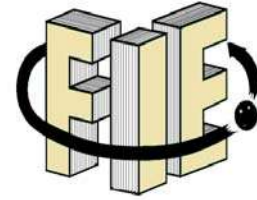




**UNIVERSIDAD MICHOACANA
DE SAN NICOLÁS DE
HIDALGO**



FACULTAD DE INGENIERÍA ELÉCTRICA

“Tarjeta multitarea configurable por puerto USB” que presenta:

Tesis

Para obtener el grado de **INGENIERO EN ELECTRÓNICA**

ORLANDO OLVERA FERNÁNDEZ

Asesor:

Ingeniero Electricista:

IGNACIO FRANCO TORRES

Morelia, Michoacán Febrero del 2015

DEDICATORIA

A lo largo de la vida las experiencias y cosas vividas que van ocurriendo te hacen la persona que eres y siempre serás, siempre existen altibajos emocionales debido a estas experiencias y sucesos ocurridos, pero siempre esto es compartido con distintas personas que por diversos motivos se han cruzado en nuestras vidas.

El tiempo en que estás viviendo una alegría es muy breve si lo pasas solo, mas el tiempo que vives una pena es en exceso largo si lo vives sin nadie a tu lado. Es por ello la importancia de tener algo de compañía siempre a tu lado.

Esta tesis quiero dedicarla a todas las personas que se han cruzado en mi vida, con las cuales he compartido alegrías, risas y sinfín de momentos agradables, pero que también hemos compartido momentos tristes, de dolor e incluso de llanto, pero a pesar de todo están ahí, y espero siempre lo estén.

Mis padres (Josefina Fernández Martínez y Roberto Olvera Ayala), novia (Ana Karen López Meza), amigos (Fer, Toño, Nacho, Alberto, Tavo, Ángel, Brian, Gil, Ada, Omar, Celeste, Marino, Diana, Zabdi, Viviana, Maribel, Érica y demás amigos que formarían una larga lista), hermano (Gabriel Olvera Fernández) y familiares que además también son mis amigos (Daniela, Adrian, Edgar, Jacinto y Pedro). Son las personas que la vida ha cruzado por mi camino, y a quienes quiero dedicar uno más de mis esfuerzos, pero principalmente lo hago porque este esfuerzo tengo la intención y deseo que al final se convierta en una alegría la cual me gustaría vivir y compartir con todas estas personas.

AGRADECIMIENTO

Existen muchas personas a quienes estoy agradecido por un sinnúmero de acciones que en determinados momentos de mi vida han realizado y espero que yo para ellos haya contribuido, pero sí no lo hice, aun hay tiempo por delante y espero poder retribuir todas las cosas buenas que por mí han hecho. De manera muy especial agradezco a mi familia ya que son ellos quienes me han hecho ser la persona que ahora soy, quizás no sea la persona más ejemplar del mundo, tampoco la peor persona de él, pero de cualquier manera todo lo que soy es gracias a mi familia.

A mi padre: Es muy difícil para mí hilar en ideas y convertir en palabras los recuerdos de mi vida entera, esos infinitos momentos vividos en los cuales tu influencia en mí, ha regido mi comportamiento, mi manera de actuar y mi manera de resolver algún problema. En muchos aspectos creo que me parezco a ti y no importa que estos sean virtudes o defectos, el hecho es que no es por casualidad y tampoco es solo por que seas mi padre, la razón por la que soy parecido a ti, es porque te admiro, porque eres mi ejemplo de vida y porque yo algún día... Quisiera ser como tú. Gracias papá por tu incondicional apoyo y cariño.

A mi madre: Quizás siempre has sido la persona que más me ha escuchado, a pesar de que sabes mamá que no soy muy comunicativo, a ti siempre te contaba las cosas que más me inquietaban y siempre a pesar de tus tareas en la casa, tenías un momento para escucharme o darme algún consejo. Mamá eras tú quien cuidabas de mí, en los tantos momentos que estaba enfermo cuando niño, tú eras quien lidiabas con todas las cosas diarias de nuestras vidas, y que quizás la rutina o monotonía, nos hace olvidar o no ver lo grande que son las pequeñas cosas. Gracias mamá por siempre ser los oídos atentos y por estar siempre a nuestro lado.

A mi hermano: Escribir algo dirigido a ti hermano, escrito de una manera propia y formal resulta un poco extraño, ya que convivo a diario contigo, eres la persona con quien convivo cada día, esto hace que nuestra manera de expresarnos y referirnos uno al otro sea digamos un poco más informal. Hermano quizás alguna vez te comente esto o alguien más lo hizo, pero de lo haré de nuevo, cuando yo supe de que tendría un hermano, jamás me sentí celoso ni amenazado de perder la atención de nuestros padres, yo deseaba con ansias tu llegada, sabía

que el compartir el cariño de los padres contigo, no era algo que me hiciera sentir mal y nunca lo hizo, al contrario logro ser otro cariño más en mi vida, algo que no conocía y deseaba, un cariño de hermanos, si compartir el cariño de los padres contigo fue un precio que tuve que pagar, resulte ganador ya que tuve mucho más de lo que pague. Gracias, además de mi hermano te has convertido en mi amigo.

Muchas gracias por existir a mi lado y por todo lo que me han dado, espero siempre poder retribuir tantas buenas acciones hacia mí. Los quiero mi familia.

ÍNDICE

Dedicatoria	ii
Agradecimiento	iii
Índice	v
Resumen	viii
Palabras clave	ix
Abstract.....	x
Keywords.....	xi
Lista de figuras	xii
Lista de tablas	xiv
Lista de símbolos y abreviaturas	xv
Capítulo 1 Descripción del proyecto	1
1.1 Introducción.....	1
1.2 Descripción de capítulos.....	2
Capítulo 1	2
Capítulo 2	2
Capítulo 3	2
Capítulo 4	3
Capítulo 5	3
1.3 Justificación	3
1.4 Objetivos de la tesis.....	4
1.5 Metodología.....	4
Capitulo 2 Marco teórico.....	5
2.1 Pre-inicio	5
2.1.1 Conocimientos básicos	5
2.1.2 Hardware	6
2.1.3 Software.....	7
2.1.4 Habilidades	8
2.2 Descripción del microcontrolador	8
2.2.1 Unidad central de procesamiento (CPU).....	9

2.2.1.1	Unidad aritmética lógica (ALU)	9
2.2.1.2	Unidad de control (UC).....	10
2.2.1.3	Registros.....	10
2.2.2	Memorias	10
2.2.2.1	Tipos de memorias ROM	10
2.2.3	Puertos Entrada/Salida (E/S)	12
2.3	PIC18F4550.....	12
2.3.1	Características principales	12
2.3.2	Diagrama de bloques y pines.....	13
2.3.3	Puertos E/S	14
2.3.3.1	Puerto A	17
2.3.3.2	Puerto B.....	18
2.3.3.3	Puerto C.....	19
2.3.3.4	Puerto D	19
2.3.3.5	Puerto E.....	19
2.3.4	Funciones del microcontrolador en la tarjeta	20
2.3.4.1	Modulador de Ancho de Pulso (PWM).....	20
2.3.4.2	Convertidos Analógico-Digital (ADC)	21
2.3.4.3	Canal de comunicación USART	22
2.4	MPLAB	24
Capítulo 3	Firmware del microcontrolador.....	25
3.1	Lógica del firmware.....	25
3.1.1	Secuencia del programa.....	26
3.1.1.1	Comandos de control.....	26
3.2	Registros asociados al firmware.....	30
3.2.1	Registros del ADC.....	30
3.2.1.1	Registro ADCON0	30
3.2.1.2	Registro ADCON1	32
3.2.1.3	Registro ADCON2.....	33
3.2.2	Registros del PWM.....	35
3.2.2.1	Registro CCP2CON	35

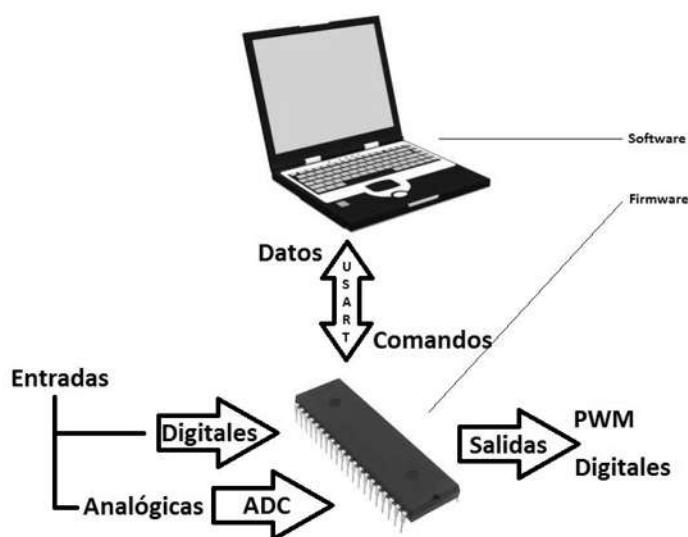
3.2.2.2	Registro CCP1CON	36
3.2.2.3	Registro T2CON	37
3.2.3	Registros del puerto serial	38
3.2.3.1	Registro TXSTA	38
3.2.3.2	Registro RCSTA	40
3.2.3.3	Registro BAUDCON	41
3.2.3.4	Registro RCREG	42
3.2.3.5	Registro TXREG	42
3.2.4	Registros de configuración de puertos	43
3.2.4.1	Registros TRIS	43
3.2.4.2	Registros LAT	43
3.2.4.3	Registros PORT	43
3.3	Códigos	44
3.3.1	Código Principal	45
3.3.2	Código de la librería PWM.h	58
3.3.3	Código de la librería display.h	59
Capítulo 4	Pruebas y resultados.	61
4.1	Software para comunicación serial “HERCULES”	61
4.2	Pruebas ADC	64
4.3	Pruebas PWM	71
4.4	Pruebas salidas digitales	74
4.5	Pruebas entradas digitales	78
4.6	Comunicación USB - Serial	79
Capítulo 5	Conclusiones, experiencias y trabajos futuros.	81
5.1	Conclusiones	81
5.2	Experiencias	82
5.3	Trabajos futuros	82
Bibliografía		83

RESUMEN

El proyecto que en este documento se describe, se trata de una tarjeta creada a partir de un microcontrolador PIC18F4550. Esta tarjeta tiene como función principal establecer comunicación bidireccional con una computadora a través del puerto serie, y con ello obtener datos de variables de entradas analógicas y digitales que se encuentren conectadas a la tarjeta y que el usuario solicite. Posteriormente a la obtención de datos el usuario puede hacer uso de una serie de comandos para establecer alguna acción de salida.

Para lograr todo esto, la tarjeta cuenta con:

- 6 entradas analógicas.
- 8 entradas digitales.
- 4 salidas digitales.
- 2 salidas de señal PWM



Esquema general del proyecto.

Un objetivo adicional con el desarrollo de esta tarjeta es que el usuario pueda usarla con el lenguaje de programación que el desee ya que solo requiere de la recepción de comandos por el puerto serie de una computadora.

PALABRAS CLAVE

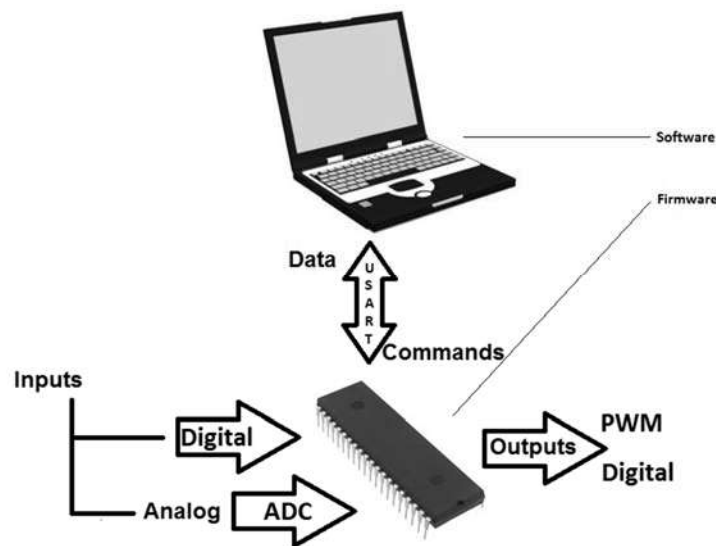
Microcontrolador, tarjeta, PIC, comunicación bidireccional, puerto serial, com putadora, comandos de control, convertidor analógico – digital, modulación por ancho de pulso.

ABSTRACT

The project described in this document, is about a card created from a PIC18 F4550 microcontroller. The card main function is to establish a two-way communication with a computer via serial port, and thereby obtain variable data from analog and digital inputs that are connected to the card at user requests. After data collecting the user may use a series of commands to set any output action.

To achieve this, the card has:

- 6 analog inputs.
- 8 digital inputs.
- 4 digital outputs.
- 2 PWM signal outputs.



General scheme of the project.

One additional objective in the development of this card is that the user can use the programming language he wants since it only requires the reception of commands through the serial port of a computer.

KEYWORDS

Microcontroller, card, P IC, bidirectional communication, serial port, computer, control commands, digital analog converter, pulse width modulation.

LISTA DE FIGURAS

Figura 2-1 Cadena de materias para microcontroladores.	6
Figura 2-2: Encapsulado del PIC18F4550.	7
Figura 2-3: Programador de PIC's Pickit3.....	7
Figura 2-4: Esquema básico de un microcontrolador.	8
Figura 2-5: Diagrama de bloques de una CPU.....	9
Figura 2-6: Diagrama de pines del PIC18F4550.....	14
Figura 2-7: Diagrama de bloques del PIC18F4550.....	16
Figura 2-8: Distribución de los puertos del PIC18F4550.....	17
Figura 2-9: Señal PWM.....	21
Figura 2-10: Conversión de señal analógica a digital.....	22
Figura 2-11: Bits del protocolo USART.	23
Figura 2-12: Diagrama de conexión del CI MAX232.....	24
Figura 3-1 Diagrama de bloques general.....	26
Figura 3-2: Voltajes de referencia.	33
Figura 3-3: ADRESH y ADRESL.....	34
Figura 3-4: Diagrama de flujo del firmware.....	44
Figura 4-1: Ventana principal del Hércules.....	61
Figura 4-2: Ventana del puerto serial.	62
Figura 4-3: Ventana del puerto serial envió de datos.	63
Figura 4-4: Circuito de pruebas.....	64
Figura 4-5: Valor leído de la conversión del AD0.	65
Figura 4-6: Valor medido en la entrada del AD0.	65
Figura 4-7: Valor leído de la conversión del AD1.	66
Figura 4-8: Valor medido en la entrada del AD1.	66
Figura 4-9: Valor leído de la conversión del AD2.	67
Figura 4-10: Valor medido en la entrada del AD2.	67
Figura 4-11: Valor leído de la conversión del AD3.	68
Figura 4- 12: Valor medido en la entrada del AD3.....	68
Figura 4-13: Valor leído de la conversión del AD4.	69
Figura 4-14: Valor medido en la entrada del AD4.	69
Figura 4-15: Valor leído de la conversión del AD5.	70
Figura 4-16: Valor medido en la entrada del AD5.....	70
Figura 4-17: PWM1 activado al 1% y PWM0 activado al 0%.....	71
Figura 4-18: PWM1 Activado al 1% y PWM0 activado al 10%.	72

Figura 4-19: PWM1 Activado al 40% y PWM0 activado al 10%.....	73
Figura 4-20: PWM1 Activado al 40% y PWM0 activado al 99%.....	74
Figura 4-21: Acción en del comando DO0.....	75
Figura 4-22: Acción en del comando DO1.....	75
Figura 4-23: Acción en del comando DO2.....	75
Figura 4-24: Acción en del comando DO3.....	75
Figura 4-25: Acción en del comando DO4.....	76
Figura 4-26: Acción en del comando DO5.....	76
Figura 4-27: Acción en del comando DO6.....	76
Figura 4-28: Acción en del comando DO7.....	76
Figura 4-29: Acción en del comando DO8.....	77
Figura 4-30: Acción en del comando DO9.....	77
Figura 4-31: Acción en del comando DOA.....	77
Figura 4-32: Acción en del comando DOB.....	77
Figura 4-33: Acción en del comando DOC.....	77
Figura 4-34: Acción en del comando DOD.....	78
Figura 4-35: Acción en del comando DOE.....	78
Figura 4-36: Acción en del comando DOF.....	78
Figura 4-37: Respuesta del comando DIT con un dato de entrada 10101010.....	79
Figura 4-38: Respuesta del comando DIT con un dato de entrada 01010101.....	79
Figura 4-39 Cable USB a serial.....	80

LISTA DE TABLAS

Tabla 2-1: Características de la familia PIC18F.....	13
Tabla 2-2: Distribución del puerto A.....	18
Tabla 2-3: Distribución del puerto B.....	18
Tabla 2-4: Distribución del puerto C.....	19
Tabla 2-5: Distribución del puerto D.....	19
Tabla 2-6: Distribución del puerto E.....	20
Tabla 3-1: Relación de comandos ADC, puertos y pines.....	27
Tabla 3-2: Relación de comandos PWM, puertos y pines.....	28
Tabla 3-3: Relación de comandos de salidas digitales, puertos y pines.	29
Tabla 3-4: Relación de puertos y pines de las entradas digitales.	29
Tabla 3-5: Bits del registro ADCON0.....	30
Tabla 3-6: Bits de selección del canal.	31
Tabla 3-7: Bits del registro ADCON1.....	32
Tabla 3-8: Bits del registro ADCON2.....	33
Tabla 3-9: Bits de configuración de las entradas analógicas.....	34
Tabla 3-10: Bits del registro CCP2CON.....	35
Tabla 3-11: Bits de configuración del modo CCP del registro CCP2CON.....	36
Tabla 3-12: Bits del registro CCP1CON.....	36
Tabla 3-13: Bits de configuración del modo CCP del registro CCP1CON.....	37
Tabla 3- 14: Bits de configuración del modo CCP del registro CCP1CON.....	37
Tabla 3-15: Bits del registro T2CON.....	38
Tabla 3- 16: Bits del registro TXSTA.....	38
Tabla 3-17: Bits del registro RCSTA.....	40
Tabla 3-18: Bits del registro BAUDCON.....	41

LISTA DE SÍMBOLOS Y ABREVIATURAS

A	<i>Amperes.</i>
ADC	<i>Convertidor analógico digital.</i>
ALU	<i>Unidad aritmética lógica.</i>
BJT	<i>Transistor de unión bipolar.</i>
CA	<i>Corriente alterna.</i>
CD	<i>Corriente directa.</i>
CMOS	<i>Semiconductor complementario de óxido metálico</i>
CPU	<i>Unidad central de procesamiento.</i>
DVD	<i>Disco versátil digital</i>
E/S	<i>Entrada / Salida.</i>
LED	<i>Diodo emisor de luz</i>
MOSFET	<i>Transistor de efecto de campo metal-óxido-semiconductor.</i>
PWM	<i>Modulador de ancho de pulso.</i>
RISC	<i>Computador con conjunto de instrucciones reducidas.</i>
SCR	<i>Rectificador controlado de silicio</i>
TRIAC	<i>Triodo para corriente alterna.</i>
TTL	<i>Lógica transistor a transistor.</i>
UC	<i>Unidad de control.</i>
USART	<i>Trasmisor receptor universal síncrono - asíncrono.</i>
USB	<i>Bus Universal en Serie.</i>
V	<i>Voltaje.</i>

CAPÍTULO 1 DESCRIPCIÓN DEL PROYECTO

1.1 Introducción

El diseño que en este documento se describe una tarjeta creada a partir de un microcontrolador PIC18F4550, la cual tiene múltiples entradas analógicas y digitales; a través de estas se puede recopilar información proveniente de variables físicas, esta información es introducida a una computadora, donde el usuario puede hacer uso de ella como él lo desee. Dicho dispositivo cuenta además con salidas analógicas y digitales para poder realizar algunas acciones de control.

La comunicación que se emplea en este diseño se realiza a través del puerto serial, sin embargo a sabiendas de que el puerto serial es un puerto que algunos equipos no poseen ya, es posible utilizar un cable el cual emula un puerto serial en un puerto USB, así finalmente el microcontrolador transmite datos por un puerto serie y la computadora los recibe por un puerto USB.

El procesamiento de la información puede realizarse bajo cualquier lenguaje de programación en el cual se pueda acceder al puerto serial, esto hace que el usuario del dispositivo, tenga la libertad de elegir el lenguaje que mejor le parezca y en el cual tenga mayor habilidad, ó también el más adecuado para el proyecto, ya que si pretende realizar una aplicación para alguna empresa tiene la posibilidad de usar software libre para crear su aplicación propia, la cual puede comercializar como un diseño de su autoría.

Finalmente para realizar algunas acciones de control, el dispositivo cuenta con un par de salidas analógicas y algunas digitales.

1.2 Descripción de capítulos

La documentación de este proyecto se realiza en cinco capítulos, donde cada uno de ellos contiene información tanto de la operación de la tarjeta como de conocimientos, habilidades y materiales necesarios para el desarrollo de la misma.

Capítulo 1

Descripción del proyecto

El capítulo uno describe de manera breve la idea central, la razón o motivación por la cual se optó por este proyecto, el objetivo que se pretende alcanzar y finalmente la manera en cómo lograr el desarrollo de este.

Capítulo 2

Marco teórico

El capítulo dos, o marco teórico, contiene una recopilación de información con la finalidad de proporcionar conocimientos necesarios para la comprensión de este documento; dentro de esta información se encuentra la descripción de hardware y software necesarios para el desarrollo de la tarjeta planteada en este documento.

Capítulo 3

Firmware del microcontrolador

En el capítulo tres como su nombre lo indica se describe el firmware del microcontrolador, dentro de esta descripción se encuentran las funciones y comandos que son leídas e interpretadas por el firmware y las cuales rigen el comportamiento de la tarjeta. También en este capítulo se describen los registros y funciones del PIC18F4550 que son usadas en esta aplicación.

Capítulo 4

Pruebas y resultados

En este capítulo se muestran fotografías de las pruebas que se realizaron así como también a los resultados que se llegaron.

Capítulo 5

Conclusiones, experiencias y trabajos futuros

Finalmente el capítulo cinco narra conclusiones, experiencias y trabajos futuros de este proyecto. Se compara los objetivos con las conclusiones y se busca el porqué de las posibles diferencias entre ellos. También en este capítulo se plasman las experiencias que se obtuvieron a lo largo de la elaboración del proyecto y la tesis.

1.3 Justificación

El siguiente proyecto es realizado pensando en la necesidad de los compañeros estudiantes de ingenierías. Por mi estancia en la escuela conozco lo necesario que en ocasiones resulta adquirir datos de variables físicas, para procesarlos dentro de una computadora. Las tarjetas que están disponibles en el mercado en ocasiones resultan ser mucho más sofisticadas de lo que se requiere, esto hace que adquirir este producto sea extremadamente costoso para un estudiante.

En la universidad los estudiantes estamos acostumbrados a usar únicamente lo que en ella se nos brinda y por comodidad no buscamos más opciones ya que cada que se nos presenta un problema, la solución la encontramos haciendo uso de lo que ahí tenemos, sin embargo esto resulta perjudicial ya que cuando se egresa, no nos damos cuenta de que ya no es tan simple resolver algunos problemas, esto debido a no tener acceso de manera tan simple a todas las facilidades que en la escuela se nos brindaban y para obtener todas esas herramientas resulta necesario invertir grandes cantidades de dinero.

1.4 Objetivos de la tesis

Elaborar un dispositivo que apoye a los alumnos en el desarrollo de sus proyectos, para que puedan trabajar con él, sin la necesidad de estar en la escuela y de tener que usar software que únicamente tienen las escuelas o público que paga por ello, pero que a la vez no deje a un lado la posibilidad de usarlos.

Se pretende que la construcción de este dispositivo no resulte un fuerte gasto para los alumnos, ya que ellos mismos podrán crear su propio dispositivo.

Que el alumno pueda desarrollar proyectos propios los cuales el podría comercializar sin problemas de derechos de autor.

1.5 Metodología

La metodología seguida es: primeramente explicar la arquitectura básica de los microcontroladores y el funcionamiento de las principales partes que lo componen, posterior a este esquema general de los microcontroladores, se procede a describir uno en específico el cual es el PIC18F4550 así como sus características, distribución de pines, puertos y funciones necesarias para este proyecto (PWM, ADC, E/S digitales y comunicación serial).

El siguiente paso es el desarrollo y operación del firmware. Una serie de comandos son los encargados de controlar las aplicaciones que se encuentran programadas, se describe cada uno de los comandos al igual que los registros que habilitan y configuran las funciones del microcontrolador.

Finalmente gracias a un software llamado Hércules se establece comunicación entre la computadora y la tarjeta, esto a través de un cable que establece una interface de comunicación USB-Serial; son enviados comandos desde la computadora hasta la tarjeta, esta interpreta los comandos y realiza la acción correspondiente a él.

CAPITULO 2 MARCO TEÓRICO

2.1 Pre-inicio

Esta sección está dedicada a las personas que deseen desarrollar el dispositivo que en este proyecto describe. Aquí se enlistan algunos conocimientos, hardware, software y habilidades necesarias para la creación de este dispositivo el cual es llamado: *“Tarjeta multitareas configurable por puerto USB”*. Aunque se pretende que la información plasmada en este documento sea comprensible por cualquier alumno, la realidad es que siempre es necesario tener conocimientos básicos.

2.1.1 Conocimientos básicos

Antes de iniciar con la información técnica de este documento, es necesario mencionar que los compañeros que pretendan hacer uso de ella, son libres de hacerlo, pero para esto es necesario que tengan algunos conocimientos básicos, los cuales les ayudaran a entender mejor cada parte de este documento.

Siguiendo el plan de estudios de la Facultad de Ingeniería Eléctrica, de la Universidad Michoacana de San Nicolás de Hidalgo, en su carrera de Ingeniería Electrónica, los conocimientos mínimos que se recomiendan, se obtendrían una vez que el alumno haya acreditado la materia de Laboratorio de microcontroladores I, y toda la cadena de materias que existe detrás de ella, dicha cadena se muestran en la figura 2-1

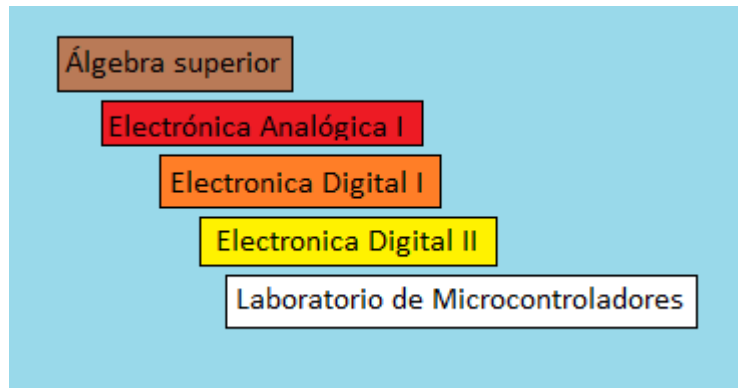


Figura 2-1 Cadena de materias para microcontroladores.

Sin embargo, además de estas materias, es necesario también tener conocimientos de lenguajes de programación, por lo cual es recomendable haber aprobado la materia de Programación de computadoras. Con el conocimiento de las materias anteriormente mencionadas, un alumno sería capaz de entender este documento, pero entre más conocimientos de electrónica y programación se tengan, más fácil resultará realizar sus proyectos.

No obstante, a pesar de los conocimientos básicos requeridos, esta tesis se elabora de la manera más clara posible para que no resulte muy compleja su comprensión, adjunto a este documento se encuentra un DVD donde se podrá encontrar materiales extra para auxiliar a quien pretenda realizar este proyecto, entre estos materiales se encuentran: manuales, tutoriales, diagramas de circuitos, imágenes, software, el código del proyecto entre algunos otros todo esto con la intención de que quien haga uso del proyecto pueda auxiliarse con este disco y también aprender algo nuevo.

2.1.2 Hardware

En el desarrollo de este dispositivo se utiliza el PIC18F4550 (ver la figura 2-2), por la razón de que este es el microcontrolador que se usa en la Facultad de Ingeniería Eléctrica, además de que la marca que fabrica este dispositivo, proporciona un software libre para programarlo que se puede descargar del sitio oficial de manera gratuita.



Figura 2-2: Encapsulado del PIC18F4550.

La programación de este microcontrolador se puede hacer a través del puerto USB cargando un archivo que genera el compilador, o por un dispositivo llamado Pickit (como el que se muestra en la figura 2-3). Para esta aplicación ha sido elegido el Pickit como programador, ya que solo se requerirá programar una vez el chip y resulta más simple solicitar apoyo en los laboratorios para la programación, que instalar todo lo necesario para poder cargar los programas desde el USB (Podrá encontrar el diagrama de conexión del pickit al PIC18F4550 en el disco adjunto).



Figura 2-3: Programador de PIC's Pickit3.

2.1.3 Software

Como ya se menciono anteriormente, el software necesario distribuido de manera gratuita por la compañía que lo crea, lo único que se requiere es descargarlo de la pagina de microchip e instalarlo, el software es el MPLAB y un compilador de lenguaje C llamado C18, los cuales

se podrán encontrar también en el DVD adjunto, así como también un tutorial de instalación y de los pasos a seguir para iniciar un nuevo proyecto.

2.1.4 Habilidades

Además de contar con los conocimientos, software y hardware necesarios, es también indispensable tener algunas habilidades en la comprensión y elaboración de circuitos, así como también algunas herramientas para desarrollar el circuito periférico básico del microcontrolador, estas habilidades no pueden ser adquiridas a través de sólo leer un documento, se necesita práctica; es por ello que se menciona en este documento, porque no hay forma de transmitir esta habilidad por este medio.

2.2 Descripción del microcontrolador

Un microcontrolador es un circuito integrado programable capaz de realizar las instrucciones que tenga almacenadas dentro de su memoria. Éstos están compuestos de varios sectores encargados de realizar tareas específicas. Un microcontrolador, al igual que una computadora, posee tres importantes unidades, éstas son: CPU (Unidad central de procesamiento), memoria y periféricos ó puertos de entrada/salida. *(Galeon.com hispavista)*

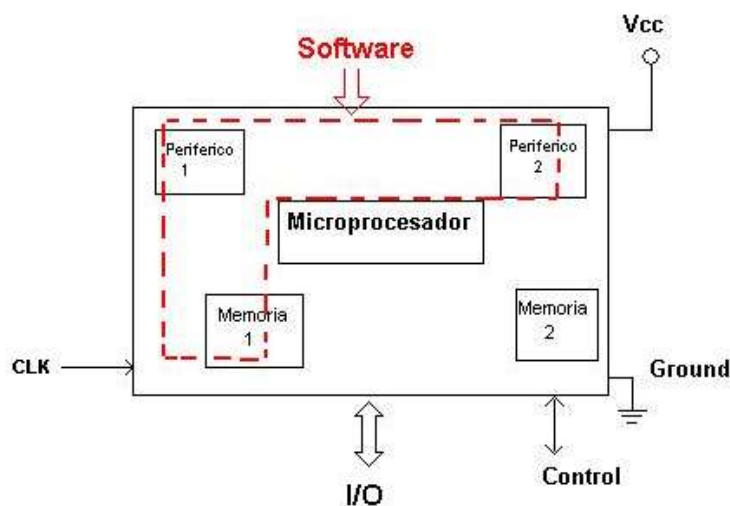


Figura 2-4: Esquema básico de un microcontrolador.

2.2.1 Unidad central de procesamiento (CPU)

La CPU o unidad central de procesamiento es un componente fundamental de todos los dispositivos programables y se encarga de interpretar las instrucciones contenidas en la memoria de programa. Las CPU son también las encargadas de controlar y dirigir las operaciones básicas del microcontrolador enviando y recibiendo direcciones de memoria, datos de un lugar a otro y señales de control, todo a través de canales de comunicación internos llamados buses.

Las CPU están constituidas internamente por otros dispositivos que realizan tareas específicas (como se muestra en la figura 2-5), la unidad aritmética lógica (ALU), la unidad de control (UC) y la unidad de memoria o también llamados registros.

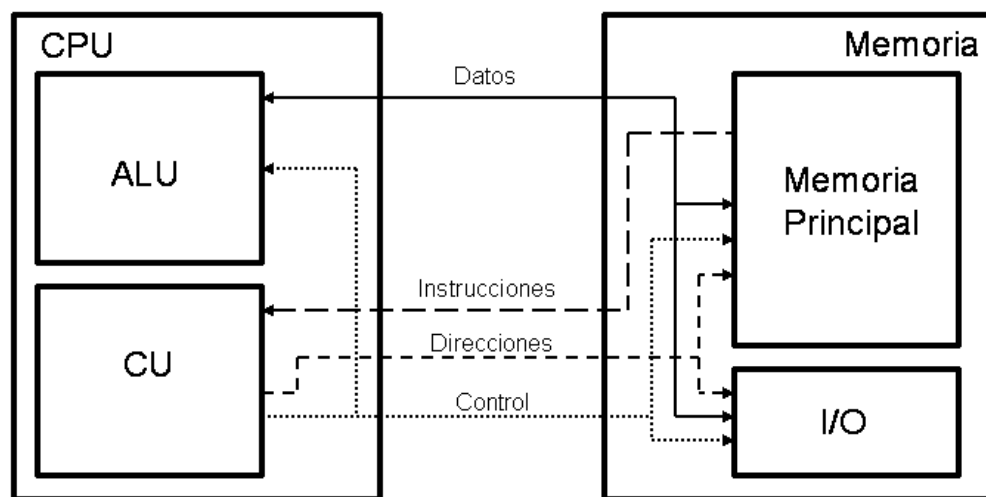


Figura 2-5: Diagrama de bloques de una CPU.

2.2.1.1 Unidad aritmética lógica (ALU)

Es la unidad encargada de realizar las operaciones aritméticas como: sumas, restas, multiplicaciones y divisiones; también determina cuando un número es negativo, positivo o cero. La ALU también ejecuta y realiza operaciones lógicas como: comparaciones, conjunciones, disyunciones y negaciones. Todas las acciones de la ALU se deciden mediante las señales de control enviadas por la UC.

2.2.1.2 Unidad de control (UC)

Controla y coordina el funcionamiento de las partes que integran un microcontrolador, determina que operaciones se deben realizar y en qué orden; asimismo sincroniza todo el proceso dependiendo de la interpretación de las instrucciones que integran los programas. Además genera el conjunto de órdenes elementales para que se realice los procesos necesarios.

2.2.1.3 Registros

El procesador cuenta con una serie de celdas o localidades de memoria que utiliza con mucha frecuencia y que forman parte de la CPU. Las localidades son también conocidas con el nombre de registros y son componentes muy importantes de un procesador, en otras palabras, son una memoria temporal para almacenar los datos básicos que el procesador requiere.

(MikroElektronika, 1998)

2.2.2 Memorias

A diferencia de las computadoras personales, un microcontrolador no posee unidades de almacenamiento masivo, sino que la memoria se encuentra integrada dentro del chip. La memoria en un microcontrolador debe ser de dos clases, una debe ser una memoria tipo ROM donde se almacenan las instrucciones del programa cargado al microcontrolador, y una memoria tipo RAM destinada al almacenamiento de variables y datos volátiles del programa contenido en la ROM.

2.2.2.1 Tipos de memorias ROM

Memoria ROM: (*Read-Only Memory*) En este caso no se “graba” el programa en memoria sino que el microcontrolador se fabrica con el programa. El costo inicial de producir un circuito de este tipo es alto, porque el diseño y producción de la máscara es un proceso costoso, sin embargo, cuando se necesitan varios miles o incluso cientos de miles de microcontroladores para una aplicación determinada, el costo inicial de producción de la máscara y el de fabricación del circuito se distribuye entre todos los circuitos de la serie y el costo final, es bastante menor que el de sus semejantes con otro tipo de memoria.

Memoria PROM: (*Programmable Read-Only Memory*) Los microcontroladores con memoria PROM se pueden programar una sola vez, con algún tipo de programador. Se utilizan en sistemas donde el programa no requiera futuras actualizaciones y para series relativamente pequeñas, donde la variante de máscara sea muy costosa, también para sistemas que requieren serialización de datos, almacenados como constantes en la memoria de programas.

Memoria EPROM: (*Erasable Programmable Read Only Memory*) Los microcontroladores con este tipo de memoria son muy fáciles de identificar porque su encapsulado es de cerámica y llevan encima una ventanita de vidrio desde la cual puede verse la oblea de silicio del microcontrolador. Se fabrican así porque la memoria EPROM es reprogramable, pero antes debe borrarse, y para ello hay que exponerla a una fuente de luz ultravioleta. (Porat & Barna, 1992, pág. 338)

Memoria EEPROM: (*Electrical Erasable Programmable Read Only Memory*) Fueron el sustituto natural de las memorias EPROM, la diferencia fundamental es que pueden ser borradas eléctricamente, por lo que la ventanilla de cristal de cuarzo y los encapsulados cerámicos no son necesarios. Al disminuir los costos de los encapsulados, los microcontroladores con este tipo de memoria se hicieron más baratos y cómodos para trabajar que sus equivalentes con memoria EPROM

Memoria Flash: En el campo de las memorias reprogramables para microcontroladores, son el último avance tecnológico en uso a gran escala, y han sustituido a los microcontroladores con memoria EEPROM.

A las ventajas de las memorias flash se le adicionan su gran densidad respecto a sus predecesoras lo que permite incrementar la cantidad de memoria de programas a un costo muy bajo. Pueden además ser programadas con las mismas tensiones de alimentación del microcontrolador, el acceso en lectura y la velocidad de programación es superior, disminución de los costos de producción.

2.2.3 Puertos Entrada/Salida (E/S)

Los puertos de entrada salida o puertos de propósito general, son registros de un microcontrolador (generalmente de longitud de 8 bits) por donde se permite leer y escribir datos en el exterior, dicho de otra forma, permiten la entrada y salida de datos. Los puertos E/S pueden ser usados en aplicaciones muy simples o muy complejas según la manera en que sean programados.

Existe en algunos microcontroladores hardware para establecer algunos tipos de comunicación, que se efectúan a través de estos puertos. Comúnmente las entradas salidas de estos puertos son digitales, sin embargo existen algunas técnicas de diseño para crear señales analógicas como el PWM (*Pulse-width modulation*). Algunos fenómenos físicos y químicos que existen a nuestro alrededor son medibles y gracias a diferentes tipos de sensores esas mediciones son convertidas a variables eléctricas analógicas y digitales; siendo los microcontroladores digitales las variables digitales pueden ser conectadas directamente a los puertos y almacenar sus datos en la memoria, sin embargo para el caso de las analógicas dentro de los puertos E/S de un microcontrolador existen convertidores analógico-digitales (ADC) los cuales como su nombre lo indica convierten variables de entrada analógicas en datos digitales.

Por el conjunto de todas estas cualidades, los puertos E/S son una parte muy importante en los microcontroladores, y es en ellos donde se convierte datos virtuales en reales y datos reales en virtuales.

2.3 PIC18F4550

2.3.1 Características principales

- Diferentes modos de manejo de energía. Tecnología nanoWatt.
- Disponibilidad de varios periféricos como pueden ser puertos I/O, módulos de timer, módulos PWM, módulos USART, convertidores A/D entre algunos otros.

- Gran cantidad de memoria.
- Arquitectura RISC avanzada tipo Harvard: 16 bits con 8 bits de datos.
- Facilidad de programación en lenguaje C.
- Hasta 32 KB de memoria de programa.
- Pila de 31 niveles.
- Hasta 2MB de RAM y 256B de EEPROM.
- Múltiples interrupciones.
- Bajo costo.

Véase la tabla 2-1

Tabla 2-1: Características de la familia PIC18F.

CARACTERISTICAS	PIC18F2455	PIC18F2450	PIC18F4455	PIC18F4450
Frecuencia de Operación	Hasta 48MHz	Hasta 48MHz	Hasta 48MHz	Hasta 48MHz
Memoria de Programa (bytes)	24.576	32.768	24.576	32.768
Memoria RAM de Datos (bytes)	2.048	2.048	2.048	2.048
Memoria EEPROM Datos (bytes)	256	256	256	256
Interrupciones	19	19	20	20
Líneas de E/S	24	24	35	35
Temporizadores	4	4	4	4
Módulos de Comparación/Captura/PWM (CCP)	2	2	1	1
Módulos de Comparación/Captura/PWM mejorado (ECCP)	0	0	1	1
Canales de Comunicación Serie	MSSP,EUSART	MSSP,EUSART	MSSP,EUSART	MSSP,EUSART
Canal USB	1	1	1	1
Puerto Paralelo de Transmisión de Datos (SPP)	0	0	1	1
Canales de Conversión A/D de 10 bits	10 Canales	10 Canales	13 Canales	13 Canales
Comparadores analógicos	2	2	2	2
Juego de instrucciones	75 (83 ext.)	75 (83 ext.)	75 (83 ext.)	75 (83 ext.)
Encapsulados	PDIP 28 pines SOIC 28 pines	PDIP 28 pines SOIC 28 pines	PDIP 40 pines QFN 40 pines TQFP 40 pines	PDIP 40 pines QFN 40 pines TQFP 40 pines

(Departamento de Ingeniería electrónica, 2005, pág. 5)

2.3.2 Diagrama de bloques y pines

El PIC18F4550 está fabricado con tecnología CMOS de altas prestaciones y encapsulado de 40 pines. En la figura 2-6 se muestra el diagrama de pines del microcontrolador.

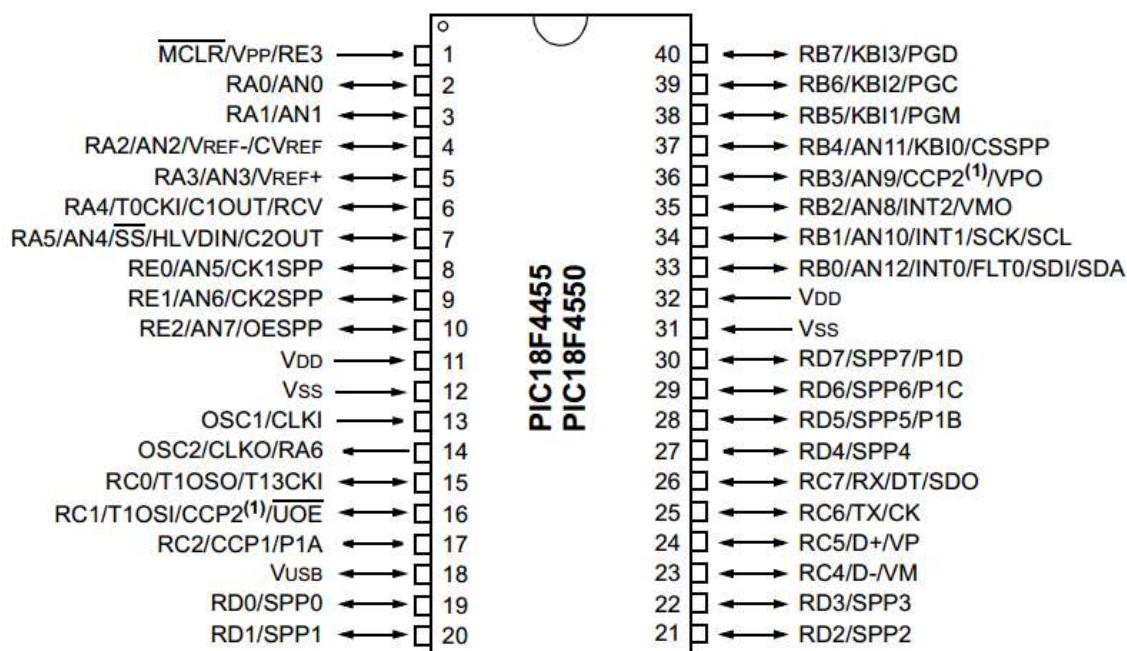


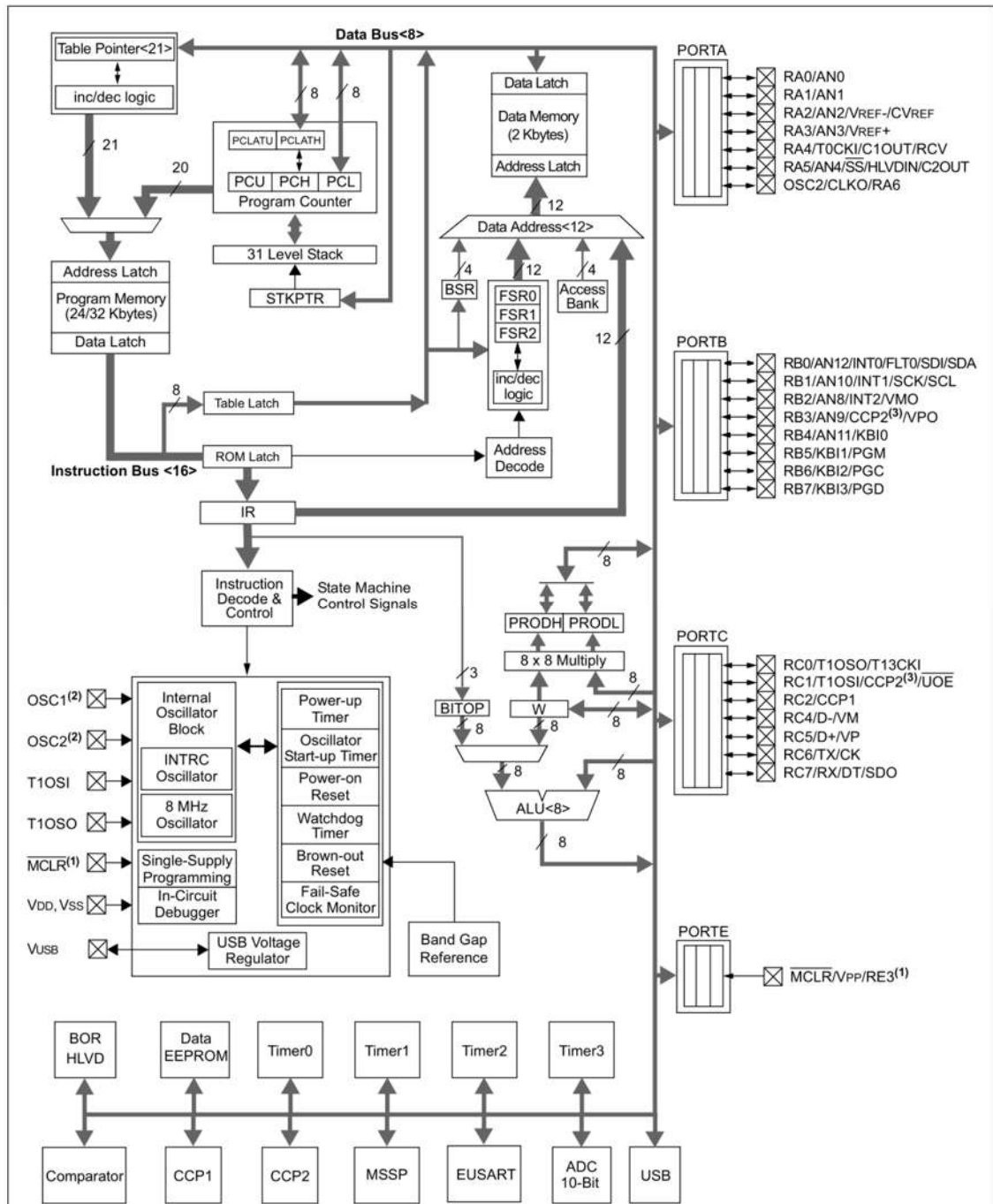
Figura 2-6: Diagrama de pines del PIC18F4550.

En la figura 2-7, se muestra el diagrama de bloques del microcontrolador, en esta figura es importante ver como los pines se dividen en grupos y son nombrados puertos, esto es algo importante ya que son en estos donde entra y sale la información, dicho en otras palabras son nuestras puertas de comunicación con el microcontrolador o también **Puertos E/S** (Si se desea información más detallada de cada pin o bloque del PIC18F4550 se debe consultar el manual de este microcontrolador, disponible en el disco adjunto). (Microchip, 2006, pág. 2)

2.3.3 Puertos E/S

Este microcontrolador cuenta con 35 líneas de entrada/salida digitales que están divididas en 5 puertos los cuales son:

- Puerto A de 7 pines y 7 bits.
- Puerto B de 8 pines y 8 bits.
- Puerto C de 8 pines y 8 bits.
- Puerto D de 8 pines y 8 bits.
- Puerto E de 4 pines y 4 bits.



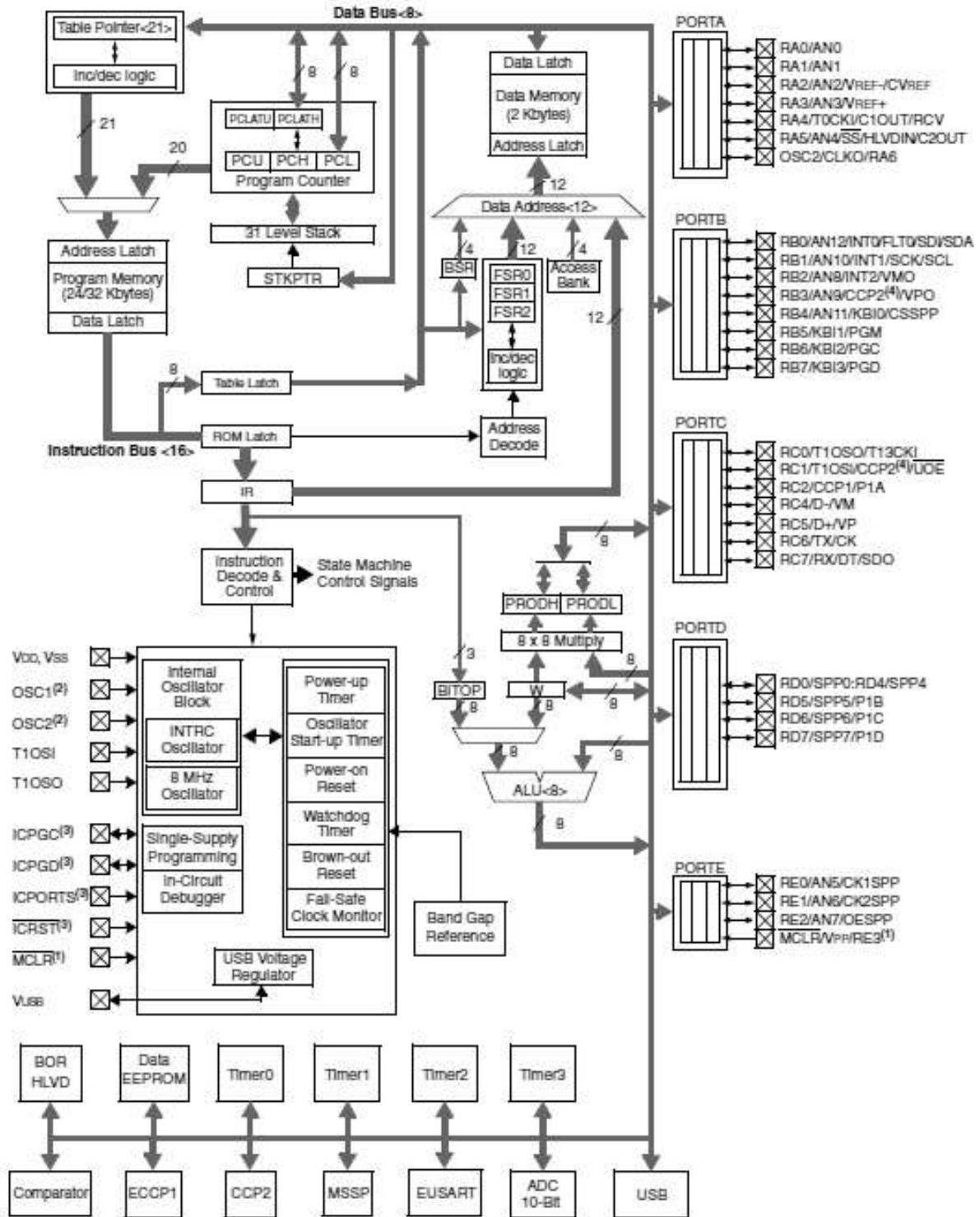


Figura 2-7: Diagrama de bloques del PIC18F4550.

(Microchip, 2006, pág. 10)

Todas las líneas digitales de E /S disponen de al menos una función alternativa. Cuando una línea trabaja en el modo alternativo no puede ser utilizada como línea digital E/S, y de igual manera en el caso opuesto.

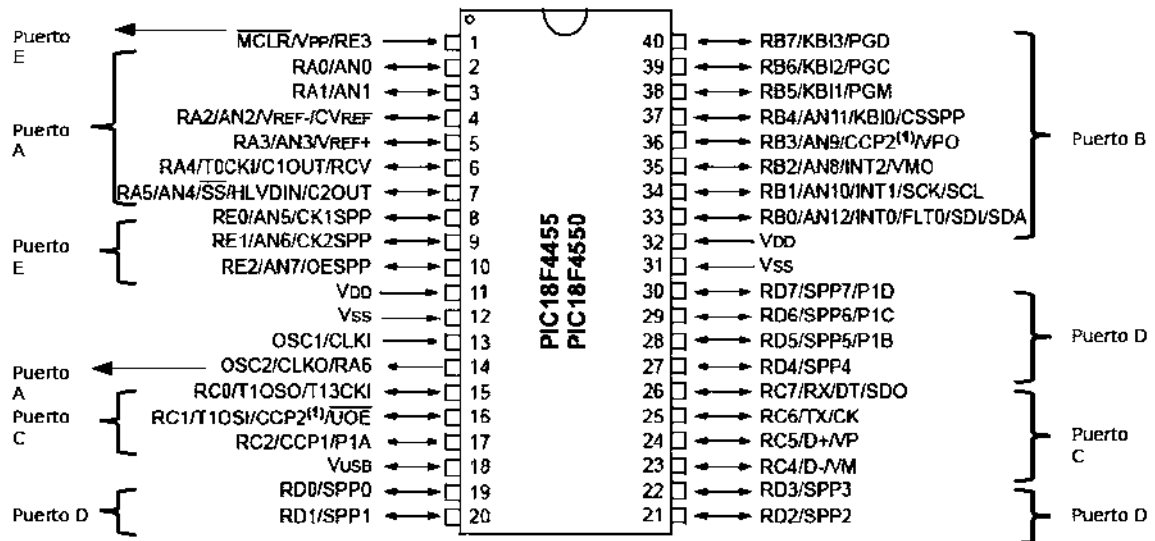


Figura 2-8: Distribución de los puertos del PIC18F4550.

Se deben tener algunas consideraciones especiales, al momento de utilizar los puertos como entrada o salida, estas consideraciones son las siguientes:

1. El pin **RA6** es compartido con el oscilador principal, en el caso de utilizar el oscilador interno, este pin puede ser utilizado como entrada o salida.
2. El pin **RC3** no está implementado
3. Los pines **RC4** y **RC5** son utilizados para la transmisión de datos del módulo **USB**, en el caso de que el módulo **USB** este deshabilitado, estos pines pueden ser utilizados únicamente como entradas.
4. El pin **RE3** es compartido con la función de **MASTER CLEAR (MCLR)**, en el caso que esta función sea deshabilitada este bit se puede utilizar únicamente como un pin de entrada, no admite ser salida.

2.3.3.1 Puerto A

- Algunas de las funciones con las que cuenta puerto A son:

- Entradas/Salidas digitales.
- Convertidor analógico/digital.
- Comparador analógico.

La tabla 2-2 indica los pines donde se encuentra el puerto y algunas de las funciones alternativas.

Tabla 2-2: Distribución del puerto A.

Puerto A	A7	A6	A5	A4	A3	A2	A1	A0
Pin		14	7	6	5	4	3	2
ADC			AN4		AN2/Vref+	AN2/Vref-	AN1	AN0
Comparador			C2OUT	C1OUT	C1IN+	C2IN+	C2IN-	C1IN-

2.3.3.2 Puerto B

- Algunas de las funciones con las que cuenta puerto B son:
- Entradas/Salidas digitales.
- Convertidor analógico/digital.
- Interrupciones Externa.
- Interrupciones por cambios en pines.
- Comunicación I2C.
- Resistencias pull-up en todas las líneas.

La tabla 2-3 indica los pines donde se encuentra el puerto y algunas de las funciones alternativas.

Tabla 2-3: Distribución del puerto B.

Puerto B	B7	B6	B5	B4	B3	B2	B1	B0
Pin	40	39 38		37	36	35	34	33
ADC				AN11	AN9	AN8	AN10	AN12
Interrupción	KBI3	KBI2 KBI1		KBI0		INT2	INT1	INT0
I2C							SCL	SDA

2.3.3.3 Puerto C

Algunas de las funciones con las que cuenta puerto C son:

- Entradas/Salidas digitales.
- Puerto USART.
- Puerto USB.
- PWM.

La tabla 2-4 indica los pines donde se encuentra el puerto y algunas de las funciones alternativas.

Tabla 2-4: Distribución del puerto C.

Puerto C	C7	C6	C5	C4	C3	C2	C1	C0
Pin	26	25	24 23			17	16 15	
USB			D+	D-				
PWM						PWM	PWM	
I2C							SDL	SDA
USART	RX	TX						

2.3.3.4 Puerto D

Algunas de las funciones con las que cuenta puerto D son:

- Entradas/Salidas digitales.
- Resistencias pull-up en todas las líneas.

La tabla 2-5 indica los pines donde se encuentra el puerto.

Tabla 2-5: Distribución del puerto D.

Puerto D	D7	D6	D5	D4	D3	D2	D1	D0
Pin	30	29	28 27 22 21 20 19					

2.3.3.5 Puerto E

Algunas de las funciones con las que cuenta puerto E son:

- Entradas/Salidas digitales
- Convertidor analógico/digital
- Reset externo

La tabla 2-6 indica los pines donde se encuentra el puerto y algunas de las funciones alternativas.

Tabla 2-6: Distribución del puerto E.

Puerto E	E7	E6	E5	E4	E3	E2	E1	E0
Pin					1	10	9	8
ADC						AN7	AN6	AN5
Reset externo					MCLR			

(Departamento de Ingeniería electrónica, 2005, págs. 70 - 83)

2.3.4 Funciones del microcontrolador en la tarjeta

La tarjeta descrita en este documento, hará uso principalmente de cuatro funciones del PIC18F4550, estas funciones son:

- PWM (Pulse-width modulation).
- ADC (Analog-to-digital converter).
- Canal de comunicación serie USART.
- Entradas y salidas Digitales.

Para cada una de estas funciones están destinados pines y pines específicos, los cuales muestran en la figura 2-6 así como también en las tablas de los puertos (Tablas 2-2, 2-3, 2-4, 2-5 y 2-6). En esta sección se describirán más a detalle cada una de estas funciones.

2.3.4.1 Modulador de Ancho de Pulso (PWM)

Las siglas PWM se ven frecuentemente en las hojas técnicas de los microcontroladores y se refieren a una forma de manejar señales digitales llamada modulación por ancho de pulso, o Pulse Width Modulation (de ahí las siglas)

El principio de funcionamiento es simple, el microcontrolador puede generar una señal cuadrada periódica con frecuencia determinada y hacer variar la relación de tiempo en el que un pin que porta esta señal permanece encendido, esta relación de tiempo encendido/apagado es llamada **ciclo de trabajo**. El hecho de que la señal esté mucho o poco tiempo encendida hace que el voltaje promedio que se refleje sea variable (Ver la figura 2-9).

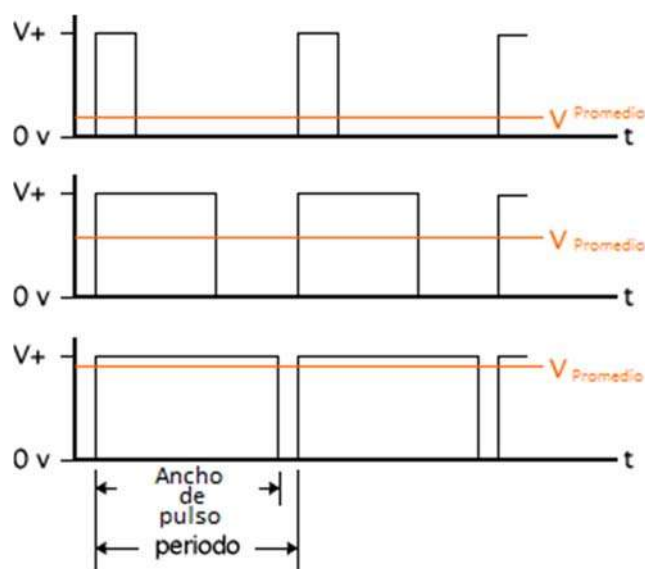


Figura 2-9: Señal PWM.

Si una señal de este tipo hace transiciones lo suficientemente rápido y es conectada a un LED (Diodo emisor de luz), nuestro ojo no será capaz de percibir esas transiciones de encendido/apagado y lo único que percibiría sería variaciones en la intensidad luz emitida por el LED.

También si una señal PWM es conectada a través de dispositivos de electrónica de potencia a un motor, esta es capaz de variar la velocidad angular del giro del motor. En general la más común de las funciones de una señal PWM es limitar con transiciones de encendido/apagado la cantidad de energía que recibe un dispositivo actuador.

2.3.4.2 Convertidos Analógico-Digital (ADC)

El desarrollo de los microcontroladores, ha permitido realizar tareas que durante años fueron hechas por sistemas electrónicos analógicos, y aunque el mundo real es análogo, una

forma de enlazar las variables analógicas con los procesos digitales es a través de los sistemas llamados convertidores de analógico-digital (ADC - Analog to Digital Converter).

El objetivo básico de un ADC es transformar una señal eléctrica analógica en un número digital equivalente.

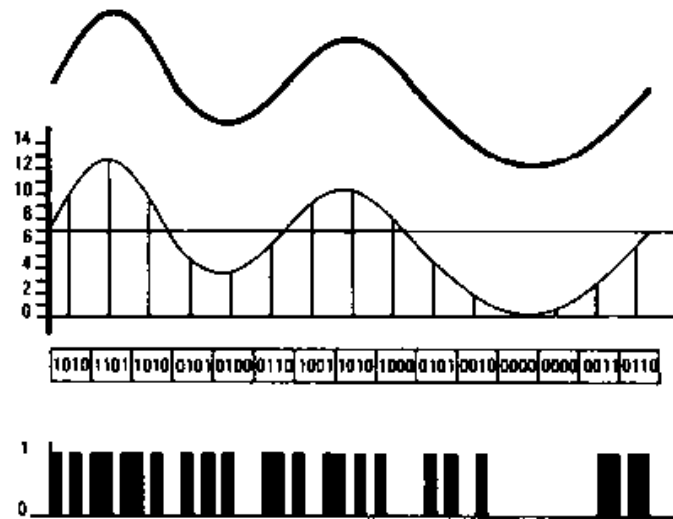


Figura 2-10: Conversión de señal analógica a digital.

Las gráficas de la figura 2-10 muestran la secuencia de conversión la señal analógica a la señal digital (código binario). Para que dicha señal ingrese al ADC, ésta debe pasar por un proceso muestreo, es decir, se toman valores discretos en instantes de tiempo de la señal analógica, lo que recibe el nombre de muestras (sampling). Finalmente como resultado se obtiene un tren de pulsos con amplitudes limitadas a la lógica usada.

2.3.4.3 Canal de comunicación USART

Es común encontrar en la mayoría de los microcontroladores un protocolo de comunicación llamado USART, este sirve para transmitir datos en serie entre varios dispositivos, ya sea entre microcontroladores o microcontroladores y computadoras. La principal diferencia con otros protocolos, es que el USART no necesita un pin de señal de reloj entre ambos dispositivos para realizar la comunicación. Cada dispositivo se programa con la velocidad a la que se va a realizar la transmisión, y después de una condición de inicio

cada dispositivo lee los bits en el tiempo acordado, por lo que sólo son necesarios dos pines para que el micro pueda enviar y recibir datos.

Para poder realizar la comunicación, lo primero que se debe establecer entre todos los dispositivos a comunicar es la velocidad a la que se va a realizar la comunicación. La comunicación comienza con una señal de Start, seguida de los bits a enviar, y se pueden seleccionar entre 5 y 9 bits a mandar, después tenemos que seleccionar si va a haber un bit de paridad para comprobar errores y por último si tenemos uno o dos bits de Stop (Figura 2-11). Estos parámetros deben de estar configurados de igual manera en todos dispositivos que se van a comunicar.

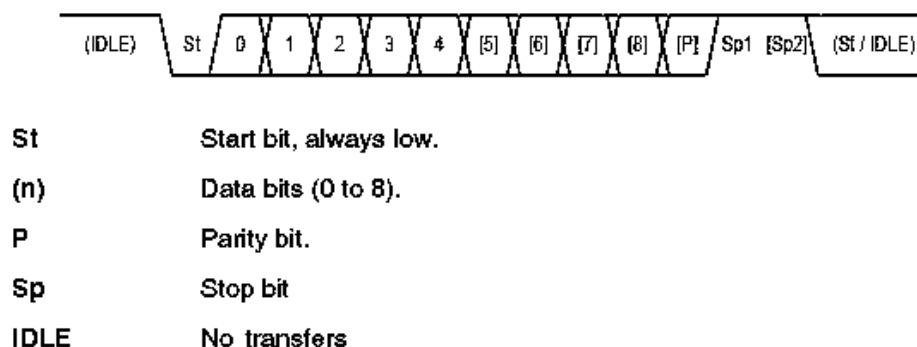


Figura 2-11: Bits del protocolo USART.

Es importante señalar que para la realización de esta comunicación entre microcontroladores y computadoras es necesario el uso de un dispositivo que sirve de interfaz de transmisión y recepción para las señales TX y RX. Este dispositivo es un circuito integrado llamado **MAX 232** y se encarga de convertir las señales del puerto serie de la computadora a niveles **TTL** compatibles con los circuitos lógicos. *(PIC's Electronica y Robótica, 2010)*

Los niveles de tensión del puerto serie de un ordenador son de entre 3v y 15v para un 0 lógico y -3v y -15v para un 1 lógico mientras que para un microcontrolador utiliza 0v para un 0 lógico y entre 4.75v y 5.25v para un uno lógico (Valores de tensión TTL), la conexión directa sin una interface provocaría daños en el microcontrolador, es por ello la necesidad del circuito MAX 232.

En la figura 2-12 se muestra el diagrama a básico de conexión de la interfaz de comunicación PC – microcontrolador con el circuito integrado MAX 232.

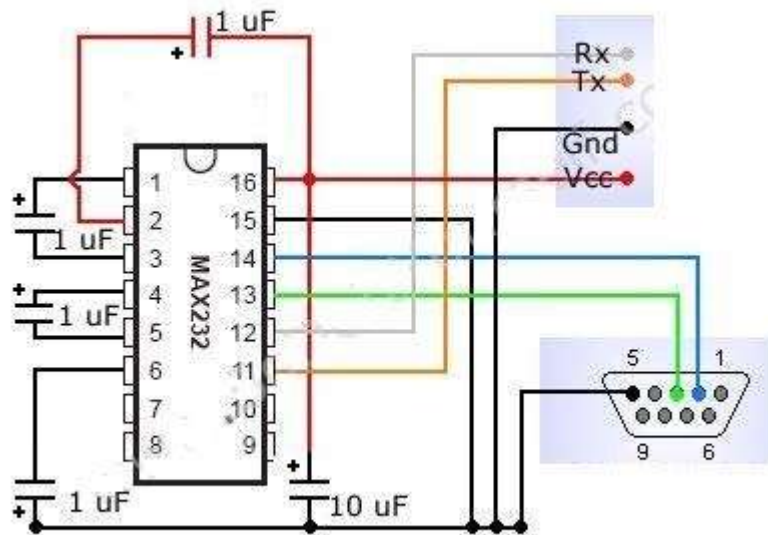


Figura 2-12: Diagrama de conexión del CI MAX232.

2.4 MPLAB

MPLAB es un editor software gratuito, destinado a productos de la marca Microchip. Este programa, permite seleccionar los distintos microcontroladores soportados, además de permitir la grabación de estos circuitos integrados directamente al programador.

Es un programa que corre bajo Windows, Mac OS y Linux. Presenta las clásicas barras de programa, de menú, de herramientas de estado, etc. El ambiente MPLAB® posee editor de texto, compilador y simulación (no en tiempo real). Para comenzar un programa desde cero para luego grabarlo al microcontrolador.

CAPITULO 3 FIRMWARE DEL MICROCONTROLADOR.

Este capítulo está enfocado exclusivamente al desarrollo del programa principal de la tarjeta o como ya se ha mencionado antes, el firmware del dispositivo.

El firmware posee las instrucciones de más bajo nivel del sistema, y están encargadas de interpretar los datos que reciban y actuar de acuerdo a ellos. Esta serie de instrucciones programadas en el microcontrolador son capaces de manipular la operación del mismo y con ello controlar el hardware que se conecta en sus puertos o terminales.

El código para este proyecto está pensado o para controlar cuatro funciones del microcontrolador:

- Comunicación serie.
- Entradas y salidas digitales.
- Entradas analógicas.
- Salidas tipo PWM.

Estas funciones son básicas para la operación de la tarjeta que se desarrolló.

3.1 Lógica del firmware.

El firmware está encargado de controlar al microcontrolador para que este reciba comandos provenientes de una computadora a través del puerto serie, el microcontrolador interpreta el comando que ha sido enviado y con ello elegirá que acción debe tomar. Para el microcontrolador es indistinta la fuente de donde provenga el dato, simplemente lo toma e interpreta, es por ello que es posible desarrollar programas para controlar estas funciones desde cualquier lenguaje de programación. La figura 3-1 muestra el diagrama de bloques general del proyecto.

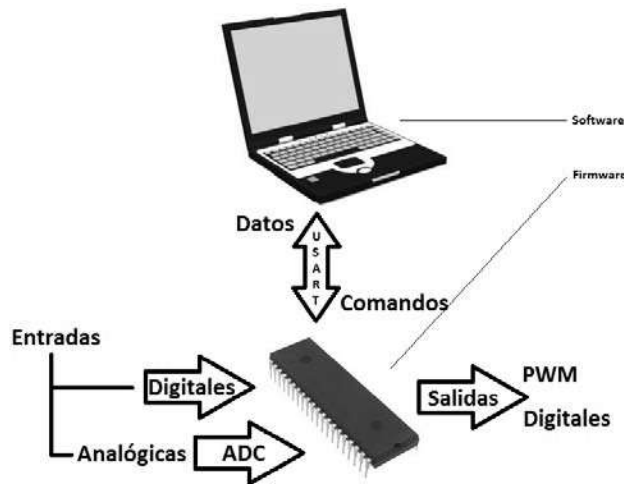


Figura 3-1 Diagrama de bloques general.

3.1.1 Secuencia del programa

La parte más importante y esencial de este firmware es la lectura de caracteres en puerto serie, ya que este es leído constantemente y se almacenan los caracteres leídos en un vector, para posteriormente compararse con una serie de condiciones programadas, si una de estas lecturas coincide con alguna de las condiciones, el programa entra a una sección específica del código que configura parámetros de operación del microcontrolador, logrando con ello ejecutar acciones de lectura o escritura en los puertos de el dispositivo.

3.1.1.1 Comandos de control

A continuación se describen los comandos para cada una de las acciones programadas en el firmware de este dispositivo.

Lectura del convertidor analógico digital: Como se ha venido mencionando en secciones anteriores de este documento, el ADC es el encargado de convertir en valores digitales las señales analógicas que en él se encuentren conectadas.

En esta aplicación haremos uso de un convertidor con seis entradas, el cual puede únicamente hacer una conversión a la vez y por ello resulta necesario seleccionar cuál de estos seis canales deseamos que convierta el ADC, para hacerlo se tienen los siguientes comandos:

AD0, AD1, AD2, AD3, AD4, AD5

Cada uno de estos comandos se envían desde una computadora a través del puerto serie y se encargan respectivamente de habilitar una entrada del convertidor analógico digital. El valor leído de la entrada que haya sido habilitada por el comando, se digitaliza y almacenada en una variable del microcontrolador, para posteriormente enviar esta variable de regreso por el puerto serie y finalmente llegar a la computadora que envió el comando.

Cada uno de los comandos habilita un puerto de entrada del microcontrolador y por lo tanto esto está asociado a un pin del mismo. La tabla 3-1 muestra la relación de los comandos, pines y puertos del microcontrolador para el ADC.

Tabla 3-1: Relación de comandos ADC, puertos y pines.

Comando	Puerto	Pin
AD0	AN0/A0	2
AD1	AN1/A1	3
AD2	AN2/A2	4
AD3	AN3/A3	5
AD4	AN4/A5	7
AD5	AN5/E0	8

Señal de salida PWM: El objetivo del PWM en este dispositivo, es emular las señales analógicas de salida, ya que los microcontroladores son elementos digitales y no generan señales analógicas de manera directa, para generarlas es necesario implementar algunos métodos de filtrado de señales. Sin embargo aunque no son señales analógicas es posible con ellas controlar algunos dispositivos analógicos haciendo uso de elementos como MOSFET, BJT, SCR, TRIAC y demás elementos de conmutación, es por todo esto que resulta necesario que para este proyecto se cuente con este tipo de señal.

En la tarjeta que se plantea en este documento existen dos comandos para la generación de señales PWM, cada comando enviado hacia el microcontrolador habilita un pin de salida y establece el ciclo de trabajo de entre 0% y 99%.

Los comandos para el PWM constan de tres caracteres y cada uno proporciona información al firmware de cómo debe ser la señal generada. La tabla 3-2 muestra la relación de comandos, pines y puertos del microcontrolador para el PWM.

- pXY
- PXY

Tabla 3-2: Relación de comandos PWM, puertos y pines.

Comando	Puerto	Pin
pXY	PWM0/C1	16
PXY	PWM1/C2	17

pXY: el primer carácter de este comando "p" (minúscula) indica al firmware que debe activar el PWM0. El carácter "X" toma valores entre 0 y 9, los cuales representan las decenas del porcentaje del ciclo de trabajo. De igual forma el carácter "Y" toma valores entre 0 y 9, y estos representan el valor de las unidades del ciclo de trabajo; es decir si "X" toma un valor de 5 y "Y" toma un valor de 4, por lo tanto el ciclo de trabajo que se obtiene a la salida es XY=54%.

PXY: para este comando la interpretación de los datos es igual a la anterior en los caracteres XY, la diferencia en este es que el primer carácter "P" (mayúscula) indica al firmware que debe activar el PWM1 con ciclo de trabajo XY.

Una vez que son enviados los comandos para el PWM, el microcontrolador genera la señal con el ciclo de trabajo indicado y esta señal se mantendrá en ese pin con los parámetros indicados, hasta que reciba otro comando que modifique el ciclo de trabajo. Una vez activados los PWM, la manera de desactivarlos es poniendo su ciclo de trabajo a 0% esto se logra colocando 00 en los caracteres XY del comando correspondiente al PWM que se desea apagar.

E/S Digitales: En la mayoría de los proyectos de control a pesar de los complejos estándares que se toman para el control, es necesario tener acciones **ON/OFF** ya sea para algunas entradas o para activar salidas que no requieren más control que encendido y apagado, esta sección del firmware está pensada en este tipo de dispositivos que solo requieren de una

salida en alto para operar y una salida en bajo para dejar de hacerlo, o en los switch mecánicos instalados en algunos equipos, por ejemplo el fin de carrera en un riel o una puerta abierta o cerrada. La tarjeta de este proyecto cuenta con cuatro salidas y ocho entradas digitales.

Salidas: El comando encargado de controlar las salidas digitales en este proyecto es el comando **DOX** donde **X** toma valores hexadecimales de entre 0 y F, en la tabla 3-3 muestra la relación del comando y los pines de salida.

Tabla 3-3: Relación de comandos de salidas digitales, puertos y pines.

Comando	B3 / Pin 36	B2 / Pin 35	B1 / Pin 34	B0 / Pin 33
DO0	0	0 0 0		
DO1	0	0 0 1		
DO2	0	0 1 0		
DO3	0	0 1 1		
DO4	0	1 0 0		
DO5	0	1 0 1		
DO6	0	1 1 0		
DO7	0	1 1 1		
DO8	1	0 0 0		
DO9	1	0 0 1		
DOA	1	0 1 0		
DOB	1	0 1 1		
DOC	1	1 0 0		
DOD	1	1 0 1		
DOE	1	1 1 0		
DOF	1	1 1 1		

Entradas: Para las entradas digitales únicamente hay un comando que es llamado **DIT**, este comando es enviado al microcontrolador, el cual enviara de regreso el estado de los pines del todo el puerto D en una serie de ocho caracteres como se muestra en la tabla 3-4.

Tabla 3-4: Relación de puertos y pines de las entradas digitales.

Puerto	Pin	Estado
D7	30	0 / 1
D6	29	0 / 1
D5	28	0 / 1
D4	27	0 / 1
D3	22	0 / 1
D2	21	0 / 1

Puerto	Pin	Estado
D1	20	0 / 1
D0	19	0 / 1

3.2 Registros asociados al firmware.

En la programación del firmware es necesario hacer uso de algunos registros especiales del microcontrolador, que configuran parámetros de operación. En esta sección se mencionaran y describirán los registros que se usan en el código del firmware de esta aplicación.

3.2.1 Registros del ADC

Para lograr un correcto funcionamiento del convertidor analógico digital, este debe ser configurado en el microcontrolador, para ello existen algunos registros los cuales proporcionan la información de operación.

Los registros asociados al ADC son:

- ADCON0
- ADCON1
- ADCON2

3.2.1.1 Registro ADCON0

Este registro es el encargado de poner en marcha el convertidor y también de seleccionar que canal será el que se va a convertir. En tabla 3-5 se numeran los bits del registro y el nombre de cada bit.

Tabla 3-5: Bits del registro ADCON0.

Numero del Bit	Nombre del Bit
Bit 7	--
Bit 6	--
Bit 5	CHS3

Numero del Bit	Nombre del Bit
Bit 4	CHS2
Bit 3	CHS1
Bit 2	CHS0
Bit 1	GO / DONE
Bit 0	ADON

ADON: Bit activación del ADC cuando en él se coloca un 1, y lo desactiva cuando en él se coloca un 0.

GO/DONE: Este bit indica el estado del proceso de la conversión en este bit se coloca de manera automática un 1 cuando se está realizando una conversión y un 0 cuando el proceso está detenido.

CHS3...CHS0: Bits de selección del canal de conversión. Una vez configurado como línea de entrada analógica el canal se selecciona por medio de estos bits, la tabla 3-6 muestra la relación de bits con el canal analógico.

Tabla 3-6: Bits de selección del canal.

CHS3	CHS2	CHS1	CHS0	Canal seleccionado
0	0	0	0	AN0 (A0)
0	0	0	1	AN1 (A1)
0	0	1	0	AN2 (A2)
0	0	1	1	AN3 (A3)
0	1	0	0	AN4 (A5)
0	1	0	1	AN5 (E0)
0	1	1	0	AN6 (E1)
0	1	1	1	AN7 (E2)
1	0	0	0	AN8 (B2)
1	0	0	1	AN9 (B3)
1	0	1	0	AN10 (B1)
1	0	1	1	AN11 (B4)
1	1	0	0	AN12 (B0)
1	1	0	1	No implementado
1	1	1	0	No implementado
1	1	1	1	No implementado