * \frac{1}{f} \Rightarrow \boxed{\lambda = \frac{v}{f}}$$

Como la mayoría de avances de la tecnología, la primera aplicación que se dio a los ultrasonidos, fue con fines bélicos, en la primera guerra mundial, para sondeos subacuáticos con el fin de detectar submarinos enemigos. Afortunadamente en el futuro se han hecho uso de estos avances, incluso mejorándolos, para aplicarlos en fines constructivos, comenzando por la localización de icebergs, continuando con la: detección de defectos en piezas, medición de espesores, validación de soldaduras, elaboración de reacciones químicas, y entrando en campos de la medicina y la bioquímica.

Los progresos en la teoría y aplicaciones de los ultrasonidos llevan un crecimiento exponencial desde hace algunos años, existiendo una enorme variedad de aparatos que responden a especificaciones muy concretas. Según el empleo de la onda ultrasónica, las aplicaciones se pueden clasificar en dos grupos, donde el 1º grupo se caracteriza por el uso de la energía transmitida para realizar la función deseada, y el 2º grupo se caracteriza por el análisis energético y geométrico del rayo utilizado. En el primer grupo, la emisión de la onda ultrasónica se realiza de forma continua, mientras que en el segundo grupo la onda se emite con forma de impulsos separados en el tiempo.

### 1.1.2 Generadores de ultrasonidos

Los generadores de ultrasonidos se definen como, dispositivos de los que se obtiene una onda ultrasónica en un determinado medio, para ello lo que hacen es transformar la energía recibida (eléctrica, magnética o mecánica) en energía mecánica, la cual se propaga en forma de onda con la frecuencia característica del generador.

Existen distintos tipos de generadores de ultrasonidos, y se pueden agrupar según la energía que se les suministra. Dentro de los generadores sensibles a energía eléctrica, destacan los: piezoeléctricos, electrostáticos, emisores de chispa, etc. dentro de los generadores sensibles a energía magnética, destacan los: magnetostrictivos, electromagnéticos, vibradores, etc. Dentro de los generadores sensibles a energía mecánica, destacan: los emisores por frotamiento, sirenas, etc.

Debido a los avances en el campo de la electrónica se hará un principal hincapié en los generadores ultrasónicos eléctricos y electrónicos los cuales están

destinados a la generación de ondas con una frecuencia determinada o en un rango de muy limitado de frecuencias. La energía eléctrica la aporta un circuito electrónico, el cual oscila a la frecuencia deseada para la onda ultrasónica, generada por el transductor vibratorio.

### 1.1.3 Aplicaciones de los ultrasonidos

A lo largo de la historia, desde que se descubrieron los ultrasonidos, se pueden distinguir tres momentos señalados en los que apareció un avance en las utilidades de los ultrasonidos. En 1918, Langevin, consiguió un transmisor de ultrasonidos, con un cristal de cuarzo piezoeléctrico puesto en resonancia. En 1927, Wood y Loomis, demostraron la posibilidad de obtener multitud de efectos físicos y biológicos, con el empleo de ondas ultrasónicas muy intensas. Tras la segunda guerra mundial, en 1945, se desarrollo la técnica de impulsos, abriendo un gran abanico de utilidades, tanto a nivel científico, como a nivel industrial.

Por medio de la propagación de haces concentrados de ultrasonidos, se han conseguido dispositivos de sondeo, de localización y de ensayo de materiales.

En cada aplicación del uso de las ondas de ultrasonidos, hay que tener en cuenta los factores que se enuncia a continuación.

- ◆ La frecuencia.
- ◆ La potencia radiada.
- ◆ La duración de las radiaciones.
- ◆ La superficie de radiación.
- ◆ Mapa energético sobre la superficie radiante.
- ◆ Influencia del medio, donde se incluye la atenuación, la velocidad de las ondas, la presión acústica, etc.

A continuación se clasifican las aplicaciones de los ultrasonidos según el criterio de los principales campos en los que se separa la ciencia investigadora de nuestros días.

◆ Guiado y sondeo

En este campo se pueden incluir las siguientes utilidades, el sondeo del fondo del mar y restos submarinos (sonar), la medida de la profundidad de las capas líquida, la navegación con la detección de obstáculos (icebergs), o de objetos buscados (bancos de peces), la señalización submarina, el estudio hidrográfico, la detección y estimación de yacimientos minerales y petrolíferos, etc.

◆ Medicina y biología

En este campo se pueden incluir las siguientes utilidades, el tratamiento de algunas enfermedades, la diagnosis por el empleo del ultrasonoscopio, el efecto sobre las bacteriofagas, y la destrucción de las bacterias estafilocócicas, el desarrollo de antibióticos, la destrucción de plagas de pequeños peces y batracios, etc.

◆ Tratamiento de productos alimenticios

En este campo se pueden incluir las siguientes utilidades, la aceleración del envejecimiento del vino y licores, la esterilización de productos alimenticios, la homogeneización de la leche, la emulsión de la mayonesa, etc.

◆ Aplicaciones físicas

En este caso se pueden incluir las siguientes utilidades, el estudio de algunas características de los sólidos, la determinación de las propiedades físicas de líquidos y gases, la detección de ruidos y vibraciones en determinados ambientes, el aumento de temperatura de sustancias, la localización de baches de aire en la navegación aérea, el comportamiento de explosiones, la rotura de estructuras cristalinas, etc.

◆ Aplicaciones químicas

En este caso se pueden incluir las siguientes utilidades, la transformación de compuestos químicos, la mejora en el proceso de reacciones de oxidación, la impresión de placas fotográficas, el tratamiento de óxidos de hierro, etc.

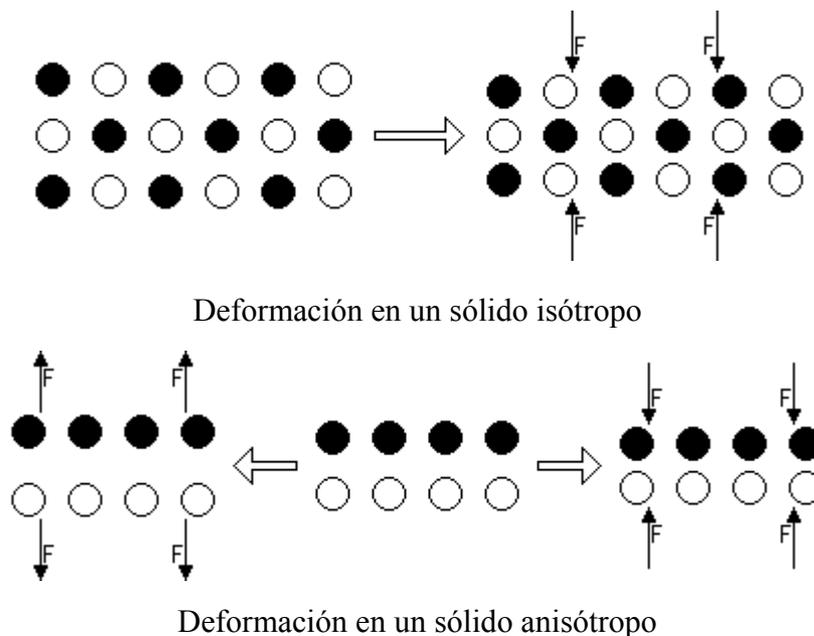
◆ Aplicaciones técnicas

En este caso se pueden incluir las siguientes utilidades, la detección de defectos y medida de espesores en piezas mecánicas, la apertura automática de puertas, el ensayo de vibraciones en aviación, etc.

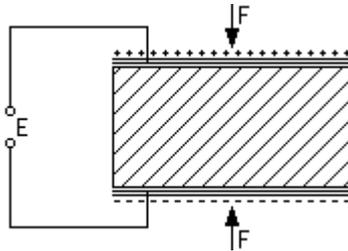
Aunque en algunas de las aplicaciones mencionadas se ha alcanzado un estado de perfección muy notable, siempre existe la posibilidad de un mayor acercamiento a la optimización, con ayuda de las novedades tecnológicas, descubiertas en los centros de investigación. Estos avances suelen aparecer por la necesidad de mejorar las aplicaciones que se encuentran en desarrollo.

## 1.2 PIEZOELECTRICIDAD

La piezoelectricidad es una propiedad común, de los materiales, en los que aparecen la agrupación de cargas eléctricas, debido a la deformación que experimentan al aplicarles un esfuerzo mecánico. La existencia de esta característica en los materiales, exige que estos sean sólidos cristalinos anisótropos, es decir, que el material presente distintas propiedades en las diferentes direcciones del espacio. Esto hace que se puedan separar las cargas eléctricas, con las deformaciones mecánicas, como se distingue en las siguientes figuras.



En términos eléctricos, un cristal de estas características se comporta como un condensador, donde el dieléctrico lo forma el propio material piezoeléctrico, como se muestra en la siguiente figura.



Muchos cristales poseen esta propiedad y los mas conocidos son: el cuarzo, la sal de Rochelle, la turmalina, la blenda, el ácido tartarico, el azúcar de caña, etc.

Con esta característica de los materiales se pueden distinguir el efecto piezoeléctrico directo, en el cual se transforma la energía mecánica en energía eléctrica, del efecto piezoeléctrico inverso, el cual proporciona energía mecánica al aplicar un potencial eléctrico.

El descubrimiento del efecto piezoeléctrico directo data del año 1880, por los hermanos Curie y el efecto piezoeléctrico inverso se predijo en 1881 por Lippmann, completándose el estudio en 1890 por W. Voigt.

Los cristales piezoeléctricos se emplean en la construcción de aparatos de encendido electrónico, reguladores de la frecuencia de los aparatos de radio, relojes de cuarzo, etc.

### 1.2.1 Transductores piezoeléctricos

Una utilidad que se da a los materiales piezoeléctricos, debido a la reversibilidad de su efecto característico, es la de transductores, distinguiéndose los emisores, con los que se permite generar ondas ultrasónicas, al aportarles energía eléctrica, de los receptores, con los que se puede detectar estas mismas ondas, realizando el proceso de transformación de vibraciones a señales eléctricas.

Aunque en la naturaleza existen materiales que presentan el efecto piezoeléctrico, en la fabricación de transductores se emplean materiales cerámicos, como el titanato-circonato de plomo (PZT), o materiales plásticos, como el fluoruro

de polivinilideno (PVF<sub>2</sub>), los cuales presentan un efecto piezoeléctrico neto gracias a que en su fabricación han sufrido un proceso llamado prepolarización.

Aun que la complejidad del diseño de los transductores piezoeléctricos es alta, su utilidad es muy variada, debido a las grandes ventajas que ofrecen, como se enuncian a continuación:

- ◆ Puede trabajar en márgenes de frecuencia que van desde las décimas de Hz hasta los GHz.
- ◆ Al ser sólido el material activo, su alta impedancia específica, facilita la transferencia de energía a otros sólidos.
- ◆ Sus bajas pérdidas internas les permite trabajar con alto rendimiento y en bandas de frecuencia muy estrechas.

### 1.3 EFECTO DEL AIRE SOBRE LAS ONDAS ULTRASÓNICAS

La transmisión de las ondas sonoras en el aire, se puede considerar como el movimiento de estas ondas en un gas ideal. Por tratarse del estudio de ultrasonidos, se puede considerar a este tipo de ondas como ondas planas, debido a que su alta frecuencia proporciona una baja longitud de onda. Finalmente hay que destacar, que las condiciones ambientales influyentes sobre las ondas acústicas, son la temperatura y la humedad relativa del fluido sobre el que se desplazan, que para este estudio, será el aire.

La temperatura del ambiente influye en la velocidad con que se transmiten estas ondas ultrasónicas, de forma individual, y en unión con la humedad relativa ambiental, se ve afectada la atenuación de dichas ondas.

#### 1.3.1 Velocidad de propagación de los ultrasonidos

Como se ha distinguido antes, la velocidad con que se propagan las ondas ultrasónicas, solo se ve alterada por la variación de la temperatura ambiental. Estos dos parámetros se pueden relacionar en forma de ecuación, la cual se obtiene realizando los siguientes cálculos.

La ecuación con la que se parte, determina la velocidad de propagación de la onda en el aire, en función de la presión, de la densidad y del índice adiabático del gas.

$c = \sqrt{\gamma \times \frac{P_0}{\rho_0}}$ ; donde:	$P_0$ : es la presión del aire en el estado de equilibrio $\rho_0$ : es la densidad del aire en el estado de equilibrio $\gamma$ : es el índice adiabático del gas
--	--

Por tener el conocimiento de estos términos en estado de equilibrio a una temperatura de 0 °C, se determina la velocidad de las ondas en esta situación.

$$\left. \begin{array}{l} T = 0^\circ C \\ c \rightarrow c_0 \end{array} \right\} \Rightarrow \left\{ \begin{array}{l} \gamma = 1.402 \\ P_0 = 1.013 * 10^5 \text{ N/m}^2 \\ \rho_0 = 1.293 \text{ Kg/m}^3 \end{array} \right\} \rightarrow c_0 = \sqrt{1.402 \times \frac{1.013 * 10^5}{1.293}}$$

$c_0 = 331.6 \text{ m/s}$

Una vez conseguida la velocidad de las ondas acústicas a la temperatura de 0 °C, aplicando la ecuación de estado del gas ideal, se puede conseguir una expresión, donde se relacione la velocidad con la temperatura del aire, por él que se propaga la onda.

$P \times V = n \times R \times T$ ; donde:	$P$ : es al presión a la que se encuentra el gas $V$ : es el volumen que ocupa el gas $n$ : es el numero de moles del gas $R$ : es la constante universal de los gases $T$ : es la temperatura a la que se encuentra el gas en "K"	
$n = \frac{m}{M}$ ; donde:	$m$ : es la masa del gas $M$ : es la masa molar del gas	Se consigue: $P \times V = \frac{m}{M} \times R \times T$
$m = V \times \rho$ ; donde:	$V$ : es el volumen del gas $\rho$ : es la densidad del gas	Resulta: $P \times V = \frac{V \times \rho}{M} \times R \times T$
Realizando operaciones	Sustituyendo	Finalmente se obtiene
$\frac{P}{\rho} = \frac{R}{M} \times T$	$\frac{R}{M} = r$ ; parámetro constante	$\frac{P}{\rho} = r \times T$

Ahora solo quedan sustituir los factores necesarios, para lograr la dependencia de la velocidad del sonido, con la temperatura.

$$\left. \begin{array}{l} c = \sqrt{\gamma \times \frac{P}{\rho}} \\ \frac{P}{\rho} = r \times T \end{array} \right\} \rightarrow c = \sqrt{\gamma \times r \times T}$$

Con el conocimiento de esta dependencia buscada y la velocidad del sonido en unas condiciones de temperatura ambiental fija, se puede llegar a una ecuación, con la que se calcule, la velocidad de las ondas en su desplazamiento por el aire.

$$\left. \begin{array}{l} c_T = \sqrt{\gamma \times r \times T} \\ c_0 = \sqrt{\gamma \times r \times T_0} \end{array} \right\} \rightarrow \frac{c_T}{c_0} = \sqrt{\frac{T}{T_0}} \Rightarrow c_T = c_0 \times \sqrt{\frac{T}{T_0}}$$

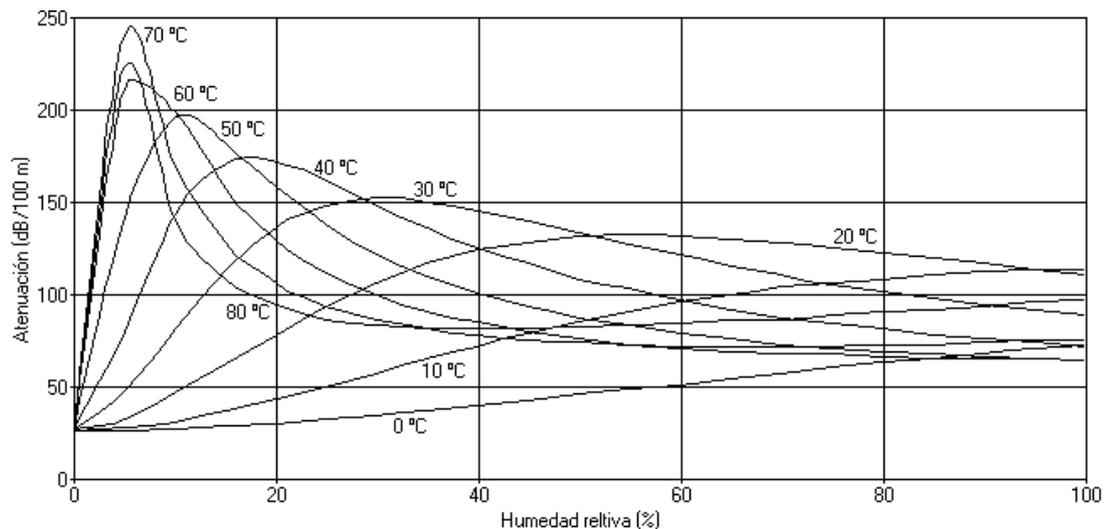
Para equiparar esta ecuación a los valores de la temperatura en °C y aplicando el dato conocido de la velocidad para 0 °C, se obtiene la ecuación deseada.

$$\left. \begin{array}{l} T(K) = 273 + t(^{\circ}C) \\ c_0 = 331.6 \frac{m}{s} \end{array} \right\} \rightarrow c_t = 331.6 \times \sqrt{\frac{273+t}{273}} \Rightarrow c_t = 331.6 \times \sqrt{1 + \frac{t}{273}}$$

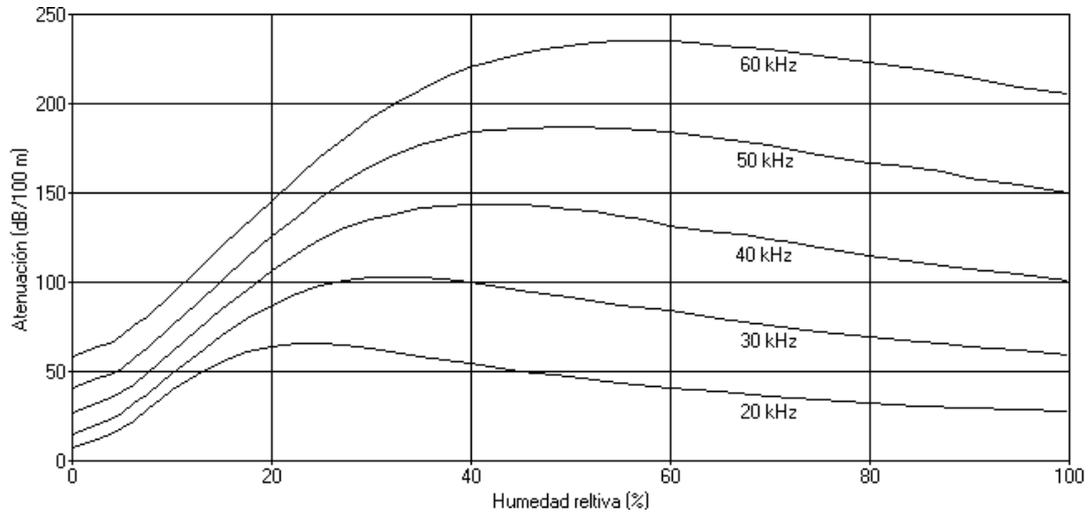
### 1.3.2 Atenuación de la señal de ultrasonidos

Las ondas ultrasónicas tendrán fijado su máximo alcance, por la atenuación, la cual depende de la humedad relativa y la temperatura del ambiente por el que se desplazan dichas ondas.

Para tener una idea de la influencia de estos dos parámetros ambientales sobre la atenuación, se hace uso de la siguiente gráfica, en la cual se relaciona la atenuación que sufre una onda de 40 kHz en una distancia de 100 m para distintos valores de temperatura, en función de la humedad relativa.



También hay que destacar que esta atenuación es diferente para distintas frecuencias de la señal ultrasónica, como se puede apreciar en la siguiente gráfica donde la temperatura esta fijada a 25 °C.



## **2 sensores de ultrasonidos**

### **Índice:**

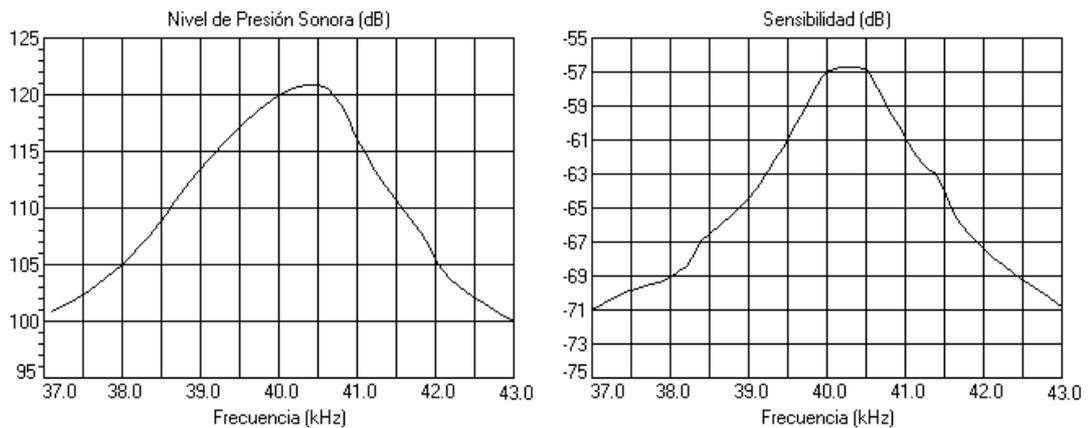
<b>2.1</b>	<b>características de los sensores.....</b>	<b>23</b>
<b>2.2</b>	<b>antenas parabólicas .....</b>	<b>24</b>



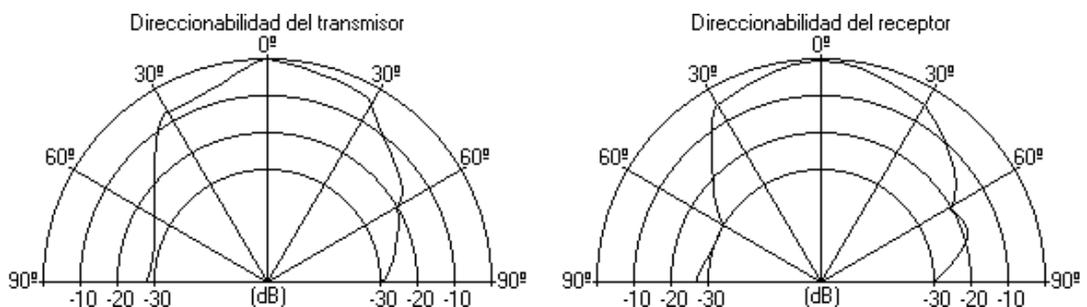
## 2.1 CARACTERÍSTICAS DE LOS SENSORES

Los sensores de ultrasonidos empleados en este aparato de medida, es una pareja de transductores piezoeléctricos cerámicos, diseñados para trabajar **apareados**. El elemento transmisor, transforma la energía eléctrica en energía mecánica, apareciendo ondas sobre el ambiente. Con el elemento receptor, se detectan estas ondas acústicas, realizando la transformación de la energía mecánica en energía eléctrica.

Estos transductores, por ser piezoeléctricos, **se tiene** limitada la frecuencia de operación dentro de un rango, viéndose fijado el valor óptimo por las dimensiones de la pieza cerámica piezoeléctrica. En las siguientes gráficas se muestra el nivel de presión sonora del transmisor y la sensibilidad del receptor, en función de la frecuencia.



La carcasa exterior de los sensores, hace que las ondas acústicas (de salida para el transmisor y de entrada para el receptor) se vean diseccionadas de la forma que se muestra en la siguiente figura.



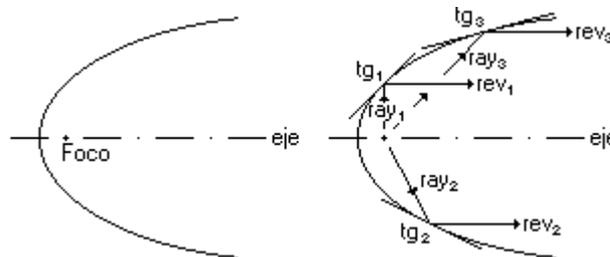
Para completar las características de la pareja de traductores se utiliza la siguiente tabla.

Característica	Transmisor	Receptor
Frecuencia nominal (kHz)	40±1	40±1
Nivel de presión sonora (dB)	>115; (0dB = 0.02mPa)	
Sensibilidad (dB)		>-65; (0dB = 1V/μbar)
Dirección (°)	80	80
Temperatura de uso (°C)	-40↔80	-40↔80
Distancia mínima detectable (m)		0.2
Distancia máxima detectable (m)		4
Resolución (mm)		9
Máxima tensión de entrada (Vpp)		20

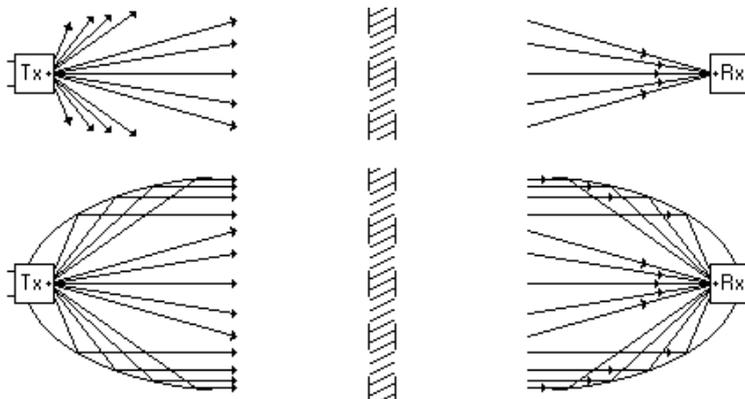
## 2.2 ANTENAS PARABÓLICAS

La manera de ampliar la distancia detectable, es con la introducción de cada sensor en un paraboloide de revolución, de forma que el material piezoeléctrico del sensor, se encuentre en el foco del paraboloide.

La característica ventajosa que presenta el foco de un paraboloide de revolución, es la coincidencia, con el eje, en dirección y sentido, de todos los rayos rebotados en la pared del paraboloide, originados en el punto focal. Esta peculiaridad se define gráficamente con la siguiente figura.



La ventaja que presenta la utilización de este tipo de antenas, se puede ver en la siguiente figura, donde se comparan los resultados, al emplear los sensores sin antenas parabólicas, de emplearlos con estas antenas.



### **3 ETAPAS DE POTENCIA**

#### **ÍNDICE:**

<b>3.1</b>	<b>INTRODUCCIÓN .....</b>	<b>27</b>
<b>3.2</b>	<b>ETAPA DE TRANSMISIÓN.....</b>	<b>27</b>
<b>3.3</b>	<b>ETAPA DE RECEPCIÓN.....</b>	<b>30</b>
<b>3.3.1</b>	<b>Grupo amplificador.....</b>	<b>30</b>
<b>3.3.2</b>	<b>Grupo seleccionador .....</b>	<b>32</b>
<b>3.4</b>	<b>ALIMENTACIÓN DEL CIRCUITO .....</b>	<b>35</b>



### 3.1 INTRODUCCIÓN

Para conseguir una mejor conversión de la señal eléctrica a onda acústica en el transmisor y la conversión inversa en el receptor se hace uso de unas etapas de potencia, las cuales se explican a continuación.

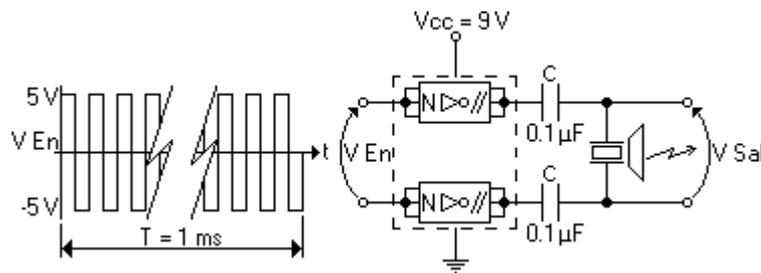
### 3.2 ETAPA DE TRANSMISIÓN

El circuito electrónico empleado en la etapa de transmisión, esta formado por puertas inversoras, según indica la hoja de especificaciones de los sensores de ultrasonidos, pero se han realizado modificaciones hasta conseguir la mejor señal de salida, sin aumentar el numero de circuitos integrados.

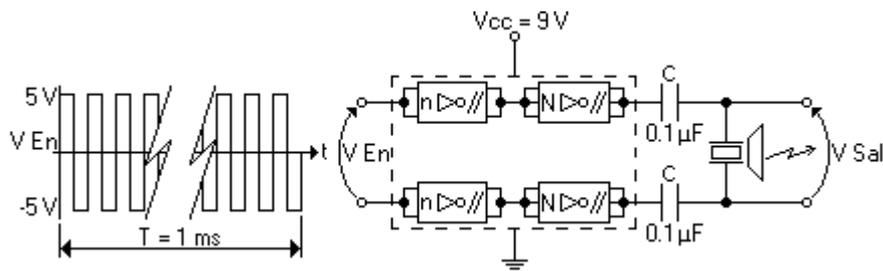
Las pruebas realizadas para evaluar las señales de salida de las diferentes configuraciones del circuito transmisor, se llevan a cabo con una señal de entrada de 40 kHz y de 1 ms de duración, donde la amplitud de la señal eléctrica es de  $\pm 5$  V.

A continuación se muestran las configuraciones genéricas de los circuitos con que se han realizado dichas pruebas. Además se representan las gráficas, de la señal entregada al transmisor, para los distintos casos, donde varía el numero de puertas asociadas en paralelo.

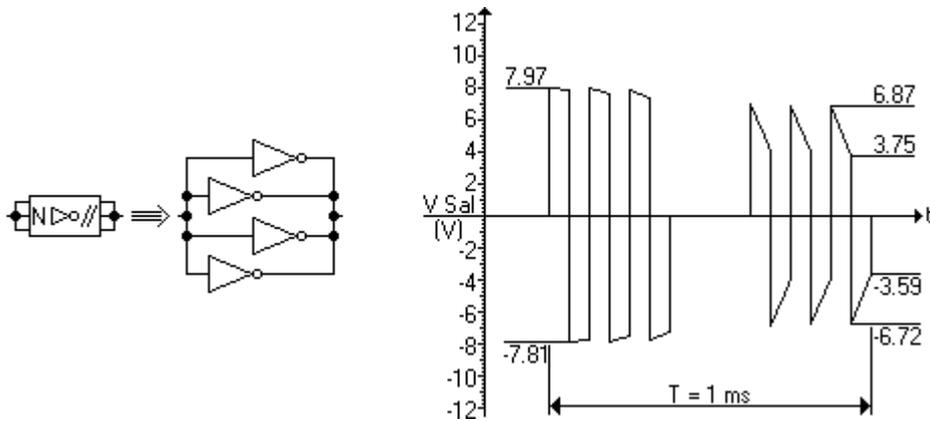
Configuración de una sola agrupación de puertas en paralelo.



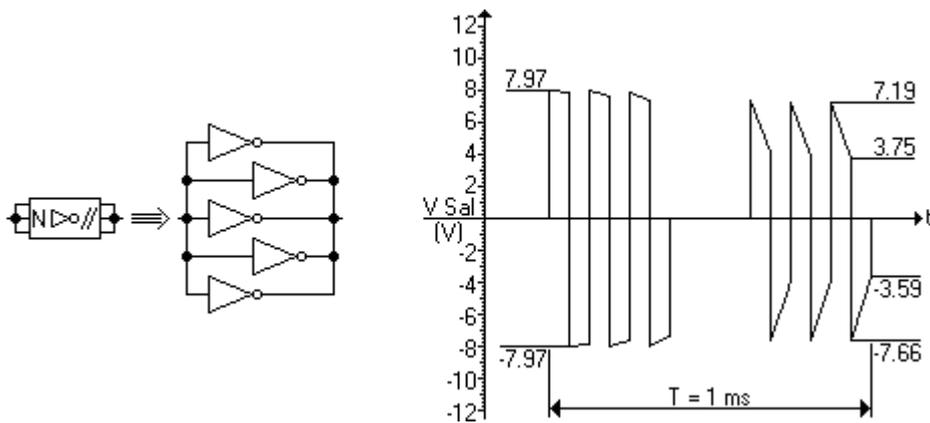
Configuración de dos agrupaciones en serie de puertas en paralelo.



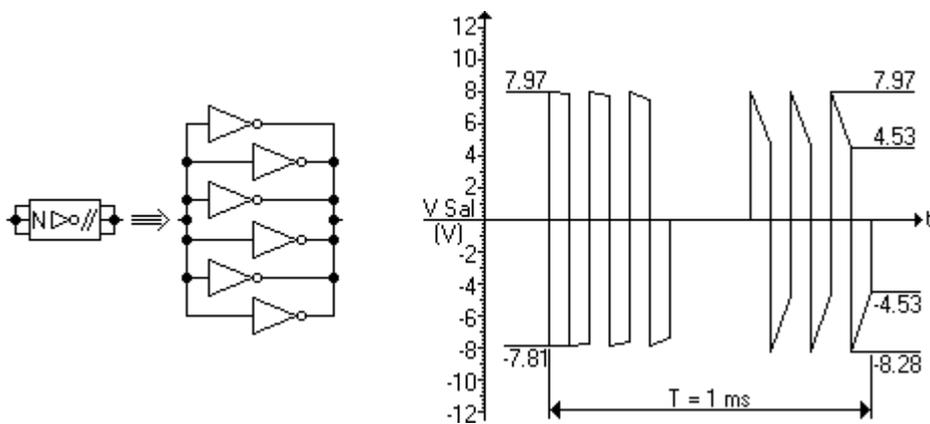
Para una sola agrupación de 4 “NOT” en paralelo, se obtiene la siguiente salida.



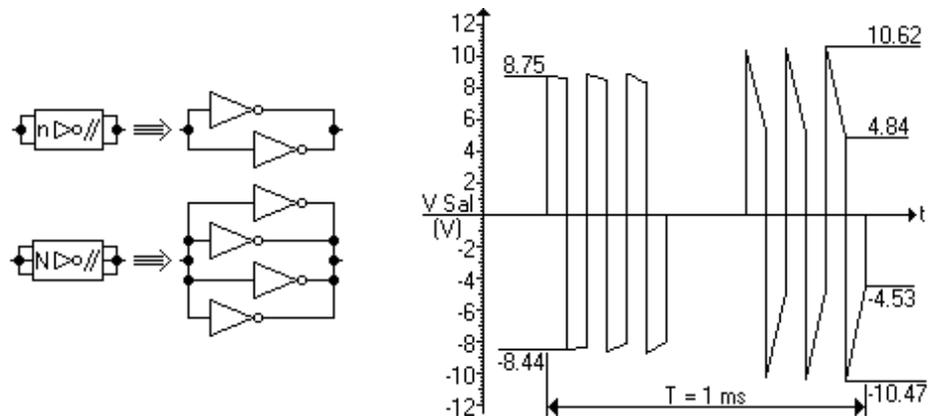
Para una sola agrupación de 5 “NOT” en paralelo, se obtiene la siguiente salida.



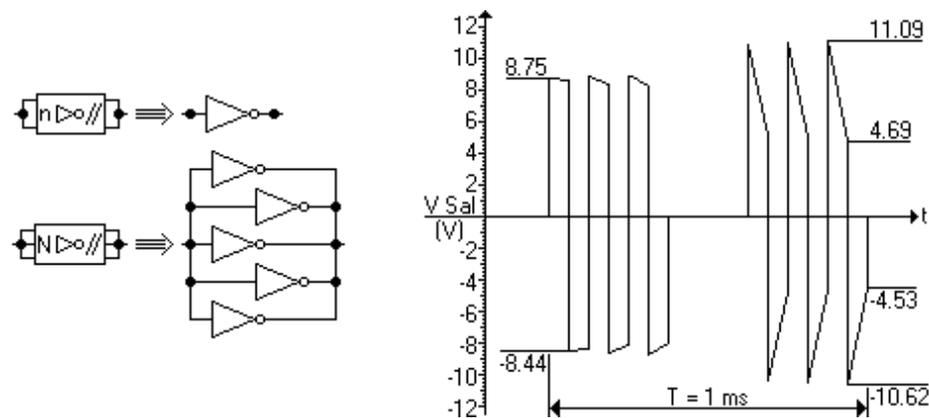
Para una sola agrupación de 6 “NOT” en paralelo, se obtiene la siguiente salida.



Para dos agrupaciones en serie, donde la inicial dispone de 2 “NOT” y la conectada al transmisor dispone de 4 “NOT” en paralelo, se obtiene la siguiente salida.

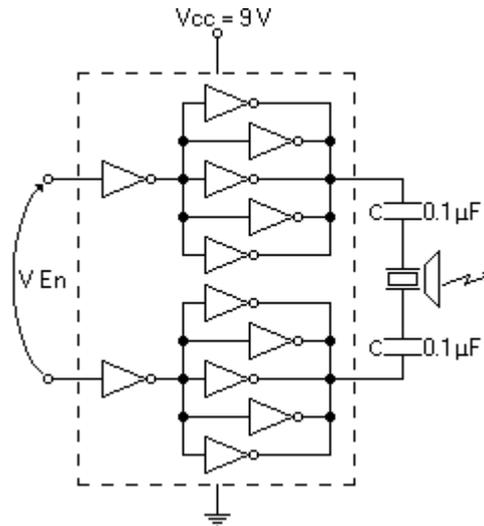


Para dos agrupaciones en serie, donde la inicial dispone de 1 “NOT” y la conectada al transmisor dispone de 5 “NOT” en paralelo, se obtiene la siguiente salida.



Evaluando las señales eléctricas aplicada al sensor transmisor, se escoge la configuración de dos agrupaciones en serie de puertas inversoras, donde la señal generada se aplica a una puerta inversora, y la salida de esta puerta, se introduce en la agrupación en paralelo de cinco puertas inversoras, las cuales entregan la señal, con un alto nivel de corriente, al transmisor.

El esquema del circuito eléctrico escogido se muestra a continuación.



La realización de este circuito se lleva a cabo con puertas inversoras de tecnología MOST, debido a la alimentación de 9 V, y también se exige que estas puertas entreguen alta corriente, para mejorar las prestaciones de la transmisión de la onda acústica. Con estas condiciones el dispositivo elegido es el “4069”.

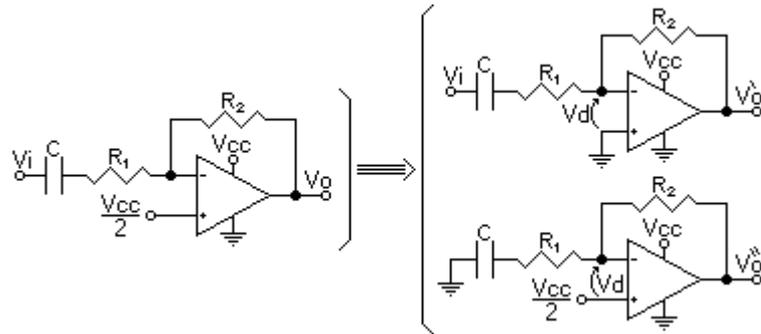
### 3.3 ETAPA DE RECEPCIÓN

El circuito electrónico empleado en la etapa de recepción, esta formado por dos grupos, donde con el primero, se amplifica la señal eléctrica aportada por el sensor receptor de ultrasonidos, y con el segundo, se seleccionan las señales recibidas que superan los niveles de tensión fijados como umbrales.

#### 3.3.1 Grupo amplificador

El modo de amplificar la señal eléctrica alterna, asociada al eco recibido, con amplificadores alimentados con tensión positiva ( $V_{cc}-0$ ), es con una configuración de realimentación negativa y centrado la señal de salida a un nivel de continua de “ $V_{cc}/2$ ”. De esta forma se consigue una amplificación completa de los niveles de entrada, tanto los positivos, como los negativos.

Los cálculos explicativos de esta teoría se exponen a continuación, justificando la elección del circuito eléctrico del grupo amplificador.



Por tener el circuito dos tensiones de entrada, se aplica el teorema de superposición, y de esta forma se consigue la señal de salida “Vo” como la suma de las señales de salida correspondientes a cada entrada.

Para el circuito donde se tiene en cuenta la entrada alterna “Vi”, el condensador no supone ninguna influencia sobre el valor eficaz de la corriente “i1”, por lo que el circuito equivalente a estudiar es el que se muestra a continuación.

	<p><b>Teoría del Amplificador Operacional</b></p> <p><math>V_d \rightarrow 0 \rightarrow V^-</math> es una masa virtual</p> <p><math>Z(\text{entrada}) \rightarrow \infty \rightarrow i_1 = i_2</math></p>
--	--

Aplicando la teoría del amplificador operacional, se puede obtener el valor de “Vo” en función de “Vi”.

$$i_1 = \frac{V_i}{R_1} = i_2 = \frac{-V_{O'}}{R_2}; \text{ despejando "Vo"} \text{ se consigue: } \boxed{V_{O'} = -\frac{R_2}{R_1} \times V_i}$$

Para el circuito donde se tiene en cuenta la componente continua “Vcc/2”, el condensador se comporta como un circuito abierto en régimen permanente, puesto que no existe un camino por el que descargarse de dicho valor de tensión. El circuito equivalente a estudiar en esta situación se muestra a continuación.

	<p><b>Teoría del Amplificador Operacional</b></p> <p><math>V_d \rightarrow 0 \rightarrow V^-</math> vale <math>V_{cc}/2</math></p> <p><math>Z(\text{entrada}) \rightarrow \infty \rightarrow i_1 = i_2</math></p>
--	---

Aplicando la teoría del amplificador operacional, para el estado de régimen permanente se puede obtener el valor de “Vo” en función de “Vcc/2”.

$$i_1 = \frac{V_{cc} - V_{cc}}{2R_1} = i_2 = \frac{V_{cc} - V_{o''}}{2R_2}; \text{ despejando "Vo''"} \text{ se consigue: } \boxed{V_{o''} = \frac{V_{cc}}{2}}$$

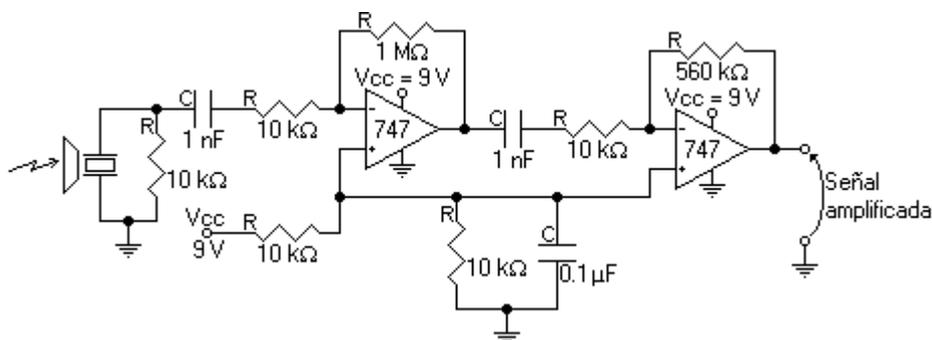
El valor de "Vo" resulta de la suma de ambas componentes.

$$V_o = V_{o'} + V_{o''} = -\frac{R_2}{R_1} \times V_i + \frac{V_{cc}}{2} \Rightarrow \boxed{V_o = -\frac{R_2}{R_1} \times V_i + \frac{V_{cc}}{2}}$$

A la vista de esta expresión, que define la señal de salida, se distingue la ganancia de amplificación, con el cociente de la tensión de salida (Vo) entre la tensión de entrada (Vi), sin tener en cuenta la componente continua. La expresión se muestra a continuación.

$$\boxed{G = \frac{V_o}{V_i} = -\frac{R_2}{R_1}}$$

Para concluir con el estudio de este primer grupo de la etapa receptora, se muestra el circuito que opera de amplificador de la señal eléctrica referente a las ondas de ultrasonidos recibidas.



La ganancia total de este circuito resulta del producto de la ganancia de cada amplificador.

$$G_1 = -\frac{1 \cdot 10^6}{10 \cdot 10^3} \rightarrow G_1 = -100 \text{ y } G_2 = -\frac{560 \cdot 10^3}{10 \cdot 10^3} \rightarrow G_2 = -56$$

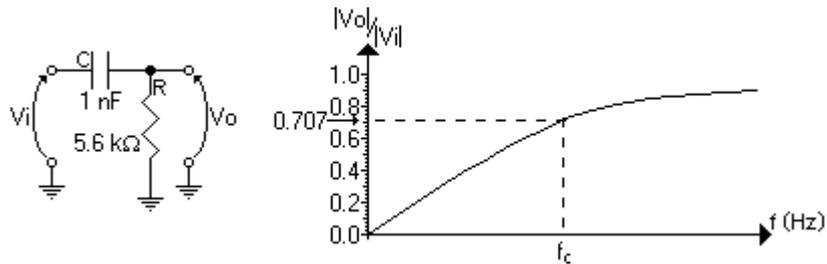
$$G_T = G_1 \times G_2 \rightarrow G_T = -100 \times -56 \Rightarrow \boxed{G_T = 5.6 \cdot 10^3}$$

### 3.3.2 Grupo seleccionador

Con este circuito se entrega un pulso eléctrico, siempre que las señales eléctricas amplificadas, corresponden a las ondas acústicas de un eco directo. Por

medio de niveles umbrales se pueden distinguir las señales eléctricas asociadas a las ondas acústicas deseables, del resto de ondas acústicas existentes en el ambiente.

En primer lugar hay que eliminar la componente continua de la señal amplificada, con el uso de un filtro pasaalta, permitiendo el paso de frecuencia de la onda a detectar (40 kHz). Este filtro pasaalta se forma con una red “RC”, y a continuación se muestra el circuito correspondiente, junto con la gráfica que representa la ganancia en función de la frecuencia.



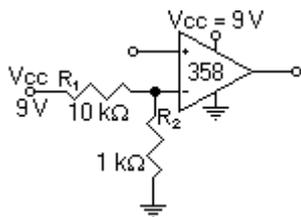
Con la ecuación de la frecuencia de corte, se determina el limite inferior del rango de frecuencias que permite pasar dicho filtro.

$$f_c = \frac{1}{2 \times \pi \times R \times C}; \left\{ \begin{array}{l} C = 1nF \\ R = 5.6k\Omega \end{array} \right\} \rightarrow f_c = \frac{1}{2 \times \pi \times 5.6 * 10^3 \times 1 * 10^{-9}}$$

$$\boxed{f_c = 28.42kHz}$$

Tras ser filtrada la señal amplificada, es necesario eliminar las señales eléctricas asociadas a ondas acústicas indeseadas. Por esta razón se hace uso de un comparador, el cual tiene fijado un nivel de tensión umbral. Este valor umbral, se a determinado de forma empírica con la elección de las resistencias que componen el divisor de tensión.

A continuación se muestra, el circuito que desempeña la función de comparador junto con la determinación del nivel umbral.

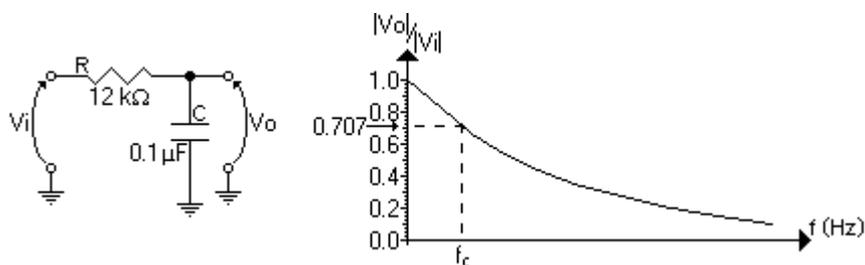


$$V_{UMBRAL} = V^- = \frac{R_2}{R_1 + R_2} \times V_{CC}$$

$$V_{UMBRAL} = \frac{1 * 10^3}{10 * 10^3 + 1 * 10^3} \times 9 \Rightarrow \boxed{V_{UMBRAL} = 0.82V}$$

Con la salida de tensión, en forma de pulsos, se lleva a cabo un integración de los mismos, con el uso de un filtro pasabajo, el cual elimina la frecuencia de 40 kHz.

Este filtro pasabajas también se forma con una red “RC”, y a continuación se muestra el circuito correspondiente, junto con la gráfica que representa la ganancia en función de la frecuencia.



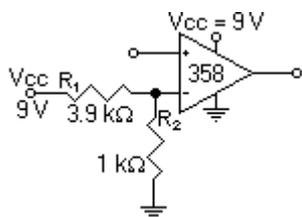
Con la ecuación de la frecuencia de corte, se determina el límite superior del rango de frecuencias que permite pasar dicho filtro.

$$f_c = \frac{1}{2 \times \pi \times R \times C}; \left\{ \begin{array}{l} R = 12k\Omega \\ C = 100nF \end{array} \right\} \rightarrow f_c = \frac{1}{2 \times \pi \times 12 * 10^3 \times 100 * 10^{-9}}$$

$$f_c = 133Hz$$

Por último, queda generar un pulso eléctrico, indicativo de que la onda acústica asociada a los niveles de tensión integrados, corresponde al eco directo de la onda emitida. Para llevar a cabo la generación de este pulso se emplea un segundo comparador, el cual, al igual que el anterior circuito, tiene fijada el valor de la tensión umbral de forma empírica, con la elección de las resistencias del divisor de tensión.

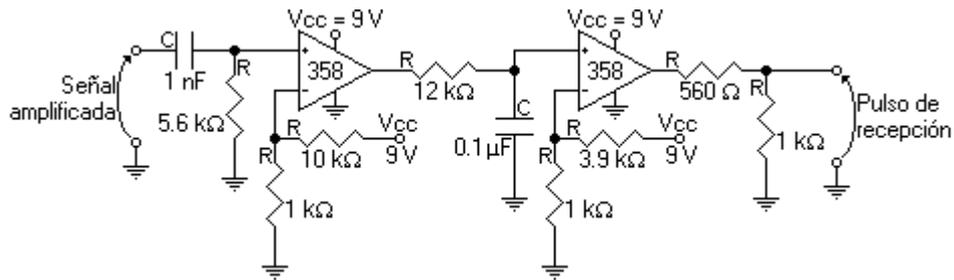
A continuación se muestra, el circuito que desempeña la función de comparador junto con la determinación del nivel umbral.



$$V_{UMBRAL} = V^- = \frac{R_2}{R_1 + R_2} \times V_{CC}$$

$$V_{UMBRAL} = \frac{1 * 10^3}{3.9 * 10^3 + 1 * 10^3} \times 9 \Rightarrow V_{UMBRAL} = 1.84V$$

Para concluir con el estudio del segundo grupo de la etapa receptora, se muestra el circuito que opera de selector de las señales eléctricas amplificadas, correspondientes a las ondas acústicas del eco directo.



Referente a los cálculos realizados para los filtros de frecuencias, hay que indicar, que son validos para el circuito en el que se han utilizado, puesto que la impedancia de entrada del componente al que se conecta, es de valor infinito, y por esta razón la corriente que circula por ambos componentes (resistencia y condensador) tiene el mismo valor.

Tampoco hay que olvidar la existencia del divisor de tensión, al final del circuito, empleado para la adaptación del pulso al circuito de control.

Los circuitos integrados empleados en este circuito son el “747” y el “LM358N”. Ambos son amplificadores operacionales, y el “747” dispone de dos amplificadores cuyas características son muy usuales, en cambio el “LM358N” esta formado por dos amplificadores de bajo consumo.

### 3.4 ALIMENTACIÓN DEL CIRCUITO

Por ser el medidor de distancias, un aparato que requiere un alto grado de desplazamiento, el tipo de alimentación más favorable, es el uso de baterías, descartándose la conexión a la red eléctrica de baja tensión. Dentro de la alimentación con baterías aparecen distintas posibilidades, y teniendo en cuenta el limite de tensión de los dispositivos que componen las etapas de potencia, se elige el uso de una batería de 9 V. Esta elección ofrece una un nivel de tensión apropiado y resulta ventajosa con respecto a la conexión en serie de baterías de menor voltaje, que para aportar el mismo nivel de tensión, ocupan un mayor volumen.



## **4 CIRCUITO DE CONTROL**

### **ÍNDICE:**

<b>4.1</b>	<b>INTRODUCCIÓN .....</b>	<b>39</b>
<b>4.2</b>	<b>GENERACIÓN DE LA SEÑAL DE 40 KHZ.....</b>	<b>39</b>
<b>4.3</b>	<b>CONTAJE DEL TIEMPO TRANSCURRIDO EN EL VIAJE DE LA ONDA .....</b>	<b>41</b>
<b>4.4</b>	<b>MEDIDA DE LOS FACTORES AMBIENTALES .....</b>	<b>41</b>
4.4.1	Medida de la temperatura .....	42
4.4.2	Medida de la humedad relativa.....	43
<b>4.5</b>	<b>CONTROL SOBRE EL “LCD” Y RECEPCIÓN DE LOS PULSADORES .....</b>	<b>45</b>
4.5.1	Control sobre el LCD .....	45
4.5.2	Recepción de los pulsadores .....	46
<b>4.6</b>	<b>ALIMENTACIÓN DEL CIRCUITO DE CONTROL .....</b>	<b>47</b>
<b>4.7</b>	<b>ESQUEMA ELÉCTRICO DEL CIRCUITO DE CONTROL .....</b>	<b>48</b>



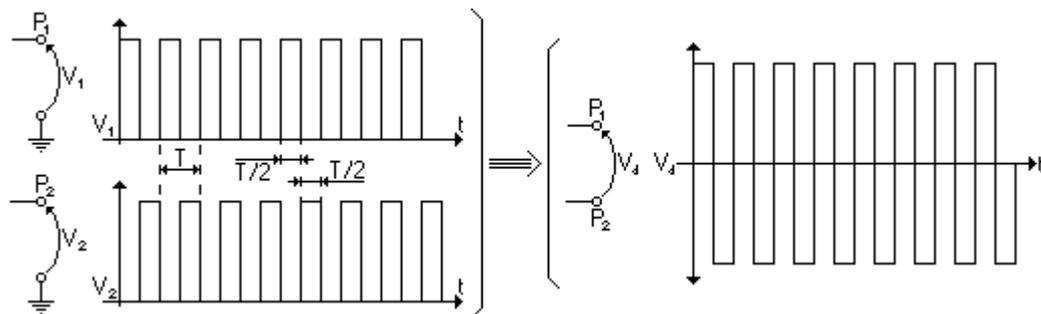
## 4.1 INTRODUCCIÓN

Para el circuito de control se emplea un microcontrolador de la casa Microchip Technology (PIC16F876A), capaz de ofrecernos todas las prestaciones necesarias para generar la señal de 40 kHz, también se encarga de contar el tiempo de viaje de las ondas ultrasónicas, realiza la medida de los factores ambientales, que influyen sobre dichas ondas, y además de recibir las señales de control de los pulsadores, dirige los mensajes a visualizar sobre el LCD.

## 4.2 GENERACIÓN DE LA SEÑAL DE 40 KHZ

La señal de 40 kHz aplicada a la etapa de transmisión, ha de ser alterna, y por el hecho de que la alimentación del microcontrolador es únicamente positiva (5-0) V, se necesita operar en modo diferencial, entre dos salidas del microcontrolador.

La salida en modo diferencial entre dos patillas se consigue, cuando el nivel de tensión de una patilla, toma como referencia la otra patilla, y viceversa. Para que la señal de salida además de alterna sea simétrica, por ambas patillas se ha de extraer una señal cuadrada con un ciclo de trabajo del 50 % y entre ellas desfasadas 180 °, de esta forma se consiguen semiciclos positivos y semiciclos negativos, como se muestra con las siguientes gráficas.



El mínimo intervalo de tiempo que ha de contar el microcontrolador, ha de ser el correspondiente a un semiperiodo, puesto que es la permanencia en alta de cada patilla. A continuación se muestran los cálculos realizados para identificar el máximo tiempo permitido en el ciclo de instrucción del microcontrolador y así determinar el cristal de cuarzo empleado en el oscilador.

$$T = \frac{1}{f}; f = 40kHz \rightarrow T = \frac{1}{40 * 10^3} \Rightarrow T = 25 * 10^{-6} seg \Rightarrow \boxed{T = 25\mu s}$$

$$T/2 = \frac{T}{2}; T = 25\mu s \rightarrow T/2 = \frac{25 * 10^{-6}}{2} \Rightarrow T/2 = 12.5 * 10^{-6} \text{ seg} \Rightarrow \boxed{T/2 = 12.5\mu s}$$

La forma de identificar el tiempo de duración del ciclo de instrucción, es haciendo que este sea submúltiplo del semiperiodo, por esta razón se tienen los siguientes valores de ciclos de instrucción posibles: 0.5  $\mu s$ , 0.25  $\mu s$ , 125 ns, etc.

Los ciclos de instrucción tienen una duración, cuatro veces superior, que el periodo de oscilación del cristal de cuarzo. Teniendo en cuenta esta explicación se puede relacionar, la frecuencia de oscilación del cristal, con su periodo de oscilación y la duración del correspondiente ciclo de instrucción, como se aprecia en la siguiente tabla.

Frecuencia del cristal (MHz)	Periodo de oscilación (ns)	Ciclo de instrucción ( $\mu s$ )
4	250	1
8	125	0.5
16	62.5	0.25
20	50	0.2

A la vista de estos valores, se tiene que elegir entre un cristal de 8 MHz o de 16 MHz, y teniendo en cuenta que el consumo de energía por parte del microcontrolador, es directamente proporcional a la rapidez con que se ejecuta el programa, se elige el cristal permitido de menor frecuencia, para reducir así el consumo de energía. Por ello se hará uso del cristal de cuarzo de 8 MHz.

Una vez definida la frecuencia del oscilador, se realiza el calculo del numero de instrucciones que debe ejecutar el microcontrolador, durante la permanencia en un semiciclo de la señal de salida.

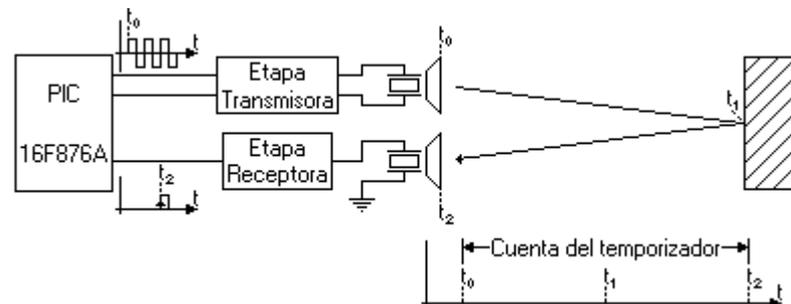
$$N^{\circ} \text{ Instrucciones} = \frac{\text{Semiperiodo}}{\text{CicloInstrucción}} \cdot \left\{ \begin{array}{l} \text{Semiperiodo} = 12.5 * 10^{-6} \\ \text{CicloInstrucción} = 0.5 * 10^{-6} \end{array} \right\}$$

$$N^{\circ} \text{ Instrucciones} = \frac{12.5 * 10^{-6}}{0.5 * 10^{-6}} \Rightarrow \boxed{N^{\circ} \text{ Instrucciones} = 25}$$

Con este dato, la forma de generar la señal alterna de 40 kHz, es con la conmutación del estado de las patillas utilizadas como salidas, durante la ejecución de 25 instrucciones.

### 4.3 CONTAJE DEL TIEMPO TRANSCURRIDO EN EL VIAJE DE LA ONDA

Con el empleo del temporizador, se pretende realizar la cuenta del tiempo transcurrido desde que se envía la onda ultrasónica, hasta que esta llega al extremo opuesto de la distancia deseada de medir. Por las razones de funcionamiento del medidor, el temporizador comenzara la cuenta con el envío de la onda ultrasónica, y finalizara, con la llegada del eco correspondiente a la onda emitida, como se muestra en el siguiente esquema.



Puesto que el intervalo de tiempo deseado, para la obtención de la distancia medida, es  $(t_0 - t_1)$  dado en “ $\mu\text{s}$ ”, es necesaria la configuración de un preescalado del temporizador, con el que se consigue corregir la cuenta de ciclos de instrucción durante el intervalo que permanece contando el temporizador  $(t_0 - t_2)$ . Con un factor  $\frac{1}{2}$ , se reduce a la mitad la cuenta de ciclos de instrucción, consiguiendo que el temporizador cuente “ $\mu\text{s}$ ” puesto que el ciclo de instrucción tiene una duración de 0.5  $\mu\text{s}$ . Con otro factor  $\frac{1}{2}$ , se reduce a la mitad la cuenta del temporizador, adaptando el intervalo  $(t_0 - t_2)$  al intervalo  $(t_0 - t_1)$ . Finalmente, se obtiene el resultado correcto almacenado sobre el registro del temporizador, al configurarlo con la preescala “ $\frac{1}{4}$ ”.

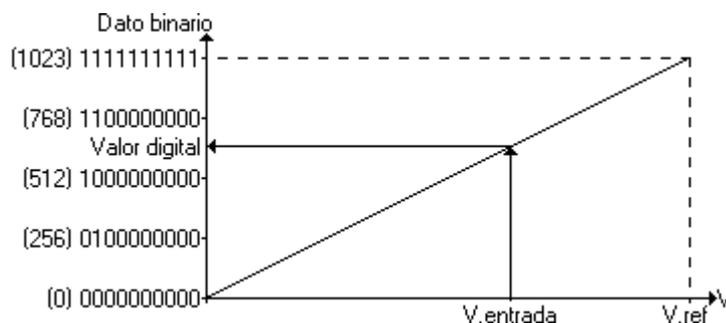
### 4.4 MEDIDA DE LOS FACTORES AMBIENTALES

Los factores ambientales que influyen sobre las ondas ultrasónicas son la temperatura y la humedad relativa del ambiente por el que se mueven. Ambos factores repercuten en la atenuación de la señal ultrasónica, mientras que la temperatura es el único factor influyente sobre la velocidad de propagación de las ondas acústicas.

El modo de realizar la medida de estos factores, es por medio de unos sensores, que entregan un nivel de tensión proporcional al valor del factor medido.

La lectura de estos niveles de tensión a determinar, se realiza con el conversor A/D del microcontrolador.

El conversor A/D empleado, almacena el resultado digital en un registro de 10 bit, presentando una variedad de  $2^{10}$  posibles valores, por lo que se completa el rango [0 – 1023]. El valor digital correspondiente a la tensión analógica del sensor, resulta de aplicar la misma proporción con **laque** al máximo valor digital (1023) le corresponde la tensión de referencia, la explicación gráfica se tiene con la siguiente figura.



El cálculo que se ha de realizar con el microcontrolador, para visualizar el correspondiente valor decimal sobre el LCD, es una regla de tres, como se muestra a continuación.

$$\left. \begin{array}{l} 1023 \rightarrow V.ref \\ VDigital \rightarrow ? \end{array} \right\} \Rightarrow ? = \frac{V.ref}{1023} \times VDigital$$

Este cálculo se lleva a cabo con la subrutina que realiza el producto entre dos términos, el dato almacenado en el registro del contador (VDigital) y la proporcionalidad fijada por la tensión de referencia (V.ref/1023).

#### 4.4.1 Medida de la temperatura

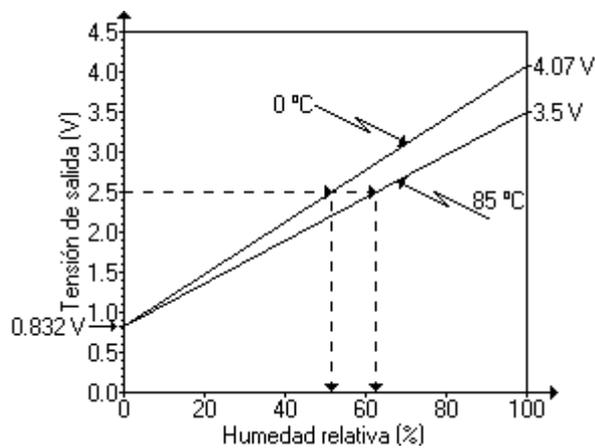
El sensor de temperatura elegido es el LM35DZ, el cual dispone de una proporcionalidad de 1 °C por cada 10 mV de tensión. Puesto que el rango de temperatura asociado a este sensor, es (0 – 100) °C, el valor de tensión entregado, se encuentra en el rango (0 – 1) V. Con el fin de conseguir una mayor precisión en la medida de la temperatura, se fija el nivel de tensión de referencia a un valor próximo al valor de 1 V, y el escogido es el estabilizador de tensión MAX6018, que entrega una tensión a la salida de 1.6 V.

Con este nivel de tensión de referencia, la precisión en la medida de la temperatura se consigue con el siguiente cálculo.

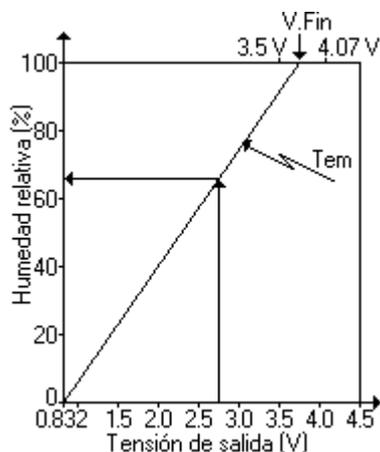
$$\Delta V = \frac{V.ref}{1023} \rightarrow \Delta V = \frac{1.6}{1023} \Rightarrow \boxed{\Delta V = 1.56mV} \quad \boxed{1.56mV \Leftrightarrow 0.156^\circ C}$$

#### 4.4.2 Medida de la humedad relativa

El sensor de humedad relativa elegido es el HIH-3610, el cual presenta una dependencia con la temperatura, puesto que a un mismo nivel de tensión, le corresponde distintos niveles de humedad relativa para diferentes temperaturas. En la siguiente gráfica se muestra la proporcionalidad entre el nivel de tensión de salida, y la humedad relativa para los límites de temperatura de operación.



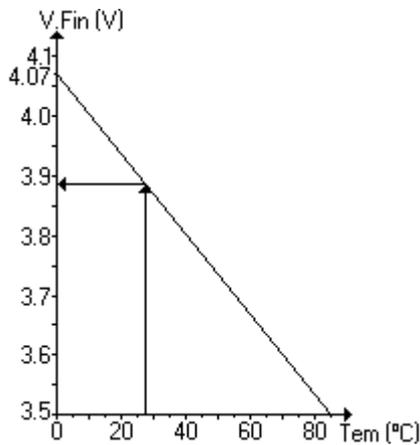
El modo de conseguir el valor de la humedad relativa en un ambiente donde se conozca la temperatura, es a partir del factor de proporcionalidad dado por el rango de tensión, en que se encuentra la salida de este sensor ( $V.Fin - 0.832$ ). Este factor de proporcionalidad, se consigue con la pendiente de la recta representada en la siguiente gráfica y calculada con la expresión mostrada.



Expresión de la pendiente

$$\boxed{Pendiente = \frac{100}{V.Fin - 0.832}}$$

También hay que determinar el valor que toma el limite superior de este rango de tensión, en función de la temperatura ambiental, puesto que existe una linealidad entre la temperatura y el limite superior de tensión. A continuación se expone la gráfica representativa de esta linealidad y la ecuación de la recta.



Expresión de la recta

$$V.Fin = \frac{4.07 - 3.5}{0 - 85} \times Tem + 4.07$$

Por ultimo, queda determinar la precisión obtenida en la medida de la humedad relativa, la cual viene fijada por la tensión de referencia con que se alimenta el conversor A/D, que para esta medida toma la alimentación del microcontrolador (5 V). La mínima precisión, se obtiene para la temperatura de 85 °C, puesto que su correspondiente rango de tensión es el menor, y por esta razón también será mínimo el numero de datos digitales de salida del conversor A/D.

Los valores numéricos, que toma el rango de datos digitales y la precisión mínima de la humedad, se muestran a continuación con los siguientes cálculos.

$$\Delta V = \frac{V.ref}{1023} \rightarrow \Delta V = \frac{5}{1023} \Rightarrow \boxed{\Delta V = 4.89mV}$$

$$\left. \begin{array}{l} 5V \rightarrow 1023 \\ (V.Fin - 0.832)V \rightarrow ? \end{array} \right\} \Rightarrow ? = \frac{V.Fin - 0.832}{5} \times 1023$$

$$N^{\circ} DatosDigitales(85^{\circ} C) = \frac{3.5 - 0.832}{5} \times 1023 \Rightarrow \boxed{N^{\circ} DatosDigitales = 546}$$

$$Re\ soluci\acute{o}n = \frac{RangoCompletoHR}{N^{\circ} DatosDigitales} \rightarrow Re\ soluci\acute{o}n(Min) = \frac{100\%}{546}$$

$$Re\ soluci\acute{o}n(Min) = 0.183\% \Rightarrow \boxed{4.89mV \Leftrightarrow 0.183\%}$$

## 4.5 CONTROL SOBRE EL "LCD" Y RECEPCIÓN DE LOS PULSADORES

El usuario del medidor, necesita una pantalla para visualizar en todo momento la operación que esta realizando, y unos pulsadores con los que se elige la decisión deseada de las ofrecidas.

### 4.5.1 Control sobre el LCD

El LCD elegido, dispone del controlador HD44780, el cual requiere una serie de líneas de comunicación, en las que hay que distinguir, tres terminales de control, y ocho terminales de datos, con los cuales existe una posibilidad añadida de una comunicación con un bus de cuatro hilos. No hay que olvidar que además de la alimentación se necesita una conexión, del terminal del contraste, a un nivel de tensión, y la forma de conseguir el máximo contraste es uniéndolo a masa.

La comunicación con el LCD, se realiza con una serie de subrutinas que se encuentran desarrolladas en un archivo incluido en el programa del microcontrolador.

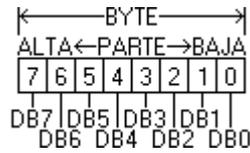
Los terminales de control y de datos se describen a continuación.

RS: es el terminal con el que se distingue el destino donde se almacena la información enviada por el bus de datos. Con el terminal a uno ( $RS = 1$ ), el destino es el registro de caracteres, los cuales se visualizaran en la pantalla, en cambio con el terminal a cero ( $RS = 0$ ), el destino es el registro de comandos, los cuales proceden a realizar operaciones sobre el LCD.

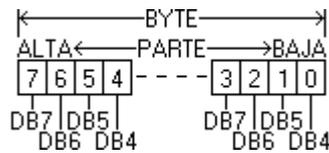
RW: es el terminal con el que se distingue la operación de lectura o escritura sobre el controlador del LCD. Con el terminal a uno ( $RW = 1$ ), se indica la operación de lectura, en cambio con el terminal a cero ( $RW = 0$ ), se indica que la operación de escritura.

EN: es el terminal con el que se distingue la habilitación, o la inhabilitación del LCD. Con el terminal a uno ( $EN = 1$ ), el LCD esta habilitado, mientras que con el terminal a cero ( $EN = 0$ ), el LCD esta deshabilitado.

DB7 – DB0: son los terminales que contienen el dato que se encuentra en el bus de 8 líneas. El Byte se dispone de la forma que muestra la siguiente figura.



DB7 – DB4: son los terminales que contienen el dato que se encuentra en el bus de 4 líneas. El modo de repartiéndose en el tiempo, el medio Byte alto y el medio Byte bajo, sobre las cuatro líneas existentes se muestra la siguiente figura.



#### 4.5.2 Recepción de los pulsadores

Existen cuatro pulsadores con los que se pueden utilizar todas las opciones que presenta el programa del microcontrolador.

Estos pulsadores tienen fijada una utilidad genérica sobre el programa, las cuales vienen serigrafiadas sobre el aparato medidor. En el caso que sobre la segunda línea del LCD, se muestren diferentes operaciones a poder llevar a cabo, se asociara al pulsador próximo a dicha opción, la posibilidad de poder entrar en ella, por medio de su activación.

Las utilidades genéricas asociadas a los pulsadores son las siguientes.

**←**: es el pulsador BAJAR, con el cual se realiza el decremento de la posición de memoria a la que se desea acceder, tanto en el proceso de lectura, como en el de escritura. Con el marcado de la segunda línea del LCD, este pulsador corresponde a la realización de medidas.

**X**: es el pulsador CANCELAR, con el que se deshace la opción seleccionada, retrocediendo a la etapa desde la que se había entrado. Con el marcador de la segunda línea del LCD, este pulsador corresponde a la realización del borrado de las medidas memorizadas.

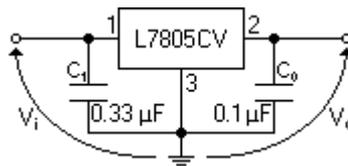
➔: es el pulsador SUBIR, con el cual se realiza el aumento de la posición de memoria a la que se desea acceder, tanto en el proceso de lectura, como en el de escritura. Con el marcado de la segunda línea del LCD, este pulsador corresponde a la lectura de las medidas guardadas.

V: es el pulsador ACEPTAR, con el que se valida la opción seleccionada, retrocediendo a la etapa desde la que se había entrado.

Puesto que los pulsadores originan rebotes durante su activación, en la patilla correspondiente del microcontrolador, el programa contenido en el mismo, dispone de la subrutina con la que se realiza la lectura de dichas patillas, tras haber acabado el intervalo de los rebotes ocurridos.

#### 4.6 ALIMENTACIÓN DEL CIRCUITO DE CONTROL

La tensión de alimentación requerida por el microcontrolador, para su deseado funcionamiento en estas condiciones, se encuentra entre 4.5 V y 5.5V. Además la utilización de esta tensión, como referencia para el conversor A/D del propio PIC, es necesaria una alta estabilidad en dicho nivel de tensión. El dispositivo con el que se consiguen estas prestaciones requeridas, es el L7805CV, y el circuito aconsejado para este dispositivo, en las hojas de especificaciones, se muestra en la siguiente figura.





## 5 PROGRAMA DEL MICROCONTROLADOR

### ÍNDICE:

<b>5.1</b>	<b>INTRODUCCIÓN</b> .....	<b>51</b>
<b>5.2</b>	<b>ARCHIVO PRINCIPAL</b> .....	<b>51</b>
5.2.1	Etapa de inicializado .....	54
5.2.2	Etapa indicadora de memoria vacía (Memoria Vacía) .....	54
5.2.3	Etapa de bifurcación (Bifurca).....	56
5.2.4	Etapa de identificación del origen.....	57
5.2.5	Etapa de medida (MEDIR).....	58
5.2.6	Etapa de almacenado en memoria (Memorizar) .....	60
5.2.7	Etapa de lectura (LEER) .....	61
5.2.8	Etapa de calibración del medidor (Calibrado) .....	62
5.2.9	Subrutina que direcciona la posición inmediata superior (Incremen).....	63
5.2.10	Subrutina que direcciona la posición inmediata inferior (Decremen) .....	64
5.2.11	Subrutina de espera a la desactivación de los pulsadores (Teclas Activas) .....	64
5.2.12	Subrutina de espera a las pulsaciones sin rebotes (Antirrebotes) .....	65
5.2.13	Subrutina de representación en el LCD “Posición” (UbicaPos).....	65
<b>5.3</b>	<b>ARCHIVO VERMEN</b> .....	<b>66</b>
<b>5.4</b>	<b>ARCHIVO CALCULO</b> .....	<b>66</b>
<b>5.5</b>	<b>ARCHIVO LINELCD</b> .....	<b>67</b>
<b>5.6</b>	<b>ARCHIVO ACCESMEN</b> .....	<b>67</b>
<b>5.7</b>	<b>ARCHIVO PARAMETR</b> .....	<b>67</b>
<b>5.8</b>	<b>ARCHIVO SENULT</b> .....	<b>68</b>
<b>5.9</b>	<b>ARCHIVO TABLAS</b> .....	<b>69</b>



## 5.1 INTRODUCCIÓN

El programa se organiza en una serie de archivos, donde se distingue un fichero principal, el cual presenta la estructura de dicho programa, y lleva asignados los archivos secundarios, sobre los cuales se encuentran las subrutinas y datos necesarios para la correcta ejecución de dicho programa.

Sobre la memoria de datos se encuentran los registros de función especial y registros de propósito general, estando estos últimos a disposición del usuario, para almacenar las variables necesarias en la ejecución del programa. Desde cada archivo, han sido definidas las variables que en él se han necesitado, consiguiéndose una colocación de forma consecutiva sobre la memoria de datos.

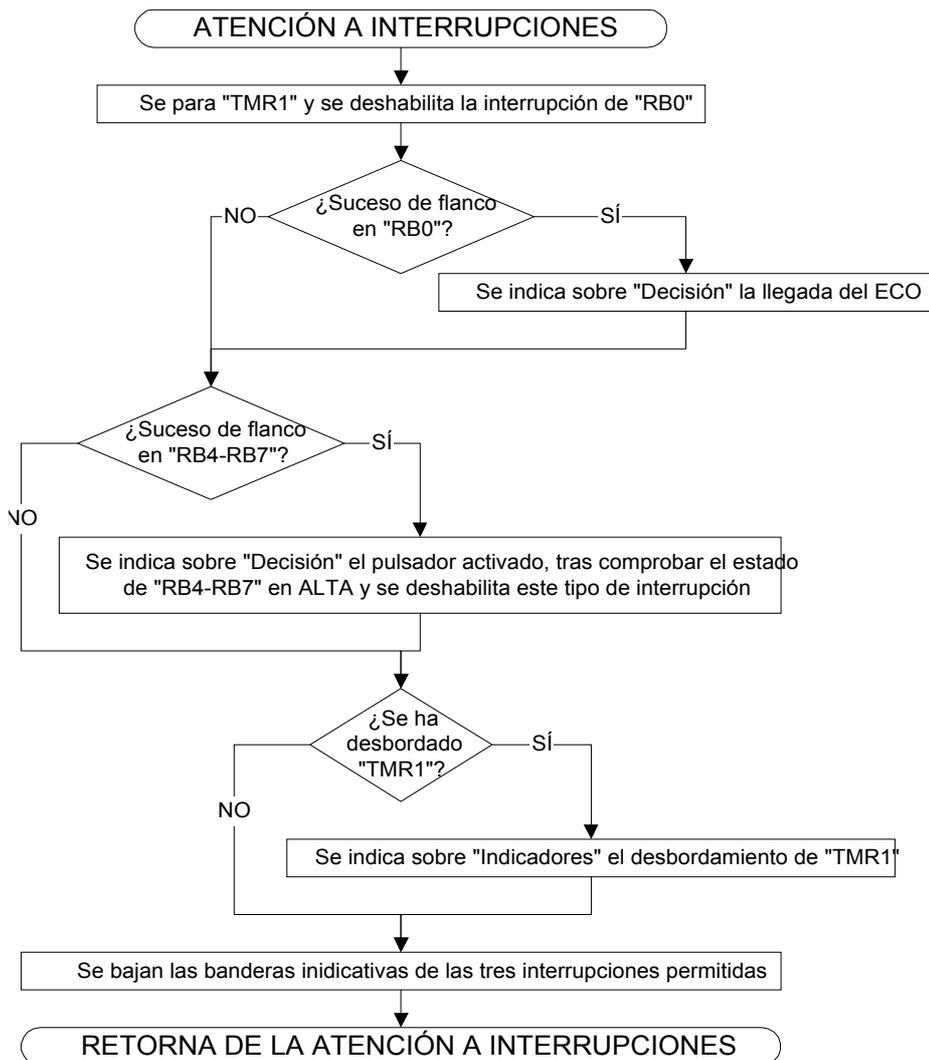
A continuación se detallan los archivos empleados por este programa.

## 5.2 ARCHIVO PRINCIPAL

Por ser empleado el lenguaje ensamblador para microcontroladores PIC, es necesario comenzar el programa, identificando el PIC elegido (PIC16F876A), junto con el *include* correspondiente a dicho PIC, donde se encuentran enunciados los registros de que dispone. También se identifica la base de los valores numéricos usados sin sufijo especificativo, además de la relación de errores falsos que impiden la compilación correcta del programa. La siguiente tarea a realizar, es la modificación de los bits identificativos de la palabra de configuración, con los que se consigue desactivar la protección del código, activar el perro guardián, identificar que el oscilador usado es de alta velocidad, activar el temporizador de arranque y la anulación de programar el PIC en el circuito de uso.

Tras la definición de las variables de uso específico en este archivo, se procede con la escritura del código, distinguiendo que el programa principal comienza en la posición 0h y ante el suceso de una interrupción se ejecuta desde la posición 4h.

En la atención a interrupciones se realiza la evaluación de los posibles sucesos que la originan, siguiendo un orden de prioridad. En el siguiente diagrama de flujo se muestra la secuencia que se realiza.



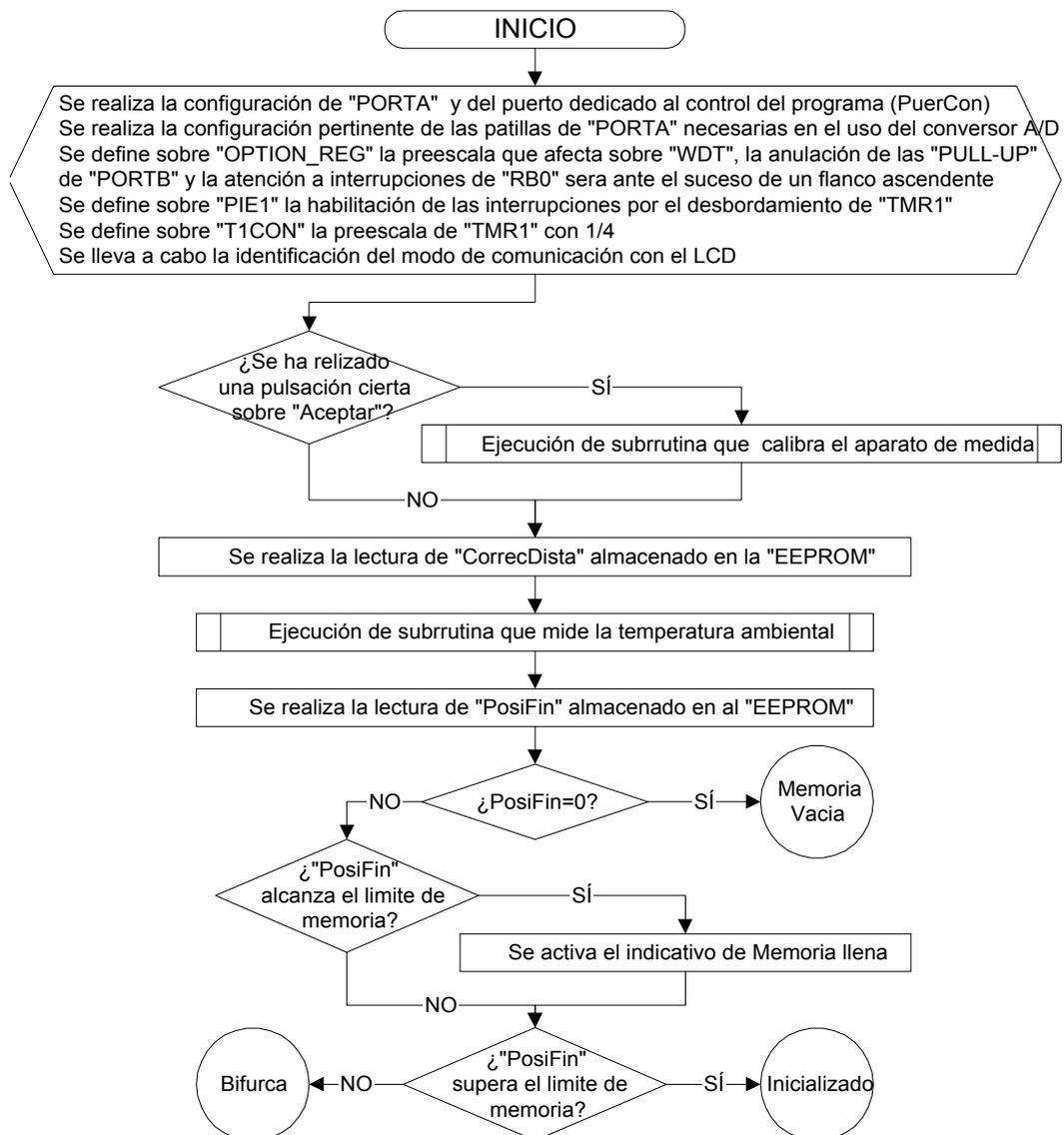
El programa principal, da comienzo con la carga sobre los registros de función específica necesarios, los Bytes requeridos para el uso deseado de los puertos y los periféricos internos del microcontrolador. La función que desempeñan los puertos y los periféricos del microcontrolador, para la realización de este proyecto se muestra en las siguientes tablas.

PATILLAS DEL MICROCONTROLADOR	
IDENTIFICACIÓN	USO QUE SE LE ASIGNA
RA0 y RA1	Son entradas analógicas para realizar su conversión a digital
RA3	Es la entrada que marca la referencia positiva para el CAD
RA2 y RA5	Son salidas digitales que entregan la señal de ultrasonidos
RB0	Es la entrada digital de recepción del ECO
RB1, RB2 y RB3	Son salidas digitales para el control del LCD
RB4 - RB7	Son entradas digitales asociadas a los pulsadores
RC4 - RC7	Son patillas digitales conectadas al bus de datos del LCD

REGISTROS ESPECIFICOS DEL MICROCONTROLADOR	
IDENTIFICACIÓN	USO QUE SE LE ASIGNA
ADCON1	Se justifican a la derecha los 10 bits resultantes del CAD
OPTION_REG	Se carga la preescala máxima del "WDT"
PIE1	Se habilitan la interrupción por desbordamiento de "TMR1"
T1CON	Se carga la preescala deseada en "TMR1"

Tras la configuración general del microcontrolador, se realiza la obtención de una serie de datos, relacionados con el correcto funcionamiento del medidor, y con los que se elige la secuencia de instrucciones a continuar ejecutando.

Tanto las tareas de configuración del microcontrolador, como la actualización de las variables, se realizan en la secuencia mostrada por el siguiente diagrama de flujo.



Este archivo principal dispone de una estructura, formada por una serie de etapas, las cuales tienen en común la modificación de la pantalla del LCD, notificando, al usuario, la etapa que se está ejecutando. Una explicación de las tareas que se realizan dentro de cada etapa se realiza a continuación.

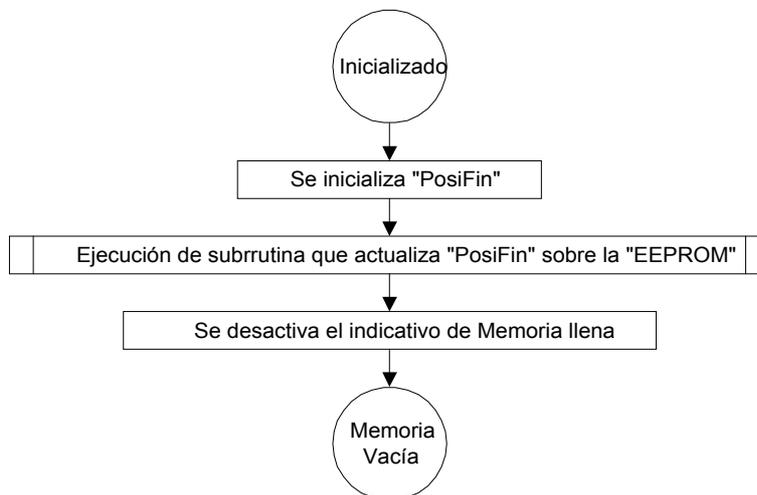
También en este archivo se incluyen una serie de subrutinas empleadas dentro de las mencionadas etapas, por lo que conviene comentar cada una de ellas.

### 5.2.1 Etapa de inicializado

En esta sección de código se entra en la primera ejecución del programa, puesto que la memoria EEPROM dispone de los datos que se cargan por defecto en la grabación del programa. La otra situación en la que se ejecuta esta secuencia de instrucciones, es ante el deseo confirmado de borrar de los datos almacenados.

Hay que indicar que esta etapa es pasajera, es decir, no se permanece en ella y debido a la rapidez con que termina, no se muestra sobre el LCD que está en ejecución.

El diagrama de flujo correspondiente a esta etapa se muestra a continuación.



### 5.2.2 Etapa indicadora de memoria vacía (MemoriaVacía)

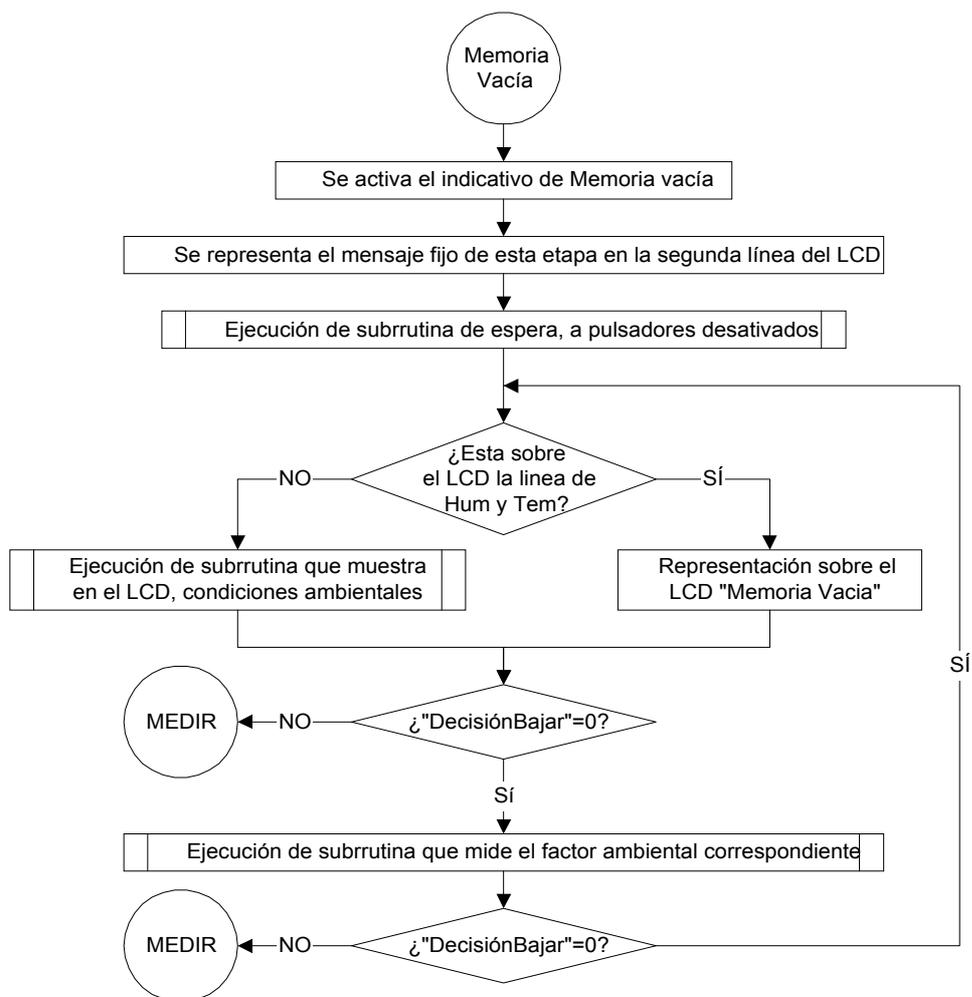
Esta etapa, tiene la misión de ofrecer al usuario el único camino que se puede seguir, representándose sobre la segunda línea del LCD (MEDIR). La manera de seleccionar este camino, es con la activación del pulsador Bajar, por encontrarse bajo este indicativo. Sobre la primera línea del LCD se conmutan, en el tiempo, dos

mensajes informativos, donde uno de ellos, indica que la memoria de distancias se encuentra vacía y el otro, muestra las condiciones ambientales de humedad relativa y temperatura.

La entrada en esta etapa, corresponde a la situación en que la memoria de distancias se encuentre vacía, y por esta razón no se incluyen los caminos para realizar un borrado o una lectura de distancias, cuando no existen datos en esta memoria.

La salida de esta etapa, se lleva a cabo con la activación del pulsador Bajar, entrando en la etapa de medida.

El diagrama de flujo, que determina la secuencia realizada dentro de esta etapa, se muestra a continuación.



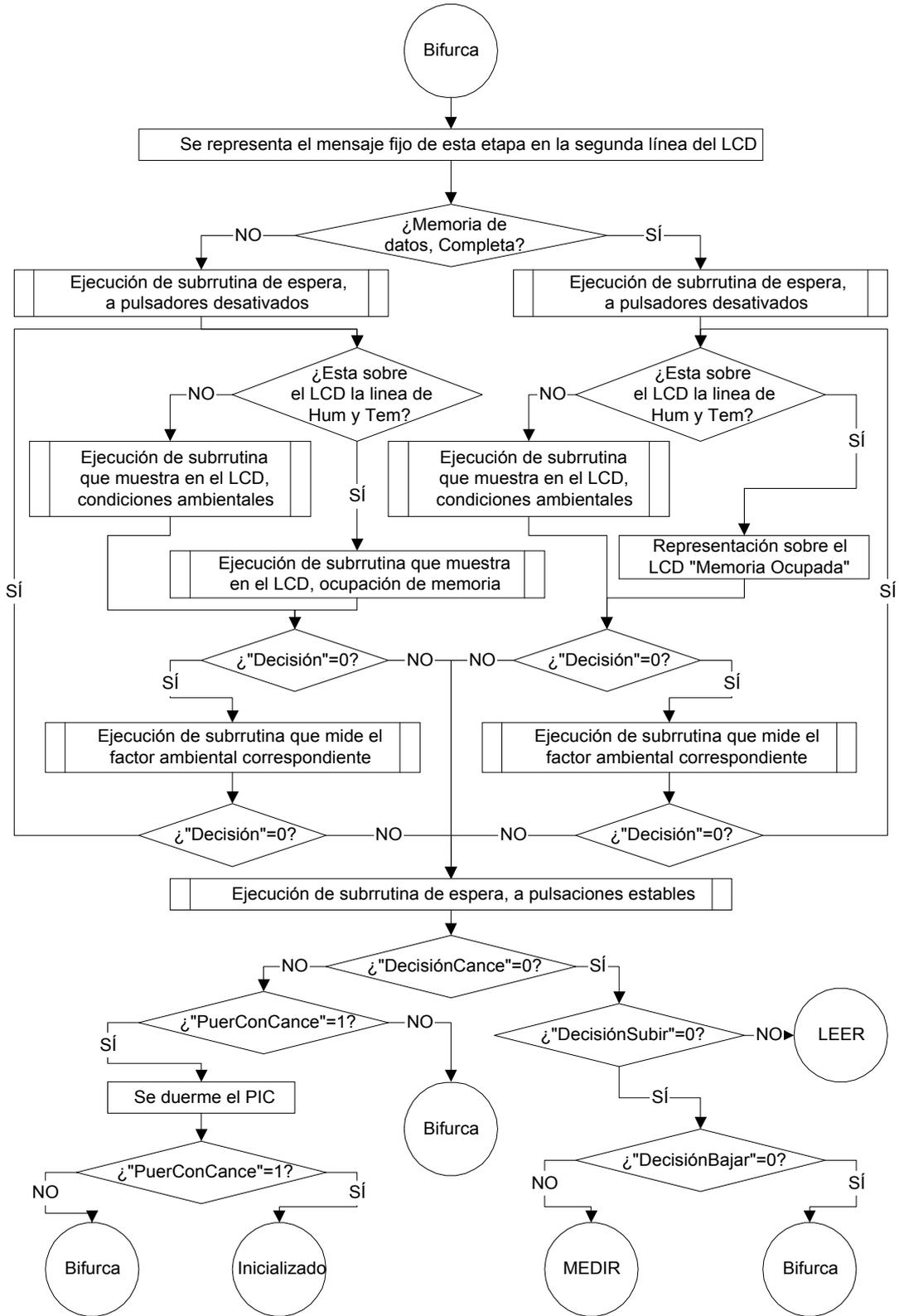
### 5.2.3 Etapa de bifurcación (Bifurca)

Esta etapa, tiene la misión de ofrecer al usuario tres posibles caminos a seguir, representados en la segunda línea del LCD (MEDIR BORRA LEER). La manera de seleccionar uno de ellos, es con la activación del pulsador que se encuentra bajo el indicativo deseado. Sobre la primera línea del LCD se conmutan, en el tiempo, dos mensajes informativos, donde uno de ellos, indica el estado de ocupación de la memoria de distancias, y el otro, muestra las condiciones ambientales de humedad relativa y temperatura.

La entrada en esta etapa, corresponde a la situación en que la memoria de distancias no se encuentre vacía, puesto que no tiene sentido, realizar un borrado o una lectura de distancias, cuando esta memoria se encuentra sin ellas.

La salida de esta etapa, se lleva a cabo con la evaluación del estado en que se encuentran los pulsadores asociados a otra etapa, tras detectar la activación de un pulsador.

El diagrama de flujo, que determina la secuencia realizada dentro de esta etapa, se muestra a continuación.



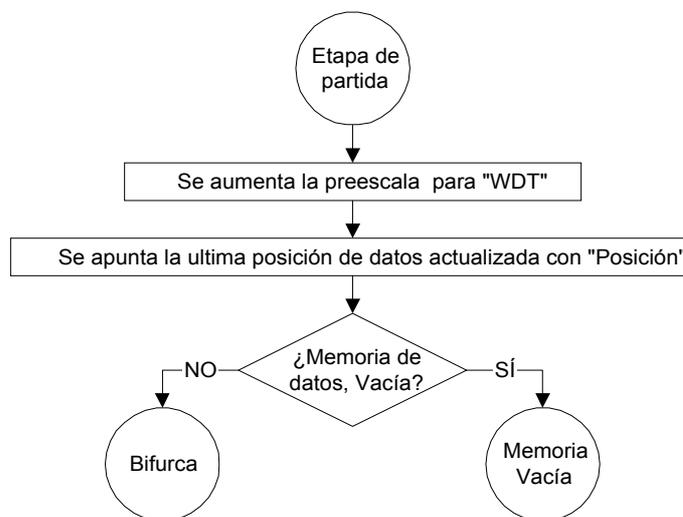
### 5.2.4 Etapa de identificación del origen

Debido a las diferencias que presentan las etapas anteriormente vistas (Bifurca y MemoriaVacía), referente al estado en que se encuentra la “EEPROM” y

la coincidencia, del posible acceso a “MEDIR”, es necesaria la identificación de la etapa desde la que se ha entrado, en el mencionado camino común. La condición que exigen esta identificación, es el deseo de retroceder de la mencionada etapa coincidente, tras tomar el usuario esta decisión.

Hay que indicar que esta etapa es pasajera, es decir, no se permanece en ella y debido a su transparencia para el usuario, no se indica sobre el LCD que esta en ejecución.

El diagrama de flujo correspondiente a esta etapa se muestra a continuación.



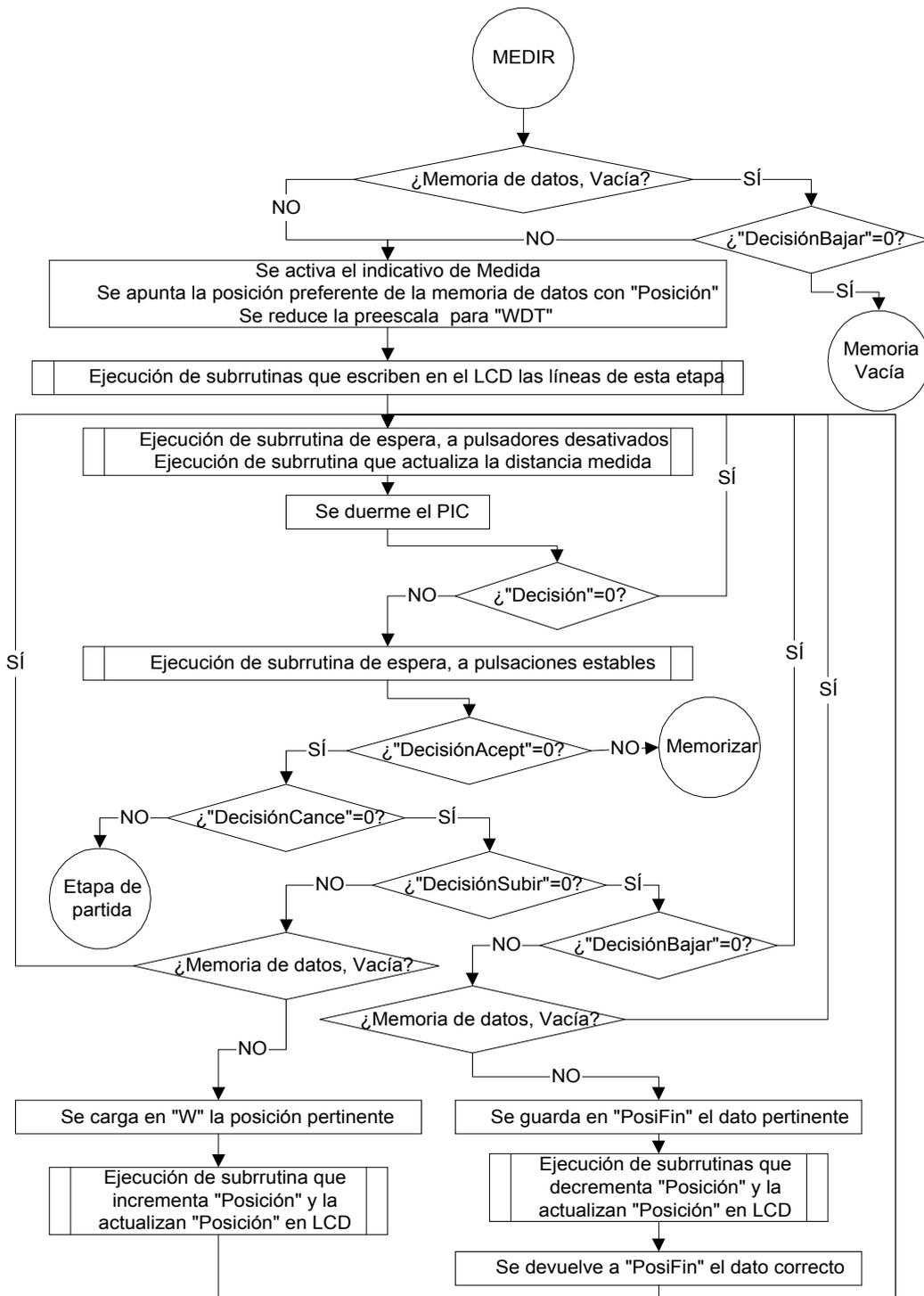
### 5.2.5 Etapa de medida (MEDIR)

En esta etapa, se lleva a cabo la misión principal de este proyecto, que consiste en visualizar sobre el LCD, la medida de la distancia existente entre el aparato de medida, y el objeto donde rebota la señal emitida. La obtención de esta medida, se consigue al realizar una serie de cálculos con datos disponibles por el microcontrolador.

Sobre la pantalla del LCD se muestra información referente a la memoria de distancias, para que el usuario conozca la posición de memoria a la que se esta apuntando, el estado de dicha posición (Vacía → □ o Llena → ■) y la relación entre posiciones ocupadas y totales. También representa la opción que se realiza con la pulsación de “Aceptar” (Salvar) y como ya se ha mencionado antes, el resultado del calculo realizado indicativo de la distancia medida, presentando la opción de guardarla en la posición deseada.

En esta etapa tienen utilidad los cuatro pulsadores, los cuales son evaluados con la activación de uno de ellos. Por medio de los pulsadores Subir y Bajar se consigue apuntar a la posición de memoria de distancias deseada, sin salir de esta etapa.

El diagrama de flujo, que determina la secuencia realizada dentro de esta etapa, se muestra a continuación.



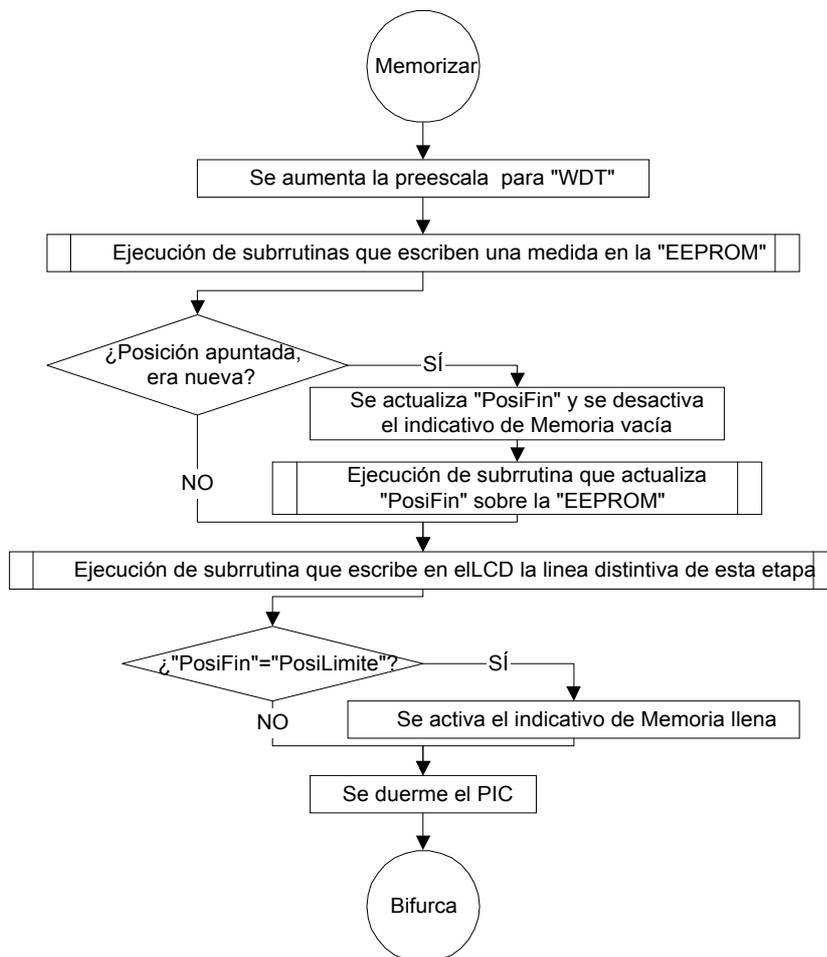
### 5.2.6 Etapa de almacenado en memoria (Memorizar)

En esta etapa, se realiza todas las operaciones necesarias, para el almacenado de una medida de distancia en los Bytes de la memoria EEPROM, ofreciendo la permanencia de estos datos almacenados, ante la ausencia de alimentación. También se tienen que actualizar los punteros de esta memoria, por medio de los cuales se consigue el grado de ocupación de la misma.

Sobre la pantalla del LCD se muestra información referente a la posición de memoria de distancias, en la que se acaba de almacenar la medida, para que el usuario distinga entre una posición libre (□), o posición ocupada(■).

La salida de esta etapa se realiza, tras la visualización sobre el LCD de la información antes mencionada durante un tiempo determinado.

El diagrama de flujo, que determina la secuencia realizada dentro de esta etapa, se muestra a continuación.

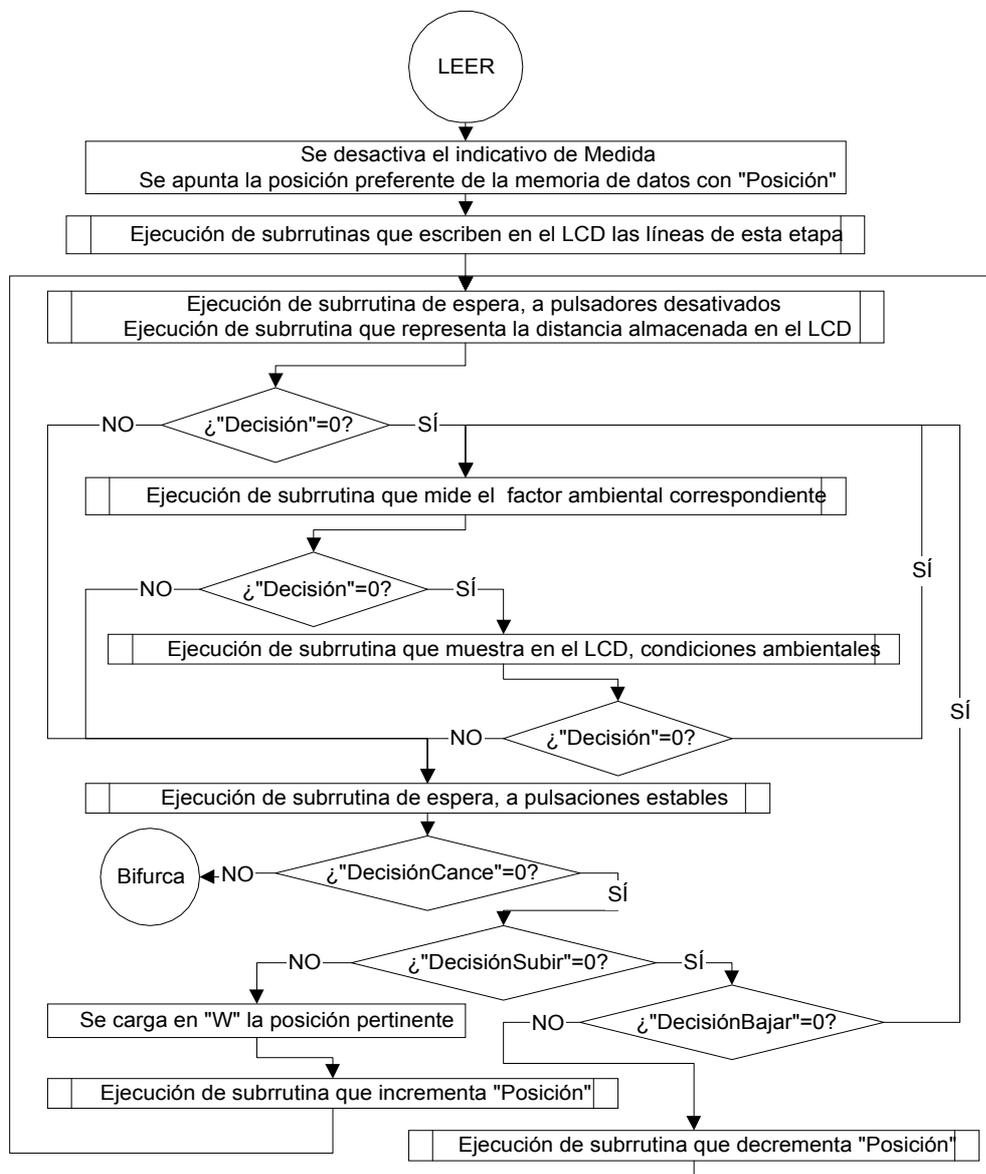


### 5.2.7 Etapa de lectura (LEER)

En esta etapa se cumplen dos misiones, siendo prioritaria, la visualización de la medida, apuntada por “Posición”, sobre el LCD. La misión secundaria consiste en representar en el LCD, los parámetros ambientales de humedad relativa y temperatura.

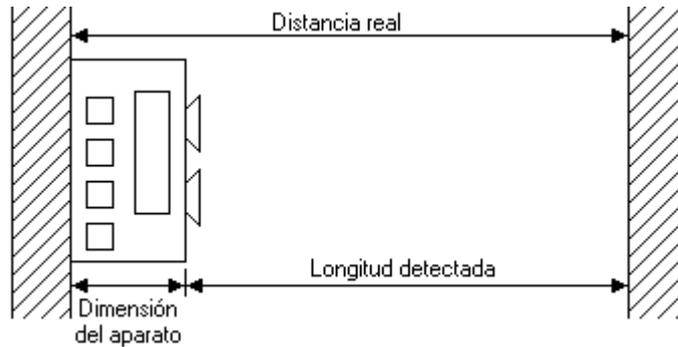
El pulsador Aceptar, carece de utilidad, puesto que es una etapa puramente informativa, y no se ofrece ninguna opción de modificar los parámetros almacenados en el microcontrolador por deseo del usuario. Por similitud con “MEDIR” el uso de “Subir” y “Bajar” permite elegir al usuario, la medida memorizada deseada.

El diagrama de flujo, correspondiente a esta etapa, se muestra a continuación.



### 5.2.8 Etapa de calibración del medidor (Calibrado)

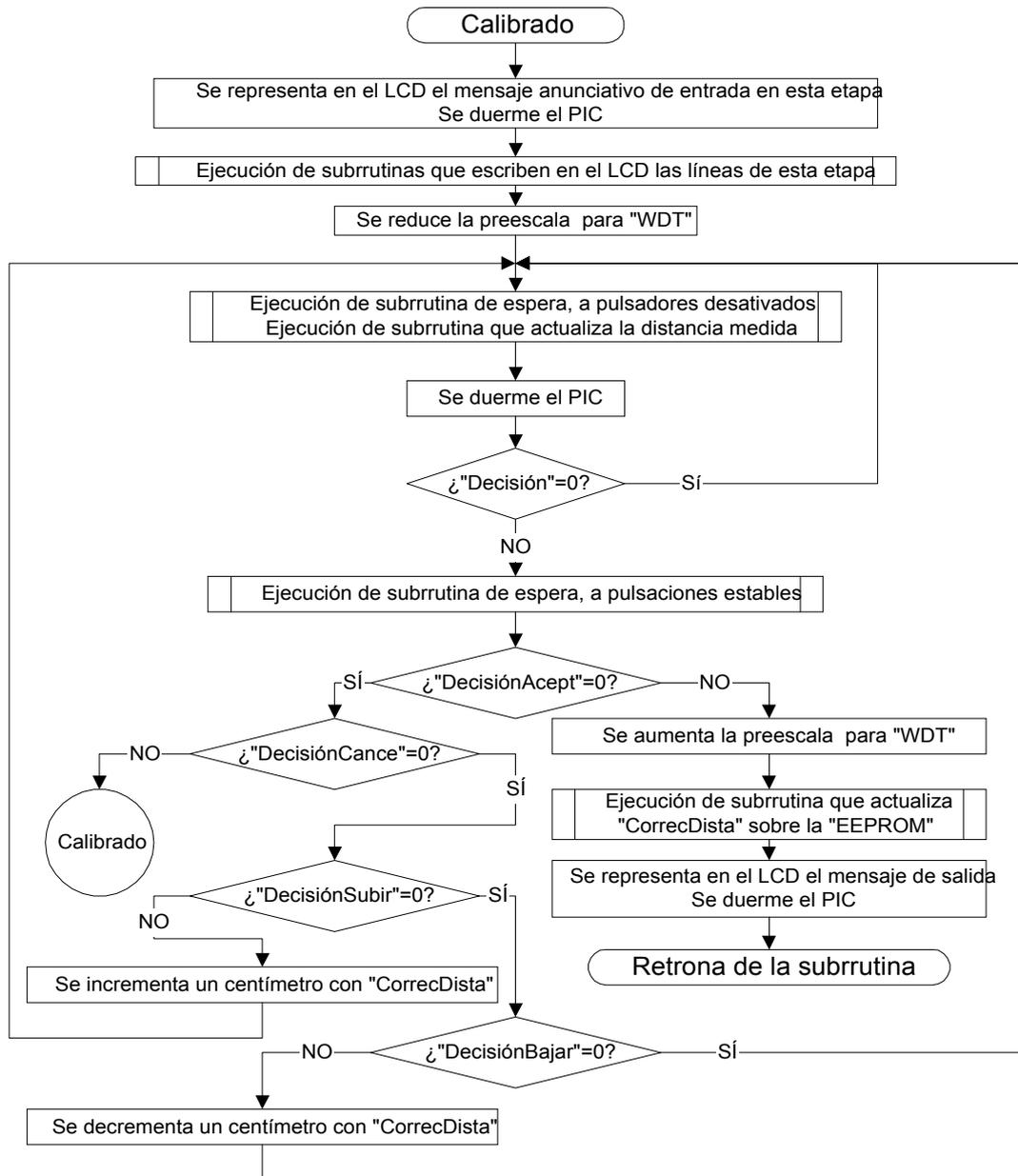
En esta etapa se trata de estimar la dimensión del aparato de medida que afecta en la distancia visualizada, puesto que la medida realizada tiene como punto inicial, a los transductores de ultrasonidos, en lugar de la base del aparato. Para explicarlo de forma gráfica se hace uso de la siguiente figura.



Para llevar a cabo esta estimación, es necesaria la realización de una medida conocida, y así aproximar la distancia visualizada en el LCD, con la actuación sobre los pulsadores Subir y Bajar, puesto que con cada pulsación de los mismos se modifica '1' centímetro la distancia visualizada. Con la aceptación de la medida visualizada, se asegura la consideración de esta dimensión en el resto de medidas realizadas, puesto que se mantiene almacenada en la memoria EEPROM.

Esta etapa ofrece la ventaja, a este programa, de que al aparato medidor no se le exige unas dimensiones fijadas.

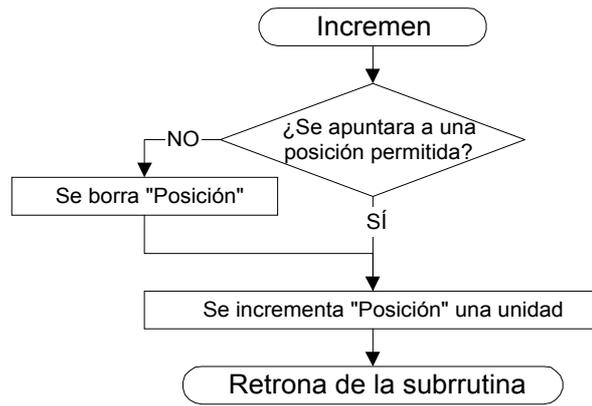
El diagrama de flujo, que determina la secuencia realizada dentro de esta etapa, se muestra a continuación.



### 5.2.9 Subrutina que direcciona la posición inmediata superior (Increment)

Esta subrutina se encarga de apuntar con "Posición", la siguiente zona de memoria EEPROM, permitida para su uso. El incremento ante el direccionamiento de la posición de memoria máxima permitida, hace que se apunte la posición más baja de la memoria, puesto que se está operando con una memoria circular.

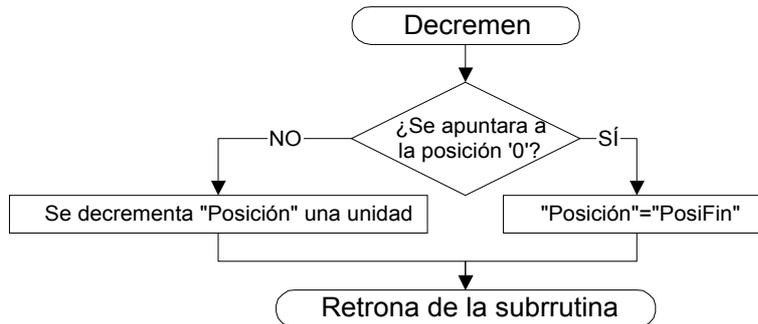
El diagrama de flujo empleado en esta subrutina se muestra a continuación.



### 5.2.10 Subrutina que direcciona la posición inmediata inferior (Decremen)

Esta subrutina se encarga de apuntar con “Posición”, la anterior zona de memoria EEPROM, permitida para su uso. El decremento ante el direccionamiento de la posición más baja de la memoria, hace que se apunte la posición máxima permitida, puesto que se esta operando con una memoria circular.

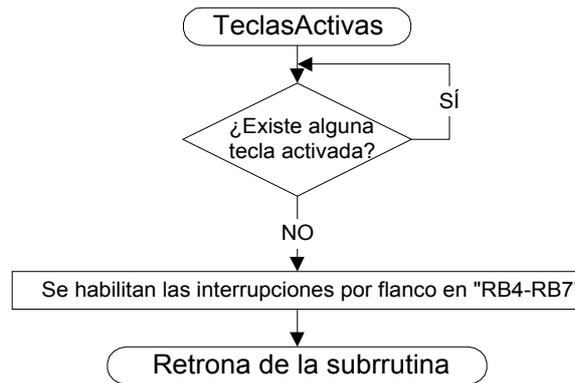
El diagrama de flujo empleado en esta subrutina se muestra a continuación.



### 5.2.11 Subrutina de espera a la desactivación de los pulsadores (TeclasActivas)

Esta subrutina se encarga de permanecer dentro de un bucle mientras alguno de los pulsadores se encuentre activado. De esta forma se consigue la detección de las pulsaciones, con flancos ascendentes y no con el estado en nivel alto.

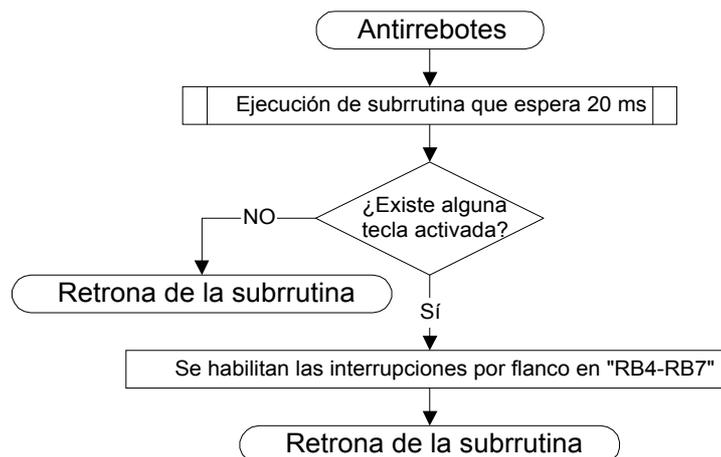
El diagrama de flujo empleado en esta subrutina se muestra a continuación.



### 5.2.12 Subrutina de espera a las pulsaciones sin rebotes (Antirrebotes)

Esta subrutina se encarga de confirmar el suceso de una pulsación cierta tras la finalización de los rebotes que aparecen al comienzo de la pulsación.

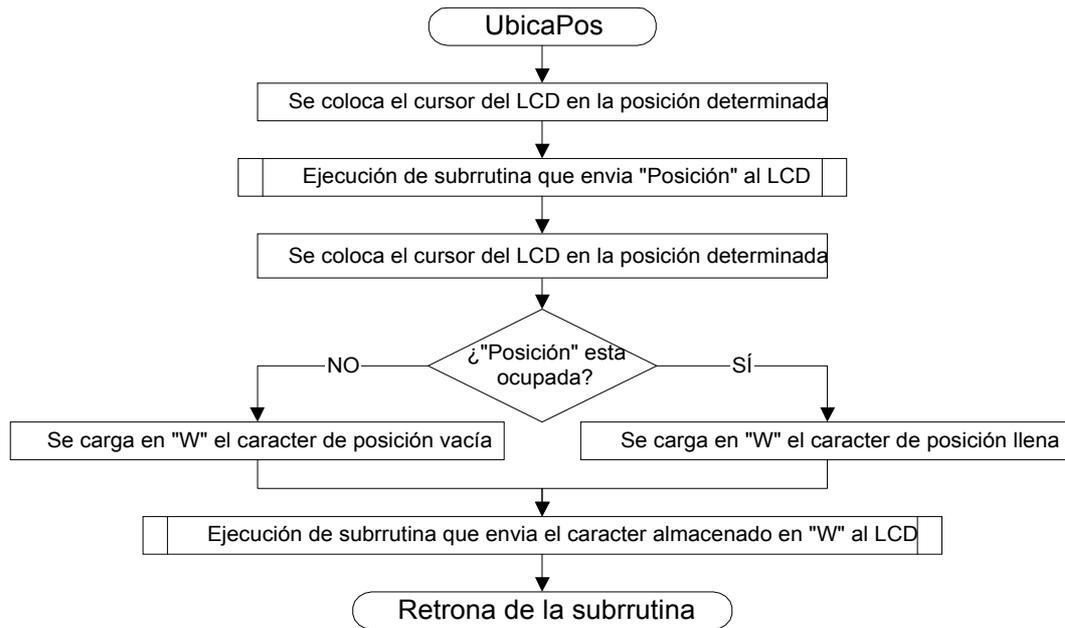
El diagrama de flujo empleado en esta subrutina se muestra a continuación.



### 5.2.13 Subrutina de representación en el LCD "Posición" (UbicaPos)

Esta subrutina es empleada únicamente en las tareas de direccionamiento de la posición de memoria de distancias, dentro de la etapa MEDIR. El motivo por el que no se emplea en las mismas tareas de "LEER", es por que añade la peculiaridad de mostrar el estado de la memoria apuntada (Vacía → □ o Llena → ■) en su posición determinada de la pantalla, siendo incompatible e innecesario en dentro de la etapa de lectura.

El diagrama de flujo empleado en esta subrutina se muestra a continuación.



### 5.3 ARCHIVO VERMEN

En este archivo se encuentra las subrutinas que logran establecen la comunicación con el LCD, puesto que con ellas se sigue el protocolo de comunicación de dicho dispositivo.

A causa de una gran repetición de la escritura de mensajes de texto en la pantalla del LCD, se han colocado en este mismo archivo las subrutinas que realizan esta operación. Para seleccionar el mensaje deseado, hay que introducir su código en la variable que evalúa esta subrutina, algo parecido sucede con la escritura de valores numéricos, donde por medio de una variables se define el dato a visualizar y el formato deseada.

Por motivo de la necesidad de realizar esperas de tiempo, en el cumplimiento del mencionado protocolo, se disponen en este archivo, todas las subrutinas de espera empleadas en programa completo.

### 5.4 ARCHIVO CALCULO

En este archivo se encuentran las subrutinas con las que se realiza los productos entre dos factores y la conversión de datos, de base binaria a base decimal, dando el resultado con los dígitos en código ASCII.

El límite que presentan ambas operaciones, viene dado por los valores de entrada en Bits. Para la multiplicación los factores pueden ser de 24 y 16 bits dando un resultado que no supera los 40 bits, y para la conversión de binario a ASCII el dato binario puede ser de 40 bits, no superando la representación en decimal los 13 dígitos.

## 5.5 ARCHIVO LINELCD

En este archivo se encuentran las subrutinas con las que se representa sobre las líneas de LCD, todos los caracteres (letras números y símbolos) precisos, para la completa información que se desea dar al usuario.

Por la distinta periodicidad con que se desea actualizar los datos, resulta de gran utilidad la anidación entre subrutinas.

## 5.6 ARCHIVO ACCESMEN

En este archivo se encuentran las subrutinas relacionadas con la escritura y lectura de la memoria EEPROM y las subrutinas que realizan la lectura de datos que se encuentran en tablas de la memoria de programa.

La ventaja que presenta la memoria EEPROM, es el mantenimiento de los datos almacenados ante la ausencia de alimentación. Por esta razón interesa guardar los datos necesitados en el posterior encendido del medidor.

El motivo por el que se realiza la lectura de la memoria de programa, es la necesidad de extraer datos de tablas que tienen un tamaño de 1024 posiciones.

## 5.7 ARCHIVO PARAMETR

En este archivo se encuentran las subrutinas con las que se consiguen la lectura de los sensores de temperatura y de humedad relativa y las subrutinas que realizan los cálculos pertinentes, para obtener los parámetros útiles en la información entregada al usuario.

En la lectura de los sensores, se hace uso del conversor A/D, y dependiendo de la etapa que solicite esta lectura, se presenta realizara la conversión con el PIC dormido o ejecutando instrucciones.

Para conseguir el resultado de los parámetros, se requiere el uso del archivo calculo, puesto que en todas las subrutinas realizan la operación de multiplicar, y además, en el caso de visualizar su resultado en pantalla, se guarda dicho parámetro en las variables de entrada de la subrutina de conversión a dígitos en código ASCII.

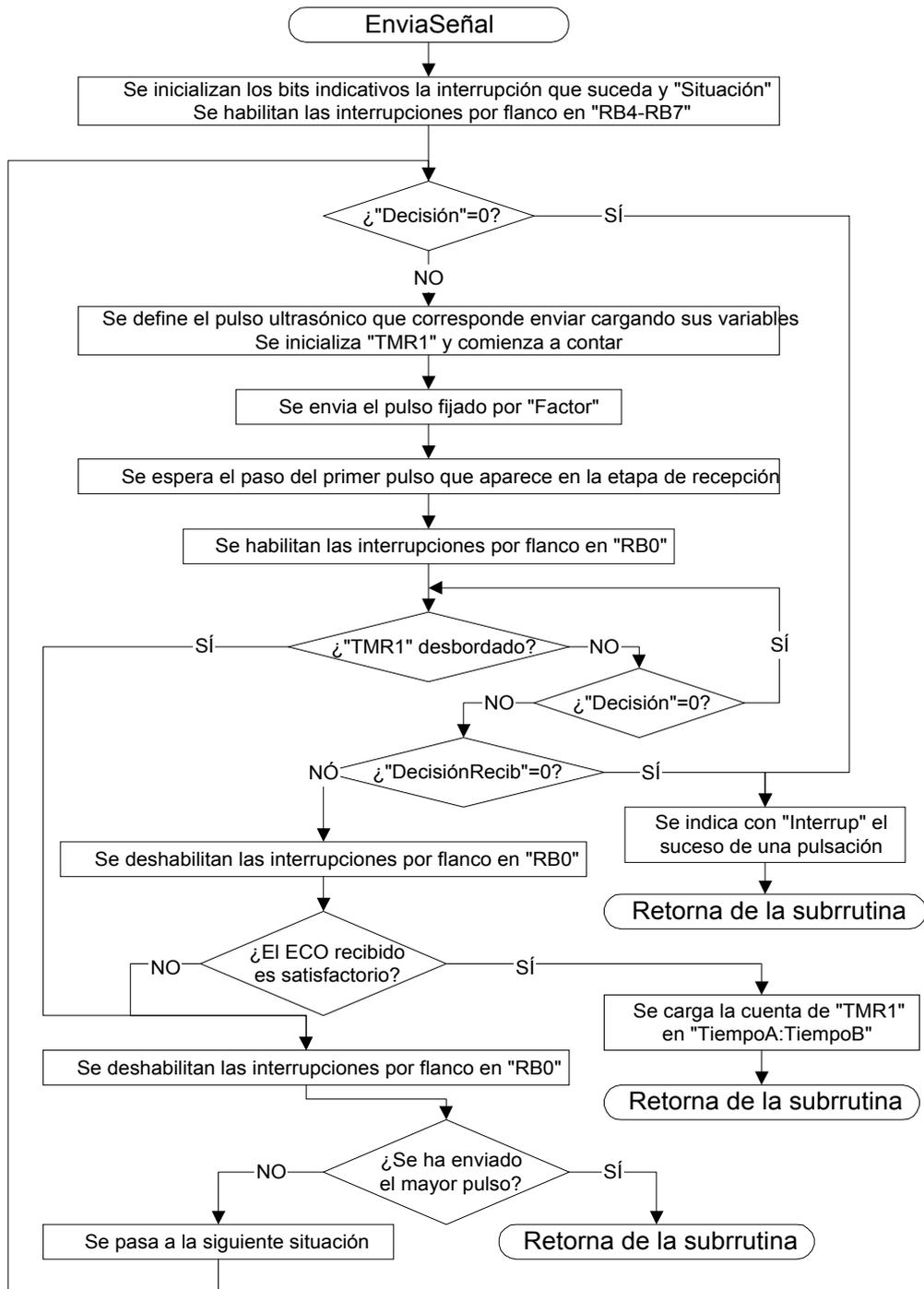
## 5.8 ARCHIVO SENULT

En este archivo se encuentra la subrutina relacionada íntegramente con la señal ultrasónica que recorre la distancia a medir, desde su generación, hasta la recepción del ECO.

En lo que respecta a la generación de la señal de ultrasonidos, para conseguir un mayor alcance de la misma, esta subrutina emplea un algoritmo con el que a partir de un pulso elemental, se transmiten pulsos de mayor duración, cada vez que el ECO recibido no cumple las condiciones definidas. Este aumento de duración del pulso eléctrico aplicado al transmisor, corresponde a un incremento de potencia de la onda ultrasónica emitida, consiguiendo el mayor alcance buscado.

El pulso elemental tiene una duración de 0.5 ms, y es con el que se comienza todas las medidas. Ante la recepción del ECO, se le realiza una evaluación del pulso entregado por la etapa receptora, con el criterio de juicio dependiente de factor multiplicador del pulso elemental. Con el resultado satisfactorio de esta evaluación, se pone fin a la subrutina tras guardar el tiempo medido por "TMR1", en caso contrario se envía un pulso de mayor duración. Cuando el máximo pulso no es capaz de recorrer la distancia que se pretende medir, se desbordara el temporizador "TMR1" y se retornara de la subrutina, indicando la imposibilidad de medir esa distancia.

El organigrama que se ha seguido en la realización de esta subrutina se muestra a continuación.



## 5.9 ARCHIVO TABLAS

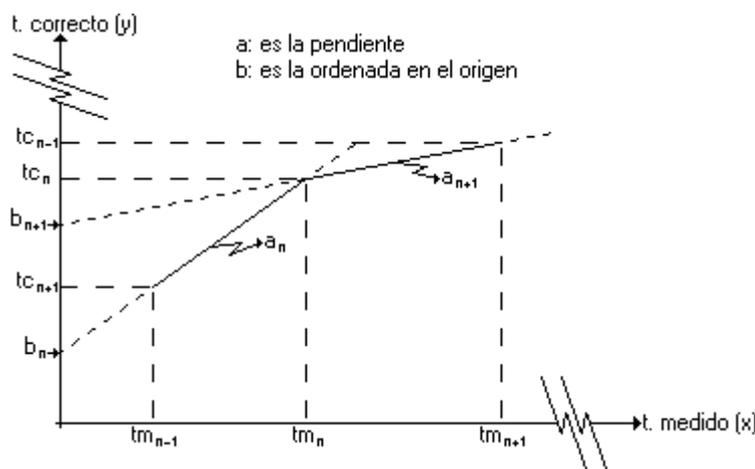
En este archivo se encuentra las tablas necesarias en la ejecución del programa, puesto que se tienen las que almacenan los mensajes a representar en el LCD y las que contienen datos numéricos, relacionados con la ejecución de cálculos.

Relacionándose con la temperatura medida se dispone de dos tablas de gran tamaño, puesto que la entrada esta definida, por la combinación binaria de los 10 bits

que entrega el conversor A/D al realizarse la medida de la temperatura, esto supone que las tablas son de 1024 posiciones. En ellas se encuentra la velocidad del sonido y el factor definidor de la humedad, ocupando dos tablas cada uno de estos parámetros, puesto que tienen un tamaño de 16 bits y cada Byte se encuentra en una tabla. El modo de operación con cada uno de estos factores, es almacenándolos en un par de variables, distinguiéndose la parte alta de la parte baja.

El resto de tablas son de un tamaño más reducido y contienen los mensajes que se visualizan en el LCD, y los datos necesarios para realizar una corrección del tiempo medido. Esta corrección del tiempo es necesaria para reducir el error cometido en la medida representada.

La existencia de este error de medida, tiene su origen por el retardo cometido desde que llega el ECO sobre el receptor, hasta que es entregado el pulso de llegada por la etapa de recepción. El método empleado para corregir este tipo de error se denomina “rectas de calibrado”, y consiste en la realización de una serie de medidas de distancias conocidas, de las que se extrae el tiempo contado por el temporizador, y se calcula el tiempo correcto. Con la relación entre ambos tipos de valores se asignan a cada tramo una recta de la que se extrae la pendiente y la ordenada en el origen. La forma que toman estas rectas de calibrado se muestra en la siguiente figura.



La obtención del tiempo correcto, se realiza aplicando la expresión que define una recta, y con los parámetros conocidos toma la siguiente forma.

$$y = a \times x + b \rightarrow \begin{cases} y \equiv tc \\ x \equiv tm \end{cases} \Rightarrow \boxed{tc = a \times tm + b}$$

Una vez conocida la operación que se debe realizar, se necesita los valores de los términos empleados en dicha operación, y para ello hay que conocer el tramo en el que se encuentra el tiempo medido. De esta forma se explica la necesidad de los tres términos encontrados en las **tabas** de datos de este archivo. Con los límites superiores de los tramos se consigue la identificación del tramo con el que hay que operar y así están definidos sobre las siguientes tablas, la pendiente y la constante de la recta.

El motivo de doblarse y triplicarse, estas tablas es la necesidad de usar dos y tres Bytes respectivamente, los términos que se encuentran en ellas.



## 6 DISEÑO DE CIRCUITOS

### ÍNDICE:

<b>6.1</b>	<b>INTRODUCCIÓN .....</b>	<b>75</b>
<b>6.2</b>	<b>DIAGRAMA DE BLOQUES.....</b>	<b>75</b>
<b>6.3</b>	<b>CIRCUITOS ELÉCTRICOS .....</b>	<b>75</b>
6.3.1	Etapa de transmisión .....	75
6.3.2	Circuito de control .....	76
6.3.3	Etapa de recepción .....	76
<b>6.4</b>	<b>PLACAS DE CIRCUITO IMPRESO .....</b>	<b>77</b>
6.4.1	Cara de componentes .....	77
6.4.2	Cara de pistas .....	77
6.4.3	Placa para los pulsadores.....	78
<b>6.5</b>	<b>FOTOGRAFÍAS DE LA PLACA CONSEGUIDA.....</b>	<b>78</b>
6.5.1	Cara de pistas .....	78
6.5.2	Cara de componentes .....	78
6.5.3	Pantalla LCD y pulsadores.....	79
6.5.4	Sensores de ultrasonidos .....	80
6.5.5	Montaje completo.....	80
<b>6.6</b>	<b><u>COSIDERACIONES</u> EN LA REALIZACIÓN DE LA PLACA .....</b>	<b>80</b>



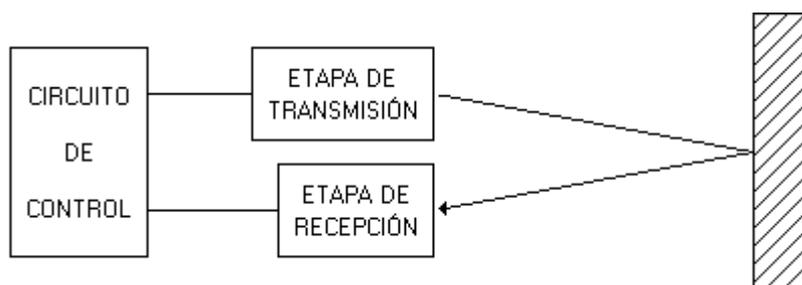
## 6.1 INTRODUCCIÓN

En este aparato medidor se han distinguido tres circuitos, encargándose cada uno de realizar tareas muy concretas. Es fundamental que exista una relación entre estas tareas y se consigue por medio de la correspondiente unión eléctrica.

En este capítulo se muestra el diseño hardware, que cumple el objetivo fijado por este proyecto.

## 6.2 DIAGRAMA DE BLOQUES

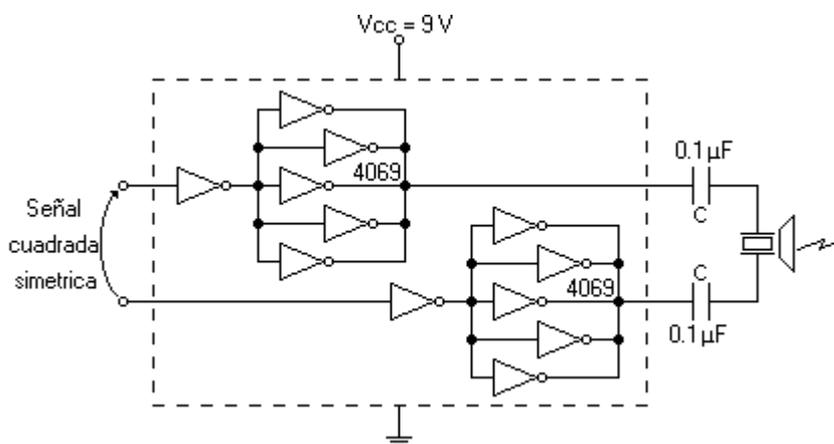
La mejor forma de comprender la necesidad de todas las tareas realizadas por cada bloque explicado, es con la unión de dichos bloques. En la siguiente figura se muestra el diagrama de bloques del aparato medidor completo.



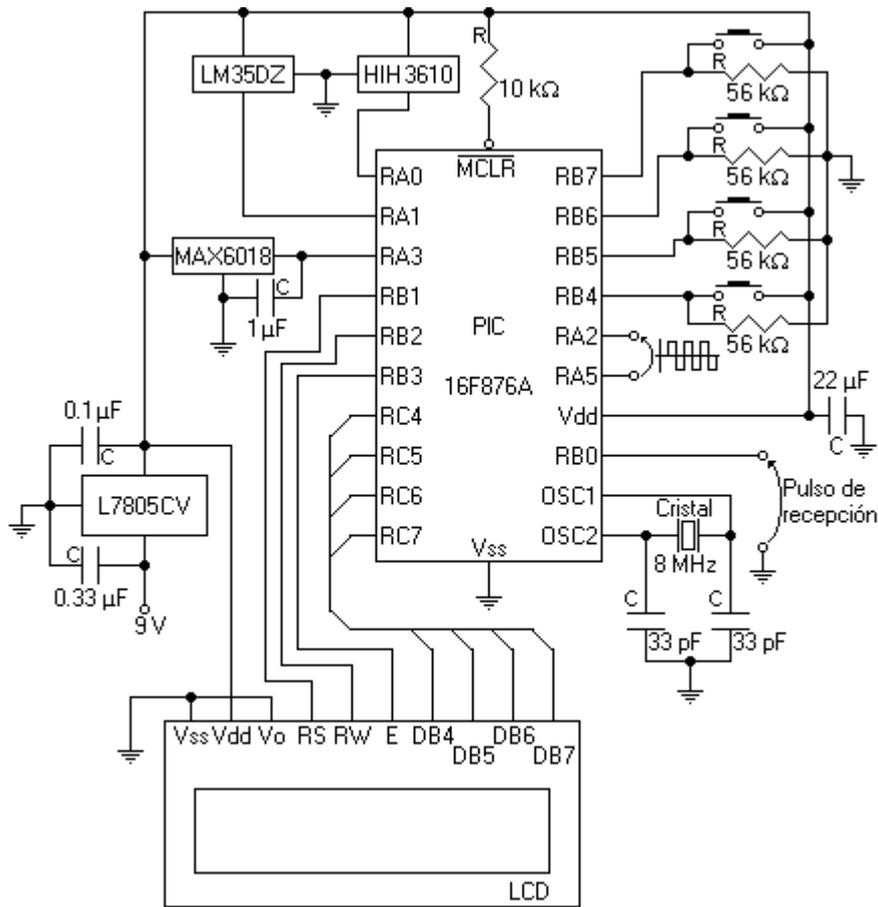
## 6.3 CIRCUITOS ELÉCTRICOS

A continuación se muestra el circuito eléctrico de cada bloque, y son ellos los que dan a conocer los puntos por los que se unen, para que exista la compaginación entre las tareas deseada.

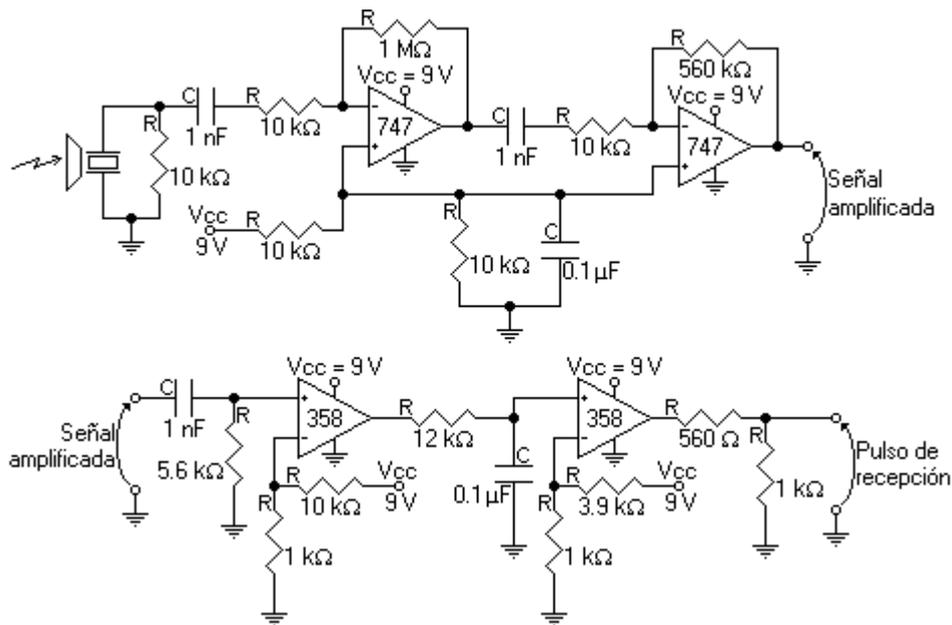
### 6.3.1 Etapa de transmisión



6.3.2 Circuito de control

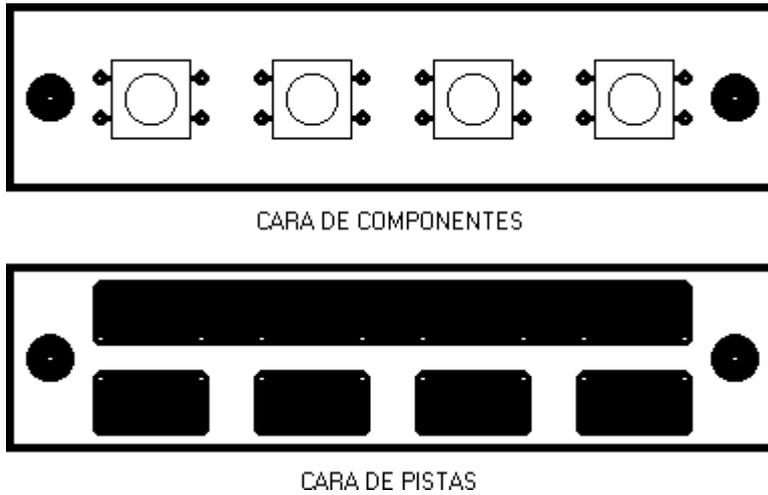


6.3.3 Etapa de recepción





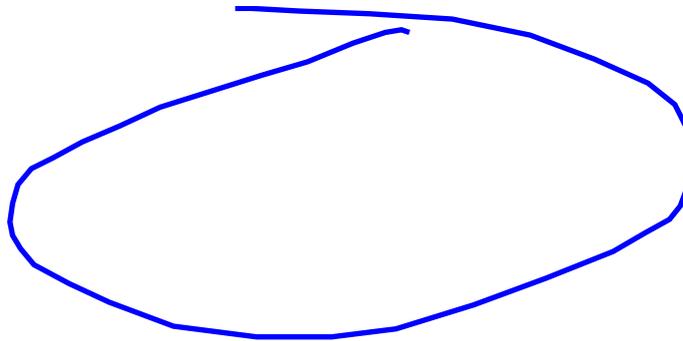
### 6.4.3 Placa para los pulsadores



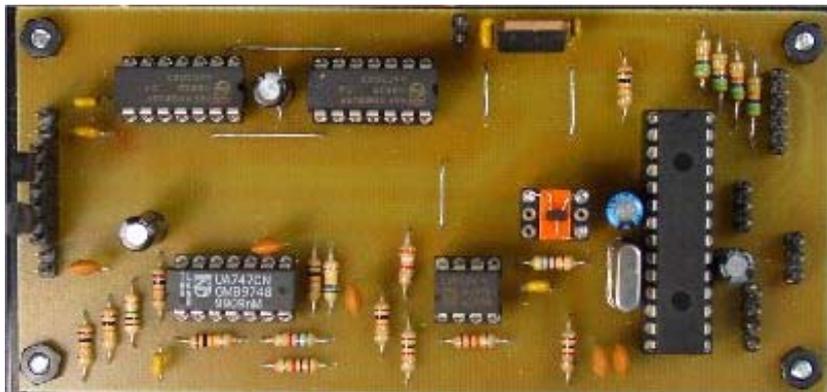
## 6.5 FOTOGRAFÍAS DE LA PLACA CONSEGUIDA

Estas fotografías muestran distintas vistas y zonas concretas del circuito electrónico definitivo.

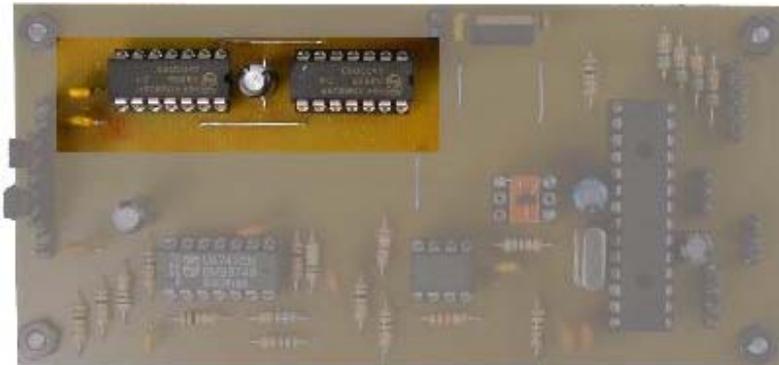
### 6.5.1 Cara de pistas



### 6.5.2 Cara de componentes

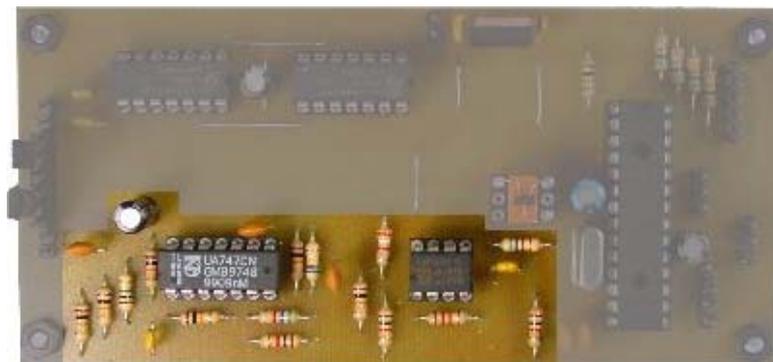
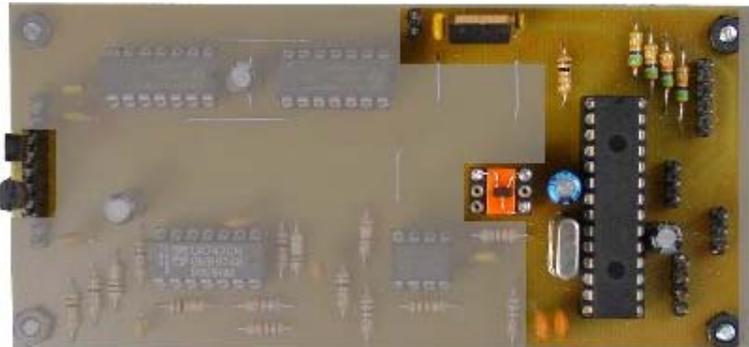


Sobre esta imagen, a continuación se distinguen los bloques que la componen.



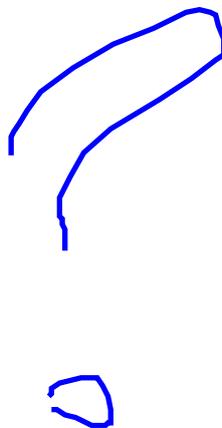
Etapa de transmisión

Circuito de control



Etapa de recepción

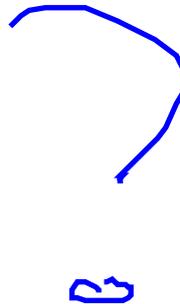
### 6.5.3 Pantalla LCD y pulsadores



#### 6.5.4 Sensores de ultrasonidos



#### 6.5.5 Montaje completo



### 6.6 COSIDERACIONES EN LA REALIZACIÓN DE LA PLACA

En cuanto a la ubicación de los componentes y el trazado de las pistas, se ha hecho uso de la herramienta informática Proteus 6. Una vez conseguido el diseño se ha pasado a archivos reconocidos por la maquina fresadora, por medio del programa IsoCam. Esta maquina fresadora se encarga tanto de la definición de las pistas de cobre, como de realizar los talados, por los que se introducen las patillas de los componentes.

Como se ha podido apreciar en las fotografías, la cara de pistas obtenida se diferencia del circuito diseñado, en la existencia de cobre en zonas donde no se necesita. Este cobre se puede eliminar, de dos formas. El modo empleado por la fresadora consiste en señalar esta superficie de cobre indeseada, con el programa

IsoCam. Esto resulta una tarea muy engorrosa, por el amplio tiempo que tarda en realizar el fresado completo. La otra forma consiste en el proceso de ataque por ácido, donde la capa de cobre de la placa de partida, se ve protegida por las zonas que forman las pistas y de esta forma el ácido elimina el cobre que no esta protegido.

Se puede dar la situación, de que estas zonas muertas realicen la función de antenas, en cuyo caso se captaran interferencias que pueden resultar perjudiciales para el funcionamiento del circuito. Evitar que estas zonas muertas formen una antena se consigue uniéndolas a un nivel de tensión fijo, y puede ser los valores de tierra o alimentación positiva.



## **7 GUÍA DE USUARIO**

### **ÍNDICE:**

<b>7.1</b>	<b>INTRODUCCIÓN .....</b>	<b>85</b>
<b>7.2</b>	<b>TABLAS DE CARACTERISTICAS .....</b>	<b>85</b>
<b>7.3</b>	<b>REALIZACIÓN DE MEDIDAS .....</b>	<b>86</b>
<b>7.4</b>	<b>LECTURA DE MEDIDAS .....</b>	<b>86</b>
<b>7.5</b>	<b>BORRADO DE LA MEMORIA .....</b>	<b>86</b>
<b>7.6</b>	<b>INFORMACIÓN ADICIONAL SOBRE EL LCD.....</b>	<b>86</b>



## 7.1 INTRODUCCIÓN

Este medidor ha sido diseñado para realizar un pequeño número de funciones, pero es misión del usuario, el elegir una de ellas. Todas estas funciones se encuentran detalladas en este capítulo.

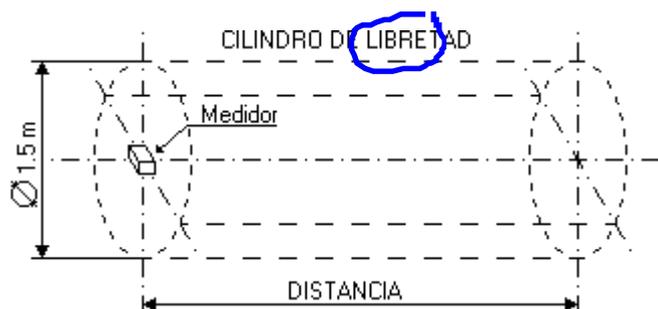
La comunicación con el usuario es continua, desde que se enciende hasta que se apaga, con el interruptor de la cara lateral, por medio de un display LCD y cuatro pulsadores. También es conveniente definir el significado dado a los símbolos que se emplean, con la siguiente tabla.

PULSADORES		CARACTERES EN PANTALLA	
←	Bajar	/	Relación entre valores (parcial/total)
X	Cancelar	→	Cambio de estado (Inicial→Final)
→	Subir	□	Memoria en estado libre
V	Aceptar	■	Memoria en estado ocupada

## 7.2 TABLAS DE CARACTERÍSTICAS

CARACTERÍSTICAS ELÉCTRICAS			
ALIMENTACIÓN DEL APARATO		Batería de 9 V	
CONSUMO DE POTENCIA EN LAS DISTINTAS FUNCIONES			
En espera de decisión		116.1 mW	
Realizando medidas		119.7 mW	
Realizando lecturas		114.3 mW	
PRESTACIONES OFRECIDAS			
MAGNITUD A MEDIR	MÍNIMO VALOR	MÁXIMO VALOR	ERROR
Distancia (m)	0,65	9	±0,02
Temperatura (°C)	0	125	±0,2
Humedad relativa (%)	2	100	±1
La temperatura de operación, se encuentra dentro del rango (0-85) °C			
Dimensiones (mm)		150 x 145 x 50	

No se garantizan las medidas correctas, ante la inexistencia de un camino libre sobre la distancia a medir, de sección circular de 1.5 m de diámetro, como se muestra en la siguiente figura.



### 7.3 REALIZACIÓN DE MEDIDAS

El modo de comenzar a realizar medidas, es con la pulsación de la tecla Bajar (**⬅**), cuando en el LCD se muestra una de las dos posibles pantallas iniciales. Tras esta pulsación, comenzara a realizar una medida continua, visualizándose en la pantalla correspondiente a esta función.

Sobre esta pantalla, con las teclas Bajar y Subir (**⬅ ➡**), se podrá seleccionar la posición de memoria donde se desee almacenar la medida realizada. Es con la tecla Aceptar (**V**) con la que se confirma el guardado de esta medida. Y con la tecla Cancelar (**X**) se sale de esta función.

### 7.4 LECTURA DE MEDIDAS

En la función de lectura, se entra con la pulsación de la tecla Subir (**➡**), cuando en el LCD se muestra la pantalla inicial que contiene LEER. Tras esta pulsación, se entra en la pantalla que muestra la ultima medida almacenada, y la posición que esta ocupa.

Permaneciendo en esta pantalla, con las teclas Bajar y Subir (**⬅ ➡**), se podrá seleccionar la posición de memoria que se desee leer y con la tecla Cancelar (**X**) se sale de esta función.

### 7.5 BORRADO DE LA MEMORIA

Ante el deseo de borrar las medidas que se encuentran en la memoria, hay que mantener pulsada ( $\approx 2.5$  seg) la tecla Cancelar (**X**). Tras el borrado se retrocede a la pantalla inicial, donde únicamente se ofrece la realización de medidas.

### 7.6 INFORMACIÓN ADICIONAL SOBRE EL LCD

También se ofrece información acerca del estado de ocupación de la memoria, mostrándose la relación entre posiciones **Ocupadas/Totales** y del estado ambiental, con la humedad relativa y la temperatura.

## 8 PRESUPUESTO

### ÍNDICE:

<b>7.1</b>	<b>EVALUACIÓN DE COSTE DEL PROTOTIPO.....</b>	<b>89</b>
7.1.1	Coste del material.....	89
7.1.2	Coste del diseño .....	90
<b>7.2</b>	<b>EVALUACIÓN DE COSTE DEL PRODUCCIÓN.....</b>	<b>90</b>
7.2.1	Coste del material.....	90
7.2.2	Coste de la mano de obra.....	91
<b>7.3</b>	<b>PRECIO DE SALIDA AL MERCADO.....</b>	<b>91</b>



## 8.1 EVALUACIÓN DE COSTE DEL PROTOTIPO

Los gastos económicos originados en el desarrollo de este prototipo, son los tipos de costes detallados en los siguientes apartados.

### 8.1.1 Coste del material

En este apartado se incluyen, los precios de todos los componentes empleados en este diseño. El calculo de este coste se muestra en la siguiente tabla.

COMPONENTE	PRECIO/unidad (€)	CANTIDAD	PRECIO/total (€)
Microcontrolador 16F876A	6,51	1	6,51
Integrado HEF4069	0,16	2	0,32
Integrado UA747	0,10	1	0,10
Integrado LM358N	0,08	1	0,08
Regulador L7805CV	0,20	1	0,20
Regulador MAX6018	0,80	1	0,80
Zócalo Pin torneado de 14	0,46	7	3,22
Zócalo Pin torneado de 8	0,18	1	0,18
Zócalo Pin torneado de 6	0,08	2	0,16
Cristal resonador 8 MHz	0,28	1	0,28
Condensador electrolítico	0,08	4	0,32
Condensador de 0.33 $\mu$ F	0,14	1	0,14
Condensador de 0.1 $\mu$ F	0,06	4	0,24
Condensador de 1 nF	0,04	3	0,12
Condensador de 33 pF	0,06	2	0,12
Resistencias de 0.25W	0,02	20	0,40
Kit de ultrasonidos C-7210	4,66	1	4,66
Sensor LM35DZ	1,63	1	1,63
Sensor HIH-3610	23,34	1	23,34
Pulsador	0,16	4	0,64
Interruptor	0,61	1	0,61
Conector de pilas de 9V	0,14	1	0,14
Display LCD 2x16 C-2602	11,30	1	11,30
Caja para montajes	12,88	1	12,88
Conjunto tornillos tuercas	0,04	10	0,40
Material adicional	3,00	1	3,00
Presupuesto total del material			71,79

Por material adicional, se entiende que es el material que no se ha llegado a gastar las unidades con que se mide al completo, y se ha hecho una estimación aproximada del coste de todos ellos. En este conjunto se cuenta con la placa, el cable paralelo, la funda termorretráctil conectores, estaño, etc.

### 8.1.2 Coste del diseño

En este apartado se cuenta el tiempo empleado por el ingeniero en llevar a cabo el diseño completo del medidor de distancias. Posteriormente se opera con el salario hora, estipulado para este tipo de personal cualificado.

El tiempo empleado en la realización del diseño, se considera de 7 meses, a jornada completa y el precio hora de esta persona esta estipulado en 10€. A continuación se realizan los cálculos con estos factores.

$$N^{\circ} deHoras = 7(meses) \times 20\left(\frac{dias}{mes}\right) \times 8\left(\frac{horas}{dia}\right) \Rightarrow N^{\circ} deHoras = 1120$$

$$CosteDiseño = N^{\circ} deHoras \times Precio(\epsilon) / Hora \rightarrow CosteDiseño = 1120 \times 10$$

$$CosteDiseño = 11200 \epsilon$$

## 8.2 EVALUACIÓN DE COSTE DEL PRODUCCIÓN

Para conseguir recuperar el coste del prototipo, se estima que pueden llegar a realizarse 1000 ventas de este medidor. La realización de estas unidades, también generan gastos que hay que tener en cuenta, y se calculan en los siguientes apartados.

### 8.2.1 Coste del material

Aquí se parte del resultado obtenido en el coste del material del prototipo, al cual se le aplica un descuento sobre los componentes, debido al pedido de esta gran cantidad de unidades. Con este descuento fijado a un 35 %, se puede calcular el coste del material para las 1000 unidades a poner a la venta.

$$CosteMaterial_{(1000Uni)} = CosteMaterial_{(Prototipo)} \times N^{\circ} Unidades \times (1 - Descuento_{(1000Uni)})$$

$$CosteMaterial_{(1000Uni)} = 71,79 \epsilon \times 1000 \times (1 - 0.35)$$

$$CosteMaterial_{(1000Uni)} = 46663,5 \epsilon$$

### 8.2.2 Coste de la mano de obra

En este apartado se cuenta el tiempo empleado, por un operario en el montaje de los 1000 medidores que se pretenden vender. Posteriormente se opera con el salario hora, estipulado para un operario.

El tiempo empleado en montar un medidor, se considera de 2 horas, y el precio hora de esta persona esta estipulado en 6€. A continuación se realizan los cálculos con estos factores.

$$N^{\circ} de Horas = 2 \left( \frac{Horas}{Unidad} \right) \times 1000 Unidades \Rightarrow \boxed{N^{\circ} de Horas = 2000}$$

$$Coste Mano Obra = N^{\circ} de Horas \times \frac{Precio(\text{€})}{Hora} \rightarrow Coste Mano Obra = 2000 \times 6$$

$$\boxed{Coste Mano Obra = 12000 \text{ €}}$$

### 8.3 PRECIO DE SALIDA AL MERCADO

Teniendo ya conocidos todos los gastos que supone el conseguir las 1000 unidades de este producto, se puede calcular a que precio debe salir al mercado para obtener un beneficio neto del 10%. Los cálculos realizados se desarrollan a continuación.

$$Gasto Unidad = \frac{Coste Diseño + Coste Material + Coste Mano Obra}{1000}$$

$$Gasto Unidad = \frac{11200 \text{ €} + 46663.5 \text{ €} + 12000 \text{ €}}{1000} \Rightarrow \boxed{Gasto Unidad = 69,86 \text{ €}}$$

$$Precio Salida = Gasto Unidad + 10\%(Gasto Unidad)$$

$$Precio Salida = 69.86 \text{ €} \times (1 + 0.1) \Rightarrow \boxed{Precio Salida = 76.85 \text{ €}}$$



## **9 CONCLUSIONES**

### **ÍNDICE:**

<b>9.1 CONCLUSIONES TÉCNICAS.....</b>	<b>95</b>
<b>9.2 CONCLUSIONES PERSONALES .....</b>	<b>95</b>



## 9.1 CONCLUSIONES TÉCNICAS

En el desarrollo de este proyecto, se han tocado varios temas del apasionante mundo de la electrónica. Donde el mayor protagonismo se lo lleva la programación del microcontrolador. Aunque tampoco hay que olvidar la parte analógica, encargada de la adaptación de señales con los transductores ultrasónicos.

Es ya conocida, la importancia que tienen los microcontroladores, sobre cualquier campo de la vida cotidiana o tecnológica, puesto que se encuentran en la mayoría de los aparatos de uso diario. Esta importancia está justificada por la gran cantidad de ventajas que ofrecen.

En el tema de los sensores de ultrasonidos, se ve como la tecnología avanza con respecto al descubrimiento y la utilidad de las diversas propiedades que ofrecen los distintos materiales encontrados en la naturaleza, incluso llegando a realizar compuestos artificiales que mejoren estas propiedades. Un claro ejemplo de ello se ve con la “piezoelectricidad”.

## 9.2 CONCLUSIONES PERSONALES

Como en la mayoría de los proyectos, se tocan varios campos tecnológicos. De los que se trata es de encontrar la relación existente entre las materias necesarias, para conseguir el desarrollo del proyecto.

Con la relación de este proyecto he conseguido varias ventajas, ~~de~~ las que espero hacer uso en mi futura vida profesional.

La que considero más importante, es la cantidad de conocimientos adquiridos con la búsqueda de soluciones a los problemas que iban surgiendo en el desarrollo de este proyecto, por hacer una puesta en común de todo lo aprendido, en las materias que tengan relación con este campo.

Otra ventaja a destacar, es la satisfacción personal de llegar a ver el funcionamiento de un aparato completo, que cumple todas las expectativas para las que se ha diseñado, en este caso la de medir distancia.

Como dice el refrán “La unión hace la fuerza”, considero que la consulta o comentario, sobre problemas, soluciones y dudas, que surgen a lo largo del desarrollo del proyecto de cualquier persona, con otras personas resulta ventajoso, tanto para el que los plantea, como para el que los conoce.

## **A APÉNDICE RECOPIULATORIO DE IMÁGENES VARIAS**

### **ÍNDICE:**

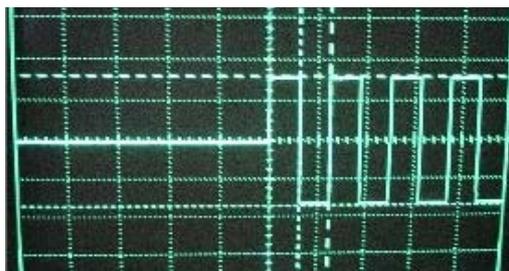
**A.I SEÑALES CAPTURADAS CON EL OSCILOSCOPIO..... 99**

**A.II FOTOGRAFÍAS DEL MEDIDOR EN SUS DISTINTAS FUNCIONES  
100**

**A.III FOTOGRAFÍAS DEL MODO EN REALIZAR LOS ENSALLOS.....101**

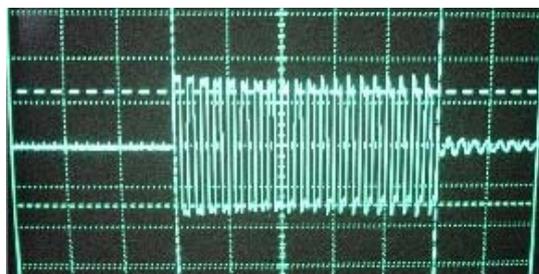
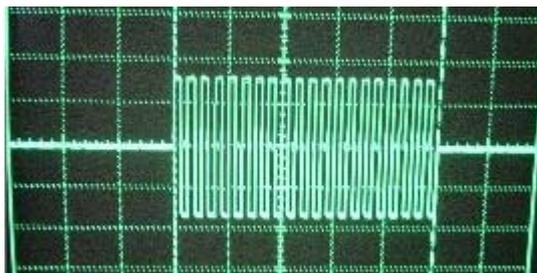


A.1 SEÑALES CAPTURADAS CON EL OSCILOSCOPIO



Generación de la señal alterna

Pulso elemental



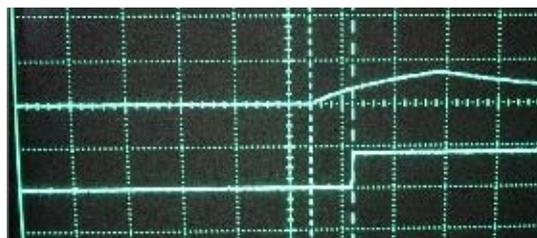
Señal de transmisión

Señal recibida amplificada y rectificada



Señal recibida integrada y en forma de pulso

Retado sobre el pulso, respecto a la señal integrada

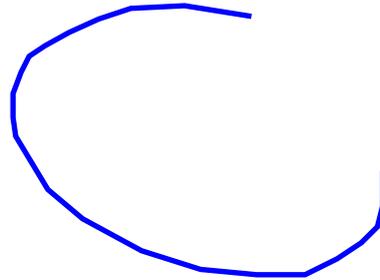


A.II FOTOGRAFÍAS DEL MEDIDOR EN SUS DISTINTAS FUNCIONES

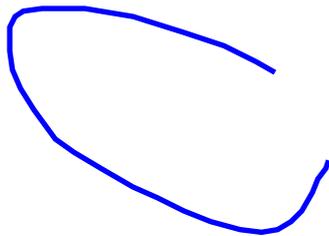


Con la función de Calibrado

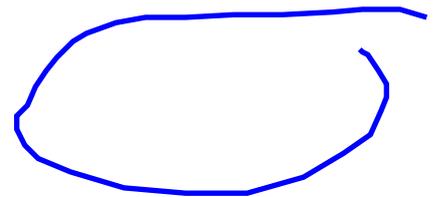
A la espera de MEDIR

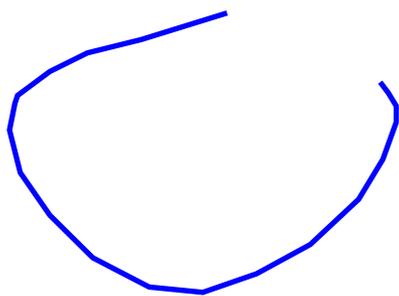


A la espera de decisión



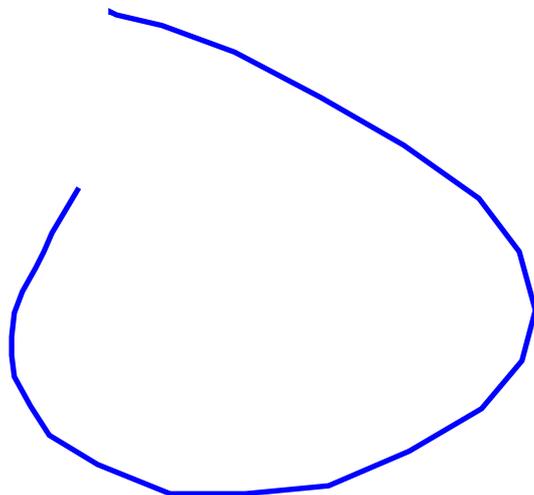
Realizando Medidas



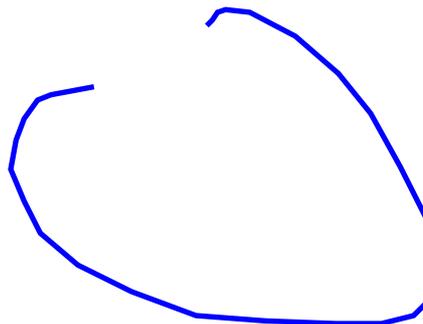


Almacenando la medida en la posición 12, inicialmente vacía

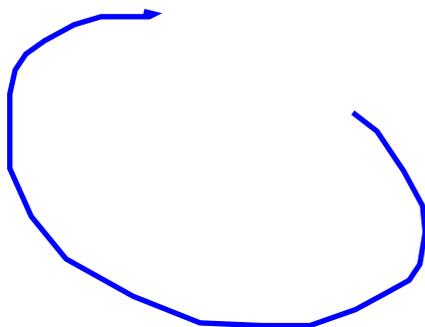
LEER y se ven los parámetros ambientales



A.III FOTOGRAFÍAS DEL MODO EN REALIZAR LOS ENSALLOS



Sobre el suelo se encuentran las referencias conocidas



Las ondas ultrasónicas rebotan con las puertas del armario



## **B APÉNDICE DEL CÓDIGO DE PROGRAMA**

### **ÍNDICE:**

<b>B.I ARCHIVO PRINCIPAL .....</b>	<b>105</b>
<b>B.II ARCHIVO VERMEN .....</b>	<b>116</b>
<b>B.III ARCHIVO CÁLCULO .....</b>	<b>123</b>
<b>B.IV ARCHIVO LINELCD .....</b>	<b>126</b>
<b>B.V ARCHIVO ACCESMEM.....</b>	<b>130</b>
<b>B.VI ARCHIVO PARAMETR .....</b>	<b>136</b>
<b>B.VII ARCHIVO SENULT .....</b>	<b>143</b>
<b>B.VIII ARCHIVO TABLAS.....</b>	<b>147</b>



B.I ARCHIVO PRINCIPAL

```

; PROGRAMA PRINCIPAL PARA REALIZAR MEDIDAS DE DISTANCIAS A PARTIR DEL TIEMPO EMPLEADO
;EN RECORRER UNA SEÑAL DE ULTRASONIDOS, UN CAMINO DE IDA Y VUELTA ENTRE LOS PUNTOS
;QUE DELIMITAN LA DISTANCIA A MEDIR

; La señal de ultrasonidos de 40 kHz se genera con el PIC, y la emision se realiza
;durante periodos de tiempo multiples del intervalo elemental (0.5 ms) incrementandose
;el factor multiplicador al no recibirse la señal rebotada en las condiciones deseadas
; El tiempo lo cuenta el temporizador TMR1, el cual dispone de 16 Bits
; Hay que tener en cuenta la temperatura del ambiente por donde se mueve la señal
;acustica, y por ello se hace uso del "CAD"
; Se dispone de una memoria EEPROM, sobre la que se almacenan las medidas deseadas

Title "Medidor de Distancia por Ultrasonidos"
LIST P=16F876A          ; Se trabajara con el 16F876A
RADIX DEC              ; Por defecto se consideran los numeros decimales
INCLUDE "P16F876A.INC" ; Las variables permitidas se encuentran en el include
ERRORLEVEL -305, -302 ; No se tiene en cuenta los avisos 305 y 302

__CONFIG _CP_OFF & _WDT_ON & _HS_OSC & _PWRTE_OFF & _LVP_OFF
; Esta desactivada la proteccion de codigo (CP)
; Se activa el perro guardian (WDT)
; El oscilador es un resonador de alta velocidad (HS)
; El temporizador de arranque esta activado (PWRTE)
; Se anula la programacion dentro del circuito (LVP)

; Confiracion de los puertos utilizados en el programa
;
; "PORTA" dispone de dos entradas analogicas "RA0 y RA1"
; de una entrada para la referencia de tension positiva "RA3"
; de dos salidas digitales "RA2 Y RA5"
; de una entrada digital"RA4"
; El BYTE necesario para el funcionamiento del conversor A/D se carga en "ADCON1"
;
; "PORTB" dispone de tres salidas digitales "RB1-RB3" para el control del LCD
; de cinco entradas digitales "RB0, RB4-RB7"
; "PORTC" dispone de cuatro salidas "RC4-RC7" para el envio de datos al LCD
;

VARIABLE IniVariables = 0x20 ; Las variables se disponen a partir de la posicion 32
ORG IniVariables
; Variables locales

Decision res 1 ; Variable indicativa de la decision tomada
CorrecDista res 1 ; Variable que almacena el corrector de la distancia

Indicadores res 1 ; Variable que almacena bits indicativos del camino a seguir
; La estructura que toma la variable Indicadores, es la siguiente:
Interrup equ 0 ; El bit '0' se asocia a Interrup/NoInterrup (1/0)
MemVacua equ 1 ; El bit '1' se asocia a MemVacua/NoMemVacua (1/0)
MemLlena equ 2 ; El bit '2' se asocia a MemLlena/NoMemLlena (1/0)
Añadida equ 3 ; El bit '3' se asocia a Añadida/Sustituida (1/0)
MensaPart equ 4 ; El bit '4' se asocia a MensaParteA/MensaParteB (1/0)
ADormir equ 5 ; El bit '5' se asocia a Dormido/Despierto (1/0)
Medir equ 6 ; El bit '6' se asocia a Medir/Leer (1/0)
TmpDesbor equ 7 ; El bit '7' se asocia a TmpDesbordado/TmpNoDesbordado (1/0)

; Se realiza el renombrado de algunos parametros
PuerCon equ PORTB ; Se bautiza el puerto PORTB como Puerto de control
PuerDat equ PORTC ; Se bautiza el puerto PORTC como Puerto de datos

; Identificacion de las patillas asociadas al control del LCD
ComanCarac equ 1 ; La patilla "RB1" se asocia a Comando/Caracter (0/1)
EscriLectu equ 2 ; La patilla "RB2" se asocia a Escritura/Lectura (0/1)
InhibHabil equ 3 ; La patilla "RB3" se asocia a Inhibe/Habilita (0/1)

; Identificacion de las patillas asociadas al control del aparato completo
Recib equ 0 ; La patilla "RB0" se asocia a la recepcion del eco
Acept equ 4 ; La patilla "RB4" se asocia al pulsador de Aceptar
Subir equ 5 ; La patilla "RB5" se asocia al pulsador de Subir
Cance equ 6 ; La patilla "RB6" se asocia al pulsador de Cancelar
Bajar equ 7 ; La patilla "RB7" se asocia al pulsador de Bajar

VARIABLE IniVariables = $ ; Se actualiza la posicion de comienzo

```

## MEDIDOR DE DISTANCIA POR ULTRASONIDOS

```

                                ;de las variables libres

ORG    0                        ; Comienza el codigo en la posicion 0
goto   Programa                ; Se salta a la etiqueta indicada

;#####
;                                # ATENCION A INTERRUPCIONES #
;#####

org    4                        ; Posicion de comienzo de la atencion de las interrupciones
                                ; Al entrar en la atencion de interrupciones, automaticamente "GIE" 1->0

bcf    T1CON,TMR1ON            ; Se da la orden de fin de cuenta al temporizador
bcf    INTCON,INTE             ; Se deshabilita la interrupcion por flanco en "RB0/INT"

clrwdt                            ; Se inicializa el contador del Perro Guardian
                                ; Se comienza con la comprobacion del motivo de la interrupcion segun el orden
                                ; prioritario, que comienza con la llegada del ECO rebotado, se sigue con la
                                ; decision tomada por el usuario y se finaliza con el deabordamiento de "TMR1"

                                ; Se atiende a la interrupcion producida por la recepcion de la señal
btfsc  INTCON,INTF            ; Se comprueba la bandera indicativa de flanco en "RB0"
bsf    Decision,Recib         ; Se activa el indicativo sobre "Decision" del ECO recibido

                                ; Se atiende a la interrupcion producida por la toma de decision
btfss  INTCON,RBIF           ; Se comprueba la bandera indicativa de flanco en "RB4-RB7"
goto   RebosaTMR1            ; Se salta a la etiqueta indicada

                                ; Se identifica el pulsador que ha originado la interrupcion
btfsc  PuerCon,Acept          ; Se comprueba el estado del pulsador Aceptar
bsf    Decision,Acept         ; Se activa el indicativo sobre "Decision" de "Acept" pulsado
btfsc  PuerCon,Cance          ; Se comprueba el estado del pulsador Cancelar
bsf    Decision,Cance         ; Se activa el indicativo sobre "Decision" de "Cancel" pulsado
btfsc  PuerCon,Subir          ; Se comprueba el estado del pulsador Subir
bsf    Decision,Subir         ; Se activa el indicativo sobre "Decision" de "Subir" pulsado
btfsc  PuerCon,Bajar          ; Se comprueba el estado del pulsador Bajar
bsf    Decision,Bajar         ; Se activa el indicativo sobre "Decision" de "Bajar" pulsado

                                ; Se comprueba que tipo de flanco (0->1 o 1->0) a originado la interrupcion
                                ; siendo valido el suceso de un flanco ascendente en "RB4-RB7"
movlw  11110000b             ; Se contruye una mascara en "W" para aislar las entradas
andwf  PuerCon                ; correspondientes a los pulsadores sobre "PuerCon"
btfss  STATUS,Z               ; Z = 1 cuando no ha ocurrido un flanco ascendente
bcf    INTCON,RBIE            ; Se deshabilitan las interrupciones por flanco en "RB4-RB7"

RebosaTMR1                       ; Se atiende a la interrupcion producida por desbordarse "TMR1"
btfsc  PIR1,TMR1IF            ; Se comprueba la bandera indicativa de "TMR1" desbordado
bsf    Indicadores,TmpDesbor  ; Se activa el indicativo del desbordamiento de "TMR1"

                                ; Se bajan las banderas indicativas de las tres interrupciones posibles
bcf    INTCON,INTF            ; Se baja la bandera indicativa de flanco en "RB0"
bcf    INTCON,RBIF           ; Se baja la bandera indicativa de flanco en "RB4-RB7"
bcf    PIR1,TMR1IF            ; Se baja la bandera indicativa de "TMR1" desbordado

retfie                            ; Se retorna a la instruccion donde se origino la interrupcion
                                ; Al salir de la atencion de interrupciones, automaticamente "GIE" 0->1

;#####
;                                # PROGRAMA PRINCIPAL #
;#####

Programa

                                ; Se comienza con las instrucciones de configuracion del PIC
bcf    STATUS,RP1              ; Seleccion del Banco 1
bsf    STATUS,RP0

                                ; Configuracion de los puertos, para las operaciones necesarias
movlw  11011011b              ; "RA2 y RA5" son salidas, "RA4" es entrada digital
movwf  TRISA                   ; Seleccion de "RA0, RA1 y RA3" como entradas del CAD
movlw  10000101b              ; "RA0 y RA1" son entradas analogicas y "RA3" es "Vref+"
movwf  ADCON1                  ; Se justifican los 10 bits del conversor a la derecha
movlw  11110001b              ; Se carga a traves de "W" la configuracion elegida para
movwf  PuerCon                 ; "PuerCon" en su registro de control asociado

                                ; Configuracion de "OPTION_REG"
                                ; Se hace uso del perro guardian para despertar al micro en el estado "SLEEP"
movlw  11001111b              ; Se anulan las resistencias de "PULL-UP" de "PORTB"

```

## APÉNDICE B

```

movwf OPTION_REG      ; La intrrupcion por "RB0", es por un flanco ascendente
                        ; La preescala escogida inicialmente para el "WDT" es de 1:128
                        ; Configuracion de "PIE1"
bsf   PIE1,TMR1IE     ; Se habilita (unicamente) la interrupcion por desbordamiento de "TMR1"
                        ; El desbordamiento de "TMR1" puede interrumpir el programa

bcf   STATUS,RP1      ; Seleccion del Banco 0
bcf   STATUS,RP0

                        ; Configuracion del temporizador
movlw 00100000b       ; Se elige una preescala de 1:4, con lo que el resultado de
movwf T1CON           ; la cuenta, son los "uS" del viaje de ida de la señal

;#####
                        ; Configuracion del modo de comunicacion del PIC con el LCD
;#####

call  LCDPuert        ; Se configura el puerto de datos de comunicacion con el LCD
call  LCDReset        ; Se lleva a cabo la configuracion inicial del LCD

;#####
                        ; Operacion del programa, una vez configurado el PIC
;#####

                        ; Con "Aceptar" activado en el arranque del aparato,se entra en la subrutina
                        ; de calibrado, donde se compensa la dimension longitudinal del aparato
btfss PuerCon,Acept   ; Se comprueba el estado del pulsador Aceptar
goto  Comienzo        ; Se salta a la etiqueta indicada
call  Espe20ms        ; Llamada a la subrutina de espera de 20 ms
btfsc PuerCon,Acept   ; Se comprueba nuevamente el estado del pulsador Aceptar
call  Calibrado       ; Llamada a la subrutina de calibrado del aparato

Comienzo
                        ; Se extrae el corrector de distancia del aparato, guardado en la "EEPROM"
movlw 1                ; Se direcciona la posicion de la memoria EEPROM, donde se
movwf Posicion         ; encuentra el Byte buscado
call  LeeBYTE          ; Llamada a la subrutina que lee un BYTE de la "EEPROM"
movf  ByteMem,W        ; Se realiza el traspaso de "ByteMem" a "CorrecDista"
movwf CorrecDista     ; a traves del registro de trabajo (w)

                        ; Se realiza la primera medida del parametro meteorologico, Temperatura
bcf   Indicadores,ADormir ; Se desactiva el indicativo de convertir AD durmiendo
call  MedTempera       ; Llamada a la subrutina que mide la temperatura con el "CAD"

clrwdt                 ; Se inicializa el contador del Perro Guardian
                        ; Se extrae de la "EEPROM" la ultima posicion de datos ocupada
clrf  Posicion         ; Se direcciona con "Posicion" el 1º Byte de la "EEPROM"
call  LeeBYTE          ; Llamada a la subrutina que lee un BYTE de la "EEPROM"
movf  ByteMem,W        ; Se realiza el traspaso de "ByteMem" a "PosiFin"
movwf PosiFin          ; a traves del registro de trabajo (w)
btfsc STATUS,Z        ; Z = 0 mientras no este vacia la memoria de datos
goto  MemoriVacía     ; Se salta a la etiqueta indicada

                        ; Se continua ejecutando el codigo para la situación de memoria ocupada
sublw PosiLimite      ; Se realiza la resta (PosiLimite - PosiFin)
btfsc STATUS,Z        ; Z = 0 mientras no este completa la memoria de datos
bsf   Indicadores,MemLlena ; Se activa el indicativo de memoria llena
btfss STATUS,C        ; C = 1 mientras "PosiFin" apunta dentro del rango permitido
goto  Inicializado   ; Se salta a la etiqueta indicada

;#####
Bifurca
                        ; Se entra en la pantalla inicial, desde la que se elige el camino deseado
clrwdt                 ; Se inicializa el contador del Perro Guardian
clrf  Decision         ; Se inicializa la variable Decision
bsf   BitsCont,Seleclin ; Se activa el indicativo de segunda linea del LCD
movlw 10000100b
movwf Mensaje         ; Se carga en "Mensaje" el BYTE asociado al mensaje 13
call  MensajeSem       ; Llamada a la subrutina que visualiza un mensaje de 1 linea

;### En este momento aparece representado en la segunda linea del LCD,
;el mensaje fijo de la situacion en que la memoria no esta vacia ###

btfsc Indicadores,MemLlena ; Se comprueba el estado de ocupacion de la "EEPROM"
goto  MemoriLlena     ; Se salta a la etiqueta indicada

                        ; Se continua ejecutando el codigo para la situación de memoria incompleta

```

## MEDIDOR DE DISTANCIA POR ULTRASONIDOS

```
call   TeclasActivas ; Llamada a la subrutina de espera a pulsadores desactivados

;### A partir de ahora se muestra sobre la primera linea del LCD, dos mensajes
;de forma alterna, de la situacion en que la memoria no esta vacia ###

bsf    Indicadores,ADormir ; Se activa el indicativo de convertir AD durmiendo

CambLin1
bsf    BitsCont,BorSbesc ; Se activa el indicativo de sobreescritura en el LCD
btfsc  Indicadores,MensaPart ; Se comprueba el indicativo selector del mensaje
goto   MeteoroOcupa ; Se salta a la etiqueta indicada
call   LineaOcuMem ; Llamada a la subrutina que representa el mensaje inicial
goto   ParteOcupa ; Se salta a la etiqueta indicada

MeteoroOcupa
call   LineaHT ; Llamada a la subrutina que representa el mensaje secundario

ParteOcupa
movf   Decision,F ; Se comprueba la existencia de decision
btfss  STATUS,Z ; Z = 1 mientras no se defina una decision
goto   ExaBifurca ; Se salta a determinar la decision tomada
call   MedirSensor ; Llamada a la subrutina que mide el sensor correspondiente
; mientras se encuentra dormido el microcontrolador
movf   Decision,F ; Se comprueba la existencia de decision
btfss  STATUS,Z ; Z = 1 mientras no se defina una decision
goto   ExaBifurca ; Se salta a determinar la decision tomada

goto   CambLin1 ; Se salta a conmutar el mensaje alterno sobre la 1ª linea

MemoriLlena
; Se continua ejecutando el codigo para la situacion de memoria completa
call   TeclasActivas ; Llamada a la subrutina de espera a pulsadores desactivados

;### A partir de ahora se muestra sobre la primera linea del LCD, dos mensajes
;de forma alterna, de la situacion en que la memoria no esta vacia ###

bsf    Indicadores,ADormir ; Se activa el indicativo de convertir AD durmiendo

OtrLlena
bsf    BitsCont,BorSbesc ; Se activa el indicativo de sobreescritura en el LCD
btfsc  Indicadores,MensaPart ; Se comprueba el indicativo selector del mensaje
goto   MeteoroLlena ; Se salta a la etiqueta indicada
movlw  10000001b
movwf  Mensaje ; Se carga en "Mensaje" el BYTE asociado al mensaje 11
call   MensajeSem ; Llamada a la subrutina que visualiza un mensaje de 1 linea
;### En este momento aparece representado el mensaje inicial sobre el LCD###

goto   ParteLlena ; Se salta a la etiqueta indicada

MeteoroLlena
call   LineaHT ; Llamada a la subrutina que representa el mensaje secundario

ParteLlena
movf   Decision,F ; Se comprueba la existencia de decision
btfss  STATUS,Z ; Z = 1 mientras no se defina una decision
goto   ExaBifurca ; Se salta a determinar la decision tomada
call   MedirSensor ; Llamada a la subrutina de medida del sensor correspondiente
; mientras se encuentra dormido el microcontrolador
movf   Decision,F ; Se comprueba la existencia de decision
btfss  STATUS,Z ; Z = 1 mientras no se defina una decision
goto   ExaBifurca ; Se salta a determinar la decision tomada

goto   OtrLlena ; Se salta a conmutar el mensaje alterno sobre la 1ª linea

ExaBifurca
; Se elige el camino determinado por la decision tomada
call   Antirrebotes ; Llamada a la subrutina de espera a pulsaciones sin rebotes
btfsc  Decision,Cance ; Se comprueba el estado del indicativo "Cance" en "Decision"
goto   QuizRESET ; Se salta a confirmar el deseo de RESETEAR la memoria
btfsc  Decision,Subir ; Se comprueba el estado del indicativo "Subir" en "Decision"
goto   LEER ; Se salta a realizar la lectura de la "EEPROM"
btfsc  Decision,Bajar ; Se comprueba el estado del indicativo "Bajar" en "Decision"
goto   MEDIR ; Se salta a realizar medidas

goto   Bifurca ; Se salta al comienzo de la etapa, ante una decision incierta

QuizRESET
```

## APÉNDICE B

```

; Se confirma la certeza del deseo de RESETEAR la memoria
clrf Decision ; Se inicializa la variable Decision
btfss PuerCon,Cance ; Se comprueba el estado del pulsador Cancelar
goto Bifurca ; Se salta a la etiqueta indicada

sleep ; Se realiza una espera, durante el sueño del Perro Guardian
btfss PuerCon,Cance ; Se comprueba nuevamente el estado del pulsador Cancelar
goto Bifurca ; Se salta a la etiqueta indicada

; Se realiza el RESETEADO de la memoria, inicializando el puntero PosiFin
Inicializado
; Se actualiza "PosiFin" con '0' y se guarda este dato en la "EEPROM"
clrf PosiFin ; Se inicializa la variable PosiFin
call GuardPosFin ; Se actualiza "PosiFin" sobre su posición en la "EEPROM"
bcf Indicadores,MemLlena ; Se desactiva el indicativo de memoria llena

;#####
MemoriVacía
; Se entra en la pantalla inicial, cuando no existen medidas guardadas
clrwdt ; Se inicializa el contador del Perro Guardian
bsf Indicadores,MemVacía ; Se activa el indicativo de memoria vacía
bsf BitsCont,Seleclín ; Se activa el indicativo de segunda línea del LCD
movlw 00000100b
movwf Mensaje ; Se carga en "Mensaje" el BYTE asociado al mensaje 03
call MensajeSem ; Llamada a la subrutina que visualiza un mensaje de 1 línea

;### En este momento aparece representado en la segunda línea del LCD,
;el mensaje fijo de la situación en que la memoria se encuentra vacía ###

call TeclasActivas ; Llamada a la subrutina de espera a pulsadores desactivados

;### A partir de ahora se muestra sobre la primera línea del LCD, dos mensajes
;de forma alterna, de la situación en que la memoria se encuentra vacía ###

bsf Indicadores,ADormir ; Se activa el indicativo de convertir AD durmiendo

PrimLinVac
bsf BitsCont,BorSbesc ; Se activa el indicativo de sobrescritura en el LCD
btfsc Indicadores,MensaPart ; Se comprueba el indicativo selector del mensaje
goto MeteoroVacía ; Se salta a la etiqueta indicada
movlw 00000001b
movwf Mensaje ; Se carga en "Mensaje" el BYTE asociado al mensaje 01
call MensajeSem ; Llamada a la subrutina que visualiza un mensaje de 1 línea
;### En este momento aparece representado el mensaje inicial sobre el LCD###

goto ParteVacía ; Se salta a la etiqueta indicada

MeteoroVacía
call LineaHT ; Llamada a la subrutina que representa el mensaje secundario

ParteVacía
btfsc Decision,Bajar ; Se comprueba el estado del indicativo "Bajar" en "Decision"
goto MEDIR ; Se salta a realizar medidas
bsf INTCON,RBIE ; Se habilitan las interrupciones por flanco en "RB4-RB7"
call MedirSensor ; Llamada a la subrutina de medida del sensor correspondiente
; mientras se encuentra dormido el microcontrolador
btfsc Decision,Bajar ; Se comprueba el estado del indicativo "Bajar" en "Decision"
goto MEDIR ; Se salta a realizar medidas
bsf INTCON,RBIE ; Se habilitan las interrupciones por flanco en "RB4-RB7"
goto PrimLinVac ; Se salta a conmutar el mensaje alterno sobre la 1ª línea

;#####
MEDIR
; En esta etapa se realizan medidas de distancia
clrwdt ; Se inicializa el contador del Perro Guardian
btfss Indicadores,MemVacía ; Se comprueba el estado de vaciado de la "EEPROM"
goto AMedir ; Se salta a la etiqueta indicada

; Se confirma la certeza de la pulsación "Bajar"
call Espe20ms ; Llamada a la subrutina de espera de 20 ms
btfss PuerCon,Bajar ; Se comprueba el estado del pulsador Bajar
goto PrimLinVac ; Se salta a la etiqueta indicada

AMedir
bsf Indicadores,Medir ; Se activa el indicativo de realizar medida
clrf Decision ; Se inicializa la variable Decision

```

## MEDIDOR DE DISTANCIA POR ULTRASONIDOS

```

; Se direcciona la posicion de datos preferente para almacenar la medida
incf PosiFin,W ; Se guarda en "W" la 1ª posicion de datos vacante
btfsc Indicadores,MemLlena ; Se comprueba el estado de ocupacion de la "EEPROM"
incf UltiEscrita,W ; Se guarda en "W" la posicion siguiente a la ultima apuntada
movwf Posicion ; Se direcciona la posicion de datos preferente para su uso

; Se comprueba correcta localizacion del puntero Posicion
sublw PosiLimite ; Se realiza la resta (PosiLimite - Posicion)
movf Posicion,W ; Se guarda en "W" el dato contenido en "Posicion"
btfss STATUS,C ; C = 1 mientras "Posicion" apunta dentro del rango permitido
movlw 1 ; Se guarda en "W" el dato '1'
movwf Posicion ; Se direcciona la posicion de datos preferente para su uso

;### A partir de ahora se reduce el tiempo que duerme el perro guardian
;para que con la instruccion Sleep se origine una espera mas corta ###

bcf INTCON,GIE ; Se deshabilitan todas las interrupciones permitidas
;durante la permanencia en el banco 1
bcf STATUS,RP1 ; Seleccion del Banco 1
bsf STATUS,RP0

; Se modifica la configuracion inicial de "OPTION_REG"
movlw 11001100b ; Se reduce la preescala del "WDT" a 1:16 para
movwf OPTION_REG ;que la espera con "Sleep" dure 288ms
bcf STATUS,RP1 ; Seleccion del Banco 0
bcf STATUS,RP0
bsf INTCON,GIE ; Se habilitan todas las interrupciones permitidas
;al pasar de nuevo al banco 0
clrwdt ; Se inicializa el contador del Perro Guardian

; Se escriben los mensajes correspondientes a esta etapa sobre el LCD
call LineaPosMed ; Llamada a la subrutina que completa la 2ª linea del LCD
call LineaSalvar ; Llamada a la subrutina que completa la 1ª linea del LCD

NuevaMedida
; Se realiza la medida de la distancia de forma repetitiva,
;saliedo de esta situacion al tomarse una decision
call TeclasActivas ; Llamada a la subrutina de espera a pulsadores desactivados
call CaptDistan ; Llamada a la subrutina que mide y visualiza la distancia
sleep ; Se realiza una espera, durante el sueño del Perro Guardian
movf Decision,F ; Se comprueba la existencia de decision
btfss STATUS,Z ; Z = 1 mientras no se defina una decision
goto ExaMEDIR ; Se salta a determinar la decision tomada

goto NuevaMedida ; Se salta a la actualizacion de la medida

ExaMEDIR
; Se elige el camino determinado por la decision tomada
call Antirrebotes ; Llamada a la subrutina de espera a pulsaciones sin rebotes
btfsc Decision,Accept ; Se comprueba el estado del indicativo "Accept" en "Decision"
goto Memorizar ; Se salta a la escritura de la medida en la "EEPROM"
btfsc Decision,Cance ; Se comprueba el estado del indicativo "Cance" en "Decision"
goto EtapaIni ; Se salta a la etapa desde la que se ha entrado en MEDIR
btfsc Decision,Subir ; Se comprueba el estado del indicativo "Subir" en "Decision"
goto PosMayorEs ; Se salta a apuntar la posicion inmediata superior permitida
btfsc Decision,Bajar ; Se comprueba el estado del indicativo "Bajar" en "Decision"
goto PosMenorEs ; Se salta a apuntar la posicion inmediata inferior permitida

goto NuevaMedida ; Se salta a la etiqueta indicada, ante una decision incierta

PosMayorEs
clrf Decision ; Se inicializa la variable Decision
btfsc Indicadores,MemVacía ; Se comprueba el estado de vaciado de la "EEPROM"
goto NuevaMedida ; Se salta a la etiqueta indicada

; Se guarda en "W" el dato con el que se opera en "Incremen"
movf Posicion,W ; Se guarda en "W" la ultima posicion de datos apuntada
btfsc Indicadores,MemLlena ; Se comprueba el estado de ocupacion de la "EEPROM"
incf Posicion,W ; Se guarda en "W" la posicion siguiente a la apuntada
call Incremen ; Llamada a la subrutina que incrementa una posicion
call UbicaPos ; Llamada a la subrutina que actualiza "Posicion" en el LCD
goto NuevaMedida ; Se salta a la etiqueta indicada

PosMenorEs
clrf Decision ; Se inicializa la variable Decision
btfsc Indicadores,MemVacía ; Se comprueba el estado de vaciado de la "EEPROM"
goto NuevaMedida ; Se salta a la etiqueta indicada

```

## APÉNDICE B

```

; Se guarda en "PosiFin" el dato con el que se opera en "Decremen"
incf PosiFin,W ; Se guarda en "W" la 1ª posición de datos vacante
btfsc Indicadores,MemLlena ; Se comprueba el estado de ocupación de la "EEPROM"
movf PosiFin,W ; Se guarda en "W" la última posición de datos ocupada
movwf PosiFin ; Se actualiza la variable PosiFin
call Decremen ; Llamada a la subrutina que decrementa una posición

; Se devuelve el valor correcto a "PosiFin" tras retornar de "Decremen"
decf PosiFin,W ; Se guarda en "W" la posición anterior a la 1ª vacante
btfsc Indicadores,MemLlena ; Se comprueba el estado de ocupación de la "EEPROM"
movf PosiFin,W ; Se guarda en "W" la última posición de datos ocupada
movwf PosiFin ; Se carga en "PosiFin" el contenido de "W"
call UbicaPos ; Llamada a la subrutina que actualiza "Posición" en el LCD
goto NuevaMedida ; Se salta a la etiqueta indicada

#####
Memorizar

;### Tras salir de la operación de medida se actualiza el preescalado del "WDT"
;para que la instrucción Sleep, duerma el microcontrolador durante 2.3 segundos ###

bcf INTCON,GIE ; Se deshabilitan todas las interrupciones permitidas
;durante la permanencia en el banco 1
bcf STATUS,RP1 ; Selección del Banco 1
bsf STATUS,RP0

; Se devuelve la configuración inicial de "OPTION_REG"
movlw 11001111b ; Se amplía el preescalado del contador del Perro Guardian
movwf OPTION_REG ;al máximo permitido, resultando ser 1:128
bcf STATUS,RP1 ; Selección del Banco 0
bcf STATUS,RP0
bsf INTCON,GIE ; Se habilitan todas las interrupciones permitidas
;al pasar de nuevo al banco 0

; Se realiza la memorización de la medida en la posición de datos seleccionada
call EscriDato ; Llamada a la subrutina que escribe un dato en la "EEPROM"

; Se comprueba el estado de la posición de datos donde se guarda la medida
btfss Indicadores,Añadida ; Se comprueba el estado de la posición apuntada
goto VerEscritura ; Se salta a la etiqueta indicada

; Se actualizan los punteros de la "EEPROM"
movf Posicion,W ; Se realiza el traspaso de "Posicion" a "PosiFin"
movwf PosiFin ;a través del registro de trabajo (w)
call GuardPosFin ; Se actualiza "PosiFin" sobre su posición en la "EEPROM"
bcf Indicadores,MemVacía ; Se desactiva el indicativo de memoria vacía

VerEscritura

; Se escribe el mensaje correspondiente a esta etapa sobre el LCD
call LineaGuardada ; Llamada a la subrutina que actualiza la 1ª línea del LCD

; Se comprueba el alcance de ocupación de la memoria
movf PosiFin,W ; Se guarda en "W" la última posición de datos ocupada
sublw PosiLimite ; Se realiza la resta (PosiLimite - PosiFin)
btfsc STATUS,Z ; Z = 0 mientras no este completa la memoria de datos
bsf Indicadores,MemLlena ; Se activa el indicativo de memoria llena
sleep ; Se realiza una espera, durante el sueño del Perro Guardian
goto Bifurca ; Se salta a la etiqueta indicada

EtapaIni

;### Tras salir de la operación de medida se actualiza el preescalado del "WDT"
;para que la instrucción Sleep, duerma el microcontrolador durante 2.3 segundos ###

bcf INTCON,GIE ; Se deshabilitan todas las interrupciones permitidas
;durante la permanencia en el banco 1
bcf STATUS,RP1 ; Selección del Banco 1
bsf STATUS,RP0

; Se devuelve la configuración inicial de "OPTION_REG"
movlw 11001111b ; Se amplía el preescalado del contador del Perro Guardian
movwf OPTION_REG ;al máximo permitido, resultando ser 1:128
bcf STATUS,RP1 ; Selección del Banco 0
bcf STATUS,RP0
bsf INTCON,GIE ; Se habilitan todas las interrupciones permitidas
;al pasar de nuevo al banco 0

; Se busca la etapa desde la que se entro en MEDIR

```

## MEDIDOR DE DISTANCIA POR ULTRASONIDOS

```

decf   Posicion,F      ; Se direcciona la ultima posicion de datos actualizada
btfsc  Indicadores,MemVacía ; Se comprueba el estado de vaciado de la "EEPROM"
goto   MemoriVacía    ; Se salta a la etiqueta indicada

goto   Bifurca        ; Se salta a la etiqueta indicada

;#####
LEER
; En esta etapa se realiza la lectura de medidas guardadas
clrwdt ; Se inicializa el contador del Perro Guardian
bcf    Indicadores,Medir ; Se desactiva el indicativo de realizar medida
clrf   Decision         ; Se inicializa la variable Decision

; Se direcciona la posicion definida por defecto
movf   PosiFin,W        ; Se guarda en "W" la ultima posicion ocupada
btfsc  Indicadores,MemLlena ; Se comprueba el estado de ocupacion de la "EEPROM"
movlw  1                ; Se guarda en "W" el dato '1'
movwf  Posicion         ; Se direcciona la posicion preferente para leer el dato

; Se visualizan sobre el LCD los caracteres de permanencia en esta etapa
call   LineaPosMed     ; Llamada a la subrutina que completa la 2ª linea del LCD
bcf    Indicadores,ADormir ; Se desactiva el indicativo de convertir AD durmiendo
call   MedirSensor    ; Llamada a la subrutina que mide el sensor correspondiente
bsf    BitsCont,BorSbesc ; Se activa el indicativo de sobrescritura en el LCD
call   LineaHT        ; Llamada a la subrutina que completa la 1ª linea del LCD
bsf    Indicadores,ADormir ; Se activa el indicativo de convertir AD durmiendo

NuevaLectura
call   TeclasActivas ; Llamada a la subrutina de espera a pulsadores desactivados
; Se visualiza en el LCD el dato guardado en la posicion apuntada
call   PosiMed       ; Llamada a la subrutina que actualiza la 2ª linea del LCD
movf   Decision,F    ; Se comprueba la existencia de decision
btfss  STATUS,Z      ; Z = 1 mientras no se defina una decision
goto   ExaLEER       ; Se salta a determinar la decision tomada

NuevaCAD
; Se actualizan de forma alternativa los parametros meteorologicos en el LCD
clrf   Decision      ; Se inicializa la variable Decision
bsf    INTCON,RBIE   ; Se habilitan las interrupciones por flanco en "RB4-RB7"
call   MedirSensor   ; Llamada a la subrutina de medida del sensor correspondiente
; mientras se encuentra dormido el microcontrolador
movf   Decision,F    ; Se comprueba la existencia de decision
btfss  STATUS,Z      ; Z = 1 mientras no se defina una decision
goto   ExaLEER       ; Se salta a determinar la decision tomada

call   HumeTempe     ; Llamada a la subrutina que representa la "Hum" y la "Temp"
movf   Decision,F    ; Se comprueba la existencia de decision
btfss  STATUS,Z      ; Z = 1 mientras no se defina una decision
goto   ExaLEER       ; Se salta a determinar la decision tomada

goto   NuevaCAD      ; Se salta a la actualizacion de "Hum" y "Temp"

ExaLEER
; Se elige el camino determinado por la decision tomada
call   Antirrebotes ; Llamada a la subrutina de espera a pulsaciones sin rebotes
btfsc  Decision,Cance ; Se comprueba el estado del indicativo "Cance" en "Decision"
goto   Bifurca      ; Se regresa a la etapa inicial
btfsc  Decision,Subir ; Se comprueba el estado del indicativo "Subir" en "Decision"
goto   PosMayorLe   ; Se salta a apuntar la posicion inmediata superior permitida
btfsc  Decision,Bajar ; Se comprueba el estado del indicativo "Bajar" en "Decision"
goto   PosMenorLe   ; Se salta a apuntar la posicion inmediata inferior permitida

goto   NuevaCAD     ; Se salta a la etiqueta indicada, ante una decision incierta

PosMayorLe
; Se guarda en "W" el dato con el que se opera en "Incremen"
clrf   Decision      ; Se inicializa la variable Decision
incf   Posicion,W    ; Se guarda en "W" la posicion siguiente a la apuntada
call   Incremen      ; Llamada a la subrutina que incrementa una posicion
goto   NuevaLectura ; Se salta a la etiqueta con la que se lee de la "EEPROM"

PosMenorLe
clrf   Decision      ; Se inicializa la variable Decision
call   Decremen      ; Se llama a la subrutina que decrementa una posicion
goto   NuevaLectura ; Se salta a la etiqueta con la que se lee de la "EEPROM"

;#####

```

## APÉNDICE B

```

; Subrutinas relacionadas con el control del programa, por medio de los pulsadores
;#####
; Subrutina de direccionamiento a la posicion inmediata superior
;#####
; En esta subrutina se opera con el dato cargado en "W"

Increment:
    subwf  PosiFin,W      ; Se realiza la resta (PosiFin - W)
    btfss  STATUS,C      ; C = 1 mientras "Posicion" no sobrepase el margen superior
    clrf   Posicion      ; Se borra la variable Posicion
    incf   Posicion,F    ; Se direcciona la posicion inmedidata superior permitida
    return                                ; Retorna de la subrutina

;#####
; Subrutina de direccionamiento a la posicion inmediata inferior
;#####
; En esta subrutina se opera con los datos cargados en "Posicion" y "PosiFin"

Decremen:
    decf   Posicion,W    ; Se guarda en "W" la posicion anterior a la apuntada
    btfsc  STATUS,Z      ; Z = 0 mientras "Posicion" no sobrepase el margen inferior
    movf   PosiFin,W     ; Se guarda en "W" la ultima posicion ocupada
    movwf  Posicion      ; Se direcciona la posicion inmedidata inferior permitida
    return                                ; Retorna de la subrutina

;#####
; Subrutina de espera a la desactivacion de los pulsadores
;#####

TeclasActivas:
    clrwdt                                ; Se inicializa el contador del Perro Guardian
    movlw  11110000b                      ; Se contruye una mascara en "W" para aislar las entradas
    andwf  PuerCon                         ; correspondientes a los pulsadores sobre "PuerCon"
    btfss  STATUS,Z                       ; Z = 1 cuando ningun pulsador este activado
    goto   TeclasActivas                  ; Se salta al comienzo de esta subrutina

    movlw  10001000b                      ; Se actiban las interrupciones Globales y de PORTB
    movwf  INTCON                          ; GIE = Gobales RBIE = flanco en "RB4-RB7"
    return                                ; Retorna de la subrutina

;#####
; Subrutina que anula los rebotes iniciales en la evaluacion de las pulsaciones
;#####

Antirrebotes:
    clrwdt                                ; Se inicializa el contador del Perro Guardian
    call   Espe20ms                       ; Llamada a la subrutina de espera de 20 ms
    movlw  11110000b                      ; Se contruye una mascara en "W" para aislar las entradas
    andwf  PuerCon                         ; correspondientes a los pulsadores sobre "PuerCon"
    btfss  STATUS,Z                       ; Z = 1 cuando ningun pulsador este activado
    return                                ; Retorna de la subrutina

    clrf   Decision                       ; Se borra la variable Decision ante una falsa pulsacion
    movlw  10001000b                      ; Se actiban las interrupciones Globales y de PORTB
    movwf  INTCON                          ; GIE = Gobales RBIE = flanco en "RB4-RB7"
    return                                ; Retorna de la subrutina

;#####
; Subrutina de representacion sobre el LCD, de la posicion de memoria apuntada
;#####

UbicaPos:
    ; Se visualiza la posicion de datos apuntada sobre el LCD
    movlw  LCDLinea2+6                    ; Se coloca el cursor en la posicion deseada de la linea 2
    call   LCDComando                     ; Se envia dicho comando al LCD
    movf   Posicion,W                     ; Se guarda en "W" el dato contenido en "Posicion"
    call   Dato2Cifras                     ; Llamada a la subrutina que visualiza datos de 2 cifras

    ; Se visualiza el estado actual de la posicion de datos apuntada
    movlw  LCDLineal+7                    ; Se coloca el cursor en la posicion deseada de la linea 1
    call   LCDComando                     ; Se envia dicho comando al LCD
    movf   Posicion,W                     ; Se guarda en "W" la posicion direccionada
    subwf  PosiFin,W                       ; Se realiza la resta (PosiFin - Posicion)
    movlw  255                             ; Se guarda en "W" el valor del caracter deseado de visualizar
    btfss  STATUS,C                       ; C = 1 cuando se direcciona una posicion de datos ocupada
    movlw  219                             ; Se guarda en "W" el valor del caracter deseado de visualizar

```

## MEDIDOR DE DISTANCIA POR ULTRASONIDOS

```

    call  LCDCaracter    ; Se envia dicho caracter al LCD
    return                ; Retorna de la subrutina

;#####
; Subrutina donde se realiza el calibrado en diatancia del aparato de medida
;#####

Calibrado:
    ; Se visualiza el mensaje anunciativo de entrada al calibrado
    movlw 00000001b
    movwf Mensaje        ; Se carga en "Mensaje" el BYTE asociado al mensaje 1
    call  MensajeCom     ; Llamada a la subrutina que visualiza un mensaje de 2 linea
    sleep                ; Se realiza una espera, durante el sueño del Perro Guardian
    clrf  Decision       ; Se inicializa la variable Decision

    ; Se visualiza el mensaje fijo de la realizacion del calibrado
    movlw 00000010b
    movwf Mensaje        ; Se carga en "Mensaje" el BYTE asociado al mensaje 2
    call  MensajeCom     ; Llamada a la subrutina que visualiza un mensaje de 2 linea
    movlw LCDLineal     ; Se coloca el cursor en la posicion deseada de la linea 1
    call  LCDComando     ; Se envia dicho comando al LCD
    movlw 127           ; Se guarda en "W" el valor correspondiente al caracter "<-
    call  LCDCaracter    ; Se envia dicho caracter al LCD
    movlw LCDLineal+15  ; Se coloca el cursor en la posicion deseada de la linea 1
    call  LCDComando     ; Se envia dicho comando al LCD
    movlw 126           ; Se guarda en "W" el valor correspondiente al caracter "->
    call  LCDCaracter    ; Se envia dicho caracter al LCD

;### A partir de ahora se reduce el tiempo que duerme el perro guardian
;para que con la instruccion Sleep se origine una espera mas corta ###

    bcf  INTCON,GIE     ; Se deshabilitan todas las interrupciones permitidas
                        ;durante la permanencia en el banco 1
    bcf  STATUS,RP1     ; Seleccion del Banco 1
    bsf  STATUS,RP0

    ; Se modifica la configuracion inicial de "OPTION_REG"
    movlw 11001100b    ; Se reduce la preescala del "WDT" a 1:16 para
    movwf OPTION_REG   ;que la espera con "Sleep" dure 288ms
    bcf  STATUS,RP1     ; Seleccion del Banco 0
    bcf  STATUS,RP0
    bsf  INTCON,GIE     ; Se habilitan todas las interrupciones permitidas
                        ;al pasar de nuevo al banco 0
    clrwdt              ; Se inicializa el contador del Perro Guardian

OtraDistancia
    ; Se realiza la medida de la distancia de forma repetitiva,
    ;saliendo de esta situacion al tomarse una decision
    call  TeclasActivas ; Llamada a la subrutina de espera a pulsadores desactivados
    call  CaptDistan    ; Llamada a la subrutina que mide y visualiza la distacia
    sleep                ; Se realiza una espera, durante el sueño del Perro Guardian
    movf  Decision,F    ; Se comprueba la existencia de decision
    btfsc STATUS,Z      ; Z = 0 cuando se defina una decision
    goto  OtraDistancia ; Se salta a la etiqueta indicada

ExaDistan
    ; Se elige el camino determinado por la decision tomada
    call  Antirrebotes  ; Llamada a la subrutina de espera a pulsaciones sin rebotes
    btfsc Decision,Acept ; Se comprueba el estado del indicativo "Acept" en "Decision"
    goto  SalvaCorrecDis ; Se salta a guardar el factor corrector de la distancia
    btfsc Decision,Cance ; Se comprueba el estado del indicativo "Cance" en "Decision"
    goto  Calibrado     ; Se regresa al comienzo de esta etapa
    btfsc Decision,Subir ; Se comprueba el estado del indicativo "Subir" en "Decision"
    goto  IncreCorrecDis ; Se salta a aumentar la distancia un centimetro
    btfsc Decision,Bajar ; Se comprueba el estado del indicativo "Bajar" en "Decision"
    goto  DecreCorrecDis ; Se salta a disminuir la distancia un centimetro
    goto  OtraDistancia ; Se salta a la etiqueta indicada, ante una decision incierta

SalvaCorrecDis
    ;### Tras salir de la operacion de medida se actualiza el preescalado del "WDT"
    ;para que la instruccion Sleep, duerma el microcontrolador durante 2.3 segundos ###

    bcf  INTCON,GIE     ; Se deshabilitan todas las interrupciones permitidas
                        ;durante la permanencia en el banco 1
    bcf  STATUS,RP1     ; Seleccion del Banco 1

```

## APÉNDICE B

```

bsf     STATUS,RP0

        ; Se devuelve la la configuracion inicial de "OPTION_REG"
movlw   11001111b   ; Se amplia el preescalado del contador del Perro Guardian
movwf   OPTION_REG  ;al maximo permitido, resultando ser 1:128
bcf     STATUS,RP1   ; Seleccion del Banco 0
bcf     STATUS,RP0
bsf     INTCON,GIE   ; Se habilitan todas las interrupciones permitidas
        ;al pasar de nuevo al banco 0

        ; Se actualiza el corretor de distancia sobre su posicion en la "EEPROM"
clrf    Decision     ; Se inicializa la variable Decision
call    GuardCorrecDista ; Se actualiza "CorrecDista" sobre la "EEPROM"

        ; Se visualiza el mensaje indicativo de la finalizacion del calibrado
movlw   00000100b
movwf   Mensaje      ; Se carga en "Mensaje" el BYTE asociado al mensaje 3
call    MensajeCom    ; Llamada a la subrutina que visualiza un mensaje de 2 linea
sleep
return  ; Retorna de la subrutina

IncreCorrecDis
clrf    Decision     ; Se inicializa la variable Decision
movlw   6             ; Se incrementa "CorrecDista" la cantidad correspondiente
addwf   CorrecDista,W ; a 1 cm guardando el resultado en "W"
btfsc   STATUS,C      ; C = 0 mientras "CorrecDista" no sobrepase el limite superior
movf    CorrecDista,W ; Se guarda en "W" el anterior valor corrector

movwf   CorrecDista   ; Se carga en "CorrecDista" el contenido de "W"
goto    OtraDistancia ; Se salta a la etiqueta indicada

DecreCorrecDis
clrf    Decision     ; Se inicializa la variable Decision
movlw   6             ; Se decrementa "CorrecDista" la cantidad correspondiente
subwf   CorrecDista,W ; a 1 cm guardando el resultado en "W"
btfss   STATUS,C      ; C = 1 mientras "CorrecDista" no sobrepase el limite inferior
movf    CorrecDista,W ; Se guarda en "W" el anterior valor corrector

movwf   CorrecDista   ; Se carga en "CorrecDista" el contenido de "W"
goto    OtraDistancia ; Se salta a la etiqueta indicada

;#####

INCLUDE "vermen.asm"
INCLUDE "calculo.asm"
INCLUDE "linelcd.asm"
INCLUDE "accesmem.asm"
INCLUDE "parametr.asm"
INCLUDE "senult.asm"
INCLUDE "tablas.asm"

END

```

B.II ARCHIVO VERMEN

```

; VERMEN.ASM
; Este archivo contiene todas las subrutinas necesarias para la visualizacion
; de mensajes y valores numericos sobre el LCD y las subrutinas con las que se
; cuenta un tiempo de espera fijado

; Para la visualizacion de los caracteres sobre la pantalla se agrupan en:

; Mensajes de dos lineas
; MENSAJE X
; MenXLin1: " _____" ;X° Mensaje 1ª Linea
; MenXLin2: " _____" ;X° Mensaje 2ª Linea

; Mensajes de una linea
; MENSAJE Y
; SemiMenY: " _____" ;Y° Mensaje de 1Lin"

; Valores numericos reales positivos
; NEntero,NDecimal/
; DATO Z -----,---/---
; NEntero,PotDiez
;#####

; Definicion de los comandos del LCD como constantes
BorrarLCD equ 00000001b ; Borra el LCD y coloca el cursor al principio
IncrementLCD equ 00000110b ; Incrementa una posicion el cursor tras realizar
; la escritura de cada caracter
DecremLCD equ 00000100b ; Decrementa una posicion el cursor tras realizar
; la escritura de cada caracter
SinCurLCD equ 00001100b ; Elimina la representacion visual tanto fijo como
; con parapadeo del cursor sobre la pantalla
SlzIzqLCD equ 00010000b ; Desplaza el cursor una posicion a la izquierda
SlzDerLCD equ 00010100b ; Desplaza el cursor una posicion a la derecha
FuncioLCD equ 00101000b ; Define el modo de funcionamiento del LCD para un
; bus de datos de 4 Hilos
LCDLinea1 equ 10000000b ; Posiciona el cursor al comienzo de la linea 1
LCDLinea2 equ 11000000b ; Posiciona el cursor al comienzo de la linea 2
;#####

IniSubrVermen equ $ ; Se guarda la direccion de memoria donde comenzara el codigo
; Variables locales
org IniVariables ; Las nuevas variables se disponen en posiciones libres

; Variables compartidas por las subrutinas que representan caracteres sobre el LCD
DatoBus res 1 ; Variable que guarda el dato a enviar por el BUS
Cuenta res 1 ; Variable que apunta el carracter deseado a representar

BitsCont res 1 ; Variable que almacena bits de control para la escritura
; La estructura que toma la variable BitsCont, es la siguiente:
SelecLin equ 0 ; El bit '0' se asocia a Linea1/Linea2 (0/1)
BorSbesc equ 1 ; El bit '1' se asocia a Borra/SobreEscr (0/1)
DereIzqu equ 2 ; El bit '2' se asocia a Derecha/Izquierda (0/1)
EnteDeci equ 3 ; El bit '3' se asocia a PEntera/PDecimal (0/1)

; Variable asociada a las subrutinas MensajeCom y MensajeSem
Mensaje res 1 ; Variable que guarda el indicador del mensaje elegido

; Variables asociadas a la subrutina BaseDiez
NEntero res 1 ; Variable que almacena el n° de digitos de la parte entera
NDecimal res 1 ; Variable que almacena el n° de digitos de la parte decimal
PotDiez res 1 ; Variable que almacena el factor divisor en potencia de '10'

; Variables asociadas a las subrutinas de retardo de tiempo
Conta011 res 1 ; Variable con carga inicial de 65 para Espe0.1ms
Conta51 res 1 ; 1ª Variable con carga inicial de 38 para Espe5ms
Conta52 res 1 ; 2ª Variable con carga inicial de 86 para Espe5ms
Conta201 res 1 ; 1ª Variable con carga inicial de 95 para Espe20ms
Conta202 res 1 ; 2ª Variable con carga inicial de 139 para Espe20ms

VARIABLE IniVariables = $ ; Se actualiza la posicion de comienzo
; de las variables aun libres
org IniSubrVermen ; Comienza el codigo de la subrutina en la dirreccion guardada
;#####

```

## APÉNDICE B

```

; Subrutinas de comunicacion con el controlador del LCD
;#####
; Subrutina de configuracion del numero de salidas del puerto de datos (PuerDat)
;#####

LCDPuert:
    bcf     STATUS,RP1           ; Seleccion del Banco 1
    bsf     STATUS,RP0
    movlw  00001111b           ; Se carga a traves de "W" la configuracion elegida para
    movwf  PuerDat             ; "PuerDat" en su registro de control asociado
    bcf     STATUS,RP1           ; Seleccion del Banco 0
    bcf     STATUS,RP0

    bcf     PuerCon,InhibHabil   ; Se deshabilita el LCD

    return                          ; Retorno de subrutina
;#####
; Subrutina de configuracion del modo de funcionamiento del LCD
;#####

LCDReset:
    ; Se define el modo de operacion deseado para el LCD
    movlw  FuncioLCD           ; Se carga en "W" el BYTE que define el funcionamiento
    call   LCDComandoInicio    ; Llamada a la subrutina que manda un comando inicial
    call   Espe5ms             ; Llamada a la subrutina de espera de 5 ms
    movlw  FuncioLCD           ; Se usa un bus de 4 hilos, indicando en "BYTE" Bit4=0
    call   LCDComandoInicio    ; Llamada a la subrutina que manda un comando inicial
    call   Espe5ms             ; Llamada a la subrutina de espera de 5 ms
    movlw  FuncioLCD           ; El display es de 2 lineas, indicando en "BYTE" Bit3=1
    call   LCDComandoInicio    ; Llamada a la subrutina que manda un comando inicial
    call   Espe5ms             ; Llamada a la subrutina de espera de 5 ms
    movlw  FuncioLCD           ; La fuente del LCD es 5*8, indicando en "BYTE" Bit2=0
    call   LCDComando         ; Llamada a la subrutina que manda un comando al LCD
    call   Espe5ms             ; Llamada a la subrutina de espera de 5 ms
    movlw  IncremLCD           ; Se incrementa el cursor con el display inmovil
    call   LCDComando         ; Llamada a la subrutina que manda un comando al LCD
    movlw  SinCurLCD          ; Se enciende el display y se oculta el cursor
    call   LCDComando         ; Llamada a la subrutina que manda un comando al LCD

    return                          ; Retorno de subrutina
;#####
; Subrutina que envia el comando de configuracion del LCD
;#####
; A esta subrutina se entra con el dato distintivo cargado en "W"

LCDComandoInicio:
    bcf     PuerCon,ComanCarac   ; Se procede a transferir un comando por el bus
    movwf  PuerDat             ; Se carga en "PuerDat" el BYTE asociado al comando
    goto   LCDHabilita         ; Se salta a habilitacion momentanea del LCD
;#####
; Subrutina que envia un comando al LCD
;#####
; A esta subrutina se entra con el dato distintivo cargado en "W"

LCDComando:
    movwf  DatoBus             ; Se almacena en "DatoBus" el BYTE del comando deseado
    bcf     PuerCon,ComanCarac   ; Se procede a transferir un comando por el bus
    movf   DatoBus,W           ; Se realiza el traspaso de "DatoBus" a "PuerDat"
    movwf  PuerDat             ; a traves del registro de trabajo (w)
    call   LCDChequea          ; Llamada a la subrutina liberalizadora del LCD
    call   LCDHabilita         ; Llamada a la subrutina de habilitacion del LCD
    swapf  DatoBus,W           ; Se realiza el traspaso de "DatoBus" conmutado
    movwf  PuerDat             ; a "PuerDat", a traves del registro de trabajo (w)
    goto   LCDHabilita         ; Se salta a habilitacion momentanea del LCD
;#####
; Subrutina que envia un caracter al LCD
;#####
; A esta subrutina se entra con el dato distintivo cargado en "W"

LCDCaracter:
    movwf  DatoBus             ; Se almacena en "DatoBus" el caracter deseado
    bcf     PuerCon,ComanCarac   ; Se procede a transferir un comando por el bus
    movf   DatoBus,W           ; Se realiza el traspaso de "DatoBus" a "PuerDat"
    movwf  PuerDat             ; a traves del registro de trabajo (w)
    call   LCDChequea          ; Llamada a la subrutina liberalizadora del LCD
    bsf     PuerCon,ComanCarac   ; Se procede a transferir un caracter por el bus

```

## MEDIDOR DE DISTANCIA POR ULTRASONIDOS

```

call    LCDHabilita      ; Llamada a la subrutina de habilitacion del LCD
swapf   DatoBus,W        ; Se realiza el traspaso de "DatoBus" conmutado
movwf   PuerDat          ; a "PuerDat", a traves del registro de trabajo (w)
goto    LCDHabilita      ; Se salta a habilitacion momentanea del LCD
;#####
; Subrutina de permanencia hasta que el LCD se liberalice
;#####

LCDChequea:
bsf     PuerCon,EscriLectu ; Se procede a realizar una lectura del LCD
bcf     STATUS,RP1        ; Seleccion del Banco 1
bsf     STATUS,RP0
movlw   11111111b        ; Se carga a traves de "W" la configuracion de todo entradas
movwf   PuerDat          ; para "PuerDat" en su registro de control asociado
bcf     STATUS,RP1        ; Seleccion del Banco 0
bcf     STATUS,RP0
bsf     PuerCon,InhibHabil ; Se habilita el LCD
nop     ; Operacion de espera de un ciclo de instruccion

Bucle
btfsc   PuerDat,7        ; Se comprueba el estado de liberalizacion del LCD
goto    Bucle           ; Se salta a la etiqueta indicada

bcf     PuerCon,InhibHabil ; Se deshabilita el LCD
bcf     STATUS,RP1        ; Seleccion del Banco 1
bsf     STATUS,RP0
movlw   00001111b        ; Se carga a traves de "W" la configuracion elegida para
movwf   PuerDat          ; "PuerDat" en su registro de control asociado
bcf     STATUS,RP1        ; Seleccion del Banco 0
bcf     STATUS,RP0

bcf     PuerCon,EscriLectu ; Se procede a realizar una escritura sobre LCD

return  ; Retorno de subrutina
;#####
; Subrutina que mantiene, durante un pulso, habilitado el LCD
;#####

LCDHabilita:
bsf     PuerCon,InhibHabil ; Se habilita el LCD
nop     ; Operacion de espera de un ciclo de instruccion
bcf     PuerCon,InhibHabil ; Se deshabilita el LCD

return  ; Retorno de subrutina
;#####
; Subrutinas para la visualizacion de mensajes y valores numericos sobre el LCD
;#####
; Subrutina de representacion de mensajes que utilizan las dos lineas del LCD
;#####
; Se parte de un dato seleccionador almacenado en "Mensaje"

MensajeCom:
clrwdt  ; Se inicializa el contador del Perro Guardian

        ; Se prepara el LCD, para una escritura sobre su pantalla
movlw   BorrarrLCD      ; Se borra la pantalla del LCD
call    LCDComando      ; Se envia dicho comando al LCD
movlw   LCDLinea1      ; Se coloca el cursor en la posicion inicial de la linea 1
call    LCDComando      ; Se envia dicho comando al LCD
clrf    Cuenta          ; Se inicializa el puntero de caracteres

        ; Se procede a la escritura de los caracteres en la 1ª linea

TomaCarac1
        ; Se identifica el mensaje definido por "Mensaje"
btfsc   Mensaje,0       ; Se comprueba la seleccion del mensaje 1
call    Men1Lin1        ; Llamada a la subrutina donde se encuentra el texto
btfsc   Mensaje,1       ; Se comprueba la seleccion del mensaje 2
call    Men2Lin1        ; Llamada a la subrutina donde se encuentra el texto
btfsc   Mensaje,2       ; Se comprueba la seleccion del mensaje 3
call    Men3Lin1        ; Llamada a la subrutina donde se encuentra el texto
btfsc   Mensaje,3       ; Se comprueba la seleccion del mensaje 4
call    Men4Lin1        ; Llamada a la subrutina donde se encuentra el texto
btfsc   Mensaje,4       ; Se comprueba la seleccion del mensaje 5
call    Men5Lin1        ; Llamada a la subrutina donde se encuentra el texto
btfsc   Mensaje,5       ; Se comprueba la seleccion del mensaje 6
call    Men6Lin1        ; Llamada a la subrutina donde se encuentra el texto
btfsc   Mensaje,6       ; Se comprueba la seleccion del mensaje 7

```

## APÉNDICE B

```

call    Men7Lin1      ; Llamada a la subrutina donde se encuentra el texto
btfsc  Mensaje,7     ; Se comprueba la seleccion del mensaje 8
call    Men8Lin1     ; Llamada a la subrutina donde se encuentra el texto

clrwdt                ; Se inicializa el contador del Perro Guardian
iorlw  0              ; Se comprueba la finalizacion de la linea seleccionada
btfsc  STATUS,Z      ; Z = 0 mientras quedan caracteres para finalizar la linea
goto   Linea2        ; Se salta a la escritura de la linea 2

call    LCDCharacter  ; Se envia dicho caracter al LCD
incf   Cuenta,F      ; Se apunta al siguiente caracter de la linea 1
goto   TomaCarac1   ; Se salta a la etiqueta indicada

Linea2
clrwdt                ; Se inicializa el contador del Perro Guardian
movlw  LCDLinea2     ; Se coloca el cursor en la posicion inicial de la linea 2
call   LCDComando    ; Se envia dicho comando al LCD
clrf   Cuenta        ; Se inicializa el puntero de caracteres

; Se procede a la escritura de los caracteres en la 2ª linea
TomaCarac2
; Se identifica el mensaje definido por "Mensaje"
btfsc  Mensaje,0     ; Se comprueba la seleccion del mensaje 1
call   Men1Lin2     ; Llamada a la subrutina donde se encuentra el texto
btfsc  Mensaje,1     ; Se comprueba la seleccion del mensaje 2
call   Men2Lin2     ; Llamada a la subrutina donde se encuentra el texto
btfsc  Mensaje,2     ; Se comprueba la seleccion del mensaje 3
call   Men3Lin2     ; Llamada a la subrutina donde se encuentra el texto
btfsc  Mensaje,3     ; Se comprueba la seleccion del mensaje 4
call   Men4Lin2     ; Llamada a la subrutina donde se encuentra el texto
btfsc  Mensaje,4     ; Se comprueba la seleccion del mensaje 5
call   Men5Lin2     ; Llamada a la subrutina donde se encuentra el texto
btfsc  Mensaje,5     ; Se comprueba la seleccion del mensaje 6
call   Men6Lin2     ; Llamada a la subrutina donde se encuentra el texto
btfsc  Mensaje,6     ; Se comprueba la seleccion del mensaje 7
call   Men7Lin2     ; Llamada a la subrutina donde se encuentra el texto
btfsc  Mensaje,7     ; Se comprueba la seleccion del mensaje 8
call   Men8Lin2     ; Llamada a la subrutina donde se encuentra el texto

clrwdt                ; Se inicializa el contador del Perro Guardian
iorlw  0              ; Se comprueba la finalizacion de la linea seleccionada
btfsc  STATUS,Z      ; Z = 0 mientras quedan caracteres para finalizar la linea

return                ; Retorna de la subrutina

call   LCDCharacter  ; Se envia dicho caracter al LCD
incf   Cuenta,F      ; Se apunta al siguiente caracter de la linea 1
goto   TomaCarac2   ; Se salta a la etiqueta indicada

;#####
; Subrutina de representacion de mensajes que utilizan una linea del LCD
;#####
; Se parte de un dato seleccionador almacenado en "Mensaje"

MensajeSem:
clrwdt                ; Se inicializa el contador del Perro Guardian

; Se prepara el LCD, para una escritura deseada sobre su pantalla
movlw  BorrarLCD     ; Se carga en "W" el comando que borra el LCD
btfss  BitsCont,BorSbesc ; Se comprueba el indicativo de antiborrado del LCD
call   LCDComando    ; Se envia dicho comando al LCD

movlw  LCDLinea1     ; Se coloca el cursor en la posicion inicial de la linea 1
btfsc  BitsCont,Seleclin ; Se comprueba indicativo seleccionador de linea
movlw  LCDLinea2     ; Se coloca el cursor en la posicion inicial de la linea 2
call   LCDComando    ; Se envia dicho comando al LCD
clrf   Cuenta        ; Se inicializa el puntero de caracteres
clrf   BitsCont      ; Se inicializa la variable que contiene los bits de control

; Se procede a la escritura de los caracteres en la linea

TomaCarac
; Se identifica el grupo al que pertenece el mensaje deseado de escribir
btfsc  Mensaje,7     ; Se comprueba la seleccion del grupo
goto   GrupoBSem    ; Se salta a la etiqueta indicada

; Se identifica el mensaje dentro del grupo "A"
btfsc  Mensaje,0     ; Se comprueba la seleccion del mensaje 1

```

## MEDIDOR DE DISTANCIA POR ULTRASONIDOS

```

call   SemiMen1   ; Llamada a la subrutina donde se encuentra el texto
btfsc  Mensaje,1 ; Se comprueba la seleccion del mensaje 2
call   SemiMen2   ; Llamada a la subrutina donde se encuentra el texto
btfsc  Mensaje,2 ; Se comprueba la seleccion del mensaje 3
call   SemiMen3   ; Llamada a la subrutina donde se encuentra el texto
btfsc  Mensaje,3 ; Se comprueba la seleccion del mensaje 4
call   SemiMen4   ; Llamada a la subrutina donde se encuentra el texto
btfsc  Mensaje,4 ; Se comprueba la seleccion del mensaje 5
call   SemiMen5   ; Llamada a la subrutina donde se encuentra el texto
btfsc  Mensaje,5 ; Se comprueba la seleccion del mensaje 6
call   SemiMen6   ; Llamada a la subrutina donde se encuentra el texto
btfsc  Mensaje,6 ; Se comprueba la seleccion del mensaje 7
call   SemiMen7   ; Llamada a la subrutina donde se encuentra el texto

goto   ComunSem  ; Se salta a la etiqueta indicada

```

### GrupoBSem

```

; Se identifica el mensaje dentro del grupo "B"
btfsc  Mensaje,0 ; Se comprueba la seleccion del mensaje 11
call   SemiMen11 ; Llamada a la subrutina donde se encuentra el texto
btfsc  Mensaje,1 ; Se comprueba la seleccion del mensaje 12
call   SemiMen12 ; Llamada a la subrutina donde se encuentra el texto
btfsc  Mensaje,2 ; Se comprueba la seleccion del mensaje 13
call   SemiMen13 ; Llamada a la subrutina donde se encuentra el texto
btfsc  Mensaje,3 ; Se comprueba la seleccion del mensaje 14
call   SemiMen14 ; Llamada a la subrutina donde se encuentra el texto
btfsc  Mensaje,4 ; Se comprueba la seleccion del mensaje 15
call   SemiMen15 ; Llamada a la subrutina donde se encuentra el texto
btfsc  Mensaje,5 ; Se comprueba la seleccion del mensaje 16
call   SemiMen16 ; Llamada a la subrutina donde se encuentra el texto
btfsc  Mensaje,6 ; Se comprueba la seleccion del mensaje 17
call   SemiMen17 ; Llamada a la subrutina donde se encuentra el texto

```

### ComunSem

```

clrwdt          ; Se inicializa el contador del Perro Guardian
iorlw  0        ; Se comprueba la finalizacion de la linea seleccionada
btfsc  STATUS,Z ; Z = 0 mientras quedan caracteres para finalizar la linea

return          ; Retorna de la subrutina

```

```

call   LCDCharacter ; Se envia dicho caracter al LCD
incf  Cuenta,F     ; Se apunta al siguiente caracter de la linea
goto   TomaCarac   ; Se salta a la etiqueta indicada
;#####
; Subrutina de representacion de un dato en base 10 con parte entera y parte decimal
;#####
; Se parte de los digitos a representar, almacenados en las variables "Digitos:...:Digitos+12"
; y se controla su representacion con las variables: NEntero, NDecimal, PotDiez

```

### BaseDiez:

```

; Se comprueba la posibilidad de aplicar un redondeo al dato
movf  NDecimal,W  ; Se guarda en "W" el numero de digitos decimales a visualizar
subwf PotDiez,W   ; Se realiza la resta (PotDiez - NDecimal)
btfsc STATUS,Z   ; Z = 0 mientras existe al menos un digito oculto
goto  VerDato     ; Se salta a la etiqueta indicada

```

```

; Se realiza la busqueda del primer digito oculto
sublw 13         ; Se guarda en "W" el orden del primer digito oculto
addlw Digitos   ; Se añade a "W" la posicion de memoria que ocupa "Digitos"
movwf FSR       ; Se apunta con "FSR" la direccion del digito buscado
movf  INDF,W    ; Se carga en "W" el digito oculto buscado

```

```

; Se determina el tipo de redondeo a realizar
sublw '4'       ; Se realiza la comprobacion definitoria de redondeo
btfsc STATUS,C ; C = 0 cuando se necesite un redondeo ascendente
goto  VerDato   ; Se salta a la etiqueta indicada

```

```

; Se modifican él, o los, digitos influenciados por el redondeo
movf  INDF,W    ; Se guarda en "W" el digito oculto buscado

```

### Modifica

```

movwf INDF      ; Se almacena en el digito apuntado el contenido de "W"
decf  FSR,F     ; Se apunta al siguiente digito a visualizar
incf  INDF,W    ; Se guarda en "W" el digito actualizado
sublw '9'       ; Se realiza la comprobacion para continuar con el redondeo
movlw '0'       ; Se guarda en "W" el dato correspondiente al '0' en ASCII
btfss STATUS,C ; C = 1 mientras el digito apuntado no supere al '8'
goto  Modifica  ; Se salta a la modificacion de mas digitos

```

## APÉNDICE B

```

incf   INDF,F           ; Se incrementa el ultimo digito afectado por el redondeo

VerDato
; Se calcula cual es la posicion sobre el LCD del primer digito a visualizar
movf   PotDiez,W        ; Se guarda en "W" el numero total de digitos decimales
sublw  13                ; Se realiza sobre "W" el calculo posicionador de la coma
movwf  Cuenta           ; Se guarda en "Cuenta" el numero total de digitos enteros
movf   NEntero,W        ; Se guarda en "W" el numero de digitos de la parte entera
subwf  Cuenta,F         ; Se guarda en "Cuenta" la resta (Cuenta - NEntero)
movlw  Digitos          ; Se guarda en "W" la posicion de memoria que ocupa "Digitos"
addwf  Cuenta,W         ; Se añade a "W" el n° de digitos enteros que no se visualizan
movwf  FSR              ; Se apunta con "FSR" la direccion del primer digito para ver
bcf    BitsCont,EnteDeci ; Se activa el indicativo de parte entera

; Se comprueba la existencia de parte entera
movf   NEntero,F        ; Se realiza el movimiento de "NEntero" sobre si mismo
btfsc  STATUS,Z         ; Z = 0 mientras existe parte entera para representar
goto   NoEntera        ; Se salta a la etiqueta indicada

; Se eliminan los ceros iniciales, con la justificacion a la izquierda
; Se sustituyen, por espacios, los ceros iniciales, con la justificacion a la derecha
DigNulo
movlw  '0'              ; Se guarda en "W" el dato correspondiente al '0' en ASCII
xorwf  INDF,W           ; Se compara el digito apuntado con el valor ASCII del '0'
btfss  STATUS,Z         ; Z = 1 cuando el digito apuntado sea el '0'
goto   PartEnte        ; Se salta a la etiqueta indicada

movlw  ' '              ; Se guarda en "W" el dato ASCII correspondiente al espacio
btfss  BitsCont,DereIzqu ; Se evalua el modo de justificacion elegido
call   LCDCharacter    ; Se envia dicho caracter al LCD
incf   FSR              ; Se apunta al digito inmediato inferior
decfsz NEntero         ; Se decrementa el n° de digitos enteros que quedan por ver
goto   DigNulo         ; Se salta a la etiqueta indicada

goto   NoEntera        ; Se salta a la etiqueta indicada

PartEnte
; Se envian los digitos restantes de la parte entera
movf   NEntero,W        ; Se realiza el traspaso de "NEntero" a "Cuenta"
movwf  Cuenta          ; a traves del registro de trabajo (w)

OtroDigito
movf   INDF,W          ; Se guarda en "W" el dato ASCII del digito deseado de ver
call   LCDCharacter    ; Se envia dicho caracter al LCD
incf   FSR              ; Se apunta al digito inmediato inferior
decfsz Cuenta         ; Se decrementa el numero de digitos que quedan por ver
goto   OtroDigito     ; Se salta a la etiqueta indicada

btfsc  BitsCont,EnteDeci ; Se comprueba la parte del dato ya representada

return                                ; Retorna de la subrutina

PartDeci
; Se comprueba la existencia de la parte decimal
movf   NDecimal,F      ; Se realiza el movimiento de "NDecimal" sobre si mismo
btfsc  STATUS,Z         ; Z = 0 mientras existe parte decimal para representar

return                                ; Retorna de la subrutina

; Se envian los digitos corresponcientes a la parte decimal
movlw  ','              ; Se guarda en "W" el dato ASCII correspondiente a la coma
call   LCDCharacter    ; Se envia dicho caracter al LCD
bsf    BitsCont,EnteDeci ; Se activa el indicativo de parte decimal
movf   NDecimal,W      ; Se realiza el traspaso de "NDecimal" a "Cuenta"
movwf  Cuenta          ; a traves del registro de trabajo (w)
goto   OtroDigito     ; Se salta a la etiqueta indicada

NoEntera
movlw  SltIzqLCD       ; Se carga en "W" el comando de salto a izquierdas del cursor
btfss  BitsCont,DereIzqu ; Se evalua el modo de justificacion elegida
call   LCDComando     ; Se envia dicho comando al LCD
movlw  '0'              ; Se guarda en "W" el dato correspondiente al '0' en ASCII
call   LCDCharacter    ; Se envia dicho caracter al LCD
goto   PartDeci       ; Se salta a la etiqueta indicada
;#####
; Subrutinas de retardo de tiempo para el correcto desarrollo del programa
;#####

```

## MEDIDOR DE DISTANCIA POR ULTRASONIDOS

```

#####
; Subrutina de espera de 0.1 ms, para la comprobacion de valided del ECO recibido
;#####
; La operacion matematica que define el tiempo de espera exacto correspondiente al siguiente
;codigo, es: tTotal = 2+2+3*Conta011-1+2 ==> Conta011 = (tTotal-5)/3
; Espera de 0.1 ms ==> 200 ciclos de instruccion
Espe01ms: ;2 ; Empleo de dos ciclos de instruccion en la llamada
    movlw 65 ;X
    movwf Conta011 ; Inicialización de la variable emplea dos ciclos
Cuenta011 ;Z;3*X-1; Bucle que emplea tres instrucciones
    decfsz Conta011,F ; tCuen = 3*Conta011-1
    goto Cuenta011 ; Cierre del bucle
    return ;2 ; Retorno de subrutina, emplea dos ciclos

; 200 = 2+2+3*65-1+2
;#####
; Subrutina de espera de 5 ms, para obtener con exito el reseteado del LCD
;#####
; La operacion matematica que define el tiempo de espera exacto correspondiente al siguiente
;codigo, es: tTotal = 2+3+(3*Conta5_2-1+6)*Conta5_1-1+2
; Espera de 5 ms ==> 10000 ciclos de instruccion
Espe5ms: ;2 ; Empleo de dos ciclos de instruccion en la llamada
    movlw 38 ;X
    movwf Conta51 ; Inicialización de la variable emplea dos ciclos
    nop ; Instruccion de ajuste del tiempo
Cuenta51 ;(Z+6)*X; Bucle que emplea (seis + bucle interior) ciclos
    movlw 86 ;Y
    movwf Conta52 ; Inicialización de la variable emplea dos ciclos
    nop ; Instruccion de ajuste del tiempo
Cuenta52 ;Z;3*Y-1; Bucle que emplea tres instrucciones
    decfsz Conta52,F ; tCuen2 = 3*Conta52-1
    goto Cuenta52 ; Cierre del bucle interior
    decfsz Conta51,F ; tCuen1 = (3*Conta52-1+6)*Conta51
    goto Cuenta51 ; Cierre del bucle exterior
    return ;2 ; Retorno de subrutina emplea dos ciclos

; 10000 = 2+3+(3*86-1+6)*38-1+2
;#####
; Subrutina de espera de 20 ms, con la que se anulan los rebotes iniciales en la pulsacion
;#####
; La operacion matematica que define el tiempo de espera exacto correspondiente al siguiente
;codigo; es: tTotal = 2+2+(3*Conta202-1+5)*Conta201-1+2
; Espera de 20 ms ==> 40000 ciclos de instruccion
Espe20ms: ;2 ; Empleo de dos ciclos de instruccion en la llamada
    movlw 95 ;X
    movwf Conta201 ; Inicialización de la variable emplea dos ciclos
Cuenta201 ;(Z+5)*X; Bucle que emplea (cinco + bucle interior) ciclos
    movlw 139 ;Y
    movwf Conta202 ; Inicialización de la variable emplea dos ciclos
Cuenta202 ;Z;3*Y-1; Bucle que emplea tres instrucciones
    decfsz Conta202,F ; tCuen2 = 3*Conta202-1
    goto Cuenta202 ; Cierre del bucle interior
    decfsz Conta201,F ; tCuen1 = (3*Conta202-1+5)*Conta201
    goto Cuenta201 ; Cierre del bucle exterior
    return ;2 ; Retorno de subrutina emplea dos ciclos

; 40000 = 2+2+(3*139-1+5)*95-1+2
;#####

```

B.III ARCHIVO CÁLCULO

```

; CALCULO.ASM
; Este archivo contiene dos subrutinas con las que se realiza un producto
;entre dos factores y una conversion de datos en binarios a codigo ASCII

; La limitacion que se tienen en estas subrutinas, es el tamaño de los datos
;de entrada.
; La subrutina del producto entrega un resultado maximo de 40 Bit's, cuyos
;factores pueden ser de 24 y 16 Bit's
; La subrutina de la conversion a codigo ASCII admite un dato de 40 Bit's
;cuyo maximo valor en decimal requiere el uso de 13 digitos

IniSubrCalculo equ $      ; Se guarda la direccion de memoria donde comenzara el codigo
; Variables locales
org IniVariables        ; Las nuevas variables se disponen en posiciones libres

; Variables asociadas a la subrutina MULT24x16
Producto    res    5      ; Variables donde se almacena el resultado del producto
Multiplicando res  5      ; Variables donde se almacenan los bit's del primer factor
Multiplicador res  2      ; Variables donde se almacenan los bit's del segundo factor

; Variables asociadas a la subrutina BIN2DEC
Cifras      res    1      ; Variable que guarda el n° de cifras restantes a determinar
Digitos     res   13      ; Variables donde se almacenan los digitos en codigo ASCII
Digito      res    1      ; Variable donde se guarda cada digito en formato decimal
Dato0       res    1      ; Variable que almacena el primer BYTE del dato en binario
Dato1       res    1      ; Variable que almacena el segundo BYTE del dato en binario
Dato2       res    1      ; Variable que almacena el tercer BYTE del dato en binario
Dato3       res    1      ; Variable que almacena el cuarto BYTE del dato en binario
Dato4       res    1      ; Variable que almacena el quinto BYTE del dato en binario

; Variable compartida por ambas subrutinas
NBit        res    1      ; Variable que guarda el n° de bit's restantes en la operacion

VARIABLE IniVariables = $      ; Se actualiza la posicion de comienzo
                                ;de las variables aun libres
org IniSubrCalculo            ; Comienza el codigo de la subrutina en la direccion guardada

;#####
; Subrutina de multiplicacion de dos numeros binarios de 24 y 16 bit's
;#####
; La operacion matematica que determina el producto de ambos datos es la siguiente
; Producto(40bits) = Multiplicando(24bits extendido a 40)*Multiplicador(16bits)

MULT24x16:
    clrf    Producto          ; Se inicializa el primer BYTE del resultado
    clrf    Producto+1        ; Se inicializa el segundo BYTE del resultado
    clrf    Producto+2        ; Se inicializa el tercer BYTE del resultado
    clrf    Producto+3        ; Se inicializa el cuarto BYTE del resultado
    clrf    Producto+4        ; Se inicializa el quinto BYTE del resultado

    movlw  16                  ; Se carga en "NBit" el numero de bits del
    movwf  NBit                ;multiplicador (Multiplicador+1:Multiplicador)

OtroBit
    rrf    Multiplicador+1,F    ; Se dispone sobre el bit de acarreo, el siguiente
    rrf    Multiplicador,F      ;bit menos significativo del multiplicador
    btfss STATUS,C             ; C = 1 mientras el bit multiplicador es '1'
    goto   FinSuma             ; Se salta a la etiqueta indicada

                                ; Se realizan las correspondientes sumas con los terminos de BYTES
                                ; Suma del primer BYTE
    movf   Multiplicando,W
    addwf  Producto,F          ; Se consigue la suma del 1° BYTE
    btfss STATUS,C             ; C = 1 cuando se desborde el resultado de la suma
    goto   Byte2               ; Se salta a realizar la suma del siguiente BYTE

    movlw  1
    addwf  Producto+1,F        ; Se suma el acarreo al siguiente BYTE
    btfss STATUS,C             ; C = 1 cuando se desborde el resultado de la suma
    goto   Byte2               ; Se salta a realizar la suma del siguiente BYTE

    movlw  1
    addwf  Producto+2,F        ; Se suma el acarreo al siguiente BYTE

```

## MEDIDOR DE DISTANCIA POR ULTRASONIDOS

```

btfss STATUS,C ; C = 1 cuando se desborde el resultado de la suma
goto Byte2 ; Se salta a realizar la suma del siguiente BYTE

movlw 1
addwf Producto+3,F ; Se suma el acarreo al siguiente BYTE
btfsc STATUS,C ; C = 0 mientras no se desborde el resultado de la suma
incf Producto+4,F ; Se incrementa el BYTE de mayor peso

Byte2 ; Suma del segundo BYTE
movf Multiplicando+1,W
addwf Producto+1,F ; Se consigue la suma del 2° BYTE
btfss STATUS,C ; C = 1 cuando se desborde el resultado de la suma
goto Byte3 ; Se salta a realizar la suma del siguiente BYTE

movlw 1
addwf Producto+2,F ; Se suma el acarreo al siguiente BYTE
btfss STATUS,C ; C = 1 cuando se desborde el resultado de la suma
goto Byte3 ; Se salta a realizar la suma del siguiente BYTE

movlw 1
addwf Producto+3,F ; Se suma el acarreo al siguiente BYTE
btfsc STATUS,C ; C = 0 mientras no se desborde el resultado de la suma
incf Producto+4,F ; Se incrementa el BYTE de mayor peso

Byte3 ; Suma del tercer BYTE
movf Multiplicando+2,W
addwf Producto+2,F ; Se consigue la suma del 3° BYTE
btfss STATUS,C ; C = 1 cuando se desborde el resultado de la suma
goto Byte4 ; Se salta a realizar la suma del siguiente BYTE

movlw 1
addwf Producto+3,F ; Se suma el acarreo al siguiente BYTE
btfsc STATUS,C ; C = 0 mientras no se desborde el resultado de la suma
incf Producto+4,F ; Se incrementa el BYTE de mayor peso

Byte4 ; Suma del cuarto BYTE
movf Multiplicando+3,W
addwf Producto+3,F ; Se consigue la suma del 4° BYTE
btfsc STATUS,C ; C = 0 mientras no se desborde el resultado de la suma
incf Producto+4,F ; Se incrementa el BYTE de mayor peso

; Suma del quinto BYTE
movf Multiplicando+4,W
addwf Producto+4,F ; Se consigue la suma del 5° BYTE

FinSuma ; Se multiplica por la base binaria (2) al multiplicando, antes de operar
; con el siguiente bit del multiplicador
bcf STATUS,C ; Se borra el bit de acarreo
rlf Multiplicando,F
rlf Multiplicando+1,F ; Se lleva a cabo esta multiplicacion por '2'
rlf Multiplicando+2,F ; con el desplazamiento a derechas de todos los BYTES
rlf Multiplicando+3,F ; del multiplicando
rlf Multiplicando+4,F ; Se usa el acarreo como bit de entrada en cada BYTE
decfsz NBit,F ; Se decreuenta el numero de bits restantes
goto OtroBit ; Se salta a la etiqueta indicada

return ; Retorna de la subrutina
; #####
; Subrutina de conversion de binario a codigo ASCII
; #####
; Se parte de un dato binario almacenado en "Dato4:Dato3:Dato2:Dato1:Dato0" y se consiguen
; los digitos decimales en codigo ASCII correspondientes al dato de entrada
; en los BYTES "Digitos:...:Digitos+12"

BIN2DEC:
movlw 13 ; Se carga en "Cifras" el n° de digitos necesarios
movwf Cifras ; para representar el dato de entrada en base decimal
movlw Digitos ; Se guarda en "W" la posicion de memoria que ocupa "Digitos"
addwf Cifras,W ; Se añade "Cifras" a la posicion de "Digitos"
movwf FSR ; Se apunta con "FSR" a la posicion donde se
decf FSR,F ; encuentra el digito de menor peso (Digitos+12)

OtroMas
call Div10 ; Llamada a la subrutina de obtencion de un digito
movf Digito,W ; Se guarda en "W" el digito decimal obtenido
andlw 00001111b ; Se enmascara la parte alta de "Digito"

```

## APÉNDICE B

```

addlw 00110000b ; Se convierte el digito decimal a codigo ASCII
movwf INDF ; Se almacena en "Digito+N" el correspondiente termino ASCII
decf FSR,F ; Se apunta al digito inmediato superior
decfsz Cifras,F ; Se decreuenta el numero de cifras restantes
goto OtroMas ; Se salta a la etiqueta indicada

return ; Retorna de la subrutina

Div10:
clrf Digito ; Se inicializa la variable "Digito"
movlw 40 ; Se carga en "NBit" el n° de bits de todos los
movwf NBit ;registros de entrada (DatoX:...:DatoZ)

NuevoBit
bcf STATUS,C ; Se borra el bit de acarreo
rlf Dato0,F
rlf Dato1,F ; Se realiza un desplazamiento a derechas del dato binario
rlf Dato2,F ;de entrada, introducciendo el "MSB" de cada BYTE en su
rlf Dato3,F ;inmediatosuperior por medio del bit de acarreo
rlf Dato4,F
rlf Digito,F ; Se introduce el "MSB" del dato resultante en "Digito"
movlw 10 ; Se guarda en "W" el dato '10'
subwf Digito,W ; Se realiza la resta (Digito - 10)
btfss STATUS,C ; C = 1 cuando "Digito" supere al '9'
goto Menor10 ; Se salta a la etiqueta indicada
movwf Digito ; Se guarda en "Digito" el valor de las unidades resultantes
incf Dato0,F ; Se guarda en "Dato0" el valor de las decenas

Menor10
decfsz NBit,F ; Se decreuenta el numero de bits restantes
goto NuevoBit ; Se salta a la etiqueta indicada

return ; Retorna de la subrutina
;#####

```

B.IV ARCHIVO LINELCD

```

; LINELCD.ASM
; Este archivo contiene las subrutinas con las que se realiza la escritura
; completa de las lineas del LCD

; No es necesaria la definicion de variables locales

;#####
; Subrutinas para la visualizacion de mensajes y valores numericos (definidos) sobre el LCD
;#####
;#####
; Subrutina de visualizacion del estado de ocupacion de la memoria, con su mensaje de texto
;#####

LineaOcuMem: ; Subrutina de visualizacion de la linea completa de ocupacion de memoria
; Se visualiza el mensaje de texto de ocupacion de memoria
movlw 10000010b
movwf Mensaje ; Se carga en "Mensaje" el BYTE asociado al mensaje 12
call MensajeSem ; Llamada a la subrutina que visualiza un mensaje de 1 linea

EstadoMem: ; Subrutina de visualizacion de la relacion "PosiFin/PosiLimite"
; Se visualizan los valores numericos de "PosiFin" y "PosiLimite"
movlw LCDLineal+10 ; Se coloca el cursor en la posicion deseada de la linea 1
call LCDComando ; Se envia dicho comando al LCD
movf PosiFin,W ; Se guarda en "W" el n° de posiciones de memoria ocupadas
call Dato2Cifras ; Llamada a la subrutina que visualiza datos de 2 cifras
movlw '/' ; Se guarda en "W" el valor correspondiente al caracter "/"
call LCDCaracter ; Se envia dicho caracter al LCD
movlw PosiLimite ; Se guarda en "W" el n° total de posiciones de la memoria
call Dato2Cifras ; Llamada a la subrutina que visualiza datos de 2 cifras
movlw ' ' ; Se guarda en "W" el valor correspondiente al caracter " "
call LCDCaracter ; Se envia dicho caracter al LCD

return ; Retorna de la subrutina
;#####
; Subrutina de visualizacion de los parametros "Hum" y "Temp", con su mensaje de texto
;#####

LineaHT: ; Subrutina de visualizacion de la linea completa de factores ambientales
; Se visualiza el mensaje de texto de factores ambientales
movlw 00000010b
movwf Mensaje ; Se carga en "Mensaje" el BYTE asociado al mensaje 02
call MensajeSem ; Llamada a la subrutina que visualiza un mensaje de 1 linea

HumeTempe: ; Subrutina de visualizacion de la Humedad y de la Temperatura
; Se convierte la medida del sensor al valor numerico de la humedad
call FactorHum ; Llamada a subrutina que define el factor humedad
call OperaHume ; Llamada a la subrutina que calcula la humedad en binario
call BIN2DEC ; Llamada a la subrutina que realiza el paso a decimal

; Se define el formato del dato para su correcta visulaizacion
movlw 2
movwf NEntero ; Se carga en "NEntero" el n° de digitos enteros a ver
movlw 0
movwf NDecimal ; Se carga en "NDecimal" el n° de digitos decimales a ver
movlw 5
movwf PotDiez ; Se carga en "PotDiez" el n° de decimales asignados al dato
bcf BitsCont,DereIzqu ; Se activa el marcador de justificacion a la derecha

; Se realiza la visualizacion de la humedad
movlw LCDLineal+5 ; Se coloca el cursor en la posicion deseada de la linea 1
call LCDComando ; Se envia dicho comando al LCD
call BaseDiez ; Llamada a subrutina que representa los digitos elegidos

; Se convierte la medida del sensor al valor numerico de la temperatura
call OperaTemp ; Lamada a la subrutina que calcula la temperatura en binario
call BIN2DEC ; Llamada a la subrutina que realiza el paso a decimal

; Se define el formato del dato para su correcta visulaizacion
movlw 2
movwf NEntero ; Se carga en "NEntero" el n° de digitos enteros a ver
movlw 1
movwf NDecimal ; Se carga en "NDecimal" el n° de digitos decimales a ver
movlw 2

```

## APÉNDICE B

```

movwf PotDiez      ; Se carga en "PotDiez" el n° de decimales asignados al dato
bcf BitsCont,DereIzqu ; Se activa el marcador de justificacion a la derecha

; Se realiza la visualizacion de la temperatura
movlw LCDLineal+10 ; Se coloca el cursor en la posicion deseada de la linea 1
call LCDComando    ; Se envia dicho comando al LCD
call BaseDiez      ; Llamada a subrutina que representa los digitos elegidos
movlw 223          ; Se guarda en "W" el valor correspondiente al caracter ""
call LCDCaracter   ; Se envia dicho caracter al LCD

return            ; Retorna de la subrutina
;#####
; Subrutina de visualizacion de la posicion y la medida, con su mensaje de texto
;#####
LineaPosMed:     ; Subrutina de visualizacion de la linea completa de Posicion y Medida
                ; Se visualiza el mensaje de texto de medida de distancia
                ; Se inicializa el contador del Perro Guardian
clrwtd          ; Se inicializa el contador del Perro Guardian
bsf BitsCont,Seleclin ; Se activa el indicativo de segunda linea del LCD
movlw 00010000b
movwf Mensaje    ; Se carga en "Mensaje" el BYTE asociado al mensaje 05
call MensajeSem  ; Llamada a la subrutina que visualiza un mensaje de 1 linea

PosiMed:        ; Subrutina de visualizacion de la posicion y la medida resultante
                ; se visualiza la posicion de la memoria apuntada en ese momento
                ; Se inicializa el contador del Perro Guardian
clrwtd          ; Se inicializa el contador del Perro Guardian
movlw LCDLinea2+6 ; Se coloca el cursor en la posicion deseada de la linea 2
call LCDComando ; Se envia dicho comando al LCD
movf Posicion,W ; Se guarda en "W" el n° de la posicion apuntada
call Dato2Cifras ; Llamada a la subrutina que visualiza datos de 2 cifras

                ; Se busca la etapa que ha solicitado esta subrutina
btfss Indicadores,Medir ; Se comprueba el indicativo de realizar medida
goto LeerVerDato ; Se salta a la etiqueta indicada

CaptDistan:   ; Subrutina que realiza una medida de distancia y la representa en el LCD
                ; Se realiza el proceso de captura de la distancia
call EnviaSeñal ; Llamada a la subrutina relacionada con las ondas acusticas

                ; Se realizan las operaciones del motivo causante del retorno de "EnviaSeñal"
btfsc Indicadores,Interrup ; Se comprueba el indicativo de secuencia interrumpida

return        ; Retorna de la subrutina

btfsc Indicadores,TmpDesbor ; Se comprueba el indicativo de "TMR1" desbordado
goto Desbordado ; Se salta a la etiqueta indicada

                ; Se realizan las operaciones que consiguen la medida
bcf Indicadores,ADormir ; Se desactiva el indicativo de convertir AD durmiendo
call MedTempera ; Llamada a la subrutina que mide la temperatura con el "CAD"
call FactorVel ; Llamada a la subrutina que define el factor de la velocidad
call CorrecTiemp ; Llamada a la subrutina que realiza la correccion del tiempo
call OperaDist ; Llamada a la subrutina que calcula la distancia en binario
call BIN2DEC ; Llamada a la subrutina que realiza el paso a decimal

                ; Se evalua la necesidad de realizar el redondeo sobre la distancia
movf Digitos+5,W ; Se guarda en "W" el primer dígito oculto
sublw '4' ; Se realiza la resta (4 - Digitos+5)
btfsc STATUS,C ; C = 0 cuando sea necesario aplicar un redondeo ascendente
goto SacarDigitos ; Se salta a la etiqueta indicada

                ; Se realiza el redondeo ascendente modificando los digitos necesarios
movlw 5
addlw Digitos ; Se guarda en "W" la posicion del primer dígito oculto
movwf FSR ; Se apunta a la posicion del primer dígito oculto

RedonAscen
movwf INDF ; Se carga el '0' sobre el dígito correspondiente
decf FSR,F ; Se apunta al dígito inmediato superior
incf INDF,W ; Se guarda en "W" el dígito a visualizar ya actualizado
sublw '9' ; Se realiza la resta ( DígitoN - '9')
movlw '0' ; Se guarda en "W" el valor '0' en código ASCII
btfss STATUS,C ; C = 1 mientras no exista la necesidad de seguir en el bucle
goto RedonAscen ; Se salta a la etiqueta indicada

incf INDF,F ; Se incrementa el ultimo dígito por modificar

```

## MEDIDOR DE DISTANCIA POR ULTRASONIDOS

```

SacarDigitos
    ; Se realiza la extraccion de los digitos validos que representan la distancia
    movlw '1' ; Se guarda en "W" el valor ASCII correspondiente a "1"
    subwf Digitos+1,W ; Se realiza la resta (Digitos+1 - '1')
    movf Digitos+1,W ; Se guarda en "W" el dato de "Digito+1"
    btfs STATUS,C ; C = 1 mientras "Digitos+1" supera al '0'
    movlw ' ' ; Se guarda en "W" el valor correspondiente al caracter " "
    movwf ByteH ; Se guarda en "ByteH" el valor actual contenido en "W"
    movf Digitos+2,W ; Se realiza el traspaso de "Digitos+2" a "ByteM"
    movwf ByteM ; a traves del registro de trabajo (w)
    movf Digitos+3,W ; Se realiza el traspaso de "Digitos+3" a "ByteN"
    movwf ByteN ; a traves del registro de trabajo (w)
    movf Digitos+4,W ; Se realiza el traspaso de "Digitos+4" a "ByteL"
    movwf ByteL ; a traves del registro de trabajo (w)

VerMedida
    ; Se realiza la visualización de la distancia
    movlw LCDLinea2+10 ; Se coloca el cursor en la posicion deseada de la linea 2
    call LCDCComando ; Se envia dicho comando al LCD
    movf ByteH,W ; Se guarda en "W" el valor correspondiente a los decametros
    call LCDCCaracter ; Se envia dicho caracter al LCD
    movf ByteM,W ; Se guarda en "W" el valor correspondiente a los metros
    call LCDCCaracter ; Se envia dicho caracter al LCD
    movlw ',' ; Se guarda en "W" el valor correspondiente al caracter ","
    call LCDCCaracter ; Se envia dicho caracter al LCD
    movf ByteN,W ; Se guarda en "W" el valor correspondiente a los decimetros
    call LCDCCaracter ; Se envia dicho caracter al LCD
    movf ByteL,W ; Se guarda en "W" el valor correspondiente a los centimetros
    call LCDCCaracter ; Se envia dicho caracter al LCD

    return ; Retorna de la subrutina

Desbordado
    ; Se representa en pantalla la medida '##,##'
    bcf Indicadores,TmpDesbor ; Se desactiva el indicativo de "TMR1" desbordado
    movlw LCDLinea2+10 ; Se coloca el cursor en la posicion deseada de la linea 2
    call LCDCComando ; Se envia dicho comando al LCD
    movlw 35 ; Se guarda en "W" el valor correspondiente al caracter "#"
    call LCDCCaracter ; Se envia dicho caracter al LCD
    movlw 35 ; Se guarda en "W" el valor correspondiente al caracter "#"
    call LCDCCaracter ; Se envia dicho caracter al LCD
    movlw ',' ; Se guarda en "W" el valor correspondiente al caracter ","
    call LCDCCaracter ; Se envia dicho caracter al LCD
    movlw 35 ; Se guarda en "W" el valor correspondiente al caracter "#"
    call LCDCCaracter ; Se envia dicho caracter al LCD
    movlw 35 ; Se guarda en "W" el valor correspondiente al caracter "#"
    call LCDCCaracter ; Se envia dicho caracter al LCD

    return ; Retorna de la subrutina
;#####
; Subrutina de visualizacion de la linea 1, correspondiente a MEDIR
;#####

LineaSalvar: ; Subrutina de visualizacion de la linea completa de Salvar
;Se visualiza el mensaje de texto de esta linea
    bsf BitsCont,BorSbesc ; Se activa el indicativo de sobreescritura en el LCD
    movlw 00001000b
    movwf Mensaje ; Se carga en "Mensaje" el BYTE asociado al mensaje 04
    call MensajeSem ; Llamada a la subrutina que visualiza un mensaje de 1 linea

    ; Se visualizan los simbolos de esta linea
    movlw LCDLinea1+6 ; Se coloca el cursor en la posicion deseada de la linea 1
    call LCDCComando ; Se envia dicho comando al LCD
    movlw 126 ; Se guarda en "W" el valor correspondiente al caracter "->"
    call LCDCCaracter ; Se envia dicho caracter al LCD
    movlw 219 ; Se guarda en "W" el valor correspondiente al caracter "?"
    btfs Indicadores,MemLlena ; Se comprueba el estado de ocupacion de la "EEPROM"
    movlw 255 ; Se guarda en "W" el valor correspondiente al caracter "?"
    call LCDCCaracter ; Se envia dicho caracter al LCD
    goto EstadoMem ; Se salta a la etiqueta indicada
;#####
; Subrutina de visualizacion de la linea 1, correspondiente a Memorizar
;#####

LineaGuardada: ; Subrutina de visualizacion de la linea completa de Guardada
; Se visualiza el mensaje de texto de esta linea
    bsf BitsCont,BorSbesc ; Se activa el indicativo de sobreescritura en el LCD

```

## APÉNDICE B

```

movlw 10001000b
movwf Mensaje ; Se carga en "Mensaje" el BYTE asociado al mensaje 14
call MensajeSem ; Llamada a la subrutina que visualiza un mensaje de 1 linea

; Se visualizan los simbolos de esta linea
movlw LCDLineal+10 ; Se coloca el cursor en la posicion deseada de la linea 1
call LCDComando ; Se envia dicho comando al LCD
movlw 219 ; Se guarda en "W" el valor correspondiente al caracter "?"
btfss Indicadores,Añadida ; Se comprueba el estado de la posicion apuntada
movlw 255 ; Se guarda en "W" el valor correspondiente al caracter "?"
call LCDCaracter ; Se envia dicho caracter al LCD
bcf Indicadores,Añadida ; Se desactiva el indicativo una memorizacion añadida
movlw ' ' ; Se guarda en "W" el valor correspondiente al caracter " "
call LCDCaracter ; Se envia dicho caracter al LCD
movlw 126 ; Se guarda en "W" el valor correspondiente al caracter "->"
call LCDCaracter ; Se envia dicho caracter al LCD
movlw ' ' ; Se guarda en "W" el valor correspondiente al caracter " "
call LCDCaracter ; Se envia dicho caracter al LCD
movlw 255 ; Se guarda en "W" el valor correspondiente al caracter "?"
call LCDCaracter ; Se envia dicho caracter al LCD

return ; Retorna de la subrutina
;#####
; Subrutina de visualizacion de un numero entero menor de dos cifras decimales
;#####
; Se parte de un dato binario almacenado en "W" y el cursor del LCD en la posicion de comienzo

Dato2Cifras:
; Se prepara el dato de entrada para pasarlo a dos digitos decimales
movwf Dato0 ; Se carga en "Dato0" el dato deseado de visualizar
clrf Dato1 ; Se borra el segundo BYTE
clrf Dato2 ; Se borra el tercer BYTE
clrf Dato3 ; Se borra el cuarto BYTE
clrf Dato4 ; Se borra el quinto BYTE
call BIN2DEC ; Llamada a la subrutina que realiza el paso a decimal
; Se define el formato del dato para su correcta visulaizacion
movlw 2
movwf NEntero ; Se carga en "NEntero" el n° de digitos enteros a ver
movlw 0
movwf NDecimal ; Se carga en "NDecimal" el n° de digitos decimales a ver
movlw 0
movwf PotDiez ; Se carga en "PotDiez" el n° de decimales asignados al dato
bcf BitsCont,DereIzqu ; Se activa el marcador de justificacion a la derecha
call BaseDiez ; Llamada a subrutina que representa los digitos elegidos

return ; Retorna de la subrutina
;#####

```

B.V ARCHIVO ACCESMEM

```

; ACCESMEM.ASM
; Este archivo contiene todas las subrutinas necesarias para la escritura
; y lectura de datos de distintos tamaños relacionados con la memoria EEPROM
; y las subrutinas relacionadas con la lectura de tablas que se encuentran
; en la memoria de programa

IniSubrAccesMem      equ $          ; Se guarda la direccion de memoria donde comenzara el
codigo

; Variables locales
org IniVariables     ; Las nuevas variables se disponen en posiciones libres

PosiLimite           equ    60      ; Constante indicativa de la capacidad de la memoria de datos

; Variables asociadas a las subrutinas relacionadas con la memoria EEPROM
UltiEscrita          res    1       ; Variable que almacena la ultima posicion escrita en memoria
PosiFin              res    1       ; Variable que almacena la ultima posicion ocupada en memoria
Posicion             res    1       ; Variable que guarda la posicion a acceder de la "EEPROM"
PosicionTem          res    1       ; Variable que guarda temporalmente la posicion de datos
ByteMem              res    1       ; Variable que guarda el BYTE deseado escribir en la "EEPROM"
ByteL                res    1       ; Variable que almacena el digito en ASCII de los centimetros
ByteN                res    1       ; Variable que almacena el digito en ASCII de los decimetros
ByteM                res    1       ; Variable que almacena el digito en ASCII de los metros
ByteH                res    1       ; Variable que almacena el digito en ASCII de los decametros

; Variables asociadas a las subrutinas relacionadas con las tablas de la memoria
FacVelA              res    1       ; Variable que almacena la parte alta del factor de velocidad
FacVelB              res    1       ; Variable que almacena la parte baja del factor de velocidad
FacHumA              res    1       ; Variable que almacena la parte alta del factor de humedad
FacHumB              res    1       ; Variable que almacena la parte baja del factor de humedad

VARIABLE IniVariables = $          ; Se actualiza la posicion de comienzo
; de las variables aun libres
org IniSubrAccesMem  ; Comienza el codigo de la subrutina en la direccion guardada

; #####
; Subrutina de escritura de una DoblePalabra (32 bit) en la memoria EEPROM
; #####
; Se parte de que los 4 BYTES se encuentra en "ByteH:ByteM:ByteN:ByteL,
; y que "Posicion" direcciona el dato completo

EscriDato:

btfsc INTCON,GIE      ; Se comprueba la habilitacion de las interrupciones
bsf   Indicadores,Interrup ; Se activa el indicativo de interrupciones en uso
bcf   STATUS,C        ; Se pone a cero el bit de acarreo
rlf   Posicion         ; Se realiza el producto (*2) a la variable Posicion
bcf   STATUS,C        ; Se pone a cero el bit de acarreo
rlf   Posicion         ; Se realiza el producto (*2) a la variable Posicion

; Con esto la variable Posicion puede direccionar la "EEPROM" al completo
movf  ByteL,W         ; Se realiza el traspaso de "ByteL" a "ByteMem"
movwf ByteMem         ; a traves del registro de trabajo (w)
call  EscriBYTE       ; Se realiza la llamada a la subrutina que escribe un BYTE
; en la direccion indicada por "Posicion"
; Queda guardado el BYTE de "ByteL"

incf  Posicion        ; Se direcciona el siguiente BYTE
movf  ByteN,W         ; Se realiza el traspaso de "ByteN" a "ByteMem"
movwf ByteMem         ; a traves del registro de trabajo (w)
call  EscriBYTE       ; Se realiza la llamada a la subrutina que escribe un BYTE
; en la direccion indicada por "Posicion"
; Queda guardado el BYTE de "ByteN"

incf  Posicion        ; Se direcciona el siguiente BYTE
movf  ByteM,W         ; Se realiza el traspaso de "ByteM" a "ByteMem"
movwf ByteMem         ; a traves del registro de trabajo (w)
call  EscriBYTE       ; Se realiza la llamada a la subrutina que escribe un BYTE
; en la direccion indicada por "Posicion"
; Queda guardado el BYTE de "ByteM"

incf  Posicion        ; Se direcciona el siguiente BYTE
movf  ByteH,W         ; Se realiza el traspaso de "ByteH" a "ByteMem"
movwf ByteMem         ; a traves del registro de trabajo (w)

```

## APÉNDICE B

```

call  EscriBYTE      ; Se realiza la llamada a la subrutina que escribe un BYTE
                        ;en la direccion indicada por "Posicion"
                        ; Queda guardado el BYTE de "ByteH"

bcf   STATUS,C       ; Se pone a cero el bit de acarreo
rrf   Posicion       ; Se realiza la division (/2) a la variable Posicion
bcf   STATUS,C       ; Se pone a cero el bit de acarreo
rrf   Posicion       ; Se realiza la division (/2) a la variable Posicion
                        ; Con esto la variable Posicion vuelve a direccionar el dato completo

movf  Posicion,W     ; Se realiza el traspaso de "Posicion" a "UltiEscrita"
movwf UltiEscrita    ;a traves del registro de trabajo (w)
subwf PosiFin,W      ; Se realiza la resta (PosiFin - Posicion)
btfss STATUS,C      ; C = 1 mientras no se añada una nueva posicion
bsf   Indicadores,Añadida ; Se indica sobre "Indicadores", que se a ocupado una
                        ;posicion vacante

return                ; Retorna de la subrutina

;#####
; Subrutina de visualizacion de una DoblePalablr (32 bit) leida de la memoria EEPROM
;#####
                        ; Se parte de que en "Posicion" se encuentra la direccion del dato completo,
                        ;y que los 4 Digos se guardaran en "Dato4:Dato3:Dato2:Dato1:Dato0"

LeerVerDato:

bcf   STATUS,C       ; Se pone a cero el bit de acarreo
rlf   Posicion       ; Se realiza el producto (*2) a la variable Posicion
bcf   STATUS,C       ; Se pone a cero el bit de acarreo
rlf   Posicion       ; Se realiza el producto (*2) a la variable Posicion
                        ; Con esto la variable Posicion puede direccionar la "EEPROM" al completo

call  LeeBYTE        ; Se realiza la llamada a la subrutina que lee el BYTE
                        ;de la direccion indicada por "Posicion"
movf  ByteMem,W      ; Se realiza el traspaso de "ByteMem" a "ByteL"
movwf ByteL          ;a traves del registro de trabajo (w)
                        ; Ya esta cargado el digito de centrimetros en "ByteL"

incf  Posicion       ; Se direcciona el siguiente BYTE
call  LeeBYTE        ; Se realiza la llamada a la subrutina que lee el BYTE
                        ;de la direccion indicada por "Posicion"
movf  ByteMem,W      ; Se realiza el traspaso de "ByteMem" a "ByteN"
movwf ByteN          ;a traves del registro de trabajo (w)
                        ; Ya esta cargado el digito de decimetros en "ByteN"

incf  Posicion       ; Se direcciona el siguiente BYTE
call  LeeBYTE        ; Se realiza la llamada a la subrutina que lee el BYTE
                        ;de la direccion indicada por "Posicion"
movf  ByteMem,W      ; Se realiza el traspaso de "ByteMem" a "ByteM"
movwf ByteM          ;a traves del registro de trabajo (w)
                        ; Ya esta cargado el digito de metros en "ByteM"

incf  Posicion       ; Se direcciona el siguiente BYTE
call  LeeBYTE        ; Se realiza la llamada a la subrutina que lee el BYTE
                        ;de la direccion indicada por "Posicion"
movf  ByteMem,W      ; Se realiza el traspaso de "ByteMem" a "ByteH"
movwf ByteH          ;a traves del registro de trabajo (w)
                        ; Ya esta cargado el digito de decametros en "ByteH"

bcf   STATUS,C       ; Se pone a cero el bit de acarreo
rrf   Posicion       ; Se realiza la division (/2) a la variable Posicion
bcf   STATUS,C       ; Se pone a cero el bit de acarreo
rrf   Posicion       ; Se realiza la division (/2) a la variable Posicion
                        ; Con esto la variable Posicion vuelve a direccionar el dato completo

                        ; Se realiza la visualizacion del dato leido

goto  VerMedida

;#####
; Subrutina de escritura de "PosiFin" en su posicion de la memoria EEPROM
;#####

GuardPosFin:

btfsc INTCON,GIE    ; Se comprueba la habilitacion de las interrupciones

```

## MEDIDOR DE DISTANCIA POR ULTRASONIDOS

```

bsf     Indicadores,Interrup ; Se activa el indicativo de interrupciones en uso

movf    Posicion,W           ; Se realiza el traspaso de "Posicion" a "PosicionTem" con "w"
movwf   PosicionTem         ;durante la actualizacion de "PosiFin"
clrf    Posicion            ; Se apunta al primer BYTE de la "EEPROM"
movf    PosiFin,W           ; Se realiza el traspaso de "PosiFin" a "ByteMem"
movwf   ByteMem             ;a traves del registro de trabajo (w)

call    EscriBYTE           ; Se realiza la llamada a la subrutina que escribe un BYTE
                                ;en la direccion indicada por "Posicion"

movf    PosicionTem,W       ; Se realiza el traspaso de "PosicionTem" a "Posicion" con "W"
movwf   Posicion            ;tras acabar la escritura en la "EEPROM"

return                                     ; Retorna de la subrutina

```

```

;#####
; Subrutina de escritura de "CorrecDista" en su posicion de la memoria EEPROM
;#####

```

GuardCorrecDista:

```

btfsc   INTCON,GIE         ; Se comprueba la habilitacion de las interrupciones
bsf     Indicadores,Interrup ; Se activa el indicativo de interrupciones en uso

movf    Posicion,W         ; Se realiza el traspaso de "Posicion" a "PosicionTem" con "w"
movwf   PosicionTem        ;durante la actualizacion de "CorrecDista"
movlw   1
movwf   Posicion          ; Se apunta al segundo BYTE de la "EEPROM"
movf    CorrecDista,W      ; Se realiza el traspaso de "CorrecDista" a "ByteMem"
movwf   ByteMem           ;a traves del registro de trabajo (w)

call    EscriBYTE         ; Se realiza la llamada a la subrutina que escribe un BYTE
                                ;en la direccion indicada por "Posicion"

movf    PosicionTem,W      ; Se realiza el traspaso de "PosicionTem" a "Posicion" con "W"
movwf   Posicion          ;tras acabar la escritura en la "EEPROM"

return                                     ; Retorna de la subrutina

```

```

;#####
; Subrutina de escritura de un BYTE en la memoria EEPROM
;#####

```

EscriBYTE:

```

bsf     STATUS,RP1         ; Seleccion del Banco 3
bsf     STATUS,RP0

```

EstaEscrib

```

btfsc   EECON1,WR         ; Se comprueba la actividad de escritura en memoria
goto    EstaEscrib        ; Permanencia en el bucle mientras se este escribiendo en
                                ;memoria

bcf     STATUS,RP1         ; Seleccion del Banco 0
bcf     STATUS,RP0
movf    Posicion,W         ; Se guarda la posicion de la memoria en el registro de
                                ;trabajo

bsf     STATUS,RP1         ; Seleccion del Banco 2
bcf     STATUS,RP0

movwf   EEADR              ; Tras el cambio de banco se actualiza la direccion

bcf     STATUS,RP1         ; Seleccion del Banco 0
bcf     STATUS,RP0

movf    ByteMem,W          ; Se guarda el BYTE a escribir en el registro de trabajo

bsf     STATUS,RP1         ; Seleccion del Banco 2
bcf     STATUS,RP0

movwf   EEDATA            ; Tras el cambio de banco se actualiza el BYTE a escribir

bsf     STATUS,RP1         ; Seleccion del Banco 3
bsf     STATUS,RP0

bcf     EECON1,EEPGD       ; Se selecciona el acceso a la "EEPROM"

```

## APÉNDICE B

```

bsf    EECON1,WREN          ; Se habilita la escritura en memoria

bcf    STATUS,RP1          ; Seleccion del Banco 0
bcf    STATUS,RP0

btfsc  Indicadores,Interrup ; Se comprueba si se hace uso de las interrupciones,
bcf    INTCON,GIE          ;deshabilitandosen de forma momentanea

bsf    STATUS,RP1          ; Seleccion del Banco 3
bsf    STATUS,RP0

movlw  0x55                 ; Se carga el dato 55(hex) en "EECON2"
movwf  EECON2               ;a traves del registro de trabajo (w)
movlw  0xAA                 ; Se carga el dato AA(hex) en "EECON2"
movwf  EECON2               ;a traves del registro de trabajo (w)
bsf    EECON1,WR           ; Se da comienzo al proceso de escritura en memoria

bcf    STATUS,RP1          ; Seleccion del Banco 0
bcf    STATUS,RP0

btfsc  Indicadores,Interrup ; Se comprueba si se hace uso de las interrupciones,
bsf    INTCON,GIE          ;para habilitaralasd de nuevo

bsf    STATUS,RP1          ; Seleccion del Banco 3
bsf    STATUS,RP0

bcf    EECON1,WREN          ; Se deshabilita la escritura en memoria

bcf    STATUS,RP1          ; Seleccion del Banco 0
bcf    STATUS,RP0

return                          ; Retorna de la subrutina

;#####
; Subrutina de lectura de un BYTE de la memoria EEPROM
;#####

LeeBYTE:

    movf  Posicion,W          ; Se guarda la posicion de la memoria en el registro de
                                ;trabajo

    bsf   STATUS,RP1          ; Seleccion del Banco 2
    bcf   STATUS,RP0

    movwf EEADR               ; Tras el cambio de banco se actualiza la direccion

    bsf   STATUS,RP1          ; Seleccion del Banco 3
    bsf   STATUS,RP0

    bcf   EECON1,EEPGD        ; Se selecciona el acceso a la "EEPROM"
    bsf   EECON1,RD           ; Se comienza el proceso de lectura de la memoria

    bsf   STATUS,RP1          ; Seleccion del Banco 2
    bcf   STATUS,RP0

    movf  EEDATA,W           ; Se guarda el BYTE leído en el registro de trabajo

    bcf   STATUS,RP1          ; Seleccion del Banco 0
    bcf   STATUS,RP0

    movwf ByteMem            ; Tras el cambio de banco se carga el BYTE leído en
                                ;"ByteMem"

    return                    ; Retorna de la subrutina

;#####
;#####
; Subrutina de busqueda de factores correctores dependientes de la Ta
;#####

                                ; Se procede a la busqueda del factor corrector de velocidad
FactorVel:
    movlw 00001000b          ; Se inidica el inicio de la tabla en al posicion 0x0800
                                ; Esta tabla corresponde a la parte alta del factor de velocidad

```

## MEDIDOR DE DISTANCIA POR ULTRASONIDOS

```

call    LecturMem          ; Llamada a subrutina de lectura sobre la memoria de
                        ; programa
                        ; El dato buscado se encuentra almacenado en el registro de trabajo (w)
movwf   FacVelA           ; Se guarda la parte alta del factor de velocidad en la
                        ; variable "FacVelA"
movlw   00001100b        ; Se indica el inicio de la tabla en la posición 0x0C00
                        ; Esta tabla corresponde a la parte baja del factor de velocidad

call    LecturMem          ; Llamada a subrutina de lectura sobre la memoria de
                        ; programa
                        ; El dato buscado se encuentra almacenado en el registro de trabajo (w)
movwf   FacVelB           ; Se guarda la parte baja del factor de velocidad en la
                        ; variable "FacVelB"
return  ; Retorna de subrutina

; Se procede a la búsqueda del factor corrector de humedad
FactorHum:
movlw   00010000b        ; Se indica el inicio de la tabla en la posición 0x1000
                        ; Esta tabla corresponde a la parte alta del factor de humedad

call    LecturMem          ; Llamada a subrutina de lectura sobre la memoria de
                        ; programa
                        ; El dato buscado se encuentra almacenado en el registro de trabajo (w)
movwf   FacHumA           ; Se guarda la parte alta del factor de humedad en la
                        ; variable "FacHumA"
movlw   00010100b        ; Se indica el inicio de la tabla en la posición 0x1400
                        ; Esta tabla corresponde a la parte baja del factor de humedad

call    LecturMem          ; Llamada a subrutina de lectura sobre la memoria de
                        ; programa
                        ; El dato buscado se encuentra almacenado en el registro de trabajo (w)
movwf   FacHumB           ; Se guarda la parte baja del factor de humedad en la
                        ; variable "FacHumB"
return  ; Retorna de subrutina

; #####
; Subrutina de búsqueda en la memoria de programa
; #####

LecturMem:                ; Subrutina que direcciona la posición "PosiciA:PosiciB"
                        ; en la memoria de programa
iorwf   TemperA,W         ; Se carga la parte alta de la posición en el registro de
                        ; trabajo

bsf     STATUS,RP1        ; Selección del Banco 2
bcf     STATUS,RP0

clrf    EEDATA            ; Se inicializa el registro "EEDATA" donde se guarda el
                        ; dato buscado

                        ; Se coloca la posición de la tabla sobre los registros "EEADRH:EEADR"

movwf   EEADRH            ; Se pasa la parte alta de la posición desde "w" al
                        ; registro "EEADRH"

bcf     STATUS,RP1        ; Selección del Banco 0
bcf     STATUS,RP0

movf    TemperB,W         ; Se carga la parte baja de la posición en el registro de
                        ; trabajo

bsf     STATUS,RP1        ; Selección del Banco 2
bcf     STATUS,RP0

movwf   EEADR             ; Se pasa la parte baja de la posición desde "w" al
                        ; registro "EEADR"

bsf     STATUS,RP1        ; Selección del Banco 3
bsf     STATUS,RP0

bsf     EECON1,EEPGD      ; Se selecciona el acceso a la memoria de programa
bsf     EECON1,RD         ; Se da la orden de lectura
nop     ; Instrucciones de espera para llevar a cabo la correcta
                        ; lectura a memoria

nop

bsf     STATUS,RP1        ; Selección del Banco 2

```

## APÉNDICE B

---

```
bcf     STATUS,RP0
movf    EEDATA,W           ; El dato deseado se encuentra en el registro "EEDATA" y
                           ; se copia en "w"
bcf     STATUS,RP1       ; Seleccion del Banco 0
bcf     STATUS,RP0
return                                     ; Retorna de subrutina
;#####
```

B.VI ARCHIVO PARAMETR

```

; PARAMETR.ASM
; Este archivo contienen las subrutinas que realizan la lectura de los sensores
;de temperatura y humedad relativa y las subrutinas que realizan los calculos
;necesarios para determinar parametros utiles en la informacion del usuario

IniSubrParametr      equ $          ; Se guarda la direccion de memoria donde comenzara el
codigo
; Variables locales
org IniVariables     ; Las nuevas variables se disponen en posiciones libres

FonEscalaA          equ    62      ; Constante que fija la parte alta de la "Vref" del CAD
FonEscalaB          equ    117     ; Constante que fija la parte baja de la "Vref" del CAD
OrigenTabla         equ    173     ; Constante indicativa del origen de la tabla de Humedad

; Variables asociadas a la subrutina de la lectura de los sensores
Decremente          res    1       ; Variable con la que se consigue un ciclo de espera de 20 us
TemperA             res    1       ; Variable que almacena la parte alta de la medida de Tª
TemperB             res    1       ; Variable que almacena la parte baja de la medida de Tª
HumedaA             res    1       ; Variable que almacena la parte alta de la medida de Hum
HumedaB             res    1       ; Variable que almacena la parte baja de la medida de Hum

; Variables asociadas a la subrutina que realiza la corrección del tiempo
TiemCorA            res    1       ; Variable que almacena la parte alta del tiempo corregido
TiemCorM            res    1       ; Variable que almacena la parte media del tiempo corregido
TiemCorB            res    1       ; Variable que almacena la parte baja del tiempo corregido

; Variables asociadas a la recta correctora del tiempo capturado por "TMR1"
Limite              res    1       ; Variable que guarda el BYTE limite superior del tramo recta
Pendiente           res    1       ; Variable que almacena la pendiente de la recta
CorteYAl            res    1       ; Variable que almacena la parte alta de la constante
CorteYMe            res    1       ; Variable que almacena la parte media de la constante
CorteYBa            res    1       ; Variable que almacena la parte baja de la constante

VARIABLE IniVariables = $          ; Se actualiza la posicion de comienzo
;de las variables aun libres
org IniSubrParametr ; Comienza el codigo de la subrutina en la direccion guardada

;#####
; Subrutina de medida de los sensores de Temperatura y Humedad relativa
;#####

MedirSensor:       ; Subrutina de medida de los parametros meteorologicos de forma alternada

; Se selecciona la medida a realizar segun el indicador "MensaPart" de "Indicadores"

btfsc Indicadores,MensaPart ; Se comprueba que parametro hay que medir con el "CAD"
goto  MedTempera          ; Se salta a la realizacion de la medida de la Tª

MedHumeda:         ; Subrutina de medida de la humeda relativa

bcf  STATUS,RP1      ; Seleccion del Banco 1
bsf  STATUS,RP0

; Configuracion del puerto "PORTA" para la conversion Analogico / Digital

movlw 10000100b      ; "RA0 y RA1" continuan cocmo entradas analogicas y "RA3"
;pasa ser una entrada
movwf ADCON1         ;analogica, dejando la tension de referencia a la
;alimentacion

bcf  STATUS,RP1      ; Seleccion del Banco 0
bcf  STATUS,RP0

; Configuracion del conversor Analogico / Digital
; Se realiza la medida de la "Hum" por "RA1"
; El registro "ADCON1" contiene el dato '10000101b' cuando entra en la subrutina

; Se configura el conversor A/D para que funcione con el oscilador interno RC
movlw 11000001b     ; La entrada analogica a seleccionada es "RA0"
movwf ADCON0        ; Tambien se activa el conversor A/D

;##### ESPERA #####
movlw 20             ; Se espera un tiempo antes de que de comienzo la conversion

```

## APÉNDICE B

```

movwf Decreme      ; Este tiempo lo determina el valor inicial de "Decre"
OtroH
decfsz Decreme     ; Bucle de espera para que se configure debidamente el CAD
goto OtroH         ; Del bucle se sale despues de 2+13*3 ciclos de instruccion
                  ; (20.5 us)
;##### ESPERA #####

                  ; Comienza la conversion
bsf  ADCON0,GO     ; Se da la orden de comienzo de conversion
btfsc Indicadores,ADormir ; Se comprueba el modo de realizar la conversion
sleep              ; Se duermen el microcontrolador

ConvirH
btfsc ADCON0,GO   ; Se sale del bucle al finalizar la conversion, indecado
                  ;por el bit "GO" a cero
goto ConvirH     ; Permanece en el bucle mientras este ocupado el CAD
                  ; Una vez finalizada la conversion se extrae el valor obtenido

bcf  STATUS,RP1
bsf  STATUS,RP0   ; Selecccion del Banco 1

movf  ADRESL,W    ; Se guarda la parte baja del dato digital en el registro
                  ;de trabajo
clrf  ADRESL

bcf  STATUS,RP1
bcf  STATUS,RP0   ; Selecccion del Banco0

movwf HumedaB     ; Tras pasar de nuevo al Banco0 se almacena el dato en la
                  ;variable "HumedaB"
movf  ADRESH,W    ; Extraccion del la parte alta del dato a la variable
                  ;"HumedaA"
clrf  ADRESH
movwf HumedaA     ;a taves del registro de trabajo

bcf  ADCON0,ADON  ; Se desactiva el conversor A/D

; Se resta la ordenada en el origen de las rectas que determinan la humedad relativa
; correspondiente a 0,832 V, indicativo de 0 % HR
; Para una resolucion de 10 bits el valor digital asociado con Vref = 5 V es 170

movlw OrigenTabla ; Se carga la ordenada en el origen en "W"
subwf HumedaB,F    ; Se resta el contenido de "W" de la parte baja del dato
                  ;convertido (HumedaB)
btfss STATUS,C    ; Se comprueba si se ha producido acarreo en la operacion
                  ;resta
decf  HumedaA     ; En el caso de NO existir acarreo se decrementa la
                  ;variable "HumedaA"
bsf  Indicadores,MensaPart ; Se pasa el relevo de la conversion a la Temperatura

return            ; Retorna de la subrrutina
;#####

MedTempera:      ; Subrrutina de medida de la temperatura del entorno

                  ; Configuracion del conversor Analogico / Digital
                  ; Se realiza la medida de la Tª por "RA0"

                  ; Hay que asegurarse que "ADCON1" contiene el dato '1000101b'
bcf  STATUS,RP1   ; Selecccion del Banco 1
bsf  STATUS,RP0

movlw 1000101b    ; "RA0 y RA1" son entradas analogicas y "RA3" es la
                  ;entrada de referencia positiva
movwf ADCON1     ; Los diez bits obtenidos se ajustan a la derecha

bcf  STATUS,RP1   ; Selecccion del Banco 0
bcf  STATUS,RP0

; Se configura el conversor A/D para que funcione con el oscilador interno RC
movlw 11001001b  ; La entrada analogica a seleccionada es "RA1"
; movlw 11000001b ; La entrada analogica a seleccionada es "RA0"
movwf ADCON0     ; Tambien se activa el conversor A/D

;##### ESPERA #####
movlw 20         ; Se espera un tiempo antes de que de comienzo la conversion
movwf Decreme    ; Este tiempo lo determina el valor inicial de "Decre"

```

## MEDIDOR DE DISTANCIA POR ULTRASONIDOS

```

OtroT
    decfsz Decreme          ; Bucle de espera para que se configure debidamente el CAD
    goto  OtroT             ; Del bucle se sale despues de 2+13*3 ciclos de
                            ;instruccion (20.5 us)
;##### ESPERA #####

                            ; Comienza la conversion
    bsf  ADCON0,GO          ; Se da la orden de comienzo de conversion de la señal
    btfsc Indicadores,ADormir ; Se comprueba el modo de realizar la conversion
    sleep                               ; Se duermen el microcontrolador

Convirt
    btfsc ADCON0,GO          ; Se sale del bucle al finalizar la conversion, indicado
                            ;por el bit "GO" a cero
    goto  Convirt           ; Permanece en el bucle mientras este ocupado el CAD

                            ; Una vez finalizada la conversion se extrae el valor obtenido
    bcf  STATUS,RP1
    bsf  STATUS,RP0          ; Seleccion del Banco 1

    movf  ADRESL,W           ; Se guarda la parte baja del dato digital en el registro
                            ;de trabajo
    clrf  ADRESL

    bcf  STATUS,RP1
    bcf  STATUS,RP0          ; Seleccion del Banco0

    movwf TemperB           ; Tras pasar de nuevo al Banco0 se almacena el dato en la
                            ;variable "TemperB"

    movf  ADRESH,W          ; Extraccion del la parte alta del dato a la variable
                            ;"TemperA"
    clrf  ADRESH
    movwf TemperA           ; a taves del registro de trabajo

    bcf  ADCON0,ADON        ; Se desactiva el conversor A/D

    bcf  Indicadores,MensaPart ; Se pasa el relevo de la conversion a la Humedad
    return                   ; Retorna de la subrutina

;#####
; Subrutina de operaciones para la obtencion de los parametros meteorologicos
;#####
;#####
; Subrutina de operaciones para la obtencion de la humedad relativa
;#####

; Se han de tener definidos los datos "HumedaA:HumedaB" y "FacHumA:FacHumB"
; El resultado de la operacion se entrega en "Dato4:Dato3:Dato2:Dato1:Dato0"

OperaHume:           ; Subrutina que realiza las operaciones para corregir la humedad relativa
;#####
    clrwdt ; Se inicializa el contador del Perro Guardian
;#####
    ; La operacion a realizar para obtener el valor correcto es la que se muestra
        ; HR(%) = HumedaA:HumedaB * FactorHum (T^a)

    ; La subrutina que realiza el producto, opera con "Multiplicando" y "Multiplicador"

    ; "Multiplicando" de 40 bits:
    ; "Multiplicando+4:Multiplicando+3:Multiplicando+2:Multiplicando+1:Multiplicando"
    ;cargado con la Humedad Relativa ==> 0...0:0...0:0...0:HumedaA:HumedaB

    ; "Multiplicando" de 32 bits:
    ;
    ; "Multiplicador+1:Multiplicador"
    ;cargado con el Factor de la Humedad ==> FacHumA:FacHumB

    ; Se carga el valor del multiplicando
    clrf  Multiplicando+4          ; Se borra el quinto BYTE
    clrf  Multiplicando+3          ; Se borra el cuarto BYTE
    clrf  Multiplicando+2          ; Se borra el tercer BYTE
    movf  HumedaA,W               ; Se carga "HumedaA" en el segundo BYTE
    movwf Multiplicando+1         ; a traves del registro de trabajo (w)
    movf  HumedaB,W               ; Se carga "HumedaB" en el primer BYTE
    movwf Multiplicando           ; a traves del registro de trabajo (w)

    ; Se carga el valor del multiplicador

```

## APÉNDICE B

```

movf   FacHumA,W           ; Se carga "FacHumA" en el segundo BYTE
movwf  Multiplicador+1     ;a traves del registro de trabajo (w)
movf   FacHumB,W           ; Se carga "FacHumB" en el primer BYTE
movwf  Multiplicador       ;a traves del registro de trabajo (w)

call   MULT24x16          ; Llamada a subrutina que realiza el producto
      ; El resultado de la operacion se encuentra en "Producto"
      ; "Producto+4:Producto+3:Producto+2:Producto+1:Producto"

; Se realiza la extracion del resultado de la multiplicacion a "Dato"
movf   Producto+4,W       ; Se realiza el traspaso del 5° BYTE a "Dato"
movwf  Dato4               ;a traves del registro de trabajo (w)
movf   Producto+3,W       ; Se realiza el traspaso del 4° BYTE a "Dato3"
movwf  Dato3               ;a traves del registro de trabajo (w)
movf   Producto+2,W       ; Se realiza el traspaso del 3° BYTE a "Dato2"
movwf  Dato2               ;a traves del registro de trabajo (w)
movf   Producto+1,W       ; Se realiza el traspaso del 2° BYTE a "Dato1"
movwf  Dato1               ;a traves del registro de trabajo (w)
movf   Producto,W         ; Se realiza el traspaso del 1° BYTE a "Dato0"
movwf  Dato0               ;a traves del registro de trabajo (w)

return                               ; Retorna de la subrutina

;#####
; Subrutina de operaciones para la obtencion de la temperatura
;#####

; Se han de tener definidos los datos "TemperA:TemperB"
; El resultado de la operacion se entrega en "Dato4:Dato3:Dato2:Dato1:Dato0"

OperaTemp:      ; Subrutina que realiza las operaciones para conseguir la temperatura en °C
;#####
clrwdt ; Se inicializa el contador del Perro Guardian
;#####
; Ahora hay que pasar el resultado de decimas de ° a °C con la operacion
      ; T(°C) = (TemperA:TemperB * FS) / (2^10)
      ; Por "FS" se entiende que es el Fondo de Escala que para este caso es 1.6016

; La subrutina que realiza el producto, opera con "Multiplicando" y "Multiplicador"

; "Multiplicando" de 40 bits:
; "Multiplicando+4:Multiplicando+3:Multiplicando+2:Multiplicando+1:Multiplicando"
; cargado con la Temperatura ==> 0...0:0...0:0...0:TemperA:TemperB

; "Multiplicando" de 32 bits:
; "Multiplicador+1:Multiplicador"
; cargado con el Fondo de Escala ==> FonEscalaA:FonEscalaB

      ; Se carga el valor del multiplicando
clrf   Multiplicando+4     ; Se borra el quinto BYTE
clrf   Multiplicando+3     ; Se borra el cuarto BYTE
clrf   Multiplicando+2     ; Se borra el tercer BYTE
movf   TemperA,W           ; Se carga "TemperA" en el segundo BYTE
movwf  Multiplicando+1     ;a traves del registro de trabajo (w)
movf   TemperB,W           ; Se carga "TemperB" en el primer BYTE
movwf  Multiplicando       ;a traves del registro de trabajo (w)

      ; Se carga el valor del multiplicador
movlw  FonEscalaA          ; Se carga "FonEscalaA" en el segundo BYTE
movwf  Multiplicador+1     ;a traves del registro de trabajo (w)
movlw  FonEscalaB          ; Se carga "FonEscalaB" en el primer BYTE
movwf  Multiplicador       ;a traves del registro de trabajo (w)

call   MULT24x16          ; Llamada a subrutina que realiza el producto
      ; El resultado de la operacion se encuentra en "Producto"
      ; "Producto+4:Producto+3:Producto+2:Producto+1:Producto"

; Se realiza la extracion del resultado de la multiplicacion a "Dato"
clrf   Dato4               ; Se borra "Dato4"
clrf   Dato3               ; Se borra "Dato3"
movf   Producto+3,W       ; Se realiza el traspaso del 4° BYTE a "Dato2"
movwf  Dato2               ;a traves del registro de trabajo (w)
movf   Producto+2,W       ; Se realiza el traspaso del 3° BYTE a "Dato1"
movwf  Dato1               ;a traves del registro de trabajo (w)
movf   Producto+1,W       ; Se realiza el traspaso del 2° BYTE a "Dato0"
movwf  Dato0               ;a traves del registro de trabajo (w)
      ; Se desprecia el BYTE de menor peso, con lo que se realiza la division [/2^8]

```

## MEDIDOR DE DISTANCIA POR ULTRASONIDOS

```

bcf     STATUS,C           ; Se carga '0' en el bit de acarreo
rrf     Dato2,F           ;
rrf     Dato1,F           ; Con esta rotacion a la derecha del dato completo,
rrf     Dato0,F           ; se consigue la division [/2]

bcf     STATUS,C           ; Se carga '0' en el bit de acarreo
rrf     Dato2,F           ;
rrf     Dato1,F           ; Con esta rotacion a la derecha del dato completo,
rrf     Dato0,F           ; se consigue la division [/2]
; Se ha realizado la division completa [/2^10]

return          ; Retorna de la subrutina

;#####
; Subrutina de operaciones para la correccion del tiempo medido
;#####

; Se parte del resultado del temporizador TMR1, localizado en "TiempoA:TiempoB"
; Se opera con las variables "Pendiente" y "CorteYAl:CorteYMe:CorteYBa"
; El resultado se devuelve sobre "TiemCorA:TiemCorM:TiemCorB"

CorrecTiemp:   ; Subrutina que realiza las operaciones para corregir el tiempo

;#####
clrwdt ; Se inicializa el contador del Perro Guardian
;#####
; Se inicializan las variables que almacenan el resultado de la correccion
clrf   TiemCorA   ; Se borra "TiemCorA"
clrf   TiemCorM   ; Se borra "TiemCorM"
clrf   TiemCorB   ; Se borra "TiemCorB"

; En primer lugar hay que localizar el rango al que pertenece el tiempo
;medido, para aplicar la correspondiente recta correctora
movlw  255
movwf  Cuenta     ; Se inicializa la variable de entrada a las tablas
;tras entrar en el bucle de localizacion del rango

TiemAlta
incf   Cuenta,F   ; Se pasa al siguiente tramo de la recta correctora
call   TabLimA    ; Se llama a la subrutina que entrega la parte alta del limite
movwf  Limite     ; Se traspasa a "Limite" el BYTE extraido de la tabla
movf   TiempoA,W  ; Se carga en "W" la parte alta del tiempo medido
subwf  Limite,W   ; Se realiza la resta (Limite - TiempoA)
btfss  STATUS,C   ; C = 1 mientras "TiempoA" no supere a "Limite"
goto   TiemAlta

btfss  STATUS,Z   ; Z = 1 cuando "TiempoA" coincida con "Limite"
goto   DatosTramo
; En el caso de que la parte alta del tiempo medido coincida con el limite de tiempo
;se realizara la comparacion entre las partes bajas de ambos tiempos
call   TabLimB    ; Se llama a la subrutina que entrega la parte baja del limite
movwf  Limite     ; Se traspasa a "Limite" el BYTE extraido de la tabla
movf   TiempoB,W  ; Se carga en "W" la parte alta del tiempo medido
subwf  Limite,W   ; Se realiza la resta (Limite - TiempoB)
btfss  STATUS,C   ; C = 1 mientras "TiempoB" no supere a "Limite"
incf   Cuenta,F   ; Se pasa al siguiente tramo de la recta correctora

; En este momento "Cuenta" almacena la posicion del tramo corrector, y se
;procede a la extraccion de las variables asociadas a dicho tramo
DatosTramo
call   Pendi      ; Se llama a la subrutina que entrega la pendiente
movwf  Pendiente  ; Se traspasa a "Pendiente" el BYTE extraido de la tabla
call   ConstanA   ; Se llama a la subrutina que entrega la parte alta de la CTE
movwf  CorteYAl   ; Se traspasa a "CorteYAl" el BYTE extraido de la tabla
call   ConstanM   ; Se llama a la subrutina que entrega la parte media de la CTE
movwf  CorteYMe   ; Se traspasa a "CorteYMe" el BYTE extraido de la tabla
call   ConstanB   ; Se llama a la subrutina que entrega la parte baja de la CTE
movwf  CorteYBa   ; Se traspasa a "CorteYBa" el BYTE extraido de la tabla

; Una vez definidas todas las variables se realizan los calculos necesarios
; La subrutina que realiza el producto, opera con "Multiplicando" y "Multiplicador"

; "Multiplicando" de 40 bits:
; "Multiplicando+4:Multiplicando+3:Multiplicando+2:Multiplicando+1:Multiplicando"
; cargado con el Tiempo Medido ==> 0...0:0...0:0...0:TiempoA:TiempoB

; "Multiplicando" de 32 bits:

```

## APÉNDICE B

```

;                                     "Multiplicador+1:Multiplicador"
; cargado con la Pendiente de la recta ==> 0...0:Pendiente

; Se carga el valor del multiplicando
clrf Multiplicando+4          ; Se borra el quinto BYTE
clrf Multiplicando+3          ; Se borra el cuarto BYTE
clrf Multiplicando+2          ; Se borra el tercer BYTE
movf TiempoA,W                ; Se carga "TiempoA" en el segundo BYTE
movwf Multiplicando+1         ; a traves del registro de trabajo (w)
movf TiempoB,W                ; Se carga "TiempoB" en el primer BYTE
movwf Multiplicando           ; a traves del registro de trabajo (w)

; Se carga el valor del multiplicador
clrf Multiplicador+1          ; Se borra el segundo BYTE
movf Pendiente,W              ; Se carga "Pendiente" en el primer BYTE
movwf Multiplicador           ; a traves del registro de trabajo (w)

call MULT24x16                ; Llamada a subrutina que realiza el producto
; El resultado de la operacion se encuentra en "Producto"
; "Producto+4:Producto+3:Producto+2:Producto+1:Producto"

; Se procede a realizar la suma ==> Producto + CorteY

; Se realiza la suma del primer BYTE
movf CorteYBa,W              ; Se carga en "W" el primer Byte de la ordenada en el origen
addwf Producto,W             ; Se realiza la suma (Producto + CorteYBa)
movwf TiemCorB                ; Se guarda el resultado en "TiemCorB"
; Se introduce el valor del acarreo al siguiente BYTE
rlf TiemCorM,F                ; Se introduce el acarreo en el 2° BYTE

; Se realiza la suma del segundo BYTE
movf CorteYMe,W              ; Se carga en "W" el segundo Byte de la ordenada en el origen
addwf TiemCorM,W             ; Se realiza la suma (C + CorteYMe), sin producir acarreo
addwf Producto+1,W           ; Se completa la suma (Producto+1 + (C + CorteYMe))
movwf TiemCorM                ; Se guarda el resultado en "TiemCorM"
; Se introduce el valor del acarreo al siguiente BYTE
rlf TiemCorA,F                ; Se introduce el acarreo en el 3° BYTE

; Se realiza la suma del tercer BYTE
movf CorteYAl,W              ; Se carga en "W" el segundo Byte de la ordenada en el origen
addwf TiemCorA,W             ; Se realiza la suma (C + CorteYAl), sin producir acarreo
addwf Producto+2,W           ; Se completa la suma (Producto+2 + (C + CorteYAl))
movwf TiemCorA                ; Se guarda el resultado en "TiemCorA"

; El acarreo de esta operacion no se tiene en cuenta, ya que es el bit
; de signo, y se consideran todos los numeros positivos

return                        ; Retorna de la subrutina

#####
; Subrutina de operaciones para la obtencion de la distancia
#####

; Se han de tener definidos los datos "TiemCorA:TiemCorM:TiempoB" y "FacVelA:FacVelB"
; El resultado de la operacion se entrega en "Dato4:Dato3:Dato2:Dato1:Dato0"

OperaDist:                    ; Subrutina que realiza las operaciones para obtener la distancia

#####
clrwdt ; Se inicializa el contador del Perro Guardian
#####
; Se procede a realizar el calculo de la distancia por determinar
; La operacion a realizar es el producto ==> v(T^a) * TiempoCorregido

; La subrutina que realiza el producto, opera con "Multiplicando" y "Multiplicador"

; "Multiplicando" de 40 bits:
; "Multiplicando+4:Multiplicando+3:Multiplicando+2:Multiplicando+1:Multiplicando"
; cargado con el Tiempo Corregido==> 0...0:0...0:TiemCorA:TiemCorM:TiemCorB

; "Multiplicando" de 32 bits:
;                                     "Multiplicador+1:Multiplicador"
; cargado con el Factor de la Velocidad==> FacVelA:FacVelB

; Se carga el valor del multiplicando
clrf Multiplicando+4          ; Se borra el quinto BYTE
clrf Multiplicando+3          ; Se borra el cuarto BYTE

```

## MEDIDOR DE DISTANCIA POR ULTRASONIDOS

```
movf   TiemCorA,W           ; Se carga "tiemCorA" en el tercer BYTE
movwf  Multiplicando+2     ;a traves del registro de trabajo (w)
movf   TiemCorM,W           ; Se carga "TiemCorM" en el segundo BYTE
movwf  Multiplicando+1     ;a traves del registro de trabajo (w)
movf   TiemCorB,W           ; Se carga "TiemCorB" en el primer BYTE
movwf  Multiplicando       ;a traves del registro de trabajo (w)

; Se carga el valor del multiplicador
movf   FacVelA,W           ; Se carga "FacVela" en el segundo BYTE
movwf  Multiplicador+1     ;a traves del registro de trabajo (w)
movf   FacVelB,W           ; Se carga "FacVelB" en el primer BYTE
movwf  Multiplicador       ;a traves del registro de trabajo (w)

call   MULT24x16           ; Llamada a subrutina que realiza el producto
; El resultado de la operacion se encuentra en "Producto"
; "Producto+4:Producto+3:Producto+2:Producto+1:Producto"

; Se realiza la extracion del resultado de la multiplicacion a "Dato"
movf   Producto+4,W        ; Se realiza el traspaso del 5° BYTE a "Dato4"
movwf  Dato4               ;a traves del registro de trabajo (w)
movf   Producto+3,W        ; Se realiza el traspaso del 4° BYTE a "Dato3"
movwf  Dato3               ;a traves del registro de trabajo (w)
movf   Producto+2,W        ; Se realiza el traspaso del 3° BYTE a "Dato2"
movwf  Dato2               ;a traves del registro de trabajo (w)
movf   Producto+1,W        ; Se realiza el traspaso del 2° BYTE a "Dato1"
movwf  Dato1               ;a traves del registro de trabajo (w)
movf   Producto,W          ; Se realiza el traspaso del 1° BYTE a "Dato0"
movwf  Dato0               ;a traves del registro de trabajo (w)

; Se realiza la correccion de la dimension longitudinal del aparato
movf   CorrecDista,W       ; Se guarda en "W" el valor corrector de la distancia
addwf  Dato3,F             ; Se guarda en "Dato3" la suma (CorrecDista + Dato3)
clrw
btfsc  STATUS,C           ; C = 0 mientras no produzca acarreo en la suma
movlw  1
; "W" contiene el valor del acarreo
addwf  Dato4,F             ; Se guarda en "Dato4" la suma (C + Dato4)

return                       ; Retorna de la subrutina
```

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

B.VII ARCHIVO SENULT

```

; SENULT.ASM
; Este archivo contiene la subrutina que genera la señal ultrasonica y
;posteriormente evalua el tiempo que tarda en llegar el ECO

IniSubrSeñul equ $          ; Se guarda la direccion de memoria donde comenzara el codigo
; Variables locales
org IniVariables          ; Las nuevas variables se disponen en posiciones libres

Retardo      res    1      ; Variable empleada en el ajuste del semiperiodo de 12,5 us
Pulso        res    1      ; Variable que guarda el n° de semiciclos del pulso elemental
Factor        res    1      ; Variable que guarda n° repeticiones del pulso elemental
Situacion     res    1      ; Variable que guarda la identificacion del pulso de emision

; Variables asociadas a la cuenta del tiempo empleado en el viaje de la señal
TiempoA      res    1      ; Variable que almacena la parte alta del tiempo alcanzado
TiempoB      res    1      ; Variable que almacena la parte baja del tiempo alcanzado

VARIABLE IniVariables = $    ; Se actualiza la posicion de comienzo
                             ;de las variables aun libres
org IniSubrSeñul          ; Comienza el codigo de la subrutina en la direccion guardada

;#####
; Subrutina que genera la señal de 40 kHz con distintas anchuras de pulso
;#####

EnviaSeñal:  ; Subrutina que genera una señal con un periodo de 25 uS
;#####
clrwdt      ; Se inicializa el contador del Perro Guardian
;#####

bcf    Indicadores,Interrup  ; Se inicializa el indicativo secuencia interrumpida
bcf    Indicadores,TmpDesbor ; Se inicializa el indicativo de "TMR1" desbordado

movlw    1
movwf    Situacion          ; Se inicializa "Situacion" indicando la 1ª

; Se activan las interrupciones del desbordamiento de "TMR1"
bsf    INTCON,PEIE        ;Se activan ahora las interrupciones por perifericos
                             ; PEIE = Perifericos

Carga
; Se carga el factor multiplicador de la anchura del pulso
; Antes del envio de cada pulso se comprueba "Decision"
movf    Decision,F        ; Se comprueba la existencia de decision
btfss   STATUS,Z          ; Z = 1 cuando no hay una decision definida
goto    EnvioFin          ; Se aborta el envio de la señal

btfsc   Situacion,0
movlw   1                  ; Se carga el factor 1
btfsc   Situacion,1
movlw   2                  ; Se carga el factor 2
btfsc   Situacion,2
movlw   6                  ; Se carga el factor 6
btfsc   Situacion,3
movlw   10                 ; Se carga el factor 10
btfsc   Situacion,4
movlw   16                 ; Se carga el factor 16
btfsc   Situacion,5
movlw   20                 ; Se carga el factor 20
btfsc   Situacion,6
movlw   30                 ; Se carga el factor 30
btfsc   Situacion,7
movlw   40                 ; Se carga el factor 40
movwf   Factor             ; Se actualiza el factor que multiplica al pulso elemental

movlw   40                 ; Se carga en "Pulso" el numero de semiciclos que tiene
movwf   Pulso              ; del pulso elemental, a traves de "W"

movlw   3                  ; Se carga en "Retardo" el numero repeticiones de un bucle
movwf   Retardo            ;para completan cada semiperiodo, a traves de "W"
; Inicializacion del temporizador "TMR1"
clrf    TMR1H              ; Se borra la parte alta del registro temporizador
clrf    TMR1L              ; Se borra la parte baja del registro temporizador

```

## MEDIDOR DE DISTANCIA POR ULTRASONIDOS

```

bsf      T1CON,TMR1ON    ; Se da la orden de comienzo de cuenta al temporizador "TMR1"

; Un vez concluida la inicializacion se genera la señal deseada
NCiclo   ;24
btfsc   PORTA,2         ;0          ; Se textea el valor anterior que toma la
;salida
goto    ABajo           ;1          ; Salta a la puesta a nivel bajo
nop     ;2              ; Operacion de retardo para ajustar el
;semiciclo
nop     ;3              ; Operacion de retardo para ajustar el
;semiciclo
nop     ;4              ; Operacion de retardo para ajustar el
;semiciclo
AAalto   ;4
movlw   00000100b       ;5
movwf   PORTA           ;6          ; Se pone "RA2" a '1' y "RA5" a '0'
goto    Decre           ;7          ; Se salta a completar el semiciclo

ABajo    ;2
nop     ;3              ; Operacion de retardo para ajustar el
;semiciclo
nop     ;4              ; Operacion de retardo para ajustar el
;semiciclo
movlw   00100000b       ;5
movwf   PORTA           ;6          ; Se pone "RA2" a '0' y "RA5" a '1'
nop     ;7              ; Operacion de retardo para ajustar el
;semiciclo
nop     ;8              ; Operacion de retardo para ajustar el
;semiciclo

Decre    ;11           ;8          ; Se completa el tiempo del retardo
decfsz  Retardo         ;9          ; Se decrementa "Retardo" hasta llegar a
;'0'
goto    Decre           ;10         ; Permanece en el bucle hasta completar el
;semiciclo

; Se decrementa y textea la variable que fija la duracion del pulso

movlw   3                ;17         ; Se inicializa de nuevo "Retardo",
movwf   Retardo         ;18         ; para completar el siguiente semiciclo
movlw   40               ;19         ; Se guarda en "W" el valor inicial de
;"Pulso"
nop     ;20             ; Operacion de retardo para ajustas el
;semiciclo
nop     ;21             ; Operacion de retardo para ajustas el
;semiciclo
decfsz  Pulso,F         ;22         ; Se decrementa la variable cada semiciclo
goto    NCiclo         ;23         ; Se continua generando el pulso elemental

;#####
; Se comprueba que ya es el ultimo semiciclo del pulso completandolo con 7 ciclos de
;instruccion
;#####
decfsz  Factor,F        ;24         ; Se decrementa el factor multiplicador del pulso
;elemental
goto    AunMas          ;0          ; Se continua mandando otro pulso elemental
nop     ;1              ; Operacion de retardo para ajustar el semiciclo
nop     ;2              ; Operacion de retardo para ajustar el semiciclo
nop     ;3              ; Operacion de retardo para ajustar el semiciclo
nop     ;4              ; Operacion de retardo para ajustar el semiciclo
nop     ;5              ; Operacion de retardo para ajustar el semiciclo
clrf   PORTA           ;6          ; Se pone a cero "PORTA" al completo
goto    Deteccion       ; Se salta a la espera la respuesta de la señal
;emitida
;#####

AunMas   ;1            ; Se ajusta los ciclos de instruccion para enlazar
;dos pulsos elementales
movwf   Pulso          ;2          ; Se carga de nuevo el numero de semiciclos en la
;variable "Pulso"
goto    AAalto         ;3          ; El enlace se realiza al comienzo del estado en
;alto
;#####
; Se espera la llegada del eco de la señal emitida
Deteccion

; Se evalua el estado de "Recib" para eliminar el primer pulso

```

## APÉNDICE B

```

Recib0          ; Bucle de permanencia mientras "Recib" esta a '0'
btfss PuerCon,Recib
goto Recib0

Recib1          ; Bucle de permanencia mientras "Recib" esta a '1'
btfsc PuerCon,Recib
goto Recib1

; Se activan las interrupciones por flanco en "RB0" para detectar el ECO recibido
bcf INTCON,INTF ; Se baja la bandera indicativa de flanco en "RB0"
bsf INTCON,INTE ; Se habilita la interrupcion por flanco en "RB0/INT"

; Se entra en la espera del ECO de la señal emitida, evaluandose continuamente
;el motivo de interrupcion, que origina el fin de cuenta de "TMR1"
NoSeñal        ; Bucle de permanencia mientras la señal esta en el camino
;#####
clrwdt ; Se inicializa el contador del Perro Guardian
;#####
btfsc Indicadores,TmpDesbor ; Se comprueba el estado de la cuenta de "TMR1"
goto SePaso ; Se salta a la alternativa ante el desbordamiento de "TMR1"
movf Decision,F ; Se comprueba la existencia de decision
btfsc STATUS,Z ; Z = 0 cuando "Decision" no es nula
goto NoSeñal ; Continua con la cuenta de tiempo

btfss Decision,Recib
goto EnvioFin

; Se anula el suceso de interrupcion por flanco en "RB0/INT"
bcf INTCON,INTE ; Se deshabilita la interrupcion por flanco en "RB0/INT"
bcf Decision,Recib ; Se pone a cero el bit indicativo de ECO recibido

;#####
clrwdt ; Se inicializa el contador del Perro Guardian
;#####
;#####
; Se aplica un recorte en el tiempo de espera, para cada tipo de pulso emitido
btfsc Situacion,0
movlw 90 ; Se carga en "W" el maximo valor permitido para "TMR1H"
btfsc Situacion,1
movlw 120 ; Se carga en "W" el maximo valor permitido para "TMR1H"
btfsc Situacion,2
movlw 150 ; Se carga en "W" el maximo valor permitido para "TMR1H"
btfsc Situacion,3
movlw 255 ; Se carga en "W" el maximo valor permitido para "TMR1H"
btfsc Situacion,4
movlw 255 ; Se carga en "W" el maximo valor permitido para "TMR1H"
btfsc Situacion,5
movlw 255 ; Se carga en "W" el maximo valor permitido para "TMR1H"
btfsc Situacion,6
movlw 255 ; Se carga en "W" el maximo valor permitido para "TMR1H"
btfsc Situacion,7
movlw 255 ; Se carga en "W" el maximo valor permitido para "TMR1H"

;#####
; Se comprueba las condiciones en que se recibe el ECO del pulso emitido
subwf TMR1H,W ; Se realiza la resta (TMR1H - W)
btfsc STATUS,C ; C = 0 mientras TMR1H no supera el limite
goto SePaso ; Se salta a la etiqueta indicada

;###
; Se comprueba que el ECO recibido, cumple las condiciones asociadas al pulso emitido
; Se exige una duracion minima al pulso que origina el ECO recibido
btfsc Situacion,0
movlw 6 ; Se carga el factor apropiado para comprobar el pulso 1
btfsc Situacion,1
movlw 12 ; Se carga el factor apropiado para comprobar el pulso 2
btfsc Situacion,2
movlw 31 ; Se carga el factor apropiado para comprobar el pulso 3
btfsc Situacion,3
movlw 50 ; Se carga el factor apropiado para comprobar el pulso 3
btfsc Situacion,4
movlw 50 ; Se carga el factor apropiado para comprobar el pulso 4
btfsc Situacion,5
movlw 50 ; Se carga el factor apropiado para comprobar el pulso 5
btfsc Situacion,6
movlw 50 ; Se carga el factor apropiado para comprobar el pulso 6
btfsc Situacion,7
movlw 50 ; Se carga el factor apropiado para comprobar el pulso 7

```

## MEDIDOR DE DISTANCIA POR ULTRASONIDOS

```

movwf  Factor          ; Se actualiza el factor que multiplica la espera elemental
;#####
clrwdt ; Se inicializa el contador del Perro Guardian
;#####

;#####
btfss  PuerCon,Recib
goto   SePaso

AunEspera          ; t = 101.5*(Factor-1)+101 [uS]
call   Espe01ms
decfsz Factor
goto   AunEspera

btfss  PuerCon,Recib
goto   SePaso
;#####
clrwdt ; Se inicializa el contador del Perro Guardian
;#####

; Cuando el ECO recibido, cumple ambos requisitos se continua con el codigo
;#####
; Se extrae el tiempo empleado en llegar la señal al extremo opuesto
movf   TMR1H,W        ; Se realiza el traspaso de "TMR1H" a "TiempoA"
movwf  TiempoA        ; a traves del registro de trabajo (w)
clrf   TMR1H

movf   TMR1L,W        ; Se realiza el traspaso de "TMR1L" a "TiempoB"
movwf  TiempoB        ; a traves del registro de trabajo (w)
clrf   TMR1L

return          ; Retorna de la subrutina

;#####
SePaso
; Se anula el suceso de interrupcion por flanco en "RB0/INT"
bcf    INTCON,INTE    ; Se deshabilita la interrupcion por flanco en "RB0/INT"
;#####
clrwdt ; Se inicializa el contador del Perro Guardian
;#####
; Se pasa a la siguiente situacion en que se encuentra la emision
btfsc  Situacion,0
movlw  2          ; Se carga la situacion correspondiente al factor 5
btfsc  Situacion,1
movlw  4          ; Se carga la situacion correspondiente al factor 10
btfsc  Situacion,2
movlw  8          ; Se carga la situacion correspondiente al factor 30
btfsc  Situacion,3
movlw  16         ; Se carga la situacion correspondiente al factor 80
btfsc  Situacion,4
movlw  32         ; Se carga la situacion correspondiente al factor 100
btfsc  Situacion,5
movlw  64         ; Se carga la situacion correspondiente al factor 150
btfsc  Situacion,6
movlw  128        ; Se carga la situacion correspondiente al factor 200
btfsc  Situacion,7
return          ; Retorna de la subrutina

movwf  Situacion    ; Se pasa el dato de "W" en la variable "Situacion"
bcf    Indicadores,TmpDesbor ; Se desactiva el indicativo de "TMR1"desbordado
call   Espe20ms    ; Se espera la anulacion de todos los ecos anteriores
goto   Carga        ; Se salta a la carga de un nuevo factor multiplicador del
; pulso elemental y al envio del pulso correspondiente

EnvioFin          ; Salida de envio de señal por toma de decision
bsf    Indicadores,Interrup ; Se activa el indicativo de secuencia interrumpida
;#####
clrwdt ; Se inicializa el contador del Perro Guardian
;#####
; Se anula el suceso de interrupcion por flanco en "RB0/INT"
bcf    INTCON,INTE ; Se deshabilita la interrupcion por flanco en "RB0/INT"

return          ; Retorna de la subrutina

;#####

```

B.VIII ARCHIVO TABLAS

```

; TABLAS.ASM
; Este archivo contiene las tablas empleadas por el programa, distinguiendo
; las de mensajes y de datos asociados a la medida del tiempo medido
; a las que se accede por medio de una llamada a subrutina
; de las de factores correctores accesibles desde la lectura de la memoria de programa

; No es necesaria la definicion de variables locales

;#####          TABLAS DE MENSAJES          #####

org      0x0600

Men1Lin1:          ; El mensaje queda definido por una tabla con "dt"
movlw  $>>8        ; Se carga el BYTE de mayor peso de la direccion en
movwf  PCLATH       ; "PCLATH", a traves del registro de trabajo (w)
movf   Cuenta,W     ; Se carga la posicion a leer de la tabla en "W"
addwf  PCL          ; Se carga en "W" el siguiente caracter a pintar
dt     "Calibrado de",0 ; El fin del mensaje se señala con un "0"

Men1Lin2:          ; El mensaje queda definido por una tabla con "dt"
movlw  $>>8        ; Se carga el BYTE de mayor peso de la direccion en
movwf  PCLATH       ; "PCLATH", a traves del registro de trabajo (w)
movf   Cuenta,W     ; Se carga la posicion a leer de la tabla en "W"
addwf  PCL          ; Se carga en "W" el siguiente caracter a pintar
dt     "la Distancia",0 ; El fin del mensaje se señala con un "0"

Men2Lin1:          ; El mensaje queda definido por una tabla con "dt"
movlw  $>>8        ; Se carga el BYTE de mayor peso de la direccion en
movwf  PCLATH       ; "PCLATH", a traves del registro de trabajo (w)
movf   Cuenta,W     ; Se carga la posicion a leer de la tabla en "W"
addwf  PCL          ; Se carga en "W" el siguiente caracter a pintar
dt     " AJUSTE",0     ; El fin del mensaje se señala con un "0"

Men2Lin2:          ; El mensaje queda definido por una tabla con "dt"
movlw  $>>8        ; Se carga el BYTE de mayor peso de la direccion en
movwf  PCLATH       ; "PCLATH", a traves del registro de trabajo (w)
movf   Cuenta,W     ; Se carga la posicion a leer de la tabla en "W"
addwf  PCL          ; Se carga en "W" el siguiente caracter a pintar
dt     "Distancia:      m",0 ; El fin del mensaje se señala con un "0"

Men3Lin1:          ; El mensaje queda definido por una tabla con "dt"
movlw  $>>8        ; Se carga el BYTE de mayor peso de la direccion en
movwf  PCLATH       ; "PCLATH", a traves del registro de trabajo (w)
movf   Cuenta,W     ; Se carga la posicion a leer de la tabla en "W"
addwf  PCL          ; Se carga en "W" el siguiente caracter a pintar
dt     "Ha concluido el",0 ; El fin del mensaje se señala con un "0"

Men3Lin2:          ; El mensaje queda definido por una tabla con "dt"
movlw  $>>8        ; Se carga el BYTE de mayor peso de la direccion en
movwf  PCLATH       ; "PCLATH", a traves del registro de trabajo (w)
movf   Cuenta,W     ; Se carga la posicion a leer de la tabla en "W"
addwf  PCL          ; Se carga en "W" el siguiente caracter a pintar
dt     "Calibrado",0   ; El fin del mensaje se señala con un "0"

Men4Lin1:          ; El mensaje queda definido por una tabla con "dt"
; movlw  $>>8        ; Se carga el BYTE de mayor peso de la direccion en
; movwf  PCLATH       ; "PCLATH", a traves del registro de trabajo (w)
; movf   Cuenta,W     ; Se carga la posicion a leer de la tabla en "W"
; addwf  PCL          ; Se carga en "W" el siguiente caracter a pintar
; dt     "",0         ; El fin del mensaje se señala con un "0"

Men4Lin2:          ; El mensaje queda definido por una tabla con "dt"
; movlw  $>>8        ; Se carga el BYTE de mayor peso de la direccion en
; movwf  PCLATH       ; "PCLATH", a traves del registro de trabajo (w)
; movf   Cuenta,W     ; Se carga la posicion a leer de la tabla en "W"
; addwf  PCL          ; Se carga en "W" el siguiente caracter a pintar
; dt     "",0         ; El fin del mensaje se señala con un "0"

Men5Lin1:          ; El mensaje queda definido por una tabla con "dt"
; movlw  $>>8        ; Se carga el BYTE de mayor peso de la direccion en
; movwf  PCLATH       ; "PCLATH", a traves del registro de trabajo (w)
; movf   Cuenta,W     ; Se carga la posicion a leer de la tabla en "W"
; addwf  PCL          ; Se carga en "W" el siguiente caracter a pintar

```

## MEDIDOR DE DISTANCIA POR ULTRASONIDOS

---

```

;      dt "",0          ; El fin del mensaje se señala con un "0"

Men5Lin2:              ; El mensaje queda definido por una tabla con "dt"
;      movlw $>>8      ; Se carga el BYTE de mayor peso de la direccion en
;      movwf PCLATH    ; "PCLATH", a traves del registro de trabajo (w)
;      movf  Cuenta,W  ; Se carga la posicion a leer de la tabla en "W"
;      addwf PCL       ; Se carga en "W" el siguiente caracter a pintar
;      dt  "",0        ; El fin del mensaje se señala con un "0"

Men6Lin1:              ; El mensaje queda definido por una tabla con "dt"
;      movlw $>>8      ; Se carga el BYTE de mayor peso de la direccion en
;      movwf PCLATH    ; "PCLATH", a traves del registro de trabajo (w)
;      movf  Cuenta,W  ; Se carga la posicion a leer de la tabla en "W"
;      addwf PCL       ; Se carga en "W" el siguiente caracter a pintar
;      dt  "",0        ; El fin del mensaje se señala con un "0"

Men6Lin2:              ; El mensaje queda definido por una tabla con "dt"
;      movlw $>>8      ; Se carga el BYTE de mayor peso de la direccion en
;      movwf PCLATH    ; "PCLATH", a traves del registro de trabajo (w)
;      movf  Cuenta,W  ; Se carga la posicion a leer de la tabla en "W"
;      addwf PCL       ; Se carga en "W" el siguiente caracter a pintar
;      dt  "",0        ; El fin del mensaje se señala con un "0"

Men7Lin1:              ; El mensaje queda definido por una tabla con "dt"
;      movlw $>>8      ; Se carga el BYTE de mayor peso de la direccion en
;      movwf PCLATH    ; "PCLATH", a traves del registro de trabajo (w)
;      movf  Cuenta,W  ; Se carga la posicion a leer de la tabla en "W"
;      addwf PCL       ; Se carga en "W" el siguiente caracter a pintar
;      dt  "",0        ; El fin del mensaje se señala con un "0"

Men7Lin2:              ; El mensaje queda definido por una tabla con "dt"
;      movlw $>>8      ; Se carga el BYTE de mayor peso de la direccion en
;      movwf PCLATH    ; "PCLATH", a traves del registro de trabajo (w)
;      movf  Cuenta,W  ; Se carga la posicion a leer de la tabla en "W"
;      addwf PCL       ; Se carga en "W" el siguiente caracter a pintar
;      dt  "",0        ; El fin del mensaje se señala con un "0"

Men8Lin1:              ; El mensaje queda definido por una tabla con "dt"
;      movlw $>>8      ; Se carga el BYTE de mayor peso de la direccion en
;      movwf PCLATH    ; "PCLATH", a traves del registro de trabajo (w)
;      movf  Cuenta,W  ; Se carga la posicion a leer de la tabla en "W"
;      addwf PCL       ; Se carga en "W" el siguiente caracter a pintar
;      dt  "",0        ; El fin del mensaje se señala con un "0"

Men8Lin2:              ; El mensaje queda definido por una tabla con "dt"
;      movlw $>>8      ; Se carga el BYTE de mayor peso de la direccion en
;      movwf PCLATH    ; "PCLATH", a traves del registro de trabajo (w)
;      movf  Cuenta,W  ; Se carga la posicion a leer de la tabla en "W"
;      addwf PCL       ; Se carga en "W" el siguiente caracter a pintar
;      dt  "",0        ; El fin del mensaje se señala con un "0"

;#####

SemiMen1:              ; El mensaje queda definido por una tabla con "dt"
      movlw $>>8      ; Se carga el BYTE de mayor peso de la direccion en
      movwf PCLATH    ; "PCLATH", a traves del registro de trabajo (w)
      movf  Cuenta,W  ; Se carga la posicion a leer de la tabla en "W"
      addwf PCL       ; Se carga en "W" el siguiente caracter a pintar
      dt  " MEMORIA VACIA ",0          ; El fin del mensaje se señala con un 0

SemiMen2:              ; El mensaje queda definido por una tabla con "dt"
      movlw $>>8      ; Se carga el BYTE de mayor peso de la direccion en
      movwf PCLATH    ; "PCLATH", a traves del registro de trabajo (w)
      movf  Cuenta,W  ; Se carga la posicion a leer de la tabla en "W"
      addwf PCL       ; Se carga en "W" el siguiente caracter a pintar
      dt  "Hum:  %      C",0          ; El fin del mensaje se señala con un 0

SemiMen3:              ; El mensaje queda definido por una tabla con "dt"
      movlw $>>8      ; Se carga el BYTE de mayor peso de la direccion en
      movwf PCLATH    ; "PCLATH", a traves del registro de trabajo (w)
      movf  Cuenta,W  ; Se carga la posicion a leer de la tabla en "W"
      addwf PCL       ; Se carga en "W" el siguiente caracter a pintar
      dt  "MEDIR",0          ; El fin del mensaje se señala con un 0

SemiMen4:              ; El mensaje queda definido por una tabla con "dt"
      movlw $>>8      ; Se carga el BYTE de mayor peso de la direccion en
      movwf PCLATH    ; "PCLATH", a traves del registro de trabajo (w)

```

## APÉNDICE B

```

movf  Cuenta,W          ; Se carga la posicion a leer de la tabla en "W"
addwf PCL               ; Se carga en "W" el siguiente caracter a pintar
dt "Salvar",0          ; El fin del mensaje se señala con un 0

SemiMen5:              ; El mensaje queda definido por una tabla con "dt"
movlw $>>8             ; Se carga el BYTE de mayor peso de la direccion en
movwf PCLATH           ; "PCLATH", a traves del registro de trabajo (w)
movf  Cuenta,W         ; Se carga la posicion a leer de la tabla en "W"
addwf PCL              ; Se carga en "W" el siguiente caracter a pintar
dt "Medida :          m",0 ; El fin del mensaje se señala con un 0

SemiMen6:              ; El mensaje queda definido por una tabla con "dt"
; movlw $>>8           ; Se carga el BYTE de mayor peso de la direccion en
; movwf PCLATH         ; "PCLATH", a traves del registro de trabajo (w)
; movf  Cuenta,W       ; Se carga la posicion a leer de la tabla en "W"
; addwf PCL            ; Se carga en "W" el siguiente caracter a pintar
; dt "",0              ; El fin del mensaje se señala con un 0

SemiMen7:              ; El mensaje queda definido por una tabla con "dt"
; movlw $>>8           ; Se carga el BYTE de mayor peso de la direccion en
; movwf PCLATH         ; "PCLATH", a traves del registro de trabajo (w)
; movf  Cuenta,W       ; Se carga la posicion a leer de la tabla en "W"
; addwf PCL            ; Se carga en "W" el siguiente caracter a pintar
; dt "",0              ; El fin del mensaje se señala con un 0

;#####
org 0x0700

SemiMen11:             ; El mensaje queda definido por una tabla con "dt"
movlw $>>8             ; Se carga el BYTE de mayor peso de la direccion en
movwf PCLATH           ; "PCLATH", a traves del registro de trabajo (w)
movf  Cuenta,W         ; Se carga la posicion a leer de la tabla en "W"
addwf PCL              ; Se carga en "W" el siguiente caracter a pintar
dt "MEMORIA COMPLETA",0 ; El fin del mensaje se señala con un 0

SemiMen12:             ; El mensaje queda definido por una tabla con "dt"
movlw $>>8             ; Se carga el BYTE de mayor peso de la direccion en
movwf PCLATH           ; "PCLATH", a traves del registro de trabajo (w)
movf  Cuenta,W         ; Se carga la posicion a leer de la tabla en "W"
addwf PCL              ; Se carga en "W" el siguiente caracter a pintar
dt "OCUPACION",0      ; El fin del mensaje se señala con un 0

SemiMen13:             ; El mensaje queda definido por una tabla con "dt"
movlw $>>8             ; Se carga el BYTE de mayor peso de la direccion en
movwf PCLATH           ; "PCLATH", a traves del registro de trabajo (w)
movf  Cuenta,W         ; Se carga la posicion a leer de la tabla en "W"
addwf PCL              ; Se carga en "W" el siguiente caracter a pintar
dt "MEDIR BORRA LEER",0 ; El fin del mensaje se señala con un 0

SemiMen14:             ; El mensaje queda definido por una tabla con "dt"
movlw $>>8             ; Se carga el BYTE de mayor peso de la direccion en
movwf PCLATH           ; "PCLATH", a traves del registro de trabajo (w)
movf  Cuenta,W         ; Se carga la posicion a leer de la tabla en "W"
addwf PCL              ; Se carga en "W" el siguiente caracter a pintar
dt "Guardada",0       ; El fin del mensaje se señala con un 0

SemiMen15:             ; El mensaje queda definido por una tabla con "dt"
; movlw $>>8           ; Se carga el BYTE de mayor peso de la direccion en
; movwf PCLATH         ; "PCLATH", a traves del registro de trabajo (w)
; movf  Cuenta,W       ; Se carga la posicion a leer de la tabla en "W"
; addwf PCL            ; Se carga en "W" el siguiente caracter a pintar
; dt "",0              ; El fin del mensaje se señala con un 0

SemiMen16:             ; El mensaje queda definido por una tabla con "dt"
; movlw $>>8           ; Se carga el BYTE de mayor peso de la direccion en
; movwf PCLATH         ; "PCLATH", a traves del registro de trabajo (w)
; movf  Cuenta,W       ; Se carga la posicion a leer de la tabla en "W"
; addwf PCL            ; Se carga en "W" el siguiente caracter a pintar
; dt "",0              ; El fin del mensaje se señala con un 0

SemiMen17:             ; El mensaje queda definido por una tabla con "dt"
; movlw $>>8           ; Se carga el BYTE de mayor peso de la direccion en
; movwf PCLATH         ; "PCLATH", a traves del registro de trabajo (w)
; movf  Cuenta,W       ; Se carga la posicion a leer de la tabla en "W"
; addwf PCL            ; Se carga en "W" el siguiente caracter a pintar
; dt "",0              ; El fin del mensaje se señala con un 0

```

## MEDIDOR DE DISTANCIA POR ULTRASONIDOS

```

;#####          TABLAS DE MENSAJES          #####
;#####          TABLAS ASOCIADAS AL TIEMPO MEDIDO          #####

TabLimA:
    movlw    $>>8                ; Se carga el BYTE de mayor peso de la direccion en
    movwf   PCLATH                ;"PCLATH", a traves del registro de trabajo (w)
    movf    Cuenta,W              ; Se carga la posicion a leer de la tabla en "W"
    addwf   PCL                    ; Se carga en "W" el siguiente caracter a pintar

    dt 007,008,009,010,012,015,017,020,023,029,035,040,046,052
;Tramo     1   2   3   4   5   6   7   8   9  10  11  12  13  14
           058,063,069,075,081,086,092,098,255
           ;15 16 17 18 19 20 21 22 23

TabLimB:
    movlw    $>>8                ; Se carga el BYTE de mayor peso de la direccion en
    movwf   PCLATH                ;"PCLATH", a traves del registro de trabajo (w)
    movf    Cuenta,W              ; Se carga la posicion a leer de la tabla en "W"
    addwf   PCL                    ; Se carga en "W" el siguiente caracter a pintar

    dt 045,118,180,241,044,019,248,224,186,108,046,229,162,106
;Tramo     1   2   3   4   5   6   7   8   9  10  11  12  13  14
           034,212,146,078,022,212,145,083,255
           ;15 16 17 18 19 20 21 22 23

Pendi:
    movlw    $>>8                ; Se carga el BYTE de mayor peso de la direccion en
    movwf   PCLATH                ;"PCLATH", a traves del registro de trabajo (w)
    movf    Cuenta,W              ; Se carga la posicion a leer de la tabla en "W"
    addwf   PCL                    ; Se carga en "W" el siguiente caracter a pintar

    dt 085,089,092,092,092,098,098,098,100,100,099,100,099,098
;Tramo     1   2   3   4   5   6   7   8   9  10  11  12  13  14
           100,100,099,099,098,099,099,098,099
           ;15 16 17 18 19 20 21 22 23

ConstanA:
    movlw    $>>8                ; Se carga el BYTE de mayor peso de la direccion en
    movwf   PCLATH                ;"PCLATH", a traves del registro de trabajo (w)
    movf    Cuenta,W              ; Se carga la posicion a leer de la tabla en "W"
    addwf   PCL                    ; Se carga en "W" el siguiente caracter a pintar

    dt 000,000,000,000,000,000,255,255,255,255,255,255,255,255
;Tramo     1   2   3   4   5   6   7   8   9  10  11  12  13  14
           255,255,255,255,000,255,255,000,255
           ;15 16 17 18 19 20 21 22 23

ConstanM:
    movlw    $>>8                ; Se carga el BYTE de mayor peso de la direccion en
    movwf   PCLATH                ;"PCLATH", a traves del registro de trabajo (w)
    movf    Cuenta,W              ; Se carga la posicion a leer de la tabla en "W"
    addwf   PCL                    ; Se carga en "W" el siguiente caracter a pintar

    dt 074,044,019,019,019,202,203,203,161,161,190,153,193,242
;Tramo     1   2   3   4   5   6   7   8   9  10  11  12  13  14
           138,136,200,201,021,198,198,034,191
           ;15 16 17 18 19 20 21 22 23

ConstanB:
    movlw    $>>8                ; Se carga el BYTE de mayor peso de la direccion en
    movwf   PCLATH                ;"PCLATH", a traves del registro de trabajo (w)
    movf    Cuenta,W              ; Se carga la posicion a leer de la tabla en "W"
    addwf   PCL                    ; Se carga en "W" el siguiente caracter a pintar

    dt 042,254,046,005,078,195,140,251,229,181,170,167,237,184
;Tramo     1   2   3   4   5   6   7   8   9  10  11  12  13  14
           017,168,198,038,179,085,022,113,047
           ;15 16 17 18 19 20 21 22 23

;#####          TABLAS ASOCIADAS AL TIEMPO MEDIDO          #####
;#####          TABLAS DE FACTORES          #####
;                ; Tablas que determinan el factor de velocidad dependiente
;                ; de la Temperatura, medida con FS = 1.5973 V
;#####

```





APÉNDICE B

dt	231,240,249,003,012,021,030,039,048,058,067,076,085,094,103,112,122,131,140,149,158,167,176,185,194,204,213,222,231,240,249,002,011,020,030,039,048,057,066,075		
	; 20 21	31	41
		51	
dt	084,093,102,111,120,129,138,148,157,166,175,184,193,202,211,220,229,238,247,000,009,018,027,036,045,054,063,072,081,090,099,108,117,126,135,144,153,162,171,180		
	; 60 61	71	81
		91	
dt	189,198,207,216,225,234,243,252,005,014,023,032,041,050,059,068,077,086,095,103,112,121,130,139,148,157,166,175,184,193,202,211,219,228,237,246,255,008,017,026		
	; 200 01	11	21
		31	
dt	035,044,052,061,070,079,088,097,106,115,124,132,141,150,159,168,177,186,194,203,212,221,230,239,247,000,009,018,027,036,044,053,062,071,080,089,097,106,115,124		
	; 40 41	51	61
		71	
dt	133,141,150,159,168,177,185,194,203,212,221,229,238,247,000,008,017,026,035,043,052,061,070,079,087,096,105,114,122,131,140,148,157,166,175,183,192,201,210,218		
	; 80 81	91	01
		11	
dt	227,236,244,253,006,015,023,032,041,049,058,067,076,084,093,102,110,119,128,136,145,154,162,171,180,188,197,206,214,223,232,240,249,002,010,019,028,036,045,053		
	; 20 21	31	41
		51	
dt	062,071,079,088,097,105,114,123,131,140,148,157,166,174,183,191,200,209,217,226,234,243,252,004,013,021,030,039,047,056,064,073,081,089,098,106,115,124,132,141		
	; 60 61	71	81
		91	
dt	150,158,167,176,184,193,201,210,218,227,235,244,252,005,013,022,031,039,048,056,065,073,082,090,099,107,116,124,133,141,150,158,167,175,184,192,201,209,218,226		
	; 400 01	11	21
		31	
dt	235,243,251,004,012,021,029,038,046,055,063,072,080,089,097,105,114,122,131,139,148,156,165,173,181,190,198,207,215,223,232,240,249,001,010,018,026,035,043,052		
	; 40 41	51	61
		71	
dt	060,068,077,085,094,102,110,119,127,136,144,152,161,169,177,186,194,203,211,219,228,236,244,253,005,013,022,030,038,047,055,063,072,080,089,097,105,114,122,130		
	; 80 81	91	01
		11	
dt	138,147,155,163,172,180,188,197,205,213,222,230,238,247,255,007,015,024,032,040,049,057,065,073,082,090,098,107,115,123,131,140,148,156,164,173,181,189,198,206		
	; 20 21	31	41
		51	
dt	214,222,231,239,247,255,008,016,024,032,040,049,057,065,073,082,090,098,106,115,123,131,139,147,156,164,172,180,188,197,205,213,220,229,237,245,253,005,013,022		
	; 60 61	71	81
		91	
dt	031,039,047,055,063,072,080,088,096,104,112,121,129,137,145,153,161,170,178,186,194,202,210,218,227,235,243,251,003,011,019,028,036,044,052,060,068,076,084,093		
	; 600 01	11	21
		31	
dt	101,109,117,125,133,141,149,157,165,174,182,190,198,206,214,222,230,238,246,254,007,015,023,031,039,047,055,063,071,079,087,095,103,111,120,128,136,144,152,160		
	; 40 41	51	61
		71	
dt	168,176,184,192,200,208,216,224,232,240,248,000,008,016,024,032,040,048,056,064,072,080,088,096,104,112,120,128,136,144,152,160,168,176,184,192,200,208,216,224		







APÉNDICE B

dt			
215, 221, 227, 233, 240, 246, 252, 002, 009, 015, 021, 027, 034, 040, 046, 052, 059, 065, 071, 078, 084, 090, 096, 103, 109, 115, 122, 128, 134, 141, 147, 153, 159, 166, 172, 178, 185, 191, 198, 204			
; 60 61		71	81
	91		
dt			
211, 217, 224, 230, 236, 243, 249, 000, 006, 013, 019, 025, 032, 038, 045, 051, 058, 064, 071, 077, 084, 090, 097, 103, 110, 116, 123, 129, 136, 142, 149, 155, 162, 169, 175, 182, 188, 195, 201, 208			
; 400 01		11	21
	31		
dt			
215, 221, 228, 234, 241, 248, 254, 005, 011, 018, 025, 031, 038, 045, 051, 058, 065, 071, 078, 085, 091, 098, 105, 112, 118, 125, 132, 138, 145, 152, 159, 165, 172, 179, 186, 192, 199, 206, 213, 220			
; 40 41		51	61
	71		
dt			
226, 233, 240, 247, 254, 004, 011, 018, 025, 032, 039, 045, 052, 059, 066, 073, 080, 087, 094, 100, 107, 114, 121, 128, 135, 142, 149, 156, 163, 170, 177, 184, 191, 198, 205, 212, 219, 226, 233, 240			
; 80 81		91	01
	11		
dt			
247, 254, 005, 012, 019, 026, 033, 040, 047, 054, 061, 068, 075, 082, 089, 096, 103, 111, 118, 125, 132, 139, 146, 153, 160, 168, 175, 182, 189, 196, 203, 211, 218, 225, 232, 239, 247, 254, 005, 012			
; 20 21		31	41
	51		
dt			
019, 027, 034, 041, 048, 056, 063, 070, 078, 085, 092, 099, 107, 114, 121, 129, 136, 143, 151, 158, 165, 173, 180, 187, 195, 202, 210, 217, 224, 232, 239, 247, 253, 005, 012, 020, 027, 034, 042, 049			
; 60 61		71	81
	91		
dt			
058, 065, 073, 080, 088, 095, 103, 110, 118, 125, 133, 140, 148, 155, 163, 170, 178, 186, 193, 201, 208, 216, 224, 231, 239, 246, 254, 006, 013, 021, 029, 036, 044, 052, 059, 067, 075, 082, 090, 098			
; 600 01		11	21
	31		
dt			
105, 113, 121, 129, 136, 144, 152, 160, 167, 175, 183, 191, 199, 206, 214, 222, 230, 238, 245, 253, 005, 013, 021, 029, 037, 044, 052, 060, 068, 076, 084, 092, 100, 108, 116, 124, 132, 140, 148, 155			
; 40 41		51	61
	71		
dt			
163, 171, 179, 187, 195, 203, 211, 219, 228, 236, 244, 252, 004, 012, 020, 028, 036, 044, 052, 060, 068, 077, 085, 093, 101, 109, 117, 125, 134, 142, 150, 158, 166, 175, 183, 191, 199, 207, 216, 224			
; 80 81		91	01
	11		
dt			
232, 240, 249, 001, 009, 017, 026, 034, 042, 051, 059, 067, 076, 084, 092, 101, 109, 117, 126, 134, 143, 151, 159, 168, 176, 185, 193, 202, 210, 218, 227, 235, 244, 252, 005, 013, 022, 030, 039, 047			
; 20 21		31	41
	51		
dt			
056, 065, 073, 082, 090, 099, 107, 116, 125, 133, 142, 150, 159, 168, 176, 185, 194, 202, 211, 220, 228, 237, 246, 254, 007, 016, 025, 033, 042, 051, 060, 068, 076, 085, 093, 102, 111, 120, 129, 138			
; 60 61		71	81
	91		
dt			
148, 157, 165, 174, 183, 192, 201, 210, 219, 228, 237, 245, 254, 007, 016, 025, 034, 043, 052, 061, 070, 079, 088, 097, 106, 115, 124, 133, 142, 152, 161, 170, 179, 188, 197, 206, 215, 224, 234, 243			
; 800 01		11	21
	31		
dt			
252, 005, 014, 023, 033, 042, 051, 060, 069, 079, 088, 097, 106, 116, 125, 134, 144, 153, 162, 171, 181, 190, 199, 209, 218, 228, 237, 246, 000, 009, 019, 028, 037, 047, 056, 066, 075, 085, 094, 104			
; 40 41		51	61
	71		
dt			
113, 123, 132, 142, 151, 161, 170, 180, 189, 199, 209, 218, 228, 237, 247, 001, 010, 020, 030, 039, 049, 059, 068, 078, 088, 097, 107, 117, 127, 136, 146, 156, 166, 175, 185, 195, 205, 215, 225, 234			
; 80 81		91	01
	11		
dt			
244, 254, 008, 018, 028, 038, 048, 057, 067, 077, 087, 097, 107, 117, 127, 137, 147, 157, 167, 177, 187, 197, 207, 217, 227, 238, 248, 002, 012, 022, 032, 042, 052, 063, 073, 083, 093, 103, 114, 124			

MEDIDOR DE DISTANCIA POR ULTRASONIDOS

---

```

; 20 21                                31                                41
                                51
dt
134,144,155,165,175,185,196,206,216,227,237,247,002,012,022,033,043,053,064,074,085,095,106,11
6,126,137,147,158,168,179,189,200,209,219,230,240,251,005,016,027
; 60 61                                71                                81
                                91
;#####
```

## **C APÉNDICE DE HOJAS DE ESPECIFICACIONES**

En este apéndice se encuentran las hojas de especificaciones de todos los componentes electrónicos empleados en el medidor de distancia y se numeran a continuación.

- ◆ Hojas de especificaciones del microcontrolador PIC16F87X
- ◆ Hojas de especificaciones de los transductores de ultrasonidos 400(T/R)160
- ◆ Hojas de especificaciones del sensor de temperatura LM35
- ◆ Hojas de especificaciones del sensor de humedad relativa HIH-3610
- ◆ Hojas de especificaciones del regulador de tensión L7800
- ◆ Hojas de especificaciones del regulador de tensión MAX6018
- ◆ Hojas de especificaciones del controlador del LCD HD44780
- ◆ Hojas de especificaciones del C.I. de puertas inversoras, HCF4069UB
- ◆ Hojas de especificaciones del C.I. con los amplificadores operacionales,  $\mu A747$
- ◆ Hojas de especificaciones del C.I. con los amplificadores operacionales de bajo consumo, LM358