



**UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO  
CARRERA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y  
AUTOMATISMO

TEMA:

**Desarrollo de una tarjeta de entrenamiento para prácticas de  
microcontroladores utilizando el dispositivo PIC 18F8720**

AUTOR:

Larenas Marín, David Daniel

Trabajo de Titulación previo a la obtención del título de  
**INGENIERO ELECTRÓNICO EN CONTROL Y AUTOMATISMO**

TUTOR:

Córdova Rivadeneira, Luis Silvio

Guayaquil, Ecuador

12 de Marzo del 2018



**UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO  
CARRERA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y  
AUTOMATISMO

**CERTIFICACIÓN**

Certificamos que el presente trabajo fue realizado en su totalidad por el Sr.  
**Larenas Marín, David Daniel** como requerimiento para la obtención del  
título de **INGENIERO ELECTRÓNICO EN CONTROL Y AUTOMATISMO**.

TUTOR

---

Córdova Rivadeneira, Luis Silvio

DIRECTOR DE CARRERA

---

Heras Sánchez, Miguel Armando

Guayaquil, a los 12 días del mes de marzo del año 2018



**UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO  
CARRERA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y  
AUTOMATISMO

**DECLARACIÓN DE RESPONSABILIDAD**

Yo, **Larenas Marín, David Daniel**

**DECLARÓ QUE:**

El trabajo de titulación **“Desarrollo de una tarjeta de entrenamiento para prácticas de microcontroladores utilizando el dispositivo PIC 18F8720”** previo a la obtención del Título de **Ingeniero Electrónico en Control y Automatismo**, ha sido desarrollado respetando derechos intelectuales de terceros conforme las citas que constan en el documento, cuyas fuentes se incorporan en las referencias o bibliografías. Consecuentemente este trabajo es de mi total autoría.

En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance del Trabajo de Titulación referido.

Guayaquil, a los 12 días del mes de Marzo del año 2018

EL AUTOR

---

LARENAS MARÍN, DAVID DANIEL



**UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO  
CARRERA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y  
AUTOMATISMO

**AUTORIZACIÓN**

Yo, **Larenas Marín, David Daniel**

Autorizó a la Universidad Católica de Santiago de Guayaquil, la publicación, en la biblioteca de la institución del Trabajo de Titulación: **“Desarrollo de una tarjeta de entrenamiento para prácticas de microcontroladores utilizando el dispositivo PIC 18F8720”**, cuyo contenido, ideas y criterios son de mi exclusiva responsabilidad y total autoría.

Guayaquil, a los 12 días del mes de Marzo del año 2018

EL AUTOR

---

LARENAS MARÍN, DAVID DANIEL

# REPORTE DE URKUND

**URKUND**

<b>Documento</b>	<a href="#">Larenas_David.docx</a> (D35538324)
<b>Presentado</b>	2018-02-12 01:32 (-05:00)
<b>Presentado por</b>	fernandopm23@hotmail.com
<b>Recibido</b>	edwin.palacios.ucsg@analysis.orkund.com
<b>Mensaje</b>	Revisión TT David Larena <a href="#">Mostrar el mensaje completo</a> 1% de estas 17 páginas, se componen de texto presente en 1 fuentes.

Lista de fuentes	Bloques		
+	Categoría	Enlace/nombre de archivo	▣
+		<a href="#">Formato TT (2).docx</a>	<input type="checkbox"/>
+		<a href="#">TT-DANIEL-SOLIS-NN-1-2016-FINAL-3...</a>	<input type="checkbox"/>
+		<a href="#">Tesis Gary Torres.docx</a>	<input checked="" type="checkbox"/>
+		<a href="https://doi.org/10.5121/ijesa.2015.5302">https://doi.org/10.5121/ijesa.2015.5302</a>	<input type="checkbox"/>
+		<a href="https://doi.org/10.4236/ica.2012.33025">https://doi.org/10.4236/ica.2012.33025</a>	<input type="checkbox"/>

Reiniciar Exportar Compartir

UNIVERSIDAD CATOLICA DE SANTIAGO DE GUAYAQUIL  
FACULTAD DE EDUCACIÓN TÉCNICA PARA EL  
DESARROLLO CARRERA DE INGENIERÍA ELECTRÓNICA  
EN CONTROL Y AUTOMATISMO

TEMA: Desarrollo de una tarjeta de entrenamiento  
para prácticas de microcontroladores utilizando el  
dispositivo PIC 18F8720

AUTOR: Larenas Marín,  
David Daniel  
Trabajo de Titulación  
previo a la obtención del título de INGENIERO  
ELECTRÓNICO EN CONTROL Y AUTOMATISMO

TUTOR: Córdova Rivadeneira, Luis Silvio  
Guayaquil, Ecuador

1 Advertencia

## DEDICATORIA

Dedico este trabajo a Dios y a mis padres. A Dios porque ha estado a mi lado toda mi vida que me ha brindado apoyo incondicional y sabiduría para elaborar mi trabajo de titulación. A mis padres que me han brindado confianza, y consejos para resolver todos mis asuntos tanto en lo personal como en lo académico como en lo profesional. Dedico este proyecto a mi abuelo Honorato Marín quién en vida fue mi guerrero y que me está enseñando a ser fuerte y humilde. "Para ti Papi Tato". Y a mí abuelita Leonor Onofre mi fiel compañía quien me da un gran abrazo cada vez que la veo y me da la bendición cada vez que salgo. "Para ti Mami Luz"

EL AUTOR

---

LARENAS MARÍN, DAVID DANIEL

## **AGRADECIMIENTO**

Quiero agradecer a mi madre mi pilar fundamental en apoyarme con mis estudios mis problemas, por comprenderme y ser mi mejor amiga, quien me ha aconsejado en el trayecto de mis estudios en la UCSG. A mi padre que me ha enseñado la virtud de la firmeza antes los acontecimientos de la vida. A mi familia mis primos y tíos que han demostrado que somos una gran familia y nos apoyamos incondicionalmente y que juntos somos fuertes. A mis amigos que nunca me abandonaron en momentos difíciles de la vida, y me siguen apoyando en todo momento Gino, Karlita.

A los Docentes de la Universidad Católica de Santiago de Guayaquil agradecidos por sus consejos profesionales y sus enseñanzas teóricas y prácticas a lo largo de la carrera universitaria. A mi tutor M. Sc. Luis Córdova Rivadeneira y en especial al M. Sc. Fernando Palacios Meléndez por sus consejos y sus ayudas a lo largo de la carrera, más que un docente un Gran Amigo.

EL AUTOR

---

LARENAS MARÍN, DAVID DANIEL



**UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO  
CARRERA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y  
AUTOMATISMO

**TRIBUNAL DE SUSTENTACIÓN**

f. \_\_\_\_\_

**ROMERO PAZ, MANUEL DE JESÚS**  
DECANO

f. \_\_\_\_\_

**MENDOZA MERCHÁN, EDUARDO VICENTE**  
COORDINADOR DE TITULACIÓN

f. \_\_\_\_\_

**PALACIOS MELÉNDEZ, EDWIN FERNANDO**  
OPONENTE

## Índice General

Índice de Figuras .....	XII
Índice de Tablas .....	XV
Resumen .....	XVI
<b>CAPÍTULO 1: INTRODUCCIÓN .....</b>	<b>17</b>
1.1. Introducción.....	17
1.2. Antecedentes. ....	18
1.3. Definición del Problema.....	21
1.4. Justificación del Problema.....	21
1.5. Objetivos del Problema de Investigación.....	22
1.5.1. Objetivo General.....	22
1.5.2. Objetivos Específicos.....	22
1.6. Hipótesis. ....	22
1.7. Metodología de Investigación.....	23
<b>CAPÍTULO 2: FUNDAMENTACIÓN TEÓRICA.....</b>	<b>25</b>
2.1. Introducción.....	25
2.2. Unidades de microcontroladores (MCU).....	26
2.3. Microcontroladores PIC. ....	27
2.4. Arquitectura de los microcontroladores PIC.....	30
2.5. Descripción general de los sistemas de entrenamiento de microcontroladores.....	35
2.5.1. Sistema de entrenamiento de microcontroladores PICAXE. .	35
2.5.2. Sistema de entrenamiento de microcontroladores EasyPic 6. .....	41
2.5.3. Sistema de entrenamiento de microcontroladores PIC EDUTRONICA JJTSA.....	44

2.6.	Análisis de prestaciones de los sistemas de entrenamiento de microcontroladores PIC.....	47
2.6.1.	Análisis de prestaciones del entrenador de microcontroladores PICAXE .....	47
2.6.2.	Análisis de prestaciones del entrenador de microcontroladores EasyPic6.....	48
2.6.3.	Análisis de prestaciones del Entrenador de Microcontroladores PIC EDUTRONICA JJTSA.....	49
CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS EXPERIMENTALES.....		50
3.1.	Introducción del sistema de entrenamiento propuesto.....	50
3.2.	Módulo de entrenamiento para el microcontrolador PIC18F8720. ....	51
3.2.1.	Microcontrolador PIC18F8720 con cristal de cuarzo de 40 MHz. ....	53
3.2.2.	Bloque de 8 entradas digitales.....	54
3.2.3.	Bloque de 8 salidas digitales.....	55
3.2.4.	Bloque de display LCD de 4x20.....	56
3.2.5.	Bloque de Display GLCD. ....	57
3.2.6.	Bloque de 4 displays de 7 segmentos.....	58
3.2.7.	Bloque de 4 matrices de LEDs de 8x8. ....	59
3.2.8.	Bloque del teclado matricial de 4X3. ....	60
3.2.9.	Bloque de 4 potenciómetros para entradas analógicas al PIC18F8720. ....	61
3.2.10.	Salida de relé.....	62
3.2.11.	Salida de señal audible.....	63
3.2.12.	Módulos de comunicación serial asíncrona.....	63
3.3.	Aplicación práctica 1: conversión a código gray usando led's. ....	64
3.4.	Aplicación práctica 2: contador ascendente de 0000 a 9999. ....	68
3.5.	Aplicación práctica 3: apertura de cerraduras eléctricas.....	70

3.6. Aplicación práctica 4: manejo de GLCD y teclado matricial.....	74
CAPÍTULO 4: CONCLUSIONES Y RECOMENDACIONES.....	78
4.1. Conclusiones.....	78
4.2. Recomendaciones.....	79
Bibliografía.....	80

## Índice de Figuras

### Capítulo 2

Figura 2. 1: Diagrama de bloques de microcontroladores PIC. ....	26
Figura 2. 2: Diagrama de pines del uC PIC18F8520/8620/8720 de 8 bits. ....	28
Figura 2. 3: Arquitectura del uC PIC18F8X20 de 8 bits. ....	29
Figura 2. 4: Enfoque de la arquitectura Von Neumann, (a) convencional Von Neumann, y (b) arquitectura Harvard. ....	30
Figura 2. 5: Diagrama de bloques para un sistema de control de nivel de agua basado en microcontroladores PIC. ....	32
Figura 2. 6: Diagrama de bloques para un sistema de control de nivel de agua basado en microcontroladores PIC utilizando teclado matricial y visualizador LCD. ....	33
Figura 2. 7: Diagrama de bloques para un sistema de control de nivel de agua basado en microcontroladores PIC más sofisticado. ....	34
Figura 2. 8: Diagrama de bloques para un sistema de control de nivel de agua basado en microcontroladores PIC más sofisticado. ....	34
Figura 2. 9: Diagrama de bloques del entrenador PICAXE. ....	36
Figura 2. 10: Sistema de entrenamiento de microcontroladores PICAXE. ....	37
Figura 2. 11: Circuito esquemático de la conexión de los 2 displays. ....	38
Figura 2. 12: Circuito esquemático de la conexión de 4 LED's. ....	38
Figura 2. 13: Circuito esquemático de la salida con relé. ....	39
Figura 2. 14: Circuito esquemático de pulsadores. ....	39
Figura 2. 15: Circuito esquemático del Dipswitch. ....	39
Figura 2. 16: Circuito esquemático del Teclado Matricial. ....	40
Figura 2. 17: Circuito esquemático del Buzzer. ....	40
Figura 2. 18: Circuito esquemático de los dispositivos de salida. ....	41
Figura 2. 19: Sistema modular de entrenamiento EasyPic6. ....	42
Figura 2. 20: Diagrama esquemático del sistema modular de entrenamiento EasyPic6 ....	44

Figura 2. 21: Tarjeta de la etapa de programación de PIC.....	45
Figura 2. 22: Tarjeta de la etapa de evaluación de programas .....	46

### Capítulo 3

Figura 3. 1: Diseño del sistema de entrenamiento en Proteus para PIC18F8720. ....	52
Figura 3. 2: Diseño esquemático del sistema de entrenamiento de uC PIC18F8720 en Eagle. ....	53
Figura 3. 3: Etapa 1 del diseño esquemático del sistema de entrenamiento de uC PIC18F8720. ....	54
Figura 3. 3: Etapa 2 del diseño esquemático del sistema de entrenamiento de uC PIC18F8720. ....	55
Figura 3. 3: Etapa 3 del diseño esquemático del sistema de entrenamiento de uC PIC18F8720. ....	55
Figura 3. 6: Etapa 4 del diseño esquemático del sistema de entrenamiento de uC PIC18F8720. ....	56
Figura 3. 7: Etapa 5 del diseño esquemático del sistema de entrenamiento de uC PIC18F8720. ....	58
Figura 3. 8: Etapa 6 del diseño esquemático del sistema de entrenamiento de uC PIC18F8720. ....	59
Figura 3. 9: Etapa 7 del diseño esquemático del sistema de entrenamiento de uC PIC18F8720. ....	60
Figura 3. 10: Etapa 8 del diseño esquemático del sistema de entrenamiento de uC PIC18F8720. ....	61
Figura 3. 11: Etapa 9 del diseño esquemático del sistema de entrenamiento de uC PIC18F8720. ....	62
Figura 3. 12: Etapa 10 del diseño esquemático del sistema de entrenamiento de uC PIC18F8720. ....	62
Figura 3. 13: Etapa 11 del diseño esquemático del sistema de entrenamiento de uC PIC18F8720. ....	63

Figura 3. 14: Etapa 9 del diseño esquemático del sistema de entrenamiento de uC PIC18F8720.....	64
Figura 3. 15: Aplicación práctica #1 del sistema de entrenamiento de uC PIC18F8720. ....	65
Figura 3. 16: Aplicación práctica #2 del sistema de entrenamiento de uC PIC18F8720. ....	68
Figura 3. 17: Aplicación práctica #3 del sistema de entrenamiento de uC PIC18F8720. ....	71
Figura 3. 18: Aplicación práctica #4 del sistema de entrenamiento de uC PIC18F8720. ....	75

## Índice de Tablas

### Capítulo 2

Tabla 2. 1: Diferencias entre las arquitecturas de microcontroladores .....31

### Capítulo 3

Tabla 3. 1: Representación de numeraciones decimal a binaria y código Gray.  
.....66

## Resumen

El presente trabajo de titulación consistió en el desarrollo de una tarjeta de entrenamiento para prácticas de microcontroladores utilizando el dispositivo PIC 18F8720. En el capítulo 1 se describe los antecedentes, definición y justificación del problema de investigación. También se especifica el objetivo general, objetivos específicos, hipótesis y metodología de investigación utilizada en el presente proyecto de titulación. En el capítulo 2, se realiza la descripción de los fundamentos teóricos de los microcontroladores, así como sus respectivas arquitecturas. Se incluye una descripción y análisis de las prestaciones de cada uno de los sistemas de entrenamiento de microcontroladores disponibles para la enseñanza en carreras de ingenierías. En el capítulo 3, se realiza el diseño del sistema de entrenamiento de microcontroladores utilizando el dispositivo PIC 18F8720, así como también, se incluyen en el trabajo de titulación tres aplicaciones prácticas desarrolladas a nivel de simulación ISIS Proteus, programación de alto nivel C e implementación para la validación de la correcta funcionalidad de la plataforma de entrenamiento.

**Palabras claves:** MICROCONTROLADORES, PROTEUS, CCS C, ARQUITECTURA, PIC, SIMULACIÓN

## CAPÍTULO 1: INTRODUCCIÓN

### 1.1. Introducción.

Los microcontroladores se han convertido en ayudantes omnipresentes en nuestra vida cotidiana. Son computadoras compactas y de propósito único que ejecutan aplicaciones de software integradas que se utilizan ampliamente en dispositivos y sistemas eléctricos modernos para controlar operaciones, como ajustes de temperatura de hornos, control remoto de televisores o funciones extendidas de teléfonos celulares. Inclusive, la tecnología ha permitido que la mecánica automotriz debe trabajar con microcontroladores para controlar las mezclas de combustible y el tiempo de ignición.

A nivel mundial, los estudiantes tienden a estar menos interesados en los cursos que involucran sistemas integrados, debido a la experiencia menos técnica y práctica que han adquirido a lo largo de los años. Los expertos limitados en sistemas incorporados también tienen recursos inadecuados como las herramientas de desarrollo de hardware y software que explican la pérdida de interés. (Lukman Adewale Ajao, Agajo, Kolo, Adegboye, & Yusuf, 2016)

A pesar de estas dificultades, hoy los estudiantes pueden avanzar en el área de sistemas integrados mediante la realización de ciertos experimentos internos simples con prácticas frecuentes en el diseño y simulación de circuitos electrónicos basados en microcontroladores utilizando herramientas de diseño asistido por computadora (CAD). tales como el modelado del

sistema virtual de Proteus (PVSM), Multisim, TinyCAD, el sistema de diseño de Hamburgo (HADES), el simulador de circuito universal (QUCS), las herramientas de FidoCadJ, entre otras.

## **1.2. Antecedentes.**

La búsqueda de información realiza en el presente trabajo de titulación fue relevante para saber de los trabajos o proyectos publicados en repositorios y revistas científicas, que estén relacionados en el estudio y desarrollo de sistemas de entrenamiento de microcontroladores. A continuación, se describen los trabajos relacionados al trabajo de titulación:

1. Somantri & Fushshilat, (2017) en su proyecto desarrollaron un kit de entrenamiento de microcontroladores integrado, avanzado y de bajo costo. En el artículo publicado por ellos, describen el diseño de un kit de entrenamiento de microcontroladores AVR con un bajo costo y la característica adicional de tener integrado un descargador (quemador). Los principales componentes de este dispositivo incluyen: microcontrolador, terminal, teclado de E/S, botón pulsador, LED, pantalla de siete segmentos, LCD, motor paso a paso y sensores. La configuración del dispositivo resulta en bajo costo y facilidad de uso; este dispositivo es adecuado para laboratorios con fondos limitados. El dispositivo también se puede usar como un kit de capacitación para la enseñanza y el aprendizaje de microcontroladores.
2. Vassiliev, Ivanova, Tapia, & Luong, (2016) publicaron el desarrollo de equipos de sistemas integrados basados en microcontroladores para investigación y apoyo educativo. Los autores investigan el caso del

desarrollo de equipos para educación, investigación y diseño de sistemas de microcontroladores integrados, efectivos desde el punto de vista de la minimización de costos para el desarrollador cuando se diseñan prototipos a escala real, así como efectivos en términos de la calidad de la capacitación en esta área.

3. (Lukman Adewale Ajao et al., 2016) publicaron el trabajo “Aprendizaje de diseño, simulación e implementación de sistemas integrados: un enfoque técnico”. Este documento muestra el método de aprendizaje del diseño del circuito del sistema con el uso de herramientas de simulación de diseño asistido por computadora (CAD). Para demostrar este procedimiento sobre cómo se llevaron a cabo en el laboratorio los experimentos de circuitos basados en microcontroladores virtuales, y como principal componente lógico se utilizó el chip PIC16F887. Se demostraron varias prácticas en experimentos manuales con la ayuda de PVSM, y el resultado de cada experimento realizado se mostró con una instantánea. Aunque, la intención de este documento es presentar un enfoque técnico para el aprendizaje y la participación de los estudiantes en las prácticas, experto en la progresión del aprendizaje de los sistemas integrados. Este experimento práctico será una ventaja añadida además de la enseñanza de los alumnos en la sala de clases, y permitirá a los aficionados, científicos profesionales e ingenieros diseñar y analizar el diseño del sistema antes de construir un prototipo para aprender el tablero y antes de realizar el embalaje final.
4. Lukman A. Ajao, Olaniyi, Kolo, & Ajao, (2015) desarrollaron el laboratorio del sistema de microcontroladores basados en proyectos

utilizando la tarjeta de desarrollo BK300 con Chip PIC 16F877. En este trabajo, resuelven el problema de la comprensión del alumno y el desarrollo de habilidades en el diseño de sistemas integrados utilizando el microcontrolador PIC16F887 mediante la demostración de experimentos prácticos de laboratorio. Además, se llevaron a cabo desarrollos de código de software, simulación de diagrama de circuito. Esto es para ayudar a los estudiantes a conectar sus conocimientos teóricos con la experiencia práctica.

5. Uttarkar & Kanchi, (2014) realizaron el diseño y desarrollo de un laboratorio de sistema integrado de bajo costo utilizando TI MSP430 LaunchPad. En este documento, se presenta una serie de experimentos comenzando desde ¡Hello World!, hasta experimentos basados en proyectos utilizando Texas Instrument (TI) MSP430 LaunchPad, que es un kit económico de TI, EE. UU. Dos paneles dedicados están diseñados y desarrollados para realizar los experimentos descritos en este artículo científico, que son el resultado de la experiencia del autor en la enseñanza de la arquitectura de computadoras y los sistemas integrados en la teoría. Estos experimentos pueden ser presentados a estudiantes de pregrado de Ingeniería: electricidad, electrónica y telecomunicaciones, informática y estudiantes de postgrado en ciencias especializados en electrónica y telecomunicaciones. La característica más destacada de los experimentos presentados en diferentes módulos es que cada experimento se explica por su descripción de hardware, desarrollo de

software que incluye la descarga de programas en la memoria flash del microcontrolador, lo que lleva a una experiencia práctica.

### **1.3. Definición del Problema.**

En el laboratorio de electrónica se realizan las prácticas experimentales de la asignatura de microcontroladores, a nivel de modelado en Proteus, programación en CCS C Compiler e implementación sobre EasyPic 6 o protoboard. La primera, es una plataforma desarrollada por Mikroelektronika, muy útil para el aprendizaje de los estudiantes de las carreras de Telecomunicaciones, y Electrónica, pero limitada por el alto costo a nivel de software y de ciertos elementos electrónicos extras para la implementación de futuras prácticas de laboratorio. La segunda, es muy limitada porque se deben integrar al protoboard una serie de dispositivos electrónicos, lo que perjudicaría en el tiempo de aprendizaje de los estudiantes.

Por lo anteriormente explicado, se crea la necesidad de realizar el desarrollo de una tarjeta de entrenamiento para prácticas de microcontroladores utilizando el dispositivo PIC 18F8720, en el cual tiene incorporado varios dispositivos microelectrónicos que son necesarios para cubrir los contenidos del syllabus de microcontroladores.

### **1.4. Justificación del Problema.**

El diseño e implementación de este sistema de entrenamiento de microcontroladores también puede ser utilizado para el desarrollo de proyectos prácticos de aprendizaje en educación continua. Inclusive a

distancia, ya que brinda oportunidades a los estudiantes en áreas rurales y urbanas para que aprendan conceptos actuales de tecnología y estén preparados para calificar para trabajos de alta tecnología.

## **1.5. Objetivos del Problema de Investigación.**

### **1.5.1. Objetivo General.**

Desarrollar una tarjeta de entrenamiento para prácticas de microcontroladores utilizando el dispositivo PIC 18F8720

### **1.5.2. Objetivos Específicos.**

- Describir la fundamentación teórica de los microcontroladores y de los sistemas de entrenamientos de microcontroladores.
- Analizar las prestaciones de otros sistemas de entrenamiento de microcontroladores disponibles para el proceso de aprendizaje de estudiantes de ingeniería electrónica en control y automatismo.
- Realizar el diseño esquemático del sistema de entrenamiento para prácticas de microcontroladores utilizando el dispositivo PIC 18F8720.
- Elaborar escenarios de simulación e implementación de prácticas de sistemas microcontroladores de acuerdo a los contenidos del syllabus de la asignatura de microcontroladores.

## **1.6. Hipótesis.**

Debido a que los microcontroladores son tan importantes para nuestro mundo de alta tecnología, la demanda es alta para los futuros profesionales en Ingeniería Electrónica en Control y Automatismo, que estén capacitados

para diseñar, mantener y ponerlas en práctica. Aunque, muchas personas en lo laboral no tienen título de tercer nivel, pero no pueden tomarse tiempo fuera del trabajo o de las familias para inscribirse en programas de tecnología o de ingeniería en los campus universitarios.

### **1.7. Metodología de Investigación.**

El método de investigación es una estrategia de investigación, que se mueve desde los supuestos subyacentes hasta el diseño de la investigación y la recopilación de datos. Aunque existen otras distinciones en los modos de investigación, la clasificación más común de los métodos de investigación es cualitativa y cuantitativa. En un nivel, cualitativo y cuantitativo se refiere a las distinciones sobre la naturaleza del conocimiento: cómo uno entiende el mundo y el propósito final de la investigación. En otro nivel del discurso, los términos se refieren a métodos de investigación, es decir, la forma en que se recopilan y analizan los datos, y el tipo de generalizaciones y representaciones derivadas de los datos.

Los métodos de investigación cuantitativa se desarrollaron originalmente en las ciencias naturales para estudiar los fenómenos naturales. Se desarrollaron métodos de investigación cualitativa en las ciencias sociales para permitir a los investigadores estudiar fenómenos sociales y culturales. Tanto los estudios de investigación cuantitativa como cualitativa se llevan a cabo en educación.

Ninguno de estos métodos es intrínsecamente mejor que el otro; cuya idoneidad debe decidirse por el contexto, el propósito y la naturaleza del estudio de investigación en cuestión; de hecho, a veces uno puede ser una alternativa al otro dependiendo del tipo de estudio. Algunos investigadores prefieren utilizar el enfoque de métodos mixtos aprovechando las diferencias entre los métodos cuantitativos y cualitativos, y combinan estos dos métodos para su uso en un solo proyecto de investigación, dependiendo del tipo de estudio y su fundamentación metodológica.

Por lo tanto, el presente trabajo de titulación tiene enfoque cuantitativo cuyo método de investigación es empírico-analítico y el diseño de investigación es descriptivo y explicativo.

## **CAPÍTULO 2: FUNDAMENTACIÓN TEÓRICA**

### **2.1. Introducción.**

El gran éxito de los microcontroladores es que nos permite diseñar sistemas programables de control mínimos de bajo coste, flexibles y muy fiables. En el mercado existen herramientas como software y hardware que facilitan el trabajo de diseñar, compilar y poner a punto las distintas aplicaciones en el ámbito profesional e inclusive en el área de la medicina.

Múltiples son las ofertas en este sentido y como se mencionó con anterioridad como no se puede abarcar todo el universo de los microcontroladores, en este estudio se tratará sólo sobre las propuestas de tres empresas que enfocan con mayor énfasis el estudio de los microcontroladores de la gama media (Valdivieso, 09). Estas herramientas son empleadas en los laboratorios, centros de enseñanza e investigación en electrónica específicamente en la asignatura de microcontroladores como plataformas que simplifican a los estudiantes el diseño de aplicaciones.

Permiten trabajar con diferentes dispositivos PIC, aportan una serie de periféricos relevantes que permiten probar y simular el funcionamiento de aplicaciones, así como también se puede grabar (la programación al PIC) el microcontrolador sin sacar del equipo (entrenador). A continuación, se detallan los diferentes entrenadores de microcontroladores, que en las siguientes secciones se analizan en detalle las características de cada una de las plataformas propuestas:

1. Entrenador de Microcontroladores PICAXE
2. Entrenador de Microcontroladores EasyPic6
3. Tarjeta PVK40

Finalmente, en el capítulo 3 realizar el diseño de un sistema de entrenamiento novedoso y que va acorde al syllabus de la asignatura de microcontroladores para la implementación en el laboratorio de electrónica.

## 2.2. Unidades de microcontroladores (MCU).

Una unidad de microcontrolador (MCU) es una computadora de un solo chip. Micro, sugiere que el dispositivo es pequeño, y el controlador sugiere que se use en aplicaciones de control. Otro término para el microcontrolador es el controlador integrado, ya que la mayoría de los microcontroladores están integrados en los dispositivos que controlan.

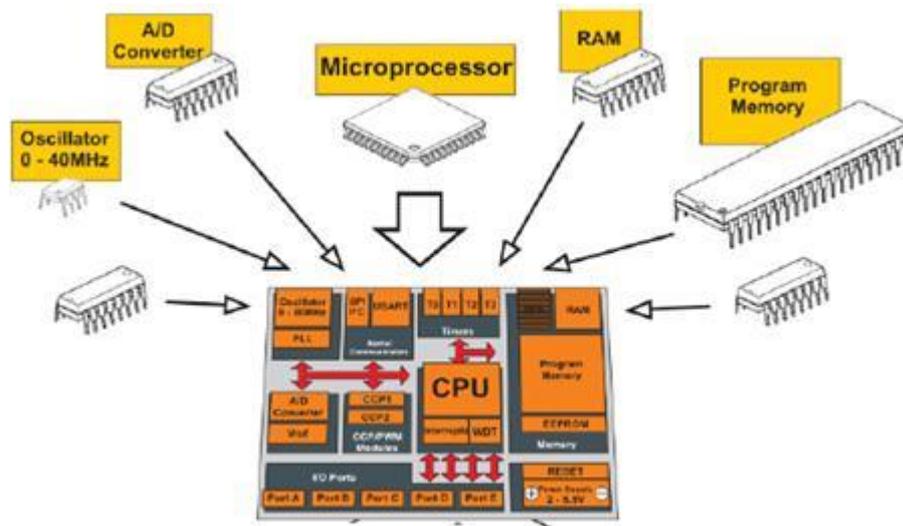


Figura 2. 1: Diagrama de bloques de microcontroladores PIC.

Fuente: (Lukman Adewale Ajao et al., 2016)

Por ejemplo, un microprocesador difiere de un microcontrolador en muchos aspectos. La principal diferencia es que un microprocesador requiere varios otros componentes para su funcionamiento, como memoria de

programa y memoria de datos, interfaces de E/S y circuito de reloj externo. Por otro lado, un microcontrolador tiene todos los chips de soporte incorporados dentro del mismo chip. En esencia, el microprocesador es solo un componente en el chip del microcontrolador.

Mashhadany, (2012) manifiesta que el microcontrolador y el microprocesador difieren en muchos aspectos. Por ejemplo, en funcionalidad, un microprocesador necesita componentes externos, tales como, recibir/enviar datos y memoria. Un microcontrolador no necesita componentes externos porque todos los periféricos necesarios están integrados, ahorrando tiempo y espacio.

Los microprocesadores modernos pueden almacenar cientos de miles de transistores cada uno. Los primeros microprocesadores tenían periféricos externos como memoria, líneas de entrada-salida y temporizadores. Según Barsoum, (2010) con el tiempo llegó un nuevo dispositivo llamado circuito integrado (*Integrated Circuit, IC*), que contiene tanto el procesador como los periféricos. También llamado microcontrolador, este fue el primer chip con una microcomputadora.

### **2.3. Microcontroladores PIC.**

Una de las familias de microcontroladores más populares y fáciles de usar disponibles en el mercado hoy en día es el "microcontrolador PIC" de Microchip Technology Inc. Originalmente conocido como controlador de interfaz periférica (*Peripheral Interface Controller, PIC*), el uC PIC consta de

más de 400 variaciones, cada uno diseñado para ser óptimo en diferentes aplicaciones. Estas variaciones consisten en una cantidad de configuraciones de memoria, arreglos de clavijas de E/S diferentes, cantidad de hardware de soporte requerido, empaquetado y funciones periféricas disponibles. En la figura 2.2 se muestra el diagrama de pines del uC PIC18F8520/8620/8720 correspondiente a la familia de 8 bits de Microchip.

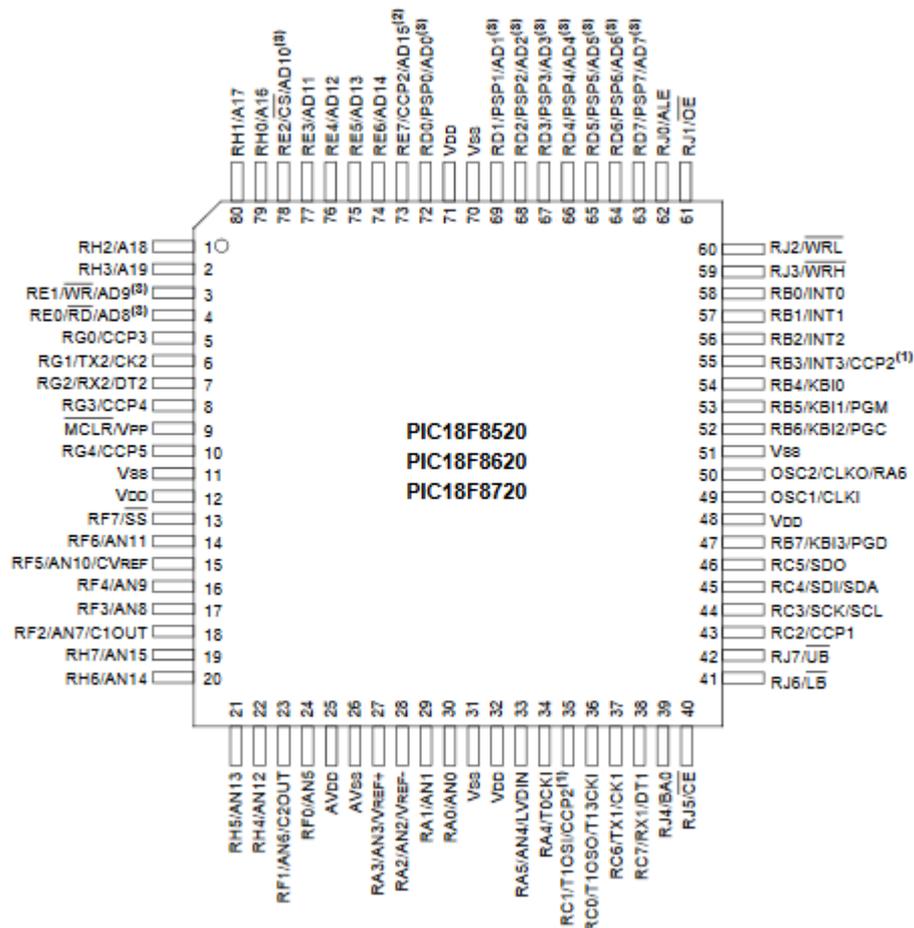


Figura 2. 2: Diagrama de pines del uC PIC18F8520/8620/8720 de 8 bits.

Fuente: (Microchip, 2013)

Según Ibrahim, (2015) la arquitectura del microcontrolador PIC se basa en una configuración modificada de Harvard RISC (Computador con Conjunto de Instrucciones Reducidas) con arquitectura de bus dual, que proporciona un diseño rápido y flexible con una ruta de migración fácil. El conjunto básico de

instrucciones de ensamblador de microcontroladores PIC consta de solo 33 instrucciones y la mayoría de los miembros de la familia (excepto los dispositivos recientemente desarrollados) usan el mismo conjunto de instrucciones. Esta es la razón por la cual un programa desarrollado para un modelo se puede ejecutar en otro modelo con una arquitectura similar sin ningún cambio. En la figura 2.3 se muestra la arquitectura del PIC18F8X20, pero en la sección 2.4 se describen los tipos de arquitecturas de los microcontroladores.

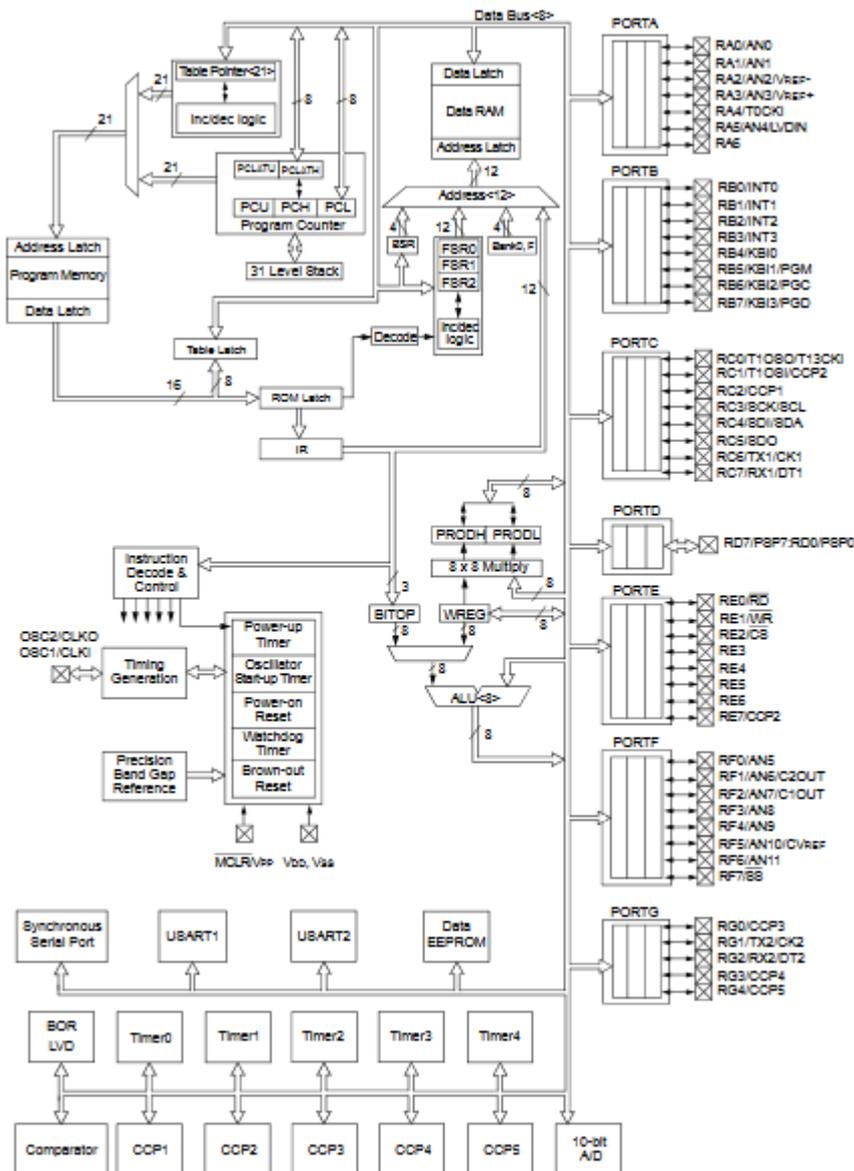


Figura 2. 3: Arquitectura del uC PIC18F8X20 de 8 bits.  
Fuente: (Microchip, 2013)

## 2.4. Arquitectura de los microcontroladores PIC.

Dos tipos de arquitecturas son convencionales en microcontroladores. La arquitectura Von Neumann, utilizada por un gran porcentaje de microcontroladores, coloca todo el espacio de memoria (programa y datos) en el mismo bus. Las instrucciones y los datos deben buscarse en orden secuencial (conocido como el cuello de botella de Von Neumann), lo que limita el ancho de banda de la operación. En la arquitectura de Harvard (utilizada por los microcontroladores PIC), el código y los datos se encuentran en buses separados y esto permite que el código y los datos se obtengan simultáneamente, lo que resulta en un mejor rendimiento.

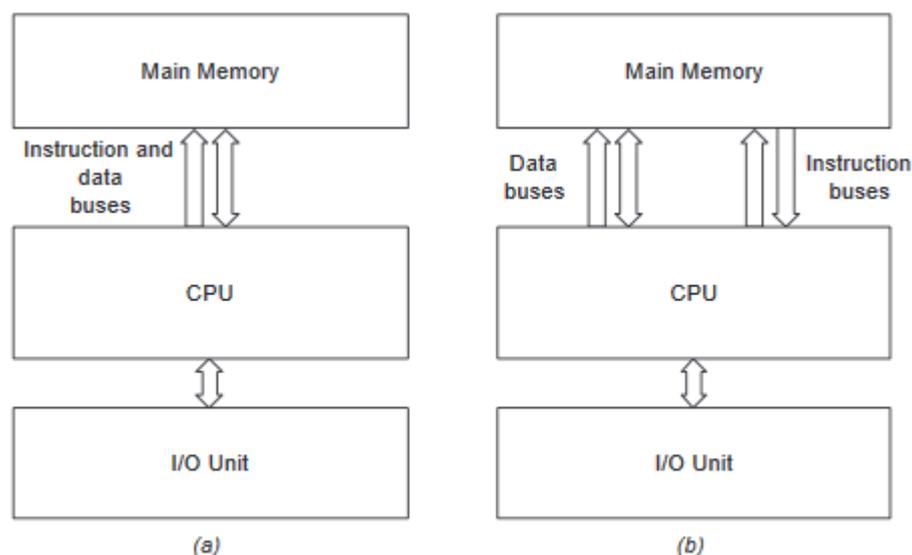


Figura 2. 4: Enfoque de la arquitectura Von Neumann, (a) convencional Von Neumann, y (b) arquitectura Harvard.

Fuente: (Allam, 2006)

Para Allam, (2006) la arquitectura Harvard es una clase de organización de computadoras de Von Neumann. Mientras que en las computadoras convencionales de Von Neumann, se usa el mismo conjunto de buses (dirección y datos) para las instrucciones del programa y los datos, tal como se muestra en la figura 2.4(a). En la arquitectura de Harvard, se usan dos

conjuntos de buses separados para las instrucciones y datos del programa. En dicha arquitectura, las instrucciones del programa y los datos parecen tener acceso simultáneamente, tal como se ilustra la figura 2.4(b).

En los sistemas microcontroladores estas dos arquitecturas tienen sus diferencias. A continuación, en la tabla 2.1 se muestra el análisis comparativo entre la arquitectura de Harvard y la arquitectura de Von Neumann.

Tabla 2. 1: Diferencias entre las arquitecturas de microcontroladores

<b>Arquitectura Harvard</b>	<b>Arquitectura Von Neumann</b>
El nombre proviene de "Harvard Mark I", una computadora antigua basada en retransmisión.	Lleva el nombre del matemático y científico informático John Von Neumann.
Se requiere dos memorias para su instrucción y datos.	Se requiere solamente una memoria para su instrucción y datos.
El diseño de la arquitectura de Harvard es complicado.	El diseño de la arquitectura de Von Neumann es simple.
Memorias separadas para códigos y datos.	Memoria única para ser compartida por código y datos.
La arquitectura de Harvard requiere un bus separado para la instrucción y los datos.	La arquitectura de Von Neumann solo requiere un bus para instrucción y datos.
Más lenta en velocidad, por lo tanto, consume más tiempo.	Mayor velocidad, por lo tanto, consume menos tiempo.
El procesador puede completar una instrucción en un ciclo	El procesador necesita dos ciclos de reloj para completar una instrucción.
Más fácil de canalizar, por lo que se puede lograr un alto rendimiento.	Bajo rendimiento en comparación con la arquitectura de Harvard.
Complejo en diseño	Simple en diseño
Costo comparativamente alto.	Es más barato.

Fuente: (Technical Editor, 2017)

A continuación, se exponen algunas aplicaciones prácticas en los que utilizan los sistemas microcontroladores. Por ejemplo, la figura 2.5 muestra un sistema de control de nivel de agua utilizando microcontroladores PIC. El nivel de agua es leído por el uC mediante un sensor de nivel, para lo cual, el uC se encarga de controlar el nivel de agua al valor requerido. Es decir, que, si el nivel de agua es bajo entonces el uC opera una bomba para extraer agua de la cisterna a fin de mantener el nivel requerido. (Ibrahim, 2014)

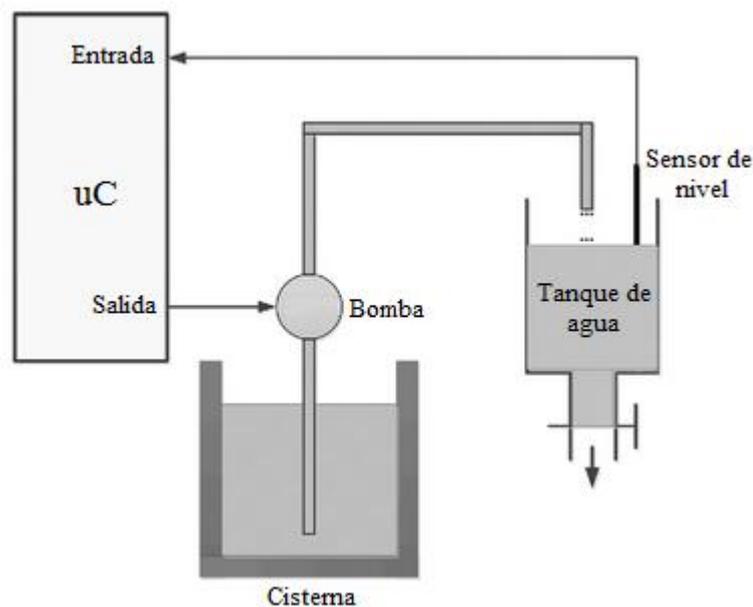


Figura 2. 5: Diagrama de bloques para un sistema de control de nivel de agua basado en microcontroladores PIC.

Fuente: (Ibrahim, 2015)

A continuación, en la figura 2.6 se muestra un sistema más sofisticado con relación al mostrado en la figura 2.5. Se observa, dos dispositivos adicionales, tales como, un teclado (keyboard) que sirve para la configuración de los niveles de agua que se requieren en el sistema; y, una pantalla de cristal líquido (*Liquid Cristal Display, LCD*) que permite visualizar por pantalla el nivel actual en el tanque. (Ibrahim, 2014)

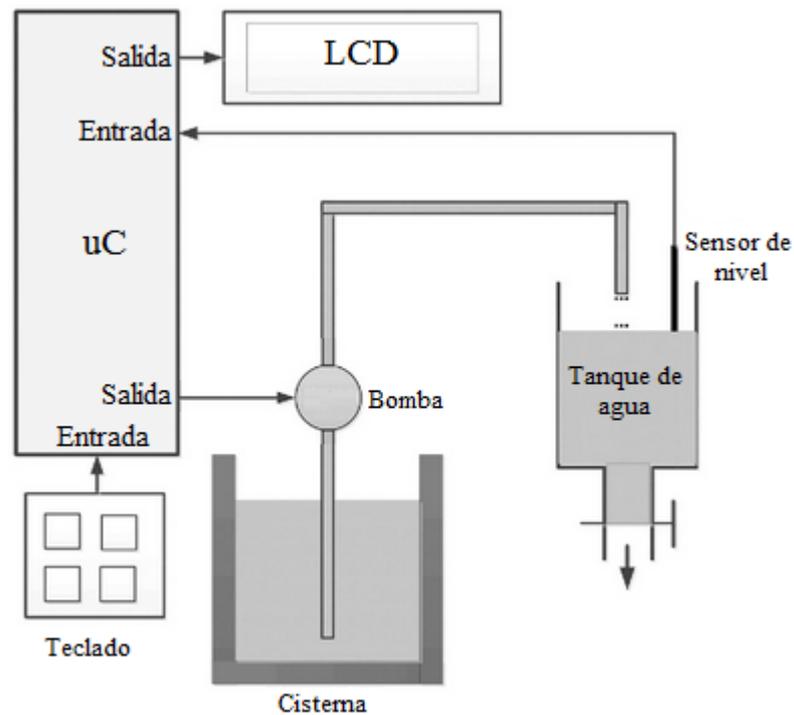


Figura 2. 6: Diagrama de bloques para un sistema de control de nivel de agua basado en microcontroladores PIC utilizando teclado matricial y visualizador LCD.  
Fuente: (Ibrahim, 2015)

Con relación a la figura 2.6 se pueden agregar dispositivos para que el sistema sea más sofisticado. Por ejemplo, se agrega una alarma sonora (véase la figura 2.7) para saber si el nivel de fluido está fuera del valor requerido. Además, el nivel real en cualquier momento se puede enviar a una PC cada segundo para archivarlo y procesarlo. Por ejemplo, un gráfico de los cambios de nivel de agua diarios se puede trazar en la PC. En general debido a que los microcontroladores son programables, es muy fácil hacer que el sistema final sea tan simple o complicado como nos plazca. (Ibrahim, 2014).

La interfaz inalámbrica (por ejemplo, Bluetooth o RF) o la conectividad a Internet se pueden agregar al sistema para que el nivel de agua se pueda controlar o controlar de forma remota. La figura 2.8 muestra el diagrama de bloques con un módulo Bluetooth conectado al microcontrolador.

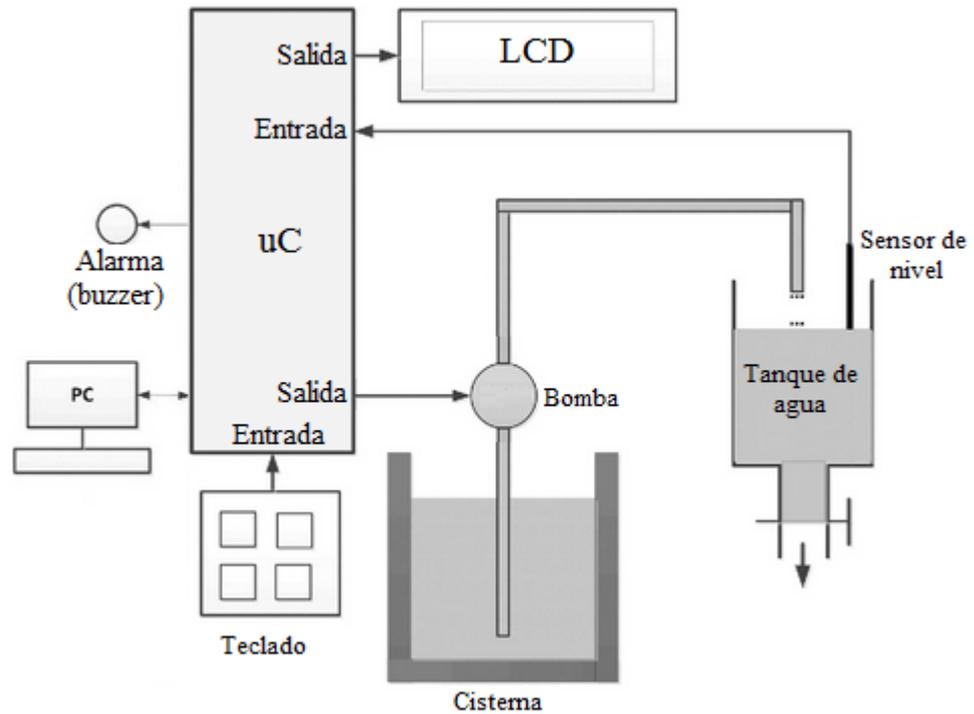


Figura 2. 7: Diagrama de bloques para un sistema de control de nivel de agua basado en microcontroladores PIC más sofisticado.  
Fuente: (Ibrahim, 2015)

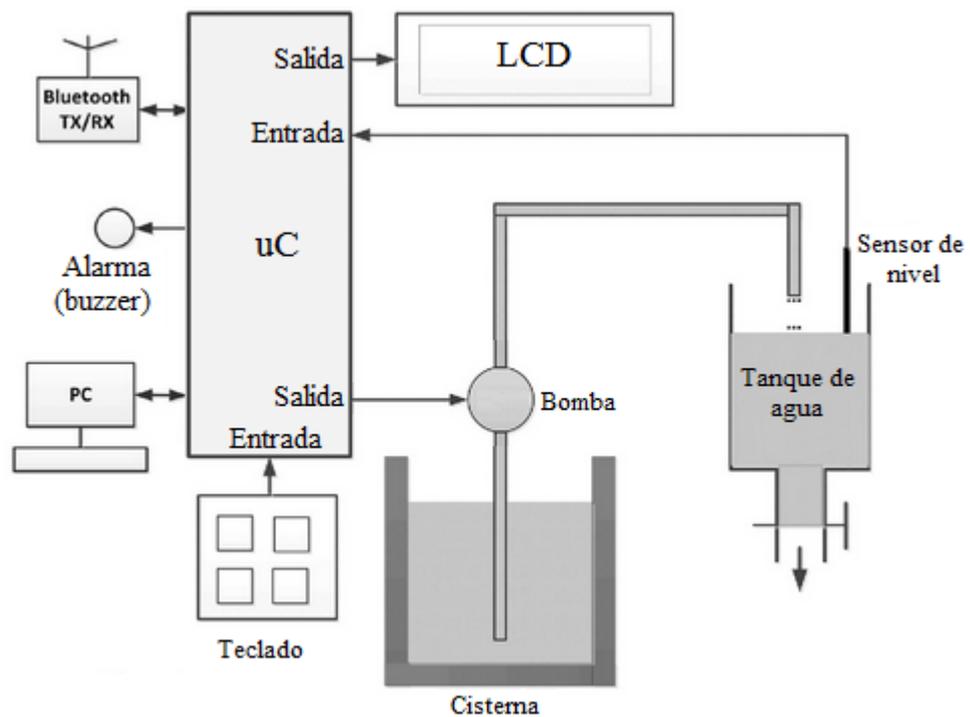


Figura 2. 8: Diagrama de bloques para un sistema de control de nivel de agua basado en microcontroladores PIC más sofisticado.  
Fuente: (Ibrahim, 2015)

## **2.5. Descripción general de los sistemas de entrenamiento de microcontroladores.**

### **2.5.1. Sistema de entrenamiento de microcontroladores PICAXE.**

En esta sección se describe la funcionalidad del sistema de entrenamiento y programación de microcontroladores Picaxe. La plataforma de entrenamiento dispone varios bloques de circuitos, que funcionan de manera independiente. Cada bloque de circuitos posee un etiquetado de conexiones a la fuente de voltaje y a cada uno de los elementos que lo componen. Tiene integrado terminales de conexión, por ejemplo, se dispone de un cable telefónico (RJ-45) para la transmisión de señales de control de un bloque a otro. Todas las configuraciones de manera independiente dependerán del bloque que se desea probar y configurar.

La figura 2.9 muestra el diagrama de bloques para el sistema de entrenamiento, se observa algunos bloques de circuitos conectados a una fuente de alimentación sencilla, donde el voltaje proveniente del transformador llega rectificadora y filtrada, gracias al uso del rectificador de voltaje IC 7805 (rectifica a +5Vcc). El voltaje que sale del IC 7805 es utilizada para la programación de los microcontroladores Picaxe. Por ejemplo, el diodo LED (color verde) se encuentra encendido (véase la figura 2.10) para saber que el programador está alimentado. (Abalos, 2014)

De la figura 2.9 se observa que los pines 2, 3 y 7 del puerto serial (DB25) son utilizados para la comunicación con un ordenador (PC). Para la transmisión de datos entre el computador y el uC se utiliza el pin 2 (serial in),

esto permite cargar la programación en el uC. Mientras que el ordenador lee los datos enviados por el uC siempre por el pin 3 (serial out). La conexión a tierra (GND) se utiliza el pin 7.

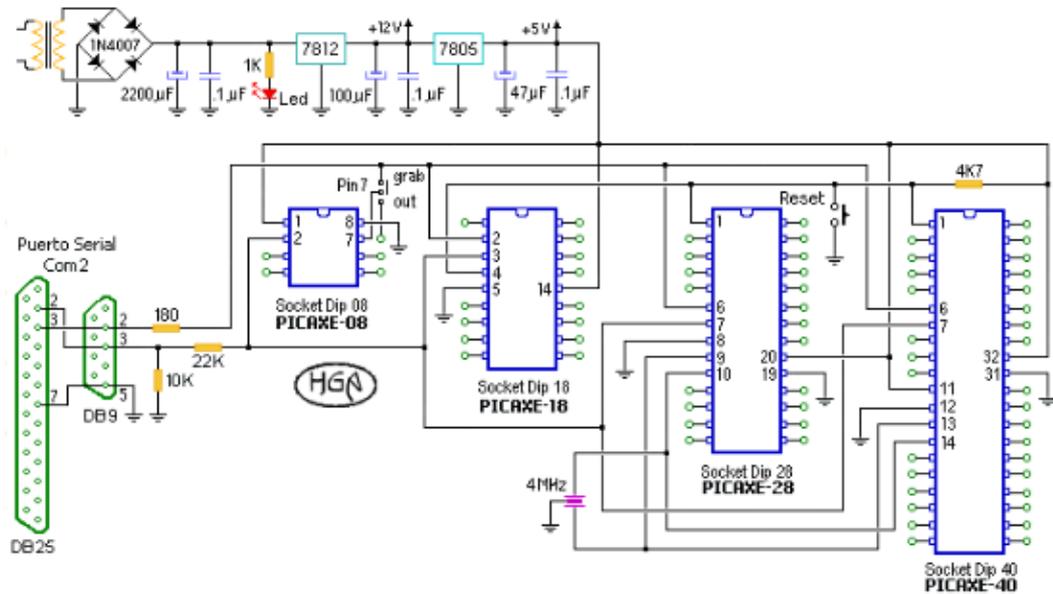


Figura 2. 9: Diagrama de bloques del entrenador PICAXE.

Fuente: (Abalos, 2014; Albán Q. & Peña D., 2015)

Elaborado por: Picaxe

Los microcontroladores PICAXE internamente cuentan con un pequeño programa (firmware), lo que hace que su uso y programación sean fáciles de realizar. Es decir, que este firmware se utiliza para que los uC's PICAXE sean programados en lenguaje de alto nivel "Basic" y a través de "Diagramas de flujo". Una ventaja es que no necesitan de un cargador externo, y son programados sin que sean retirados del circuito donde están funcionando.

Así como cualquier otro microcontrolador, el de PICAXE guarda información de la programación en una memoria FLASH (no volátil). Para modificar la programación grabada en el uC, es necesario que se grabe el código de programación. Adicionalmente, se puede reprogramar el uC

PICAXE de una manera sencilla. Es decir, que el nuevo código de programación se carga en el uC PICAXE logrando borrar el anterior código almacenado inicialmente en la memoria y almacena el nuevo código en la memoria.

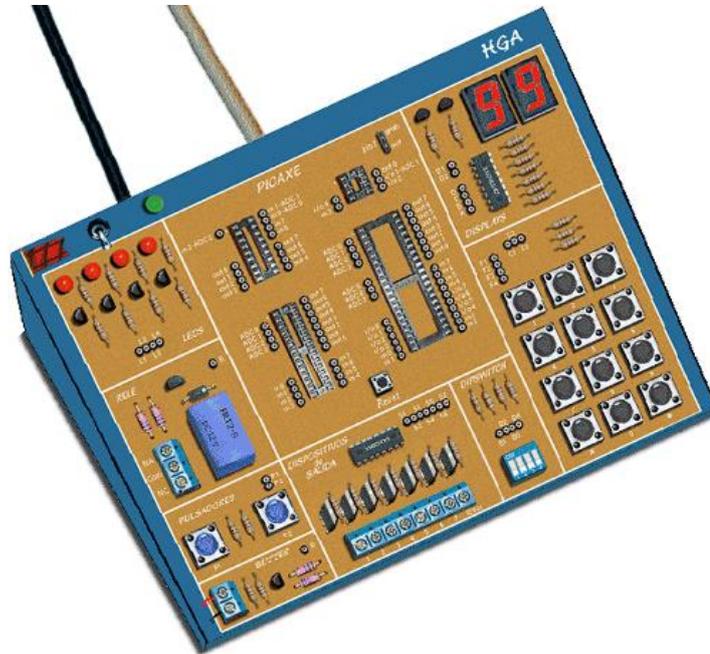


Figura 2. 10: Sistema de entrenamiento de microcontroladores PICAXE.  
Fuente: (Abalos, 2014; Albán Q. & Peña D., 2015)  
Elaborado por: Picaxe

Para proceder a programar (quemar) el uC Picaxe se debe conectar el cable del sistema de entrenamiento (ver figura 2.10) y el software de programación (computador). En la tarjeta de entrenamiento hay un zócalo independiente para utilizar uC's de diversos números de pines. Para resetear el sistema completamente se utiliza el pulsador (P1). El sistema incluye conectores que permiten expandir cada uno de los pines del uC Picaxe, de tal manera, que las señales de E/S sean llevadas a los demás bloques del circuito. Para el caso del uC Picaxe-08 es necesario el uso exclusivo del pin 7 llamado "Jumper", el mismo que permite grabar el uC (Grab), y en la posición Out, para probar el programa.

La figura 2.11 muestra el diagrama esquemático del bloque de circuito, llamado “DISPLAYS”. La tarjeta de entrenamiento Picaxe cuenta dos 2 displays (D1 y D2) de 7 segmentos (ánodo-común) que son conectados en el mismo bus de datos y los ánodos manejados independientemente. Para la funcionalidad de los displays, es necesario utilizar el decodificador 74LS47 que entrega el microcontrolador (A, B, C, D).

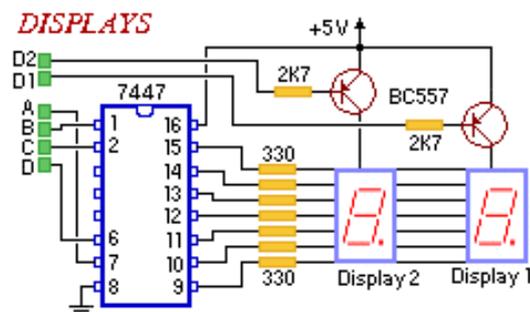


Figura 2. 11: Circuito esquemático de la conexión de los 2 displays.  
Fuente: (Abalos, 2014; Albán Q. & Peña D., 2015)  
Elaborado por: Picaxe

En la figura 2.12 muestra el circuito esquemático de los LEDS, el cual dispones de 4 diodos (L1, L2, L3, L4). Estos diodos son encendidos con un nivel lógico alto en su respectivo acceso.

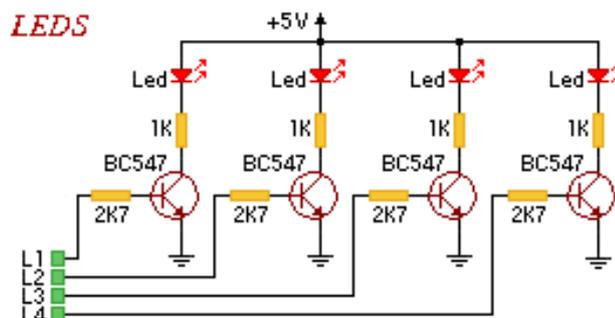


Figura 2. 12: Circuito esquemático de la conexión de 4 LED's.  
Fuente: (Abalos, 2014; Albán Q. & Peña D., 2015)  
Elaborado por: Picaxe

En la figura 2.13 se muestra el circuito esquemático de la salida con RELE, que es utilizado para operar cargas de potencia desde el uC Picaxe. Para su operación, es necesario un nivel lógico alto en el pin R.

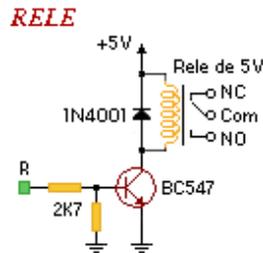


Figura 2. 13: Circuito esquemático de la salida con relé.  
 Fuente: (Abalos, 2014; Albán Q. & Peña D., 2015)  
 Elaborado por: Picaxe

El circuito esquemático de los PULSADORES (véase la figura 2.14) dispone de dos pulsadores normalmente abiertos, tales como, P1 y P2 conectados a GND con resistencias de 10KΩ. Los pulsadores tienen un nivel lógico bajo (0V), pero cuando son pulsados tienen un nivel lógico alto (5V).

**PULSADORES**

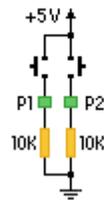


Figura 2. 14: Circuito esquemático de pulsadores.  
 Fuente: (Abalos, 2014; Albán Q. & Peña D., 2015)  
 Elaborado por: Picaxe

El circuito esquemático de los DIPSWITCH (ver figura 2.15) dispone de cuatro interruptores (S1, S2, S3, S4) conectados a GND con resistencias de 10KΩ. El funcionamiento es que se tiene un nivel lógico bajo, si el interruptor está abierto, pero si el interruptor se cierra, se tiene un nivel lógico alto.

**DIPSWITCH**

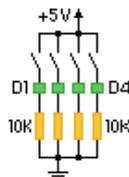


Figura 2. 15: Circuito esquemático del Dipswitch.  
 Fuente: (Abalos, 2014; Albán Q. & Peña D., 2015)  
 Elaborado por: Picaxe

En la figura 2.16 se ilustra la conexión del TECLADO MATRICIAL, este tiene un teclado con una matriz de 3x4, con caracteres de 0 a 9, \*, y #. Las columnas (C1, C2, C3) del teclado tienen conectadas unas resistencias de 2,7K $\Omega$ , que sirven para fijar un nivel lógico alto cuando no se está oprimiendo ninguna tecla.

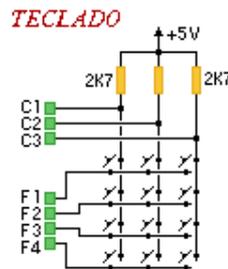


Figura 2. 16: Circuito esquemático del Teclado Matricial.  
 Fuente: (Abalos, 2014; Albán Q. & Peña D., 2015)  
 Elaborado por: Picaxe

En la figura 2.17 se observa el circuito de conexión BUZZER, el cual puede crear automáticamente sonidos de diferentes frecuencias utilizando el comando sound.

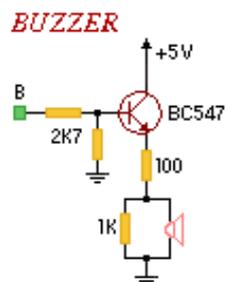


Figura 2. 17: Circuito esquemático del Buzzer.  
 Fuente: (Abalos, 2014; Albán Q. & Peña D., 2015)  
 Elaborado por: Picaxe

En el circuito de la figura 2.18 se ilustra el circuito para los DISPOSITIVOS DE SALIDA, que provee de siete salidas (+5 Vcc o +12 Vcc) con óptima capacidad de corriente. Para amplificar la corriente de salida del microcontrolador se utiliza el IC ULN2003 como driver de los transistores de salida. La habilitación del transistor de salida correspondiente se hace

conectando un nivel lógico alto en los pines de conexión (S1, S2, S3, S4, S5, S6, S7). El transistor de salida conmuta a tierra, por lo tanto, la carga hay que colocarla entre las salidas (1, 2, 3, 4, 5, 6, 7), y la alimentación +5 Vcc, o +12 Vcc.

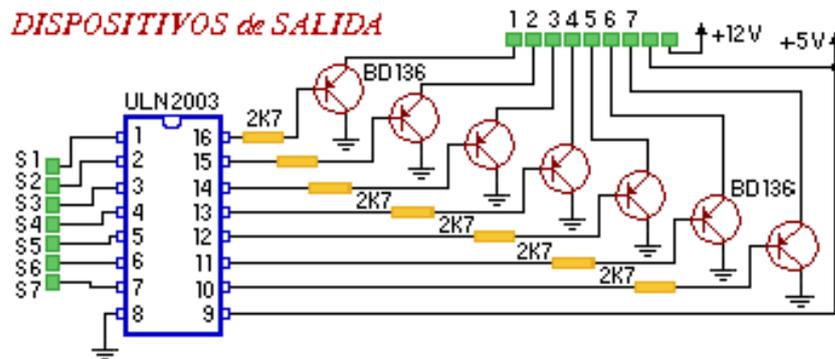


Figura 2. 18: Circuito esquemático de los dispositivos de salida.  
Fuente: (Abalos, 2014; Albán Q. & Peña D., 2015)  
Elaborado por: Picaxe

### 2.5.2. Sistema de entrenamiento de microcontroladores EasyPic 6.

La tarjeta EasyPic6 es producida por la empresa Mikroelektronika la cual presenta una propuesta muy interesante de desarrolladores y compiladores en lenguaje Basic, Ensamblador, Pascal, Lenguaje C entre otros. Se puede observar una imagen de la tarjeta en la figura 2.19, en la que se han conservado números referenciales para destacar las características de la tarjeta de una mejor manera.

El Sistema Modular EasyPic6 desarrolla todas las funciones para casi todos los microcontroladores PIC de la familia Microchip. Está diseñado para permitir a los estudiantes y docentes para probar fácilmente y explorar las capacidades de los microcontroladores PIC. También permite a los microcontroladores PIC una interfaz con circuitos externos y una amplia gama

de dispositivos periféricos. Los estudiantes pueden, por tanto, centrarse en el desarrollo de software solamente.

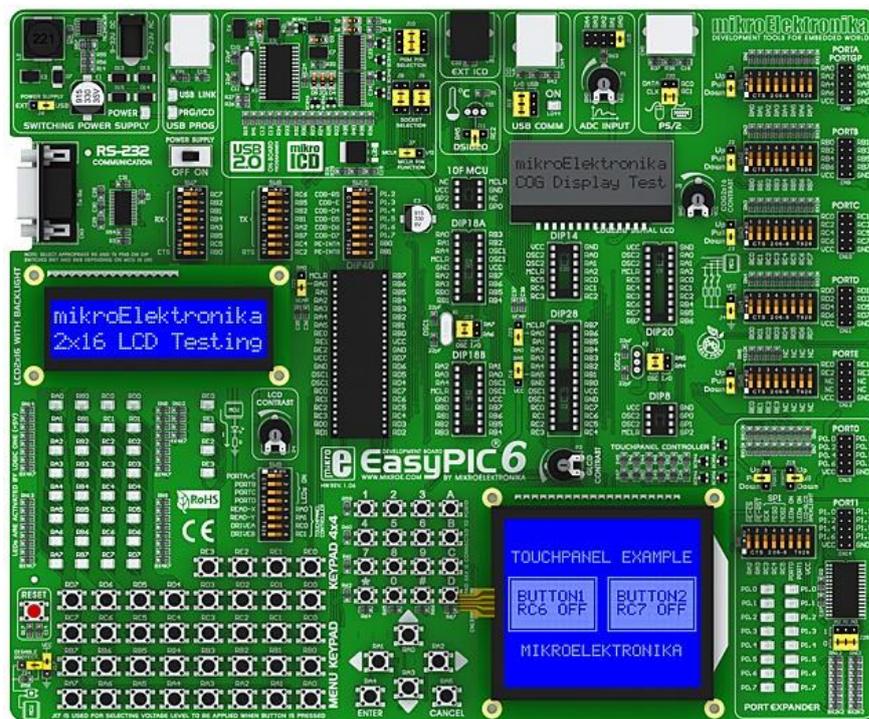


Figura 2. 19: Sistema modular de entrenamiento EasyPic6.  
Fuente: (Mikroelektronika, 2018)

Hay marcas de identificación a cada componente en una serigrafía, tanto en la parte superior e inferior. Estas marcas describen la conexión con el microcontrolador, los modos de operación y proporcionar información útil adicional de modo que casi no hay necesidad de esquemas adicionales. En la figura 2.20 se observa con mayor detalle las siguientes características de la numeración presentada:

1. Regulador del voltaje de la fuente de alimentación.
2. Conector USB para el programador incorporado.
3. Programador USB 2.0 con soporte de mikroICD.
4. Zócalo para el sensor de temperatura DS1820 que permite mediciones de temperatura con 0.5 grados de exactitud.

5. Conector para el depurador externo (ICD2 o ICD3) de Microchip.
6. Conector para la comunicación USB.
7. Entradas para probar el convertidor A/D.
8. Conector PS/2.
9. Visualizador LCD 2x16 incorporado.
10. Interruptor DIP que permiten el funcionamiento de las resistencias pull-up/pull-down.
11. Puente para seleccionar las resistencias pull-up/pull-down.
12. Conectores de los puertos de E/S.
13. Zócalo para colocar el microcontrolador PIC.
14. Controlador del panel táctil. microcontrolador.
15. Extensor de puertos.
16. Conector del visualizador LCD gráfico 128x64.
17. Potenciómetro de contraste del visualizador LCD gráfico 128x64.
18. Conector de panel táctil.
19. Teclado menú.
20. Teclado 4x4.
21. Botones de presión para simular las entradas digitales.
22. Puente para seleccionar el estado lógico de los botones de presión.
23. Puente para poner en cortocircuito una resistencia de protección.
24. Botón para resetear el microcontrolador.
25. 36 diodos LED que indican el estado lógico en los pines.
26. Ajuste de contraste del visualizador LCD alfanumérico.
27. Conector del visualizador LCD alfanumérico.
28. Conector para la comunicación RS-232.

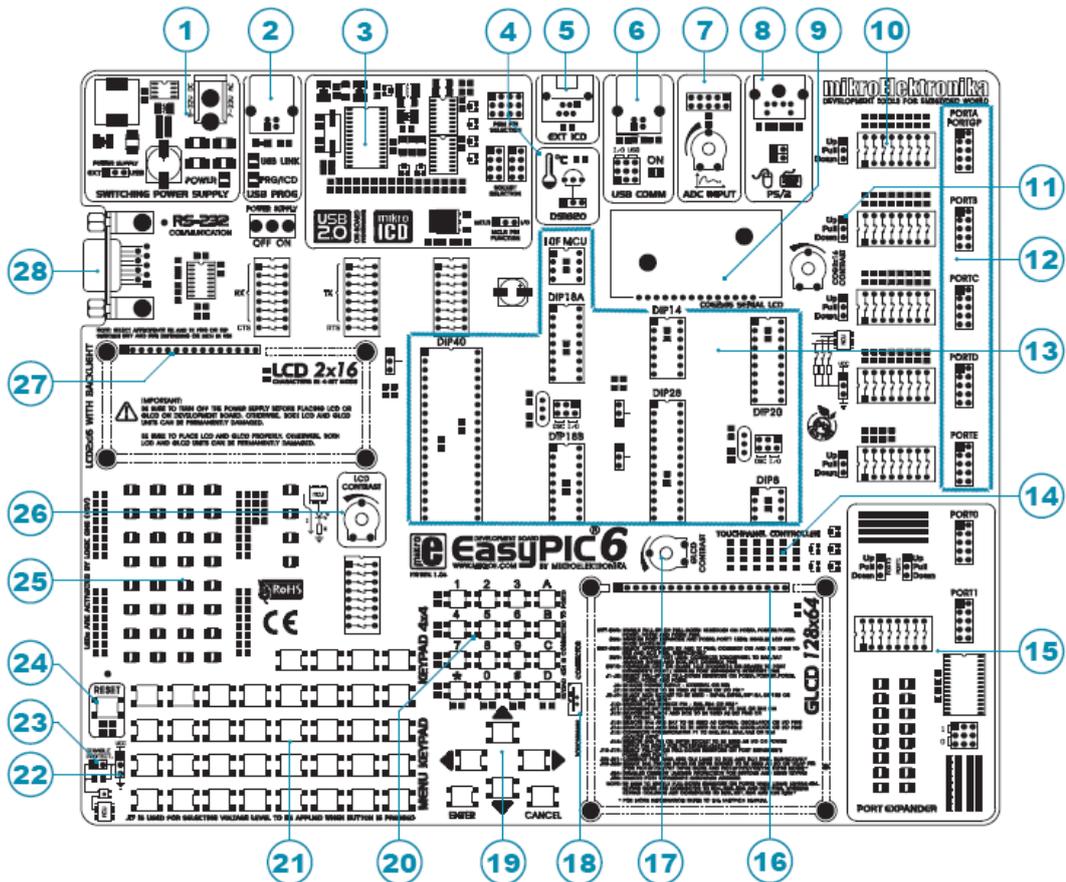


Figura 2. 20: Diagrama esquemático del sistema modular de entrenamiento EasyPic6  
Fuente: (Mikroelektronika, 2018)

EasyPic6 fue diseñado como herramienta de apoyo a la enseñanza de los microcontroladores, cuyas características son las siguientes:

1. Prueba de instrucciones a través de un ICD incorporado;
2. Programación en circuito
3. El desarrollo de aplicaciones.

### 2.5.3. Sistema de entrenamiento de microcontroladores PIC EDUTRONICA JJTSA.

El sistema se compone de dos tarjetas: Quemador (Programador) y el Sistema de Evaluación de programas. Mediante este sistema se pueden elaborar y probar una serie de programas que se aproximan a la realidad de

control que alguien quiera realizar. Se ahorra, tiempo y esfuerzo al evitar conexiones en un “protoboard”, el cual puede generar demasiados problemas como son falsos contactos e incertidumbre en el funcionamiento al no saber dónde está la falla.

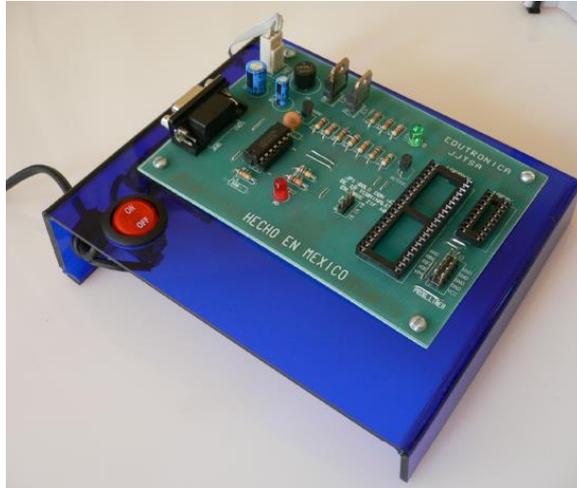


Figura 2. 21: Tarjeta de la etapa de programación de PIC.  
Fuente: (Edutrónica, 2018)

En la figura 2.21 se muestra el programador que permite grabar una gran cantidad de modelos de microcontrolador PIC, mediante el uso de dos bases una de 40 pines y otra de 18 pines. Para mayor flexibilidad, se puede insertar una base tipo ZIF (Zero In Force) sobre la base de 40 pines para programar la mayoría de los modelos. Cuenta además con un conector que puede usarse para programar externamente circuitos en línea.

En la figura 2.22 se ilustra la tarjeta que permite realizar la evaluación de programas, tiene una gran flexibilidad en el manejo de señales de entradas analógicas /digitales y salidas digitales. Los tipos de microcontrolador que se pueden utilizar en este sistema son los modelos de PIC 16F84, 16F627, 16F628, 16F877, 16F874 y 18F452 y aquellos otros que sean compatibles en PINOUT con estos modelos.



Figura 2. 22: Tarjeta de la etapa de evaluación de programas  
Fuente: (Edutrónica, 2018)

Para este sistema de evaluación de programas se detalla sus componentes:

1. Display de Cristal Líquido de 2x20 caracteres (LCD).
2. Doble Display de 7 segmentos.
3. Puertos con salida de LED's.
4. Un puerto con Interruptor Táctil.
5. Interruptor táctil para interrupción externa (RB0/INT).
6. Buzzer Piezoeléctrico.
7. Sensor analógico de luz (Fotorresistencia).
8. Sensor de Temperatura (LM35DZ).
9. Dos resistencias variables para voltaje variable (0-5 VCD).
10. Sensor infrarrojo para decodificación de señales infrarrojas. (Control Remoto).
11. Interruptor de Reset.
12. Conector "header" macho del Puerto B para cargas externas al sistema.
13. Conector "header" macho para programación en línea del PIC, sin necesidad de estar quitando el PIC para programarlo.

Mediante estos elementos se pueden realizar programas que le permiten al usuario rápidamente elevar el nivel y habilidad en el desarrollo de programas de interés para sus aplicaciones.

## **2.6. Análisis de prestaciones de los sistemas de entrenamiento de microcontroladores PIC.**

A continuación, se realiza un análisis de prestaciones de los sistemas de entrenamiento de microcontroladores que fueron descritos en la sección 2.2. El análisis se realiza en función del programa de estudios y del syllabus de la asignatura de Microcontroladores, tanto para la Carrera de Ingeniería en Telecomunicaciones, como la Carrera de Ingeniería Electrónica en Control y Automatismo.

### **2.6.1. Análisis de prestaciones del entrenador de microcontroladores PICAXE**

Con este entrenador de microcontroladores PICAXE, existe la posibilidad de armar circuitos básicos que son necesarios para cubrir el programa de estudios de la asignatura de Microcontroladores. Debiendo indicarse que se dedica exclusivamente a los microcontroladores PICAXE. Para el caso de los microcontroladores de los fabricantes Intel y Microchip, es necesario utilizar otras tarjetas de entrenamiento para cada una de ellas. La desventaja de este sistema de entrenamiento es que los microcontroladores PICAXE no son considerados en el aprendizaje de los estudiantes que toman la asignatura. Por lo tanto, no es gran ventaja el uso de este sistema modular de microcontroladores.

## **2.6.2. Análisis de prestaciones del entrenador de microcontroladores EasyPic6.**

El sistema modular EasyPic6 es una tarjeta muy amplia y completa, el mismo que viene incorporado de un programador, herramienta importante en el trabajo con microcontroladores. En este caso un ICD que permite la prueba paso a paso y la depuración de instrucciones. Trae también incorporados un panel sensible al tacto (touch screen) y un panel gráfico (GLCD). Estos dispositivos son empleados con frecuencia en sistemas de automatización.

Lo descrito anteriormente y unido a la flexibilidad para manejar pines (entrada y salida) con o sin el uso de pull ups, pull downs, interruptores o LED's, lo hacen atractivo para múltiples aplicaciones a implementarse. Adicional hay que añadir que el sistema de entrenamiento facilitaría el proceso de enseñanza-aprendizaje para la práctica de laboratorio. Desde el punto de vista didáctico esta herramienta aporta mayor flexibilidad a la enseñanza y que el docente puede utilizar para afinar sus prácticas antes de exponerlas al estudiantado.

Una vez realizado el análisis de prestaciones se puede concluir, que el EasyPic6 sería una solución factible por su compatibilidad con las distintas gamas (160 Pic's) de la familia de los microcontroladores MicroChip, para su implementación en el Laboratorio de Electrónica. La desventaja es el costo de adquisición, que por motivos de importación saldría muy elevado el costo por unidad.

### **2.6.3. Análisis de prestaciones del Entrenador de Microcontroladores PIC EDUTRONICA JJTSA.**

Las tarjetas de programación y evaluación de programas EDUTRÓNICA presentan un entorno similar de trabajo que del PICAXE en el sentido de su apariencia con display LCD, LED's, potenciómetros, área para prototipos. Sus diferencias fundamentales radican en que trabajan con microcontroladores PIC (de Microchip) y PICAXE, así como también la cantidad de pulsadores que es muy inferior. La fortaleza de la tarjeta radica en la presentación de la utilización de características avanzadas en microcontroladores.

## **CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS EXPERIMENTALES**

### **3.1. Introducción del sistema de entrenamiento propuesto.**

En este trabajo, resuelven el problema de la comprensión del alumno y el desarrollo de habilidades en el diseño de sistemas integrados utilizando el microcontrolador PIC18F8720 mediante la demostración de experimentos prácticos de laboratorio. Además, se llevan a cabo el desarrollo de código de programación de alto nivel C (software CCS C), simulación de diagrama de circuito utilizando ISIS Proteus. Esto es para ayudar a los estudiantes de las Carreras de Ingeniería Electrónica en Control y Automatismo y de Telecomunicaciones a conectar sus conocimientos teóricos con la experiencia práctica. Cada uno de los experimentos se llevó a cabo utilizando el sistema de entrenamiento diseñado para utilizar el PIC18F8720, el programador CCS C, el software de simulación Proteus 8.0.

A través de la experiencia del docente, M. Sc. Luis Córdova Rivadeneira el mismo que imparte la asignatura de Microcontroladores. De acuerdo a esa experiencia del curso de microcontroladores y la participación activa de los estudiantes se manifiestan en proyectos de laboratorio completos y prácticos sobre la resolución de problemas de la vida real. La sesión de laboratorio con la placa de desarrollo y el software demostrados en este trabajo de titulación no es ambigua. Durante las pruebas experimentales con el sistema de entrenamiento al final del semestre muestra que, con el aumento cronológico en el diseño del hardware, el estudiante mostró la confianza en el diseño de nuevos sistemas independientes con hardware y software bastante

complicado. La instalación de un laboratorio de este tipo es fácil con el material descrito en este documento.

### **3.2. Módulo de entrenamiento para el microcontrolador PIC18F8720.**

El presente módulo de entrenamiento permite realizar una serie de prácticas muy útiles para el aprendizaje de los microcontroladores de gama alta, como lo es el PIC18F8720. Este módulo consiste en una serie de elementos asociados que están disponibles para que sean controlados por un microcontrolador, estos bloques de desarrollo son:

1. Microcontrolador PIC18F8720 con cristal de cuarzo de 40 MHz
2. Bloque de 8 entradas digitales.
3. Bloque de 8 salidas digitales
4. Bloque de Display LCD de 4x20
5. Bloque de display GLCD
6. Bloque de 4 display de 7 segmentos
7. Bloque de 4 matrices de led de 8x8
8. Bloque de teclado matricial de 4x3
9. Bloque de 4 potenciómetros para entradas analógicas al micro
10. Salida de relé
11. Salida de señal audible
12. 2 módulos de comunicación serial asíncrona

La figura 3.1 muestra el circuito esquemático dibujado en el programa PROTEUS

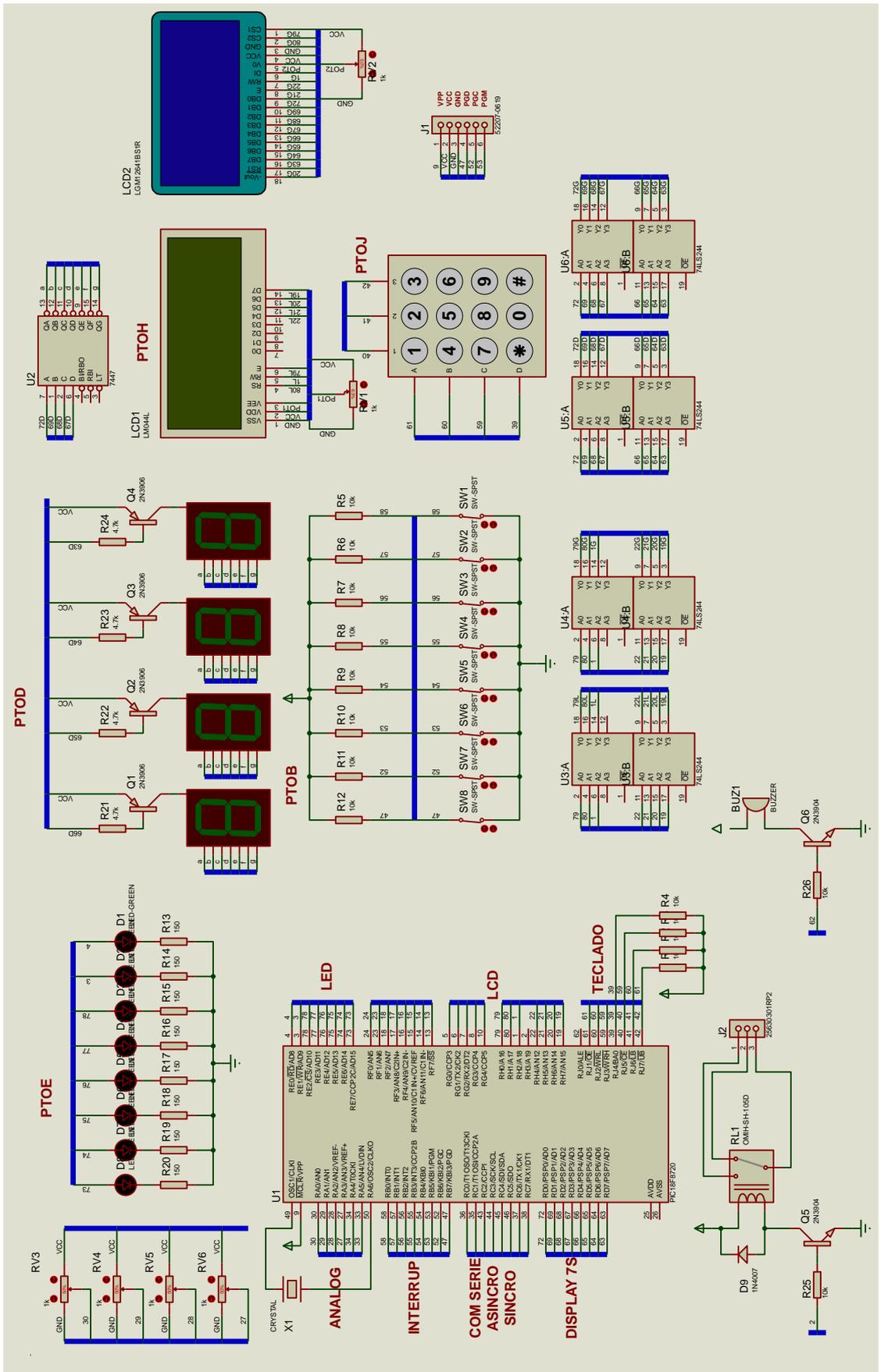


Figura 3. 1: Diseño del sistema de entrenamiento en Proteus para PIC18F8720.

Elaborado por: Autor.

La figura 3.2 muestra el diseño esquemático por bloques del sistema de entrenamiento de microcontroladores PIC18F8720 realizado sobre el software Eagle. En las siguientes subsecciones se describen por etapas o bloques el sistema de entrenamiento con el microcontrolador PIC18F8720.

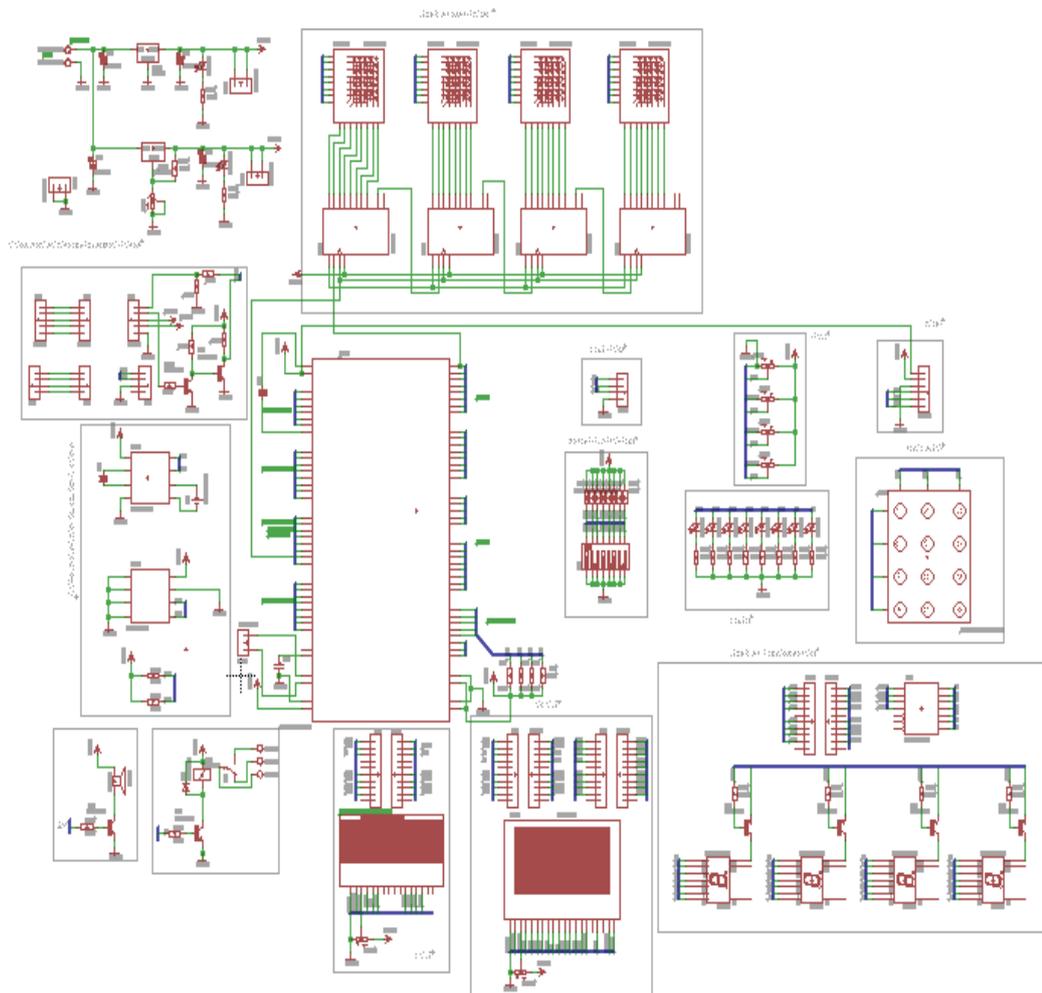


Figura 3. 2: Diseño esquemático del sistema de entrenamiento de uC PIC18F8720 en Eagle.

Elaborado por: Autor.

### 3.2.1. Microcontrolador PIC18F8720 con cristal de cuarzo de 40 MHz.

Este bloque (véase la figura 3.3) contiene el elemento fundamental de todo el sistema de entrenamiento, esto es el microcontrolador de la familia Microchip PIC18F8720 en encapsulado TQFP de 80 pines con un cristal de cuarzo de 40 MHz.

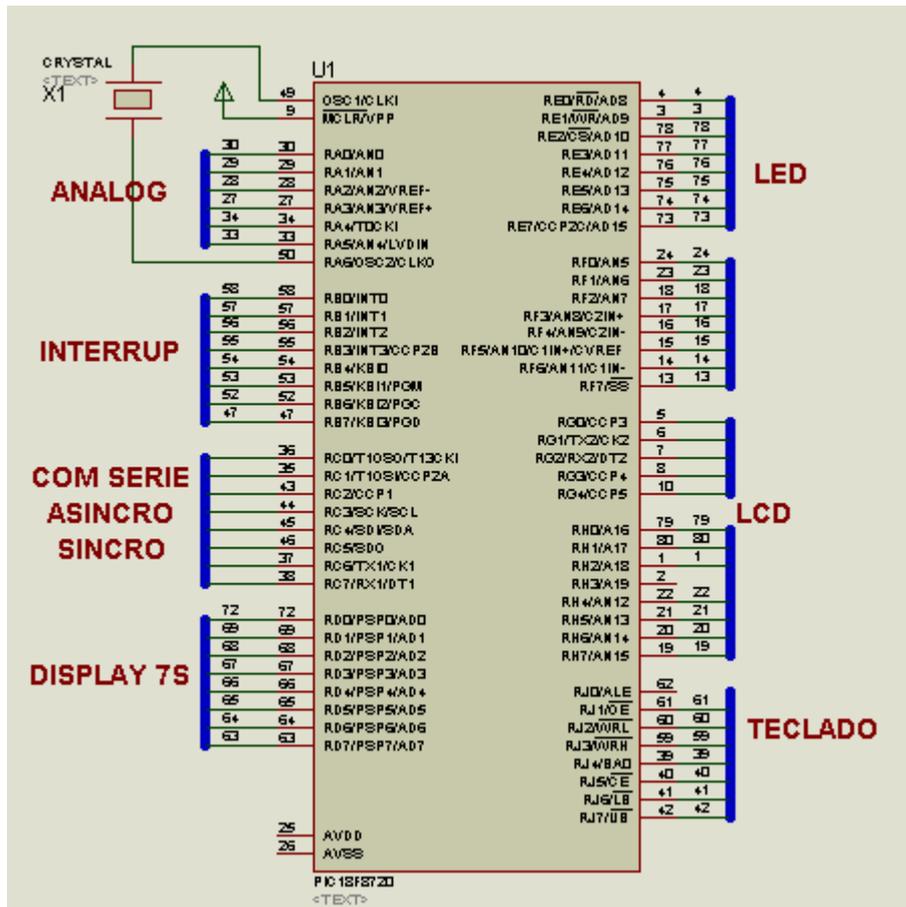


Figura 3. 3: Etapa 1 del diseño esquemático del sistema de entrenamiento de uC PIC18F8720.

Elaborado por: Autor.

### 3.2.2. Bloque de 8 entradas digitales.

Este bloque posee un dipswitch de 8 interruptores con 8 resistencias de 10 KΩ conectadas en configuración PULL UP (véase la figura 3.4), para el ingreso de señales digitales al microcontrolador. Estas 8 señales entran al microcontrolador a través del puerto B, el cual está diseñado para el ingreso de interrupciones externas como B0 y B4 hasta B7. Al cerrarse cada interruptor ingresa un bajo, es decir, que se trabaja con lógica negada para el ingreso de las señales.

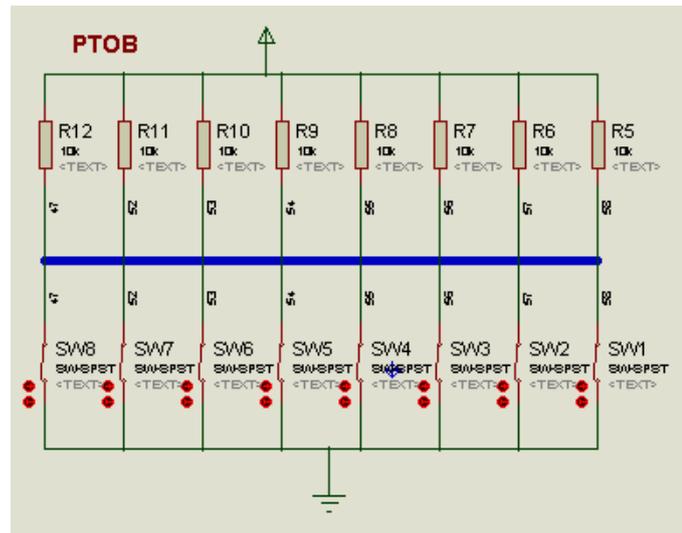


Figura 3. 4: Etapa 2 del diseño esquemático del sistema de entrenamiento de uC PIC18F8720.

Elaborado por: Autor.

### 3.2.3. Bloque de 8 salidas digitales.

Este bloque posee un conjunto de 8 led's con 8 resistencias de 150  $\Omega$  conectadas en serie, para la salida de señales digitales desde el microcontrolador, tal como se muestra en la figura 3.3. Estas 8 señales salen del microcontrolador a través del puerto E (PTOE), por lo que se requiere un nivel alto para encender cada led, es decir, que trabaja con lógica positiva para la salida de señales.

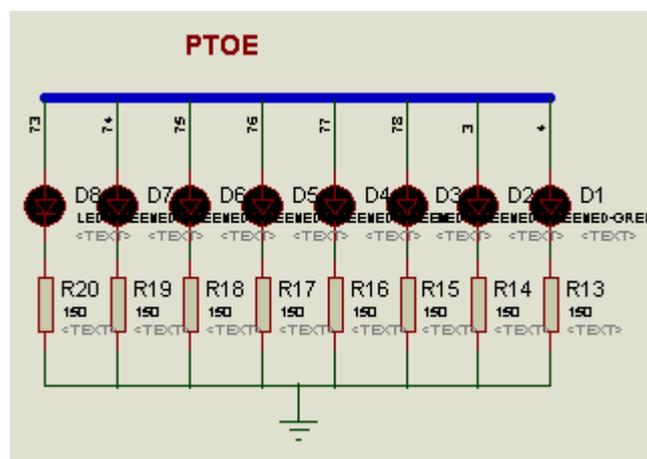


Figura 3. 5: Etapa 3 del diseño esquemático del sistema de entrenamiento de uC PIC18F8720.

Elaborado por: Autor.

### 3.2.4. Bloque de display LCD de 4x20.

Este bloque posee un LCD de 4x20 (4 filas por 20 columnas), para la visualización de mensajes provenientes del microcontrolador PIC18F8720. El puerto utilizado para la conexión del LCD al microcontrolador es el puerto H, tal como se muestra en la figura 3.6.

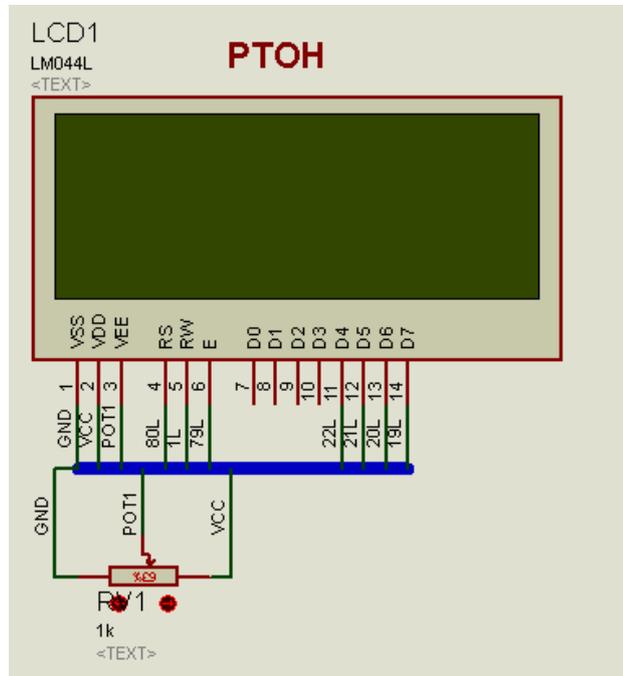


Figura 3. 6: Etapa 4 del diseño esquemático del sistema de entrenamiento de uC PIC18F8720.

Elaborado por: Autor.

A continuación, se indican las conexiones de los puertos del microcontrolador al display LCD de 4x20 y del potenciómetro:

H0 => PIN ENABLE

H1 => PIN RS

H2 => PIN RW

H4 => PIN D4

H5 => PIN D5

H6 => PIN D6

H7 => PIN D7

GND => PIN 1

VDD => PIN 2

POTENCIÓMETRO => PIN 3

### **3.2.5. Bloque de Display GLCD.**

Este bloque posee un GLCD de 128 pixeles de largo por 64 pixeles de ancho (véase la figura 3.7), para la visualización de mensajes provenientes del microcontrolador PIC18F8720. Los puertos utilizados para la conexión del GLCD al microcontrolador son el puerto D y el puerto H. Hay que recordar que el LCD también utiliza el puerto H por tal motivo no se puede trabajar al mismo tiempo con los dos dispositivos conectados al microcontrolador. Los pines del GLCD se conectan de la siguiente manera:

PIN 1: VSS conectado a GND

PIN 2: VDD conectado a +5 V

PIN 3: V0 - LCD conectado a un potenciómetro de 20 k $\Omega$

PIN 4: D/I – Datos conectados a H2

PIN 5: R/W – Lectura o escritura conectado a H4

PIN 6: Enable conectado a H5

PIN 7-14: Data Bus 0 a 7 está conectado al puerto D

PIN 15: Chip Select 1 está conectado a H0

PIN 16: Chip Select 2 está conectado a H1

PIN 17: Reset está conectado a H0

PIN 18: Voltaje negativo conectado a un potenciómetro de 20 k $\Omega$

PIN 19: Voltaje positivo para LED backlight está conectado a +5V

PIN 20: Voltaje negativo para LED backlight conectado a GND.

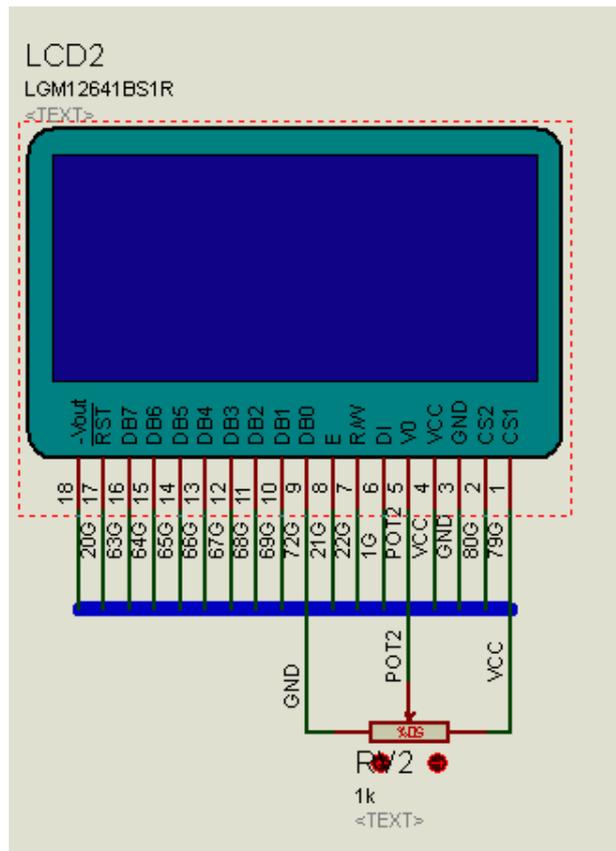


Figura 3. 7: Etapa 5 del diseño esquemático del sistema de entrenamiento de uC PIC18F8720.

Elaborado por: Autor.

### 3.2.6. Bloque de 4 displays de 7 segmentos.

Este bloque posee 4 display de 7 segmentos ánodo común, multiplexados mediante 4 transistores 2N3906 que trabajan en corte y saturación para habilitarlos o deshabilitarlos. De tal forma, que solo uno se encienda a la vez, los cuatro displays de 7 segmentos están conectados en paralelo con la señal que sale de un decodificador de BCD a 7 segmentos 74LS47 tal como se muestra en la figura 3.8. Los datos de entrada del decodificador, así como las señales que habilitan a los transistores por sus bases son dados por el puerto D del microcontrolador

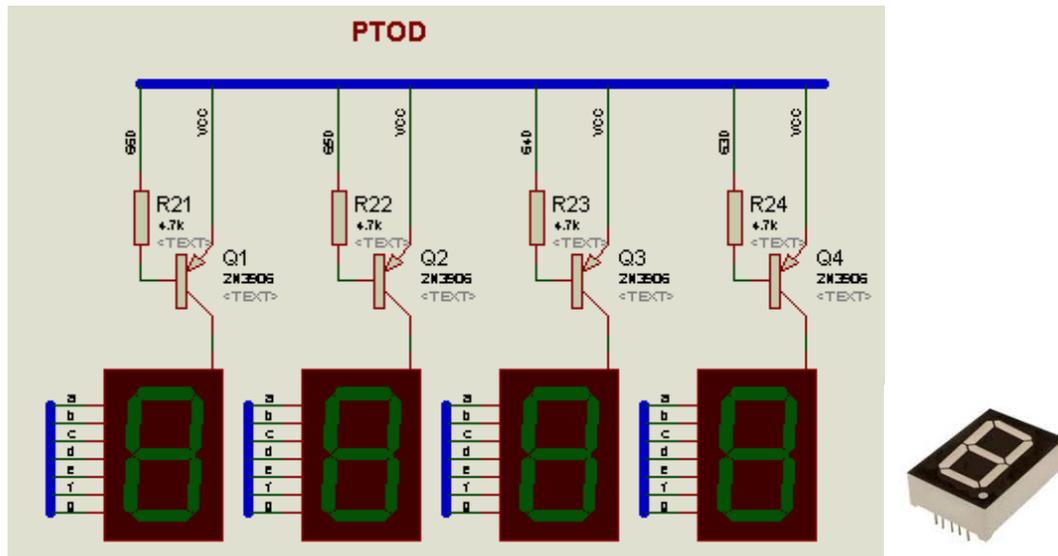


Figura 3. 8: Etapa 6 del diseño esquemático del sistema de entrenamiento de uC PIC18F8720.

Elaborado por: Autor.

### 3.2.7. Bloque de 4 matrices de LEDs de 8x8.

Este bloque posee 4 matrices de led de 8x8 (véase la figura 3.9) que son controladas por el micro 18F8720 a través de comunicación serie síncrona SPI para lo cual se requiere de 4 registros de desplazamiento serie-paralelo 4094 de familia lógica CMOS, componente que controla cada una de las columnas de las 4 matrices led. Las filas de las matrices se encuentran en paralelo y la información que reciben del micro lo hacen a través del puerto F.

Las conexiones por SPI el microcontrolador se lo hace a través de:

Salida de SPI => PIN\_E0

Clock de SPI => PIN\_C6

Las conexiones de la matriz de led están configuradas de la siguiente manera:

Por la parte de arriba:

C8 C7 F2 C1 F4 C6 C4 F1

Por la parte de abajo:

F5    F7    C2    C3    F8    C5    F6    F3



Figura 3. 9: Etapa 7 del diseño esquemático del sistema de entrenamiento de uC PIC18F8720.

Elaborado por: Autor.

### 3.2.8. Bloque del teclado matricial de 4X3.

Este bloque permite el ingreso de valores numéricos al micro controlador a través de un teclado matricial de 4 filas por 3 columnas con un total de 12 teclas que van del 0 al 9, una tecla \* y otra tecla #. Como las filas trabajan como entradas es necesario ponerles resistencias de pull up en el orden de los 10 K $\Omega$ . El teclado matricial se encuentra conectado en el puerto J del microcontrolador PIC18F8720 de la siguiente manera:

Los pines correspondientes a las columnas

C1 => PIN\_J1

C2 => PIN\_J2

C3 => PIN\_J3

Los pines correspondientes a las filas

F1 => PIN\_J4

F2 => PIN\_J5

F3 => PIN\_J6

F4 => PIN\_J7

Mientras que el teclado puesto en el entrenador tiene la siguiente configuración:

C2   F1   C1   F4   C3   F3   F2



Figura 3. 10: Etapa 8 del diseño esquemático del sistema de entrenamiento de uC PIC18F8720.

Elaborado por: Autor.

### **3.2.9. Bloque de 4 potenciómetros para entradas analógicas al PIC18F8720.**

Este bloque posee 4 potenciómetros de precisión para el ingreso de hasta cuatro señales analógicas que se deseen medir, estos potenciómetros se encuentran polarizados entre 0 y 5 VDC para poder trabajar con el convertidor analógico a digital que posee en el interior del microcontrolador, ya sea con resolución de 8 bits o de 10 bits dependiendo de la aplicación que se desee hacer. Las señales de los potenciómetros ingresan a los siguientes pines:

POT1 => PIN\_A0

POT2 => PIN\_A1

POT3 => PIN\_A2

POT4 => PIN\_A3

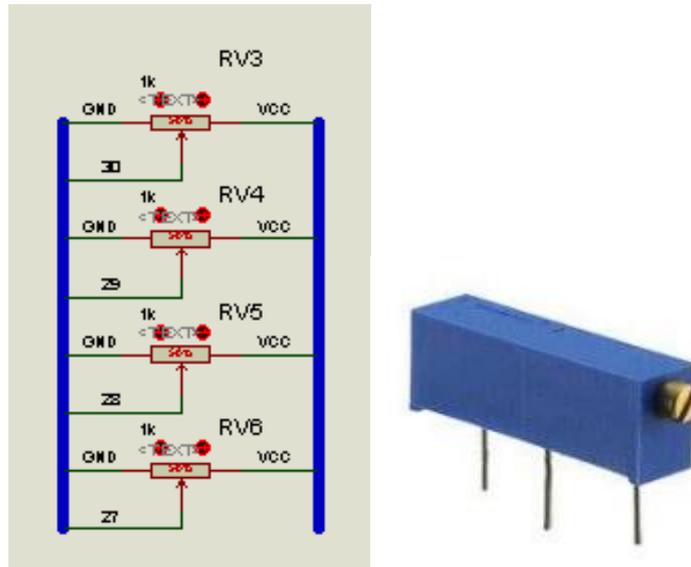


Figura 3. 11: Etapa 9 del diseño esquemático del sistema de entrenamiento de uC PIC18F8720.  
Elaborado por: Autor.

### 3.2.10. Salida de relé.

Este bloque posee una salida de relé para activar cargas de mayor potencia, sean estas en voltaje continuo o en voltaje alterno. El relé está controlado por el PIN\_H3 del microcontrolador, a través de un transistor 2N3904 con una resistencia de 10 KΩ en su base y un diodo de paso libre en anti paralelo con la bobina, tal como se muestra en la figura 3.12.

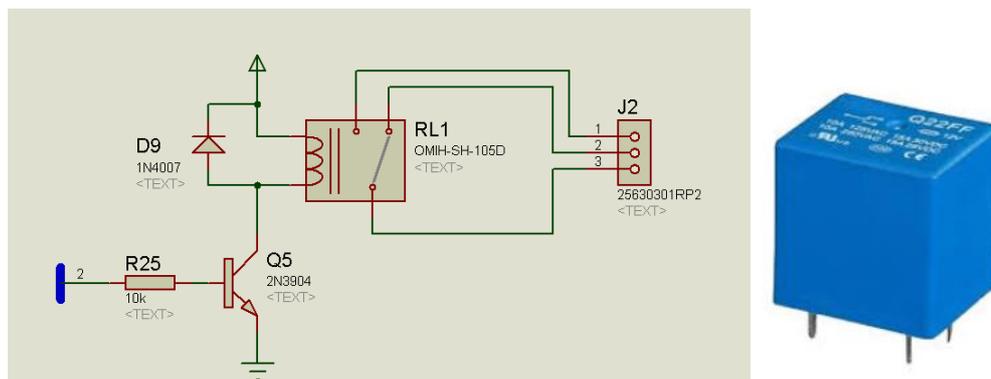


Figura 3. 12: Etapa 10 del diseño esquemático del sistema de entrenamiento de uC PIC18F8720.  
Elaborado por: Autor.

### 3.2.11. Salida de señal audible.

Este bloque posee una salida de señal audible mediante un buzzer, para realizar prácticas que lo requieran. El buzzer está controlado por el PIN\_J0 del micro controlador, a través de un transistor 2N3904 con una resistencia de 10 K $\Omega$  en su base, tal como se muestra en la figura 3.13.

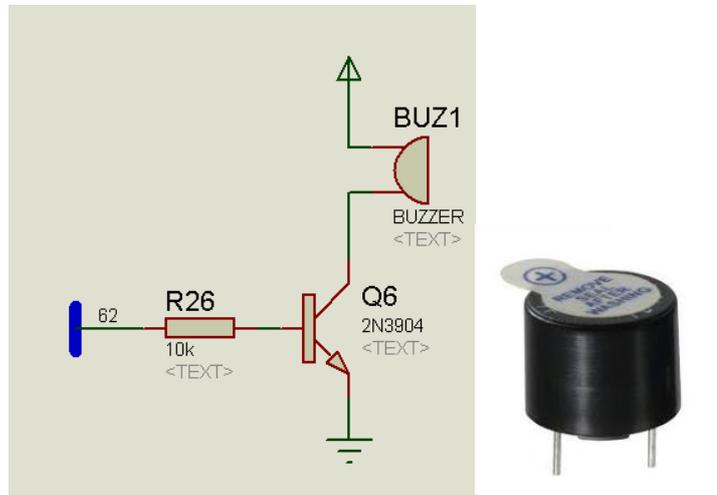


Figura 3. 13: Etapa 11 del diseño esquemático del sistema de entrenamiento de uC PIC18F8720.

Elaborado por: Autor.

### 3.2.12. Módulos de comunicación serial asíncrona.

Este bloque cuenta con dos módulos de comunicación serie asíncrona UART con interrupción por Hardware para ser utilizados para la comunicación serial entre el microcontrolador y un sin número de dispositivos que funcionan con el protocolo de comunicación RS232 TTL.

El primer módulo TX1 y RX1 utiliza los siguientes pines:

TX1 => PIN\_C6      => PIN37

RX1 => PIN\_C7      => PIN38

Funciona con voltajes de 0 y 5V, mientras que el segundo módulo TX2 y RX2 utiliza los siguientes pines:

TX2 => PIN\_G1      => PIN6

RX2 => PIN\_G2      => PIN7

Cabe resaltar que este segundo módulo posee una circuitería compuesta de transistores y resistencias para convertir los niveles lógicos de funcionamiento tanto de entrada como de salida de 0 a 3.3V tal como se muestra en la figura 3.14.

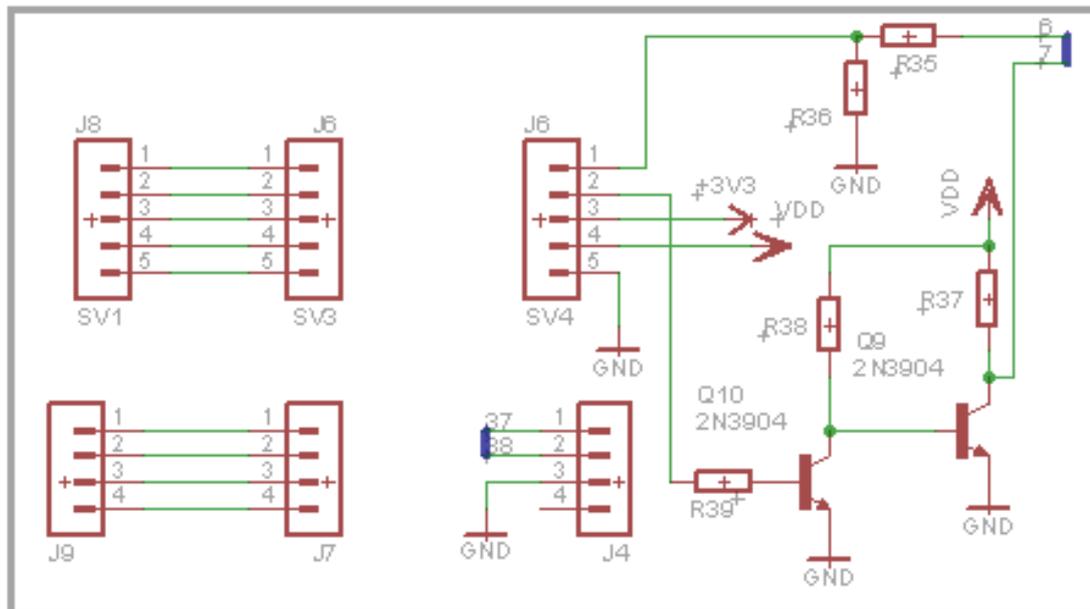


Figura 3. 14: Etapa 9 del diseño esquemático del sistema de entrenamiento de uC PIC18F8720.

Elaborado por: Autor.

### 3.3. Aplicación práctica 1: conversión a código gray usando led's.

En la figura 3.15 se muestra un circuito esquemático que permite el manejo de los puertos de entrada y salida del microcontrolador PIC18F8720.

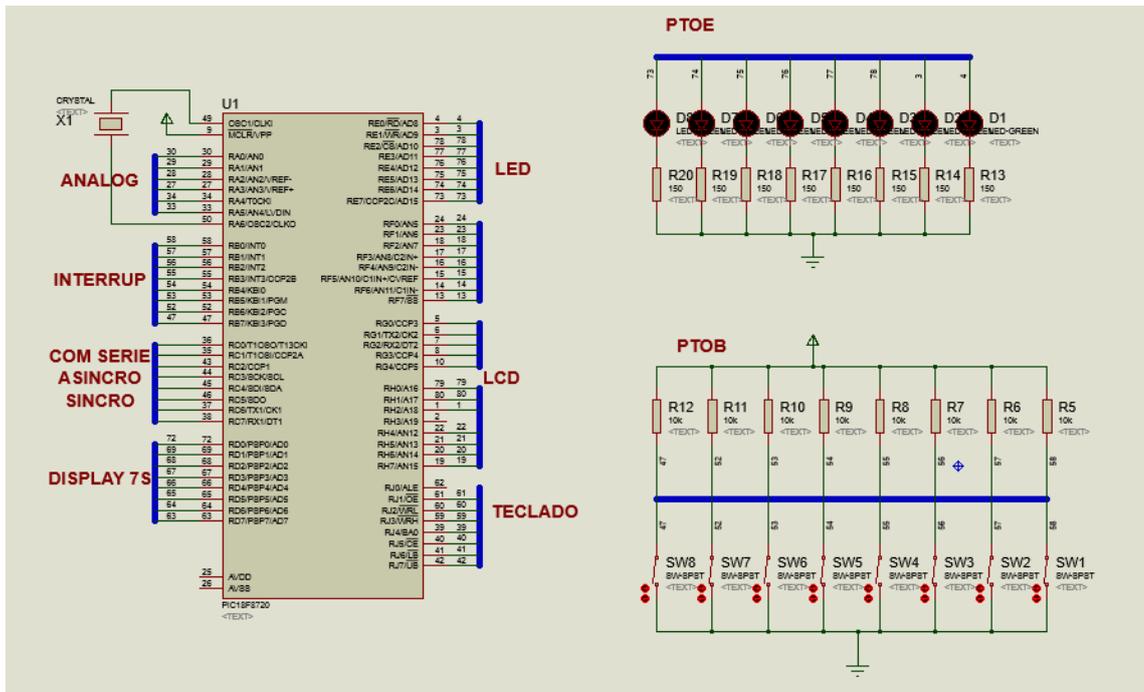


Figura 3. 15: Aplicación práctica #1 del sistema de entrenamiento de uC PIC18F8720.

Elaborado por: Autor.

Como se puede observar se utilizó el PUERTO E para conectar 8 diodos led con sus respectivas resistencias en serie de 150  $\Omega$  y el PUERTO B para conectar en sus pines 8 switch que servirán como elementos de entrada. El ingreso de datos trabaja con lógica negada, es decir al cerrar cada uno de los switch se ingresa un nivel de voltaje bajo, por lo que se requiere resistencias de pull up de 10 K $\Omega$  para su correcto funcionamiento.

Como ejemplo práctico para el circuito se realizará un programa capaz de convertir un código binario de 4 bits en su respectivo código GRAY. La tabla 3.1 muestra la equivalencia entre los números decimales binarios y los códigos GRAY.

Tabla 3. 1: Representación de numeraciones decimal a binaria y código Gray.

Decimal	Binario	Gray
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100
8	1000	1100
9	1001	1101
10	1010	1111
11	1011	1101
12	1100	1010
13	1101	1011
14	1110	1001
15	1111	1000

Elaborado por: Autor.

A continuación, se muestra las directivas a usarse en este programa, las cuales indican el microcontrolador a utilizar (PIC18F8720), la velocidad de reloj (20 MHz), y la declaración de los puertos con su respectiva dirección asignada en memoria RAM en los registros especiales.

```

#include <18f8720.h>
#fuses HS
#use delay(clock=20000000)
#byte portb=0xf81
#byte portd=0xf83
#byte porte=0xf84

```

En el programa principal se configura al puerto B como un puerto de salida, ya que es aquí donde se van a conectar los diodos led, y al puerto E como un puerto de entrada, debido a que los switch se conectarán aquí. Esto es posible con las siguientes instrucciones:

```

void main()
{
  set_tris_b(0xFF);
  set_tris_e(0x00);
  porte=0;
}

```

Para la conversión del número binario que se ingresa por el puerto B se utiliza la instrucción SWITCH, la cual hace la pregunta al puerto B sobre su estado actual para realizar la respectiva asignación del código GRAY al puerto de salida E, tal como se indica en las siguientes líneas de programación:

```

while(true)
{
  switch (portb) {
    case 0:porte=0; break;
    case 1:porte=1; break;
    case 2:porte=3; break;
    case 3:porte=2; break;
    case 4:porte=6; break;
    case 5:porte=7; break;
    case 6:porte=5; break;
    case 7:porte=4; break;
    case 8:porte=12; break;
    case 9:porte=13; break;
    case 10:porte=15; break;
    case 11:porte=14; break;
    case 12:porte=10; break;
    case 13:porte=11; break;
    case 14:porte=9; break;
    case 15:porte=8; break;
    default:porte=0;
             break; }
}

```

Una vez terminado el programa se procede a compilar el archivo .C, si no hubiese errores de sintaxis automáticamente se generan varios archivos, entre ellos el archivo hex, el cual debe ser ingresado al microcontrolador para su respectivo funcionamiento.

### 3.4. Aplicación práctica 2: contador ascendente de 0000 a 9999.

En la figura 3.16 se muestra un circuito esquemático que permite el manejo de cuatro displays de 7 segmentos ánodo común de manera multiplexada con el microcontrolador 18F8720, un decodificador 74LS47, cuatro transistores 2N3906 y cuatro resistencias de 4.7 KΩ.

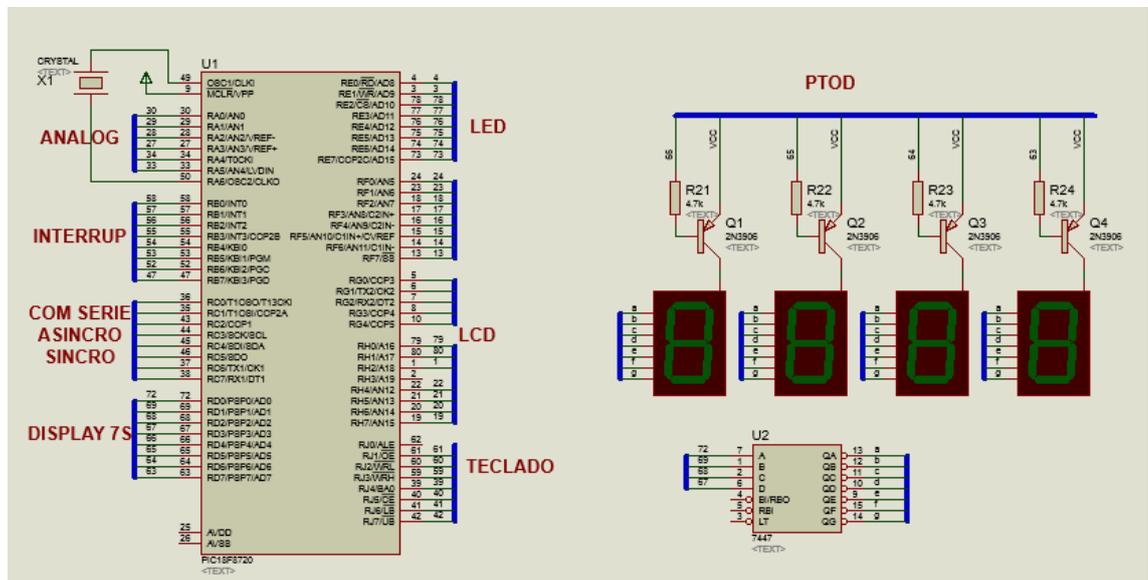


Figura 3. 16: Aplicación práctica #2 del sistema de entrenamiento de uC PIC18F8720.

Elaborado por: Autor.

Como ejemplo práctico para el circuito se realizará un programa capaz de contar desde 0000 hasta 9999 con intervalo de tiempo de 1 segundo. A continuación, se muestra las directivas a usarse en este programa, las cuales incluyen la librería del microcontrolador PIC18F8720, la velocidad de reloj (20 MHz), y la asignación del puerto a utilizar con su respectiva dirección asignada

en memoria RAM en los registros especiales, además se declara las dos funciones que van a ser invocadas desde el programa MAIN, y estas son; mostrar () y contar ().

```
#include <18f8720.h>
#fuses HS
#use delay(clock=20000000)
#byte portd=0xf83
void mostrar(void);
void contar(void);
```

Es necesario declarar e inicializar con valor de cero, cuatro variables que manejarán cada uno de los dígitos correspondiente al número que se presentará en los display de 7 segmentos.

```
int unidad=0;
int decena=0;
int centena=0;
int umil=0;
```

En el programa principal MAIN se debe programar al puerto D como salida y luego en un lazo infinito se debe llamar a las funciones que harán la visualización de los números en el display de 7 segmentos (mostrar) y a la que incrementa el número cada segundo (contar).

```
void main ()
{
  set_tris_d(0x00);
  While(true)
  {
    mostrar();
    contar();
  }
}
```

Las funciones mostrar y contar se detallan a continuación

```
void mostrar(void)
{
    portd=112+unidad;
    delay_ms(100);
    portd=176+decena;
    delay_ms(100);
    portd=208+centena;
    delay_ms(100);
    portd=224+umil;
    delay_ms(100);
}

void contar(void)
{
    unidad++;
    if (unidad>9){
        unidad=0;
        decena++;}
    if (decena>9){
        decena=0;
        centena++;}
    if (centena>9){
        centena=0;
        umil++;}
    if (umil>9){
        umil=0;
    }
}
```

### 3.5. Aplicación práctica 3: apertura de cerraduras eléctricas.

En la figura 3.17 se muestra un circuito esquemático de la aplicación práctica 3, con un microcontrolador PIC18F8720 que se encarga de controlar el manejo de un display LCD de 4x20 y un teclado matricial de 3x4. Además, se puede observar dos dispositivos de salida como son, el relé electromecánico y un buzzer para emitir señales audibles, estos dispositivos se encuentran acompañados por transistores 2N3904, resistencias y diodos rectificadores.

Como ejemplo práctico para el circuito se realizará un programa que permita el ingreso de una clave de 4 dígitos (1234), si el valor ingresado corresponde al de la clave se deberá activar un relé para la apertura de una cerradura eléctrica, caso contrario deberá mostrarse en el LCD un mensaje

de clave equivocada, si se llegase a introducir por tres veces consecutivas una clave equivocada el sistema se bloqueará por un tiempo determinado y se activará una señal audible como muestra de lo ocurrido.

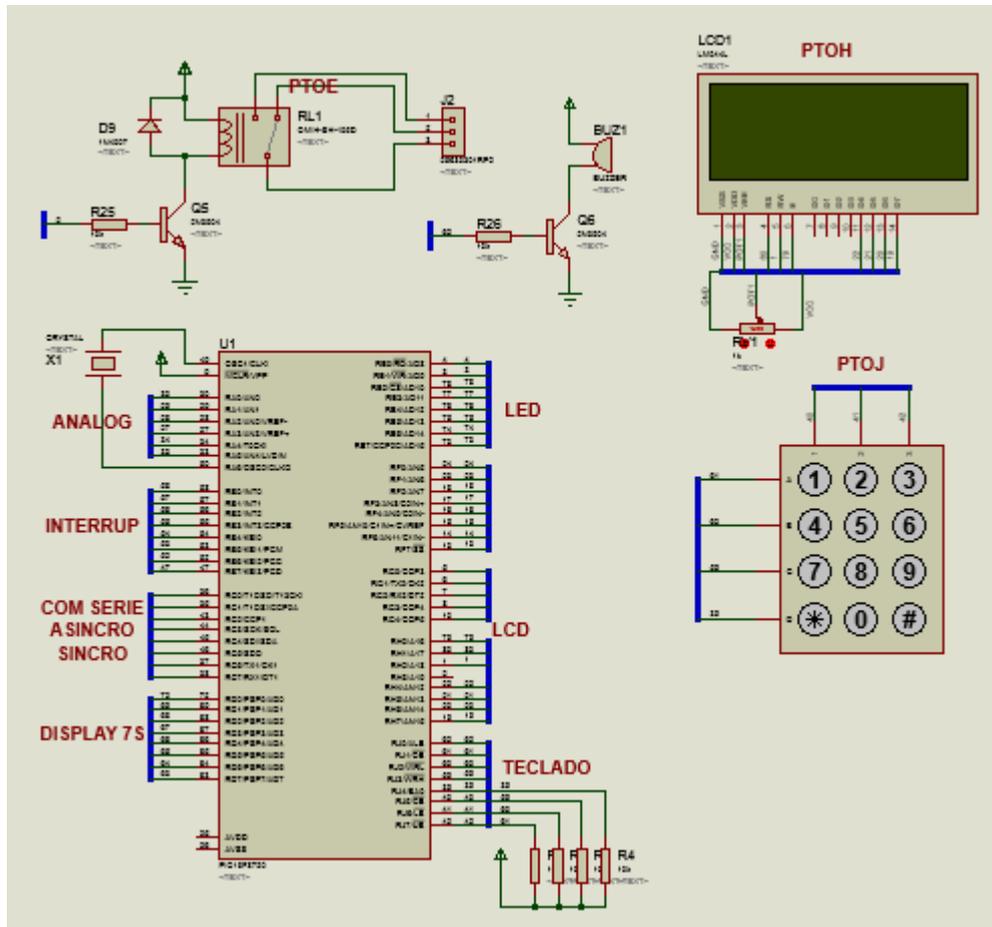


Figura 3. 17: Aplicación práctica #3 del sistema de entrenamiento de uC PIC18F8720.  
Elaborado por: Autor.

A continuación, se muestra las directivas que se utilizará en este programa, las cuales incluyen la librería del microcontrolador PIC18F8720, la velocidad de reloj (20 MHz), y la asignación del puerto a usar con su respectiva dirección asignada en memoria RAM en los registros especiales. También, se incluyen librerías para manejar LCD's y teclado matricial, para lo cual, se declara las tres funciones que serán invocadas, estas son: ingreso (), compara () y bloqueo ().

```

#include <18F8720.h>
#fuses HS
#use delay(clock=20000000)
#include <KBD2.c>
#include <LCD420_E.c>
#byte PORTH=0xF87

void ingreso(void);
void compara(void);
void bloqueo(void);

```

Es necesario declarar, cuatro variables globales que se necesitarán en el programa, que son:

```

int16 valor;
int error;
int16 clave=1234;
int k;

```

En el programa principal MAIN se debe programar a los puertos que estarán involucrados en el proyecto, esto es el puerto H como salida para conectar el LCD, el puerto J combinado entre entrada y salida para el teclado, acto seguido se debe inicializarlas librerías del LCD y del Teclado para luego en un lazo infinito llamar a las funciones de ingreso de clave y la de comparación del número ingresado con la clave.

```

void main()
{
    set_tris_j(0x1E);
    set_tris_h(0x00);
    lcd_init();
    kbd_init();
    while(true)
    {
        ingreso();
        compara();
    }
}

```

La función de ingreso(), permite al usuario ingresar la clave a través del teclado matricial, al finalizar la clave se debe validar con la tecla #

```
void ingreso(void)
{
  lcd_putc("\fIngrese clave ");
  lcd_gotoxy(1,2);
  k=0;
  valor=0;
  while(k!=-13)
  {
    k=0;
    while(k==0)
    k=kbd_getc();
    k=k-48;
    if(k!=-13)
    {
      printf(lcd_putc,"%D",k);
      valor=(valor*10)+k;
    }
  }
}
```

La función compara () chequea que la clave ingresada por el usuario sea la misma que el sistema reconoce como válida, esto es 1234

```
void compara(void)
{
  if (valor==clave)
  {
    output_high(PIN_E0);
    lcd_putc("\fPuerta abierta");
    delay_ms(1000);
    output_low(PIN_E0);
  } else{
    lcd_putc("\fclave incorrecta");
    error++;
  }
  if(error>2)
    bloqueo();
}
```

Por último, la función bloqueo (), muestra un mensaje en el LCD y activa una salida audible por el PIN\_J0 como muestra de lo sucedido

```
void bloqueo(void)
{
  int y;
  output_high(PIN_J0);
  for(y=0; y<=100; y++)
  {
    lcd_putc("\fSist Bloqueado");
    delay_ms(500);
    lcd_putc("\f");
    delay_ms(500);
  }
  output_low(PIN_J0);
  error=0;
}
```

### 3.6. Aplicación práctica 4: manejo de GLCD y teclado matricial.

En la figura 3.18 se muestra un circuito esquemático de la aplicación práctica 4, que utiliza el microcontrolador PIC18F8720 que se va encargar de controlar el manejo de un display GLCD de 4x20 y un teclado matricial de 3x4. Además, se puede observar dos dispositivos de salida como son, el relé electromecánico y un buzzer para emitir señales audibles, estos dispositivos se encuentran acompañados por transistores 2N3904, resistencias y diodos rectificadores.

Como ejemplo práctico para el circuito se realizará un programa que permita el ingreso de un voltaje analógico de 0 V hasta 5 V. El ingreso se lo hará mediante un potenciómetro, este valor ingresado debe visualizarse en el GLCD mediante una barra horizontal en la parte superior del GLCD.

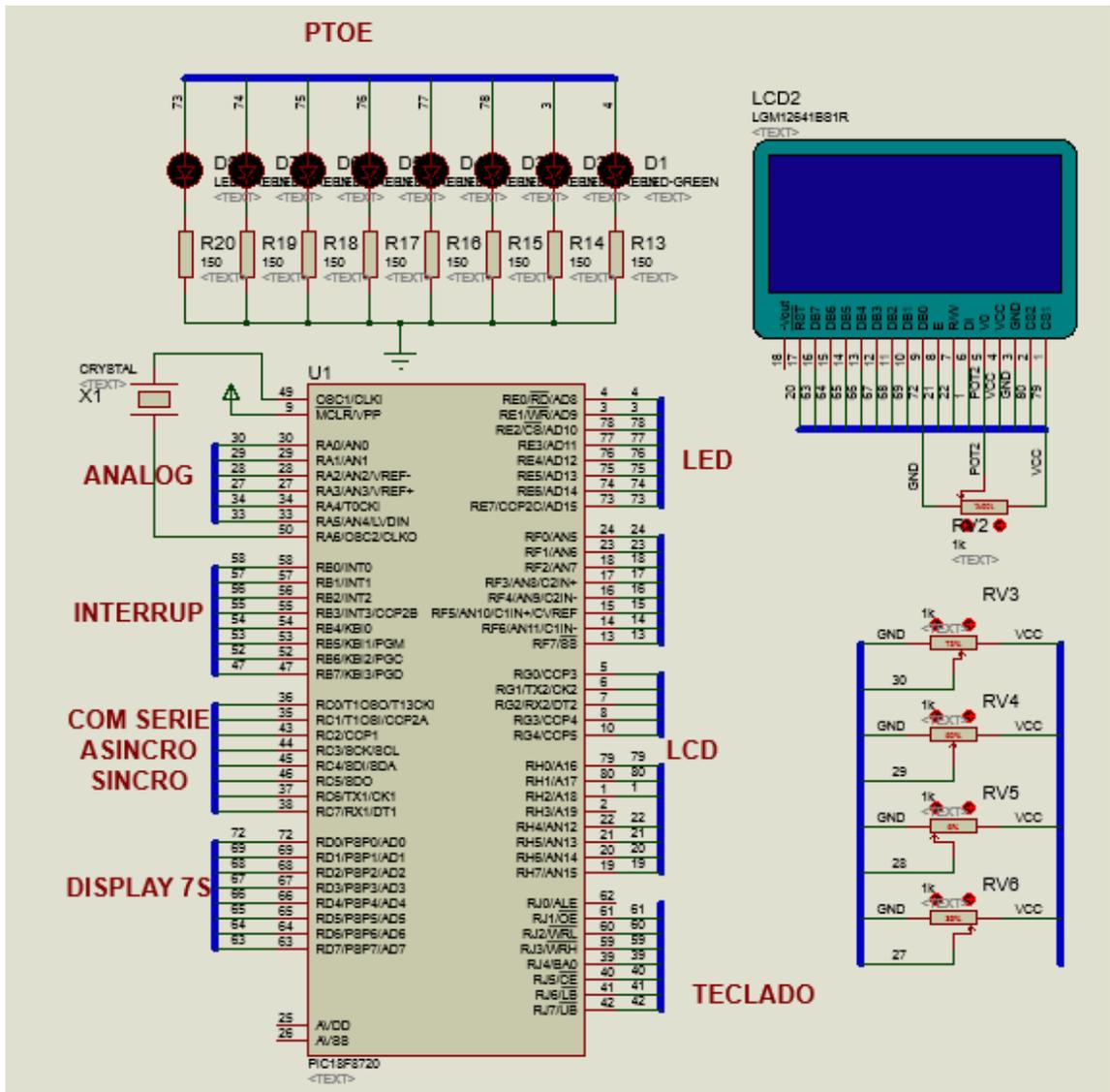


Figura 3. 18: Aplicación práctica #4 del sistema de entrenamiento de uC PIC18F8720.

Elaborado por: Autor.

Adicionalmente se debe activar una salida cada vez que se supere en la entrada un valor de 3.2 V, de igual manera se debe presentar mensajes de LUZ ACTIVADA/DESACTIVADA dependiendo si se supera o no ese voltaje. A continuación, se muestra las directivas necesarias, las cuales incluyen la librería del PIC18F8720, la velocidad de reloj (20 MHz), y la asignación del puerto a usar con su respectiva dirección asignada en memoria RAM en los registros especiales, además se incluye la librería para manejar al GLCD.

```

#include <18f8720.h>
#DEVICE ADC=10
#fuses HS
#use delay(clock=20Mhz)
#include <GLCD_2.C>
#byte portd=0xf83
#byte porte=0xf84
#byte porth=0xf87

```

Es necesario declarar variables globales que se necesitarán en el programa, así como variables tipo char con los mensajes que deberán aparecer en el transcurso del programa.

```

float ri1,ri2;
float dato1;
char men1[ ]="LUZ ACTIVADA ",men2[ ]="LUZ DESACTIVADA",men3[ ]="3.4V";
char autor[ ]="Larenas";

```

En el programa principal MAIN se debe programar a los puertos que estarán involucrados en el proyecto, esto es el puerto E como salida para conectar los LED que se deben encender y el puerto h como salida para el GLCD, acto seguido se debe inicializar la librería del GLCD, configurar al convertidor ADC del micro con el reloj interno, setea 2 entradas analógicas desde AN0 hasta AN1, y presentar en el GLCD un mensaje con el nombre del autor

```

void main()
{
set_tris_e(0x00);
set_tris_h(0x00);
glcd_init(on);
setup_adc(ADC_clock_internal);
setup_adc_ports(AN0_TO_AN1,VSS_VREF);
delay_ms(100);
glcd_text57(5,45,autor,1,1);
glcd_rect(86,9,87,12,1,1);
glcd_text57(86,0,men3,1,1);

```

Mediante un lazo infinito se debe seleccionar el canal analógico que se desea medir en este caso el canal 0, luego se debe hacer una conversión del valor medido que puede ser de 0 a 1023 a un valor de voltaje de 0 a 5v. Con este valor se debe graficar de manera proporcional una barra horizontal en el GLCD, si el valor de voltaje llega a superar los 3.2v aparece un mensaje que dice LUZ ACTIVADA caso contrario aparece otro mensaje de LUZ DESACTIVADA. Además, si el voltaje supera ese valor se debe activar una salida en el PIN\_E0

```
while(true)
{
    set_adc_channel(0);
    dato1=read_adc();
    ri1=(127*dato1)/1023;
    if(ri2!=ri1)
    {
        glcd_rect(0,15,127,20,1,0);
        glcd_rect(0,15,ri1,20,1,1);
        ri2=ri1;
        if(ri1>86){
            glcd_text57(5,27,men2,1,0);
            glcd_text57(5,27,men1,1,1);
            output_high(PIN_E0);}
        else{
            glcd_text57(5,27,men1,1,0);
            glcd_text57(5,27,men2,1,1);
            output_low(PIN_E0);}
    }
}
```

## **CAPÍTULO 4: CONCLUSIONES Y RECOMENDACIONES.**

### **4.1. Conclusiones.**

- El impacto inmediato del trabajo de titulación es el desarrollo de la experiencia de diseñar una tarjeta entrenadora de microcontroladores PIC18F8720. Fue una experiencia de aprendizaje muy exigente, el mismo se determinó a partir de varias conversaciones con el tutor y docente de la Facultad de Educación Técnica para el Desarrollo.
- Durante el período de diseño y desarrollo de 6 meses en la materia de actualización de conocimiento 1 (Microcontroladores) dictada por el tutor, se logró realizar las aplicaciones prácticas o circuitos de prueba, así como el ensamblaje de PCB, organización de componentes electrónicos, evaluación y prueba del sistema y realización de instrumentos para evaluar la efectividad de este sistema de entrenamiento.
- El inconveniente del presente trabajo fue el presupuesto; siempre hay falta de fondos para hacer el trabajo. Por ejemplo, el presupuesto del sistema se establece en alrededor de \$ 170 y la lista de materiales por un total de \$ 224.93. Esto no incluye el costo de montaje. Esto hace que el presupuesto supere en más del 25%. Pero al realizar la búsqueda ampliada del proveedor de partes, se puede mejorar el precio por volumen, los costos de la lista de materiales pueden reducirse entre un 15% y 20%. Mientras, que con la ayuda de los estudiantes se realizaría el montaje y al mismo tiempo enseñar el proceso de fabricación, control de calidad y solución de problemas que

pueden reducir el costo de mano de obra del montaje del sistema. También proporciona a los estudiantes experiencias de capacitación muy prácticas para sus futuros lugares de trabajo. Estos enfoques hacen que el sistema de \$ 170 sea una realidad más cercana.

- El objetivo principal fue proporcionar una herramienta didáctica que permita realizar prácticas de laboratorio, utilizando el sistema de entrenamiento (hardware) y del software para los estudiantes que no tienen acceso a la capacitación en microcontroladores que se requiere para muchos trabajos de alta tecnología.
- Las actividades individuales de laboratorio también permitirán reforzar el aprendizaje y el desarrollo de habilidades de los estudiantes en conceptos de programación, así como proporcionar una plataforma para la investigación y el desarrollo individual de los estudiantes una vez finalizada la asignatura de microcontroladores. Los resultados esperados serán estudiantes mejor entrenados/educados que calificarán para puestos en la fuerza de trabajo basada en el conocimiento.

#### **4.2. Recomendaciones.**

- Desarrollar sistemas de entrenamiento para prácticas de microcontroladores AVR utilizando el entorno de programación de Arduino.
- Desarrollar una interfaz web de simulación de microcontroladores para capacitación de educación continua a distancia.

## Bibliografía

- Abalos, H. (2014). Entrenador de Microcontroladores PICAXE. Recuperado el 11 de enero de 2018, a partir de <https://es.scribd.com/document/240136245/Entrenador-de-Microcontroladores-PICAXE-pdf>
- Ajao, L. A., Agajo, J., Kolo, J. G., Adegboye, M. A., & Yusuf, Y. (2016). Learning of Embedded System Design, Simulation and Implementation: A Technical Approach. *American Journal of Embedded Systems and Applications*, 3(3), 35. <https://doi.org/10.11648/j.ajes.20150303.12>
- Ajao, L. A., Olaniyi, O. M., Kolo, J. G., & Ajao, A. O. (2015). Project-Based Microcontroller System Laboratory Using BK300 Development Board With PIC16F887 Chip. *International Journal of Embedded Systems and Applications*, 5(3), 15–28. <https://doi.org/10.5121/ijesa.2015.5302>
- Albán Q., D. M., & Peña D., G. A. (2015). Evaluación del sistema de entrenamiento PIC18F4550 mediante aplicaciones prácticas para la asignatura de microcontroladores. (Trabajo de Titulación). Universidad Católica de Santiago de Guayaquil, Guayaquil. Recuperado a partir de <http://repositorio.ucsg.edu.ec/handle/3317/3808>
- Allam, O. (2006). Design and Implementation of a 32-bit RISC Microprocessor (Master). University of Applied Sciences Darmstadt. Recuperado a partir de [https://www.dte.us.es/docencia/master/micr/dapa/Lab4\\_micro6\\_v2.pdf](https://www.dte.us.es/docencia/master/micr/dapa/Lab4_micro6_v2.pdf)
- Barsoum, N. (2010). Speed control of the induction drive by temperature and light sensors via PIC. *Global Journal on Technology & Optimization (GJTO)*, 1, 53–59.
- Ibrahim, D. (2014). PIC microcontroller projects in C: basic to advanced (Second edition). Kidlington, Oxford, UK: Newnes.

- Ibrahim, D. (2015). PIC32 microcontrollers and the diligent chipKIT: introductory to advanced projects (1. ed). Amsterdam: Elsevier Newnes.
- Mashhadany, Y. I. A. (2012). Design and Implementation of Electronic Control Trainer with PIC Microcontroller. *Intelligent Control and Automation*, 3(3), 222. <https://doi.org/10.4236/ica.2012.33025>
- Microchip. (2013). PIC18F6520/8520/6620/8620/6720/8720 Datasheet. Microchip. Recuperado a partir de <http://ww1.microchip.com/downloads/en/DeviceDoc/39609C.pdf>
- Somantri, Y., & Fushshilat, I. (2017). Low Cost, Advanced, Integrated Microcontroller Training Kit. *IOP Conference Series: Materials Science and Engineering*, 180, 12107. <https://doi.org/10.1088/1757-899X/180/1/012107>
- Technical Editor. (2017, abril 26). Difference between Harvard architecture and von-Neumann architecture. Recuperado el 11 de febrero de 2018, a partir de <http://www.polytechnichub.com/difference-harvard-architecture-von-neumann-architecture/>
- Uttarkar, N. K., & Kanchi, R. R. (2014). Design and Development of a Low-Cost Embedded System Laboratory Using TI MSP430 LaunchPad. *American Journal of Embedded Systems and Applications*, 1(2), 37. <https://doi.org/10.11648/j.ajes.20130102.12>
- Vassiliev, A. E., Ivanova, T. Y., Tapia, D. F. C., & Luong, Q. T. (2016). Microcontroller-based embedded system equipment development for research and educational support (pp. 219–223). IEEE. <https://doi.org/10.1109/ICIMTech.2016.7930333>



Presidencia  
de la República  
del Ecuador



Plan Nacional  
de Ciencia, Tecnología,  
Innovación y Saberes



SENESCYT  
Secretaría Nacional de Educación Superior,  
Ciencia, Tecnología e Innovación

## DECLARACIÓN Y AUTORIZACIÓN

Yo, **Larenas Marín, David Daniel** con C.C: # 120747333-9 autor del Trabajo de Titulación: **Desarrollo de una tarjeta de entrenamiento para prácticas de microcontroladores utilizando el dispositivo PIC 18F8720** previo a la obtención del título de **INGENIERO ELECTRÓNICO EN CONTROL Y AUTOMATISMO** en la Universidad Católica de Santiago de Guayaquil.

1.- Declaro tener pleno conocimiento de la obligación que tienen las instituciones de educación superior, de conformidad con el Artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de titulación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizo a la SENESCYT a tener una copia del referido trabajo de titulación, con el propósito de generar un repositorio que democratice la información, respetando las políticas de propiedad intelectual vigentes.

Guayaquil, 12 de Marzo de 2018

f. \_\_\_\_\_

Nombre: Larenas Marín, David Daniel

C.C: 120747333-9

## **REPOSITORIO NACIONAL EN CIENCIA Y TECNOLOGÍA**

### **FICHA DE REGISTRO DE TESIS/TRABAJO DE TITULACIÓN**

<b>TÍTULO Y SUBTÍTULO:</b>	Desarrollo de una tarjeta de entrenamiento para prácticas de microcontroladores utilizando el dispositivo PIC 18F8720		
<b>AUTOR(ES)</b>	LARENAS MARÍN, DAVID DANIEL		
<b>REVISOR(ES)/TUTOR(ES)</b>	M. Sc. CÓRDOVA RIVADENEIRA, LUIS SILVIO		
<b>INSTITUCIÓN:</b>	Universidad Católica de Santiago de Guayaquil		
<b>FACULTAD:</b>	Facultad de Educación Técnica para el Desarrollo		
<b>CARRERA:</b>	Ingeniería Electrónica en Control y Automatismo		
<b>TÍTULO OBTENIDO:</b>	Ingeniero Electrónico en Control y Automatismo		
<b>FECHA DE PUBLICACIÓN:</b>	12 de Marzo de 2018	<b>No. DE PÁGINAS:</b>	81
<b>ÁREAS TEMÁTICAS:</b>	Microcontroladores, Diseño Electrónico y Sistemas Digitales		
<b>PALABRAS CLAVES/ KEYWORDS:</b>	MICROCONTROLADORES, PROTEUS, CCS C, ARQUITECTURA, PIC, SIMULACIÓN		
<b>RESUMEN/ABSTRACT (150-250 palabras):</b>			
<p>El presente trabajo de titulación consistió en el desarrollo de una tarjeta de entrenamiento para prácticas de microcontroladores utilizando el dispositivo PIC 18F8720. En el capítulo 1 se describe los antecedentes, definición y justificación del problema de investigación. También se especifica el objetivo general, objetivos específicos, hipótesis y metodología de investigación utilizada en el presente proyecto de titulación. En el capítulo 2, se realiza la descripción de los fundamentos teóricos de los microcontroladores, así como sus respectivas arquitecturas. Se incluye una descripción y análisis de las prestaciones de cada uno de los sistemas de entrenamiento de microcontroladores disponibles para la enseñanza en carreras de ingenierías. En el capítulo 3, se realiza el diseño del sistema de entrenamiento de microcontroladores utilizando el dispositivo PIC 18F8720, así como también, se incluyen en el trabajo de titulación tres aplicaciones prácticas desarrolladas a nivel de simulación ISIS Proteus, programación de alto nivel C e implementación para la validación de la correcta funcionalidad de la plataforma de entrenamiento.</p>			
<b>ADJUNTO PDF:</b>	<input checked="" type="checkbox"/> SI	<input type="checkbox"/> NO	
<b>CONTACTO CON AUTOR/ES:</b>	<b>Teléfono:</b> +593991592741	<b>E-mail:</b> <a href="mailto:danny_lm15@hotmail.es">danny_lm15@hotmail.es</a>	
<b>CONTACTO CON LA INSTITUCIÓN: COORDINADOR DEL PROCESO DE UTE</b>	<b>Nombre:</b> Mendoza Merchán, Eduardo Vicente		
	<b>Teléfono:</b> +593-9-68366762		
	<b>E-mail:</b> <a href="mailto:eduardo.mendoza@cu.ucsg.edu.ec">eduardo.mendoza@cu.ucsg.edu.ec</a>		
<b>SECCIÓN PARA USO DE BIBLIOTECA</b>			
<b>Nº. DE REGISTRO (en base a datos):</b>			
<b>Nº. DE CLASIFICACIÓN:</b>			
<b>DIRECCIÓN URL (tesis en la web):</b>			