

ESCUELA POLITÉCNICA NACIONAL

ESCUELA DE FORMACIÓN DE TECNÓLOGOS

CONSTRUCCIÓN DE UN DISPOSITIVO DE PROTECCIÓN PARA MOTORES TRIFÁSICOS DE INDUCCIÓN BASADO EN EL MICROCONTROLADOR PIC

PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE TECNÓLOGO EN ELECTROMECAÁNICA

Barros Aragón Miguel Ángel

mickiangeleo@hotmail.com

Tipán Quinatoa Sergio Adrián

sergio92adrian@gmail.com

DIRECTOR: Ing. Jorge Eduardo Rafael Loza Cedeño

jorge.loza@epn.edu.ec

Quito, Febrero, 2017

DECLARACIÓN

Nosotros, Barros Aragón Miguel Ángel y Tipán Quinatoa Sergio Adrián, declaramos bajo juramento que el trabajo aquí descrito es de nuestra autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que hemos consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedemos nuestros derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Barros Aragón Miguel Ángel

Tipán Quinatoa Sergio Adrián

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por los señores Barros Aragón Miguel Ángel y Tipán Quinatoa Sergio Adrián, bajo mi supervisión.

Ing. Loza Cedeño Jorge
DIRECTOR DE PROYECTO

AGRADECIMIENTOS

Ante todo a DIOS, por permitirme estar con vida y tener a los seres que más quiero junto a mí.

El agradecimiento es también para nuestro director de proyecto, que con su conocimiento ayudó a culminarlo. Y todas aquellas personas, ingenieros que con su valiosa colaboración hicieron posible el desarrollo del presente.

Miguel Barros

A nuestro Padre celestial, por regalarme un día más de vida junto a mis seres queridos.

Al Ing. Loza por su acertada guía durante la ejecución del proyecto; y en general a la Escuela de Formación de Tecnólogos que durante años nos brindó los conocimientos para aportar al desarrollo.

Sergio Tipán

DEDICATORIA

La concepción de este proyecto está dedicada a mi madre, padre y hermanas con mucho amor y cariño, pilares fundamentales en mi vida. Sin ellos, jamás hubiese podido conseguir lo que hasta ahora he logrado.

Miguel Barros

A mí querida abuela Dolores, quien con su inmensa sabiduría, cariño y paciencia nos educó a mis hermanos y a mí para ser mejores seres humanos.

A mis padres, Enrique y María por su apoyo incondicional y sus buenos consejos durante todos estos años de formación.

En especial a mi tío Nelson, aunque no estés con nosotros siempre te llevo presente día tras día.

Sergio Tipán

TABLA DE CONTENIDOS

INTRODUCCIÓN.....	1
OBJETIVOS.....	2
OBJETIVO GENERAL.....	2
OBJETIVOS ESPECÍFICOS.....	2
1. CAPITULO 1 MARCO CONCEPTUAL.....	3
1.1. SISTEMAS TRIFÁSICOS.....	3
1.1.1. FUNDAMENTOS.....	3
1.1.2. SISTEMAS TRIFÁSICOS BALANCEADOS.....	4
1.1.3. MOTOR TRIFÁSICO DE INDUCCIÓN.....	4
1.2. DESBALANCE DE VOLTAJE.....	5
1.2.1. DEFINICIÓN DEL DESBALANCE DE VOLTAJE SEGÚN LA ASOCIACIÓN NACIONAL DE FABRICANTES ELÉCTRICOS.....	5
1.2.2. DEFINICIÓN DEL DESBALANCE DE VOLTAJE SEGÚN EL INSTITUTO DE INGENIEROS ELÉCTRICOS Y ELECTRÓNICOS (IEEE).....	5
1.2.2. EFECTOS DEL DESBALANCE DE VOLTAJE SOBRE LOS MOTORES TRIFÁSICOS DE INDUCCIÓN.....	5
1.3. FALLO DE FASE.....	6
1.3.1. DEFINICIÓN.....	6
1.3.2. EFECTOS DEL FALLO DE FASE EN LOS MOTORES TRIFÁSICOS DE INDUCCIÓN.....	6
1.4. SECUENCIA DE FASES.....	7
1.4.1. DEFINICIÓN.....	7
1.5. MICROCONTROLADORES PIC.....	8
1.5.1. ¿QUÉ ES UN MICROCONTROLADOR?.....	8

1.5.2.	MICROCONTROLADOR (MCU).....	8
1.5.2.1.	CARACTERÍSTICAS GENERALES.....	9
1.5.2.2.	DISTRIBUCIÓN DE PINES.....	9
1.5.3.	ORGANIZACIÓN DE LA MEMORIA.....	13
1.5.4.	MÓDULO CONVERTIDOR DE 10 BITS ANÁLOGO A DIGITAL.....	13
1.5.5.	TIMER 1.....	14
1.5.5.1.	INTERRUPCIÓN DEL TIMER 1.....	15
1.5.6.	PANTALLA GRÁFICA DE CRISTAL LÍQUIDO GLCD.....	15
1.6.	MIKRO C PRO FOR PIC.....	16
1.6.1.	CARACTERÍSTICAS PRINCIPALES DE MIKROC PRO FOR PIC.....	17
1.6.2.	PROJECT SETTINGS (CONFIGURACIÓN DE PROYECTOS).....	17
1.6.3.	LIBRARY MANAGING (GESTIÓN DE LIBRERÍAS)	17
1.6.4.	CODE EDITOR (EDITOR DE CÓDIGO).....	18
1.6.5.	COMPILAR Y SOLUCIONAR LOS ERRORES.....	18
2.	CAPITULO 2 CONSTRUCCIÓN DEL DISPOSITIVO DE PROTECCIÓN.....	19
2.1.	ANÁLISIS DE FALLAS Y SUS VALORES PERMISIBLES.....	19
2.2.	DIAGRAMAS DE BLOQUES DEL DISPOSITIVO DE PROTECCIÓN.....	20
2.3.	FUENTE DE ALIMENTACIÓN DC (CORRIENTE DIRECTA).....	21
2.4.	DESARROLO DEL CIRCUITO MONITOR DE VOLTAJE.....	22
2.4.1.	ACONDICIONAMIENTO PARA LECTURA DE LA SEÑAL DE VOLTAJE ALTERNO.....	22
2.4.2.	RECTIFICADORES DE PRECISIÓN CON AMPLIFICADORES OPERACIONALES.....	25
2.4.2.1	RECTIFICADOR INVERSOR DE MEDIA ONDA LINEAL CON SALIDA NEGATIVA.....	25
2.4.2.2.	RECTIFICADOR DE PRECISIÓN O AMPLIFICADOR DE VALOR ABSOLUTO CON ENTRADAS CONECTADAS A TIERRA.....	28

2.4.3.	CONVERTIDOR DE AC A DC.....	29
2.4.4.	CIRCUITO ELÉCTRICO PARA EL ACONDICIONAMIENTO DE LA SEÑAL DE VOLTAJE AC.....	30
2.5.	CIRCUITO ELÉCTRICO PARA EL ACONDICIONAMIENTO DE LAS SEÑALES PARA IDENTIFICACIÓN DE SECUENCIA FASES.....	31
2.6.	UNIDAD DE POTENCIA.....	33
2.6.1.	MÓDULO RELÉ.....	33
2.6.2.	DIMENSIONAMIENTO DEL CONTACTOR.....	34
2.7.	UNIDAD DE PROCESAMIENTO Y CONTROL.....	37
2.7.1.	ANÁLISIS DE LOS RECURSOS DEL PIC 18F4550.....	37
2.7.2.	ASIGNACIÓN DE PINES.....	38
2.7.3.	CONFIGURACIÓN DEL OSCILADOR PRINCIPAL.....	39
2.7.4.	INTERFAZ DEL MÓDULO GLCD Y EL PIC18F4550.....	41
2.8.	DESARROLLO DEL SOFTWARE PARA EL MICROCONTROLADOR.....	43
2.8.1.	CONFIGURACIÓN DEL MICROCONTROLADOR PIC18F4550.....	44
2.8.2.	IMÁGENES Y MENSAJES DE PRESENTACIÓN.....	49
2.8.3.	RUTINA PARA VERIFICAR LA CONEXIÓN DE LAS 3 FASES.....	52
2.8.4.	RUTINA PARA IDENTIFICAR LA SECUENCIA DE FASES.....	54
2.8.5.	PROGRAMA PRINCIPAL.....	56
2.8.5.1.	SUBRUTINA PARA LEER EL MÓDULO A/D Y CALCULAR EL VALOR DE VOLTAJE.....	57
2.8.5.2.	CÁLCULO DEL DESBALANCE DE VOLTAJE.....	57
2.8.5.3.	DETECCIÓN DE FALLAS.....	58
2.8.5.4.	VISUALIZACIÓN DE PARÁMETROS DE MEDICIÓN.....	61
2.9.	MENSAJES ESCRITOS EN LA MEMORIA EEPROM.....	63

3. CAPITULO 3 IMPLEMENTACIÓN DEL DISPOSITIVO DE PROTECCIÓN	66
3.1. DISEÑO DE PISTAS EN ARES.....	66
3.2. EXPORTACIÓN E IMPRESIÓN DEL CIRCUITO.....	68
3.3. FABRICACIÓN DEL CIRCUITO IMPRESO.....	71
3.4. MONTAJE DEL DISPOSITIVO DE PROTECCIÓN.....	73
3.5. RECURSOS UTILIZADOS.....	75
4. CAPITULO 4 PRUEBAS DE FUNCIONAMIENTO Y ANÁLISIS DE RESULTADO	78
4.1. INTRODUCCIÓN.....	78
4.2. CONDICIONES INICIALES PARA LAS PRUEBAS.....	79
4.3. IDENTIFICACIÓN DE LA SECUENCIA DE FASES.....	81
4.4. PÉRDIDA DE FASE.....	83
4.5. DESBALANCE DE VOLTAJE.....	84
4.6. SOBRE Y BAJO VOLTAJES.....	84
5. CAPÍTULO 5 CONCLUSIONES Y RECOMENDACIONES	86
5.1. CONCLUSIONES.....	86
5.2. RECOMENDACIONES.....	87
BIBLIOGRAFÍA	88
ANEXOS	89
ANEXO A: AMPLIFICADOR OPERACIONAL μ A741.....	90
ANEXO B: COMPARADOR DE VOLTAJE LM339.....	92
ANEXO C: FOTOTRANSISTOR DPC-817C.....	94
ANEXO D: CONTACTOR LC1D12M7.....	96
ANEXO E: PLANOS DIAGRAMAS ELÉCTRICOS Y ELECTRÓNICOS.....	97
ANEXO F: CÓDIGO FUENTE.....	100

ÍNDICE DE FIGURAS

Figura 1.1 Generador Trifásico.....	3
Figura 1.2 Voltajes generados desfasados 120° entre sí.....	3
Figura 1.3 Campo magnético giratorio en la carcasa.....	4
Figura 1.4 Motor trifásico de jaula de ardilla.....	4
Figura 1.5 Secuencia positiva <i>abc</i>	7
Figura 1.6 Secuencia negativa <i>acb</i>	7
Figura 1.7 Microcontrolador PIC.....	8
Figura 1.8 Distribucion de pines del PIC18F4550.....	9
Figura 1.9 Función de transferencia del conversor A/D.....	14
Figura 1.10 Diagrama del TIMER1 (Modo de lectura/escritura 16 bits).....	14
Figura 1.11 Pantalla gráfica de cristal líquido GLCD 128x64.....	15
Figura 1.12 Entorno de desarrollo en Mikro C for PIC.....	17
Figura 1.13 Ventana de mensajes después de compilar.....	18
Figura 2.1 Diagrama de bloques del dispositivo de protección.....	20
Figura 2.2 Circuito de la fuente de voltaje DC +5 V.....	21
Figura 2.3 Circuitos de las fuentes de voltajes DC	22
Figura 2.4 Amplificador operacional uA741.....	23
Figura 2.5 Transformador de potencial y divisor de voltaje.....	23
Figura 2.6 Rectificador inversor de media onda lineal con salida negativa.....	25
Figura 2.7 La salida V_o es negativa y de igual magnitud de V_{in} para el semiciclo positivo.....	26
Figura 2.8 La salida V_o limitada a 0 V para el semiciclo negativo.....	27
Figura 2.9 Formas de onda del circuito rectificador de precisión de media onda.....	28
Figura 2.10a Amplificador de valor absoluto semiciclo positivo.....	28
Figura 2.10b Amplificador de valor absoluto semiciclo negativo.....	29
Figura 2.11 Rectificador de onda completa de precisión, formas de onda.....	29
Figura 2.12 Convertidor de CA a CD o amplificador de valor absoluto medio (MAV)...	30
Figura 2.13 Diagrama eléctrico del circuito de acondicionamiento de señal AC.....	31

Figura 2.14 Circuito detector de cruce por cero.....	32
Figura 2.15 Forma de onda del circuito detector de cruce por cero.....	32
Figura 2.16 Esquema y configuración de pines del DPC-817C.....	33
Figura 2.17 Circuito esquemático del módulo relé.....	34
Figura 2.18 Contactor usado para la conexión/desconexión del motor.....	36
Figura 2.19 Asignación de pines.....	38
Figura 2.20 Configuración del oscilador primario.....	39
Figura 2.21 Diagrama de bloques del oscilador y su configuración.....	41
Figura 2.22 Diagrama eléctrico de la interfaz PIC y módulo GLCD.....	42
Figura 2.23 Diagrama de flujo general.....	43
Figura 2.24 Configuración del Registro ADCON1.....	44
Figura 2.25 Configuración del Registro T1CON.....	46
Figura 2.26 Edición de imagen.....	49
Figura 2.27 Herramienta Graphic LCD bitmap editor.....	50
Figura 2.28 Imagen desplegada en el GLCD.....	51
Figura 2.29 Información del proyecto.....	51
Figura 2.30 Pantalla y señales luminosas de líneas conectadas.....	52
Figura 2.31 Ondas cuadradas correspondientes a cada fase.....	54
Figura 2.32 Secuencia de fases positiva, formas de onda.....	55
Figura 2.33 Secuencia de fases negativa, formas de onda.....	55
Figura 2.34 Resultado de la identificación de secuencia de fases.....	56
Figura 2.35 Límites de voltajes para motores según norma IEC60034-1.....	59
Figura 2.36 Impresión de los valores de voltaje y el desbalance.....	63
Figura 2.37 Ventana correspondiente a EEPROM Editor.....	64
Figura 3.1 Vista del entorno de desarrollo <i>ARES</i>	66
Figura 3.2 Creación del primer PCB.....	67
Figura 3.3 PCB de la fuente de alimentación DC.....	68
Figura 3.4 PCB del bloque de acondicionamiento de señal AC.....	69

Figura 3.5 PCB de la tarjeta principal, contiene el PIC18F4550.....	70
Figura 3.6 Transferencia de las pistas en la baquelita.....	72
Figura 3.7 Ubicación y soldadura de los componentes.....	73
Figura 3.8 Vista de los circuitos impresos terminados.....	73
Figura 3.9 Vista exterior del dispositivo.....	74
Figura 3.10 Ubicación interna de los componentes del dispositivo	74
Figura 4.1 Conexión del dispositivo para las pruebas.....	78
Figura 4.2 Recursos, equipos e instrumentos para las pruebas.....	79
Figura 4.3 Curva de respuesta de las lecturas de voltaje.....	81
Figura 4.4 Conexiones para pruebas de secuencia de fases	82
Figura 4.5 Resultado de la identificación de la secuencia positiva.....	82
Figura 4.6 Diagrama de bloques para imitar pérdida de fase.....	83
Figura 4.7 Lectura de la pérdida de fase.....	83
Figura 4.8 Lectura del desbalance de voltaje.....	84
Figura 4.9 Límites de voltajes.....	85
Figura 4.10 Detección de un voltaje fuera de rango y un bajo voltaje.....	85

ÍNDICE DE TABLAS

Tabla 1.1 Descripción del puerto A.....	10
Tabla 1.2 Descripción del puerto B.....	11
Tabla 1.3 Descripción del puerto C.....	11
Tabla 1.4 Descripción del puerto D.....	12
Tabla 1.4 Descripción del puerto E.....	12
Tabla 1.6 Descripción de los pines del módulo glcd 128x64.....	16
Tabla 2.1 Cargas con sus respectivas fuentes.....	21
Tabla 2.2 Características del fototransistor DPC-817C.....	34
Tabla 2.3 Datos del motor.....	36
Tabla 2.4 Entradas analógicas y digitales.....	37
Tabla 2.5 Salidas digitales.....	38
Tabla 2.6 Recursos necesarios.....	38
Tabla 2.7 Múltiples frecuencias de trabajo del CPU.....	40
Tabla 2.8 Relación de pines del PIC y el GLCD.....	41
Tabla 2.9 Mensajes escritos en la memoria EEPROM.....	64
Tabla 4.1 Placa de datos del motor.....	80
Tabla 4.2 Análisis de voltajes entre el instrumento patrón y el dispositivo.....	80

INTRODUCCIÓN

Los motores eléctricos son dispositivos electromecánicos usados en la mayoría de aplicaciones industriales para la conversión de la energía eléctrica a mecánica. Entre ellos sobresalen los motores de inducción; son máquinas robustas usadas en todo tipo de aplicaciones y, además, son altamente confiables, requiere poco mantenimiento y se encuentran en el mercado en una amplia gama de potencias para satisfacer las necesidades de los procesos industriales.

Sin embargo, cuando el motor trifásico de inducción es alimentado por el sistema trifásico de voltajes desbalanceados por un largo periodo de tiempo, se pueden producir efectos anormales en su funcionamiento. Debido a la simetría de su estructura electromagnética interna se comporta como un conjunto equilibrado de impedancias, por lo tanto, solo aparecerán desequilibrios en las corrientes, las que producen en cada fase campos magnéticos de diferente magnitud.

Consecuentemente, para un determinado porcentaje de desbalance de voltaje y una carga dada, las corrientes en las fases y la elevación de la temperatura serán mucho mayores que cuando operan bajo las mismas condiciones de carga, pero con voltajes balanceados, afectando por lo tanto la potencia nominal del motor. Además, la pérdida de una de las líneas de alimentación, por algún problema externo al motor, hará que la máquina permanezca girando y moviendo la carga, y las 2 fases que permanecen energizadas tienen un incremento de corriente de aproximadamente el 75%; ésto produce excesiva vibración, ruido fuera de lo normal y calentamiento en los bobinados por la sobrecarga adquirida de forma repentina.

Por último, el sentido de giro del motor trifásico está gobernado por la secuencia de fases; y si dos de las tres líneas que energizan a dicho motor son invertidas, causará que gire en el sentido contrario. Por ejemplo, en las bombas de agua, si la máquina gira en sentido opuesto al predeterminado, el impulsor se desenroscará en el extremo del eje del motor produciendo daños a la propia máquina.

Existen en el mercado varios dispositivos de protección electromecánicos que solventan algunos de estos fallos, pero son insensibles a las variaciones de voltaje de la red eléctrica, se deben calibrar con mucha precisión y son relativamente costosos. El presente proyecto propone construir el dispositivo de protección para la identificación de la pérdida de fase, desbalance de voltaje y la inversión en la secuencia de fases, utilizando un microcontrolador PIC con la finalidad de que el dispositivo opere antes de que ocurra algún daño.

OBJETIVOS

OBJETIVO GENERAL

Construir un dispositivo de protección para motores trifásicos de inducción basado en el microcontrolador PIC.

• OBJETIVOS ESPECÍFICOS

- Especificar la información relacionada con el desempeño del motor trifásico de inducción frente a los fallos: la pérdida de fase, el desbalance de voltaje y la inversión en la secuencia de fases.
- Identificar los valores y porcentajes admisibles en cada fallo en el sistema.
- Definir los métodos para monitorear el voltaje e identificar la secuencia de fases.
- Modelar los circuitos acondicionadores de señales utilizando como herramienta el software ISIS Proteus.
- Desarrollar un programa para supervisar las fallas durante y después de la puesta en marcha del motor, empleando un microcontrolador y el compilador Mikro C for PIC.
- Realizar las simulaciones en el programa ISIS Proteus, verificando la correcta ejecución del programa desarrollado; determinando su efectividad.
- Dimensionar el actuador en función de la potencia del motor y las señales de respuesta del microcontrolador producida por los fallos.
- Determinar los circuitos impresos necesarios para el asentamiento de los elementos que conformarán el dispositivo de protección.
- Evaluar la funcionalidad del dispositivo de protección mediante pruebas que simulen los fallos para las que está diseñado

CAPITULO 1

MARCO CONCEPTUAL

1.1. SISTEMAS TRIFÁSICOS

1.1.1. FUNDAMENTOS [2]

Los sistemas trifásicos son sistemas de producción, distribución y consumo de energía eléctrica formado por un generador trifásico de corriente alterna (CA) o tres fuentes independientes de voltaje monofásicos. Este generador consta básicamente de un imán giratorio (rotor) rodeado por un devanado estacionario (estator), tres devanados o bobinas independientes con terminales a-a', b-b', y c-c' que se disponen físicamente alrededor del estator a 120° de distancia entre sí.

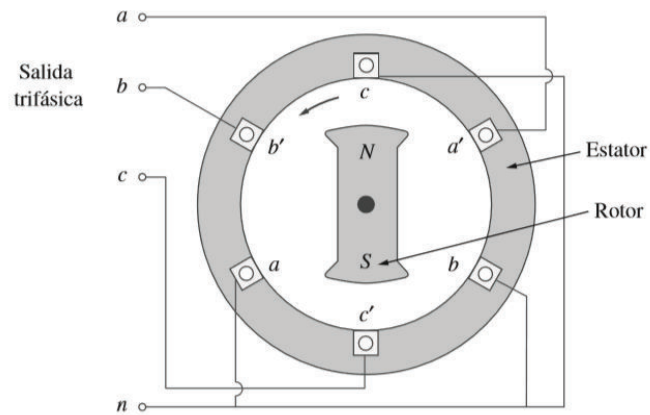


Figura 1.1 Generador Trifásico. [2]

Al girar el rotor, su campo magnético “corta” el flujo de las tres bobinas e induce voltajes en ellas. A causa de que las bobinas se hallan a 120° de distancia entre sí, los voltajes inducidos en ellas son iguales en magnitud desfasados 120° y se encuentran en un orden determinado (secuencia definida).

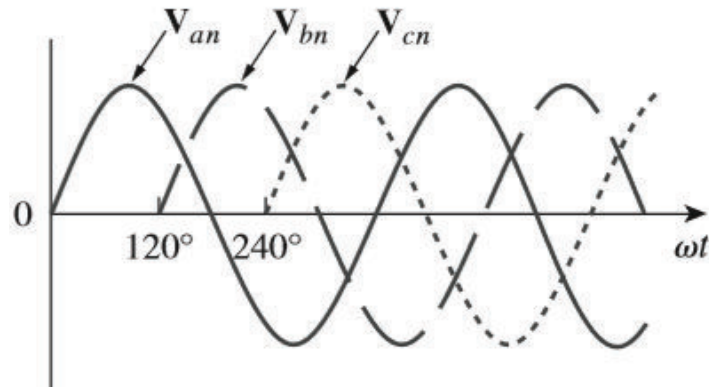


Figura 1.2 Voltajes generados desfasados 120° entre sí. [2]

1.1.2. SISTEMAS TRIFÁSICOS BALANCEADOS

En los sistemas trifásicos balanceados sus corrientes son iguales, por ende, se puede hablar de que sus cargas están balanceadas o equilibradas; caso contrario, si sus corrientes son desiguales se tiene un sistema desbalanceado o desequilibrado. [10]

1.1.3. MOTOR TRIFÁSICO DE INDUCCIÓN [3]

El motor trifásico de inducción también conocido como motor asíncrono, es hoy en día el motor eléctrico que más se utiliza en las aplicaciones industriales, sobre todo el motor con rotor de jaula de ardilla por su simplicidad, confiabilidad y bajo costo.

El motor trifásico basa su funcionamiento en la inducción electromagnética. Es así que cuando se conecta el motor a la red trifásica, se establece un campo magnético giratorio alrededor del grupo o núcleo de chapas del estator.

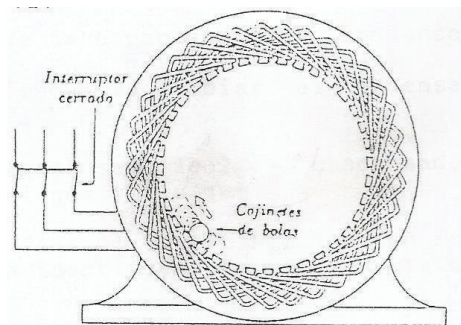


Figura 1.3 Campo magnético giratorio en la carcasa. [3]

Cuando el motor está completo, este campo induce voltaje en las barras del rotor jaula de ardilla; a su vez este voltaje hace que circule corriente por las barras produciendo otro campo magnético alrededor de las mismas. Entonces, la interacción de los 2 campos magnéticos, el de las barras del rotor y el principal que está girando alrededor del núcleo, hace que el rotor gire y desarrolle un torque en su respectivo eje.

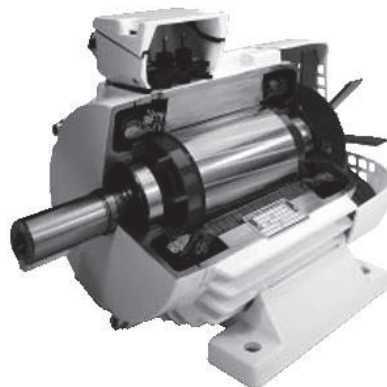


Figura 1.4 Motor trifásico de jaula de ardilla. [3]

1.2. DESBALANCE DE VOLTAJE

1.2.1. DEFINICIÓN DEL DESBALANCE DE VOLTAJE SEGÚN LA ASOCIACIÓN NACIONAL DE FABRICANTES ELÉCTRICOS (NEMA). [5]

Según la NEMA, el desbalance de voltaje es la relación entre la desviación máxima del voltaje promedio y el voltaje promedio de las tres líneas.

$$\% LVUR = \frac{\text{Desviación máxima del promedio de } \{V_{ab}, V_{bc}, V_{ca}\}}{\text{Promedio de } \{V_{ab}, V_{bc}, V_{ca}\}} \times 100 \% \quad (1.1)$$

Donde:

% LVUR: Porcentaje de desbalance de voltaje de línea, con sus siglas en inglés “Line Voltage Unbalance Rate (LVUR)”.

V_{ab} , V_{bc} y V_{ca} son los voltajes de línea del sistema eléctrico que alimenta el motor.

1.2.2. DEFINICIÓN DEL DESBALANCE DE VOLTAJE SEGÚN EL INSTITUTO DE INGENIEROS ELÉCTRICOS Y ELECTRÓNICOS (IEEE). [5]

La IEEE define el desbalance de voltaje como:

$$\% PVUR = \frac{\text{Desviación máxima del promedio de } \{V_a, V_b, V_c\}}{\text{Promedio de } \{V_a, V_b, V_c\}} \times 100 \% \quad (1.2)$$

Donde:

% PVUR: Porcentaje de desbalance de voltaje de fase, con sus siglas en inglés “Phase Voltage Unbalance Rate (PVUR)”.

V_a , V_b y V_c son los voltajes fase-neutro del sistema que alimenta el motor.

La IEEE y NEMA usan la misma definición para el desbalance de voltaje, la única diferencia es: NEMA utiliza el voltaje entre líneas y la IEEE el voltaje entre fase-neutro.

Por otra parte, ambas definiciones extraen la información de los ángulos de fase y solo consideran magnitudes.

1.2.3. EFECTOS DEL DESBALANCE DE VOLTAJE SOBRE LOS MOTORES TRIFÁSICOS DE INDUCCIÓN [5] [6]

Los efectos adversos de este fallo sobre estos motores han sido verificados experimentalmente y documentados enfocándose en las siguientes variables:

El torque de arranque y el torque máximo disminuyen cuando el voltaje es desbalanceado; por lo tanto, si el desbalance de voltaje es severo, es probable que el torque del motor ya no sea adecuado para la aplicación.

La velocidad a plena carga disminuye levemente si el motor es alimentado con voltaje desbalanceado.

La corriente de rotor bloqueado tendrá un desbalance similar al desbalance de voltaje, pero los KVA de rotor bloqueado se incrementarán levemente.

Cuando el motor está operando con una carga normal, el porcentaje de desbalance de corrientes causado será aproximadamente de 6 a 10 veces el porcentaje del desbalance de voltaje.

Por otra parte, un pequeño desbalance de voltaje en las fases podría causar un incremento significativo en la temperatura del motor, llegando a disminuir la vida del aislante en aproximadamente la mitad por cada 10°C de incremento de temperatura de estos.

1.3. FALLO DE FASE

1.3.1. DEFINICIÓN [1]

El fallo de una fase es la interrupción de energía en una línea de alimentación cualquiera hacia la carga.

Entonces, el motor sigue girando con solo dos fases y puede sufrir daños. La causa es, por ejemplo, un fusible que se funde. Los motores de tamaño pequeño y mediano son más críticos en el estator; esto significa que solo puede sufrir daños el estator

1.3.2. EFECTOS DEL FALLO DE FASE EN LOS MOTORES TRIFÁSICOS DE INDUCCIÓN [1]

Los efectos potenciales a los que puede conducir la pérdida de fase son similares a aquellos producidos por el desbalance de voltaje; sin embargo, la pérdida de una fase representa uno de los peores casos de desbalance de voltaje. Adicionalmente, en los devanados de las dos fases restantes que quedan en operación se produce:

- Calentamiento excesivo.
- Calentamiento en el rotor (barras del rotor y problemas con chumaceras y lubricantes).
- Incremento desproporcionado de la temperatura del motor.

1.4. SECUENCIA DE FASES

1.4.1. DEFINICIÓN [2]

La secuencia de fases es el orden temporal en que los voltajes pasan por sus respectivos valores máximos, y está determinada por el orden en que los fasores pasan por un punto fijo en el diagrama de fases.

Para la secuencia de fases positiva, los fasores deben girar en sentido contrario a las manecillas del reloj con la frecuencia ω (Figura 1.5); pasan por el eje horizontal en una secuencia *abcabca*... Así, esta secuencia es *abc*, *bca* o *cab*.

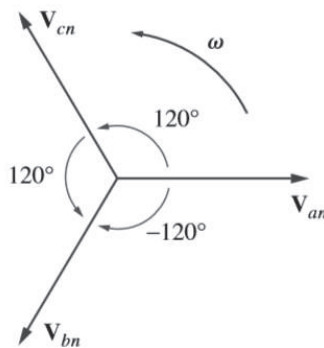


Figura 1.5 Secuencia positiva *abc*. [2]

De igual manera, en cuanto a los fasores de la figura (Figura. 1.6), al girar en dirección contraria a las manecillas del reloj pasan por el eje horizontal en una secuencia *acbacba*... Esto describe a la secuencia negativa *acb*, *cba* o *bac*.

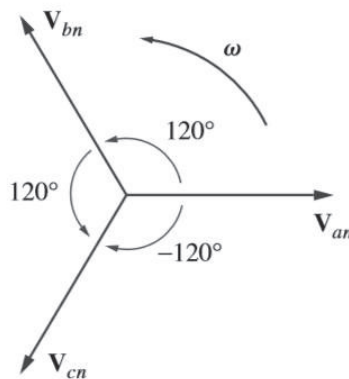


Figura 1.6 Secuencia negativa *acb*. [2]

La inversión en la secuencia de fases ocasiona que los motores trifásicos giren en dirección opuesta, lo cual puede causar lesiones al personal de operación y desperfectos al propio equipo.

NOTA: El sentido de giro de los motores trifásicos es posible invertirlo cambiando las conexiones de línea de dos terminales cualesquiera del estator, ya que simultáneamente se cambia el sentido de giro del campo magnético giratorio.

1.5. MICROCONTROLADORES PIC

1.5.1. ¿QUÉ ES UN MICROCONTROLADOR? [7]

Un microcontrolador es un circuito integrado, en cuyo interior posee toda la arquitectura de un computador, esto es CPU, memorias RAM, EEPROM, memoria de programa, circuitos de entrada y salidas, etc.

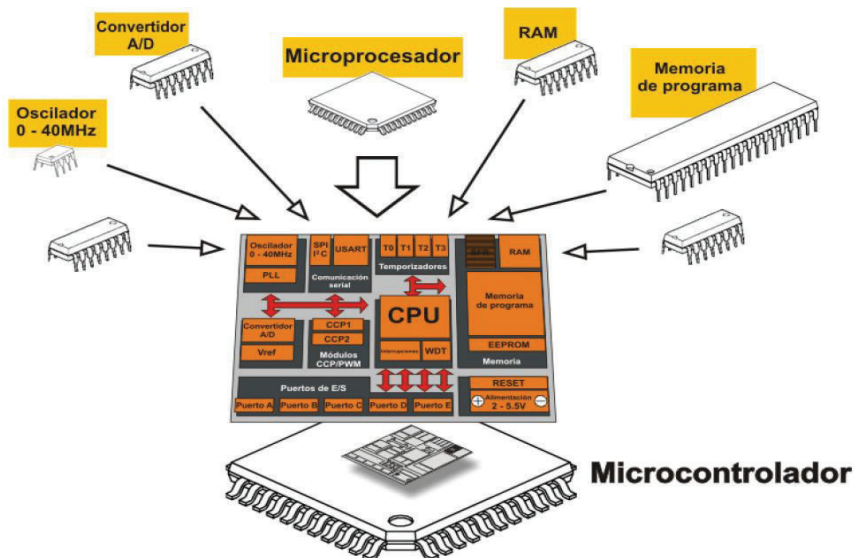


Figura 1.7 Microcontrolador PIC. [8]

Un microcontrolador de fábrica, no realiza tarea alguna, éste debe ser programado para que realice un simple parpadeo de un LED hasta un sofisticado control de un robot.

1.5.2. MICROCONTROLADOR (MCU)

El PIC 18F4550 es un microcontrolador de propósito general, diseñado y fabricado por la empresa *Microchip Technology*. El PIC 18F4550 se encuentra en la gama de los microcontroladores de 8 bits. Sus características de memoria de instrucciones, memoria de datos, número de entradas/salidas digitales, número de canales analógicos, temporizadores/contadores y tipos de puertos de comunicación, han hecho de este PIC uno de los más usados para diversas aplicaciones de sistemas empotrados. Además, *Microchip* pone a disposición la hoja de datos de cada uno de sus productos de forma gratuita, la cual se puede descargar directamente desde su página web: www.microchip.com.

1.5.2.1. CARACTERÍSTICAS GENERALES

Las características principales del PIC18F4550 son las siguientes:

- Juego de 77 instrucciones.
- Frecuencia máxima de operación de 48 MHz.
- Memoria Flash de 32 Kbytes.
- Memoria RAM de 2 Kbytes.
- Memoria EEPROM de 256 Bytes.
- 20 fuentes de interrupción.
- Encapsulado de 40 o 44 pines, de los cuales 35 son de entrada-salida divididos en 5 puertos (A, B, C, D, E).
- Un temporizador de 8 bits y tres de 16 bits.
- Módulo CCP de 16 bits para comparación, captura y generación de señal PWM de 10 bits.
- Módulos USB, USART y MSSP para comunicación I2C o SPI.
- Módulo 13 canales de conversión analógico-digital, con resolución de 10 bits.
- 2 comparadores.
- Voltaje de operación entre 2 y 5.5 volts.
- Arquitectura de 8 bits.
- Oscilador interno configurable entre 32 kHz y 8 MHz.
- Prioridad en interrupciones.
- Watchdog Timer.
- Detección de voltaje Alto/Bajo.

1.5.2.2. DISTRIBUCIÓN DE PINES

En la Figura 1.8 se muestra la distribución de pines del PIC18F4550.

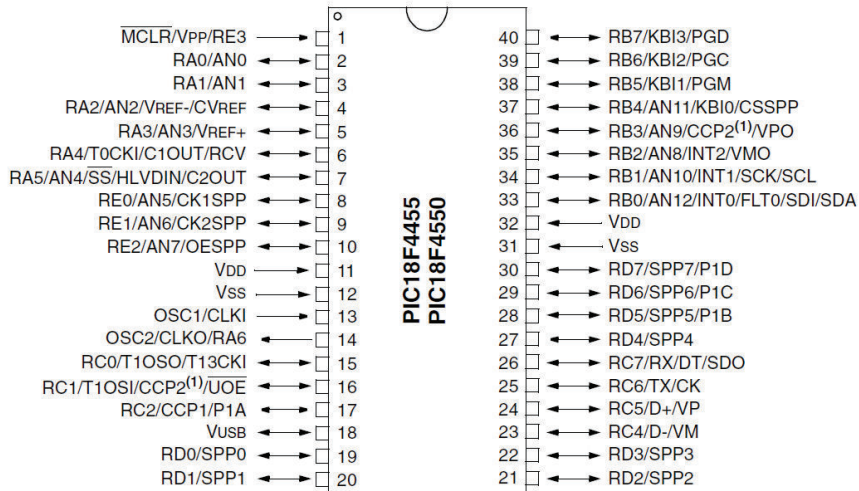


Figura 1.8 Distribución de pines del PIC18F4550. [7]

A continuación, en las siguientes tablas se resume las funciones de cada puerto y sus respectivos pines, extraídas de la hoja de datos proporcionada por el fabricante [7].

➤ Puerto A

Es un puerto bidireccional de Entrada/Salida (E/S).

Nombre del Pin	Descripción
RA0/AN0	E/S Digital/ Entrada analógica 0
RA1/AN1	E/S Digital/ Entrada analógica 1
RA2/AN2/V _{REF-} /CV _{REF}	E/S Digital/ Entrada analógica 2/ Entrada V _{REF-} A/D/ Salida V _{REF-} Comparador
RA3/AN3/V _{REF+}	E/S Digital/ Entrada analógica 3/ Entrada V _{REF+} A/D
RA4/T0CKI/C1OUT/RCV	E/S Digital/ Entrada de reloj Timer0/ Salida de Comparador 1/ Entrada RCV
RA5/AN4/SS/HLVDIN/C2OUT	E/S Digital/Entrada analógica 4/Entrada SPI/Entrada de detección voltaje alto/bajo/Salida de Comparador 2
RA6	Entrada del oscilador principal (OSC2)/salida de señal de reloj (CLKO)

Tabla 1.1 Descripción del puerto A.

Después del reset, los comparadores y el convertidor Analógico/Digital configuran el respectivo pin en modo analógico. Para poder utilizar un pin en modo digital hay que desactivar la función analógica.

➤ Puerto B

Es un puerto bidireccional de E/S. Los pull-ups internos (resistencias) pueden activarse o desactivarse por software en todas las entradas. Si un pin se configura como salida, la resistencia pull-up correspondiente se desactiva automáticamente.

Nombre del Pin	Descripción
RB0/AN12/INT0/FLT0/SDI/SDA	E/S Digital / Entrada analógica 12 / Interrupción externa 0 / Entrada de fallo del ECCP / Entrada de datos del SPI / E/S de I2C.
RB1/AN10/INT1/SCK/SCL	E/S Digital / Entrada analógica 10 / Interrupción externa 1 / Línea de reloj del SPI / Línea de reloj del I2C
RB2/AN8/INT2/VMO	E/S Digital / Entrada analógica 8 / Interrupción externa 2 / Salida de datos del USB (VCMO)
RB3/AN9/CCP2/VPO	E/S Digital / Entrada analógica 9 / E/S de CCP2 / Salida de datos del USB (VPO)

RB4/AN11/KBI0/CSSPP	E/S Digital / Entrada analógica 11 / Interrupción por cambio en pin (KBI0) / Salida de CS del SSP (CSSP)
RB5/KBI1/PGM	E/S Digital / Interrupción por cambio en pin (KBI1) / Línea de programación (PGM)
RB6/KBI2/PGC	E/S Digital / Interrupción por cambio en pin (KBI2) / Línea de programación (PGC)
RB7/KBI3/PGD	E/S Digital / Interrupción por cambio en pin (KBI3) / Línea de programación (PGD)

Tabla 1.2 Descripción del puerto B.

Una característica importante de este puerto es el manejo de interrupciones externas. Las terminales RB0, RB1 y RB2 aceptan solicitudes de interrupción externa, por flancos de subida o bajada. Por su parte, los terminales RB4 a RB7 pueden detectar un cambio en el nivel lógico (digital) de la señal de entrada para generar una interrupción.

➤ Puerto C

Es un puerto bidireccional de E/S digital. Los pines del puerto C son compartidos por dispositivos de entrada y salida (temporizadores, módulo CCP y módulo USB).

Nombre del Pin	Descripción
RC0/T1OSO/T13CKI	E/S Digital / Salida del oscilador del Temporizador 1 / Entrada del reloj externo Timer 1 y Timer 3
RC1/T1OSI/CCP2/ UOE	E/S Digital / Entrada del oscilador del Temporizador 1 / E/S del módulo CCP2 / Salida OE del transceptor del USB (UOE)
RC2/CCP1/P1A	E/S Digital / E/S del módulo CCP1 / Salida PWM del ECCP1 (P1A)
RC4/D-/VM	Entrada Digital / Línea menos del bus USB (D-) / Línea de entrada del USB (VM)
RC5/D+/VP	Entrada Digital / Línea más del bus USB (D+) / Línea de entrada del USB (VP)
RC6/TX/CK	E/S Digital / Salida de transmisión del EUSART (TX) / Línea de reloj del EUSART (CK)
RC7/RX/DT/SDO	E/S Digital / Entrada de recepción del EUSART (RX) / Línea de datos síncrona del EUSART (DT) / Salida de datos del SPI (SDO)

Tabla 1.3 Descripción del puerto C.

Después del reset, todos los pines quedan configurados como entradas.

➤ Puerto D

Puede funcionar como puerto bidireccional de E/S digital o como puerto SPP (Streaming Parallel Port). Esos pines tienen buffers de entrada cuando el módulo SPP está habilitado.

Nombre del Pin	Descripción
RD0/SPP0	E/S Digital / Línea de datos del SPP (SPP0)
RD1/SPP1	E/S Digital / Línea de datos del SPP (SPP1)
RD2/SPP2	E/S Digital / Línea de datos del SPP (SPP2)
RD3/SPP3	E/S Digital / Línea de datos del SPP (SPP3)
RD4/SPP4	E/S Digital / Línea de datos del SPP (SPP4)
RD5/SPP5/P1B	E/S Digital / Línea de datos del SPP (SPP5) / Salida PWM del ECCP1 (P1B)
RD6/SPP6/P1C	E/S Digital / Línea de datos del SPP (SPP6) / Salida PWM del ECCP1 (P1C)
RD7/SPP7/P1D	E/S Digital / Línea de datos del SPP (SPP7) / Salida PWM del ECCP1 (P1D)

Tabla 1.4 Descripción del puerto D.

➤ Puerto E

Es un puerto bidireccional de E/S digital. En el reset, RE2 a RE0 se configuran como entradas analógicas.

Nombre del Pin	Descripción
RE0/AN5/CK1SPP	E/S Digital / Entrada analógica 5 / Salida de reloj 1 del SPP
RE1/AN6/CK2SPP	E/S Digital / Entrada analógica 6 / Salida de reloj 2 del SPP
RE2/AN7/OESPP	E/S Digital / Entrada analógica 7 / Salida de habilitación del SPP
RE3	Línea de <i>reset</i> externo MCLR / Línea de programación (VPP)
V _{SS}	GND, referencia a tierra
V _{DD}	Fuente de alimentación DC positivo
V _{USB}	Salida del regulador de voltaje interno USB 3.3V

Tabla 1.5 Descripción del puerto E.

1.5.3. ORGANIZACIÓN DE LA MEMORIA

Existen 3 tipos de memoria en los PIC18F:

- Memoria de programa
- RAM de datos
- EEPROM de datos

Como dispositivos de arquitectura Harvard, los buses de la memoria de datos y de programa están separados; esto permite el acceso a la vez en las dos memorias. La EEPROM de datos, en la práctica, se puede utilizar como un dispositivo periférico, puesto que se maneja a través de un sistema de registros de control.

- El PIC18F4550 tiene 32KBytes de memoria Flash y puede almacenar hasta 16.384 instrucciones de palabra única.
- 2.048 Bytes de memoria SRAM interna en la cual se incluyen registros de función especial, almacena datos de forma temporal durante la ejecución del programa.
- Memoria EEPROM de datos (memoria no volátil) de 256 Bytes
- Pila: bloque de 31 palabras de 21 bits.
- Memoria de configuración en la que se incluye la palabra de configuración del dispositivo y los registros de identificación.

1.5.4. MÓDULO CONVERTIDOR DE 10 BITS ANÁLOGO A DIGITAL (A/D)

El módulo conversor de analógico a digital (A/D) tiene 13 entradas para el PIC18F4550. Este módulo permite la conversión de una señal de entrada analógica a un número de 10 bits. El módulo tiene 5 registros:

- ADRESH (A/D Result High Register).
- ADRESL (A/D Result Low Register).
- ADCON0 (A/D Control Register 0).
- ADCON1 (A/D Control Register 1).
- ADCON2 (A/D Control Register 2).

El registro ADCON0 controla la operación del módulo A/D. El registro ADCON1 configura las funciones de los pines del puerto. El registro ADCON2 configura la fuente de reloj, el tiempo de adquisición de la señal y el formato del resultado.

En la figura 1.9 se muestra la función de transferencia que permite determinar el número resultante de la conversión en función del voltaje analógico de entrada ($1LSb=5/1024=4.88$ [mV]).

Por ejemplo, la salida será 0x002h para un intervalo del voltaje de entrada entre 9.76 mV (2LSb) y 14.64 mV (3LSb).

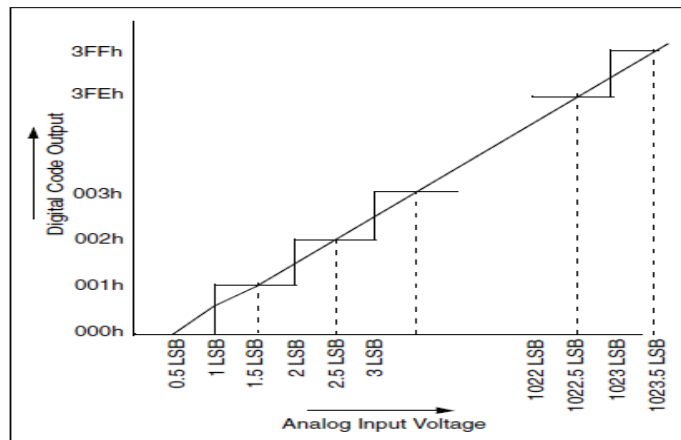


Figura 1.9 Función de transferencia del convertor A/D. [7]

1.5.5. TIMER 1

El módulo temporizador/contador TIMER 1 incorpora las siguientes características:

- Temporizador/contador de 16bits seleccionable por software.
- Registros legibles y escribibles de 8 bits (TMR1H y TMR1L)
- Fuente seleccionable de reloj (interna o externa) con opciones del reloj o del oscilador interno Timer1 del dispositivo
- Interrupción por desbordamiento
- El módulo reset en un evento especial de disparo del CCP
- Flag de estado del reloj del dispositivo (T1RUN)

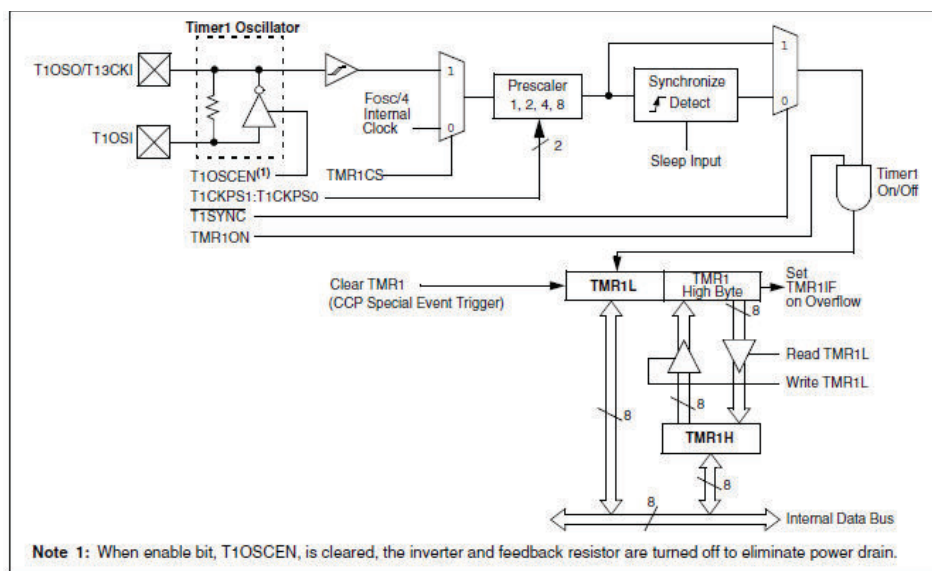


Figura 1.10 Diagrama del TIMER1 (Modo de lectura/escritura 16 bits). [7]

1.5.5.1. INTERRUPCIÓN DEL TIMER1

El par de registros TMR1 (TMR1H:TMR1L) incrementan de 0000h a FFFFh y vuelven a 0000h. La interrupción Timer1, si está permitida, se genera en el desbordamiento que activa el flag de la interrupción, TMR1IF (PIR1<0>). Esta interrupción se puede permitir o inhabilitar activando o borrando el bit, TMR1IE (PIE1<0>).

1.5.6. PANTALLA GRÁFICA DE CRISTAL LÍQUIDO GLCD

Una pantalla gráfica de cristal líquido o GLCD (acrónimo del inglés Graphic Liquid Crystal Display) es una pantalla plana formada por una matriz de píxeles monocromos colocados delante de una fuente de luz o reflectora (ver Figura 1.11). A menudo, se utiliza en dispositivos electrónicos de baterías, ya que utiliza cantidades muy pequeñas de energía eléctrica.

Dispone de una memoria RAM interna del mismo tamaño de la capacidad que dispone la pantalla; por ejemplo, si una pantalla tiene un tamaño de 128 píxeles de largo por 64 píxeles de alto (128x64) tiene una memoria RAM interna de la misma capacidad (128x64).

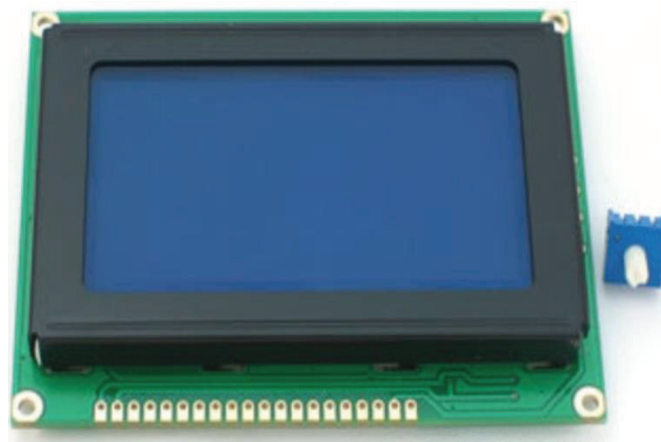


Figura 1.11 Pantalla gráfica de cristal líquido GLCD 128X64. [8]

Por lo general, es manejada por microcontroladores de 8 bits para la configuración y utilización de la misma.

En la tabla 1.6 se describe la función de cada uno de los pines del GLCD 128X64 píxeles utilizada en el presente proyecto.

PIN	Función
PIN 1: VSS	Conexión a tierra (GND).
PIN 2: VDD	Conexión de alimentación (+ 5V).
PIN 3: V0	Voltaje de salida negativa, por lo general es usada con un potenciómetro con el PIN 18, para el ajuste del contraste de los pixeles.
PIN 4: RS	Datos de E/S para el cambio de registro
PIN 5: R/W	Determina si los datos se van a leer o escribir.
PIN 6: E	Habilita la comunicación con la GLCD.
PIN 7: D0	Especifica un dato de 8-bits de información, bus de datos 1.
PIN 8: D1	Bus de datos 2.
PIN 9: D2	Bus de datos 3.
PIN 10: D3	Bus de datos 4.
PIN 11: D4	Bus de datos 5.
PIN 12: D5	Bus de datos 6.
PIN 13: D6	Bus de datos 7.
PIN 14: D7	Bus de datos 8.
PIN 15: CS1	Indica si se selecciona la primera mitad de la pantalla, pixeles 0-63.
PIN 16: CS2	Indica si se selecciona la segunda mitad de la pantalla, pixeles 64-127.
PIN 17: RST	Señal de reinicio.
PIN 18: VOUT	Conexión de ajuste de contraste de los pixeles.
PIN 19: LED+	Conexión positiva de la luz de fondo, por lo general son +5V
PIN 20: LED-	Conexión negativa de la luz de fondo, por lo general es tierra

Tabla 1.6 Descripción de los pines del módulo GLCD 128x64.

1.6. MIKRO C PRO FOR PIC [8][9]

Desarrollado por Mikroelektronika, es un compilador compatible con el estándar ANSI C; no obstante, en determinados aspectos, difiere del estándar ANSI en algunas características. Algunas de estas diferencias se refieren a las mejoras, destinadas a facilitar la programación de los microcontroladores PIC, mientras que las demás son la consecuencia de la limitación de la arquitectura del hardware de los PIC, desarrollado para trabajar con microcontroladores de 8 bits de Microchip® (PIC12XXX – PIC18XXX). Además, es una herramienta de desarrollo de gran alcance, rica en características de los microcontroladores PIC. Está diseñado para proporcionar al programador la solución más fácil posible para el desarrollo de aplicaciones para sistemas empotrados, sin comprometer el rendimiento o control.

1.6.1. CARACTERÍSTICAS PRINCIPALES DE MIKRO C PRO FOR PIC

Al iniciar el IDE del compilador mikroC PRO for PIC por primera vez, aparecerá una ventana como se muestra en la figura 1.12

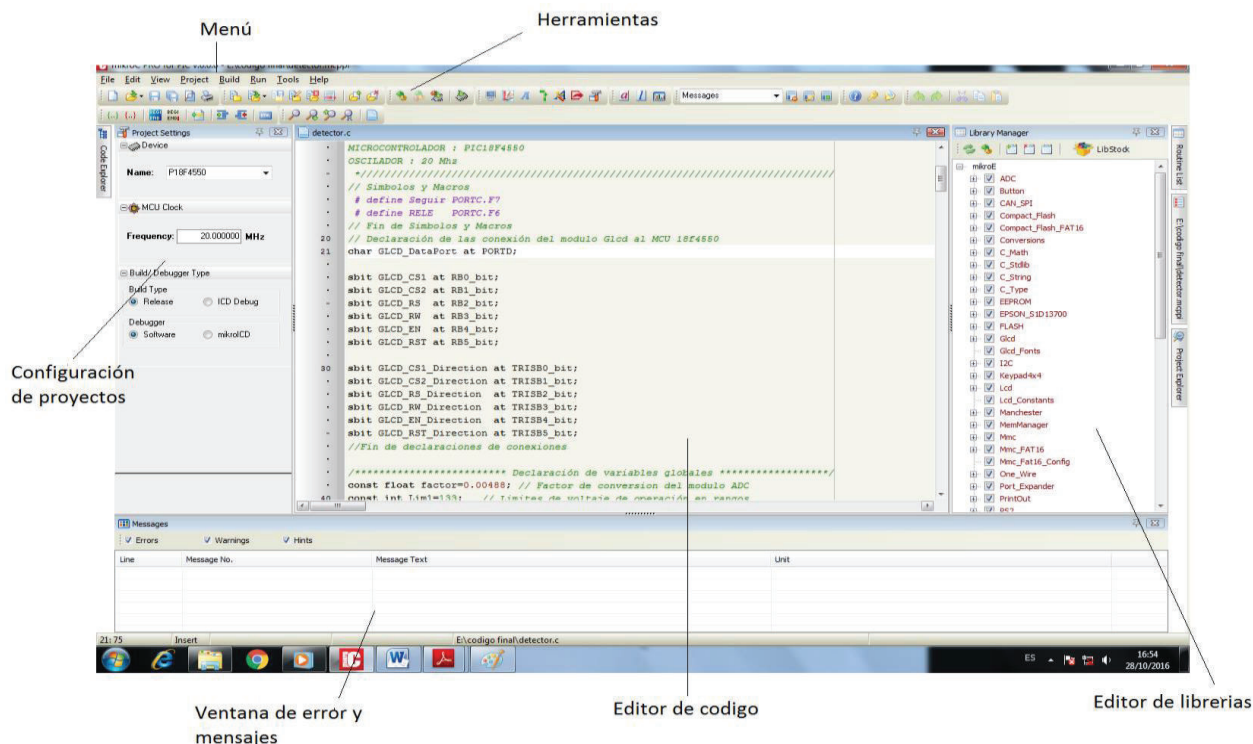


Figura 1.12 Entorno de desarrollo en Mikro C for PIC.

1.6.2. PROJECT SETTINGS (CONFIGURACIÓN DE PROYECTOS)

Al compilar un proyecto, el compilador genera el archivo .hex que se cargará en el microcontrolador. Estos archivos serán diferentes, lo que depende del tipo del microcontrolador, así como del propósito de la compilación. Por esta razón es necesario ajustar algunos parámetros de proyectos utilizando la ventana Project Settings.

1.6.3. LIBRARY MANAGING (GESTIÓN DE LIBRERÍAS)

El compilador tiene que conocer todas las dependencias de su archivo fuente en mikroC para compilarlo apropiadamente. Por ejemplo, si las librerías forman parte de su proyecto, debe especificar cuáles de ellas se utilizan. Las librerías contienen un gran número de funciones listas para ser utilizadas. Las librerías en mikroC proporcionan muchas facilidades para escribir programas para los microcontroladores PIC. Se abre la ventana Library Manager, y se marca las que se desea utilizar en el programa. Al marcar una librería, se añade automáticamente al proyecto y se enlaza durante el proceso de la compilación. Así, no se necesita incluir las librerías manualmente en los archivos del código fuente por medio de la directiva del preprocesador #include.

1.6.4. CODE EDITOR (EDITOR DE CÓDIGO)

El proceso de editar programas se debe realizar dentro de la ventana principal del IDE denominada Code Editor. Al escribir el programa, se sugiere incluir comentarios. Los comentarios son muy importantes para depurar y mejorar el programa. Además, aunque el compilador no tenga las restricciones de formateo, siempre debe seguir a las mismas reglas de editar. Dado que no hay limitaciones de tamaño, no se debe prescindir de utilizar los espacios en blanco para hacer códigos más legibles.

1.6.5. COMPILAR Y SOLUCIONAR LOS ERRORES

Para compilar el código, se pulsa sobre la opción *Build* en el menú *Project*. En realidad, el proyecto entero se compila, y si la compilación se ha realizado con éxito, se generarán los archivos de salida (*asm. hex* etc.). Una compilación se realiza con éxito si no se ha encontrado ningún error. Durante el proceso de compilación se generan muchos mensajes que se visualizan en la ventana *Messages*. Estos mensajes consisten en información, advertencia y errores (ver Figura 1.13). Cada error encontrado se asocia con su línea de programa y su descripción.

Line	Message No.	Message Text	Unit
0	126	All files Preprocessed in 188 ms	
0	122	Compilation Started	detector.c
332	123	Compiled Successfully	detector.c
318	1164	Variable 'resultado' has been eliminated by optimizer	detector.c
0	127	All files Compiled in 297 ms	
0	1144	Used RAM (bytes): 210 (10%) Free RAM (bytes): 1817 (90%)	Used RAM (bytes): 210 (10%) Free RAM (bytes): 1817 (90%)
0	1144	Used ROM (bytes): 9892 (30%) Free ROM (bytes): 22876 (70%)	Used ROM (bytes): 9892 (30%) Free ROM (bytes): 22876 (70%)
0	125	Project Linked Successfully	detector.mcppl
0	128	Linked in 1046 ms	
0	129	Project 'detector.mcppl' completed: 1890 ms	
0	103	Finished successfully: 28 oct 2016, 17:16:00	detector.mcppl

Figura 1.13 Ventana de mensajes después de compilar.

Como un error en el código puede generar muchos más errores, simplemente se debe intentar solucionar el primer error en la lista y después recompilar el programa. En otras palabras, es recomendable solucionar los errores uno a uno.

CAPÍTULO 2

CONSTRUCCIÓN DEL DISPOSITIVO DE PROTECCIÓN

2.1. ANÁLISIS DE FALLAS Y SUS VALORES PERMISIBLES

FALLA	CAUSAS	EFFECTOS	TOLERANCIAS
Desbalance De Voltaje	<p>Este fenómeno usualmente puede ocurrir por la asimetría de las impedancias conectadas en las líneas de distribución y/o transmisión, distribución desigual de las cargas monofásicas en un sistema trifásico; tales como motores de inducción, convertidores de potencia, hornos eléctricos, iluminación, etc. Además, en sistemas residenciales y/o comerciales por ser la mayoría de consumidores de energía eléctrica, es donde se presenta el mayor porcentaje de desbalance de voltaje, debido a la variación de las cargas y cambios continuos en la configuración de las redes eléctricas</p>	<p>Cuando los voltajes de línea que alimentan al motor trifásico no son balanceados, se originará un desbalance en la corriente del bobinado estatórico, encontrando valores entre 6 a 10 veces el porcentaje de desbalance de voltaje en un motor con carga máxima. El efecto de los voltajes desbalanceados sobre el motor de inducción es equivalente a introducir un "voltaje de secuencia negativa" que "gira" en sentido opuesto al que giraría si el voltaje fuera balanceado. Este voltaje de secuencia negativa produce un flujo giratorio en el entrehierro en sentido contrario a la rotación del rotor, induciendo altas corrientes en el rotor. El torque de arranque y el torque máximo disminuyen. La velocidad del motor a plena carga disminuye levemente. Otro de los efectos más perjudiciales es que la vida del aislante de los devanados disminuye aproximadamente a la mitad por cada 10º C de incremento en la temperatura de los devanados.</p>	<p>Según la Norma NEMA MG1.2009 y la IEC60034-2, se establece que, el desbalance de voltaje no debe ser mayor al 3% en el sistema eléctrico que energiza al motor trifásico.</p>
Pérdida De Fase	<p>La pérdida de una fase representa uno de los peores casos de desbalance de voltaje, causada por ejemplo por un fusible que se funde, malas instalaciones, contactores deteriorados, etc.</p>	<p>Los efectos potenciales a los que puede conducir la pérdida de fase son similares a aquellos producidos por el desbalance de voltaje; adicionalmente, también en los devanados de las dos fases restantes que quedan en operación se produce:</p> <ul style="list-style-type: none"> - Calentamiento excesivo. - Calentamiento en el rotor 	<p>Los Relés electrónicos de pérdida de fase se disparan en 2 a 3 segundos en un motor totalmente cargado</p>

FALLA	CAUSAS	EFFECTOS	TOLERANCIAS
Inversión de Secuencia de Fases	<p>Este fallo se produce por descuido del personal técnico, durante el mantenimiento o cambio de máquinas y motores al conectarlos a la red eléctrica.</p> <p>Otros aspectos que pueden influir son la falta de señalización, tanto en la fuente de alimentación trifásica como en los bornes del motor.</p>	La inversión en la secuencia de fases ocasiona que los motores trifásicos giren en dirección opuesta, lo cual puede causar lesiones al personal de operación y desperfectos al propio equipo.	La inversión en la secuencia de fases es un fallo que se corrige antes de poner en marcha al motor trifásico.

2.2. DIAGRAMA DE BLOQUES DEL DISPOSITIVO DE PROTECCIÓN

El dispositivo de protección se puede dividir en cuatro bloques principales: fuente de alimentación DC, acondicionamiento de señales, unidad de procesamiento y control (microcontrolador PIC), y la unidad de potencia (contactor y módulo relé). En la figura 2.1 se observa un esquema con sus componentes más relevantes, los cuales se detallará más adelante.

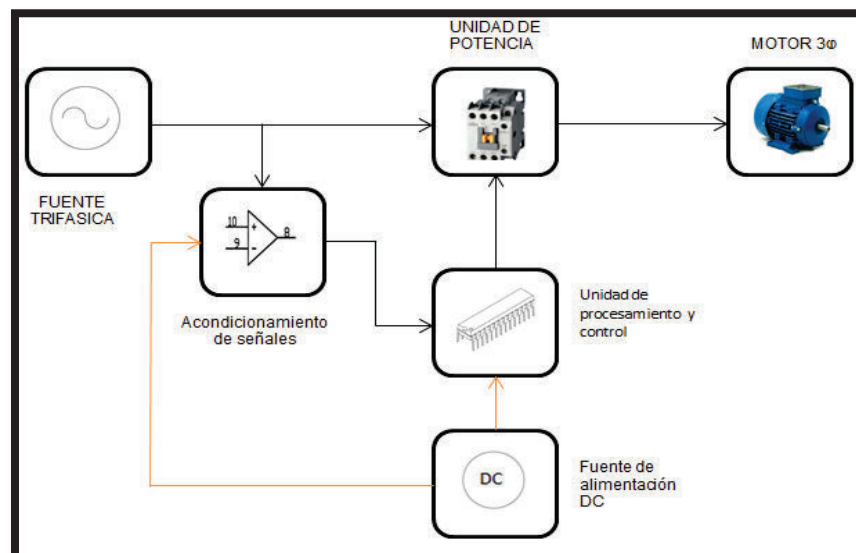


Figura 2.1 Diagrama de bloques del dispositivo de protección.
[Elaboración propia]

A su vez, el bloque de acondicionamiento de señales para el microcontrolador se divide en dos subgrupos; el primero para acondicionar las señales para la medición de voltajes, y el segundo para la identificación de la secuencia de fases.

2.3. FUENTE DE ALIMENTACIÓN DC (CORRIENTE DIRECTA)

El dispositivo de protección utiliza fuentes de DC de +5 V, +12 V, y -12 V para polarizar los diferentes circuitos integrados que se detalla en la siguiente tabla.

Fuente	Carga
+5 V	Microcontrolador, LCD y circuitos integrados.
±12 V	Amplificadores operacionales, circuitos integrados y relé.

Tabla 2.1 Cargas con sus respectivas fuentes.

En la figura 2.2 se muestra un diagrama del circuito implementado para la fuente de voltaje DC.

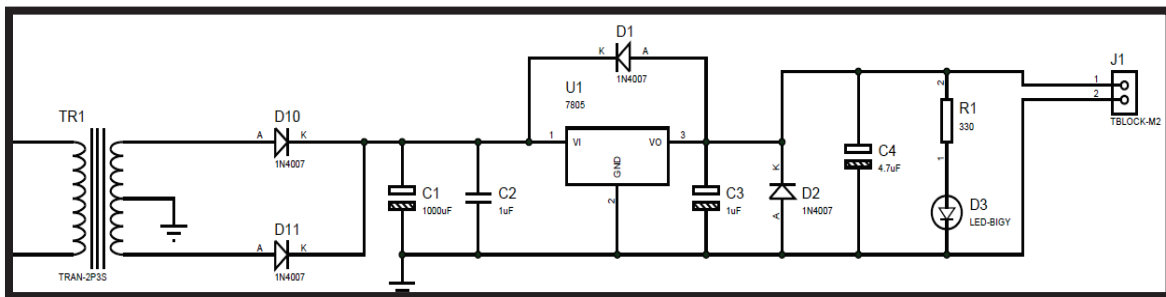


Figura 2.2 Circuito de la fuente de voltaje DC +5 V.
[Elaboración propia]

El transformador, el primer elemento que aparece en el diagrama de la fuente, tiene como objetivo reducir el nivel de voltaje de 120 V AC (Corriente alterna) proveniente de la red eléctrica, a un valor de voltaje adecuado a la aplicación. Para ello se utilizó un transformador de TAP central de relación 10:1 para las 3 fuentes de voltaje DC.

En la etapa de rectificación, como la meta es obtener un valor de voltaje continuo, mediante la rectificación, se convierte la señal de voltaje alterno en una señal de voltaje pulsatoria positiva. Para ello, se dispuso de 2 diodos, como se observa en el circuito de la figura 2.2; los diodos tienen la característica especial de dejar circular la corriente en un único sentido (de ánodo hacia cátodo).

A la salida del rectificador, se dispone de capacitores para el filtrado de la señal pulsatoria. Mediante el capacitor C1 (normalmente de alta capacitancia) se obtiene un aplanamiento de la señal con un determinado valor de rizado, que aún no es adecuado para la polarización de los circuitos integrados, el microcontrolador y otros elementos.

Para estabilizar el voltaje en los valores deseados se utilizó los reguladores de voltaje de la familia LM 78/79XX. Para +5 V se usó el regulador LM 7805, +12 V el regulador LM 7812 y finalmente para -12 V el regulador LM 7912.

Los valores de los capacitores C2, C3, C4 son recomendados por el fabricante. El diodo D1 se utiliza como protección del regulador contra polarización invertida, mientras que D2 se utiliza como protección contra polarización inversa (cuando existe una caída de voltaje en la entrada, más rápida que en la salida). [11]

Por último, se utilizó LED's para señalar el correcto funcionamiento de cada una de las fuentes, figura 2.3

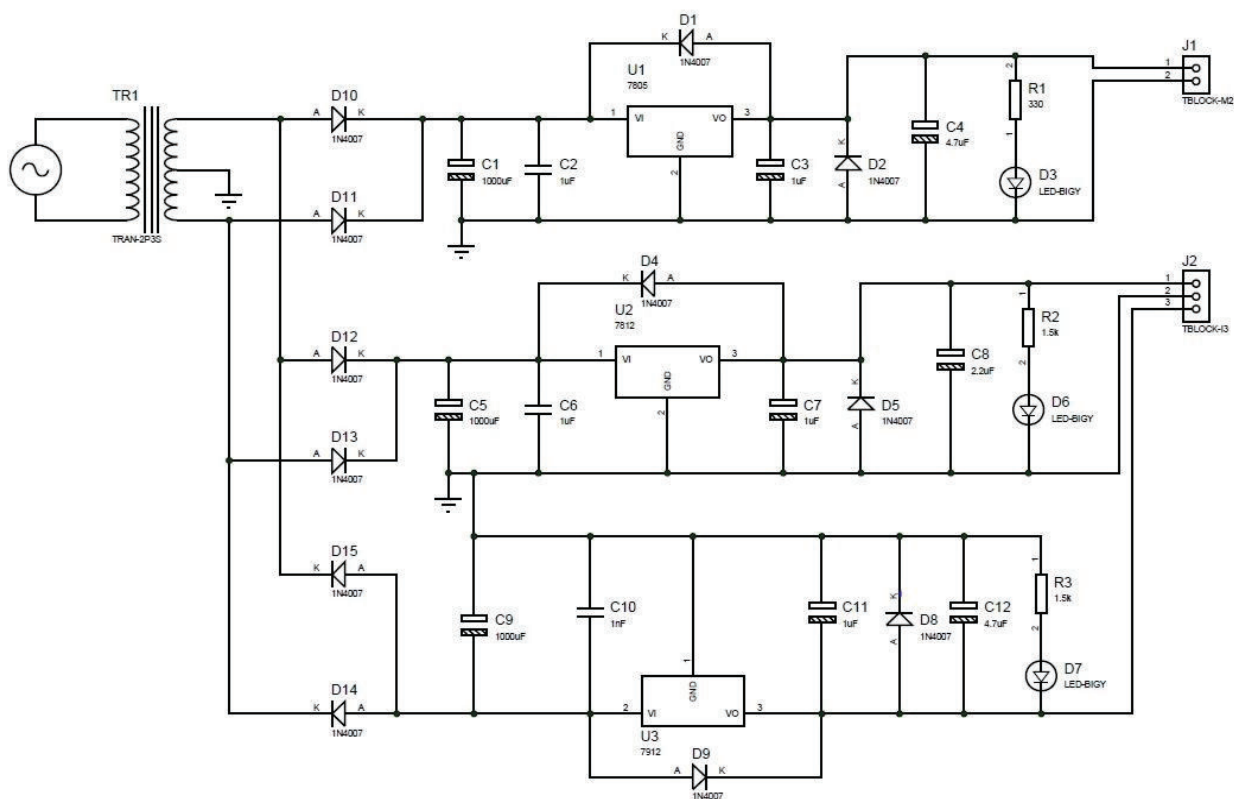


Figura 2.3 Circuitos de las fuentes de voltajes DC.
[Elaboración propia]

2.4. DESARROLLO DEL CIRCUITO MONITOR DE VOLTAJE

2.4.1. ACONDICIONAMIENTO PARA LECTURA DE LA SEÑAL DE VOLTAJE ALTERNO

Para medir el voltaje AC presente en las 3 fases del motor (fase-neutro), se utilizaron varios recursos; entre ellos, el más relevante el amplificador operacional en varias

configuraciones. En la figura 2.4, se observa la distribución de pines del amplificador operacional uA741 empleado en este proyecto (ver anexo A).

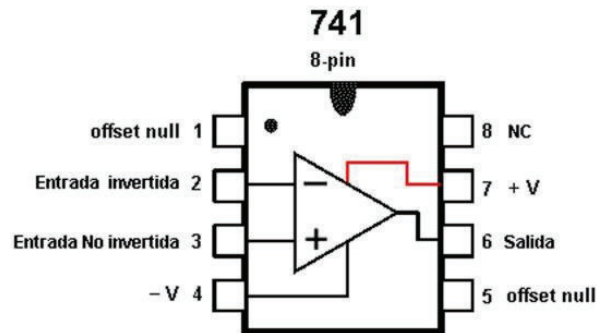


Figura 2.4 Amplificador operacional uA741. [4]

Debido a que el voltaje a medir es de AC (corriente alterna) y el microcontrolador de *Microchip PIC* admite en sus canales analógicos voltajes comprendidos entre 0 a 5 V DC, es necesario disminuir de forma proporcional el voltaje para luego rectificarlo y así convertirlo a DC. Por lo tanto, la primera etapa para el acondicionamiento de la señal de medición del voltaje está conformada por un transformador de potencial de relación 10:1 y un divisor de voltaje.

La división de voltaje ocurre cuando una fuente dependiente o independiente de voltaje se conecta en serie con dos resistencias, como se ilustra en la figura 2.5. Además, se toma del secundario del mismo transformador señales para procesarlas en el módulo de acondicionamiento de señales de la identificación de la secuencia de fases que se describe más adelante.

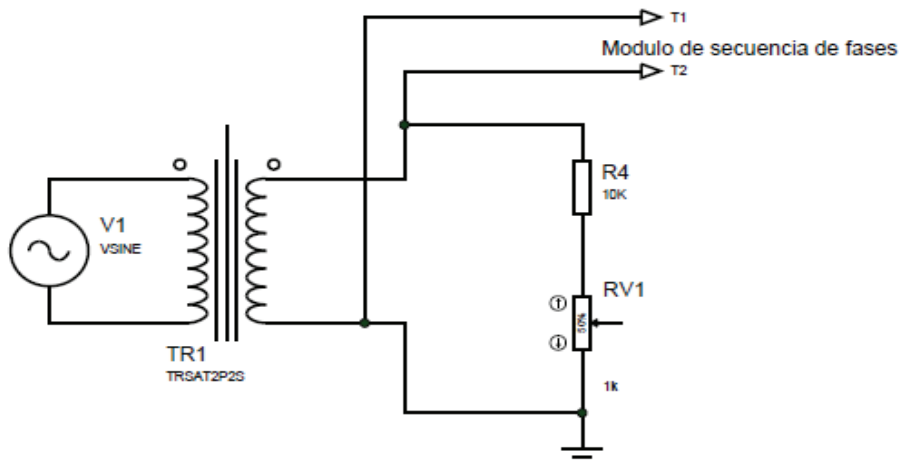


Figura 2.5 Transformador de potencial y divisor de voltaje.
[Elaboración propia]

El valor de voltaje en el lado secundario del transformador está dado por la ecuación 2.1

$$\frac{V_2}{V_1} = \frac{N_2}{N_1} \quad (2.1)$$

Donde:

V_2 : Voltaje en el lado secundario del transformador.

V_1 : Voltaje en el lado primario del transformador.

$\frac{N_2}{N_1} = \frac{1}{10}$: Relación de transformación.

Despejando la ecuación 2.1 se obtiene el valor de voltaje en el lado secundario

$$V_2 = \frac{N_2}{N_1} * V_1 \quad (2.2)$$

Para realizar las mediciones de voltaje, se consideró el valor nominal obtenido mediante la lectura de un voltímetro digital, el cual fue de 120 V entre una fase y el neutro de una red trifásica industrial; sin embargo, se tiene que prever que este valor de voltaje puede exceder en condiciones de fallo (sobre voltaje). Por ende, para el dimensionamiento del circuito se escogió un límite superior de 150 V_{RMS} (valor eficaz).

En base a las características de voltaje del canal analógico del microcontrolador se seleccionó el valor de la resistencia $R_1=10K\Omega$ y el valor de voltaje en la resistencia variable RV_1 de 1 V (RMS), para el límite superior de 150 V.

Así tenemos, para la resistencia variable RV_1

$$V_{RV_1} = V_2 * \frac{RV_1}{RV_1+R_1} \quad (2.3)$$

Reemplazando la ecuación (2.2) en (2.3):

$$V_{RV_1} = V_1 * \frac{N_2}{N_1} * \frac{RV_1}{RV_1+R_1} \quad (2.4)$$

$$1 V = 150 V * \frac{1}{10} * \frac{RV_1}{RV_1+10 K\Omega}$$

$$RV_1 = 714.28\Omega$$

El valor de la resistencia de precisión que nos permita obtener dicho valor con una buena exactitud y disponible en el mercado es de 1 K Ω .

Se justifica estos valores de voltaje y resistencia debido a que nuestro acondicionador de señal es un rectificador de precisión de onda completa o también denominado circuito de valor absoluto, y sus rangos de operación idóneos son de 0 a 1.5 V_{RMS} AC.

2.4.2. RECTIFICADORES DE PRECISIÓN CON AMPLIFICADORES OPERACIONALES

En concordancia con las especificaciones fijadas anteriormente, el bloque de acondicionamiento de la señal de voltaje AC, debe convertir la señal de voltaje de 0 a 150 V AC (rms) a un voltaje entre 0 a 5 V DC. La primera etapa se encarga de disminuir el voltaje proporcionalmente y adecuarlo, para ello se utilizó el transformador y el divisor de voltaje.

Es de conocimiento que para rectificar una señal AC se emplea diodos y filtros; sin embargo, el problema es que los diodos tienen un voltaje umbral, para diodos de silicio de 0.6 V y para diodos de germanio de 0.3 V; esto ocasiona que no se puedan rectificar voltajes pequeños. La solución es diseñar un circuito que se comporte como diodo ideal, se utiliza amplificadores operacionales y dos diodos comunes, así se obtiene un poderoso circuito capaz de rectificar señales en la entrada, incluso de unos cuantos mV.

2.4.2.1. RECTIFICADOR INVERSOR DE MEDIA ONDA LINEAL CON SALIDA NEGATIVA

Los rectificadores de media onda transmiten solamente la mitad de un ciclo de una señal y eliminan la otra al limitar su salida a 0 V. La mitad del ciclo que sí se transmite puede estar invertida o no, también puede experimentar una ganancia o una atenuación, o permanecer inalterada en cuanto a la magnitud; todo lo cual dependerá de las resistencias elegidas y de la colocación de los diodos en el circuito del amplificador operacional.

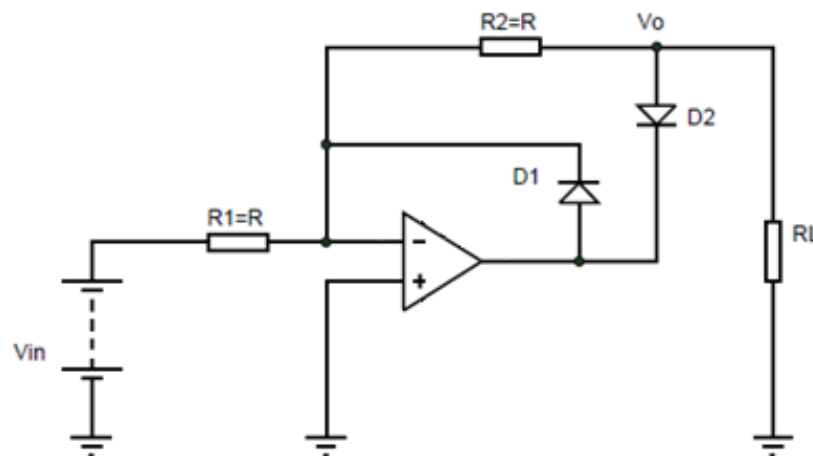


Figura 2.6 Rectificador inversor de media onda lineal con salida negativa. [4]

El análisis del funcionamiento del circuito de la figura 2.6 es el siguiente:

a) Cuando V_{in} (voltaje de entrada) > 0 (semiciclo positivo)

Los diodos $D1$ y $D2$ inicialmente son circuitos abiertos (estado OFF), el amplificador operacional (*amp op*) no tiene retroalimentación negativa, por lo cual este elemento opera en lazo abierto y se satura con valores de $-V_{cc}$ (voltaje DC de polarización negativa) en su salida. Sin embargo, en el momento que el valor de voltaje de la salida del *amp op* (V_o') sea mayor o igual al valor de voltaje umbral del diodo $D2$ ($V_{th}=0.6$ V), este último elemento se polariza directamente, cierra el circuito de la retroalimentación negativa y conecta al amplificador operacional como inversor de ganancia unitaria dado que $R2=R1$.

Puesto que la entrada inversora (e^-) está al potencial de tierra (tierra virtual), el diodo $D1$ está en polarización inversa (estado OFF) figura 2.7.

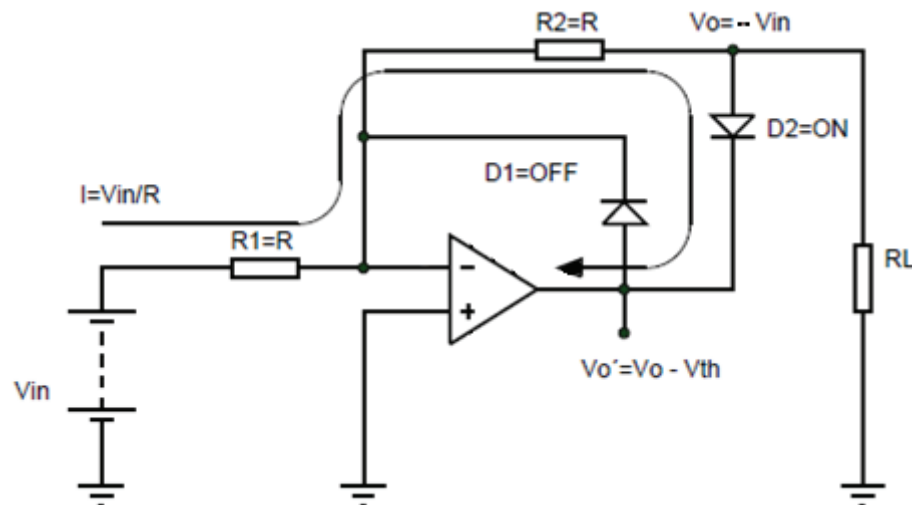


Figura 2.7 La salida V_o es negativa y de igual magnitud de V_{in} para el semiciclo positivo. [4]

Para el amplificador inversor:

$$V_o = -\frac{R_2}{R_1} V_{in}$$

Donde $R_2=R_1$

$$V_o = -V_{in} \quad (2.5)$$

Para la salida V_o' se establece lo siguiente:

$$V_o' = V_o - V_{th} \quad (2.6)$$

Reemplazado 2.5 en 2.6:

$$V_{o'} = -V_{in} - V_{th}$$

$$V_{o'} = -(V_{in} + V_{th})$$

b) Cuando $V_{in} < 0$ (semiciclo negativo)

En la figura 2.8, de igual forma los diodos D1 y D2 no conducen (OFF), la salida del *amp op* $V_{o'}$ se satura con valores de $+V_{cc}$ (voltaje DC de polarización positivo). Esto produce que el diodo D1 conduzca (ON) y fuerza al diodo D2 a estar en polarización inversa (OFF). Producto de ello, no circula corriente por R2 y el voltaje de salida del circuito V_o se iguala a cero porque la entrada inversora (e-) está al potencial de tierra (tierra virtual)

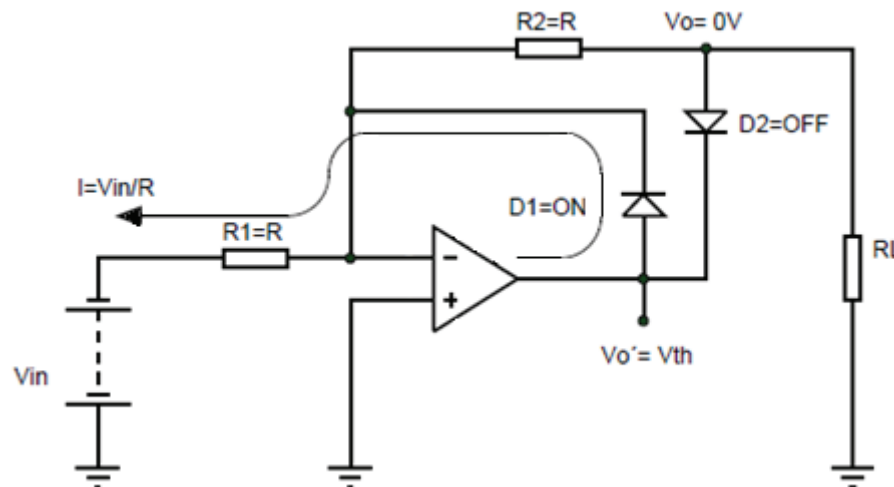


Figura 2.8 La salida V_o limitada a 0 V para el semiciclo negativo. [4]

El funcionamiento del rectificador de precisión de media onda se resume por las formas de ondas de la figura 2.9.

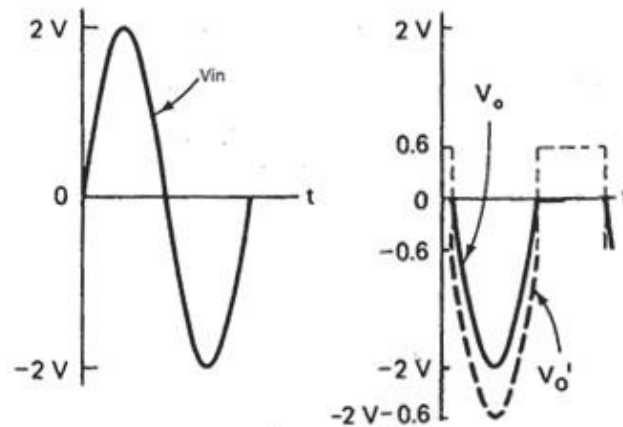


Figura 2.9 Formas de onda del circuito rectificador de precisión de media onda. [4]

2.4.2.2. RECTIFICADOR DE PRECISIÓN O AMPLIFICADOR DE VALOR ABSOLUTO CON ENTRADAS CONECTADAS A TIERRA [4]

Para construir un convertidor de voltaje AC a voltaje DC, se inicia con el rectificador de precisión o amplificador de valor absoluto de la figura 2.10. Para las entradas positivas de la figura 2.10(a), el amplificador operacional A invierte V_{in} . El amplificador B suma las salidas de A y V_{in} para producir una salida del circuito $V_o = +V_{in}$.

Para entradas negativas como se muestra en la figura 2.10(b), el amplificador operacional B invierte $-V_{in}$, mientras que el amplificador operacional A limita su salida a 0 voltios, de esa forma la salida del circuito $V_o = +V_{in}$. Por lo tanto, la salida del circuito V_o es positiva e igual al valor rectificado o absoluto de la entrada figura 2.11.

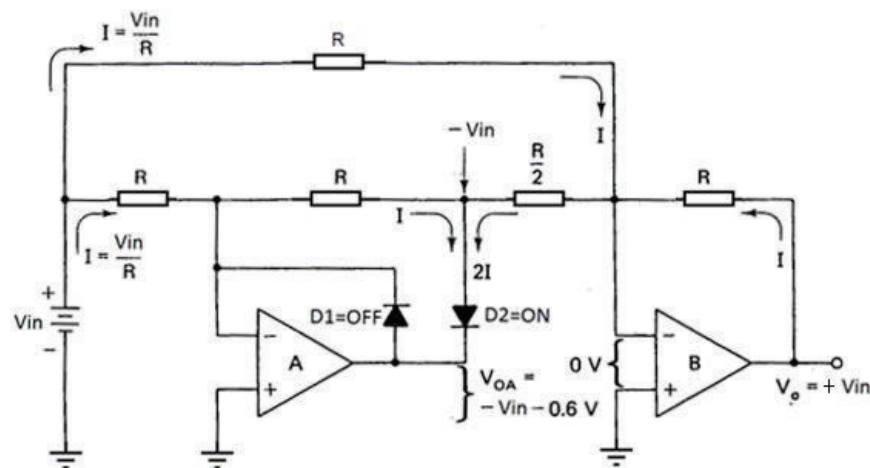


Figura 2.10a Amplificador de valor absoluto semiciclo positivo. [4]

(a) Para entradas positivas el amplificador operacional A invierte V_{in} ; el amplificador operacional B es un inversor sumador, de modo que $V_o = +V_{in}$ [4]

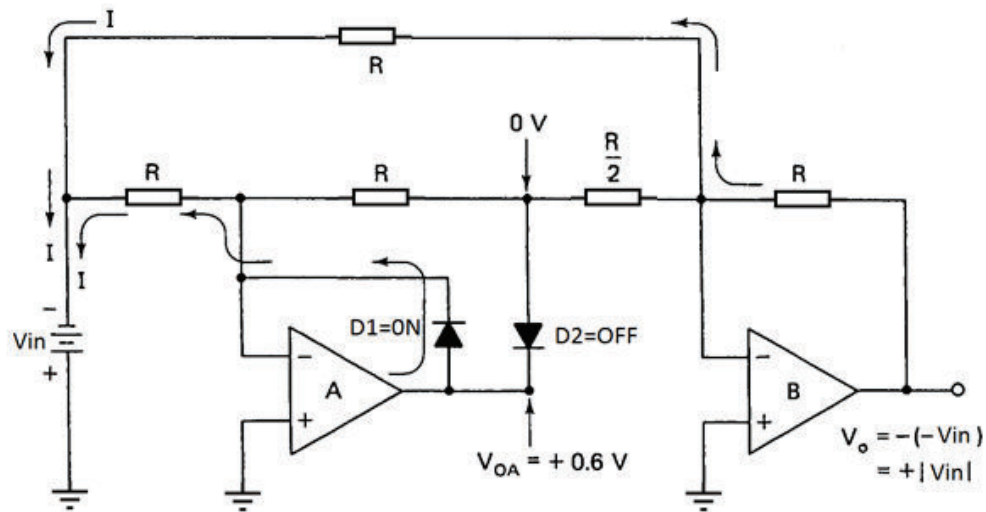


Figura 2.10b Amplificador de valor absoluto semiciclo negativo. [4]

(b) Para entradas negativas, la salida de A se limita 0 voltios; el amplificador operacional B invierte V_{in} , de modo que $V_o = -(-V_{in}) = +|V_{in}|$

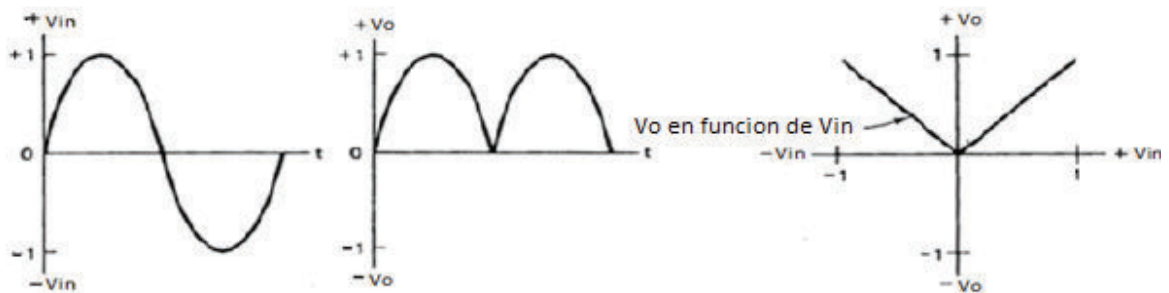


Figura 2.11 Rectificador de onda completa de precisión, formas de onda. [4]

2.4.3. CONVERTIDOR DE AC A DC [4]

Al circuito de valor absoluto en la figura 2.10 se añade un capacitor de valor alto (10 μF). El circuito resultante es el amplificador de valor medio absoluto (MAV) o convertidor de AC a DC que se muestra en la figura 2.12. El capacitor C proporciona el promedio de la salida rectificadora del amplificador operacional B, el cual toma cerca de 60 ciclos del voltaje de entrada antes de que el voltaje del capacitor se establezca a su lectura final.

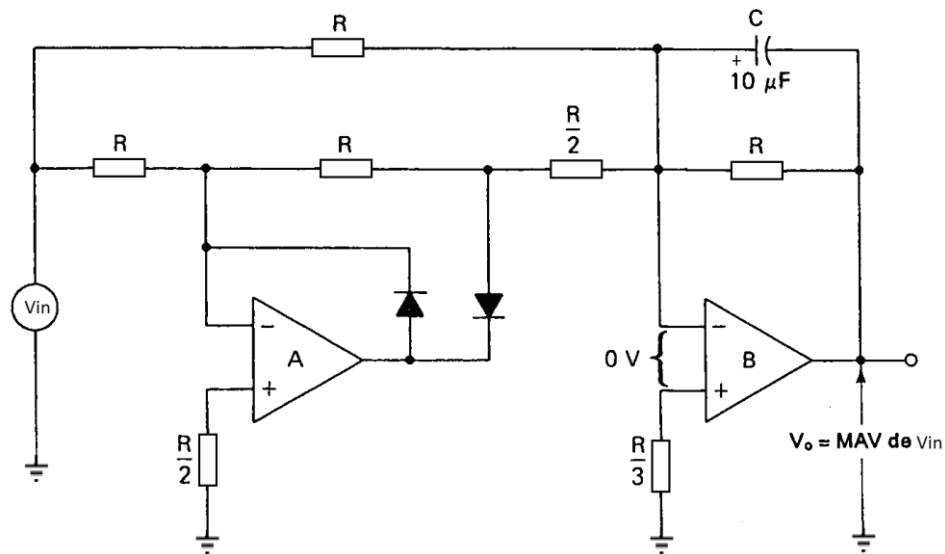


Figura 2.12 Convertidor de CA a CD o amplificador de valor absoluto medio (MAV). [4]

2.4.4. CIRCUITO ELÉCTRICO PARA EL ACONDICIONAMIENTO DE LA SEÑAL DE VOLTAJE AC

Para acondicionar la señal de voltaje AC no solo es necesario usar el convertidor de CA a CD, hace falta usar circuitos adicionales como: seguidores de voltaje y amplificador no inversor como se observa en la figura 2.13.

El voltaje de entrada al amplificador A de la figura 2.13, es proporcionado por el bloque del transformador y del divisor de voltaje. Dicha señal entra a un seguidor de voltaje (amplificador A), el cual evita una caída de voltaje de la señal AC proporcionada por el bloque anterior. Seguidamente, entra al circuito MAV (amplificadores B y C) para rectificarla y promediarla a un valor máximo de 1 V DC cuando en la entrada de todo el circuito de la figura 2.13 sea 150 V_{RMS} AC. Una vez convertida a una señal DC, se amplifica por medio de un amplificador no inversor (amplificador D) con ganancia de 5 con el fin de obtener a la salida de todo el circuito 5 V DC, el cual es el valor máximo que se puede aplicar a las entradas del conversor análogo a digital del microcontrolador PIC de *Microchip*.

Finalmente, la señal de salida del amplificador D entra a un seguidor de voltaje (amplificador E) para evitar una vez más cualquier caída de voltaje a la salida. Además, el diodo Zener (D3-1N4733A) proporciona protección para el microcontrolador, con un valor de voltaje de polarización inversa de 5.1 V es ideal para recortar valor superiores que podrían generar daños al chip.

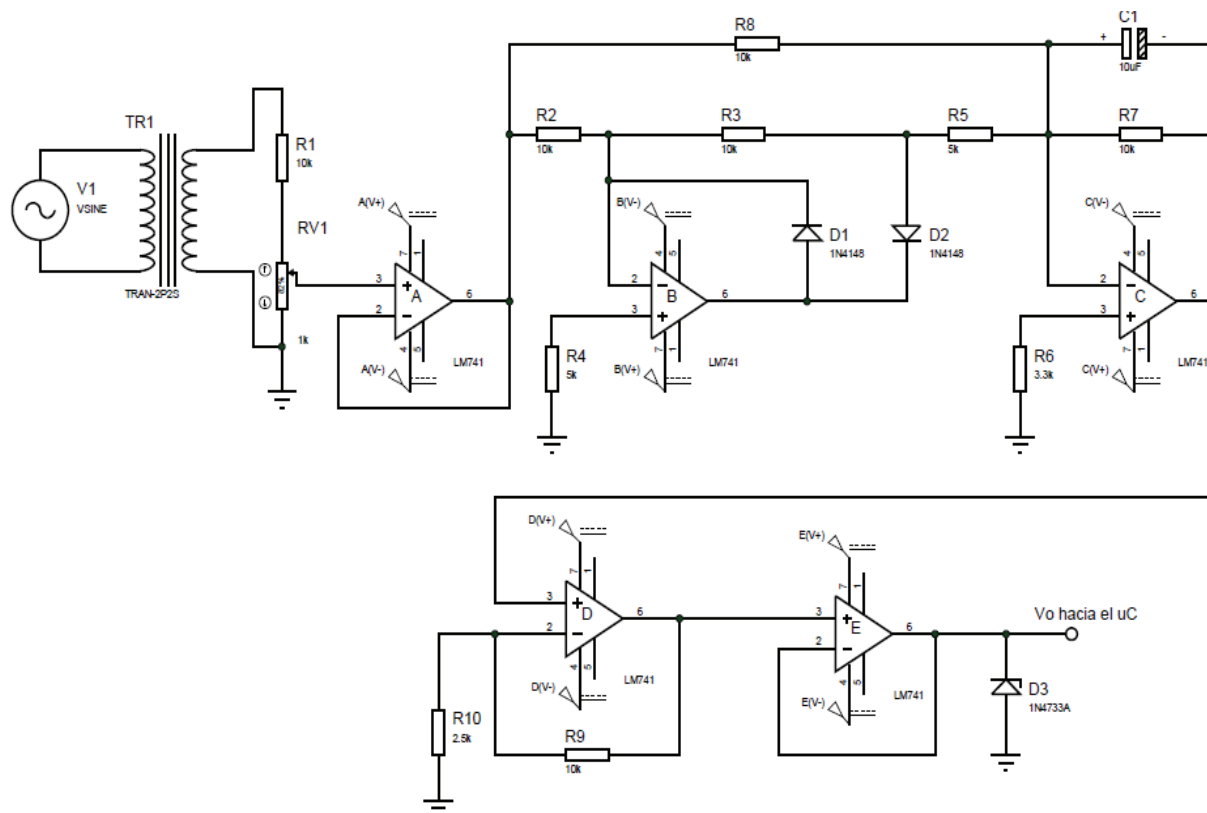


Figura 2.13 Diagrama eléctrico del circuito de acondicionamiento de señal AC.
[Elaboración propia]

2.5. CIRCUITO ELÉCTRICO PARA EL ACONDICIONAMIENTO DE LAS SEÑALES PARA IDENTIFICACIÓN DE SECUENCIA DE FASES.

Para la identificación de la secuencia de fases del sistema trifásico se optó por usar amplificadores operacionales, específicamente el uso del comparador Lm339 por sus características mejoradas frente a otros comparadores como: tiempo de respuesta, voltaje offset, voltaje de alimentación simple o dual, ganancia de voltaje (V/mV), etc (ver anexo B).

En el módulo de acondicionamiento de señal de voltaje AC, el primer bloque está conformado por un transformador de relación 10:1, el cual reduce el voltaje a valores aptos para la aplicación. En esta sección también se hizo uso de dicho transformador, para esto se ha tomado la señal del secundario de cada uno de los tres transformadores para cada una de las fases (R, S, T).

De igual forma, la señal obtenida del secundario del transformador, terminal T1 y terminal T2, se la conecta a otro divisor de voltaje para adecuar el nivel de voltaje y

limitar la corriente hacia el comparador (protección del elemento) como se observa en la figura 2.14.

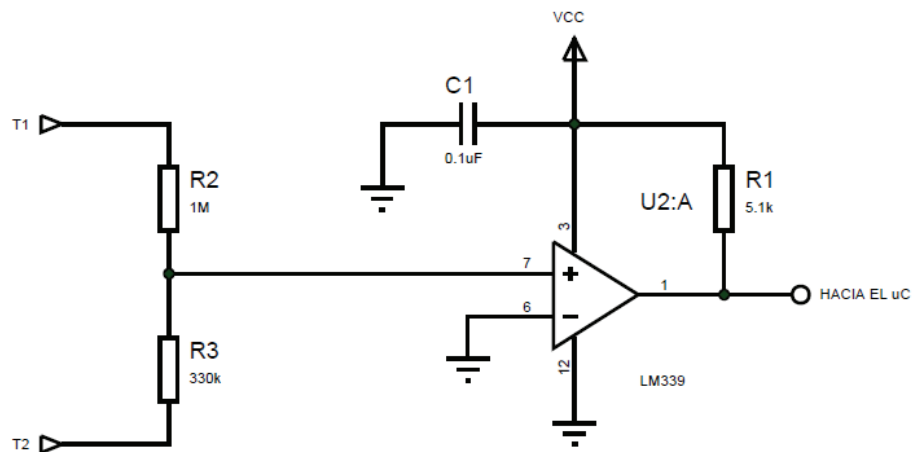


Figura 2.14 Circuito detector de cruce por cero.
[Elaboración propia]

El circuito de la figura 2.14 se encarga de generar una onda cuadrada entre 0 y 5V DC sincronizado con los cruces por cero. El circuito conmuta de 0 a 5 V (flanco ascendente) cuando la señal de voltaje alterna cruza por cero hacia los valores positivos de dicha señal AC. Por otro lado, el circuito conmuta de 5 a 0 V (flanco descendente) cuando la señal de voltaje alterna cruza por cero hacia los valores negativos de la señal evaluada (ver figura 2.15).

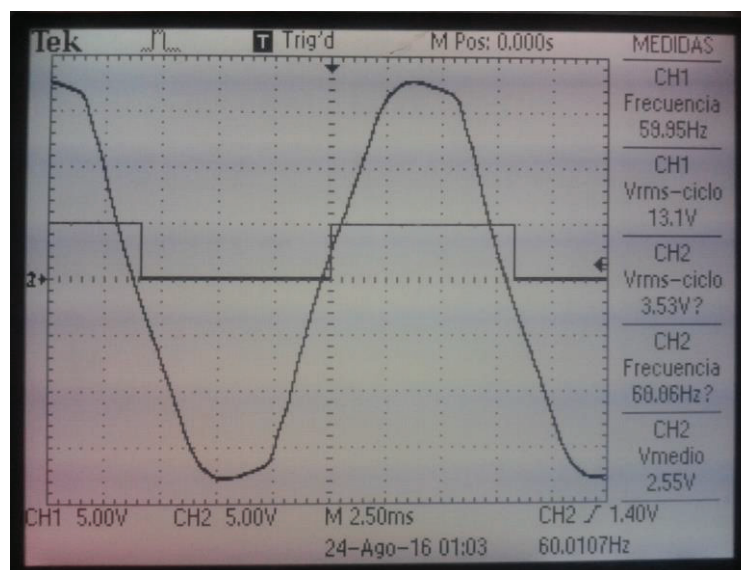


Figura 2.15 Forma de onda del circuito detector de cruce por cero.
[Elaboración propia]

El comparador LM339 tiene su salida en colector abierto, para la cual el fabricante recomienda el valor de la resistencia de pull up $R1=5.1K\Omega$.

Para las aplicaciones de respuesta rápida y comparación con entradas ruidosas o señal de voltaje alterna, que puede afectar el rango de voltaje de modo común de la entrada del comparador, de igual forma el fabricante recomienda usar un condensador de $0.1\mu F$ en derivación entre el voltaje de alimentación y tierra para conseguir la precisión deseada. [12]

2.6. UNIDAD DE POTENCIA

Los pines configurados como salidas del microcontrolador son útiles para controlar cargas que no consuman demasiada corriente, por ejemplo, un Led. Cabe recordar que el PIC18F4450 puede suministrar un máximo de corriente de 20 mA por cada pin, pero son insuficientes para controlar cargas mayores como una lámpara, bocinas o motores; por lo cual se decidió usar un módulo relé, que hará el acoplamiento entre la parte digital y la parte de potencia o fuerza.

Para ello, el módulo relé recibe las señales de respuesta del microcontrolador y gobierna la conexión o la desconexión de la bobina del contactor tripolar que enciende o apaga el motor trifásico.

2.6.1. MÓDULO RELÉ

Este driver contiene un optoacoplador, fototransistor DPC-817C (ver Figura 2.16), diseñado específicamente para aplicaciones de aislamiento entre la parte de control y potencia, cuyas características se detallan en la Tabla 2.2, y un relé o relay que son relevadores electromecánicos energizados con 5 V, capaces de manejar cargas hasta 10 A en 250 V AC.

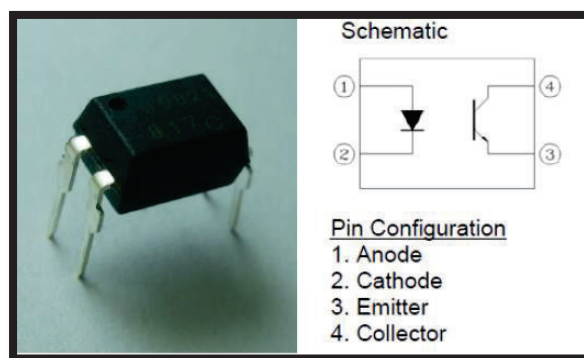


Figura 2.16 Esquema y configuración de pines del DPC-817C. [14]

Simbolo	Descripción	Valor
V_F	Voltaje directo de polarización	1.2 V
I_F	Corriente directa de polarización	5 Ma
T_J	Temperatura de juntura	-40 A 85 °C

Tabla 2.2 Características del fototransistor DPC-817C.

El optoacoplador proporciona un aislamiento óptico entre la parte digital y el circuito de encendido o apagado de la bobina del relé, también cuenta con un LED que sirve para indicar el estado del relé. El circuito de este driver se muestra en la Figura 2.17

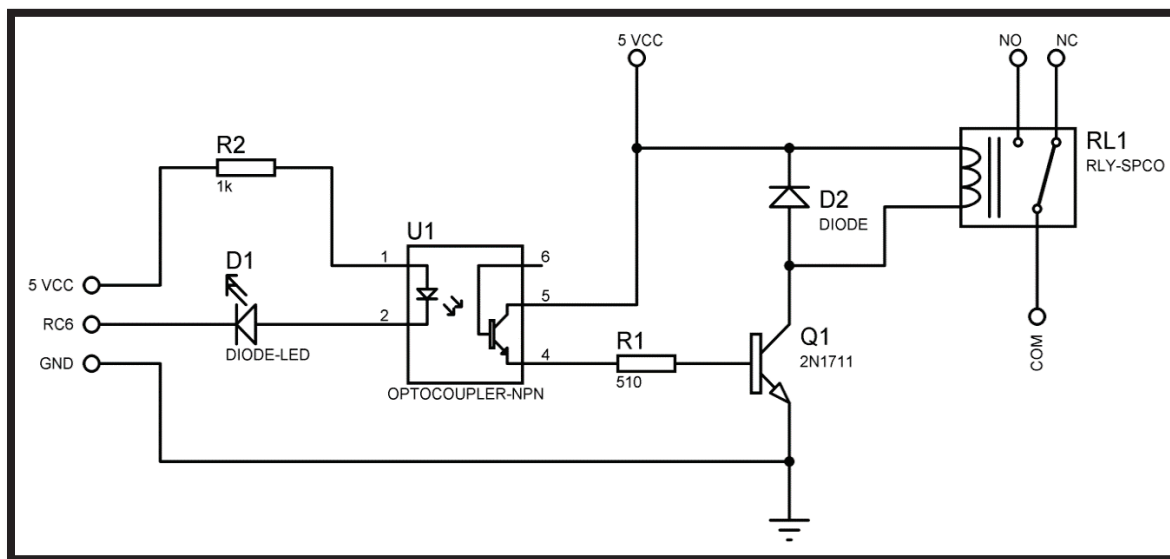


Figura 2.17 Circuito esquemático del modulo relé.
[Elaboración propia]

La salida RC6 del microcontrolador utilizado para el control del relé electromecánico tiene un voltaje de salida de 5 voltios. La resistencia R_2 limita la corriente directa de polarización del diodo del optoacoplador, y se calcula con la ecuación 2.7, utilizando el V_F e I_F proporcionados por la hoja de datos del elemento (ver anexo C).

$$R_2 = \frac{5V - V_F - V_{PD}}{I_F} \quad (2.7)$$

Donde:

V_F : Voltaje directo de polarización del diodo del optoacoplador

I_F : Corriente directa de polarización del diodo del optoacoplador

V_{PD} : Voltaje directo de polarización del diodo indicador de estado

$$R_2 = \frac{5V - 1.2V - 0.7V}{5 \text{ mA}}$$

$$R_2 = 620 \Omega$$

Por lo tanto, se seleccionó un valor estandar para $R_2=1K\Omega$

A partir del circuito de la Figura 2.17 se analiza el funcionamiento del circuito: la entrada RC6 está conectada al cátodo del diodo del optoacoplador a través del led indicador. El ánodo del diodo del optoacoplador se conecta a V_{cc} (+5 V) por intermedio de R_2 , una resistencia de $1K\Omega$. Estos tres componentes, el diodo indicador, el diodo del optoacoplador y la R_2 forman un circuito serie por el cual circula la corriente cuando la entrada está a un nivel BAJO (conectada a GND) y no circula si la entrada está a un nivel ALTO (conectada a V_{cc}).

El transistor del optoacoplador tiene su colector a V_{cc} y su emisor conectado a Q1 a través de una resistencia de 510Ω . Este es otro circuito serie por el cual circula corriente cuando el transistor del optoacoplador conduce al ser “iluminado” por su diodo, con lo que se introduce corriente en la base de Q1 a través de R_1 .

Finalmente, Q1 está conectado en una típica configuración emisor común, con su emisor a tierra (GND) y la bobina del relé como carga en el colector. Cuando circula corriente por la base desde el optoacoplador, Q1 se satura permitiendo el paso de la corriente a través de la bobina del relé, lo que produce el cierre del contacto normalmente abierto (NO) . El diodo D1 protege al transistor del voltaje que aparece en la bobina del relé cuando deja de circular corriente por la misma.

En resumen, al ponerse la entrada a nivel BAJO (0 V), se pone en saturación el transistor Q1 a través del optoacoplador, con lo que se cierra el contacto normal abierto del relé y posteriormente energizando la bobina del contactor bipolar.

2.6.2. DIMENSIONAMIENTO DEL CONTACTOR

El contactor es el elemento encargado de realizar la conexión o desconexión de la carga que supervisa el dispositivo desarrollado, para la cual se debe dimensionar y seleccionar el adecuado a la aplicación.

Los parámetros para el dimensionamiento y su selección están condicionados a las características de la carga, teniendo en cuenta que la carga es un motor trifásico de inducción. Se debe analizar la placa de datos del motor para obtener datos como la corriente nominal, voltaje de alimentación, potencia, frecuencia de operación, entre otras que son muy útiles para la selección. A continuación, en la Tabla 2.3 se muestra los datos de placa del motor a utilizar en la fase de pruebas.

Voltaje [V]: 220 / 380	Factor de potencia: 0,75
Corriente [A]: 1.4 / 0.8	Rpm: 1740
HP: ½	Frecuencia [Hz]: 60

Tabla 2.3 Datos del motor.

Además de las características de la carga, para el dimensionamiento se debe considerar algunos aspectos del contactor a seleccionar como:

- Voltaje de excitación de la bobina del contactor
- Frecuencia de operación de la bobina
- Números de contactos auxiliares y principales
- Voltaje de aislamiento
- Y la categoría de utilización establecida por la norma IEC

Se consideró un factor de seguridad de 125% de la potencia nominal del motor, para la cual la potencia que maneja el contactor se calcula con la ecuación 2.8:

$$P_{\text{contactor}} = 1.25 \times 0.5 \text{ HP} \quad (2.8)$$

$$P_{\text{contactor}} = 0.625 \text{ HP}$$

La categoría de utilización es AC3, que se refiere a los motores de jaula de ardilla con corrientes de arranque de 5 a 7 veces la corriente nominal del motor.

Por lo tanto, se seleccionó el contactor de la marca Schneider Electric, Modelo LC1D12 M7 (Ver Figura 2.18) el cual cumple con las necesidades y requerimientos de la carga (ver anexo D).



Figura 2.18 Contactor usado para la conexión/desconexión del motor. [15]

El contactor seleccionado tiene las siguientes características:

- Tipo de accionamiento: electromecánico
- Voltaje de servicio: 220 V AC y de aislamiento: 690 V
- Corriente de servicio: 12 A
- Número de polos: 3
- Contactos auxiliares: 1NC +1NO
- Categoría de utilización: AC3
- Frecuencia: 60 Hz
- Voltaje de excitación de la bobina: 220 V

2.7. UNIDAD DE PROCESAMIENTO Y CONTROL

El microcontrolador PIC18F4550 es el elemento principal del dispositivo de protección, el cual se encarga de procesar las señales (analógicas y digitales) enviadas desde el bloque anterior, para gobernar el relé y a su vez el encendido o apagado del motor en función del voltaje de alimentación en el caso de existir las fallas contempladas en su concepción. Además, debe presentar el valor correspondiente de voltaje de cada una de las tres fases, el porcentaje de desbalance y la secuencia de fases en el LCD (Display de Cristal Líquido).

2.7.1. ANÁLISIS DE LOS RECURSOS DEL PIC18F4550

En el momento que se ha definido todos los elementos que formarán parte del dispositivo de protección (medición de voltaje, pulsos sincronizados por los cruces por cero de cada forma de onda de las 3 fases, visualización de los parámetros de medición y otros), se hará un análisis de los recursos necesarios; para posteriormente, asignar cada función a un determinado puerto.

En la Tabla 2.4 y 2.5 se detalla el número de entradas y salidas necesarias dependiendo de la función a cumplir.

Entradas analógicas	
Función	Entradas
Voltaje de la L1 (fase-neutro)	1
Voltaje de la L2	1
Voltaje de la L3	1
TOTAL	3
Entradas digitales	
Pulsos para adquisición de la secuencia de Fases (L1-L2-L3)	3
Reset exterior (MCLR)	1
Pulsador "Seguir"	1
TOTAL	5

Tabla 2.4 Entradas analógicas y digitales.

Salidas digitales	
Función	Salidas
Señal de habilitación del Relé	1
LED's correspondientes a la conexión de las respectivas líneas	3
LCD Gráfico	14
TOTAL	18

Tabla 2.5 Salidas digitales.

En la tabla 2.6 se detallan los recursos necesarios del microcontrolador para la ejecución de presente proyecto.

Recursos Necesarios del MCU
Módulo Conversor Análogo a Digital de 10 bits
Modulo Timer 16 bits
Interrupción por desbordamiento del Timer

Tabla 2.6 Recursos necesarios.

2.7.2. ASIGNACIÓN DE PINES

En la figura 2.19 se observa el diagrama del microcontrolador con la respectiva asignación de los pines, tanto de entrada como de salida, dependiendo de la función a desempeñar.

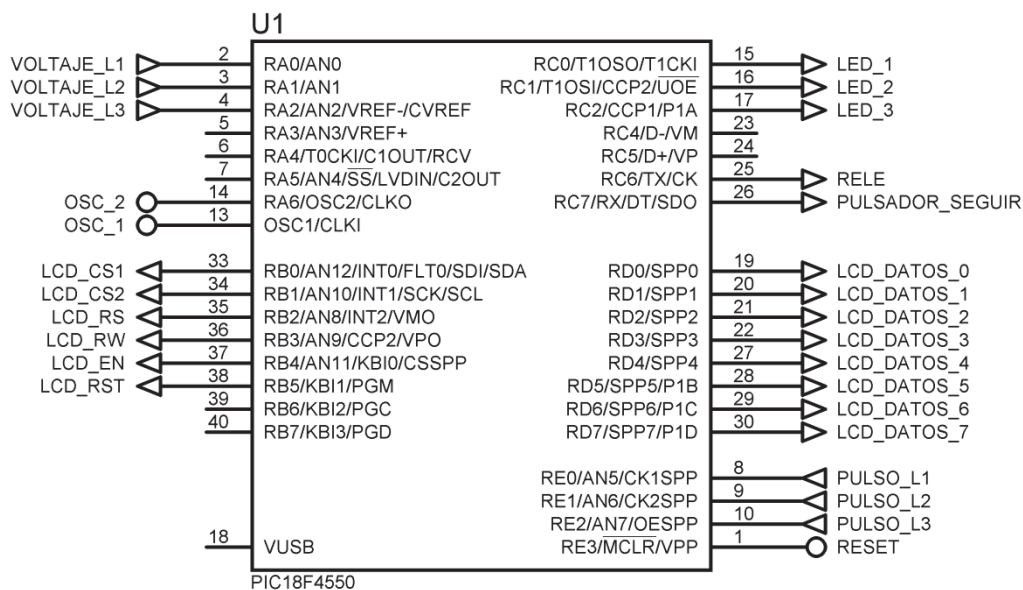


Figura 2.19 Asignación de pines.
[Elaboración propia]

Los pines sin utilizar, se configuran como salidas digitales y programadas en nivel alto.

2.7.3. CONFIGURACIÓN DEL OSCILADOR PRINCIPAL

El PIC18F4550 cuenta con tres tipos de osciladores, uno primario (utiliza cristal o señal externa), otro secundario (utiliza el Timer 1), y el último un bloque de oscilador interno (utiliza el oscilador interno o el INTRC),

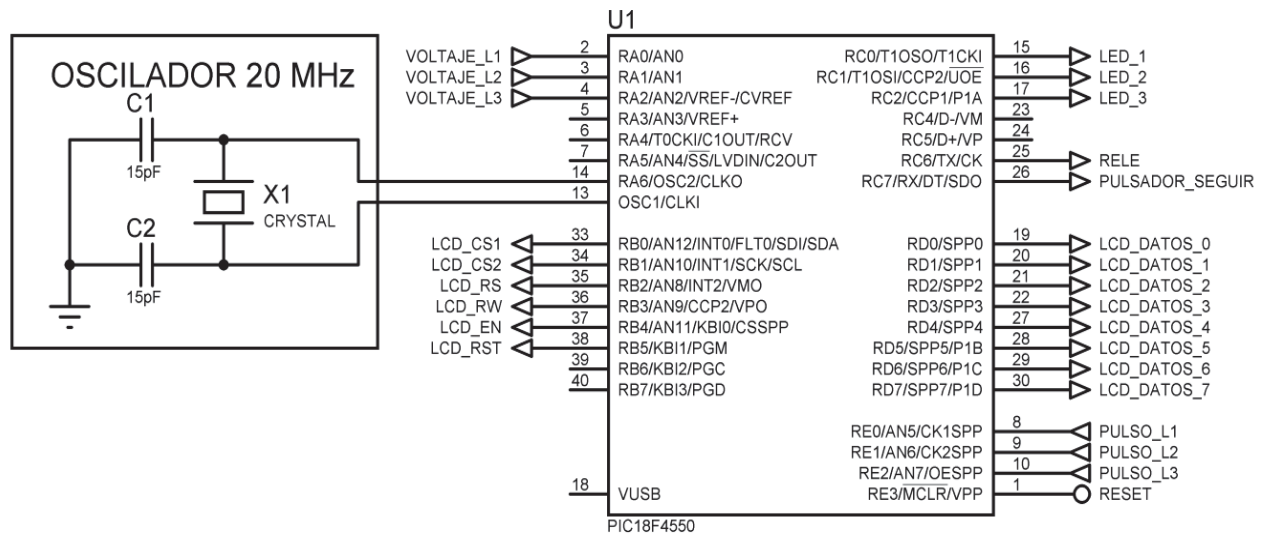


Figura 2.20 Configuración del oscilador primario.
[Elaboración propia]

La Figura 2.20 muestra el diagrama eléctrico del oscilador primario elegido para este proyecto, en este modo un cristal o resonador cerámico se conecta a los pines del PIC identificados como OSC1 y OSC2 que corresponde a los pines número 13 y 14 respectivamente.

Esta configuración es la que se utiliza en la mayoría de los proyectos en los que no se utiliza el módulo USB. Normalmente se usa el modo de oscilador XT para cristales hasta 4 MHz y la configuración HS para cristales más de 4 MHz.

Mediante el uso de un Postscaler interno se puede reducir la frecuencia que entrega el cristal para hacer funcionar al CPU a diferentes velocidades dividiendo dicha frecuencia para el factor de Postscaler (determinada por los bits de configuración de CPUDIV). Para propósitos de nuestro proyecto, se decidió usar la frecuencia que entrega el cristal directamente en el CPU del microcontrolador; para ello se debe utilizar uno de los modos de oscilador externo que no utilice el PLL Prescaler (Ver figura 2.21).

La frecuencia de trabajo del CPU se puede seleccionar mediante el uso del "Oscillator Postscaler (CPUDIV)", una vez definido el cristal de cuarzo a utilizar (20 Mhz); se puede escoger la frecuencia directa del reloj, o 1/2 o 1/3 o 1/4 de dicha frecuencia.

La frecuencia seleccionada se resalta en amarillo conforme se detalla en la tabla 2.7.

Frecuencia del Oscilador	Divisor CPUDIV1:CPUDIV0 Oscillator Postcaler	Configuración en c18 para los modos: FOSC = XT_XT o FOSC = HS	Frecuencia de trabajo del CPU
4 Mhz	Sin división	#pragma config CPUDIV = OSC1_PLL2	4 Mhz
4 Mhz	/ 2	#pragma config CPUDIV = OSC2_PLL3	2 Mhz
4 Mhz	/ 3	#pragma config CPUDIV = OSC3_PLL4	1,33 Mhz
4 Mhz	/ 4	#pragma config CPUDIV = OSC4_PLL6	1 Mhz
8 Mhz	Sin división	#pragma config CPUDIV = OSC1_PLL2	8 Mhz
8 Mhz	/ 2	#pragma config CPUDIV = OSC2_PLL3	4 Mhz
8 Mhz	/ 3	#pragma config CPUDIV = OSC3_PLL4	2,67 Mhz
8 Mhz	/ 4	#pragma config CPUDIV = OSC4_PLL6	2 Mhz
12 Mhz	Sin división	#pragma config CPUDIV = OSC1_PLL2	12 Mhz
12 Mhz	/ 2	#pragma config CPUDIV = OSC2_PLL3	6 Mhz
12 Mhz	/ 3	#pragma config CPUDIV = OSC3_PLL4	4 Mhz
12 Mhz	/ 4	#pragma config CPUDIV = OSC4_PLL6	3 Mhz
16 Mhz	Sin división	#pragma config CPUDIV = OSC1_PLL2	16 Mhz
16 Mhz	/ 2	#pragma config CPUDIV = OSC2_PLL3	8 Mhz
16 Mhz	/ 3	#pragma config CPUDIV = OSC3_PLL4	5,33 Mhz
16 Mhz	/ 4	#pragma config CPUDIV = OSC4_PLL6	4 Mhz
20 Mhz	Sin división	#pragma config CPUDIV = OSC1_PLL2	20 Mhz
20 Mhz	/ 2	#pragma config CPUDIV = OSC2_PLL3	10 Mhz
20 Mhz	/ 3	#pragma config CPUDIV = OSC3_PLL4	6,67 Mhz
20 Mhz	/ 4	#pragma config CPUDIV = OSC4_PLL6	5 Mhz
24 Mhz	Sin división	#pragma config CPUDIV = OSC1_PLL2	24 Mhz
24 Mhz	/ 2	#pragma config CPUDIV = OSC2_PLL3	12 Mhz
24 Mhz	/ 3	#pragma config CPUDIV = OSC3_PLL4	8 Mhz
24 Mhz	/ 4	#pragma config CPUDIV = OSC4_PLL6	6 Mhz
40 Mhz	Sin división	#pragma config CPUDIV = OSC1_PLL2	40 Mhz
40 Mhz	/ 2	#pragma config CPUDIV = OSC2_PLL3	20 Mhz
40 Mhz	/ 3	#pragma config CPUDIV = OSC3_PLL4	13,33 Mhz
40 Mhz	/ 4	#pragma config CPUDIV = OSC4_PLL6	10 Mhz
48 Mhz	Sin división	#pragma config CPUDIV = OSC1_PLL2	48 Mhz
48 Mhz	/ 2	#pragma config CPUDIV = OSC2_PLL3	24 Mhz
48 Mhz	/ 3	#pragma config CPUDIV = OSC3_PLL4	16 Mhz
48 Mhz	/ 4	#pragma config CPUDIV = OSC4_PLL6	12 Mhz

Tabla 2.7 Múltiples frecuencias de trabajo del CPU. [7]

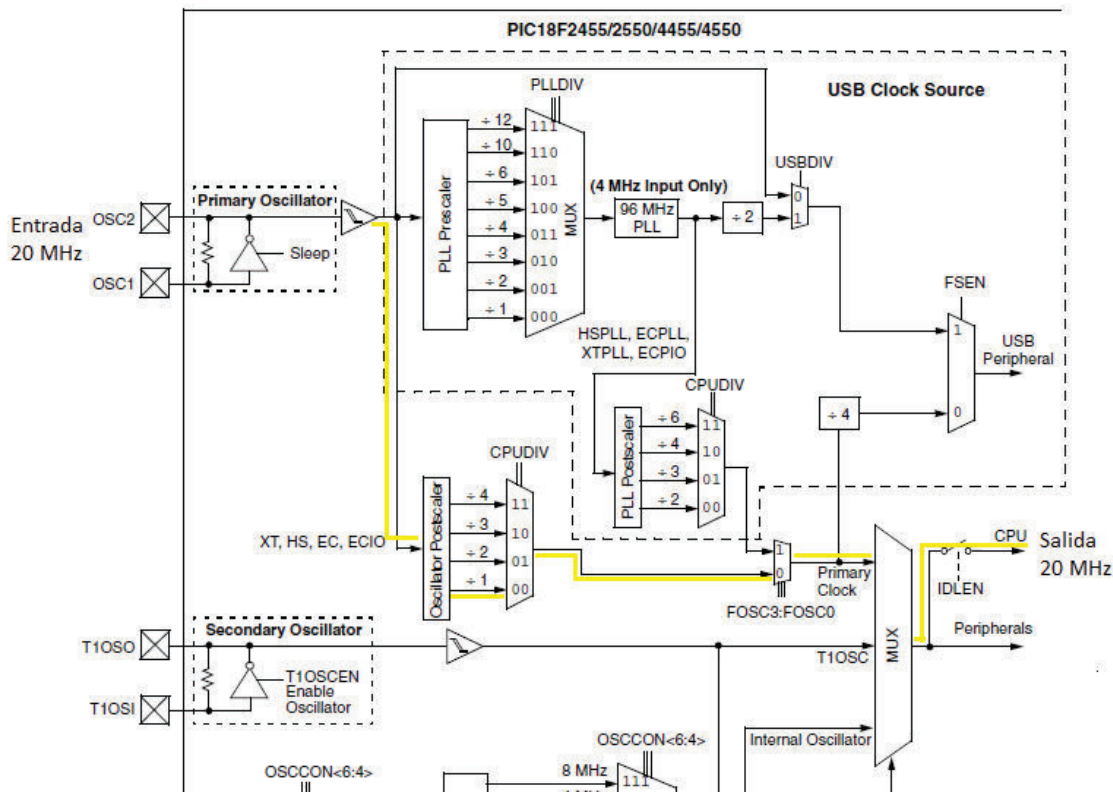


Figura 2.21 Diagrama de bloques del oscilador y su configuración. [7]

En la figura 2.21 se indica en amarillo el recorrido que hace la señal desde que sale del oscilador primario, pasando por el Oscillator Postscaler CPUDIV que divide la frecuencia por 1, hasta llegar al núcleo del microcontrolador (CPU Clock).

2.7.4. INTERFAZ DEL MÓDULO GLCD Y EL PIC18F4550

Para la interconexión con el GLCD se utilizó los puertos D y B del PIC. El puerto B es utilizado como puerto de control y el puerto D como puerto de datos. La relación de los pines del PIC y el GLCD se detallan en la tabla 2.8.

PIN [PIC]	PIN [GLCD]
RB0	Línea CS1 de GLCD
RB1	Línea CS2 de GLCD
RB2	Línea RS de GLCD
RB3	Línea RW de GLCD
RB4	Línea EN de GLCD
RB5	Línea RST de GLCD
RD0-RD7	Bus de datos de 8 bits

Tabla 2.8 Relación de pines del PIC y el GLCD.

En la figura 2.22 se muestra el diagrama eléctrico de la interfaz del PIC y el módulo GLCD.

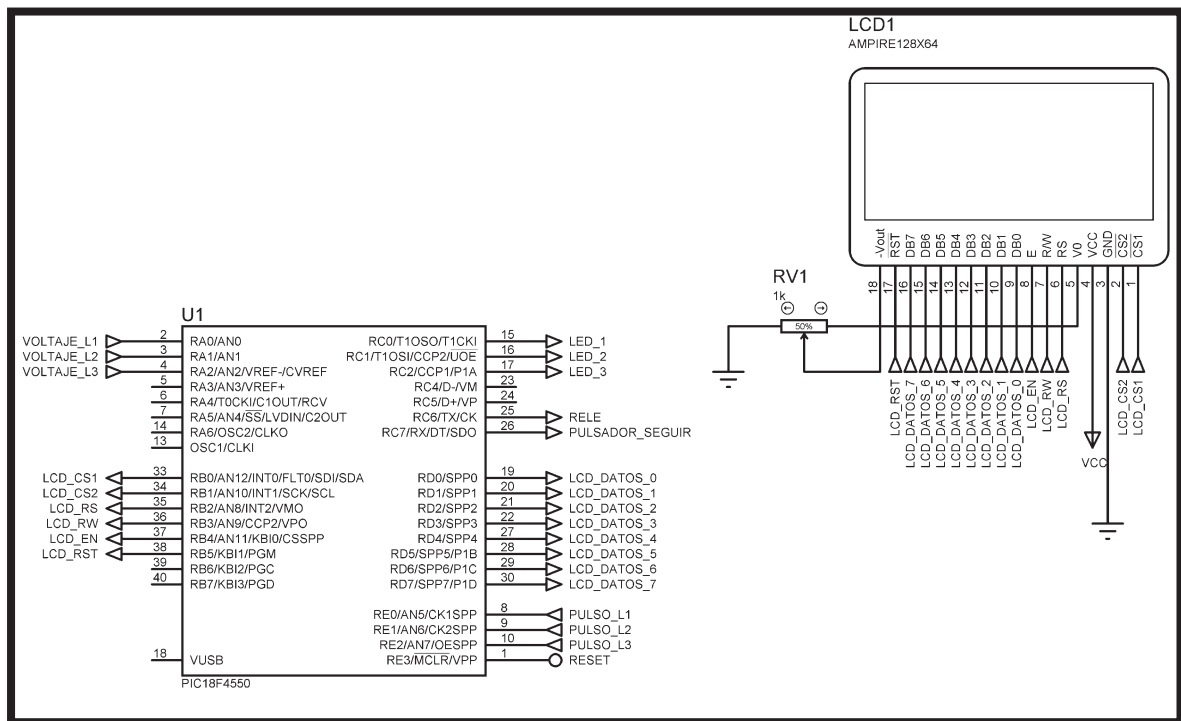


Figura 2.22 Diagrama eléctrico de la interfaz PIC y módulo GLCD.
[Elaboración propia]

Para tener una correcta comunicación entre el PIC y el GLCD, se debe declarar la relación de los pines de microcontrolador y la pantalla. Para ello se asigna, mediante programación un identificador a cada pin del MCU.

Los datos se envían hacia la pantalla de manera paralela desde el puerto D. Por la razón que se envían 8 bits simultáneamente, los datos se declaran con tipo char (carácter).

```
char GLCD_DataPort at PORTD; //Se asigna los pines de datos del GLCD
                                // al puerto D del MCU
```

Mientras que para el puerto de control se declara individualmente porque cada bit desempeña una función distinta.

```
sbit GLCD_CS1 at RB0_bit; // Se asigna los pines de control del GLCD
sbit GLCD_CS2 at RB1_bit; // al puerto B del MCU
sbit GLCD_RS at RB2_bit;
sbit GLCD_RW at RB3_bit;
sbit GLCD_EN at RB4_bit;
sbit GLCD_RST at RB5_bit;
```

Y también se asigna identificadores a los registros de control de E/S (TRISX).

```
sbit GLCD_CS1_Direction at TRISB0_bit;
sbit GLCD_CS2_Direction at TRISB1_bit;
sbit GLCD_RS_Direction at TRISB2_bit;
sbit GLCD_RW_Direction at TRISB3_bit;
sbit GLCD_EN_Direction at TRISB4_bit;
sbit GLCD_RST_Direction at TRISB5_bit;
```

2.8. DESARROLLO DEL SOFTWARE PARA EL MICROCONTROLADOR

El software usado para escribir el programa para el microcontrolador es MikroC *PRO for PIC*, el cual es un paquete computacional creado por *Mikroelectronica*, para desarrollar y depurar aplicaciones basadas en los microcontroladores PIC de las familias 16XXXX-18XXXX, fabricados por *Microchip*. Este entorno de desarrollo integrado brinda al usuario un ambiente amigable, al contar con librerías que facilitan el desarrollo de aplicaciones y de fácil aprendizaje; lo que lo hace una herramienta potente para la creación de aplicaciones escritas en lenguaje C/C++.

El programa desarrollado para esta aplicación se lo puede dividir en partes, la configuración del microcontrolador, inicialización de las variables y constantes necesarias, el bucle de trabajo principal y por último la subrutina de interrupción por desbordamiento del TIMER 1. En la figura 2.23, se muestra un diagrama de flujo general.

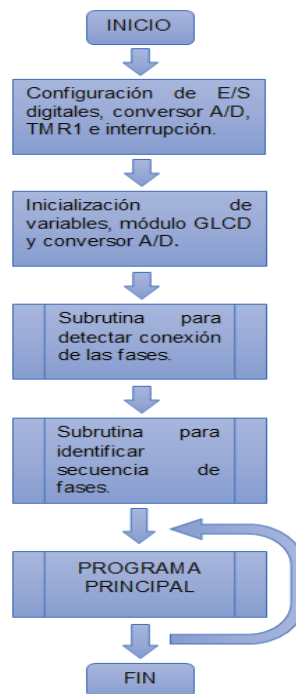


Figura 2.23 Diagrama de flujo general.
[Elaboración propia]

2.8.1. CONFIGURACIÓN DEL MICROCONTROLADOR PIC18F4550

Una vez que ya se ha asignado a cada PIN una función específica, procederemos a la configuración del chip. El microcontrolador PIC18F4550 tiene 13 canales analógicos distribuidos en los puertos A, B y E, de los cuales se usarán solo 3 canales ubicados en el puerto A. Para ello, se debe configurar los canales AN0, AN1 y AN2 como entradas analógicas y el resto de canales se debe configurar como digitales. Para cumplir el cometido se debe manipular el registro ADCON1, el cual se encarga de seleccionar y habilitar los canales según las necesidades del proyecto y además nos permite escoger el voltaje de referencia para el módulo A/D.

REGISTER 21-2: ADCON1: A/D CONTROL REGISTER 1

U-0	U-0	R/W-0	R/W-0	R/W-0 ⁽¹⁾	R/W ⁽¹⁾	R/W ⁽¹⁾	R/W ⁽¹⁾
—	—	VCFG0	VCFG0	PCFG3	PCFG2	PCFG1	PCFG0
bit 7							bit 0

Legend:

R = Readable bit W = Writable bit U = Unimplemented bit, read as '0'
 -n = Value at POR '1' = Bit is set '0' = Bit is cleared x = Bit is unknown

bit 7-6 **Unimplemented:** Read as '0'

bit 5 **VCFG0:** Voltage Reference Configuration bit (VREF- source)

1 = VREF- (AN2)

0 = Vss

bit 4 **VCFG0:** Voltage Reference Configuration bit (VREF+ source)

1 = VREF+ (AN3)

0 = VDD

bit 3-0 **PCFG3:PCFG0:** A/D Port Configuration Control bits:

PCFG3: PCFG0	AN12	AN11	AN10	AN9	AN8	AN7 ⁽²⁾	AN6 ⁽²⁾	AN5 ⁽²⁾	AN4	AN3	AN2	AN1	AN0
0000 ⁽¹⁾	A	A	A	A	A	A	A	A	A	A	A	A	A
0001	A	A	A	A	A	A	A	A	A	A	A	A	A
0010	A	A	A	A	A	A	A	A	A	A	A	A	A
0011	D	A	A	A	A	A	A	A	A	A	A	A	A
0100	D	D	A	A	A	A	A	A	A	A	A	A	A
0101	D	D	D	A	A	A	A	A	A	A	A	A	A
0110	D	D	D	D	A	A	A	A	A	A	A	A	A
0111 ⁽¹⁾	D	D	D	D	D	A	A	A	A	A	A	A	A
1000	D	D	D	D	D	D	A	A	A	A	A	A	A
1001	D	D	D	D	D	D	D	A	A	A	A	A	A
1010	D	D	D	D	D	D	D	D	A	A	A	A	A
1011	D	D	D	D	D	D	D	D	D	A	A	A	A
1100	D	D	D	D	D	D	D	D	D	D	A	A	A
1101	D	D	D	D	D	D	D	D	D	D	D	A	A
1110	D	D	D	D	D	D	D	D	D	D	D	D	A
1111	D	D	D	D	D	D	D	D	D	D	D	D	D

A = Analog input

D = Digital I/O

Figura 2.24 Configuración del Registro ADCON1. [7]

En la Figura 2.24, se muestra resaltado en amarillo los canales seleccionados. Además, se asignó el voltaje de referencia del módulo A/D a los pines de alimentación del chip VDD y VSS.

MikroC es un compilador con acceso a cada uno de estos registros de función especial, entonces la línea de código para configurar el módulo A/D quedará de la siguiente manera:

```
ADCON1=0B00001100;           //Configurar AN1, AN2 y AN3 como entradas analógicas
```

Nótese que se carga con un valor en notación binaria (base 2) en el registro ADCON1, que previamente fue analizada según las necesidades del proyecto.

Ahora, el resto de pines se configuran como entradas o salidas digitales. Mediante la manipulación del registro TRIS se puede configurar el sentido de dirección de los datos; si le asignamos un valor de “1”, se configura como entradas digitales; por otra parte, si se le asigna “0” se configura como salidas digitales a los pines del puerto seleccionado.

Por ejemplo, se desea configurar al nibble alto RD<7:4> del puerto D como entradas digitales, y el nibble bajo RD<3:0> como salidas digitales; en *MikroC* quedaría de la siguiente forma:

```
TRISD=0B11110000; //Configurar entradas y salidas digitales
```

Analizando este fragmento de código lo interpretaríamos de la siguiente forma: Se carga un número binario, al nibble bajo se le asigna un 0 (salidas) y al nibble alto se le asigna un 1 (entradas) al registro TRIS del puerto D.

Otro recurso necesario que se usará a lo largo de este proyecto es el TMR1, como temporizador. La habilitación de dicho módulo se dará solo en casos de falla; durante ese tiempo funcionará básicamente como un temporizador regresivo, en el momento que llegue a cero inhabilitará la señal de activación del módulo relé.

De igual forma al TMR1 lo gobiernan varios registros, pero solo nos interesa el registro que gobierna directamente su control y es el registro T1CON.

REGISTER 12-1: T1CON: TIMER1 CONTROL REGISTER

R/W-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
RD16	T1RUN	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	TMR1CS	TMR1ON
bit 7							bit 0

Legend:

R = Readable bit W = Writable bit U = Unimplemented bit, read as '0'
 -n = Value at POR '1' = Bit is set '0' = Bit is cleared x = Bit is unknown

bit 7	RD16: 16-Bit Read/Write Mode Enable bit 1 = Enables register read/write of Timer1 in one 16-bit operation 0 = Enables register read/write of Timer1 in two 8-bit operations
bit 6	T1RUN: Timer1 System Clock Status bit 1 = Device clock is derived from Timer1 oscillator 0 = Device clock is derived from another source
bit 5-4	T1CKPS1:T1CKPS0: Timer1 Input Clock Prescale Select bits 11 = 1:8 Prescale value 10 = 1:4 Prescale value 01 = 1:2 Prescale value 00 = 1:1 Prescale value
bit 3	T1OSCEN: Timer1 Oscillator Enable bit 1 = Timer1 oscillator is enabled 0 = Timer1 oscillator is shut off The oscillator inverter and feedback resistor are turned off to eliminate power drain.
bit 2	T1SYNC: Timer1 External Clock Input Synchronization Select bit When TMR1CS = 1; 1 = Do not synchronize external clock input 0 = Synchronize external clock input When TMR1CS = 0; This bit is ignored. Timer1 uses the internal clock when TMR1CS = 0.
bit 1	TMR1CS: Timer1 Clock Source Select bit 1 = External clock from RC0/T1OSO/T13CKI pin (on the rising edge) 0 = Internal clock (Fosc/4)
bit 0	TMR1ON: Timer1 On bit 1 = Enables Timer1 0 = Stops Timer1

Figura 2.25 Configuración del Registro T1CON. [7]

En la figura 2.25 se puede observar la configuración del módulo Timer 1 como temporizador en modo de 16 bits. En *MikroC*, la configuración del registro *T1CON* quedaría de la siguiente forma:

```
T1CON=OB00110000; // Configuración del TMR1 como temporizador 16 bits
```

La interpretación es el módulo TMR1 se fija como temporizador de 16 bits con preescaler 1:8 y además se encuentra apagado.

Cabe recalcar la importancia del ciclo de oscilador (T_{OSC}) y el ciclo de instrucción (T_{CI}); el ciclo de oscilador se obtiene directamente del cristal externo conectado a los pines OSC1 y OSC2 en este caso, y el ciclo de instrucción se define como cuatro veces el ciclo de oscilador.

Por ejemplo, si se requiere usar un cristal de 4 MHz y se desea determinar el ciclo de instrucción, las ecuaciones 2.9 y 2.10 ayudan a facilitar el cálculo:

$$T_{osc} = 1/F_{osc} \quad (2.9)$$

Donde:

T_{osc} : Ciclo de oscilador

F_{osc} : Frecuencia del oscilador (cristal)

T_{ci} : Ciclo de instrucción

$$T_{osc} = 1/4 \text{ MHz}$$

$$T_{osc} = 0.25 \mu\text{seg}$$

Ahora el ciclo de instrucción es:

$$T_{ci} = 4 * T_{osc} \quad (2.10)$$

$$T_{ci} = 4 * 0.25 \mu\text{seg}$$

$$T_{ci} = 1 \mu\text{seg}$$

Conocer el ciclo de instrucción es un dato fundamental para usar el módulo TMR1 como temporizador, por la sencilla razón que se produce un incremento en los registros TMR1H y TMR1L con cada ciclo de instrucción ejecutado; cuando la relación del preescalador es 1:1.

Para temporizar intervalos de tiempo en segundos, minutos u horas, es recomendable usar la interrupción del TMR1.

El desbordamiento de los registros TMR1H y TMR1L (desde 0x0000 a 0xFFFF) genera una interrupción por desbordamiento, la cual hace que el bit TMR1IF del registro PIR sea igual a 1. La generación de esta interrupción se puede habilitar/deshabilitar con el bit TMR1IE del registro PIE1. El bit TMR1IF tiene que ser borrado por software antes de volver a habilitarlo.

Para el proyecto se desea temporizar un intervalo de 100 ms, con un cristal de 20 MHz, entonces la solución es la siguiente:

$$T_{osc} = 1/20 \text{ MHz} = 0.05 \mu\text{seg.}$$

$$T_{ci} = 4 * T_{osc} = 0.2 \mu\text{seg.}$$

Se denomina intervalo de tiempo a temporizar como "T" → T = 100 ms.

La ecuación 2.11 nos ayuda a encontrar el valor de carga inicial para el TMR1 en sus registros (TMR1H y TMR1L).

$$T = N * T_{ci} * n \quad (2.11)$$

Donde:

N = Número de ciclos de instrucción/preescaler.

n = Valor de preescaler

Reemplazando valores:

$$100 \text{ mseg} = N * 0.2 \mu\text{seg} * 8$$

$$N = 62500$$

Cabe notar que el módulo TMR1 va desde 0 a 65535 y existen 65536 números. El valor de “ N ” no puede superar dicho valor, por la razón que no se puede cargar números negativos en estos registros.

Para calcular el valor de carga “ Q ”, resta un simple paso y se calcula usando la ecuación 2.12:

$$Q = 62536 - N \quad (2.12)$$

$$Q = 65536 - 62500$$

$$Q = 3036$$

En hexadecimal

$$Q = 0BDC$$

Este valor se debe cargar en los dos registros TMR1H y TMR1L.

A continuación se muestra las líneas de programación correspondientes a la configuración del microcontrolador, del módulo TMR1 y su interrupción.

```
ADCON1=0B00001100, // RA <2:0> fijar como entradas analógicas
UCON=0B00000000; // Módulo USB desactivado

PORTA=0B00000000; // Valor de inicialización de los puertos
PORTB=0B00000000;
PORTC=0B00000000;
PORTD=0B00000000;
PORTE=0B0000;

TRISA=0B00001111; // RA <2:0> fijar como entradas, el resto como salidas
TRISB=0B00000000; // RB <7:0> fijar como salidas digitales
TRISC=0B10000000; // RC7 fijar como entrada digital, el resto como salidas
TRISD=0B00000000; // RD <7:0> fijar como salidas digitales
TRISE=0B1111; // RE <2:0> fijar como entradas digitales
RELE=1; // inhabilitar la señal del RELÉ

T1CON=0B00110000; // TMR1 como temporizador, preescaler 1:8, TMR1 OFF
```

```

TMR1IF_bit=0;          // Bandera de Interrupción encendida en 0

TMR1H=0X0B;           // Carga inicial 3036(0X0BDC), para interrupción de
TMR1L=0XDC;           // intervalos de tiempo cada 100 ms

TMR1IE_bit=1;        // Habilitar interrupción del TMR1 por desbordamiento
INTCON=0XC0;         // Habilitar interrupciones globales y periféricas
Glcd_Init ();        // Inicializar módulo GLCD con la configuración
predeterminada

```

2.8.2. IMÁGENES Y MENSAJES DE PRESENTACIÓN

Una vez que se han realizado todas las configuraciones e inicializaciones de los recursos requeridos para el proyecto, se ejecuta la rutina para la presentación e información del proyecto en la pantalla gráfica.

Esta rutina consiste en el despliegue del escudo de la Escuela Politécnica Nacional y el logo de la carrera de Tecnología en Electromecánica en la pantalla, además presenta la información de los autores.

La pantalla gráfica es monocromática de 128x64 píxeles, es decir, la resolución es limitada para la cual se debe editar la imagen a desplegar, Esto provocó la necesidad de redimensionar y editar la imagen en *Microsoft Paint* (Ver figura 2.26).

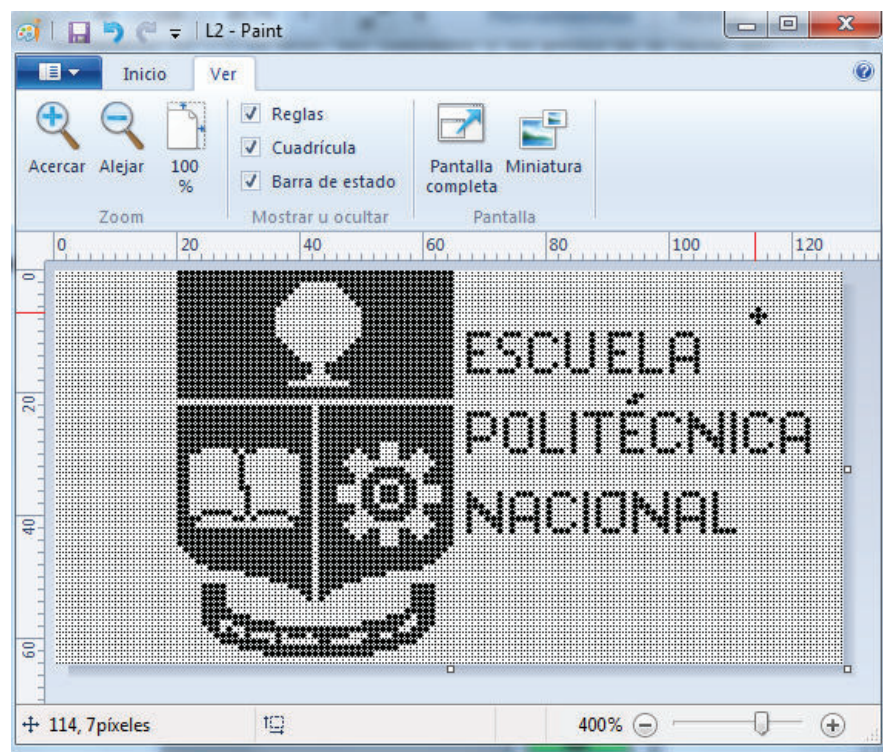


Figura 2.26 Edición de imagen.
[Elaboración propia]

Una vez obtenido el bitmap (mapa de bits monocromático), se utilizó la herramienta *Graphic LCD bitmap editor* disponible en el menú de herramientas (*Tools*) que incluye el compilador de *MikroC* para generar un arreglo de 1024 Bytes (Ver figura 2.27).

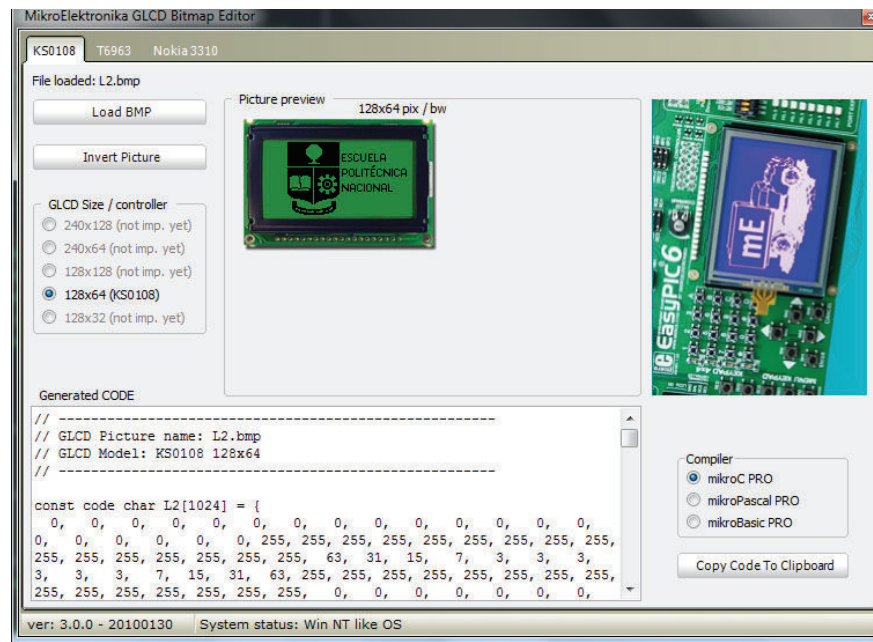


Figura 2.27 Herramienta *Graphic LCD bitmap editor*.
[Elaboración propia]

El arreglo es utilizado por las siguientes líneas de programación para la graficación de la imagen en la pantalla.

```

Glcd_Image(EPN);           // Graficar escudo de la EPN
Delay_1sec();              // Esperar 2 segundos
Delay_1sec();

Glcd_Fill(0x00);           //Limpiar LCD gráfico
Delay_ms(10);

Glcd_Image(Electromecanica); // Graficar logo de electromecánica
Delay_1sec();
Delay_1sec();

Glcd_Fill(0x00);           //Limpiar LCD gráfico
Delay_ms(10);

```

El código completo con el arreglo para las dos imágenes se incluye al final en el ANEXO F.

En la figura 2.28 muestra el resultado del despliegue de la imagen en la pantalla.



Figura 2.28 Imagen desplegada en el GLCD.
[Elaboración propia]

Después de los gráficos, se presenta la información de los autores, carrera y el tipo de proyecto. Las siguientes líneas de código solo imprimen los mensajes previamente grabados en la memoria EEPROM interna del MCU que se explicará en la sección de: Mensajes escritos en la memoria EEPROM.

```
// Escribir los mensajes grabados en la EEPROM en la posición deseada
mensajes(9,18,0,1);
mensajes(10,18,1,1);
Glcd_Box(0,22,127,32,2);
mensajes(11,18,3,0);
mensajes(12,12,5,1);
mensajes(13,12,6,1);
mensajes(14,18,7,1);

Delay_ms(2000);           // Retardo de 2 segundos
Glcd_Fill(0x00);         //Limpiar LCD gráfico
Delay_ms(10);
```

La figura 2.29 muestra una captura donde se aprecia información importante del proyecto



Figura 2.29 Información del proyecto.
[Elaboración propia]

2.8.3. RUTINA PARA VERIFICAR LA CONEXIÓN DE LAS 3 FASES

En la primera rutina, el programa verifica si las tres fases están conectadas al dispositivo de protección. Esto con el objetivo de asegurar que las 3 líneas tengan un nivel de voltaje, y de esa forma facilitar la operación del dispositivo desarrollado.

Para ello, se utilizó el conversor A/D del microcontrolador y los respectivos circuitos acondicionadores de voltaje de cada una de las fases. Ahora, recordemos que cuando se conecte una fase y el neutro del sistema trifásico a los respectivos terminales identificados en el dispositivo, éste hará que el bloque que acondicionaba la señal AC, devuelva una señal DC en un rango de 0 a 5 V DC.

Mediante esta respuesta del bloque anterior y haciendo uso del módulo conversor A/D, se desarrolló la rutina para identificar la conexión o la desconexión de las líneas de alimentación (L1-L2-L3) al dispositivo.

Si la fase está conectada, se encenderá el LED correspondiente a dicha línea evaluada y además se podrá visualizar en la pantalla la línea conectada (Ver figura 2.30).



Figura 2.30 Pantalla y señales luminosas de líneas conectadas.
[Elaboración propia]

El código para verificar el estado de conexión o desconexión de cada línea es el siguiente:

```
*----- Identificación de las líneas conecadas -----*/
do {
  if (ADC_Read(0)>700){      // Leer y evaluar la señal de L1, ¿L1 conectado?
    RC0_bit=1;              // Encender LED correspondiente a L1
    Glcd_Write_Char('L',36,4,1);
    Glcd_Write_Char('1',42,4,1);
  }
  else {
    RC0_bit=0;              // Apagar LED de L1
    Glcd_Write_Char(' ',36,4,1);
    Glcd_Write_Char(' ',42,4,1);
  }
  Delay_us(8);

  if (ADC_Read(1)>700){      // Leer y evaluar la señal de L2, ¿L2 conectado?
    RC1_bit=1;              // Encender LED correspondiente a L2
    Glcd_Write_Char('L',60,4,1);
    Glcd_Write_Char('2',66,4,1);
  }
  else{
    RC1_bit=0;              // Apagar LED de L2
    Glcd_Write_Char(' ',60,4,1);
    Glcd_Write_Char(' ',66,4,1);
  }
  Delay_us(8);

  if (ADC_Read(2)>700){      // Leer y evaluar la señal de L3, ¿L3 conectado?
    RC2_bit=1;              // Encender LED correspondiente a L3
    Glcd_Write_Char('L',84,4,1);
    Glcd_Write_Char('3',90,4,1);
  }
  else{
    RC2_bit=0;              // Apagar LED de L3
    Glcd_Write_Char(' ',84,4,1);
    Glcd_Write_Char(' ',90,4,1);
  }
  Delay_us(8);

  if (RC0_bit&RC1_bit&RC2_bit)
    Line_OK=0;              //Variable para indicar que las líneas están conecadas
  else Line_OK=1;
}
while(Line_OK || Seguir);
```

Como se observa, la lectura del módulo A/D determina si la línea está o no conecada.

Cabe mencionar que para seguir con el flujo del programa, las tres líneas deben estar conecadas al dispositivo desarrollado y además se debe presionar el botón“SEGUIR”.

2.8.4. RUTINA PARA IDENTIFICAR LA SECUENCIA DE FASES

Continuando con el diagrama de flujo general, al finalizar la rutina de verificación de las líneas conectadas, procedemos a identificar la secuencia de fases.

El bloque de acondicionamiento de señales, para identificar la secuencia de fases, transforma la señal senoidal en onda cuadrada para cada fase, como se ilustra en la Figura 2.31.

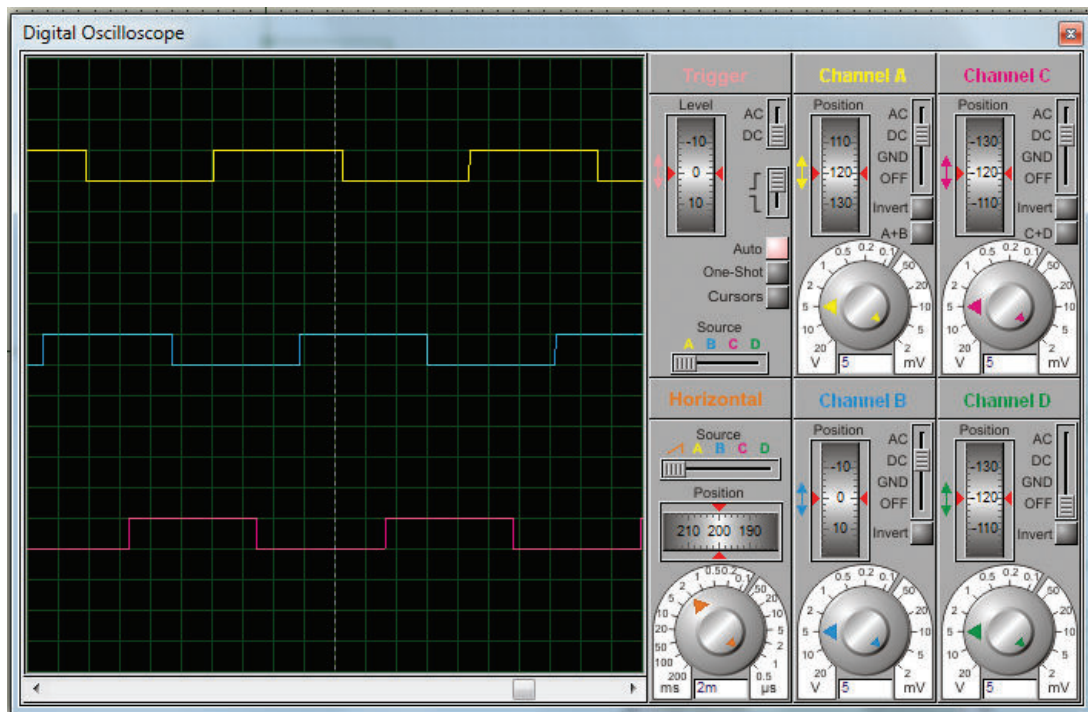


Figura 2.31 Ondas cuadradas correspondientes a cada fase.
[Elaboración propia]

De esta forma podemos tratar las señales como de tipo digital. Así, para la identificación de la secuencia de fases, tomamos como referencia el pulso de la línea L1; la L1 está asignada al pin número 8 (RE0).

La identificación de la secuencia comienza cuando se detecta el cambio de nivel de voltaje (0 a 5 V o flanco ascendente) en la línea de referencia. Una vez que se ha detectado el flanco ascendente de dicha línea, el programa debe esperar 15 ms aproximadamente (330°); ésto con el objetivo de leer el valor máximo en la línea L3 y el valor L2, que son de tipo digital 1 ó 0 lógico.

Para la secuencia positiva, transcurrido los 15 ms de espera, el valor lógico en la línea L2 será "0" y en la línea L3 será "1". Esta combinación define la secuencia positiva, como se indica en la Figura 2.32.

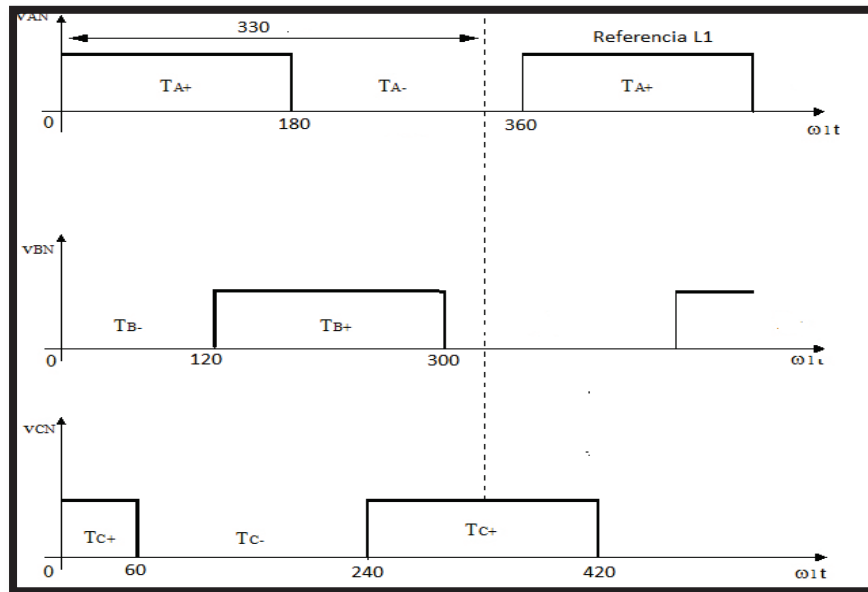


Figura 2.32 Secuencia de fases positiva, formas de onda.
[Elaboración propia]

Por último, para la secuencia negativa, transcurrido el mismo tiempo de espera, el valor lógico en la línea L2 será “1” y en la línea L3 será “0”. Esta combinación define la secuencia negativa, Figura 2.33.

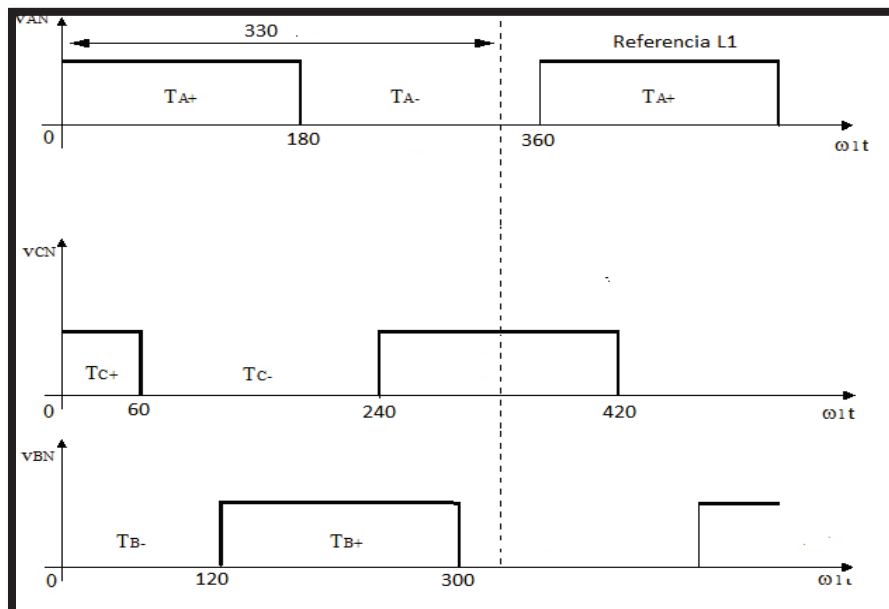


Figura 2.33 Secuencia de fases negativa, formas de onda.
[Elaboración propia]

Cabe recalcar que la línea L2 está asignada al pin RE1 y la línea L3 al pin RE2.

A continuación se observa el código implementado para la identificación de la secuencia de fases del sistema trifásico.

```
do {
  if (Line_OK==0)
  {
    L_1=RE0_bit; // Memoriza el estado lógico de la línea L1
    L_2=RE1_bit; // Memoriza el estado lógico de la línea L2

    if (L_1|L_2)//Comprobar los estados de L1 y L2 ¿están en "0" lógico?
      continue;
    if (!L_1) // Esperar hasta que la línea L1 (referencia) cambie de
      while(!RE0_bit); // estado, 0 a 1(detectar flanco positivo en L1)
    Delay_ms(15); // retardo de 15 milisegundos
    sec_fase=PORTE; // leer el Puerto E y determinar la secuencia de
    switch (sec_fase){ // fases basado en los estados lógicos de L2 y L3
      case 2: mensajes(7,18,6,1);
        break;
      case 4: mensajes(6,18,6,1);
        break;
      default: break;
    }
    Line_OK=1; // Variable para indicar que la secuencia fue identificada
  }
} while(Seguir);
```

En la figura 2.34 se muestra la secuencia negativa identificada por el dispositivo desarrollado.

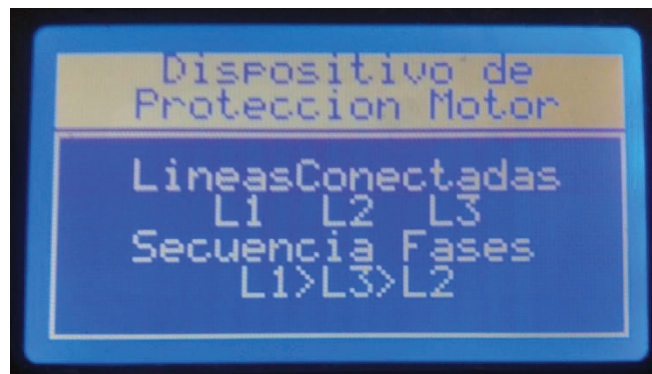


Figura 2.34 Resultado de la identificación de secuencia de fases.
[Elaboración propia]

2.8.5. PROGRAMA PRINCIPAL

Una vez identificada la secuencia de fases, el programa continúa con el bucle del programa principal, el cual se encarga de leer los niveles de voltaje correspondientes a

cada una de las 3 fases haciendo uso del módulo conversor A/D interno del microcontrolador y el bloque de acondicionamiento de señal AC.

2.8.5.1. SUBROUTINA PARA LEER EL MÓDULO A/D Y CALCULAR EL VALOR DE VOLTAJE

Para evitar fluctuaciones en las mediciones de voltaje por ruido u otras circunstancias, se toma una cierta cantidad de muestras de la misma señal analógica para promediarlas y así eliminar lecturas erróneas.

La función retorna el valor de voltaje de las 3 fases y se almacenan en variables de tipo *float* *VL1*, *VL2* y *VL3* en el programa principal; estos datos son de enorme importancia para la rutina de cálculo de desbalance de voltaje y detección de bajos y sobre voltajes.

La subrutina requiere de un solo parámetro, y es el canal a ser evaluado como se observa en el siguiente código.

```
float Volts (char canal){
    char cnt_mues;           // Variable para el número de muestras del ADC
    unsigned int add_adc=0; // Variable para la sumatoria de muestras
                           // y el promedio final
    float resultado=0;      // Variable para almacenar el valor de voltaje

    // Tomar 100 muestras de la misma señal analógica y luego promediarla
    for (cnt_mues=0;cnt_mues<100;cnt_mues++){
        add_adc+=ADC_Read(canal);
        Delay_us(80);       // Retardo mínimo recomendado por el fabricante
    }                       // para la siguiente adquisición

    add_adc=(int)(add_adc/100); // Promedio de muestras

    . . .

    resultado=(float)(add_adc*factor); // Calcular valor de
    resultado*=29;                    // voltaje

    return resultado;                // Devolver valor de voltaje
}
```

El factor de conversión del módulo A/D “factor” se ha declarado como una constante global en la sección de constantes y variables (ver en el ANEXO F).

```
const float factor=0.00489; // Factor de conversion del módulo A/D
```

2.8.5.2. CÁLCULO DEL DESBALANCE DE VOLTAJE

Debido a que las lecturas de voltaje son entre la fase y el neutro (referencia del sistema trifásico), se adopta la definición del desbalance de voltaje según el Instituto de Ingenieros Eléctricos y Electrónicos (IEEE).

Para determinar el porcentaje de desbalance de voltaje, se debe calcular primero el promedio de las tres lecturas de voltaje, el cual, es la referencia para identificar la máxima desviación entre las tres lecturas de voltaje.

Aplicando la ecuación 2.13 descrita en el capítulo 1 del presente proyecto, se puede determinar el porcentaje de desbalance existente en la red trifásica.

$$\% \text{ PVUR} = \frac{\text{Desviación máxima del promedio de } \{V_a, V_b, V_c\}}{\text{Promedio de } \{V_a, V_b, V_c\}} \times 100 \% \quad (2.13)$$

El código que permite realizar el trabajo descrito se muestra a continuación.

```
//----- Rutina para calcular el desbalance de voltaje -----
Promedio=(float) (VL1+VL2+VL3)/3; // Calcular el Promedio
// de los 3 voltajes
Desv_1=fabs (VL1-Promedio); // Calcular las Desviaciones para cada
Desv_2=fabs (VL2-Promedio); // línea con respecto al promedio
Desv_3=fabs (VL3-Promedio);

if ((Desv_1>Desv_2)&&(Desv_1>Desv_3)) // Determinar la máxima
Des_eq=(float) (Desv_1*100)/Promedio;// desviación entre las 3
// líneas y calcular
if ((Desv_2>Desv_1)&&(Desv_2>Desv_3)) // el desbalance
Des_eq=(float) (Desv_2*100)/Promedio;// existente en la red

if ((Desv_3>Desv_1)&&(Desv_3>Desv_2))
Des_eq=(float) (Desv_3*100)/Promedio;
//----- FIN Rutina para calcular el desbalance de voltaje -----
```

2.8.5.3. DETECCIÓN DE FALLAS

Es muy importante garantizar una correcta alimentación de energía eléctrica para el motor. Los conductores, así como todo el sistema de protección, deben garantizar una calidad de energía eléctrica del motor dentro de los parámetros, conforme la norma **IEC60034-1** que contempla la operación de motores trifásicos de inducción de bajo voltaje.

El voltaje podrá variar dentro de un rango de $\pm 10\%$ del valor nominal. Considerando el voltaje nominal de 220 V (línea - línea); cabe recordar que las lecturas del prototipo son entre una línea y el neutro, por lo cual se aplica la ecuación 2.14 para obtener el valor de voltaje de fase aproximado partiendo de los datos del motor.

$$V_{FASE} = \frac{V_{NOMINAL}}{\sqrt{3}} \quad (2.14)$$

Reemplazando valores:

$$V_{FASE} = \frac{220 \text{ V}}{\sqrt{3}}$$

$$V_{FASE} = 127 \text{ V}$$

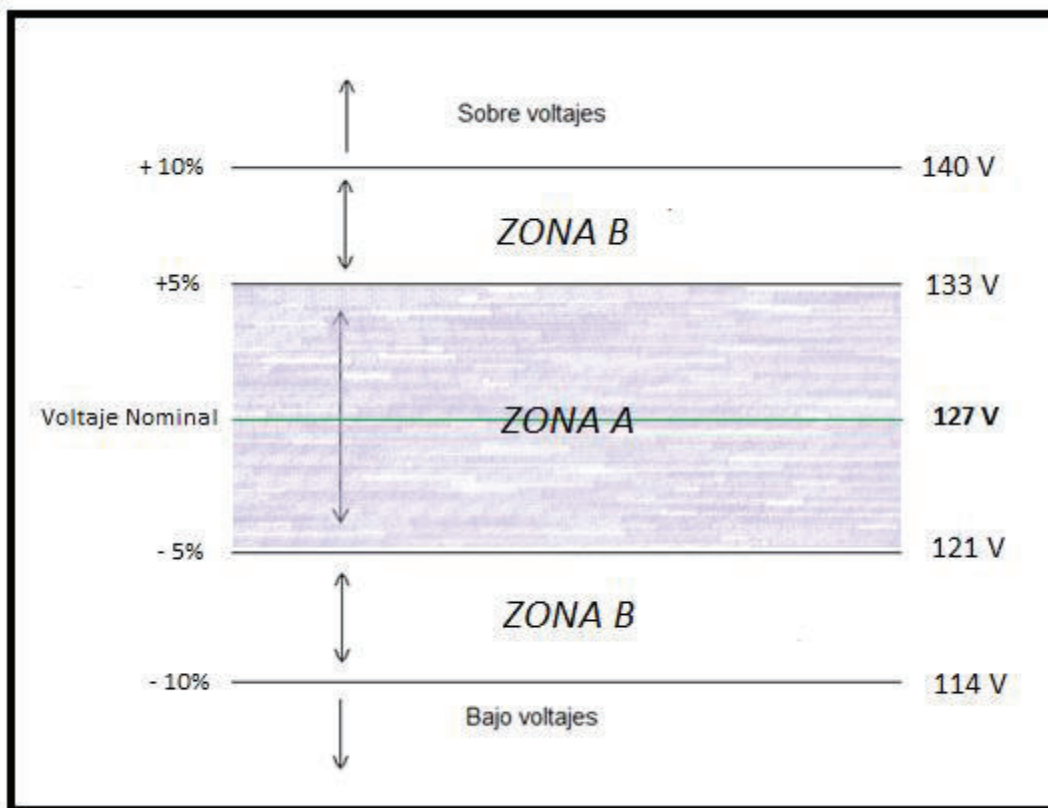


Figura 2.35 Límites de voltajes para motores según norma IEC60034-1. [16]

Detalles de la figura 2.35:

- ZONA A (Operación segura)
- ZONA B (Fuera de rango)

De acuerdo con la norma IEC60034-1, el motor operará satisfactoriamente dentro de la zona A presentando ligeros incrementos de temperatura con respecto al voltaje nominal. Por otro lado, el motor debe ser capaz de desempeñar su función principal en la zona B, aún así no se recomienda su operación prolongada, ya que los incrementos de temperatura muy probablemente serán más altos que aquellas en la zona A.

Si las lecturas de voltajes se encuentran por encima de la zona B (fuera de rango superior), se consideran sobre voltajes y si las lecturas de voltajes se encuentran por debajo de la zona B (fuera de rango inferior) se consideran bajo voltajes.

La subrutina propuesta para la detección de fallas en las lecturas de voltajes es la siguiente:


```

void detector (float voltio){
if((voltio<121&&voltio>=114)|| (voltio>133&&voltio<=140))//Voltaje dentro de
    fuera_rango=1;                                //la zona B
if(voltio>140)    // Sobre voltaje
    sobre_volt=1;

if(voltio<114)    // Bajo voltaje o pérdida de fase
    bajo_volt=1;
}

```

Si los valores de voltaje de las tres, dos o una de las líneas, se encuentra en la zona B (superior o inferior), en la pantalla se desplegará un temporizador previamente cargado con un valor de 60 seg. Si transcurrido dicho tiempo, la anomalía se mantiene, el MCU inhabilitará la señal de respuesta que se envía al módulo relé, para desconectar al motor de la fuente de alimentación; además, se visualizará “PELIGRO” en la pantalla.

Para el caso de sobre voltajes, bajo voltajes o la pérdida de una fase, el procedimiento es similar al anterior con la diferencia que el tiempo a temporizar es mucho más corto que el anterior, de 3 seg.

Si el porcentaje de desbalance de voltaje se encuentra en el rango entre 3 a 6%, se verificará esta falla durante 60 segundos y de mantenerse, el procedimiento a seguir será igual a los dos anteriores. Pero si el desbalance supera el 6% el tiempo de verificación será de 3 seg.

El siguiente código permite ejecutar el trabajo descrito, si no existe fallas de ninguna índole, el MCU activará el módulo relé sin ningún problema.

```

/*===== Detección de fallas =====*/
    detector(VL1);// Ejecutar función "detector" por medio de los
    detector(VL2);// parámetros de valores de voltaje correspondientes a
    detector(VL3);// cada línea
    if (Des_eq>3&&Des_eq<=6) //Desbalance de la red mayor a 3% y <= a 6%
        fuera_rango=1;
    if (Des_eq>6)
        sobre_volt=1;
    if (sobre_volt==1||bajo_volt==1)//Priorizar fallas más
        fuera_rango=0;                                // graves
    if (fuera_rango==1||sobre_volt==1||bajo_volt==1) // ¿Existen fallas?
    {
        if(fuera_rango){ // ¿Existen fallas leves?
            if (ctrl_1==0)
            {
                timetest=60;    // Cargar 60 segundos para verificar
                ctrl_1=1;        // fallas leves o fuera de rango
                ctrl_2=0;
                cnt=0;
            }
        }
        if(sobre_volt||bajo_volt) // ¿Existen sobre o bajo voltajes?
        {
            if (ctrl_2==0) {

```

```

        timetest=3;           // Cargar 3 segundos para verificar
        ctrl_2=1;
        ctrl_1=0;           // sobre y bajo voltajes
        cnt=0;
    }
}
if (TMR1ON_bit==0&&timetest!=0)
    TMR1ON_bit=1;         // Activar TMR1
    if (timetest==0) {
        TMR1ON_bit=0;
        RELE=1;
        ctrl_3=1;
        Glcd_Write_Text("PELIGRO",75,5,1);
    }
ByteToStr(timetest,txt); // Imprimir temporizador
Glcd_Write_Char(*(txt+1),87,6,1);
Glcd_Write_Char(*(txt+2),93,6,1);

fuera_rango=0; // Encerar variables de detección de fallas
sobre_volt=0; // a cero
bajo_volt=0;
}
else { // No existen fallas
    if (ctrl_3==0){
        TMR1ON_bit=0; // Desactivar TMR1
        if (RELE==1) // ¿RELE apagado?
            RELE=0; // Activar RELE
        ctrl_1=0; // Encerar variables de control en cero
        ctrl_2=0;
        cnt=0;
    }
    if (ctrl_3==1){ //Rutina para reconectar //la carga después de //superar la falla
        Glcd_Write_Text("Reconec",81,5,1);
        if (Seguir==0){
            ctrl_3=0;
            RELE=0;
            borrar(81,5);
            borrar(93,5);
            borrar(105,5);
            borrar(111,5);
        }
        borrar(69,4);
        borrar(69,5);
        borrar(69,6);
        borrar(93,6);
    }
}
}
}
/*===== FIN Detección de fallas =====*/

```

2.8.5.4. VISUALIZACIÓN DE PARÁMETROS DE MEDICIÓN

Por último, todas las lecturas de voltaje de cada una de las fases, porcentaje de desbalance y los mensajes necesarios, se imprimen en la pantalla gráfica con el fin de facilitar la detección de fallas para el operador.

El procedimiento para la impresión de los voltajes de cada línea es similar, por esta razón, se desarrolló una subrutina para realizar esta tarea.

```
void imprimir (float voltaje, char pagina){
    char txt2 [17];

    FloatToStr(voltaje,txt2); // Transformar de float a cadena de caracteres

    Glcd_Write_Char(*(txt2),21,pagina,1); // Imprimir las centenas
    Glcd_Write_Char(*(txt2+1),27,pagina,1); // Imprimir las decenas
    Glcd_Write_Char(*(txt2+2),33,pagina,1); // Imprimir las unidades
    Glcd_Write_Char(*(txt2+3),39,pagina,1); // Punto decimal
    Glcd_Write_Char(*(txt2+4),45,pagina,1); // Imprimir el primer decimal
}
```

El siguiente código corresponde a la rutina de impresión escrita en el programa principal

```
//////////Impresión de los voltajes y desbalance en el Glcd//////////

imprimir(VL1,4); //Para el valor de la línea L1
imprimir(VL2,5); //Para el valor de la línea L2
imprimir(VL3,6); //Para el valor de la línea L3

if (Des_eq>=1) //Para el porcentaje de desbalance de la red
{
    FloatToStr(Des_eq,txt1); // Transformar float a cadena de caracteres

    Glcd_Write_Char(*(txt1),84,4,1); // Imprimir unidades
    Glcd_Write_Char(*(txt1+1),90,4,1); // Punto decimal
    Glcd_Write_Char(*(txt1+2),96,4,1); // imprimir el primer decimal

}
else{

    FloatToStr(Des_eq,txt1); // Transformar float a cadena de caracteres

    Glcd_Write_Char('0',84,4,1);
    Glcd_Write_Char('.',90,4,1);
    Glcd_Write_Char(*(txt1),96,4,1); // Imprimir primer decimal

}
```

El resultado obtenido de la ejecución de la rutina del programa principal se puede resumir en la imagen de la figura 2.36

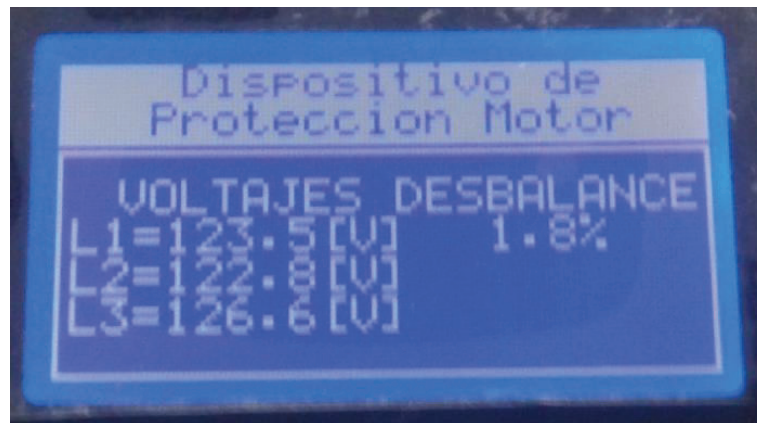


Figura 2.36 Impresión de los valores de voltaje y el desbalance.
[Elaboración propia]

2.9. MENSAJES ESCRITOS EN LA MEMORIA EEPROM

Una de las ventajas de usar un módulo LCD es facilitar la comunicación entre el dispositivo (máquina) y el humano, para ello se debe escribir mensajes en el programa principal para que los despliegue el microcontrolador en el LCD, Esta técnica de escribir mensajes largos directamente en el programa del microcontrolador es muy poco recomendable, por la razón que *Mikro C* usa mucha memoria de datos para realizar el trabajo mencionado y a la vez, reduciendo el espacio disponible en la memoria, que se puede dar un mejor uso para la aplicación desarrollada.

La solución a este inconveniente es recurrir a los recursos físicos del microcontrolador. Los microcontroladores de *Microchip* están equipados con una memoria EEPROM interna que se puede leer y escribir mediante software. El PIC18F4550 tiene a disposición una memoria EEPROM interna de 256 Bytes en la cual podemos grabar varios mensajes para posteriormente leerlos durante la ejecución del programa.

Para grabar los mensajes necesarios, el compilador *Mikro C* en su menú “*Tools*” tiene a disposición la opción *EEPROM Editor*; esta herramienta permite al desarrollador grabar previamente cualquier información en dicha memoria. Al seleccionar la opción *EEPROM Editor*, aparecerá una ventana como se muestra en la siguiente Figura 2.37

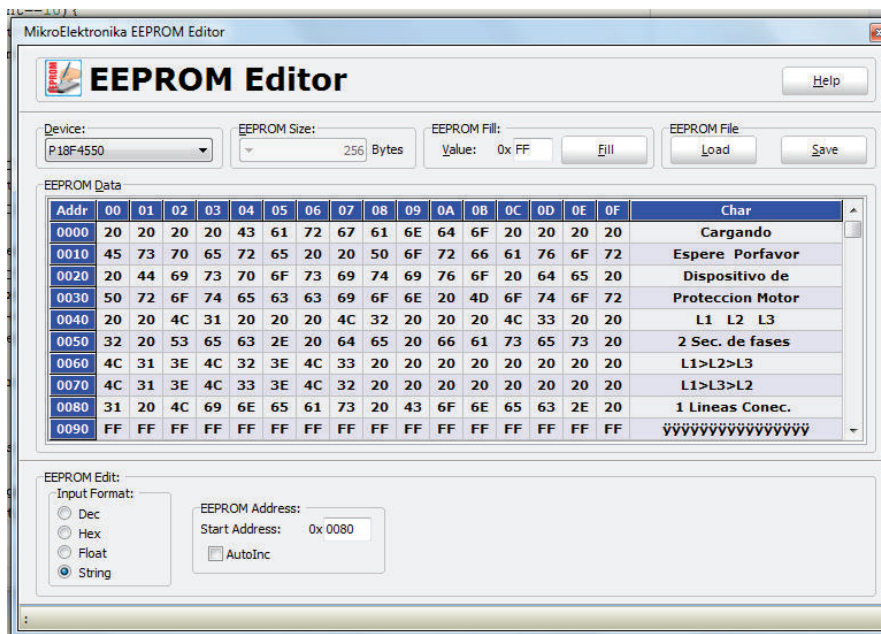


Figura 2.37 Ventana correspondiente a *EEPROM Editor*.
[Elaboración propia]

En la tabla 2.9 se detalla los mensajes escritos en la memoria EEPROM, su dirección y su contenido. Cabe mencionar que cada mensaje tiene un total de 16 caracteres alfanuméricos.

Número de Mensaje	Contenido	Dirección
0	Cargando	0x00 a 0x0F
1	Espere Porfavor	0x10 a 0x1F
2	Dispositivo de	0x20 a 0x2F
3	Protección Motor	0x30 a 0x3F
4	L1 L2 L3	0x40 a 0x4F
5	Secuencia Fases	0x50 a 0x5F
6	L1>L2>L3	0x60 a 0x6F
7	L1>L3>L2	0x70 a 0x7F
8	LíneasConectadas	0x80 a 0x8F
9	PROYECTO	0x90 a 0x9F
10	INTEGRADOR	0xA0 a 0xAF
11	T E S I S	0xB0 a 0xBF
12	Miguel Barros	0xC0 a 0xCF
13	Sergio Tipán	0xD0 a 0xDF
14	ELECTROMECAÁNICA	0xE0 a 0xEF
15	-----sin utilizar-----	0xF0 a 0xFF

Tabla 2.9 Mensajes escritos en la memoria EEPROM.

Para la lectura de los mensajes grabados en la memoria EEPROM se desarrolló una subrutina, capaz de determinar la dirección del inicio del mensaje a ser leído a partir del número de dicho mensaje, para luego almacenarlo en una cadena de caracteres para su posterior impresión en la pantalla gráfica.

```
void mensajes (int num_men,char num_pix,char num_pag, char color){
    char txt [16];          // Array para almacenar el mensajes a leer
    int j;
    int k;

    k=0;
    num_men=num_men*0x10; // Operación para determinar la dirección del
                          // mensaje seleccionado

    // Leer mensaje de 16 caracteres y guardar en el array "txt"
    for (j=num_men;j<=num_men+0x0F&&j!=0xFF;j++){

        txt[k]=EEPROM_Read(j);
        k++;
        Delay_ms(20);
    }

    txt[16]='\0';          // Caracter nulo para indicar el fin del array
    Glcd_Write_Text(txt,num_pix,num_pag,color);//Presentar el mensaje en GLCD
}
```

El inconveniente que se presentó durante el desarrollo de esta subrutina fue omitir el caracter nulo en el final de la cadena de caracteres, el cual produjo múltiples errores en la impresión de los mensajes. En lenguaje C/C++ el caracter nulo '\0' indica el final de un arreglo de caracteres.

En este capítulo se ha tratado sobre el desarrollo tanto de hardware y software del dispositivo de protección, para visualizar de manera detallada los planos electrónicos de los diferentes circuitos que conforman el dispositivo y el código fuente para el microcontrolador (véase los anexos E y F).

CAPÍTULO 3

IMPLEMENTACIÓN DEL DISPOSITIVO DE PROTECCIÓN

En este capítulo se detallan los pasos efectuados para la implementación del dispositivo de protección. Enfocándose en el diseño de los circuitos impresos *PCB* (*Printed Circuit Board*), después de haber verificado el funcionamiento de cada circuito; primero en el simulador de *PROTEUS* y, posteriormente las respectivas pruebas en un protoboard.

El diseño de cada circuito impreso se realizó usando *PROTEUS*, entorno de desarrollo que integra diversas herramientas de software que permiten la captura de esquemáticos (módulo *ISIS*), fabricación de circuitos impresos (módulo *ARES*) y un entorno de simulación avanzado (módulo *VSM*).

3.1. DISEÑO DE PISTAS EN *ARES*

Para la elaboración de los *PCB* de los respectivos circuitos, se usó la herramienta *ARES* de Proteus, que facilita la transferencia de los componentes desde *ISIS* a *ARES* y posee dos características importantes que brinda esta herramienta, los cuales son: auto-posicionamiento de elementos y auto-ruteo. En la figura 3.1 se muestra el ambiente de desarrollo de *ARES*.

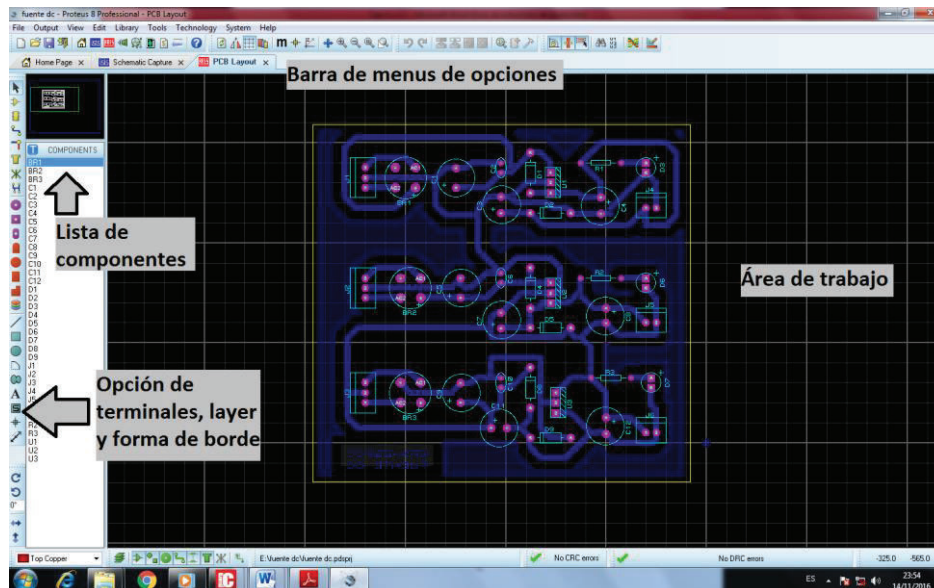


Figura 3.1 Vista del entorno de desarrollo *ARES*.
[Elaboración propia]

Debido a que el dispositivo de protección se divide en varios bloques y además el número de componentes es considerable. Se optó por fabricar varios circuitos impresos con el objeto de reducir el tamaño del dispositivo, separar la parte analógica de la digital y, además, aislar la parte de potencia que es alimentada con 220 voltios AC.

Para ello, se elaboró un circuito impreso para la fuente de DC, tres circuitos acondicionadores de señales para cada una de las fases, un circuito impreso para el módulo relé y por último la tarjeta principal la cual contiene al microcontrolador.

A continuación, se detalla los requerimientos necesarios para poder llevar a cabo el ruteado de una placa PCB a partir del esquemático generado en ISIS:

Antes de comenzar a colocar los componentes en el área de trabajo de *ARES*, es necesario crear los bordes de la PCB. Entonces, hay que definir, delimitar la forma y el tamaño de la placa que se va a utilizar, usando la herramienta “*2D Graphics Box Mode*” y “*Board Edge*” como capa sobre la que se trabajará. En la figura 3.2 se delimita el tamaño y forma con el rectángulo de color amarillo.

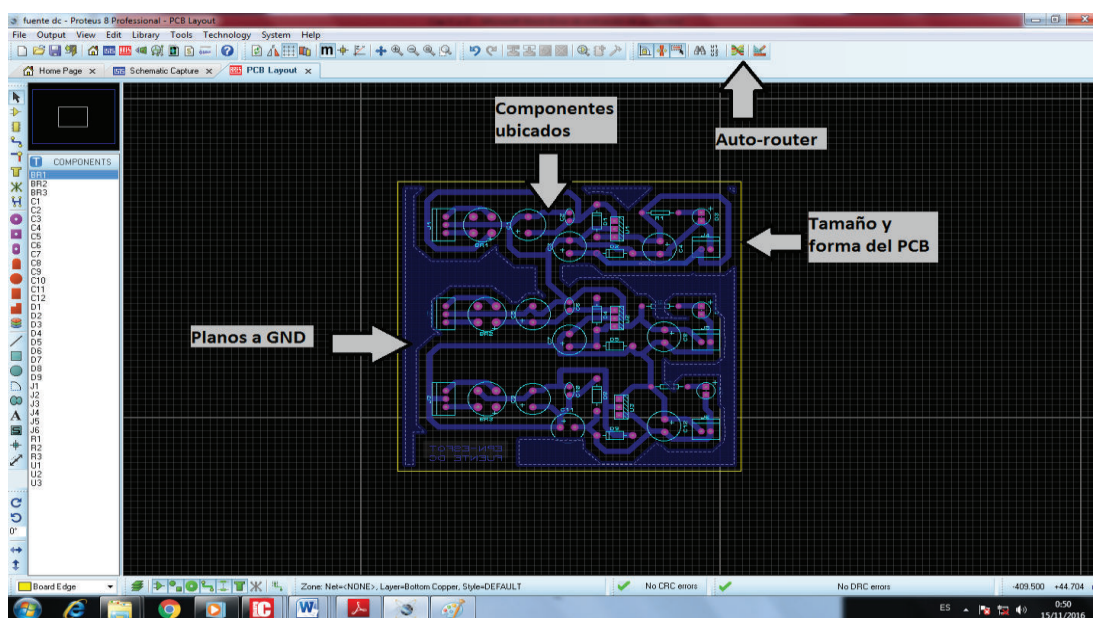


Figura 3.2 Creación del primer PCB.
[Elaboración propia]

Ya definidos los bordes, procedemos a la ubicación de cada componente dentro del área establecida. Se toma en cuenta la trayectoria que puedan tomar las pistas, el acceso a los elementos de ajuste como potenciómetros de precisión, en el caso de los reguladores de voltaje y componentes que necesiten de disipadores de calor; ubicarlos de forma que éste no interfiera ni bloquee a otros componentes. Para los componentes como terminales para la conexión entre placas o alimentación de las mismas deben localizarse en los bordes de la placa.

El “*Auto Router*” es una herramienta avanzada para el trazado automático de las pistas, basado en rejilla. Su potencia, rapidez y flexibilidad, permite generar todas las rutas de la placa de los circuitos impresos con el grosor y separación seleccionados.

El grosor de la pista es fundamental para el buen desempeño del circuito impreso fabricado, para ello se utilizó el grosor T30 para las pistas de las señales analógicas y digitales y T80 para las pistas de la fuente de DC. Una separación entre pistas T30 (0.762 mm) es recomendado en tutoriales acerca de elaboración de PCB's. [13]

Por último, se crea un plano a masa o GND usando la herramienta "Zone Mode", que ayuda a establecer un plano a masa general entre los puntos de GND de cada componente y el resto de cobre de la placa, con el fin de reducir las interferencias y el ruido eléctrico como se observa en la Figura 3.2.

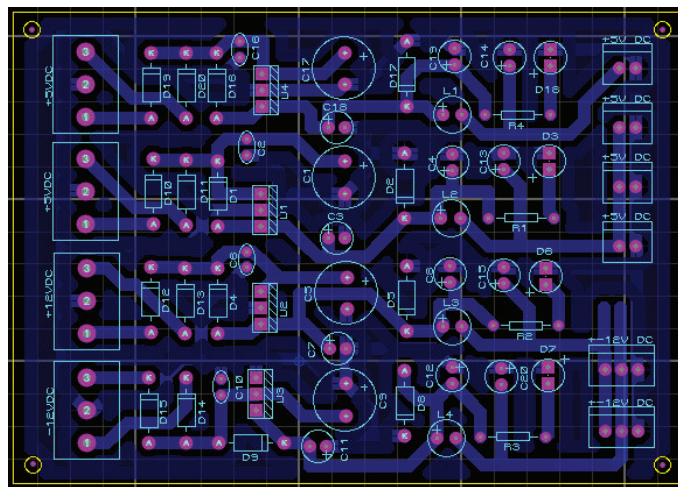
3.2. EXPORTACIÓN E IMPRESIÓN DEL CIRCUITO

Proteus ofrece la facilidad de exportar a PDF o a imagen BITMAP cada cara o plantilla para su posterior impresión y continuar con la transferencia o serigrafía sobre la placa del circuito impreso.

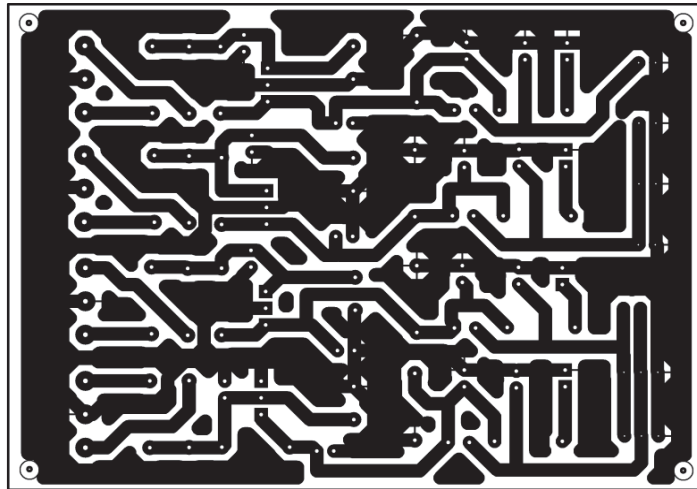
Para ello, se selecciona la opción "Export Graphics => Export Adobe PDF File o Export to Bitmap" desde el menú "Output". En la ventana que aparecerá habrá que seleccionar las vistas que se requiera exportar. [13]

A continuación, se presenta los diagramas finales de cada PCB (Figuras 3.3, 3.4 y 3.5) que conformarán el dispositivo de protección, los gráficos más importantes para la elaboración del circuito impreso: Fotolitos de las pistas de la cara inferior y las plantillas de la capa superior que serán transferidos a la placa.

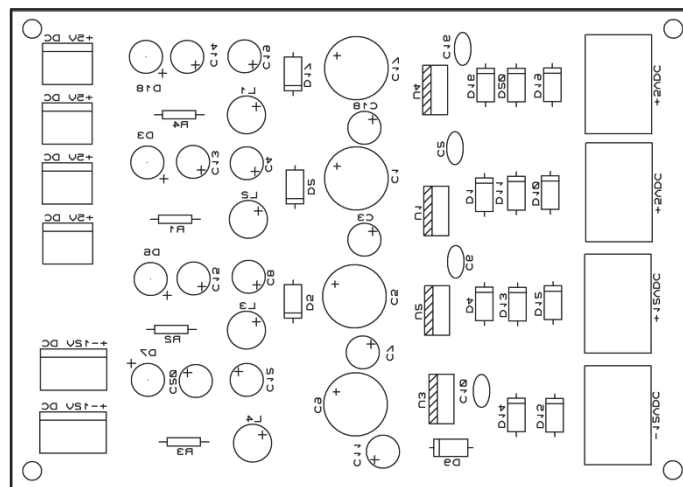
Figura 3.3 PCB de la fuente de alimentación DC.
[Elaboración propia]



a) Ruteo de las pistas en ARES.

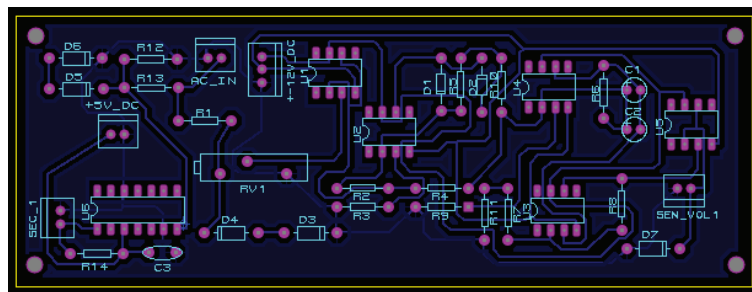


b) Fotolito de las pistas de la cara inferior.

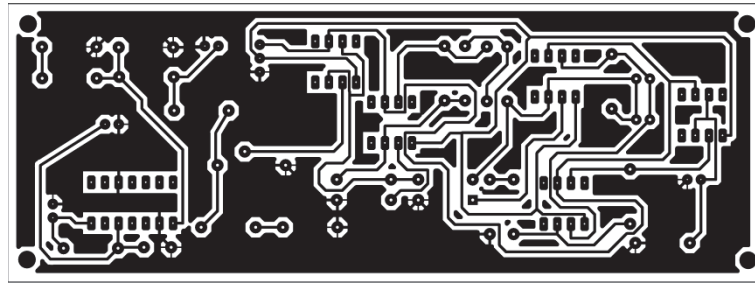


c) Plantilla de la cara superior, ubicación de componentes.

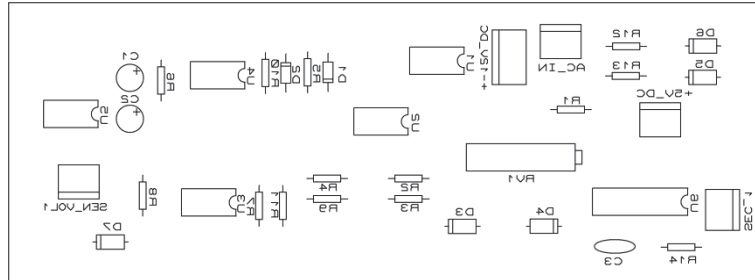
Figura 3.4 PCB del bloque de acondicionamiento de señal AC.
[Elaboración propia]



a) Ruteo de las pistas en ARES.

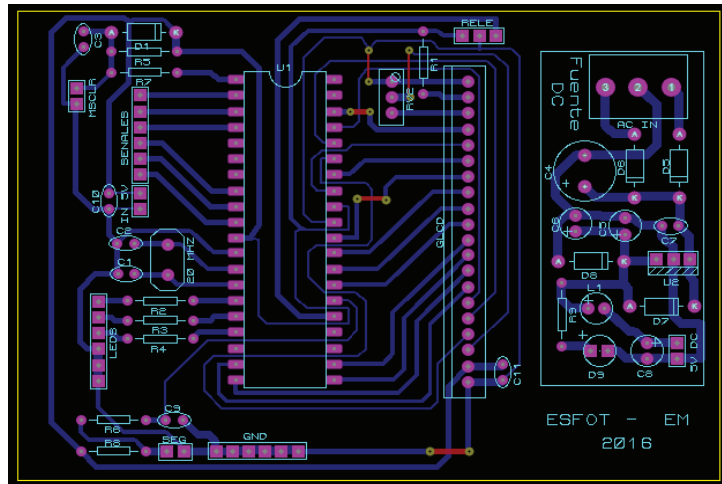


b) Fotolito de las pistas de la cara inferior.

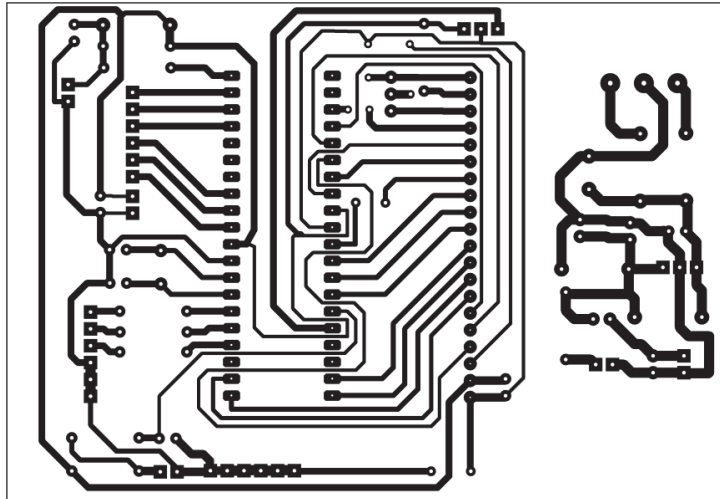


c) Plantilla de la cara superior, ubicación de componentes.

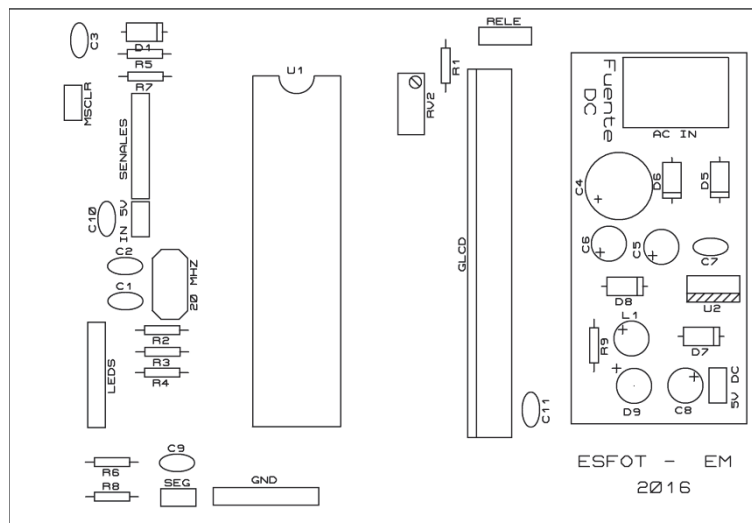
Figura 3.5 PCB de la tarjeta principal, contiene el PIC18F4550.
[Elaboración propia]



a) Ruteo de las pistas en ARES.



b) Fotolito de las pistas de la cara inferior.



c) Plantilla de la cara superior, ubicación de componentes.

3.3. FABRICACIÓN DEL CIRCUITO IMPRESO

Para la elaboración de la placa física (circuitos impresos), se tiene el diseño de los circuitos realizado anteriormente e impresos en papel transferible para el caso de las pistas; y en el caso de la plantilla de la capa superior que contiene la ubicación de los componentes, en una lámina de acetato.

Los pasos a seguir para transferir las pistas sobre la baquelita son los siguientes:

1. Cortar la baquelita según las dimensiones especificadas en el diseño esquemático
2. Emparejar la superficie del cobre con una lija fina 200 de agua para obtener una superficie lisa y posteriormente limpiarla con alcohol para eliminar impurezas y lograr una buena transferencia de las pistas.

3. Asegurar con cinta adhesiva el lado del papel transferible que contiene la impresión de las pistas. Con la ayuda de una plancha y el calor que ésta genera, se transferirá las pistas hacia el cobre mediante el proceso de planchado que dura entre 10 a 15 minutos.
4. Retirar el papel transferible de la baquelita, para ello se debe sumergir la baquelita junto con el papel en agua durante varios minutos; si se da el caso de haber perdido alguna pista, se debe utilizar un marcador para retocar las pistas perdidas.
5. En un recipiente con agua caliente, suficiente para cubrir la placa, sumergir la baquelita y luego agregar el ácido (cloruro-férrico); mientras el ácido actúa se tiene que agitar el recipiente, esto ayuda a acelerar el proceso. Una vez que el ácido haya actuado en el cobre excedente, se debe retirar con mucho cuidado la placa y enjuagarla con agua limpia.

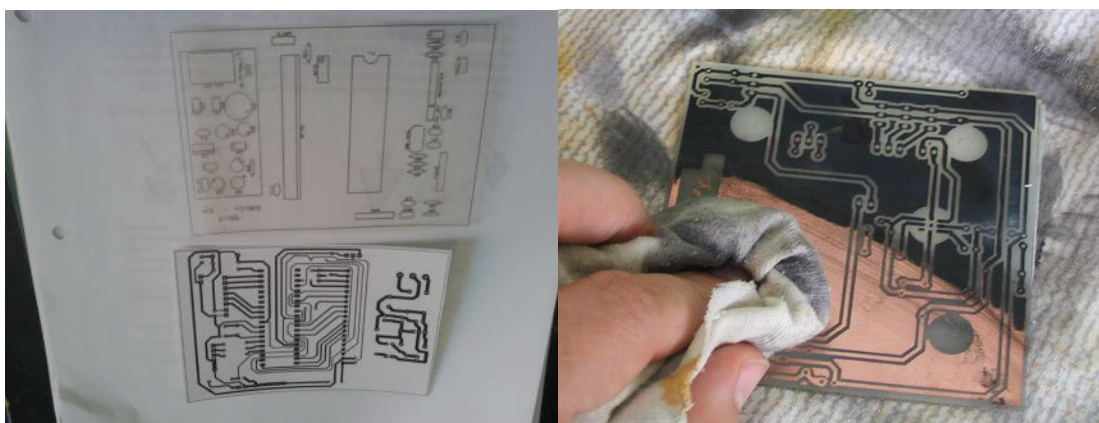


Figura 3.6 Transferencia de las pistas en la baquelita.
[Elaboración propia]

6. Para remover la película de tinta sobre las pistas, se debe limpiar con thinner y algodón, seguidamente lo secamos y procedemos a realizar las perforaciones necesarias (ver figura 3.6).
7. Con la ayuda de la plancha y la plantilla de acetato que contiene la impresión de la capa superior (ubicación de componentes), proceder a la transferencia.
8. Ubicación y soldadura blanda de cada componente (ver figura 3.7).
9. Para finalizar, con una esponja abrasiva de acero pulir suavemente y dar brillo a las pistas, limpiarlo y dar una capa de esmalte transparente para evitar la oxidación de las pistas.

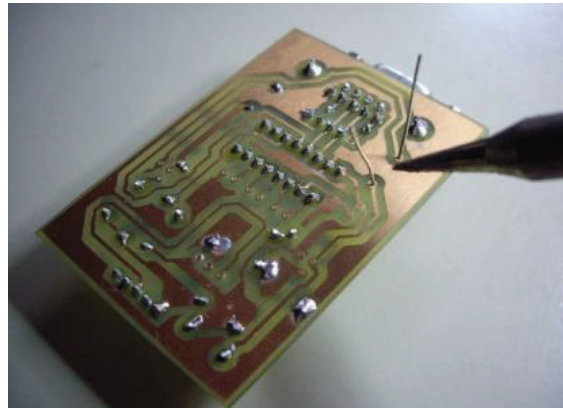


Figura 3.7 Ubicación y soldadura de los componentes.
[Elaboración propia]

Para facilitar la extracción o colocación de algún circuito integrado o el mismo microcontrolador, se optó por usar zócalos con el objeto de evitar el desgaste en los pines de dichos elementos. A continuación, en la figura 3.8 se muestra los circuitos impresos terminados.

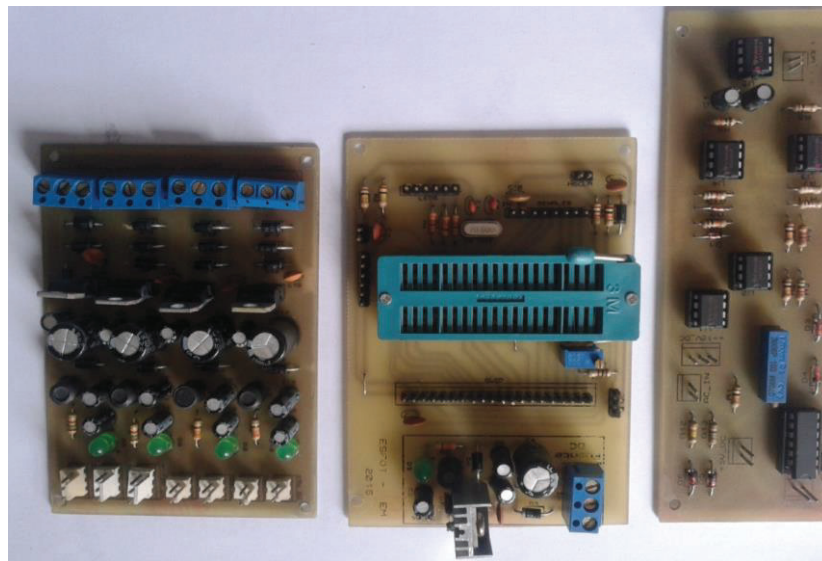


Figura3.8 Vista de los circuitos impresos terminados.
[Elaboración propia]

3.4. MONTAJE DEL DISPOSITIVO DE PROTECCIÓN

Para el montaje del dispositivo y cada uno de los componentes que lo conforman, se usó una caja metálica con tapa de dimensiones (25 cm x 25 cm x 13 cm).

En la cara superior se realizaron perforaciones y cortes para colocar el GLCD, led's indicadores para cada línea, y pulsadores para control del dispositivo (reset y seguir). En las caras laterales de la caja se realizaron perforaciones para 4 jacks, donde se

conectará la alimentación trifásica: 3 líneas y neutro a sensor, 2 jacks de alimentación 220 V para el control del contactor desde el módulo relé, 2 jacks para el control de encendido y apagado del contactor. Además se ha considerado perforaciones adicionales para ubicación del switch interruptor y del conector tripolar. En la Figura 3.9 se observa la distribución antes mencionada.



Figura 3.9 Vista exterior del dispositivo.
[Elaboración propia]

En el interior de la caja se ubicó las 3 tarjetas acondicionadoras de señales y transformadores de potencial para cada línea, el transformador para la tarjeta de la fuente de alimentación DC, la tarjeta principal que contiene el PIC y el módulo relé. Todos ubicados en un doble fondo metálico para fácil extracción de cada componente como se muestra en la Figura 3.10

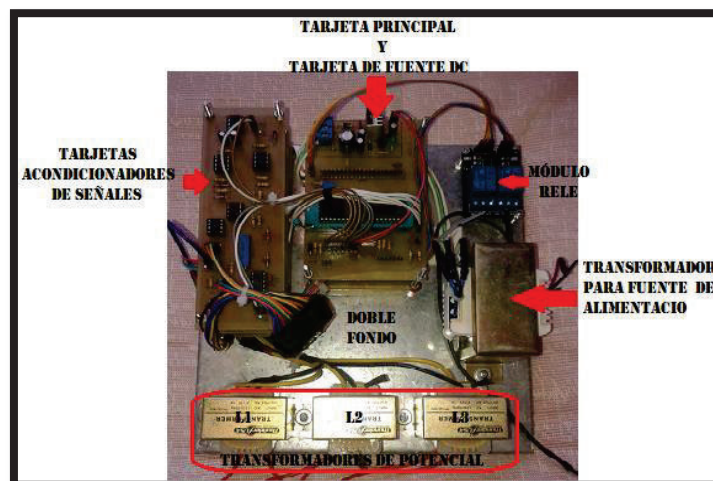


Figura 3.10 Ubicación interna de los componentes del dispositivo.
[Elaboración propia]

Una vez ubicados todos los componentes y realizadas las respectivas conexiones entre ellos, se procedió a verificar el encendido del dispositivo y que su funcionamiento sea el esperado. Esto se comprobó con el encendido de los led's indicadores en la fuente de alimentación, del switch interruptor, el correcto encendido del GLCD y mediante mediciones de voltaje en puntos específicos como: entradas a los transformadores, en salida de señales en las tarjetas acondicionadoras, en fuente independiente para alimentación del PIC y GLCD.

3.5. RECURSOS UTILIZADOS

A continuación, se presenta la información detallada de la cantidad de elementos y componentes utilizados en la construcción del dispositivo, con su respectivo precio y el costo total invertido en la fabricación del dispositivo.

Elementos / Componentes	Cantidad	P.Unitario	P.Total (USD)
PIC 18F4550	1	7,50	7,50
GLCD 128x64 (KS0108)	1	10,00	10,00
Módulo Relé 5V 2 canales	1	4,50	4,50
Contactador Schneider Electric LC1D12	1	36,50	36,50
Transformadores			
(110/220 Vac a 12Vac) 3A	1	10,40	10,40
(120 Vac a 12Vac) 1A	3	7,90	23,70
Resistencias			
10K	20	0,02	0,40
1k	4	0,02	0,08
5.1K	3	0,02	0,06
1.5K	9	0,02	0,18
3.3K	3	0,02	0,06
5K	6	0,02	0,12
330K	3	0,02	0,06
1.5M	3	0,02	0,06
330 Ω	6	0,02	0,12
100 Ω	1	0,02	0,02
100K	1	0,02	0,02
Diodos Rectificadores			
1N4007	20	0,08	1,60
1N4001	1	0,08	0,08
Diodos Zener			
1N4744A	6	0,28	1,68
1N4733A	9	0,28	2,52

Potenciómetro Precisión				
	10K	3	1,35	4,05
	5K	1	0,7	0,70
Reguladores				
	7805	3	0,57	1,71
	7812	1	0,57	0,57
	7912	1	0,57	0,57
Capacitores Electrolíticos				
	1000 uf / 35V	5	0,40	2,00
	1 uf / 25v	5	0,09	0,45
	4.7 uf / 25v	5	0,09	0,45
	2.2 uf / 25v	6	0,09	0,54
	22 uf / 25v	3	0,09	0,27
	47 uf / 25v	3	0,09	0,27
Capacitores Cerámicos				
	# 104	13	0,08	1,04
	# 15	2	0,08	0,16
Operacionales				
	Comparador LM339	3	0,85	2,55
	Amplificador LM741	15	0,45	6,75
	Bobina 1 mH	5	0,60	3,00
	Cristal 20 Mhz	1	0,50	0,50
	Led 5 mm Verde	8	0,08	0,64
	Terminal Hembra	20	0,10	2,00
	Terminal Macho	20	0,10	2,00
	Conector header 3 pin	5	0,25	1,25
	Conector header 2 pin	16	0,35	5,60
	C. header hembra 40 pin	1	0,55	0,55
	C. Header macho 40 pin	2	0,55	1,10
	Conector jack banana	12	0,40	4,80
	Conector tripolar macho	1	0,60	0,60
	Borneras 3 Puntos	5	0,35	1,75
	Termoencogible 6 mm	1	0,60	0,60
	Switch	1	0,35	0,35
	Espárrago 3 mm x 1 m	1	1,00	1,00
	Tuercas 3 mm	40	0,04	1,60
	Portafusible	1	0,60	0,60

Fusible 1A	1	0,30	0,30
Cable arduino 20 cm x 40 u	1	8,00	8,00
Pulsador NO	2	1,20	2,40
Baquelita 30 cm x 20 cm	1	3,50	3,50
Papel Transferible	4	1,00	4,00
Funda ácido 20 gr	8	0,31	2,48
Broca 0,8 mm	2	0,50	1,00
Broca 1 mm	1	0,60	0,60
Zócalo socket dip 8	15	0,10	1,50
Zócalo socket dip 14	3	0,12	0,36
Zócalo zip universal	1	2,15	2,15
Estaño metros	3	0,50	1,50
Spray transparente	1	2,99	2,99
Lija # 150 agua	1	0,80	0,80
Lustre acero fino	1	0,60	0,60
Etiquetas Adhesivas	1	1,00	1,00
TOTAL 1			183,34

RECURSOS HUMANOS	VALOR
Desarrollo del programa para el MCU (1 mes y 2 semanas x 6 horas diarias)	405,00
Desarrollo del dispositivo (4 meses x 6 horas diarias)	
• Acondicionamiento de señales	1080,00
• Fabricación de PCB's	
• Implementación y pruebas	
TOTAL 2 1485,00	

TOTAL, INVERSIÓN TOTAL 1 + TOTAL 2	USD 1668,34
---	--------------------

La inversión total en la construcción del dispositivo que contempla elementos, componentes y recurso humano es de **1668,34** dólares.

CAPÍTULO 4

PRUEBAS DE FUNCIONAMIENTO Y ANÁLISIS DE RESULTADOS

4.1. INTRODUCCIÓN

Para comprobar el funcionamiento del dispositivo de protección desarrollado se realizaron varias pruebas; con el fin de verificar su funcionalidad y el cumplimiento de los objetivos establecidos en el presente proyecto.

Las pruebas realizadas son:

- Identificación de la secuencia de fases.
- Pérdida de fase.
- Desbalance de voltaje superior al 3%.
- Sobre y bajo voltajes.

Se utilizó un variac trifásico en conexión Y-Y con neutro para obtener sobre y bajo voltajes, y además simular un desbalance de voltaje superior al 3%.

Para la pérdida de fase se utilizó un interruptor tipo cuchilla unipolar que facilita la interrupción de energía en una línea cualquiera de alimentación hacia la carga.

En la Figura 4.1 se muestra las conexiones necesarias del dispositivo para las pruebas; y en la figura 4.2 se observa su implementación.

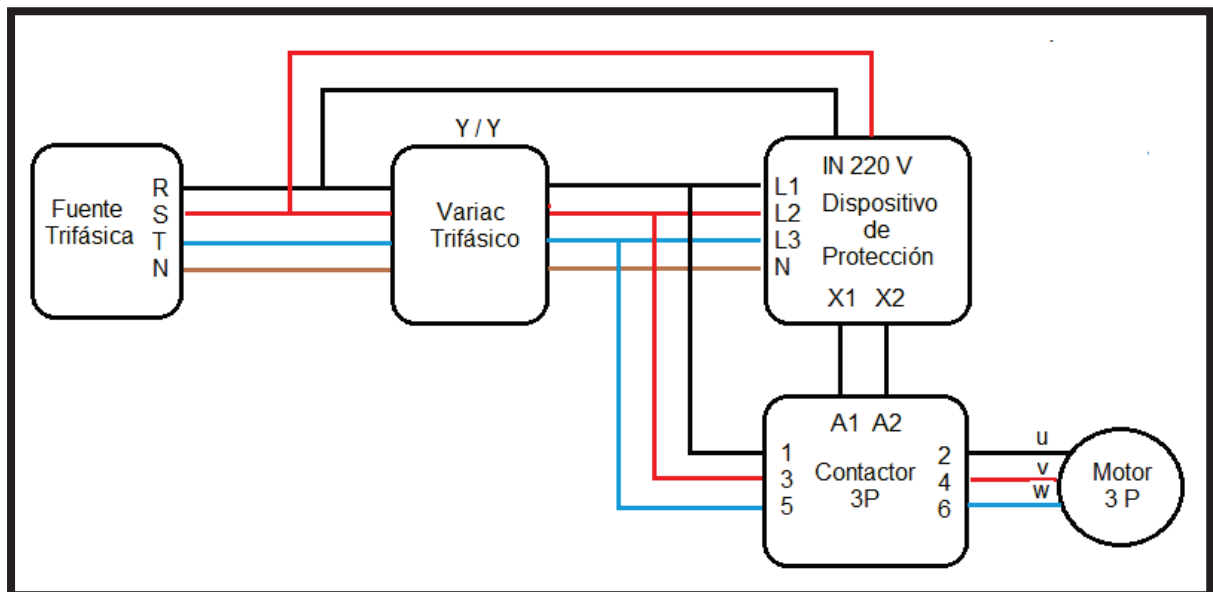


Figura 4.1 Conexión del dispositivo para las pruebas.
[Elaboración propia]



Figura 4.2 Recursos, equipos e instrumentos para las pruebas.
[Elaboración propia]

4.2. CONDICIONES INICIALES PARA LAS PRUEBAS

Las pruebas de funcionamiento del dispositivo de protección se realizaron en el Laboratorio de Máquinas Eléctricas ubicado en el edificio de la Facultad de Ingeniería Eléctrica de la Escuela Politécnica Nacional.

Los equipos, recursos e instalaciones necesarias para realizar las pruebas mencionadas son las siguientes:

- Fuente trifásica de 4 hilos.
- Variac trifásico.
- Motor trifásico de jaula de ardilla.
- Pinza amperimétrica.
- Multímetro digital.
- Interruptor tipo cuchilla unipolar.
- Secuencímetro.
- Contactor tripolar
- Mesa de trabajo

Los datos de placa del motor utilizado para las distintas pruebas se muestran en la Tabla 4.1.

Voltaje [V]: 220 / 380	Factor de potencia: 0,75
Corriente [A]: 1.4 / 0.8	Rpm: 1740
HP: 1/2	Frecuencia [Hz]: 60

Tabla 4.1 Placa de datos del motor.

Para verificar que las lecturas de voltaje mostradas por el dispositivo desarrollado son confiables, se procedió a analizar y comparar las lecturas con el instrumento patrón (Pinza amperimétrica Fluke 337). En la Tabla 4.2 se muestra los valores de voltaje obtenidos por el dispositivo desarrollado y el instrumento patrón de una de las fases. Para determinar si las lecturas de voltajes se encuentran en un rango de error tolerable por debajo del 2%, se analizó las lecturas de una de las líneas con la ecuación 4.1

$$\% \text{ Error} = \frac{|L_p - L_{bp}|}{L_p} \times 100 \quad (4.1)$$

Donde:

L_p : Lectura mostrada por el instrumento patrón.

L_{bp} : Lectura mostrada por el dispositivo bajo prueba.

Número de lectura	Instrumento Patrón Voltios [V]	Dispositivo Voltios [V]	Porcentaje de Error (%)
1	0	0	0
2	5,7	5,66	0,70
3	10,7	10,75	0,47
4	20	20,66	3,30
5	31	31,3	0,97
6	40	39,99	0,02
7	50	49,95	0,10
8	60	60,14	0,23
9	70	69,62	0,54
10	80	79,67	0,41
11	90	89,01	1,10
12	100	99,34	0,66
13	110	109,3	0,64
14	121	120,2	0,66
Error Promedio			0,70

Tabla 4.2 Análisis de voltajes entre el instrumento patrón y el dispositivo.

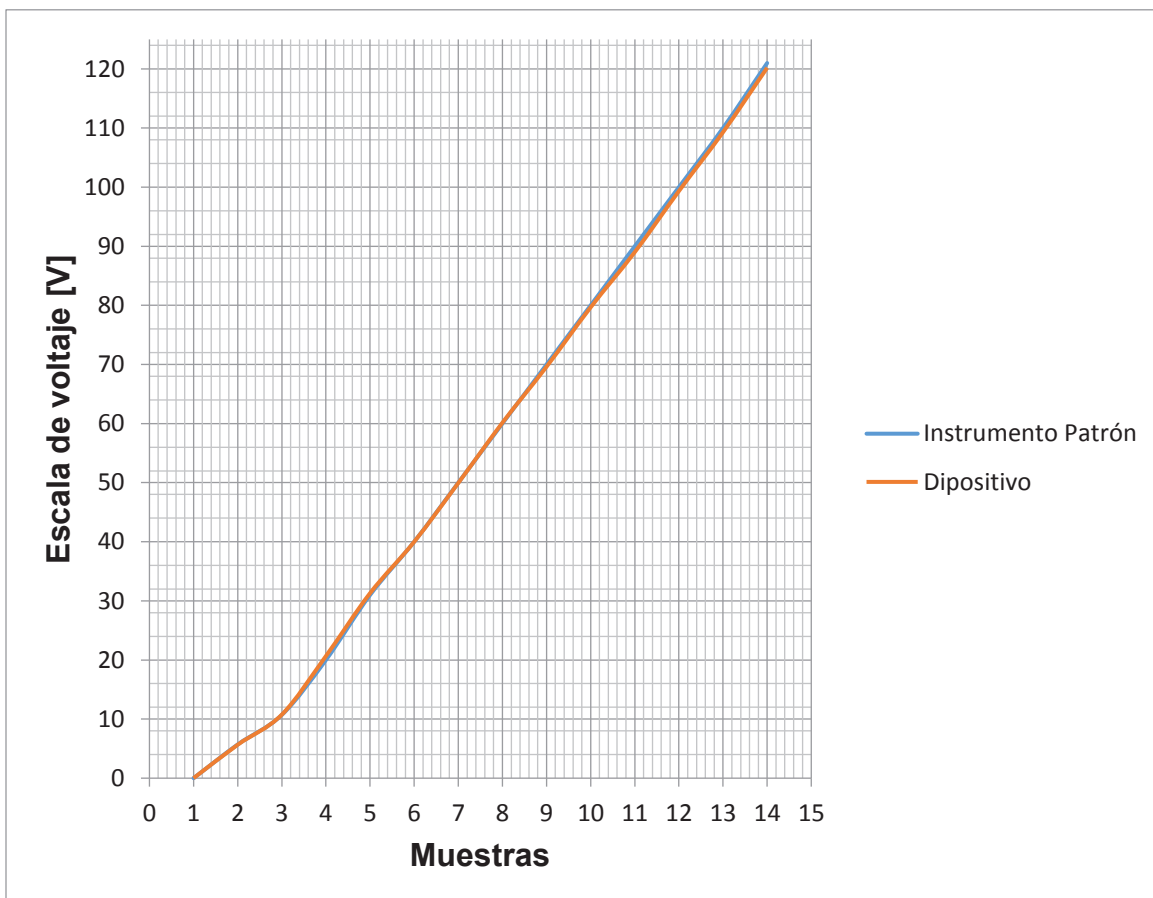


Figura 4.3. Curva de respuesta de las lecturas de voltaje.
[Elaboración propia]

El porcentaje de error obtenido a partir de las lecturas tomadas, tanto del dispositivo desarrollado como del instrumento patrón, están por debajo del 1.5%, por lo cual las lecturas de voltaje son confiables para las siguientes pruebas de bajo voltajes, sobre voltajes, y desbalance de voltaje.

4.3. IDENTIFICACIÓN DE LA SECUENCIA DE FASES

Para evaluar la correcta identificación de la secuencia de fases se utilizó como instrumento patrón un secuencímetro de marca *TECPEL* modelo *PHS 863*. En la figura 4.4 se observa el diagrama eléctrico para realizar esta prueba.

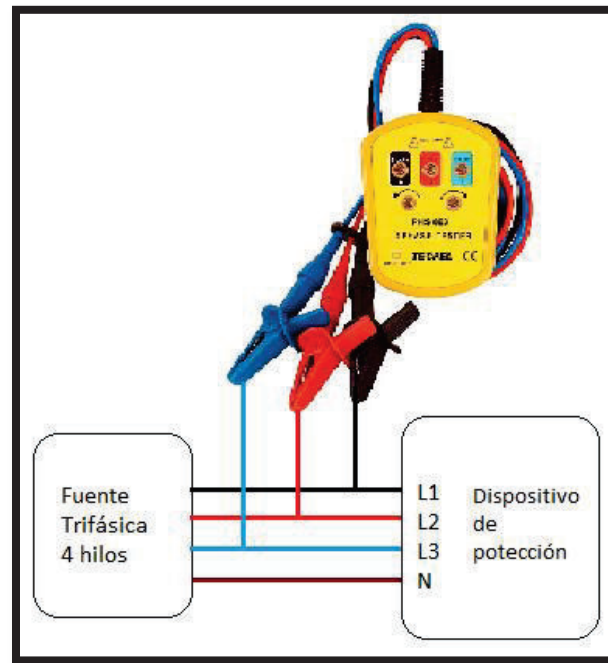


Figura 4.4 Conexiones para pruebas de secuencia de fases.
[Elaboración propia]

Una vez realizadas las conexiones a los terminales identificados tanto del dispositivo desarrollado y el secuencímetro, se procedió a identificar la secuencia de fases.

En la figura 4.5 se muestra el resultado de la prueba de identificación de la secuencia de fases. El secuencímetro muestra la secuencia positiva con el giro horario, mientras que el dispositivo desarrollado indica la secuencia positiva con el orden temporal en que los voltajes pasan por sus respectivos valores máximos, para la secuencia positiva $L1 > L2 > L3$.



Figura 4.5 Resultado de la identificación de la secuencia positiva.
[Elaboración propia]

4.4. PÉRDIDA DE FASE

La pérdida de una fase durante la operación del motor trifásico constituyó una de las fallas más graves; experimentalmente se verificó el incremento de corriente en las dos fases restantes, el exceso de vibración y el aumento de temperatura del motor.

Para imitar la pérdida de fase se usó un interruptor de tipo cuchilla en una de las líneas de alimentación hacia el motor, como se muestra en la Figura 4.6

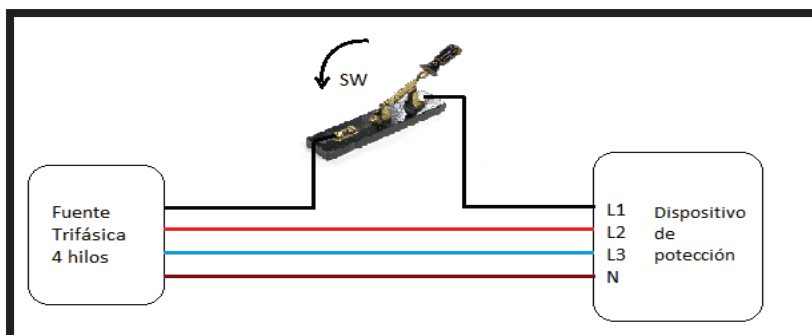


Figura 4.6 Diagrama de bloques para imitar pérdida de fase.
[Elaboración propia]

Ante una pérdida de fase, en la pantalla se pudo observar la lectura de la fase interrumpida (línea L1) con un valor correspondiente a 0 voltios, véase la Figura 4.7



Figura 4.7 Lectura de la pérdida de fase.
[Elaboración propia]

Transcurrido el tiempo de verificación de la falla (3 segundos), el MCU actúa inhabilitando la señal de encendido del módulo relé, con el propósito de desenergizar la bobina del contactor tripolar. Como resultado, se da la apertura de los contactos principales del contactor para desconectar la carga.

4.5. DESBALANCE DE VOLTAJE

Para esta prueba, con la ayuda del variac se manipuló una de las fases de alimentación del motor (L3) para simular un desbalance de voltaje superior al 3%, obteniendo como resultado las lecturas de la Figura 4.8

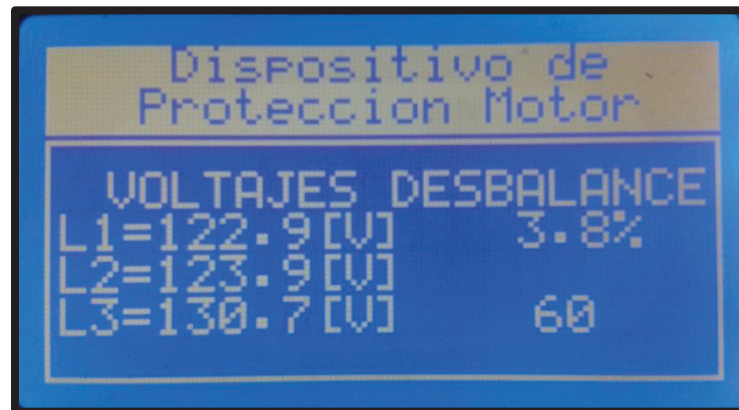


Figura 4.8 Lectura del desbalance de voltaje.
[Elaboración propia]

El desbalance que se observa en la pantalla, es de 3.8% . A continuación se verifica dicho valor.

Lecturas de voltaje:

- L1=122.9 V
- L2=123.9 V
- L3=130.7 V
- Promedio de lecturas: 125.8 V
- La máxima desviación corresponde a la L3 y su valor es de 4.9 V

Para calcular el porcentaje de desbalance de voltaje aplicamos la ecuación 1.2 analizada en el capítulo 1.

$$\text{Desbalance} = \frac{4.9 \text{ V}}{125.8 \text{ V}} \times 100 \%$$

$$\text{Desbalance} = 3.8 \%$$

Con el resultado anterior se demuestra que la rutina desarrollada para obtener el valor de porcentaje de desbalance de voltaje es correcta.

4.6. SOBRE Y BAJO VOLTAJES

De acuerdo a los límites establecidos en la detección de fallas según la norma IEC60034-1, se considera una falla por sobre voltaje cuando cualquiera de las 3 líneas

presentan lecturas superiores al valor de 140 V (+ 10% del voltaje nominal), y para bajo voltajes lecturas inferiores de 114 V (-10 % del voltaje nominal). Si los valores de voltaje están por fuera de dichos límites, el prototipo verificará la falla por 3 segundos y de mantenerse la anomalía el MCU desconectará el motor de la fuente de alimentación. Cabe mencionar que si se identifica un sobre voltaje en la pantalla se desplegará las siglas “SV”, para bajo voltajes se desplegará las siglas “BV” y si los voltajes están dentro de la zona B (superior o inferior) se visualizara las siglas “FR” que corresponden a voltajes fuera de rango de operación segura como se observa en el diagrama de la figura 4.9

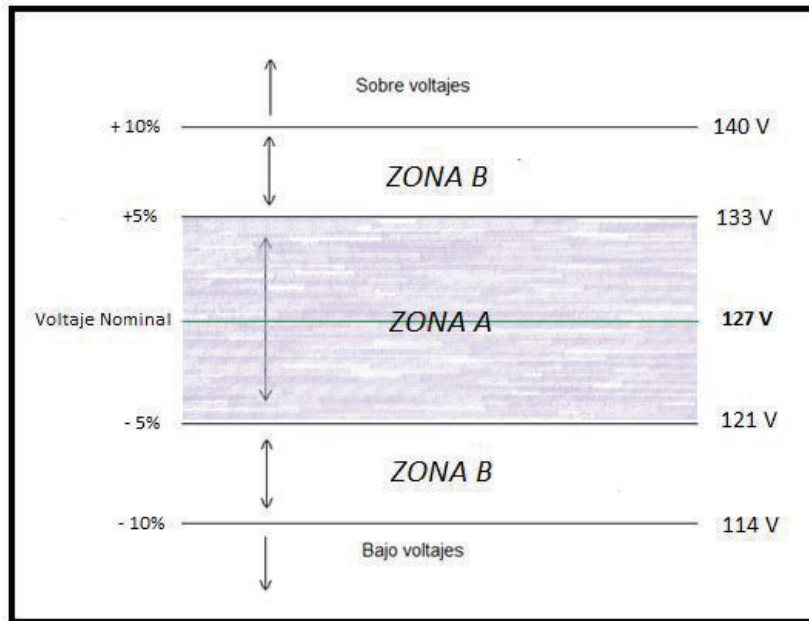


Figura 4.9 Zonas y límites de voltajes. [16]

A continuación, en la figura 4.10 se verifica la detección de un bajo voltaje, y un voltaje fuera de rango.



Figura 4.10 Detección de un voltaje fuera de rango y un bajo voltaje.
[Elaboración propia]

CAPÍTULO 5

CONCLUSIONES Y RECOMENDACIONES

5.1. CONCLUSIONES

- Con el conocimiento adquirido de programación, microcontroladores, máquinas eléctricas y electrónica durante nuestra etapa de aprendizaje, se logró ejecutar y cumplir con los objetivos planteados dentro del proyecto, que consiste en la construcción de un dispositivo de protección para motores trifásicos de inducción, basado en el microcontrolador PIC.
- En el acondicionamiento de señales para la lectura de voltaje se usó el amplificador operacional uA741, porque presenta un voltaje de desplazamiento (offset) de entrada bajo y, además; un ancho de banda típico de 1Mhz.
- Las señales de los voltajes de cada una de las fases a medir fueron acondicionadas en un rango de 0 a 5 V DC, que es el máximo rango de voltaje que admite los microcontroladores PIC de la familia 18XXXX.
- La configuración del oscilador principal (cristal externo) del microcontrolador a una frecuencia de 20 Mhz, ayudó a facilitar la comunicación entre el PIC y el módulo GLCD. Usando frecuencias menores, el GLCD presentaba los textos y lecturas de voltaje extremadamente lentos y además se observaba caracteres extraños.
- Mediante el uso del microcontrolador PIC18F4550 de *Microchip* se consiguió procesar las señales analógicas y digitales enviadas desde el bloque de acondicionamiento, para medir los valores de voltaje e identificar la secuencia de fases a fin de presentarlos en la pantalla gráfica de cristal líquido (GLCD).
- Para el desarrollo del programa se usó el compilador MikroC compatible con el lenguaje C/C++; debido a la ventaja de contar con librerías hechas y de fácil aprendizaje, lo convirtió en una herramienta adecuada para la ejecución de esta parte trascendental del proyecto.
- Las pruebas de funcionamiento del programa desarrollado para el PIC y el bloque de acondicionamiento de señales para medir voltajes mostraron que el dispositivo responde efectivamente a las variaciones de voltaje, producto de una pérdida de fases, bajo o sobre voltaje; estos dos últimos generados por un Variac.
- Se consiguió un completo aislamiento entre la parte digital (PIC) y la de potencia usando un optoacoplador en el módulo relé. La ventaja óptica del optoacoplador, separa físicamente las partes mencionadas para evitar daños en el microcontrolador producto de un cortocircuito o algún otro tipo de anomalía eléctrica producida en la parte de potencia.

- La interfaz del módulo GLCD con el microcontrolador PIC proporciona al usuario una forma sencilla y segura para la localización de las fallas contempladas en este proyecto, que se pueden generar en la red eléctrica que energiza al motor.
- Las pruebas de funcionamiento del dispositivo de protección desarrollado arrojaron resultados válidos, con un porcentaje de error en las lecturas de voltajes, con respecto al instrumento patrón, menor a 1,5 %; y además se aseguró la pronta desconexión de la carga al detectar cualquier falla. Con estas pruebas se garantiza su adecuado funcionamiento.

5.2. RECOMENDACIONES

- Se debe hacer un análisis detallado para la selección del microcontrolador, tomando en cuenta características como la capacidad de memoria de programa, de datos, periféricos a gobernar, número de canales del conversor análogo a digital, frecuencia de trabajo del CPU y el número de pines necesarios para la aplicación a desarrollar.
- Se recomienda para trabajos futuros integrar otros parámetros a monitorear, tales como: corriente y temperatura de los bobinados, con el fin de obtener un sistema de protección más eficiente.
- Un plan de mantenimiento adecuado se logra con un historial de fallas de cada maquinaria, ya sea esta eléctrica, mecánica o electromecánica; para ello se recomienda desarrollar un algoritmo que archive la incidencia de las fallas y luego descargarlas mediante un protocolo de comunicación hacia la PC.
- Hay que tomar en cuenta la señalización de las borneras de conexión para la entrada y salida trifásica, puesto que una conexión incorrecta produciría errores en las lecturas de voltajes y en la identificación de la secuencia de fases.
- Con el fin de reducir el tamaño del dispositivo, se recomienda reemplazar la circuitería del acondicionamiento de la señal de voltaje AC por el C.I AD736 (convertidor de RMS a DC), el cual tiene un costo elevado.
- Hay que tomar precaución con la conexión de la red trifásica al dispositivo de protección, por la razón de que la carcasa del dispositivo es metálica. Para evitar cualquier posible contacto indeseado como cortocircuitos, se usó jacks plásticos para aislar la carcasa de los puntos de conexión.

BIBLIOGRAFÍA

- [1] Caltenco, M. (2008). Criterios de ingeniería aplicables en la selección óptima de motores trifásicos de inducción tipo jaula de ardilla. Proyecto previo a la obtención del Título de Ingeniero Electricista. Instituto Politécnico Nacional, México.
- [2] Charles, A., y Matthew, S. (2006). Fundamentos de Circuitos eléctricos. México: McGraw-Hill.
- [3] Chiluisa, C. (s.f.). Motores de corriente alterna, Reparación y Diseño. Quito.
- [4] Coughlin, R., y Driscoll, F. (1993). Amplificadores operacionales y circuitos integrados lineales. México: Prentice-Hall HISPANOAMÉRICA.
- [5] Jami, F. (2003). Desbalance de Voltaje. Proyecto previo a la obtención del Título de Ingeniero Eléctrico. Escuela Politécnica Nacional, Quito.
- [6] Quispe, E. (2004). Efectos del desbalance y los armónicos de voltaje en la operación de los Motores de Inducción. Universidad Autónoma de Occidente, Departamento de Energética y Mecánica. Cali: Grupo de Investigación de Energías GIEN-UAO.
- [7] Microchip Technology, <<PIC18F2455/2550/4455/4550>> [En línea] Disponible en: <http://www.microchip.com/wwwproducts/en/PIC18F4550> [Último acceso: 08/ 12/ 2016]
- [8] Mendoza, J. (2011). Diseño y simulación de sistemas microcontrolados en lenguaje C [En línea] Disponible en: http://www.mikroe.com/download/eng/documents/publications/otherbooks/libro_simulacion_mikroc.pdf [Último acceso: 15 / 07 / 2016]
- [9] Mikroelectronika (2011). PIC microcontrollers – Programing in C [En línea] Disponible en: <http://learn.mikroe.com/ebooks/picccprogramming/front-matter/introduction/> [Último acceso: 10 / 08 / 2016]
- [10] Wikipedia (s.f.) Sistemas trifásicos [En línea] Disponible en: http://es.wikipedia.org/wiki/Sistema_trif%C3%A1sico [Último acceso: 02 / 08 / 2016]
- [11] Texas Instrument, <<Reguladores de voltaje series 7800>> [En línea] Disponible en: <https://www.sparkfun.com/datasheets/Components/LM7805.pdf> [Último acceso: 10 / 08 / 2016]
- [12] Texas Instrument, <<LMx39x, LM2901xx Quad Differential Comparators>> [En línea] Disponible en: <http://www.ti.com/lit/ds/symlink/lm2901.pdf> [Último acceso: 29 / 08 / 2016]
- [13] Centro de Automatización Industrial, SENA (2008). Manual de uso del programa de diseño de circuitos y simulación Proteus Layout Editor [En línea] Disponible en: <http://www.tecnica1lomas.com.ar/tutoriales/manual-proteus.pdf> [Último acceso: 26 10 2016]

[14] Keep Jump, << DPC-817C Phototransistor / Photocoupler>> [En línea] Disponible en: <http://www.keepjump.com.tw/DataSheet/Others/DPC-817C.pdf> [Último acceso: 20 / 01 / 2017]

[15] Schneider Electric, <<LC1D12M7 / Contactor 3P – AC 3>> [En línea] Disponible en: <http://www.alliedelec.com/m/d/496b082aafb4361bd51519bba481d322.pdf> [Último acceso: 20 / 01 / 2017]

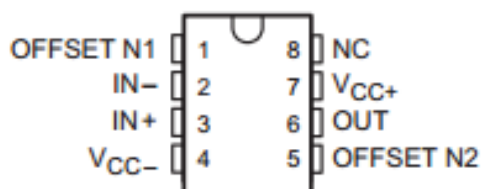
[16] IEC 60034-1: 2010, << Máquinas eléctricas rotativas. Parte 1: Características asignadas y características de funcionamiento. >> [En línea] Disponible en: <https://law.resource.org/pub/in/bis/S05/is.iec.60034.1.2010.pdf> [Último acceso: 20 / 01 / 2017]

ANEXOS

ANEXO A: AMPLIFICADOR OPERACIONAL μ A741

Pin Configurations

μ A741M . . . JG PACKAGE
 μ A741C, μ A741I . . . D, P, OR PW PACKAGE
(TOP VIEW)



Absolute Maximum Ratings

over virtual junction temperature range (unless otherwise noted)⁽¹⁾

	μ A741C		μ A741M		UNIT	
	MIN	MAX	MIN	MAX		
V_{CC} Supply voltage ⁽²⁾	-18	18	-22	22	C	
V_{ID} Differential input voltage ⁽³⁾	-15	15	-30	30	V	
V_I Input voltage, any input ⁽²⁾⁽⁴⁾	-15	15	-15	15	V	
Voltage between offset null (either OFFSET N1 or OFFSET N2) and V_{CC-}	-15	15	-0.5	0.5	V	
Duration of output short circuit ⁽⁵⁾	Unlimited					
Continuous total power dissipation	See Table 1					
T_A Operating free-air temperature range	0	70	-55	125	°C	
Case temperature for 60 seconds	FK package				260	°C
Lead temperature 1.6 mm (1/16 inch) from case for 60 seconds	J, JG, or U package				300	°C
Lead temperature 1.6 mm (1/16 inch) from case for 10 seconds	D, P, or PS package		260		N/A	°C
T_{stg} Storage temperature range	-65	150	-65	150	°C	

Switching Characteristics μ A741C, μ A741M

over operating free-air temperature range, $V_{CC\pm} = \pm 15$ V, $T_A = 25^\circ\text{C}$ (unless otherwise noted)

PARAMETER	TEST CONDITIONS	μ A741C			μ A741M			UNIT
		MIN	TYP	MAX	MIN	TYP	MAX	
t_r Rise time	$V_I = 20$ mV, $R_L = 2$ k Ω , $C_L = 100$ pF, See Figure 1		0.3			0.3		μ s
Overshoot factor			5%			5%		—
SR Slew rate at unity gain	$V_I = 10$ V, $R_L = 2$ k Ω , $C_L = 100$ pF, See Figure 1		0.5			0.5		V/ μ s

Electrical Characteristics μ A741C, μ A741M

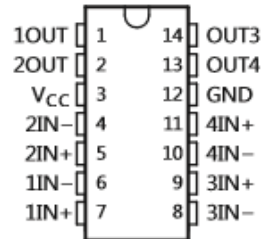
at specified virtual junction temperature, $V_{CC2} = \pm 15$ V (unless otherwise noted)

PARAMETER	TEST CONDITIONS	$T_A^{(1)}$	μ A741C			μ A741M			UNIT
			MIN	TYP	MAX	MIN	TYP	MAX	
V_{IO} Input offset voltage	$V_O = 0$	25°C		1	6		1	5	mV
		Full range			7.5		± 15	6	
$\Delta V_{IO(Adj)}$ Offset voltage adjust range	$V_O = 0$	25°C		± 15			20	200	mV
I_{IO} Input offset current	$V_O = 0$	25°C		20	200			500	nA
		Full range			300			500	
I_{IB} Input bias current	$V_O = 0$	25°C		80	500		80	500	nA
		Full range			800			1500	
V_{ICR} Common-mode input voltage range		25°C	± 12	± 13		± 12	± 13		V
		Full range	± 12			± 12			
V_{OM} Maximum peak output voltage swing	$R_L = 10$ k Ω	25°C	± 12	± 14		± 12	± 14		V
		Full range	± 12			± 12			
		25°C	± 10			± 10	± 13		
		Full range	± 10			± 10			
A_{VD} Large-signal differential voltage amplification	$R_L \geq 2$ k Ω	25°C	20	200		50	200		V/mV
		Full range	15			25			
r_i Input resistance		25°C	0.3	2		0.3	2		M Ω
r_o Output resistance	$V_O = 0$, See ⁽²⁾	25°C		75			75		Ω
C_i Input capacitance		25°C		1.4			1.4		pF
CMRR Common-mode rejection ratio	$V_{IC} = V_{ICRmin}$	25°C	70	90		70	90		dB
		Full range	70			70			
k_{SVS} Supply voltage sensitivity ($\Delta V_{IO}/\Delta V_{CC}$)	$V_{CC} = \pm 9$ V to ± 15 V	25°C		30	150		30	150	μ V/V
		Full range			150			150	
I_{OS} Short-circuit output current		25°C		± 25	± 40		± 25	± 40	mA
I_{CC} Supply current	$V_O = 0$, No load	25°C		1.7	2.8		1.7	2.8	mA
		Full range			3.3			3.3	
P_D Total power dissipation	$V_O = 0$, No load	25°C		50	85		50	85	mW
		Full range			100			100	

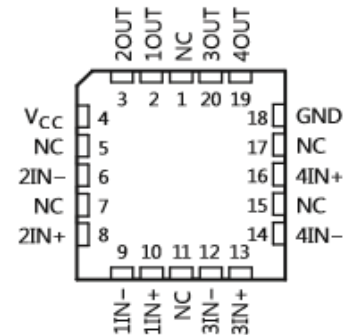
ANEXO B: COMPARADOR DE VOLTAJE LM339

6 Pin Configuration and Functions

D, DB, N, NS, PW, J, or W Package
SOIC, SSOP, PDIP, SO, TSSOP, CDIP, or CFP
Top View



FK Package
20-Pin LCCC⁽¹⁾
Top View



(1) NC = no internal connection.

Pin Functions

NAME	PIN		I/O ⁽¹⁾	DESCRIPTION
	D, J, W, B, PW, DB, N, NS	FK		
1IN+	7	10	I	Positive input pin of the comparator 1
1IN-	6	9	I	Negative input pin of the comparator 1
1OUT	1	2	O	Output pin of the comparator 1
2IN+	5	8	I	Positive input pin of the comparator 2
2IN-	4	6	I	Negative input pin of the comparator 2
2OUT	2	3	O	Output pin of the comparator 2
3IN+	9	13	I	Positive input pin of the comparator 3
3IN-	8	12	I	Negative input pin of the comparator 3
3OUT	14	20	O	Output pin of the comparator 3
4IN+	11	16	I	Positive input pin of the comparator 4
4IN-	10	14	I	Negative input pin of the comparator 4
4OUT	13	19	O	Output pin of the comparator 4
GND	12	18	I	Ground
V _{CC}	3	4	I	Supply pin
NC	—	1	—	No connect (no internal connection)
		5		
		7		
		11		
		15		
		17		

(1) I = Input, O = Output

Electrical Characteristics: LM239, LM339, LM2901, LM3302

($V^* = 5 V_{DC}$, $T_A = 25^\circ\text{C}^{(1)}$ unless otherwise stated)

PARAMETER	TEST CONDITIONS	LM239, LM339			LM2901			LM3302			UNIT
		MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX	
Input Offset Voltage	See ⁽²⁾	2.0	5.0		2.0	7.0		3	20		mV _{DC}
Input Bias Current	$I_{IN(+)}$ or $I_{IN(-)}$ with Output in Linear Range ⁽³⁾ , $V_{CM} = 0\text{ V}$	25	250		25	250		25	500		nA _{DC}
Input Offset Current	$I_{IN(+)} - I_{IN(-)}$, $V_{CM} = 0\text{ V}$	5.0	50		5	50		3	100		nA _{DC}
Input Common-Mode Voltage Range	$V^* = 30 V_{DC}$ (LM3302, $V^* = 28 V_{DC}$) ⁽⁴⁾	0	$V^* - 1.5$		0	$V^* - 1.5$		0	$V^* - 1.5$		V _{DC}
Supply Current	(LM3302, $V^* = 28 V_{DC}$) $R_L = \infty$ on all Comparators	0.8	2.0		0.8	2.0		0.8	2.0		mA _{DC}
	(LM3302, $V^* = 28 V_{DC}$) $R_L = \infty$, $V^* = 36\text{ V}$	1.0	2.5		1.0	2.5		1.0	2.5		mA _{DC}
Voltage Gain	$R_L \geq 15\text{ k}\Omega$, $V^* = 15 V_{DC}$ $V_O = 1 V_{DC}$ to $11 V_{DC}$	50	200		25	100		2	30		V/mV
Large Signal Response Time	$V_{IN} = \text{TTL Logic Swing}$, $V_{REF} = 1.4 V_{DC}$, $V_{RL} = 5 V_{DC}$, $R_L = 5.1\text{ k}\Omega$	300			300			300			ns
Response Time	$V_{RL} = 5 V_{DC}$, $R_L = 5.1\text{ k}\Omega$ ⁽⁵⁾	1.3			1.3			1.3			μs
Output Sink Current	$V_{IN(-)} = 1 V_{DC}$, $V_{IN(+)} = 0$, $V_O \leq 1.5 V_{DC}$	6.0	16		6.0	16		6.0	16		mA _{DC}
Saturation Voltage	$V_{IN(-)} = 1 V_{DC}$, $V_{IN(+)} = 0$, $I_{SINK} \leq 4\text{ mA}$	250	400		250	400		250	500		mV _{DC}
Output Leakage Current	$V_{IN(+)} = 1 V_{DC}$, $V_{IN(-)} = 0$, $V_O = 5 V_{DC}$	0.1			0.1			0.1			nA _{DC}
Input Offset Voltage	See ⁽²⁾		9.0		9	15		40			mV _{DC}
Input Offset Current	$I_{IN(+)} - I_{IN(-)}$, $V_{CM} = 0\text{ V}$		150		50	200		300			nA _{DC}
Input Bias Current	$I_{IN(+)}$ or $I_{IN(-)}$ with Output in Linear Range, $V_{CM} = 0\text{V}$ ⁽³⁾		400		200	500		1000			nA _{DC}
Input Common-Mode Voltage Range	$V^* = 30 V_{DC}$ (LM3302, $V^* = 28 V_{DC}$)		$V^* - 2.0$		0	$V^* - 2.0$		0	$V^* - 2.0$		V _{DC}
Voltage Range	See ⁽⁴⁾										
Saturation Voltage	$V_{IN(-)} = 1 V_{DC}$, $V_{IN(+)} = 0$, $I_{SINK} \leq 4\text{ mA}$		700		400	700		700			mV _{DC}
Output Leakage Current	$V_{IN(+)} = 1 V_{DC}$, $V_{IN(-)} = 0$, $V_O = 30 V_{DC}$, (LM3302, $V_O = 28 V_{DC}$)		1.0		1.0			1.0			μA_{DC}
Differential Input Voltage	Keep all V_{IN} 's $\geq 0 V_{DC}$ (or V^* , if used) ⁽⁶⁾		36		36			28			V _{DC}

Recommended Operating Conditions

over operating free-air temperature range (unless otherwise noted)

			MIN	MAX	UNIT
Supply Voltage	Single Supply	LM139N, LM239N, LM339N, LM2901N	2	36	V
		LM3302N	2	28	
	Dual Supply	LM139N, LM239N, LM339N, LM2901N	± 1	± 18	
		LM3302N	± 1	± 14	
Operating Temperature	LM139/LM139A		-55	125	$^\circ\text{C}$
	LM2901/LM3302		-40	85	
	LM239/LM239A		-25	85	
	LM339/LM339A		0	70	

ANEXO C: FOTOTRANSISTOR DPC-817C

● Features

- Current transfer ratio (CTR: 50~600% at $I_F = 5\text{mA}$, $V_{CE} = 5\text{V}$)
- High isolation voltage between input and output ($V_{iso} = 5000\text{ V rms}$)
- Creepage distance > 7.62 mm
- Operating temperature up to +110°C
- Compact small outline package
- Pb free and RoHS compliant.

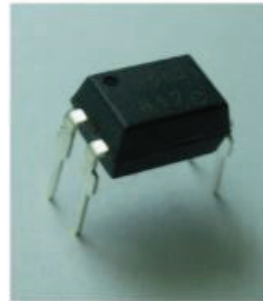
● Description

1. The DPC-817 series are optically coupled isolators containing a GaAs light emitting diode and an NPN silicon phototransistor
2. The lead pitch is 2.54mm

● Applications:

1. Computer terminals.
2. System appliances, measuring instruments.
3. Registers, copiers, automatic vending machines.
4. Electric home appliances, such as fan heaters, etc. of different potentials and impedances.
5. Signal transmission between circuits

● Dimensions

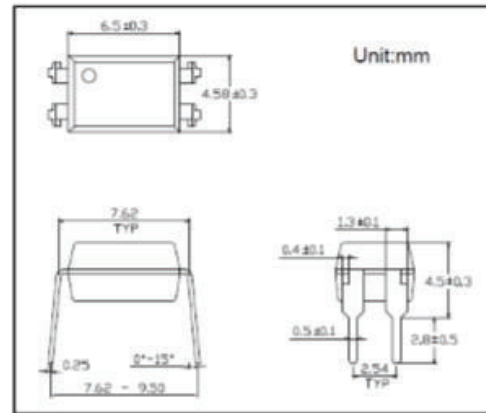


Schematic



Pin Configuration

1. Anode
2. Cathode
3. Emitter
4. Collector



● Absolute Maximum Ratings (Ta=25°C)

	Parameter	Symbol	Rating	Unit
INPUT	Forward Current	I_F	50	mA
	Reverse Voltage	V_R	6	V
	Power Dissipation	P	70	mW
OUTPUT	Collector-Emitter Voltage	V_{CEO}	35	V
	Emitter-Collector Voltage	V_{ECO}	6	V
	Collector Current	I_C	50	mA
	Collector Power Dissipation	P_C	150	mW
Total Power Dissipation		P_{tot}	200	mW
*1 Isolation Voltage		V_{iso}	5,000	Vrms
Rated impulse isolation voltage		V_{IOTM}	6,000	V
Rated repetitive peak isolation voltage		V_{IORM}	630	V
Operating Temperature		T	-40 to +85	°C

Electro-Optical Characteristics (Ta=25°C)

Parameter		Symbol	Conditions	MIN.	TYP.	MAX.	Unit.
INPUT	Forward Voltage	V_F	$I_F=20\text{mA}$	---	1.2	1.4	V
	Reverse Current	I_R	$V_R=4\text{V}$	---	---	10	μA
	Terminal Capacitance	C_T	$V=0, f=1\text{KHz}$	---	30	250	pF
OUTPUT	Collector Dark Current	I_{CE0}	$V_{CE}=20\text{V}, I_F=0$	---	---	100	nA
	Collector-Emitter Breakdown Voltage	BV_{CEO}	$I_C=0.1\text{mA}$ $I_F=0$	35	---	---	V
	Emitter-Collector Breakdown Voltage	BV_{ECO}	$I_E=10\mu\text{A}$ $I_F=0$	6	---	---	V
TRANSFER CHARACTERISTICS	Collector Current	I_C	$I_F=5\text{mA}$	2.5	---	30	mA
	*1 Current Transfer Ratio	CTR	$V_{CE}=5\text{V}$	50	---	600	%
	Collector-Emitter Saturation Voltage	$V_{CE(sat)}$	$I_F=20\text{mA}$ $I_C=1\text{mA}$	---	0.1	0.2	V
	Isolation Resistance	R_{iso}	DC500V 40~60%R.H.	5×10^{10}	1×10^{11}	---	
	Floating Capacitance	C_f	$V=0, f=1\text{MHz}$	---	0.6	1	pF
	Cut-Off Frequency	f_c	$V_{CE}=5\text{V},$ $I_C=2\text{mA}$ $R_L=100 \Omega,$ -3dB	---	80	---	kHz
	Response Time(Rise)	t_r	$V_{CE}=2\text{V},$ $I_C=2\text{mA}$	---	4	18	μs
	Response Time(Fall)	t_f	$R_L=100 \Omega$	---	3	18	μs

ANEXO D: CONTACTOR LC1D12M7

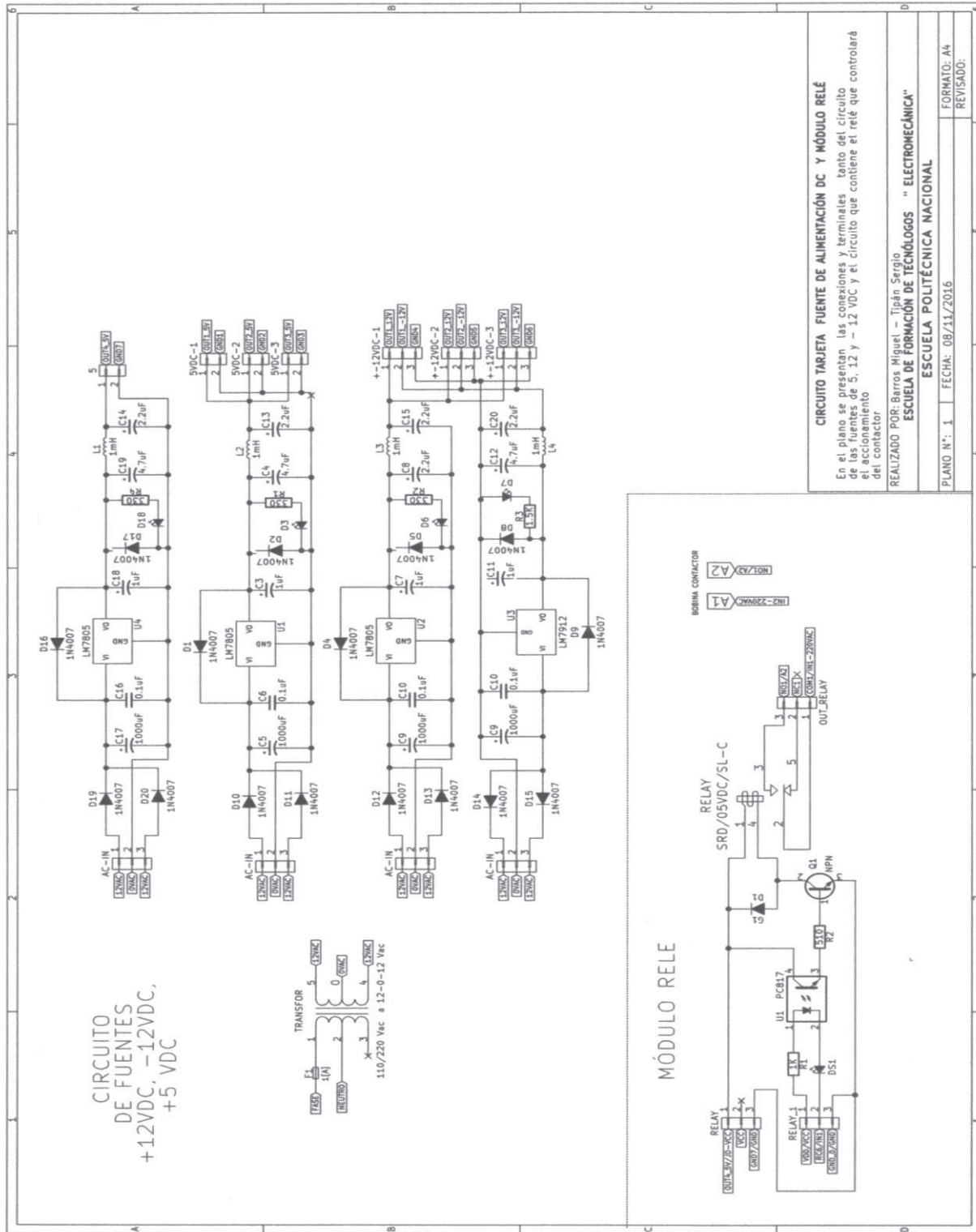
LC1D12M7

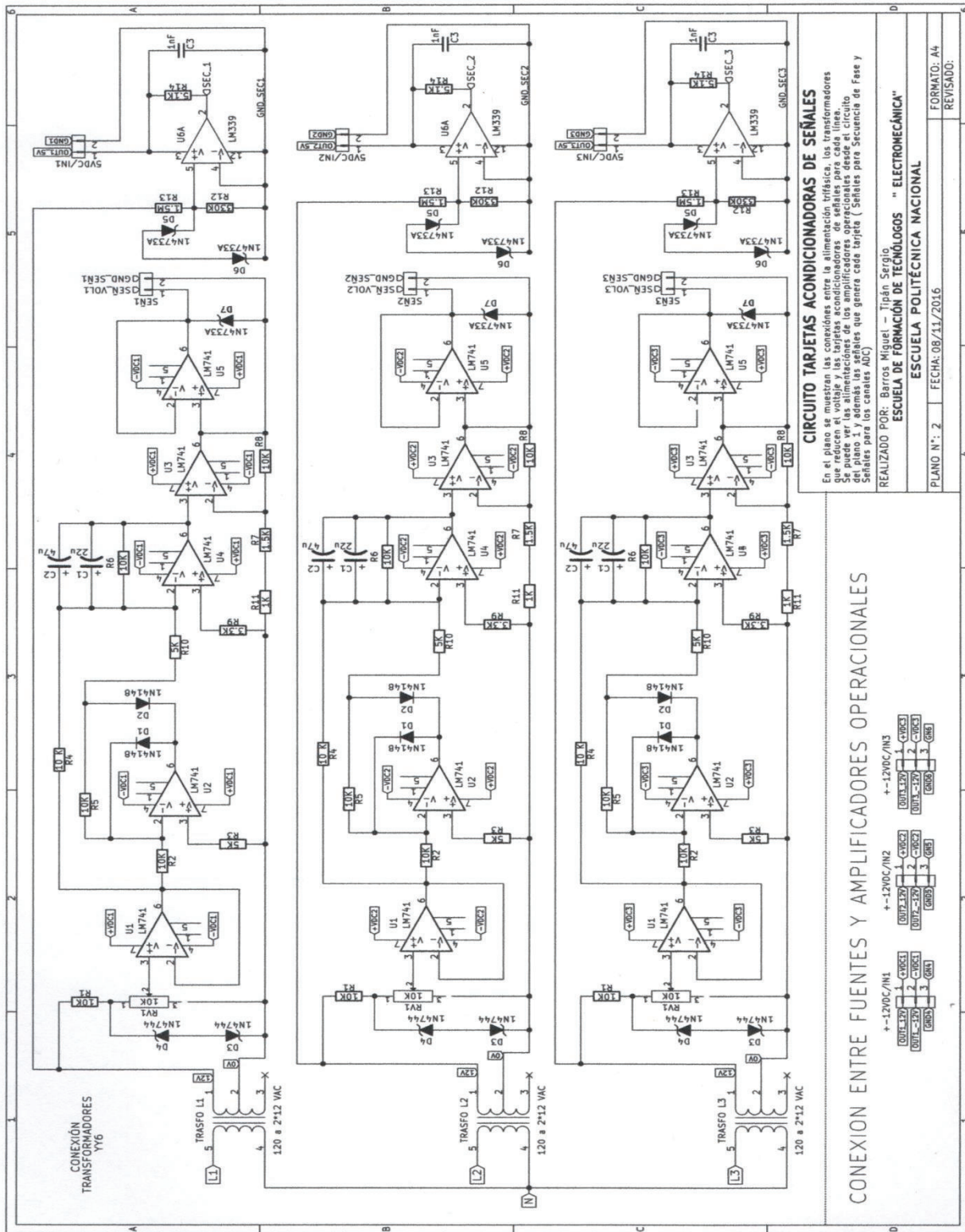
TeSys D contactor - 3P(3 NO) - AC-3 - ≤ 440
V 12 A - 220 V AC coil

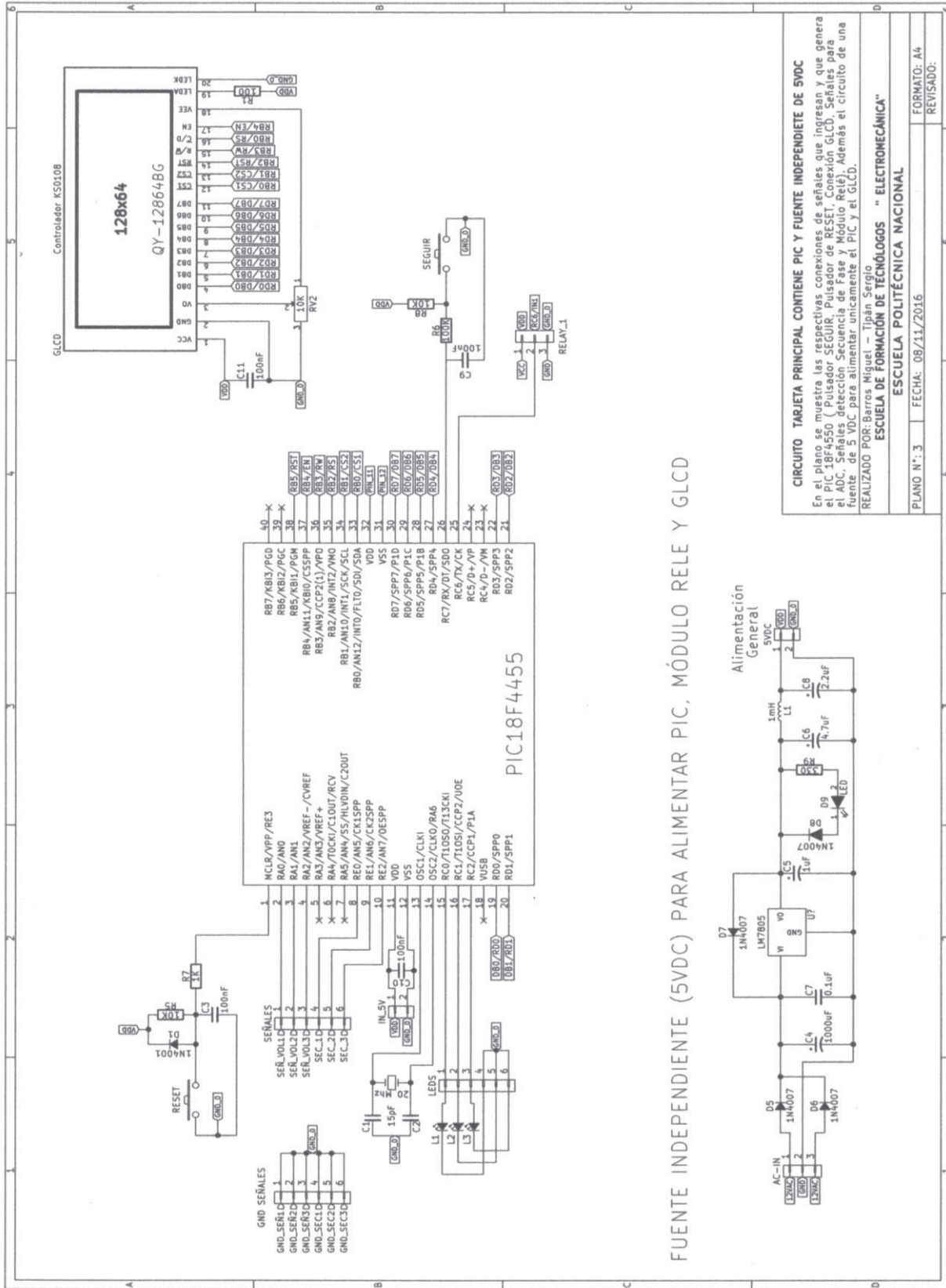
Main

Range of product	TeSys D
Product or component type	Contactor
Device short name	LC1D
Contactor application	Motor control Resistive load
Utilisation category	AC-1 AC-3
Poles description	3P
Power pole contact composition	3 NO
[Ue] rated operational voltage	≤ 690 V DC for power circuit ≤ 690 V AC 25...400 Hz for power circuit
[Ie] rated operational current	25 A (≤ 60 °C) at ≤ 440 V AC AC-1 for power circuit 12 A (≤ 60 °C) at ≤ 440 V AC AC-3 for power circuit
Motor power kW	7.5 kW at 660...690 V AC 50/60 Hz 7.5 kW at 500 V AC 50/60 Hz 5.5 kW at 415...440 V AC 50/60 Hz 5.5 kW at 380...400 V AC 50/60 Hz 3 kW at 220...230 V AC 50/60 Hz
Motor power HP (UL / CSA)	10 hp at 575/600 V AC 50/60 Hz for 3 phases motors 7.5 hp at 460/480 V AC 50/60 Hz for 3 phases motors 3 hp at 230/240 V AC 50/60 Hz for 3 phases motors 3 hp at 200/208 V AC 50/60 Hz for 3 phases motors 2 hp at 230/240 V AC 50/60 Hz for 1 phase motors 1 hp at 115 V AC 50/60 Hz for 1 phase motors
Control circuit type	AC 50/60 Hz
Control circuit voltage	220 V AC 50/60 Hz
Auxiliary contact composition	1 NO + 1 NC

ANEXO E: PLANOS DIAGRAMAS ELÉCTRICOS Y ELECTRÓNICOS







ANEXO F: CÓDIGO FUENTE

```

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
//          ESCUELA POLITÉCNICA NACIONAL          //
//          ESCUELA DE FORMACIÓN DE TECNÓLOGOS    //
//          CARRERA ELECTROMECAÁNICA              //
// FECHA: 02/09/2016                             //
// ESTUDIANTES: - Tipán Sergio                    //
//               - Barros Miguel                  //
// TEMA: Programa para dispositivo de protección.c //
//       - Detector de secuencia de fase          //
//       - Sensor de voltajes (FASE-NEUTRO)      //
//       - Detector de porcentaje de desbalance de voltajes //
//                                               //
// MICROCONTROLADOR: PIC18F4550                  //
// OSCILADOR:      20 Mhz                        //
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

//===== Símbolos y Macros =====
# define Seguir PORTC.F7
# define RELE   PORTC.F6
//===== Fin de Símbolos y Macros =====
// Declaración de las conexiones del modulo Glcd al MCU 18F4550

char GLCD_DataPort at PORTD;

sbit GLCD_CS1 at RB0_bit;
sbit GLCD_CS2 at RB1_bit;
sbit GLCD_RS  at RB2_bit;
sbit GLCD_RW  at RB3_bit;
sbit GLCD_EN  at RB4_bit;
sbit GLCD_RST at RB5_bit;

sbit GLCD_CS1_Direction at TRISB0_bit;
sbit GLCD_CS2_Direction at TRISB1_bit;
sbit GLCD_RS_Direction  at TRISB2_bit;
sbit GLCD_RW_Direction  at TRISB3_bit;
sbit GLCD_EN_Direction  at TRISB4_bit;
sbit GLCD_RST_Direction at TRISB5_bit;
//Fin de declaraciones de conexiones

/***** Declaración de variables globales *****/

const float factor=0.00489;          // Factor de conversión del modulo ADC
const code char EPN[1024];
const code char Electromecanica[1024];

bit L_1;
bit L_2;
char i;

char ctrl_1=0;
char ctrl_2=0;
char ctrl_3=0;

char txt [4];
char txt1[15];

char fuera_rango=0;
char sobre_volt=0;

```

```

char bajo_volt=0;

short int Line_OK=0;
short int sec_fase=0;
short int timetest;
short int cnt=0;

float VL1,VL2,VL3,Promedio,Desv_1,Desv_2,Desv_3,Des_eq;
/***** Fin declaración de variables globales *****/

/*===== Declaración de prototipo de funciones =====*/

// Prototipo de función para leer los mensajes de la memoria EEPROM interna
void mensajes (int num_men,char num_pix,char num_pag,char color);
// Prototipo de función para leer el A/D y calcular el nivel de voltaje
float Volts (char canal);
// Prototipo de función para imprimir voltajes
void imprimir (float voltaje, char pagina);
// Prototipo de función para detección fallas, sobre y bajos voltajes
void detector (float voltio, char pag);
// Prototipo de función para borrar textos
void borrar (char pix, char pag_n);
/*===== Fin Declaración de prototipo de funciones =====*/

void main()
{
/*----- Configuración interna del MCU -----*/
ADCON1=0B00001100, // RA <2:0> fijar como entradas analógicas
UCON=0B00000000; // Módulo USB desactivado

PORTA=0B00000000; // Valor de inicialización de los puertos
PORTB=0B00000000;
PORTC=0B00000000;
PORTD=0B00000000;
PORTE=0B0000;

TRISA=0B0000111; // RA <2:0> fijar como entradas, el resto como salidas
TRISB=0B00000000; // RB <7:0> fijar como salidas digitales
TRISC=0B10000000; // RC7 fijar como entrada digital, el resto como salidas
TRISD=0B00000000; // RD <7:0> fijar como salidas digitales
TRISE=0B111; // RE <2:0> fijar como entradas digitales
RELE=1; // inhabilitar la señal del RELÉ

T1CON=0B00110000; // TMR1 como temporizador, prescaler 1:8, TMR1 OFF
TMR1IF_bit=0; // Bandera de Interrupción encerada en 0

TMR1H=0X0B; // Carga inicial 3036(0X0BDC), para interrupción de
TMR1L=0XDC; // intervalos de tiempo cada 100 ms

TMR1IE_bit=1; // Habilitar interrupción del TMR1 por desbordamiento (overflow)
INTCON=0XC0; // Permiso para habilitar interrupciones globales y periféricas

Glcd_Init(); // Inicializar módulo Glcd con la configuración predeterminada

/*----- Fin de configuración del MCU -----*/

Glcd_Fill(0x00); //Limpiar Lcd gráfico
Delay_ms(10); //Retardo de 10 ms

//-----Animación del mensaje de presentación-----

```

```

Glcd_Rectangle(5,15,120,25,1);
mensajes(0,18,4,1);
mensajes(1,18,6,1);

for (i=1;i<115;i++){
    Glcd_V_Line(15,25,i+5,1);
    Delay_ms(50);    //50ms
}
Glcd_Fill(0x00);    //Limpiar LCD gráfico
//----- Limpiar Lcd y fin de la presentación -----
Delay_ms(10);

Glcd_Image(EPN);    // Graficar escudo de la EPN
Delay_1sec();
Delay_1sec();

Glcd_Fill(0x00);    //Limpiar LCD gráfico
Delay_ms(10);

Glcd_Image(Electromecanica);    // Graficar escudo de la EPN
Delay_1sec();
Delay_1sec();

Glcd_Fill(0x00);    //Limpiar Lcd gráfico
Delay_ms(10);
// Escribir los mensajes grabados en la EEPROM en la posición deseada
mensajes(9,18,0,1);
mensajes(10,18,1,1);
Glcd_Box(0,22,127,32,2);
mensajes(11,18,3,0);
mensajes(12,12,5,1);
mensajes(13,12,6,1);
mensajes(14,18,7,1);

Delay_ms(2000);    // Retardo de 2 segundos

Glcd_Fill(0x00);    //Limpiar Lcd gráfico
Delay_ms(10);
// Escribir mensajes en la posición deseada
Glcd_Box(0,0,127,16,2);
mensajes(2,18,0,0);
mensajes(3,18,1,0);
Glcd_Rectangle(0,18,127,63,1);
mensajes(8,18,3,1);

/*----- Identificación de las líneas conectadas -----*/
do {
    if (ADC_Read(0)>700){    // Leer y evaluar la señal de L1, ¿L1 conectado?
        RC0_bit=1;    // Encender LED correspondiente a L1
        Glcd_Write_Char('L',36,4,1);
        Glcd_Write_Char('1',42,4,1);
    }
    else {
        RC0_bit=0;    // Apagar LED de L1
        borrar(36,4);
    }
    Delay_us(8);

    if (ADC_Read(1)>700){    // Leer y evaluar la señal de L2, ¿L2 conectado?
        RC1_bit=1;    // Encender LED correspondiente a L2

```

```

        Glcd_Write_Char('L',60,4,1);
        Glcd_Write_Char('2',66,4,1);
    }
else{
    RC1_bit=0;           // Apagar LED de L2
    borrar(60,4);
    }
Delay_us(8);

if (ADC_Read(2)>700){    // Leer y evaluar la señal de L3, ¿L3 conectado?
    RC2_bit=1;          // Encender LED correspondiente a L3
    Glcd_Write_Char('L',84,4,1);
    Glcd_Write_Char('3',90,4,1);
    }
else{
    RC2_bit=0;          // Apagar LED de L3
    borrar(84,4);
    }
Delay_us(8);

if (RC0_bit&RC1_bit&RC2_bit)
    Line_OK=0;          //Variable para indicar que las líneas están conectadas
else Line_OK=1;
}
while(Line_OK || Seguir);

/*----- Fin Identificación de las líneas conectadas -----*/

/*=====Identificación de secuencia de fases=====*/
mensajes(5,18,5,1);

do {
    if (Line_OK==0)
    {
        L_1=RE0_bit;    // Memoriza el estado lógico de la línea L1
        L_2=RE1_bit;    // Memoriza el estado lógico de la línea L2

        if (L_1|L_2)    // Comprobar los estados de L1 y L2 ¿están en "0" lógico?
            continue;
        if (!L_1)       // Esperar hasta que la línea L1 (referencia) cambie de
            while(!RE0_bit); // estados, 0 a 1(detectar flanco positivo en L1)
        Delay_ms(15);   // retardo de 15 milisegundos
        sec_fase=PORTE; // leer el Puerto E y determinar la secuencia de
        switch (sec_fase){ // fases basado en los estados lógicos de L2 y L3
            case 2: mensajes(7,18,6,1);
                break;
            case 4: mensajes(6,18,6,1);
                break;
            default: break;
        }
        Line_OK=1; // Variable para indicar que la secuencia fue identificada
    }
} while(Seguir);
/*=====Fin Identificación de secuencia de fases=====*/

Glcd_Box(1,24,126,62,0);

Glcd_Write_Text("VOLTAJES",12,3,1); // Escribir los textos en la posición

```

```

Glcd_Write_Text("DESBALANCE",66,3,1); // especificada
Glcd_Write_Text("L1=",3,4,1);
Glcd_Write_Text("L2=",3,5,1);
Glcd_Write_Text("L3=",3,6,1);

Glcd_Write_Text("[V]",51,4,1);
Glcd_Write_Text("[V]",51,5,1);
Glcd_Write_Text("[V]",51,6,1);
Glcd_Write_Char('%',108,4,1);

while(1){ // BUCLE INFINITO

    VL1=Volts(0); // Ejecuta la función "volts" por medio del parámetro
                // "el canal del modulo ADC"
                // el valor devuelto se le asigna a la variable "VL1"

    VL2=Volts(1); // Ejecuta la función "volts" por medio del parámetro
                // "el canal del modulo ADC"
                // el valor devuelto se le asigna a la variable "VL2"

    VL3=Volts(2); // Ejecuta la función "volts" por medio del parámetro
                // "el canal del modulo ADC"
                // el valor devuelto se le asigna a la variable "VL3"

//----- Rutina para calcular el desbalance de voltaje -----
Promedio=(float)(VL1+VL2+VL3)/3; // Calcular el Promedio
                                // de los 3 voltajes
Desv_1=fabs(VL1-Promedio); // Calcular las Desviaciones para cada
Desv_2=fabs(VL2-Promedio); // línea con respecto al promedio
Desv_3=fabs(VL3-Promedio);

if ((Desv_1>Desv_2)&&(Desv_1>Desv_3)) // Determinar la máxima
    Des_eq=(float)(Desv_1*100)/Promedio; // desviación entre las 3
                                // líneas y calcular
if ((Desv_2>Desv_1)&&(Desv_2>Desv_3)) // el porcentaje de desbalance
    Des_eq=(float)(Desv_2*100)/Promedio; // existente en la red

if ((Desv_3>Desv_1)&&(Desv_3>Desv_2))
    Des_eq=(float)(Desv_3*100)/Promedio;
//----- FIN Rutina para calcular el desbalance de voltaje -----

/*===== Detección de fallas =====*/

    detector(VL1,4); // Ejecutar función "detector" por medio de los
    detector(VL2,5); // parámetros de valores de voltaje correspondientes a
    detector(VL3,6); // cada línea

    if (Des_eq>3&&Des_eq<=6) //Desbalance de la red mayor a 3% y <=6%
        fuera_rango=1;
    if (Des_eq>6)
        sobre_volt=1;
    if (sobre_volt==1||bajo_volt==1) //Priorizar fallas más graves
        fuera_rango=0;
    if (fuera_rango==1||sobre_volt==1||bajo_volt==1) // ¿Existen fallas?
    {
        if(fuera_rango){ // ¿Existen fallas leves?
            if (ctrl_1==0 && ctrl_3==0)
            {
                timetest=60; // Cargar 60 segundos para verificar
            }
        }
    }
}

```

```

        ctrl_1=1;           // fallas leves
        ctrl_2=0;
        cnt=0;
    }
}

if(sobre_volt||bajo_volt)      // ¿Existen sobre o bajo voltajes?
{
    if (ctrl_2==0 && ctrl_3==0)
    {
        timetest=3;           // Cargar 5 segundos para verificar
        ctrl_2=1;
        ctrl_1=0;           // sobre y bajo voltajes
        cnt=0;
    }
}

if(TMR1ON_bit==0 && timetest!=0){
    if (ctrl_3==0)
        TMR1ON_bit=1;       // Activar TMR1
    }
if (timetest==0)
{
    TMR1ON_bit=0;
    RELE=1;
    ctrl_3=1;
    Glcd_Write_Text("PELIGRO",81,5,1);
}
ByteToStr(timetest,txt);
Glcd_Write_Char(*(txt+1),93,6,1);
Glcd_Write_Char(*(txt+2),99,6,1);

fuera_rango=0;
sobre_volt=0;
bajo_volt=0;
}
else {
    if (ctrl_3==0){
        TMR1ON_bit=0;       // Desactivar TMR1
        if (RELE==1)       // ¿RELE apagado?
            RELE=0;       // Activar RELE

        ctrl_1=0;         // Encerar variables de control en cero
        ctrl_2=0;
        cnt=0;

    }

    if (ctrl_3==1){
        Glcd_Write_Text("Reconec",81,5,1); //Rutina para reconectar
        //la carga después de
        if(Seguir==0){    //superar la falla
            ctrl_3=0;
            RELE=0;
            borrar(81,5);
            borrar(93,5);
            borrar(105,5);
            borrar(111,5);
        }
    }
}

```

```

        borrar(69,4);
        borrar(69,5);
        borrar(69,6);
        borrar(93,6);
    }

/*===== FIN Detección de fallas =====*/

//////////Impresión de los voltajes y desbalance en el Glcd//////////
imprimir(VL1,4); //Para el valor de la línea L1
imprimir(VL2,5); //Para el valor de la línea L2
imprimir(VL3,6); //Para el valor de la línea L3

FloatToStr(Des_eq,txt1); // Transformar float a cadena de caracteres
if (Des_eq<1){

    Glcd_Write_Char('0',90,4,1);
    Glcd_Write_Char('.',96,4,1);
    Glcd_Write_Char(*(txt1),102,4,1); // Imprimir primer decimal
}
if (Des_eq>=1&&Des_eq<10) //Para el porcentaje de desbalance de la red
{
    Glcd_Write_Char(*(txt1),90,4,1); // Imprimir unidades
    Glcd_Write_Char(*(txt1+1),96,4,1); // Punto decimal
    Glcd_Write_Char(*(txt1+2),102,4,1); // Imprimir el primer decimal
}

if (Des_eq>=10){

    Glcd_Write_Char(*(txt1),90,4,1);
    Glcd_Write_Char(*(txt1+1),96,4,1);
    Glcd_Write_Char(' ',102,4,1);
}
Delay_ms(400);
}
}

//-----Definición de prototipo de funciones-----

void interrupt (){
    TMR1IF_bit=0; // Borra bandera de interrupción TMR1
    TMR1H=0X0B; // Cargar valor para generar interrupción cada 100 ms
    TMR1L=0XDC;
    cnt++; // Aumentar una unidad el contador por cada interrupción
    if (cnt==10){
        cnt=0; // Encerar contador en 0
        timetest--;
    }
}

void borrar (char pix, char pag_n){
    Glcd_Write_Text(" ",pix,pag_n,1);
}

void mensajes (int num_men,char num_pix,char num_pag, char color){
    char txt [16]; // Array para almacenar el mensajes a leer

```



```

int j;
int k;

k=0;
num_men=num_men*0x10;          // Operación para determinar la dirección del
                               // mensaje seleccionado

// Leer mensaje de 16 caracteres y guardar en el array "txt"
for (j=num_men;j<=num_men+0x0F&&j!=0xFF;j++){

    txt[k]=EEPROM_Read(j);
    k++;
    Delay_ms(20);
}
txt[16]='\0';                  // Caracter nulo para indicar el fin del array
Glcd_Write_Text(txt,num_pix,num_pag,color); // Presentar el mensaje en GLCD
}

float Volts (char canal){
    char cnt_mues;             // Variable para el numero de muestras del ADC
    unsigned int add_adc=0;    // Variable para la sumatoria de muestras
                               // y el promedio final
    float resultado=0;        // Variable para almacenar el valor de voltaje

    // Tomar 100 muestras de la misma señal analógica y luego promediarla
    for (cnt_mues=0;cnt_mues<100;cnt_mues++){
        add_adc+=ADC_Read(canal);
        Delay_us(80);         // Retardo mínimo recomendado por el fabricante
        }                     // para la siguiente adquisición de dato

    add_adc=(int)(add_adc/100); // Promedio de muestras

    if (canal==0){            // Línea conectada, encender LED correspondiente
        if (add_adc>700)
            RC0_bit=1;
        else RC0_bit=0;      // Línea desconectada, apagar LED
    }
    if (canal==1){
        if (add_adc>700)
            RC1_bit=1;
        else RC1_bit=0;
    }
    if (canal==2){
        if (add_adc>700)
            RC2_bit=1;
        else RC2_bit=0;
    }

    resultado=(float)(add_adc*factor); // Calcular valor de
    resultado*=29;                     // voltaje

    return resultado;                 // Devolver valor de voltaje
}

void detector (float voltio, char pag){

```

```

if((voltio<121&&voltio>=114)|| (voltio>133&&voltio<=140)){ // Voltaje dentro de
    fuera_rango=1;
    Glcd_Write_Text("FR",69,pag,1);          //la zona B rango fallas leves
}
if(voltio>140){                                // Sobre voltaje
    sobre_volt=1;
    Glcd_Write_Text("SV",69,pag,1);          //rango de fallas graves
}
if(voltio<114){                                // Bajo voltaje o pérdida de fase
    bajo_volt=1;
    Glcd_Write_Text("BV",69,pag,1);          //rango de fallas graves
}
}

void imprimir (float voltaje, char pagina){
    char txt2 [15];

    FloatToStr(voltaje,txt2); // Transformar de float a cadena de caracteres
    if (voltaje<1){
        Glcd_Write_Char('0',21,pagina,1); // Imprimir las centenas
        Glcd_Write_Char('.',27,pagina,1); // Imprimir las decenas
        Glcd_Write_Char('0',33,pagina,1); // Imprimir las unidades
        Glcd_Write_Char('0',39,pagina,1); // Punto decimal
        Glcd_Write_Char('0',45,pagina,1); // Imprimir el primer decimal
    }
    if(voltaje>=1){
        Glcd_Write_Char(*(txt2),21,pagina,1); // Imprimir las centenas
        Glcd_Write_Char(*(txt2+1),27,pagina,1); // Imprimir las decenas
        Glcd_Write_Char(*(txt2+2),33,pagina,1); // Imprimir las unidades
        Glcd_Write_Char(*(txt2+3),39,pagina,1); // Punto decimal
        Glcd_Write_Char(*(txt2+4),45,pagina,1); // Imprimir el primer decimal
    }
}
//-----Fin Definición de prototipo de funciones-----
// -----
// GLCD Picture name: EPN.bmp
// GLCD Model: KS0108 128x64
// -----

const code char EPN[1024] = {
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255,
63, 31, 15, 7, 3, 3, 3, 3, 3, 3, 7, 15, 31, 63, 255, 255, 255,
255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255,
248, 240, 224, 192, 128, 128, 0, 0, 128, 128, 192, 224, 240, 248, 255, 255, 255,
255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 0, 0, 252, 36, 36,
4, 0, 24, 36, 36, 196, 0, 248, 4, 4, 4, 136, 0, 252, 0, 0, 0, 252,
0, 0, 252, 36, 36, 4, 0, 252, 0, 0, 0, 0, 248, 68, 68, 68, 248, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 223, 223, 223, 223, 223, 223, 223, 223, 223, 223, 223, 223, 223, 223, 223, 223, 223, 223, 223,
223, 223, 223, 223, 223, 223, 219, 219, 219, 217, 24, 216, 217, 219, 219, 219, 223,
223, 223, 223, 223, 223, 223, 223, 223, 223, 223, 223, 223, 223, 223, 223, 223, 223, 223, 223, 0,
0, 129, 129, 129, 129, 0, 1, 1, 129, 128, 128, 0, 1, 129, 1, 0, 0, 0,

```

```

129, 1, 129, 128, 128, 128, 129, 1, 129, 161, 176, 145, 1, 1, 129, 128, 129,
0, 0, 128, 1, 0, 0, 0, 128, 0, 128, 0, 0, 128, 128, 128, 0, 0, 0,
128, 128, 128, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 255, 255, 63, 63, 31, 31,
31, 31, 31, 63, 223, 63, 63, 31, 31, 31, 31, 31, 63, 63, 255, 255, 0,
255, 255, 255, 255, 127, 63, 31, 63, 127, 127, 15, 15, 15, 127, 127, 63, 31,
63, 127, 255, 255, 255, 0, 0, 63, 8, 8, 8, 7, 0, 31, 32, 32, 32,
31, 0, 63, 32, 32, 32, 0, 63, 0, 0, 0, 63, 0, 0, 0, 63, 36,
36, 32, 0, 31, 32, 32, 32, 17, 0, 63, 3, 6, 12, 24, 63, 0, 63,
0, 31, 32, 32, 32, 17, 0, 63, 8, 8, 8, 63, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 255, 255, 0, 0, 128, 128, 128, 128, 128, 128, 0, 15, 248, 0, 128, 128, 128,
128, 128, 0, 0, 255, 255, 0, 255, 255, 255, 199, 199, 198, 0, 124, 254, 199,
187, 187, 187, 199, 254, 124, 0, 198, 199, 199, 255, 255, 0, 0, 240, 96, 192,
128, 0, 240, 0, 224, 16, 16, 16, 224, 0, 224, 16, 16, 16, 32, 0, 240,
0, 224, 16, 16, 16, 224, 0, 240, 96, 192, 128, 0, 240, 0, 224, 16, 16, 16,
224, 0, 240, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 31, 63, 127, 253, 252, 252, 252, 252, 253,
253, 252, 253, 252, 252, 252, 252, 252, 253, 255, 255, 255, 0, 255, 255, 255, 255,
253, 248, 240, 248, 252, 253, 225, 225, 225, 253, 252, 248, 240, 248, 253, 127, 63,
31, 0, 0, 7, 0, 0, 1, 3, 7, 0, 7, 1, 1, 1, 7, 0, 3,
4, 4, 4, 2, 0, 7, 0, 3, 4, 4, 4, 3, 0, 7, 0, 0, 1, 3,
7, 0, 7, 1, 1, 1, 7, 0, 7, 4, 4, 4, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 248,
249, 249, 249, 129, 1, 1, 3, 3, 3, 3, 7, 7, 7, 15, 15, 31, 63,
0, 63, 31, 15, 15, 7, 7, 7, 3, 3, 3, 3, 1, 1, 1, 129, 249, 249,
248, 248, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 3, 7, 15, 25, 29, 27, 51, 63, 54, 54, 118, 102, 124, 100, 108,
108, 108, 100, 108, 124, 124, 108, 108, 108, 100, 100, 118, 126, 62, 54, 55, 63,
27, 25, 28, 15, 7, 3, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0
};

// -----
// GLCD Picture name: Electromecanica.bmp
// GLCD Model: KS0108 128x64
// -----

const code char Electromecanica[1024] = {
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 128, 128, 128, 128, 128,
128, 128, 0, 0, 0, 0, 0, 128, 224, 32, 32, 32, 32, 32, 96, 224,
192, 128, 128, 128, 128, 128, 128, 128, 192, 112, 24, 206, 252, 48, 112, 224, 128, 128,
128, 128, 128, 192, 96, 112, 112, 96, 96, 96, 96, 224, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 128, 192,
96, 32, 32, 32, 32, 120, 76, 74, 50, 2, 2, 2, 2, 12, 8, 8,
15, 135, 128, 128, 128, 128, 64, 64, 64, 67, 34, 34, 34, 34, 19, 17, 16, 16,
16, 16, 8, 8, 8, 8, 8, 8, 136, 196, 124, 31, 15, 3, 96, 120, 78, 127,
79, 72, 72, 72, 72, 72, 72, 200, 8, 8, 200, 104, 24, 0, 0, 0, 0,

```

```

3, 7, 12, 24, 96, 192, 128, 0, 0, 0, 0, 0, 128, 192, 224, 176, 152, 204,
198, 227, 230, 246, 244, 252, 248, 192, 128, 128, 128, 192, 192, 192, 192, 128, 128,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 224, 248,
60, 44, 44, 47, 207, 9, 9, 14, 0, 0, 0, 0, 0, 0, 192, 32, 16,
16, 8, 4, 4, 2, 2, 1, 129, 128, 192, 64, 96, 32, 32, 48, 16, 16,
24, 8, 8, 12, 12, 4, 4, 4, 6, 6, 6, 6, 254, 14, 11, 136, 200,
104, 56, 24, 0, 0, 0, 0, 0, 0, 128, 192, 112, 28, 198, 241, 220, 231,
225, 224, 224, 224, 240, 240, 240, 248, 248, 248, 252, 252, 252, 255, 254, 254, 252,
254, 255, 255, 255, 255, 255, 255, 255, 255, 127, 255, 63, 127, 255, 255, 255, 255,
255, 255, 255, 255, 255, 255, 251, 251, 255, 126, 120, 120, 120, 240, 224, 0, 0,
0, 0, 0, 0, 0, 0, 255, 255, 9, 201, 201, 201, 249, 241, 192, 128, 0, 0,
0, 0, 0, 0, 0, 49, 81, 145, 17, 17, 17, 17, 17, 17, 17, 17, 17, 17, 17,
17, 17, 17, 17, 17, 17, 17, 17, 17, 17, 17, 17, 17, 17, 17, 17, 249,
13, 7, 255, 248, 254, 255, 249, 252, 252, 252, 254, 254, 254, 254, 255, 255, 255,
248, 254, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 127,
255, 63, 127, 255, 255, 255, 255, 255, 127, 63, 31, 15, 7, 3, 129, 192, 96,
63, 48, 224, 192, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255,
222, 222, 222, 223, 223, 254, 252, 0, 0, 0, 0, 0, 3, 127, 252, 255,
129, 129, 249, 249, 25, 57, 255, 126, 124, 124, 120, 240, 240, 224, 224, 224, 225,
226, 226, 228, 228, 232, 248, 248, 248, 248, 248, 248, 232, 232, 232, 200, 200, 200,
200, 136, 136, 136, 136, 8, 8, 8, 15, 254, 7, 7, 7, 7, 7, 7, 7, 255,
255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255,
255, 255, 255, 255, 63, 15, 3, 193, 96, 63, 16, 32, 64, 129, 3, 255, 192,
96, 48, 24, 12, 6, 3, 241, 248, 252, 254, 255, 255, 255, 255, 255, 255, 63,
63, 63, 63, 255, 255, 59, 251, 243, 131, 131, 255, 255, 127, 15, 7, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 7, 15, 24, 48, 63, 48, 48,
48, 16, 127, 255, 193, 131, 7, 255, 255, 15, 135, 255, 255, 135, 135, 247, 31,
31, 31, 63, 255, 255, 63, 31, 255, 255, 31, 31, 31, 255, 255, 255, 255, 255,
255, 255, 127, 63, 255, 255, 254, 255, 255, 255, 127, 31, 7, 255, 255, 255, 255,
255, 63, 63, 63, 63, 63, 255, 255, 31, 7, 129, 224, 112, 28, 7, 1, 0,
0, 0, 0, 0, 0, 1, 3, 0, 0, 0, 0, 0, 0, 255, 3, 3, 3,
3, 255, 255, 7, 131, 255, 48, 48, 48, 48, 48, 63, 63, 24, 15, 7, 1, 1,
1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 3, 3, 3, 1, 1, 1,
1, 1, 3, 15, 24, 24, 24, 24, 31, 15, 12, 14, 15, 15, 12, 12, 12, 63,
96, 96, 96, 96, 127, 63, 48, 24, 31, 255, 127, 31, 7, 193, 48, 12, 3,
127, 64, 64, 127, 63, 16, 16, 16, 16, 16, 31, 31, 12, 6, 3, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
1, 3, 3, 3, 3, 3, 3, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 127, 48, 12, 3, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
};

```