



**UNIVERSIDAD ESTATAL
PENÍNSULA DE SANTA ELENA
FACULTAD DE SISTEMAS Y TELECOMUNICACIONES
CARRERA DE ELECTRÓNICA Y
TELECOMUNICACIONES**

TRABAJO DE TITULACIÓN

Propuesta Tecnológica, previo a la obtención del título de:

**INGENIERO EN ELECTRÓNICA Y
TELECOMUNICACIONES**

**“APLICACIONES DE MICROCONTROLADORES PARA
INTERACCIÓN ENTRE ACTUADORES Y SENSORES
UTILIZANDO UN MÓDULO EASYPIC V8”**

AUTOR: CRISTIAN GEOVANNY MARCATOMA PALTA

TUTOR: ING. SENDEY VERA GONZÁLEZ, MACI

**LA LIBERTAD – ECUADOR
2020**

AGRADECIMIENTO

El objetivo que me he planteado desde la preparatoria “ser un profesional” se está cumpliendo. El camino que me toco recorrer fue duro, pero gracias a mi Dios por darme la inteligencia y fortaleza que he necesitado para seguir de pie y llegar hasta donde ahora estoy.

Agradezco profundamente a mis padres, a mis hermanos y a mi esposa por la motivación, el apoyo y el sacrificio que hicieron al permitirme seguir adelante en mis estudios.

A mis amigos y amigas que de una u otra forma estaban ahí para brindarme una voz de aliento, haciéndome sentir que, sí se podía. “Gracias por formar parte de este camino”.

Al personal docente de la Universidad Estatal Península de Santa Elena, y aquellos los que me brindaron su amistad, paciencia, y sus valiosos conocimientos impartidos en este proyecto de titulación.

Termino agradeciendo a todas las personas que rodean mi vida, sus palabras y consejos fueron muy valiosos para ser la persona quien soy, los estaré siempre agradecido.

Cristian Geovanny Marcatoma Palta

DEDICATORIA

Dedico el presente trabajo a mis padres y hermanos; ya que son la razón principal de que yo pueda culminar esta etapa, por su ayuda incondicional, sus consejos y todo el esfuerzo y confianza que depositaron en mí. Les estaré eternamente agradecido por todo lo que hicieron por mí.

A mi esposa y en especial a mi hija Victoria, siendo ellas la fuente de inspiración para llevar a cabo este documento. En un futuro muy próximo los veré lograr sus objetivos, hoy me tocó a mí, estoy seguro ustedes también lo van a lograr.

Cristian Geovanny Marcatoma Palta

APROBACIÓN DEL TUTOR

En mi calidad de Tutor del trabajo de titulación denominado “**Aplicaciones de Microcontroladores para interacción entre actuadores y sensores utilizando un módulo EASYPIC v8**” elaborado por el egresado **Cristian Geovanny Marcatoma Palta**, de la carrera de Electrónica y Telecomunicaciones de la Universidad Estatal Península de Santa Elena, me permito declarar que luego de haber orientado, estudiado y revisado, lo apruebo en todas sus partes y autorizo al estudiante para que inicie los trámites legales correspondientes.

La Libertad, Febrero del 2020

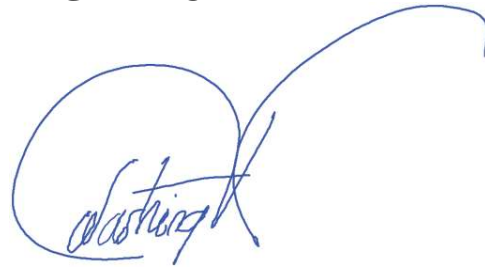
A handwritten signature in blue ink, appearing to read 'SENDY VERA G', with a large circular flourish at the end.

Ing. Sendey Vera González, MACI

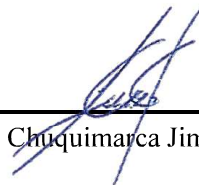
TRIBUNAL DE GRADO



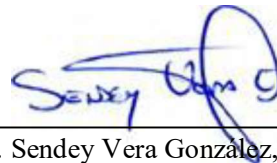
Ing. Freddy Villao Santos
**Decano de la facultad de Sistemas
Telecomunicaciones**



Washington Torres Güin
**Director de la Carrera de Electrónica
y Telecomunicaciones**



Ing. Luis Chuquimarca Jiménez, MGT
Profesor del área



Ing. Sendency Vera González, MACI
Tutor



Ab. Víctor Coronel Ortíz, MSc.
Secretario General

UNIVERSIDAD ESTATAL PENÍNSULA DE SANTA ELENA
FACULTAD DE SISTEMAS Y TELECOMUNICACIONES
CARRERA ELECTRÓNICA Y TELECOMUNICACIONES

“Aplicaciones de microcontroladores para interacción entre actuadores y sensores utilizando un módulo EasyPic V8”

Autor: CRISTIAN GEOVANNY MARCATOMA PALTA

Tutor: SENDEY VERA GONZÁLEZ

RESUMEN

La Universidad Estatal Península de Santa Elena cuenta con la carrera de Electrónica y Automatización que hizo su apertura en el periodo 2018-1, se hace mención a esta carrera debido a que este proyecto va dirigido a la asignatura de Microcontroladores la cual se brinda en la carrera antes mencionada. En la presente asignatura es necesario realizar ejercicios prácticos para tener en claro la teoría, no basta con solo simular el circuito ya que no es factible al momento de la implementación. Es por eso que la siguiente propuesta se basa en desarrollar aplicaciones de microcontroladores que manipulen actuadores, sensores e indicadores empleando un módulo de última generación EASYPIC v8 fabricado por la empresa MIKROE, que permite la interacción de sus diversos periféricos que lo componen, siendo esto un material de apoyo para que el estudiante refuerce sus habilidades en el análisis teórico y práctico de los microcontroladores. Este incluye una guía de aplicaciones prácticas donde se emplean dos lenguajes de programación y varios softwares entre ellos MPLAB, MikroC, MATLAB, Proteus, además interfaces de usuario donde se exponen los resultados de las cuatro últimas prácticas planteadas.

ABSTRACT

The State University of the Peninsula of Santa Elena has a degree in Electronics and Automation that opened in the period 2018-1, it is made specific to this career because this project is aimed at the subject of Microcontrollers which is offered in the aforementioned race. In this subject it is necessary to carry out practical exercises to be clear about the theory, it is not enough to just simulate the circuit and that it is not feasible at the time of implementation. That is why the following proposal is based on developing microcontroller applications that manipulate actuators, sensors and indicators using a state-of-the-art EASYPIC v8 module manufactured by the MIKROE company, which allows the interaction of its various peripherals that compose it, this being a Support material for the student reinforces his skills in the theoretical and practical analysis of microcontrollers. This includes a guide of practical applications where two programming languages are used and several softwares including MPLAB, MikroC, MATLAB, Proteus, as well as user interfaces where the results of the last four practical practices are presented.

DECLARACIÓN

El contenido del presente trabajo de Graduación es de mi responsabilidad; el patrimonio intelectual del mismo pertenece a la Universidad Estatal Península de Santa Elena.



Cristian Geovanny Marcatoma Palta

ÍNDICE GENERAL

AGRADECIMIENTO	I
DEDICATORIA	II
APROBACIÓN DEL TUTOR	III
TRIBUNAL DE GRADO	IV
RESUMEN	V
ABSTRACT	VI
DECLARACIÓN	VII
ÍNDICE GENERAL	VIII
ÍNDICE DE FIGURAS	XI
ÍNDICE DE TABLAS	XIV
INTRODUCCIÓN	1
CAPITULO I	2
GENERALIDADES DE LA PROPUESTA	2
1.1 ANTECEDENTES	2
1.2 DESCRIPCIÓN DEL PROYECTO	4
1.3 OBJETIVOS DEL PROYECTO	5
1.3.1 OBJETIVO GENERAL	5
1.3.2 OBJETIVOS ESPECÍFICOS	5
1.4 RESULTADOS ESPERADOS	5
1.5 JUSTIFICACIÓN DEL TEMA	6
1.6 ALCANCE DEL PROYECTO	7
1.7 METODOLOGÍA	7
TIPOS DE INVESTIGACIÓN	7
1.7.1 INVESTIGACIÓN EXPLORATORIA	7
1.7.2 INVESTIGACIÓN EXPERIMENTAL	8
1.7.3 INVESTIGACIÓN BIBLIOGRÁFICA	8
CAPITULO II	9
PROPUESTA TECNOLÓGICA	9
2.1 MARCO CONTEXTUAL	9

2.2	MARCO CONCEPTUAL	10
2.2.1	TARJETAS DE DESARROLLO	10
2.2.2	CARACTERISTICAS DE LA PLACA EASYPIC v8	11
2.2.2.1	PUERTOS DE ALIMENTACIÓN DE ENTRADA.....	12
2.2.2.2	SISTEMA DE ALIMENTACIÓN.....	14
2.2.2.3	BOTON DE ENCENDIDO Y DE REINICIO.....	14
2.2.2.4	ZÓCALOS DIP PARA MICROCONTROLADORES	15
2.2.2.5	OSCILADOR DE CRISTAL	16
2.2.2.6	SECCIÓN DE ENTRADA Y SALIDA	17
2.2.2.7	CONECTOR DE PANTALLA GRÁFICA 1x20	22
2.2.2.8	CONECTOR PARA PANTALLA DE CARACTERES 2x16	24
2.2.2.9	ZÓCALOS MIKROBUS.....	26
2.2.2.10	COMUNICACIÓN.....	27
2.2.3	MICROCONTROLADORES PIC	31
2.2.4	SENSORES	32
2.2.5	ACTUADORES	32
2.2.6	INDICADORES.....	33
2.2.7	CONVERSIÓN ANALÓGICA DIGITAL	34
2.2.8	PROTOCOLO DE COMUNICACIÓN.....	34
2.3	MARCO TEÓRICO	36
	CAPÍTULO III	38
	DESARROLLO DE LA PROPUESTA	38
3.1	COMPONENTES DE LA PROPUESTA	38
3.1.1	COMPONENTES LÓGICOS	38
3.1.1.1	MIKROC PRO FOR PIC.....	38
3.1.1.2	MPLAB IDE	39
3.1.1.3	PROTEUS.....	40
3.1.1.4	MATLAB	40
3.1.1.5	CODEGRIP	41
3.1.2	COMPONENTES FÍSICOS	42
3.1.2.1	MÓDULO EASYPIC v8.....	42

3.1.2.2	MICROCONTROLADOR PIC 16F887	44
3.1.2.3	SENSOR SHARP 2Y0A21	45
3.1.2.4	SENSOR LM35	46
3.1.2.5	MOTOR DC	48
3.1.2.6	CONTROLADOR L298N	49
3.1.2.7	LCD 2X16	51
3.1.2.8	DISPLAY DE 7 SEGMENTOS	52
3.1.2.9	RESISTENCIAS	54
3.2	GUÍA DE APLICACIONES PRÁCTICAS	54
3.2.1	PRÁCTICA 1:	55
3.2.2	PRÁCTICA 2:	58
3.2.3	PRÁCTICA 3:	61
3.2.4	PRACTICA 4:	63
3.2.5	PRÁCTICA 5:	65
3.2.6	PRÁCTICA 6:	67
3.2.7	PRÁCTICA 7:	69
3.3	PRESUPUESTO DE LA PROPUESTA	71
	RESULTADOS	73
	CONCLUSIONES	95
	RECOMENDACIONES	96
	BIBLIOGRAFÍA	97
	ANEXOS	98
	ANEXO A: GUÍA DE APLICACIONES PRÁCTICAS	98
	ANEXO B:	99

ÍNDICE DE FIGURAS

Ilustración 1 Placa de desarrollo EasyPIC V8	11
Ilustración 2 Puertos de alimentación de entrada hacia la placa EasyPIC v8	12
Ilustración 3 Fuente de alimentación regulada y filtrada	14
Ilustración 4 Botón POWER y RESET de la placa EasyPIC v8.....	15
Ilustración 5 Indica los 8 zócalos de la tarjeta EasyPIC v8	16
Ilustración 6 Socket para oscilador de cristal.....	16
Ilustración 7 Interruptor DIP para OSC1	17
Ilustración 8 Interruptor DIP para OSC2	17
Ilustración 9 Esquema de conexión de los puertos de E/S.....	18
Ilustración 10 Puerto A del MCU con Pulsadores	18
Ilustración 11 Switch para controlar los puertos de los pulsadores	19
Ilustración 12 Switch de 8 polos (UP-PULL-DOWN) del puerto A	19
Ilustración 13 Indicadores LEDS conectado al puerto B de la placa EasyPIC v8.....	20
Ilustración 14 DIP de 10 polos para habilitar los puertos de los LEDS.....	21
Ilustración 15 Conectores macho 2x5 del puerto A	22
Ilustración 16 Socket para GLCD en EasyPIC v8	22
Ilustración 17 Interruptor (SW6) para GLCD e interruptor (SW2) panel táctil.....	23
Ilustración 18 Socket para LCD (esquema electrónico)	24
Ilustración 19 Interruptor DIP (SW6) para LCD	25
Ilustración 20 Zócalos mikroBUS en placa EasyPIC v8	26
Ilustración 21 Esquema electrónico del mikroBUS1	26
Ilustración 22 Bloque de comunicación USB-UART.....	27
Ilustración 23 Bloque de comunicación CAN en la placa EasyPIC v8	28
Ilustración 24 Bloque USB(DISPOSITIVO) en EasyPIC v8	30
Ilustración 25 Encapsulado de un MCU PIC de 40 pines.....	31
Ilustración 26 Actuador Eléctrico	33
Ilustración 27 Dispositivos indicadores	33
Ilustración 28 Proceso de conversión Analógica a Digital	34

Ilustración 29 Entorno del software MikroC versión 7.6.....	38
Ilustración 30 Software MPLAB IDE.....	39
Ilustración 31 Software Proteus	40
Ilustración 32 Software MATLAB	41
Ilustración 33 Software CODEGRIP	42
Ilustración 34 EasyPIC v8.....	43
Ilustración 35 MCU PIC 16F887	45
Ilustración 36 Sensor SHARP 2Y02A1 (10-80cm) [15]	45
Ilustración 37 Características del sensor Voltaje vs Distancia [15].....	46
Ilustración 38 Sensor LM35.....	46
Ilustración 39 Circuito básico de conexión.....	47
Ilustración 40 Motorreductor 75:1 HP 6V	48
Ilustración 41 Características técnicas del motorreductor	48
Ilustración 42 L298N controlador de motores DC.....	49
Ilustración 43 Bloque de conexiones L298N y MCU.....	50
Ilustración 44 Pantalla de Cristal Líquido LCD 2x16.....	51
Ilustración 45 Display de siete segmentos	53
Ilustración 46 Configuración Ánodo y Cátodo Común	53
Ilustración 47 Control de un depósito de líquidos	60
Ilustración 48 Estacionamiento de vehículos.....	62
Ilustración 49 Práctica #4 esquema representativo de conexiones	64
Ilustración 50 Práctica #5 esquema representativo de conexiones	66
Ilustración 51 Practica #7 esquema representativo de conexiones	70
Ilustración 52 Habilitando los LEDs del puerto D y los botones del puerto A.....	74
Ilustración 53 Introduciendo un 1 o un 0 lógico por el switch del puerto A.....	74
Ilustración 54 Manipulando los botones del puerto A	75
Ilustración 55 Cuando ninguno sensor detecta la presencia de líquido	76
Ilustración 56 Cuando el sensor "SV" detecta el liquido	77
Ilustración 57 Cuando el líquido a llegado al sensor "SLL"	77
Ilustración 58 los tres sensores detectan el agua.....	78

Ilustración 59 Conexión entre EasyPIC y prototipo realizado en baquelita	79
Ilustración 60 Configuración del Board Setup para la práctica #3	80
Ilustración 61 Primer vehículo registrado	80
Ilustración 62 Registrando el vehículo #9.....	81
Ilustración 63 Led RB1 activo, indicando que el estacionamiento está lleno.....	81
Ilustración 64 Interfaz de usuario práctica #4, manipulación de puertos.....	82
Ilustración 65 Manipulando el puerto A desde la interfaz de MATLAB	83
Ilustración 66 Cuando se presiona el botón DEMO desde la interfaz de Usuario	84
Ilustración 67 Conexión entre el módulo EasyPIC y el sensor.....	85
Ilustración 68 Interfaz de usuario practica #5, puerta cerrada	86
Ilustración 69 Interfaz de usuario práctica #5, puerta abierta	86
Ilustración 70 Conexión entre EasyPIC, L298N y motor DC.....	88
Ilustración 71 Conexiones practica #6, EasyPIC y prototipo 3D.....	89
Ilustración 72 Pruebas de velocidad del motor DC.....	89
Ilustración 73 interfaz de usuario práctica #6, motor en sentido Antihorario.....	90
Ilustración 74 Interfaz de usuario práctica #6, motor en sentido horario.....	91
Ilustración 75 Conexión para habilitar la salida del motor A	92
Ilustración 76 Conexión de componentes: Práctica #7	93
Ilustración 77 Interfaz de usuario práctica #7, temperatura baja	93
Ilustración 78 Interfaz de usuario práctica #7, temperatura alta	94

ÍNDICE DE TABLAS

Tabla 1 Capacidades de alimentación de entrada mediante USB	12
Tabla 2 Capacidades de alimentación mediante una Batería Li-Po y Li.Ion	13
Tabla 3 Capacidades de alimentación a través de fuente externa de 12V	13
Tabla 4 Switch (SW6) para habilitar e inhabilitar los puertos LEDS	21
Tabla 5 Configuración de los polos del interruptor SW6	23
Tabla 6 Interruptor (SW2) habilita e inhabilita el panel táctil	24
Tabla 7 Configuración de los polos del SW6.3 y SW6.4 para el LCD	25
Tabla 8 Control de Pines TX y RX del MCU (SW21 y SW23)	28
Tabla 9 Indicador LED (Muestra el tráfico UART).....	28
Tabla 10 Configuración de pines RB2 y RB3 para comunicación CAN.....	29
Tabla 11 Configuración de pines para usar el periférico USB.....	30
Tabla 12 Configurar pines RA2 y RA3 como fuente de alimentación o GPIO	30
Tabla 13 Datos técnicos PIC 16F887.....	44
Tabla 14 Detalles técnicos del sensor SHARP	46
Tabla 15 Detalles técnicos del sensor LM35	47
Tabla 16 Tabla de verdad para control de dos motores	49
Tabla 17 Características principales del L298N	50
Tabla 18 Pines funcionamiento en LCD	52
Tabla 19 Costo de equipos de Importación.....	71
Tabla 20 Costos de dispositivos adquiridos dentro del país	71
Tabla 21 Costos de Desarrollo	72
Tabla 22 Costo final de la propuesta.....	72
Tabla 23 Conexión para salida del motor B.....	88
Tabla 24 Conexión para salida del motor A	92

INTRODUCCIÓN

El presente trabajo es una herramienta de desarrollo que guiará a los estudiantes en la resolución de prácticas de laboratorio basadas en temas relacionados a los microcontroladores (MCU) PIC pertenecientes a la familia 16F de microchip, facilitando el estudio y la programación de algunos de los módulos internos que conforman el microcontrolador; optimizando el tiempo en el montaje del hardware y agilizando la resolución de las guías de actividades prácticas.

El capítulo I describe cómo surgió la idea de la propuesta detallando ciertos problemas y a la vez justificando la solución planteada, además se proyectan los objetivos y los resultados que se esperan obtener al finalizar los capítulos propuestos.

En el Capítulo II se desarrollan los tres tipos de marcos entre ellos el marco contextual, conceptual y teórico, dando a conocer los elementos utilizados para desarrollar la propuesta, además conceptos generales de cada uno de ellos en relación al tema planteado.

En el Capítulo III se detalla el desarrollo de la propuesta, realizando las 7 aplicaciones prácticas empleando MCU PIC de 8 bits. Se utiliza dos lenguajes de programación, en C y assembler, además se proyectan evidencias físicas, descriptivas y gráficas en consecuencia a la práctica planteada. Se comprueba su funcionalidad con el módulo EASYPIC V8 y la factibilidad que presenta al tener una placa de desarrollo.

CAPITULO I

GENERALIDADES DE LA PROPUESTA

1.1 ANTECEDENTES

La Universidad Estatal Península de Santa Elena cuenta con la Facultad de Sistemas y Telecomunicaciones, en el año 2010 dio apertura de dos carreras de ingeniería como Electrónica y Telecomunicaciones e Informática. Actualmente la facultad está compuesta por cinco carreras donde tres hicieron su aparición en el periodo 2018-1 entre ellas están Electrónica y Automatización, Tecnologías de la Información, Telecomunicaciones. Además, se conoce que la facultad posee 9 laboratorios técnicos los mismo que fueron distribuidos para las diferentes especializaciones que brinda la carrera [1].

La carrera de Electrónica y Telecomunicaciones aún vigente posee 139 estudiantes matriculados a partir del cuarto semestre, se sabe que la carrera de Electrónica y Automatización cuenta con 104 estudiantes matriculados hasta el tercer semestre, se hace mención de estas dos carreras debido a que en su malla curricular contienen algunas materias similares para su formación profesional, entre ellas la materia de Microcontroladores la cual requiere la realización de actividades prácticas para tener en claro la teoría. La facultad tiene asignado un lugar de trabajo para el área de Electrónica, pero cabe mencionar que las herramientas para prácticas de MCU no son del todo versátiles.

La posibilidad de encontrar los dispositivos electrónicos en el mercado local es reducida, debido a las limitadas tiendas existentes en la provincia. De modo que, si el estudiante no logra obtener los materiales necesarios no podrá llevar a cabo sus trabajos encomendados, esto hace que exista un vacío en su conocimiento práctico. El hecho de utilizar una tarjeta entrenadora con todos los elementos y dispositivos interconectados,

deja a un lado la preocupación de estar buscando los dispositivos ya que los entrenadores incorporan todos los elementos necesarios, esto facilita al momento de preparar los trabajos y también sirve de apoyo para expandir el conocimiento práctico y científico de los microcontroladores.

Otras de la causas por las cuales no se realizan actividades prácticas de MCU completas o a tiempo son: los retrasos externos y desactualización de contenidos que son puntos muy importantes al analizar la problemática, por ejemplo, el docente tiene un tiempo determinado para brindar su cátedra y en la mayoría de las prácticas, existen retrasos provocado por los estudiantes debido al tiempo que se toman al montar los circuitos electrónicos ya sea por motivos que no reviso las especificaciones técnicas o simplemente no posee los materiales respectivos, esto hace que el docente no logre avanzar con los objetivos planteados previamente. El contenido de la asignatura se ha desactualizado debido a que con el pasar de los años las nuevas tecnologías hacen presencia tanto en dispositivos como sensores, actuadores e indicadores. Actualmente el contenido del syllabus llega hasta comunicación serial y generación de señales PWM (modulación por ancho de pulsos), por lo que solo se realizan simulaciones básicas sobre temáticas como domótica, IOT (Internet de las cosas), Comunicaciones Inalámbricas, Automatización, Control y plataformas Matlab.

Es necesario mencionar que el empleo de dispositivos como el MCU PIC, se torna complejo al instante de transferir cierta trama de datos en tiempo real a una plataforma que permita monitorear sus datos y generar un análisis de los mismos, ya que el análisis sobre el comportamiento de un elemento específico es lo primero que se necesita para llevar a cabo el respectivo control.

De acuerdo con la problemática expuesta se estima necesario contar con herramientas de desarrollo que empleen la tecnología actual en este caso se propone utilizar un módulo de entrenamiento basado en MCU PIC de 8 bits con el objetivo de entrenar a los estudiantes en el tratamiento de señales analógicas y digitales (A/D) que

proporcionan los diferentes periféricos del MCU. De modo que será empleado para desarrollar diferentes actividades prácticas entregando la posibilidad de que el estudiante se concentre más en sus nuevas ideas de aplicaciones con MCU en vez de preocuparse por las conexiones entre dispositivos.

1.2 DESCRIPCIÓN DEL PROYECTO

En esta propuesta se plantea desarrollar diferentes aplicaciones prácticas basados en temas relacionados a los MCU PIC de 8 bits, de modo que se emplea una tarjeta de entrenamiento EASYPIC el cual me permite manipular los diversos periféricos del microcontrolador. Está dirigido a las personas que tienen conocimientos básicos de microcontroladores, sistemas digitales y códigos de programación. Esta tarjeta quedará en el laboratorio de electrónica dando la oportunidad de realizar actividades prácticas con las herramientas que esta posee, la misma que está compuesta por los siguientes elementos:

- Guía de aplicaciones prácticas
- Hardware
- Software

El presente proyecto se enfoca en el desarrollo de las diferentes aplicaciones prácticas basadas al MCU PIC 16F887, consiste en 7 actividades prácticas aplicando temas relacionados al lenguaje ensamblador, lenguaje C, operadores lógicos, interrupciones, conversores A/D, señales PWM, comunicación Serial (Comunicación Serial Asíncrona), entre otros. Además, se desarrolla el respectivo algoritmo de programación para realizar el tratamiento y análisis de las señales adquiridas de los diferentes periféricos del módulo de entrenamiento, también se utiliza la herramienta MATLAB-GUIDE para diseñar interfaces gráficas para que el usuario pueda visualizar y manipular los diferentes actuadores, sensores e indicadores que se emplean en algunas de las prácticas.

1.3 OBJETIVOS DEL PROYECTO

1.3.1 OBJETIVO GENERAL

Desarrollar aplicaciones de microcontroladores que manipulen actuadores y sensores, empleando el módulo EasyPIC v8 para contribuir al análisis teórico y práctico de los microcontroladores.

1.3.2 OBJETIVOS ESPECÍFICOS

- Analizar los elementos electrónicos que conforman el módulo EasyPIC.
- Desarrollar aplicaciones de microcontroladores manejando lenguajes de programación en C y ensamblador para interacción entre periféricos y el PIC 16F887.
- Desarrollar interfaz gráfica en Matlab para adquisición de datos del microcontrolador.
- Sintetizar los resultados obtenidos de las aplicaciones prácticas de microcontroladores para evidenciar el rendimiento del módulo EasyPIC v8.

1.4 RESULTADOS ESPERADOS

- Se espera como resultado obtener una documentación actualizada en español sobre la tarjeta entrenadora, una guía entrenadora de aplicaciones prácticas de microcontroladores para el instructor.
- Permitir que el instructor o el estudiante pueda hacer diferentes actividades en el módulo de entrenamiento con la libertad de desarrollar nuevas aplicaciones prácticas.

1.5 JUSTIFICACIÓN DEL TEMA

La necesidad de usar herramientas tecnológicas de nueva generación para el avance y mejoramiento del conocimiento científico en el usuario justifica el siguiente proyecto, que se orienta a guiar al estudiante en la resolución de prácticas de laboratorio a través de sistemas basados en MCU. El proyecto es una herramienta de desarrollo, que permite familiarizar al estudiante con nuevas tecnologías y técnicas. Además, facilitar el estudio y la programación de algunos de los módulos que conforman el MCU; optimizando el tiempo en el montaje del hardware y agilizando la resolución de las guías de actividades prácticas ya que estos permiten un fácil uso y una mejor comodidad al momento de la implementación.

Se elige como unidad de control a los MCU PIC de 8 bits debido a que permiten el manejo de varios lenguajes de programación como es el lenguaje ensamblador, ya que este es la base para que el estudiante pueda iniciar el estudio de cualquier otro dispositivo, además de su robustez para llevar a cabo procesos de control. Se elige al PIC 16F887 ya que internamente vienen incorporados diversos periféricos, como sistemas PWM, conversores Analógicos/Digitales, comunicación I2C, SPI, UART y permiten tener múltiples opciones de trabajos, además manejan un lenguaje de programación de alto nivel, lo que permite una programación rápida y comprensible.

El docente encargado de brindar la asignatura de Microcontroladores se verá beneficiado, ya que le permitirá concluir sus clases con el tiempo estimado y cumplir con los objetivos propuestos dado que el módulo de entrenamiento y las actividades prácticas propuestas ayudaría a brindar una explicación sencilla de la teoría y la práctica enfocándose más en la parte de la creación de la codificación de control, además de eliminar los posibles errores que suelen cometerse en la conexión de hardware.

El módulo es una herramienta importante para que el estudiante comience a conocer las posibilidades que ofrecen los diversos periféricos y poder desarrollar diferentes

aplicaciones prácticas, permitiéndoles utilizar y explorar al máximo las herramientas tecnológicas actuales, puesto que hoy en día la tendencia consiste en utilizar sistemas embebidos para el desarrollo de aplicaciones de robótica, domótica, IOT (Internet de las cosas), sensores, automatización y control, comunicaciones inalámbricas.

1.6 ALCANCE DEL PROYECTO

Con esta propuesta se pretende que cualquier estudiante que recibe materias relacionadas a los MCU, tengan la oportunidad de realizar diferentes actividades prácticas con las herramientas que posee el módulo de entrenamiento, con el objetivo de familiarizar al estudiante con nuevas tecnologías, fortalecer los conocimientos teóricos sobre el uso de los circuitos programables, desarrollar criterios para la solución a los problemas de procesamiento de datos de tipo analógico o digital que involucren las entradas y salidas de algún proceso. Los prerequisites del participante para realizar las actividades prácticas son; tener conocimientos básicos en MCU, sistemas digitales y códigos de programación.

1.7 METODOLOGÍA

TIPOS DE INVESTIGACIÓN

Este proyecto para el desarrollo de aplicaciones prácticas requiere de diferentes tipos de investigación:

1.7.1 INVESTIGACIÓN EXPLORATORIA

Se maneja una investigación exploratoria debido a que los estudiantes de las distintas carreras de la facultad de sistemas y telecomunicaciones tienden a utilizar temas relacionados a domótica, automatización y control, siendo estos algunos de los temas poco estudiados en la asignatura de Microcontroladores. Se trata de probar la efectividad y la eficiencia de un módulo de entrenamiento EASYPIC v8 ya que es una de las últimas versiones lanzadas por la empresa MikroE empleando MCU PIC de 8

bits, de modo que permita garantizar el buen desempeño de las aplicaciones prácticas planteadas.

1.7.2 INVESTIGACIÓN EXPERIMENTAL

Consiste en realizar diferentes tipos de pruebas de laboratorio con el módulo de entrenamiento EasyPIC que permita la resolución de las 7 aplicaciones prácticas planteadas. En este caso el esquema de la práctica debe ser previamente simulada para prevenir errores antes de la implementación real. Se utiliza dispositivos electrónicos como display de siete segmentos, sensores de distancia, temperatura, motores DC, LCD 2x16, donde se realizan códigos de prueba para verificar el funcionamiento de cada uno.

1.7.3 INVESTIGACIÓN BIBLIOGRÁFICA

En esta propuesta de titulación se realizarán consultas a revistas, libros y artículos científicos que se publican en la web, que servirán de apoyo para cumplir con los objetivos planteados.

CAPITULO II

PROPUESTA TECNOLÓGICA

2.1 MARCO CONTEXTUAL

La Facultad de Sistemas y Telecomunicaciones en el periodo 2018-1 ofertó dos nuevas carreras como son Telecomunicaciones, Electrónica y Automatización, actualmente los estudiantes de estas carreras se encuentran cursando el cuarto semestre del periodo 2019-2. Las carreras mencionadas contienen en su malla curricular asignaturas similares como Microcontroladores, una asignatura donde es necesario la realización de prácticas físicas para tener en claro la teoría. El laboratorio existente en la facultad posee ciertas herramientas, más no las necesarias para llevar a cabo una práctica de MCU, actualmente para la resolución de prácticas de MCU el estudiante debe adquirir la lista de materiales desde otras ciudades, y a partir de ello cumplir con lo estimado.

Esta propuesta se basa en desarrollar aplicaciones de MCU utilizando un módulo de entrenamiento EASYPIC v8, se desarrollarán aplicaciones prácticas con diferentes temas y diferentes elementos electrónicos, con el fin de que los estudiantes de las carreras mencionadas obtengan más conocimientos acerca del funcionamiento de los MCU y comiencen a conocer las posibilidades que ofrecen los diversos periféricos para poder desarrollar nuevas aplicaciones prácticas, permitiéndoles utilizar y explorar al máximo las herramientas tecnológicas actuales. Se emplean dos softwares para su respectiva codificación entre ellos: MPLAB y MikroC Pro For PIC, donde permiten los lenguajes de programación como el lenguaje C y ensamblador respectivamente, con esto se logra un despliegue total acerca de los MCU. Se utiliza la herramienta de Matlab para crear interfaces de usuario que permita la manipulación y el control del módulo EasyPIC v8, que reciba y transmita datos de manera serial entre ambos equipos, para así poder visualizar en tiempo real los datos proporcionados por el módulo.

En esta propuesta se utilizará el módulo EASYPIC v8 fabricado por la empresa Mikro lanzada en el año 2019, el cual es una herramienta de desarrollo que soporta 273 MCU producidas por microchip, tiene un diseño robusto, compacto y ergonómico que permite un desarrollo cómodo de las diferentes aplicaciones planteados. La placa de desarrollo está equipada con dos conectores de pantalla uno es de 1x16 LCD, el segundo es de 1x20 de una sola fila para GLCD con resolución 128x64 pixeles capaz de mostrar contenido gráfico. La creación rápida de prototipos permite al ingeniero tomar la manera más eficiente y sin esfuerzo, para visualizar las ideas, capacidades y limitaciones de diseño a través de medios medibles [2].

2.2 MARCO CONCEPTUAL

2.2.1 TARJETAS DE DESARROLLO

Las tarjetas o placas de desarrollo, son útiles para el desarrollo de aplicaciones y sistemas embebidos, además de ser un material didáctico que tiene los recursos suficientes para facilitar el estudio de los conceptos, habilidades y destrezas con respecto a los MCU PIC de 8 bits.

Son empleadas como un material de apoyo en el proceso de aprendizaje ya que contienen una amplia gama de herramientas que posibilitan la interacción, exploración y creación de nuevos proyectos, generando nuevos conocimientos en los estudiantes y personas que quieran iniciar el estudio de los MCU [3].

Las tarjetas de MCU son diseñadas con el objetivo de ayudar a estudiantes y personas que deseen incursionar en el proceso de aprendizaje ya que contiene un hardware diseñado acorde a las necesidades del usuario, para aquellos que lo utilizasen solo se centren en el desarrollo del código de programación sin tener que preocuparse en las conexiones de hardware y los periféricos que se desee utilizar [3].

La empresa MikroElektronika ha desarrollado varias placas de desarrollo, de las cuales ha venido presentando mejoras evidentes en cada presentación. Se menciona algunas

de las placas que específicamente son utilizadas para trabajar con los MCU de 8 bits, entre ellas están:

1. EASYPIC v5
2. EASYPIC v6
3. EASYPIC v7
4. EASYPIC PRO v7
5. EASYPIC v8

En la ilustración 1, se presenta la placa de desarrollo EasyPIC V8 con todos los elementos, características y periféricos que esta posee.



Ilustración 1 Placa de desarrollo EasyPIC V8

2.2.2 CARACTERISTICAS DE LA PLACA EASYPIC v8

Se hace énfasis la siguiente placa de desarrollo debido a que se utilizara para el desarrollo de esta propuesta, la placa de desarrollo está diseñada para el rápido desarrollo de aplicaciones embebidas, basados en MCU PIC de 8 bits, ofrece una configuración familiar con características estándar así como características nuevas y únicas en relación a las anteriores versiones: programación y depuración a través de

wi-fi o conectores USB-C, soporta una amplia gama de MCU, tiene todo lo que se necesita para el desarrollo de aplicaciones [2].

La tarjeta se divide en varias secciones, las mismas que están compuestas por interruptores, indicadores, botones y los conectores. Cada sección contiene componentes importantes para que su funcionamiento sea el correcto.

2.2.2.1 PUERTOS DE ALIMENTACIÓN DE ENTRADA

1. Conector USB Tipo C
2. Conector de Batería
3. Conector tipo barril

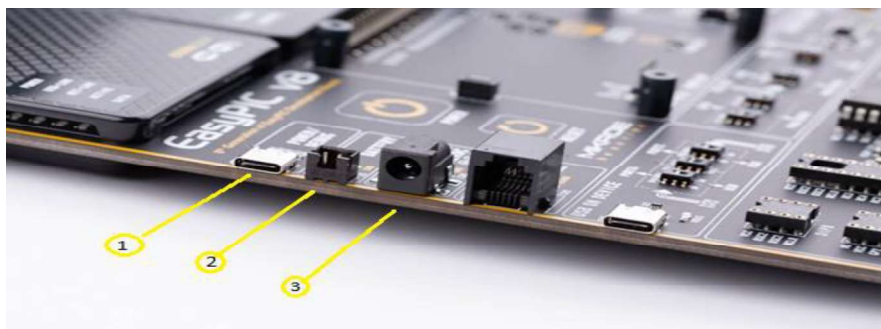


Ilustración 2 Puertos de alimentación de entrada hacia la placa EasyPIC v8

CONECTOR USB TIPO C

La placa de desarrollo puede ser alimentada a través de un conector USB tipo C, a través del host USB de un computador, o un adaptador USB de pared [2].

La siguiente tabla detalla las calificaciones máximas que soporta y entrega la placa.

Fuente de alimentación USB				
Voltaje de entrada [V]		Voltaje de salida [V]	Corriente Máxima [A]	Potencia Máxima [W]
MIN	MAX	3.3	1.8	6
4.4	5.5	5	1.4	7
		3.3 y 5	0.8 y 0.8	6.64

Tabla 1 Capacidades de alimentación de entrada mediante USB

CONECTOR DE BATERIA

La placa de desarrollo utiliza un conector de batería XH de 2,5mm. Permite conectar una serie de baterías entre ellas las de Li-Po y Li-Ion. Además, ofrece la función de carga de la batería tanto del conector USB o de la fuente de alimentación externa. [2]

La siguiente tabla detalla las calificaciones máximas que soporta y entrega la placa.

Fuente de alimentación mediante Batería				
Voltaje de entrada [V]		Voltaje de salida [V]	Corriente Máxima [A]	Potencia Máxima [W]
MIN	MAX	3.3	1.6	5.28
3.5	4.2	5	1.2	6
		3.3 y 5	0.7 y 0.7	5.81

Tabla 2 Capacidades de alimentación mediante una Batería Li-Po y Li.Ion

CONECTOR TIPO BARRIL

Permite la alimentación de una fuente externa de 12V la misma que puede ser conectado sobre el conector de barril que contiene la etiqueta 12VDC. Cuando se utiliza una fuente de alimentación externa, es posible obtener una cantidad óptima de energía. La tarjeta de desarrollo permite una corriente máxima de 2.8A y un voltaje de salida de (3,3V y 5V) cuando se utiliza una alimentación de 12V. El conector de tipo barril es útil ya que facilita la conexión con adaptadores de pared. [2]

Fuente de alimentación externa 12VDC				
Voltaje de entrada [V]		Voltaje de salida [V]	Corriente Máxima [A]	Potencia Máxima [W]
MIN	MAX	3.3	2.8	9.24
10.6	14	5	2.8	14
		3.3 y 5	2.8 y 2.8	23.24

Tabla 3 Capacidades de alimentación a través de fuente externa de 12V

El módulo de fuente de alimentación compatible con la redundancia del suministro de energía te permite estar conectado en los diferentes conectores al mismo tiempo ya que el siguiente modulo se encarga de elegir a la fuente de alimentación más apropiada para la placa de desarrollo. Así no tendrán interrupciones al momento de desconectar alguna de ellas, por ejemplo, tengo conectado la entrada de alimentación USB-C y también la conexión externa tipo barril de 12VDC, aquí entra el trabajo del módulo UPS para elegir la mejor fuente de alimentación y fácilmente se puede desconectar cualquiera de ellas sin ocasionar interrupciones en la placa de desarrollo.

2.2.2.2 SISTEMA DE ALIMENTACIÓN

La unidad de fuente de alimentación (PSU) proporciona energía limpia y regulada, necesaria para el correcto funcionamiento de la tarjeta de desarrollo. El MCU que se este utilizando, junto con el resto de los periféricos, exigen una fuente de alimentación regulada y libre de ruidos. Por lo tanto, la fuente de alimentación se encuentra cuidadosamente diseñado para regular, filtrar, y distribuir la potencia a todas las partes de la placa de desarrollo. Está equipado con tres entradas de alimentación ofreciendo toda la flexibilidad que necesita el EasyPIC V8, especialmente cuando se utilizan en el area de trabajo [2].



Ilustración 3 Fuente de alimentación regulada y filtrada

2.2.2.3 BOTON DE ENCENDIDO Y DE REINICIO

La placa de desarrollo contiene dos botones táctiles con las etiquetas de POWER y RESET, que se utilizan para encender y restablecer el MCU. Son botones sensibles al tacto específicamente diseñados para interfaces hombre-máquina.

Cuando la placa de desarrollo detecta una fuente de alimentación automáticamente entra en stand-by a la espera de que se pulse el botón capacitivo de encendido, para iniciar a distribuir la energía al resto del equipo.



Ilustración 4 Botón POWER y RESET de la placa EasyPIC v8

Debajo del botón de encendido como se observa en la ilustración 4 se encuentra el botón capacitivo de RESET, este sirve para restablecer o reiniciar el MCU, aquella no se encuentra ligada con la alimentación de energía, sino que se encuentra conectada directamente con el pin MCLR del MCU. Si el botón es presionado se activará el reinicio del MCU.

2.2.2.4 ZÓCALOS DIP PARA MICROCONTROLADORES

La placa EasyPic v8 está constituida por 8 sócalos DIP que van desde DIP8 hasta DIP40, es decir soporta MCU de 8 hasta 40 pines. Todos los zócalos DIP permiten la ubicación exclusiva de MCU, desde el más pequeño hasta el más grande, como se puede apreciar en la Ilustración 5. Por ejemplo, si se desea colocar el MCU PIC16F887 el cual posee 40 pines, esta debe ser ubicada en el DIP40 ya que si ubica en otra posición causará desalineación en los pines y otros problemas. [2]

Para instalar el MCU en el zócalo DIP primero asegúrese que el corte circular coincida con el corte circular del zócalo DIP de la placa, luego alinear ambos extremos del MCU

con el zócalo, presionar lentamente hacia abajo hasta que todos los pines encajen en el zócalo. Nota importante, solo puede instalarse un MCU a la vez.

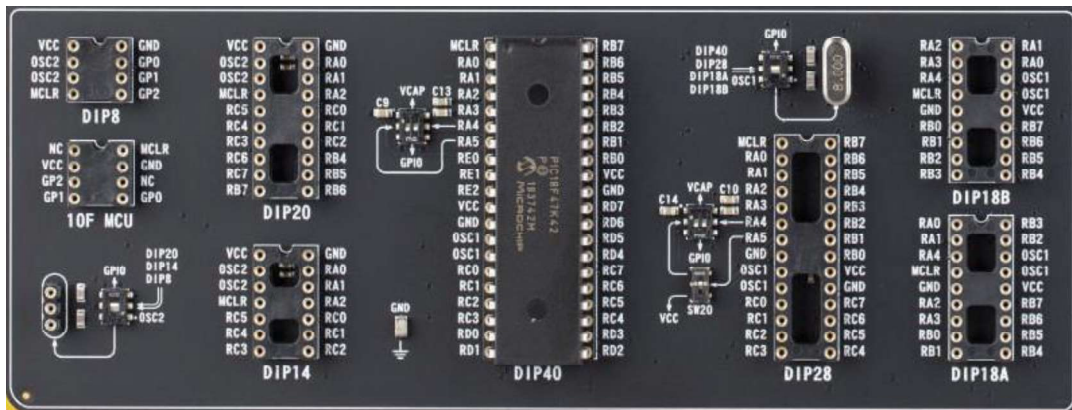


Ilustración 5 Indica los 8 zócalos de la tarjeta EasyPIC v8

2.2.2.5 OSCILADOR DE CRISTAL

La mayoría de los microcontroladores PIC a menudo requieren de un oscilador externo (cristal de cuarzo) para estabilizar la frecuencia de reloj. La EasyPIC v8 cuenta con dos zócalos para instalar el oscilador de cristal. El oscilador de cristal OSC1 de la placa se encamina a los zócalos de los MCU ubicados en los DIP40, DIP28, DIP18A y DIP18B. En caso de utilizar los zócalos DIP20, DIP14 y DIP8 se debe utilizar el oscilador de cristal OSC2. Esta placa te permite reemplazar el oscilador por cualquiera de otro valor, de acuerdo al límite de frecuencia de reloj que permite el microcontrolador. Los microcontroladores ubicados en el zócalo 10F utilizan osciladores internos es decir no se enrutan a las entras de los osciladores externos.



Ilustración 6 Socket para oscilador de cristal

El OSC1 de la placa tiene un interruptor DIP, utilizado para llevar el control de los pines RA6/RA7, los cuales serán utilizados como GPIO o como entradas para el oscilador.

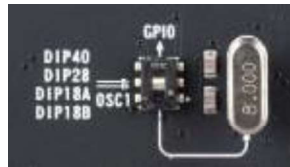


Ilustración 7 Interruptor DIP para OSC1

DIP Arriba: Configura a los pines RA6/RA7 como GPIO.

DIP Abajo: Los Pines RA6/RA7 se encuentran conectados a la entrada del oscilador.

El OSC2 ubicado por debajo de los dos zócalos DIP8 hay un interruptor DIP, utilizado para llevar el control de los pines RA4/RA5, los cuales serán utilizados como GPIO o como entradas para el oscilador.



Ilustración 8 Interruptor DIP para OSC2

DIP Arriba: Configura a los pines RA4/RA5 como GPIO.

DIP Abajo: los Pines RA4/RA5 se encuentran conectados a la entrada del oscilador.

2.2.2.6 SECCIÓN DE ENTRADA Y SALIDA

Los pines de entrada y salida de cualquier MCU son internamente agrupados como puertos, la cual se puede apreciar en la placa de desarrollo ya que nos ofrece una interfaz limpia y organizada de los puertos.

La placa de desarrollo EasyPIC v8 contiene cinco puertos, los cuales se encuentran etiquetados con el nombre PORTA, PORTB, PORTC, PORTD y PORTE. Depende del

MCU instalado en la placa se harán uso de los ciertos pines. Los puertos se encuentran ubicados en la parte inferior de la plataforma, cada puerto contiene ocho botones, ocho Leds, un switch DIP de 8 ocho polos, y un conector macho de 2x5 líneas, todos los puertos se encuentran enrutados al MCU como se observa en la ilustración 7.



Ilustración 9 Esquema de conexión de los puertos de E/S

BOTONES DE PRESIÓN

Se encuentran conectados directamente a los puertos de E/S del MCU. Son comúnmente llamados pulsadores (ver ilustración 9), estos se utilizan para aplicar un estado lógico (“1” o “0”) en las entradas digitales del MCU. Son pequeños botones metálicos SPTS que funcionan en combinación con un switch DIP(SW5), etiquetado como BUTTON PRESS LEVEL [2]

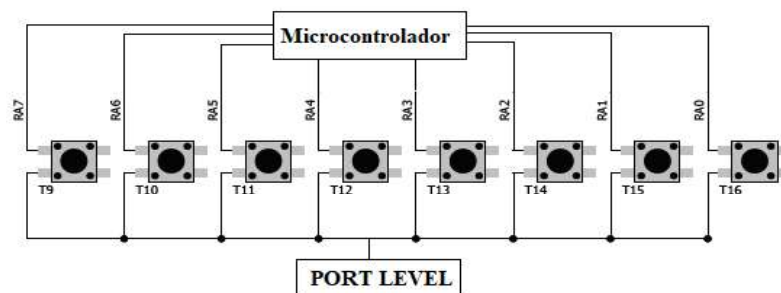


Ilustración 10 Puerto A del MCU con Pulsadores

El switch DIP de 8 polos con la etiqueta BUTTON PRESS LEVEL situada en la parte central de la placa, es utilizado para configurar los botones de cada uno de los puertos, entre ellos configurar que todos los botones de los puertos estén deshabilitados, para evitar que se presione algún botón accidentalmente. Solo cinco de polos del switch se utilizan ya que ellos permiten el control de los cinco puertos existentes (los demás polos con la etiqueta NIC no son utilizados debido a que no se encuentran conectados).

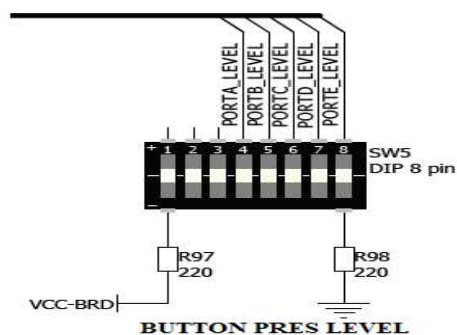


Ilustración 11 Switch para controlar los puertos de los pulsadores

Posición UP: Si hace presión a un pulsador se está aplicando un nivel lógico alto (“1”).

Posición media: Los Botones están completamente desconectados.

Posición Down: Si presiona un botón se aplica un nivel lógico bajo (“0”).

SWITCH DIP8

En la sección de entrada y salida también se encuentra los cinco interruptores DIP de 8 polos asociado cada uno de los puertos. En la placa de desarrollo lo encontramos etiquetado con el nombre de UP-PULL-DOW y es utilizado para habilitar o interrumpir el paso de corriente en la entrada del MCU.

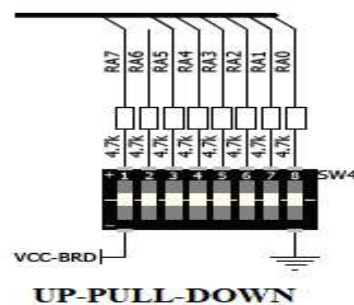


Ilustración 12 Switch de 8 polos (UP-PULL-DOWN) del puerto A

Posición UP: Conecta una resistencia de $4.7k\Omega$ entre el carril de alimentación y el pin del MCU (DIP levantado).

Posición media: Desactiva ambos estados, es decir se encuentra en un estado flotante (DIP en la parte central).

Posición DOW: Conecta una resistencia de $4.7k\Omega$ entre el GND y el pin asociado al MCU (DIP abajo).

INDICADORES LEDS

En los cinco puertos de la placa de desarrollo se encuentra instalado un grupo de 8 diodos LEDS respectivamente, utilizados para indicar de forma visual el estado lógico del pin que se esté utilizando. La corriente que circula a través del diodo es limitada por una resistencia de $4.7k\Omega$ la misma que sirve para proteger el diodo LED de posibles averías. Cada led internamente se encuentra conectado a un pin del puerto y están etiquetados de acuerdo al pin conectado, las etiquetas van desde (RA0 a RE7). Un led encendido o activo nos indica que el MCU desde el pin está enviando un estado lógico (1). En cada puerto si los LEDS no son utilizados deben ser deshabilitados, esto se logra desde el interruptor DIP (SW6) de 10 polos ubicado en la placa de desarrollo.

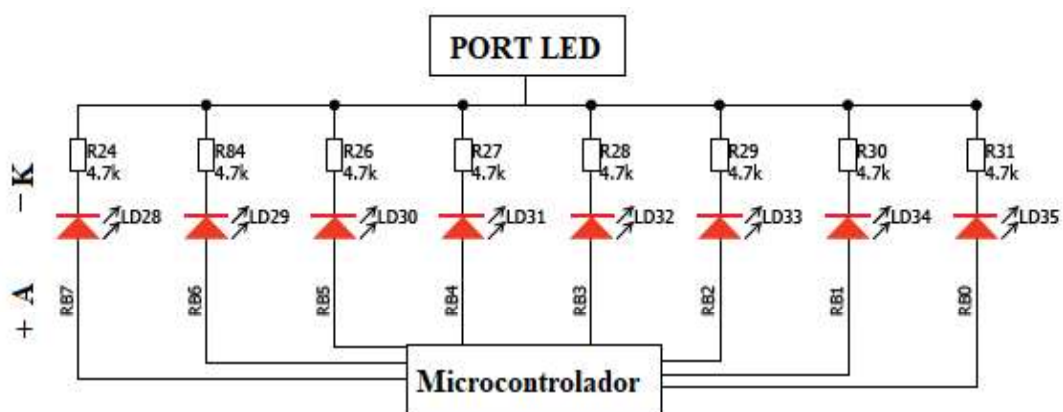


Ilustración 13 Indicadores LEDS conectado al puerto B de la placa EasyPIC v8

Para habilitar o deshabilitar los puertos de los LEDS dirigirse a la placa de desarrollo y ubicar la etiqueta BOARD SETUP y dentro de ella al switch (SW6) DIP de 10 polos.

Donde cinco de los polos del switch (SW6) etiquetados como PORT LEDS son utilizados para activar o desactivar los LED de cada puerto (ver tabla 4).

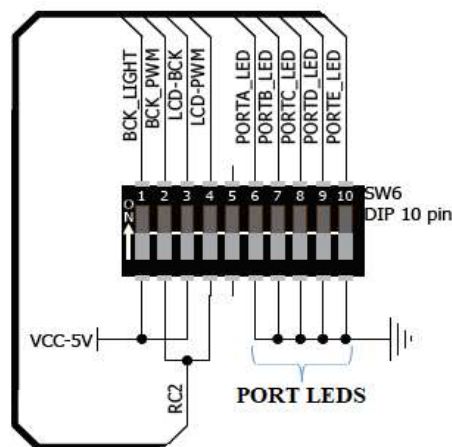


Ilustración 14 DIP de 10 polos para habilitar los puertos de los LEDS

	ON (Arriba)	OFF (Abajo)
SW6.6	Habilita LEDS del puerto A	Inhabilita LEDS del puerto A
SW6.7	Habilita LEDS del puerto B	Inhabilita LEDS del puerto B
SW6.8	Habilita LEDS del puerto C	Inhabilita LEDS del puerto C
SW6.9	Habilita LEDS del puerto D	Inhabilita LEDS del puerto D
SW6.10	Habilita LEDS del puerto E	Inhabilita LEDS del puerto E

Tabla 4 Switch (SW6) para habilitar e inhabilitar los puertos LEDS

CONECTORES MACHO 2X5

Además, incluye cinco conectores machos de 2x5 pines, los cuales se encuentran conectados a los puertos del MCU. Cada conector de 2x5 incluye un pin para GND y otro para alimentación (+5V), las cabeceras se encuentran etiquetadas de acuerdo al nombre del pin que esté conectado al MCU y van desde (RA0-RA7 a RE0-RE7).

Estos se utilizan para expandir o ampliar la conectividad entre los diferentes dispositivos, periféricos y equipos de medición. De tal forma que pueden actuar como

entradas o salidas hacia el MCU, es una de las mejoras que presenta esta placa de desarrollo EasyPIC v8, que da apertura a muchas posibilidades de conexión (entre ellos pueden ser: sensores, actuadores, indicadores o visualizadores) que se encuentren conectados desde la parte exterior de la plataforma.

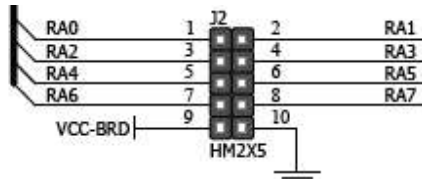


Ilustración 15 Conectores macho 2x5 del puerto A

2.2.2.7 CONECTOR DE PANTALLA GRÁFICA 1x20

La placa EasyPIC v8 está equipada con un conector de pantalla grafica de 1x20 pines de una sola fila, que admite soporte para pantallas gráfica de cristal líquido GLCD con resolución 128x64 pixeles, compuesta por el controlador de pantalla KS108 o compatible. Son utilizadas para mostrar contenido monocromático como, caracteres, textos, imágenes, etc. Contiene líneas de comandos, rieles de alimentación (GND, +5V), control de luz de fondo PWM y el control de tensión para el contraste [2].

GLCD/TFT Socket

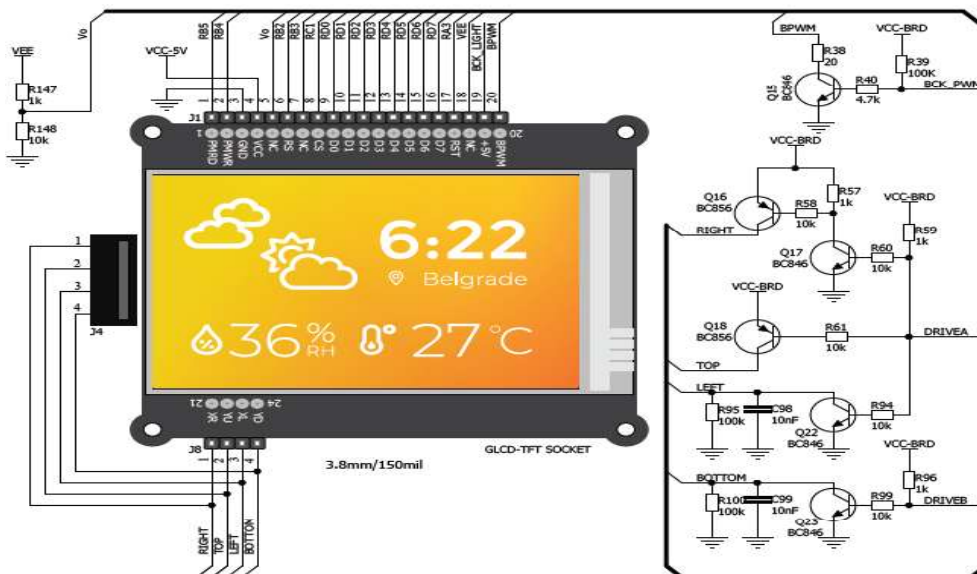


Ilustración 16 Socket para GLCD en EasyPIC v8

Los GLCD por lo general incluyen un panel táctil, los cual puede ser usado como un dispositivo de entrada. EasyPIC v8 proporciona los circuitos necesarios para interconectar el panel táctil resistivo con el MCU instalado. Consta de dos conectores para las entradas del panel táctil: el primero es un conector de cable plano de 4 polos (FFC), etiquetado como J4, el segundo es un conector macho de 1x4 polos con la etiqueta J8.

En el bloque BOARD SETUP situado en el centro de la placa EasyPIC v8, encontramos dos interruptores DIP (SW6 y SW2), utilizados para configurar la luz de fondo del GLCD y el panel táctil.

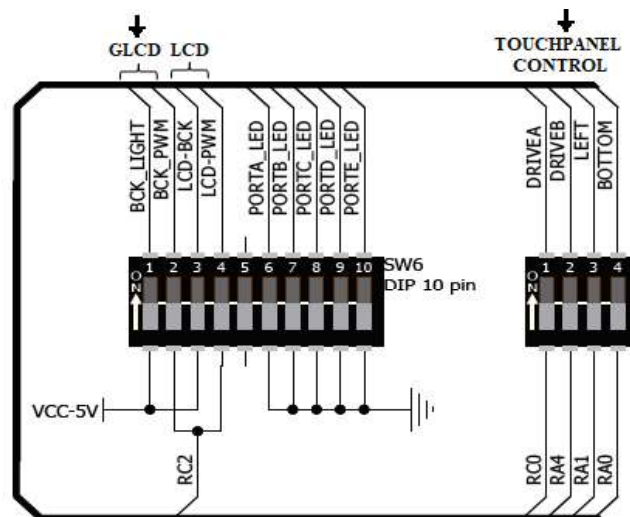


Ilustración 17 Interruptor (SW6) para GLCD e interruptor (SW2) panel táctil.

Dos polos del switch SW6 son utilizados para controlar la retroiluminación de la GLCD.

	ON (Arriba)	OFF (Abajo)
SW6.1 BCK_LIGHT	GLCD luz de fondo encendida	Retroiluminación desactivada
SW6.2 BCK_PWM	Señal PWM contrala el brillo de retroiluminación de GLCD	Brillo de fondo fijo y no puede ser controlado por el MCU

Tabla 5 Configuración de los polos del interruptor SW6

El interruptor DIP de 1x4 pines con la etiqueta SW2 permite interconectar el panel táctil con los pines RA0, RA1, RA4 y RC0 del MCU. Para hacer uso del panel táctil todos pines del interruptor deben estar en la posición ON (arriba) caso contrario ubicar en la posición OFF (abajo).

	ON (Arriba)	OFF (Abajo)
SW2	Panel táctil activo.	Panel táctil desconectado.

Tabla 6 Interruptor (SW2) habilita e inhabilita el panel tactil

2.2.2.8 CONECTOR PARA PANTALLA DE CARACTERES 2x16

EasyPIC v8 contiene un conector para pantalla de caracteres, con pines machos de 1x16 de una sola fila, admite de forma nativa la Pantalla de Cristal Líquido (LCD) de 2x16 caracteres. Siendo esta una forma barata y popular de representar caracteres, símbolos y textos de forma visual hacia los usuarios.

Internamente las líneas de datos del LCD se encuentran conectadas en modo de 4 bits, D4 a D7 conectan al MCU y D0 a D3 a GND. El resto de los pines se interconectan como se puede apreciar en la ilustración 18.

LCD SOCKET

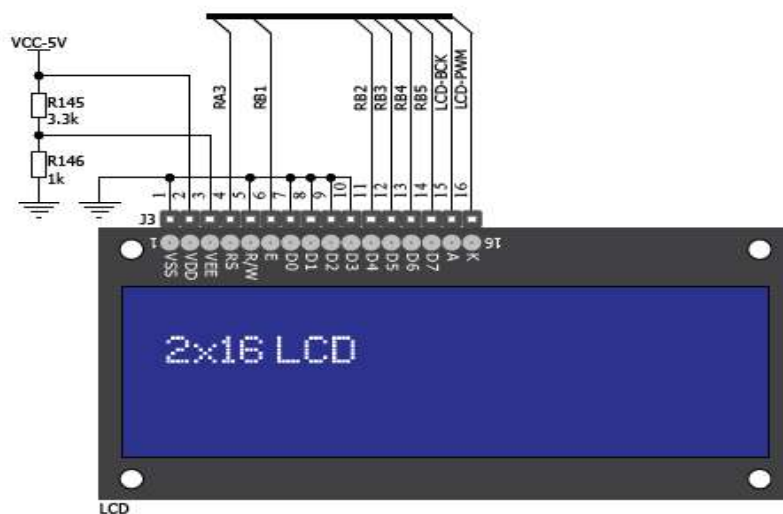


Ilustración 18 Socket para LCD (esquema electrónico)

En el bloque BOARD SETUP situado en el centro de la placa EasyPIC v8, encontramos el interruptor DIP (SW6), utilizado para configurar la luz de fondo del LCD.

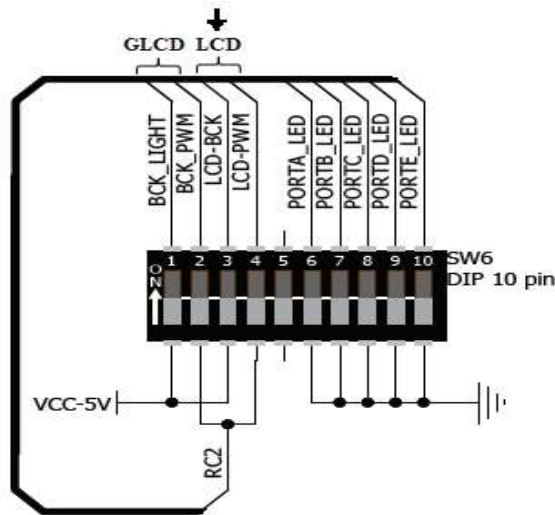


Ilustración 19 Interruptor DIP (SW6) para LCD

Dos polos del switch SW6 son utilizados para controlar la retroiluminación del LCD.

	ON (Arriba)	OFF (Abajo)
SW6.3 BCK_LIGHT	Enciende la luz de fondo del LCD	Retroiluminación desactivada
SW6.4 BCK_PWM	Controla la Señal PWM para el brillo de retroiluminación del LCD	Brillo de fondo fijo y no puede ser controlado por el MCU

Tabla 7 Configuración de los polos del SW6.3 y SW6.4 para el LCD

Nota: No pueden conectarse las dos pantallas simultáneamente (GLCD y LCD) ya que utilizan las mismas líneas de datos que unen al MCU por lo tanto causaría tráfico e interferencia en el envío o recepción de datos. La pantalla GLCD o TFT solo trabaja con los MCU de la familia 18F.

2.2.2.9 ZÓCALOS MIKROBUS

EasyPIC v8 está equipada con cinco zócalos mikroBUS, ubicados en la parte superior de la plataforma, se encuentran etiquetados como mikroBUS (1-5), que permite la conexión con circuitos electrónicos que tienen ciertos diseños de referencia y se encuentran estandarizados bajo la marca registrada como Click Board.



Ilustración 20 Zócalos mikroBUS en placa EasyPIC v8

Cada zócalo cuenta con 16 pines hembras, las cuales se encuentran interconectadas a las interfaces de comunicación del MCU (I2C, UART, SPI). Además de las líneas individuales de alimentación (5V y 3.3V), GND, PWM, interrupción, una entrada analógica, reset, donde cada una de ellas se encuentran etiquetadas con sus respectivos nombres de las funciones a realizar (ver ilustración 21).

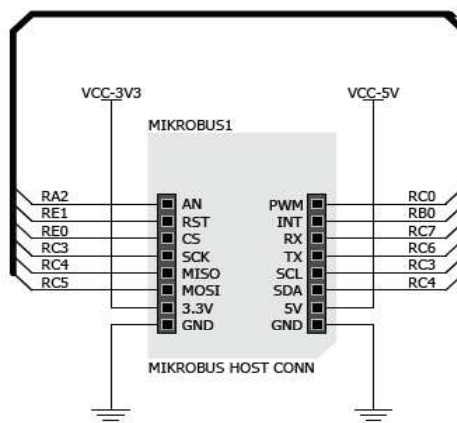


Ilustración 21 Esquema electrónico del mikroBUS1

2.2.2.10 COMUNICACIÓN

EasyPIC v8 contiene conectores de comunicación que permiten conectar diferentes dispositivos que usan protocolos de comunicación de capa alta, tales como USB, CAN, etc. Está equipada con un conversor USB-UART, para proporcionar una comunicación simplificada con el ordenador personal a través de la interfaz serial. Cada uno de los conectores se presentan más a detalle en la siguiente sección.

USB-UART

La interfaz UART (Universal Asynchronous Receiver/Transmitter), tradicionalmente la más utilizada para intercambiar datos entre el MCU y otros dispositivos externos (puede ser un ordenador). EasyPIC v8 brinda una conectividad UART a través de un conector USB tipo C, esto logra gracias al controlador FT230XQ instalado en la plataforma (Ver ilustración 22).

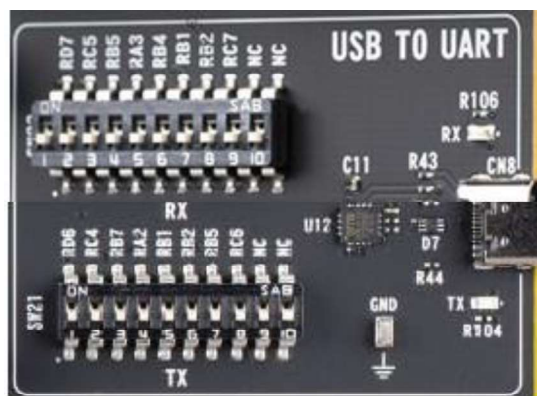


Ilustración 22 Bloque de comunicación USB-UART

Para habilitar las líneas de comunicación UART del MCU (TX, RX) e intercambiar datos a través del conector USB tipo C, ubicamos el bloque USB TO UART situado en la placa de desarrollo. En el siguiente bloque encontramos dos interruptores DIP de 10 polos cada uno, etiquetados como SW21 y SW23 respectivamente. Estos permiten el control total de las líneas UART (RX y TX) del MCU. La tabla 8 nos ayuda a apreciar el funcionamiento de los interruptores mencionados.

	ON (Arriba)	OFF (Abajo)
SW 21 (TX)	Conecta el pin TX del MCU al circuito conversor USB-UART	Desconecta el Pin TX desde el circuito conversor USB-UART
SW 23 (RX)	Conecta el pin RX del MCU al circuito conversor USB-UART	Desconecta el Pin RX desde el circuito conversor USB-UART

Tabla 8 Control de Pines TX y RX del MCU (SW21 y SW23)

Junto al conector USB-C está instalado dos indicadores LEDS, que indican el trafico UART que está ocurriendo en ese instante. Ya sea que se esté transmitiendo o recibiendo datos el led se encenderá.

LED Amarillo (TX)	Indica que los datos están saliendo del MCU.
LED Rojo (RX)	Indica que los datos están entrando al MCU

Tabla 9 Indicador LED (Muestra el trafico UART)

CAN

CAN responde a la abreviatura “Controllor Area Network” correspondiente a una Red de Área Controlada, en sus inicios fue diseñado para ser utilizada en la industria automotriz, pero debido a su inmunidad al ruido sea expandido su uso en otras aplicaciones. El bus CAN permiten establecer una rápida comunicación entre el MCU y los diferentes periféricos, la cual está compuesta de dos cables que proporcionan altas velocidades y mensajes de prevención en la colisión de datos [2].



Ilustración 23 Bloque de comunicación CAN en la placa EasyPIC v8

EasyPIC v8 incorpora el MCP2561FD, siendo este un transceptor de microchip que trabaja a una alta velocidad y permite una flexibilidad en los datos. Se encuentra etiquetado con el nombre de U19. El transceptor reúne los requisitos de automoción ya que excede una velocidad de datos de 2Mbps, baja corriente de reposo, compatibilidad electromagnética y la inmunidad a la descarga electrostática.

Para habilitar los pines que permitan la comunicación CAN, la placa de desarrollo tiene instalado un interruptor ubicado en el bloque CAN COMM, donde especifica la configuración de los pines RB2 y RB3 los cuales permiten utilizarse como GPIO o estar conectados al transceptor CAN (TX, RX). Solo permite esta comunicación para MCU de 28 y 40 pines.

	ON (Arriba)	OFF (Abajo)
SWITCH CAN	Configura los pines RB2 y RB3 como GPIO	Los pines RB2 y RB3 conecta al transceptor CAN.

Tabla 10 Configuración de pines RB2 y RB3 para comunicación CAN

Para conectarse al Bus CAN, la placa de desarrollo contiene un terminal de tornillo de dos polos, ubicado en la sección CAN COMM. Su conexión debe realizarse de acuerdo a las etiquetas CAN H y CAN L para tener una adecuada comunicación.

USB (DISPOSITIVO)

Universal Serial Bus (USB), es un estándar en la industria de los cables y son utilizados para la comunicación y fuente de alimentación entre diferentes dispositivos. USB presento su última versión de cables y es conocida como USB-C, este presenta muchas ventajas como el pin-out simétrica, mayor capacidad de corriente, mayor velocidad en la transferencia de datos. Esto hace que los MCU equipados con el periférico USB puedan conectarse aun periférico USB externo (puede ser una PC, portátil, etc.). Un indicador LED amarillo etiquetado como VBUS se encuentra instalado en el bloque

USB ON DEVICE este indica la conexión entre los dispositivos HOST del otro extremo.

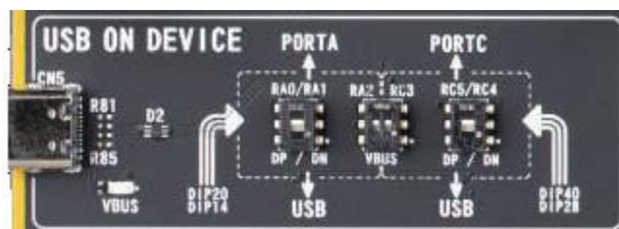


Ilustración 24 Bloque USB(DISPOSITIVO) en EasyPIC v8

En la placa de desarrollo están instalados dos interruptores DIP situados en el bloque USB ON DEVICE que permiten la configuración de los pines RA0, RA1, RC4 y RC5 para que funcionen como GPIO o líneas de datos USB. Solo admite ciertos zócalos de MCU entre ellos el DIP14, DIP20, DIP28 y DIP40.

	ON (Arriba)	OFF (Abajo)
PORTA (DIP14, DIP20)	Los pines RA0 y RA1 son configurados como GPIO	Conecta los pines RA0 y RA1 a las líneas de datos USB
PORTC (DIP28, DIP40)	Los pines RC4 y RC5 son configurados como GPIO	Conecta los pines RC4 y RC5 a las líneas de datos USB

Tabla 11 Configuración de pines para usar el periférico USB

Además, incluye un interruptor de dos polos con la etiqueta VBUS, es utilizado para especificar o configurar los pines RA2 o RC3 (depende del MCU), tanto como líneas GPIO o como fuente de alimentación para el transceptor USB interno.

	ON (Arriba)	OFF (Abajo)
RA2 (DIP14 DIP20)	Configura el pin RA2 como GPIO	El pin RA2 se conecta la línea de alimentación de 3.3V
RC3 (DIP28 DIP40)	Configura el pin RC3 como GPIO	El pin RC3 se conecta la línea de alimentación de 3.3V

Tabla 12 Configurar pines RA2 y RA3 como fuente de alimentación o GPIO

2.2.3 MICROCONTROLADORES PIC

Un microcontrolador es un circuito integrado programable, que en su interior posee una arquitectura de un computador, esto es CPU, memorias RAM, EEPROM, y circuitos de entrada y salida. [4] Este contiene los elementos necesarios para llevar el control de una o varias tareas específicas, como el control de una lavadora, un sistema de alarma, etc. Para esto se ayuda de una memoria interna donde se almacena el programa que gobierna el funcionamiento del prototipo, el mismo luego de ser configurado y programado solo sirve para realizar la tarea asignada.

Microchip Technology INC es la empresa que produce los MCU PIC, esta ocupa el primer lugar en ventas de microcontroladores de 8 bits desde el año 2002. Su gran aceptación es gracias a las buenas características que brinda el dispositivo, los bajos precios, tamaño reducido, gran velocidad, bajo consumo de potencia, abundancia de información y gran disponibilidad de herramientas para llevar a cabo su programación.



Ilustración 25 Encapsulado de un MCU PIC de 40 pines

Además, un MCU incorpora los elementos necesarios para generar el desarrollo de diferentes aplicaciones prácticas, entre ellos incorpora módulos como PWM, USART, Comparadores, convertidores A/D, fuentes de interrupción, temporizadores. En el desarrollo de esta propuesta se cubre temas fundamentales para exponer a los estudiantes su funcionamiento.

2.2.4 SENSORES

En electrónica, un sensor es conocido como un dispositivo de entrada, encargado de proporcionar una respuesta o información frente a estímulos o señales físicas, químicas o biológicas que se generan en el entorno. Son técnicamente llamados transductores ya que convierte cualquier variable física en otra señal de naturaleza distinta, comúnmente a una señal eléctrica [5].

La señal que proporciona un sensor fácilmente puede ser traducida por una unidad de control, llegando a ser esta una información muy útil para un sistema de medición. La unidad de control puede ser un MCU el cual tiene como objetivo evaluar los datos de las señales externas, y en función a ellas generar conjunto de salidas hacia un actuador o indicador [6].

Los sensores de proximidad infrarrojos constan de un emisor y un receptor conformado por un fotodiodo y un fototransistor. Estos tienen la capacidad de captar radiación electromagnética infrarroja, generada por partículas eléctricas en movimientos. La ventaja de estos dispositivos es que no necesita de un contacto físico para detectar la proximidad de un objeto, además de que trabajan en el espectro infrarrojo donde no se ven tan afectados por la luz ambiente.

Los sensores de temperatura también conocidos como termostatos, estos dispositivos son capaces de medir las propiedades térmicas del ambiente o de un cuerpo. Estos entregan una señal eléctrica de salida ya sea analógica o digital siendo esta un valor proporcional a la temperatura real [8].

2.2.5 ACTUADORES

Se denominan actuadores a todos aquellos dispositivos que obedecen o acatan las ordenes que ejecuta una unidad de control, es decir efectúan una acción en respuesta a la petición solicitada. Al igual que los sensores, existen diferentes tipos de actuadores y ellos son: Hidráulicos, Neumáticos, Eléctricos [9].

En los actuadores eléctricos la fuente de energía es la electricidad y basan su funcionamiento en el principio de que la energía eléctrica se puede transformar en energía mecánica [10].

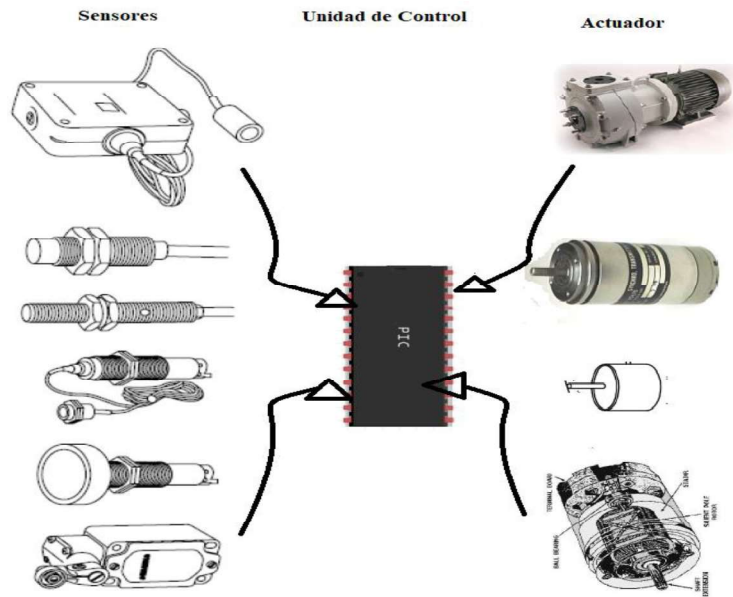


Ilustración 26 Actuador Eléctrico

2.2.6 INDICADORES

Se le conoce como indicadores a todo dispositivo que puede dar una respuesta visible hacia el ser humano. Son muy utilizados en procesos de control ya que se necesita tener ciertos indicadores para determinar el estado de un proceso, puede ser por medio una luz, una pantalla, display, un Buzzer, etc. El más común es el diodo led, siendo este el más económico y fácil de usar.

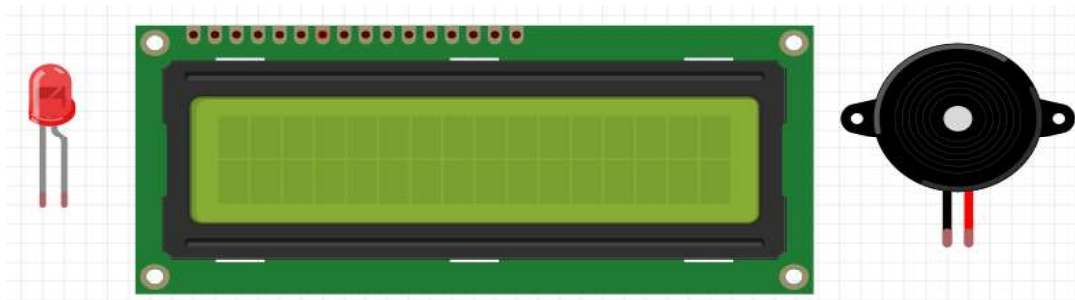


Ilustración 27 Dispositivos indicadores

2.2.7 CONVERSIÓN ANALÓGICA DIGITAL

La conversión de datos consiste en la toma de muestras del mundo real (señales analógicas) para que desde un ordenador sus datos puedan ser tratados o manipulados. Básicamente se trata de adquirir un conjunto de señales físicas utilizando una unidad de control para convertirlas en señales eléctricas y luego digitalizarlas a tal punto que se pueda procesar desde una maquina eléctrica. El dispositivo que hace como unidad de control en la presente propuesta es un MCU PIC el cual es el encargado de adquirir y digitalizar las señales entrantes que se desea medir [11].

Las propiedades físicas a medir pueden darse en diversos campos de investigación como por ejemplo la temperatura en una habitación, el valor de los infrarrojos cuando un obstáculo se encuentra cerca. Estos datos procesados digitalmente son de apoyo para hacer uso de los actuadores e indicadores para llevar a cabo el control de algún proceso.

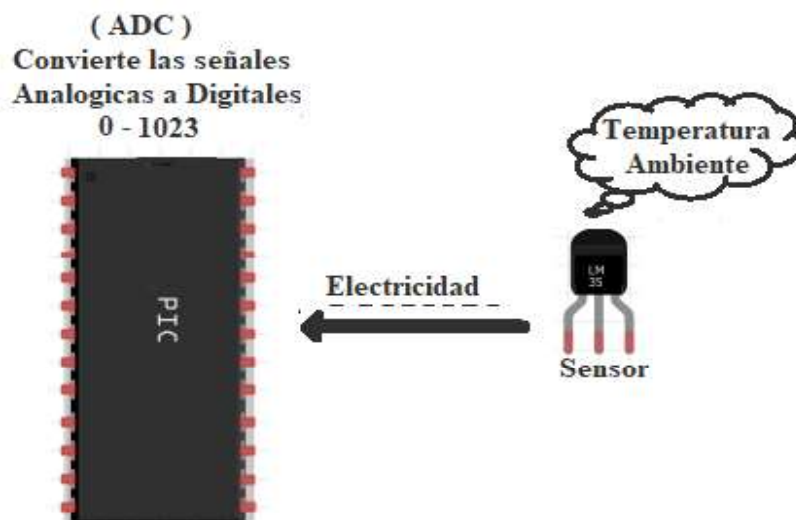


Ilustración 28 Proceso de conversión Analógica a Digital

2.2.8 PROTOCOLO DE COMUNICACIÓN

COMUNICACIÓN SERIAL

Es una interfaz de comunicación de datos digitales la cual trasmite bit a bit los datos a través de las dos líneas del MCU (RX y TX), es decir una comunicación asíncrona.

Para ser enviados, se convierten en un formato serie en donde los bits se encuentran organizados uno tras otro, de tal manera que en el extremo receptor llegara primero el bit menos significativo.

Para esta comunicación es importante que tanto emisor y receptor se encuentren sincronizados a una cierta velocidad medida en Baudios.

Existen varios modos de transmisión de datos estos son:

Simplex es una comunicación que puede darse un solo sentido, sea para recibir o transmitir.

Half-duplex es una transmisión que puede ocurrir en ambos sentidos, pero no al mismo tiempo.

Full-duplex es una transmisión que puede ocurrir en ambos sentidos y al mismo tiempo. También conocido como líneas de doble sentido.

2.3 MARCO TEÓRICO

En la siguiente sección contiene un resumen de las investigaciones realizadas, con el objetivo de conocer las diferentes publicaciones, revistas, artículos, tesis y proyectos, que proporcionaron información para el desarrollo de esta propuesta; se brinda una breve descripción de cada uno de ellos.

En Guatemala en el año 2015 se realizó el trabajo titulado: **Diseño e implementación de prácticas del laboratorio de Microcontroladores**, donde detallan el fundamento de la programación y su elemento principal el MCU PIC, hacen uso del software MikroBasic Pro For PIC. Plantean tres prácticas, donde utilizan un LCD, un motor DC y un potenciómetro, en la última práctica realizan una calculadora básica para dar a conocer las estructuras lógicas y las subrutinas que pueden aportar en el estudio de la programación.

En Bucaramanga en el año 2011 se realizó el trabajo titulado: **Diseño y fabricación de un módulo entrenador implementando la familia de microcontroladores dsPIC33F**, en el siguiente trabajo se construyó un módulo entrenador para ayudar al estudiante en el proceso de aprendizaje sobre los microcontroladores dsPIC33F, este documento contiene ayudas didácticas, hojas técnicas y manuales de uso y prácticas para interactuar con el usuario. La placa está compuesta por diodos emisores de luz, una pantalla LCD, pulsadores, switch, memoria SD, display de 7 segmentos, puerto serie, módulo bluetooth, potenciómetros y borneras de entradas. Además, está equipada con un programador para que el usuario tenga la facilidad de verificar su codificación. [12]

En Pamplona en el año 2017 se realizó el trabajo titulado: **Proyectos electrónicos con microcontroladores PIC16F877A**, en el siguiente trabajo se realizan tres proyectos electrónicos en el que cada proyecto compone de varias etapas: fase de pruebas y programación, diseño de esquema electrónico, fabricación de PCB, montaje de componentes e impresión 3D. El primer proyecto hace uso de la comunicación I2C

para simular un reloj despertador, el siguiente utiliza un joystick para interactuar con dos servomotores y un diodo laser, el ultimo es un cubo de leds RGB. [13]

En Quito en el año 2018 se realizó el trabajo titulado: **Construcción de módulos con Bluetooth, motores y sensores utilizando Arduino Mega para el laboratorio de microprocesadores de la ESFOT** en el siguiente trabajo se realizan 10 módulos didácticos, que componen de dispositivos como diodos Leds, Motores, Bluetooth, Buzzer, sensores de temperatura, distancia y luz. Este documento cuenta con un manual de prácticas la cual está compuesta de tres prácticas de laboratorio donde su elemento principal el Arduino Mega.

CAPÍTULO III

DESARROLLO DE LA PROPUESTA

3.1 COMPONENTES DE LA PROPUESTA

3.1.1 COMPONENTES LÓGICOS

A continuación, se da a conocer los softwares empleados en el desarrollo del proyecto, se describen sus características y el uso.

3.1.1.1 MIKROC PRO FOR PIC

Es un compilador de alto nivel para microcontroladores PIC de MICROCHIP creada por la empresa Mikroelectronica. Está diseñado para desarrollar, construir y depurar aplicaciones embebidas basadas en PIC. Tienen un entorno de desarrollo con una amplia variedad de características como la sintaxis en C la cual es muy fácil de aprender, IDE fácil de usar, un código muy compacto y eficiente, bibliotecas de software, depurador de hardware, la generación de archivos (.hex y .COFF).

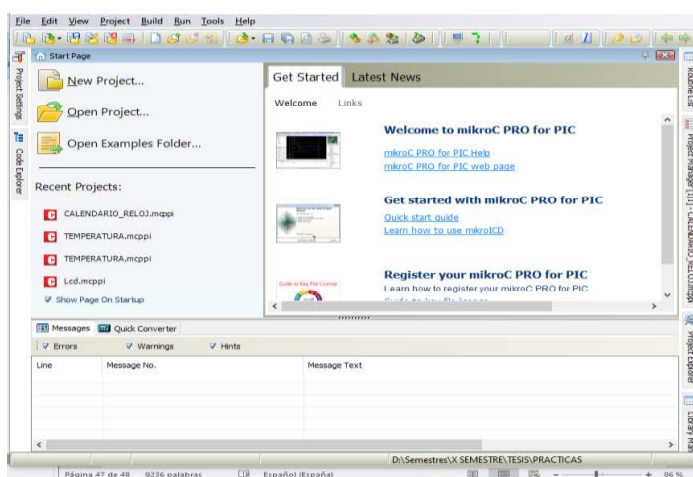


Ilustración 29 Entorno del software MikroC versión 7.6

Este es uno de los softwares que emplean para realizar la codificación y dar solución a cada una de las prácticas que se plantearan más adelante. Este me permite ir comprobando la codificación paso a paso para corregir previos errores que se presenten.

3.1.1.2 MPLAB IDE

El software MPLAB IDE es un ambiente de desarrollo integrado creada por la empresa Microchip Technology que permite la creación de proyectos basados en microcontroladores PIC. Para llevara a cabo las respectivas etapas de un proyecto el software está equipado con varios módulos que permiten la Edición, Ensamblaje, Simulación y Programación. Este entorno de desarrollo es muy amigable a la hora de realizar la programación ya que presta una interfaz muy detallada para el usuario.

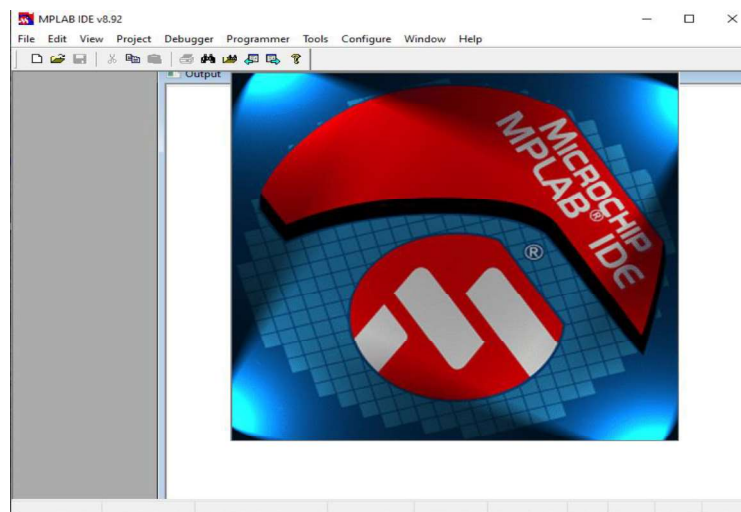


Ilustración 30 Software MPLAB IDE

Este software es utilizado para dar solución a las tres primeras prácticas, ya que permite realizar la codificación en lenguaje ensamblador. Siendo esto uno de los lenguajes de programación que sirve de base para que el estudiante pueda iniciar el estudio de cualquier otro dispositivo PIC.

3.1.1.3 PROTEUS

Proteus es una herramienta desarrollada por Labcenter Electronics que consta de dos programas principales: Ares e Isis. En Proteus Isis se puede encontrar una gran variedad de dispositivos electrónicos, como circuitos integrados, capacitores, resistencias, diodos, conectores, amplificadores, etc. En donde se puede construir y simular circuitos electrónicos para verificar su funcionamiento antes de realizar la implementación física del prototipo. En Proteus Ares permite construir esquemas electrónicos o PCB con las medidas adecuadas de cada elemento que se necesite. Una herramienta muy utilizada por los estudiantes de ingeniería en electrónica ya que permite simular previamente sus prototipos electrónicos antes de su implementación.



Ilustración 31 Software Proteus

Por medio de ISIS Proteus se procedió realizar el esquema electrónico y a simular cada una de las aplicaciones prácticas planteadas, ya que este incorpora un entorno completo de los elementos a utilizar.

3.1.1.4 MATLAB

MATLAB viene de la abreviatura “Matrix Laboratory”, laboratorio matricial. Es un entorno de computación orientado para llevar a cabo proyectos que impliquen cálculos matemáticos elevados y la visualización gráfica de los mismos. MATLAB es capaz de

realizar las mismas funciones que una calculadora científica tales como el análisis numérico, calculo matricial, proceso de señales y visualización gráfica en un entorno completo. Este maneja un lenguaje de programación de alto nivel, es una herramienta muy utilizada por los ingenieros y científicos, dado que MATLAB incorpora diferentes modos de trabajo y es muy fácil de usar ya que presenta una interfaz amigable hacia los usuarios.

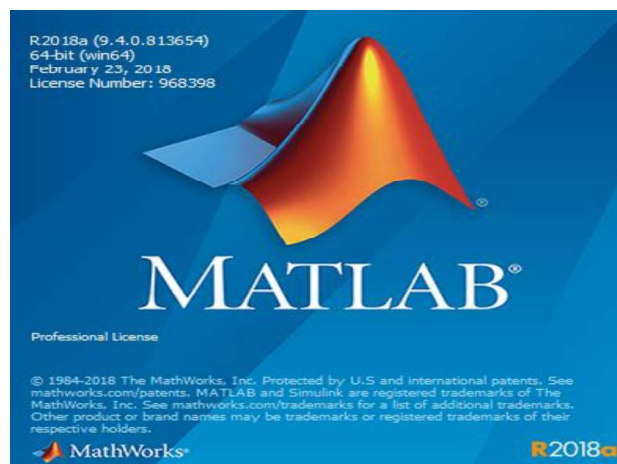


Ilustración 32 Software MATLAB

Se hace uso de la herramienta MATLAB GUIDE, para crear interfaces gráficas y permitir una interacción activa con el usuario. Es un programa muy útil cuando se tiene en la entrada un ingreso continuo de datos, ya que estos pueden ser manipulados fácilmente desde la plataforma. Desde la interfaz gráfica se establecerá una comunicación serial entre el software MATLAB y la placa EasyPIC v8 cabe debe mencionar que se desarrollaran prácticas que emitan y reciban datos.

3.1.1.5 CODEGRIP

Codegrip Suite es un software que permite llevar el control completo sobre la placa de desarrollo EasyPIC v8. Se utiliza para cargar las tareas de programación el archivo (.hex) de forma inteligente, así como otras opciones de configuración de la placa de desarrollo. Es una de las primeras placas que ofrece una depuración a través de WI-FI siendo esta una de las características que revoluciona la forma en que se desarrollan las

aplicaciones prácticas. Está Incorpora un módulo CODEGRIP integrado en la placa de desarrollo el mismo que se utiliza un conector USB-C para una conexión confiable y segura con la computadora personal.

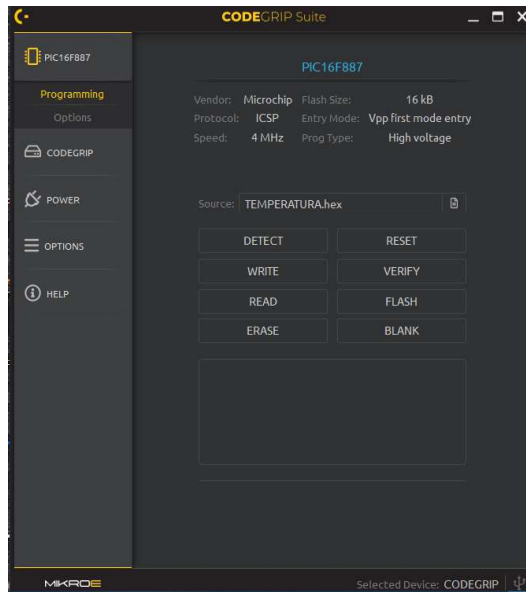


Ilustración 33 Software CODEGRIP

El software CODEGRIP SUITE se utiliza para grabar la información en el MCU PIC, para ello es necesario el archivo (.hex) de la codificación realizada previamente.

3.1.2 COMPONENTES FÍSICOS

A continuación, se detalla los componentes físicos utilizados en el desarrollo de esta propuesta, cada uno con sus características y uso.

3.1.2.1 MÓDULO EASYPIC v8

La creación rápida de prototipos permite al ingeniero tomar la manera más eficiente y sin esfuerzo, para visualizar las ideas, capacidades y limitaciones de diseño de cada proyecto a realizar. Esta placa de desarrollo soporta 273 MCU de 8 bits de Microchip desde los MCU más pequeños con solo 8 pines hasta los más gigantes de 40 pines, donde los sócalos DIP permiten la ubicación exclusiva de los MCU. Posee un diseño

robusto y compacto con una mejor calidad de las tomas y un diseño ergonómico, que permite un desarrollo cómodo de proyectos, además con dos conectores de pantallas entre ellas la LCD 2x16 y la GLCD 128x64 capaces de mostrar contenidos gráficos avanzados, interruptores, pulsadores, indicadores leds, extensor de puertos, comunicaciones USB. Ofrece cinco sockets mikroBUS mejorados, donde puede colocar cualquiera de los más de 650 click boards diferentes (ver Ilustración 34). Estas tarjetas listas para usar no requieren una configuración de hardware adicional, ahorran a los desarrolladores las pruebas y la resolución de problemas a menudo asociados con la fase de creación de prototipos. Posee un suministro de energía de primera clase que entrega una energía constante, diseñados para filtrar y distribuir la energía sin ruido. Se encuentra respaldada por un poderoso software CODEGRIP Suite, que ofrece un control completo sobre la placa de desarrollo. Se utiliza para administrar de manera inteligente las tareas de programación y depuración ya sea por medio USB, o WI-FI [2].

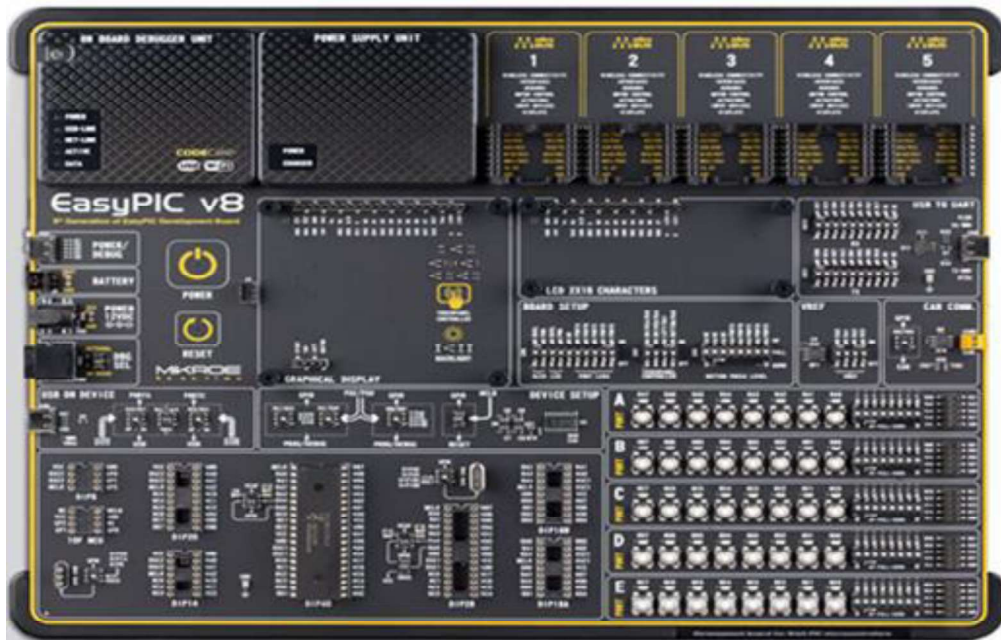


Ilustración 34 EasyPIC v8

3.1.2.2 MICROCONTROLADOR PIC 16F887

El MCU PIC 16F887 es utilizado en todas las aplicaciones prácticas que se han planteado, en particular el modelo de 40 pines con encapsulado PDIP. Este es el dispositivo central en el desarrollo de las prácticas, ya que en él se graba la codificación relacionada a los dos lenguajes de programación (C y ensamblador). Entre sus principales características tenemos (ver Tabla 13):

CARACTERÍSTICAS	
Arquitectura	RISC
Tamaño	18x51mm
Pines	40
Frecuencia de Operación	0 - 20MHz
Oscilador interno	8MHz – 31KHz selecto por software
Voltaje de Alimentación	2.0V a 5.5V
Memoria ROM	8K
Memoria EEPROM	256 Bytes
Memoria RAM	368 Bytes
Pines de propósito general	35
Pines Analógicos	14 (resolución 10 bits)
Pines PWM	4
Módulo I2C, SPI	
Módulo USART	

Tabla 13 Datos técnicos PIC 16F887

Este MCU incorpora varias interfaces de comunicación y la que se utiliza es la comunicación serial, haciendo uso de los pines RX y TX. Con objetivo de recibir o transferir datos hacia una PC, además de poder visualizar sus datos por el puerto COM interactuando desde la plataforma de MATLAB GUIDE.

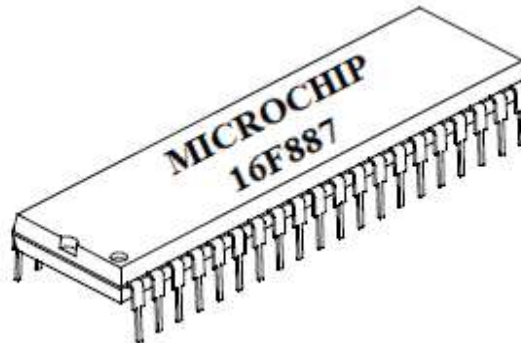


Ilustración 35 MCU PIC 16F887

3.1.2.3 SENSOR SHARP 2Y0A21

El sensor SHARP es utilizado para medir distancias entre él y un objeto, está compuesto por un emisor infrarrojo y un receptor. Calcula la distancia utilizando el método de triangulación el cual consiste en medir el ángulo que se forma entre el emisor – objeto – receptor, el receptor detecta el punto de incidencia y a su vez la distancia del objeto (ver Ilustración 36) [14].



Ilustración 36 Sensor SHARP 2Y02A1 (10-80cm) [15]

El sensor tiene un rango de medición de 10 a 80cm y entrega una salida tipo analógica. El MCU utilizado tiene un conversor ADC con resolución de 10 bits entonces obtendremos valores de 0 a 1024, si transformamos a valores de voltaje tendríamos de 0 a 5V. En la siguiente Ilustración 36 podemos observar el funcionamiento del sensor (Voltaje vs Distancia).

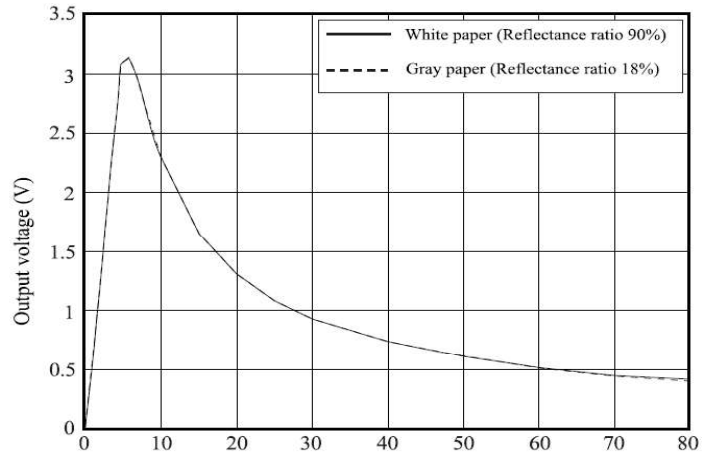


Ilustración 37 Características del sensor Voltaje vs Distancia [15]

Según la ilustración 39, el sensor tiene un buen funcionamiento a partir de los 10cm. A continuación, se detalla las características técnicas del sensor (ver Tabla 14).

CARACTERÍSTICAS	
Rango de medición	10 a 80cm
Voltaje de alimentación	4.5V a 5.5V
Consumo de corriente	30mA
Salida tipo	Analógica
Tamaño	29.5x13x13.5 mm

Tabla 14 Detalles técnicos del sensor SHARP

3.1.2.4 SENSOR LM35

El sensor LM35 (ver Ilustración 38) es la precisión de circuitos integrados capaz de medir temperatura, y entregar una tensión proporcional a la temperatura en °C. A continuación, se presenta algunas de las características de este dispositivo (ver Tabla 15).



Ilustración 38 Sensor LM35

CARACTERÍSTICAS	
Rango de medición	-55 a +150°C
Voltaje de alimentación	4 a 20V
Consumo de corriente	60μA
Salida tipo	Analógico
Factor de escala lineal	10mV/°C

Tabla 15 Detalles técnicos del sensor LM35

En la Ilustración 39 observamos el circuito básico para obtener la temperatura en el rango establecido. La salida del sensor es conectada a un pin analógico del MCU, este internamente lo convierte a digital presentado al usuario valores de lectura de 0 a 1023, de acuerdo al voltaje que proporcione el sensor. El valor del voltaje recibido fluctúa entre 0 y 5V. Para calcular el valor en [mV] por cada unidad entregada por el convertidor ADC realizamos la siguiente ecuación.

$$valor = \frac{5}{1024} = 4.8828[mV]$$

A este valor le multiplicamos por la lectura recibida del `ADC_read()`

$$[mV]leidos = 4.8828[mV] * ADC_read$$

De acuerdo a las características del sensor nos dice que por cada 10mV hay un 1°C entonces la ecuación final nos queda:

$$^{\circ}C = \frac{4.8828[mV] * ADC_read()}{10[mV]} \quad \therefore \quad ^{\circ}C = \frac{5.0 * adc_read() * 100}{1024}$$

En resumen, cualquiera de las dos ecuaciones se puede ubicar en el código para obtener los datos de temperatura en grados Celsius.

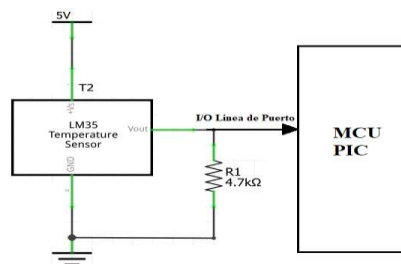


Ilustración 39 Circuito básico de conexión

3.1.2.5 MOTOR DC

Un motor de corriente continua es capaz de transformar la energía eléctrica (corriente continua), en energía mecánica (ver Ilustración 40). Su funcionamiento se basa en la fuerza electromagnética que se produce sobre un conductor cuando se aplica una corriente eléctrica. Estos dispositivos se los puede encontrar en los juguetes a pilas, que están constituidos por dos imanes fijos a la carcasa y un bobinado de cobre en el eje del motor. Este elemento forma parte de los actuadores, y para su tener su debido control se necesita de un controlador o Driver, en las prácticas 6 y 7 se ha optado por el controlador L298N (Puente H).



Ilustración 40 Motorreductor 75:1 HP 6V

El motorreductor mostrado en la ilustración 42, es un motor de corriente continua de 6V de alta potencia con una caja de engranajes de metal 75.81: 1 y el eje en su salida en forma de D con una longitud de 3x9mm. El siguiente cuadro muestra las características básicas del componente (ver Tabla 41):

CARACTERÍSTICAS	
Voltaje	6V
Reductora	75: 1
Rendimiento sin carga	410 RPM
Corriente	70mA
Dimensiones	10x12x36mm
Peso	1.3 kg * cm

Ilustración 41 Características técnicas del motorreductor

3.1.2.6 CONTROLADOR L298N

El módulo L298N es utilizado para controlar motores de corriente continua de hasta 2 amperios. Cuenta con diodos de protección y un regulador **LM7805** que provee con 5V ajustados a la parte lógica del chip L298N. Permite conectar hasta dos motores DC: las salidas se encuentran etiquetadas como OUT1 y OUT2 para el motor A, OUT3 y OUT4 para el motor B (ver Ilustración 42).



Ilustración 42 L298N controlador de motores DC

In1	In2	In3	In4	Salida motor A	Salida motor B
0	0	0	0	Detenido	Detenido
0	0	0	1	Detenido	Antihorario
0	0	1	0	Detenido	Horario
0	1	0	0	Antihorario	Detenido
0	1	0	1	Antihorario	Antihorario
0	1	1	0	Antihorario	Horario
1	0	0	0	Horario	Detenido
1	0	0	1	Horario	Antihorario
1	0	1	0	Horario	Horario
ENA	0 – 255				
ENB	0 - 255				

Tabla 16 Tabla de verdad para control de dos motores

La Tabla 16 nos indica los estados que se pueden dar al controlador L298N, para llevar a cabo el control de dos motores DC. Las etiquetas In1, In2, In3 e In4 son pines de digitales de entradas las cuales se conectan a un MCU para llevar el control de los estados mediante líneas de código. ENA y ENB son pines de entradas utilizados para variar la velocidad de los motores, desde el MCU se puede aplicar una señal PWM con valores desde 0 a 255, siendo 255 la velocidad máxima que puede alcanzar el motor. A continuación, se presenta algunas de las características principales del dispositivo L298N:

CARACTERÍSTICAS	
Rango de Alimentación	4.8V a 46V
Corriente máxima	2A
Temperatura de funcionamiento	-25 a 130°C
Dimensiones	47x53mm
Peso	29g

Tabla 17 Características principales del L298N

La Ilustración 43 nos muestra un esquema que conecta a un MCU, para que este pueda llevar el control sobre los motores.

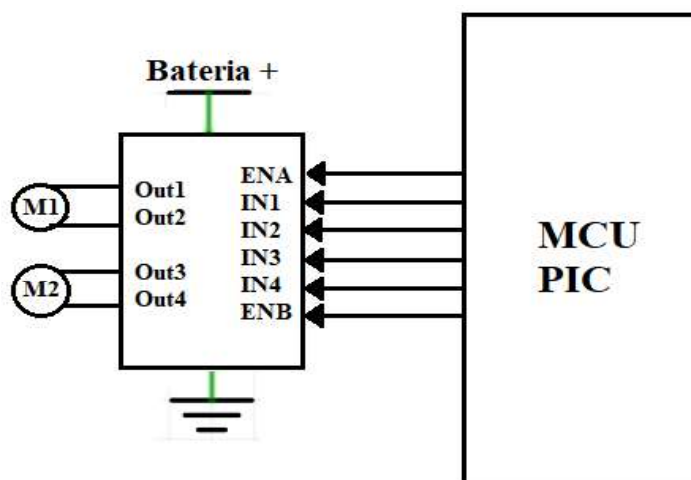


Ilustración 43 Bloque de conexiones L298N y MCU

3.1.2.7 LCD 2X16

Pantalla de Cristal Líquido más conocida como LCD, utilizado para visualizar mensajes, letras griegas, signos de puntuación, símbolos matemáticos, etc. Este elemento está específicamente fabricado para ser utilizado con los MCU. Es muy utilizado en el desarrollo de aplicaciones prácticas ya que es económico y muy completo para la visualización de textos en su pantalla (Ver Ilustración 44) [16].

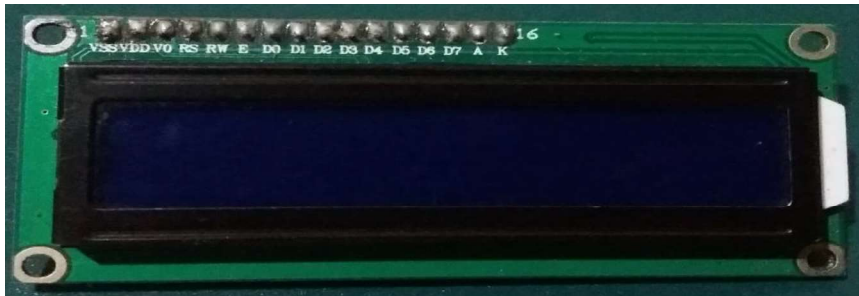


Ilustración 44 Pantalla de Cristal Líquido LCD 2x16

La pantalla LCD cuenta con 16 pines, cada uno con su etiqueta correspondiente. A continuación, se detalla la función que cumplen (ver Tabla 18).

Función	Etiqueta	Estado lógico	Descripción
Tierra	Vss	-	0V
Alimentación	Vdd	-	+5V
Contraste	Vo	-	0 – Vdd
Control de Funcionamiento	RS	0	D0-D7 Trabajan como comandos
		1	D0-D7 trabajan como datos
	RW	0	Modo escribir datos al LCD
		1	Modo leer datos del LCD
	E	0	Deshabilita acceso al visualizador
		1	Modo normal
	D0	1/0	Bit 0 LSB
	D1	1/0	Bit 1

Datos/comandos	D2	1/0	Bit 2
	D3	1/0	Bit 3
	D4	1/0	Bit 4
	D5	1/0	Bit 5
	D6	1/0	Bit 6
	D7	1/0	Bit 7 MSB

Tabla 18 Pines funcionamiento en LCD

En las conexiones no hay de qué preocuparse porque el módulo EasyPIC v8 incorpora los pines para conectar una pantalla LCD 2x16, lo que resta por realizar, es la programación adecuada para trabajar con ella.

A continuación, se indica algunas características que presenta el display LCD 2x16:

- Contiene 16 columnas y 2 filas (abarca 16 caracteres por fila)
- Encendido y apagado del display
- Parpadeo del cursor
- Retorna al inicio el cursor
- Desplazamientos de izquierda a derecha
- Reconoce caracteres ASCII

3.1.2.8 DISPLAY DE 7 SEGMENTOS

Un display de siete segmentos es un dispositivo digital de salida que es empleado para representar valores numéricos y ciertas letras del alfabeto (ilustración 45). Este se comercializa en dos configuraciones posibles: ánodo común o cátodo común. Los segmentos del display pueden ser controlados desde un MCU, para ello se demuestra su funcionamiento en la práctica dos en donde se hace uso del puerto D del MCU PIC 16F887. De acuerdo a las configuraciones del display las conexiones hacia el MCU varia [17].



Ilustración 45 Display de siete segmentos

Para gobernar la información que se va presentar en el display de siete segmentos, se hace uso de tablas, es ahí donde ubicaremos un valor hexadecimal para indicar que diodo LED debe activarse. La tabla ubicada en el Anexo B indica los segmentos que deben activarse para representar cada uno de los caracteres posibles [17].

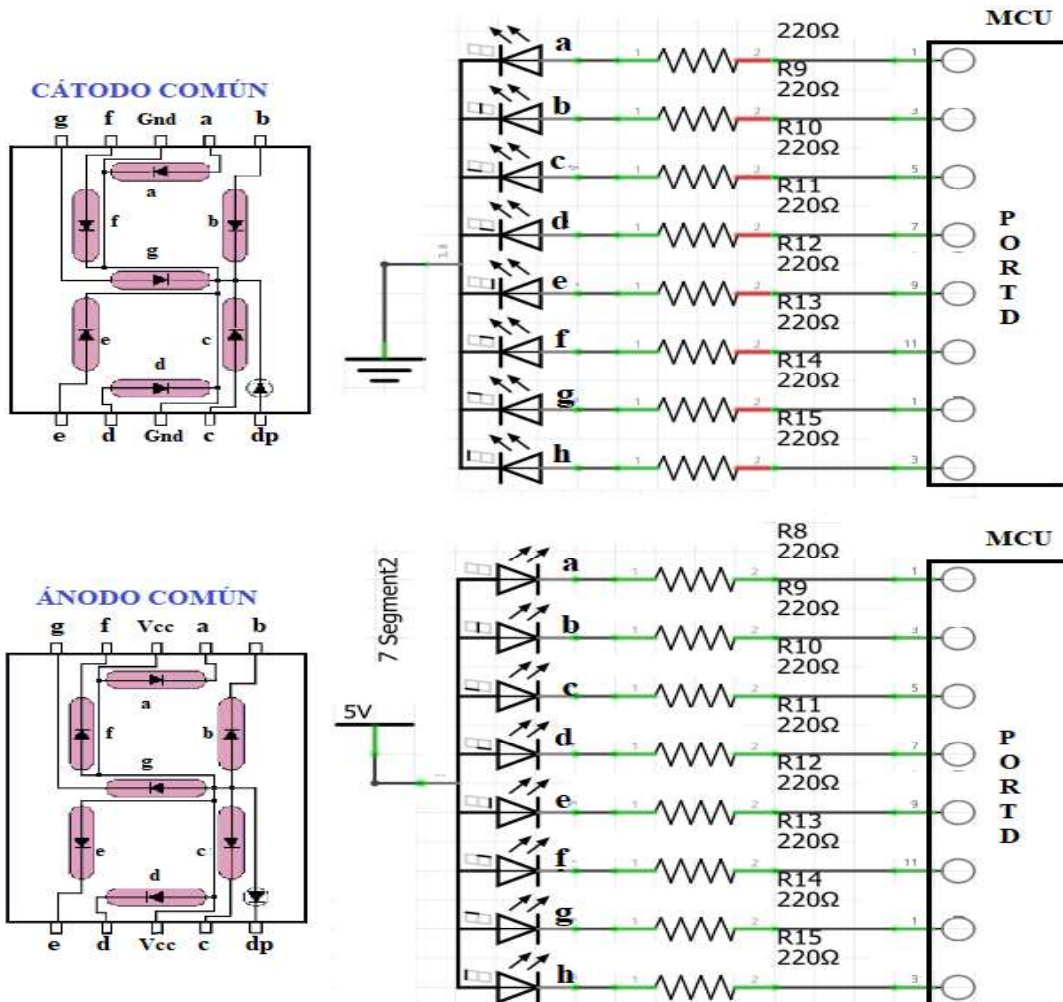


Ilustración 46 Configuración Ánodo y Cátodo Común

3.1.2.9 RESISTENCIAS

La resistencia es un dispositivo físico que puede adquirirse con ciertos valores estándar en las tiendas electrónicas. Son utilizadas para limitar el valor de la corriente a un valor apropiado para su funcionamiento. En el desarrollo de la aplicación práctica tres, se hace uso de las resistencias, estas ayudan a atenuar el voltaje y la corriente que va hacia los diodos LEDs del display de siete segmentos. En el display para proteger los LEDs se utilizan resistencias entre 220Ω y 330Ω , estos limitan la corriente a unos 10mA, generando una luminosidad suficiente en los diodos LEDs.

3.2 GUÍA DE APLICACIONES PRÁCTICAS

En el desarrollo de la guía de aplicaciones prácticas se utiliza la placa de desarrollo **EASYPIC v8** de la empresa MikroElektronika, junto con el software **CODEGRIP SUITE** el cual es el encargado de grabar la información en el microcontrolador. Para realizar la codificación se emplea dos softwares, entre ellos: **MPLAB** este maneja el lenguaje ensamblador utilizado en las tres primeras prácticas, **MikroC** en cambio maneja un lenguaje de alto nivel como es el C, utilizado a partir de la práctica 4. Luego de culminar con la programación, es necesario realizar el proceso de simulación para verificar la programación y presenciar los errores que pueden darse en la implementación, para ello se utiliza el software **PROTEUS**. Además, se hace uso de la herramienta **MATLAB GUIDE**, utilizada para diseñar interfaces gráficas que permiten una interacción constante entre los datos y el usuario.

A continuación, se exponen las siete aplicaciones prácticas, donde consta de una breve explicación del tema a tratar, sus objetivos, los materiales y la descripción de la práctica. Mas a detalle de cada una de las prácticas se encuentra en el ANEXO A del documento.

3.2.1 PRÁCTICA 1:

Compilación y grabación del microcontrolador para familiarizar al usuario con el módulo EasyPIC v8

Objetivo de la práctica

Facilitar al estudiante el desenlace del funcionamiento de cada herramienta empleada tanto el hardware como software que me permita compilar y grabar el microcontrolador.

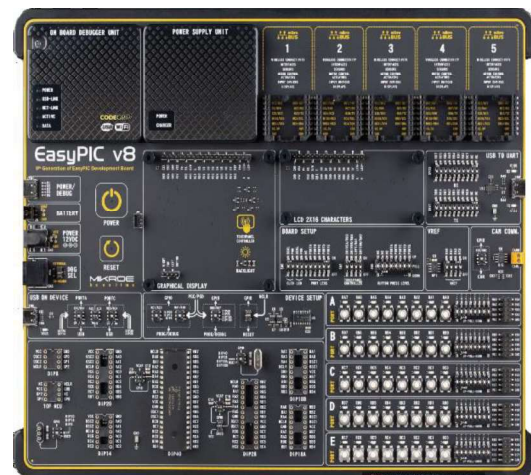
Objetivos específicos

- Aprender a utilizar el software MPLAB IDE para compilar el microcontrolador.
- Aprender a utilizar el software PROTEUS para simular circuito electrónico.
- Aprender a utilizar el software CODEGRIP SUITE para grabar la codificación en el microcontrolador mediante USB-C y WIFI.
- Manipular periféricos de entrada y salidas del módulo EasyPIC v8
- Tendrá la capacidad y habilidad para diseñar el algoritmo, editar el programa en el software MPLAB IDE, copiarlo, depurarlo y simular el comportamiento en el simulador PROTEUS.

Materiales de trabajo

HARDWARE	
1	Tarjeta de desarrollo EasyPic v8
2	Microcontrolador PIC 16F887
3	Conector Tipo C – USB
4	Fuente de alimentación 12V

SOFTWARE	
1	MPLAB
2	PROTEUS
3	CODEGRIP SUITE



Descripción de la práctica

La presente práctica se diseñó para que los usuarios aprendan de forma correcta el funcionamiento de la placa EasyPIC v8, el manejo del software de programación MPLAB, y correcta simulación desde ISIS Proteus.

La práctica consta de los siguientes puntos:

- Preparar el software MPLAB (Crear un nuevo proyecto y el compilar código)

Código de prueba desarrollado para el PIC 16F887 el cual tiene como finalidad manejar los puertos de entrada y salidas del MCU. Se utiliza el switch ubicado en el puerto A y los leds del Puerto B. En el puerto B se presentará el dato adquirido de las seis líneas del puerto A al que están conectado un arreglo de interruptores. Por ejemplo, si se introduce 111000 por el puerto A, tendré una salida del mismo valor en el puerto B.

La siguiente programación esta específicamente escrita para el PIC 16F887, donde las cinco primeras líneas son relevantes para cargar la configuración de inicio del microcontrolador.

```
LIST P=16F887
INCLUDE "P16f887.INC"

__CONFIG _CONFIG1, _FOSC_INTRC_NOCLKOUT & _WDTE_OFF & _PWRTE_ON & _MCLRE_OFF
& _CP_OFF & _CPD_OFF & _BOREN_ON & _IESO_ON & _FCMEN_ON & _LVP_OFF
__CONFIG _CONFIG2, _BOR4V_BOR21V & _WRT_OFF

    ORG    0x00                ;Inicia en la direccion 0;
    bsf    STATUS,RP0        ;Pone un 1 en el bit 5 del registro STATUS, acceso
al Banco 1

    BANKSEL ANSEL            ;Habilita los bancos digitales
    CLRF   ANSEL

    BANKSEL PORTA
    CLRF   PORTA            ;Limpia PORTA
    CLRF   PORTB            ;Limpia PORTB
    CLRF   PORTC            ;Limpia PORTC
    CLRF   PORTD            ;Limpia PORTD
    CLRF   PORTE            ;Limpia PORTE
```

```

    BANKSEL TRISA      ;Configura puerto A como digital
    CLRF    TRISD      ;Configuro PUERTO D como salida
    BANKSEL PORTA

Inicio:
    MOVLW   b'00111111' ;Asigno las seis lineas del puerto A a "w"
    MOVWF   TRISA       ;Las seis lineas del puerto A se configuran como
entradas
    BCF     STATUS,RP0  ;Pone a 0 el bit 5 del registro STATUS, acceso al
Banco 1

principal
    MOVF    PORTA,W      ;cargó los datos del puerto A en w
    MOVWF   PORTD       ;Presento el dato de "w", en el puerto D
    goto    principal   ;Bucle infinito
END

```

Desarrollo

- Simular proyecto en ISIS Proteus
- Preparar módulo EasyPIC v8 para grabar microcontrolador
- Grabación a través de USB
- Grabación a través de WIFI

Nota

Las hojas guía de las aplicaciones prácticas se encuentran en el Anexo A, incluido los códigos de programación de los respectivos softwares utilizados.

3.2.2 PRÁCTICA 2:

Control de un depósito de líquidos mediante microcontrolador PIC16F887

Objetivo de la práctica

Desarrollar el código de microcontrolador y simular el control de nivel de un depósito de líquidos, empleando el software MPLAB y la placa de desarrollo EASYPIC v8, para contribuir al estudio del lenguaje ensamblador y sus instrucciones.

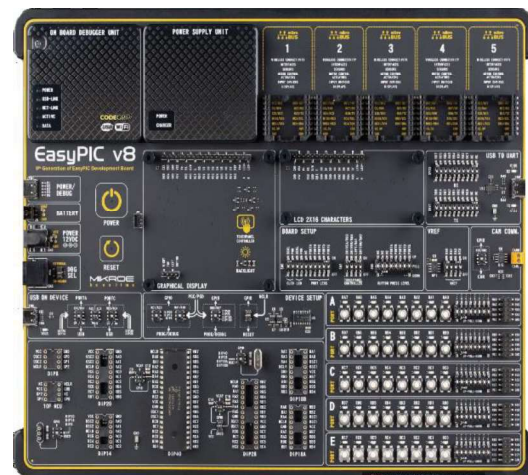
Objetivos específicos

- Diseñar el programa de solución mediante diagramas de flujo
- Desarrollar codificación en software MPLAB que permita el control del prototipo.
- Simular el proyecto utilizando software ISIS Proteus para comprobar su funcionamiento
- Grabar el microcontrolador a través de Wifi empleando CODEGRIP Suite
- Analizar y comprobar el funcionamiento físico empleando placa de desarrollo EasyPIC

Materiales de trabajo

HARDWARE	
1	Tarjeta de desarrollo EasyPic v8
2	Microcontrolador PIC 16F887
3	Conector Tipo C – USB
4	Fuente de alimentación 12V

SOFTWARE	
1	MPLAB
2	PROTEUS
3	CODEGRIP SUITE



Descripción de la práctica

Diseñar un programa que permita llevar el control del nivel de agua, del depósito de la figura.

Consta de tres sensores detectores de agua:

1. **SV**, Sensor que indica que el depósito está vacío y se encuentra conectado al pin RA0
2. **SLL**, Sensor que indica que el depósito se está llenando y se encuentra conectado al pin RA1
3. **SR**, Sensor que indica que el depósito está rebosando y está conectado al pin RA2.

Además de dos bombas de agua, con las etiquetas B1 y B2 conectadas a los pines RB5 y RB6 respectivamente.

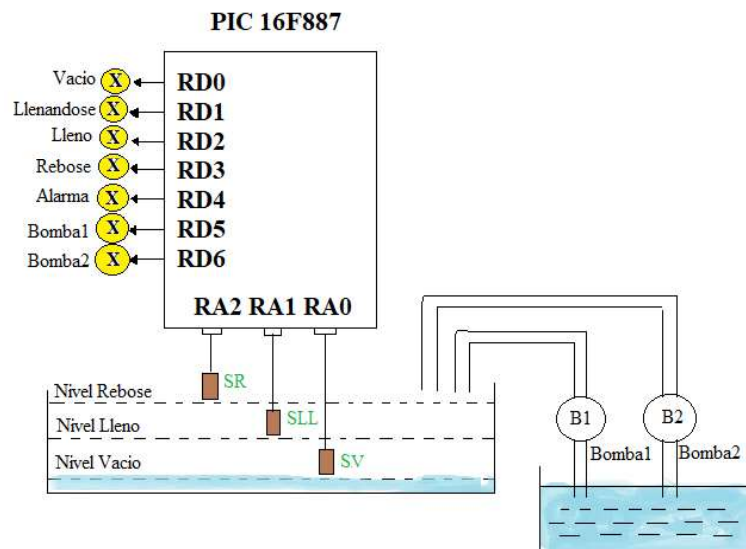
Contiene cinco indicadores que servirán para visualizar el control:

- El indicador conectado al pin RB0 nos enseña que el depósito está vacío.
- El indicador conectado al pin RB1 nos enseña que el depósito está llenándose.
- El indicador conectado al pin RB2 nos enseña que el depósito está lleno.
- El indicador conectado al pin RB3 nos enseña que el depósito está en rebose.
- El indicador conectado al pin RB4 nos enseña que existe algo irregular en el depósito “Alarma”.

Su funcionamiento:

- Cuando ninguno de los sensores se encuentra mojado se entiende que el depósito está vacío y se accionarán las dos bombas. El indicador “Vacío” se iluminará.
- Cuando el nivel del líquido toque el Sensor de vacío “SV” seguirá llenándose el depósito con las dos Bombas. El indicador “Llenándose” se iluminará.

- Cuando el nivel del líquido toque el sensor de llenado “SLL” se para la bomba B2, quedando la bomba B1 activada en modo mantenimiento. El indicador “Lleno” se ilumina.
- Si el nivel del líquido moja el sensor de rebose “SR” se apaga también la bomba B1, quedando las dos bombas fuera de servicio. El indicador “Rebose” se enciende.
- Cuando se produce un fallo o mal funcionamiento en los sensores de entrada (por ejemplo, que se active el sensor de rebose y no la de vacío) se para las dos bombas. El indicador “Alarma” se iluminará.



Desarrollo

- Configuración de entradas y salidas
- Construcción de tabla de verdad
- Realizar código en software MPLAB
- Diagrama de flujo
- Simulación en ISIS Proteus
- Resultados en el módulo EasyPIC v8

3.2.3 PRÁCTICA 3:

Contador de vehículos para estacionamiento

Objetivo de la práctica

Desarrollar un código de microcontroladores para simular el conteo de vehículos en la entrada de un estacionamiento, empleando el software MPLAB y la placa de desarrollo EASYPIC.

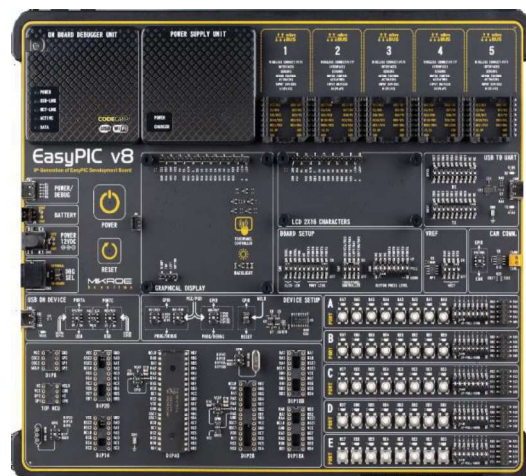
Objetivos específicos

- Diseñar el programa de solución mediante diagramas de flujo
- Desarrollar codificación en software MPLAB que permita el control del prototipo.
- Simular el proyecto utilizando software ISIS Proteus para comprobar su funcionamiento
- Grabar el microcontrolador a través de Wifi empleando CODEGRIP Suite
- Analizar y comprobar el funcionamiento físico empleando placa de desarrollo EasyPIC

Materiales de trabajo

HARDWARE	
1	Tarjeta de desarrollo EasyPic v8
2	Microcontrolador PIC 16F887
3	Conector Tipo C – USB
4	Fuente de alimentación 12V
5	Display de 7 segmentos

SOFTWARE	
1	MPLAB
2	PROTEUS
3	CODEGRIP SUITE



Descripción de la práctica

Diseñar un programa que permita llevar el conteo en la entrada de un estacionamiento.

- Consta de un pulsador conectado al Pin RA1. Si es presionado significa que un vehículo a ingresado.
- Un indicador LED conectado al Pin RB1, nos indica que el estacionamiento está lleno.
- Un display de 7 segmentos conectado en las siete líneas del puerto D, mostrara la cantidad de vehículos que ha ingresado.

Su funcionamiento:

El pulsador es utilizado para aplicar un nivel lógico “bajo” o “alto” dependiendo si es presionado o no. Al presionar el pulsador el microcontrolador reconoce un estado lógico “0” por el pin RA1, es decir que ha ingresado un vehículo y se incrementa un contador el cual es visualizado en un display de siete segmentos. El estacionamiento solo tiene espacio para nueve vehículos. Si el estacionamiento está lleno el LED conectado al pin RB1 se encenderá para indicar que el estacionamiento está lleno.

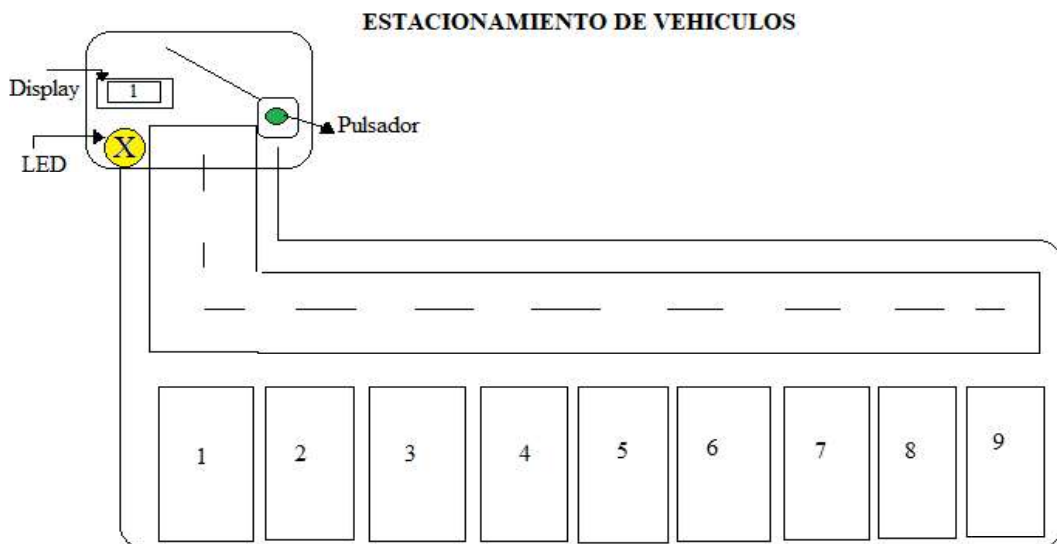


Ilustración 48 Estacionamiento de vehículos

3.2.4 PRACTICA 4:

Control de PIC16F887 desde MATLAB

Objetivo de la práctica

Desarrollar el código de microcontroladores para manipular los puertos del microcontrolador PIC 16F887, empleando el software MikroC, MATLAB y la placa de desarrollo EasyPIC v8.

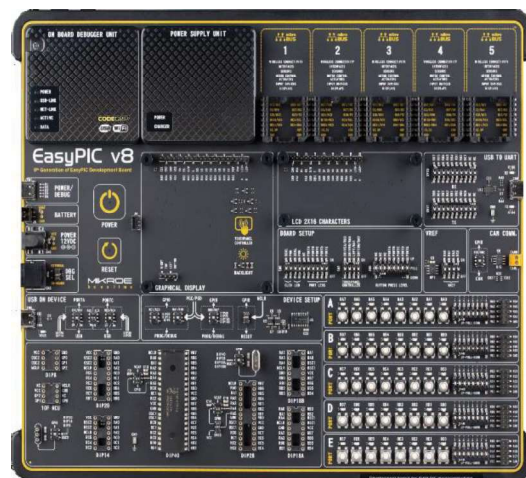
Objetivos específicos

- Desarrollar codificación en software MikroC que permita el control del prototipo.
- Simular el proyecto utilizando software ISIS Proteus para comprobar su funcionamiento.
- Diseñar el programa de solución mediante diagramas de flujo
- Diseñar interfaz de usuario (GUIDE) empleando MATLAB para manipulación de puertos.
- Grabar el microcontrolador empleando CODEGRIP Suite
- Analizar y comprobar el funcionamiento físico empleando placa de desarrollo EasyPIC

Materiales de trabajo

HARDWARE	
1	Tarjeta de desarrollo EasyPic v8
2	Microcontrolador PIC 16F887
3	Conector Tipo C – USB
4	Fuente de alimentación 12V
5	LCD 2x16

SOFTWARE	
1	MATLAB
2	PROTEUS
3	MIKROC PRO FOR PIC
4	CODEGRIP SUITE



Descripción de la práctica

Se requiere manipular los puertos del MCU PIC16F887 empleando software MATLAB.

Desde MATLAB, crear una interfaz gráfica de usuario GUIDE que me permita establecer una comunicación UART entre el MCU y la PC, se necesita diseñar mediante botones o cualquier otra herramienta que sirva de ayuda para el desarrollo de la práctica. En el software MikroC Pro For PIC crear la codificación para el MCU a utilizar, debe ser capaz de recibir las ordenes que son transmitidas desde la plataforma de Matlab, y generar la acción respectiva. Configurar los pines de los puertos RA y RD del MCU de tal forma que se puedan manipular independientemente desde MATLAB.

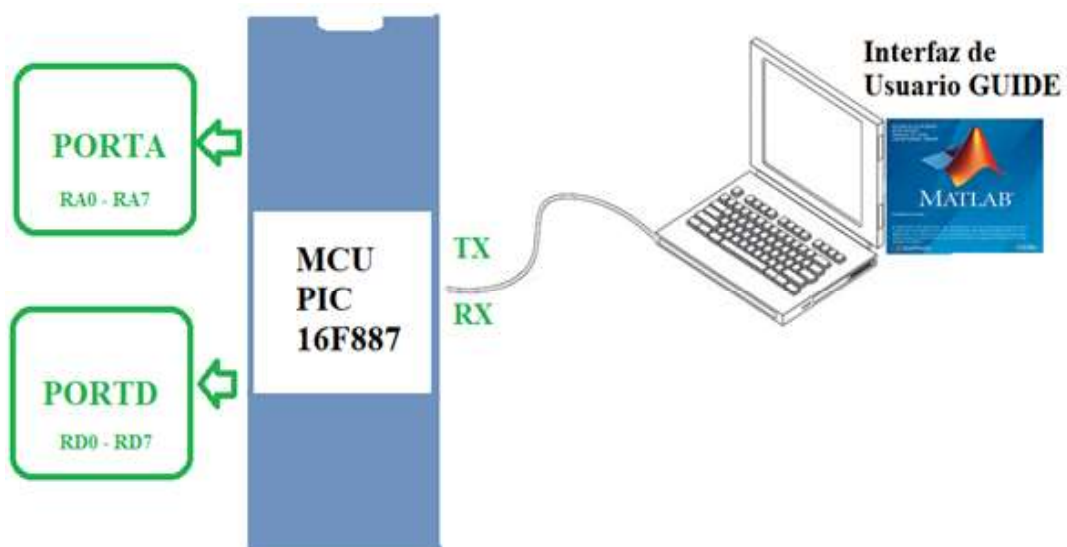


Ilustración 49 Práctica #4 esquema representativo de conexiones

Además, incorporar un botón desde MATLAB que sirva para demostración de los indicadores LEDS y la pantalla LCD. Al presionar el botón debe presentar un mensaje de bienvenida en la pantalla LCD y dando apertura a la práctica 4. Luego de unos segundos debe realizar una secuencia de luces que enciendan de izquierda a derecha de todos los puertos.

3.2.5 PRÁCTICA 5:

Control de proximidad para acceso a puerta eléctrica

Objetivo de la práctica

Desarrollar un código de microcontroladores para simular una puerta eléctrica automática, empleando un sensor de distancia, software MikroC e interfaces graficas de usuario en MATLAB.

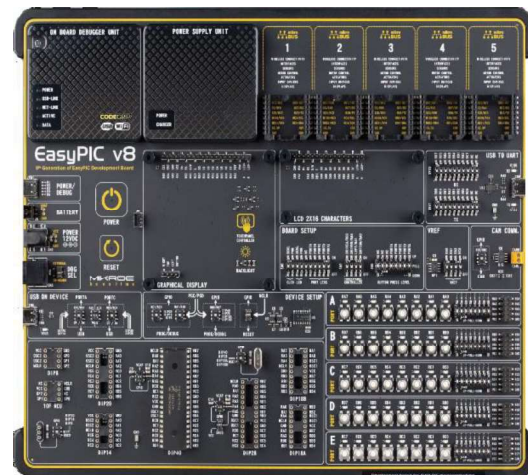
Objetivos específicos

- Desarrollar codificación en software MikroC que permita el control del prototipo.
- Simular el proyecto utilizando software ISIS Proteus para comprobar su funcionamiento.
- Desarrollar el diagrama de flujo para representar la lógica de programación.
- Diseñar interfaz gráfica de usuario (GUI) empleando comunicación UART para el ingreso continuo de datos.
- Grabar el microcontrolador empleando CODEGRIP Suite
- Analizar y comprobar el funcionamiento físico empleando placa de desarrollo EasyPIC

Materiales de trabajo

HARDWARE	
1	Tarjeta de desarrollo EasyPic v8
2	Microcontrolador PIC 16F887
3	Conector Tipo C – USB
4	Fuente de alimentación 12V
5	Sensor sharp
6	LCD 2x16

SOFTWARE	
1	MATLAB
2	PROTEUS
3	MIKROC PRO FOR PIC
4	CODEGRIP SUITE



Descripción de la práctica

Se requiere realizar un control automatizado de una puerta eléctrica. Muchos abran visitado algún centro comercial y hayan observado que en su entrada existen puertas que permiten el acceso a personas o cualquier objeto siempre y cuando se encuentre a cierta distancia de la puerta.

En la presente práctica se utiliza un sensor infrarrojo de distancia conectado al pin RA1 del MCU PIC16F887, un LCD 2x16 montado en la placa EasyPIC que muestre los valores receptados desde el sensor. Establecer una comunicación UART entre la PC y el MCU a través de los pines RC6 y RC7 (Tx y Rx) respectivamente. La lógica de programación se realiza en el software MikroC Pro For PIC y desde MATLAB en el archivo .m que se genera en la GUIDE considerando sus respectivas condiciones.

Desde MATLAB, diseñar una interfaz gráfica que permita recolectar los datos leídos por el sensor infrarrojo. Los datos recolectados representarlos en una gráfica donde se pueda apreciar sus valores de forma continua. Para indicar que existe la presencia de un objeto utilizar una etiqueta que diga (puerta abierta o puerta cerrada) de acuerdo a los valores enviados por el sensor.

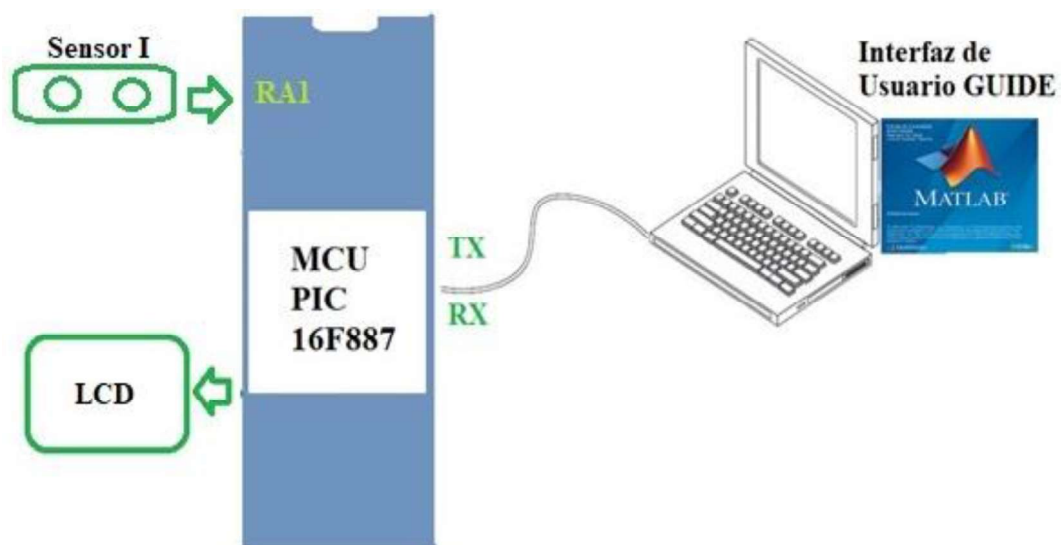


Ilustración 50 Práctica #5 esquema representativo de conexiones

3.2.6 PRÁCTICA 6:

Control de velocidad de un motor DC

Objetivo de la práctica

Desarrollar un código de microcontroladores para llevar el control de velocidad de un motor DC desde la plataforma MATLAB GUIDE, empleando interfaces graficas de usuario y software MikroC.

Objetivos específicos

- Desarrollar codificación en software MikroC que permita el control del prototipo.
- Simular el proyecto utilizando software ISIS Proteus para comprobar su funcionamiento.
- Desarrollar el diagrama de flujo para representar la lógica de programación.
- Diseñar interfaz gráfica de usuario (GUI) empleando comunicación UART para la E/S de datos.
- Grabar el microcontrolador empleando CODEGRIP Suite
- Analizar y comprobar el funcionamiento físico empleando placa de desarrollo EasyPIC

Materiales de trabajo

HARDWARE	
1	Tarjeta de desarrollo EasyPic v8
2	Microcontrolador PIC 16F887
3	Conector Tipo C – USB
4	L298N
5	Motor DC
6	LCD 2x16

SOFTWARE	
1	MATLAB
2	PROTEUS
3	MIKROC PRO FOR PIC
4	CODEGRIP SUITE



Descripción de la práctica

Se desea realizar el control de velocidad de un motor DC desde la plataforma de MATLAB GUIDE, para ello se cuenta con la estación de trabajo EasyPIC v8 y dispositivos como el MCU PIC16F887, un controlador de motores L298N y un motor DC.

Se emplea el software MikroC, para desarrollar la programación para el MCU PIC16F887. El controlador de motores también llamado puente H, conecta a los pines RC0, RC1 y RC3. Se utiliza un LCD 2x16 montado en la placa EasyPIC para mostrar los valores de PWM que se aplican a los motores, además que nos muestre el sentido de giro del motor (horario o antihorario). Establecer una comunicación UART entre la PC y el MCU a través de los pines RC6 y RC7 (Tx y Rx) respectivamente.

Desde MATLAB, diseñar una interfaz gráfica que permita enviar datos, para llevar un control constante del motor utilizado. El diseño debe constar con una barra desplazable donde su máximo valor sea 255 y el mínimo -255, esos valores deben ser enviados por el puerto serial hacia el MCU, el software desarrollado en MikroC debe ser capaz de leer los datos enviados y aplicarlos adecuadamente a la salida PWM del MCU.

Su funcionamiento:

- La barra desplazable debe iniciar en el centro enviando datos de “0”
- Si la barra se desplaza hacia la derecha, el motor gira en sentido horario y envía datos superiores a 0.
- Si la barra se desplaza hacia la izquierda, el motor gira en sentido antihorario y envía datos inferiores a 0.

3.2.7 PRÁCTICA 7:

Control ON/OFF de temperatura empleando un MCU PIC16F887

Objetivo de la práctica

Desarrollar un código de microcontroladores para realizar la medición temperatura y su respectivo envío de información hacia la plataforma de MATLAB y establecer un control digital ON/OFF.

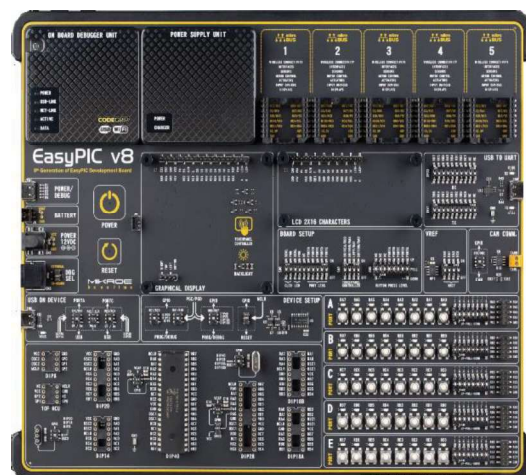
Objetivos específicos

- Desarrollar codificación en software MikroC que permita el control del prototipo.
- Simular el proyecto utilizando software ISIS Proteus para comprobar su funcionamiento.
- Diseñar el programa de solución mediante un diagrama de flujo
- Diseñar interfaz de usuario (GUIDE) empleando MATLAB.
- Grabar el microcontrolador empleando CODEGRIP Suite
- Analizar y comprobar el funcionamiento físico empleando placa de desarrollo EasyPIC

Materiales de trabajo

HARDWARE	
1	Tarjeta de desarrollo EasyPic v8
2	Microcontrolador PIC 16F887
3	Conector Tipo C – USB
4	LM35
5	Resistencia 4,7k Ω
6	Culer (mini ventilador)
7	L298N (Puente H)

SOFTWARE	
1	MATLAB
2	PROTEUS
3	MIKROC PRO FOR PIC
4	CODEGRIP SUITE



Descripción de la práctica

Se desea llevar a cabo un control ON/OFF de temperatura, para ello se cuenta con la estación de trabajo EasyPIC v8 y dispositivos como el MCU PIC16F887, un controlador de motores L298N, un mini ventilador y un sensor LM35.

Se emplea el software MikroC, para desarrollar la programación para el MCU PIC16F887, de manera que pueda medir constantemente la temperatura ambiente con el dispositivo LM35. De acuerdo a los valores de temperatura la programación debe ser capaz de activar un actuador con el objetivo de reducir la temperatura. Si la temperatura está por encima de 26°C el MCU debe accionar el mini ventilador, aumentando o reduciendo la velocidad de manera periódica. Se utiliza un LCD 2x16 montado en la placa EasyPIC para mostrar los valores de temperatura y las acciones que se han generado. Establecer una comunicación UART entre la PC y el MCU a través de los pines RC6 y RC7 (Tx y Rx) para graficar su valor de temperatura con respecto al tiempo.

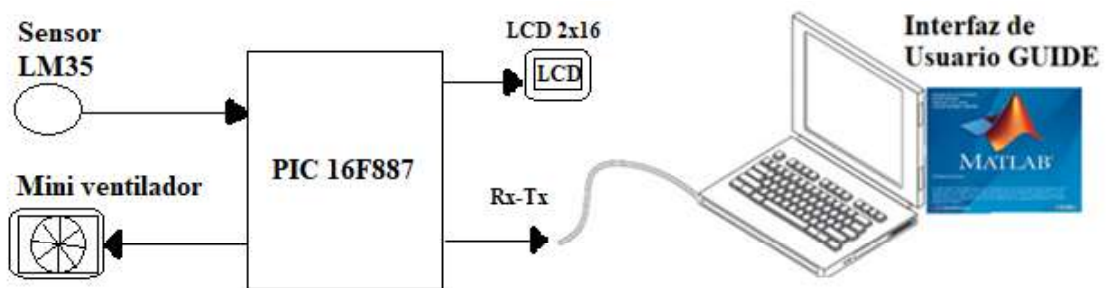


Ilustración 51 Práctica #7 esquema representativo de conexiones

3.3 PRESUPUESTO DE LA PROPUESTA

A continuación, se detalla el costo final de la propuesta planteada.

Herramientas de hardware adquiridas desde la empresa MikroE situada en el país de Serbia.

Costo del hardware de importación			
Cantidad	HERRAMIENTAS	COSTO	PRECIO FINAL
1	Placa de desarrollo EASYPIC v8	\$ 249	\$ 249
1	Fuente de alimentación 12V -2A	\$ 12.40	\$ 12.40
2	Conectores EasyPROTO	\$ 6	\$ 12
1	Pantalla TFT 320x240	\$ 33,50	\$ 33,50
Envió			\$ 25
Impuestos /Aduana			\$ 310
TOTAL			\$ 641,90

Tabla 19 Costo de equipos de Importación

Herramientas adquiridas dentro del país.

Costo de hardware			
Cantidad	HERRAMIENTAS	COSTO	PRECIO FINAL
1	Microcontrolador PIC 16F887	\$ 10	\$ 10
1	Controlador de motores L298N	\$ 6	\$ 6
1	Sensor Infrarrojo Sharp	\$ 12	\$ 12
1	Motor reductor DC	\$ 20	\$ 20
1	Mini ventilador DC 12v	\$ 2	\$ 2
1	Display siete Segmentos	\$ 0,50	\$ 0,50
10	Resistencias	\$ 0.10	\$ 1
1	Baquelita virgen 10x10 cm	\$ 1,50	\$ 1,50
1	Sensor LM35	\$ 1	\$ 1
TOTAL			\$ 54

Tabla 20 Costos de dispositivos adquiridos dentro del país

En el desarrollo de la propuesta se necesitó realizar diseños en 3D los cuales fueron realizados en el software FreCad siendo este un software libre. Todos los softwares que se han utilizado son totalmente gratuitos.

Costo de desarrollo		
HERRAMIENTAS	COSTO	PRECIO FINAL
Desarrollo Código	-	-
Diseño Prototipo	-	-
Impresión 3D	\$ 20	\$ 20
Ensamblaje	-	-
TOTAL		\$ 20

Tabla 21 Costos de Desarrollo

COSTO FINAL

En la Tabla 22 muestra el valor final de la propuesta tecnológica.

DESCRIPCIÓN	PRECIO FINAL
Costos de Hardware de importación	\$ 641,90
Costos de Hardware en Ecuador	\$ 54
Costos de Software	-
Costos de Desarrollo	\$ 20
Gastos adicionales	\$ 100
TOTAL	\$ 815

Tabla 22 Costo final de la propuesta

RESULTADOS

Como se ha mencionado la tarjeta EasyPIC v8 es utilizada para comprobar o verificar el funcionamiento de cada una de las prácticas propuestas en este documento. Se seleccionó esta tarjeta debido a que cumple con los requerimientos necesarios para facilitar en los estudiantes el proceso de aprendizaje de los MCU ya que contiene un hardware diseñado acorde a las necesidades del usuario, para que quienes la utilicen solo se centren en el desarrollo del código de programación sin tener que preocuparse en las conexiones de hardware y los periféricos que se desee utilizar.

A continuación, se detalla un resumen de los resultados obtenidos en cada una de las aplicaciones prácticas:

Práctica #1

La presente práctica fue propuesta para que los usuarios aprendan de forma correcta el funcionamiento de la placa EasyPIC v8 y los demás softwares que lo acompañan. Se llevó a cabo la siguiente práctica dando resultados favorables al momento de probar su funcionamiento con la tarjeta EasyPIC.

Esta actividad da inicio con una introducción sobre el uso y funcionamiento del Software MPLAB, utilizado para el desarrollo de la programación en lenguaje ensamblador. En el transcurso de la práctica se verifica el funcionamiento del código de prueba escrito para el PIC 16F887 el cual tiene como finalidad manejar los puertos de entrada y salidas Digitales del MCU. Para prevenir errores en la implementación real se utilizó ISIS Proteus el cual comprueba el funcionamiento de la programación escrita. De igual forma para el software Proteus se brinda los pasos necesarios para llevar el a cabo la simulación de la práctica.

La placa EasyPIC incorpora un programador para PIC con el nombre de CODEGRIP. En esta sección el usuario aprende a grabar un MCU PIC mediante las dos posibilidades que la placa posee, estas son: mediante USB o WIFI.

Tras concluir los procesos antes indicados, nos queda verificar el funcionamiento del código cargado en el microcontrolador haciendo uso de la placa de desarrollo EasyPIC v8.

Para ello habilitamos los LEDs del puerto D y adicionalmente los pulsadores del puerto A (Ilustración 54). Se trata de interactuar tanto desde los switches o los pulsadores ya que son las entradas hacia el microcontrolador y los resultados serán los mismo, siendo que en sus entradas se escribe un “1” o un “0”.

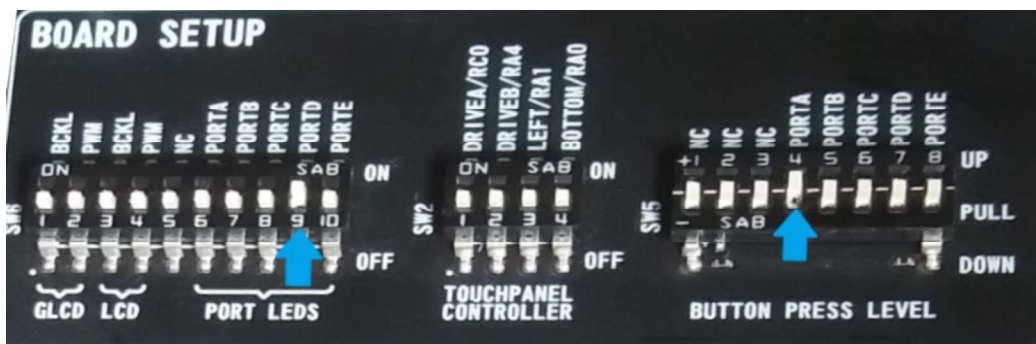


Ilustración 52 Habilitando los LEDs del puerto D y los botones del puerto A



Ilustración 53 Introduciendo un 1 o un 0 lógico por el switch del puerto A



Ilustración 54 Manipulando los botones del puerto A

Recomendaciones

- Para grabar el MCU a través de WIFI, se recomienda que todos los dispositivos externos (PCs) estén conectados a la misma red que está conectada la placa.
- No olvidar de configurar el BOARD SETUP para habilitar tanto los LEDs o los switch de acuerdo sea conveniente.

Práctica #2

Esta práctica tiene como objetivo desarrollar un programa que permita simular el control de nivel de un depósito de líquidos, empleando el software MPLAB y la placa de desarrollo EASYPIC v8. Para ello se realiza las diferentes pruebas de funcionamiento con la placa.

Cuando ninguno de los sensores se encuentra bajo agua, se entiende que el depósito está vacío y se accionarán las dos bombas.



Ilustración 55 Cuando ninguno sensor detecta la presencia de líquido

Cuando el nivel del líquido toca el Sensor de vacío “SV” seguirá llenándose el depósito con las dos Bombas activas. El indicador “Llenándose” se iluminará.



Ilustración 56 Cuando el sensor "SV" detecta el liquido

Cuando el nivel del líquido toque el sensor de llenado “SLL” se apaga la bomba B2, quedando la bomba B1 activa en modo mantenimiento. El indicador “Lleno” se ilumina.



Ilustración 57 Cuando el líquido a llegado al sensor "SLL"

Si el nivel del líquido moja el sensor de rebose “SR” se apaga también la bomba B1, quedando las dos bombas fuera de servicio. El indicador “Rebose” se enciende.



Ilustración 58 los tres sensores detectan el agua

Recomendaciones

- Antes de realizar un código de programación se recomienda identificar cuáles son las posibles entradas y salidas del sistema.
- Establecer una tabla de verdad para dar idea el funcionamiento del sistema.
- La experimentación en la tarjeta de desarrollo debe responder a la tabla de verdad planteada, caso contrario revisar codificación. Se recomienda utilizar tablas de datos en la programación.

Práctica #3

El objetivo de la siguiente aplicación práctica es desarrollar un programa que permita simular el conteo de vehículos en la entrada de un estacionamiento, empleando el software MPLAB y la placa de desarrollo EASYPIC v8.

Para evitar utilizar cables y reducir las conexiones, se ha realizado un diseño en baquelita donde compone de los diversos elementos que se utilizan en las prácticas. Consta de tres entradas el puerto A, C y D la misma que permite conectar hacia la placa EasyPIC. El puerto A es utilizado para los sensores de temperatura y distancia más adelante se detalla sus conexiones. El puerto C es empleado para hacer uso del motor DC y el miniventilador. El puerto D conecta a un display de siete segmentos con sus debidas resistencias para su funcionamiento adecuado.

En esta práctica se emplea el display de siete segmentos, para ello se realiza la respectiva conexión del puerto D de la placa EasyPIC hacia el prototipo realizado en baquelita (ver figura 61).

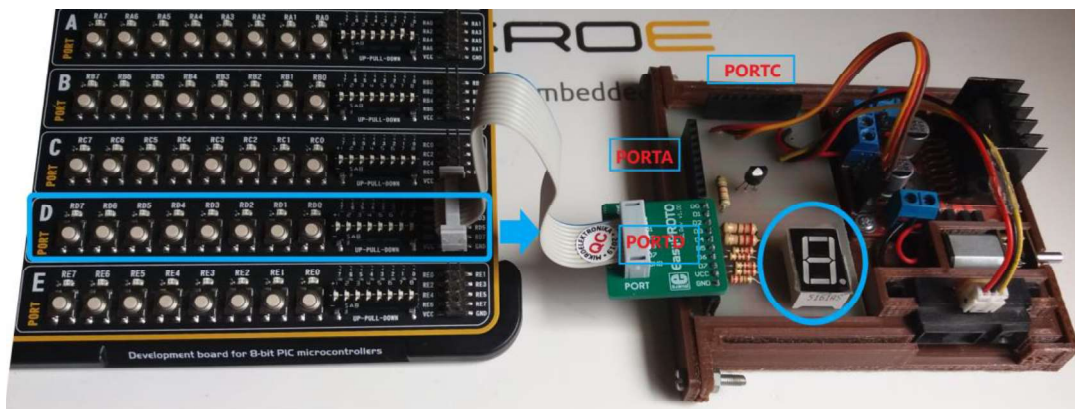


Ilustración 59 Conexión entre EasyPIC y prototipo realizado en baquelita

En la placa EasyPIC configuramos el BOARD SETUP, para habilitar los botones del puerto A y los LEDs del puerto B, observar la ilustración 62 y configurar de acuerdo a ello. Solo resta grabar la información en el MCU a través de WIFI, y se proceder con

la verificación del funcionamiento del prototipo y obtener los resultados deseados de acuerdo a la programación realizada.

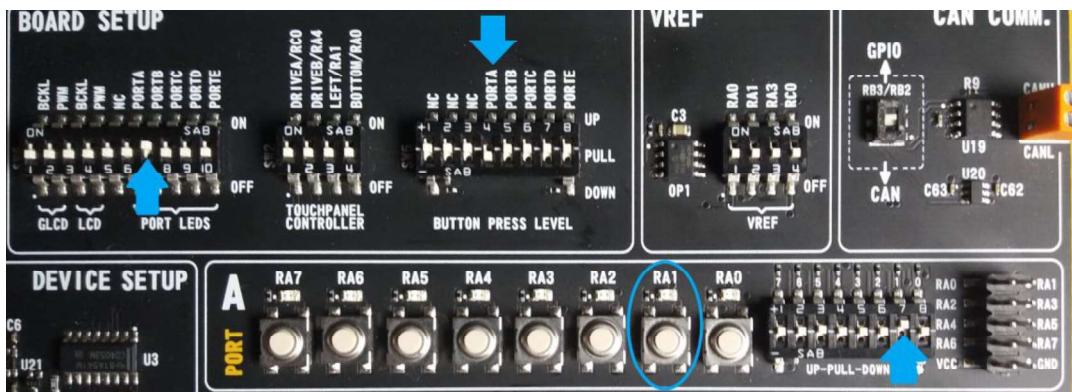


Ilustración 60 Configuración del Board Setup para la práctica #3

Cada vez que se presiona el pulsador conectado al pin RA1 quiere decir que un vehículo a ingresado y el display de siete segmentos los registra. En la siguiente ilustración se puede observar que un vehículo a ingresado.

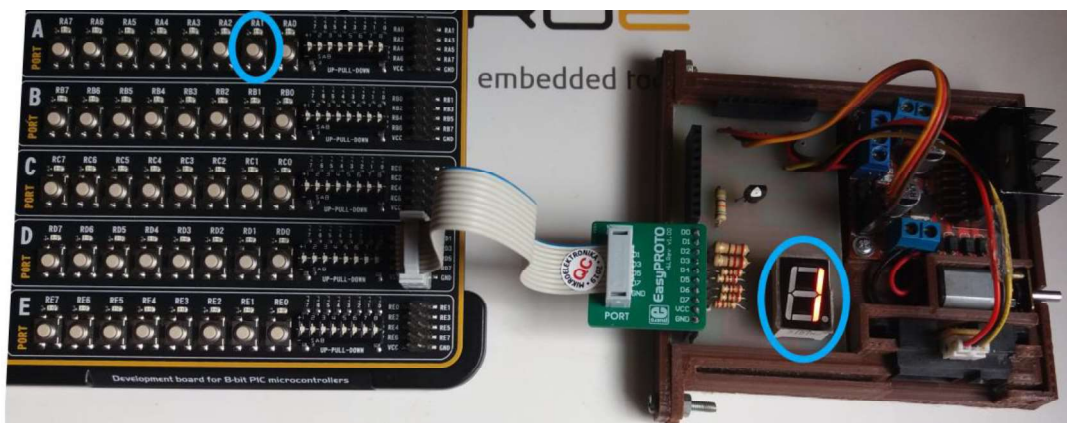


Ilustración 61 Primer vehículo registrado

De manera consecutiva se ha presionado el pulsador nueve veces para comprobar si los valores registrados son los correctos lo podemos verificar en la siguiente ilustración.



Ilustración 62 Registrando el vehículo #9

Cuando han ingresado nueve vehículos y alguien más quiere ingresar, este envía una alarma es decir se activa el LED conectado al pin RB1 de la placa EasyPIC indicando que el estacionamiento está lleno. De esta manera se culmina la práctica 3 comprobando todos los resultados y explicando su funcionamiento.

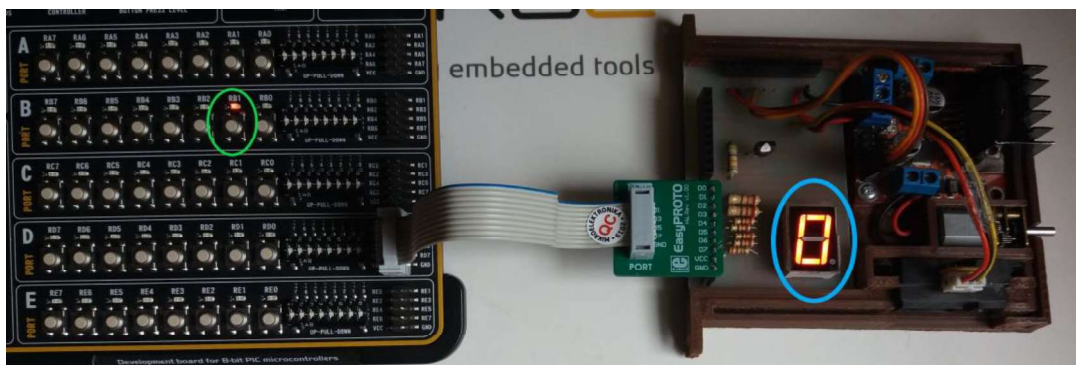


Ilustración 63 Led RB1 activo, indicando que el estacionamiento está lleno

Recomendaciones

- Se recomienda utilizar resistencias de 220 o 330Ω para realizar la conexión hacia el display de siete segmentos, ya que ayudaran a proteger los diodos LEDs que se encuentran encapsulados en el display.
- En la codificación se recomienda utilizar tablas de datos para representar el valor hexadecimal a 7 segmentos.
- Se recomienda agregar la librería RETARDOS.INC a la carpeta de trabajo, la cual es utilizada para eliminar el efecto rebote que ocasione un pulsador.

Práctica #4

Con el objetivo de desarrollar un programa que permita la manipulación de los puertos del microcontrolador PIC 16F887, empleando el software MikroC, MATLAB y la placa de desarrollo EasyPIC v8.

A partir de esta práctica se emplean los softwares: MikroC, MATLAB, Proteus, CODEGRIP. Se inicia detallando los pasos necesarios para la creación de un nuevo proyecto en cada uno de los softwares mencionados.

Tras realizar todos los procesos se procede con la verificación de la interacción entre el módulo EasyPIC v8 y la interfaz de usuario realizada en MATLAB.

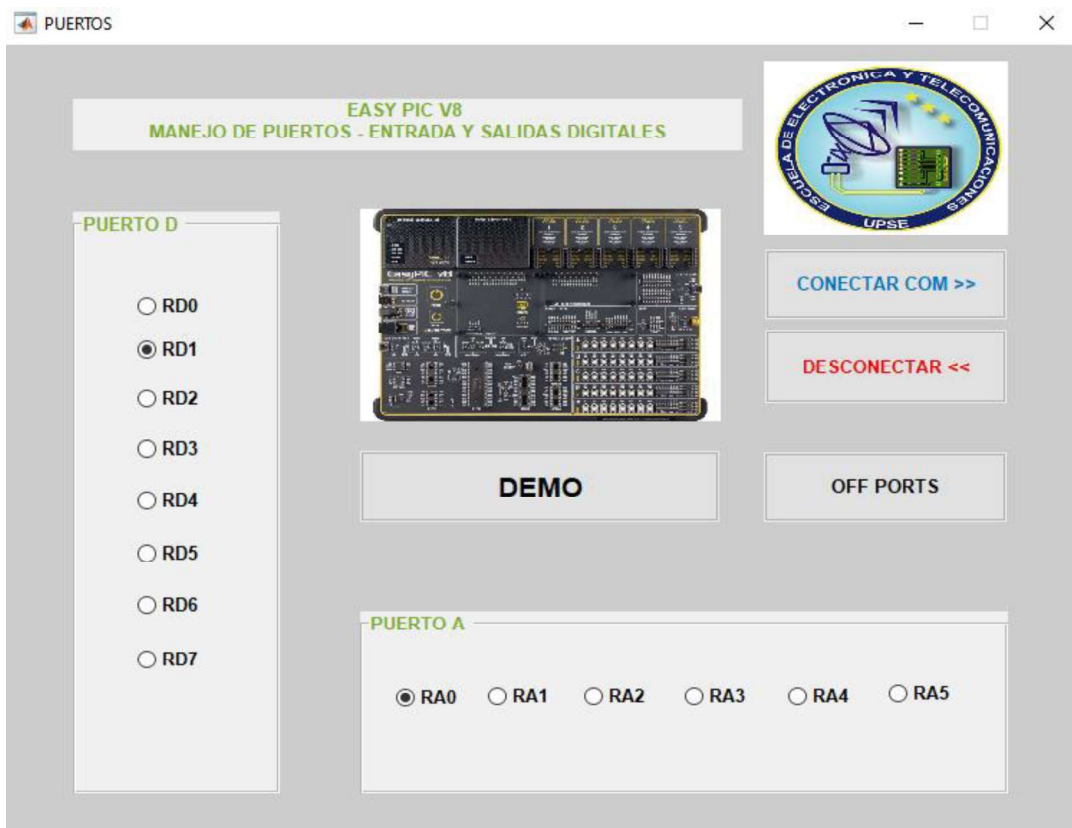


Ilustración 64 Interfaz de usuario práctica #4, manipulación de puertos

Se conecta el cable USB-C a la salida del bloque USB TO UART para proceder a manipular de forma independiente los puertos A y D del MCU. En la siguiente figura apreciamos que RA0 ha sido presionado ya que el LED conectado a ese pin esta activo. Se realiza varias pruebas presionando los diferentes pines desde la interfaz de MATLAB.

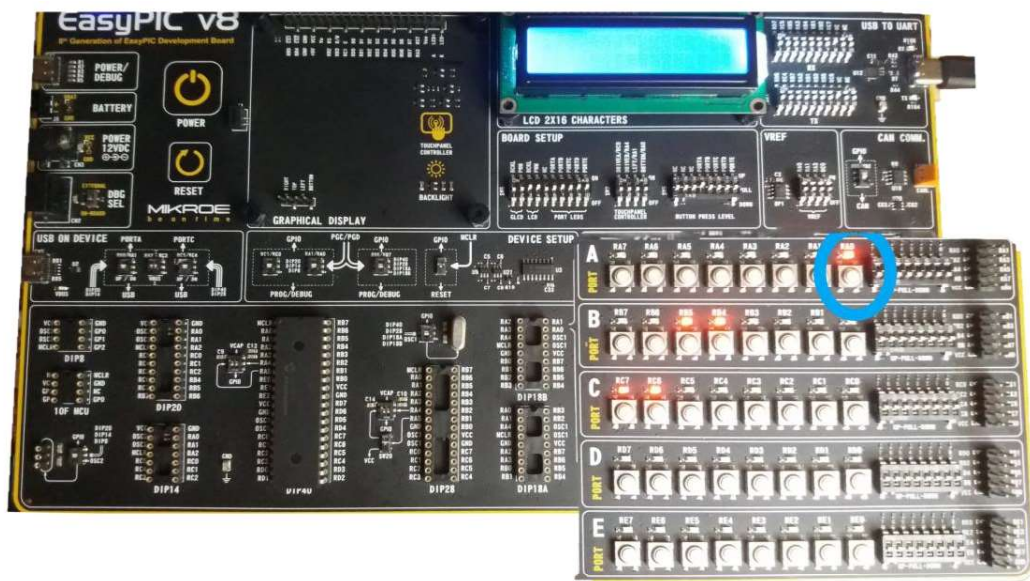


Ilustración 65 Manipulando el puerto A desde la interfaz de MATLAB

Se comprueba que la comunicación serial entre ambos dispositivos es favorable debido a que los datos que se reciben son los correctos. Desde MATLAB solo se envían caracteres (letras), para facilitar al MCU en la lectura de los datos en el puerto serie. Se ha definido que para cada pin del puerto A y D se envíen caracteres de manera independiente a cada uno. En la ilustración 63 se ha presionado el botón *DEMO* el cual hace una demostración del manejo de un LCD, los retardos y el control de todos los puertos, es decir cumple una secuencia.

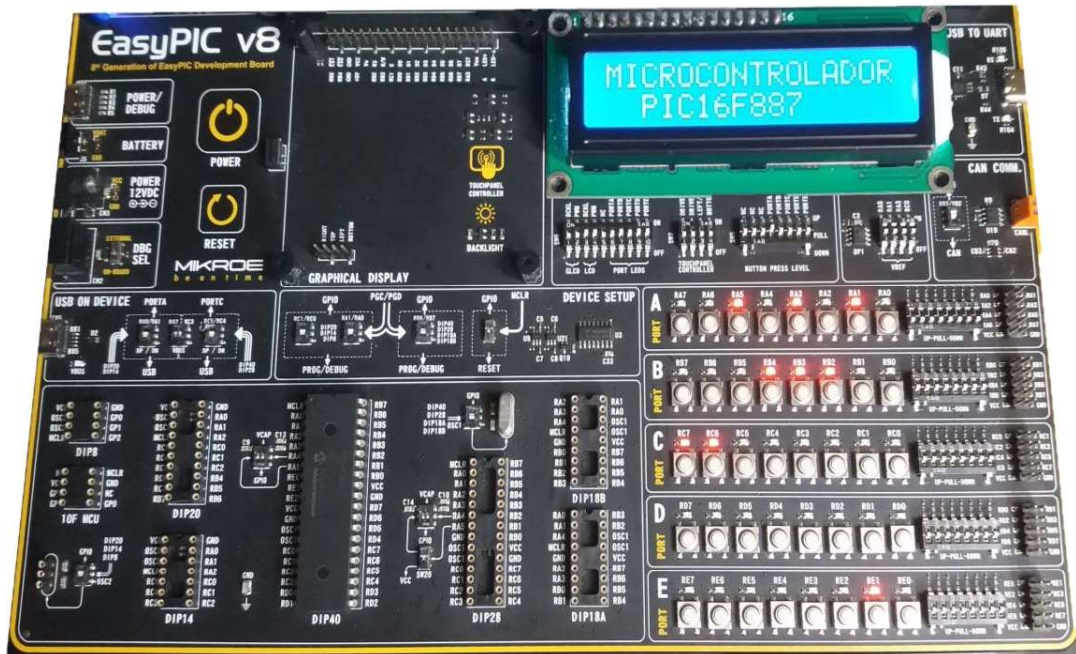


Ilustración 66 Cuando se presiona el botón DEMO desde la interfaz de Usuario

Recomendaciones

- Al utilizar el software MikroC se recomienda marcar las librerías que se hagan uso en la práctica, en este caso se utiliza un LCD, y por ende hay que habilitar la librería correspondiente al LCD, esto evitara los errores en la compilación.
- En bloque USB TO UART del módulo EasyPIC deben estar habilitados los pines RC7 y RC6, caso contrario no establecerá una comunicación entre la plataforma de MATLAB y el MCU.
- Se recomienda verificar a que puerto COM está conectada el módulo EasyPIC, mi computadora reconoce que el modulo está conectado al puerto COM10. Esto servirá al momento de conectar ambos dispositivos desde MATLAB.
- No olvidar conectar el cable USB-C desde la salida del bloque USB TO UART a la computadora personal.
- Se recomienda emitir desde MATLAB los comandos necesarios para conectar el puerto COM.

Práctica #5

El objetivo de esta práctica es desarrollar un programa que permita simular una puerta eléctrica automática, empleando un sensor de distancia, software MikroC e interfaces graficas de usuario en MATLAB.

En teoría se realiza el estudio de la señal analógica a su vez el tratamiento de ella convirtiéndola a una señal digital. Desde Matlab se ha diseñado una interfaz de usuario con el objetivo de visualizar los datos del sensor de manera externa.

Se realiza la conexión respectiva entre el prototipo construido en baquelita y la placa EasyPIC, debe conectarse al puerto A de la placa ya que en la codificación está configurado el pin RA1 como analógico y a través de él recibiremos los datos que nos proporciona el sensor. (ver ilustración 69)

Desde la interfaz de usuario en la plataforma MATLAB, se adquieren con éxito los datos enviados por el MCU, haciendo uso de la interfaz serial que está contiene. También se hace uso de una pantalla LCD ubicada en el módulo, la cual muestra el valor ADC de acuerdo a la medida del sensor.

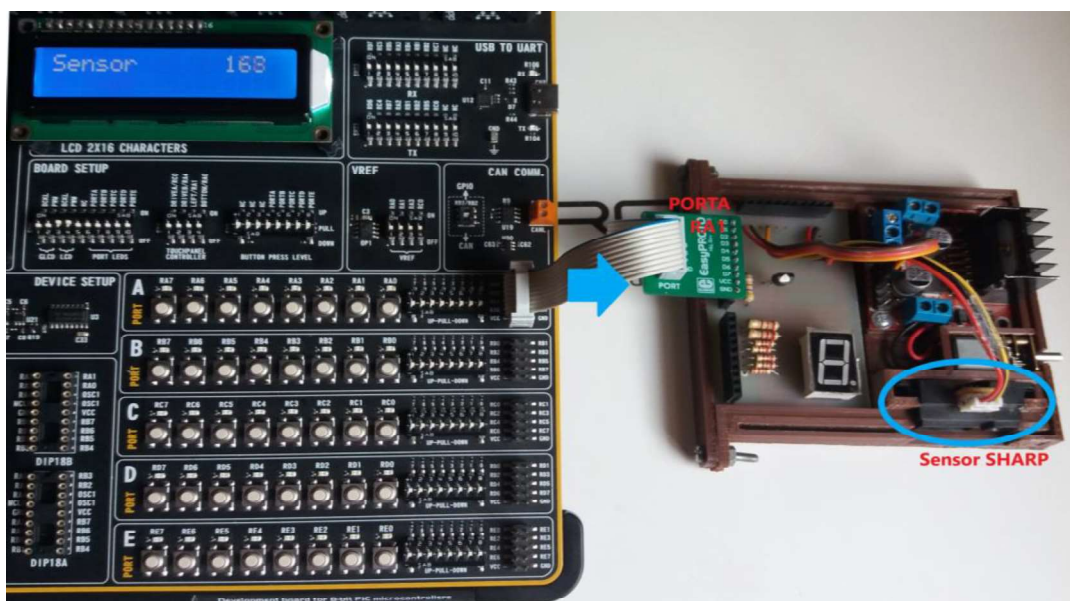


Ilustración 67 Conexión entre el módulo EasyPIC y el sensor

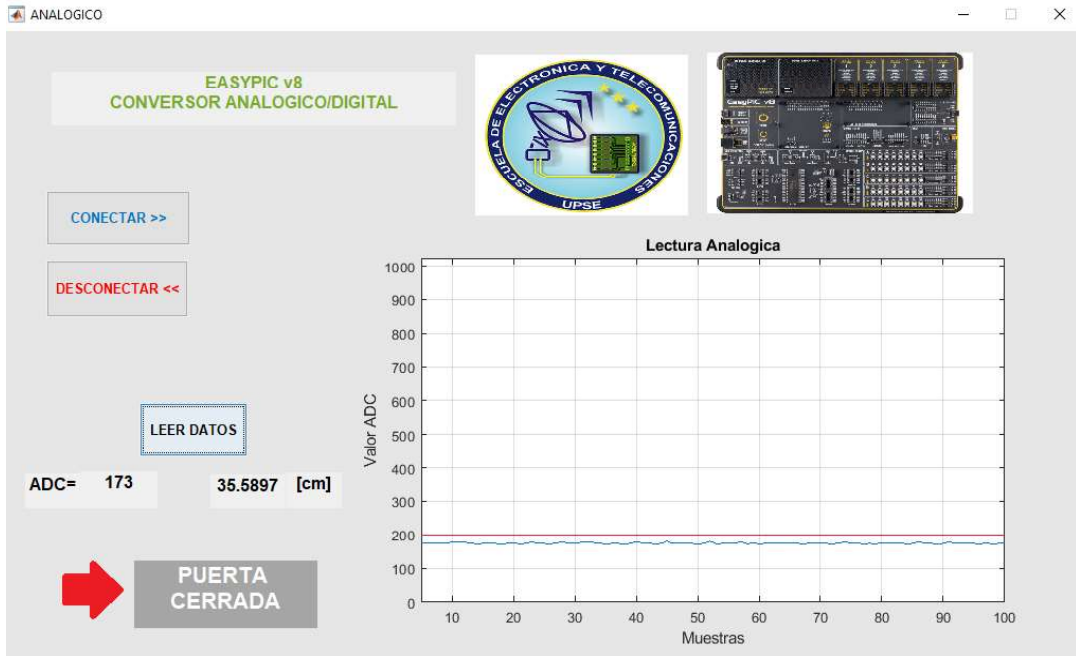


Ilustración 68 Interfaz de usuario practica #5, puerta cerrada

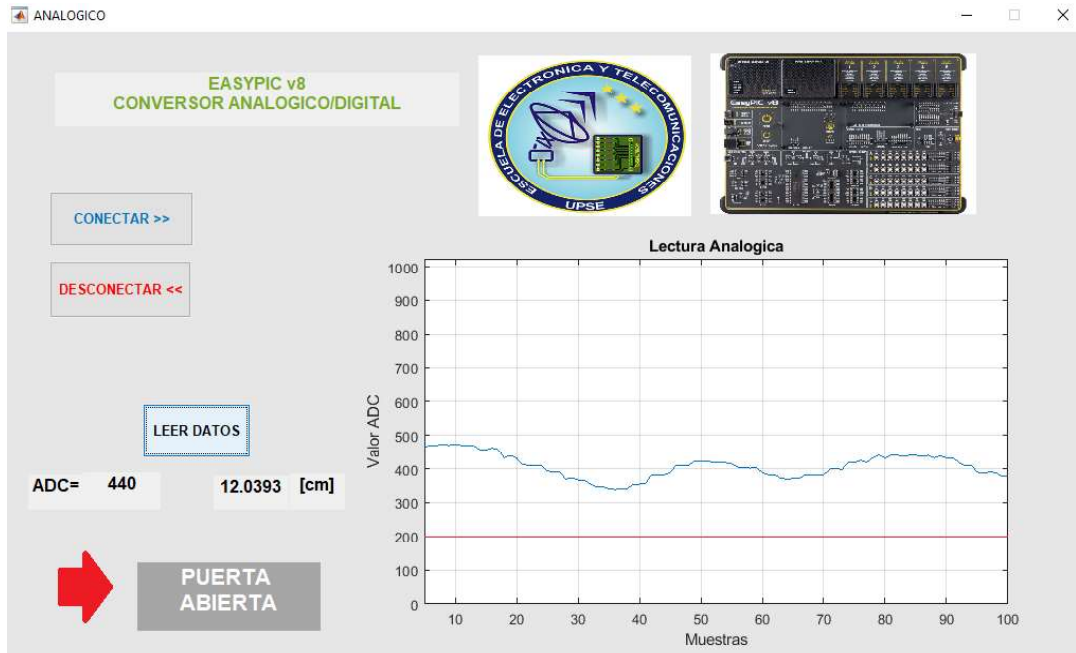


Ilustración 69 Interfaz de usuario práctica #5, puerta abierta

En las interfaces graficas expuestas se puede observar los resultados de la adquisición de datos, donde nos muestra el valor ADC que envía el MCU y su grafica correspondiente a esos valores. Al obtener el valor ADC, en MATLAB se realiza una conversión para obtener su resultado en centímetros para luego presentarlos en las etiquetas correspondientes.

Se comprueba que si el valor ADC es superior a 200 entonces nos indica que la puerta está abierta, caso contrario la puerta está cerrada. Las imágenes muestran una línea roja, la cual representa el límite para mencionar los estados de la puerta, si bien está abierta o cerrada.

Recomendaciones

- Se recomienda revisar las especificaciones técnicas del sensor infrarrojo (SHARP), revisar su conexión y su rango de medición.
- Se recomienda simular desde ISIS Proteus para corroborar que los datos se estén transmitiendo por el puerto serial
- Tener en cuenta el comando para leer los datos que han sido transmitidos por el puerto serie en MATLAB `fscanf(se, '%d');`
- Cuando se presiona el botón LEER DATOS el programa entra a un bucle infinito, leyendo los datos del puerto serie. Cada repetición que hace la condición while, esta toma 100 muestras del puerto serie, tener mucho cuidado al momento de manipular los datos.
- Se recomienda habilitar los DIPS 3 y 4 del switch (SW6) para dar contraste a la pantalla LCD y se pueda visualizar de manera adecuada.

Práctica #6

El objetivo de esta práctica es controlar un motor DC desde una interfaz de usuario diseñada en MATLAB GUIDE. Se procede a realizar las pruebas de funcionamiento donde se obtienen resultados esperados y favorables de acuerdo a la programación realizada. A continuación, se puede apreciar la conexión entre el modulo y los demás elementos electrónicos para el cumplimiento de la práctica. Para esto construye un prototipo en una placa PCB, extendiendo los pines del puerto C del MCU. La placa PCB está diseñada para recibir los pines del puerto C del MCU observar ilustración 72 para verificar sus conexiones.

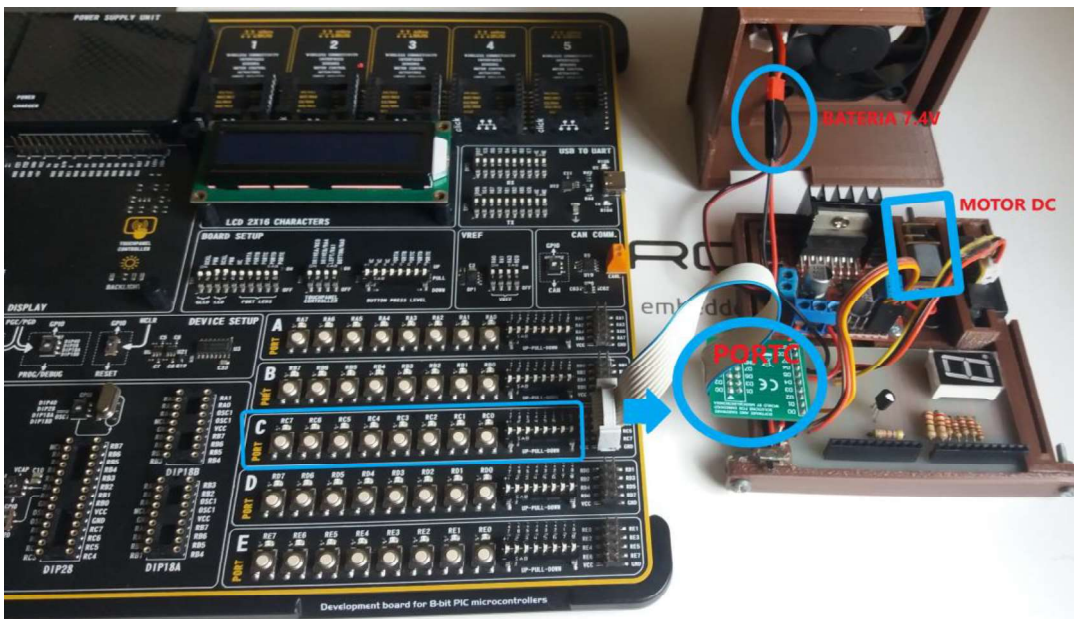


Ilustración 70 Conexión entre EasyPIC, L298N y motor DC

El dispositivo L298N permite controlar dos motores: donde el motor DC conecta a las salidas OUT3 y OUT4 del L298N y por ende hace uso de los pines *in3*, *in4* y *ENB*, a continuación, se muestra una tabla de conexiones.

Cable color	Café (marrón)	Naranja	Rojo
Conexión - Puente H	PWM - ENB	In3	In4
Conexión pines MCU	RC1	RC3	RC0

Tabla 23 Conexión para salida del motor B

Para dar potencia a los motores se utilizó una batería de 7.4V - 300mA, el cual está conectado un switch para permitir la alimentación al puente H (L298N).

El diseño completo de la impresión 3D queda montada de la siguiente forma (ilustración 73). Para trabajar sin inconvenientes es favorable cubrir con todas las piezas 3D.

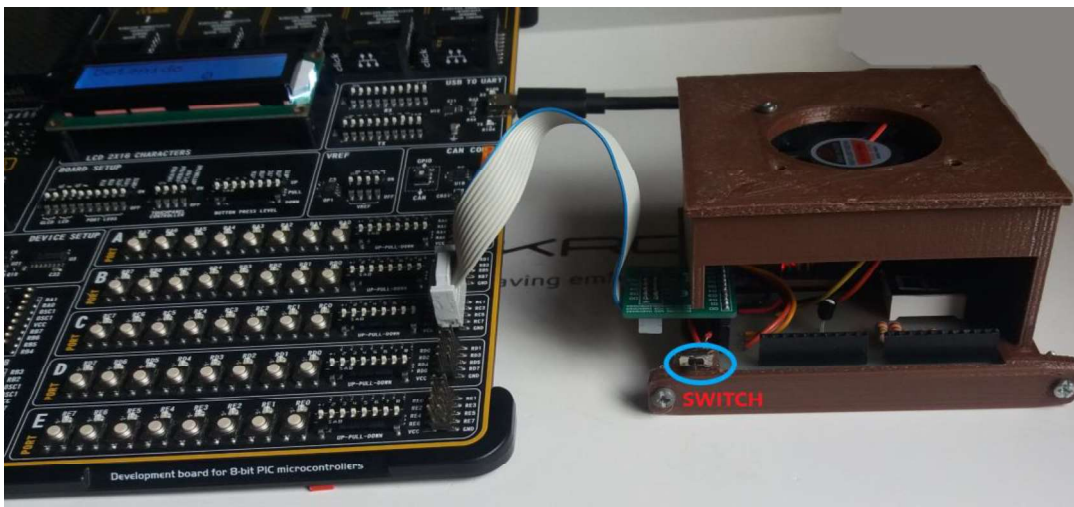


Ilustración 71 Conexiones práctica #6, EasyPIC y prototipo 3D

Se han realizado varias pruebas manipulando la velocidad del motor desde la interfaz gráfica de MATLAB. En el display LCD nos indica el sentido de giro del motor y el valor PWM aplicado.



Ilustración 72 Pruebas de velocidad del motor DC

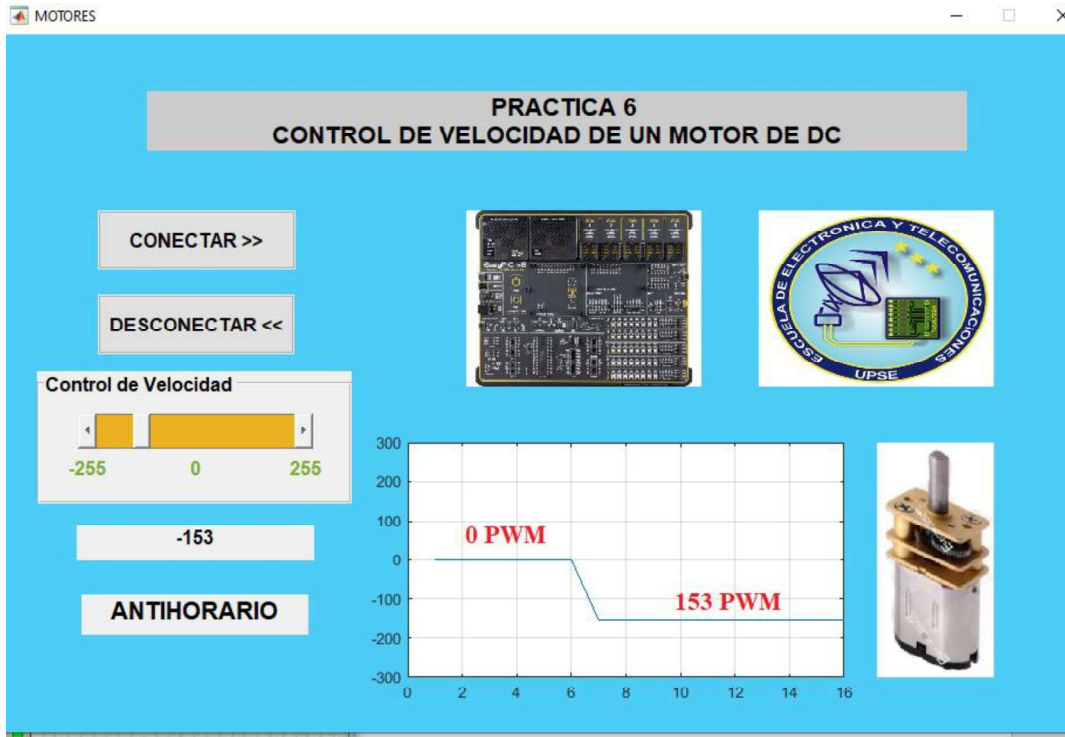


Ilustración 73 interfaz de usuario práctica #6, motor en sentido Antihorario

Desde MATLAB mediante la interfaz gráfica diseñada, se establece una comunicación Serial para transmitir datos o valores enteros. Para ello se ha utilizado el componente *Slider* el cual está compuesto de una barra desplazable, donde sus valores cambian a medida que el usuario mueve la barra. Este componente se configuró para entregar valores que van desde 0, 0.1, 0.2, hasta llegar a 1, en la programación se plantean las ecuaciones para convertir esos datos a valores que vayan de 0 a 255 siendo estos los valores enteros que pueden ser transmitidos por el puerto serie sin problemas.

En MikroC se reciben los datos que han sido transmitidos por puerto serial. Estos datos fueron modificados nuevamente para establecer valores que permitan hacer el giro Horario y Antihorario de los motores. En el LCD se muestra el valor PWM que se ha aplicado al motor también el sentido de giro de aquéllo.

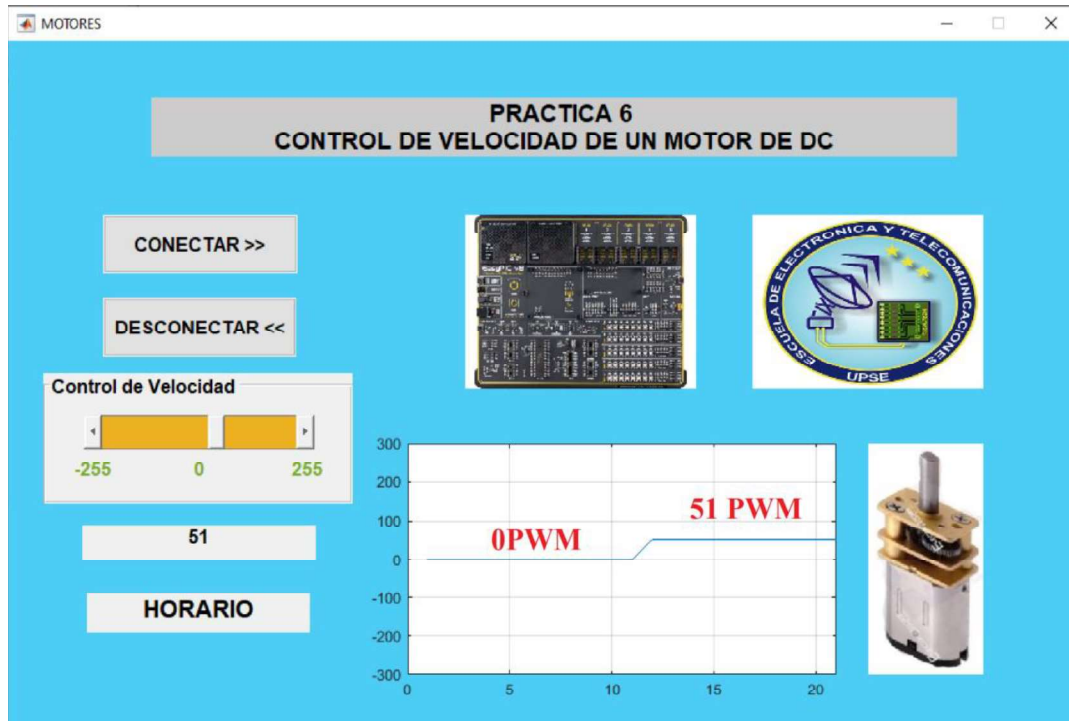


Ilustración 74 Interfaz de usuario práctica #6, motor en sentido horario

Recomendaciones

- Se recomienda revisar las características técnicas del controlador L298N y su esquema de conexiones.
- Al desarrollar la programación en MikroC se debe indicar que módulo PWM se va utilizar, ya que el MCU tiene dos módulos internos. Al utilizar PWM2 está indicando que va hacer uso del pin RC1 como salida PWM.
- Se recomienda utilizar el Software Serial Port Driver para simular un puerto COM virtual en la PC, por este medio se puede interactuar entre la plataforma de Proteus y MATLAB.
- Se recomienda que el componente *Slider* tome valores enteros positivos, ya que al momento de transmitirlo por el puerto serie los datos no se interpreta de misma forma.

Práctica #7

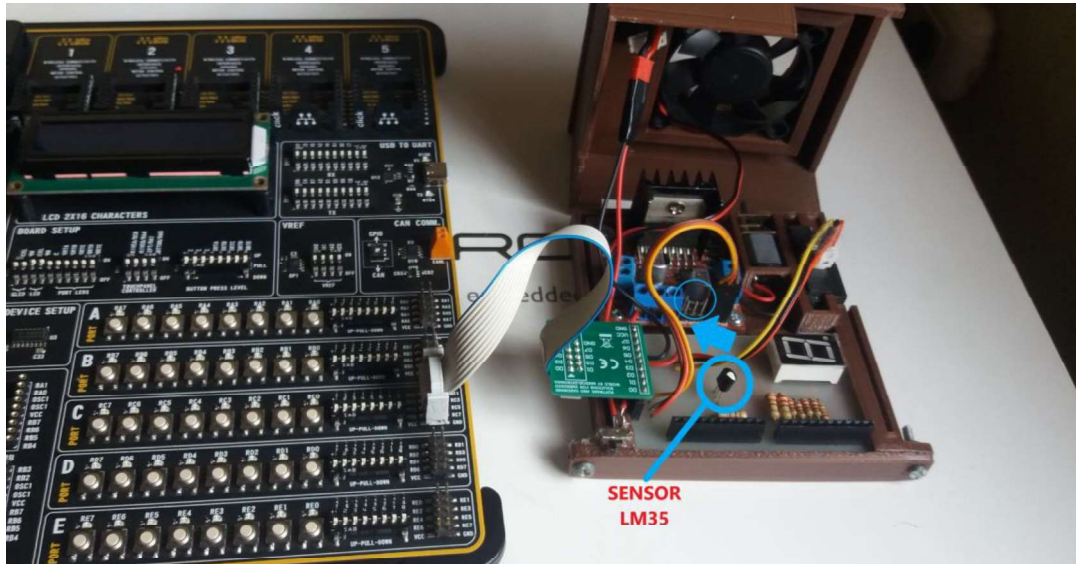


Ilustración 75 Conexión para habilitar la salida del motor A

En la práctica 6 se hacía referencia al control de un motor DC haciendo uso de la salida para el motor B es decir *OUT3* y *OUT4*, en esta práctica se utiliza las salidas para el motor A (*OUT1* y *OUT2*) para ello, debemos hacer un cambio en la conexión de la práctica 6. Desconectaremos el cable que conecta al controlador L298N etiquetados (*in3*, *in4*, *ENB*) y ubicaremos en las etiquetas *ENA*, *in1*, *in2* tomando en cuenta que el cable de color café es la señal PWM y debe ser conectado a la etiqueta ENA.

Para mayor seguridad en la conexión del elemento guiarse en la siguiente tabla:

Cable color	Café (marrón)	Naranja	Rojo
Conexión - Puente H	PWM-ENA	In2	In1
Conexión pines MCU	RC1	RC3	RC0

Tabla 24 Conexión para salida del motor A

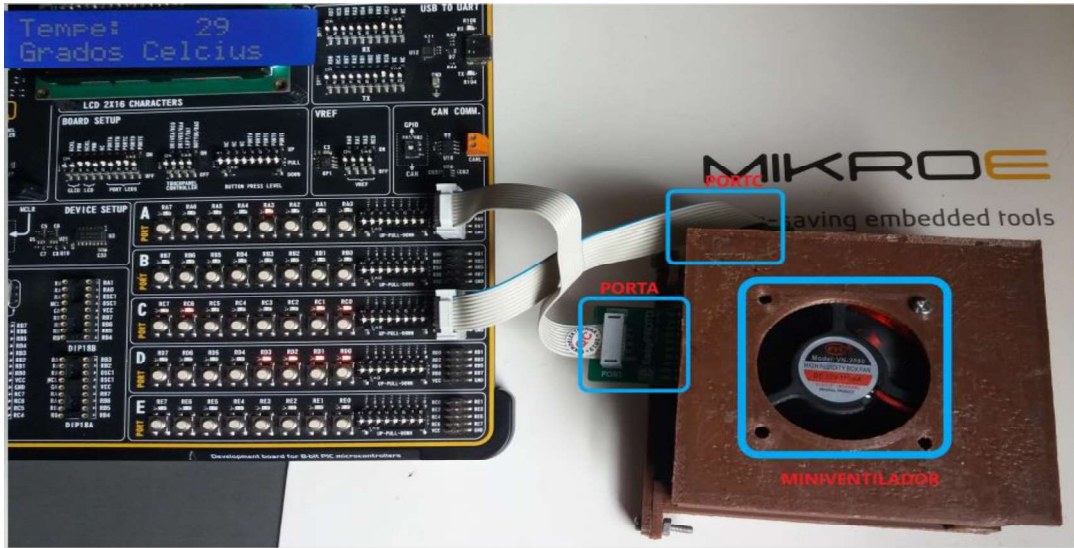


Ilustración 76 Conexión de componentes: Práctica #7

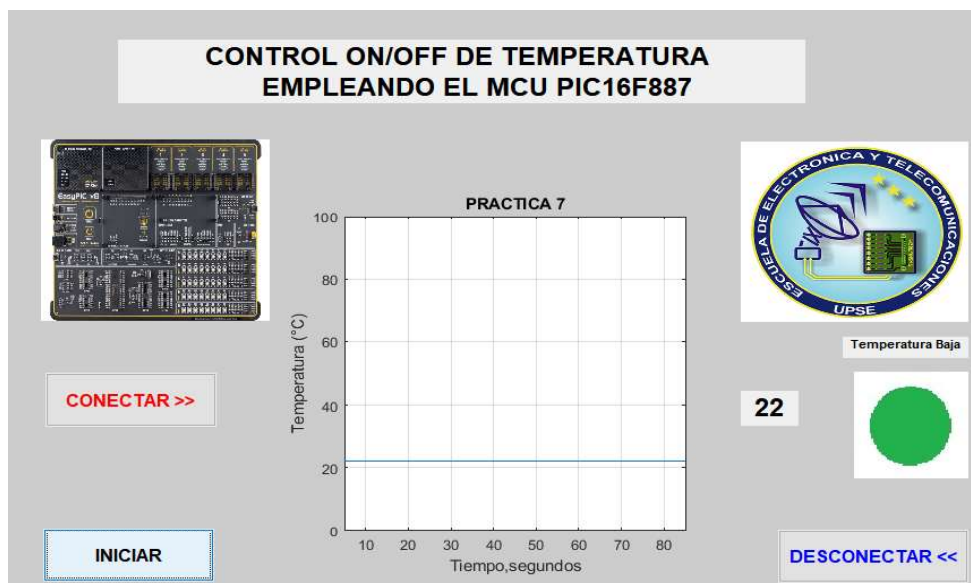


Ilustración 77 Interfaz de usuario práctica #7, temperatura baja

En la interfaz gráfica observamos los valores de temperatura que nos entrega el MCU, además su respectiva grafica indicando los cambios de temperatura que pueden darse. compuesta de varias etiquetas que ayudan a visualizar el estado de temperatura, si se encuentra a una temperatura alta o baja.

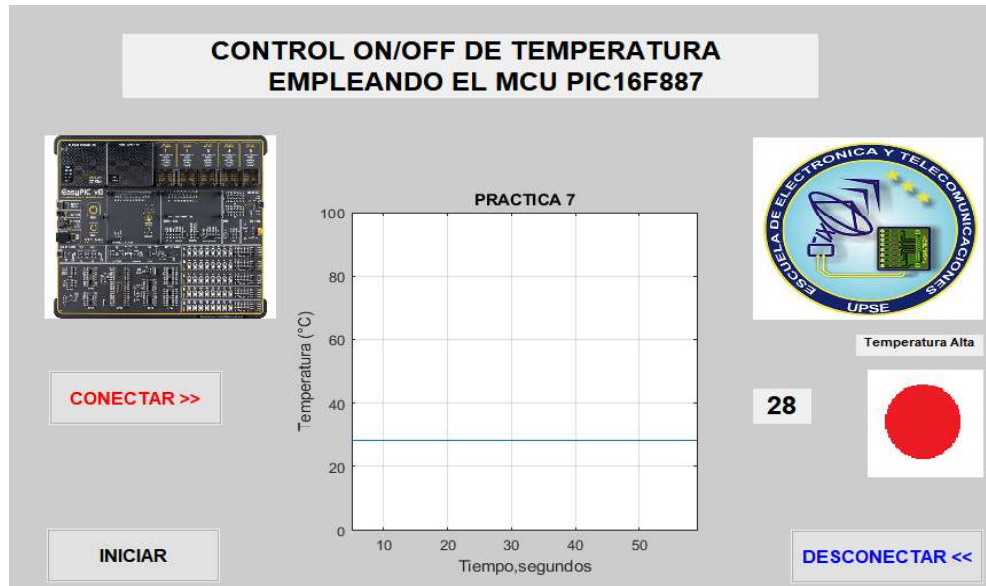


Ilustración 78 Interfaz de usuario práctica #7, temperatura alta

En el desarrollo de esta práctica se prueba la interacción entre varios componentes como son: el convertor analógico/digital (ADC) para adquirir los valores de temperatura entregados por el sensor LM35, la modulación por ancho de pulsos (PWM) en cual es entregado al controlador L298N y aplicado al motor, visualizador LCD mostrando el valor de temperatura en grados Celsius y los indicadores LEDs del puerto D para indicar que la temperatura está muy elevada. Se consiguió presenciar el cambio de velocidad en el motor, a medida que la temperatura sobre pasaba los 26°C se inicia a aumentar la velocidad con el objetivo de no sobrepasar los límites de temperatura establecida.

Recomendaciones

- Comprobar que la placa de desarrollo esté conectada en la salida de la etiqueta USB TO UART hacia la PC, luego verificar el puerto COM a la cual se encuentra conectada, en mi PC se encuentra en el COM10, establecer la comunicación a 9600 Baudios.
- Para obtener el valor de la temperatura en grados Celsius, se realiza una conversión al valor ADC adquirido.

CONCLUSIONES

- Se logró analizar satisfactoriamente todos los componentes electrónicos que incorpora el módulo EASYPIC V8 y los demás componentes externos que se han utilizado para el desarrollo de esta propuesta entre ellos los sensores, actuadores e indicadores, se han descrito detalladamente cada uno de ellos.
- La placa EasyPIC v8 acepta ocho tipos de encapsulados de microcontroladores desde 8 hasta 40 pines según su capacidad, también integra diversos periféricos de entrada y salida como: display de 2x16, LEDs, pulsadores, switches, puerto serial y el extensor de pines, entre los que se han utilizado. Este permitió comprobar el funcionamiento de cada una de las aplicaciones prácticas, demostrando una versatilidad y veracidad al instante de evidenciar los resultados.
- Para la construcción de las aplicaciones prácticas se tomó en cuenta los temas más frecuentes y trascendentales que se ven en la asignatura de Microcontroladores, se realizó tres prácticas en lenguaje ensamblador, planteando temas de control y haciendo uso de sus entradas y salidas digitales, con la finalidad de dar inicio al estudio de los microcontroladores, siendo este el lenguaje base de programación.
- Se realizó cuatro prácticas utilizando el software MikroC y MATLAB, donde la unidad de control es el PIC16F887 y su lenguaje de programación el C, se aplicó los conocimientos adquiridos en la Universidad, realizando interfaces de usuario diseñadas con la herramienta de *MATLAB GUIDE*, utilizadas para el manejo de datos.
- En las interfaces de usuario diseñadas en MATLAB se obtuvo las medidas de los sensores las cuales fueron representadas en una gráfica, además se logró controlar los periféricos tanto de entrada o salida del MCU, como el control de motores, diodos LEDS, pantalla LCD, la medición de temperatura, la medición de distancia, logrando tener una exitosa interacción entre el MCU y MATLAB.

RECOMENDACIONES

- Se recomienda que el modulo esté conectado a su fuente de alimentación la cual es de 12V y 2mA ya que se consigue un rendimiento adecuado en ella.
- Se recomienda no colocar más de un microcontrolador PIC en la plataforma EasyPIC ya que puede causar reacciones desfavorables tanto en la placa como en el desarrollo de la práctica.
- Se recomienda leer las características técnicas de los dispositivos que se vayan a conectar adicionalmente en el módulo, sea el caso de los sensores, protoboards, motores, controlador de motor, display, PICs, de no ser así puede verse afectado los elementos.
- En caso de no estar conectado a internet, y desea utilizar la comunicación serial, se recomienda utilizar dos cables USB-C, uno que sea utilizado como alimentación y a la vez para grabar el microcontrolador y el otro debe conectarse al bloque USB TO UART hacia la computadora, para no tener el inconveniente de estar desconectando el mismo cable una y otra vez.
- Al utilizar software MATLAB y hacer uso de la herramienta GUIDE, tomar en cuenta los elementos y la configuración de cada uno ellos, ya que son independientes a la hora de realizar su programación.
- Se propone agregar nuevos contenidos al syllabus para renovar las prácticas de microcontroladores que cubran temas de la tecnología actual, los cuales estarán conectados a los diferentes periféricos del módulo. Además, se creará una guía de prácticas establecidas que cubrirá todos los temas planteados.

BIBLIOGRAFÍA

- [1] J. A. T. Gonzales, «Red de Equipos mFi para la seguridad del laboratorio de telecomunicaciones aplicando estandares internacionales de cableado y administracion en infraestructura de telecomunicaciones». Ecuador Patente 1, Enero 2019.
- [2] MikroElektronika, EasyPIC v8 Develoment Board, Serbia, 2019.
- [3] J. A. M. C. Kelly Viviana Gonzalez Rojas, «Diseño y Fabricacion de un modulo entrenador implementando la familia de microcontroladores dsPIC33F». Bucaramanga 18 Julio 2011.
- [4] C. A. Reyes, Mococontroladores PIC Programacion en Basic, Quito: Rispergraf, 2008.
- [5] J. P. H. L. R. I. B. Juan E. Guarella, «Sensores y Actuadores en Motores,» Universidad Nacional de la plata, La Plata, 2011.
- [6] L. M. V. g. Jose Luis Bucheli Naranjo, «Diseño e implementación de un modulo didactico con sistemas HMI para el analisis y el estudio de sensores y transductores de proximidad analogicos por medio de la tarjeta de adquisicion de datos DAQ NI y controlada por el software LABVIEW,» Quito, 2014, p. 6.
- [7] J. E. R. Ruiz, Diseño y construcción de un dispositivo electronico para detección de obstaculos, como ayuda a personas con discapacidad visual, Bogota, 2008.
- [8] A. P. P. Jeferson, «Control inteligente de temperatura de un departamento prototipo por ventilacion natural,» Quito, 2009, pp. 39, 40.
- [9] M. Hernando, «Tema 2: Morfologia del Robot,» Madrid, 2016, pp. 19, 20.
- [10] T. R. G. Hector Ulloa López, «Análisis y estudio de los sensores aplicados a la mecatrónica para practicas en el laboratorio de electrónica control y automatismo utilizando el modulo de entrenamiento ni-qnet-015,» Guayaquil, 2012, pp. 40-41.
- [11] J. A. S. Briones, «Prototipo de dispositivo electronico IOT para la adquisicion de diversos parametros fisicos en la transportacion publica,» La Liberta - Ecuador, 2019, p. 24.

- [12] H. A. M. C. Kelly Viviana Gonzales Rojas, Diseño y fabricacion de un modulo entrenador implementando la familia de Microcontroladores dsPIC33F, Bucaramanga, 2011.
- [13] J. J. I. González, Proyectos electronicos con microcontrolador PIC16F877A, Pamplona, 2017.
- [14] N. MECHATRONICS, «Tutorial Sensor de Distancia SHARP,» 2016. [En línea]. Available: https://naylorlampmechatronics.com/blog/55_tutorial-sensor-de-distancia-sharp.html -->. [Último acceso: 09 12 2019].
- [15] SHARP, GP2Y0A21YK0F, Estados Unidos, 2011.
- [16] Mikroelektronika, Manual MikroC.
- [17] F. R. L. J. L. Enrique Palacios, Microcontrolador PIC16F84. Desarrollo de proyectos, México: Alfaomega, 2004.
- [18] MikroElektronika, «MikroE,» 5 10 2011. [En línea]. Available: www.mikroe.com. [Último acceso: 20 09 2019].
- [19] F. R. L. J. L. Enrique palacios, Microcontrolador PIC 16F84 - Desarrollo de proyectos, Mexico: Alfaomega, 2004.
- [20] R. I. Chero, Breve Manual de Maxima, Málaga: texmaker, 2012.
- [21] J. A. O. H. Suset G. Rodriguez Alemán, «Interfaces Gráficas de Usuario en MATLAB».
- [22] M. B. Manuel Vargas, Introducción a MATLAB y su aplicacion al análisis y control de sistemas, Sevilla: Universidad de Sevilla, 2004.

ANEXOS

ANEXO A: GUÍA DE APLICACIONES PRÁCTICAS

ANEXO B:

RESUMEN DE INSTRUCCIONES EN ASSEMBLER

Instrucciones orientadas a byte:

Mnemonic, Operands	Description	Cycles	12-Bit Opcode			Status Affected
			MSb	LSb		
ADDWF f,d	Add W and f	1	0001	11df	ffff	C,DC,Z
ANDWF f,d	AND W with f	1	0001	01df	ffff	Z
CLRF f	Clear f	1	0000	011f	ffff	Z
CLRWF -	Clear W	1	0000	0100	0000	Z
COMF f,d	Complement f	1	0010	01df	ffff	Z
DECf f,d	Decrement f	1	0000	11df	ffff	Z
DECFSZ f,d	Decrement f, Skip if 0	1(2)	0010	11df	ffff	None
INCF f,d	Increment f	1	0010	10df	ffff	Z
INCFSZ f,d	Increment f, Skip if 0	1(2)	0011	11df	ffff	None
IORWF f,d	Inclusive OR W with f	1	0001	00df	ffff	Z
MOVF f,d	Move f	1	0010	00df	ffff	Z
MOVWF f	Move W to f	1	0000	001f	ffff	None
NOP -	No Operation	1	0000	0000	0000	None
RLF f,d	Rotate left f through Carry	1	0011	01df	ffff	C
RRF f,d	Rotate right f through Carry	1	0011	00df	ffff	C
SUBWF f,d	Subtract W from f	1	0000	10df	ffff	C,DC,Z
SWAPF f,d	Swap f	1	0011	10df	ffff	None
XORWF f,d	Exclusive OR W with f	1	0001	10df	ffff	Z

Instrucciones orientadas a bit:

Mnemonic, Operands	Description	Cycles	12-Bit Opcode			Status Affected
			MSb	LSb		
BCF f,b	Bit Clear f	1	0100	bbbf	ffff	None
BSF f,b	Bit Set f	1	0101	bbbf	ffff	None
BTFSC f,b	Bit Test f, Skip if Clear	1(2)	0110	bbbf	ffff	None
BTFSS f,b	Bit Test f, Skip if Set	1(2)	0111	bbbf	ffff	None

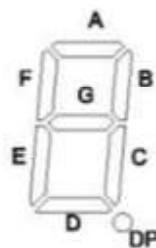
Instrucciones orientadas a literal y control:

Mnemonic, Operands	Description	Cycles	12-Bit Opcode			Status Affected
			MSb	LSb		
ANDLW k	AND literal with W	1	1110	kkkk	kkkk	Z
CALL k	Call subroutine	2	1001	kkkk	kkkk	None
CLRWDT k	Clear Watchdog Timer	1	0000	0000	0100	\overline{TO} , \overline{PD}
GOTO k	Unconditional branch	2	101k	kkkk	kkkk	None
IORLW k	Inclusive OR Literal with W	1	1101	kkkk	kkkk	Z
MOVLW k	Move Literal to W	1	1100	kkkk	kkkk	None
OPTION k	Load OPTION register	1	0000	0000	0010	None
RETLW k	Return, place Literal in W	2	1000	kkkk	kkkk	None
SLEEP -	Go into standby mode	1	0000	0000	0011	\overline{TO} , \overline{PD}
TRIS f	Load TRIS register	1	0000	0000	0fff	None
XORLW k	Exclusive OR Literal to W	1	1111	kkkk	kkkk	Z

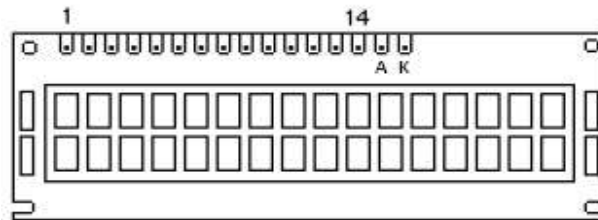
DISPLAY DE SIETE SEGMENTOS



Dígito hexadecimal presentado en el despliegue de 7 segmentos	b7	b6	b5	b4	b3	b2	b1	b0
	PB	G	F	E	D	C	B	A
0	1	1	0	0	0	0	0	0
1	1	1	1	1	1	0	0	1
2	1	0	1	0	0	1	0	0
3	1	0	1	1	0	0	0	0
4	1	0	0	1	1	0	0	1
5	1	0	0	1	0	0	1	0
6	1	0	0	0	0	0	1	0
7	1	1	1	1	1	0	0	0
8	1	0	0	0	0	0	0	0
9	1	0	0	1	0	0	0	0
A	1	0	0	0	1	0	0	0
b	1	0	0	0	0	0	1	1
C	1	1	0	0	0	1	1	0
d	1	0	1	0	0	0	0	1
E	1	1	0	0	0	1	1	0
F	1	0	0	0	1	1	1	0



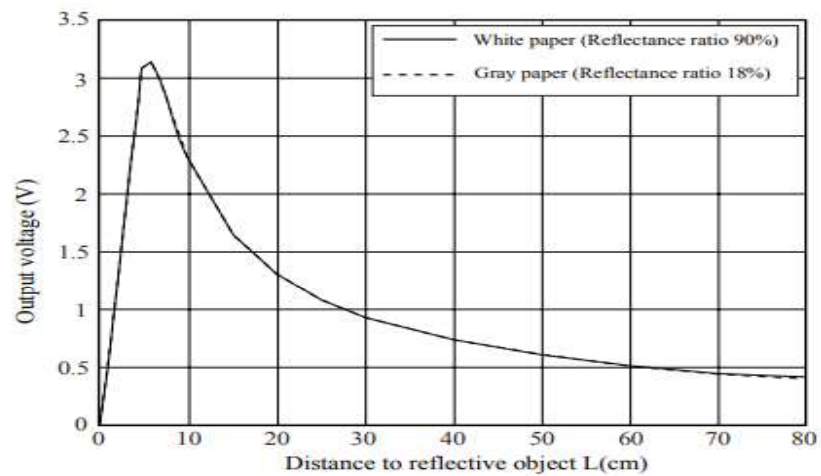
DISPLAY LCD 2X16



Pin No	Name	Description
1	V _{ss}	GND
2	V _{dd}	+5v
3	V _o	Contrast Control
4	RS	Register Select
5	R/W	Read/Write
6	E	Enable (<i>Strobe</i>)
7	D0	Data <i>LSB</i>
8	D1	Data
9	D2	Data
10	D3	Data
11	D4	Data
12	D5	Data
13	D6	Data
14	D7	Data <i>MSB</i>

SENSOR INFRARROJO DE PROXIMIDAD GP2Y0A21YK0F

Relación de Voltaje vs Distancia



Medidas del sensor

