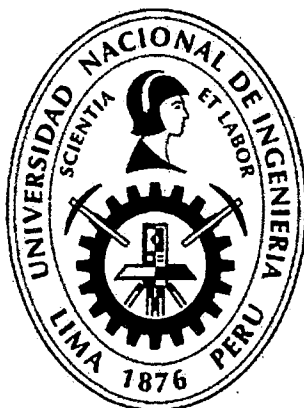


UNIVERSIDAD NACIONAL DE INGENIERÍA

Facultad de Ciencias

Escuela Profesional de Ingeniería Física



Tesis para optar el Título Profesional de:

Ingeniero Físico

Titulada:

Sistema de adquisición y registro de datos de temperatura,
humedad relativa y distancia

Presentada por:

Jorge Enrique Giraldo Mendoza

Asesor:

Jorge Luis Mírez Tarrillo

LIMA - PERU

2013

Digitalizado por:

Consortio Digital del
Conocimiento MebLatam,
Hemisferio y Dalse

*Quiero dedicar la culminación de esta
investigación a mis padres Eurípides y Betty
que me brindaron siempre su apoyo
incondicional*

Tabla de Contenidos

Tabla de Contenidos

Índice de Tablas

Índice de Figuras

Agradecimientos

Resumen

Capítulo 1: Introducción	1
1.1. Antecedentes	1
1.2. Declaración del problema	2
1.3. Objetivos	2
1.4. Definición de términos	3
Capítulo 2: Fundamento Teórico	5
2.1. Microcontrolador PIC16F876A	6
2.1.1. Puerto A	7
2.1.2. Puerto B	8
2.1.3. Puerto C	9
2.2. Conversión Análogo-Digital	10
2.3. Comunicación serial asincrónica RS232	12
2.4. Comunicación serial sincrónica I2C	13
2.5. Integrado MAX232	16
2.6. Timer 555 en modo multivibrador astable	18
2.7. Sensor de temperatura LM35	19
2.8. Sensor de humedad HS1101	19
2.9. Sensor ultrasónico UCM-R4OK1 con soporte SRF04	20
2.10. Memoria EEPROM	22
2.11. Pantalla LCD 2x16	23
2.12. Multiplexión	24
Capítulo 3: Parte experimental	26
3.1. Equipos y componentes	26

3.2. Sensores, amplificación y soporte	28
3.2.1. Sensor de humedad	28
3.2.2. Sensor de temperatura	29
3.2.3. Sensor ultrasónico	30
3.3. Procesamiento de la señal analógica	33
3.4. Visualización LCD	36
3.5. Comunicación serial con la EEPROM	38
3.6. Comunicación serial con la PC	41
3.7. Software para transferencia de datos	45
3.8. Multiplexión comunicación serial PIC – PIC	49
3.9. Resultados	51
Capítulo 4: Discusión	54
4.1. Procesamiento de datos	54
4.2. Correcciones e incertidumbres	55
4.3. Evaluación crítica de los resultados	57
4.4. Comparación de los resultados con otros equipos	59
Capítulo 5: Conclusiones	62
Apéndice A. Código Matlab asociado al entorno gráfico	64
Apéndice B. Código asociado al PIC 12F675 para la multiplexión	67
Apéndice C. Código asociado al cálculo de distancia	68
Apéndice D. Trazabilidad de los patrones	70
Apéndice E. Certificados de calibración del equipo elaborado	81
Apéndice F. Variante USB 2.0	88
Imágenes del equipo elaborado	94
Referencias	96

Índice de Tablas

1. Funciones del puerto A	7
2. Registros asociados al puerto A	7
3. Funciones del puerto B	8
4. Registros asociados al puerto B	8
5. Funciones del puerto C	9
6. Registros asociados al puerto C	10
7. Niveles lógicos	13
8. Configuración de frecuencia de conversión A/D	34
9. TAD vs máxima frecuencia de oscilador externo	35
10. Calibración de temperatura	52
11. Calibración de humedad relativa	52
12. Calibración de distancia	53
13. Costos unitarios de componentes	58
14. Radioshack sensor de temperatura interno	59
15. Radioshack sensor de temperatura externo	59
16. Radioshack higrómetro	59
17. Mannix termómetro	60
18. Mannix higrómetro	60
19. Inmetro termómetro	60
20. Inmetro higrómetro	61

Índice de Figuras

1. Microcontrolador PIC 16F876A, vista de los periféricos y pines de alimentación VDD: +5V y VSS: 0V	6
2. Diferencia entre la calidad de un muestreo de 8 y 16 bits	11
3. Formato marca espacio	12
4. Conexión del PIC con la EEPROM	14
5. Transferencia de datos en el bus I2C	15
6. Integrado MAX32, distribución de los pines de alimentación, condensadores y transmisión/recepción	17
7. Configuración del integrado MAX32	17
8. Configuración del Timer, en modo multivibrador astable	18
9. Configuración Timer, recomendada en el datasheet	20
10. Interior del sensor UCM-R4OK1	21
11. Diagrama de tiempos del SRF_04	21
12. Configuración externa de la pantalla LCD 2x16	23
13. Configuración del Timer, modificación del circuito recomendado en el datasheet	29
14. Configuración del OPAM y el LM35, amplificación de 0 a 5V	30
15. Circuito correspondiente al módulo SRF04	32
16. Registros ADCON0 y ADCON1	34
17. Conversión A/D bit por bit	35
18. Configuración PIC-LCD	36
19. Comunicación entre el PIC y el puerto serial usando el integrado MAX232	42
20. Nueva interface gráfica de usuario (GUI)	45
21. Entorno gráfico desarrollado en Matlab	46
22. Nuevo proyecto con la herramienta Deploytool	47
23. Creación del ejecutable con la herramienta Deploytool	47
24. Ejecutable adquiriendo datos de temperatura y humedad	48
25. Microcontrolador PIC 12F675, vista de los periféricos y pines de alimentación VDD: +5V y VSS: 0V	50
26. Multiplexión TDM entre Los PIC 12F675 y el PIC master 16F876A	51

Agradecimientos

En primer lugar, agradezco al Ingeniero Moisés Inga Chucos por abrirme las puertas de su laboratorio, en el cual hice las primeras pruebas de adquisición de datos de temperatura y humedad; me proporcionó los patrones con la respectiva documentación de INDECOPI, y los equipos necesarios para una adecuada calibración. También agradezco al Ingeniero Salvador Mogrovejo Barrera por permitirme usar su patrón de temperatura, y ayudarme en la calibración para las variantes del sistema de registro de datos.

Un agradecimiento especial al Ingeniero Ricardo Bernuy Casahuamán, quién me acogió en su casa y en su taller mecánico, sus consejos me ayudaron a avanzar con esta tesis a pesar del trabajo y las dificultades.

Agradezco al personal de UNIMETRO.SAC por su apoyo en las calibraciones, aún cuando estaba fuera de sus funciones y sus horas de trabajo; también al Ingeniero Jorge Mírez Tarrillo por sus consejos para la culminación de esta tesis.

Jorge Giraldo Mendoza

UNIVERSIDAD NACIONAL DE INGENIERIA

Julio 2013

Resumen

Los sistemas de registro de datos son necesarios en la industria para el monitoreo de los procesos de producción. Los primeros sistemas capaces de hacerlo fueron las tarjetas de adquisición de datos (DAQ's), la desventaja de éstas reside en que necesitan una computadora para su funcionamiento, su memoria volátil es incapaz de almacenar datos por sí misma. Ante esto tenemos el data logger, que convierte las señales analógicas en digitales, procesa la información, imprime el dato, y lo guardar por tiempo indefinido (sin energía) en la memoria; hasta pasarlos a la computadora. Esto le da una clara ventaja sobre la DAQ, ya que sólo se necesita encender el equipo para que registre datos, las horas o días que se programe.

El presente trabajo consiste en el diseño de un equipo que tiene ambas funciones, es capaz de adquirir datos online (como la DAQ) conectado a una computadora. El uso del protocolo serial asincrónico estándar RS232 entre la computadora y el equipo, permite enviar más de 9600 datos por segundo usando tres líneas, dos de datos y una de tierra. La distancia es otra ventaja de este tipo de comunicación, se puede poner sensores hasta a 30 metros del equipo (propio del protocolo) habiéndose probado 8 metros de forma experimental, esto permite monitorear ambientes distintos.

La función data logger es independiente de la computadora, en este caso se usó una memoria EEPROM 26LC256 para almacenar los datos, esta memoria se comunica con el microcontrolador usando el protocolo de comunicación serial sincrónica I2C, el microcontrolador usado es el PIC16F876A como registrador en serie con la EEPROM externa, es decir, adquiere las señales analógicas de temperatura, distancia y humedad; las convierte a datos digitales y guarda estos en la memoria externa. El equipo es capaz de medir temperatura, humedad y distancia haciendo uso de sensores adaptados al mismo. Es posible conectar varios sensores a la vez usando una multiplexión, es decir: combinar dos o más canales de información en un solo medio de transmisión. Se probó experimentalmente con 10 sensores de temperatura, pudiendo ser más; obteniéndose una resolución de 0.1 °C en cada caso.

Las incertidumbres obtenidas son:

Temperatura (LM35) : ± 0.5 °C evaluado de 0 °C a 50 °C

Humedad relativa (HS1101) : ± 4.0 % evaluado de 30 % RH a 80 % RH

Profundidad (UCM-R4OK1) : ± 0.5 cm evaluado de 3 cm a 300 cm

El complemento de este equipo es el software, que se realizó en Matlab; el programa fue diseñado como archivo *.m y posteriormente convertido a ejecutable con entorno gráfico, muestra en la pantalla el progreso de la captura y envía la data a Microsoft Excel, la lectura de la memoria EEPROM puede hacerse desde el software para pasar la data al disco duro, o manualmente si se desea visualizar en la pantalla LCD.

Palabras clave: Data logger, DAQ, serial, multiplexión, ADC, sensor.

Capítulo 1

Introducción

1.1 Antecedentes

En primer lugar hay que marcar la diferencia entre un sistema que adquiere y otro que registra datos, los sistemas de adquisición convierten una señal analógica en digital y requieren una PC (*Personal Computer*) para registrar los datos, un ejemplo de esto son las tarjetas de adquisición. Por otro lado, un equipo registrador de datos es independiente de la PC, este equipo puede adquirir y almacenar los datos ya que posee una memoria no volátil. Los primeros equipos de adquisición de datos se remontan a inicios de la década de los '70s y se asocian a los minicomputadores de Digital Equipment Corporation, cabe mencionar que en ese entonces la aplicación de los transistores y los circuitos integrados eran una novedad. Los primeros registradores usaron papel perforado y cintas magnéticas. La evolución de las tarjetas de adquisición fue lenta hasta la década de los '80s, en 1987 surge la tarjeta de sonido AdLib, la AdLib viene a ser una de las primeras DAQ's, usaba el integrado de sonido Yamaha YM3812, este integrado es un ADC (*Analog Digital Converter*) que producía sonido mediante la síntesis FM, la tarjeta se podía insertar en una PC estándar con slot ISA compatible.

De aquí en más surgió una gama de posibilidades para la adquisición de datos analógicos, actualmente pueden encontrarse ADC de hasta 64 bits, y como memoria para registradores se usan las EEPROM (*Electrically-Erasable Programmable Read-Only Memory*), memorias USB (*Universal Serial Bus*) entre otras. Entre las empresas que producen equipos registradores de datos encontramos a "Delta Ohm", "Fluke", "Lufft" entre otras, que alcanzan una gran exactitud, con una resolución de centésimas en temperatura (°C) y humedad relativa (% HR), y una precisión de centésimas en algunos casos. Estos equipos se usan como patrones para calibración, es decir: se usan como referencia para comprobar el buen funcionamiento de otros equipos, menos precisos y más comerciales por su bajo precio.

1.2 Declaración del problema

En la industria se requieren equipos de adquisición y registro para recopilar datos de diferentes parámetros físicos, como temperatura, humedad, presión, flujo, nivel, distancia, entre otros. Con la evolución de los equipos de adquisición de datos, la precisión ha aumentado; en el mercado podemos encontrar equipos desde 30 dólares hasta unos miles de dólares, dependiendo de las funciones y especificaciones de calidad mencionadas.

Por otro lado el desarrollo de un prototipo que pueda medir temperatura, humedad y distancia dentro de los parámetros comerciales es todo un reto para un recién egresado y una oportunidad para crear capacidades humanas y tecnológicas en este rubro de mediciones. Se pueden usar componentes comerciales para la elaboración, la dificultad radica en la estabilidad y la programación, tanto en el microcontrolador como en el software que usa la PC. Hay que tener, también, pleno conocimiento de la comunicación serial, pues es necesario comunicar un microcontrolador con la PC, y al mismo microcontrolador con una memoria externa. La calibración puede realizarse en empresas de metrología, de las cuales hay muchas en Lima.

1.3 Objetivos

- Crear un equipo capaz de adquirir datos de temperatura, humedad relativa y distancia, usando para esto el PIC 16F876A.
- Establecer comunicación serial entre PIC 16F876A y dispositivos externos para registrar y enviar información.
- Llegar a exactitud comercial, con correcciones menores a: temperatura ± 0.5 °C, humedad relativa ± 4.0 % RH, distancia ± 0.5 cm.
- Crear un ejecutable en el SO Windows (XP y 7) para leer la data registrada en forma serial.
- Reducir el costo unitario de producción a menos de 200 soles.

1.4 Definición de términos

En esta sección se exponen algunas definiciones que resultarán familiares para el lector, aún así, es necesario reforzarlas para comprender a cabalidad el proceso que une el mundo natural y el digital. Cabe señalar que a lo largo de la redacción se desarrollarán conceptos y definiciones nuevas a partir de estos términos base.

Temperatura: Es una medida de la actividad interna de la materia, dicho de otro modo, es una medida de la energía cinética de átomos y moléculas que componen un cuerpo. Se mide en Kelvin (K), que es el $1/273.16$ de la temperatura termodinámica del punto triple del agua.

Humedad relativa: La humedad relativa es la humedad que contiene una masa de aire en relación con la máxima humedad absoluta que podría admitir sin producirse condensación, conservando las mismas condiciones de temperatura y presión atmosférica. Esta es la forma más habitual de expresar la humedad ambiental. Se expresa en tanto por ciento (% RH).

Distancia: Es una magnitud escalar que mide la separación entre dos puntos, se mide en unidades de longitud, es decir en metros (m). El metro se define como la distancia que recorre la luz en el vacío durante un intervalo de $1/299\,792\,458$ de segundo.

Memoria RAM (*Random Access Memory*): Es en esta memoria donde se cargan todas las instrucciones que ejecuta el microprocesador. Es una memoria volátil, de lectura y escritura, es decir: la información se borra en ausencia de una fuente de energía.

Memoria ROM (*Read only memory*): Es la memoria que contiene el programa del microcontrolador y sus constantes, permite la lectura de la información, mas no su escritura. Es una memoria no volátil, independientemente de la presencia de una fuente de energía.

Microprocesador: El microprocesador es un circuito integrado que contiene la Unidad Central de Proceso (UCP), o procesador de un computador. La UCP está formada por

la Unidad de Control, que interpreta las instrucciones; y el Camino de Datos, que las ejecuta.

Microcontrolador: Es un circuito integrado programable que contiene todos los componentes de un computador, es decir: posee un microprocesador, memorias y pines para control de periféricos.

Mesurando: Magnitud particular sometida a una medición.

Transductor: Dispositivo que hace corresponder a una magnitud de entrada otra de salida según una ley determinada, un ejemplo de esto es el termopar.

Sensor: Elemento de un instrumento de medida que está directamente sometido a la acción de un mesurando, un ejemplo de esto, es el receptor fotoeléctrico de un espectrofotómetro.

Instrumento de medida: Dispositivo destinado a utilizarse para hacer mediciones, sólo o asociado a uno o varios dispositivos anexos.

Ultrasonido: Onda sonora cuya frecuencia se encuentra por encima del rango auditivo humano.

Calibración: Conjunto de operaciones que establecen (en condiciones especificadas) la relación entre los valores de una magnitud indicada por un instrumento de medida o un sistema de medida, o los valores representados por una medida materializada o por un material de referencia, y los valores correspondientes de esa magnitud realizados por patrones.

Capítulo 2

Fundamento Teórico

Los sistemas de adquisición de datos se utilizan para medir y registrar señales obtenidas, principalmente, a partir de transductores, y en algunos casos por medición directa de la señal eléctrica. Los sistemas de instrumentación se pueden clasificar en analógicos y digitales. Un sistema analógico se puede definir como una función continua en una gráfica voltaje versus tiempo. Los sistemas digitales manejan información en forma de pulsos discretos y continuos, cuya relación con el tiempo contiene información referente a la magnitud o naturaleza de la cantidad. Un sistema de adquisición de datos analógicos consta de algunos o todos los elementos siguientes:

- **Transductores:** Para la transformación de parámetros físicos en señales eléctricas.
- **Acondicionadores de señales:** Para la amplificación, modificación o selección de ciertas partes de estas señales.
- **Dispositivos de presentación visual:** Para monitoreo continuo de las señales de entrada o salida, puede ser un osciloscopio, el monitor de la PC entre otros.
- **Instrumentación de registro de datos:** Para guardar los datos de entrada y reproducirlos posteriormente para un análisis más detallado, puede ser el disco duro de la PC o una memoria USB entre otros.

El diseño de un data logger toma en consideración todos estos elementos en un solo equipo. Se usa un microcontrolador como unidad de procesamiento, como acondicionadores de señales se usa un OPAM (*operational amplifier*), un temporizador y otras componentes. En esta sección se hace una descripción de los componentes usados, cómo operan, para qué sirven, en qué rangos de tensión trabajan, entre otras características; mencionando en cada descripción cómo interactúan las componentes con el sistema.

2.1. Microcontrolador PIC 16F876A

Un microcontrolador es un circuito integrado que incluye en su interior las tres unidades principales de una computadora: unidad central de procesamiento, memoria y periféricos de entrada y salida. Para este microcontrolador, la unidad central de procesamiento o procesador opera a 4 MHz, contiene: registros, la unidad de control, la unidad aritmético-lógica, los buses y las instrucciones. La memoria se divide en RAM (*Random-Access Memory*), ROM (*Read-Only Memory*), EEPROM (*Electrically-Erasable Programmable Read-Only Memory*) y FLASH, de donde proviene la F del 16F876A, el poseer memoria FLASH nos permite borrar y grabar el PIC las veces que sean necesarias. Además este microcontrolador posee 28 pines divididos en 3 puertos de entrada/salida (para control de periféricos) y alimentación. La distribución de pines puede verse en Fig.1.

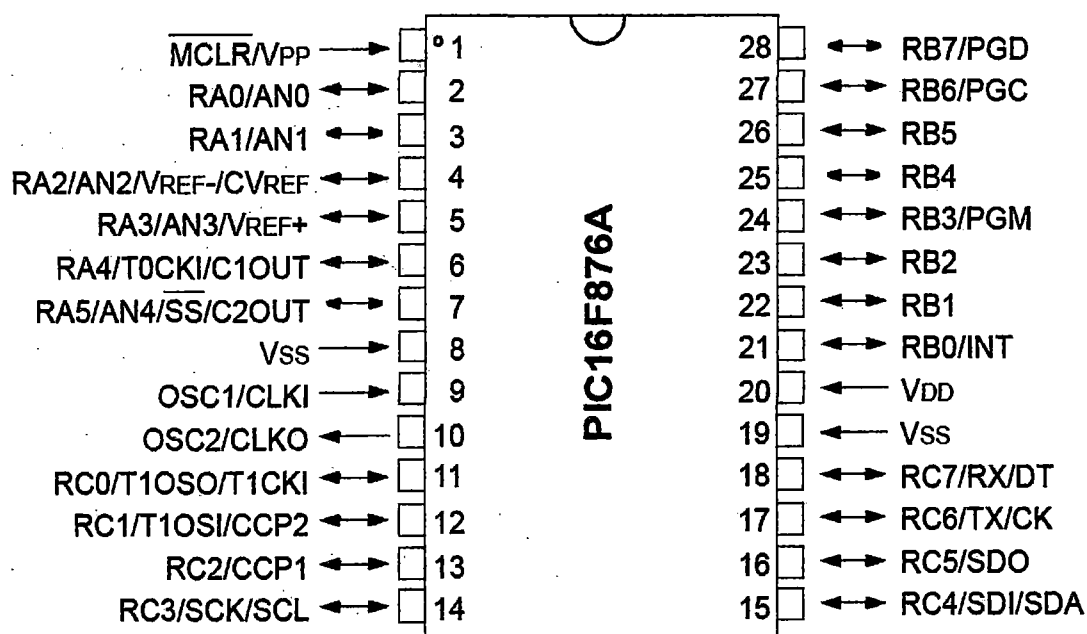


Fig.1: Microcontrolador PIC 16F876A, vista de los periféricos y pines de alimentación VDD: +5 V y VSS: 0 V [1]

Sobre los pines de alimentación, hay que aclarar el hecho de que un microcontrolador puede trabajar a diferentes voltajes, siempre con voltaje continuo; así por ejemplo en la serie 16F87XA el voltaje de alimentación puede ser desde 2 V hasta 5.5 V, siendo este voltaje el que representa al 1 lógico y 0 V representa al 0 lógico. Esto corresponde a la

tecnología CMOS (*Complementary metal-oxide-semiconductor*). Las principales características de esa familia lógicas son: bajo consumo de potencia, robustos frente al ruido o degeneración de la señal, son sencillos de diseñar y tienen bajo costo. Los demás pines del PIC corresponden a puertos analógicos y digitales, es necesario analizarlos de forma independiente.

2.1.1. Puerto A

Como se puede apreciar en la Tabla 1, el puerto A puede funcionar como entrada/salida digital o como entrada analógica con conversor A/D de 10 bits, siendo el único puerto con esta cualidad, además el pin RA4 funciona como gatillo para activar la interrupción interna del Timer 0, el pin RA5 funciona también para comunicación serial SPI, éste protocolo se usa, por ejemplo, en comunicación serial con potenciómetros digitales y sensores, pero no lo usamos en nuestro caso. Lo que si usaremos es la entrada analógica por el pin RA0 para temperatura, de modo que configuramos el puerto A como entrada analógica, los registros respectivos se pueden ver en la Tabla 2:

Tabla 1: Funciones del puerto A [1]

Name	Bit#	Buffer	Function
RA0/AN0	bit 0	TTL	Input/output or analog input.
RA1/AN1	bit 1	TTL	Input/output or analog input.
RA2/AN2/VREF-/CVREF	bit 2	TTL	Input/output or analog input or VREF- or CVREF.
RA3/AN3/VREF+	bit 3	TTL	Input/output or analog input or VREF+.
RA4/T0CKI/C1OUT	bit 4	ST	Input/output or external clock input for Timer0 or comparator output. Output is open-drain type.
RA5/AN4/SS/C2OUT	bit 5	TTL	Input/output or analog input or slave select input for synchronous serial port or comparator output.

Legend: TTL = TTL input, ST = Schmitt Trigger input

Tabla 2: Registros asociados al puerto A [1]

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other Resets
05h	PORTA	—	—	RA5	RA4	RA3	RA2	RA1	RA0	--0x 0000	--0u 0000
85h	TRISA	—	—	PORTA Data Direction Register						--11 1111	--11 1111
9Ch	CMCON	C2OUT	C1OUT	C2INV	C1INV	CIS	CM2	CM1	CM0	0000 0111	0000 0111
9Dh	CVRCON	CVREN	CVROE	CVRR	—	CVR3	CVR2	CVR1	CVR0	000- 0000	000- 0000
9Fh	ADCON1	ADFM	ADCS2	—	—	PCFG3	PCFG2	PCFG1	PCFG0	00-- 0000	00-- 0000

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by PORTA.


```

TRISA.0 = 1           'Se configure RA0 como entrada
ADCON1 = %10000001  'Se configura RA0 como entrada analógica

```

2.1.2. Puerto B

El puerto B puede funcionar como entrada/salida digital según la Tabla 3, los pines RB0, RB4:RB7 pueden funcionar como interrupción externa, las interrupciones usan el registro OPTION_REG y hacen que el programa salte de donde se encuentre hacia una etiqueta de interrupción ISR (*Interrupt service Routine*), cuando se termina de ejecutar el ISR retornamos a la sección del programa en donde se produjo la interrupción; también hay interrupciones internas que se dan por desbordamiento, estas son para los temporizadores: Timer 0, Timer 1 y Timer 2. Otra ventaja del puerto B que se usa en éste proyecto es la comunicación serial asincrónica vía RS232, ésta se usa para enviar y recibir datos de la computadora. Los registros necesarios para las interrupciones (en especial OPTION_REG) están en la Tabla 4.

Tabla 3: Funciones del puerto B [1]

Name	Bit#	Buffer	Function
RB0/INT	bit 0	TTL/ST ⁽¹⁾	Input/output pin or external interrupt input. Internal software programmable weak pull-up.
RB1	bit 1	TTL	Input/output pin. Internal software programmable weak pull-up.
RB2	bit 2	TTL	Input/output pin. Internal software programmable weak pull-up.
RB3/PGM ⁽³⁾	bit 3	TTL	Input/output pin or programming pin in LVP mode. Internal software programmable weak pull-up.
RB4	bit 4	TTL	Input/output pin (with interrupt-on-change). Internal software programmable weak pull-up.
RB5	bit 5	TTL	Input/output pin (with interrupt-on-change). Internal software programmable weak pull-up.
RB6/PGC	bit 6	TTL/ST ⁽²⁾	Input/output pin (with interrupt-on-change) or in-circuit debugger pin. Internal software programmable weak pull-up. Serial programming clock.
RB7/PGD	bit 7	TTL/ST ⁽²⁾	Input/output pin (with interrupt-on-change) or in-circuit debugger pin. Internal software programmable weak pull-up. Serial programming data.

Legend: TTL = TTL Input, ST = Schmitt Trigger input

Tabla 4: Registros asociados al puerto B [1]

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other Resets
06h, 106h	PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xxxx	uuuu uuuu
86h, 186h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111
81h, 181h	OPTION_REG	RBPU	INTEDG	TOCS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111

Legend: x = unknown, u = unchanged. Shaded cells are not used by PORTB.

2.1.3. Puerto C

El puerto C se usa principalmente para comunicación serial sincrónica (envío de datos en tiempo real) y asincrónica (envío de datos en diferido con velocidad preestablecida), para la comunicación serial sincrónica se requiere un pin para el clock y uno por donde viajan los datos, ya sea bajo el protocolo I2C (*Inter-Integrated Circuit*) o el SPI (*Serial Peripheral Interface*). En la Tabla 5 podemos ver que el pin RC3 puede funcionar como clock para comunicación SPI o I2C, mas no funciona como entrada/salida digital, algo similar pasa con RC4 que puede ser entrada de datos para comunicación SPI y entrada/salida para comunicación I2C, pero no puede ser entrada/salida digital. Los pines RC6 y RC7 en cambio si pueden ser entrada/salida digitales, además pueden ser transmisores o receptores de comunicación USART (*Universal Synchronous/Asynchronous Receiver Transmitter*), esta comunicación implica Tx/Rx del estándar asincrónico RS232, el RS485, el RS422 y data/clock de las sincrónicas mencionadas SPI e I2C. No hay un orden establecido, RC6 puede ser Tx, Rx, clock o data; del mismo modo RC7.

Tabla 5: Funciones del puerto C [1]

Name	Bit#	Buffer Type	Function
RC0/T1OSO/T1CKI	bit 0	ST	Input/output port pin or Timer1 oscillator output/Timer1 clock input.
RC1/T1OSI/CCP2	bit 1	ST	Input/output port pin or Timer1 oscillator input or Capture2 input/Compare2 output/PWM2 output.
RC2/CCP1	bit 2	ST	Input/output port pin or Capture1 input/Compare1 output/PWM1 output.
RC3/SCK/SCL	bit 3	ST	RC3 can also be the synchronous serial clock for both SPI and I ² C modes.
RC4/SDI/SDA	bit 4	ST	RC4 can also be the SPI data in (SPI mode) or data I/O (I ² C mode).
RC5/SDO	bit 5	ST	Input/output port pin or Synchronous Serial Port data output.
RC6/TX/CK	bit 6	ST	Input/output port pin or USART asynchronous transmit or synchronous clock.
RC7/RX/DT	bit 7	ST	Input/output port pin or USART asynchronous receive or synchronous data.

Legend: ST = Schmitt Trigger input

Los pines RC1 y RC2 sirven de entrada/salida digitales, y también para modulación de ancho de pulso (PWM), en el que ambos pines son entradas y pueden comparar las señales. Los registros de entrada y salida digital se encuentran en la Tabla 6, no hay registros de comunicación serial, ya que esta comunicación es una transmisión y recepción ordenada de pulsos, solo se requiere entradas y salidas digitales.

Tabla 6: Registros asociados al puerto C [1]

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other Resets
07h	PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	uuuu uuuu
87h	TRISC	PORTC Data Direction Register								1111 1111	1111 1111

Legend: x = unknown, u = unchanged

2.2. Conversión Análogo-Digital

La digitalización o conversión analógica-digital (A/D) consiste en tomar medidas de la amplitud (tensión) de una señal en forma periódica, asignar estos valores a un conjunto finito de niveles preestablecidos de tensión (conocidos como niveles de cuantificación) y registrarlos como números enteros en cualquier tipo de memoria o soporte. La conversión A/D también es conocida por el acrónimo inglés ADC. Es común que la amplitud de señal a digitalizar esté fuera del rango de entrada del conversor A/D, por esta razón es necesaria una etapa de acondicionamiento de la señal, se recomienda también hacer un filtro en esta etapa. Hablemos ahora de los cuatro procesos que intervienen en la conversión A/D:

Muestreo: Consiste en tomar muestras periódicas de la amplitud de onda. La velocidad con que se toma esta muestra, es decir, el número de muestras por segundo es lo que se conoce como frecuencia de muestreo.

Retención: Las muestras tomadas han de ser retenidas por un circuito de retención, el tiempo suficiente para permitir evaluar su nivel (cuantificación). Desde el punto de vista matemático este proceso no se contempla, ya que se trata de un recurso técnico debido a limitaciones prácticas, y carece, por tanto, de modelo matemático.

Cuantificación: En el proceso de cuantificación se mide el nivel de voltaje de cada una de las muestras. Consiste en asignar un margen de valor de una señal analizada a un único nivel de salida. Una palabra (número de bits del conversor A/D) más larga proporciona una mejor calidad en la conversión (de 8, 16 bits o 32 bits), es decir,

mientras más larga la palabra mayor la semejanza con la señal analógica, en Fig. 2 se muestra gráficamente la etapa de cuantificación.

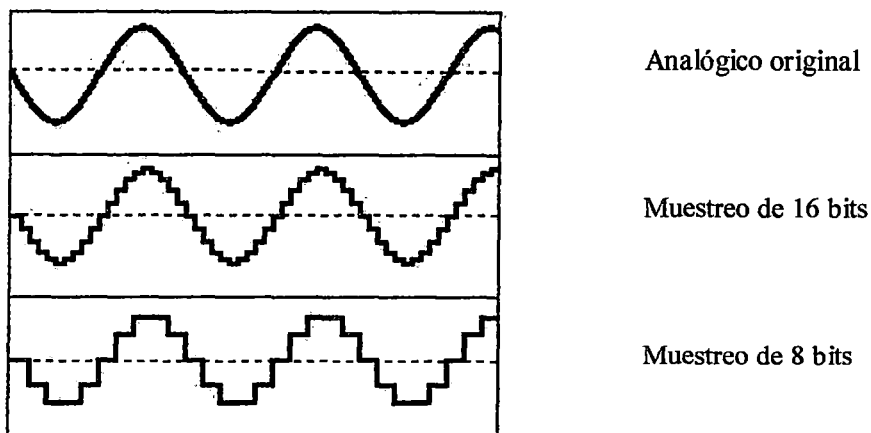


Fig.2: Diferencia entre la calidad de un muestreo de 8 y 16 bits

Codificación: La codificación consiste en traducir los valores obtenidos durante la cuantificación al código binario. Hay que tener presente que el código binario es el más utilizado, pero no es el único.

Durante el muestreo y la retención, la señal aún es analógica, puesto que aún puede tomar cualquier valor. No obstante, a partir de la cuantificación, cuando la señal toma valores finitos, podemos decir que ya es digital.

En el caso del PIC 16F876A así como en los demás PIC's de la familia 16F87X el conversor A/D es de 10 bits, para una entrada de tensión que va de 0 V a 5 V, ya que el número de estados (N) equivale a: $2^{10} - 1$ (n número de bits, 2 por el sistema binario), el número de estados disponibles será 1023. La resolución del ADC es igual a la tensión de entrada dividida por el número de estados disponibles:

$$Resolución = \frac{V_{ref}}{2^{10}-1} = \frac{5V}{2^{10}-1} = 4.88 mV$$

Según como se haya programado el PIC al capturar el dato este procede a efectuar el cálculo, el paso siguiente es enviar este nuevo dato vía serial TTL.

2.3. Comunicación Serial Asincrónica RS232

La comunicación serial consiste en el envío de un bit de información de manera secuencial, un bit a la vez y a un ritmo establecido entre el emisor y el receptor. Por otro lado, la transmisión de datos en paralelo es mucho más rápida que la transmisión serial, la razón principal por la cual se usa más la comunicación en serie es porque requiere un hilo para transmisión y otro para recepción, en comparación con la comunicación en paralelo que requiere 8. Otro punto a favor de la comunicación en serie es el alto grado de estandarización que permite la comunicación entre equipos, por ejemplo mediante RS232, USB o FireWire.

En la comunicación asincrónica, los datos se envían en intervalos de tiempo irregulares, como los caracteres en el teclado que se introducen en tiempo real. Para diferenciar la ubicación de los bits de información dentro del Byte de envío, es necesario establecer un indicador de inicio de transmisión o bit de inicio y un indicador de finalización o bit de finalización.

Dentro de las múltiples posibilidades existentes nos centraremos en la comunicación serial a través de la interpretación de dos niveles lógicos de tensión o corriente, denominado **formato marca/espacio**, que puede apreciarse en Fig.3. El nivel lógico "1" representa un estado de tensión o corriente denominado marca, el nivel lógico "0" representa un estado de tensión o corriente denominado espacio.

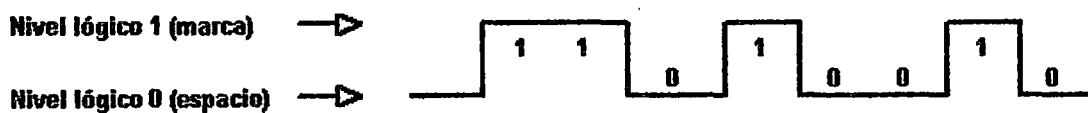


Fig.3: Formato marca espacio

Existen varios estándares que usan el formato marca/espacio, los que nos interesan se encuentran en la Tabla 7. El uso de estos formatos depende, principalmente, de la distancia que sea necesaria transmitir la información.

Tabla 7: Niveles lógicos [13]

Estándar	Nivel Lógico "1" (Marca)	Nivel Lógico "0" (Espacio)
TTL	5 V	0 V
Lazo 20 mA	20 mA	0 mA
RS232C	-3 V a -15 V	+3 V a +15 V

Usamos el nivel TTL en los microcontroladores como los PIC, hay otros microcontroladores que tienen otros niveles lógicos. La comunicación serial de nivel TTL no es comprendida por la PC, esta entiende el RS232 de modo que se necesita un transversor o adaptador de nivel como el MAX232.

El RS232 puede transmitir los datos en grupos de 5, 6, 7 u 8 bits, a velocidades determinadas que pueden ser 2400, 4800, 9600, etc. baudios por segundo. Después de la transmisión de los datos, le sigue un bit opcional de paridad que indica si el número de bits transmitidos es par o impar, y después 1 o 2 bits de parada. Normalmente, el protocolo utilizado es 8N1 (que significa, 8 bits de datos, sin paridad y con 1 bit de parada). Una vez que ha comenzado la transmisión de un dato, los bits tienen que llegar uno detrás de otro a una velocidad constante y en determinados instantes de tiempo. Por eso se dice que el RS232 es asíncrono por carácter y síncrono por bit. Los pines que portan los datos son RXD y TXD. Las demás se encargan de otros trabajos, DTR indica que el ordenador esta encendido, DSR que el aparato conectado a dicho puerto esta encendido, RTS que el ordenador puede recibir datos (porque no está ocupado), CTS que el aparato conectado puede recibir datos, y DCD detecta que existe una comunicación, presencia de datos.

2.4. Comunicación Serial Sincrónica I2C

En una comunicación sincrónica, el transmisor y el receptor están sincronizados con el mismo reloj, el receptor recibe continuamente la información a la misma velocidad que el transmisor la envía, incluso cuando no hay transmisión de bits. Es por este motivo que el receptor y el transmisor están sincronizados a la misma velocidad. Además, se inserta información suplementaria para garantizar que no se produzcan errores durante la transmisión.

En el transcurso de la transmisión sincrónica, los bits se envían sucesivamente sin que exista una separación entre cada carácter, por eso es necesario insertar elementos de sincronización; esto se denomina sincronización al nivel de los caracteres y es otra diferencia con la comunicación asincrónica.

Dentro de los varios tipos de comunicación serial sincrónica encontramos los de dos hilos como el protocolo I2C y SMBus, y los de tres hilos como el SPI; siendo el I2C el que usa la EEPROM externa para comunicarse con el PIC. La principal característica de I2C es que utiliza dos líneas para transmitir la información, una para los datos y por otra la señal de reloj. También es necesaria una tercera línea, pero esta sólo es la referencia (tierra); como suelen comunicarse circuitos en una misma placa que comparten una misma tierra esta tercera línea no suele ser necesaria. Las líneas se llaman: SDA (datos), SCL (reloj) y GND (tierra). Las dos primeras líneas son drenador abierto, por lo que necesitan resistencias de pull-up, los dispositivos conectados al bus I2C tienen una dirección única para cada uno. También pueden ser maestros o esclavos. La configuración PIC - EEPROM se muestra en Fig.4:

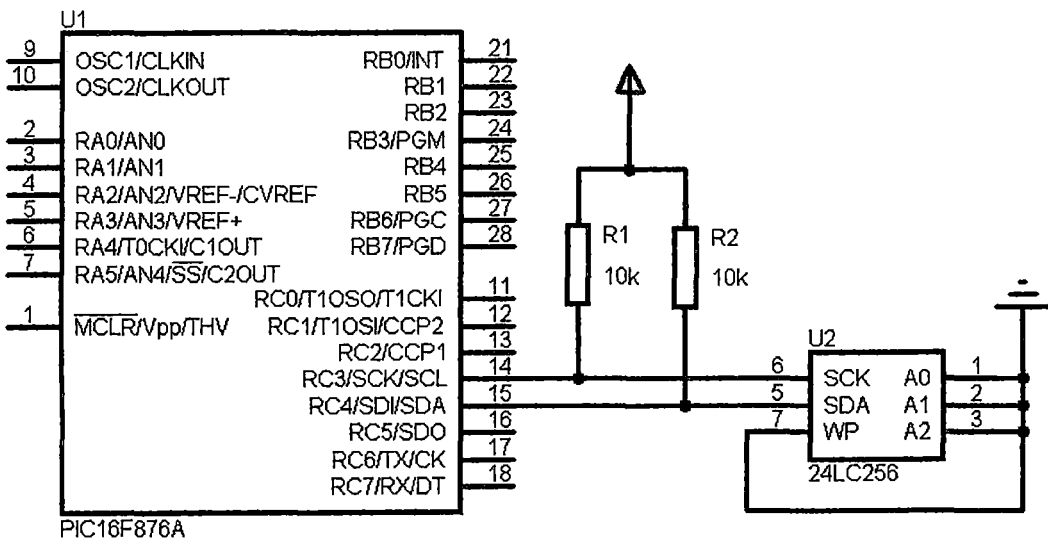


Fig.4: Conexión del PIC con la EEPROM

El dispositivo maestro inicia la transferencia de datos y además genera la señal de reloj, pero no es necesario que el maestro sea siempre el mismo dispositivo, esta característica se la pueden ir pasando a los dispositivos que tengan esa capacidad. Esta característica hace que al bus I2C se le denomine bus multimaestro. Las transacciones en el bus I2C tienen el formato que se muestra en Fig.5:

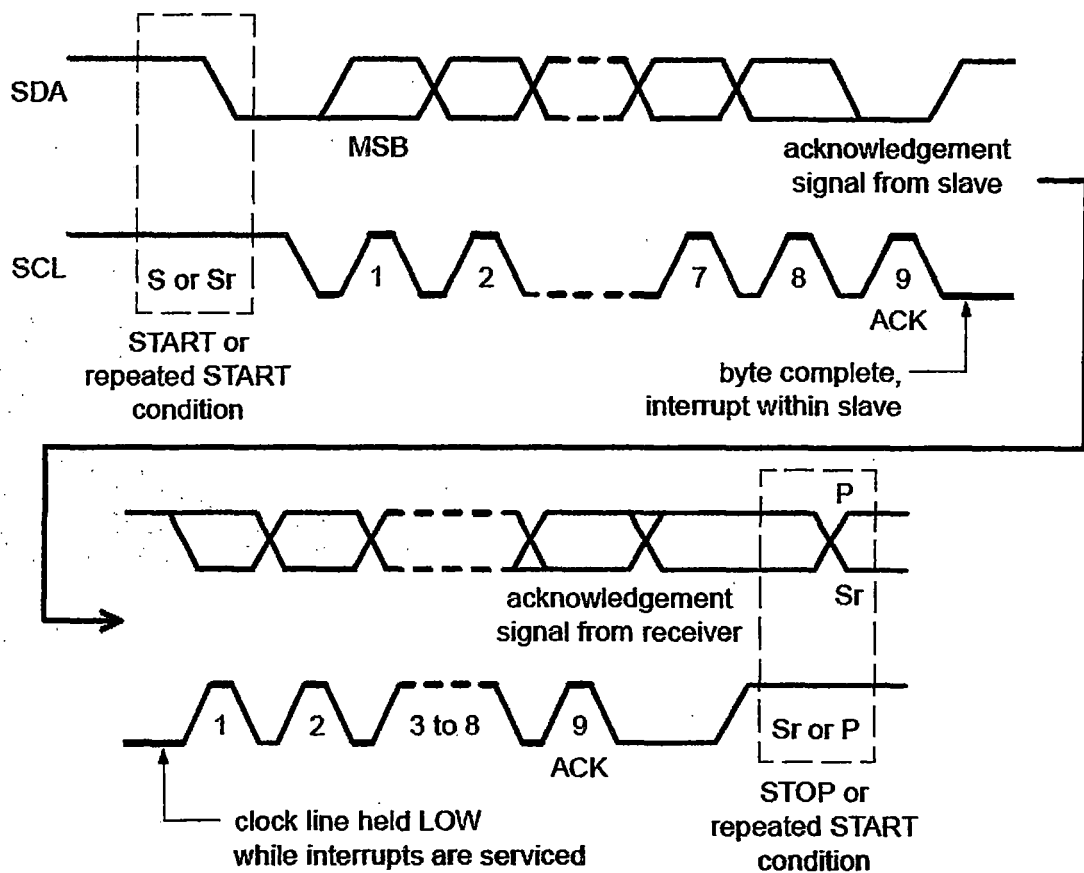


Fig.5: Transferencia de datos en el bus I2C [9]

Analicemos el envío de un dato según Fig.5:

| start | 1 2 3 4 5 6 7 | R/W | ACK | ... DATA ... | ACK | stop | idle |

- El bus esta libre cuando SDA y SCL están en estado lógico alto.
- Con el bus libre, cualquier dispositivo puede ocupar el bus I2C como maestro.
- El maestro comienza la comunicación enviando un patrón llamado "start condition", esto alerta a los dispositivos esclavos, poniéndolos a la espera de una transacción.
- El maestro se dirige al dispositivo con el que quiere comunicarse, enviando un byte que contiene los siete bits (1 - 7) que componen la dirección del dispositivo esclavo con el que se quiere comunicar, y el octavo bit (8 = R/W) de menor peso se corresponde con la operación deseada (L/E), lectura = 1 (recibir del esclavo) y escritura = 0 (enviar al esclavo).

- La dirección enviada es comparada por cada esclavo del bus con su propia dirección, si ambas coinciden, el esclavo se considera direccionado como esclavo-transmisor o esclavo-receptor dependiendo del bit R/W.
- El esclavo responde enviando un bit de ACK (acknowledgement), el cual indica al maestro que el dispositivo está en condiciones de comunicarse.
- Seguidamente comienza el intercambio de información entre los dispositivos.
- El maestro envía la dirección del registro interno del dispositivo que se desea leer o escribir.
- El esclavo responde con otro bit de ACK
- Ahora el maestro puede empezar a leer o escribir bytes de datos. Todos los bytes de datos deben constar de 8 bits, el número máximo de bytes que pueden ser enviados en una transmisión no está restringido, siendo el esclavo quien fija esta cantidad de acuerdo a sus características.
- Cada byte leído/escrito por el maestro debe ser obligatoriamente reconocido por un bit de ACK por el dispositivo maestro/esclavo.
- Se repiten los 2 pasos anteriores hasta finalizar la comunicación entre maestro y esclavo.
- Aun cuando el maestro siempre controla el estado de la línea del reloj, un esclavo de baja velocidad o que deba detener la transferencia de datos mientras efectúa otra función, puede forzar la línea SCL a nivel bajo. Esto hace que el maestro entre en un estado de espera, durante el cual, no transmite información esperando a que el esclavo esté listo para continuar la transferencia en el punto donde había sido detenida.
- Cuando la comunicación finaliza, el maestro transmite una "stop condition" para dejar libre el bus.
- Después de la "stop condition", es obligatorio para el bus, estar libre durante unos microsegundos (idle).

2. 5. Integrado MAX232

El MAX232 es un circuito integrado que convierte los niveles de las líneas de un puerto serie RS232 a niveles TTL y viceversa. Lo interesante es que sólo necesita una

alimentación de 5 V, ya que convierte internamente los valores de tensión que son necesarios para el estándar RS232 (-5 V y 5 V). Otros integrados que trabajan con RS232 requieren dos voltajes, +12 V y -12 V. Podemos ver en Fig.6 la distribución de pines, y en Fig.7 la configuración recomendada por el datasheet.

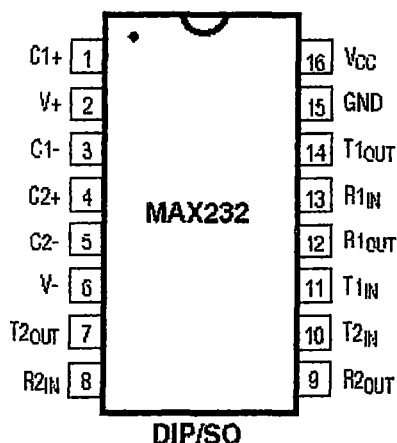


Fig.6: Integrado MAX32, distribución de los pines de alimentación, condensadores y transmisión/recepción. [8]

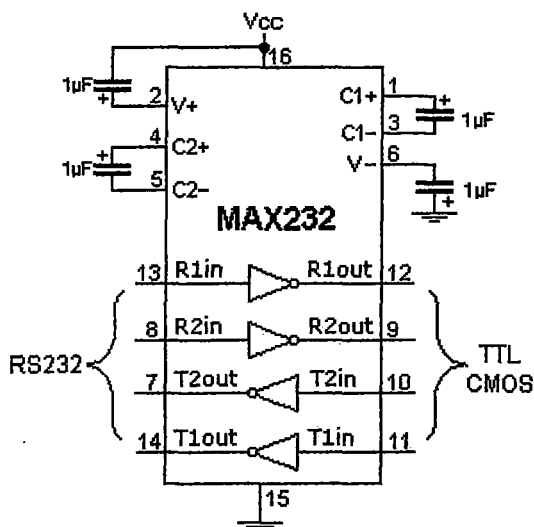


Fig.7: Configuración del integrado MAX32. [8]

El MAX232 tiene integrada la electrónica necesaria para lograr la comunicación entre el puerto serie de una PC y cualquier otro circuito con funcionamiento en base a señales de nivel TTL/CMOS. Este integrado posee dos convertidores de nivel TTL a RS232 y otros dos que, a la inversa, convierten de RS232 a TTL, estos convertidores son suficientes para manejar las cuatro señales más utilizadas del puerto serie del PC, que son TX, RX, RTS y CTS.

TX es la señal de transmisión de datos, RX es la de recepción, y RTS y CTS se utilizan para establecer el protocolo para el envío y recepción de los datos.

2.6. Timer 555 en modo multivibrador astable

El Timer 555 es uno de los circuitos integrados más versátiles, fue puesto en el mercado en 1972 por Signetics (ahora Philips). Es un multivibrador muy barato, se usa como generador de onda cuadrada, disparador, oscilador controlado por tensión (VCO), conversor DC-DC, y hasta como demodulador de señal AM. En Fig.8 podemos ver su configuración en modo multivibrador astable.

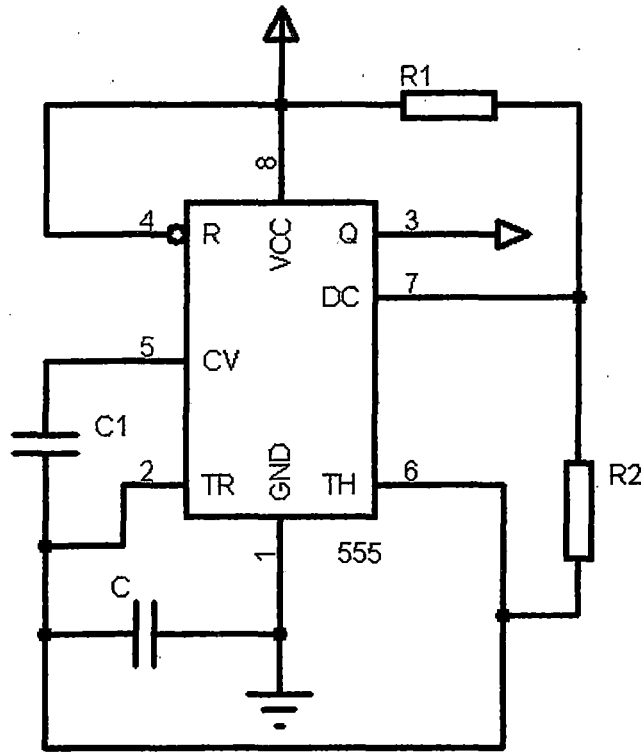


Fig.8: Configuración del Timer, en modo multivibrador astable

En modo astable es capaz de generar frecuencias de hasta 1 MHz, esta frecuencia F depende de las resistencias R1, R2 y el condensador C.

$$F = \frac{1.44}{(R1 + 2xR2)C}$$

2.7. Sensor de temperatura LM35

Sus características más relevantes son:

- Salida de tensión de 10 mV/°C
- Amplio rango de funcionamiento (desde -55 °C a 150 °C).
- Incertidumbre típica de 0.25 °C
- Incertidumbre de 0.75 °C para extremos cercanos a -55 °C y 150 °C
- Baja corriente de alimentación (60 µA)
- Baja impedancia de salida.
- Bajo costo (S/. 4,50).

Este integrado tiene una tensión de salida proporcional a la temperatura, en la escala Celsius. No necesita calibración externa y es de bajo costo. Funciona en el rango de alimentación comprendido entre 4 V y 30 V. Como ventaja adicional, el LM35 no requiere acondicionamiento externo cuando se desea obtener una precisión del orden de ± 0.25 °C a temperatura ambiente, y ± 0.75 °C para temperaturas cercanas a -55 °C y 150 °C. La baja impedancia de salida, su salida lineal y su precisa calibración inherente hace posible una fácil instalación en un circuito de control. Debido a su baja corriente de alimentación (60 µA), el calentamiento es muy reducido (menos de 0.1 °C sin ventilación).

2.8. Sensor de humedad HS1101

El sensor de humedad HS1101 es una solución efectiva cuando se desea medir la humedad relativa. El diseño del sensor está basado en una celda capacitiva; por lo que se recomienda usar en un circuito RC o un Timer 555, este último ofrece una salida cuadrada que es fácil de entender para cualquier microcontrolador.

Características principales:

- Tiempo de respuesta, corto

- Requerimiento de potencia 5 V a 10 V
- Salida análoga variando la capacitancia en respuesta al cambio de humedad
- Rango de medición de humedad: 1 % RH a 99 % RH
- Temperatura de operación: 0 °C a 100 °C

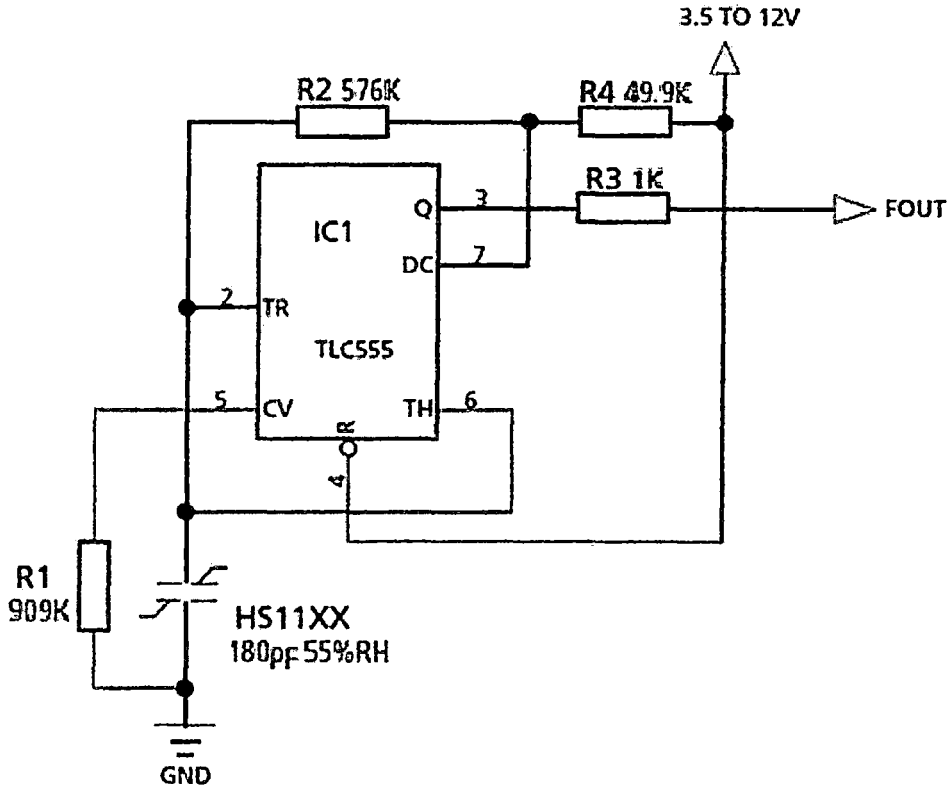


Fig.9: Configuración del Timer, recomendada en el datasheet [6]

Regla de correspondencia entre frecuencia F (Hz) y humedad relativa RH (%) según el datasheet [6]:

$$F = 6660x(1.1038 - 1.9368x10^{-3}xRH + 3.0114x10^{-6}xRH^2 - 3.440310^{-8}xRH^3)$$

2.9. Sensor ultrasónico UCM-R4OK1 con soporte SRF04

Para hablar de este sensor hay que entender primero, qué es el ultrasonido. El ser humano tiene un rango de audición en frecuencia, desde 20 Hz hasta 20 kHz,

dependiendo de cada persona. Las frecuencias inferiores a los 20 Hz entran en el rango de infrasonidos y las mayores de 20 kHz se encuentran en el rango de ultrasonidos. El sensor UCM-R4OK1 consiste en un emisor y un receptor, opera a 40 kHz correspondiente al ultrasonido; su componente principal es un resonador piezoeléctrico como se ve en Fig.10, el cual resuena con el aire a 40 kHz teniendo como entrada una onda cuadrada de amplitud 12 V.

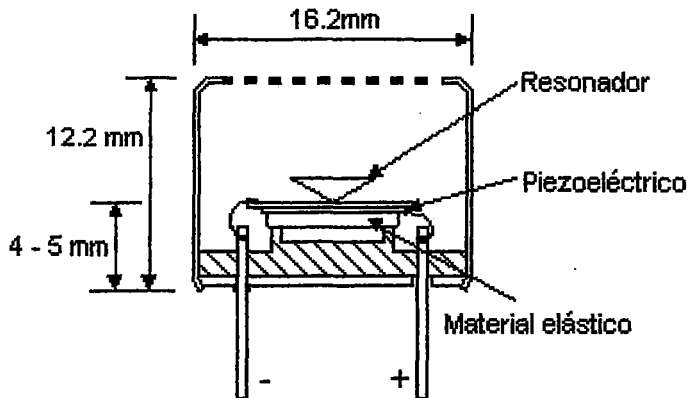


Fig.10: Interior del sensor UCM-R4OK1

El circuito de soporte para este sensor consiste en un PIC 12F675 o 12C508, requiere también un integrado MAX232 o alguna de esta familia, 2 OPAM, resistencias y condensadores. Este soporte es comercial hay por lo menos tres fabricantes y se adquirió el SRF04 por su tamaño pequeño. El diagrama de tiempos se muestra en Fig.11.

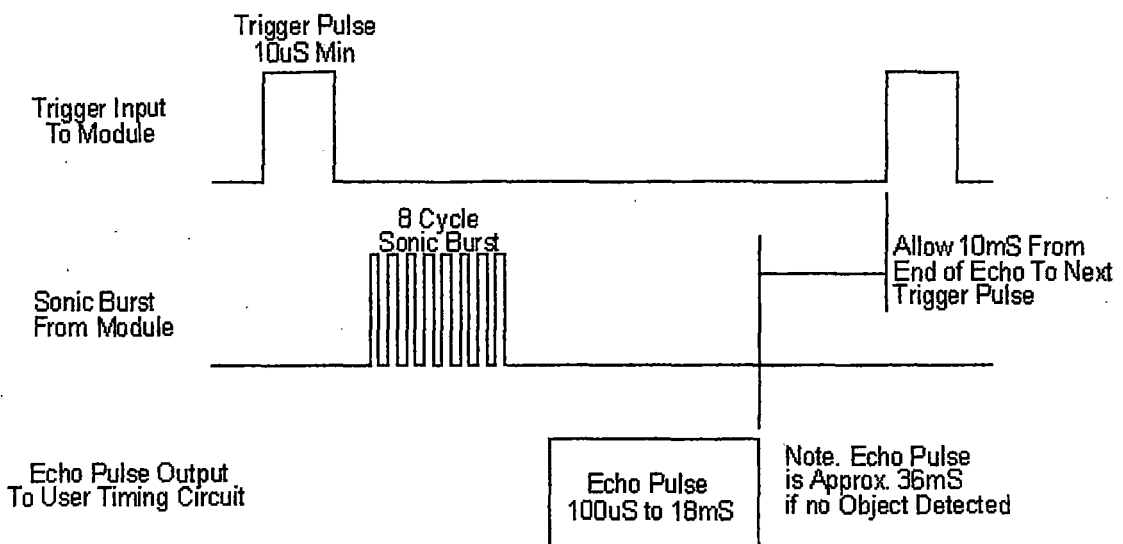


Fig.11: Diagrama de tiempos del SRF04 [10]

Resumiendo lo que se muestra en el diagrama: el PIC master 16F876A envía un pulso al módulo SRF04 cada cierto intervalo de tiempo establecido; el módulo, al percibir un pulso de entrada, emite una señal ultrasónica cuadrada; si el receptor ultrasónico percibe el eco (en forma de onda cuadrada), el módulo SRF04 envía al master un pulso alto que tiene el mismo tiempo de duración que demoró el eco en llegar al receptor. Este tiempo se procesa y opera con la rapidez del sonido (340 m/s) para hallar la distancia al plano perpendicular.

2.10. Memoria EEPROM

EEPROM o E2PROM son las siglas de *Electrically Erasable Programmable Read-Only Memory*, es una memoria no volátil, viene a ser un tipo de memoria ROM que puede ser programada, borrada y reprogramada eléctricamente, esto se puede hacer desde la PC o desde un microcontrolador externo programado para cumplir esta función, además mantienen la información por muchos años sin fuente de alimentación. Se diferencia de la EPROM porque esta última debe borrarse mediante un aparato que emite rayos ultravioleta.

Aunque una EEPROM puede ser leída un número ilimitado de veces, tiene un número límite de grabaciones que se encuentra entre 100.000 y un millón de veces. Existen dos tipos de memoria EEPROM seriales, una de ellas es la serie 24LCXX, que corresponde a los dispositivos de comunicación serial de dos hilos (I2C) y la serie 93LCXX que se comunica a través de tres hilos (SPI). Cada una de estas memorias utiliza un tipo de comunicación serial que depende de la acción a ejecutar, es decir; si se va a leer o escribir un dato, o si se va a enviar una dirección. El programa que ejecuta el computador o microcontrolador deberá seguir paso a paso la secuencia del protocolo, el cual cambia dependiendo del tipo de memoria que se desea programar o leer. Si la memoria tiene más de 256 posiciones el direccionamiento se hace por medio de páginas, siendo de 0 a 255 la página 0 (cero), las posiciones 256 a 511 la página 1 y así sucesivamente. Dentro de la información que se debe enviar a las memorias seriales se incluye uno o varios bits correspondientes a los números de página de la memoria.

2.11. Pantalla LCD 2x16

La pantalla LCD (*Liquid Crystal Display*) es un periférico que consiste en una pantalla delgada y plana formada por un número de píxeles en color o monocromos colocados delante de una fuente de luz. Una de sus ventajas más resaltantes es que usa una cantidad muy pequeña de energía, que depende del brillo que se elija y del tamaño de la pantalla. Existen varios tamaños, por ejemplo la usada es de 2x16 caracteres, también hay de 2x12, 1x16, 4x16, 4x20, entre otras. Este bajo consumo hace que dicha pantalla sea ideal para un data logger que debe funcionar de preferencia con baterías. En Fig. 12 apreciamos una configuración válida de la pantalla y el PIC 16F876A. Al usarla con un PIC hay una gran cantidad de configuraciones posibles, se elige la que mejor se acomode a las pistas o PCB que se va a diseñar. Además sólo se podrán usar los pines del PIC que tengan la opción de entrada/salida digital.

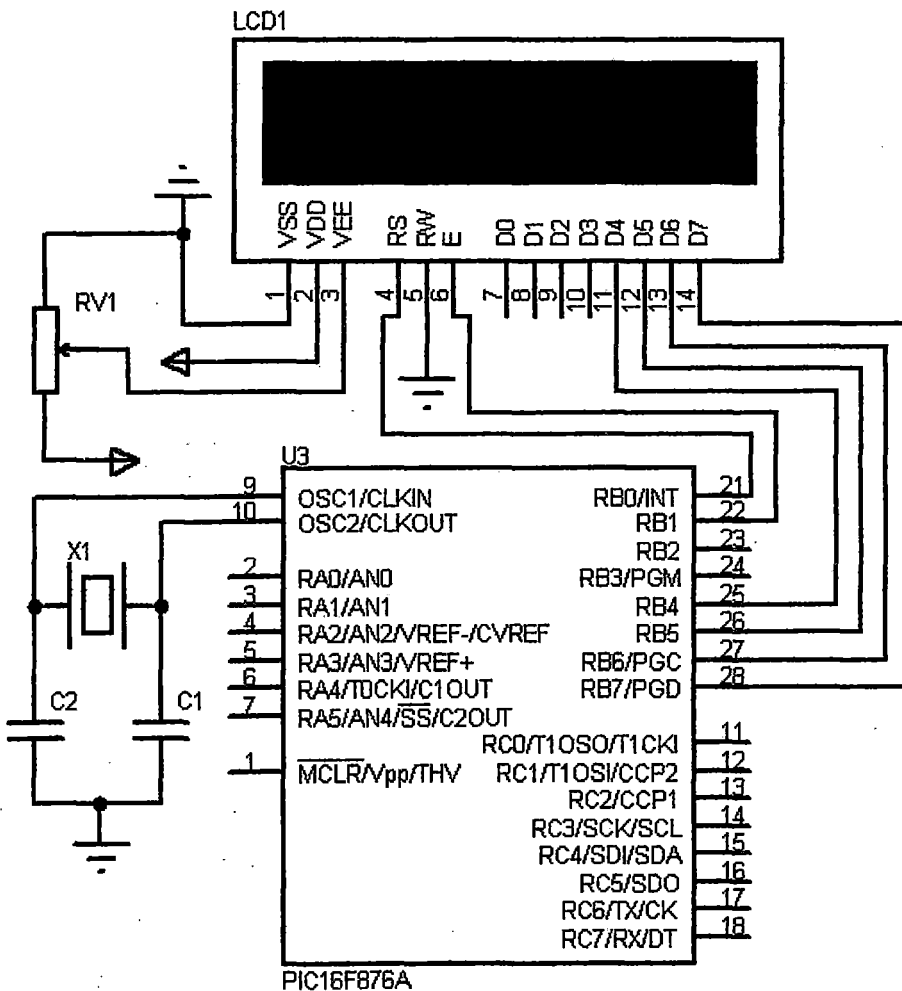


Fig.12: Configuración externa de la pantalla LCD 2x16

2.12. Multiplexión

La multiplexión o multiplexación es un proceso en el cual los datos de múltiples canales de transmisión comparten un enlace común, el proceso inverso se conoce como demultiplexación. Existen varios tipos de multiplexión, las más importantes son la multiplexión por división de frecuencia (FDM) y la multiplexión por división de tiempo (TDM), antes de tocar estas definiciones debemos entender qué es el ancho de banda:

Ancho de banda: Es la cantidad de información o de datos que se puede enviar a través de una conexión en un período de tiempo dado. El ancho de banda se indica generalmente en bits por segundo (Bps), kilobits por segundo (Kbps), o megabits por segundo (Mbps); se le asocia con frecuencias debido a que se mide como una inversa del tiempo.

Multiplexión FDM: En este tipo de multiplexión el espectro de frecuencias, representado por el ancho de banda disponible de un canal, se divide en porciones de ancho de banda más pequeños, de acuerdo a la cantidad de canales de entrada, los cuales se llaman “subcanales”.

Los subcanales se separan entre sí por una banda de guarda, para evitar las interferencias por solapamiento. El tipo de multiplexión FDM modula cada señal para su transmisión; las señales pueden ser analógicas o digitales, para las analógicas se utilizan los tipos de Modulación: AM (*Amplitude Modulation*), FM (*Frequency Modulation*) y PM (*Phase Modulation*); en el caso de las digitales utilizan ASK, FSK, PSK y DPSK.

Multiplexión TDM: La multiplexión TDM utiliza la técnica de tiempo compartido, la cual es la más utilizada en la actualidad, especialmente en los sistemas de transmisión digitales. El canal común es asignado a cada canal durante una fracción del tiempo total (intervalo de tiempo).

El multiplexor por división en el tiempo muestrea, o explora, cíclicamente las señales de entrada (datos de entrada) de los diferentes usuarios, y transmite las tramas a través de una única línea de comunicación de alta velocidad. Funcionan a nivel de bit o a nivel de carácter, a nivel de bit cada trama contiene un bit de cada dispositivo explorado; a nivel

de caracteres manda un caracter en cada canal de la trama. El segundo es generalmente más eficiente, dado que requiere menos bits de control que un TDM de bit. La operación de muestreo debe ser lo suficientemente rápida, de forma que cada buffer sea vaciado antes de que lleguen nuevos datos.

Capítulo 3

Parte experimental

3.1. Equipos y componentes

En la parte experimental se utilizaron multímetros KONSTAR UT33C, una PC (cualquiera con puerto serial), un osciloscopio digital RIGOL DS1052E y para la calibración el Termohigrómetro patrón marca LUFFT, modelo OPUS 10 THI.

Los multímetros fueron usados para evaluar voltajes de 0, 5 y 12 voltios, por esta razón su precisión no es tan importante. Para evaluar la amplificación de las señales analógicas y la comunicación serial se usó el osciloscopio. En la PC fueron instalados tres programas que han sido usados durante todo el experimento y se detalla a continuación:

ISIS Profesional v.7.5: Es conocido también como Proteus. El programa ISIS (*Intelligent Schematic Input System*) permite diseñar el diagrama eléctrico del circuito que se desea realizar con componentes muy variados, desde simples resistencias, hasta microprocesadores y microcontroladores, incluyendo fuentes de alimentación, generadores de señales y muchos otros componentes con prestaciones diferentes. Los diseños realizados en ISIS pueden ser simulados en tiempo real, mediante el módulo VSM (*Virtual System Modeling*), asociado directamente con ISIS. En este programa encontramos la mayoría de los integrados utilizados. Las imágenes de circuitos presentadas, en su mayoría, han sido obtenidas de las simulaciones hechas en este entorno.

MATLAB 7 v.R2010a: (MATrix LABoratory), es un lenguaje de alto nivel y un entorno interactivo para cálculo numérico, también conocido como lenguaje M. Ofrece un entorno de desarrollo integrado (IDE); entre sus prestaciones básicas se hallan la manipulación de matrices, la representación de datos y funciones, la implementación de

algoritmos, la creación de interfaces de usuario (GUI), y la comunicación con programas en otros lenguajes y con otros dispositivos hardware. El paquete MATLAB dispone de dos herramientas adicionales que expanden sus prestaciones, Simulink (plataforma de simulación multidominio) y GUIDE (editor de interfaces de usuario - GUI). Además, se pueden ampliar las capacidades de MATLAB con las cajas de herramientas (toolboxes); y las de Simulink con los paquetes de bloques (blocksets).

El Matlab es un lenguaje de preferencia debido a la familiaridad, es muy usado en los cursos de pregrado, más que el LabView, el cual también podría usarse. Se ha aprovechado de Matlab, las herramientas para establecer la comunicación serial, también el entorno gráfico con el cual se pueden crear ejecutables.

Proton IDE v.1.0.4.6: Proton IDE profesional es un entorno de desarrollo integrado (IDE) que ha sido diseñado específicamente para el Proton Plus compiler. Proton es un entorno de programación para PICs, su lenguaje de programación es el Basic, y posee librerías para la comunicación serial sincrónica y asincrónica. También se pueden usar entornos como MikroC, CCS, Pic Basic Pro, entre otros.

Además de los equipos y software mencionados, también se utilizaron otras herramientas del laboratorio como caudín, pinzas, desarmadores, etc. Esto es con respecto a la parte instrumental, los componentes electrónicos utilizados se mencionan a continuación:

- PIC16F876A
- PIC12F675
- EEPROM 24LC256
- Max232
- Pantalla LCD 16x2
- Transductor de temperatura LM35
- Transductor de humedad relativa HS1101
- Modulo ultrasónico SRF-04
- Timer LM555
- OPAM LM358, LM2904

- Resistencias de 0.25 watt (varios valores)
- Reguladores de voltaje 7805
- Condensadores electrolíticos (varios valores)
- Cristal de cuarzo de 4 MHz
- Batería de 9 voltios
- Molex y espadines

3.2. Sensores, amplificación y soporte

Hemos elegido tres sensores, para medir temperatura, humedad y distancia respectivamente, en ninguno de los casos conectamos directamente el sensor al PIC master. Hay sensores que pueden medir fenómenos físicos usando conexión directa al microcontrolador mediante el protocolo SPI, como el TC77 para temperatura o el MPL115A1 para temperatura y presión, también tenemos el SHT15 para temperatura y humedad, entre otros sensores. Pero la desventaja está en que el precio de estos sensores es mucho más alto y no hay grandes ventajas en precisión y resolución, comparando con los sensores usados. A continuación se expone la electrónica necesaria para que los sensores funcionen en un rango comprendido por el PIC.

3.2.1. Sensor de humedad

El sensor de humedad HS1101 es un condensador que contiene material higroscópico entre las placas, al ser capacitivo es conveniente usarlo acompañado de un Timer 555 en modo multivibrador astable pues el microcontrolador no puede evaluar en forma adecuada las caídas de tensión en un condensador; pero sí puede medir directamente un número de pulsos por intervalo de tiempo, podemos ver la configuración recomendada por el datasheet en Fig.13.

El Timer 555 en modo multivibrador astable es un generador de señal cuadrada, en donde la frecuencia F depende de las resistencias R_2 , R_1 , y del condensador C que en este caso es el HS1101. La regla de correspondencia es:

$$F = \frac{1.44}{(R2 + 2xR1)C}$$

Al aumentar la humedad, la constante dieléctrica del material higroscópico se reduce, por tanto la capacitancia se reduce; al reducirse la capacitancia la frecuencia aumenta, ya que es inversamente proporcional. De modo que a mayor humedad, mayor frecuencia y viceversa.

Cabe destacar que habrán cambios para cada variante del Timer 555, el NE 555 por ejemplo introducirá errores a temperaturas cercanas a 0 °C, debido a que es su límite de trabajo, lo que no pasará con el LM 555.

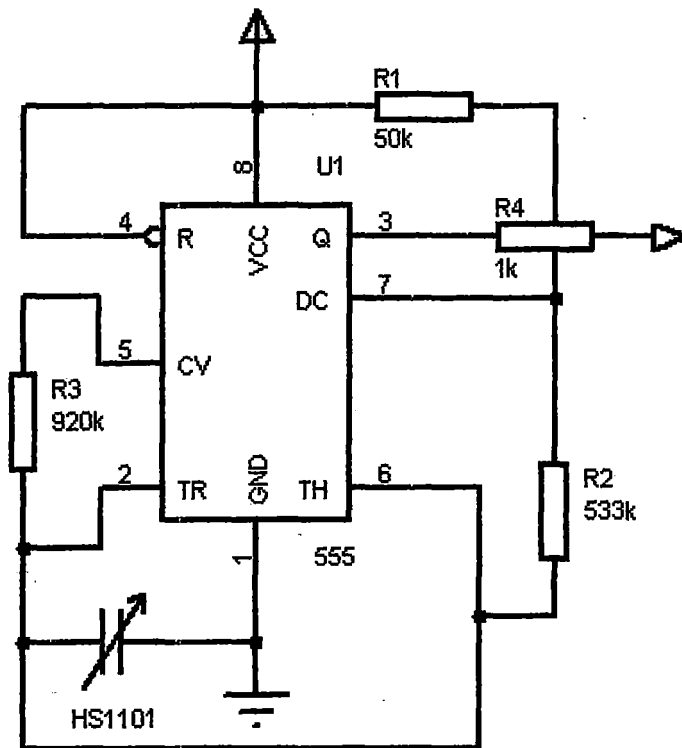


Fig.13: Configuración del Timer, modificación del circuito recomendado en el datasheet

3.2.2. Sensor de Temperatura

Con respecto a la temperatura, el sensor LM35 varía en 10 mV por cada grado Celsius. Ya que el convertor A/D del PIC16F876A tiene 1023 estados ($2^{10} - 1$), su resolución mínima será:

$$R_{min} = \frac{5 V}{2^{10} - 1} = 4.88 mV$$

De modo que el ADC del PIC puede resolver desde 0.5 °C (5 mV), hasta 50 °C (500 mV) sin problemas. Buscamos una resolución de 0.1 °C (0.08 °C es el mínimo para el sensor) y se logra usando un OPAM, elegimos el LM358 que puede amplificar hasta en un factor 100. Dado que los datos a ser amplificados pertenecen a una escala muy pequeña, se corre el riesgo de alteraciones debido a interferencias externas y a la diferencia de impedancia entre el LM35 y el PIC; por lo cual es conveniente pasar esta señal por la etapa de seguidor, aprovechando que el integrado LM358 contiene dos OPAM. Esta configuración se puede observar en Fig.14.

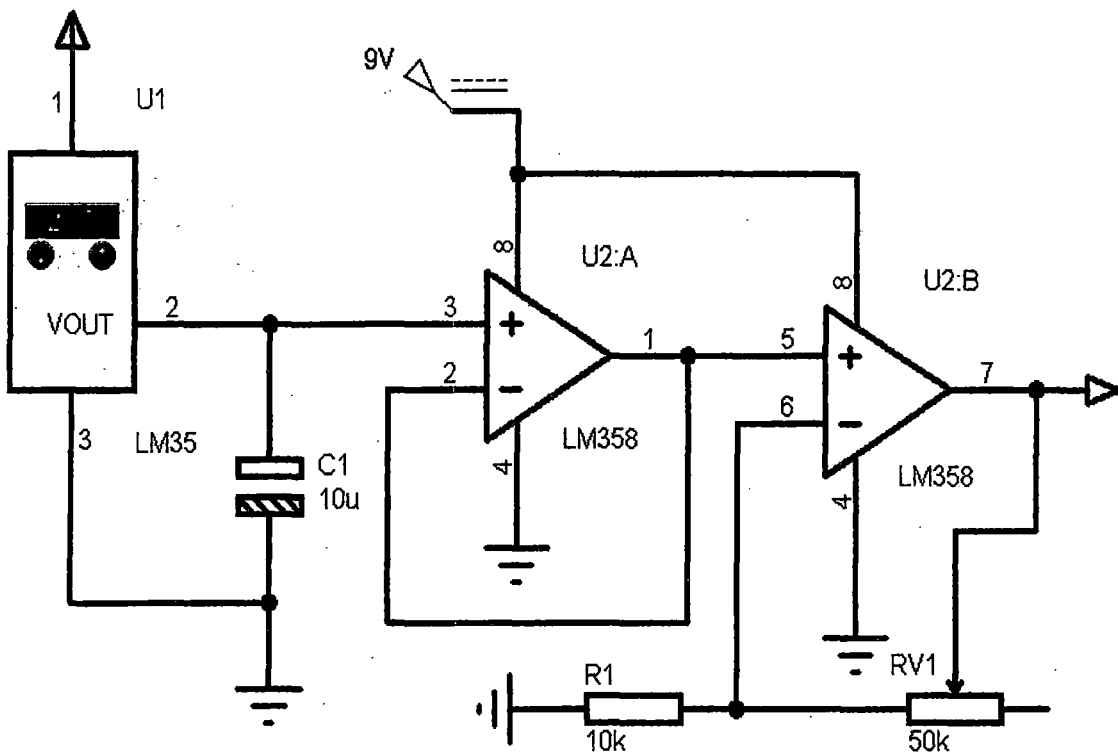


Fig.14: Configuración del OPAM y el LM35, amplificación de 0 V a 5V

3.2.3. Sensor ultrasónico

Los sensores ultrasónicos poseen un emisor y un receptor, se usan para detectar la proximidad de un objeto, y en nuestro caso para calcular la distancia entre el sensor y el

objeto. En Fig.15 se muestra el esquema básico del módulo ultrasónico SRF04, que presenta pequeñas variaciones para otros modelos.

Se utiliza un PIC12C508 para realizar las funciones de control y generar una señal cuadrada de 40 kHz, la cual es necesaria para los transductores piezoeléctricos. A modo de observación, la diferencia entre la serie 12C y la serie 12F de los PICs, es que la serie C se graba una única vez, no es regrabable, mientras que la serie F si lo es (memoria flash). La señal de salida del PIC tiene una amplitud de 5 V, la cual puede hacer funcionar el emisor ultrasónico, pero el módulo SRF04 sólo detectaría distancias cortas. El transductor puede soportar hasta 20 V de amplitud a 40 kHz, así que la mejor opción es doblar el valor de la amplitud, otros módulos ultrasónicos usan una compuerta NOR para lograrlo, en este caso se usa un integrado ST232CD, que para el caso cumple las mismas funciones que el MAX232. Ya se había mencionado que el MAX232 cambia el nivel lógico TTL (de 0 V a 5 V) a nivel lógico RS232 (de 12 V a -12 V), la entrada de alimentación del modulo SRF04 es de 5 V (voltaje que requiere el PIC12C508), la señal de salida del ST232CD es de -5 V a 5 V, la cual se conecta al emisor ultrasónico duplicando así el alcance del módulo. La señal se envía directamente hacia un objeto y el eco llega al receptor, el receptor entrega una señal a 40 kHz y de unos cuantos mV pico con mucho ruido. El circuito de recepción es similar al de cualquier micrófono, con dos etapas del amplificador operacional LMC6032 configurados en modo amplificador para un factor de ganancia de 17 cada uno ($1 + 33/2.2$). La salida va a un comparador diferencial cuya función es discriminar el ruido que también ha sido amplificado y envía una señal cuadrada al PIC, un pequeño voltaje positivo de retroalimentación proporciona histéresis para entregar una salida estable y limpia. El PIC reconoce como 1 lógico y 0 lógico la señal cuadrada que sale del comparador, este PIC, al detectar un 0 lógico en GP2 envía un pulso alto por GP0 que tiene el mismo tiempo de duración que demoró el eco en llegar al receptor (ver Fig.15). De este modo podemos medir distancias desde 3 centímetros a 3 metros, a distancias cortas (menos de 2 cm) el receptor recoge la señal directa del transmisor en lugar del eco y a más de 3 metros no llega bien el eco al receptor. El voltaje negativo conveniente para el amplificador operacional y el comparador es generado por el ST232CD que nos da un valor cercano a -5 V como se muestra en Fig.15.

Sonic Range Finder SRF04

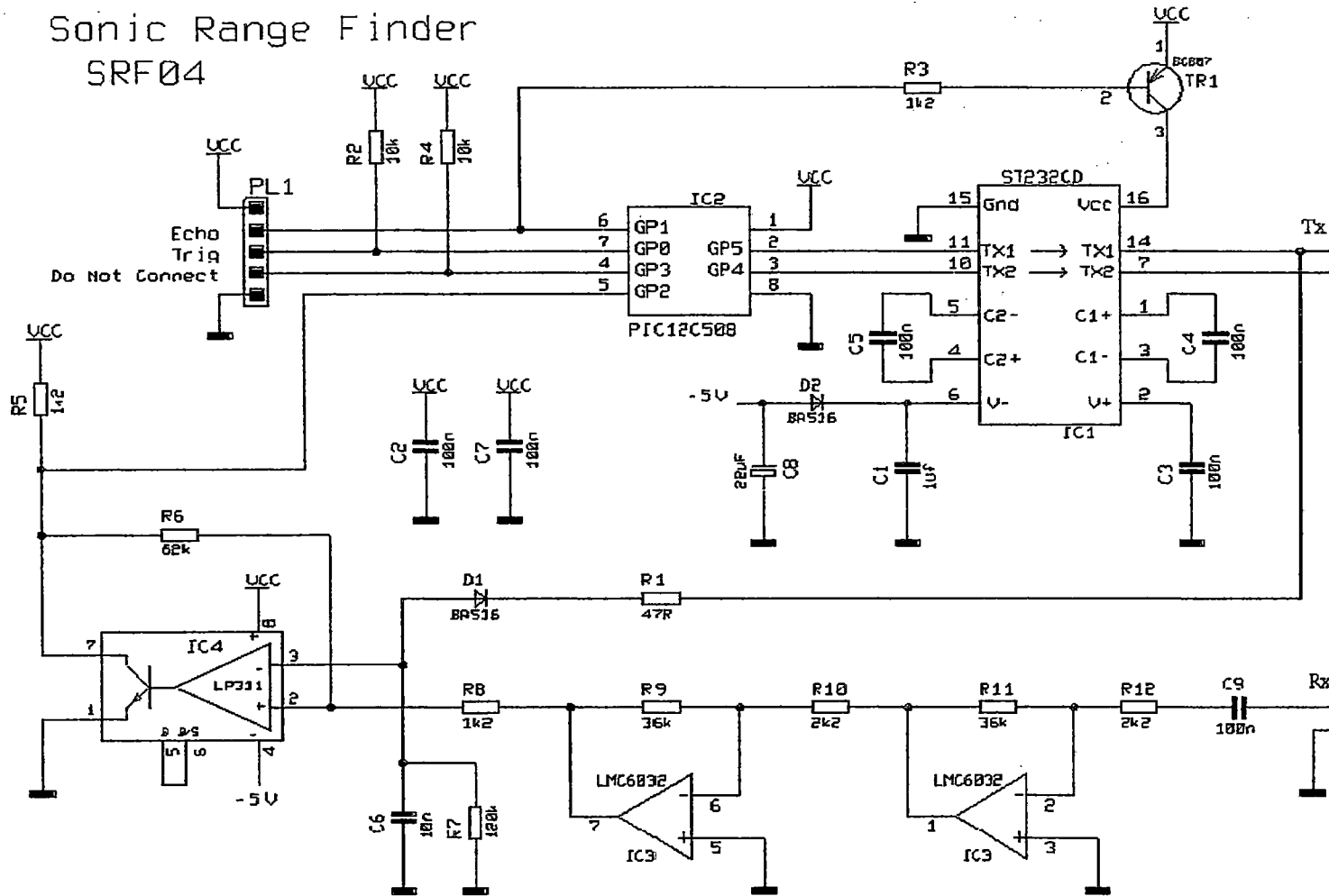


Fig.15: Circuito corrispondente al modulo SRF04 [10]

Resumiendo, el PIC master 16F876A envía un pulso cada cierto intervalo de tiempo establecido, si el receptor ultrasónico del módulo SRF04 percibe el eco (en forma de onda cuadrada), envía al master un pulso alto que tiene el mismo tiempo de duración que demoró el eco en llegar al receptor. Este tiempo se procesa y opera con la rapidez del sonido 340 m/s para hallar la distancia al plano perpendicular. En el apéndice C vemos el código del master PIC 16F876A.

3.3 Procesamiento de la señal analógica

Como se mencionó antes el PIC 16F876A posee un conversor A/D de 10 bits, para una entrada de tensión que va de 0 a 5V, el número de estados (N) está dado por 2^{10} (10 es el número de bits, 2 por el sistema binario) contados desde 0 hasta 1023. Así por ejemplo el estado que corresponde a una entrada de tensión de 2 V será:

$$N' = 2 * \frac{2^{10}-1}{5} = 409.2 = 409$$

Este estado 409 es el dato inicial con el que trabaja el PIC, dado que el integrado LM35 entrega 10 mV por grado Celsius, la temperatura correspondiente a la configuración de Fig.10 sería de 20.0 °C. Para llegar a esta temperatura partiendo del valor del registro ADC en 409, programamos en el PIC:

```

Declare ADIN_RES10      'Se declara un ADC de 10 bits
Declare ADIN_TAD2      'Configura el oscilador del ADC en FOSC/32
Declare ADIN_STIME 100  '100us de tiempo de muestreo

Dim Temp As Word
TRISA.0 = 1             'Se configura RA0 como entrada
ADCON1 = %10000001    'Se configura RA0 como entrada analógica
Temp = ADIn 0         'Inicia la conversión A/D, adquiere 409
                          'por RA0
Temp = Temp/2-3       'Cálculo de la temperatura (201)

```

En la última línea del programa se logra obtener en memoria un valor de 201 correspondiente a 20.1 °C, este 0.1 °C se encuentra dentro de la tolerancia. La configuración del oscilador externo tiene que ver con el registro ADCON0, en este caso

se declara “2” que equivale a “10” en binario correspondiente a los bits 7 y 6 de dicho registro, los bits de los registros ADCON0 y ADCON1 se encuentran en Fig.16.

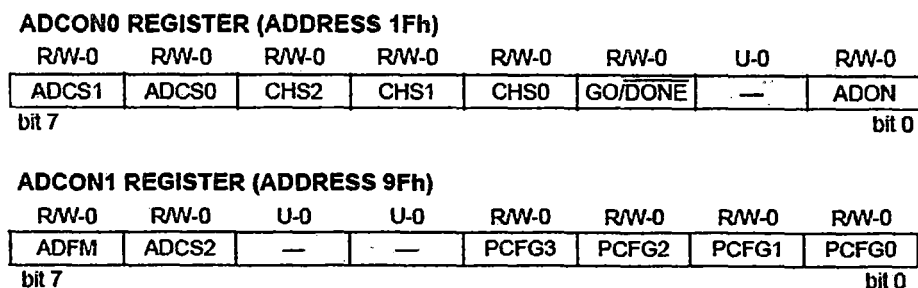


Fig.16: Registros ADCON0 y ADCON1 [1]

El bit 6 del registro ADCON1 es configurado en 0 (ADCON1 = %10000001), ubicándonos en la Tabla 8 esta configuración corresponde un TAD (A/D conversión Time) de $F_{OSC}/32$, que según la Tabla 9 se logra con oscilador externo de un máximo de 20 MHz.

Tabla 8: Configuración de frecuencia de conversión A/D [1]

ADCON1 <ADCS2>	ADCON0 <ADCS1:ADCS0>	Clock Conversion
0	00	$F_{osc}/2$
0	01	$F_{osc}/8$
0	10	$F_{osc}/32$
0	11	FRC (clock derived from the internal A/D RC oscillator)
1	00	$F_{osc}/4$
1	01	$F_{osc}/16$
1	10	$F_{osc}/64$
1	11	FRC (clock derived from the internal A/D RC oscillator)

Se busca (según el Datasheet) que el TAD mínimo sea de 1.6 μ s y para una conversión de 10 bits se requieren un mínimo de 12 veces el TAD, que en este caso resulta 19.2 μ s. Podemos ver en la Tabla 9 que para $2T_{OSC}$ la frecuencia máxima es 1.25 MHz, es decir este oscilador tiene un periodo de 0.8 μ s y requiere 2 ciclos mínimo (1.6 μ s) para cada TAD, pero la frecuencia pudo ser 1 MHz y con dos ciclos completa los 2 μ s para cada TAD lo cual sigue siendo correcto; dicho de otro modo, no hay ningún problema si el tiempo de adquisición es mayor. Siguiendo el ejemplo anterior se usó un oscilador externo de 4 MHz y $32T_{OSC}$, con esto nuestro TAD dura 8 μ s lo cual sigue siendo

correcto, incluso es mejor que trabajar en los límites dándole tiempo de sobra al microcontrolador para realizar la conversión A/D.

Tabla 9: TAD vs máxima frecuencia de oscilador externo [1]

AD Clock Source (TAD)		Maximum Device Frequency
Operation	ADCS2:ADCS1:ADCS0	
2 Tosc	000	1.25 MHz
4 Tosc	100	2.5 MHz
8 Tosc	001	5 MHz
16 Tosc	101	10 MHz
32 Tosc	010	20 MHz
64 Tosc	110	20 MHz
RC ^(1,2,3)	x11	(Note 1)

No hay que confundir el tiempo de conversión de los datos con el tiempo de muestreo, que para este caso es de 100 μ s (`Declare ADIN_STIME 100`) esto nos dice que durante 100 μ s el PIC detecta y promedia un valor de voltaje. Esta diferencia puede apreciarse en Fig.17, donde a cada bit correspondiente a la lectura analógica se le asigna un tiempo de conversión TAD.

Los resultado de la conversión de 10 bits son almacenados en los registros ADRESH:ADRESL y además el bit GO/DONE (registro ADCON0 bit 2) se pone a 0 lógico. Luego el comando "ADIn" de Proton IDE llama el dato de estos registros (`Temp = ADIn 0`) para su cálculo según la ecuación que relaciona el voltaje con el fenómeno a medir.

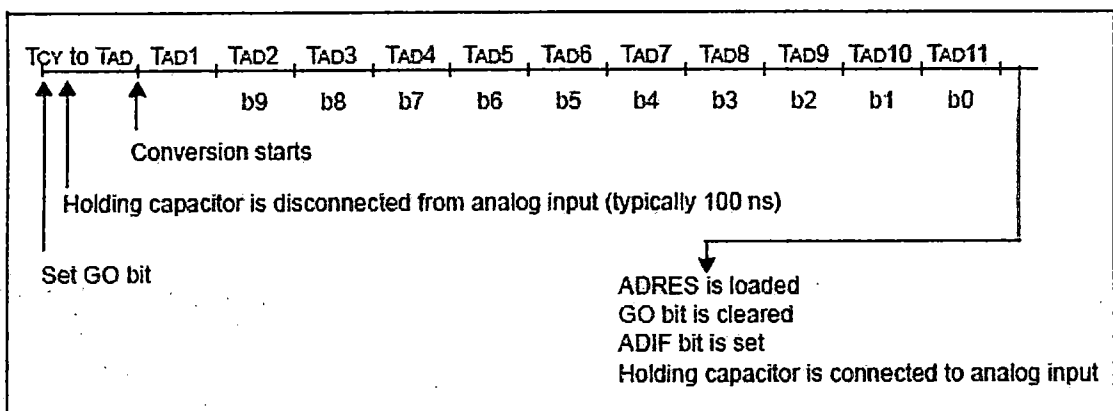


Fig.17: Conversión A/D bit por bit [1]

Una vez procesado el valor se envía de forma serial a la memoria EEPROM usando el protocolo I2C (*Inter-Integrated Circuit*).

3.4. Visualización LCD

La pantalla de cristal líquido (LCD) recibe datos en paralelo enviados desde el PIC master 16F876A. Para continuar con el código mostrado en el procesamiento analógico se usará la variable "Temp", correspondiente a la temperatura, con un valor temporal adquirido "Temp = 22.3" y la variable "Hum", correspondiente a la humedad relativa, con un valor temporal adquirido "Hum = 61.0", ambos datos se visualizan en el LCD, tal como se muestra en Fig.18.

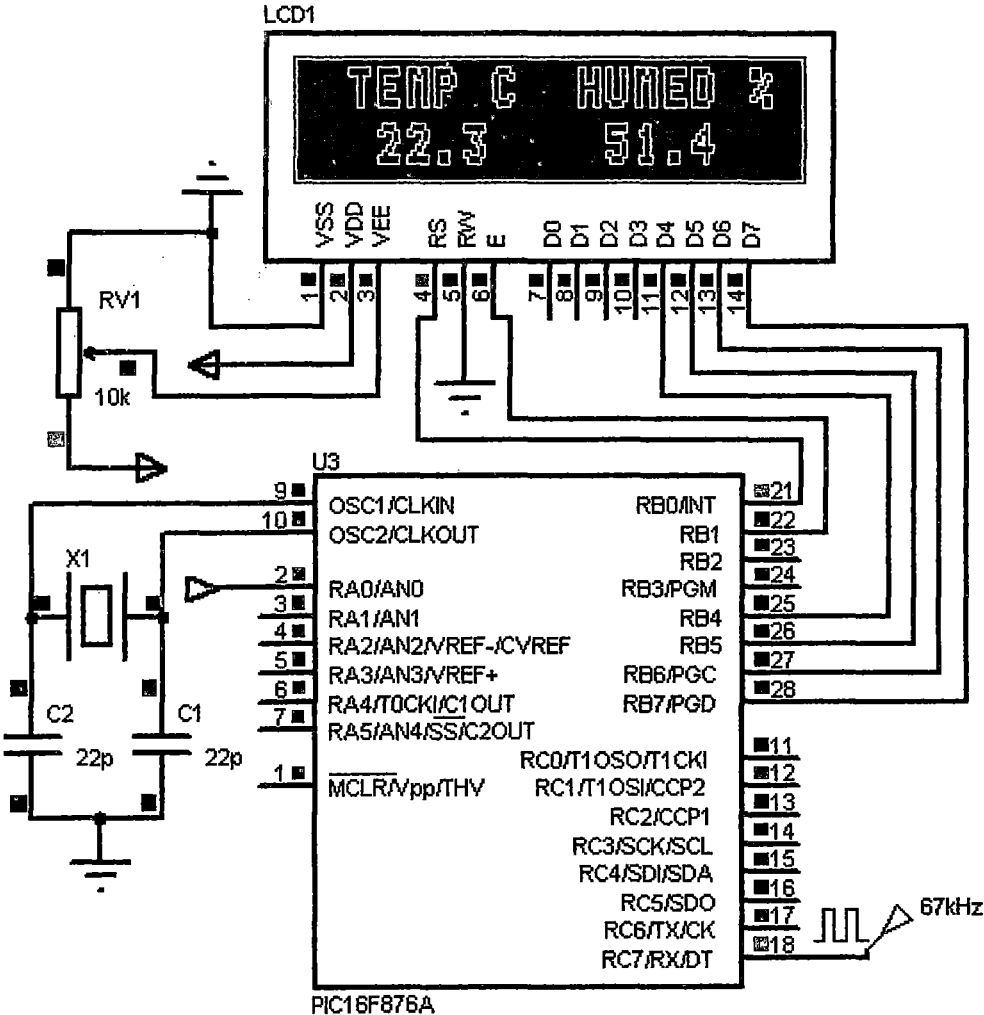


Fig.18: Configuración PIC-LCD

```

Declare LCD_INTERFACE 4
Declare LCD_LINES 2
Declare LCD_DTPIN PORTB.0
Declare LCD_ENPIN PORTB.5
Declare LCD_RSPIN PORTB.4
'////////////////////
'Declaraciones de variables'
'Conversión AD'
'////////////////////
Cls
Print At 1,1," TEMP C ", "HUMED %"
Print At 2,2," ",DEC2 Temp/10,".",DEC1 Temp//10
Print At 2,11, DEC2 Hum/10,".",DEC1 Hum//10
DelayMS 1000

```

En la primera línea del programa se declara una comunicación de 4 bits (**LCD_INTERFACE** 4) que corresponderá a los pines D4, D5, D6 y D7 del LCD. La configuración mostrada se usa cuando solamente hay escritura sobre la pantalla, cuando se requiere leer el dato mostrado en el LCD para hacer un nuevo cálculo se utilizan los 8 bits del bus de datos (desde el pin 7 al 14 del LCD), pero no es el caso. En la segunda línea se establece que el LCD tiene 2 líneas (**LCD_LINES** 2), correspondiente a nuestro LCD de 2x16 caracteres. En la línea 3 (**LCD_DTPIN** PORTB.0) se establece que la escritura sobre el LCD es a partir de RB0, quedan asignados los pines más significativos del puerto B a los pines del LCD, D4 con RB0, D5 con RB1, D6 con RB2 y D7 con RB3. Lo que sigue es la programación de los pines de control del LCD, el pin 4 RS (*Register Select*), el pin 5 R/W (*Read/Write*) y el pin 6 EN (*Enable*). R/W se coloca en baja (0) para escritura y en alta (1) para lectura, como usamos el LCD para imprimir caracteres, se conecta R/W a tierra como se ve en Fig.17. RS selecciona el registro del LCD que vamos a usar, el LCD tiene dos registros de 8 bits, un registro de instrucciones (0) y otro de datos (1), seleccionando el registro de instrucciones indicamos a la pantalla que en el bus de datos hay una instrucción, y seleccionando el registro de datos indicamos que tenemos en el bus de datos un caracter, cuyo destino son las dos zonas de memoria del LCD. EN se usa para decirle al LCD que estamos enviando datos, para enviar datos al LCD el programa debe poner dicho pin en baja (0) y a continuación establecer las otras dos líneas de control (RS y R/W) y/o colocar datos en el bus de datos. Cuando la transferencia de datos ha concluido, se pone EN alto (1) y se debe esperar la cantidad mínima de tiempo requerido por la hoja de datos (este tiempo está en nanosegundos), y al final se pone en baja (0) otra vez. Esta secuencia requerida para imprimir datos en el LCD se puede hacer enviando 0 y 1 por los pines RB4 y RB5 del

PIC o usando el comando "Print At" como se ve en las líneas 7, 8 y 9 del programa, este comando opera sobre EN y sobre RS, los pone en alta y baja formando la secuencia necesaria para imprimir datos. Así en la línea 7 del programa (**Print At 1,1,"** TEMP C ", "HUMED %") escribimos el texto en la fila 1 columna 1, y en la línea 9 del programa (**Print At 2,11, DEC2 Hum/10, ".",DEC1 Hum//10**) escribimos dos números y un punto en forma de texto en la fila 2 columna 11 como se ve en Fig.18. El comando "Cls" se usa para limpiar el LCD y dejarla en blanco, de no colocarlo los datos se sobrescriben.

Hay que mencionar también la función de los pines 3, 15 y 16 del LCD. El pin 3 es el ajuste de contraste que normalmente se hace con un potenciómetro colocado entre 5 V y tierra como se ve en Fig. 12. Por último, los pines 15 y 16 son los que se utilizan para alimentar el (o los) LEDs de fondo del LCD, estos brindan la iluminación (back light). El pin 15 debe ser conectado a una tensión de 4 V o 4,5 V y el 16 a tierra, en estas condiciones la luz de fondo está encendida a 100% de su brillo. En este caso también (como en el pin 3) se puede utilizar un potenciómetro para ajustar el brillo.

3.5 Comunicación serial con la EEPROM

La comunicación serial con la EEPROM es mediante el protocolo I2C, donde la memoria es el *Esclavo* y el PIC el *Maestro*. Proton IDE tiene incorporada la secuencia de comunicación I2C, basta con hacer uso de los comandos que llevan a esta secuencia. Veamos un poco del código en mención, iniciando con la parte de grabación de la EEPROM:

```
Device = 16F876A
XTAL 4
Declare SCL_PIN PORTB.6
Declare SDA_PIN PORTB.7
'.....'
'Declaraciones correspondientes a la pantalla'
'.....'
Dim TEMP As Word
Dim TDAT As Word
Dim DATA_OUT As Word
```

```

Dim Address As Word
Address = 0
'Programación de tiempo y datos por minuto'
INICIO:
'Programación de la conversión análogo digital'
ESCRIBIR:
Address = Address + 1      ' Address es el numero de la fila a
                          ' leer o escribir
    DATA_OUT = TEMP'ENTERO
    BStart ' Inicio de la comunicación I2C
    BusOut %10100000,Address,[DATA_OUT]

    BStop           ' parada del I2C
    DelayMS 5      ' pausa para alojamiento del dato
If Address >= TDAT Then
    GoTo LEER
Else GoTo INICIO
EndIf

```

Las declaraciones de PIC a utilizar y la frecuencia del oscilador externo son propias de cada programa, la comunicación serial de la EEPROM 24LC256 es mediante el protocolo I2C para lo cual se requieren dos líneas, una es la del reloj (SCL) y la otra la transferencia de datos (SDA), declaramos entonces a RB6 como pin de reloj (**Declare SCL_PIN** PORTB.6) y a RB7 como pin de datos (**Declare SDA_PIN** PORTB.7). Paso siguiente, declaramos las variables tipo Word, este tipo de variable posee 16 bits de espacio, lo cual es necesario para las variables como TEMP, correspondiente a la temperatura, recordemos que la temperatura puede tener un valor mayor a 256 (25.6 °C) por lo que la variable Byte (8 bits) sería insuficiente, para simplificar la comprensión del programa consideremos un valor de temperatura menor a 256, así se grabará en la memoria un solo byte, en la práctica se graban 4, 2 bytes de temperatura y 2 bytes de humedad. También declaramos una variable para el número total de datos (TDAT), una para el valor a enviar a la EEPROM (DATA_OUT), y otro para el número de fila (Address). Declaradas las condiciones iniciales y calculada la temperatura, pasamos a la parte de escritura de la EEPROM.

Iniciamos la comunicación I2C con el comando BStart, usando el comando BusOut el PIC se comunica con la memoria enviando un byte que contiene los siete bits (%10100000) que componen la dirección del dispositivo esclavo con el que se quiere

comunicar, y el octavo bit (%10100000) de menor peso se corresponde con la operación deseada, en nuestro caso escritura = 0 (enviar al esclavo). El comando BusOut también contiene la dirección de la fila en la página de la memoria (Address) y una variable que contiene el valor a guardar (DATA_OUT) en la fila de la página establecida (en este caso página 0).

Almacenado el dato en la EEPROM se detiene la comunicación serial, el proceso se repite con otro dato de temperatura hasta que el número de datos guardados sea TDAT. Una vez almacenada toda la data podemos apagar el equipo por años de ser necesario y al encenderlo la interface nos debe llevar a la opción de lectura.

```
LEER:
.....
'Condiciones para lectura por RS232 desde la PC'
.....
EXPORTAR:
DelayMS 500
For Address = 1 To TDAT
  BStart
  BusIn %10100001,Address,[TEMP] 'TEMP toma el dato de
DATA_OUT
  BStop
  .....
  'Salida por pantalla LCD'
  'Salida serial de datos '
  .....
  DelayMS 500
Next Address
GoTo LEER
Stop

GoTo LEER
```

Cuando la PC le da la orden al PIC para enviar los datos, éste los recupera desde la EEPROM, es decir, la EEPROM exporta los datos hacia el PIC; para esto se inicia la comunicación I2C con el comando BStart, usando el comando BusIn, el PIC se comunica con la memoria enviando los siete bits (%10100001) que componen la dirección del dispositivo esclavo y el octavo bit (%10100001) que corresponde a la lectura = 1 (leer al esclavo). El dato leído desde la dirección requerida se almacena en TEMP y es enviado mediante comunicación serial TTL del PIC al MAX232, y finalmente por RS232 hacia la PC.

3.6. Comunicación serial con la PC

El PIC, luego de haber calculado la temperatura y la humedad, envía en forma serial los datos a la EEPROM, cuando los datos son llamados desde la PC, estos van de la EEPROM al PIC y del PIC a la PC, para la comunicación PIC-PC se usan dos comandos en Proton IDE, que son, “SerIn” para recibir datos y “SerOut” para exportarlos, acompañados de la variable que contiene al dato en este caso VAR1:

```
SerIn PORTC.5, 3313, 200, LEER, [ESER]
```

```
SerOut PORTC.6, 3313, [DEC3 VAR1]
```

Donde PORTC.6 se refiere al pin RC6 que es configurado como salida serial, PORTC.5 se refiere al pin RC5 que es configurado como entrada serial, 3313 indica que la transmisión es de 8 bits invertida sin paridad y su velocidad es de 300 baudios por segundo, esto se puede calcular usando la ayuda de Proton IDE:

1. Determine the bit period. (bits 0 - 11) $(1,000,000 / \text{baud rate}) - 20$
2. Data bits and parity. (bit 13) 8-bit/no-parity = step 1 + 0. 7-bit/even-parity = step 1 + 8192
3. Select polarity. (bit 14) True (noninverted) = step 2 + 0. Inverted = step 2 + 16384
4. Select driven or open output (bit 15) Driven = step 3 + 0 Open = step 3 + 32768

El número 200 en el comando SerIn indica que el PIC espera durante 200 ms para recibir un dato serial; si no lo recibe el programa vuelve a la etiqueta LEER, esta etiqueta se ubica en la sección del programa donde se crea conveniente, si el PIC recibe ese dato serial durante los 200 ms el programa continua y se le asigna el dato serial a la variable ESER, que en este caso será de tipo Byte, está claro que este dato viene de la PC mediante protocolo RS232. Quien recibe directamente el dato de la PC es el integrado MAX232, que se encarga de cambiar este dato al nivel TTL.

El DEC3 antes del VAR1 indica que antes de la transmisión la variable VAR1 tiene 3 valores en base decimal, por ejemplo de calcular una temperatura 25.6 °C se enviará 256, si la temperatura calculada es 5.6 °C se enviará 056. El Software en la PC recepciona este valor y lo introduce a un vector de dimensión 3, un espacio para cada cifra. A continuación se muestra el código que permite iniciar la lectura de la EEPROM:

```

LEER:
'Configuración de la pantalla LCD'
SerIn PORTC.5,3313,200,LEER,[ESER]
If ESER="xxxx" Then
  GoTo EXPORTAR
EndIf
GoTo LEER
EXPORTAR:

```

Este dato serial recibido (ESER) es único y tiene un valor constante para cualquier lectura de datos, esta palabra condiciona la lectura del PIC, es decir que aunque alguien tenga los demás parámetros de la comunicación RS232 (baudrate, paridad, etc.) no podrá acceder a la información, ningún software de comunicación serial podría leer esta información solo el desarrollado en Matlab para este propósito, esta es la razón por la cual algunos equipos que tienen comunicación serial estándar no pueden ser leídos. La configuración externa del MAX232 se muestra en Fig.19.

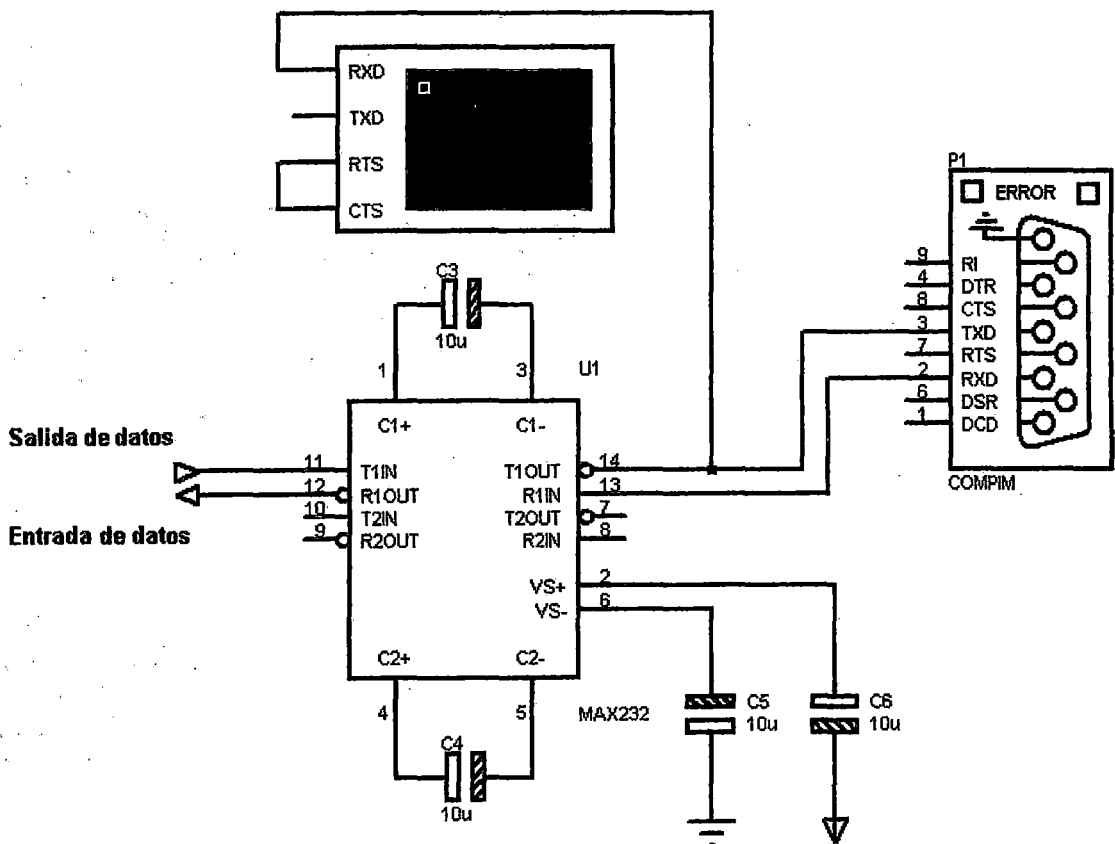


Fig.19: Comunicación entre el PIC y el puerto serial usando el integrado MAX232

El PIC enviará los datos usando el comando “Serout”, serán enviados en el nivel TTL (de 0 V a 5 V) al integrado MAX232 y los datos que salen viajarán en el nivel lógico RS232 (de -5 V a 5 V).

A continuación se muestra el código en lenguaje “M” (Matlab) necesario para la comunicación con el PIC:

```
PS = serial('COM1');
set(PS,'Baudrate',300); % velocidad a 300 Baudios por segundo
set(PS,'StopBits',1); % se configura un bit de parada
set(PS,'DataBits',8); % se configura que el dato es de 8 bits
set(PS,'Parity','none'); % se configura sin paridad
set(PS,'Terminator','CR/LF');% forma en la que termina el envío
set(PS,'OutputBufferSize',1); % "1" es el número de bytes a
                               %enviar
set(PS,'InputBufferSize',1); % "1" es el número de bytes a
                               %recibir
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

El comando “*instrfindall*” encuentra todos los objetos del puerto serie, podemos ejecutarlo desde línea de comando para asegurarnos de que hay puertos disponibles. Una vez ubicados los puertos disponibles procedemos a la programación.

Declaramos en primer lugar que vamos a trabajar con el puerto COM1 y se le asigna la variable PS, luego establecemos la velocidad de transmisión en 300, declaramos un bit de parada y 8 de datos, y se configura sin paridad. CR significa “*carriage return*” es uno de los caracteres de control de la codificación ASCII, hace que se mueva el cursor a la primera posición de una línea y en este caso se usa junto a LF que significa “*line feed*”, con lo cual ordenamos al programa que luego de recibir un dato serial, la nueva lectura se haga en la siguiente línea y al inicio de esta. Configuramos, por último, que vamos a enviar solo un byte a la vez y vamos a recibir también un byte a la vez.

A continuación se describe el código necesario para recibir los datos desde el PIC:

```

fopen(PS);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fprintf(PS,'%s','xxxx'); %envío de la condición de inicio
KT=[];
dtotal=dm*tmp;
for i=1:2*dtotal
    K1=zeros(1,3);
    for j=1:3 %datos de tres cifras
        K1(j) = fread(PS,1,'uchar');
        %entra un (1) valor, en código ASCII
    end;
    K1=char(K1);%el valor es convertido a cadena
    K1=str2double(K1); %el valor es convertido a entero
    KT=[KT K1];
end;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Operaciones con matrices, tratamiento de la data%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

Luego de declarar las condiciones que establecen la comunicación serial, se procede a abrir el puerto (PS) y enviamos la palabra de inicio al PIC. La matriz KT se declara sin dimensión, ya que depende de la cantidad de datos almacenada en la memoria, la cual varía para cada lectura; “dtotal” es el número de datos que se almacena en KT y es igual al producto de los datos por minuto (dm) y el tiempo total de medición (tmp).

El primer “for” va de 1 a 2 veces “dtotal” porque se mide a la vez temperatura y humedad, se declara un nuevo vector K1 que contendrá los tres dígitos que componen el valor de temperatura o humedad, pero en código ASCII. Hay que aclarar que los valores decimales se transfieren de un microprocesador a otro en código ASCII, por esto a la hora de recepcionar los datos, deben convertirse de ASCII a String usando “char”. Para indicar que el valor entrante es de tipo entero sin signo y de un byte se usa “uchar”, es un tipo de entero al igual que “int”, “short” y “long”, salvo que estos sí tienen signo. En el ejemplo anterior en el que se envió una temperatura de 5.6°C se tienen los datos decimales a enviar 056 desde el PIC y en código ASCII 48 53 54, correspondientes a 3 bytes, estos valores se asignan en orden a K1 y permanecen en código ASCII hasta que usamos “char”, sólo entonces los valores de K1 toman 0 5 6 en forma de cadena. Finalmente se usa “str2double” para pasar 0 5 6 cadena a 056 entero. En la parte final de este código se introduce el vector K1 dentro de la matriz KT y el proceso se repite.

3.7. Software para transferencia de datos

Para la comunicación entre la PC y el equipo se desarrolló un programa en Matlab 2010, ya que para aplicaciones comerciales no se trabaja con la extensión *.m (solo código) hubo la necesidad de usar la GUI (*Grafic Usuary Interface*) de Matlab y su compilador por defecto, el LCC. Podemos crear una nueva GUI guiándonos con Fig.20.

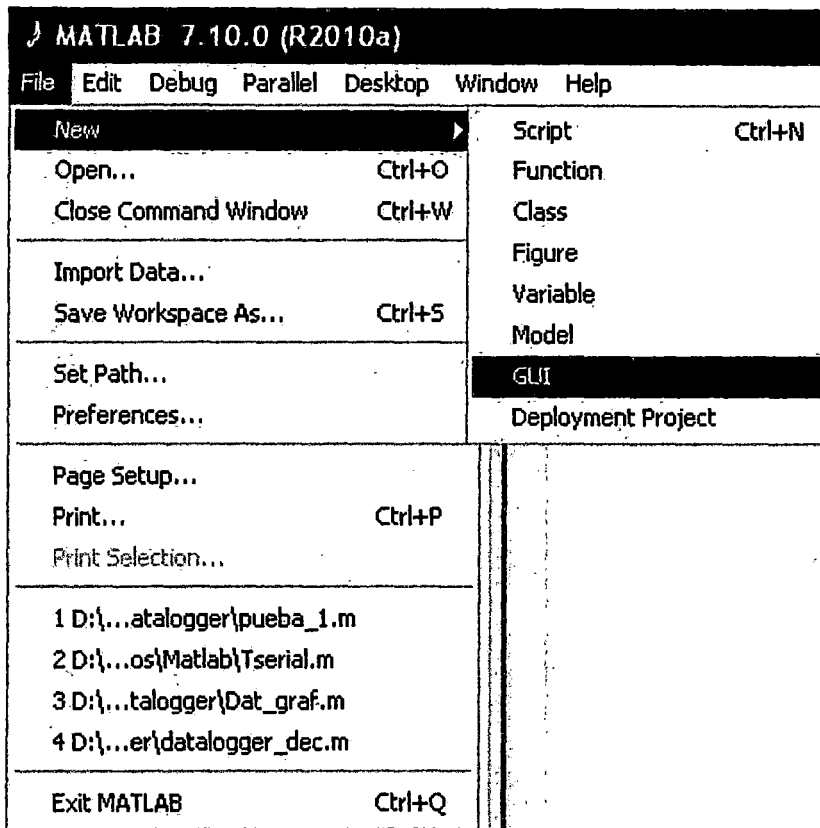


Fig.20: Nueva interface gráfica de usuario (GUI)

Hemos visto en la sección 3.6 (comunicación serial con la PC) la información y código en lenguaje M (lenguaje de Matlab) necesarios para establecer la comunicación serial RS232. Establecida la comunicación serial y una matriz donde se almacenen los datos, cerramos el script y damos clic en “file”, “new GUI”. Vale aclarar que el conjunto de figuras correspondientes al entorno gráfico se guardan en un archivo de extensión *.fig, el cual enlaza líneas de código con los objetos mostrados. Para éste entorno grafico se usaron casillas de texto (edit text), líneas de texto estático (static text), un botón de inicio (push button), un cuadro desplegable (pop-up menu) y un par de ejes (axes); cada

uno tiene un código asociado en el archivo *.m pero la totalidad del código es muy extenso, por lo cual se adjunta en los apéndices. A continuación se describe cómo ensamblar el entorno gráfico mostrado en Fig.21, en la primera casilla se establece el número de datos que se leen por minuto con valor por defecto 1, a su izquierda una casilla de texto estático. En la segunda casilla el tiempo total en minutos, también con 1 por defecto y su indicación estática a la izquierda. Luego se coloca un “Pushbotton” con el texto “Inicio”, este se enlaza con un “Callback” que inicia el programa de lectura de datos, cuando se pulsa este botón es enviada la palabra de inicio al PIC.

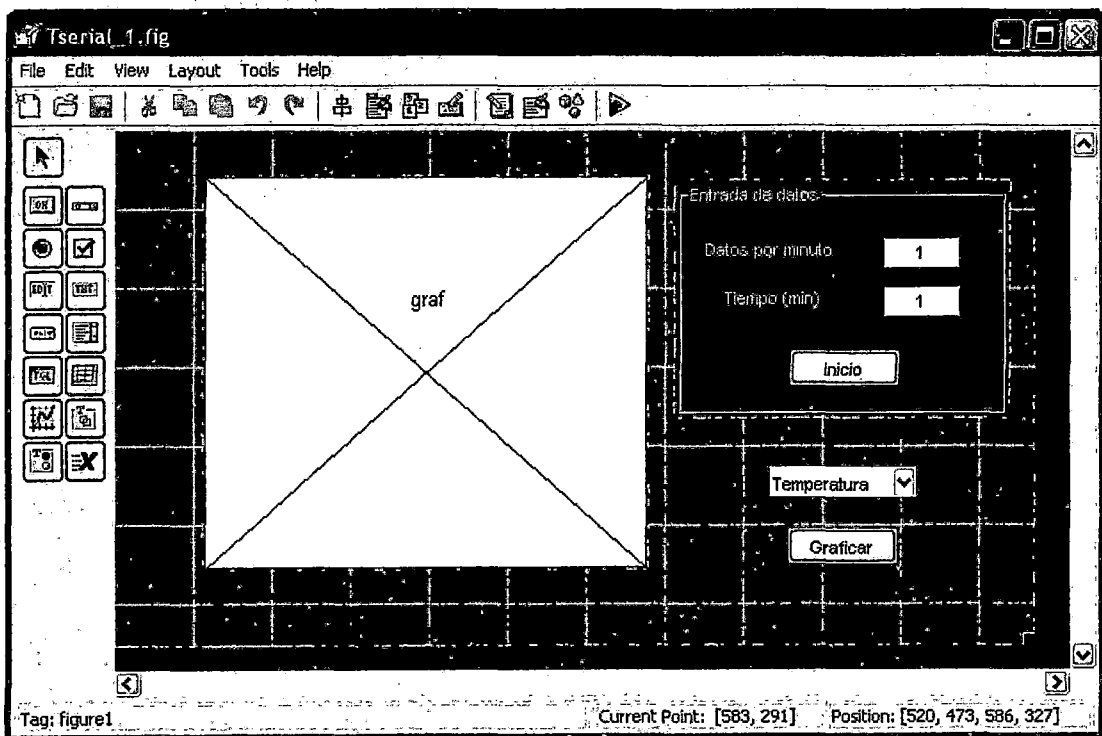


Fig.21: Entorno gráfico desarrollado en Matlab

Desde este punto ya podemos adquirir datos usando un entorno gráfico, pero aún dependiente de Matlab, terminada la parte gráfica se procede a la compilación, es decir unimos el archivo *.fig, el *.m, los convertimos en un ejecutable para Windows y lo empacamos junto al Runtime (MCR). Para esto ejecutamos el comando “deploytool”, es la herramienta a través de la cual accedemos al compilador de Matlab, el LCC. Emerge la ventana mostrada en Fig.22.

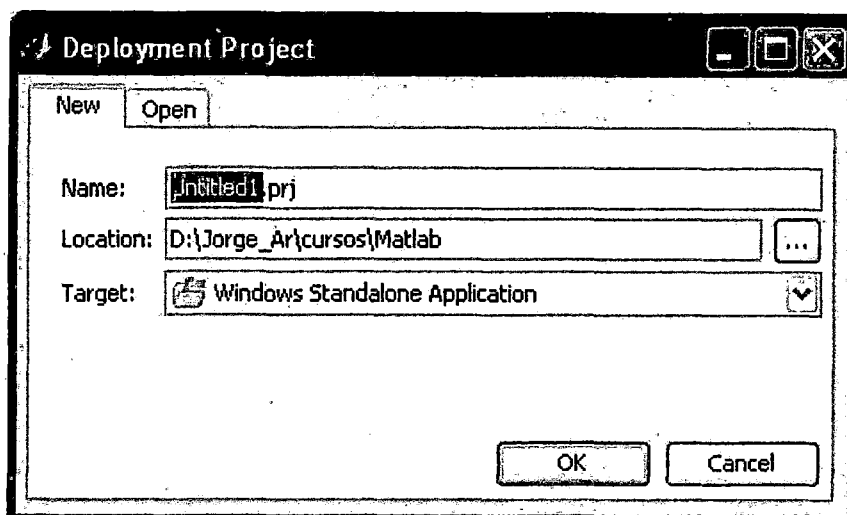


Fig.22: Nuevo proyecto con la herramienta Deploytool

Creado el proyecto aparece una ventana en donde podemos agregar los archivos *.m que sean necesarios, los archivos *.fig (objetos) ya están asociados a los *.m, no es necesario llamarlos. Luego construimos el ejecutable con la opción Build, como se muestra en Fig.23. El empaquetado se hace con la opción Package, y se le asocia el MCR, esto es lo que nos permite correr el programa en cualquier computadora que no tenga Matlab instalado.

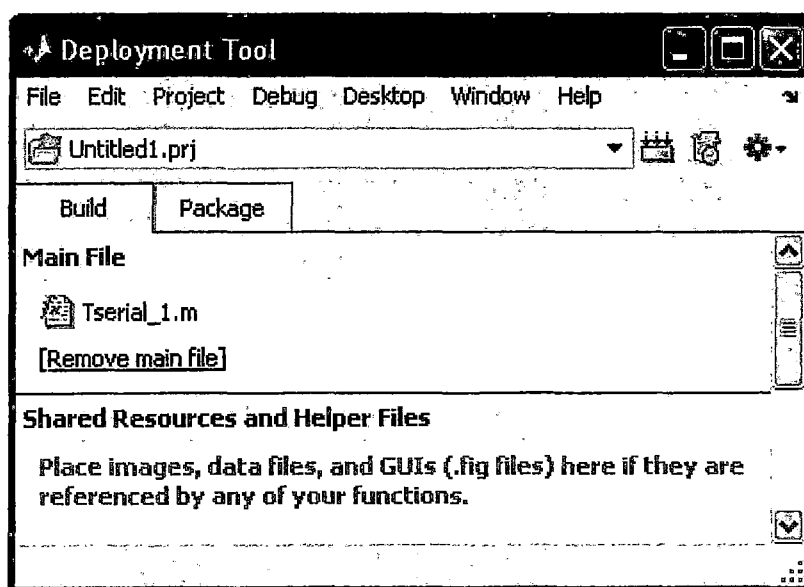


Fig.23: Creación del ejecutable con la herramienta Deploytool

Luego de compilar y empaclar se obtiene un ejecutable, puede correr en cualquier máquina que no tenga Matlab y que tenga libre el puerto programado, en este caso "COM1". Al abrir este ejecutable aparece la ventana mostrada en Fig.24.

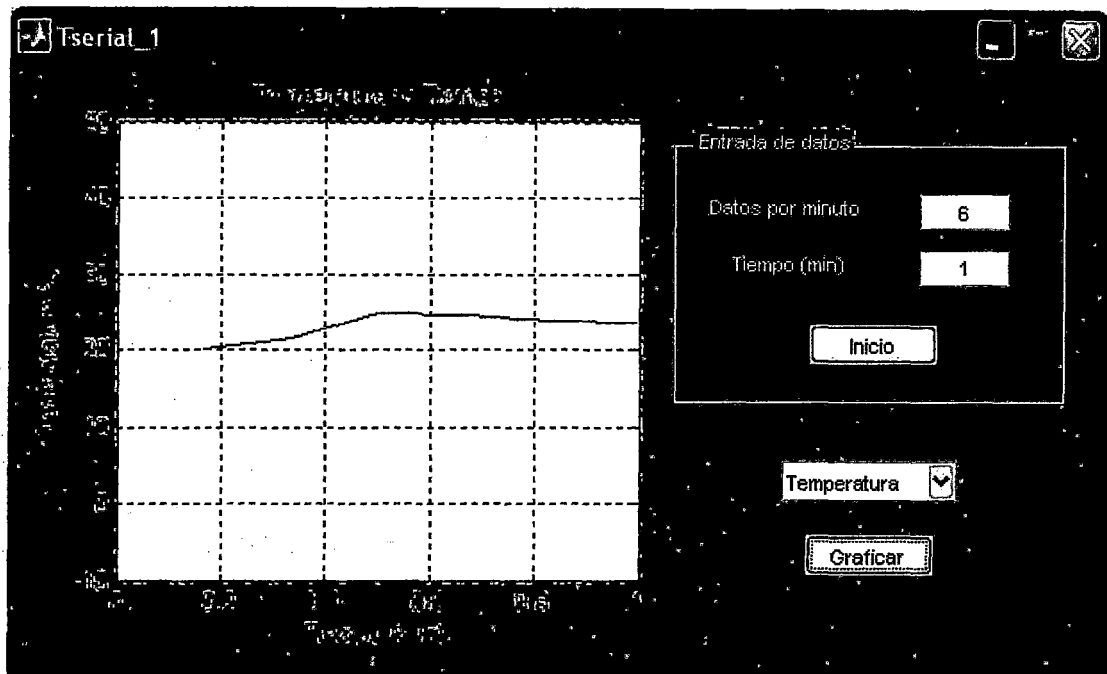


Fig.24: Ejecutable adquiriendo datos de temperatura y humedad

Para obtener una gráfica es necesario tener un vector de datos y un vector de tiempo, la matriz en donde se almacena toda la información de las mediciones es KT, es conveniente entonces tener esta matriz guardada para su uso posterior en otro programa o subprograma. La solución es guardarla con un archivo *.mat o uno *.xls, optamos por el último ya que se puede abrir desde Windows y es fácil de trabajar.

```
xlswrite('Data.xls',KT,'Hojal','A1');
```

Se hace lo propio con el tiempo y el número de datos por minuto

```
dd=[dm, tmp];
xlswrite('inicio.xls',dd,'Hojal','A1');
```

Con la información de estas dos matrices se puede construir la gráfica desde otra sección del programa usando "xlsread" y las opciones de ploteo:

```

x=xlsread('Data.xls');
dd=xlsread('Inicio.xls');
dm=dd(1);
tmp=dd(2);
dt=dm*tmp; %numero de datos totales
tt=zeros(1,dt); %eje x
tr=60/dm; %tiempo real en segundos
tr=tr/60; %tiempo real en minutos
tt(1,1)=tr;
for i=2:dt
    tt(1,i)=tt(1,i-1)+tr;
end
switch opcion1
    case 1
        plot(tt,x(:,1))%plot(tt,x(1,:))
        title('Temperatura vs Tiempo');
        xlabel('Tiempo (min)');
        ylabel('Temperatura °C');
        ylim([-10 50])
        grid;
    case 2
        plot(tt,x(:,2))
        title('Humedad vs Tiempo');
        xlabel('Tiempo (min)');
        ylabel('Humedad %');
        ylim([10 90])
        grid;
end

```

Este código va dentro de Callback asociado al Pushbutton con texto “Graficar” asociado al Listbox que contiene las opciones para graficar temperatura y humedad en este caso.

3.8. Multiplexión, comunicación serial PIC - PIC

Hasta ahora hemos visto como un solo microcontrolador 16F876A ha recibido y procesado señales analógicas, se ha comunicado en forma serial con una EEPROM, una PC y logró mostrar sus datos por una pantalla LCD. Pero qué sucede si se requiere tomar datos en los distintos puntos de un laboratorio o área de trabajo? Una solución sería tener un número N de data loggers y ubicarlos en las posiciones adecuadas. Otra solución es usar un microcontrolador adicional para adquirir y procesar las señales analógicas, este microcontrolador es el 12F675 el cual solo posee 8 pines como se muestra en Fig.25, dos de ellos son usados como entrada analógica y un tercero como salida serial, este microcontrolador al igual que el 16F876A posee un conversor

análogo/digital de 10 bits, por lo cual el tratamiento de errores sigue siendo el mismo; el 12F675 y el 18F876A se comunicarán en formato serial TTL o RS232 según la distancia de separación que se requiera, la distancia teórica en RS232 es 30 m y en TTL 2 m.

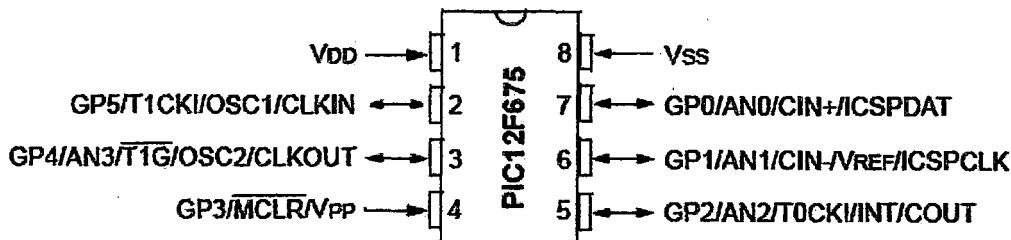


Fig.25: Microcontrolador PIC 12F675, vista de los periféricos y pines de alimentación VDD: +5V y VSS: 0V [3]

Se tiene entonces un sistema compuesto por un microcontrolador 12F675 que recibe y calcula la señal analógica del sensor de humedad HS1101 y del sensor de temperatura LM35, a este equipo ya le podemos llamar termohigrómetro. Tener N termohigrómetros que envíen sus datos en forma serial al máster 16F876A (multiplexión por división de tiempo) es más económico que tener N data loggers y cumple la misma función, para lograr esto debemos programar cada termohigrómetro con un desfase de tiempo. A continuación se muestra un ejemplo, asumiendo un desfase de 10 segundos y 6 termohigrómetros, es decir se tomarán 6 datos por minuto para el máster, un dato por minuto para cada termohigrómetro:

```

.....
'Configuración ADC y declaración de variables'
.....
DelayMS 10000 'Tiempo que varía en cada termohigrómetro
Main:
Temp = ADIn AN1 'Adquisición de la señal de temperatura
.....
'código correspondiente al cálculo de la temperatura'
.....
envio:
DelayMS 10
Puls = Counter GPIO.0,100 'Adquisición de la señal de humedad
.....
'código correspondiente al cálculo de la humedad'
.....
SerOut GPIO.2,3313, [DEC3 Temp ,DEC3 Puls]
DelayMS 59880

```

La primera pausa (**DelayMS 10000**) cambia en cada uno de los termohigrómetros, el primero tendrá un desfase de 0 segundos, 10000 ms corresponde al segundo, 20000 al tercero y así hasta 50000 ms para el sexto, esa secuencia se puede ver en Fig.26 correspondiente a una multiplexión por división de tiempo (TDM). Las siguientes líneas de código ya son conocidas, en la tercera línea (**Temp = ADIn AN1**) la variable “Temp” recibe la señal analógica de temperatura por GP1 (Equivalente a RA1 en el 16F876A), en la línea 6 la variable “Puls” recibe el número de pulsos que llegan a GP0 durante 100 ms, luego en la línea 7 se envían los datos seriales a 300 baudios por segundo por el pin GP2, en la línea 8 se programa una pausa de 59880 ms, sumando los tiempos de procesamiento de las señales analógicas se obtiene un minuto, y finalmente en la línea 9 se reinicia el proceso para tomar datos nuevos.

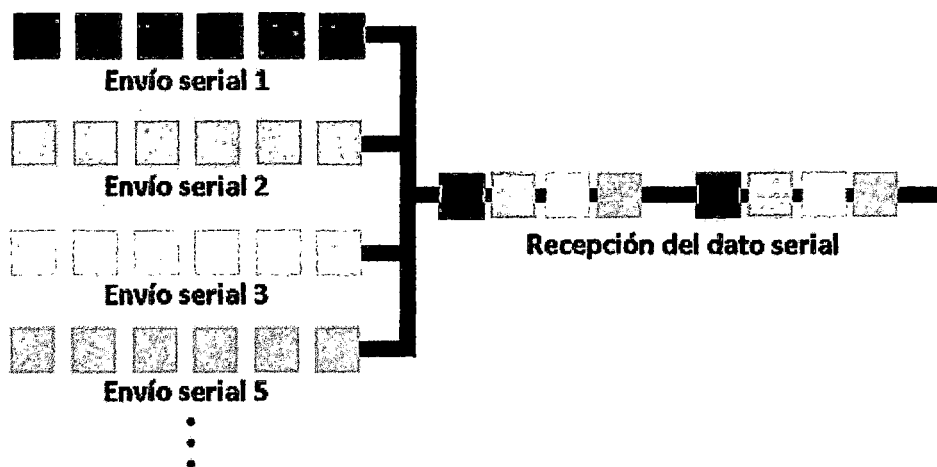


Fig.26: Multiplexión TDM entre Los PIC 12F675 y el PIC master 16F876A

3.9. Resultados

Se analiza en primer lugar la temperatura. En la Tabla 10 se muestra la temperatura que marca el data logger (termómetro), la temperatura convencionalmente verdadera (T.C.V) es la indicación del termómetro, la corrección es la diferencia de medición entre el termómetro y el patrón. El procedimiento para la evaluación de los datos tiene pequeñas variaciones en distintos laboratorios de metrología, para tomar estos datos, se

usa un baño líquido y se espera 10 minutos para alcanzar el equilibrio térmico. Una vez establecido el equilibrio térmico se toman datos cada minuto para cada temperatura, el promedio de 10 datos es el que se muestra en la Tabla 10.

Tabla 10: Calibración de temperatura

Indicación del termómetro (°C)	Corrección (°C)	T.C.V	Incertidumbre ±(°C)
0,0	0,1	0,1	0,2
10,0	0,1	10,1	0,1
20,0	0,1	20,1	0,1
30,0	-0,1	29,9	0,5
40,0	-0,1	39,9	0,9
50,0	0,0	50,0	0,5

La incertidumbre tiene que ver con el grado de dispersión de los valores obtenidos, se obtiene por medios estadísticos, en donde se estima la influencia del entorno en la medición y los resultados de mediciones anteriores. De modo que no se atribuye la incertidumbre directamente al mesurando (data logger) sino también al patrón y al entorno.

Los datos mostrados en la Tabla 11 corresponden a humedad relativa, el procedimiento fue diferente en este caso, la calibración se llevó a cabo en una cámara cerrada de vidrio y espuma de poliestireno (tecnopor), variando la humedad interior con un deshumecedor y una bandeja con agua. Se toma un solo dato una vez alcanzado el equilibrio, esto es cada 15 minutos.

Tabla 11: Calibración de humedad relativa

Indicación del higrómetro (%HR)	Corrección (%HR)	T.C.V	Incertidumbre ±(%HR)
33,0	-0,5	32,5	4,0
52,0	-0,9	51,1	4,0
71,0	-0,5	70,5	4,0
86,0	0,4	86,4	4,0

Los datos de la Tabla 12 muestran los resultados en la medición de longitud, UNIMETRO nunca había calibrado un sensor ultrasónico, no debe haber muchas empresas de metrología que lo hayan hecho. De modo que se creó un procedimiento,

empieza por una mesa nivelada junto a una pared, la regla se coloca sobre la mesa marcando cero en contacto con la pared. El modulo SRF04 se coloca sobre un soporte universal a una altura de 50 cm sobre el nivel de la mesa y a cierta distancia de la pared, se establece dicha distancia usando una escuadra calibrada y se toma el dato mostrado en la pantalla LCD. Para longitudes largas se aleja la mesa de la pared siguiendo la perpendicular.

Tabla 12: Calibración de distancia

Indicación de del modulo SRF04 (cm)	Corrección (cm)	T.C.V	Incertidumbre ±(cm)
50,0	-0,01	50,0	0,35
100,5	0,49	100,0	0,35
150,0	-0,01	150,0	0,35
200,0	-0,01	200,0	0,35
249,5	-0,51	250,0	0,35
300,0	-0,01	300,0	0,35

Capítulo 4

Discusiones

4.1. Procesamiento de datos

En el procesamiento surge un desfase de milisegundos en el PIC, al procesar datos durante horas se hace evidente, esto se debe a que la ejecución de cada comando lleva algunos microsegundos que suman un retardo no programado, cuando el programa es extenso y los datos a procesar son grandes, como en este caso, este problema se torna evidente; así para un retardo programado de 60 segundos, el retardo real es de 61 segundos. Este retardo no programado se reduce aumentando la frecuencia del cristal de cuarzo (oscilador externo), este determina la velocidad de procesamiento de los datos, si ponemos un oscilador externo de 3 GHz el retardo es ínfimo como en el caso de las PC. En la práctica los PIC de la serie 16F trabajan a lo más con cristales de 20 MHz. La solución a este problema es programar tiempos ligeramente menores, por ejemplo, se programa el PIC para adquirir y procesar los datos de temperatura y humedad en un segundo, pero de 99X ms no de 1000. Al programar en Matlab tiempos de adquisición de medio segundo, se obtiene al inicio una transmisión correcta desde el PIC, pero conforme va trabajando el sistema los datos se deforman, es por esto que el tiempo de adquisición “no es considerado” en el código:

```
%set(PS,'Timeout',0.5); % 0.5 segundos de tiempo de espera
```

Al anularlo, Matlab recibe datos seriales sin importar cuánto demoren en llegar, el envío incluso puede no ser periódico. El dato adquirido se encuentra en código ASCII y está compuesto por tres valores numéricos, se separa cada valor y se convierten a cadena, recién al ser cadena se puede convertir a entero, se juntan los tres valores y se asigna el dato a la matriz de datos. Este proceso se repite cada medio segundo aproximadamente (ver código Matlab en Apéndices).

4.2. Correcciones e incertidumbres

Corrección e incertidumbre son dos conceptos con los que se trabaja mucho en metrología, son los parámetros que tienen mayor importancia entre los resultados de un certificado de calibración.

La corrección es la diferencia entre el equipo a calibrar y el patrón que usa la institución calibradora, existe otra corrección para el patrón ya que fue calibrado con un patrón más preciso, a esto se le llama trazabilidad y se muestra en los certificados (ver Apéndice D). Por otro lado la incertidumbre tiene relación con la parte estadística y los factores externos a la medición. Según el Centro Español de Metrología (CEM), que es la base de las normativas metrológicas de INDECOPI, las posibles fuentes de incertidumbre son:

- a) Definición incompleta del mensurando
- b) Realización imperfecta de la definición del mensurando
- c) Muestra no representativa del mensurando
- d) Conocimiento incompleto de los efectos de las condiciones ambientales sobre la medición, o medición imperfecta de dichas condiciones ambientales
- e) Lectura sesgada de instrumentos analógicos, por parte del personal técnico
- f) Resolución finita del instrumento de medida o umbral de discriminación
- g) Valores inexactos de los patrones de medida o de los materiales de referencia
- h) Valores inexactos de constantes y otros parámetros tomados de fuentes externas y utilizadas en el algoritmo de tratamiento de los datos
- i) Aproximaciones e hipótesis establecidas en el método y en el procedimiento de medida
- j) Variaciones en las observaciones repetidas del mensurando, en condiciones aparentemente idénticas

Estas fuentes no son necesariamente independientes, y algunas de ellas, desde a) hasta i), pueden contribuir a la j). Claro está, que un efecto sistemático no identificado no puede ser tenido en cuenta en la evaluación de la incertidumbre del resultado de una medición, aunque contribuirá a su error.

Teniendo en cuenta esto, analicemos la incertidumbre generada por la electrónica.

El ADC del PIC 16F876A es de 10 bits, por tanto el número de estados (escalones) es 1023 y la escala de voltaje es de 0 V a 5 V, la mínima resolución será:

$$R_{\min} = \frac{5V}{2^{10} - 1} = 4.88 \text{ mV}$$

La incertidumbre producida por la digitalización quedará asociada a esta mínima resolución (que equivale a 0.5 °C), es decir la incertidumbre sería $\pm 2.44 \text{ mV}$ que corresponde a 0.25 °C. Por otro lado según el datasheet del LM35 la incertidumbre asociada a los extremos -55 °C y 150 °C es 0.75 °C, mientras la incertidumbre típica es 0.25 °C; como medimos temperaturas de 0 a 50°C, con 10 mV por grado Celsius, la mínima amplificación para lograr la escala de 0 V a 5 V del ADC es de 10, pero surgió un problema con esta amplificación. Dicho problema se debió a que el voltaje de alimentación elegido para el LM35 fue 5 V (puede ser hasta 35 V), el voltaje de salida del OPAM nunca llega a ser igual a 5 V, es decir para 50 °C no salían 500 mV del LM35, esto se soluciona de dos formas, llevando el voltaje de alimentación del LM35 a 12 V o con la misma alimentación de 5 V hacer una amplificación por 5, con salida de 0 V a 2.5 V del OPAM.

Los errores producidos por amplificación tienen dos causas principales y ambas tienen que ver con la tensión de offset (2.9 mV en condiciones normales), esta es la tensión que sale del OPAM cuando no hay voltaje de entrada. La primera causa es la temperatura, el LM358 no opera igual a 25 °C que a 50 °C, la tensión de offset aumenta con la temperatura; la segunda es el ruido de fondo, los OPAM generan tensión de offset mayor cuando aumenta el ruido de fondo, en aplicaciones con alta ganancia o gran ancho de banda el ruido puede generar errores de consideración. En el datasheet se presenta una gráfica que muestra el comportamiento del LM358 con la temperatura, pero no se habla del ruido, por lo que estimar cuantitativamente el error producido por el OPAM sería a mi criterio, inapropiado; solo se toma el valor máximo dado en el datasheet, que es 7.0 mV. La forma más confiable de saber cómo afecta la amplificación a la medición es calibrando con un patrón que tenga trazabilidad. En los resultados se muestra que el error nunca supera 0.5 °C, de modo que al 0.25 °C que es producto del

ADC, y al 0.25 °C que es propio del LM35, se suma un error mínimo debido a la amplificación.

Con respecto a la humedad, la configuración es tal como se recomienda en el datasheet, de modo que no debería haber errores mayores de 4 % HR en teoría, esto se comprueba con la calibración, una aclaración importante, en el datasheet no se habla de la ecuación de compensación entre humedad y temperatura. Este error de 4% HR planteado por los fabricantes es un estimado. Programando según la ecuación que relaciona frecuencia y humedad del datasheet, se obtuvo en las primeras pruebas menos de 3 % de corrección en el rango entre 30 % RH y 70 % RH, por encima de 70 % RH la corrección puede llegar hasta 5%, ni que decir de la incertidumbre, es necesario alterar esta ecuación en todo el rango. En el datasheet la ecuación entre frecuencia y humedad es de tercer grado, aunque se aproxima mucho a una lineal, esta resulta ser inexacta, pues depende mucho del Timer 555 que se use, en el datasheet se mencionan unos cuantos solamente, también depende de las condiciones ambientales y la calidad de los componentes como las resistencias. Al cambiar la ecuación compensando la relación entre humedad y temperatura (empíricamente) se obtiene una corrección menor al 1% y una incertidumbre de 4% en todo el rango como se muestra en el certificado (ver Apéndices).

Con respecto al sensor ultrasónico, si se busca reducir costos es correcto hacer la pregunta, ¿Por qué no hacer un modulo? Cuya respuesta es: sí se hizo. El detalle está en la estabilidad del sistema. Cuando se construye un equipo se busca, entre otras cosas, que el valor mostrado sea estable y armando una variante del SRF04 no se consiguió, el modulo diseñado tuvo un costo en materiales de S/.30.00 aproximadamente, pero la lectura era inestable, variaba en ± 2 cm. Esto se debe principalmente a la calidad de los componentes utilizados. Hay otra desventaja que es el tamaño, mientras que el SRF04 mide 5.0 cm de largo por 2.5 cm de ancho, el módulo diseñado mide el triple. No hubo problemas en el tratamiento de errores (para ver las ecuaciones de conversión revisar Apéndices).

4.3. Evaluación crítica de los resultados

La razón por la cual se realizó este proyecto fue “reducir los costos de un data logger”, medir temperatura, humedad y distancia en ambientes distintos. Este sistema lo logra y lo hace con gran precisión, el costo en materiales sin presentación plástica es de S/. 74.00 para un data logger de temperatura y humedad, y S/. 85.00 extra si se desea medir nivel de líquidos (sensor ultrasónico), los costos aumentan si se desea hacer multiplexión usando los PIC 12F675. El equipo diseñado es más versátil que una DAQ, pequeño (8x9cm) y fácil de transportar. En la Tabla 13 se muestra un presupuesto comprando componentes por unidad (hay una gran diferencia al por mayor).

Tabla 13: Costos unitarios de componentes

Componentes	Costo unitario (S/.)
LM35	4,50
HS1101	17,50
Modulo SRF 04	85,00
LM555	0,50
MAX232	1,50
LM2904	4,00
EEPROM 24LC256	4,00
PIC 16F876A	14,00
PIC 12F675	4,50
LM7805	0,80
Pantalla LCD	14,00
Baquelita (20x20 cm)	4,00
Condensadores (ciento)	4,00
Resistencias (ciento)	1,00
Conectores/cables	5,00

El proyecto ,sin embargo, tiene una debilidad y está en la parte comercial. Para la venta se requiere un control de calidad el cual debe realizarse en una empresa de metrología, esto incrementa el costo en S/. 100.00 por equipo, una solución sería comprar un patrón que cuesta por encima de S/. 2000.00 y calibrar cada equipo que se produce. Otro inconveniente es la licencia de Matlab ya que no se pueden vender ejecutables hechos en Matlab sin pagar una licencia a MathWorks. Un problema menor sería el conector DB9 cada vez menos computadoras lo tienen, lo que obliga poner un transversor extra que pase datos seriales de RS232 a USB, o usar un PIC de la serie 18F que trae

incorporada la arquitectura para la transmisión USB, como el PIC 18F2550 con el mismo número de pines y mismas funciones que el 16F876A, pero con el doble del precio de éste (se hizo una variante USB en la serie 18F, ver Apéndices).

4.4. Comparación de los resultados con otros equipos

En las empresas de metrología se tiene una gran base de datos sobre mediciones de temperatura y humedad, cito algunos equipos nuevos y los errores estimados en las calibraciones para termómetros y termohigrómetros, la calibración de distancias largas es común para reglas o cintas métricas, no se agregan al no ser comparables con un sensor ultrasónico.

Marca: RADIOSHACK

Modelo: 63-1032

Serie: No indica

Identificación: TH-10

Solicitante: TAGUMEDICA S.A.C

Tabla 14: Radioshack sensor de temperatura interno

Indicación del Termómetro	Corrección	T.C.V.	Incertidumbre
15,0	-0,6	14,4	0,1
25,0	-0,6	24,4	0,1
35,0	-0,5	34,5	0,1

T.C.V.: temperatura convencionalmente verdadera = Indicación del termómetro + Corrección

Todos los valores están reportados en grados Celsius (°C).

Tabla 15: Radioshack sensor de temperatura externo

Indicación del Termómetro	Corrección	T.C.V.	Incertidumbre
15,0	0,3	15,3	0,1
25,0	0,1	25,1	0,1
35,0	-0,1	34,9	0,1

Tabla 16: Radioshack higrómetro

Indicación del Higrómetro	Corrección	HR.C.V.	Incertidumbre
32,0	-2,3	29,7	3,5
60,0	4,1	64,1	3,5
87,0	5,3	92,3	3,5

HR.C.V.: humedad relativa convencionalmente verdadera = Indicación del higrómetro + Corrección

Todos los valores están reportados en (% HR).

Marca: MANNIX

Modelo: SAM990DW

Serie: 9466883

Identificación: No indica

Solicitante: WORLDTRONIC S.A.C

Tabla 17: Mannix termómetro

Indicación del Termómetro	Corrección	T.C.V.	Incertidumbre
15,0	-0,1	14,9	0,1
25,0	-0,1	24,9	0,1
35,0	0,1	35,1	0,1

T.C.V.: temperatura convencionalmente verdadera = Indicación del termómetro + Corrección

Todos los valores están reportados en grados Celsius (°C).

TABLA 18: Mannix higrómetro

Indicación del Higrómetro	Corrección	HR.C.V.	Incertidumbre
30,0	-0,3	29,7	3,5
56,0	8,1	64,1	3,5
94,7	-2,5	92,3	3,5

HR.C.V.: humedad relativa convencionalmente verdadera = Indicación del higrómetro + Corrección

Todos los valores están reportados en (% HR).

Marca: No indica

Origen: Nacional

Modelo: No indica

Serie: No indica

Identificación: No indica

Solicitante: INMETRO S.A.C

Tabla 19: Inmetro termómetro

Indicación del Termómetro	Corrección	T.C.V.	Incertidumbre
15,0	-0,5	14,5	0,1
25,0	-0,2	24,8	0,1
35,0	0,6	35,6	0,1

T.C.V.: temperatura convencionalmente verdadera = Indicación del termómetro + Corrección

Todos los valores están reportados en grados Celsius (°C).

Tabla 20: Inmetro higrómetro

Indicación del Higrómetro	Corrección	HR.C.V.	Incertidumbre
38,0	-6,8	31,2	4,0
64,0	-0,6	63,4	3,5
86,0	-0,5	85,5	4,0

HR.C.V.: humedad relativa convencionalmente verdadera = Indicación del higrómetro + Corrección

Todos los valores están reportados en (% HR).

Se puede apreciar que en otros equipos la corrección en temperatura puede llegar a más de 0.6 °C y en humedad relativa a más de 6 % RH, por lo que nuestros resultados quedan dentro de lo que se usa comercialmente.

Capítulo 5

Conclusiones

Habiendo terminado con el diseño, programas y arquitectura de dos data loggers con precisión comercial, se puede llegar a las siguientes conclusiones:

- El ADC de 10 bits del PIC16F876A es capaz de adquirir una señal analógica entre 0 y 5V, a la vez es suficiente para conseguir precisión comercial en los equipos que se diseñen a partir de este. Dicho microcontrolador puede funcionar también como conversor de frecuencia a voltaje (puede contar pulsos).
- El PIC16F876A es capaz de transmitir y recibir datos en forma serial, lo hace en el nivel TTL, de 0 V a 5 V. De este modo pueden transmitirse datos entre dos PICs. Para la comunicación con la PC se requiere llevar el nivel TTL (0 V a 5 V) al nivel RS232 (12 V a -12 V), lo cual se logra usando el transversor MAX232. También puede comunicarse mediante I2C con una EEPROM y otros periféricos, gracias a las características de este protocolo.
- Luego de realizar las calibraciones en empresas de metrología, los resultados arrojan una exactitud comparable a equipos comerciales, con lo cual se comprueba que el equipo diseñado es apto para los requerimientos del mercado, superando el objetivo planteado en temperatura, humedad relativa y distancia.
- El entorno de programación Matlab 2010 contiene las librerías necesarias para la comunicación serial RS232, que son accesibles a través de código. También posee un entorno gráfico para usuario que permite la programación con objetos como complemento a la programación por línea de comando. Desde Matlab se pueden crear ejecutables, de modo que los programas ya no requiere dicho entorno para funcionar.

- El costo de producción, en materiales, de un Data Logger en temperatura y humedad relativa es S/.74.00 fuera de los costos de presentación, dicho precio se reduce a menos de la mitad comprando componentes al por mayor.

Podemos concluir en general, que en este proyecto se lograron reproducir con éxito todas las etapas de los sistemas de adquisición y registro de datos, desde los sensores hasta el almacenamiento de la data en una EEPROM y en un archivo de Microsoft Excel, e incluso cumpliendo con la normativa de INDECOPI y requerimientos del mercado.

Apéndice A. Código Matlab asociado al entorno gráfico

```
function varargout = Tserial_1(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @Tserial_1_OpeningFcn, ...
                  'gui_OutputFcn',  @Tserial_1_OutputFcn, ...
                  'gui_LayoutFcn',  [], ...
                  'gui_Callback',    []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
function Tserial_1_OpeningFcn(hObject, eventdata, handles, varargin)
handles.output = hObject;
guidata(hObject, handles);
function varargout = Tserial_1_OutputFcn(hObject, eventdata, handles)
varargout{1} = handles.output;
function pushbutton_Callback(hObject, eventdata, handles)
dm = str2double(get(handles.dm, 'string'));
tmp = str2double(get(handles.tmp, 'string'));

delete(instrfindall)
PS=serial('COM3');
set(PS, 'Baudrate', 300); % se configura la velocidad a 2400 Baudios
set(PS, 'StopBits', 1); % se configura bit de parada a uno
set(PS, 'DataBits', 8); % se configura que el dato es de 8 bits, debe
estar entre 5 y 8
set(PS, 'Parity', 'none'); % se configura sin paridad
set(PS, 'Terminator', 'CR/LF'); % salto de línea, volver al inicio
set(PS, 'OutputBufferSize', 1); % "n" es el número de bytes a enviar
set(PS, 'InputBufferSize', 1); % "n" es el número de bytes a recibir
%set(PS, 'Timeout', t); % 0.2 segundos de tiempo de espera

fopen(PS);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fprintf(PS, '%s', '7');
KT=[];
dtotal=dm*tmp;
for i=1:2*dtotal % numero de datos
    K1=zeros(1,3);
    for j=1:3 %datos de tres cifras
        K1(j) = fread(PS,1,'uchar'); %entra un (1) valor, normalmente
en codigo ascci
    end;

    K1=char(K1);%el valor es convertido a cadena
    K1=str2double(K1);
    KT=[KT K1];
end;

mf=zeros(2, dtotal);
```

```

for s=1:2*dtotal
    if mod(s,2)==0
        mf(2,s/2)=KT(s); % datos de humedad
    else
        if KT(s)>600 % 60.0°C límite para transferencia de datos
            KT(s)=-1*(KT(s)-600);
        end
        mf(1,(s+1)/2)=KT(s); % datos de temperatura
    end
end
mf=mf./10;
dd=[dm,tmp];
xlswrite('inicio.xls',dd,'Hojal','A1');
xlswrite('Data.xls',mf,'Hojal','A1');
fclose(PS);
delete(PS);
clear PS;

```

```

function dm_Callback(hObject, eventdata, handles)
function dm_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
function tmp_Callback(hObject, eventdata, handles)
function tmp_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
function popupmenu_Callback(hObject, eventdata, handles)
function popupmenu_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
function graf_but_Callback(hObject, eventdata, handles)

```

```

opcion1=get(handles.popupmenu,'value');
dd=xlsread('Inicio.xls');
dm=dd(1);
tmp=dd(2);
dt=dm*tmp; %numero de datos totales
tt=zeros(1,dt); %eje x
tr=60/dm; %tiempo real en segundos
tr=tr/60; %tiempo real en minutos
tt(1,1)=tr;
for i=2:dt
    tt(1,i)=tt(1,i-1)+tr;
end

```

```

x=xlsread('Data.xls');
switch opcion1
    case 1
        plot(tt,x(:,1))
        title('Temperatura vs Tiempo');
        xlabel('Tiempo (min)');
        ylabel('Temperatura °C');
        ylim([-10 50])
        grid;
    case 2

```

```
plot(tt,x(:,2))
title('Humedad vs Tiempo');
xlabel('Tiempo (min)');
ylabel('Humedad %');
ylim([10 90])
grid;
end
```

Apéndice B. Código asociado al PIC 12F675 para la multiplexión

```
Device 12F675
Declare XTAL 4
CMCON = 0x07
ANSEL = 0x02
TRISIO = %0011

Declare ADIN_RES 10
Declare ADIN_TAD 2
Declare ADIN_STIME 100

Dim Temp As Word
Dim Puls As Word

Symbol N300 = 3313
Symbol AN1 = 1

ADCON0 = %10000000
DelayMS 9000
Main:
Temp = ADIn AN1
'.....
'código correspondiente al cálculo de la temperatura'
'.....
envio:
DelayMS 10
Puls = Counter GPIO.0,100
'.....
'código correspondiente al cálculo de la humedad'
'.....
DelayMS 8880
SerOut GPIO.2,N300, [DEC3 Temp ,DEC3 Puls]
GoTo Main
```

Apéndice C: Código asociado a la medición de distancia

```
Device = 16F876A
Include "modedefs.bas"

Declare LCD_INTERFACE 4      'Definiciones LCD
Declare LCD_LINES 2         'Definiciones LCD
Declare LCD_DTPIN PORTB.0   'Definiciones LCD
Declare LCD_ENPIN PORTB.5   'Definiciones LCD
Declare LCD_RSPIN PORTB.4   'Definiciones LCD
Declare SCL_PIN PORTB.6     'RB6 y RB7 asignados a comunicación I2C
Declare SDA_PIN PORTB.7
Start:

Symbol Ping = PORTC.7      'Definiciones I/O

Dim Trigger As Word        '1 uS Pulso de disparo
Dim Scale As Word          'raw x 10.00 =uS
Dim DATA_OUT As Byte
Dim RawToIn As Word        '1/73.746 (ancho**) a pulgadas
Dim RawToCm As Word        '1/29.034(ancho**) a centímetros
Dim Address As Word
Dim IsHigh As Byte
Dim IsLow As Byte
Dim ESER As Byte

Trigger = 1
Scale = 10
RawToIn = 889
RawToCm = .2257
IsHigh = 1
IsLow = 0
Address = 0

Dim rawDist As Word        'raw medida
Dim inches As word
Dim cm As Word

'Programa principal
main:
GoTo Get_Sonar            'leer el valor del sensor

after_Get_Sonar:
inches = rawDist ** RawToIn 'convertir a pulgadas
cm = rawDist ** RawToCm    'convertir a centímetros

Cls ' Limpiar pantalla
Print At 1,1, "Tiempo ",Dec rawDist 'mostrar tiempo (eco)
Print At 2,1, "Distancia ", Dec cm, " cm" 'mostrar distancia
DelayMS 300 'Esperar 300ms

'Es escritura de la data en la EEPROM'
GoTo main

'Subroutines
Get_Sonar:
```

```

Low Ping                                'hacer el disparo 0-1-0
PulsOut Ping, Trigger                   'activar sensor
rawDist = PulsIn Ping, 1                 'medir el tiempo del pulso alto
rawDist = rawDist * Scale                 'convertir a uS
rawDist = rawDist / 2                     'por el tiempo de ida y vuelta
GoTo after_Get_Sonar

```

Read:

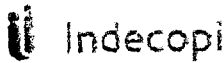
```

Print At 1,1," PARA LEER DATA "
Print At 2,1," PRES. ENTER "
SerIn PORTC.5,3313,100,Read,[ESER]
If PORTC.1 = 0 Then Or ESER = "xxx" Then
'Si el pulsador es presionado la EEPROM es leída, si el ejecutable en
'la PC envía la palabra de ejecución también se lee la EEPROM
'.....
'Lectura de la data en la EEPROM'
'.....
'Imprimir la data en la pantalla LCD'
'.....
'Envío de datos seriales'
'.....
GoTo Read

```

Apéndice D: Trazabilidad de los patrones

D.1: Certificado de calibración del patrón de humedad



"Decenio de las Personas con Discapacidad en el Perú"
"Año del Centenario de Machu Picchu para el Mundo"

Instituto Nacional de Defensa de la Competencia y de la Protección de la Propiedad Intelectual



Servicio Nacional de Metrología
Laboratorio de Temperatura

Certificado de Calibración

LT - 429 - 2011

Página 1 de 4

Expediente	57314	<p>Este certificado de calibración documenta la trazabilidad a los patrones nacionales, que realizan las unidades de medida de acuerdo con el Sistema Internacional de Unidades (SI)</p> <p>El SNM custodia, conserva y mantiene los patrones nacionales de las unidades de medida, calibra patrones secundarios, realiza mediciones y certificaciones metroológicas a solicitud de los interesados, promueve el desarrollo de la Metrología en el país y contribuye a la difusión del Sistema Legal de Unidades de medida del Perú (SLUMP).</p> <p>El SNM es miembro del Sistema Interamericano de Metrología (SIAM) y participa activamente en las intercomparaciones que éste realiza en la región.</p> <p>Con el fin de asegurar la calidad de sus mediciones el usuario está obligado a recalibrar sus instrumentos a intervalos apropiados.</p>
Solicitante	UNIMETRO S.A.C.	
Dirección	Av. Gran Chimú 451 Urb. Zárate - San Juan de Lurigancho - Lima	
Instrumento de Medición	TERMOHIGROMETRO	
Indicación	DIGITAL	
Alcance de Indicación	-20 °C a 50 °C ; 0 % a 95 % (**)	
División de escala / Resolución	0,1 °C ; 0,5 %	
Marca	LUFFT	
Modelo	OPUS 10 THI (**)	
Número de Serie	032.0510.0003.0.A.1.30 (**)	
Procedencia	ALEMANIA (**)	
Cantidad	1	
Fecha de Calibración	2011-08-09	

Este certificado de calibración sólo puede ser difundido completamente y sin modificaciones. Los extractos o modificaciones requieren la autorización del Servicio Nacional de Metrología. Certificados sin firma y sellos carecen de validez.

Sello	Fecha	Sub Jefe del Servicio Nacional de Metrología	Responsable del Laboratorio
	2011-08-09	HENRY POSTIGO LINARES	EDWIN FRANCISCO GUILLEN MESTAS

Instituto Nacional de Defensa de la Competencia y de la Protección de la Propiedad Intelectual - Indecopi
Servicio Nacional de Metrología
Calle De La Prusa 164, San Basilio Lima Perú | Telf: 2247000 Anexo 1111 | Fax: Anexo 1204
correo: snm@indecopi.gob.pe
Web: www.indecopi.gob.pe

Método de Calibración

Calibración por comparación empleando cámaras de humedad y temperatura ambientales con condiciones controladas

Lugar de Calibración

Laboratorio de Temperatura
 Calle de la Prosa 104, San Borja - Lima

Condiciones Ambientales

Temperatura	23 °C ± 1 °C
Humedad Relativa	60 % ± 2 %

Patrones de referencia

Trazabilidad	Patrón utilizado	Certificado de calibración
INMETRO BRASIL	Termohigrómetro con incertidumbre del orden de 1,1%	DIMCI 0447/2011 Marzo 2011
INDECOPI/SNM	Termómetro Digital con incertidumbre del orden de 21 mK	INDECOPI/SNM LT-307-2011 Junio 2011

Observaciones

Con fines de identificación se ha colocado una etiqueta autoadhesiva de color verde INDECOPI-SNM

(*) Datos grabado en una etiqueta adherida al instrumento de medición.

(**) Datos tomado de su manual.

Las temperaturas usadas son las de la Escala Internacional de Temperatura de 1990 (International Temperature Scale ITS-90).

Resultados de Medición

PARA EL TERMOMETRO

INDICACION DEL TERMOMETRO (°C)	TEMPERATURA CONV. VERDADERA (°C)	CORRECCION (°C)	INCERTIDUMBRE DE MEDICION (°C)
14,4	15,0	0,6	0,35
18,6	18,9	0,3	0,24
25,1	25,2	0,1	0,24

La temperatura convencionalmente verdadera (TCV) resulta de la relación:
 $TCV = \text{Indicación del termómetro} + \text{corrección}$

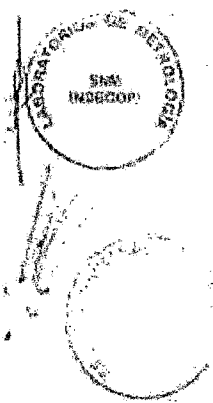
PARA EL HIGROMETRO

INDICACION DEL HIGROMETRO (% H.R.)	HUMEDAD RELATIVA CONV. VERDADERA (% H.R.)	CORRECCION (% H.R.)	INCERTIDUMBRE DE MEDICION (% H.R.)
16,0	12,4	-3,6	2,1
56,5	56,4	-3,1	2,1
80,0	85,8	-0,1	2,1

La humedad relativa convencionalmente verdadera (HCV) resulta de la relación:
 $HCV = \text{Indicación del higrómetro} + \text{corrección}$

Nota 1.- El tiempo mínimo de estabilización fue al menos de 30 minutos.

Nota 2.- La certificación IT-20 está grabada en una etiqueta adherida al instrumento de medición.



Incertidumbre

La incertidumbre reportada en el presente certificado es la incertidumbre expandida de medición que resulta de multiplicar la incertidumbre estándar por el factor de cobertura $k=2$. La incertidumbre fue determinada según la "Guía para la Expresión de la Incertidumbre en la Medición", segunda edición, julio del 2001 (Traducción al castellano efectuada por Indecopi, con autorización de ISO, de la GUM, "Guide to the Expression of Uncertainty in Measurement", corrected and reprinted in 1995, equivalente a la publicación del BIPM JCGM:100 2008, GUM 1995 with minor corrections "Evaluation of Measurement Data - Guide to the Expression of Uncertainty in Measurement").

La incertidumbre expandida de medición fue calculada a partir de los componentes de incertidumbre de los factores de influencia en la calibración. La incertidumbre indicada no incluye una estimación de variaciones a largo plazo.

Recalibración

Los resultados son válidos en el momento de la calibración. Al solicitante le corresponde disponer en su momento la ejecución de una recalibración, la cual está en función del uso, conservación y mantenimiento del instrumento de medición o a reglamentaciones vigentes.

SERVICIO NACIONAL DE METROLOGIA - SNM

El Servicio Nacional de Metrología (SNM) fue creado el 6 de Enero de 1983 mediante la Ley N° 23560 y ha sido encomendado al INDECOPI - mediante el Decreto Supremo DS-024-93 ITINCI.

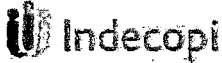
El SNM cuenta con Laboratorios Metroológicos debidamente acondicionados, instrumentos de medición de alta exactitud y personal calificado. Cuenta con un Sistema de Gestión de la Calidad que cumple con los requisitos de las Normas ISO 9001 o ISO/IEC 17025 con lo cual se constituye en una entidad capaz de brindar un servicio integral, confiable y eficaz de aseguramiento metroológico para la industria, la ciencia y el comercio.

El SNM cuenta con la cooperación técnica de organismos metroológicos internacionales de alto prestigio tales como: el Physikalisch-Technische Bundesanstalt (PTB) de Alemania; el Centro Nacional de Metrología (CENAM) de México; el National Institute of Standards and Technology (NIST) de USA; el Centro Español de Metrología (CEM) de España; el Instituto Nacional de Tecnología Industrial (INTI) de Argentina; el Instituto Nacional de Metrología (INMETRO) de Brasil; entre otros.

SISTEMA INTERAMERICANO DE METROLOGIA- SIM

El Sistema Interamericano de Metrología (SIM) es una organización regional auspiciada por la Organización de Estados Americanos (OEA), cuya finalidad es promover y fomentar el desarrollo de la metrología en los países americanos. El Servicio Nacional de Metrología - Indecopi es miembro del SIM a través de la subregión ANDIMET (Bolivia, Colombia, Ecuador, Perú y Venezuela) y participa activamente en las inter comparaciones realizadas por el SIM.

D.2: Certificado de calibración del patrón de temperatura



"Decenio de las Personas con Discapacidad en el Perú"
"Año del Centenario de Machu Picchu para el Mundo"

Instituto Nacional de Defensa de la Competencia y de la Protección de la Propiedad Intelectual



Servicio Nacional de Metrología
Laboratorio de Temperatura

Certificado de Calibración

LT - 408 - 2011

Página 1 de 4

Expediente	56750	Este certificado de calibración documenta la trazabilidad a los patrones nacionales, que realizan las unidades de medida de acuerdo con el Sistema Internacional de Unidades (SI).
Solicitante	INGENIERIA DE CALIDAD Y METROLOGIA S.A.C. - ICYM S.A.C.	
Dirección	Jr. Echeñique 973 - Magdalena del Mar,	El SNM custodia, conserva y mantiene los patrones nacionales de las unidades de medida, calibra patrones secundarios, realiza mediciones y certificaciones metrológicas a solicitud de los interesados, promueve el desarrollo de la Metrología en el país y contribuye a la difusión del Sistema Legal de Unidades de medida del Perú. (SLUMP). El SNM es miembro del Sistema Interamericano de Metrología (SIM) y participa activamente en las Intercomparaciones que éste realiza en la región. Con el fin de asegurar la calidad de sus mediciones el usuario está obligado a recalibrar sus instrumentos a intervalos apropiados.
Instrumento de Medición	TERMOMETRO DE INDICACION DIGITAL	
Alcance de Indicación	-50 °C a 400 °C (°C)	
División de escala / Resolución	0,01 °C ; 0,1 °C (°C)	
Marca	DELTA DHM	
Modelo	HD 2127.1	
Precedencia	ITALIA (°C)	
Número de Serie	10027513 (Para el indicador); 10032083 y 10032064 (Para los sensores) (°C)	
Elemento Sensor	Dos Termorresistencia de Platino de 100 Ohms	
Fecha de Calibración	2011-07-26 al 2011-08-01	

Este certificado de calibración sólo puede ser utilizado completamente y sin modificaciones. Los extractos o modificaciones requieren la autorización del Servicio Nacional de Metrología. Certificado sin firma y sello carecen de validez.

Sello	Fecha	Sub Jefe del Servicio Nacional de Metrología	Responsable del Laboratorio
	2011-08-01	 HENRY POSTIGO LINARES	 EDWIN FRANCISCO GUILLÉN MEJÍAS

Instituto Nacional de Defensa de la Competencia y de la Protección de la Propiedad Intelectual - Indecopi
Servicio Nacional de Metrología
Calle de La Prusa 104, San Jorge Lima - Perú / Tel: 2242800, Anexo 1331, Fax: Anexo 1264
e-mail: metrologia@indecopi.gob.pe
WEB: www.indecopi.gob.pe

Método de Calibración

Calibración por comparación siguiendo el procedimiento INDECOPI-SNM PC-017 "Procedimiento de Calibración de Termómetros Digitales" (1era Edición Noviembre 2007)

Lugar de Calibración

Laboratorio de Temperatura
 Calle de la Prosa 104, San Borja - Lima

Condiciones Ambientales

Temperatura	23.0 °C ± 1 °C
Humedad Relativa	59 % ± 2 %

Patrones de referencia

Trazabilidad	Patrón utilizado	Certificado de calibración
NVLAP-FLUKE-HART	Indicador Digital con 2 Módulos de barrido para PRT's con Incertidumbre del orden de 1 mΩ	A8B13D43 ; A8B13D44 ; A8B13D45 Setiembre 2008
NVLAP-BURNS	Dos Termorresistencias de Platino con Incertidumbre del orden de 10 mK	92605 y 92606 Setiembre 2008

Observaciones

Con fines de identificación se ha colocado una etiqueta autoadhesiva de color verde INDECOPI-SNM
 (*) Datos grabados en etiquetas adheridas al indicador y a los sensores de temperatura respectivamente.
 (**) Dato tomado de su manual.
 (***) La resolución de pantalla es de 0,01 °C entre -50,00 °C hasta 199,99 °C. Fuera de este alcance la resolución es de 0,1 °C.
 Las temperaturas usadas son las de la Escala Internacional de Temperatura de 1990 (International Temperature Scale ITS-90).



**Resultados de Medición
PARA EL SENSOR 70P100A**

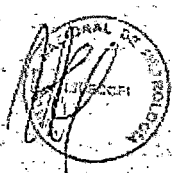
INDICACION DEL TERMOMETRO (°C)	TEMPERATURA CONVENCIONALMENTE VERDADERA (°C)	CORRECCION (°C)	INCERTIDUMBRE (°C)
-30,00	-29,756	0,244	0,011
-0,07	0,000	0,070	0,009
60,00	59,641	-0,050	0,019
120,00	119,648	-0,052	0,014
180,00	179,678	-0,022	0,014
240,0	240,08	0,08	0,07

PARA EL SENSOR 70P100B

INDICACION DEL TERMOMETRO (°C)	TEMPERATURA CONVENCIONALMENTE VERDADERA (°C)	CORRECCION (°C)	INCERTIDUMBRE (°C)
-30,00	-29,586	0,414	0,010
-0,10	0,000	0,100	0,008
60,00	60,781	-0,101	0,019
120,00	120,178	-0,178	0,014
180,00	180,178	-0,178	0,014
240,0	240,18	-0,18	0,07

La temperatura convencionalmente verdadera (TCV) resulta de la relación:
 $TCV = \text{Indicación del termómetro} + \text{Corrección}$

- Nota 1.** La profundidad de inmersión del sensor fue de 3 cm aproximadamente.
Nota 2. Tiempo de estabilización no menor a 5 minutos.
Nota 3. Las identificaciones 70P100, 70P100A y 70P100B están grabadas en etiquetas adhesivas al indicador y a los sensores respectivamente.



Incertidumbre

La incertidumbre reportada en el presente certificado es la incertidumbre expandida de medición que resulta de multiplicar la incertidumbre estándar por el factor de cobertura $k=2$. La incertidumbre fue determinada según la "Guía para la Expresión de la Incertidumbre en la Medición", segunda edición, julio del 2003 (Traducción al castellano efectuada por Indecopi, con autorización de ISO, de la GUM, "Guide to the Expression of Uncertainty in Measurement", corrected and reprinted in 1995, equivalente a la publicación del BIPM JCGM:100 2008, GUM 1995 with minor corrections "Evaluation of Measurement Data - Guide to the Expression of Uncertainty in Measurement"). La incertidumbre expandida de medición fue calculada a partir de los componentes de incertidumbre de los factores de influencia en la calibración. La incertidumbre indicada no incluye una estimación de variaciones a largo plazo.

Recalibración

Los resultados son válidos en el momento de la calibración. Al solicitante le corresponde disponer en su momento la ejecución de una recalibración, la cual está en función del uso, conservación y mantenimiento del instrumento de medición o a reglamentaciones vigentes.

SERVICIO NACIONAL DE METROLOGIA - SNM

El Servicio Nacional de Metrología (SNM) fue creado el 6 de Enero de 1983 mediante la Ley N° 23560 y ha sido encomendado al INDECOPÍ - mediante el Decreto Supremo DS-024-93 ITINCI.

El SNM cuenta con Laboratorios Metroológicos debidamente acondicionados, Instrumentos de medición de alta exactitud y personal calificado. Cuenta con un Sistema de Gestión de la Calidad que cumple con los requisitos de las Normas ISO 9001 e ISO/IEC 17025 con lo cual constituye en una entidad capaz de brindar un servicio integral, confiable y eficaz de aseguramiento metrológico para la industria, la ciencia y el comercio.

El SNM cuenta con la cooperación técnica de organismos metroológicos internacionales de alto prestigio tales como: el Physikalisch-Technische Bundesanstalt (PTB) de Alemania; el Centro Nacional de Metrología (CENAM) de México; el National Institute of Standards and Technology (NIST) de USA; el Centro Español de Metrología (CEM) de España; el Instituto Nacional de Tecnología Industrial (INTI) de Argentina; el Instituto Nacional de Metrología (INMETRO) de Brasil; entre otros.

SISTEMA INTERAMERICANO DE METROLOGIA - SIM

El Sistema Interamericano de Metrología (SIM) es una organización regional auspiciado por la Organización de Estados Americanos (OEA), cuya finalidad es promover y fomentar el desarrollo de la metrología en los países americanos. El Servicio Nacional de Metrología - Indecopi es miembro del SIM a través de la subregión ANDIMET (Bolivia, Colombia, Ecuador, Perú y Venezuela) y participa activamente en las Inter comparaciones realizadas por el SIM.

D.3: Certificado de calibración del patrón de longitud



"Decenio de las Personas con Discapacidad en el Perú"
"Año de la Integración Nacional y el Reconocimiento de Nuestra Diversidad"

Instituto Nacional de Defensa de la Competencia y de la Protección de la Propiedad Intelectual



Servicio
Nacional de Metrología

Laboratorio de Longitud y Angulo

Certificado de Calibración

LLA - 228 - 2012

Página 1 de 4

Expediente	62087	<p>Este certificado de calibración documenta la trazabilidad a los patrones nacionales, que realizan las unidades de medida de acuerdo con el Sistema Internacional de Unidades (SI)</p> <p>El SNM custodia, conserva y mantiene los patrones nacionales de las unidades de medida, calibra patrones secundarios, realiza mediciones y certificaciones metrológicas a solicitud de los interesados, promueve el desarrollo de la Metrología en el país y contribuye a la difusión del Sistema Legal de Unidades de medida del Perú. (SLUMP).</p> <p>El SNM es miembro del Sistema Interamericano de Metrología (SIM) y participa activamente en las Intercomparaciones que éste realiza en la región.</p> <p>Con el fin de asegurar la calidad de sus mediciones el usuario está obligado a recalibrar sus instrumentos a intervalos apropiados.</p>
Solicitante	UNIMETRO S.A.C.	
Dirección	Av. Gran Chimú 451 Urb. Zárate - San Juan de Lurigancho	
Instrumento de Medición	REGLA METALICA	
Alcance de Indicación	0 mm a 100 mm; 100 mm a 1 000 mm	
División de Escala	0,5 mm; 1 mm	
Marca	UYUSTOOLS	
Modelo	REJ040	
Clase de Exactitud	NO INDICA	
Número de Serie	IL-02 (*)	
Fecha de Calibración	2012-05-10	

Este certificado de calibración sólo puede ser difundido completamente y sin modificaciones. Los extractos o modificaciones requieren la autorización del Servicio Nacional de Metrología. Certificados sin firma y sellos carecen de validez.



Fecha

2012-05-10

Sub Jefe del Servicio Nacional de Metrología

HENRY POSTIGO LINARES

Responsable del laboratorio

JANNET CARRASCO TUESTA

Instituto Nacional de Defensa de la Competencia y de la Protección de la Propiedad Intelectual - Indecopi
Servicio Nacional de Metrología
Calle De La Prasa 104, San Borja Lima - Perú / Telf.: 2247800 Anexo 1331 / Fax: Anexo 1264
email: metrologia@indecopi.gob.pe
WEB: www.indecopi.gob.pe

Método de Calibración

Determinación del error de indicación por el método de comparación, utilizando una regla metálica y un magnificador óptico.
 Se tomó como referencia la Norma OIML R 35 - 1 2007 (E)

Lugar de Calibración

Laboratorio de Longitud y Angulo
 Calle de la Frosa 104, San Borja - Lima

Condiciones Ambientales

	Inicial	Final
Temperatura	20,1 °C	20,1 °C
Humedad Relativa	64,8 %	64,8 %
Presión Atmosférica	994,8 mbar	994,8 mbar

Patrones de referencia

Trazabilidad	Patrón utilizado	Certificado/Informe de calibración
Servicio Nacional de Metrología Láser estabilizado de He-Ne 633 nm Certificado 4989	LA 07 039 Regla metálica clase I INOX	INDECOPI SNM/LLA - 273 -2011
Servicio Nacional de Metrología Microscopio de herramientas INDECOPI SNM/LLA - 419 - 2009	LA 06 016 Magnificador óptico MITUTOYO Modelo 193 Con incertidumbre del orden de 5 µm	INDECOPI SNM/LLA - 082 -2011

Observaciones

Con fines de identificación se ha colocado una etiqueta autoadhesiva de color verde INDECOPI SNM.
 Los errores máximos permisibles especificados en la tabla de resultados son referenciales, debido a que el instrumento no especifica la clase de exactitud.
 El instrumento no tiene número de serie. Esta identificación se encuentra pegada sobre el instrumento.

Resultados de Medición

INDICACIÓN DE LA REGLA (mm)	ERROR ENCONTRADO (mm)	ERROR MÁXIMO PERMISIBLE CLASE II ± (mm)
0,0	0,0	0,6
20,0	0,0	0,6
40,0	0,0	0,6
60,0	0,0	0,6
80,0	0,0	0,6
100,0	0,0	0,6
200,0	0,0	0,6
300,0	0,0	0,6
400,0	-0,1	0,6
500,0	-0,1	0,6
600,0	-0,1	1,0
700,0	-0,1	1,0

INCERTIDUMBRE DE MEDICION: 0,1 mm



Instituto Nacional de Defensa de la Competencia y de la Protección de la Propiedad Intelectual - Indecopi
 Servicio Nacional de Metrología
 Calle De La Prosa 104, San Borja Lima - Perú / Telf.: 2247800 Anexo 1331 / Fax: Anexo 1264
 email: metrologia@indecopi.gob.pe
 WEB: www.indecopi.gob.pe

Incertidumbre

La incertidumbre reportada en el presente certificado es la incertidumbre expandida de medición que resulta de multiplicar la incertidumbre estándar por el factor de cobertura $k=2$. La incertidumbre fue determinada según la "Guía para la Expresión de la Incertidumbre en la Medición", segunda edición, julio del 2001 (Traducción al castellano efectuada por Indecopi, con autorización de ISO, de la GUM, "Guide to the Expression of Uncertainty in Measurement", corrected and reprinted in 1995, equivalente a la publicación del BIPM JCGM:100 2008, GUM 1995 with minor corrections "Evaluation of Measurement Data - Guide to the Expression of Uncertainty in Measurement").

La incertidumbre expandida de medición fue calculada a partir de los componentes de incertidumbre de los factores de influencia en la calibración. La incertidumbre indicada no incluye una estimación de variaciones a largo plazo.

Recalibración

Los resultados son válidos en el momento de la calibración. Al solicitante le corresponde disponer en su momento la ejecución de una recalibración, la cual está en función del uso, conservación y mantenimiento del instrumento de medición o a reglamentaciones vigentes

SERVICIO NACIONAL DE METROLOGIA - SNM

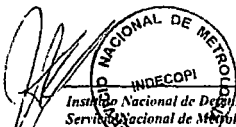
El Servicio Nacional de Metrología (SNM) fue creado el 6 de Enero de 1983 mediante la Ley N° 23560 y ha sido encomendado al INDECOPI - mediante el Decreto Supremo DS-024-93 ITINCI.

El SNM cuenta con Laboratorios Metroológicos debidamente acondicionados, instrumentos de medición de alta exactitud y personal calificado. Cuenta con un Sistema de Gestión de la Calidad que cumple con los requisitos de las Normas ISO 9001 e ISO/IEC 17025 con lo cual se constituye en una entidad capaz de brindar un servicio integral, confiable y eficaz de aseguramiento metroológico para la industria, la ciencia y el comercio.

El SNM cuenta con la cooperación técnica de organismos metroológicos internacionales de alto prestigio tales como: el Physikalisch-Technische Bundesanstalt (PTB) de Alemania; el Centro Nacional de Metrología (CENAM) de México; el National Institute of Standards and Technology (NIST) de USA; el Centro Español de Metrología (CEM) de España; el Instituto Nacional de Tecnología Industrial (INTI) de Argentina; el Instituto Nacional de Metrología (INMETRO) de Brasil; entre otros.

SISTEMA INTERAMERICANO DE METROLOGIA- SIM

El Sistema Interamericano de Metrología (SIM) es una organización regional auspiciado por la Organización de Estados Americanos (OEA), cuya finalidad es promover y fomentar el desarrollo de la metrología en los países americanos. El Servicio Nacional de Metrología - Indecopi es miembro del SIM a través de la subregión ANDIMET (Bolivia, Colombia, Ecuador, Perú y Venezuela) y participa activamente en las Inter comparaciones realizadas por el SIM.



Apéndice E: Certificados de calibración del equipo elaborado

E.1: Certificado de calibración de humedad



INGENIERÍA EN METROLOGÍA

Página 1 de 2

CERTIFICADO DE CALIBRACIÓN Y TRAZABILIDAD

N° **CTU-312-2012**

Fecha de emisión: 2012-05-18
Expediente: 510-2012

UNIDAD BAJO PRUEBA: REGISTRADOR DE DATOS DE TEMPERATURA Y HUMEDAD.
Marca: No indica Rango de indicación (HR): 10 % HR a 90 % HR
Modelo: IMR-1 División mínima (HR): 0,2 % HR
Serie: 1.0.3 Ubicación: Laboratorio
Identificación: DATALOGGER 1 (*) Procedencia: Nacional

SOLICITANTE: BSC. JORGE ENRIQUE GIRALDO MENDOZA
Dirección: Calle Santo Domingo 130 Urb. El Parral, Comas - Lima

DE LA CALIBRACIÓN: Fecha: 2012-05-16
Lugar: Laboratorio de Calibración de UNIMETRO S.A.C.
Método: Tomando como referencia la OIML R 121, Edition 1996 (E), "The scale of relative humidity of air certified against saturated salt solution".

RESULTADO DE LAS MEDICIONES:

Los resultados de las mediciones efectuadas se muestran en la página 02 del presente documento. La incertidumbre de la medición que se presenta esta basada en una incertidumbre estándar multiplicado por un factor de cobertura $k=2$, el cual proporciona un nivel de confianza de aproximadamente 95 %.

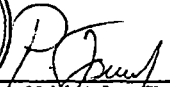
CONDICIONES AMBIENTALES:

	Inicial	Final
Temperatura	23,4°C	23,6°C
Humedad Relativa	62%	61%

PATRONES DE REFERENCIA:

Trazabilidad	Patrón utilizado	Certificado de calibración
Patrones de referencia del SNM-INDECOPI	Higrómetro patrón de 0,5 %HR de resolución	LT-429-2011 SNM-INDECOPI




Ing. Moisés A. Inga Chucos
Gerente de Metrología

PROHIBIDA SU REPRODUCCION PARCIAL O TOTAL DE ESTE DOCUMENTO SIN AUTORIZACION ESCRITA DE UNIMETRO S.A.C.

Av. Gran Chimú N° 451 Urb. Zárate, San Juan de Lurigancho - Lima
Telf.: 376-8271 Cel.: 998446498 Nextel: 142*1743 RPM *110362
Web: www.unimetrosac.com E-mail: ventas@unimetrosac.com / unimetrosac@hotmail.com

OBSERVACIONES:

- (*) Identificación asignada por BSC. JORGE ENRIQUE GIRALDO MENDOZA .
- Se colocó una etiqueta con la indicación "CALIBRADO".
- La periodicidad de la calibración depende del uso, mantenimiento y conservación del instrumento.

RESULTADOS

PARA EL SENSOR DE HUMEDAD			
Indicación del Higrómetro (%HR)	Corrección (%HR)	HRC.V. (%HR)	Incertidumbre ± (%HR)
33,0	-0,5	32,5	4,0
52,0	-0,9	51,1	4,0
71,0	-0,5	70,5	4,0
86,0	0,4	86,4	4,0

HRC.V.: humedad relativa convencionalmente verdadera = Indicación del higrómetro + Corrección
Todos los valores están reportados en (% HR).

FIN DEL DOCUMENTO



P. J. G.

PROHIBIDA SU REPRODUCCION PARCIAL O TOTAL DE ESTE DOCUMENTO SIN AUTORIZACION ESCRITA DE UNIMETRO S.A.C.

Av. Gran Chimú N° 451 Urb. Zárate, San Juan de Lurigancho - Lima
Telf.: 376-8271 Cel.: 998446498 Nextel: 142*1743 RPM *110362
Web: www.unimetrosac.com E-mail: ventas@unimetrosac.com / unimetrosac@hotmail.com

E.2: Certificado de calibración de longitud



INGENIERÍA EN METROLOGÍA

Página 1 de 2

CERTIFICADO DE CALIBRACIÓN Y TRAZABILIDAD

N° CLU-168-2012

Fecha de Emisión : 2012-05-18

Expediente : 510-2012

UNIDAD BAJO PRUEBA: REGISTRADOR DE DATOS DE LONGITUD
Marca: No indica Alcance de indicación: 3,0 cm a 320,0 cm
Modelo: IMR-2 División de escala: 0,5 cm
Identificación: DATALOGGER 2 (*) Procedencia: Nacional
Serie: 1.0.1 Ubicación: Laboratorio

SOLICITANTE: BSC. JORGE ENRIQUE GIRALDO MENDOZA
Dirección: Calle Santo Domingo 130 Urb. El Parral, Comas - Lima.

DE LA CALIBRACIÓN: Fecha: 2012-05-16
Lugar: Laboratorio de Calibración de UNIMETRO S.A.C.
Método: Tomando como referencia la Norma OIML-R035 "Medidas Materializadas de Longitud".
Tomando como referencia la norma DI-012 "Procedimiento para la Calibración de Reglas Rígidas de Trazos", Edición Digital 1, CEM de España.

RESULTADO DE LAS MEDICIONES

Los resultados de las mediciones efectuadas se muestran en la página 02 del presente documento.
La incertidumbre de la medición que se presenta esta basada en una incertidumbre estándar multiplicado por un factor de cobertura $k=2$, el cual proporciona un nivel de confianza de aproximadamente 95 %.

CONDICIONES DE CALIBRACIÓN:

	Inicial	Final
Temperatura	21,2 °C	20,7 °C
Humedad Relativa	75 %	74 %


PATRONES DE REFERENCIA:

Trazabilidad	Patrón utilizado	Certificado de calibración
Patrones de referencia del SNM-INDECOPI	Regla Metálica - Clase II	LLA-228-2012 - SNM-INDECOPI

OBSERVACIONES:

- Se colocó una etiqueta con la indicación "CALIBRADO".
- La periodicidad de la calibración depende del uso, mantenimiento y conservación del equipo.
- (*) Identificación asignada por BSC. JORGE ENRIQUE GIRALDO MENDOZA.




Ing. Moisés A. Inga Chucos
Gerente de Metrología

PROHIBIDA SU REPRODUCCION PARCIAL O TOTAL DE ESTE DOCUMENTO SIN AUTORIZACION ESCRITA DE UNIMETRO S.A.C.

Av. Gran Chimú N° 451 Urb. Zárate, San Juan de Lurigancho - Lima
Telf.: 376-8271 Cel.: 998446498 Nextel: 142*1743 RPM *110362
Web: www.unimetrosac.com E-mail: ventas@unimetrosac.com / unimetrosac@hotmail.com

RESULTADOS

INDICACIÓN DEL INSTRUMENTO (cm)	DESVIACIÓN ENCONTRADA (cm)
50,0	-0,01
100,5	0,49
150,0	-0,01
200,0	-0,01
249,5	-0,51
300,0	-0,01

INCERTIDUMBRE DE LA MEDICIÓN: 0,35 cm

FIN DEL DOCUMENTO



P. J. Cruz

PROHIBIDA SU REPRODUCCION PARCIAL O TOTAL DE ESTE DOCUMENTO SIN AUTORIZACION ESCRITA DE UNIMETRO S.A.C.

Av. Gran Chimú N° 451 Urb. Zárate, San Juan de Lurigancho - Lima
Telf.: 376-8271 Cel.: 998446498 Nextel: 142*1743 RPM *110362
Web: www.unimetrosac.com E-mail: ventas@unimetrosac.com / unimetrosac@hotmail.com

E.3: Certificado de calibración de temperatura

INMETRO[®]

Instrumentación y Gestión en Metrología

Área de Metrología

Laboratorio de Temperatura

CERTIFICADO DE CALIBRACIÓN

NÚMERO LTI-01502-2012

Expediente: 836-M-2012

Página 1 de 2

Fecha de recepción:	24 de junio de 2012	Este certificado de calibración es trazable a patrones nacionales o internacionales, los cuales realizan las unidades de acuerdo con el Sistema Internacional de Unidades (SI). Los resultados del certificado se refieren al momento y condiciones en que se realizaron las mediciones. El usuario está en la obligación de recalibrar el instrumento a intervalos adecuados, los cuales deben ser elegidos con base en las características del trabajo realizado y el tiempo de uso del instrumento. INMETRO S.A.C. no se responsabiliza de los perjuicios que pueda ocasionar el uso inadecuado de este instrumento, ni de una incorrecta interpretación de los resultados de la calibración aquí declarados.	
Objeto de Calibración:	REGISTRADOR DE DATOS DE TEMPERATURA Y HUMEDAD		
Marca / Fabricante:	No indica		
Modelo:	MR-1		
Serie / Identificación:	1.0.3 / DATALOGGER		
Procedencia:	Nacional		
Ubicación:	Laboratorio		
Alcance de indicación:	-10 °C a 60 °C; 14 °F a 140 °F		
División mínima:	0,1 °C; 0,1 °F		
Elemento sensor:	Termistor		
Longitud del sensor:	133 mm (cable)	Este certificado de calibración no podrá ser reproducido parcialmente, excepto con autorización previa por escrito del laboratorio que lo emite.	
Serie / Identificación:	No indica / No indica		
Solicitante:	JORGE ENRIQUE GIRALDO MENDOZA	Este certificado de calibración sin firma y sello carece de validez.	
Dirección:	Calle Santo Domingo 130 Urb. El Parral, Comas - Lima		
Fecha de calibración:	24 de junio de 2012	El certificado de calibración sin firma y sello carece de validez.	
Lugar de calibración:	Laboratorio de Temperatura - Área de Metrología Jr. Antisuyo 280, Urb. Zarata, San Juan de Lurigancho, Lima.		
Método de calibración:	Comparación directa con termómetro patrón calibrado, comparación realizada en un medio isotérmico de temperatura controlada, tomando como referencia el procedimiento PC-017 "Procedimiento para la calibración de termómetros digitales", 1ra. Edición, Noviembre - 2007, SNM-INDECOPI		
Condiciones ambientales:			
Temperatura inicial:	20,2 °C	Humedad relativa inicial:	58,0 %
Temperatura final:	20,3 °C	Humedad relativa final:	59,0 %
Sello	Firma/s autorizada/s	Fecha de emisión	
	Ing. Aníbal Páucar Curasima Gerencia del Servicio de Metrología	26 de Junio de 2012	

Patrones de referencia:

Patrón utilizado	Número de certificado	Trazabilidad de referencia
Termómetro patrón de 0,01 °C de resolución	LT-408-2011 Agosto 2011	GNM-00200PI

Resultados de medición

Indicación del Termómetro (°C)	Corrección (°C)	T.C.V. (°C)	Incertidumbre (°C)	Precisión (°C)
0,0	0,1	0,1	0,2	± 1,0
10,0	0,1	10,1	0,1	± 1,0
20,0	0,1	20,1	0,1	± 1,0
30,0	-0,1	29,9	0,5	± 1,0
40,0	-0,1	39,9	0,9	± 1,0
50,0	0,0	50,0	0,5	± 2,0

T.C.V.: Temperatura Comenzionalmente Verdadera = Indicación del termómetro + Corrección

Observaciones

Se adjunta una etiqueta autoadhesiva con la indicación "CALIBRADO".

Se realizó el ajuste al instrumento de medición antes de su calibración.

La profundidad de inmersión del sensor fue de 60 mm.

El tiempo de estabilización no fue menor a 15 minutos.

Incertidumbre

La incertidumbre expandida de la medición que se presenta está basada en una incertidumbre estándar multiplicado por un factor de cobertura $k=2$, el cual proporciona un nivel de confianza de aproximadamente 95 %.

La incertidumbre expandida de medición fue calculada a partir de los componentes de incertidumbre de los factores de influencia en la calibración. La incertidumbre indicada no incluye una estimación de variaciones a largo plazo.

FIN DEL DOCUMENTO



Apéndice F: Variante USB 2.0

Los puertos serie se hacen cada vez menos comunes a causa de su limitada velocidad de transferencia y su asincronía, este es el caso de RS232 y RS485. En este contexto aparece la tecnología USB, que soluciona estos problemas haciendo uso de una tecnología que ya no es cara y que supera las limitaciones anteriores. Un dispositivo USB (*Universal Serial Bus*) proporciona la información sobre sus características mediante descriptores. A partir de ahí, un cliente (maestro) puede solicitar que el dispositivo esclavo trabaje para él, si su clase y capacidades son acordes (similar al I2C). Los PIC de la serie 18F solo pueden funcionar como esclavos.

Hay 2 parámetros importantes (descriptores) en todos los dispositivos USB, su Vendor ID (VID) y su Product ID (PID); estos valores identifican al tipo de dispositivo. Partiendo de estos parámetros, se puede programar un PIC de la serie 18F para que se comunique vía USB con la PC, lo cual no es tan fácil como comunicarlo por RS232, no hay un comando específico que realice la secuencia propia del protocolo, por esto el Proton IDE tiene una herramienta llamada “Easy HID USB”. Esta herramienta nos genera archivos fuente con los que podemos establecer la comunicación USB HID 2.0, es decir, nos permite una tasa de transferencia de 60 Mega bytes por segundo, muy por encima del límite RS232. El Easy HID USB nos proporciona 3 archivos fundamentales, uno es el archivo de inclusión “USBProjectDESC.inc” que contiene las rutinas propias del protocolo. Un archivo de inclusión es una serie de líneas de código que pueden ser llamadas desde cualquier entorno de programación, es decir se puede llamar desde Mikroc, desde CCS, de MPLAB, etc. Se presenta el código a continuación con sus comentarios en inglés.

LIST

```
#DEFINE    __EPO_BUFF_SIZE          64          ; 8,
16, 32, or 64
#DEFINE    __MAX_NUM_INT            1           ; For
tracking Alternate Setting
#DEFINE    __MAX_EP_NUMBER          1           ;
UEP1
#DEFINE    __NUM_CONFIGURATIONS     1
#DEFINE    __NUM_INTERFACES         1

#DEFINE    __MODE_PP                __PPBMO
#DEFINE    __UCFG_VAL                __PUEN | __TRINT | __FS | __MODE_PP
; Full-speed
```

```
#DEFINE __USB_USE_HID ; Inform the rest of the
library subroutines that we're using a HID interface
```

```
; HID
```

```
; Endpoints Allocation
```

```
#DEFINE __HID_INTF_ID 0
#DEFINE __HID_UEP UEPI
#DEFINE __HID_BD_OUT __ep1Bo
#DEFINE __HID_INT_OUT_EP_SIZE 64
#DEFINE __HID_BD_IN __ep1Bi
#DEFINE __HID_INT_IN_EP_SIZE 64
#DEFINE __HID_NUM_OF_DSC 1
```

```
; -----
```

```
; This table is polled by the host immediately after USB Reset has
been released.
```

```
; The table defines the maximum packet size EPO can take.
```

```
; See section 9.6.1 of the Rev 1.0 USB specification.
```

```
; These fields are application dependent to be modified to meet your
specifications.
```

```
;
```

```
__DeviceDescriptor
```

```
retlw (__EndDeviceDescriptor - __DeviceDescriptor) / 2 ;
```

```
bLength Length of this descriptor
```

```
retlw __DSC_DEV ; bDescType This
```

```
is a DEVICE descriptor
```

```
dt 0x10, 0x01 ; bcdUSB Revision
```

```
1.10 low byte, high byte
```

```
retlw 0x00 ; bDeviceClass
```

```
zero means each interface operates independently
```

```
retlw 0x00 ; bDeviceSubClass
```

```
retlw 0x00 ; bDeviceProtocol
```

```
retlw __EPO_BUFF_SIZE ;
```

```
bMaxPacketSize for EPO
```

```
; idVendor low byte, high byte
```

```
dt 0x81, 0x17
```

```
; idProduct low byte, high byte
```

```
dt 0xD0, 0x07
```

```
dt 0x00, 0x00 ; bcdDevice low
```

```
byte, high byte
```

```
retlw 0x01 ; iManufacturer
```

```
retlw 0x02 ; iProduct
```

```
; iSerialNumber index
```

```
retlw 0x00
```

```
retlw __NUM_CONFIGURATIONS ;
```

```
bNumConfigurations
```

```
__EndDeviceDescriptor
```

```
; -----
```

```
; This table is retrieved by the host after the address has been set.
```

```
; This table defines the configurations available for the device.
```

```
; See section 9.6.2 of the Rev 1.0 USB specification (page 184).
```

```
; These fields are application dependent to be modified to meet your
specifications.
```

```
;
```

```
; Configuration pointer table
```

```

__USB_CD_Ptr
__Configs
    db    Low __Config1, High __Config1
    db    Low __Config1, High __Config1

; Configuration Descriptor
__Config1
    retlw (__Interfacel-__Config1) / 2 ; bLength Length of this
descriptor
    retlw __DSC_CFG                      ; bDescType 2 =
CONFIGURATION
__Config1Len
    retlw Low ((__EndConfig1 - __Config1) / 2) ; Length of this
configuration
    retlw High ((__EndConfig1 - __Config1) / 2)
    retlw 0x01                             ; bNumInterfaces
Number of interfaces
    retlw 0x01                             ; bConfigValue
Configuration Value
    retlw 0x04                             ; iConfig String
Index for this config = #01
    retlw 0xA0                             ; bmAttributes
attributes - bus powered

    ; Max power consumption (2X mA)
    retlw 0x32

__Interfacel
    retlw (__HIDDescriptor1 - __Interfacel) / 2 ; length of
descriptor
    retlw __DSC_INTF
    retlw 0x00                             ; number of
interface, 0 based array
    retlw 0x00                             ; alternate
setting
    retlw 0x02                             ; number of
endpoints used in this interface
    retlw 0x03                             ; interface CLASS
- assigned by the USB
    retlw 0x00                             ; boot device
    retlw 0x00                             ; interface
protocol - mouse
    retlw 0x00                             ; index to
string descriptor that describes this interface
__HIDDescriptor1
    retlw (__Endpoint1In - __HIDDescriptor1) / 2 ; descriptor size
(9 bytes)
    retlw 0x21                             ; descriptor type
(HID)
    dt 0x00, 0x00                          ; HID CLASS
release number (1.00) low byte, high byte
    retlw 0x00                             ; Localised
country code (none)
    retlw 0x01                             ; Number of HID
CLASS descriptor to follow (1)
    retlw 0x22                             ; Report
descriptor type (HID)
__ReportDescriptor1Len
    retlw Low ((__EndReportDescriptor1 - __ReportDescriptor1) / 2)
    retlw High ((__EndReportDescriptor1 - __ReportDescriptor1) / 2)
__Endpoint1In

```

```

        retlw (__EndPoint1Out - __Endpoint1In) / 2      ; length of
descriptor
        retlw __DSC_EP
        retlw 0x81      ; EP1, IN
        retlw 0x03      ; Interrupt
        dt    0x40, 0x00 ; This should be
the size of the endpoint buffer low byte, high byte
        retlw 0x0A      ; Polling
interval
__EndPoint1Out
        retlw (__EndConfig1 - __EndPoint1Out) / 2 ; Length of this
Endpoint Descriptor
        retlw __DSC_EP      ; bDescriptorType
= 5 for Endpoint Descriptor
        retlw 0x01      ; Endpoint number
& direction
        retlw 0x03      ; Transfer type
supported by this Endpoint
        dt    0x40, 0x00 ; This should be
the size of the endpoint buffer low byte, high byte
        retlw 0x0A      ; Polling
interval
__EndConfig1

__ReportDescriptor1
; vendor defined usage page
retlw 0x06
retlw 0x00
retlw 0xFF

; vendor defined usage
retlw 0x09
retlw 0x00

; collection(application)
retlw 0xA1
retlw 0x01

; *** INPUT REPORT ***

; vendor defined usage
retlw 0x09
retlw 0x01

retlw 0x15      ; logical minimum (-128)
retlw 0x80      ;
retlw 0x25      ; logical maximum (127)
retlw 0x7F      ;
retlw 0x35      ; Physical Minimum (0)
retlw 0x00      ;
retlw 0x45      ; Physical Maximum (255)
retlw 0xFF      ;

; report size in bits
retlw 0x75
retlw 0x08

; report count (number of fields)
retlw 0x95
retlw 0x08

```

```

; Input (Data, Variable, Absolute)
retlw 0x81
retlw 0x02

; *** OUTPUT REPORT ***

; vendor defined usage
retlw 0x09 ; usage (Vendor Defined)
retlw 0x02 ;

retlw 0x15 ; logical minimum (-128)
retlw 0x80 ;
retlw 0x25 ; logical maximum (127)
retlw 0x7F ;
retlw 0x35 ; Physical Minimum (0)
retlw 0x00 ;
retlw 0x45 ; Physical Maximum (255)
retlw 0xFF ;

; report size in bits
retlw 0x75
retlw 0x08

; report count (number of fields)
retlw 0x95
retlw 0x08

; Output (Data, Variable, Absolute)
retlw 0x91
retlw 0x02

retlw 0xC0 ; end collection
__EndReportDescriptor1

; String pointer table
__USB_SD_Ptr
__Strings
db Low __String0, High __String0
db Low __String1, High __String1
db Low __String2, High __String2
db Low __String3, High __String3
db Low __String4, High __String4

__String0
retlw (__String1 - __String0) / 2 ; Length of string
retlw __DSC_STR ; Descriptor type
3
dt 0x09, 0x04 ; Language
ID (0x0409 as defined by MS)

;company name
__String1
retlw (__String2 - __String1) / 2
retlw __DSC_STR

retlw 'u'
retlw 0x00
retlw 's'
retlw 0x00
retlw 'b'
retlw 0x00

```

```

;product name
__String2
    retlw (__String3 - __String2) / 2
    retlw __DSC_STR

    retlw 'u'
    retlw 0x00
    retlw 's'
    retlw 0x00
    retlw 'b'
    retlw 0x00

; serial number
__String3
    retlw (__String4 - __String3) / 2
    retlw __DSC_STR

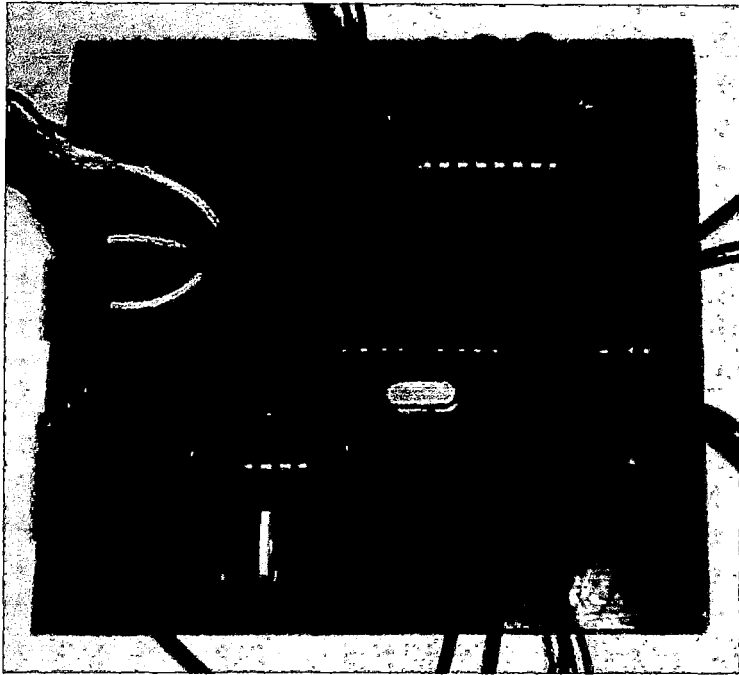
__String4
    retlw (__String5 - __String4) / 2
    retlw __DSC_STR
    dt    0
__String5

```

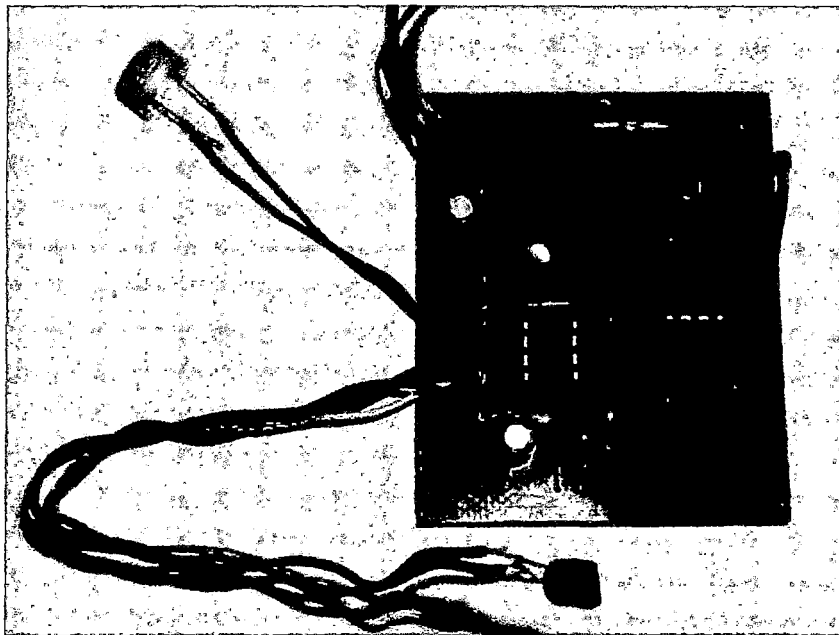
El siguiente archivo es un archivo de Visual Basic “mcHIDInterface.bas”, que usa una librería llamada “mcHID.dll” propia del Easy HID USB, el tercer archivo es una plantilla para programar en Proton IDE “USBProject.bas”. Tanto el mcHIDInterface.bas como el USBProjectDESC.inc tienen el código Vendor ID (VID) y el Product ID (PID) que identifican el dispositivo, esto permite comunicar el PIC con la PC.

Cabe agregar que tanto el PIC 18F2550 como el 18F4550 tienen la arquitectura para soportar estas velocidades de transferencia, operan a 48 MHz, tienen ADC, opción de comunicación I2C y RS232 estándar, siendo el 18F2550 el más parecido al 16F876A por sus 28 pines y disposición de sus puertos. El entorno gráfico se realiza en Visual Basic 5 CCE, que es la versión libre de Visual Basic, para esto se usa el mcHIDInterface.bas y la “form” asociada. Visual Basic 5 CCE posee grandes ventajas sobre el Matlab tales como la velocidad de procesamiento, la ligereza del ejecutable, la facilidad para trabajar con objetos y sobre todos, que es software libre.

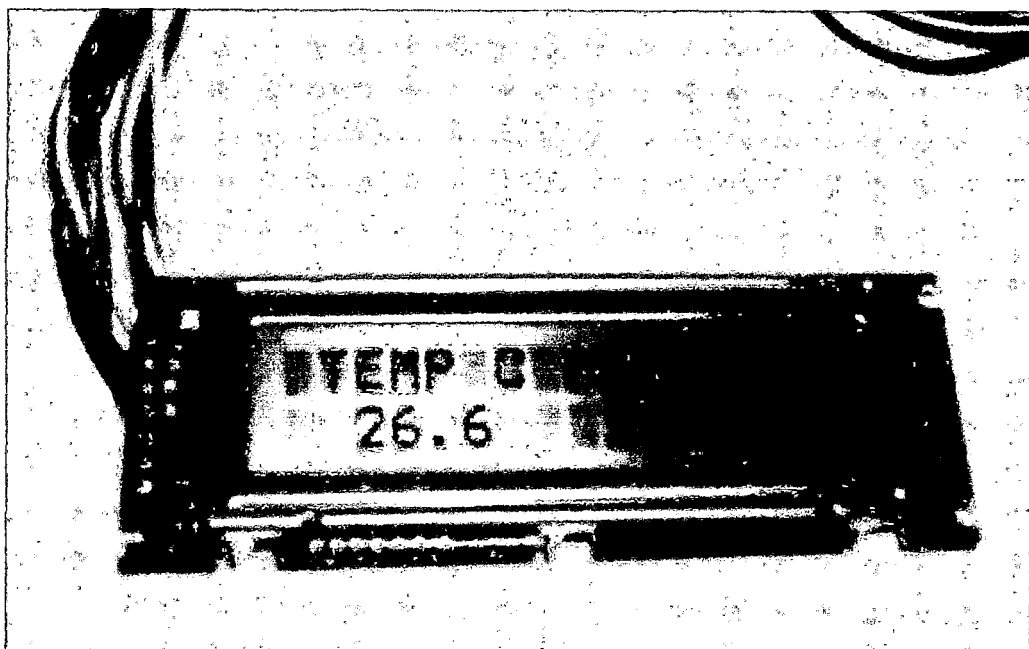
Imágenes del equipo elaborado



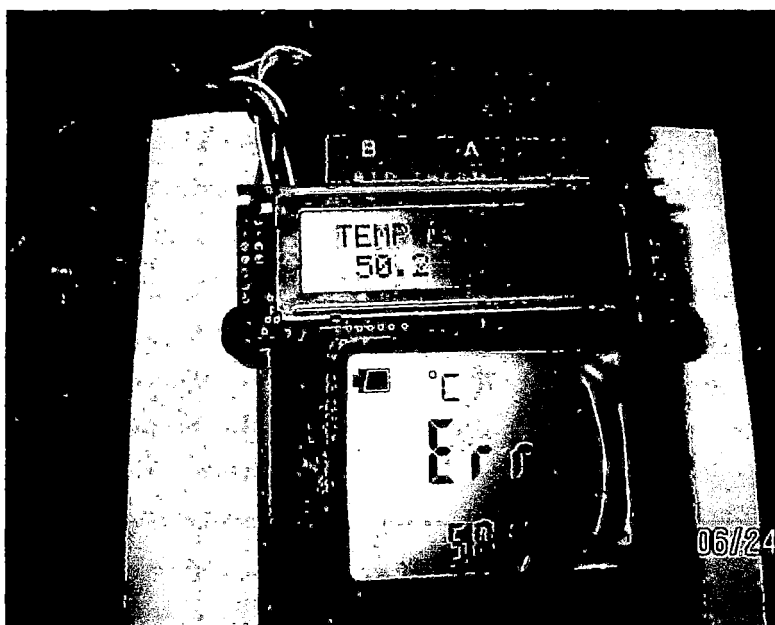
Imag.1: Registrador, resaltan el PIC 16F876A, la EEPROM 24LC256 y el MAX232



Imag.2: Termohigrómetro, resaltan el OPAM 2904, el Timer LM555, el sensor LM35 y el HS1101



Imag.3: Lectura de un valor cualquiera de temperatura y humedad



Imag.4: Comparación con un patrón de temperatura

Referencias:

- [1] Microchip. PIC16F87XA. 28/40/44 - Pin Enhanced Flash Microcontrollers, 2003
- [2] Microchip. 24AA256/24LC256/24FC256 – 256K I2C CMOS Serial EEPROM, 2004
- [3] Microchip. PIC12F629/675. 8 - Pin Flash based 8 - bit. CMOS Microcontrollers, 2010.
- [4] National Semiconductor. LM35/LM35A/LM35C/LM35CA/LM35D, 1994
- [5] Fairchild Semiconductor. LM2904, LM358/LM358A, LM258/LM258A, 2002.
- [6] Humirel. HS1100/HS1101 Relative Humidity Sensor, 2002.
- [7] ST Microelectronics. NE555/SA555/SE555 General Purpose Single Bipolar Timers, 1998.
- [8] Texas Instruments. MAX232, MAX232I Dual EIA-232 Drivers/Receivers, 2002.
- [9] Philips Semiconductors. The I2C-bus Specification, 2000.
- [10] Devantech Ltd. SRF04 Ultrasonic Ranger, Technical Specification.
- [11] Dogan Ibrahim. Programación de Microcontroladores PIC. Elsevier, 2006.
- [12] Angulo Usategui, José. Microcontroladores PIC, 3ª Edición. España, Mc Graw-Hill, 2003.
- [13] Boylestad, Robert L. Electrónica: Teoría de Circuitos y Dispositivos Electrónicos, 6ª Edición. México, Pearson, 2003.
- [14] Morales Marchena, Herón. Matlab: Métodos Numéricos y Visualización Gráfica. Megabyte, 2005.
- [15] Glosario de Términos del Centro Español de Metrología CEM.
<http://www.cem.es>