



# Instituto Politécnico Nacional

Escuela Superior de Ingeniería Mecánica y Eléctrica  
Sección de Estudios de Posgrado e Investigación  
Unidad Profesional Adolfo López Mateos Zacatenco

---

---

## Control de Torque de un Motor sin Escobillas de Corriente Directa

T E S I S

QUE PARA OBTENER EL GRADO DE: MAESTRO EN  
CIENCIAS EN INGENIERÍA MECÁNICA

P R E S E N T A

Ing. Adrián Esteban Mejía García

DIRIGEN:

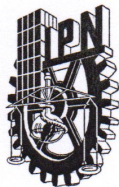
Dr. Juan José Muñoz Cesar

Dr. Juan Alfonso Beltrán Fernández

*México D. F. Agosto 2013*







**INSTITUTO POLITÉCNICO NACIONAL**  
**SECRETARÍA DE INVESTIGACIÓN Y POSGRADO**

SIP-14 BIS

**ACTA DE REVISIÓN DE TESIS**

En la Ciudad de México, D. F. siendo las 13:00 horas del día 31 del mes de Enero del 2013 se reunieron los miembros de la Comisión Revisora de la Tesis, designada por el Colegio de Profesores de Estudios de Posgrado e Investigación de E.S.I.M.E. para examinar la tesis titulada:

**"CONTROL DE TORQUE DE MOTOR SIN ESCOBILLAS DE CORRIENTE DIRECTA".**

Presentada por el alumno:

**MEJÍA**

Apellido paterno

**GARCÍA**

Apellido materno

**ADRÍAN ESTEBAN**

Nombre(s)

Con registro: 


B	0	9	1	7	3	4
---	---	---	---	---	---	---

aspirante de:

**MAESTRO EN CIENCIAS EN INGENIERIA MECÁNICA**

Después de intercambiar opiniones los miembros de la Comisión manifestaron **SU APROBACIÓN DE LA TESIS**, en virtud de que satisface los requisitos señalados por las disposiciones reglamentarias vigentes.

LA COMISIÓN REVISORA  
Directores de tesis

  
DR. JUAN ALFONSO BELTRÁN FERNÁNDEZ

Presidente

  
DR. JUAN JOSÉ MUÑOZ CÉSAR

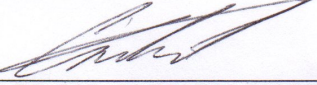
Segundo Vocal


  
DR. GUILLERMO MANUEL URRIOLAGOITIA CALDERÓN

Tercer Vocal


  
DR. JUAN JOSÉ MUÑOZ CÉSAR

Secretario

  
DR. CARLOS TORRES TORRES

  
DR. LUIS HÉCTOR HERNÁNDEZ GÓMEZ

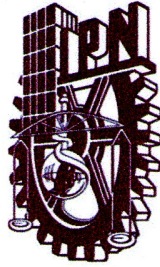
EL PRESIDENTE DEL COLEGIO

  
DR. MAURO ALBERTO ENCISO AGUILAR  
SECRETARÍA DE INVESTIGACIÓN Y POSGRADO









## ***Instituto Politécnico Nacional***

### ***SECRETARÍA DE INVESTIGACIÓN Y POSGRADO***

#### ***CARTA DE SESIÓN DE DERECHOS***

En la ciudad de México el día 1° del mes Febrero del año 2013, el que suscribe, Ing. Mejía García Adrián Esteban, alumno del Programa de Maestría en Ciencias en Ingeniería Mecánica con número de registro B091734, adscrito a la Sección de Estudios de Posgrado e Investigación de la ESIME Unidad Zacatenco, manifiesta que es autor intelectual del presente trabajo de Tesis bajo la dirección del Dr. Juan Alfonso Beltrán Fernández y Dr. Juan José Muñoz Cesar y cede los derechos del trabajo intitulado CONTROL DE TORQUE DE MOTOR SIN ESCOBILLAS DE CORRIENTE DIRECTA., al Instituto Politécnico Nacional para su difusión, con fines académicos y de investigación.

Los usuarios de la información no deben reproducir el contenido textual, gráficas o datos del trabajo sin el permiso expreso del autor y/o directores de trabajo. Este puede ser obtenido escribiendo a la dirección de correo electrónico [aemejiag@ipn.mx](mailto:aemejiag@ipn.mx) y si el permiso se otorga, el usuario deberá dar el agradecimiento correspondiente y citar la fuente del mismo.

Ing. Adrián Esteban Mejía García

Nombre y firma



# Dedicatoria

A mi papá Isidoro Mejía López, a mi mamá Misnereyda García Gonzales  
y a mi hermano Javier Alejandro Mejía García, por su aliento,  
comprensión, consejos y por ese ejemplo de lucha ante la adversidad.

A mis amigos Luis, Alejandra, Ivan y Mauricio, gracias por su ayuda,  
consejos, críticas y observaciones que me han animado a mejorar día con  
día.

Un especial agradecimiento al Dr. Luis Hector Hernández Gómez y, al  
profesor y amigo Dr. Juan José Muñoz Cesar por su apoyo y paciencia  
para que este trabajo halla sido concluido.

Gracias a todos por estar a mi lado cada instante.

Sinceramente  
Adrián Esteban Mejía García



# Agradecimientos

Al:

## Instituto Politécnico Nacional

Escuela Superior de Ingeniería Mecánica y Eléctrica  
Sección de Estudios de Posgrado e Investigación

## Consejo Nacional de Ciencia y Tecnología

### **A mis Asesores:**

Dr. Juan Alfonso Beltrán Fernández  
Dr. Juan José Muñoz Cesar

### **A mis Amigos:**

M. en C. Mauricio Pérez Romero  
Ing. Godoy Rodríguez Luis  
Ing. Hernández Velázquez Ivan  
Ing. Corona Calderón Maribel Alejandra



# Resumen

En este trabajo se presenta el desarrollo e implementación de un control de par para un motor sin escobillas de tres fases de corriente directa, también conocido como motor BLDC (del inglés *Brushless Direct Current*).

Las pruebas se realizaron utilizando un motor de la marca LINIX, modelo 45ZWN24-40, a un voltaje de 24 V RMS, con un par de 0.924 Nm, 2.3 A y una velocidad nominal de 4000 rpm.

El control se realiza partiendo de una secuencia de seis pasos, que permite una conducción de 120° entre cada fase del motor, utilizando una conmutación unipolar complementaria, la cual permite un mejor aprovechamiento del par.

Para controlar el par se actúa primeramente sobre la variable velocidad haciendo uso de un algoritmo computacional tipo Proporcional Integral Derivativo (PID), posteriormente se realiza un control Proporcional Integral (PI) para el par, el cual funciona a partir de la salida generada por la primera estrategia aplicada; logrando así, regular el flujo de corriente que pasa a través de cada una de las fases del motor.

La programación se realizó sobre la plataforma de Freescale CodeWarrior, programando en lenguaje C, y Procesador Experto. El algoritmo generado se implementó en un Controlador Digital de Señales modelo 56F8006, donde los resultados generados por parte de los controladores programados se obtuvieron a través de FreeMaster, que es una interfaz gráfica de Freescale; y la modulación de ancho de pulso se muestra utilizando una tarjeta de adquisición de datos modelo 6211 de National Instruments.



Para un funcionamiento correcto de los controladores se establecieron las constantes proporcional, integral y derivativa con valores de 1.20 para  $k_p$ , 3 para  $k_i$ , y 0.04 para  $k_d$ . Con estas constantes se observó un buen funcionamiento por parte del controlador de velocidad, lo que permitió establecer las constantes para el controlador de corriente las cuales fueron de 0.05 para la constante proporcional  $k_{pc}$  y 0.15 para la constante integral  $k_{ic}$ , las cuales presentaron mejor respuesta debido a que el error en el lazo de control de par, resultó ser aproximadamente de un 6 %.

# Abstract

This thesis presents the design and implementation of a torque control for a 3-Phase brushless direct current (BLDC) motor.

The testing was performed using a motor model Linux 45ZWN24-40 at a voltage of 24 V RMS with a torque of 0.924 Nm, 2.3 A and a nominal speed of 4000 RPM.

The control is performed starting from a 6-step sequence, which allows  $120^\circ$  conduction between each motor phase. It uses a complementary unipolar switching that enables a better utilization of the torque.

The speed variable has to be governed firstly to control the torque. The regulation is done by utilizing a Proportional Integral Derivative (PID) computational algorithm. Subsequently, a Proportional Integral (PI) control is executed for the torque. The torque operates from the output generated by the first strategy applied. Therefore, it regulates the current flow passing through each of the motor phases.

Programming is performed on Freescale's CodeWarrior -using C language- and Processor Expert platforms. The generated algorithm was implemented on a 56F8006 Digital Signal Controller, where the results generated by the controllers set were obtained through Freescale's graphical interface FreeMaster. Besides, the pulse-width modulation is shown using a data acquisition board, model 6211 from National Instruments.

For proper operation of the controllers, the proportional (P), integral (I) and derivative (D) constants (K) are set to the following values: 1.20 for  $k_p$ , 3 for  $k_i$ , and 0.04 for  $k_d$ . To conclude, these constants show the correct performance of the speed controller. Thereby, it allows to set the current controller constants to 0.05 for the proportional constant  $k_{pc}$  and 0.15 for integral constant  $k_{ic}$ . Both constants had a better response because the error in the torque control loop turned out to be about six percent.

# Índice General

Resumen	XI
Índice General	XV
Índice de Figuras	XIX
Índice de Tablas	XXIII
Objetivo	XXV
Introducción	XXVII
Justificación	XXIX
<b>1. Antecedentes Generales</b>	<b>1</b>
1.1. Estado del Arte . . . . .	1
1.2. Planteamiento del Problema . . . . .	4
1.3. Aportaciones . . . . .	6
1.4. Metodología de Desarrollo . . . . .	6
1.5. Sumario . . . . .	7
<b>2. Control de Motor BLDC</b>	<b>9</b>
2.1. Clasificación de Máquinas Eléctricas . . . . .	9
2.2. Motor sin Escobillas ( <i>Brushless</i> ) . . . . .	11
2.2.1. Modelo Matemático del Motor . . . . .	15
2.3. Control Trapezoidal . . . . .	18
2.4. Control de Torque y Velocidad . . . . .	22
2.5. Esquema de Control . . . . .	24

2.6.	Modulación de Ancho de Pulso (PWM) . . . . .	26
2.6.1.	Conmutación Bipolar . . . . .	30
2.6.2.	Conmutación unipolar . . . . .	31
2.6.3.	Control de voltaje en inversores de 3 Fases . . . . .	34
2.6.4.	Efecto del tiempo muerto en Inversores controlados por PWM . . . . .	36
2.7.	Control PID . . . . .	41
2.7.1.	Respuesta proporcional . . . . .	41
2.7.2.	Respuesta Integral . . . . .	42
2.7.3.	Respuesta Derivativa . . . . .	42
2.7.4.	Control PID Digital . . . . .	42
2.8.	Sumario . . . . .	47
<b>3.</b>	<b>Estrategia de Control para Motor BLDC</b>	<b>49</b>
3.1.	Introducción . . . . .	49
3.2.	Conmutación Trapezoidal . . . . .	50
3.3.	Microcontrolador MC56F8002/MC56F8006 . . . . .	54
3.3.1.	Diagrama Bloques del Sistema . . . . .	56
3.4.	Convertidor Analógico-Digital (ADC) . . . . .	57
3.4.1.	Estatus y Control de Registro 1A y 1B (ADCn_ADCSC1A y ADCn_ADCSC1B)	58
3.4.2.	Resultados de los Registros de Datos A y B (ADCn_ADCRA y ADCn_ADCRB) . . . . .	58
3.4.3.	Descripción Funcional . . . . .	59
3.5.	Modulación de Ancho de Pulso (PWM) . . . . .	63
3.5.1.	Características . . . . .	64
3.5.2.	Modos de operación . . . . .	65
3.5.3.	Descripción de Funcionamiento . . . . .	65
3.5.4.	Operación de Canal Independiente o Complementario . . . . .	69
3.5.5.	Generación de Tiempo Muerto . . . . .	71
3.5.6.	Salida PWM Asimétrica . . . . .	73
3.5.7.	Colocación de Borde Variable PWM de Salida . . . . .	74
3.5.8.	Polaridad de PWM . . . . .	74
3.5.9.	Software de Control de Salida . . . . .	75
3.5.10.	Generador de Carga . . . . .	77
3.6.	Programación del Control . . . . .	77

3.6.1. Bit de entrada/salida de propósito general . . . . .	80
3.6.2. Interrupciones Externas . . . . .	81
3.6.3. Contador libre . . . . .	82
3.6.4. Convertidor Analógico–Digital . . . . .	83
3.6.5. Modulación de Ancho de Pulso Complementario . . . . .	84
3.6.6. Detalle del Algoritmo de Control . . . . .	85
3.7. Características del Control . . . . .	90
3.8. Sumario . . . . .	91
<b>4. Pruebas y Resultados</b>	<b>93</b>
4.1. Técnica de 6 pasos . . . . .	93
4.2. Control de Velocidad . . . . .	94
4.3. Control en Cascada . . . . .	97
4.4. Sumario . . . . .	102
4.5. Conclusiones . . . . .	103
4.6. Sugerencias de Trabajos a Futuro . . . . .	104
<b>Referencias</b>	<b>107</b>
<b>Anexos</b>	<b>113</b>
<b>A. Esquema de la Etapa de Potencia</b>	<b>115</b>
A.1. Descripción Operacional . . . . .	116
A.2. Consideraciones de Diseño . . . . .	118
A.2.1. Puente de Tres Fases . . . . .	118
A.2.2. Corriente de Retroalimentación y Corriente de Bus . . . . .	120
A.2.3. Sobrecorriente, Bajo Voltaje, y otras Funciones de Seguridad . . . . .	121
A.2.4. Detección de Corriente de Fase . . . . .	122
A.2.5. Fuentes de Alimentación y Voltaje de Referencia . . . . .	123
<b>B. Tarjeta Controladora MC56F8006</b>	<b>127</b>
B.1. Listado de Partes . . . . .	127
B.2. Esquemático de la Tarjeta Controladora MC56F8006 para el Control de Motores BLDC/PMSM . . . . .	128
B.3. Esquemático . . . . .	128

B.4. Diseño de la Tarjeta Impresa (PCB) del Controlador MC56F8006 para el Control de Motores BLDC/PMSM . . . . .	129
<b>C. Etapa de Potencia</b>	<b>131</b>
C.1. Lista de Materiales . . . . .	131
<b>D. Esquemáticos de Etapa de Potencia</b>	<b>135</b>
<b>E. Programación del Algoritmo</b>	<b>147</b>
E.1. Declaración de Librerías . . . . .	147
E.2. Declaración de variables . . . . .	148
E.3. Registros para comunicación . . . . .	149
E.4. Detección de Posición del Motor . . . . .	152
E.5. Conmutación . . . . .	152
E.6. Calculo de velocidad . . . . .	156
E.7. Control PID para Velocidad . . . . .	158
E.8. Control PI para Corriente . . . . .	158



# Índice de Figuras

1.1. Desarrollo del trabajo de control de torque para motor de trifásico de corriente directa . . . . .	7
2.1. Clasificación de las máquinas eléctricas. . . . .	10
2.2. Sección transversal de un motor sin escobillas. . . . .	12
2.3. Rotor colocado exteriormente e interiormente. . . . .	13
2.4. Sistema de Tres Fases. . . . .	13
2.5. Fuerza contraelectromotriz del motor sin escobillas y Flujo Magnético. . . . .	14
2.6. Topología del motor. . . . .	15
2.7. Inversor de 3 fases para motor brushless. . . . .	19
2.8. Secuencia de conmutación. . . . .	20
2.9. Representación de un motor sin escobillas con sensores de efecto Hall. Se pueden apreciar los 6 sectores definidos por la combinación de señales Hall, así como los vectores de excitación de cada sector para movimientos en sentido de las manecillas de reloj. . . . .	21
2.10. Operación del motor en los cuatro cuadrantes. . . . .	24
2.11. Diagrama de control. . . . .	26
2.12. Formas de ondas eléctricas en las dos fases de operación y par rizado. . . . .	26
2.13. Formas de onda de voltaje y corriente de salida en un recortador genérico. . . . .	28
2.14. Formas de onda de voltaje y corriente de salida, en el recortador con una frecuencia de conmutación similar al de la figura 2.13. . . . .	29
2.15. Modulación por ancho de pulsos bipolar. (a) Referencia cuadrada y portadora triangular. (b) La salida es $+V_{CC}$ cuando $v_{cuadrada} > v_{tri}$ y es $-V_{CC}$ cuando $v_{cuadrada} < v_{tri}$ . . . . .	31

2.16. Puente inversor completo, cuando $S_1$ y $S_2$ están cerrados el voltaje de salida $v_O$ es igual a $V_{CC}$ ; y cuando $S_3$ y $S_4$ están cerrados el voltaje de salida $v_O$ es $-V_{CC}$ . . . . .	32
2.17. (a) Puente convertidor de onda completa para PWM unipolar. (b) Señales de referencia y portadora. (c) Tensiones eléctricas $v_a$ y $v_b$ del puente. (d) Tensión eléctrica de salida. . . . .	33
2.18. PWM unipolar con interruptores de alta y baja frecuencia. (a) Señales de referencia y de control. (b) $v_a$ . (c) $v_b$ . (d) Salida $v_a - v_b$ . . . . .	34
2.19. Inversor trifásico. . . . .	35
2.20. (a) Ondas portadora y de referencia para un control PWM. (b) Formas de onda de salida. . . . .	36
2.21. Configuración básica de PWM (solo se muestra una fase del inversor). . . .	37
2.22. Señales de control de retraso y voltaje de salida . . . . .	38
2.23. Relación entre el voltaje de fase $v_a(t)$ y la señal de error $e(t)$ para (a) $i_a > 0$ ; (b) $i_a < 0$ . . . . .	39
2.24. Diagrama de bloques para un inversor PWM con un circuito de tiempo de retardo . . . . .	39
2.25. Diagrama a bloques equivalente de la figura 2.24 . . . . .	40
2.26. Diagrama que muestra la función $f(hT)$ . . . . .	45
2.27. Diagrama a bloques de la realización del esquema de control PID en la forma de velocidad . . . . .	47
3.1. Motor sin escobillas y etapa de potencia. . . . .	51
3.2. Relación entre fuerza contraelectromotriz, corriente, y diferencia de potencial eléctrico para las tres fases del motor BLDC. . . . .	52
3.3. Desactivación de transistores en conmutación unipolar complementaria. . .	54
3.4. Diagrama a bloques del Sistema. . . . .	57
3.5. Resultados de los Registros de Datos (ADCn_ADCRA y ADCn_ADCRB) .	60
3.6. Ejemplo de segmento de código que describe la operación de la función de la selección de muestreo. . . . .	61
3.7. PWM de Alineación Central . . . . .	66
3.8. PWM de Alineación al Borde . . . . .	67
3.9. Periodo PWM de Alineación Central . . . . .	68
3.10. Periodo PWM de Alineación al Borde . . . . .	68

3.11. Ancho de Pulso PWM de Centro Alineado . . . . .	69
3.12. Ancho de Pulso PWM de Alineación al Borde. . . . .	69
3.13. Canales pares complementarios. . . . .	70
3.14. Distorsión del Tiempo Muerto . . . . .	73
3.15. Forma de onda asimétrica en el cambio de fase del PWM de salida . . . . .	74
3.16. Inserción de forma de onda de borde variable al cambio de fase del PWM . . . . .	75
3.17. Polaridad PWM. . . . .	76
3.18. Software de Control de Salida en Modo Complementario . . . . .	78
3.19. Ambiente de Programación en Procesador Experto . . . . .	79
3.20. Componentes de los periféricos de control . . . . .	80
3.21. Configuración del bit de propósito general . . . . .	81
3.22. Configuración de Interrupciones . . . . .	82
3.23. Configuración del Contador . . . . .	83
3.24. Componente ADC . . . . .	84
3.25. Diagrama de flujo del algoritmo de control . . . . .	86
4.1. Generación de 6 PWM's . . . . .	94
4.2. Primera prueba realizada al controlador de velocidad . . . . .	95
4.3. Pruebas realizadas con $k_p = 0,80$ , $k_d = 0,03$ y $k_i = 5,00$ . . . . .	95
4.4. Pruebas realizadas con $k_p = 1,20$ , $k_d = 0,04$ y $k_i = 3,00$ . . . . .	96
4.5. Pruebas realizadas con $k_p = 1,80$ , $k_d = 0,06$ y $k_i = 4,25$ . . . . .	97
4.6. Ventana de control . . . . .	97
4.7. Control en cascada donde las constantes del PID tienen valores de 2.00 (proporcional), 5.00 (integral) y 0.05 (derivativa) . . . . .	98
4.8. Pruebas realizadas con $k_{pc} = 0.50$ y $k_{ic} = 0.01$ . . . . .	99
4.9. Pruebas realizadas con $k_{pc} = 0.05$ y $k_{ic} = 1.50$ . . . . .	99
4.10. Pruebas realizadas con $k_{pc} = 0.01$ y $k_{ic} = 2.50$ . . . . .	100
4.11. Gráfica de torque y corriente, con cambios en las constantes del PID . . . . .	101
4.12. Gráfica de torque y corriente sin cambios en las contantes integral y pro- porcional del controlador de corriente . . . . .	101
4.13. Control en cascada donde las constantes del PID con $k_p=1.20$ , $k_i=3.00$ y $k_d=0.04$ . . . . .	102
A.1. Configuración del Sistema. . . . .	115
A.2. Diagrama a Bloques . . . . .	117

A.3. Distribución de los conectores J7 y J8. . . . .	118
A.4. Fase de Salida . . . . .	119
A.5. Bus de retroalimentación . . . . .	121
A.6. Detección de Corriente de Fase . . . . .	123
A.7. Circuito de entrada del Bus de CD . . . . .	124
A.8. Fuente de Alimentación de 5V . . . . .	124
A.9. Fuente de Alimentación de 3.3V . . . . .	125
A.10. Interfaz CAN . . . . .	125
B.1. Conectores y Cabeceras ( <i>headers</i> ) de la Tarjeta Hija . . . . .	128
B.2. Capa superior . . . . .	129
B.3. Capa inferior . . . . .	129
B.4. Controlador y Cabeceras . . . . .	130
D.1. Esquema general de la Etapa de Potencia . . . . .	135
D.2. Fase para el sensado de corriente . . . . .	136
D.3. Fase para sensar la fuerza contraelectromotriz . . . . .	137
D.4. Conectores y otros circuitos para señales de sensores de efecto hall . . . . .	138
D.5. Conectores, interruptores e indicadores de señales PWM . . . . .	139
D.6. Conectores para la tarjeta de control . . . . .	140
D.7. Etapa de control y direccionamiento de señales hacia los MOSFET . . . . .	141
D.8. Circuito para comunicación serial . . . . .	142
D.9. Circuito de Potencia . . . . .	143
D.10. Fuentes de alimentación . . . . .	144
D.11. Circuito impreso de la etapa de potencia en la parte superior . . . . .	145
D.12. Circuito impreso de la etapa de potencia en la parte inferior . . . . .	146

# Índice de Tablas

3.1. Descripción de los campos de los registros ADCn_ADCS1A/B . . . . .	59
3.2. Modos de PWM cuando la operación restringida . . . . .	65
B.1. Lista de Partes . . . . .	127
C.1. Lista de Materiales para la Etapa de Potencia . . . . .	131



# Objetivo

Desarrollar un control de par, empleando la tecnología de los controladores digitales de señales (DSC, por sus siglas en inglés, *Digital Signal Controller*) de última generación que permita establecer un control en cascada para la manipulación del flujo de corriente en un motor sin escobillas de corriente directa.

Derivados del objetivo general se plantean los siguientes objetivos específicos:

- Recopilar la información necesaria, que proporcione las bases suficientes, para establecer la conmutación del motor.
- Plantear la información pertinente, para el entendimiento del problema, la cual dará cuenta de la extensión de los parámetros a considerar, tales como: las variables relacionadas con la rotación de la máquina trifásica, estrategia de control, así como la modulación por ancho de pulso.
- Establecer los lazos control necesarios, para actuar sobre la velocidad y el par del motor BLDC, partiendo del análisis de los registros adecuados a utilizar en el controlador digital de señales.
- Realizar la programación necesaria para afectar las señales de voltaje y corriente, que son adecuadas de acuerdo al algoritmo seleccionado para cumplir con el propósito de regular el par a través de la corriente.





# Introducción

El presente trabajo trata sobre el control de par para un motor sin escobillas trifásico de corriente directa conocido como *Brushless* o BLDC ( del ingles, *Brushless Direct Current*). La investigación aborda los antecedentes generales relacionados a las investigaciones que se han hecho con motores síncronos de imanes permanentes donde se encuentra clasificada la máquina con la que se trabaja en esta tesis, abordando los tipos de control que existen así como las maneras de cerrar el lazo.

A partir de lo anterior se encuentra que el motor sin escobillas trabaja con una fuerza contraelectromotriz con una forma trapezoidal, lo que permite encontrar que el tipo de conmutación a utilizar es conocida como 6 pasos, derivando entonces en el estudio de una etapa de potencia compuesta por seis transistores donde la activación de cada uno de estos es en pares accionado de esta manera dos bobinas al mismo tiempo. Respecto a la retroalimentación esta puede ser con o sin sensor donde por conveniencia debido a que el motor adquirido trae integrado 3 sensores de efecto hall, la manera de cerrar el lazo es con sensor.

Los transistores son activados con 6 señales conocidas como modulación por ancho de pulso (PWM, del ingles *Pulse Width Modulator*), con esto se aborda también una explicación breve acerca de la generación de este tipo de señal así como sus variantes que existen incluyendo la manera que existe para mandar dos señales de este tipo al mismo tiempo.

Al cerrar el lazo para realizar un control de par, primero se encuentra que es necesario programar un algoritmo que regule la velocidad y posteriormente uno que limite el flujo de corriente hacia el estator, variable que permite manipular el par de manera indirecta.

Una vez entendida la estructura que debe tener el algoritmo para el control de par, se procede a describir las características del microcontrolador a utilizar, que es un procesador

digital de señales (DSP, *Digital Signal Processing*) de la familia 56F80xx de Freescale, el cual a través del uso de un entorno de diseño conocido como Procesador Experto hallado en la plataforma de CodeWarrior, es posible la programación de 6 PWM con la generación de tiempo muerto para evitar algún tipo de cortocircuito así como la frecuencia de 20 a 26 kHz que evita ruido en la máquina derivado de la conmutación. Aunado a esto se aborda una explicación de como el procesador genera la modulación de ancho de pulso y la conversión analógico-digital, necesaria para captar el flujo de corriente que atraviesa cada una de las fases del motor.

Con un algoritmo que debe ser programado, convenientemente se aborda una redacción acerca de las funciones a utilizar y que permite llevar a cabo el programa ya mencionado, derivando en un análisis de pruebas y resultados, partiendo de los 6 PWM, seguido de un estudio del control de velocidad elaborado considerando los parámetros mas adecuados para las constantes de un PID, y posteriormente realizando el mismo estudio el lazo interior relacionado con el flujo de corriente.

Finalmente se presenta una explicación de los resultados en el entorno de desarrollo gráfico obtenidos, derivados de las pruebas elaboradas utilizando el control en cascada que se realizo concluyendo cuales son las contantes mas idóneas para el trabajo elaborado.

# Justificación

Los motores sin escobillas debido a su gran popularidad, son usados en la industria Automotriz, Aeroespacial, Medica, en la Instrumentación y Automatización de equipo industrial, ganando terreno sobre los motores de corriente directa con escobillas [Yedamale (2003), Lee (2004) y Wiberg (2003)]. Las ventajas que presentan estos motores, son: menor tamaño, mayor potencia, menor desgaste, y un amplio rango de control, que propicia que sean ampliamente estudiados [Yan (2010), y Ohm (1999)]. Existen dos tipos de motores que funcionan a base de conmutación electrónica: los motores sin escobillas (BLDC, por sus siglas en ingles) y los motores síncronos de imán permanente (PMSM por sus siglas en ingles, *Permanent Magnet Synchronous Motor*) [Colton (2008)]. Los primeros hacen uso de la fuerza contraelectromotriz para garantizar un buen control, a partir de la detección del cruce por cero de esta o haciendo uso de sensores, y los segundos pueden ser controlados con diferentes técnicas desde el control directo de par [Dalib (2007)], control vectorial difuso [César (2005), Kaliappan (2010), Ömer AYDOĞDU (2010), Shannon (2001) y Soter (2005)] hasta el uso de redes neuronales y algoritmos genéticos [Urasaki (2006) y XiaChangliang (2000)]. Es por esto que los motores BLDC son más fáciles de controlar que los PMSM, debido a que estos necesitan un mayor control sobre la corriente [Lee (2004), Margarita (2004) y Mevey (2006)].

La técnica de control por retroalimentación de la fuerza contraelectromotriz (*back-EMF*) puede ser con control trapezoidal y utilizando retroalimentación con o sin sensor, donde este es usado con ambas técnicas, y las cuales presentan el mismo par rizado necesario para el motor [Prokop (2005), Ozkut (2008), Sozer (1998) y Su (2001)].

El control de torque es importante en ciertas aplicaciones, como en los vehículos eléctricos donde se usa en la tracción, en aplicaciones como esta, el sistema puede ser a lazo abierto, donde el usuario proporciona el torque de entrada [Salgues (2008), Atallah (2003) y Chu.L.C (2001)], y cuando se desea que el sistema de tracción eléctrica tienda a ser más robusto, el lazo se puede cerrar agregando un control de velocidad [Dilmi (2006)]. Otra de sus aplicaciones se da en los robots manipuladores, donde estos requieren actuadores con un alto torque a bajas velocidades, y las técnicas de control aplicadas van en función del tipo de actuador a utilizar, los motores sin escobillas, debido a su reducido mantenimiento tienden a ser usados comúnmente en el efector final del manipulador [Taghirad (2001) y Ellis (2003)].

# Capítulo 1

## Antecedentes Generales

### 1.1. Estado del Arte

Las técnicas de control para motores BLDC se pueden clasificar según el tipo de conmutación implementada. Las más utilizadas actualmente son: Conmutación trapezoidal (*6 steps mode*) [Akin (2000), Condit (2007), Elevich (2005) y Ficapal (2003)], Conmutación Sinusoidal (SPWM por sus siglas en ingles, *Sine Pulse Width Modulation*) [Hobetler (1992) y Vas (1998)], Control del Campo Orientado (FOC por sus siglas en ingles, *Field Orientated Control*) [Texas (1998)], y Modulación del Ancho de Pulso del Espacio Vectorial (SVPWM por sus siglas en ingles, *Space Vector Pulse Width Module*) [Michael (2010) y Yedamale (2003)], de donde la técnica de control varia dependiendo del tipo de motor sin escobillas con el que se trabaje [Tolrá (2003)].

Un motor sin escobillas gira por la interacción que existe entre el rotor de imanes permanentes con un campo magnético generado cuando circula un voltaje de corriente directa a través de las bobinas del estator [Electronics (2006) y Elliott (2004)]. La forma más sencilla de controlar la rotación de un BLDC es a través de la técnica de los 6 pasos de conmutación (a veces llamada 60, 120 grados de control, como lo muestra Freescale (2009a), Kettle (1998), King (2008) y Singh (2003) ). La conmutación es posible con un inversor de 3 fases, que puede ser construido con transistores MOSFET, IGBT o simplemente de transistores bipolares [Viramontes (2010), Dixon (2006) y Minciunescu (1998)]. En este tipo de control cada transistor conduce durante  $120^\circ$  [Rashid (1992)]. La posición del rotor se determina usando elementos de retroalimentación, tales como sensores de efecto hall y sensores de posición angular (*encoders*), e incluso una combinación de am-

bos [Mani (2003)]. Esta técnica de control retroalimentado es buena, pero frecuentemente resulta ser más cara [Jani (1999) y Ellis (2003)].

Para aplicaciones que requieren que el motor opere con un torque específico sin tomar en cuenta como prioridad máxima la velocidad, un controlador de corriente puede ser usado, ya que el torque es directamente proporcional a la corriente [Im (2010)]. En este modo, la velocidad se mantendrá en el valor de la señal de referencia para todas las cargas hasta el punto en el que la corriente en la armadura es máxima. Si aumenta la carga del esfuerzo de torsión, la velocidad disminuirá por que dejara de fluir corriente hacia la armadura. Por el contrario, si la carga provoca que la velocidad se incremente por arriba del valor de referencia, la corriente se regresara automáticamente, provocando que el motor se comporte como un generador, regresando energía a la red. En este tipo de control la señal de salida de un controlador de velocidad es la señal de entrada al controlador de torque [Hattori (2002)].

Debido a lo costoso que resulta cerrar el lazo de control con encoders [Xu (2008)]. Existen otros métodos de medición que permiten retroalimentar el sistema, tales como las mediciones a través de sensores DC transformadores de corriente (DCCT por sus siglas en ingles, *DC Current Transformer*) o sensores de derivación de la corriente (OSDC por sus siglas en ingles, *One-Shunt Current Detection*) según Awadallah (2009), Karthikeyan (2010) y López (2010). Ambas técnicas de medición estiman la posición y velocidad, usando las mediciones de corriente y voltaje aplicados en el ciclo de modulación de ancho de pulso (PWM por sus siglas en ingles, *Pulse Width Module*)[Gamazo (2010)]. Estos métodos emplean un filtro Kalman, un observador y una matriz de transición; sin embargo esto provoca que el numero de cálculos sea mayor, provocando un mayor numero de instrucciones por segundo del CPU [King (2008)].

La diferencia que existe entre DCCT y OSDC, es que la primera usa dos sensores de corriente conectados en el inversor, que sirven como elementos de retroalimentación, y en la segunda se usa una resistencia de derivación colocada también en la etapa de inversión; ambos métodos han demostrado tener éxito en el control sin sensores acoplados a alguna parte del motor, demostrando ser eficaces y semejantes al control con sensores [Xu (2008)]. Otra técnica de control es usar la fuerza contraelectromotriz de cruce por cero, para la estimación de la posición del motor, haciendo uso de los parámetros del



motor, y de voltajes y corrientes que se manejan [Kumar (2009)]. Por lo tanto la fuerza contraelectromotriz de cruce por cero es usada para sincronizar la conmutación de los 6 pasos con la posición del rotor en el orden en que se consiga el desarrollo de un torque constante del motor sin escobillas.

El panorama general para usar esta técnica de control se puede resumir en cuatro pasos: 1) Generación de modulación de ancho pulso (PWM, por sus siglas en ingles *Pulse Width Modulation*), 2) Detección del cruce por cero de la fuerza contraelectromotriz, 3) Control de conmutación, 4) Control de velocidad [Lawler (2004)].

El primer punto se refiere a las señales que activaran los transistores del inversor, la generación de los voltajes de modulación de ancho de pulso que se pueden dar en modo complementario y en modo independiente. En el modo complementario, los interruptores de una fase son operados complementariamente: si un interruptor esta activado, el otro está desactivado. Este modo necesita la inserción de un tiempo muerto entre los interruptores superior e inferior para evitar cualquier corto circuito en la fase. La conmutación complementaria puede incorporarse de manera unipolar o bipolar. La conmutación unipolar proporciona bajas perdidas en la conmutación y rizado en la corriente. Sin embargo, para el uso de una fuerza contraelectromotriz, la conmutación bipolar es la mejor elección porque esto permite un ciclo útil en el rango del 50 % al 100 %. En modo independiente los interruptores superior e inferior son operados independientemente sobre un periodo de conmutación, pero en este modo de operación se consigue un bajo torque y el control de velocidad es muy difícil [Freescale (2009a) y Frogal (2009)].

El cruce por cero de la fuerza electromotriz de las tres fases se comprueba usando comparadores. Las salidas de los comparadores se llevan a las entradas de decodificadores. Donde un bloque de filtración digital se utiliza para filtrar el cruce de cero de las señales, en Grasblum (2008). El control de velocidad y corriente se hace con un control PID para el primero y un control PI para el segundo [Ohm (1999)].

## 1.2. Planteamiento del Problema

Los sistemas eléctricos que emplean motores de corriente directa con escobillas como elemento final de control, que son necesarios para regular la fuerza (ejemplo de esto se da en la robótica en los actuadores finales, en el área automotriz, en el aeromodelismo, etc.,) requieren de una inversión en mantenimiento debido al desgaste mecánico que se genera en las escobillas. Los motores BLDC eliminan el costo en mantenimiento, pues no utilizan escobillas para su conmutación, sin embargo requieren de un algoritmo computacional para la conmutación y el control de par.

Se ha revisado en distintos trabajos citados anteriormente y en observaciones experimentales que cuando una máquina eléctrica rotatoria necesita de una mayor fuerza para mover una carga, es necesario reducir la velocidad de esta para que el par sea mayor. Un control de velocidad puede controlar el par de manera indirecta pues si este detecta que el motor necesita mover una carga, el controlador calcula que es indispensable inyectar más energía al sistema, lo cual ocasiona que la máquina mantenga su velocidad y a su vez pueda mover esa carga. El problema consecuente de tener solo un lazo de control, que en este caso, solo es para la velocidad es que, si el motor necesita manipular esa carga con un par determinado, con este tipo de arquitectura de control será difícil, pues la variable manipulada es la velocidad. Debido a que la velocidad es inversamente proporcional al par y que además la variable controlada es el voltaje que a su vez depende de otra variable que es la corriente, y en consecuencia todas estas variables están relacionadas, y que en lazo de control al aumentar el voltaje en el sistema eléctrico aumenta la velocidad, y al incrementar la corriente incrementa el par, es evidente que la arquitectura de control debe ser de tipo cascada, donde la variable interna y por lo tanto, la más rápida sea la corriente y la externa sea el voltaje.

Un motor sin escobillas de corriente directa es empleado en sustitución de los motores con escobillas, en aplicaciones donde se requiere una mayor: velocidad, par de arranque, y potencia; proporcionando ventajas en costo y mantenimiento. Aplicaciones tales como la tracción de automóviles eléctricos, vehículos no tripulados, en la robótica, exoesqueletos, entre otros, hacen uso de motores BLDC, donde además de mantener una velocidad es de suma importancia controlar el par con el que se mantenga funcionando este efector final.

El no contar con un control de par apropiado en los motores trifásicos de corriente directa implica que muchos mecanismos incrementen su costo, pues en busca de una solución rápida serían nuevamente sustituidos por otro tipo de actuadores eléctricos que a largo plazo propiciarían que fuesen sustituidos debido a un mantenimiento constante, limitando su funcionamiento y en consecuencia dejando al mecanismo completamente en desuso, pues generaría una gran cantidad de problemas al no poder ser utilizados correctamente ya que no cumplirían con las exigencias del usuario final.

Hasta este punto se ha hablado de cómo controlar el par y la importancia que este tiene, pero también es relevante mencionar que en un motor sin escobillas existen otros aspectos que influyen para la activación del motor, pues éste, como se mencionó en un principio requiere de una conmutación, es decir que su funcionamiento requiere de un par de señales en cada una de las bobinas, y como el motor cuenta con 3 bobinas, el par de señales deben permitir la combinación de un par de bobinas, es por esta razón que debe de existir una conmutación que derive en una secuencia de 6 pasos que genere una rotación eléctrica, y dependiendo del número de polos que tenga el motor, cierto número de rotaciones eléctricas permitirán una rotación mecánica. El tipo de señales que deben generarse para la conmutación son de tipo modulación de ancho de pulso (PWM, por sus siglas en inglés, *Pulse Width Modulation*). Estas señales deben activar un par de interruptores (elementos semiconductores) los cuales admiten la conmutación y por ende la activación del motor.

Derivado de todo lo anterior se puede decir que: la manera de generar un control de par es a través de un control en cascada, donde el lazo externo sea un control de velocidad y el interno sea uno de corriente, ambos algoritmos se deben programar a través de una plataforma de desarrollo computacional que genere el código necesario para poder ser implementado en un sistema embebido, hablese de un Microcontrolador, un Controlador Digital de Señales, un Procesador Digital de Señales, FPGA, etc. En cuanto a la parte que compete al funcionamiento del controlador, una manera de observar su funcionamiento sería a través de una interfaz gráfica que permite la visualización de las variables que se involucran en los algoritmos. Por otra parte para que una máquina eléctrica como lo es el motor BLDC funcione, es importante que se utilice un algoritmo de conmutación que funcione a través que 6 PWM que activen el conjunto de interruptores que son parte de la etapa de potencia que permite la rotación de la máquina. Las funciones dedicadas a cada

uno de los PWM, deben de ser dependientes de los algoritmos de control, que en esta caso, debe ser la salida del control de par.

### 1.3. Aportaciones

- Desarrollo de un algoritmo de control de par, para un motor BLDC el cual permite mantener el torque constante a un mínimo cambio de carga no constante (esto significa que debido a las dimensiones del motor, no se contempla una carga en especifico), haciendo énfasis en los trabajos investigados con anterioridad, y uso de elementos de retroalimentación que permitan la ejecución del algoritmo.
- Desarrollo de software para el control en procesador experto, permitiéndose así un algoritmo de control mas fácil, haciendo uso de herramientas de software que permitirán una programación más rápida y fácil de entender.

Las aportaciones de este trabajo se presentan en la facilidad de programación del algoritmo, el bajo costo que se presenta para efecto de pruebas y estudios del control, debido a que este puede ser implementado en motores BLDC de mayor potencia.

### 1.4. Metodología de Desarrollo

El trabajo de la tesis se encuentra estructurado en 4 capítulos, y un apartado para conclusiones y recomendaciones, los cuales contienen lo siguiente:

- El capítulo 1, como se ha mostrado, contiene los antecedentes generales, aportaciones y desarrollo del trabajo.
- El capítulo dos contiene la descripción del tipo de control usado para del motor sin escobillas propuesto para este trabajo, con la descripción general de las partes que conforman el arranque y paro de este, así como la conmutación que se aplica para su funcionamiento.

- El capítulo tres explica los tipos de algoritmos básicos, que permiten realizar una conmutación en el motor, la estrategia de modulación de ancho de pulso aplicada, así como el desarrollo de los controles PID.
- El capítulo cuatro expone los resultados obtenidos en las pruebas realizadas a partir de la conmutación, hasta los controles PID de velocidad y PI para el par.
- El apartado final presenta las conclusiones, aportaciones y sugerencias para trabajos futuros.

La figura 1.1 muestra el desarrollo metodológico que se llevara para el presente trabajo:

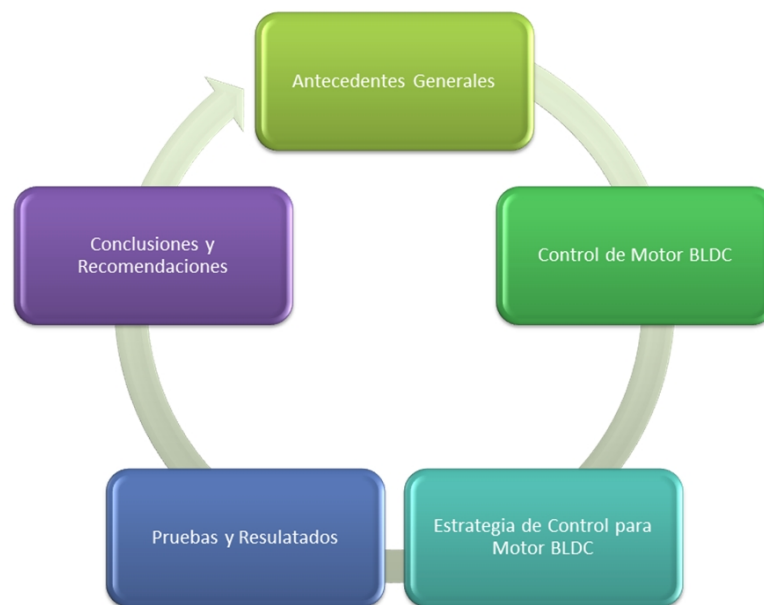


Figura 1.1: Desarrollo del trabajo de control de torque para motor de trifásico de corriente directa

## 1.5. Sumario

En este capítulo se presentó el estado del arte del control de motores sin escobillas de 3 fases de corriente directa, las aportaciones que se darán en este trabajo y se explicó brevemente el contenido de los capítulos subsecuentes.

El siguiente capítulo da explicación del tipo de motor que se utiliza en el presente trabajo partiendo de la clasificación en la que se encuentra dentro de las máquinas eléctricas rotatorias, la conmutación mas apropiada para este tipo de motor, el control que puede ser empleado y las técnicas que permiten poder utilizarlo.

# Capítulo 2

## Control de Motor BLDC

### 2.1. Clasificación de Máquinas Eléctricas

En la actualidad las máquinas eléctricas rotatorias pueden clasificarse en dos grandes grupos:

1. Máquinas de corriente directa.
2. Máquinas de corriente alterna, las cuales pueden clasificarse como sigue:
  - (a) Máquinas síncronas.
  - (b) Máquinas asíncronas o de inducción.
  - (c) Máquinas de reluctancia variable.

En este trabajo se utilizó la máquina síncrona de imanes permanentes de tipo imanes superficiales (MSIPS) con distribución trapezoidal. En la figura 2.1 se muestra un diagrama de clasificación de las máquinas eléctricas y se señala la ubicación de la máquina que se utilizó mediante rectángulos sombreados.

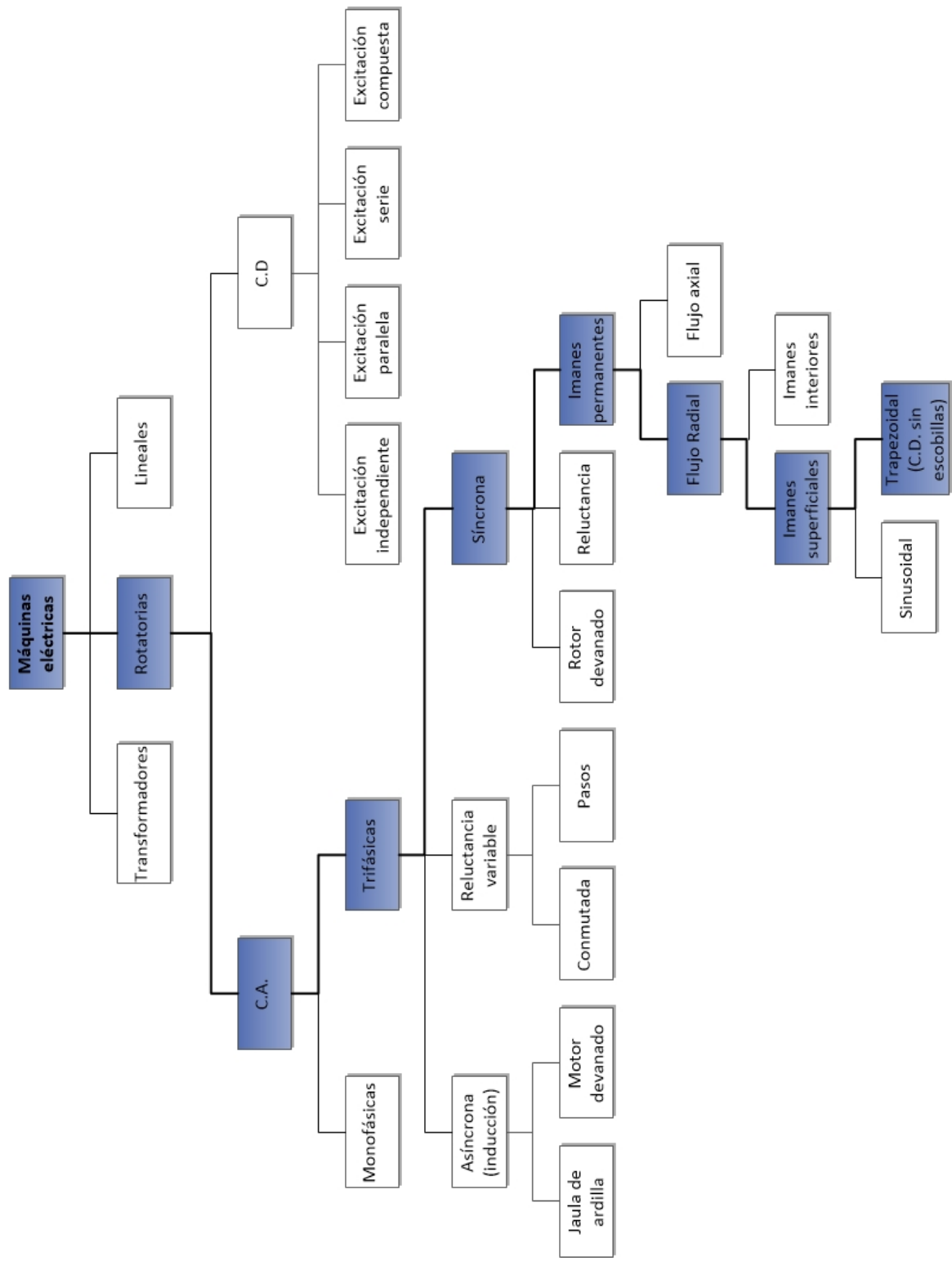


Figura 2.1: Clasificación de las máquinas eléctricas.



Los motores sin escobillas son muy populares y tienen una amplia área de aplicaciones. Los motores BLDC al carecer de un conmutador, son más confiables que los motores convencionales de corriente directa. Estos presentan ventajas cuando se comparan con los motores de inducción de corriente alterna. Ya que logran una mayor eficiencia al generar el flujo magnético del rotor [Frogal (2009)], también presentan gran par de arranque, un control suave de la velocidad, y la posibilidad de un control fácil del torque [Atallah (2003)].

A diferencia de los motores con escobillas convencionales donde la conmutación entre fases se realiza internamente de forma mecánica, en los motores sin escobillas las corrientes y los voltajes aplicados a cada uno de los bobinados del motor deben ser controlados independientemente mediante una conmutación electrónica [Tolrá (2003)]. Tradicionalmente, la implementación del control para este tipo de motores ha sido compleja y ha supuesto una fuerte inversión de tiempo, lo que requiere combinar diseño de hardware y recursos de programación de software [Rao (2009)]. Hoy la rápida disponibilidad y el bajo costo de los microcontroladores, han posibilitado el desarrollo para la aplicación de sistemas de control avanzado, a base de sistemas digitales de alto rendimiento, para el control de motores sin escobillas [Xu (2008)].

Existen dos grandes familias de controladores para motores diferenciadas principalmente en la utilización de sensores o sin sensores para determinar la posición del rotor [Tolrá (2003)]. El uso de sensores de efecto hall o de posición angular (encoders) es indispensable para el primer caso [Viramontes (2010)], mientras el uso de la fuerza contraelectromotriz es necesario para el otro [Kumar (2009)].

## **2.2. Motor sin Escobillas (*Brushless*)**

Los motores BLDC son de conmutación electrónica debido a que no tienen escobillas. El estator usualmente está hecho de material magnético. Los devanados del estator contienen ranuras las cuales están embobinadas tal como se muestra en la figura 2.2 [Viramontes (2010)].

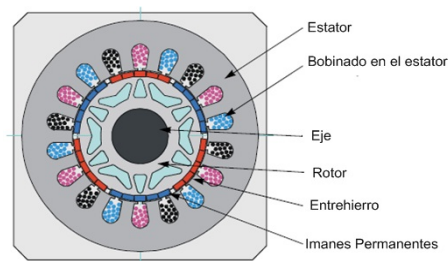


Figura 2.2: Sección transversal de un motor sin escobillas.

Los motores sin escobillas consisten de una parte estacionaria llamada estator, y una rotatoria llamada rotor. El espacio entre el estator y el rotor es llamada entrehierro. El estator lleva los devanados y el rotor lleva los imanes. Los motores sin escobillas pueden ser de polos salientes o interiores. Estos dos casos se muestran en la figura 2.3. En el otro caso el estator y los devanados son estacionarios, esto provoca que no tenga escobillas o anillos colectores [Colton (2008)]. Tres fases crean seis diferentes cambios que permite que la corriente fluya a través de las bobinas. Un microcontrolador induce la corriente para un rápido cambio en el campo magnético que provoca el giro del rotor. Para que el microcontrolador haga posible esto eficientemente, es necesario saber la posición del rotor, información que es proporcionada por sensores de efecto hall. Para evaluar la posición y direccionar la corriente, se necesita que el microcontrolador tenga un algoritmo computacional, que es la principal razón por lo cual los motores sin escobillas no eran populares en el pasado. Esto sin embargo no es un problema hoy en día. Las ventajas de un motor BLDC comparado con un motor de CD son muchas, tales como:

- La fricción es baja y la durabilidad es mejor, esto es debido a la falta de escobillas, las cuales no generan un desgaste.
- Al no existir un desgaste de las escobillas, no hay partículas que puedan causar daños.
- Un motor sin escobillas presenta una relación torque/ volumen y debido a que la bobina se encuentra en la parte exterior, ayuda a un rápido enfriamiento del motor [Wiberg (2003)].

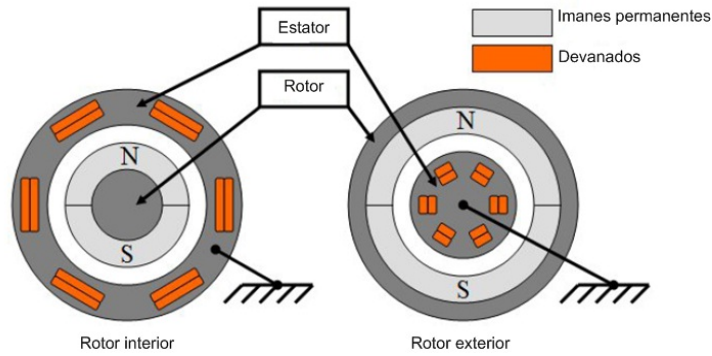


Figura 2.3: Rotor colocado exteriormente e interiormente.

La magnetización de los imanes permanentes y su desplazamiento en el rotor permite que la fuerza contraelectromotriz (tensión inducida en el estator debida al movimiento del rotor) de una forma trapezoidal. Esto permite que el voltaje de CC, con forma rectangular, sea usado para crear una rotación con señales de bajo par (figura 2.4).

En el motor solo puede haber 2 polos por fase; los dos polos por fase definen el radio entre la revolución eléctrica y la revolución mecánica. Por ejemplo, el motor mostrado en la figura 2.3 tiene 3 fases por polo; esto representa seis revoluciones eléctricas por una revolución mecánica.

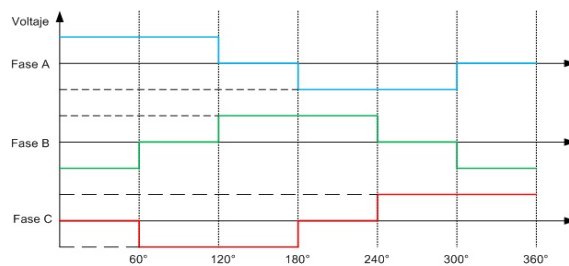


Figura 2.4: Sistema de Tres Fases.

La simplicidad de la forma rectangular del voltaje aplicado, garantiza la singularidad del control. Sin embargo, la posición del rotor debe ser conocida en ciertos ángulos con el fin de ajustar el voltaje aplicado por la fem. El acoplamiento entre cada conmutación y la fem es muy importante, con esta condición el motor estará en el mejor punto de trabajo.

Por lo tanto, la sencillez del control y su alto rendimiento hacen del motor sin escobillas, la mejor opción de bajo costo para aplicaciones de alta eficiencia.

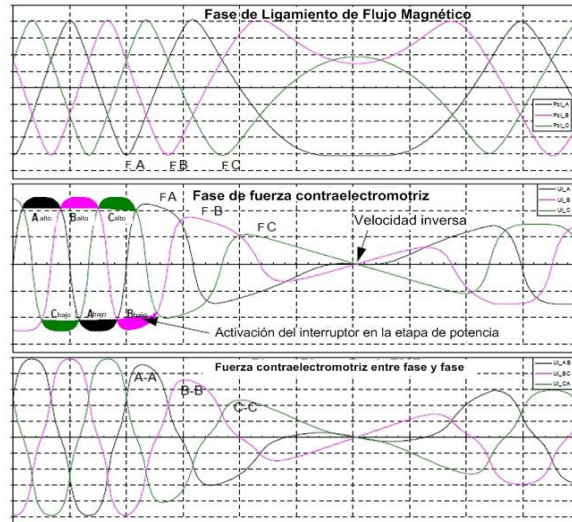


Figura 2.5: Fuerza contraelectromotriz del motor sin escobillas y Flujo Magnético.

La figura 2.5 muestra el número de formas de onda, el flujo magnético, el voltaje de fase de la fem y el voltaje de fase a fase de la fem. El flujo magnético fue obtenido mediante el cálculo del voltaje de fase de la fem, que se encuentra en las terminales no alimentadas del motor brushless. Como se observa, la forma de la fem es aproximadamente trapezoidal y la amplitud está en función de la velocidad. Durante la inversión de la velocidad las amplitudes cambian, así como también existe un cambio en la secuencia de las fases.

Las áreas rellenas en la parte superior de las formas de onda del voltaje de la fem, indican los intervalos donde la conmutación de las etapas de potencia se llevan a cabo. Como puede verse, los cambios de potencia están cíclicamente conmutados a través de los 6 pasos. Los puntos donde se cruzan los voltajes de fase de la fem representan la conmutación [Freescale (2009b) y Staszko (2010)].

### 2.2.1. Modelo Matemático del Motor

El modelo matemático del motor puede obtenerse a partir del circuito mostrado en la figura 2.6

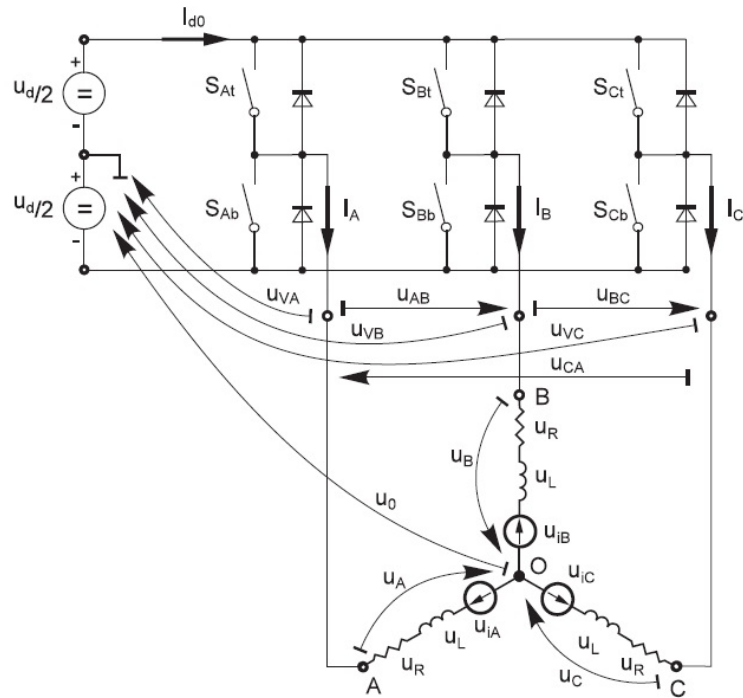


Figura 2.6: Topología del motor.

El objetivo del modelo es encontrar, como las características del motor dependen de el ángulo de cambio. El ángulo de cambio es la diferencia angular entre la conmutación real y la ideal (en el punto donde la fuerza contraelectromotriz de fase a fase cruza por cero).

El modelo del controlador del motor, consiste en una etapa de potencia de 3 fases conectada a un motor sin escobillas de 3 fases. La alimentación al sistema es provista por una fuente de voltaje  $U_d$ . Seis semiconductores conmutan ( $U_{A/B/Ct/b}$ ), ya controlados, permiten las formas de voltaje rectangular aplicadas. La conmutación de los semiconductores y los diodos son tomados como dispositivos ideales.

Las siguientes ecuaciones son validas para la topologia representada:

$$\begin{aligned}
u_A &= \frac{1}{3} \left( 2u_{VA} - u_{VB} - u_{VC} + \sum_{x=A}^C u_{ix} \right) \\
u_B &= \frac{1}{3} \left( 2u_{VB} - u_{VC} - u_{VA} + \sum_{x=A}^C u_{ix} \right) \\
u_C &= \frac{1}{3} \left( 2u_{VC} - u_{VA} - u_{VB} + \sum_{x=A}^C u_{ix} \right)
\end{aligned} \tag{2.1}$$

$$u_B = \frac{1}{3} \left( \sum_{x=A}^C u_{Vx} + \sum_{x=A}^C u_{ix} \right)$$

$$0 = i_a + i_B + i_C$$

donde:

$U_{VA} \cdots U_{VC}$ : Subdivisión de tensiones entre una salida de etapa de potencia y su cero natural.

$U_A \cdots U_C$  : Tensión del Motor fase-bobina.

$U_{iA} \cdots U_{iC}$ : Fase de fuerza contraelectromotriz (back-EMF) inducida en la bobina del estator.

$U_0$  : Tensión diferencial entre el punto central de la conexión en estrella de bobinado del motor y la etapa de potencia cero natural.

$i_A \cdots i_C$  : Corrientes de fase.

$R_s$  : Resistencias de fase.

La ecuación (2.1) puede ser reescrita, tomando en cuenta la resistencia y la inductancia en la fase del motor. La inductancia mutua entre las dos bobinas de fase del motor puede ignorarse, debido a que son muy pequeñas y no tiene un efecto significativo.

$$\begin{aligned}
u_{VA} - u_{iA} - \frac{1}{3} \left( \sum_{x=A}^C u_{Vx} - \sum_{x=A}^C u_{ix} \right) &= R \cdot i_A + L \frac{di_A}{dt} \\
u_{VB} - u_{iB} - \frac{1}{3} \left( \sum_{x=A}^C u_{Vx} - \sum_{x=A}^C u_{ix} \right) &= R \cdot i_B + L \frac{di_B}{dt} \\
u_{VC} - u_{iC} - \frac{1}{3} \left( \sum_{x=A}^C u_{Vx} - \sum_{x=A}^C u_{ix} \right) &= R \cdot i_C + L \frac{di_C}{dt}
\end{aligned} \tag{2.2}$$

donde:

$R$ :

Resistencia de fase del motor.

$L$ :

Inductancia de fase del motor.

El torque interno del propio motor se define como:

$$T_i = \frac{1}{\omega} \sum_{x=A}^C u_{ix} \cdot i_x = \sum_{x=A}^C \frac{d\Psi_x}{d\theta} \cdot i_x \tag{2.3}$$

donde:

$T_i$  :

Torque interno del motor (sin perdidas mecánicas).

$\omega, \theta$ :

Velocidad del rotor, posición del rotor.

$x$  :

Indice de fase, que se encuentra para A, B, C.

$\Psi_x$  :

Flujo magnético de la bobina  $x$ .

Esto es importante para entender como la fuerza contraelectromotriz, puede ser sensada y como el comportamiento del motor, depende de la alineación de la fuerza contraelectromotriz con los eventos de conmutación [Freescale (2009a) y Staszko (2010)].

## 2.3. Control Trapezoidal

Uno de los métodos más simples de control de motores sin escobillas, es el llamado conmutación trapezoidal o *6 steps mode* [Singh (2003)].

En este esquema se controla la corriente que circula por los terminales del motor, excitando un par simultáneamente y manteniendo el tercer terminal desconectado. Sucesivamente, se va alternando el par de terminales a excitar, hasta completar las seis combinaciones posibles según Tolrá (2003).

Tres sensores de efecto hall situados en el motor, son utilizados para proporcionar la posición aproximada del rotor al controlador y que éste pueda determinar el próximo par de terminales a excitar [Tolrá (2003)]. Para la rotación del motor sin escobillas, los devanados del estator, deben ser activados llevando una secuencia. Es importante conocer el orden de la posición del rotor, para entender que bobina estará energizada, según la secuencia de activación. Como ya se mencionó los sensores de efecto hall que se encuentran en el estator, sensan la posición del rotor [Viramontes (2010)].

Siempre que los polos magnéticos del rotor pasen cerca del los elementos primarios, estos, darán una señal alta o baja, indicando si el polo N o S paso cerca de alguno de ellos. Basándose en la combinación de las señales de los tres sensores de efecto hall, se determinara la secuencia de conmutación exacta en Yedamale (2003), Staszko (2010) y Yan (2010).

La generación de la rotación del campo magnético es implementada usando 6 transistores, de puente inversor como los que se muestran en la figura 2.7, donde la activación de los transistores se determinara usando 6 modulaciones de ancho de pulso, generados con un microcontrolador [Yedamale (2003)]. Los transistores que conforman la etapa de potencia de potencia pueden ser BJT, MOSFET o IGBT [Staszko (2010)].



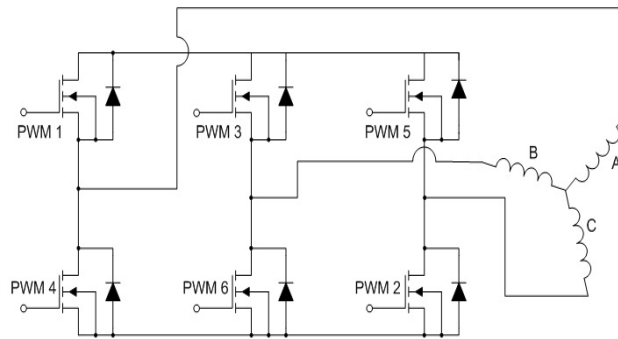


Figura 2.7: Inversor de 3 fases para motor brushless.

La secuencia de conmutación se explica a continuación:

- Se alimenta la bobina A, la bobina B no está conectada, por lo que, la tensión que regresa a la bobina C es negativa, es decir C activa un interruptor en un nivel bajo.
- La bobina A sigue alimentada, y la bobina C se desconecta, entonces el voltaje que pasa a través de B es negativo.
- La secuencia de conmutación sigue alimentando B y desconectando primero A y después C. Lo mismo ocurre con la bobina C.

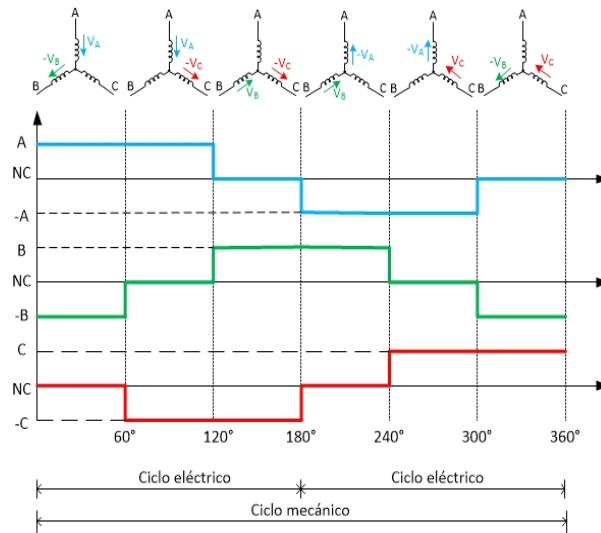


Figura 2.8: Secuencia de conmutación.

La figura 2.8 muestra la secuencia de conmutación en cada una de las bobinas.

Cada  $60^\circ$  de rotación, uno de los sensores Hall cambia de estado. Teniendo en cuenta esto, se tiene que 6 pasos completan un ciclo mecánico. En sincronismo, con cada 60 grados eléctricos, la fase de la corriente puede estar actualizándose. Sin embargo, un ciclo eléctrico no corresponderá a una revolución mecánica. El número de ciclos eléctricos se repetirá hasta completar una rotación mecánica, la cual dependerá del número de pares de polos del rotor. Por cada par de polos un ciclo eléctrico es completado. Entonces, el *numero de ciclos eléctricos / rotaciones* es igual a los *polos del rotor*. La figura 2.8 muestra la secuencia de detección de los sensores de efecto Hall en un motor con un par de polos, donde se observa que dos ciclos eléctricos, corresponden a una revolución mecánica [Akin (2000)]

Para detectar la posición del rotor, tres sensores de efecto hall están colocados alrededor del perímetro angular del rotor a iguales distancias. Para un rotor con un par de polos, los sensores están separados  $120^\circ$ . La lógica de estado de cada sensor cambia cada 180 grados pero el patrón de combinación de los tres sensores cambia en intervalos de  $60^\circ$ . Si el rotor tiene 4 polos, las distintas posiciones serán detectadas cada 30 grados del ángulo físico: (*ángulo físico = ángulo eléctrico / número de polos*) en Electronics (2006).

La alineación basada en sensores de efecto hall, consiste en combinar dos tipos posibles de conmutación de motores sin escobillas, trapezoidal y sinusoidal intentando entregar siempre el máximo par.

El algoritmo se puede dividir en dos fases:

- Arranque. En esta primera fase, se excita el motor utilizando una conmutación trapezoidal, basada únicamente en la información proporcionada por los sensores Hall. Tal como se muestra en la figura 2.9, a una posición inicial cualquiera del rotor, se puede estimar a partir de los sensores Hall, el sector en el que se encuentra, y por tanto, su posición aproximada con un error de  $\pm 30$ . El vector excitación (flechas numeradas) aplicado en cada sector garantiza que la excitación se encuentra siempre entre  $60^\circ$  y  $120^\circ$  de la posición del rotor.
- Funcionamiento normal. Una vez se detecta la transición de un sensor Hall se puede estimar con gran precisión la posición del rotor debido a que la distribución física de los sensores respecto a las fases que es conocida a una posición angular definida [Viramontes (2010)].

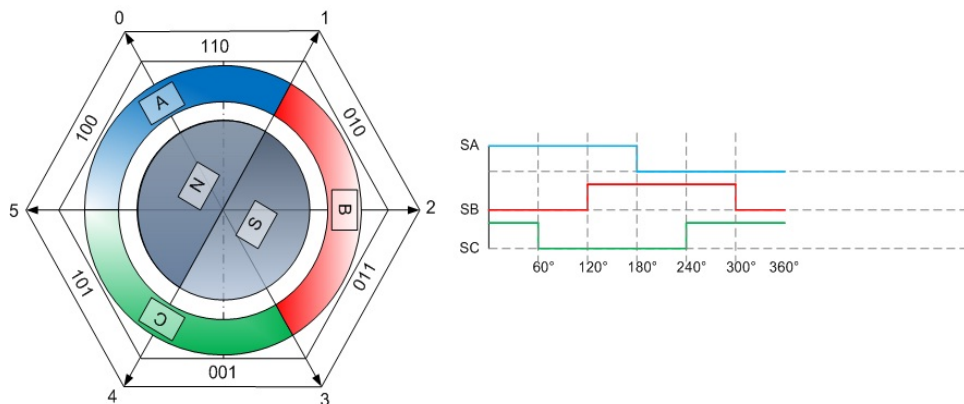


Figura 2.9: Representación de un motor sin escobillas con sensores de efecto Hall. Se pueden apreciar los 6 sectores definidos por la combinación de señales Hall, así como los vectores de excitación de cada sector para movimientos en sentido de las manecillas de reloj.

Ya sea haciendo uso de sensores de posición angular o de sensores de efecto hall, está claro que para realizar un control retroalimentado es necesario conocer la posición del rotor, esta variable puede ser detectada a partir de una propiedad del motor, la cual debe variar con la posición del rotor. Por otra parte, es aun más adecuado si la propiedad establece una posición única dentro de un ciclo eléctrico, añadiendo así robustez al sistema y dando lugar a una técnica que no haga uso de sensores. La variación del flujo en la fase, el cual se encuentra relacionado con la posición produce torque. Este efecto puede ser analizado en la reluctancia y fuerza contraelectromotriz, que pueden variar con la corriente, así como con la posición. La fuerza contraelectromotriz también varía linealmente con la velocidad. La variación de la reluctancia o de la fem puede ser registrada directamente o también su efecto en una cantidad secundaria que puede ser usada en su lugar según Elliott (2004).

## 2.4. Control de Torque y Velocidad

La conmutación permite el movimiento del motor y consigue que este funcione. El propósito del movimiento es controlar la velocidad y el torque [Dalib (2007)]. La velocidad en un motor BLDC es directamente proporcional al voltaje que se le aplica al estator. Mientras la conmutación mantiene el vector de flujo magnético rotacional, la velocidad que obliga al rotor a pasar a la siguiente posición es determinada por la intensidad de esta fuerza magnética y esta está dada por el voltaje aplicado a los anillos del estator. Mediante el uso de modulación de ancho de pulso a una frecuencia mayor que la conmutación, el aumento del voltaje aplicado al estator puede ser fácilmente controlado, por lo tanto la velocidad de el motor puede ser controlada como lo menciona Gamazo (2010). El torque es también controlado mediante la reducción del voltaje del estator cuando la carga del torque del motor es más alta que la indicada en Rao (2009).

Estos parámetros de control requieren sensores. Los sensores de efecto Hall que son usados para detectar la posición del rotor, sirven también para detectar la velocidad midiendo el tiempo que toman los sensores para cambiar. Una entrada en el canal de captura del contador de tiempo del microcontrolador es usado para medir la longitud de un pulso de uno de los sensores de efecto Hall. El tiempo medido es convertido en revoluciones por minuto (RPM) [Margarita (2004)].

El torque es sensado como una relación de la corriente consumida por el motor. El par puede ser medido individualmente por cada una de las tres fases o con el promedio del voltaje de bus de corriente directa que alimenta el motor como lo muestra Im (2010). La medición del torque en el bus de corriente directa es simple porque este requiere solo un canal de ADC y recursos de software para el tratamiento único de este canal [Frogal (2009)].

Mediante el seguimiento, de la velocidad y la corriente en el motor, mientras el proceso de la conmutación se lleva a cabo, la entrada media de tensión se incrementa y disminuye, esto afecta directamente la velocidad y el torque del motor. El controlador de seis pasos usa una de dos posibles técnicas de PWM:

**Conmutación de PWM Unipolar.** Esta técnica se refiere a las fases del motor que son cambiadas de tal manera que una de las fases regresa corriente mientras la modulación de PWM está ocurriendo en otra fase.

**Conmutación Bipolar.** Esta técnica se refiere al paso de voltaje a través de dos fases que son moduladas con el PWM, dos de entrada y una salida de corriente que están siendo moduladas.

La conmutación bipolar y unipolar tiene ventajas específicas. La conmutación unipolar reduce el ruido electromagnético y el rizo en el voltaje de bus de corriente directa. La conmutación bipolar es más adecuada para control sin sensor, donde esta es necesaria para sensar la fuerza contraelectromotriz. La aproximación bipolar tiene el punto de cero volts a un 50% de su ciclo útil, por lo tanto es más tiempo para sensar la fuerza contraelectromotriz.

Ambas aproximaciones pueden ser independientes o complementarias. La conmutación unipolar y bipolar se refiere a la relación que existe entre dos fases. La aproximación complementaria se refiere a la relación de 2 señales controlando una fase. La aproximación independiente se aplica al PWM solo en un lado de la fase. La aproximación complementaria modula ambos lados. Las aproximaciones complementarias y las independientes permiten el control en 2 o 4 cuadrantes.

La conmutación bipolar y el control de operación en cuadratura se determinada por los seis transistores en conmutación del inversor.

La operación de dos cuadrantes proporciona un voltaje fijo y corriente de la misma polaridad (voltaje positivo con corriente positiva o voltaje negativo y corriente negativa) que es la operación en los cuadrantes I y III en la figura 2.10.

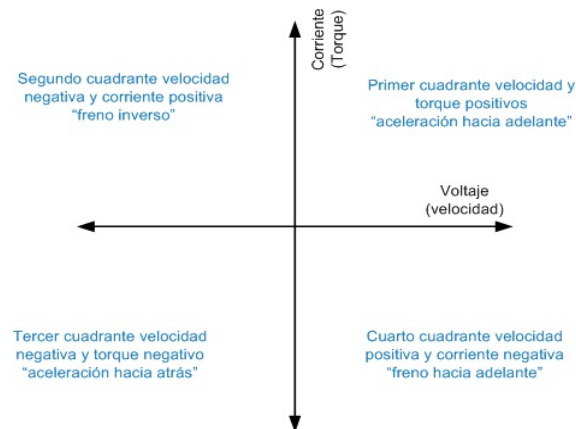


Figura 2.10: Operación del motor en los cuatro cuadrantes.

La operación de 4 cuadrantes en conmutación de PWM permite la operación de la tensión generada en los cuatro cuadrantes. Este siempre se usa en conmutación complementaria de los transistores superiores e inferiores. Los beneficios de la operación en los 4 cuadrantes con operación de PWM son: 1) Control de aceleración y freno (este freno es una posibilidad del motor a través del 0% al 50% del ciclo útil, que no es usado en la rotación del motor). 2) Operación lineal en todos los cuadrantes [Ohm (1999) y Ozkut (2008)]

## 2.5. Esquema de Control

Para el control se utilizan 2 controles PI convencionales, uno que controla la velocidad del motor, el cual proporciona la entrada a un segundo controlador proporcional integral, que controla el torque del motor. El control trapezoidal se realiza usando sensores de

efecto hall, siguiendo los vectores de corriente.

A continuación se muestra el procedimiento que se sigue para desarrollar el control empleado.

- Comunicación entre la etapa de potencia y la etapa de control.
- Generar modulación de ancho de pulso, manejando la conmutación electrónica en el motor.
- Realizar un control de velocidad a lazo abierto.
- Cerrar el lazo para el control de velocidad, utilizando sensores de efecto hall.
- Agregar un control PI de torque, conectado en serie al controlador de velocidad, tomando como retroalimentación la tensión o corriente de bus de las fases del motor.

La clave para un control efectivo de la velocidad y del par se basa relativamente en la simplicidad de las ecuaciones para torque y fuerza contraelectromotriz, que son similares a las de un motor CD convencional. La magnitud de la fuerza contraelectromotriz puede escribirse como [Salgues (2008)]:

$$E = 2NlrBw \quad (2.4)$$

Y el torque como:

$$T = \frac{1}{2}i^2\frac{dL}{d\theta} - \frac{1}{2}B^2\frac{dr}{d\theta} + \frac{4N}{\pi}Brl\pi i \quad (2.5)$$

Donde  $N$  es el numero de vueltas de la bobina por fase,  $l$  es la longitud del rotor,  $r$  es el radio interno del rotor,  $B$  es el la densidad de flujo magnético del rotor,  $w$  es la velocidad angular del motor,  $i$  es la corriente de fase,  $L$  es la inductancia de fase,  $\theta$  es la posición del rotor,  $R$  es la resistencia de fase.

Los primeros dos términos en la expresión del par son los componentes de reluctancia parásita. El tercer término produce el par, que es el mecanismo usado para la generación de par en los motores BLDC. En resumen, la fuerza contraelectromotriz es directamente proporcional a la velocidad del motor y a la producción de par que es también proporcional a la corriente de fase. Estos factores permiten el esquema de control mostrado en la

figura 2.11.

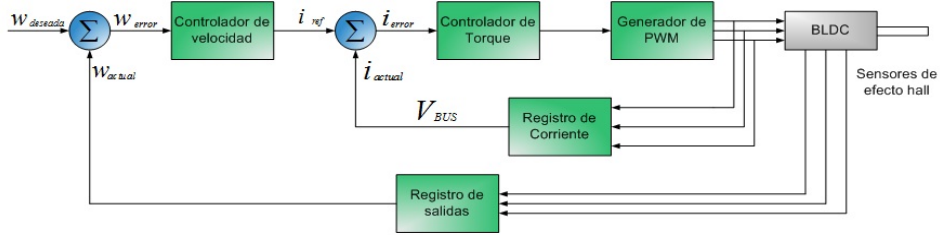


Figura 2.11: Diagrama de control.

El motor BLDC está caracterizado por dos fases en estado activo. En este esquema de control, la producción de par sigue el principio de que la corriente solo puede fluir solamente en dos de tres fases al mismo tiempo y que no hay generación de torque en la región donde la fuerza contraelectromotriz cruza por cero. La figura 2.12 describe la formas de onda en un motor sin escobillas en las dos fases de operación.

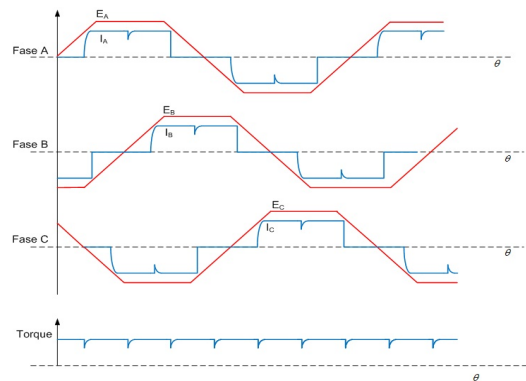


Figura 2.12: Formas de ondas eléctricas en las dos fases de operación y par rizado.

## 2.6. Modulación de Ancho de Pulso (PWM)

Los circuitos de conversión DC/AC (Corriente Directa a Corriente Alterna) tienen una amplia aplicación en la industria. Son utilizados en variadores de velocidad, sistemas de alimentación, filtros activos, etc. Los convertidores DC/AC se clasifican como inversores



con fuente de voltaje (VSI) e inversores con fuente de corriente (CSI). Los CSI se usan sistemas de alta potencia, los VSI se reservan para aplicaciones de baja y mediana potencia. Dentro de esta clasificación existen varias configuraciones de convertidores DC/AC que dependen de la aplicación final y el nivel de voltaje o corriente de su salida. En el caso de los drive para motores de baja y mediana potencia, la topología típica es el medio puente inversor trifásico con fuente de voltaje, formado por seis elementos en conmutación MOS-FET's, Transistores Bipolares de Compuerta Aislada (IGBT), Tiristores desactivados por Compuerta (GTO) o Tiristores Controlados por MOS (MCT).

Adicionalmente, se debe considerar la técnica de modulación que activará los elementos de conmutación. En el inversor como fuente de voltaje, la técnica de modulación se encarga de la forma de onda de la señal de salida AC, su nivel de tensión eléctrica y su frecuencia. Las técnicas de modulación se pueden clasificar en escalares o PWM (*Pulse Width Modulation*). Entre las técnicas escalares se encuentran la técnica de modulación de onda cuadrada (*six-step*), técnica de modulación sinusoidal, técnica de modulación sinusoidal con tercer armónico, entre otras; divisibles a la vez en técnicas de modulación basadas en portadora triangular y técnicas programadas. La técnica de modulación SVM se presenta en los años ochenta la cual maneja el puente inversor trifásico para el manejo de la potencia, disminuye las pérdidas por conmutación en el mismo y minimiza el contenido de armónicos de la señal de salida.

El principio de la modulación de ancho de pulso puede explicarse mejor considerando una conversión CD/CD (Corriente Directa a Corriente Directa) realizada por un convertidor genérico, alimentado por un voltaje de corriente directa. El convertidor es para controlar el componente de CD de la salida de voltaje. Como se muestra en la figura 2.13, esto se logra mediante el uso de un convertidor que conmuta de manera que la salida de voltaje consiste en un tren de pulsos (estado 1 del convertidor genérico) intercalado con escalamientos (estado 0). Oportunamente el circuito de potencia práctico para la conversión es llamado *recortador*. Los convertidores de baja potencia usados en fuentes de alimentación para equipo electrónico son usualmente llamados reguladores de voltaje o simplemente convertidores CD/CD.

En el caso ilustrado, los pulsos y las escalas son de la misma duración, esto es, los interruptores S1 y S2 operan con una duración de relación útil de 0.5. Una relación útil

$D$ , de un interruptor se define como:

$$D = \frac{t_{ON}}{t_{ON} + t_{OFF}} \quad (2.6)$$

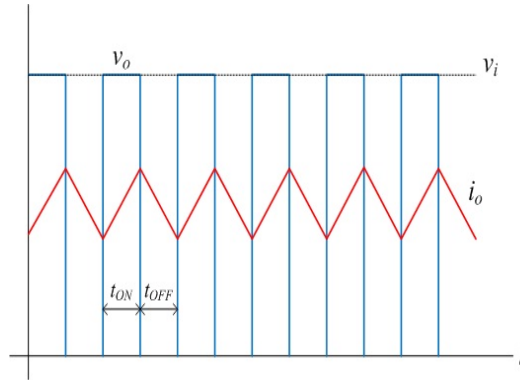


Figura 2.13: Formas de onda de voltaje y corriente de salida en un recortador genérico.

Donde  $t_{ON}$  es el tiempo activo (es decir el intervalo en el que el interruptor está cerrado), y  $t_{OFF}$  es el inactivo (el intervalo de tiempo en el que el interruptor está abierto). Aquí, el interruptor  $S_5$  también opera con una relación útil de 0.5. Sin embargo, si la relación útil de los interruptores  $S_1$  y  $S_2$  eran, por ejemplo, 0.6, la relación útil de interruptor  $S_5$  tendría que estar a 0.4. Los convertidores  $S_3$  y  $S_4$  del convertidor no son utilizados, a menos que la inversión de polaridad de la salida de voltaje sea necesaria, hasta entonces su relación útil es cero.

Es fácil ver que el valor promedio (componente de CD),  $V_{O,dc}$ , de la salida de voltaje que es proporcional al valor,  $V_i$ , de la entrada de voltaje y para la relación útil,  $D_{12}$ , de los interruptores  $S_1$  y  $S_2$ , que es

$$V_{O,dc} = D_{12}V_i \quad (2.7)$$

Desde que el rango de una relación útil es cero (el interruptor se encuentra abierto todo el tiempo) a la unidad (el interruptor está cerrado todo el tiempo), ajustando la relación útil de una apropiada conmutación que permite poner a  $V_{O,dc}$ , a cualquier valor entre cero y  $V_i$ . Como se muestra en la ecuación (2.7), la característica del voltaje de

control,  $V_{O,dc} = f(D_{12})$ , del recortador es lineal.

Aunque la frecuencia de conmutación,  $f_{sw}$ , definida como:

$$f_{sw} = \frac{1}{t_{ON} + t_{OFF}} \quad (2.8)$$

No afecta a la componente de corriente directa de la salida de voltaje, la calidad de la corriente de salida depende en gran medida de  $f_{sw}$ . Como se ilustra en la figura 2.14, si el número de pulsos por segundo se duplica en comparación con el de la figura 2.13, el incremento en la frecuencia de conmutación resulta en una reducción del el rizado de corriente de aproximadamente de un 50 %.

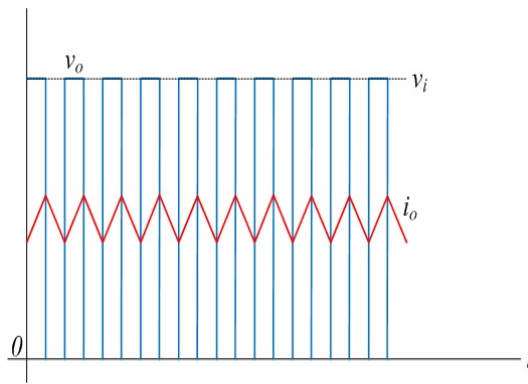


Figura 2.14: Formas de onda de voltaje y corriente de salida, en el recortador con una frecuencia de conmutación similar al de la figura 2.13.

Claramente, el tiempo entre los cambios consecutivos de estado del convertidor es suficientemente corto para prevenir un cambio significativo de la corriente entre los correspondientes saltos de la salida de voltaje.

La reducción del rizado de corriente puede explicarse también por el análisis armónico de la salida de voltaje. Hay que tener en cuenta que la salida de frecuencia fundamental es igual a la frecuencia fundamental de conmutación. Como resultado, los armónicos de la componente alterna de la tensión aparecen en las frecuencias siendo múltiplos de  $f_{sw}$ . Las correspondientes reactancias inductivas de la carga son proporcionales a estas frecuencias. Por lo tanto, si la frecuencia de conmutación es suficientemente grande, la componente de

alterna de la corriente de salida esta tan fuertemente atenuada que la corriente de CD es prácticamente ideal.

La modulación por ancho de pulso, proporciona un método para disminuir la Distorsión Armónica Total (DAT) de la corriente de carga. Una salida de un inversor PWM, con algo de filtrado, en general cumple con las especificaciones de DAT con más facilidad que el esquema de conmutación de onda cuadrada. La salida PWM sin filtrar tendrá un factor DAT relativamente elevado, pero los armónicos tendrán unas frecuencias mucho más altas que los de la onda cuadrada, haciendo más sencillo el filtrado.

En modulación PWM, la amplitud de la tensión de salida se puede controlar por medio de las formas de onda moduladoras. Dos ventajas de la modulación PWM son la reducción de los requerimientos de filtro para reducir los armónicos y el control de la amplitud de salida. Entre las desventajas se puede citar que los circuitos de control de los interruptores son más complejos, y que hay mayores pérdidas debidas a una conmutación más frecuente.

El control de los interruptores para la salida sinusoidal PWM requiere una señal de referencia, llamada a veces señal de control o moduladora, que puede ser una senoidal o una cuadrada; y una señal portadora, que es una onda triangular que controla la frecuencia de conmutación.

### 2.6.1. Conmutacion Bipolar

La figura 2.15 ilustra el principio de modulación por ancho de pulso bipolar cuadrada. La figura 2.15a muestra una señal cuadrada de referencia y una señal portadora triangular. Cuando el valor instantáneo de la cuadrada de referencia es mayor que la portadora triangular, la salida está en  $+V_{CC}$ , y cuando la referencia es menor que la portadora, la salida está en  $-V_{CC}$ .

$$V_O = +V_{CC} \quad \text{para} \quad v_{cuadrada} > v_{tri} \quad (2.9)$$

$$V_O = -V_{CC} \quad \text{para} \quad v_{cuadrada} < v_{tri}$$

Esta versión de PWM es bipolar, ya que la salida toma valores alternos entre más y

menos la tensión de la fuente de continua.

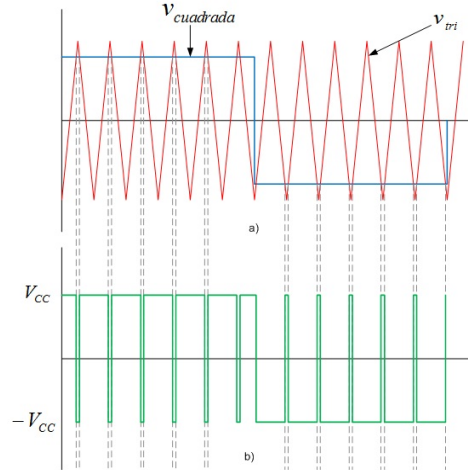


Figura 2.15: Modulación por ancho de pulsos bipolar. (a) Referencia cuadrada y portadora triangular. (b) La salida es  $+V_{CC}$  cuando  $v_{cuadrada} > v_{tri}$  y es  $-V_{CC}$  cuando  $v_{cuadrada} < v_{tri}$

El esquema de conmutación que permite implementar la conmutación bipolar utilizando el puente inversor de onda completa de la figura 2.16 se determina comparando las señales instantáneas de referencia y portadora:

$$v_{cuadrada} > v_{tri} (v_0 = +V_{CC}) \quad (2.10)$$

$$v_{cuadrada} < v_{tri} (v_0 = -V_{CC})$$

En la ecuación (2.10) cuando  $v_{cuadrada} > v_{tri}$  los interruptores  $S_1$  y  $S_2$  están conduciendo, y cuando  $v_{cuadrada} < v_{tri}$  los interruptores  $S_3$  y  $S_4$  se cierran y por lo tanto permiten el paso de la corriente.

### 2.6.2. Conmutación unipolar

En un esquema de conmutación unipolar para la modulación por ancho de pulso, la salida se conmuta de nivel alto a cero, o de nivel bajo a cero, en lugar de entre niveles alto y bajo, como en la conmutación bipolar. Un esquema de conmutación unipolar tiene

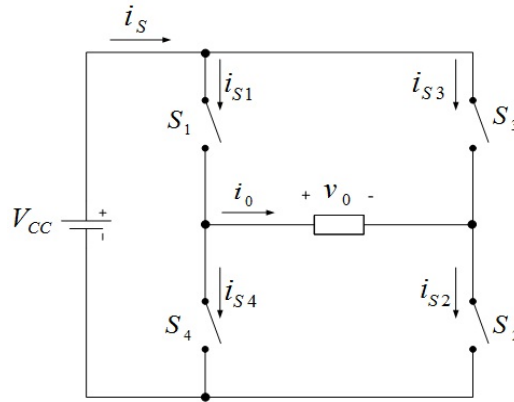


Figura 2.16: Puente inversor completo, cuando  $S_1$  y  $S_2$  están cerrados el voltaje de salida  $v_O$  es igual a  $V_{CC}$ ; y cuando  $S_3$  y  $S_4$  están cerrados el voltaje de salida  $v_O$  es  $-V_{CC}$

los siguientes controles de interruptores:

$$\begin{aligned}
 S_1 & \text{ conduce cuando} & +v_{cuadrada} > v_{tri} \\
 S_2 & \text{ conduce cuando} & -v_{cuadrada} < v_{tri} \\
 S_3 & \text{ conduce cuando} & -v_{cuadrada} > v_{tri} \\
 S_4 & \text{ conduce cuando} & +v_{cuadrada} < v_{tri}
 \end{aligned} \tag{2.11}$$

Observe que los pares de interruptores  $(S_1, S_4)$  y  $(S_2, S_3)$  son complementarios: cuando un interruptor de uno de los pares está cerrado, el otro está abierto. Las tensiones  $v_a$  y  $v_b$  en la figura 2.17a oscilan entre  $+V_{CC}$  y cero. La tensión de salida  $v_O = v_{ab} = v_a - v_b$  es tal como se muestra en la figura 2.17b.

Otro esquema de conmutación unipolar sólo tiene un par de interruptores trabajando a la frecuencia de la portadora mientras que el otro trabaja a la frecuencia de referencia, con lo que se cuenta con dos interruptores de alta frecuencia y dos de baja frecuencia. Es este esquema de conmutación:

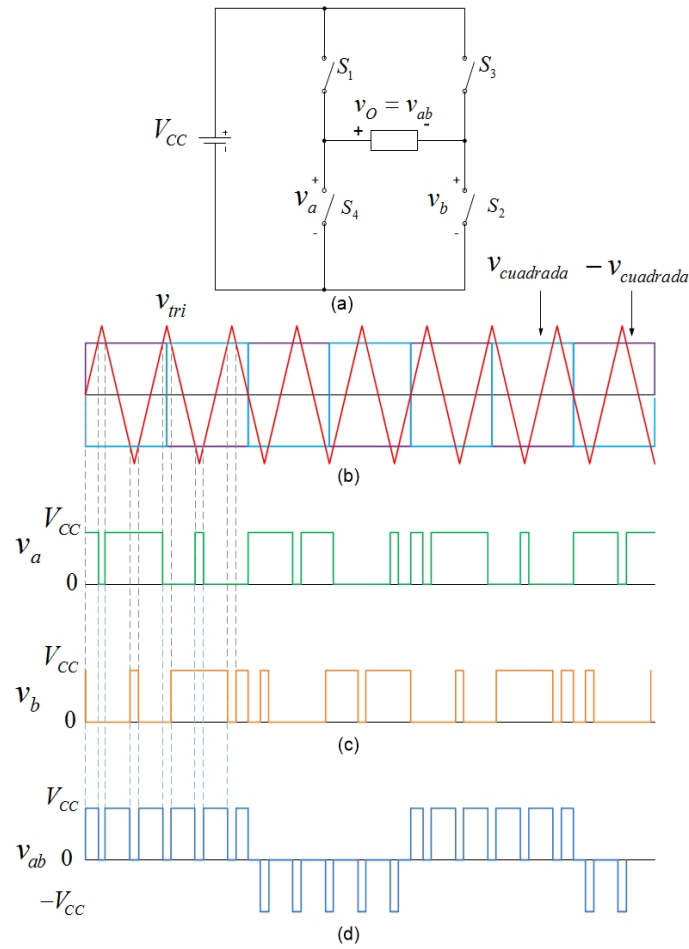


Figura 2.17: (a) Puentes convertidor de onda completa para PWM unipolar. (b) Señales de referencia y portadora. (c) Tensiones eléctricas  $v_a$  y  $v_b$  del puente. (d) Tensión eléctrica de salida.

$S_1$	conduce cuando	$v_{cuadrada} > v_{tri}$	alta frecuencia	
$S_2$	conduce cuando	$v_{cuadrada} > 0$	baja frecuencia	
$S_3$	conduce cuando	$v_{cuadrada} < 0$	baja frecuencia	(2.12)
$S_4$	conduce cuando	$v_{cuadrada} < v_{tri}$	alta frecuencia	

Donde las ondas cuadrada y triangular son como las mostradas en la figura 2.18a.

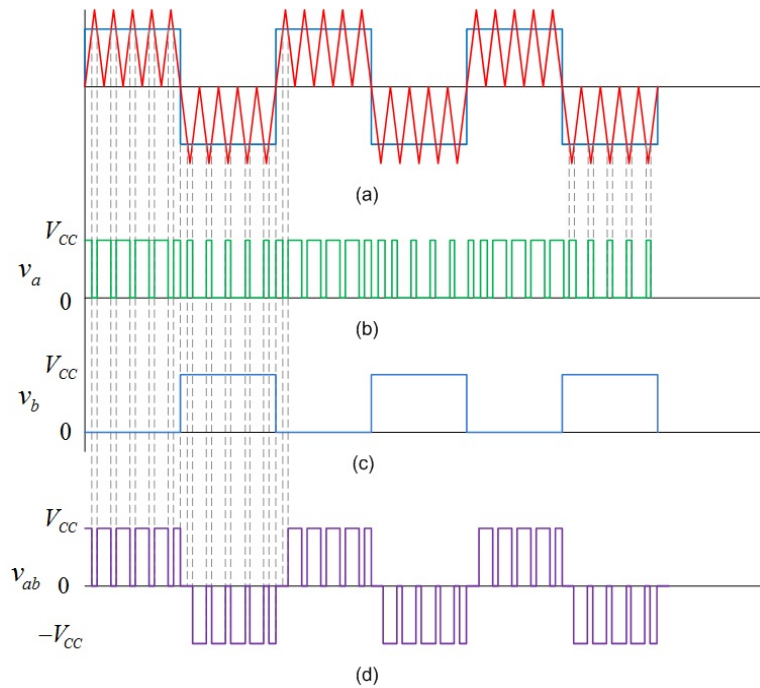


Figura 2.18: PWM unipolar con interruptores de alta y baja frecuencia. (a) Señales de referencia y de control. (b)  $v_a$ . (c)  $v_b$ . (d) Salida  $v_a - v_b$

De forma alternativa,  $S_2$  y  $S_3$  podrían ser los interruptores de alta frecuencia, y  $S_1$  y  $S_4$  podrían ser los interruptores de baja frecuencia.

### 2.6.3. Control de voltaje en inversores de 3 Fases

Un inversor trifásico puede considerarse como tres inversores monofásicos, con la salida de cada inversor monofásico desplazada  $120^\circ$ . Las técnicas de control analizadas anteriormente son aplicables a los inversores trifásicos. Las ventajas de la conmutación PWM son las mismas que en el caso monofásico; pocos requisitos de filtrado para la reducción de armónicos y el control de la amplitud a la frecuencia fundamental o de referencia.

La conmutación PWM en el inversor trifásico es similar a la del inversor monofásico. Básicamente, cada interruptor se controla comparando una onda cuadrada de referencia con una portadora triangular. La frecuencia fundamental de salida es igual que la de la onda de referencia, y la amplitud de la salida viene determinada por las amplitudes relativas de las ondas de referencia y portadora.



Al igual que en el caso del inversor trifásico de seis pasos, los interruptores de la figura 2.19 se controlan por parejas  $(S_1, S_4)$ ,  $(S_2, S_5)$  y  $(S_3, S_6)$ . Cuando uno de los interruptores de la pareja está cerrado, el otro está abierto. Cada pareja de interruptores requiere una onda cuadrada de referencia separada. Las tres ondas cuadradas de referencia están desfasadas  $120^\circ$  para producir una salida trifásica equilibrada [Senturk (2009)].

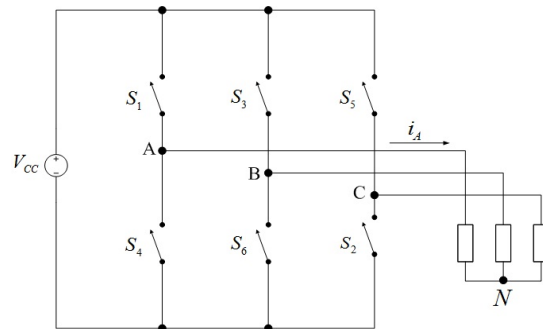


Figura 2.19: Inversor trifásico.

La figura 2.20 muestra una portadora triangular y las tres ondas de referencia. El control de interruptores es como se muestra en la ecuación (2.13).

$$\begin{aligned}
 S_1 & \quad \text{conduce cuando} & \quad v_A > v_{tri} \\
 S_2 & \quad \text{conduce cuando} & \quad v_C > v_{tri} \\
 S_3 & \quad \text{conduce cuando} & \quad v_B > v_{tri} \\
 S_4 & \quad \text{conduce cuando} & \quad v_A < v_{tri} \\
 S_5 & \quad \text{conduce cuando} & \quad v_C < v_{tri} \\
 S_6 & \quad \text{conduce cuando} & \quad v_B < v_{tri}
 \end{aligned}
 \tag{2.13}$$

Los armónicos se minimizarán si se elige una frecuencia portadora igual a un múltiplo impar de tres veces la frecuencia de referencia, es decir, 3, 9, 15, . . . tres veces la referencia.

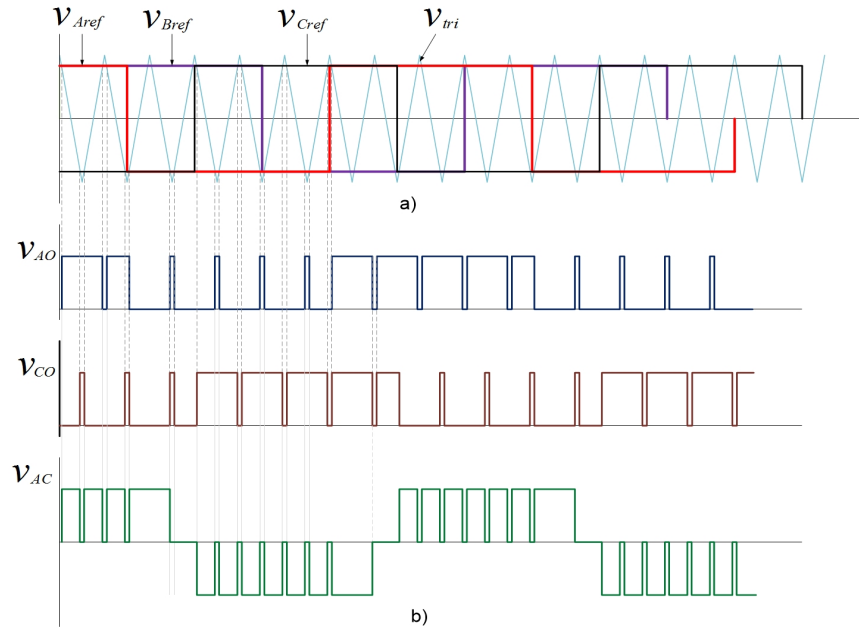


Figura 2.20: (a) Ondas portadora y de referencia para un control PWM. (b) Formas de onda de salida.

La figura 2.20b muestra las tensiones de línea de salida para un inversor trifásico PWM.

Una vez analizado los diferentes tipos de modulación de ancho de pulso, se considera que el más adecuado es utilizar un tipo unipolar en el modo de seis pasos, pues es el más simple de implementar y el cual permite una conmutación de 120 grados entre fase y fase.

#### 2.6.4. Efecto del tiempo muerto en Inversores controlados por PWM

La figura 2.21 muestra una fase del inversor PWM, donde se asume que los transistores de potencia son usados como elementos de conmutación. La señal de PWM controla los transistores a través de un circuito de retardo. Las señales en las bases  $B_1$  y  $B_2$  para los transistores  $T_1$  y  $T_2$ , respectivamente, se ilustran en la figura 2.22, donde las señales de control muestran un retardo ocasionado por un tiempo de retardo  $T_d$ , que se muestran en las líneas punteadas [Ben-Brahim (2002)].

Durante el tiempo de retardo, ambos transistores dejan de conducir, y la salida en

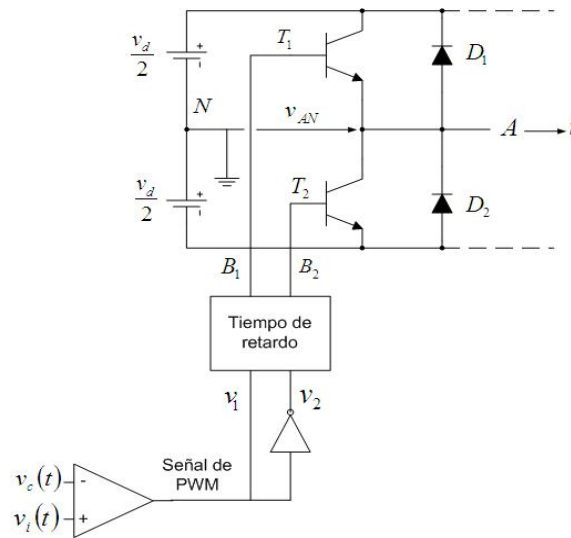


Figura 2.21: Configuración básica de PWM (solo se muestra una fase del inversor).

la terminal A esta flotando. Sin embargo, si la corriente de salida  $i$  es continua, como normalmente es el caso, la corriente fluye a través de los diodos de libre circulación  $D_1$  y  $D_2$ . La conducción de cualquiera de los diodos dependerá de la dirección del flujo de corriente. Cuando la corriente fluye hacia la carga ( $i > 0$ , de acuerdo a la figura 2.21), el diodo  $D_2$  conduce, y el voltaje negativo aparecerá en la salida de la terminal. Además, para la corriente que fluye en sentido contrario en el inversor, el voltaje positivo aparece en la salida a través de  $D_1$ . Como resultado de ello, dejando de lado el tiempo de almacenamiento inverso del transistor, la tensión eléctrica de salida se desvía de su forma de onda ideal PWM por la cantidad de áreas sombreadas en la figura 2.22 [Chen (2007)].

Obviamente, durante el tiempo de retardo, la salida de voltaje no puede ser controlada por las señales de control, pero está determinada por la condición de carga, que es, la dirección en el que fluye la corriente. Aunque la condición de carga está sujeta a cambios, puede afirmarse que la desviación del voltaje se debe a que el tiempo de retardo se opone al flujo de corriente en cualquier dirección. Por lo tanto, la desviación de voltaje, provoca que la magnitud de la corriente sea más pequeña que la esperada. Esto a su vez deriva uno de los efectos más importantes de la temporización, que es, la disminución de la tensión eléctrica efectiva de salida del inversor [Hwang (2009)].

Haciendo una referencia a la figura 2.21, se puede definir una señal de error  $e(t)$ , que se obtiene de relación que existe entre la señal de referencia  $v_i(t)$  y la señal portadora

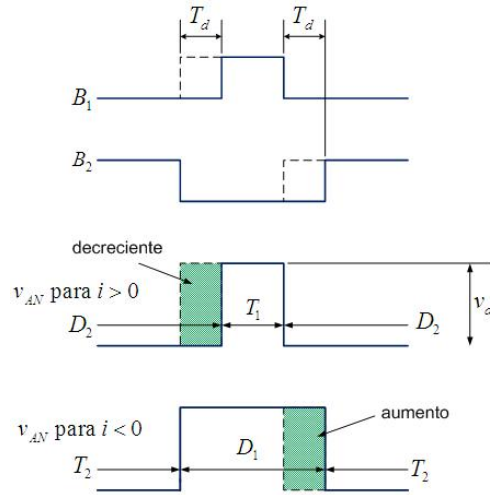


Figura 2.22: Señales de control de retraso y voltaje de salida

$v_c(t)$ , por

$$e(t) = v_i(t) - v_c(t) \quad (2.14)$$

Basándose en esta definición, se observa que la relación que existe entre la tensión de fase  $v_{AN}$  y la señal de error  $e(t)$  se ilustra en la figura 2.23 para  $i_a > 0$  y  $i_a < 0$ . La desviación de voltaje  $\epsilon$  depende del tiempo de retardo  $T_d$  y la pendiente de la onda triangular de la señal portadora  $v_c(t)$ . Se asume que la señal portadora  $v_i(t)$  varía lentamente en comparación con la señal de referencia  $v_c(t)$  que está a una alta frecuencia. El radio  $-\epsilon/T_d$  es igual a la caída de la pendiente  $-2V_c/(T_c/2)$  de la señal triangular portadora  $v_c(t)$ , por lo que se tiene

$$\epsilon = T_d \frac{2V_c}{T_c/2} = 4f_c T_d V_c \quad (2.15)$$

donde  $V_c$ ,  $T_c$  y  $f_c$  denotan la amplitud (mitad del voltaje pico a pico), periodo y frecuencia de la señal portadora  $v_c(t)$ , respectivamente [Jeong (1991)].

Es interesante notar que existe una no linealidad en la histéresis que hay entre las señales  $v_{AN}(t)$  y  $e(t)$ . Apoyándose en la ecuación (2.14), la figura 2.23 se usa en el dia-

grama a bloques de la figura 2.24 para el circuito inversor PWM con retraso de tiempo que se muestra en la figura 2.21.

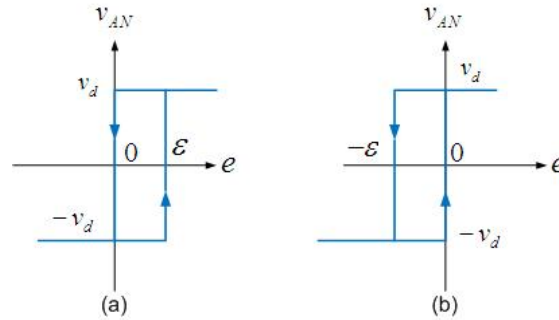


Figura 2.23: Relación entre el voltaje de fase  $v_a(t)$  y la señal de error  $e(t)$  para (a)  $i_a > 0$ ; (b)  $i_a < 0$

Observando el sistema de la figura 2.24, puede simplificarse al de la figura 2.25 donde la señal de la función  $\text{sgn}(i_a)$  se define como +1 para  $i_a > 0$  y -1 para  $i_a < 0$ . Esta simplificación se deduce directamente del hecho de que

$$\begin{aligned}
 i_a > 0; m = e - \frac{\epsilon}{2} &\Rightarrow (e = 0, \text{ entonces } m = -\frac{\epsilon}{2}) \text{ y } (e = \epsilon, \text{ entonces } m = -\frac{\epsilon}{2}) \\
 i_a < 0; m = e + \frac{\epsilon}{2} &\Rightarrow (e = -\epsilon, \text{ entonces } m = -\frac{\epsilon}{2}) \text{ y } (e = 0, \text{ entonces } m = \frac{\epsilon}{2})
 \end{aligned}
 \tag{2.16}$$

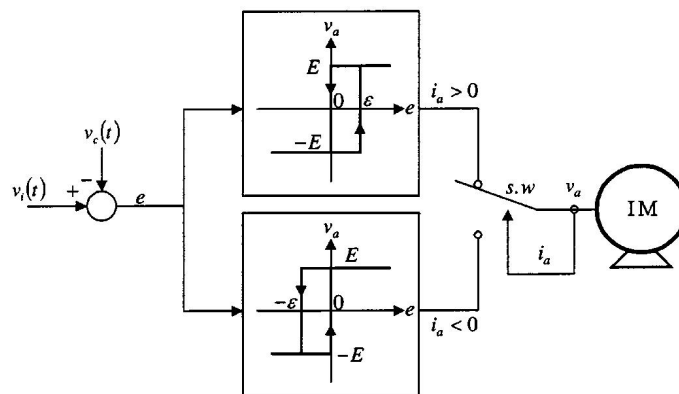


Figura 2.24: Diagrama de bloques para un inversor PWM con un circuito de tiempo de retardo

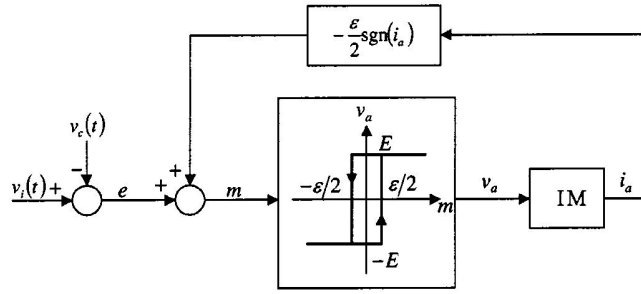


Figura 2.25: Diagrama a bloques equivalente de la figura 2.24

Por lo tanto, la histéresis que relaciona  $m$  en la figura 2.24, se desplaza a la derecha para  $i_a > 0$  y a la izquierda para  $i_a < 0$  por una cantidad  $\epsilon/2$  para dar lugar a la histéresis que relaciona a  $e$  y  $v_a$  que se muestran en la figura 2.24 [Leggate (1997)].

Para evaluar el efecto del tiempo muerto cuantitativamente, se debe seguir la hipótesis de que:

- El tiempo de almacenamiento inverso de los elementos de conmutación no es considerado.
- La frecuencia de conmutación, comparada con la frecuencia de salida fundamental, es suficientemente grande.
- La desviación de voltaje que se produce es casi equidistante.
- La caída de un pulso corto no ocurre.
- La salida de corriente del inversor es casi cuadrada.

Con estas suposiciones, el efecto acumulativo de la repetición de los tiempos de retardo (efecto del tiempo muerto) puede evaluarse por un promedio de las desviaciones de tensión en cada medio ciclo positivo y negativo de la corriente.

Desde la desviación para cada pulso  $\Delta e$  se mide por

$$\Delta e = T_d V_d \quad (2.17)$$

La desviación del voltaje promedio en cada medio ciclo de la salida del inversor  $\Delta V$  se obtiene de

$$\Delta V = \frac{MT_d}{T}V_d \quad (2.18)$$

Donde  $M$  es el número de cambios por ciclo y  $T$  es la duración de un ciclo.

En la ecuación (2.18)  $MT_d$  es la suma total del tiempo de retardo del ciclo. Por lo tanto, se puede decir que la desviación de voltaje promedio respecto al voltaje de alimentación es equivalente a la relación del total del tiempo muerto a lo largo de un ciclo completo [Lin (2002)].

## 2.7. Control PID

El control PID es usado comúnmente para ajustar la velocidad. Este recibe señales de los sensores y calcula la acción correctiva a través de un algoritmo computacional basándose en el error (proporcional), la suma de todos los errores anteriores (integral) y la tasa de cambio del error (derivativo). El modelo matemático del PID puede ser representado como:

$$U(t) = k_p \left[ e(t) + \frac{1}{T_i} \int_0^t e(t) + T_d \frac{de(t)}{dt} \right] \quad (2.19)$$

### 2.7.1. Respuesta proporcional

La componente proporcional puede ser expresada como:

$$p(t) = k_p \cdot e(t) \quad (2.20)$$

En un control PID, el efecto de controlar depende de la ganancia proporcional ( $k_p$ ). En general, incrementando la ganancia proporcional se incrementara la velocidad del sistema de control y se reducirá la respuesta en estado estable. Sin embargo, si la ganancia es también grande, el sistema comenzará a oscilar y será inestable.

### 2.7.2. Respuesta Integral

La componente integral puede ser expresada como:

$$i(t) = \frac{k_p}{T_i} \int_t^0 e(t) dt \quad (2.21)$$

Para la expresión (2.21), se observa que la componente integral suma el término del error en el tiempo. El resultado es que cada término de error pequeño puede causar que la componente integral se incremente lentamente. La respuesta integral aumenta continuamente con el tiempo a menos que el error sea cero, por lo tanto el objetivo es llevar el error en estado estacionario a cero. Pero el control integral reducirá la velocidad de respuesta total del sistema y aumentara el sobreimpulso.

Al incrementar la componente integral ( $T_i$ ) permite que la componente integral se acumule débilmente y que se reduzca el sobreimpulso, esto hará que el sistema no oscile durante el aumento del tiempo, por lo tanto, mejorara su estabilidad. Sin embargo, esto retrasa el proceso de eliminación del error en estado estable. Reduciendo  $T_i$  aumenta la componente integral y se reduce el tiempo de eliminación del error, pero el sistema oscila. Entonces  $T_i$  debe seleccionarse de acuerdo a las necesidades prácticas.

### 2.7.3. Respuesta Derivativa

La parte derivativa se puede expresar como:

$$d(t) = k_p \cdot T_d \frac{de(t)}{dt} \quad (2.22)$$

El efecto de la parte derivativa depende de la constante de tiempo derivativo ( $T_d$ ). En general, el tamaño de ( $T_d$ ), mejora el efecto de retrasar el cambio de  $e(t)$  y viceversa. Entonces la selección del ( $T_d$ ) apropiado, mejorara los requerimientos del sistema.

### 2.7.4. Control PID Digital

El esquema de de control PID analógico ha sido usado de manera exitosa en muchos sistemas de control industrial por más de medio siglo. El principio básico del esquema de



control PID es que actúa sobre la variable a ser manipulada a través de una apropiada combinación de las tres acciones de control: acción de control proporcional (donde la acción de control es proporcional a la señal del error actuante, la cual es la diferencia entre la entrada y la señal de realimentación); la acción de control integral (donde la acción de control es proporcional a la integral de la señal de error actuante).

En situaciones donde muchas plantas se controlan directamente mediante una sola computadora digital (como un esquema de control en el que se controlan desde unos cuantos lazos hasta cientos de estos mediante un solo controlador digital), la mayoría de los lazos de control se pueden manipular mediante esquemas de control PID.

La acción de control PID en controladores analógicos es la que se muestra en la ecuación (2.19) donde  $e(t)$  es la entrada al controlador (señal actuante),  $u(t)$  es la salida del controlador (señal manipulada),  $k_p$  es la ganancia proporcional,  $T_i$  es el tiempo integral (o tiempo de reajuste) y  $T_d$  es el tiempo derivativo (o tiempo de adelanto).

Para obtener la función de transferencia pulso del controlador PID digital, se puede discretizar la ecuación (2.19). Al aproximar el término integral mediante la suma trapezoidal y el término derivativo mediante la diferencia de dos puntos, se obtiene

$$\begin{aligned}
 u(kT) = k_p \left\{ e(kT) + \right. \\
 \left. \frac{T}{T_i} \left[ \frac{e(0) + e(T)}{2} + \frac{e(T) + e(2T)}{2} + \dots \right. \right. \\
 \left. \left. + \frac{e((k-1)T) + e(kT)}{2} \right] + T_d \frac{e(kT) - e((k-1)T)}{T} \right\}
 \end{aligned} \tag{2.23}$$

La expresión (2.23) también puede ser expresada como:

$$u(kT) = k_p \left\{ e(kT) + \frac{T}{T_i} \sum_{h=1}^k \frac{e((h-1)T) + e(hT)}{2} + \frac{T_d}{T} [e(kT) - e((k-1)T)] \right\} \quad (2.24)$$

Se define

$$\frac{e((h-1)T) + e(hT)}{2} = f(hT), f(0) = 0 \quad (2.25)$$

En la figura 2.26 se muestra la función  $f(hT)$ . Entonces

$$\sum_{h=1}^k \frac{e((h-1)T) + e(hT)}{2} = \sum_{h=1}^k f(hT) \quad (2.26)$$

Al tomar la transformada  $z$  de la última ecuación, se obtiene:

$$\begin{aligned} Z \left[ \sum_{h=1}^k \frac{e((h-1)T) + e(hT)}{2} \right] &= Z \left[ \sum_{h=1}^k f(hT) \right] = \frac{1}{1-z^{-1}} [F(z) - f(0)] \\ &= \frac{1}{1-z^{-1}} F(z) \end{aligned} \quad (2.27)$$

donde

$$F(z) = Z[f(hT)] = \frac{1+z^{-1}}{2} E(z) \quad (2.28)$$

Por lo tanto

$$Z \left[ \sum_{h=1}^k \frac{e((h-1)T) + e(hT)}{2} \right] = \frac{1+z^{-1}}{2(1-z^{-1})} E(z) \quad (2.29)$$

Entonces la transformada  $z$  de la ecuación (2.23) da como resultado:

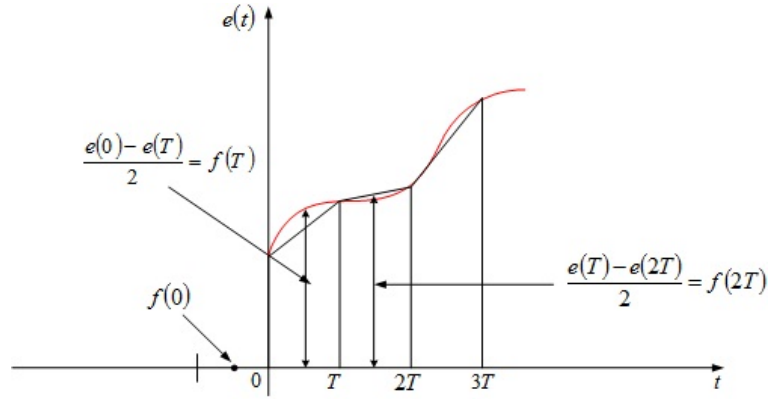


Figura 2.26: Diagrama que muestra la función  $f(hT)$

$$\begin{aligned}
 M(z) &= k_p \left[ 1 + \frac{T}{2T_i} \frac{1+z^{-1}}{1-z^{-1}} + \frac{T_d}{T} (1-z^{-1}) \right] E(z) \\
 &= \left[ k_p + \frac{k_i}{1-z^{-1}} + k_d (1-z^{-1}) \right] E(z)
 \end{aligned} \tag{2.30}$$

donde:

$$k_p = k - \frac{kT}{2T_i} = k - \frac{k_i}{2}$$

$$k_i = \frac{kT}{T_i}$$

$$k_d = \frac{kT_d}{T}$$

La ganancia proporcional  $k_p$  para el controlador PID digital es mas pequeña que la ganancia  $k$  para el controlador PID analógico por un factor de  $\frac{k_p}{k}$ .

La función de transferencia pulso para el controlador PID digital se convierte en

$$G_D(z) = \frac{U(z)}{E(z)} = k_p + \frac{k_i}{1 - z^{-1}} + k_d(1 - z^{-1}) \quad (2.31)$$

La función de transferencia pulso del controlador PID digital mostrada en la ecuación (2.31) es conocida comúnmente como *forma posicional* del esquema de control PID.

La otra forma por lo regular utilizada en el esquema de control PID digital es el esquema conocido como forma de velocidad. Para obtener la ecuación del control PID en la forma de velocidad, se considera la diferencia hacia atrás en  $m(kT)$ , esto es, la diferencia entre  $m(kT)$  y  $m((k-1)T)$ , de donde se obtiene

$$U(z) = -k_p C(z) + k_i \frac{R(z) - C(z)}{1 - z^{-1}} + k_d(1 - z^{-1}) C(z) \quad (2.32)$$

La ecuación (2.32) da el esquema de control PID en la forma de velocidad. La figura 2.27 da el esquema de control PID en la forma de velocidad. En la figura 2.27 se muestra un diagrama de bloques de la realización de un esquema de control PID digital en la forma de velocidad. Note que en la ecuación (2.32) sólo el término del control integral incluye la entrada  $R(z)$ . Por lo tanto, el término integral no se puede excluir del controlador digital si éste se utiliza en la forma de velocidad.

Una ventaja del esquema de control PID en la forma de velocidad es que no es necesaria la inicialización cuando se conmuta de operación manual a automática. De este modo, si existen cambios súbitos grandes en el punto de ajuste o en el inicio de la respuesta de operación del proceso, el esquema de control PID en la forma de velocidad es útil en la supresión de correcciones excesivas en sistemas de control.

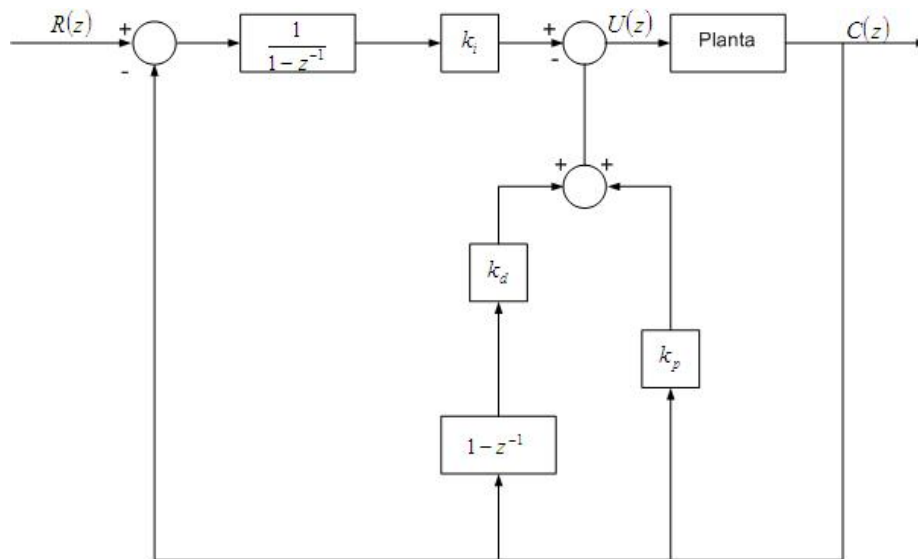


Figura 2.27: Diagrama a bloques de la realización del esquema de control PID en la forma de velocidad

## 2.8. Sumario

Este segundo capítulo ha dado una breve explicación de la clasificación de las máquinas eléctricas rotatorias, de donde se mostró que el motor BLDC es una máquina con distribución trapezoidal, de imanes superficiales, con flujo radial e imanes permanentes que es un motor de tres fases. Partiendo de lo anterior se estudio la manera en que trabaja el motor de corriente directa sin escobillas, donde se da a partir de la conmutación de 6 pasos que permite el flujo de corriente a través de las bobinas; con el fin de poder definir la alimentación de cada una de las bobinas. Posteriormente para poder controlar el motor se establecieron los pasos a seguir así como los tipos de algoritmos como: PID y PI, para las variables de velocidad y torque, y la técnica de PWM, indispensable para poder activar cada uno de los transistores que pertenecen a la etapa de potencia.

La señal PWM es generada utilizando un microcontrolador, donde también es posible implementar los algoritmos computaciones de control. El capítulo 3 explica los registros que permiten generar una modulación del ancho de de pulso, un convertidor analógico-digital que se utiliza para obtener la lectura de la corriente y cerrar el lazo de control y la manera de generar la programación para realizar los algoritmos de control utilizando los periféricos antes mencionados.



# Capítulo 3

## Estrategia de Control para Motor BLDC

### 3.1. Introducción

Los motores sin escobillas son muy populares en una amplia variedad de aplicaciones [Elevich (2005)], estos utilizan un conmutador eléctrico que es mejor que un conmutador mecánico, que es el tipo de conmutación que utilizan los motores de corriente directa convencionales [Prokop (2005)], existen tres principales estrategias de conmutación que permiten un funcionamiento correcto del motor, las cuales son: conmutación trapezoidal (6 pasos o *six steps*), Sinusoidal y Control de Campo Orientado (FOC, por sus en ingles *Field Oriented Control*) [Wiberg (2003)]. Cualquiera de estas técnicas debe aplicarse en combinación con reguladores de corriente, de velocidad o de ambas dependiendo de la aplicación, para de esta forma, poder aprovechar mejor las revoluciones por minuto o par que ofrece la máquina [Dalib (2007)]. Estas estrategias de control han sido posibles gracias al gran crecimiento de los sistemas informáticos incorporados, junto a la flexibilidad en precio que ofrecen los semiconductores de potencia y los microcontroladores, es posible tener métodos de control más sofisticados [Margarita (2004)].

Este capítulo presenta el desarrollo de una estrategia de control para un motor sin escobillas a lazo abierto, manejando una conmutación de 6 pasos. Posteriormente se presenta el desarrollo de reguladores de velocidad y de control, los cuales permitirán tener un lazo de control cerrado dando como resultado un mejor aprovechamiento tanto de la velocidad como del torque, y así, optimizar el funcionamiento de una máquina trifásica de corriente directa. Los algoritmos de control son desarrollados en la plataforma Code Warrior para poder ser implementados en un microcontrolador HCS08, aunque, el presente capítulo no desglosa el desarrollo completo del algoritmo, debido a que este se presenta en el siguiente capítulo de este trabajo, si expone parte del uso de los registros necesarios, para poder desarrollar las estrategias de control antes mencionadas.

## 3.2. Conmutación Trapezoidal

Las señales que controlan las tres fases de un motor BLDC pasan a través de una etapa de potencia, que es un puente trifásico como el que se muestra en la figura 3.1, este tipo de señales son moduladoras de ancho de pulso (PWM); cuando se desea un control a lazo cerrado, a este mismo esquema se pueden añadir diferentes tipos de sensores como: sensores de efecto hall, y sensores de posición angular (*encoders* y *resolvers*).

Existen dos tipos de motores sin escobillas de imanes permanentes, que dependen de las formas de onda de su fuerza contraelectromotriz. En los motores BLDC la forma de onda contraelectromotriz (*back-EMF*, del inglés, *Electromotive Force*) es trapezoidal (6 pasos). La forma de onda contraelectromotriz es sinusoidal en los motores Síncronos de Imanes Permanentes (PMSM, por sus siglas en inglés).

Los motores PMSM proveen un par rizado casi de cero, por lo cual entregan una alta eficiencia. Los motores BLDC tienen 15 % más de potencia que los motores PMSM, asumiendo que las pérdidas son iguales para ambos motores, y que los motores PMSM tienen un factor de potencia. Para el mismo torque los motores PMSM requieren una mayor capacidad en el manejo de la corriente [Lee (2004)].



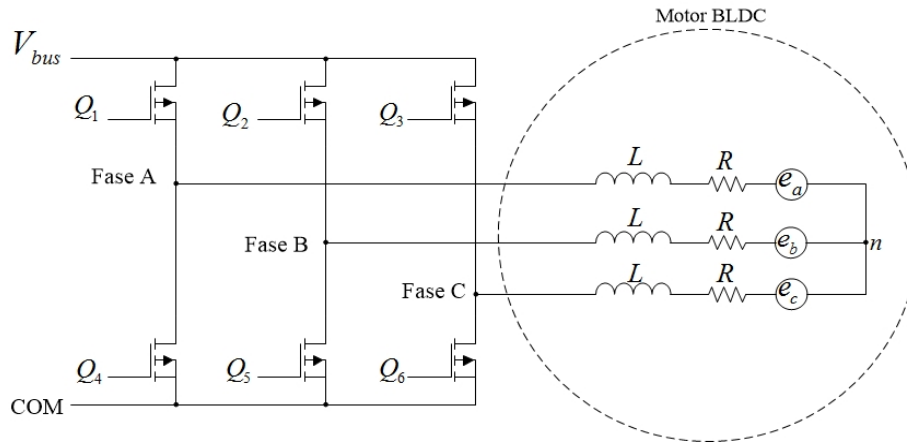


Figura 3.1: Motor sin escobillas y etapa de potencia.

El devanado del estator de un motor BLDC es típicamente trapezoidal, con el fin de que la forma de onda de la fuerza contraelectromotriz sea también trapezoidal. El torque generado tiene un considerable par rizado que ocurre en cada paso de la conmutación trapezoidal. Como ya se ha mencionado en capítulos anteriores la conmutación de 6 pasos se energizan dos fases en cada secuencia de la conmutación. Por el contrario los motores PMSM, tienen un devanado distribuido sinusoidalmente, el cual permite una fuerza contraelectromotriz de forma sinusoidal. El torque generado en los motores síncronos de imán permanente es suave con mucho menor par rizado en comparación con los BLDC. Sin embargo el par máximo que ofrecen los motores PMSM es menor. La forma de onda senoidal en los motores PMSM es debido a que en estos se activan las tres bobinas al mismo tiempo [Atallah (2003)].

La relación entre la forma trapezoidal de la *back-EMF* en las tres fases, la corriente del motor, la diferencia de potencial eléctrico del motor sin escobillas se muestran en la figura 3.2. La forma trapezoidal de la fuerza contraelectromotriz ( $e_{a,byc}$ ) tiene una magnitud constante de  $E_p$  durante 120 grados eléctricos en los ciclos positivo y negativo. La diferencia de potencial eléctrico,  $P_a$ , y el par electromagnético son continuos cuando se aplica corriente ( $i_{a,byc}$ ) al motor durante el mismo periodo en ambas mitades del ciclo.

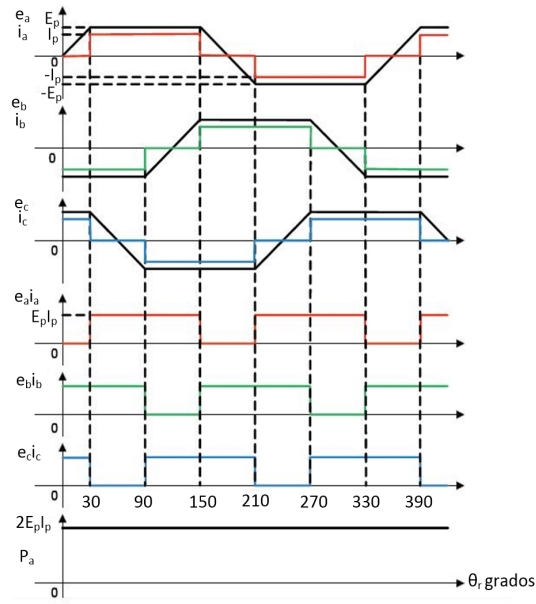


Figura 3.2: Relación entre fuerza contraelectromotriz, corriente, y diferencia de potencial eléctrico para las tres fases del motor BLDC.

La ecuación para el voltaje instantáneo y torque del motor sin escobillas, partiendo de las figuras 3.2 y 3.1, es:

$$\begin{bmatrix} v_a \\ v_b \\ v_c \end{bmatrix} = \begin{bmatrix} R & 0 & 0 \\ 0 & R & 0 \\ 0 & 0 & R \end{bmatrix} + \begin{bmatrix} L & 0 & 0 \\ 0 & L & 0 \\ 0 & 0 & L \end{bmatrix} \frac{d}{dt} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \begin{bmatrix} e_a \\ e_b \\ e_c \end{bmatrix} \quad (3.1)$$

$$T_e = e_a \cdot i_a + e_b \cdot i_b + e_c \cdot i_c \quad (3.2)$$

donde:

$v_{a,b,c}$ : Voltajes en las terminales del motor en volts.

$i_{a,b,c}$ : Corrientes de fase en el motor, en ampers.

$e_{a,b,yc}$ : Voltajes de la fuerza contraelectromotriz, en volts.

$R$ : Resistencia de los devanados del motor, en ohms.

$\omega_m$  : Velocidad angular, en rad/s

$T_e$  : Torque del motor, en Newton-metro.

El par del motor es generado por la suma de los productos de FEM (fuerza contraelectromotriz) y la corriente del motor como se muestra en la figura 3.1 y en la ecuación (3.2). Sin embargo, esta es inversamente proporcional a la velocidad del motor, proporcionando un bajo torque a velocidades bajas y un alto par a velocidades altas.

La conmutación trapezoidal activa dos dispositivos de potencia en cada fase del motor en una secuencia predeterminada. Este método es muy popular debido a la simplicidad del algoritmo de control. Para obtener información acerca de la posición del rotor, se usan tres sensores de efecto hall. Esto es muy efectivo al momento de controlar la velocidad, pero provoca un par rizado, especialmente a bajas velocidades. Por lo tanto, es muy popular para aplicaciones simples que requieren un sencillo lazo de control.

Como ya se ha explicado anteriormente para lograr que el motor sin escobillas presente una rotación mecánica es necesario que se desarrollen dos ciclo eléctricos, esto se logra a través de señales PWM que activan los transistores, haciendo uso de conmutación bipolar o conmutación unipolar, donde cada una presenta diferentes ventajas, por ejemplo, la conmutación unipolar reduce el ruido electromagnético y el par rizado debido a que conmuta menos. La conmutación bipolar presenta mayor ventaja si se desea realizar una estrategia de control sin utilizar sensores, porque, se hace uso de la fuerza contraelectromotriz para poder cerrar el lazo de control [Lee (2004)].

En la estrategia de conmutación de 6 pasos, se implementa conmutación unipolar, utilizando técnicas PWM, tal como se muestra en la figura 3.3. La conmutación unipolar con PWM puede controlar el motor BLDC con un desarrollo del voltaje y la corriente casi lineal. La única diferencia es que el ciclo de trabajo del PWM igual a cero proporciona uno de aproximadamente cero. La principal ventaja que presenta la conmutación unipolar PWM es que tiene un mejor control electromagnético debido al manejo de voltaje que presenta, lo cual no ocurre en la conmutación bipolar. Esta técnica de control es implementada usando un microcontrolador MC56F8006 de Freescale Semiconductor.

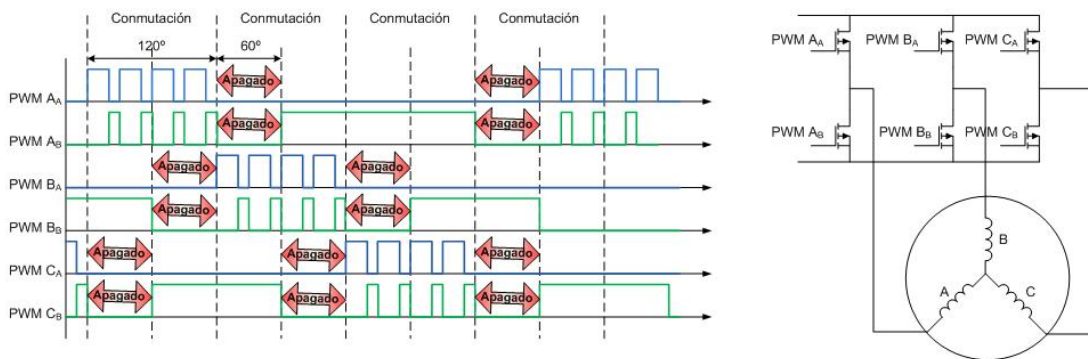


Figura 3.3: Desactivación de transistores en conmutación unipolar complementaria.

### 3.3. Microcontrolador MC56F8002/MC56F8006

Los microcontroladores 56F8006/56F8002 pertenecen a la familia 568006E, los cuales están basados en los controladores digitales de señales (DSC por sus siglas en inglés, *Digital Signal Controller*). Estos combinan, en un simple chip, la capacidad de procesamiento de los DSP (por sus siglas en inglés, *Digital Signal Processor*) y la funcionalidad de un microcontrolador, a través de la flexibilidad de sus periféricos que permiten generar soluciones a bajo costo. Debido a la flexibilidad en cuanto a configuración, y código compacto de programa, es por esto que los 56F8006/56F8002 son una buena opción para diversas aplicaciones. Los 56F8006/56F8002 incluyen periféricos que son especialmente útiles para distintas aplicaciones, como:

- Control Industrial

- Aparatos electrodomésticos
- Sensores inteligentes
- Sensores de seguridad
- Fuentes conmutadas y ahorro de energía
- Medición de energía
- Control de Motores (ACIM, BLDC, PMSM, SR y motores a pasos)
- Herramientas portátiles
- Equipo médico
- Instrumentación
- Iluminación

El núcleo 56800E se basa en una doble arquitectura Harvard, que consta de tres unidades de ejecución en paralelo, lo que permite que realice un máximo de seis operaciones por ciclo de instrucción. El estilo de programación y la optimización de instrucciones permiten la generación de un eficiente código de control, implementado en un compacto DSP. El conjunto de instrucciones es también muy eficiente para los compiladores de C, que permiten el rápido desarrollo de aplicaciones de control optimizado.

Los 56F8006/56F8002 permiten la ejecución del programa en memorias internas. Dos operadores de datos pueden ser accedidos desde la memoria RAM en el chip de datos por cada ciclo de instrucción. Los 56F8006/56F8002 también ofrecen más de 40 entradas/salidas de propósito general (GPIO, por sus siglas en ingles *General Purpose Input/Output*), dependiendo de la configuración de los periféricos.

Estos controladores digitales de señales ofrecen las siguientes características:

- Eficiente Controlador Digital de Señales (DSC) de 16 bit de la familia 56800E, con doble arquitectura Harvard.

- Más de 32 Millones de Instrucciones por Segundo (MIPS por sus siglas en ingles, *Million Instructions Per Second*) en un núcleo de 32 MHz de frecuencia.
- 155 Instrucciones básicas en conjunto con más de 20 modos de dirección.
- Un solo ciclo de  $16 \times 16$  bit en un acumulador multiplicador paralelo (MAC),
- Instrucciones en paralelo en un conjunto de modos de direccionamiento DSP.
- Tres buses internos de direccionamiento.
- Cuatro buses internos de datos.
- Alimentación de aproximadamente 1.9 V a 3.6 V.
- Soporte de temperatura en un rango de  $-40^{\circ}\text{C}$  a  $105^{\circ}\text{C}$
- Hasta 16 kbytes de memoria flash con protección.
- Un modulo con 6 canales PWM.
- Reloj de PWM con una operación de hasta 96 MHz.
- Señales PWM en modo de centro alineado y alineación al borde.
- Dos convertidores analógico-digitales de 12 bits (ADC).
- Dos Amplificadores de Ganancia Programable (PGA).
- Una interfaz de comunicación serial de alta velocidad (SCI) con funcionalidad de esclavo LIN.
- Una Interfaz de Periférico Serial (SPI).
- Un puerto de Circuito Inter-Integrado  $I^2C$ .
- 40 pines de entrada/salida de propósito general (GPIO).
- Interfaz de programación JTAG/EOnCE para debugueo en tiempo real.

### 3.3.1. Diagrama Bloques del Sistema

La figura 3.4 muestra la estructura completa de la aplicación en hardware incluyendo las conexiones y la circuitería externa.

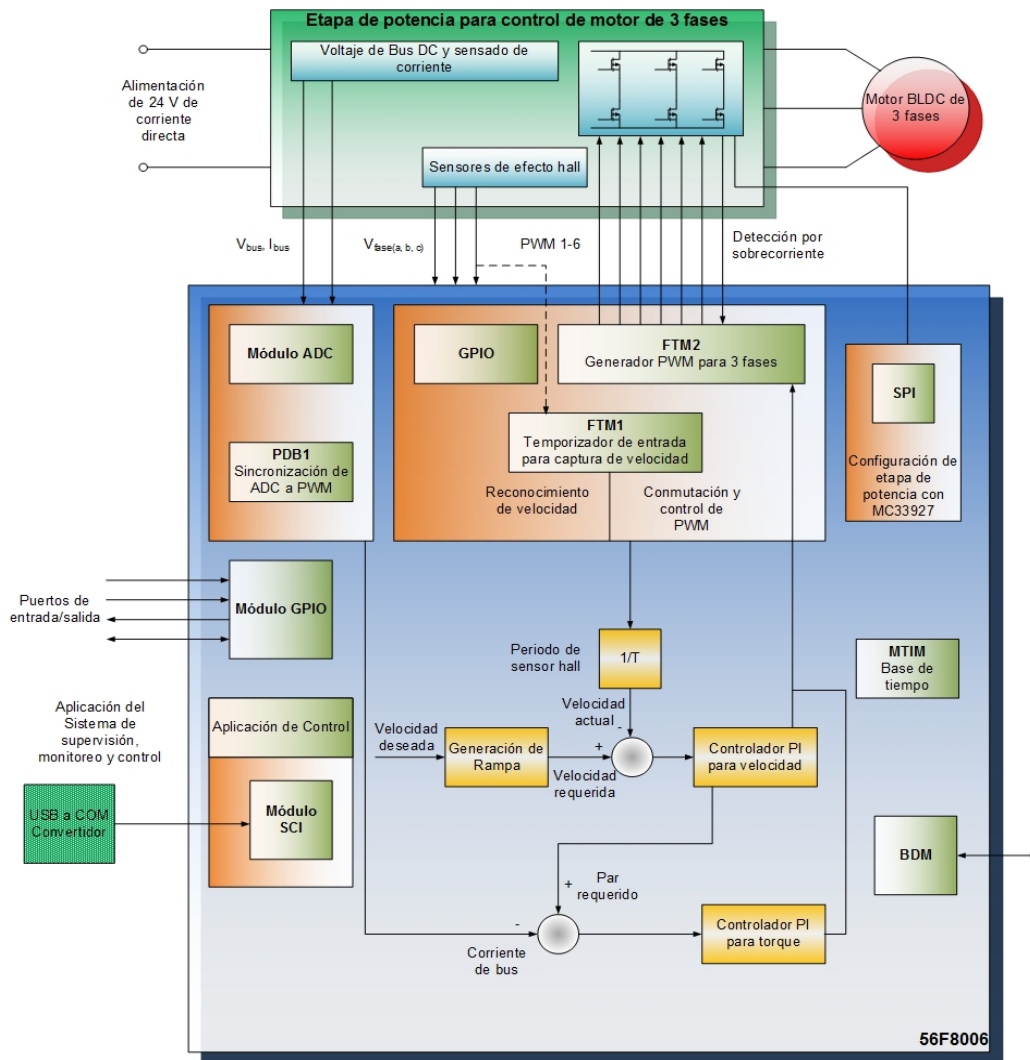


Figura 3.4: Diagrama a bloques del Sistema.

### 3.4. Convertidor Analógico-Digital (ADC)

Este convertidor analógico a digital de 12 bits es designado para operar con un DSC. En este apartado el término ADCn representa ambos módulos. ADC0 (n es 0) que es lo mismo que ADCA. ADC1 (n es 1) que es lo mismo que ADCB.

Las características del modulo ADC son:

- Entrada de voltajes en un rango de valores de  $V_{SSA}$  a  $V_{DDA}$ .
- Mas de 28 entradas analógicas.

- Salida de 12, 10 o 8 bits en formato justificado a la derecha.
- Conversión automática o continua (retorno automático después de la conversión).
- Tiempo de muestreo configurable y velocidad de conversión.
- Soporte para conversión simultánea y software de disparo.
- Sensores de temperatura dirigidos a ANA26 y ANB26.
- Puede ser configurado para tomar dos muestras (sin reconfigurar el software) basado en disparos en hardware durante los modos de ida y regreso

### 3.4.1. Estatus y Control de Registro 1A y 1B (ADCn\_ADCSC1A y ADCn\_ADCSC1B)

En este apartado se describe el funcionamiento del estado y control de registro, ADCn\_ADCSC1A y ADCn\_ADCSC1B. Estos registros tienen los mismos campos, y son usados para el aprovechamiento del control del ADC en la operación de ida y vuelta. En algún punto en el tiempo, solo un ADCn\_ADCSC1A o ADCn\_ADCSC1B es activado controlando el ADC. Esto es posible escribiendo en ADCn\_ADCSC1A, mientras ADCn\_ADCSC1B está realizando una conversión, y viceversa. Escribiendo ADCn\_ADCSC1A mientras esta activo el control de conversión, se aborta la conversión actual e inicia una nueva conversión (si los bits ADCH son iguales a un valor distinto de todos en un segundo). Lo mismo se aplica para ADCn\_ADCSC1B.

La tabla 3.1, muestra los campos de los registros ADCn\_ADCS1A y ADCn\_ADCSC1B

### 3.4.2. Resultados de los Registros de Datos A y B (ADCn\_ADCRA y ADCn\_ADCRB)

En operación de 12 bits, el ADCR [14:3] contiene el resultado para una conversión de 12 bits. En el modo de 10 bits, ADCR [12:3] contiene el resultado de la conversión para 10 bits y ADR [14:13] son cero. Igualmente, cuando se configura en el modo de 8 bits, el resultado se encuentra en ADCR [10:3] y ADR [14:11] son cero. La figura 3.5 , muestra las direcciones de los registros ADCn\_BASE + B y ADCn\_BASE + C.



Tabla 3.1: Descripción de los campos de los registros ADCn\_ADCS1A/B

Fase	Descripción
15-8	Campos reservados, leen y escriben como cero
7 COCO	Bandera de conversión completa, cuando este bit es igual a 0 la conversión no es completada, si este es igual a 1 la conversión es completada
6 AIEN	Este bit habilita la conversión. Cuando este bit (1 lógico) es activado se habilita la interrupción completa de conversión
5 ACDO	Este bit es activado para realizar una conversión continua. Con este bit igual a 0 se realiza una conversión después de escribir en ADCSC1. Si es puesto en 1 se realiza una conversión continua después de escribir en ADCSC1
4-0 ADCH	Este campo compuesto por 5 bits, es usado para seleccionar uno de los canales de entrada. El subsistema del convertidor analógico a digital es desactivado cuando todos los bits del canal seleccionado son llevados a 1. Esta característica permite explícitamente la desactivación del ADC y el aislamiento del canal de entrada

### 3.4.3. Descripción Funcional

El modulo ADC es deshabilitado durante el reseteo o cuando los bits ADCH están en un estado activo. El modulo está inactivo cuando una conversión se ha completado y otra conversión no ha sido inicializada. Cuando está inactivo, el modulo se encuentra en estado de menor consumo.

El ADC puede desarrollar una conversión analógica digital e cualquiera de los canales seleccionables del software. En el modo de 10 y 12 bits, el voltaje del canal seleccionado, es convertido hasta un resultado digital de 12 bits. En el modo de 8 bits, el voltaje del canal seleccionado, es convertido hasta un resultado digital de 9 bits.

Cuando la conversión esta completa, el resultado es colocado en el registro de datos. En el modo de 10 bits, el resultado es redondeado a 10 bits y puesto en el registro de datos. En el modo de 8 bits, el resultado es redondeado a 8 bits y puesto en ADCR. La bandera de conversión completa (COCO) es colocado entonces y una interrupción es generada si la interrupción de conversión completa es ha sido habilitada (AIEN=1).

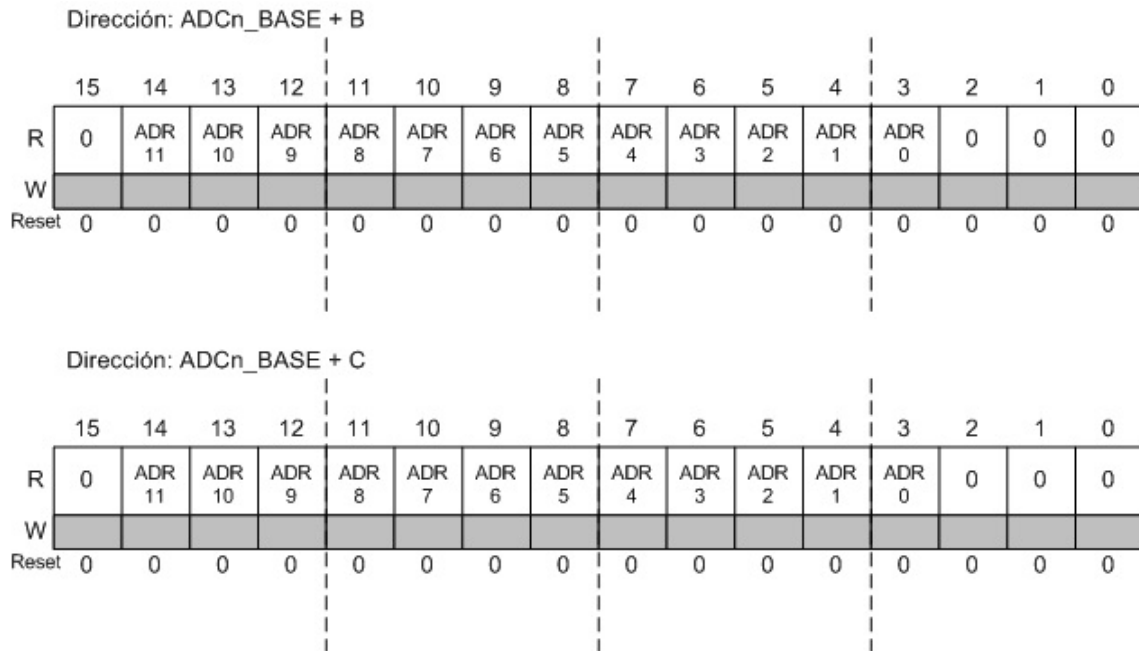


Figura 3.5: Resultados de los Registros de Datos (ADCn\_ADCRA y ADCn\_ADCRB)

### Selección de Reloj y Control Dividido

Uno de cuatro fuentes de reloj pueden ser seleccionadas como fuente de reloj del módulo del ADC. Esta fuente de reloj es entonces dividida por un valor configurable para generar el reloj de entrada del convertidor (ADCK). El reloj es seleccionado de una de las siguientes fuentes a través de los bits ADICLK.

- El reloj de bus, que es igual a la frecuencia del software es ejecutado. Esta selección predetermina permite restablecer.
- El reloj de bus dividido por dos. Para un alto alcance del reloj de bus, este puede ser dividido hasta por 16.
- El reloj asíncrono (ADACK) es generado de una fuente de reloj sin el módulo.
- El reloj asíncrono (ADACK) es generado de una fuente de reloj sin el módulo ADC. Cuando es seleccionado como fuente de reloj este se mantiene activo mientras el núcleo DSC está en el modo de paro o espera y acepta las conversiones en este modo para una operación a bajo ruido

Cualquiera que sea el reloj que se ha seleccionado, su frecuencia debe de estar dentro del rango de frecuencias para ADCK. Si los relojes disponibles son demasiado lentos, el ADC no se ejecuta de acuerdo a las especificaciones. Si los relojes disponibles son demasiado rápidos, entonces el reloj será dividido a una frecuencia apropiada, Esta división queda especificada por los bits ADIV y pueden ser divididos por 1, 2, 4 o 8.

### Disparo en Hardware

El modulo ADC tiene un hardware asíncrono seleccionable de disparo de conversión, ADHWT, que es activado cuando el bit ADTRG es puesto. La figura 3.6 muestra como se usa ADHWT para una interfaz con el PGA.

```

if (ADTRG==0) {                                     // software trigger selected
    sample_select=0;
} else {                                           // hardware trigger selected
    if (posedge SSEL[0] and NOT posedge SSEL[1]) sample_select = 0;
    else if (posedge SSEL[1] and NOT posedge SSEL[0]) sample_select = 1;
}                                                  // else NO CHANGE

```

Figura 3.6: Ejemplo de segmento de código que describe la operación de la función de la selección de muestreo.

Esto implica que el ADCn\_ADCSC1A y ADCn\_ADCRA son utilizados cada vez que el hardware no está activo. Cuando el hardware es utilizado, los bordes activos SSEL[1:0] determinan que conjunto de registros de control/ resultado son utilizados. Un borde positivo en SSEL[1] pero no en SSEL[0] selecciona a ADCn\_ADCSC1B/ ADCn\_ADCRB. Cambios simultáneos en SSEL[1] y SSEL[0] no resultan en un cambio en el conjunto de registros de control seleccionado.

En esencia, SSEL[1:0] actúan como bits de control preestablecidos para los registros control/ resultado que son usados en la siguiente conversión.

### Control de Conversión

Las conversiones pueden ser desarrolladas en el modo de 12 bits, 10 bits o 8 bits según lo determinado por los bits MODE. La conversión puede ser inicializada por cualquier disparo en software o hardware. Además, el modulo ADC puede ser configurado para

bajo consumo de energía, largo tiempo de muestreo y conversión continua.

**Inicialización de Conversiones.** Una conversión es iniciada:

- Después de escribir a ADCSC1 (con los bits ADCH no mayores a 1 segundo) si la operación en software es seleccionada.
- Después de un evento de disparo en hardware (ADHWT) si la operación de disparo en hardware es seleccionada.
- Tras la transferencia de los resultados a los registros de datos cuando una conversión continúa es habilitada.

Si las conversiones continuas son habilitadas, una conversión es automáticamente iniciada después de finalizar la conversión actual. En la operación de disparo en software, las conversiones continuas comienzan después de que ADCSC1 es escrito y continuara hasta terminar.

**Finalización de Conversiones.** Una conversión es completada cuando el resultado de la conversión es transferido hasta los resultados de los datos de registros, ADCn\_ADCRA o ADCn\_ADCRB. Esto se indica al establecer el bit COCO en ADCn\_ADCSC1A o ADCn\_ADCSC1B. Una interrupción es generada si AIEN es alto en el tiempo en que el bit COCO es activado.

**Desactivación de Conversiones.** Cualquier conversión en progreso es desactivada cuando:

- Una escritura en ADCSC1 ocurre (la conversión de corriente se interrumpe y una conversión es iniciada si ADCH no es mayor a 1s).
- Una escritura en ADCSC2, ADCCFG, o ADCCV ocurre. Esto indica que ha ocurrido un cambio en el modo de operación y que la conversión de corriente es por lo tanto invalida. No es iniciada una nueva conversión.

- El núcleo DSC es reseteado.
- El núcleo DSC entra en modo de paro con ADCK no habilitado.

Cuando una conversión es interrumpida, el contenido de los registros de datos, ADCn\_ADCRA/B, no son alterados pero continúan siendo los valores transferidos después de ser completados en la última conversión exitosa. En el caso que la conversión fuera abortada por un reset, el ADCR regresa a su estado inicial.

**Tiempo de Muestreo y Tiempo Total de Conversión.** El tiempo total de conversión depende del tiempo de muestreo (que es determinado por ADLSMP), el IP del bus de frecuencia, el modo de conversión (8 bits, 10 bits o 12 bits), y la frecuencia del reloj de conversión ( $f_{ADCK}$ ). Después de que el modulo se activa, el muestreo de entrada comienza. ADLSMP es usado para estar entre tiempos de muestro cortos (ciclos ADCK) y largos (23.5 ciclos ADCK). Cuando el muestreo esta completo, el convertidor está aislado del canal de entrada y la conversión se realiza para determinar el valor digital de la señal analógica.

Si la frecuencia del bus es menor que la frecuencia  $f_{ADCK}$ , el tiempo de muestreo para la conversión continua no se puede garantizar, cuando esta activo un muestreo corto (ADLSMP0=0). Si la frecuencia del bus es menor que 1/11 de la frecuencia  $f_{ADCK}$ , el tiempo de muestreo para conversiones continuas es habilitado (ADLSMP=1).

El tiempo máximo de conversión total se determina por la fuente de reloj elegida y la relación de división seleccionada. La fuente de reloj es seleccionada por los bits ADICLK, y la relación de división se especifica mediante los bits ADIV.

### 3.5. Modulación de Ancho de Pulso (PWM)

El PWM puede ser configurado como tres pares complementarios, seis señales PWM independientes, o sus combinaciones (como un par complementario y cuatro señales independientes). El control síncrono de pulso modulado es de borde alineado y centro alineado, de cero a 100 de la modulación.

Un contador PWM de 15 bits es utilizado para los seis canales. La resolución del PWM es un periodo de ciclo de reloj para la operación de borde alineado, y dos periodos de ciclo de reloj para la operación de centro alineado. El periodo de reloj depende de la frecuencia de la fuente de reloj del sistema o de  $3 \times$  reloj del sistema y un valor programado.

Cuando se generan señales complementarias PWM, el modulo inserta automáticamente el tiempo muerto a los pares de PWM de salida. Cada PWM de salida puede ser controlado manualmente por un generador PWM, un temporizador, resultados de la conversión del ADC, pines de propósito general (GPIO), o software, y separando el control de la polaridad de salida de la parte superior e inferior. La salida asimétrica es habilitada para cambiar el ciclo útil de trabajo del PWM, alternativamente en cada medio ciclo, sin la operación en software.

### 3.5.1. Características

- El PWM funciona en cualquier reloj del sistema o en el reloj del sistema  $3 \times$ .
- Seis señales PWM.
  - Todas independientes.
  - Pares complementarios.
  - Mezclados y complementarios.
- Características de operación del canal complementario.
  - Inserción separada de tiempos muertos para bordes de subida y de bajada.
  - Corrección de modulación de pulso a través de software.
  - Salida asimétrica de PWM sin operación de centro alineado.
  - Control de polaridad superior e inferior
- Señales en borde o centro alineado.
- Resolución de 15 bits.
- Capacidad de recarga de la mitad del ciclo.

- Software de control individual de salida PWM.
- Protección de fallas programable.
- Comparación en el control de salida de polaridad del PWM.
- Salida de control de polaridad PWM.
- Registros protegidos contra escritura.
- Fuente de alimentación seleccionable para cada par complementario de señales PWM.

### 3.5.2. Modos de operación

Se debe de tener cuidado al hacer uso de este módulo en los modos de operación. Algunas aplicaciones requieren actualizaciones periódicas de software para su correcto funcionamiento. El no hacerlo puede provocar daños en la configuración del hardware. Debido a esto, las salidas PWM son puestas en estado inactivo en el modo de paro, y opcionalmente, en los modos de espera y EOnCE. Las salidas PWM son reactivadas (asumiendo que fueron activadas anteriormente) cuando estos modos son activados. La tabla 3.2 muestra los modos de operación PWM.

Tabla 3.2: Modos de PWM cuando la operación restringida

Modo	Descripción
Paro	Las salida PWM son deshabilitadas
Espera	Las salidas PWM son deshabilitadas como una función del bit PWM_CNFG WAIT_EN
EOnCE	Las salidas PWM son deshabilitadas como una función del bit PWM_CNFG DBG_EN

### 3.5.3. Descripción de Funcionamiento

Para permitir frecuencias más bajas de PWM, el prescalar produce la frecuencia de reloj PWM, dividiendo esta frecuencia por uno, dos, cuatro u ocho. Los bits del prescalar, PRSC0 y PRSC1 en el control de registro (PWM\_CTRL), seleccionan la división del prescalar. Este prescalar es regulado y no será usado por el generador PWM hasta que el bit LDOK, es activado y un nuevo ciclo PWM comienza.

## Generador

El generador PWM contiene un contador ascendente/descendente de 15 bits, que produce señales PWM, con la alineación de software seleccionable, periodo, ciclo útil, y la inversión de la generación de señales PWM.

### Alineación y Comparación de Polaridad de Salida

El bit de alineación de borde (EDG) en el registro (PWM\_CNFG) configura la selección de centro alineado o alineación al borde en las salidas del generador PWM.

La comparación de polaridad de salida PWM es seleccionada por el bit  $CINV_n$  en la fuente de control en el registro (PWM\_SCTRL). Esta operaciones de salida se muestran en la figuras 3.7 y 3.8.

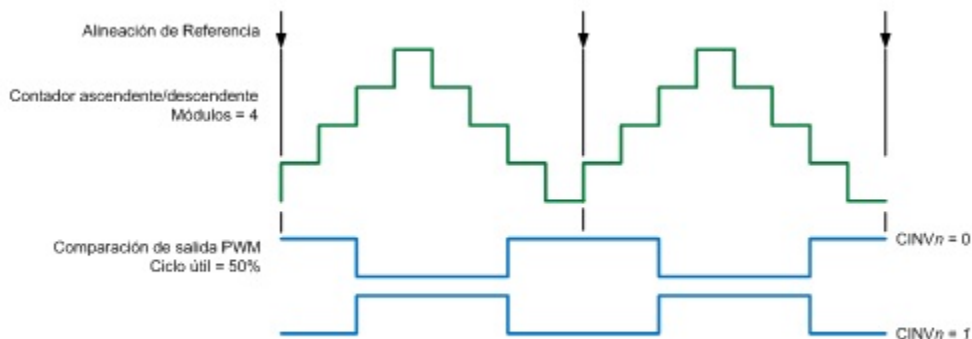


Figura 3.7: PWM de Alineación Central

La salida del PWM es llevada a un estado alto cuando el valor del PWM evaluado en el registro ( $PWM\_VAL_n$ ) es más que el valor del contador del PWM, y el comparador PWM está contando hacia abajo si el canal  $CINV_n = 0$ . O bien el comparador PWM es desactivado si el canal  $CINV_n = 1$ .

El comparador PWM es conducido a un estado bajo cuando el valor PWM contenido en el registro ( $PWM\_VAL_n$ ) coincide con el valor del contador PWM, y el contador



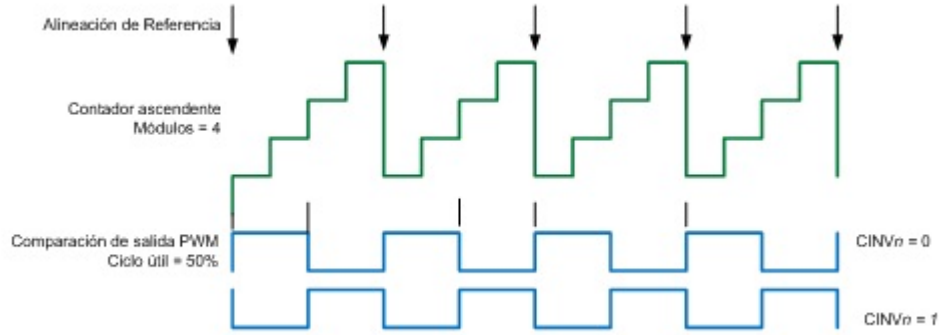


Figura 3.8: PWM de Alineación al Borde

está contando hacia arriba si el canal  $CINV_n = 0$ . O bien el comparador PWM es llevado a un estado bajo si el canal  $CINV_n = 1$ .

### Periodo

El periodo del PWM es determinado por el valor escrito en el registro (PWM\_CMOD) del modulo de conteo. El conteo del PWM es ascendente/descendente en una operación de alineación central. En este modo la mayor resolución del PWM es de  $3 \times$  ciclos de reloj del sistema si la entrada de reloj del PWM es de  $3 \times$  reloj del sistema. El modulo es la mitad del periodo PWM en los ciclos de reloj PWM.

La ecuación (3.3) muestra la forma de calcular el periodo del PWM de alineación central y la figura 3.9 ilustra este procedimiento.

$$Periodo_{PWM} = Modulo_{PWM} \times Reloj_{PWM} \times 2 \quad (3.3)$$

El contador PWM es un contador ascendente durante una operación de alineación al borde. En este modo, la resolución más alta del PWM es en un  $3 \times$  ciclo de reloj del sistema. El modulo es el periodo de el PWM en los ciclos de reloj PWM.

La ecuación (3.4) muestra la forma de calcular el periodo del PWM de alineación al borde y la figura 3.10 muestra este procedimiento.

$$Periodo_{PWM} = Modulo_{PWM} \times Reloj_{PWM} \quad (3.4)$$

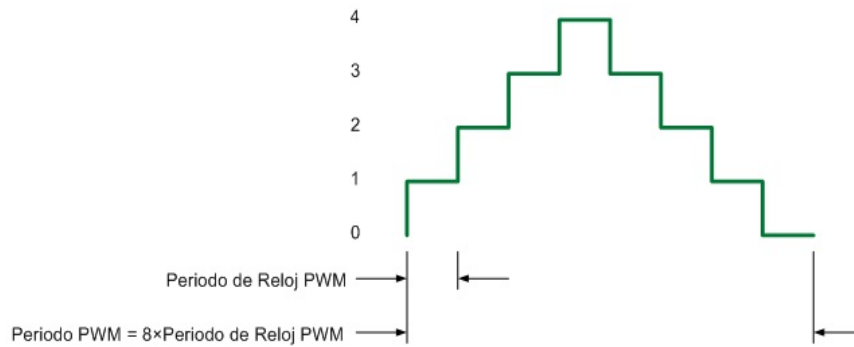


Figura 3.9: Periodo PWM de Alineación Central

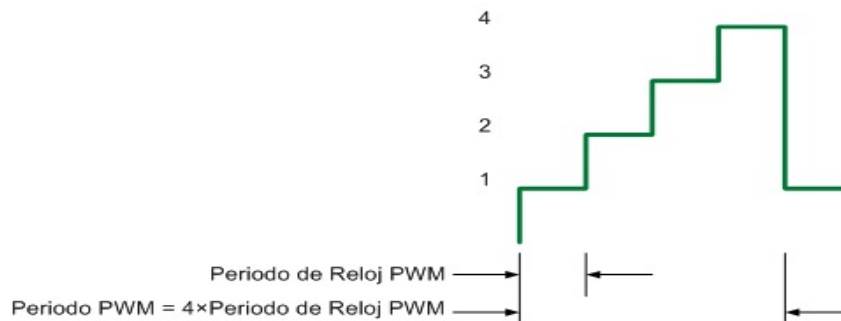


Figura 3.10: Periodo PWM de Alineación al Borde

### Ciclo de Trabajo del Ancho de Pulso

El número de 16 bits escritos en el registro de valores del PWM es el ancho de pulso en el periodo de reloj de PWM de la salida preescalar de este (o periodo menos el ancho de pulso si  $CINV_n = 1$ ). La ecuación (3.5) muestra la forma de calcular el ciclo útil.

$$C_u = \frac{Modulo_{PWM}}{Modulo} \times 100 \quad (3.5)$$

Una operación de centro alineado es ilustrada en la figura 3.11. El ancho de pulso es dos veces el valor escrito en el registro de valores de PWM con salida de alineación central en los ciclos de reloj PWM.

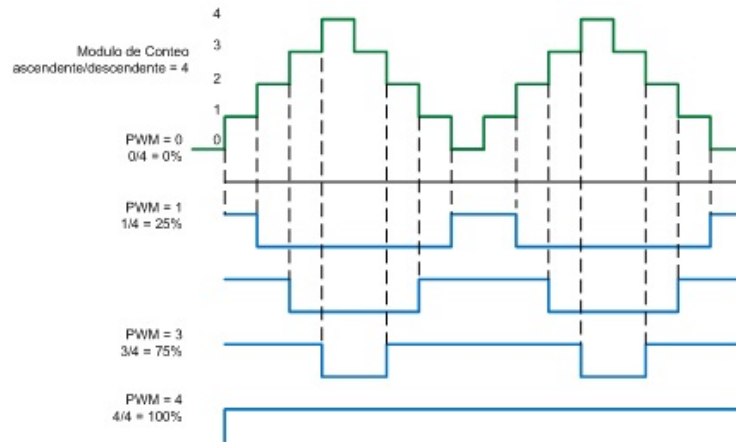


Figura 3.11: Ancho de Pulso PWM de Centro Alineado

Una operación de alineación al borde se muestra en la figura 3.12. El ancho de pulso es escrito en el valor del registro PWM con una salida de alineación al borde en los ciclos de reloj PWM.

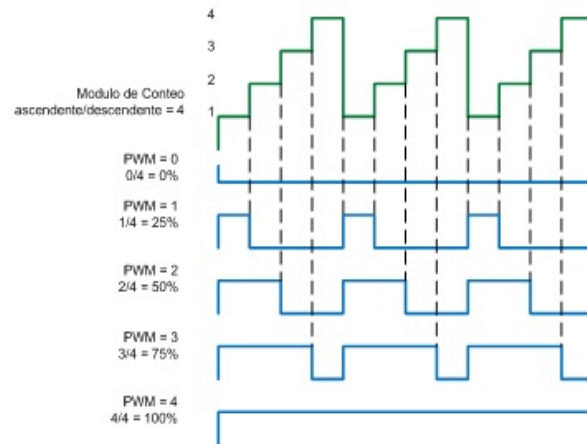


Figura 3.12: Ancho de Pulso PWM de Alineación al Borde.

### 3.5.4. Operación de Canal Independiente o Complementario

En el registro PWM\_CNFG escribiendo 1 en el bit ( $INDEP_{nn}$ ) para una operación independiente o par complementario configura un par de salidas PWM como dos canales independientes PWM. Cada salida PWM tiene su propio valor en el registro operando

independientemente de los otros canales en operación de canal independiente.

Escribiendo 0 en el bit  $INDEP_{nn}$  configura la salida PWM como un par de canales complementarios. Los PWM están en parejas en operación de canales complementarios, tal como se muestra en la figura 3.13.

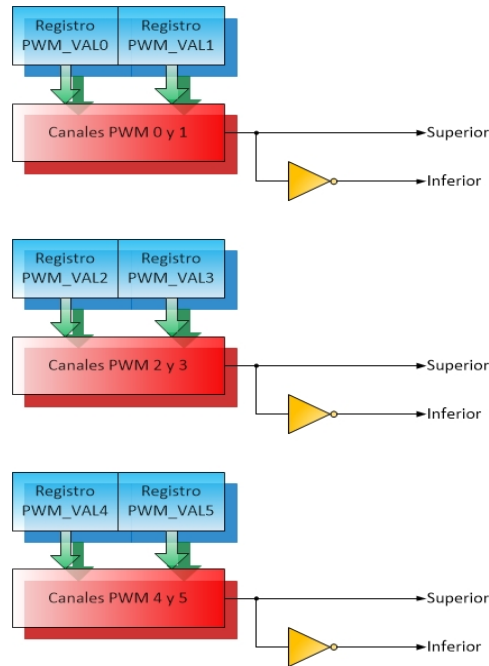


Figura 3.13: Canales pares complementarios.

Los canales complementarios controlan los transistores superiores e inferiores en un circuito inversor. La operación en canales complementarios, permite contar con las siguientes características:

- Inserción de tiempo muerto.
- Corrección de anchos de pulso superiores e inferiores para distorsiones causadas por la inserción del tiempo muerto y características de carga reactiva.
- Control de polaridad superior e inferior.

### 3.5.5. Generación de Tiempo Muerto

En el modo complementario, cada par de PWM puede ser usado para controlar los transistores superiores e inferiores. Idealmente, los pares de PWM son inversos unos respecto a otros. Cuando es activado el PWM del canal superior, el PWM del canal inferior esta desactivado y viceversa.

A fin de evitar corto circuitos entre los transistores superiores e inferiores, no debe de haber superposición en los intervalos de conducción entre los transistores superiores e inferiores. Pero las características de los transistores ocasionan tiempos de desactivación mas grandes que los tiempos de activación. Por lo tanto para evitar una conducción de los transistores superiores e inferiores, es insertado un tiempo muerto en el periodo de conmutación.

La generación de tiempo de muerto es insertada automáticamente a través de software, activando retardos entre cada par de PWM durante la conmutación. El registro (PWM\_DTIM1) de tiempo muerto especifica el número de ciclos de reloj de PWM que son usados en el retardo de tiempo muerto. Cada cierto tiempo que el generador PWM cambia de estado, el tiempo muerto es insertado. PWM\_DTIM0 controla el tiempo muerto durante la transición del estado bajo al estado alto, mientras que PWM\_DTIM1 controla la transición del estado alto al estado bajo. El tiempo muerto lleva a ambos PWM a un estado inactivo. La figura 3.14 muestra el diagrama a bloques que permite la generación e inserción del tiempo muerto, las figuras 3.15 y 3.16 muestran la generación de tiempo muerto para PWM de centro alineado y los límites del ciclo de trabajo respectivamente.

#### Control de Tiempo Muerto Superior/Inferior

En el modo complementario, cualquiera de los transistores superiores o inferiores controlan la salida de voltaje. Sin embargo, el tiempo muerto tiene que ser insertado para evitar la superposición en los intervalos de conducción entre las transiciones de superiores e inferiores. Ambos transistores en modo complementario son desactivados durante la inserción del tiempo muerto, permitiendo que el voltaje de salida quede determinado por la dirección de la corriente de la carga e introduciendo distorsión en el voltaje de salida. La distorsión se manifiesta típicamente con pobres formas de onda de salida con fallas visibles y armónicos.

La carga inductiva distorsiona la salida de voltaje manteniendo el flujo de corriente a través del diodo de protección del transistor durante el tiempo muerto. Este tiempo muerto transcurre creando una salida de voltaje variable con la dirección de corriente. Con un flujo de corriente positivo, la salida de tensión eléctrica durante el tiempo muerto es igual al voltaje inferior, controlando el transistor superior. Con un flujo negativo de corriente, la salida de voltaje durante el tiempo muerto es igual al voltaje superior, controlando el transistor inferior. Esto se deriva en el ancho de pulso original acortado por la inserción del tiempo muerto, el promedio de la salida será menor que el valor deseado. Sin embargo, cuando el tiempo muerto es agregado, este crea una distorsión en la forma de onda de la corriente de carga. Esta distorsión se ve agravada por diferentes retrasos de encendido y de apagado de cada uno de los transistores. Obteniendo información acerca de que transistor esta controlandose en un determinado momento, esta distorsión puede ser corregida.

Para un circuito típico en operación en canal complementario, solo uno de los transistores tendrá un control efectivo de la tensión de salida en un momento determinado. Esto depende de la dirección de la corriente de carga para el par de transistores (figura 3.14). Para corregir la distorsión uno de dos diferentes factores se han añadido al valor deseado de PWM, dependiendo de en que transistor (superior o inferior) esté controlando la tensión de salida.

Por lo tanto, el software se encarga de calcular los valores compensados de PWM antes de colocarlos en un par/impar de pares numerados en los registros de PWM. Cada valor de PWM par o impar en el registro ( $PWM\_VALn$ ) controla el ancho de pulso en cualquier instante. Para un determinado par de PWM, si el registro  $PWM\_VALn$  par o impar es activado, dependerá de:

- El estado de corrección del bit ( $IPOLn$ ) par/impar sin los bits ICC en el registro  $PWM\_ICCTRL$  son llevados a cero.
- La dirección del contador PWM si los bits ICC en el registro  $PWM\_ICCTRL$  son puestos en uno.
- Para corregir la distorsión en el tiempo muerto, el software puede incrementar o decrementar apropiadamente el valor en el registro  $PWM\_VALn$ .

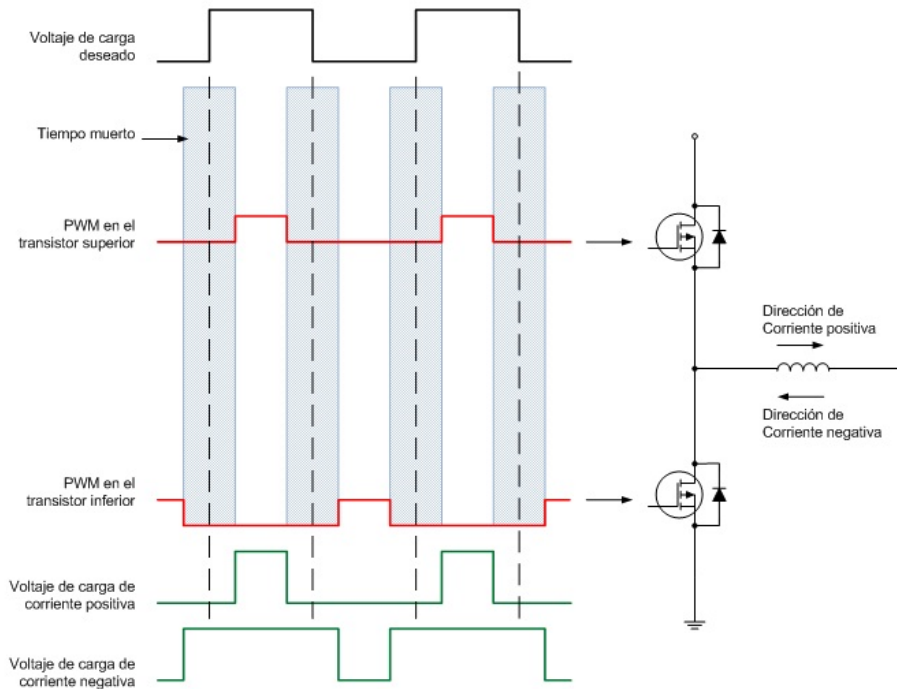


Figura 3.14: Distorsión del Tiempo Muerto

- En operación de borde alineado, decrementando o incrementando el valor de PWM por un valor de corrección igual al tiempo muerto típicamente compensado por la distorsión del tiempo muerto.
- En operación de alineación central, decrementando o incrementando el valor del PWM por una corrección igual a uno y medio del tiempo muerto compensando la distorsión provocada por el mismo.

Los bits IPOL0-IPOL2 en el registro (PWM\_CTRL) de control seleccionan cada uno de los valores PWM del registro par/impar para usarse en el siguiente ciclo PWM en modo complementario si el bit ICC $n$  correspondiente es 0.

### 3.5.6. Salida PWM Asimétrica

En modo complementario con alineación central, el ciclo de trabajo del PWM es capaz de cambiar alternativamente en cada medio ciclo. La dirección de conteo del contador PWM selecciona los registros de valor de PWM par o impar, para usarse en el ciclo de este. Para el conteo ascendente, selecciona un PWM par en el registro de valores para usarse en el ciclo PWM. Para un conteo descendente, selecciona un PWM impar en el

registro de valores para usarse en el ciclo PWM. Este tipo de salida en el PWM se puede ver en la figura 3.15

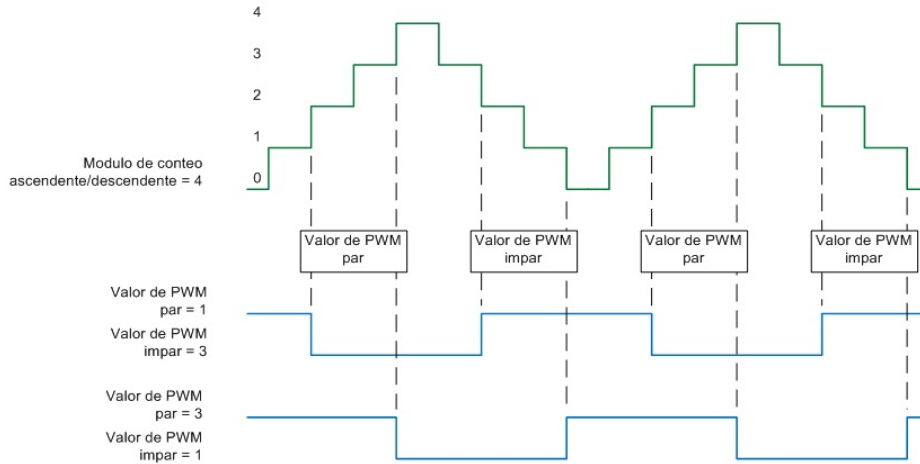


Figura 3.15: Forma de onda asimétrica en el cambio de fase del PWM de salida

### 3.5.7. Colocación de Borde Variable PWM de Salida

En modo complementario con borde alineado, el tiempo de ambos bordes de PWM puede ser controlado usando los bits  $PEC_n$  en el registro  $PWM\_ICCTRL$  y los bits  $CINV_n$  en el registro  $PWM\_SCTRL$ . El pulso de borde alineado creado por el registro del valor par y el bit asociado es XORed con el pulso creado por el registro impar y su asociado  $CINV$ . Los resultados de la XOR se introducen en la lógica complementaria y el tiempo muerto. En contraste al PWM asimétrico, el cambio de fase PWM puede pasar el límite del PWM, como se muestra en la figura 3.16.

### 3.5.8. Polaridad de PWM

Una polaridad positiva significa que el PWM se activa a la salida en alto. Inversamente, una polaridad negativa significa cuando a la salida el PWM es bajo.

La polaridad de salida de los PWMs queda determinada por dos opciones:



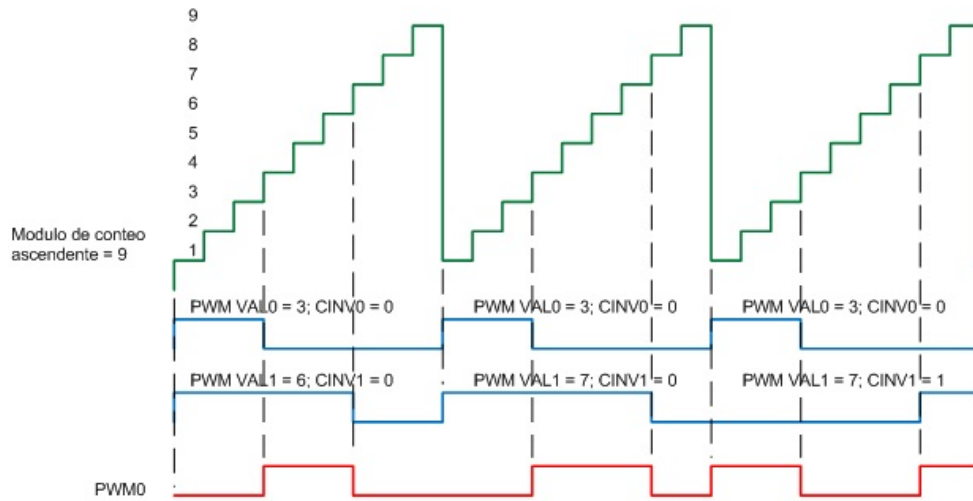


Figura 3.16: Inserción de forma de onda de borde variable al cambio de fase del PWM

- Los controles TOPNEG $nn$  de polaridad de las salidas PWM0, PWM2 y PWM4, que típicamente manejan transistores superiores del par. Cuando TOPNEG $nn$  es activado estas salidas son activadas en un estado bajo.
- Los controles BOTNEG $nn$  de polaridad de las salidas PWM1, PWM3 y PWM5, que típicamente manejan los transistores inferiores del par. Cuando BOTNEG $nn$  es activado estas salidas son activadas en un estado bajo.

Ambos bits, TOPNEG $nn$  y BOTNEG $nn$ , son configurados en el registro PWM\_CNFG. Esto se observa en la figura 3.17.

### 3.5.9. Software de Control de Salida

Habilitando el bit (OUTCTRL $n$ ) de control de salida, el PWM es controlado por el software en lugar de por el generador PWM.

En un modo independiente, con OUTCTRL $n = 1$ , el bit de salida OUT $n$ , controla el canal PWM $n$ . Habilitando y deshabilitando un bit OUT $n$  activa y desactiva el correspondiente canal PWM. Los bits OUTCTRL $n$  y OUT $n$  están en el registro (PWM\_OUT) de control del PWM. Durante el software de control de salida, TOPNEG $nn$  y BOTNEG $nn$  todavía controlan la polaridad de salida.

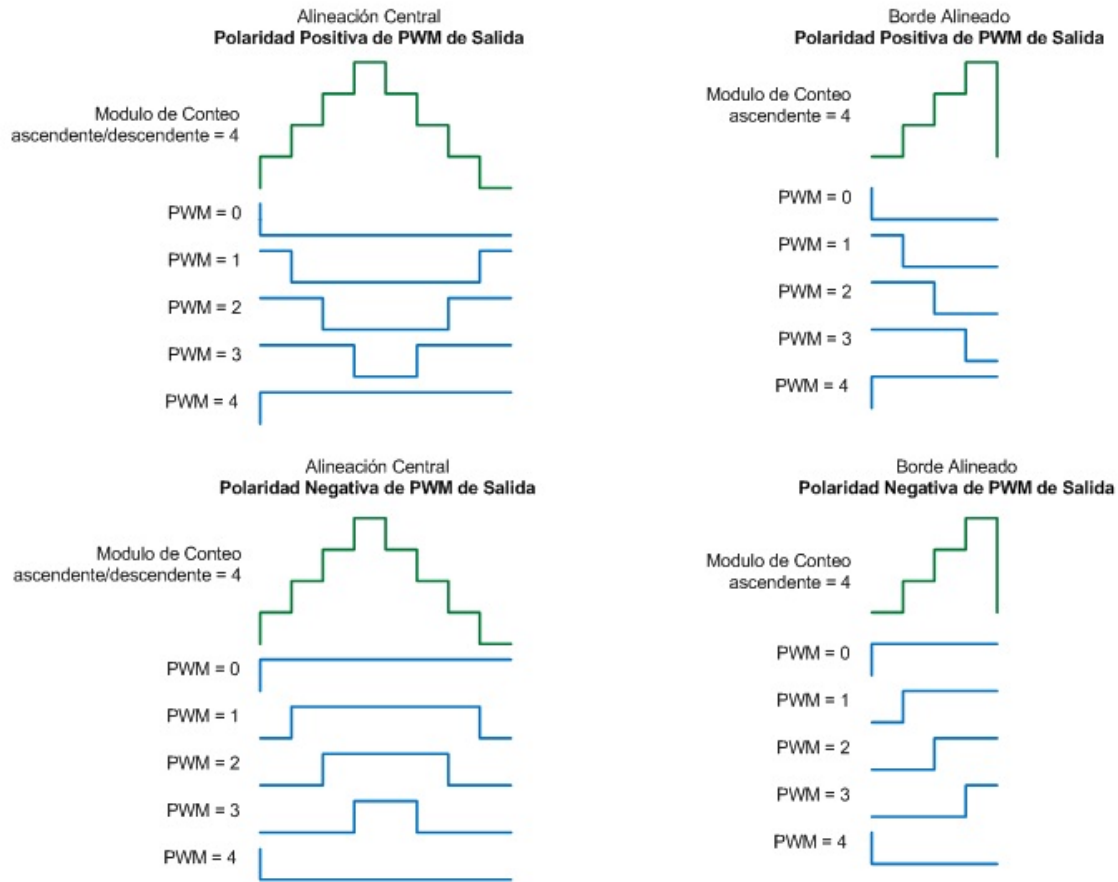


Figura 3.17: Polaridad PWM.

En operación de canal complementario, el par e impar  $OUTCTRL_n$  pueden ser idénticos y conmutados al mismo tiempo a través de una operación adecuada. Los bits  $OUT_n$  pares sustituyen las salidas del generador PWM. Los generadores de tiempo muerto agregan este tiempo siempre que un bit par  $OUT_n$  cambia. El tiempo muerto no es insertado cuando el bit impar  $OUT_n$  cambia. El bit par  $OUT_n$  controla los canales complementarios pares cuando el bit impar  $OUT_n$  es habilitado. Sin embargo, el bit par  $OUT_n$  sigue controlando los pares de canales complementarios con el impar  $PWM_n$  desactivado si el bit impar  $OUT_n$  es deshabilitado. En otras palabras, habilitando el bit impar  $OUT_n$  hace que su correspondiente  $PWM_n$  sea su complemento, mientras que deshabilitando el bit impar  $OUT_n$  desactiva el  $PWM_n$  impar, esto se observa en la figura 3.18.

Habilitando los bits  $OUTCTRL_n$  no se deshabilita el generador PWM. Este continúa funcionando, pero ya no controla los pines de salida. Cuando los bits  $OUTCTRL_n$  son

deshabilitados, las salidas del generador PWM toman el control de las salidas PWM en el comienzo del siguiente ciclo PWM, esto se observa en la figura 3.18.

### 3.5.10. Generador de Carga

#### Carga Habilitada

El bit (LDOK) de carga habilita el generador PWM con:

- Un divisor prescalar de los bits PRSC1 y PRSC0 en el registro (PWM\_CTRL) de control.
- Un periodo PWM del modulo de conteo PWM en el registro PWM\_CMOD.
- Un ancho de pulso de todos los valores de los registros PWM\_VAL $n$ .

LDOK previene la recarga simultanea de estos parámetros del PWM. Habilitando LDOK permite el preescalado de los bits, cargando los registros PWM\_CMOD y PWM\_VAL $n$ , que son usados por el generador PWM, al comenzar la recarga del siguiente ciclo.

#### Frecuencia de Carga

Los bits LDFQ3, LDFQ2, LDFQ1 y LDFQ0 en el registro PWM\_CTRL seleccionan una frecuencia de carga integral de una a 16 recargas del PWM. Los bits tienen efecto en cada recarga PWM, independientemente del estado del bit LDOK. El bit HALF en el registro PWM\_CTRL controla medio ciclo de recarga para alineación central. Si el bit HALF es habilitado, ocurre oportunamente una recarga al comienzo del ciclo PWM y en la mitad del ciclo PWM. Si este bit no es habilitado solo ocurre una recarga al comienzo del ciclo.

Cargando un nuevo modulo en un medio ciclo obligara al contador a un nuevo valor menos uno del conteo del siguiente ciclo de reloj PWM.

## 3.6. Programación del Control

En este apartado se presenta el desarrollo del algoritmo de control para velocidad y torque. En la figura 3.19, se presenta el ambiente del programa Code Warrior en el modo

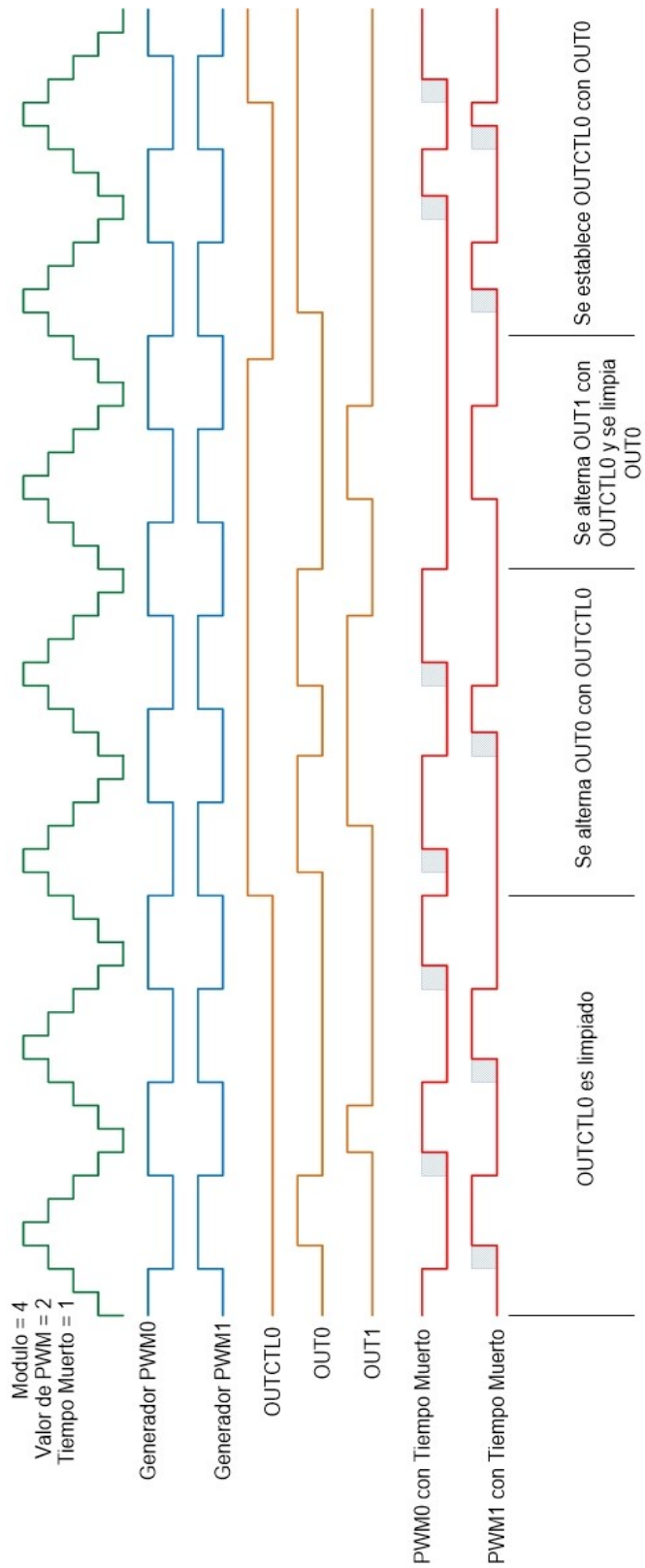


Figura 3.18: Software de Control de Salida en Modo Complementario

de programación Procesador Experto. Este programa está diseñado para trabajar en el entorno de Windows XP, por lo que comúnmente se pide el nombre del proyecto. También se selecciona el tipo de microcontrolador, que para este caso, es un DSP56F8006.

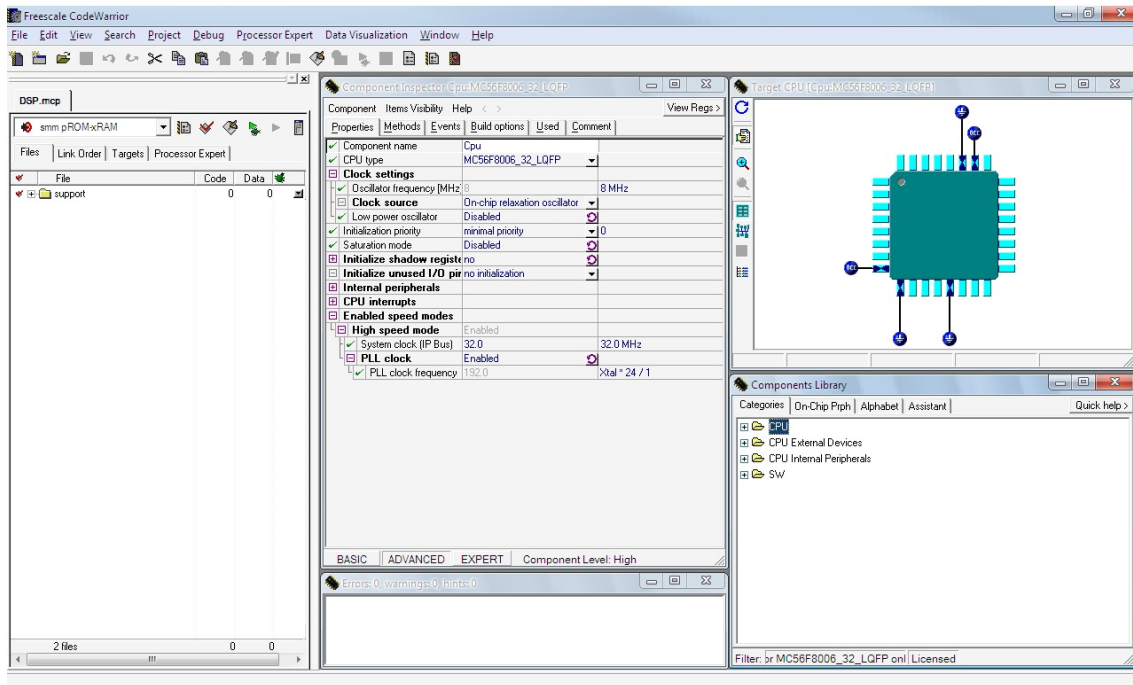


Figura 3.19: Ambiente de Programación en Procesador Experto

En este ambiente se seleccionan y cargan los componentes de los periféricos a utilizar en el proyecto, localizados en la librería de componentes, figura 3.20.

Los componentes son seleccionados con un doble clic sobre cada uno. Al escoger un componente este se agrega al árbol de la izquierda de la pantalla del programa. Cada componente contiene un inspector, el cual permite configurar el componente, En la figura 3.20 se puede observar la configuración del PWM, el cual es programado para trabajar en alineación central, y en modo complementario. Además es posible agregar el ciclo de trabajo, la frecuencia de conmutación y el tiempo muerto. Esta ventana de inspección contiene los canales de PWM, los cuales pueden ser también establecidos en modo de comparación, control en software y enmascaramiento de fallas.

De la misma forma se agregan componentes los demás componentes que son utilizados para la elaboración de los dos algoritmos de control, estos componentes son:

- Un bit.

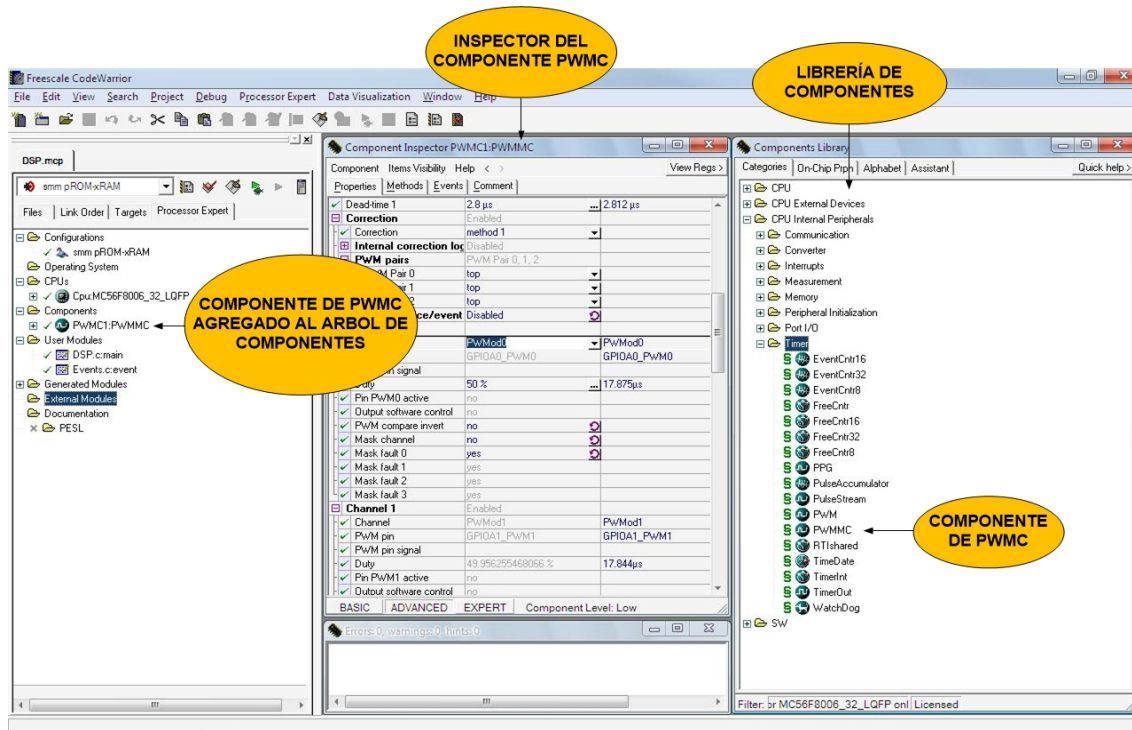


Figura 3.20: Componentes de los periféricos de control

- 5 Interrupciones externas.
- Un contador.
- Un ADC.
- Un componente PWM en modo complementario para 6 canales.

### 3.6.1. Bit de entrada/salida de propósito general

Este bit es configurado como entrada, y es activado cuando el interruptor de tres tiros dos polos, de la etapa de potencia es activado, permitiendo que sea el que arranca el sistema, y comience la conmutación del motor. El bit se encuentra en la ventana de librería de componentes, en la carpeta de periféricos internos del CPU.

La función *Bit\_GetRawVal* lee el estado de registro de entrada sin importar la dirección del pin, esto con el propósito de que inicie la conmutación sin importar si el interruptor está en la posición izquierda o derecha.

En la ventana de inspector de componente del bit, se configuran las propiedades de: el pin para entrada/salida (*PIN for I/O*) en el puerto F bit 0 (GPIOF0\_XATL) y *pull resistor* como *pull up*. La figura 3.21 muestra el procedimiento antes planteado.

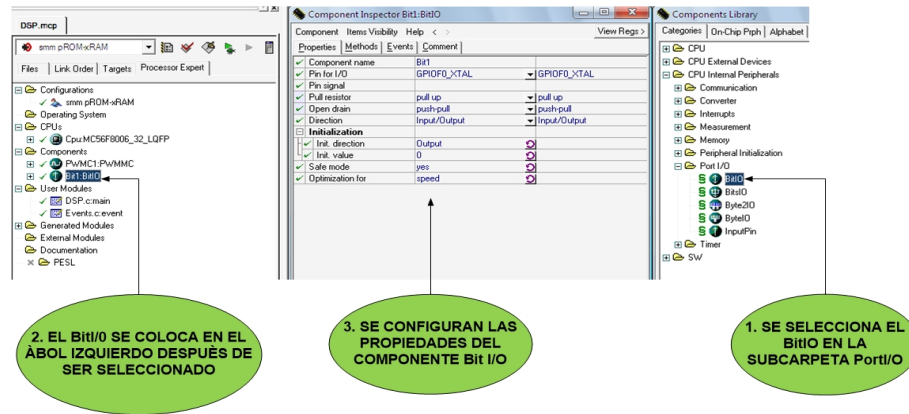


Figura 3.21: Configuración del bit de propósito general

### 3.6.2. Interrupciones Externas

Las 5 interrupciones externas utilizadas en el control cumplen diferentes propósitos, con funciones para la programación de los sensores de efecto hall, y para el aumento o reducción de la velocidad del motor.

Estos componentes implementan una interrupción externa. La interrupción será causada por una señal en un pin. Cuando este componente se activa se ejecuta el evento ligado al mismo. El componente de interrupción se encuentra ubicado en la librería de componentes, en la subcarpeta de interrupciones (*Interrupts*). En la ventana del componente de interrupción se configura las propiedades de: Pin, donde el puerto conectado al sensor A se configura en el puerto B en el bit 2 (GPIOB2\_ANA2\_ANB2\_TIN2), y la propiedad de generación de interrupción en borde ascendente (*rising edge*) también es configurada.

De la misma manera se configuran los sensores B y C, donde los únicos cambios entre uno y otro ocurre en el pin, que es el bit 3 del puerto B para el primero y el bit 6 del puerto C para el segundo. La figura 3.22 muestra la configuración de las interrupciones para el sensor A.

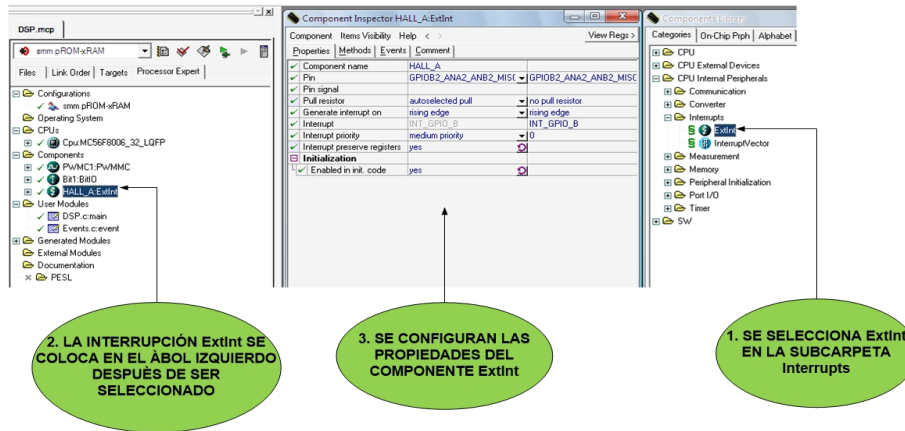


Figura 3.22: Configuración de Interrupciones

Las interrupciones utilizadas en los botones pulsadores, trabajan en borde descendente y están configuradas en los bits 1 y 5 del puerto B.

### 3.6.3. Contador libre

El contador de libre es utilizado para calcular las revoluciones por minuto del motor, a partir de los pulsos proporcionados por cada sensor. Este componente cuenta libremente durante un determinado tiempo, esto es, haciendo uso de 32 bits cuenta eventos que ocurren en intervalos de tiempo regulares. Los eventos con causados por un temporizador en hardware que es activado por el Procesador Experto para generar los eventos con una frecuencia y periodo específico.

El componente puede retornar el tiempo medido (del último reinicio) en número de oscilaciones del oscilador principal o en microsegundos o milisegundos. Cualquier anomalía provocada durante el conteo, provoca un error y el contador es reiniciado.



El byte *FC32\_GetTimeReal* regresa el tiempo en segundos desde el ultimo evento como un número (punto flotante) real. Al hacer uso del byte *FC32\_GetTimeReal* cada vez que transcurren  $4\mu s$  (resolución configurada en el inspector del componente), se obtiene el tiempo transcurrido hasta que se contaron 6 pulsos provenientes de los sensores. La figura 3.23, muestra la ubicación del contador libre, así como propiedad de resolución configurada.

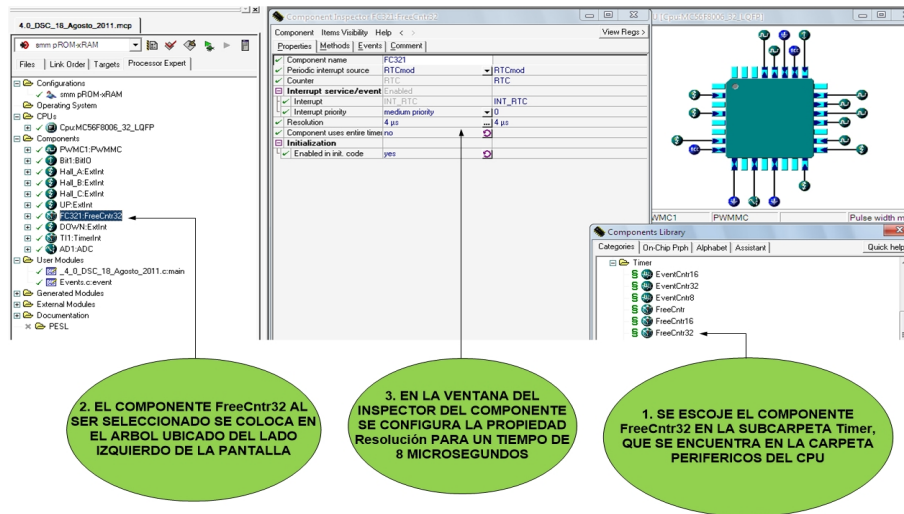


Figura 3.23: Configuración del Contador

### 3.6.4. Convertidor Analógico–Digital

El convertidor analógico-digital es utilizado para obtener una lectura de la corriente de bus ( $I_{BUS}$ ) y por consiguiente realizar el algoritmo de control para el par del motor, este hace uso de una resolución de 8 bits, y es utilizado en el canal 7, que se encuentra en el bit 1 del puerto C, con un tiempo de conversión configurado en las propiedades del componente ADC de  $26,061\mu s$ .

El byte *AD1\_Start()* comienza una conversión continua en todos los canales activados en el componente inspector. Cuando cada medición en todos los canales ha terminado, el evento *OnEnd* es llamado. El byte *AD1\_GetValue* proporciona el último valor medido para todos los canales.

La figura 3.24 muestra la ubicación y configuración en el componente inspector del ADC, donde se puede observar las funciones de los bytes *AD1\_Start()* y *AD1\_GetValue*.

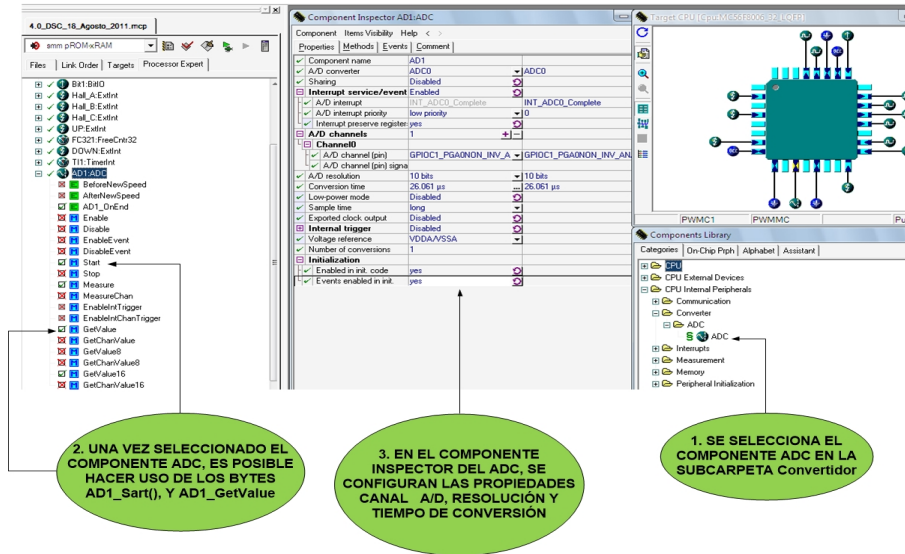


Figura 3.24: Componente ADC

### 3.6.5. Modulación de Ancho de Pulso Complementario

Este componente implementa la modulación de ancho de pulso para el control de motores (PWMMC). El PWMMC puede ser configurado como tres pares complementarios, seis señales independientes PWM, o sus combinaciones, como complementario y cuatro independientes. Con soporte para una modulación de 0 a 100 %, con control síncrono de pulso modulado en operación de borde ascendente o alineación central.

Un contador PWM de 16 bits, 15 bits o 12 bits es aplicado a los seis canales. La resolución del PWM es un periodo de reloj para una operación de borde ascendente y uno o dos periodos para una operación en centro alineado. El periodo de reloj depende de la frecuencia IPBus y el preescalador programable. Cuando se generan señales complementarias, el modulo inserta automáticamente el tiempo muerto a los pares de salida del PWMMC. Cada salida PWMMC puede ser controlado por un generador PWMMC o a través de software manualmente.

En la ventana del componente inspector, se configura el PWM en modo complementario (pares 0, 1 y 2) con polaridad negativa para las partes superiores de los pares y polaridad positiva para las partes inferiores; a una frecuencia/periodo de 20 kHz y un tiempo muerto de  $2,8\mu s$ , Los puertos configurados son: GPIOA2\_PWM2 (canal 1), GPIOB0\_ANB13\_PWM3 (canal 3), GPIOA5\_PWM5 (canal 5) y GPIOA6\_ANA1 (canal 6). La figura 3.20 muestra algunas de las propiedades antes mencionadas.

### 3.6.6. Detalle del Algoritmo de Control

La figura 3.25 describe la constitución del algoritmo de control para el motor sin escobillas de 3 fases de corriente directa. El código generado como se ha explicado es a través del uso de los componentes que se obtienen utilizando la plataforma de programación para el microcontrolador 56F8006.

Las librerías que son declaradas y que se muestran en el código del programa, se incluyen de manera automática al generar el archivo cuando se comienza a programar y también al ir agregando cada uno de los componentes que se van a utilizar. Las variables que se utilizan se declaran en los archivos main o events, estas tienen que ser escritas, pues son parte de la programación en lenguaje C que se realiza. Las variables que hacen referencia a los PWM se declaran utilizando la instrucción `define`, ya que esta permite etiquetar cada una de las direcciones en código hexadecimal, que se tienen para los componentes PWM. El bloque que hace referencia a entradas y salidas, son las instrucciones necesarias para comunicar el microcontrolador con el microcontrolador MC33927, el cual envía las señales a la etapa de potencia.

La detección de posición permite saber en donde se encuentra el rotor a través de las señales que mandan los sensores, siendo así posible iniciar la conmutación, función que permite que el motor comience a operar. También cada vez que se activa un par de bobinas, estas mandan una corriente la cual es registrada por amplificadores operacionales que se encuentran en la etapa de potencia, y que mandan una señal en voltaje que corresponde al nivel de corriente que se encuentra en cada una de las fases y que es cuantificado a través de un convertidor analógico-digital usando recursos de programación.

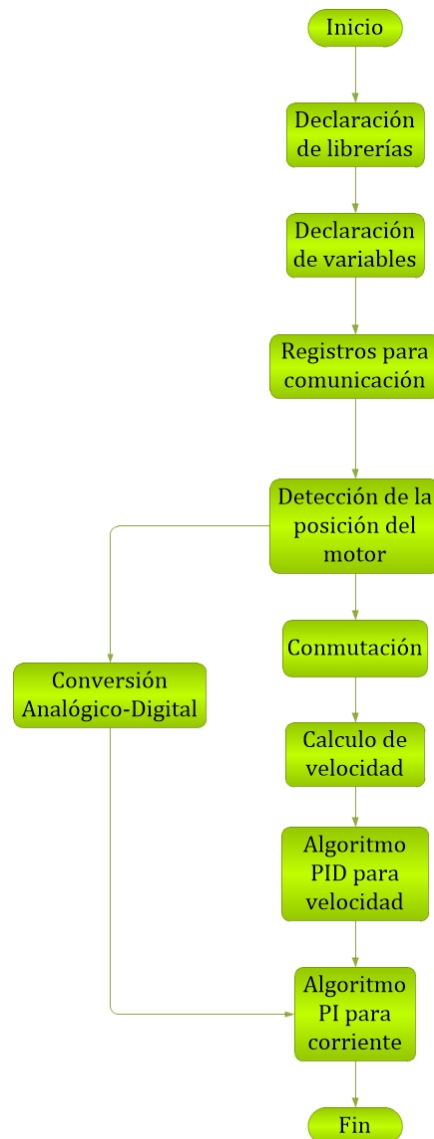


Figura 3.25: Diagrama de flujo del algoritmo de control

El calculo de la velocidad inicia cuando cada uno de los sensores mandan un estado lógico alto (1), lo cual indica que cuando se detecten los 6 pulsos, correspondientes a una rotación mecánica, y haciendo uso del contador libre (FC321), se realiza el calculo de la velocidad a través del tiempo que tardo en dar una rotación completa.

El algoritmo de velocidad calcula primero un error, que es igual al punto de ajuste de una velocidad de entrada menos la velocidad calculada (RPM); con este error es posible calcular el controlador PID, donde para la parte integral se hace uso de un calculo de velocidad anterior y de uno posterior; y para la parte derivativa se utiliza una error anterior

y uno posterior. La salida del algoritmo de control para velocidad se utiliza como punto de ajuste para el segundo controlador, que es el de corriente, el error para este segundo algoritmo de control se realiza haciendo la sustracción de salida antes mencionada menos la corriente de bus; posteriormente se procede a calcular la parte integral del controlador, y finaliza el programa.

### Unidad MC33927

El microcontrolador MC33927 es un dispositivo que permite la comunicación entre la etapa de potencia y la etapa de control, los registros de comunicación para este microcontrolador son:

**SPI\_Init:** Esta función inicia la transmisión de datos a través de la bus SPI (del ingles, *Serial Peripheral Interface*) del microcontrolador MC33927. La librería `#include "SPI_00361_01.h"` incluye los archivos que activan este bus, ejemplo de esto es el `SPI_Init`.

**MC33927\_GetSR0:** Esta función envía un comando al controlador MC33927 a través del bus SPI que hace que el controlador envíe el estado de registro 0 que contiene el estado de los bits de la unidad MC33927. La función almacena los datos en una variable de 8 bits. La función `MC33927_GetSR0(&mutdStatusRegister0)` realiza lo anteriormente mencionado.

**MC33927\_GetSR1:** Esta función envía un comando al controlador MC33927 a través de SPI que hará que el controlador envíe el estado de registro 1. Este registro contienen los bits de modo de la unidad MC33927. La función almacena los datos en una variable de 8 bits. La función `MC33927_GetSR1(&mutdStatusRegister1)` realiza lo anteriormente mencionado.

**MC33927\_GetSR2:** Esta función envía un comando al controlador MC33927 a través de SPI que hace el controlador envíe un estado de registro 2. Este estado registro contiene los bits de mascara de interrupción (`mutdMaskInterrupts.B.xx`). Los datos recibido se almacenan en una variable de 8 bits. La función `MC33927_GetSR2(&mutdStatusRegister2)` realiza lo anteriormente mencionado.

**MC33927\_GetSR3:** Esta función envía un comando al controlador MC33927 de el bus SPI que hace que el controlador envíe un registro de estado 3. Este registro contiene el valor del tiempo puerto establecido en la unidad MC33927. La función

`MC33927_GetSR3(&mudtStatusRegister3)` devuelve los datos recibidos como una variable de 8 bits.

**MC33927\_ModeCommand:** Esta función envía una orden al MC33927 a través del bus SPI que establecerá un modo de bloque, modo de fallas y modo para desaturación. Los bits `mudtModeCommands.B.xx` son llamados en una variable o independientemente.

**MC33927\_ClearFlags:** Esta función envía dos comandos al MC33927 vía SPI que limpia las banderas. Los bits `mudtClearFlags.B.xx` pueden ser configurados en una variable de 8 bits o independientemente.

**MC33927\_Deadtime:** Esta función envía una orden al MC33927 a través de SPI referente al tiempo muerto impuesto por el controlador primario. Después de comando debe establecer el tiempo muerto deseado; donde las instrucciones que permiten esto son `mbyDeadtime=MC33927_GetSR3` y `Cpu_Delay1000US`.

### Detección de la Posición del Motor y Conmutación

La función `void posicion` detecta cada uno de los sensores, a través de seis condiciones `if` que indican que sensores se encuentran activados. Esto permite que en la conmutación que depende de la variable `n` sea posible la activación de un par de transistores utilizando 2 PWM.

La conmutación inicia con la instrucción `ciclo=sector` donde, `sector=n+1`, debido a esto la función `void conmutacion` cuando `ciclo` es igual a una constante que va del 1 al 6, y a través del uso de condiciones se activa un PWM de entrada y uno de salida, por ejemplo si `ciclo==1` una señal de PWM de entrada se encuentra en el transistor superior conectado a la bobina A y la otra señal PWM se encuentra en el transistor inferior conectado a la bobina C (para mas información observar la figura 9 del anexo D) siendo esta la manera en como conmuta el motor.

### Calculo de velocidad

El calculo de velocidad comienza en el instante en que cada una de las interrupciones dedicadas a cada sensor se activan, cuando esto ocurre la función `velocidad` es mandada llamar, donde dentro de esta función se incrementa la variable `pulsos`. Cuando `pulsos` es igual a 6, el contador libre `FC321_GetTimeReal(&tiempo)`, guarda en la variable `tiempo` el

valor equivalente en microsegundos del tiempo en que `pulsos` llegó al valor de 6. Una vez realizado lo anterior se calcula la velocidad en revoluciones por minuto que se guarda en la variable llamada `RPM`.

### Algoritmo PID para velocidad

Un algoritmo PID de velocidad requiere del cálculo de una velocidad de retroalimentación y una velocidad de entrada. La variable `set_point` se incrementa cada vez que es pulsado un botón y la variable `RPM` como ya se mencionó es calculada; y utilizando ambas variables se calcula el error.

Las constantes `kp`, `ki` y `kd` se declaran con valores de 2, 5 y 0.05 respectivamente. Con la constante `kp` se calcula la parte proporcional, realizando el producto de la constante `kp` por el `error`, donde el resultado se asigna a la variable `P`.

Con la constante integral `ki`, la variable `RPM` y la constante `T` se calcula la parte integral, que se asigna a la variable `I`. La parte derivativa se calcula utilizando el `error`, y las constantes `kd` y `T`, asignado el resultado a `D`. La suma de `P`, `I` y `D` es el resultado del algoritmo de velocidad que se asigna a la variable `c_t`.

### Algoritmo PI para corriente

Controlando la corriente es posible controlar el torque, así pues si se detectan los cambios de corriente en cada una de las fases, convirtiendo estos valores utilizando un convertidor analógico digital (variable `i_bus`), a valores entendibles por el microcontrolador y con la salida del controlador de velocidad (`torque=c_t`), es posible calcular un error (`error_v`) para el algoritmo PI de corriente.

La parte proporcional (`P_v`) se calcula con el producto de la constante proporcional (`kp_v`) por el error. En cuanto a la parte integral (`I_v`) se realiza un método algebraico utilizando las constantes `ki_v`, `Ti_v` y las variables `torque` e `i_bus`.

La suma de las variables `P_v` y `I_v` proporciona el resultado (asignado a `ciclo_v`) para el algoritmo de control de corriente.

El programa completo puede ser consultado en el anexo E del presente trabajo, donde se presenta cada parte del programa en los apartados correspondientes al diagrama de la flujo presentado en esta sección.

### 3.7. Características del Control

El control generado tiene la característica de poder ser implementado en cualquier microcontrolador de la familia 56F80xx, trabajando en conjunto con una unidad de transmisión de datos como el MC33927, pues los parámetros de comunicación via SPI fueron configurados para este controlador. Las partes del código dedicados a la conmutación, posición, velocidad, y corriente pueden ser implementados en cualquier DSP de freescale, siempre y cuando sean utilizados sensores de efecto hall para la retroalimentación.

Las limitaciones que se pueden presentar es que la etapa de potencia utilizada esta diseñada para el tipo de motor utilizado, aunado a esto, la corriente de retroalimentación que es detectada en la etapa de potencia utiliza amplificadores operacionales sin etapas de reducción, lo que implica que si se intenta utilizar un motor de mayores dimensiones, es posible que la etapa quede averiada debido al exceso de corriente que puede entregar un motor de mayor tamaño. Otra limitación que se presenta en el presente algoritmo es que solo es aplicable a motores con distribución trapezoidal, independientemente del número de polos que pueda llegar a tener, el algoritmo no esta diseñado para arrancar motores de imanes permanentes con fuerza contraelectromotriz senoidal pues el código no trabaja a base de espacios vectoriales.

El programa puede ser aplicado siempre y cuando el código sea aceptable en la plataforma de CodeWarrior 8.3, pues la generación del código esta dado por este tipo de plataforma, si se requiere utilizar el mismo código para otro tipo de microcontrolador de otra familia no incluida en esta plataforma u otro de una marca diferente, solo hay que seguir la lógica de programación utilizada y tomar en cuenta las variantes en las instrucciones que puedan presentarse.



## 3.8. Sumario

El presente capítulo abordo la conmutación de 6 pasos para el motor sin escobillas de corriente directa, con esto se dio cabida al análisis de las características del microcontrolador 56F8006 con el propósito de verificar que permitirían implementar un control a lazo abierto para este motor. Posteriormente se estudiaron los periféricos del mismo microcontrolador para determinar que las atribuciones que estos poseen harían posible que se realizara un control en cascada.

Ulteriormente se describió como realizar la programación utilizando procesador experto en la plataforma de Code Warrior versión 8.3, donde se explico la manera de configurar cada uno de los componentes, tales como: Interrupciones, PWM, contadores libres, entradas digitales y conversion analógico-digital. En consecuencia se elaboró una breve explicación de cada unas de las instrucciones acerca del código de programación generado, como lo son: registros de programación, posición, conmutación, calculo de velocidad, control PID y control PI.

Con el código generado se comentaron las características que posee el algoritmo, así como las limitaciones en cuanto a implementación, generación de código y alcances que puede tener el desarrollo computacional presentado en el anexo E del presente trabajo.

El capítulo 4 presentara las pruebas y resultados que se realizaran utilizando el algoritmo computacional desarrollado en este capítulo, donde se presentaran las pruebas a las que somete el control al utilizar distintos valores para las constantes proporcional, integral y derivativa de los algoritmos mencionados en los ultimas secciones de este capítulo.



# Capítulo 4

## Pruebas y Resultados

El presente capítulo presenta las pruebas realizadas utilizando un motor LINIX de 24 V de corriente directa trifásico con un torque nominal de 0.924 Nm. Como primer punto a abordar se muestra la generación de los 6 PWM's en modo complementario con conmutación unipolar independiente, posteriormente se exponen las pruebas y resultados realizados con un control de velocidad, indispensable para un control indirecto del torque. Una vez sintonizado el control de velocidad, se realiza un lazo de control en cascada, donde la salida del controlador de velocidad es la entrada al controlador de corriente, con el propósito de controlar el flujo de corriente de bus y así tener un control sobre el torque generado por el motor.

### 4.1. Técnica de 6 pasos

La visualización de los 6 PWM que permiten aplicar la técnica de los 6 pasos, es a través de una tarjeta de adquisición de datos NI USB-6211 de NATIONAL INSTRUMENTS que proporciona 250 kS/s y la herramienta de Adquisición de Datos de MATLAB. Haciendo uso de 6 canales analógicos de esta tarjeta, que son conectados a los pines del DSP 56F8006, y con una interfaz gráfica en MATLAB, se muestran los 6 PWM generados. La figura 4.1 muestra los canales del 1 al 6, donde los pares de canales 6 y 5, corresponden a los 2 PWM dirigidos a la bobina A, los pares 4 y 3, corresponden a los 2 PWM de la bobina B, y los canales 2 y 1 se relacionan con los 2 PWM de la bobina C.

Al realizar el lazo de control de velocidad se busco a través de prueba y error la mejor respuesta de este en donde el error generado permita que la respuesta del sistema se

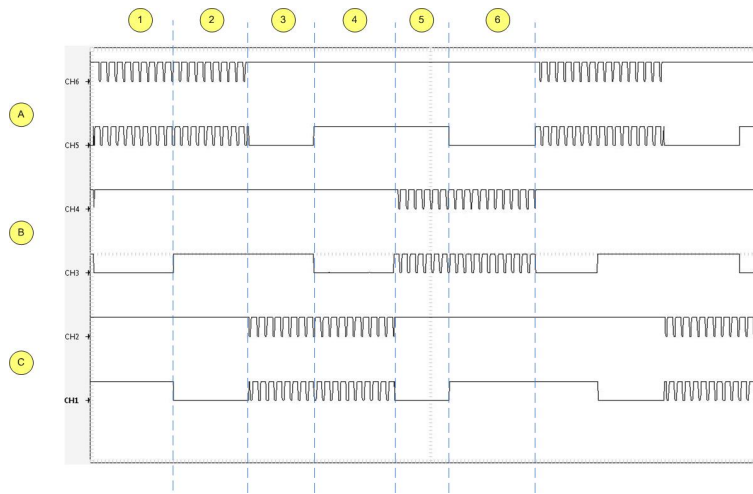


Figura 4.1: Generación de 6 PWM's

acerque al punto de ajuste impuesto. En las pruebas realizadas se usaron distintos valores para las constantes  $k_p$ ,  $k_i$  y  $k_d$ ; que permitieron observar el comportamiento del sistema y de donde aquí se muestran la que mostraron los mejores resultados.

## 4.2. Control de Velocidad

La primera prueba realizada es con una constante  $k_p = 2,00$ ,  $k_d = 0,05$  y  $k_i = 5,00$ . La figura 4.2 muestra cómo se comporta el sistema bajo estos para metros, donde el punto de ajuste primeramente cambia de 1000 rpm a 2000 rpm (línea roja en la figura 4.2) durante esta transición la salida, o sea, las rpm (línea verde en la figura 4.2) medidas muestran un sobreimpulso significativo esto es debido a la rapidez con la que el sistema trata de alcanzar el punto de ajuste. Las oscilaciones presentes en la figura muestran la forma en cómo la respuesta a la salida trata de seguir a la entrada, donde el error entre ambas es del 8.7%, debido a que el punto de ajuste en el momento en que se obtuvo la gráfica es de 1500 rpm y las revoluciones por minuto medidas son de 1369.2 rpm. La línea azul muestra el comportamiento controlador PID, que en realidad es el comportamiento a la salida del sistema y que muestra ser similar a la línea verde (rpm).

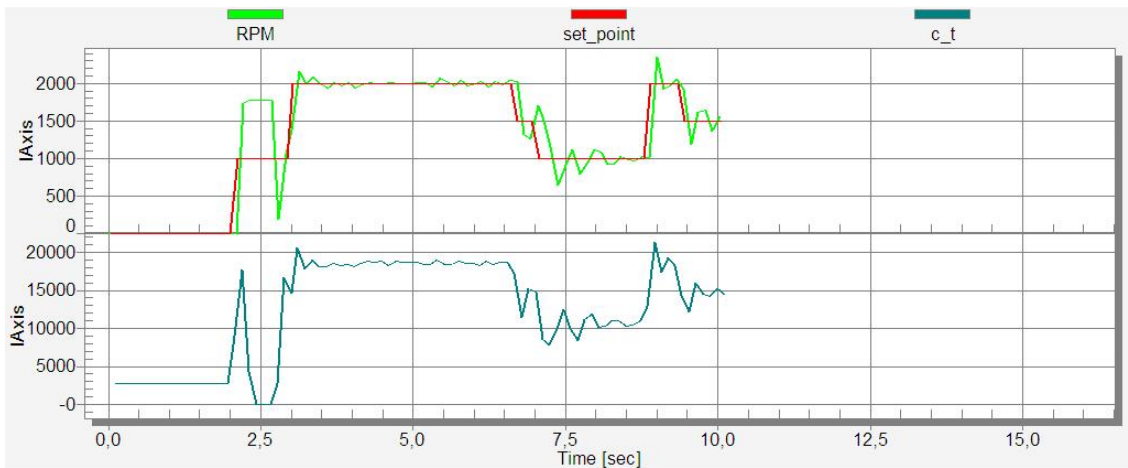


Figura 4.2: Primera prueba realizada al controlador de velocidad

La segunda prueba realizada fue cambiando los valores de las constantes proporcional y derivativa, se le asignó el valor de 0.80 a la primera y 0.03 a la segunda respectivamente. Esto se hizo con el propósito de reducir el sobreimpulso ocasionado al aumentar la velocidad del motor y reducir el error en estado estacionario, los resultados mostraron que efectivamente se generó una mejor respuesta del controlador pero aumentó la velocidad de respuesta del sistema, esto es fácil de observar en la gráfica de la figura 4.3 debido al desfase existente entre las señales de color rojo y verde. El error mostró ser del 1.02 %, ya que las rpm medidas por el software en el instante en que obtuvo la muestra tenían un valor de 2500 y la salida del sistema mostraba un valor de 2474.39 rpm.

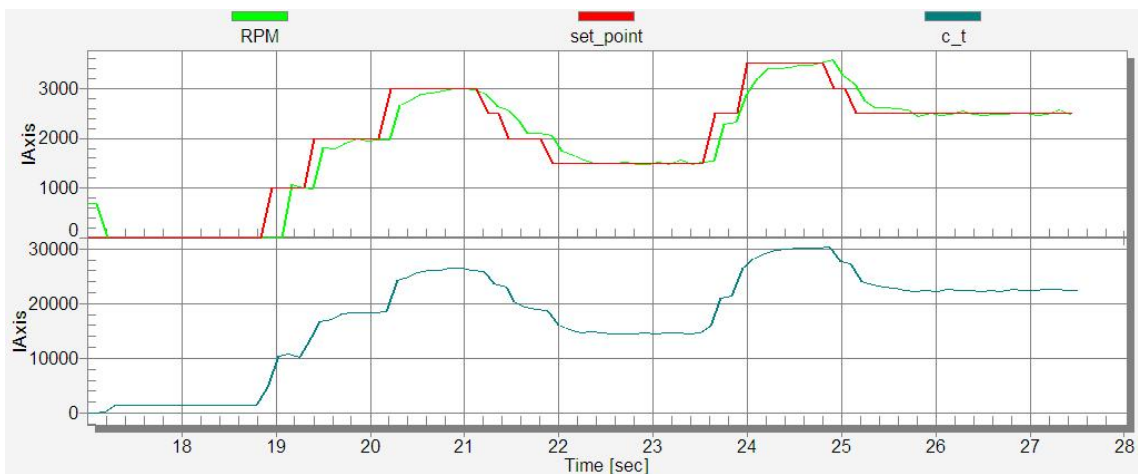


Figura 4.3: Pruebas realizadas con  $k_p = 0,80$ ,  $k_d = 0,03$  y  $k_i = 5,00$

Para reducir el tiempo de respuesta del controlador y evitar el retraso de respuesta a la salida, como el mostrado en la figura 4.3. Se le asignaron a las contantes los valores de 1.20 para la constante proporcional, 3.00 para la constante integral y 0.04 para la constante derivativa. Bajo estos nuevos parámetros el sistema mostró conservar el mismo desfaseamiento incrementando el porcentaje del error, que fue del 1.72% (figura 4.4).

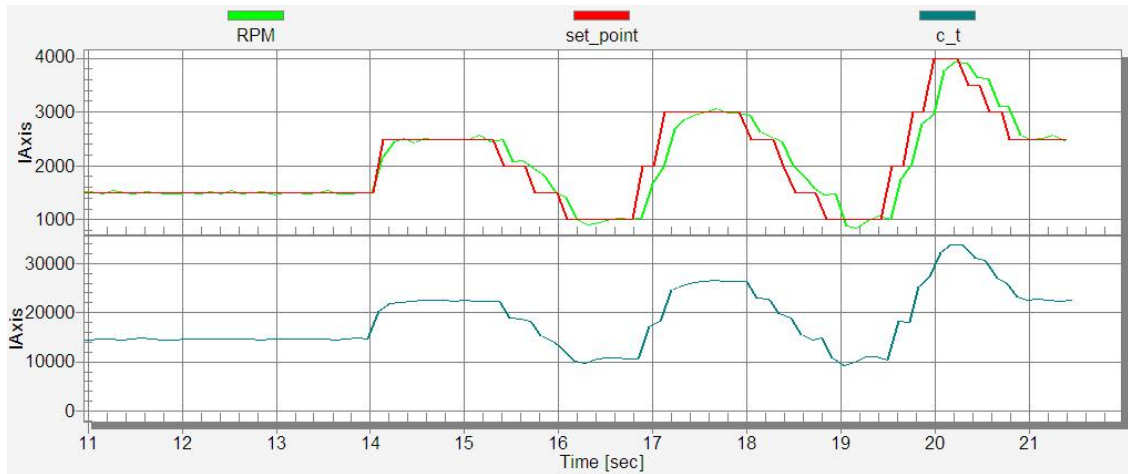


Figura 4.4: Pruebas realizadas con  $k_p = 1,20$ ,  $k_d = 0,04$  y  $k_i = 3,00$

Realizando una comparación entre las gráficas mostradas anteriormente, se puede observar que existe un mejor seguimiento del punto de ajuste por parte de la salida si las constantes son mayores, bajo esta premisa los valores son cambiados a 1.8 para  $k_p$ , 4.25 para  $k_i$  y 0.06 para  $k_d$ . Estos cambios a las constantes mostraron en la gráfica de la figura 4.5 que hay un mejor seguimiento del punto de ajuste a una velocidad constante, pero se conserva el sobreimpulso debido a la velocidad de cálculo que el controlador permite que la respuesta de salida alcance la señal de entrada, incluso debido a esto, se genera un aumento considerable del error que fue de 4.679% en el instante de obtener la gráfica.

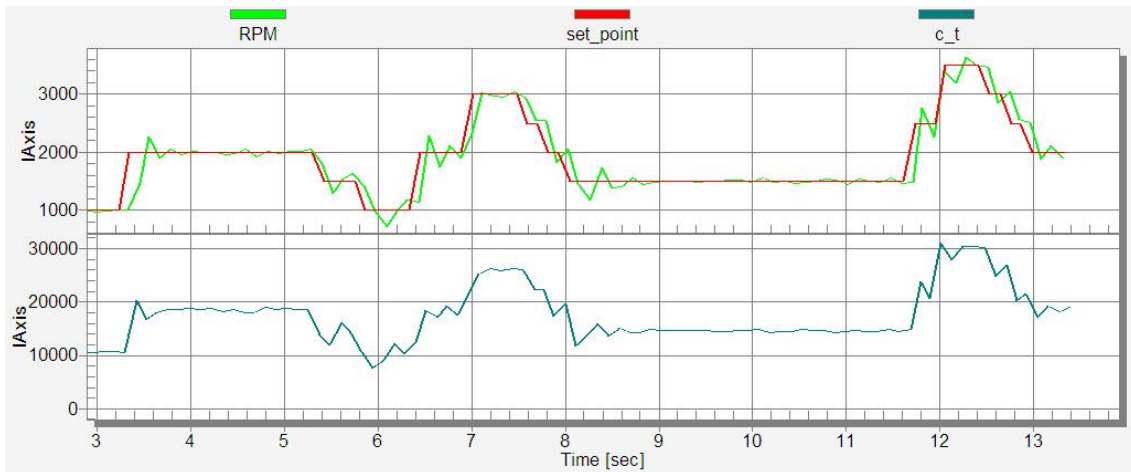


Figura 4.5: Pruebas realizadas con  $k_p = 1,80$ ,  $k_d = 0,06$  y  $k_i = 4,25$

De los resultados obtenidos para el controlador de velocidad, el que mostró tener una mejor respuesta debido a error generado de la diferencia de las señales de entrada y de salida y una rápida respuesta es el mostrado en la figura 4.2, donde la ventana de control (figura 4.6) proporciona la velocidad en a la que el sistema responde y por lo tanto, el tiempo en el que se calcula el algoritmo de control, que es de  $20.239 \mu s$ .

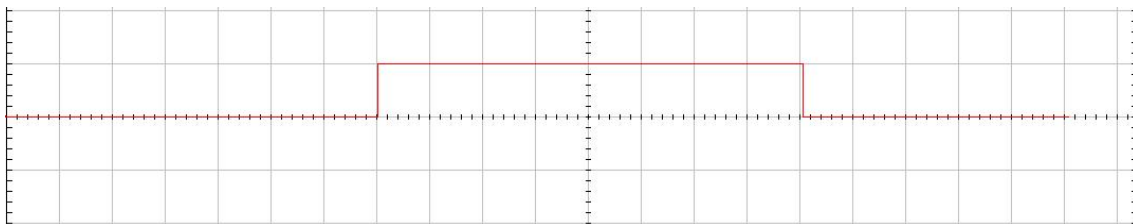


Figura 4.6: Ventana de control

### 4.3. Control en Cascada

Para implementar el control de corriente, se hace uso de un control en cascada, en donde la salida del controlador de velocidad, es la entrada al controlador de corriente, es por esta razón que primeramente se seleccionaron las constantes para el algoritmo de control de velocidad. La señal de retroalimentación para el control de corriente se toma de la corriente de bus, y aplicando un control PI se cierra el lazo de control permitiendo así, tener un control independiente e indirecto sobre la corriente. Las pruebas realizadas muestran que el control de corriente permite tener un escalamiento sobre la corriente, lo

que significa que esta no se sale de un cierto rango de valores, y admite tener un mejor control sobre el par del motor. Las pruebas realizadas no se realizaron bajo cargas constantes, simplemente se hicieron bajo cargas menores hechas al intentar parar la flecha del motor manualmente a fin de tan solo observar el cambio de la corriente y por lo tanto del torque.

Con las constantes para el algoritmo de control de velocidad con valores para  $k_p = 2.00$ ,  $k_d = 0.05$  y  $k_i = 5.00$ ; se realizaron pruebas agregando el controlador de corriente donde las constantes proporcional e integral para este controlador son 0.50 para la constante proporcional ( $k_{pc}$ ) y 0.01 para la constante integral ( $k_{ic}$ ). Al ser agregado el control de corriente en cascada al controlador de velocidad, este último no modificó su comportamiento, esto se observa en la figura 4.7, la cual muestra la respuesta del sistema para la velocidad, donde a una velocidad de 3000 rpm en la entrada y una salida de 2925.07 rpm el error es del 2.4 %, por lo que el error aumenta conforme se incrementa la velocidad aun sin el controlador de corriente.



Figura 4.7: Control en cascada donde las constantes del PID tienen valores de 2.00 (proporcional), 5.00 (integral) y 0.05 (derivativa)

Las mediciones de corriente se toman reduciendo la velocidad donde es posible ver que la corriente oscila entre 2.5 A y 7.5 A con una medición del par de 0 a 10 Nm (figura 4.8) lo cual corresponde una medición errónea del par pues el motor solo es capaz de soportar 0.925 Nm, aunado a esto la maquina trifásica mostró tender a detenerse debido a la sobreprotección en hardware que tiene, lo que significa que el flujo de corriente en el estator sin llegar a ser mayor de 2.5 A, ocasionaba que se activara la autoprotección del sistema.



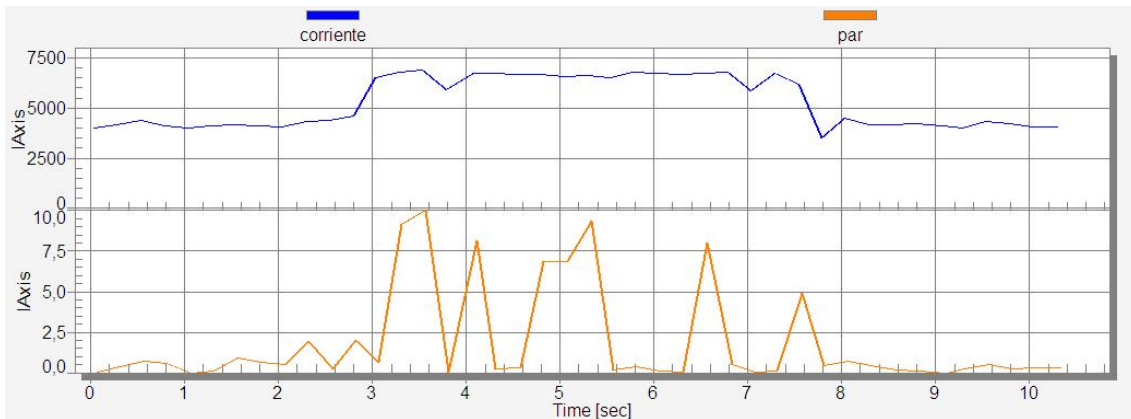


Figura 4.8: Pruebas realizadas con  $k_{pc} = 0.50$  y  $k_{ic} = 0.01$

Al cambiar el valor de las constantes en el controlador de corriente, este muestra una mejor respuesta, pues la corriente está entre 0.475 a 0.650 A con un torque que se encuentra entre 0 y 0.500 Nm donde el error para el lazo de corriente es del 5.3% , con un manejo de las constantes proporcional e integral con valor de 0.05 y 1.50. Este tipo de respuesta del control en la corriente se muestra en la figura 4.9.

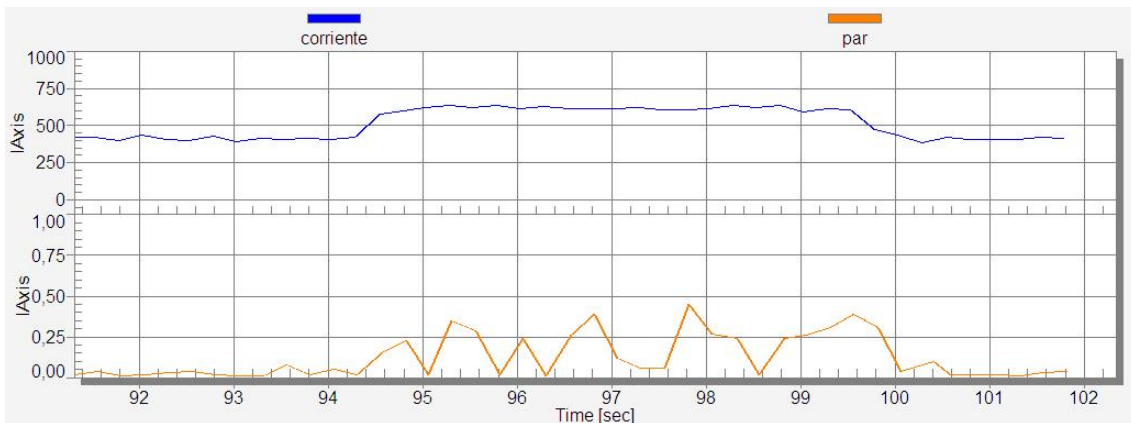


Figura 4.9: Pruebas realizadas con  $k_{pc} = 0.05$  y  $k_{ic} = 1.50$

El realizar un nuevo cambio en las constantes en el algoritmo del control de la corriente, reduciendo la constante de proporcional de 0.05 a 0.01 y aumentando la constante integral de 1.50 a 2.50 permite que la medición del par aumente, pero provoca que una vez alcanzado el par nominal, el motor tienda a detenerse debido al control de fallas del microcontrolador, esto se observa en la figura 4.10, donde el mayor par medido es de aproximadamente 2.0 Nm, con un error del 33% a una velocidad nominal de 1000 rpm,

y con un rango de corriente de 0.075 a 0.2 A, nuevamente se comprueba que el cambiar las constantes altera drásticamente el comportamiento del sistema.

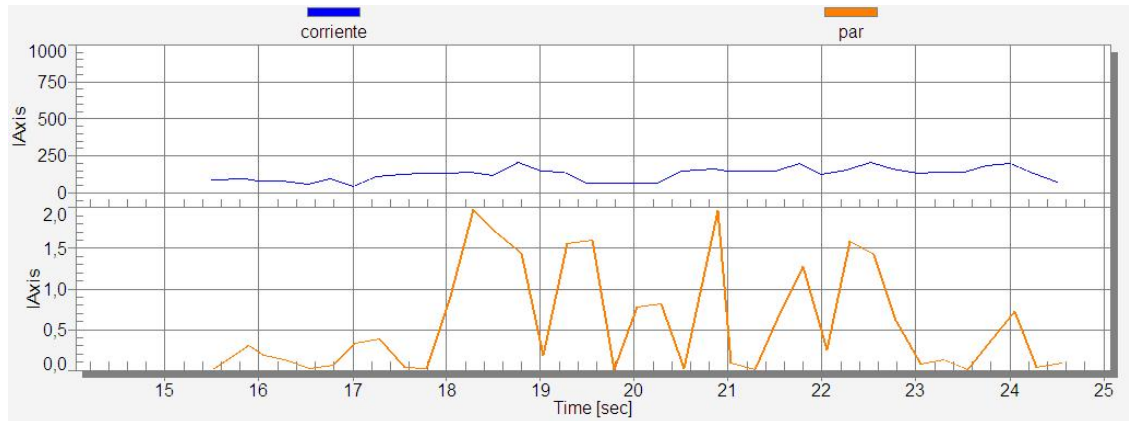


Figura 4.10: Pruebas realizadas con  $k_{pc} = 0.01$  y  $k_{ic} = 2.50$

Las pruebas muestran que el control de corriente trabaja correctamente cuando las constantes integral y proporcional tienen los valores mostrados en la figura 4.9, pues con estos valores la medición del par no pasa de 1.0 y la corriente muestra cambios de 0.05 A, en un rango de 0.25 a 0.75 A, esto también fue observado al cambiar las constantes del PID de velocidad, en donde el sistema demostró tener un comportamiento similar para los mismos valores en las constantes del control de corriente, como se puede ver en las figuras 4.11 y 4.12.

La gráfica de la figura 4.11 es conseguida a partir de las constantes  $k_p = 0.80$ ,  $k_i = 5.00$ , y  $k_d = 0.015$  y manejando las constantes  $k_{pc} = 0.05$  y  $k_{ic} = 0.15$ . Con estas constantes, el sistema mostró responder bien en la parte de velocidad permaneciendo el desfase entre las señales de entrada y de salida en la parte de velocidad. En cuanto al torque este se presentó en un rango de 0 a 0.650 Nm, con un error de 6.4% en un rango de corriente de 0.4 a 0.75 A.

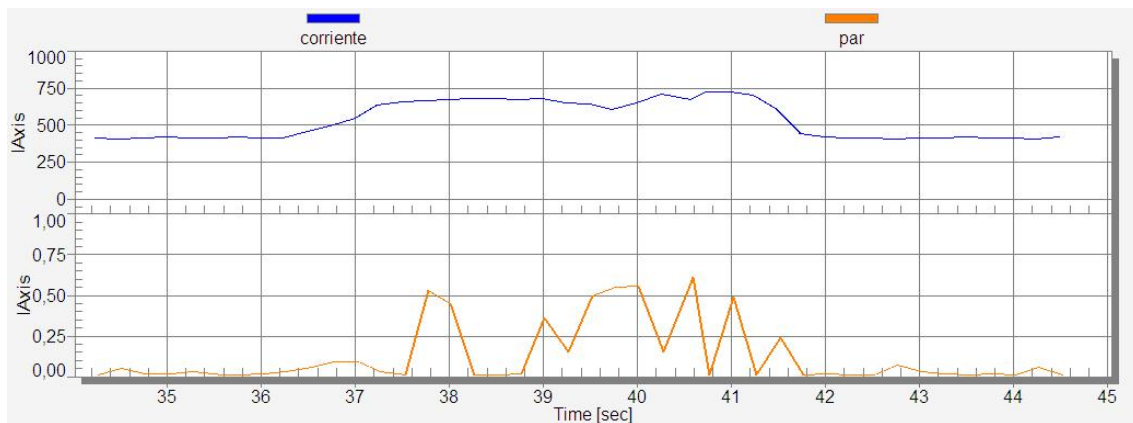


Figura 4.11: Gráfica de torque y corriente, con cambios en las constantes del PID

La figura 4.12, presenta la respuesta del control de corriente utilizando los mismos valores de las constantes utilizados en las últimas dos gráficas, pero con un cambio en las constantes de velocidad, con  $k_p=1.20$ ,  $k_i=3.00$ , y  $k_d=0.04$ , donde el controlador para esta ultima variable, presenta una buena respuesta de la salida con respecto a la entrada. El torque medido esta de 0 a 0.650 Nm y la corriente se encuentra en un rango de valores de 0.6 a 0.49 A.

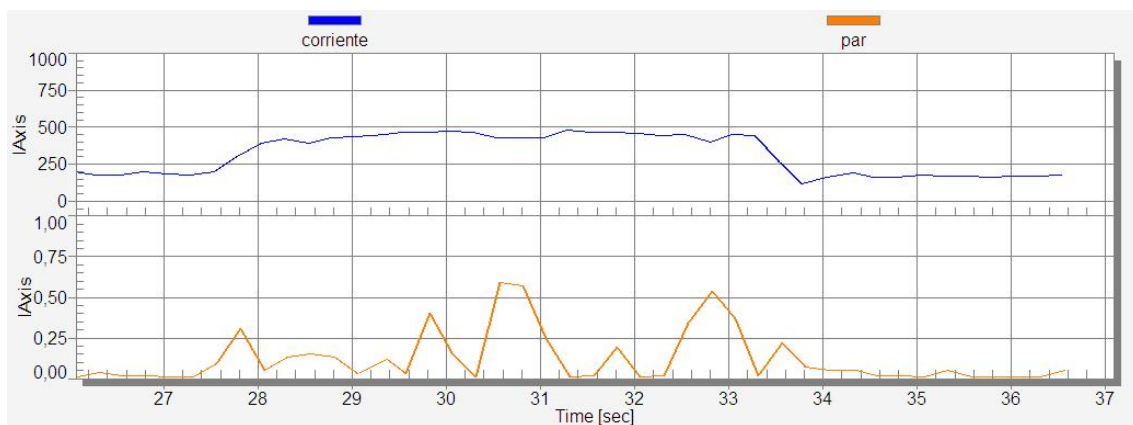


Figura 4.12: Gráfica de torque y corriente sin cambios en las constantes integral y proporcional del controlador de corriente

De las figuras 4.10, 4.11 y 4.12 la medición del par se encuentra en un rango de valores de 0 a 0.650 Nm, donde este último fue el mayor valor registrado por software, el cual pertenece a la gráfica mostrada en la figura 4.12, donde el control de velocidad mostró tener un error en estado estacionario de 0.32 % incluso a distintas velocidades, aunado a esto el error en el controlador de corriente fue 3 %. Debido a lo expuesto anteriormente la figura 4.12 expone el mejor control para la corriente. La figura 4.13 presenta la respuesta para la velocidad con este lazo de control en cascada.

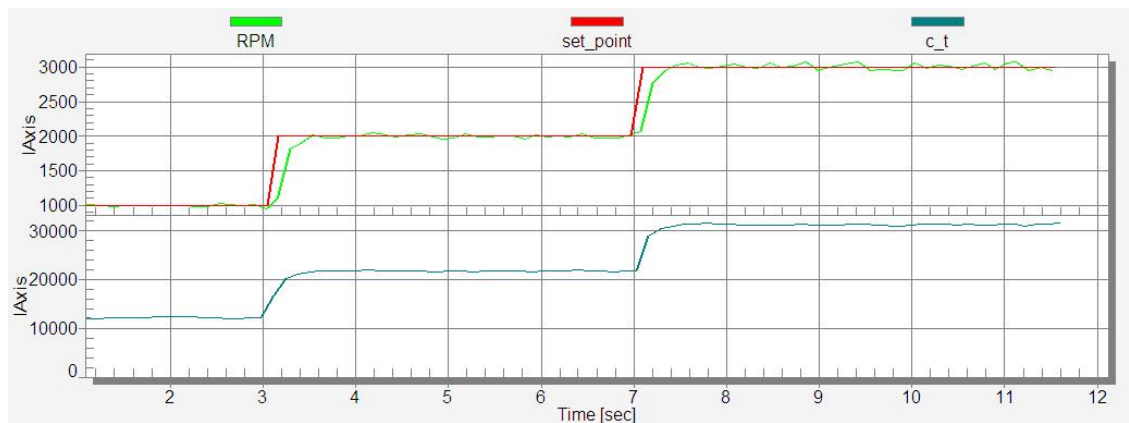


Figura 4.13: Control en cascada donde las constantes del PID con  $k_p=1.20$ ,  $k_i=3.00$  y  $k_d=0.04$

## 4.4. Sumario

El presente capítulo expuso las distintas pruebas realizadas, partiendo de la generación de las 6 modulaciones de ancho de pulso donde la figura 4.1, muestra a partir del uso de una tarjeta de adquisición de datos y el programa llamado MATLAB, cada uno de los pasos que se generaron. También se muestran las pruebas realizadas para el control de velocidad donde a partir de un ajuste en las constantes se identificaron bajo que parámetros el control de velocidad trabaja mejor, muestra de ello se puede observar en la figura 4.5, así pues finalmente se realizaron las pruebas al control en cascada, el cual se genero al agregar un algoritmo para la corriente en serie al controlador de velocidad y se ajustaron las constantes para observar la mejor respuesta del sistema (figura 4.12). Partiendo de lo anterior el apartado siguiente trata sobre las conclusiones a las que se llega a partir de los resultados obtenidos en esta parte.

# Conclusiones y Recomendaciones

Este apartado se encuentra dividido en 2 secciones, la primera parte contiene las conclusiones realizadas, y la tercera las sugerencias para trabajos futuros.

## 4.5. Conclusiones

- La información recabada en este trabajo permite establecer las bases necesarias para el control de motores síncronos de corriente directa, partiendo desde el principio básico, como la estructura del motor, con el único fin de poder establecer el tipo de técnica de control a emplear, de esta manera, con la investigación que se llevo a cabo se concluyo que debido a que el motor constaba de dos pares de polos, la técnica de los 6 pasos, utilizando 6 PWM era la mas apta para ser aplicada al motor sin escobillas.
- La técnica de los seis pasos consiste en pasar 6 PWM's a través de un inversor trifásico de corriente directa que esta constituido por 6 transistores tipo MOSFET, que permite utilizar 2 PWM's para cada una de las bobinas del motor y de esta manera establecer los seis pasos, este método permite que el motor mantenga una velocidad con un bajo ruido (20 a 26 kHz), además que es fácil de implementar. Con el ciclo de trabajo establecido por los PWM el motor mostró mantener una buena velocidad nominal, pero no controlada, es por esta razón que se recurrió a un tipo de algoritmo de control.
- Con la necesidad de mantener la velocidad a un valor deseado, se programó un control PID que permite que el motor mantenga una velocidad fija aun bajo ciertas perturbaciones, la pruebas realizadas mostraron que a diferentes constantes establecidas dentro del algoritmo de control, la velocidad se mantenía, salvo por el hecho de que en algunos casos el sistema mostró responder mas lento en comparación con

otros, esto se observó principalmente cuando las constantes eran pequeñas ( $k_p=0.8$  y  $k_d=0.03$ ) con lo cual el sistema al cambiar el punto de ajuste, la respuesta a la salida del sistema tardaba en llegar al valor impuesto a la entrada, finalmente se optó por considerar constantes grandes ( $k_p=2.0$  y  $k_d=0.05$ ), pues lo que se buscaba era tener velocidad de respuesta por parte del algoritmo de control.

- El tener un control indirecto sobre la corriente implica que debe existir un control sobre el flujo de corriente que circula a través de las bobinas del estator, en donde existen dos maneras de obtener este tipo de corriente, la primera es a través de sensores que estuvieran conectados al estator y la segunda haciendo uso de la corriente de bus, este tipo de corriente realmente no tiende a mostrar muchas variaciones, lo cual es una de las razones por la cuales el lazo de control para la corriente no mostró un amplio rango de cambio, las pruebas realizadas mostraron que manejando una constante integral pequeña (0.01) la lectura de la corriente tendía a ser errónea debido al amplio margen de error que existía, con lo que el integrador no podía regular el lazo interno del control del sistema en su conjunto. Con una constante integral grande (1.5), la corriente se presentaba en rangos no mayores a 2000 mA, lo que significa que el torque del motor era menor a 0.9 N-m.
- La limitante mas evidente en la lectura del torque se muestra en cada una de las gráficas que se presentan en el capítulo cuatro en la sección correspondiente al control en cascada que presenta un gráfica del par irregular debida posiblemente a dos factores importantes, el primero a que el código del algoritmo de control fue realizado en Procesador Experto, lo que implica utilizar mas espacio en memoria lo que deriva directamente en velocidad de procesamiento por parte del DSC, segundo a que el convertidor analógico-digital utilizado carece de un amplio margen de resolución lo que repercute directamente en la lectura de la corriente de bus.

## 4.6. Sugerencias de Trabajos a Futuro

- Implementar una técnica de PWM bipolar, utilizando el mismo DSC empleado en este trabajo.
- Buscar apoyo en una técnica de sintonización para sistemas digitales tal como la de Ziegler-Nichols.

- Uso de la misma etapa de control y de potencia con un motor de tres fases de corriente directa con un mayor número de pares de polos.
- Ampliar la presente investigación a un control de posición con algoritmos de tipo P, PI y PID.
- Depuración del algoritmo de control, de manera que permita tener una mayor velocidad de procesamiento, así como el empleo de componentes ligados a la conversión analógico-digital.
- Realizar pruebas al motor bajo distintas cargas, obteniendo un análisis de comportamiento, que permita posteriormente poder emplear un control difuso o neuro difuso.
- Ampliar la investigación a técnicas empleadas a motores de imanes permanentes con distribución senoidal.
- Implementar el mismo programa a una etapa de potencia on capacidad para un motor de mayores dimensiones.
- Comparación entre el aprovechamiento del flujo de corriente entre un motor BLDC y un motor PMSM (del inglés, Permanent Magnet Synchronous Motor).





# Referencias

- AKIN, BILIAL (2000). Trapezoidal Control of BLDC Motors Using Hall Effect Sensors. *Informe técnico*, Texas Instruments.
- ATALLAH, KAIS (2003). Torque-Ripple Minimization in Modular Permanent-Magnet Brushless Machines. *IEEE Transactions on Industry Applications*, **39**, pp. 1689–1695.
- AWADALLAH, MOHAMED A. (2009). Adaptive Deadbeat Controllers for Brushless DC Drives Using PSO and ANFIS Techniques. *ELECTRICAL ENGENIERING*, **60**, pp. 3–11.
- BEN-BRAHIM, LAZHAR (2002). Repetitive Control Based PWM Inverter Dead-Time Compensation for AC Servo Drive. *Engineering Journal of the University of Qatar*, **15**, pp. 93–109.
- CHEN, LIHUA (2007). Elimination of Dead-Time in PWM Controlled Inverters. *Informe técnico*, Universidad del Estado de Michigan.  
Recuperado de: [http://www.ee.bgu.ac.il/~pedesign/Graduate\\_problem\\_papers/papers\\_2007/PWM\\_Deadtime.pdf](http://www.ee.bgu.ac.il/~pedesign/Graduate_problem_papers/papers_2007/PWM_Deadtime.pdf).
- CHU.L.C (2001). Torque Control of Brushless DC Motor Applied to Electric Vehicles. *Informe técnico*, Universidad de Tecnología de Taiwan, Departamento de Ingeniería Eléctrica. Recuperado de:  
<http://ir.lib.stut.edu.tw/bitstream/987654321/4184/2/Torque%20control%20of%20brushless%20DC%20motors%20applied%20to%20electric%20vehicles.pdf>.
- COLTON, SHANE W. (2008). *Design and Prototyping Methods for Brushless Motors and Motor Control*. Tesina o Proyecto, Instituto Tecnológico de Massachusetts.
- CONDIT, RESTON (2007). Sensorless BLCD Control with Back-EMF Filtering. *Informe técnico*, Microchip Technology Inc..

- CÉSAR, JUAN JOSÉ MUÑOZ (2005). *Autosintonización por Búsqueda Tabú del Control Vectorial Difuso de Velocidad para un Motor de Inducción*. Tesina o Proyecto, Instituto Politécnico Nacional, Escuela Superior de Ingeniería Mecánica y Eléctrica.
- DALIB, SARIATI BINTI (2007). *The Simulation of the Direct Torque Control of Permanent Magnet Synchronous Motor*. Tesina o Proyecto, Universidad Tecnológica de Malasia.
- DILMI, SABRI (2006). Nonlinear Torque Control of the Induction Motor in Hybrid Electric Vehicle Applications. En: *American Control Conference*, .
- DIXON, JUAN W. (2006). Simplified Sensorless Control for BLDC Motor, Using DSP Technology. *Informe técnico*, Departamento de Ingeniería Eléctrica de la Universidad Pontificia de Chile. Recuperado de: <http://web.ing.puc.cl/power/paperspdf/dixon/53a.pdf>.
- ELECTRONICS, NEC (2006). *3-Phase Brushless DC Motor Control 120-Degree Trapezoidal Drive with Hall Sensors for MC-LVKIT-714 Motor Control Evaluation System*. NEC Electronics Corporation.
- ELEVICH, LEONARD N. (2005). 3-Phase BLDC Motor Control with Hall Sensors Using 56800/E Digital Signal Controllers. *Informe técnico*, Freescale Semiconductor.
- ELLIOTT, CHARLIE (2004). Using the dsPIC30F for Sensorless BLDC Control. *Informe técnico AN901*, Microchip Technology Inc..
- ELLIS, GEORGE (2003). *Control System Design Guide*. George H..
- FICAPAL, ALBERT TORNÉ (2003). Algoritmos de Modulación Para Motores Brushless. *Informe técnico*, Ingenia-CAT, Departamento de Control de Movimiento. Recuperado de: <http://www.ingenia-cat.com/reference/learn/TEC.PAP.5240009519.pdf>.
- FREESCALE (2009a). 3-Phase Sensorless BLDC Motor Control Using MC9S08MP16. *Informe técnico*, Freescale Semiconductor.
- FREESCALE (2009b). MC9S08MP16 Reference Manual. *Informe técnico*, Freescale Semiconductor.
- FROGAL, PETRI (2009). 3-Phase BLDC/PMSM Low-Voltage Motor Control Drive. *Informe técnico*, Freescale Semiconductor.

- GAMAZO, JOSÉ CARLOS (2010). Position and Speed Control of a Brushless DC Motors Using Sensorless Techniques and Application Trends. *Sensors*, **10**, pp. 6901–6947.
- GRASBLUM, PAVEL (2008). Using FlexTimer in ACIM/PMSM Motor Control Applications. *Informe técnico*, Freescale Semiconductor.
- HATTORI, SATOMI (2002). Suppression Control Method of Torque Vibration for Brushless DC Motor Utilizing Repetitive Control with Fourier Transform. *Trans. of Society of Instrument and Control Engineers*, **2**, pp. 42–53.
- HOBETLER, THOMAS G. (1992). Direct Torque Control of Induction Machines Using Space Vector Modulation. *IEEE Transactions on Industry Applications*, **28**, pp. 1045–1053.
- HWANG, SEON-HWAN (2009). Dead-Time Compensation Method for Voltage-Fer PWM Inverter. *IEEE Transactions on Energy Conversion*, **2**, pp. 1–10.
- IM, WON-SANG (2010). Torque Maximization Control of 3-Phase BLCD Motors in the High Speed Region. *Power Electronics*, **10**, pp. 717–723.
- JANI, YASHVANT (1999). Implementig Embedded Speed Control for Brushless DC Motors. *Informe técnico*, Renesas Technology.
- JEONG, SEUNG-GI (1991). The Analysis and Compensation of Dead-Time Effects in PWM Inverters. *IEEE Transactions on Industrial Electronics*, **38**, pp. 108–114.
- KALIAPPAN, E. (2010). Direct Torque Control of PMSLDC Motor using Hybrid (GA and Fuzzy logic) Controller. *Advances in Information Technology*, **1**, pp. 163–167.
- KARTHIKEYAN, J. (2010). Simulation and Implementation of Current Control of BLDC Motor Based on a Common DC Signal. *International Journal of Engineering Science and Tachnology*, **2**, pp. 1632–1639.
- KETTLE, PAUL (1998). Sensorless Control of a Brushless DC Motor Using an Extended Kalman Estimator. *Informe técnico*, Analog Devices, Motion Control Group.
- KING, KEVIN (2008). MCU Performance for Various Control Algorithms of BLDC Motors. *Informe técnico*, Renesas Technology.

- KUMAR, B. MAHESH (2009). Sensorless Speed Control of Brushless DC Motor with Fuzzy Based Estimation. *Electrical and Computer Engineering*, **8**, pp. 119–125.
- LAWLER, J.S. (2004). Extending the Constant Power Speed Range of the Brushless DC Motor Through Dual-Mode Inverter Control. *IEEE TRANSACTIONS ON POWER ELECTRONICS*, **19**, pp. 783–793.
- LEE, SHIYOUNG (2004). A Comparison Study of the Commutation Methods for the Three-Phase Permanent Magnet Brushless DC Motor. *Informe técnico*, Universidad de Pensilvania. Recuperado de: <http://www.magnelab.com/uploads/4c51d9ba6fe5a.pdf>.
- LEGGATE, DAVID (1997). Pulse Based Dead Time Compensator for PWM Voltage Inverters. *Informe técnico*, Allen-Bradley. Recuperado de: [www.ieeexplore.ieee.org](http://www.ieeexplore.ieee.org).
- LIN, JONG-LICK (2002). A New Approach of Dead-Time Compensation for PWM Voltage Inverters. *IEEE Transactions on Circuits and Systems*, **49**, pp. 476–483.
- LÓPEZ, FABIÁN JIMÉNEZ (2010). Current Control Methods for Shunt Power Active Filters to Reduce Harmonic Content in Electrical Systems. *Informe técnico*, Facultad de Ingeniería Electrónica, Universidad de Santo Tomas Tunja. Recuperado de: <http://www.eatis.org/eatis2010/portal/paper/memoria/html/files/11.pdf>.
- MANI, MARC VILA (2003). Alineación Inicial de un Motor PMSM con Encoder Incremental Mediante Sensores de Efecto Hall. *Informe técnico*, Ingenia-CAT. Recuperado de: <http://www.ingenia-cat.com/reference/learn/TEC.PAP.5379191030.pdf>.
- MARGARITA, ANDREY (2004). Brushless DC Motor Control. *Nota de Aplicación AN2227*, CYPRESS MYCROSYSTEMS.
- ÖMER AYDOGDU (2010). An Effective Real Coded GA Based Fuzzy Controller for Speed Control of a BLDC Motor without Speed Sensor. *Elec Eng & Comp Sci*, **19**, pp. 1–18.
- MEVEY, JAMES ROBERT (2006). *Sensorless Field Oriented Control of Brushless Permanent Magnet Synchronous Motors*. Tesina o Proyecto, Universidad de Kansas.
- MICHAEL, PRAWIN ANGEL (2010). Implementation of SVPWM in FPGA through MATLAB/SIMULNK Environment. *Engineering Science and Technology*, **2**, pp. 4585–4594.

- MINCIUNESCU, PETRE (1998). Sensorless Control of Brushless DC Motor Using Extended Kalman Estimator and Back-EMF Integration Algorithm: A Comparation. *Informe técnico*, Analog Devices, Inc..
- OHM, DAL Y. (1999). About Commutation and Current Control Methods for Brushless Motors. *Informe técnico*, Drivetech Research.
- OZKUT, SALIH BARIS (2008). Sensorless Direct Torque and Indirect Flux Control of Brushless DC Motor with Non-sinusoidal Back-EMF. *IEEE*, **9**, pp. 1373–1378.
- PROKOP, LIBOR (2005). 3-Phase BLDC Motor Control with Sensorless Back-EMF ADC Zero Crossing Detection Using 56F80x. *Nota de Aplicación AN1913*, Freescale Semiconductor.
- RAO, G. MADHUSUDHANA (2009). Speed Control of BLDC Motor with Common Current. *International Journal of Recent Trends in Engineering*, **2**, pp. 182–187.
- RASHID, MUHAMMAD H. (1992). *Electronica de Potencia*. Alan Apt, 2ª edición.
- SALGUES, CHRISTOPHE X. (2008). *Torque Control of a PMSM Brushless DC Machine for Hybrid Electric Vehicle*. Tesina o Proyecto, Universidad de Akron.
- SENTURK, OSMAN S. (2009). A Single Leg Switched PWM Method for Three-Phase H-Bridge Voltage Source Converters. *Informe técnico*, Universidad de Alborg, Dinamarca. Recuperado de: <http://vbn.aau.dk/files/18943553/Publication>.
- SHANNON, THADDEUS T. (2001). Adaptive Critic Based Design of a Fuzzy Motor Speed Controller. *Informe técnico*, Northwest Computational Intelligence Laboratory and Systems Science. Recuperado de: <http://www.sysc.pdx.edu/download/papers/ISIC2001.pdf>.
- SINGH, BHIM (2003). Gain Scheduling Control of Permanent Magnet Brushless DC Motor. *Informe técnico*, Departamento de Ingeniería Eléctrica, IIT Delhi, India. Recuperado de: <http://www.docstoc.com/docs/44364414/Gain-Scheduling-Control-of-Permanent-Magnet-Brushless-dc-Motor>.
- SOTER, STEFAN (2005). Low Cost Vector Control for Permanent-Magnet-Synchromotor with only one DC-Link Current Sensor for Pumping Application. *IEMDC*, **3**, pp. 1–4.

- SOZER, YILMAZ (1998). Low-Cost Sensorless Control of Brushless DC Motors with Improved Speed Range. *IEEE*, **6**, pp. 86–92.
- STASZKO, PETR (2010). MC9S08MP16 High Speed BLDC Sensorless Drive. *Informe técnico*, Freescale Czech System Laboratories Roznov Pod Radhostem.
- SU, GUI-JIA (2001). Low-Cost Sensorless Control of Brushless DC Motors with Improved Speed Range. *Informe técnico*, Oak Ridge National Laboratory. Recuperado de: <http://www.ornl.gov/webworks/cppr/y2001/pres/112957.pdf>.
- TAGHIRAD, H.D (2001). H-Based Robust Torque Control of Harmonic Drive Systems. *Journal of Dynamic Systems, Measurement and Control*, **123**, pp. 338–345.
- TEXAS (1998). Field Orientated Control of 3-Phase AC-Motors. *Informe técnico BPRA073*, Texas Instruments.
- TOLRÁ, R. J. (2003). Técnicas de Control para Motores Brushless. *Informe técnico*, Ingenia CAT.  
Recuperado de: <http://www.ingenia-cat.com/reference/learn/TEC.PAP.0422232450.pdf>.
- URASAKI, NAOMITSU (2006). An Adaptive Dead-Time Compensation Strategy for a Permanent Magnet Synchronous Motor Drive Using Neural Network. *Power Electronics*, **6**, pp. 279–289.
- VAS, PETER (1998). *Sensorless Vector and Direct Torque Control*. Peter Hammond.
- VIRAMONTES, EDUARDO (2010). BLDC Motor Control with Hall Effect Sensors Using the 9S08MP. *Informe técnico*, Freescale Semiconductor.
- WIBERG, JOHAN (2003). *Controlling a Brushless DC Motor in a Shift-by-Wire System*. Tesina o Proyecto, Universidad de Linköpings.
- XIACHANGLIANG (2000). Speed Control of Brushless DC Motor Using Genetic Algorithm based Fuzzy Controller. *Informe técnico*, Escuela de Ingeniería Eléctrica y automatización, Universidad de Tianjin. Recuperado de: <http://www.paper.edu.cn>.
- XU, HUANGSHENG (2008). Understanding Sensorless Vector Control for Brushless DC Motors. *Informe técnico*, Renesas Technology.

- YAN, GOKHAN (2010). Brushless Direct Current (BLDC) Machine Bipolar-Unipolar Driving Circuit Educational Study. *ELSEVIER*, **2**, pp. 2694–2699.
- YEDAMALE, PADMAJARA (2003). Brushless DC (BLDC) Motor Fundamentals. *Informe técnico*, Microchip Technology Inc..





# Anexo A

## Esquema de la Etapa de Potencia

La unidad de control de motores BLDC y PMSM de baja tensión de 12-24 V de CD a 4 A, es un tablero principal, que junto a una tarjeta secundaria, es una unidad de desarrollo para aplicaciones de control de motores de tres fases.

Las tarjetas secundarias, disponibles con un microcontrolador, a través de una plataforma de software de desarrollo y señales de retroalimentación permiten una variedad de algoritmos para el control de motores de tres fases. La figura A.1 muestra una ilustración de la arquitectura del sistema.

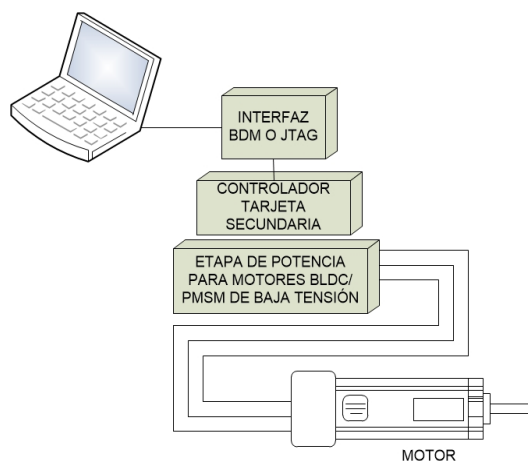


Figura A.1: Configuración del Sistema.

Las características de la tarjeta son:

- Alimentación de entrada de 12-24 V de CD, con ampliación a 50 V.
- Corriente de salida de 4A.
- Circuito de protección para alimentación de polaridad inversa.
- Inversor de tres fases (6 MOSFET's).
- MOSFET para 3 fases con sobrecarga y protección de mínima tensión.
- Sensado de corriente de bus y de derivación en las tres fases.
- Circuito de sensado de fuerza contraelectromotriz en las tres fases.
- Dos conectores para tarjeta secundaria.
- Capa física CAN.
- Interfaz USB.
- LED indicador, LED de encendido, 6 diodos LED para PWMs, diodos LED indicadores para actividad SCI.
- Interruptores de un tiro dos polos.
- Botón pulsador para reseteo.

## A.1. Descripción Operacional

La unidad de control de motores de 3 fases de baja tensión, consiste en una etapa de potencia que opera con voltajes de entrada de corriente directa en el rango de 12-24V, a 4 A.

Junto con la tarjeta secundaria, proporcionan una plataforma de desarrollo de software que permite la escritura de los algoritmos, y la construcción de cualquier hardware. Esto soporta una amplia variedad de algoritmos para motores PMSM y motores sin escobillas (BLDC).

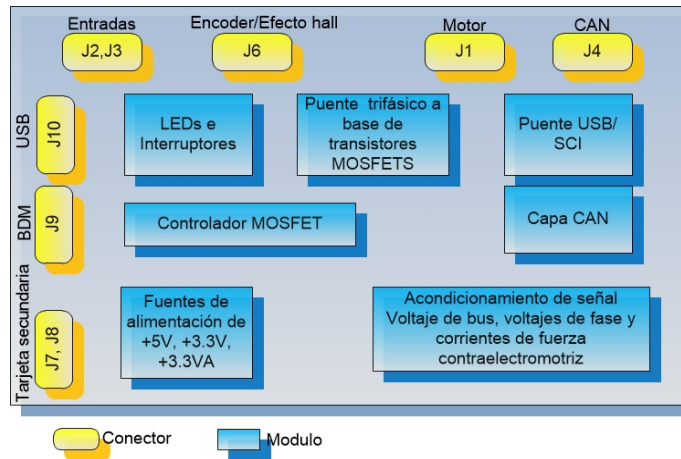


Figura A.2: Diagrama a Bloques

La figura A.2 muestra el diagrama a bloques. La tarjeta secundaria es conectada a través de 2 conectores de 20 pines J7 y J8. La figura A.3 muestra la asignación de los conectores para la tarjeta secundaria. Las conexiones de alimentación al motor se hacen en el conector de la salida J1. La fase A, fase B y fase C son llamadas A, B y C. La energía requerida se obtiene de una sola fuente de 12V y 50 V de corriente directa. Cualquier entrada es suministrada a través de los conectores J2 y J3.

Las siguientes tarjetas secundarias son validas con estos controladores:

- MC56F8013/23
- MC9S08AC60
- MCF51AC256
- MC9S08MP16
- MC56F8006



Figura A.3: Distribución de los conectores J7 y J8.

## A.2. Consideraciones de Diseño

El controlador de motores de tres fases BLDC/PMSM de baja tensión es diseñado para el desarrollo de software. Además de hardware necesario para mover un motor, se proporcionan una variedad de señales de retroalimentación que facilitan el desarrollo del algoritmo de control.

### A.2.1. Puente de Tres Fases

La etapa de salida se configura como un puente trifásico con transistores de salida MOSFET. Esto se simplifica considerablemente integrando un controlador de compuerta que presenta una sobrecarga, baja tensión y otras características de seguridad. La figura A.4 muestra el esquemático para una fase. En la entrada, la resistencia R99 configurada como *pull-down* coloca un estado lógico bajo en la ausencia de una señal para el transistor inferior. Permitir una entrada *pull-down* es importante, debido a que el transistor de potencia podría desactivarse en el caso de una interrupción en la conexión o en ausencia de energía en la tarjeta secundaria. El control de las compuertas se alimenta a 3V. El circuito integrado MC33927 de Freescale, alimenta las compuertas. El MC33927 también ofrece bajo voltaje y sobrecorriente. El bajo voltaje de umbral es de 8V. El MC33927 cuenta con una inserción de tiempo muerto, que puede ser configurado usando SPI. El valor típico para el tiempo muerto es de  $15\mu\text{s}$ . Una importante consideración de diseño en el controlador del motor es la selección de la impedancia en la compuerta para las salidas de los transistores. En la figura A.4, las resistencias R69, R70, el diodo D11, la caída de corriente nominal del MC33927 determinan la impedancia de la compuerta para la mitad inferior del puente de transistores. Una red similar se utiliza en la mitad superior del puente. Estas redes establecen el valor de la impedancia de entrada en aproximadamente  $100\Omega$  y una caída de corriente en el control de compuerta de aproximadamente 100 mA.

Estos valores producen tiempos de transición de aproximadamente 285 ns.

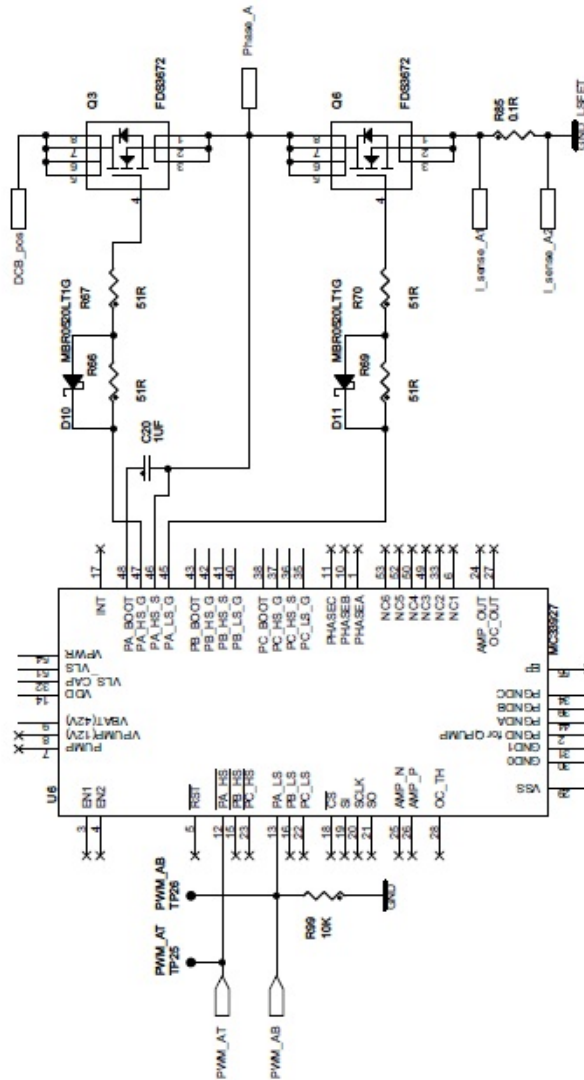


Figura A.4: Fase de Salida

Tiempos de transición de esta magnitud representan un gran compromiso de seguridad entre la disipación de energía y la generación de ruido. Generalmente, tiempos de transición más amplios que 250 ns tienden a consumir mucha energía en rangos no audibles de PWM; tiempos de transición abajo de 50 ns crean cambios en la corriente ( $di/dt$ ) tan grandes que es difícil lograr un correcto funcionamiento. El controlador de motores BLDC está diseñado con tiempos de conmutación en este amplio rango para minimizar el ruido.

Un diodo en anti paralelo es también considerado en el diseño. Si los diodos anti paralelos de un motor fuera de línea son dañados, el resultado en el cambio de corriente puede causar problemas en el manejo del ruido, que serán difíciles de resolver. En general, la caída a cero de  $di/dt$  puede ser aproximadamente igual al  $di/dt$  aplicado en la desactivación de los diodos anti paralelos. Los MOSFET's FDS3672 usados en el diseño son admisibles para la recuperación inversa.

### A.2.2. Corriente de Retroalimentación y Corriente de Bus

La figura A.5 muestra el circuito que permite la retroalimentación de señales proporcionales al voltaje de bus y a la corriente de bus. La tensión de bus se reduce a través de un divisor de tensión formado por R23, R27 y R104. Los valores se eligen de tal manera que una tensión de 36.6V de voltaje de bus corresponde a 3.3V en  $V_{sense\_DCB}$ . El  $V_{sense\_DCB}$  se incrementa cada 91 mV por volt del voltaje de bus, y se conecta a la tarjeta secundaria a través del conector J8 en el pin 9. Una salida adicional,  $V_{sense\_DCB/2}$ , proporciona una referencia en la detección de cruce por cero. El  $V_{sense\_DCB/2}$  aumenta 45.4 mV por volt del voltaje de bus, y es conectado a la tarjeta secundaria a través del conector J8 en el pin 10.

La corriente de bus es muestreada por la resistencia R88 en la figura A-9, y amplificada en los amplificadores operacionales del MC33927 (figura A.5). Este circuito proporciona una salida de voltaje adecuado para muestreo en las entradas analógico-digitales. El amplificador operacional del MC33927 es usado como amplificador diferencial para sensar la corriente de bus. Con  $R82 = R83$ ,  $R102 = R103$ , y  $R81 = R84$ , la ganancia es obtenida por:

$$A = \frac{R81}{(R82 + R102)} \quad (A.1)$$

La tensión de salida se desplaza por  $+1.65 V_{REF}$  para permitir cambios de corriente positiva y negativa. Una caída de tensión de  $\pm 400$  mV a través de la resistencia de sentido corresponde a un rango de medición de corriente de  $\pm 4A$ . La señal AMP\_OUT esta internamente conectada al comparador de sobrecorriente del MC33927, y proporciona una función de disparo de sobrecorriente. Además, el AMP\_OUT está conectado al conector J8 de la tarjeta secundaria en el pin 11 denominado  $I_{sense\_DCB}$ .

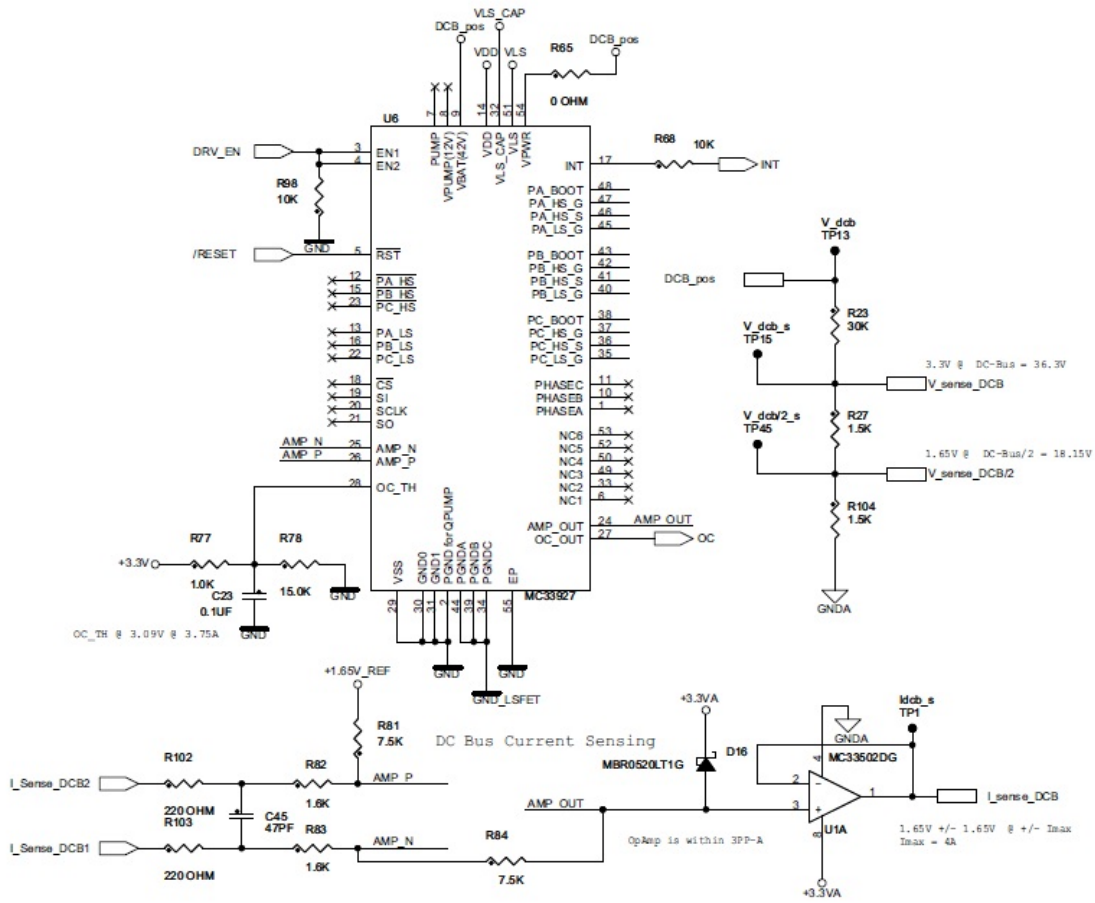


Figura A.5: Bus de retroalimentación

### A.2.3. Sobrecorriente, Bajo Voltaje, y otras Funciones de Seguridad

El MC33927 proporciona funciones de sobrecorriente y bajo voltaje (figura A.5). La retroalimentación de la corriente del bus se filtra para eliminar los picos, y esta señal es alimentada en el comparador de corriente del MC33927. Por lo tanto cuando la corriente de bus excede los 3.75 A, los seis transistores son desactivados. Una vez que el estado de fallo ocurra, las seis compuertas son desactivadas, hasta que el estado de fallo sea borrado por el reset, o por el interruptor de apagado de la tarjeta.

La función de bajo voltaje es implementada internamente. La alimentación de voltaje del MC33927 es detectada internamente. Si este voltaje es menor que 8V, el circuito desactivado es evaluado, y una interrupción es generada.

Las funciones de seguridad del MC33927 mantienen al dispositivo funcionando correctamente y dentro de los límites de seguridad. Este limita su corriente, sin embargo, no garantiza necesariamente que la tarjeta este operando dentro de los límites térmicos. El MC33927 tiene una función de advertencia térmica. Si la temperatura se eleva por encima de 170°C en uno de los tres detectores, entonces se genera una interrupción.

El MC33927 cuenta con otras funciones de seguridad como la detección de la desaturación. Error de fase, generación del error, error de escritura después del bloque, y reseteo. Todas estas características pueden ser configuradas a través de SPI para disparo de las interrupciones.

#### A.2.4. Detección de Corriente de Fase

Las resistencias de muestreo proporcionan información de la corriente de fase para las tres fases. Debido a que estas resistencias muestrean la corriente de fase inferior, estas no miden la corriente de fase directamente. Sin embargo, obteniendo las corrientes de fase para las tres fases, la corriente de fase puede ser calculada a partir de los valores de la corriente de fase. Los circuitos de medición para una fase se muestran en la figura ???. La corriente es registrada por la resistencia R85 y amplificada por el amplificador operacional U1B.

El circuito proporciona un voltaje de salida adecuado para el muestreo en las entradas A/D. El MC33052DG es usado como un amplificador diferencial. Con  $R2 = R7$ ,  $R3 = R5$  y  $R4 = R6$ , la ganancia está dada por:

$$A = \frac{R2}{(R3 + R4)} \quad (\text{A.2})$$



El voltaje de entrada cambia cada  $1.65 V_{REF}$  que permite desplazamientos de corriente negativos y positivos. Un voltaje de  $\pm 400$  mV pasa a través de la resistencia de derivación que corresponde a una corriente de medida de  $\pm 4$  A.

La ganancia de este amplificador operacional es 4.12 sin el  $+1.65$  V de *offset*, en otras palabras la salida de  $\pm 1.65$  V corresponde a  $\pm 4$  A. La salida es conectada a la tarjeta secundaria en el conector J8.

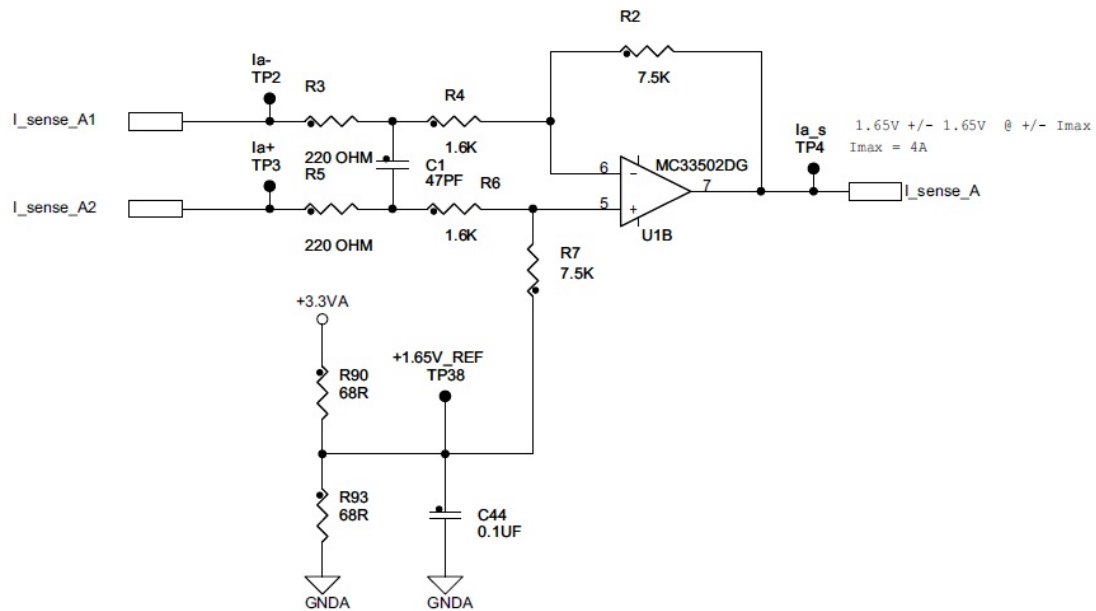


Figura A.6: Detección de Corriente de Fase

### A.2.5. Fuentes de Alimentación y Voltaje de Referencia

La etapa de control de motores de 3 fases BLDC/PMSM de baja tensión contiene distintos dispositivos que requieren varios niveles de voltaje como  $+5V$  o  $+3.3V$ .

#### Fuente de Alimentación de Entrada

El bus puede ser alimentado de 2 conectores de entrada, J2 y el conector de alimentación J3 (figura A.7). La fuente de alimentación debe de ser capaz de entregar al menos 4A. El bus de DC tiene protección de polaridad inversa. El MC33927 es alimentado directamente del bus de corriente directa. La fuente de  $+5V$  de energía se sirve de la tensión de bus.

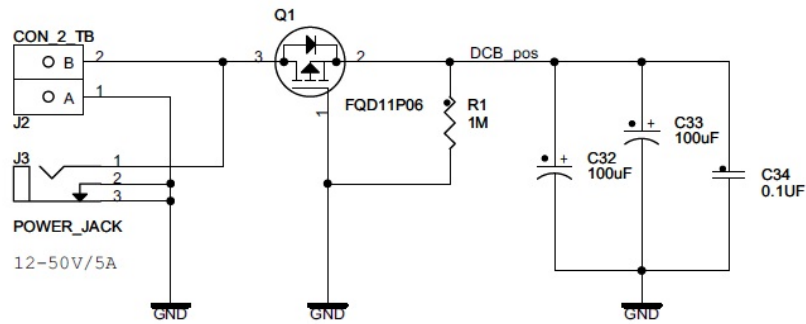


Figura A.7: Circuito de entrada del Bus de CD

### Fuente de Alimentación de 5V

El nivel de 5V se genera a través del regulador de cambio LM2594HVM (figura A.8), que genera este nivel del voltaje de bus. Este convertidor puede alimentar más de 500mA. Este nivel de voltaje lineal del MC33269D permite regular, codificar, e interconectar la capa física de la interfaz CAN. Si el convertidor LM2594HVM opera adecuadamente, el LED verde D19 enciende.

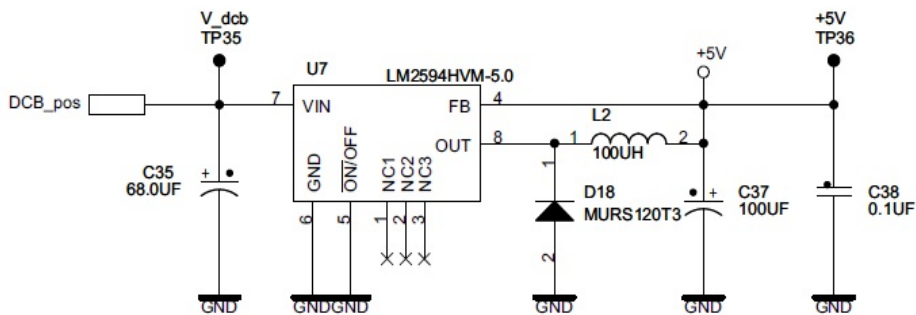


Figura A.8: Fuente de Alimentación de 5V

### Fuente de Alimentación de 3.3V

Un importante nivel de voltaje para esta tarjeta es 3.3V. Este voltaje es obtenido del MC33269D que es un regulador de voltaje lineal, que alimenta aproximadamente 800 mA (figura A.9). Este voltaje es utilizado para alimentar el inversor lógico y el controlador JM60. Este está conectado a la tarjeta secundaria a través del conector J7.

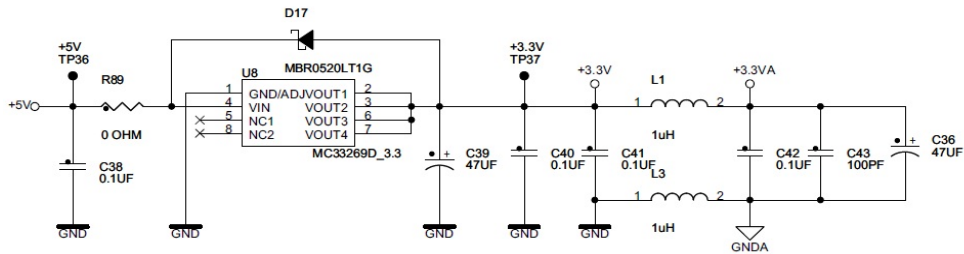


Figura A.9: Fuente de Alimentación de 3.3V

## Interfaz CAN

La tarjeta contiene una interfaz CAN. La principal parte de la interfaz esta en el controlador PCA82C250. El PCA82C250 es la interfaz entre el controlador de comunicación CAN y la capa física del bus. El dispositivo ofrece capacidad diferencial de transmitir y capacidad diferencial de recibir hacia el controlador CAN.

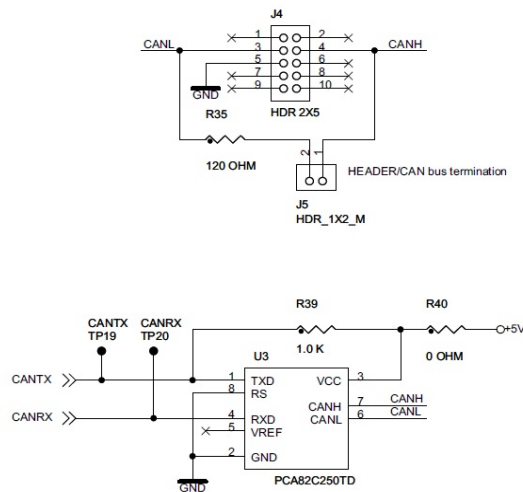


Figura A.10: Interfaz CAN

La interfaz CAN es compatible con ISO 11898, y permite un rango máximo de transferencia de datos de 1 Mbit/s. El receptor CAN está protegido contra cortocircuitos, protección contra perturbaciones en el bus, y protección térmica.

El receptor CAN está conectado vía CANTX y CANRX a la tarjeta secundaria a través del conector J7. Las señales de bus CANL y CANH son enviadas hacia el bus CAN por medio de J4. La figura A.10 muestra el circuito de la interfaz CAN.



# Anexo B

## Tarjeta Controladora MC56F8006

### B.1. Listado de Partes

Tabla B.1: Lista de Partes

Dispositivo	Cantidad	Descripción	Fabricante	Numero de Parte
C3, C2, C4, C6, C7	5	820pF/ 6.3V	Cualquiera	
C1	1	2200pF/6.3V	Cualquiera	
C8, C9, C10	3	100nF/6.3V	Cualquiera	
J1, J6	2	HDR 3×1	MOLEX	09-65-203B
J2	1	HDR 2×7	TYCO ELECTRONICS	4-103322-2
J3, J4	2	87022-610	FCI	87022-610LF
J5	1	HDR 2×6	TYCO ELECTRONICS	
R1-R4, R6, R9	6	100Ω 1/8W	Cualquiera	
R5	1	1kΩ 1/8W	Cualquiera	
R7, R8	2	47kΩ 1/8W	Cualquiera	
R10	1	33kΩ 1/8W	Cualquiera	
R11	1	18kΩ 1/8W	Cualquiera	
U1	1	DSC/LQFP-32	FREESCALE	MC56F8006VLC
U2	1	2 Compuertas-OR	ON SEMICONDUCTOR	MC74VHC1G32

## B.2. Esquemático de la Tarjeta Controladora MC56F8006 para el Control de Motores BLDC/PMSM

### B.3. Esquemático

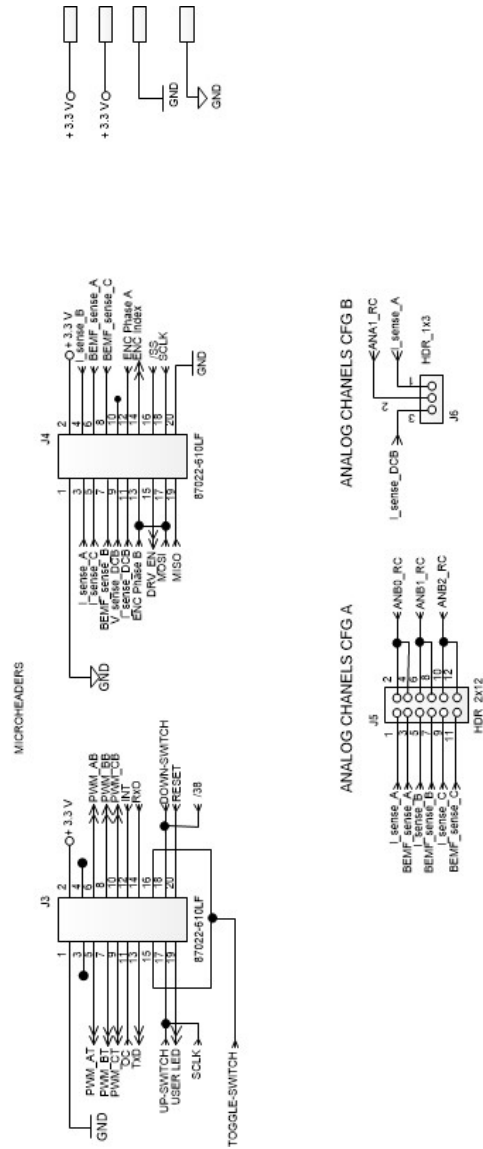


Figura B.1: Conectores y Cabeceras (*headers*) de la Tarjeta Hija

#### B.4. Diseño de la Tarjeta Impresa (PCB) del Controlador MC56F8006 para el Control de Motores BLDC/PMSM

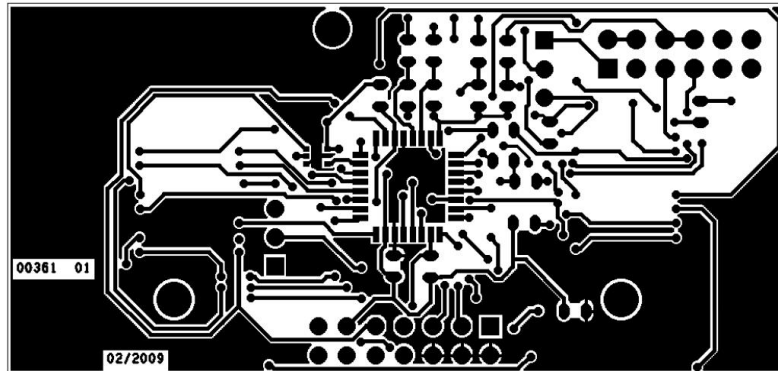


Figura B.2: Capa superior

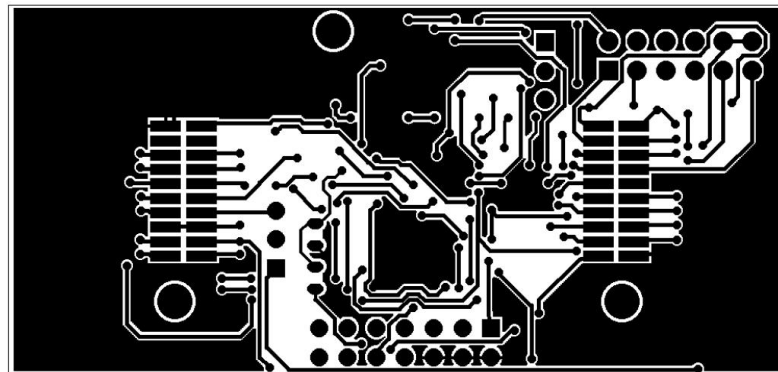


Figura B.3: Capa inferior

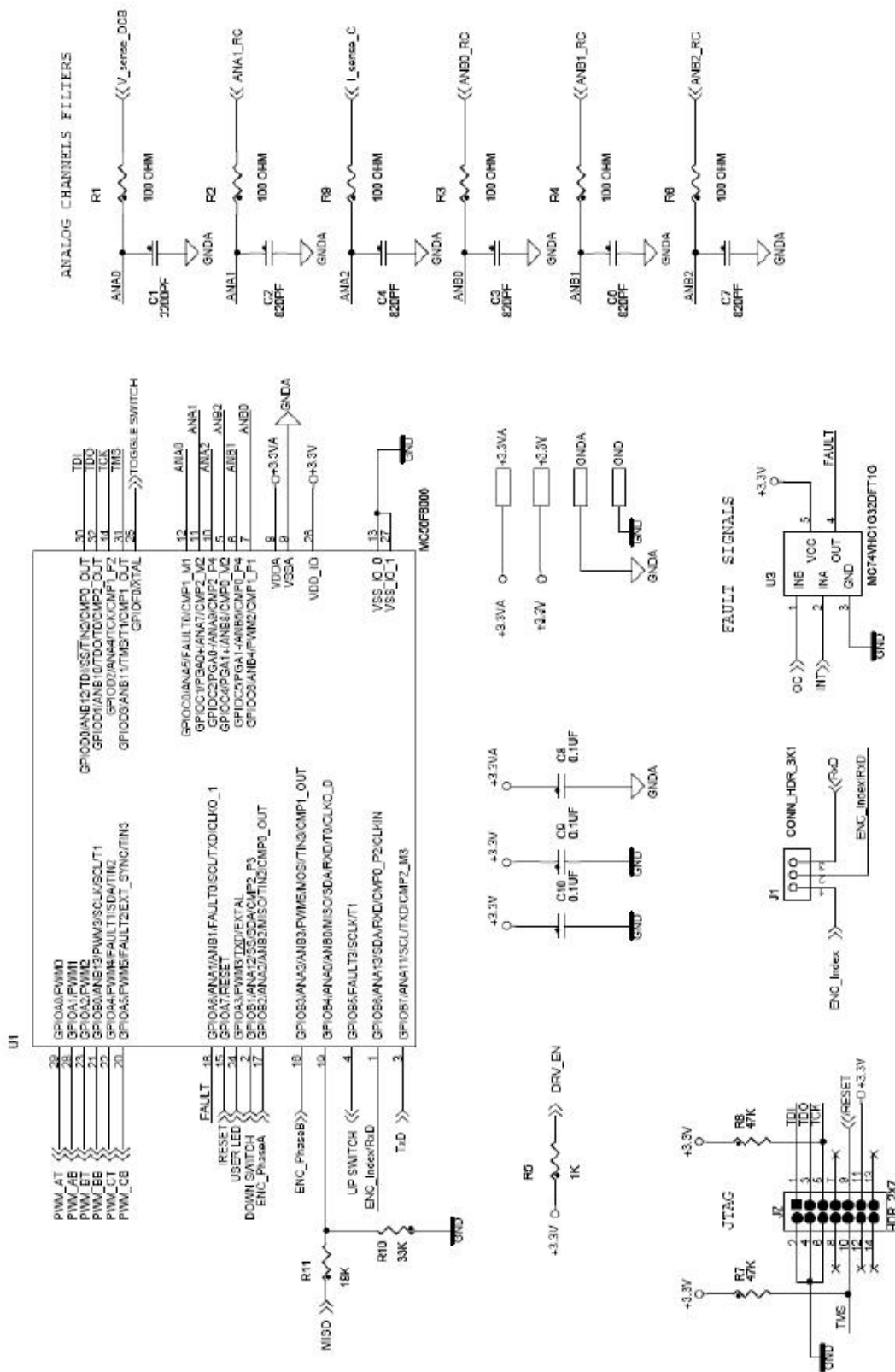


Figura B.4: Controlador y Cabeceras



# Anexo C

## Etapa de Potencia

### C.1. Lista de Materiales

Tabla C.1: Lista de Materiales para la Etapa de Potencia

Dispositivo	Cantidad	Descripcion	Fabricante	Numero de Parte
C1-3, C5	4	47pF/100V	Cualquiera	
C4, C5, C9, C10	4	100nF/100V	Cualquiera	
C12, C13, C17, C23	4	100nF/100V	Cualquiera	
C27, C29, C31, C38	4	100nF/100V	Cualquiera	
C40-C42, C44	4	100nF/100V	Cualquiera	
C6-C8	3	470pF/100V	Cualquiera	
C11, C26, C28, C30	4	2.2 $\mu$ F/100V	Cualquiera	
C14, C15	2	22pF/100V	Cualquiera	
C16	1	10 $\mu$ F/16V	Cualquiera	
C18	1	4.7 $\mu$ F/16V	Cualquiera	
C19	1	470nF/25V	Cualquiera	
C20-C22	3	1 $\mu$ F/25V	Cualquiera	
C24	1	2.2 $\mu$ F/50V	Cualquiera	
C25-C34	2	100nF/100V	Cualquiera	
C32,C33	2	100 $\mu$ F/80V	Cualquiera	

Dispositivo	Cantidad	Descripcion	Fabricante	Numero de Parte
C35	1	68 $\mu$ F/100V	Cualquiera	
C36-C39	2	47 $\mu$ F/6.3V	Cualquiera	
C37	1	100 $\mu$ F/16V	Cualquiera	
C43	1	100PF/100V	Cualquiera	
D1-D6, D9	7	LED Amarillo	AVAGO TECHNOLOGIES	HSMY-C170
D7,D8, D19	3	LED Verde	AVAGO TECHNOLOGIES	HSMG-C170
D10-D17	8	0.5A/20V Schottky	ON SEMICONDUCTOR	MBR0520LT
D18	1	1A/200V	ON SEMICONDUCTOR	MUR0520LT
J1	1	HDR 1 $\times$ 3	MOLEX	09-85-2038
J2	1	CON_2_TB	LUMBERG INC	KRM02
J3	1	Power Jack	SWITCHCRAFT	RAPC712X
J4	1	HDR 2 $\times$ 5	TYCO ELECTRONICS	4-103322-2
J5	1	HDR 1 $\times$ 2	TYCO ELECTRONICS	4-103321-4
J6	1	HDR 1 $\times$ 5	MOLEX	22-27-2051
J7,J8	2	87407-110	FCI	4-103322-2
J9	1	HDR 2 $\times$ 3	TYCO ELECTRONICS	
J10	1	CON USB	TYCO ELECTRONICS	292304-1
L1,L3	2	1 $\mu$ H	TDK	MLZ2012ATR0
L2	1	100 $\mu$ H	Wurth Elektronik	744778920
Q1	1	9.4A/60V MOSFET	FAIRCHILD SEM.	FQD11P06TF
Q2	1	200mA/60V NPN	ON SEMICONDUCTOR	MMBT2369ALT
Q3-Q8	8	7.5A/100V MOSFET	FAIRCHILD SEM.	FDS3672
R1, R60	2	1M $\Omega$ a 1/8W	Cualquiera	
R2, R7	2	7.5k $\Omega$ a 1/8W	Cualquiera	
R10, R13	2	7.5k $\Omega$ a 1/8W	Cualquiera	

Dispositivo	Cantidad	Descripcion	Fabricante	Numero de Parte
R14, R19, R81, R84	4	7.5k $\Omega$ a 1/8W	Cualquiera	
R3, R5, R8, R11	4	220 $\Omega$ a 1/8W	Cualquiera	
R15, R18, R102	3	220 $\Omega$ a 1/8W	Cualquiera	
R103	1	220 $\Omega$ a 1/8W	Cualquiera	
R4, R6, R9, R12	4	1.6k $\Omega$ a 1/8W	Cualquiera	
R16, R17, R82, R83	4	1.6k $\Omega$ a 1/8W	Cualquiera	
R20, R23, R25, R29	4	30k $\Omega$ a 1/8W	Cualquiera	
R22, R28, R31	3	3k $\Omega$ a 1/8W	Cualquiera	
R27, R104	2	1.5k $\Omega$ a 1/8W	Cualquiera	
R32, R36, R39, R41	4	1k $\Omega$ a 1/8W	Cualquiera	
R33, R37, R42	3	24 $\Omega$ a 1/8W	Cualquiera	
R34, R38, R43, R46	4	1.8k $\Omega$ a 1/8W	Cualquiera	
R50, R53, R97	3	1.6k $\Omega$ a 1/8W	Cualquiera	
R35	1	120 $\Omega$ a 1/8W	Cualquiera	
R40, R59, R85, R89	4	0 $\Omega$ a 1/8W	Cualquiera	
R44, R47, R49, R51	4	270 $\Omega$ a 1/8W	Cualquiera	
R54-R56, R63, R64	5	270 $\Omega$ a 1/8W	Cualquiera	
R45, R48, R52	3	4.7k $\Omega$ a 1/8W	Cualquiera	
R57, R58, R96	3	4.7k $\Omega$ a 1/8W	Cualquiera	
R61, R62	2	33 $\Omega$ a 1/8W	Cualquiera	
R66, R67	2	10 $\Omega$ a 1/8W	Cualquiera	
R69-R76	8	10 $\Omega$ a 1/8W	Cualquiera	
R79, R80	2	10 $\Omega$ a 1/8W	Cualquiera	
R68, R94, R95	3	10k $\Omega$ a 1/8W	Cualquiera	
R98-R101	4	10k $\Omega$ a 1/8W	Cualquiera	
R77	1	1k $\Omega$ a 1/8W	Cualquiera	
R78	1	15k $\Omega$ a 1/8W	Cualquiera	
R85-88	4	100m $\Omega$ a 1/8W	Cualquiera	
R90, R93	2	68 $\Omega$ a 1/8W	Cualquiera	
R91	1	820 $\Omega$ a 1/8W	Cualquiera	

<b>Dispositivo</b>	<b>Cantidad</b>	<b>Descripcion</b>	<b>Fabricante</b>	<b>Numero de Parte</b>
SW1, SW2	2	20mA/32V	C&K COMPONENTS	KSC821JLFS
SW3	1	20mA/20V 3 Estados	APEM COMPONENTS	TL39P0050
SW4	1	20mA/32V	C&k COMPONENTS	KSC341JLFS
TP1-TP45	45			
U1, U2	2	Amp. Operacional	ON SEMICONDUCTOR	MC33502DRG2
U3	1	Interfaz CAN	PHILIPS SEM.	PCA82C250TD
U4	1	Inversor Schmitt	ON SEMICONDUCTOR	MC74AC14DG
U5	1	Controlador de 8 bits	FREESCALE	MC9S08JM60CF
U6	1	Puente de 3 Fases	FREESCALE	MCZ33927EK
U7	1	Regulador	NATIONAL SEM.	LM2594HVM-5.0
U8	1	Regulador	ON SEM.	MC33269D-3.3
X1	1	Oscilador de 8MHz	RAKON	LFA140E



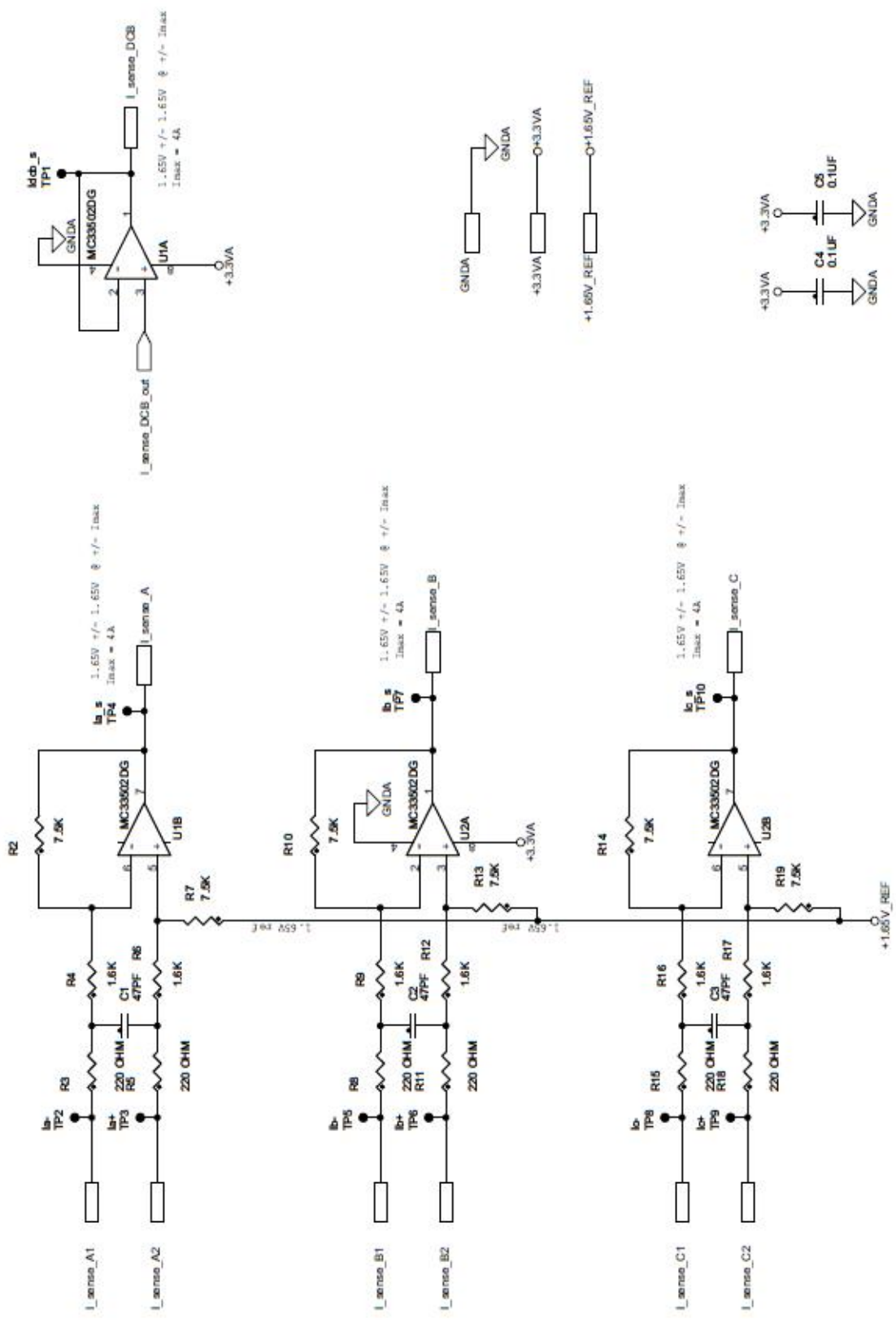


Figura D.2: Fase para el sensado de corriente

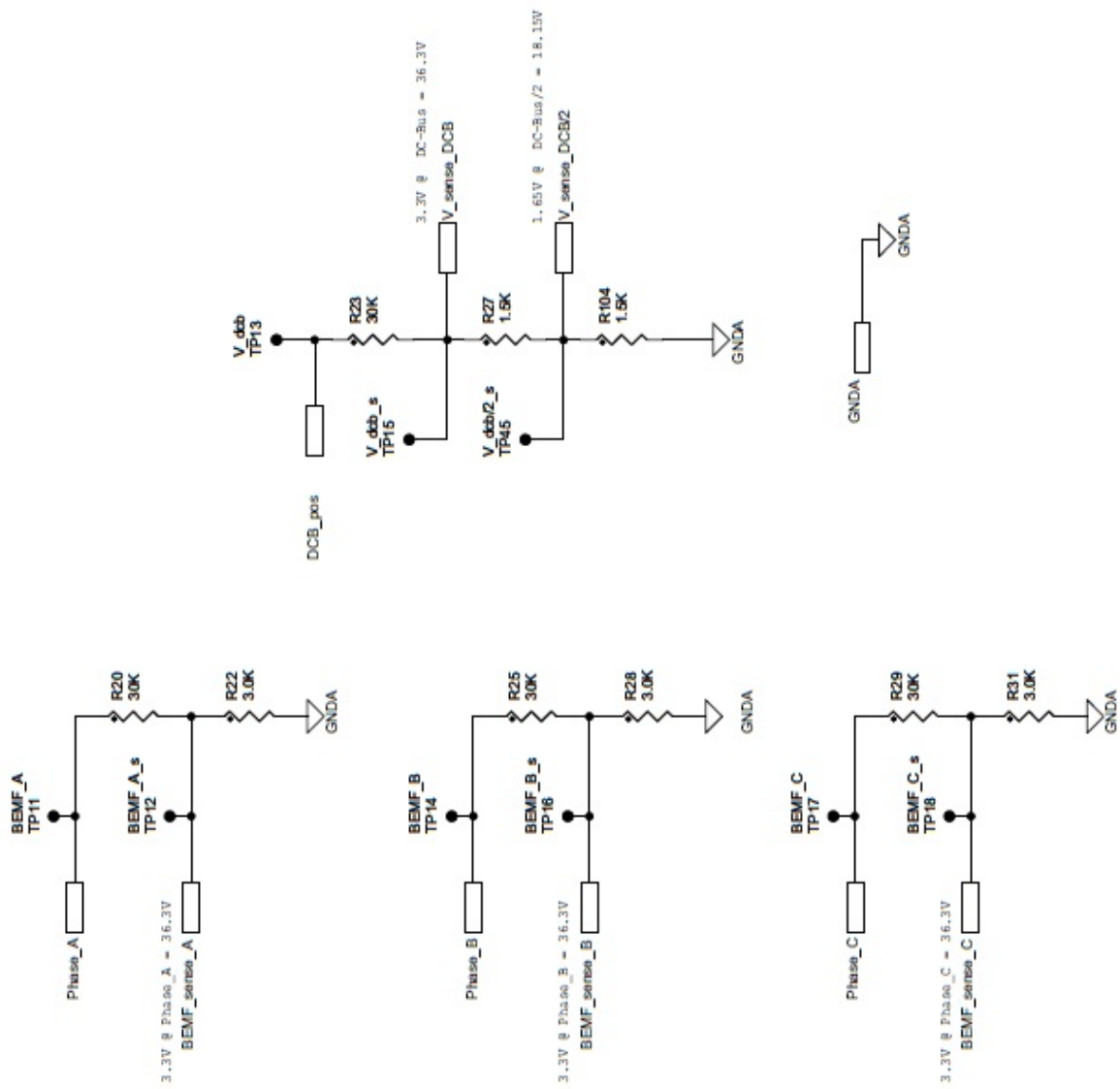


Figura D.3: Fase para sensar la fuerza contraelectromotriz

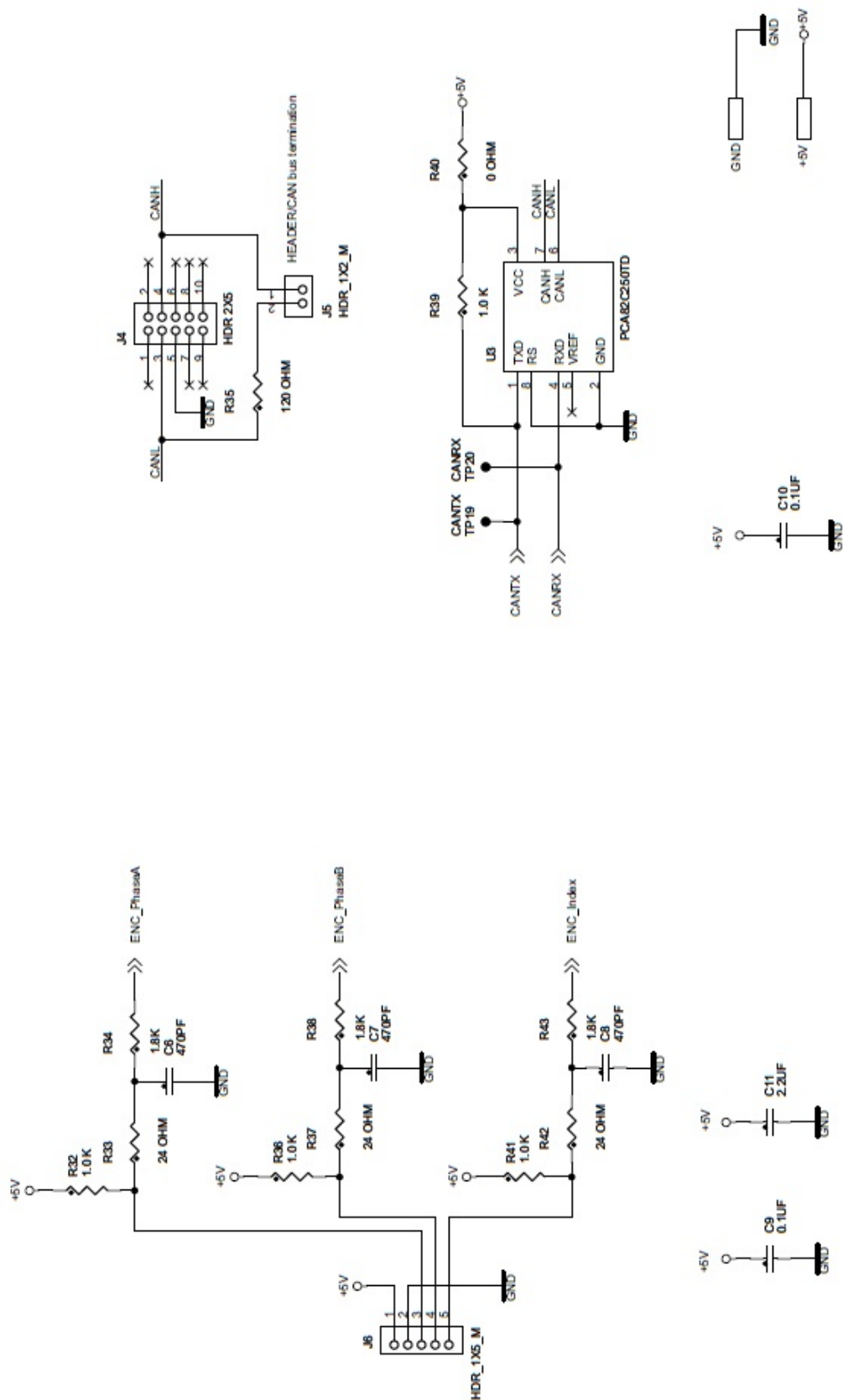


Figura D.4: Conectores y otros circuitos para señales de sensores de efecto hall



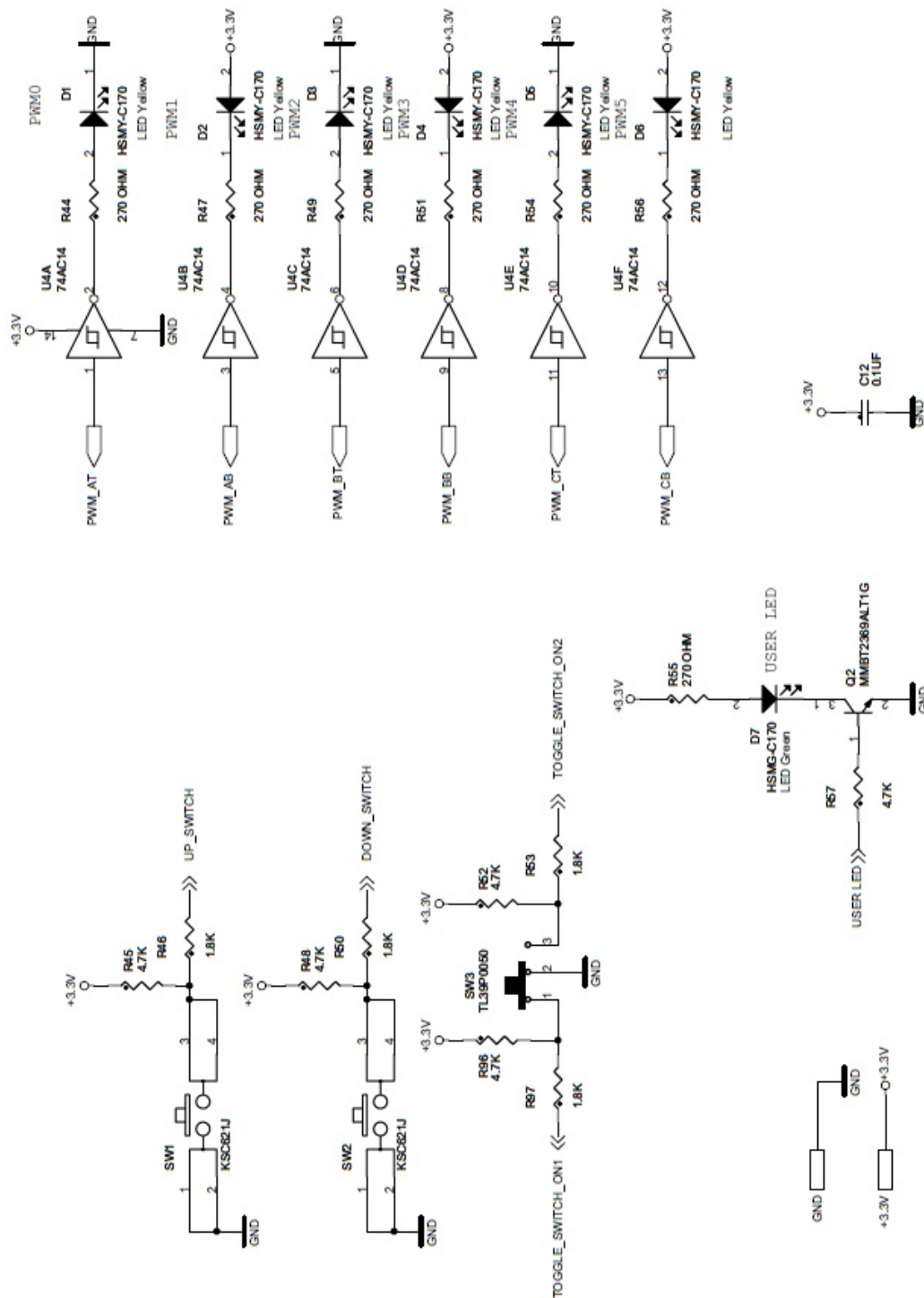


Figura D.5: Conectores, interruptores e indicadores de señales PWM

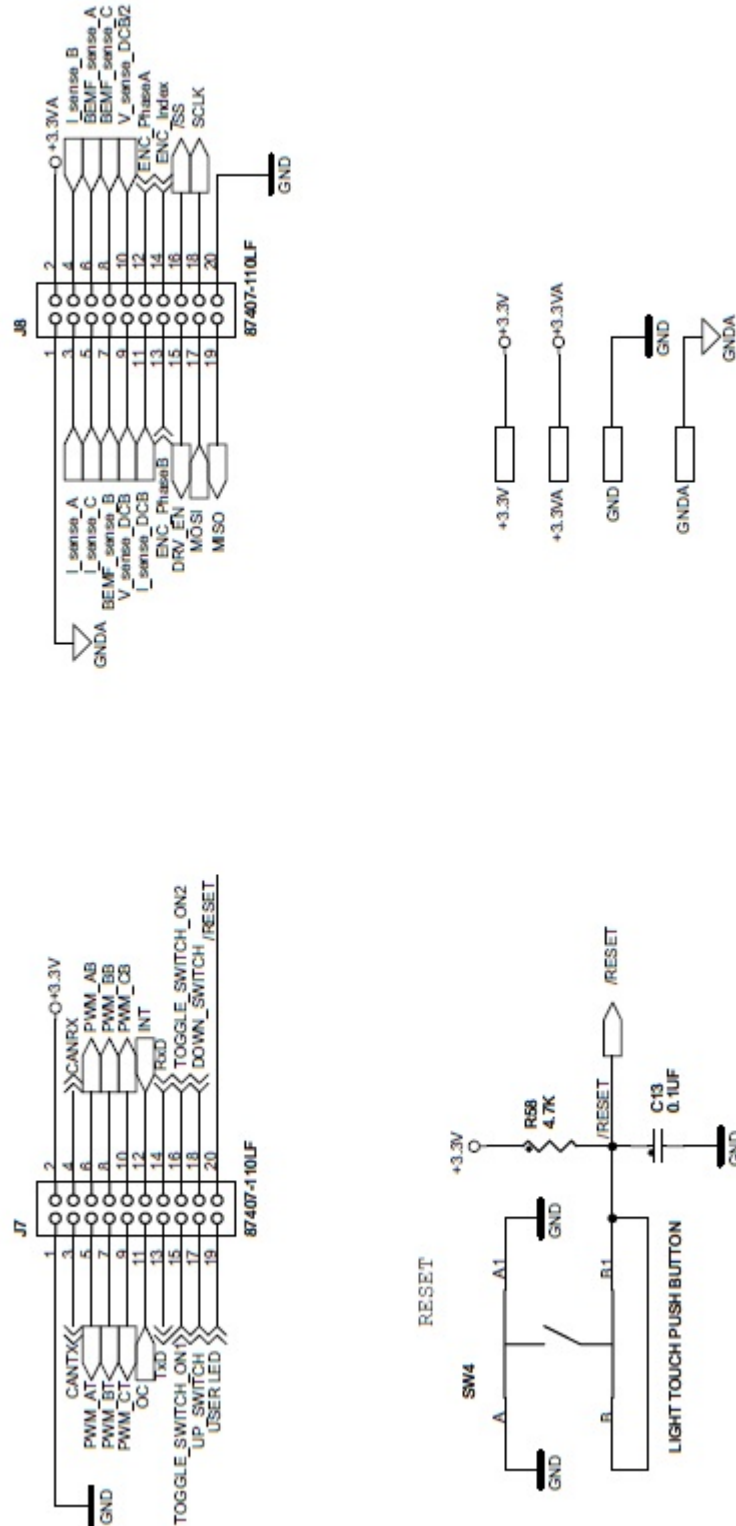


Figura D.6: Conectores para la tarjeta de control

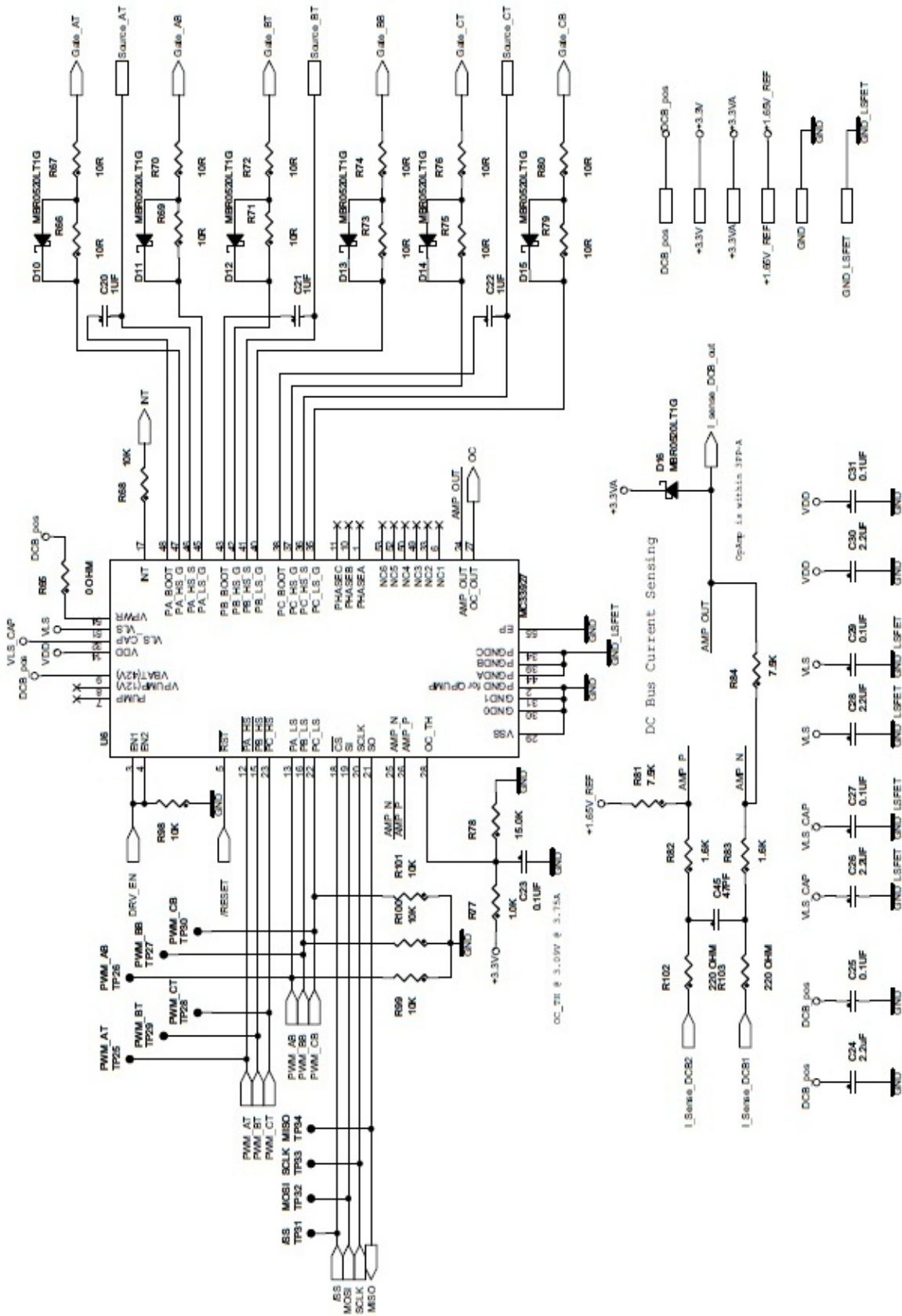


Figura D.7: Etapa de control y direccionamiento de señales hacia los MOSFET

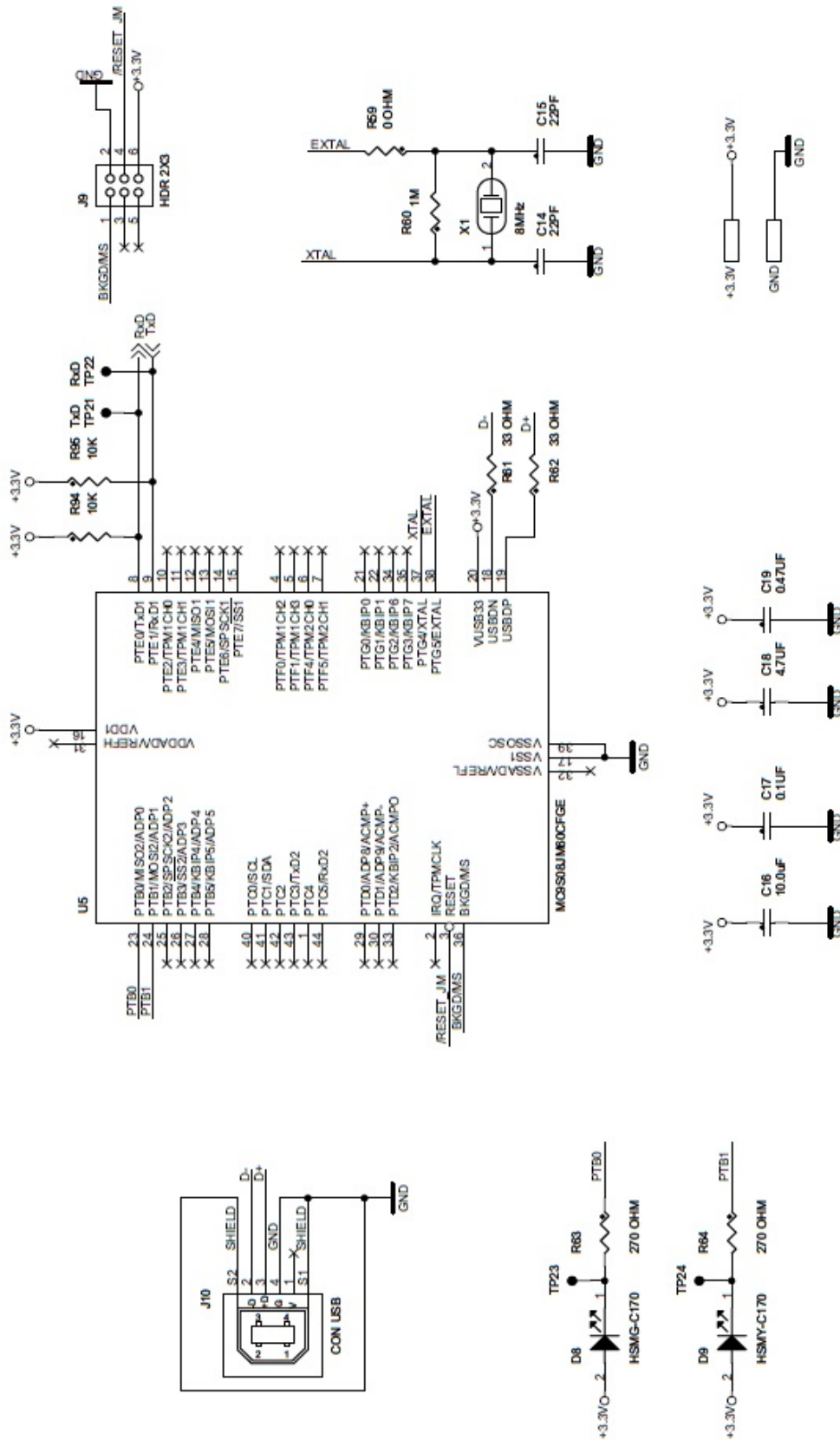


Figura D.8: Circuito para comunicación serial

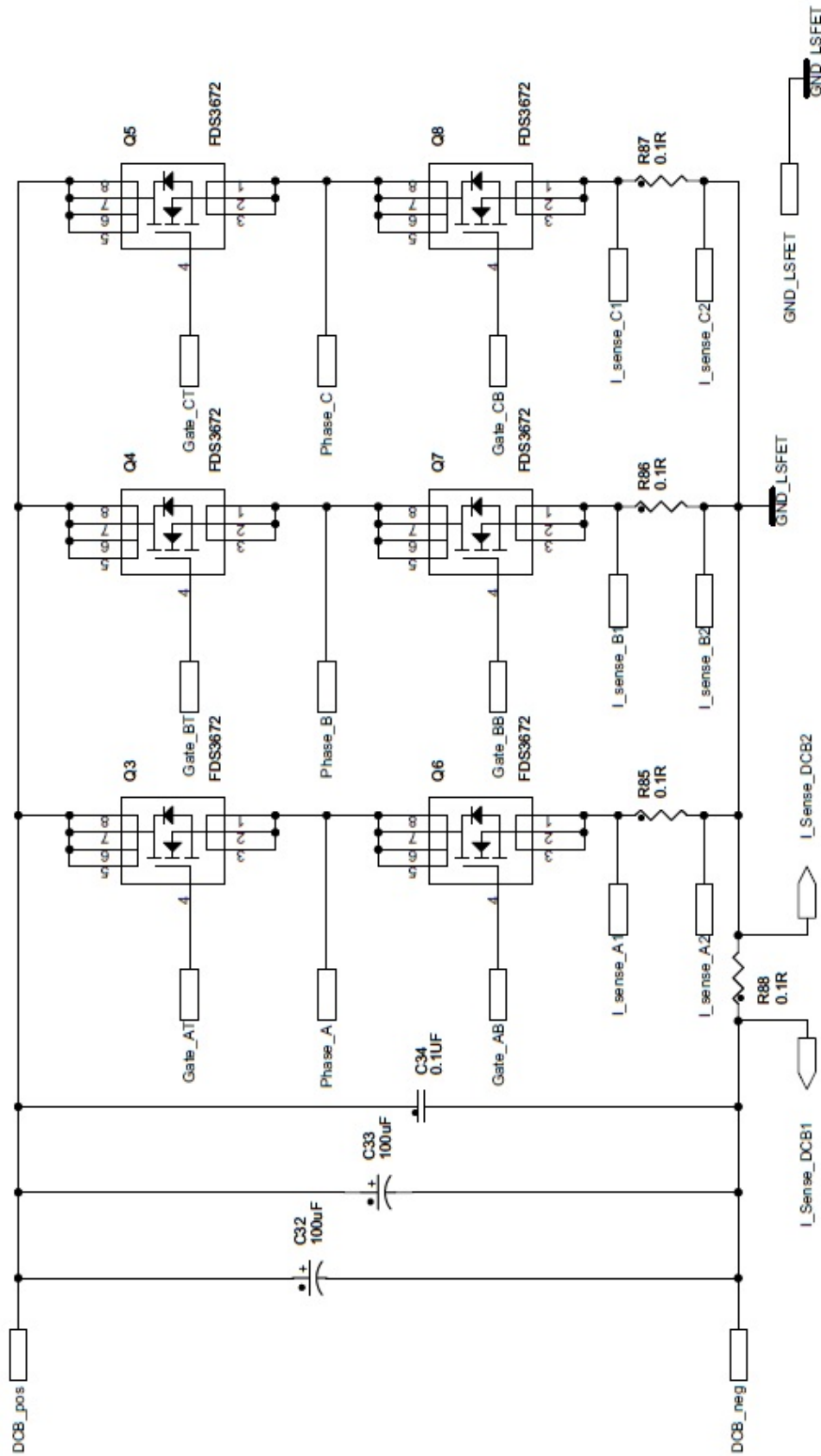


Figura D.9: Circuito de Potencia

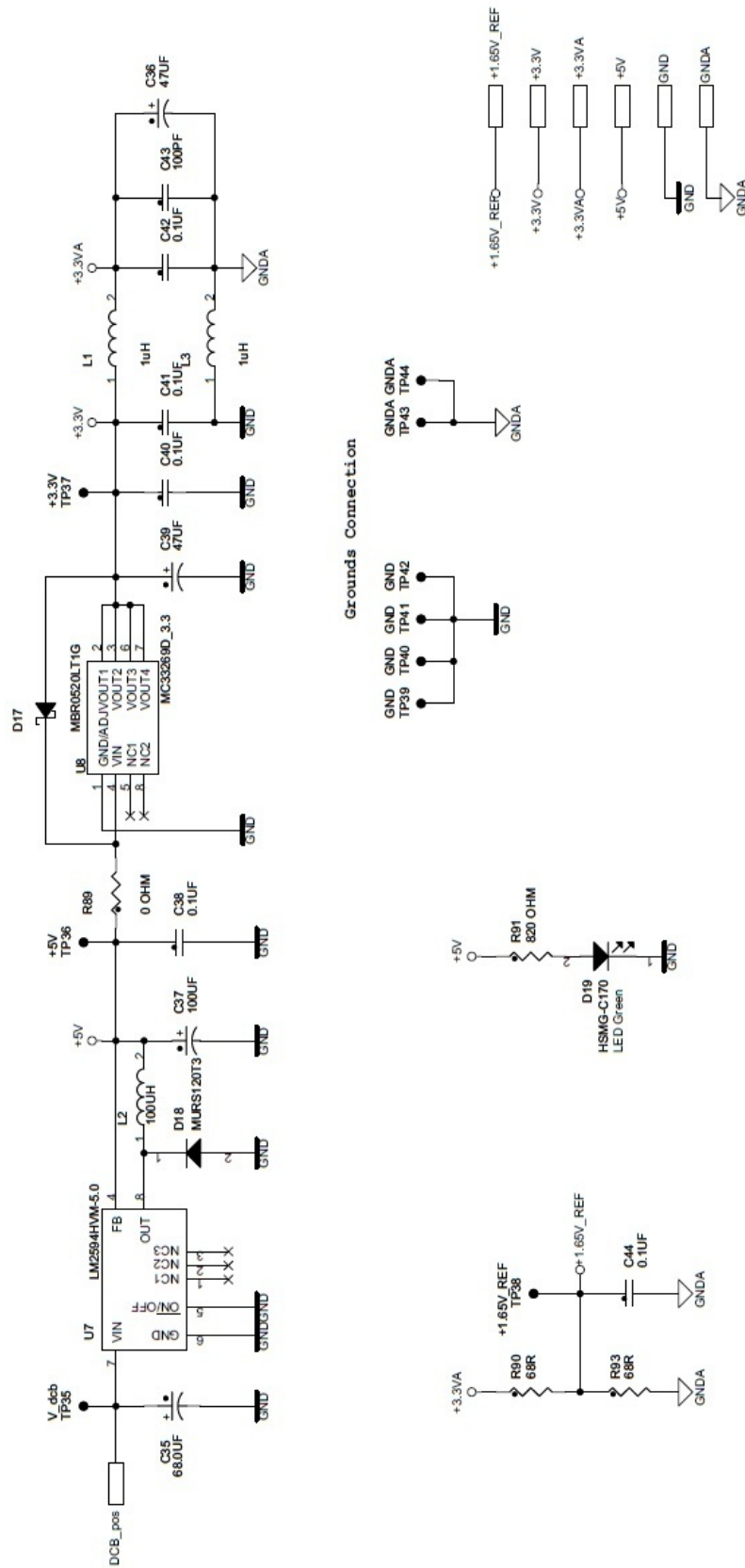


Figura D.10: Fuentes de alimentación

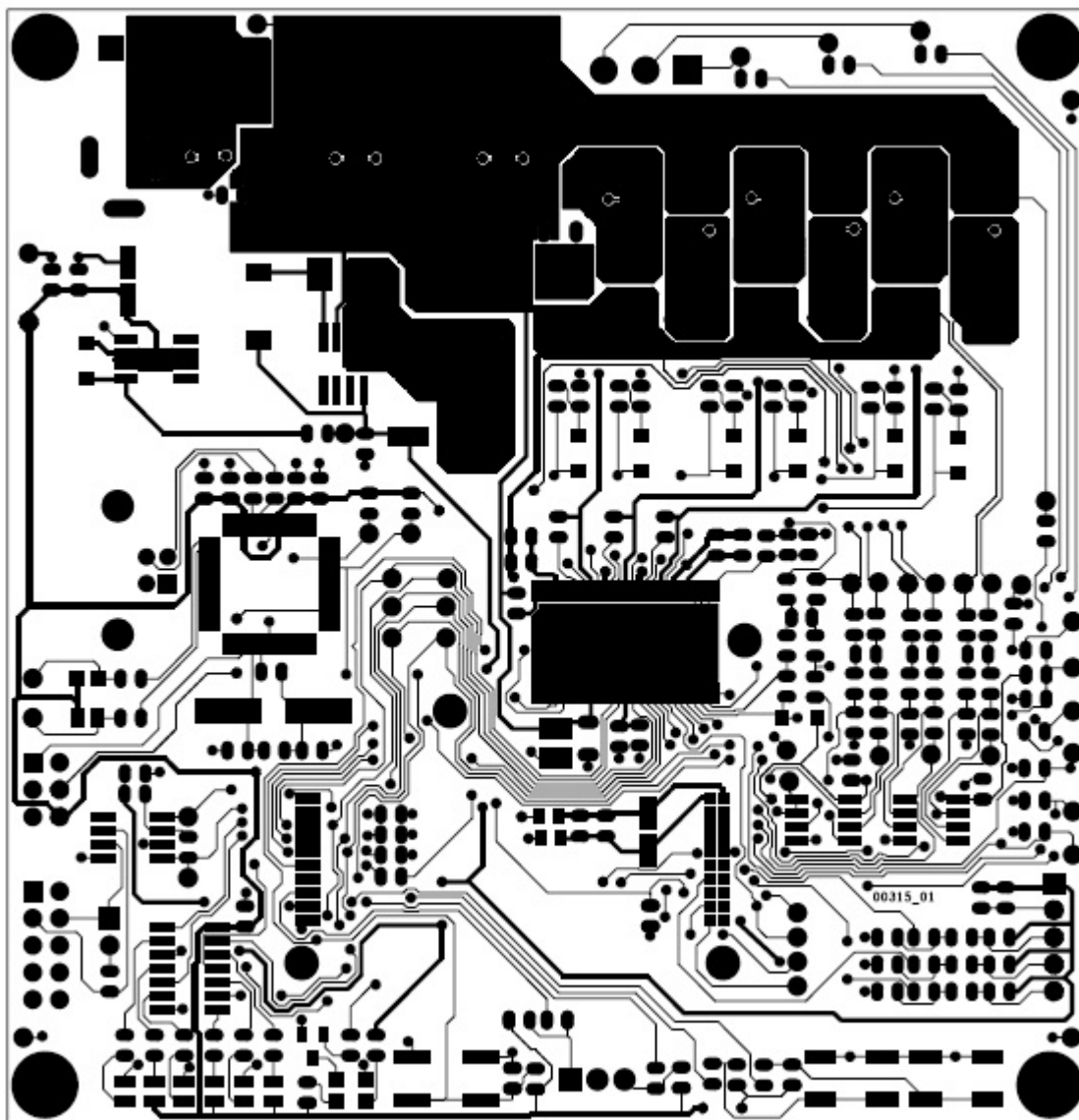


Figura D.11: Circuito impreso de la etapa de potencia en la parte superior

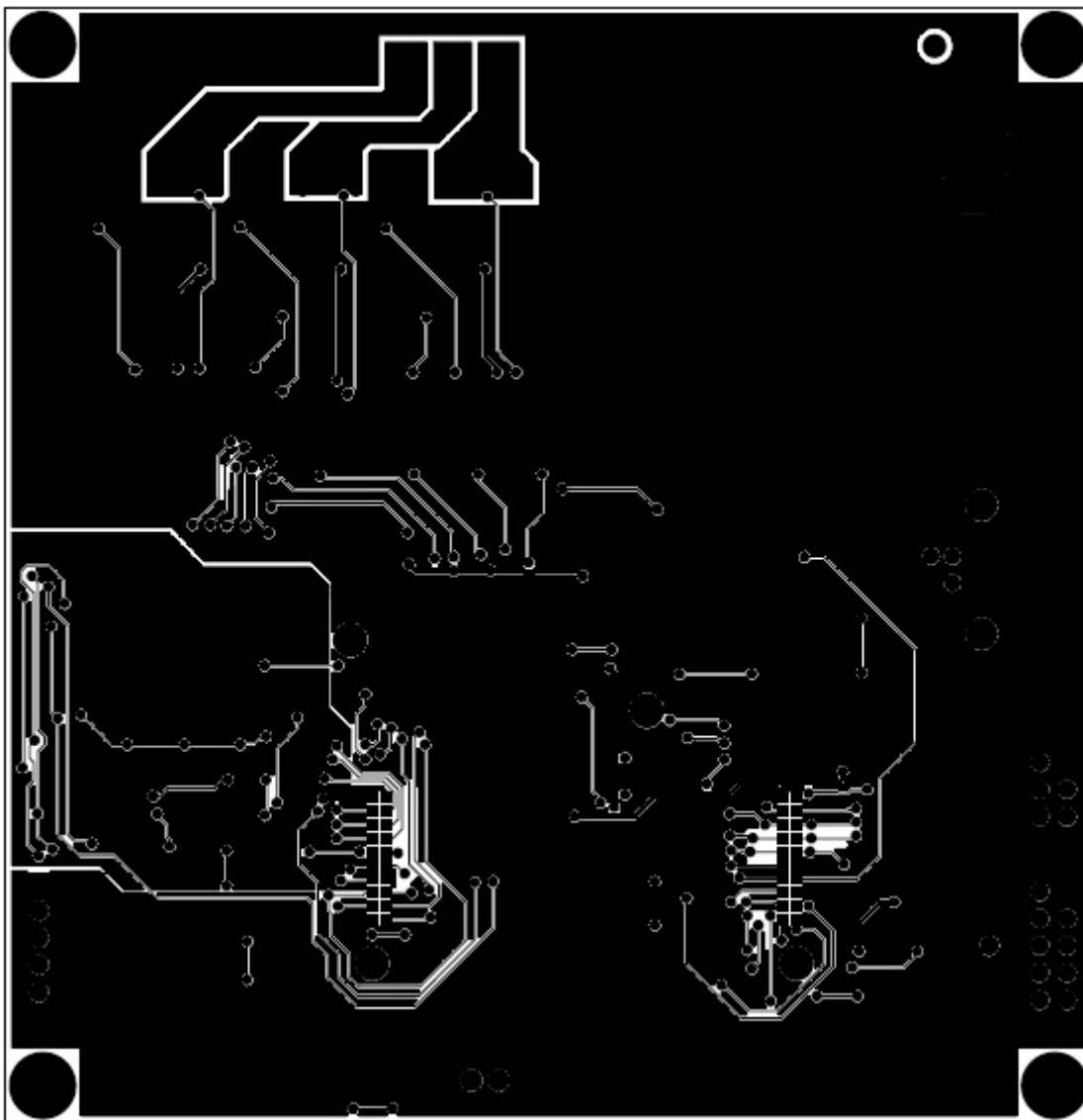


Figura D.12: Circuito impreso de la etapa de potencia en la parte inferior



# Anexo E

## Programación del Algoritmo

A continuación se presenta el código generado para para el control del motor, en Main.

### E.1. Declaración de Librerías

```
#include "Cpu.h"
#include "Events.h"
#include "PWMC1.h "
#include "Bit1.h "
#include "Hall_A.h"
#include "Hall_B.h"
#include "Hall_C.h"
#include "UP.h "
#include "DOWN.h "
#include "FC321.h "
#include "TI1.h "
#include "AD1.h "
\*Se incluyen modulos que se usan durante todo el proyecto*\
#include "AD1.h "
#include "PE.Types.h"
#include "PE.Error.h"
#include "PE.Const.h"
#include "IO.Map.h "
```

```
#include "qs.h "  
#include "gpio.h "  
#include "sys.h "  
#include "cop.h "  
#include "sci.h "  
  
#include "freemaster.h"  
#include "MC33927.h "  
#include "SPI_00361_01.h"
```

## E.2. Declaración de variables

```
static MC33927_LATCH_T mudtStatusRegister0;  
static MC33927_MODE_T mudtStatusRegister1;  
static MC33927_LATCH_T mudtStatusRegister2;  
static MC33927_MODE_COMMAND_T mudtModeCommands;  
static MC33927_LATCH_T mudtMaskInterrupts;  
static MC33927_LATCH_T mudtClearFlags;  
  
#define SET_MODE_DESAT_DISABLE    0x48  
static byte mbytDeadtime;  
static void GpioInit(void);  
  
#define PWM_A_out PWM_CTRL=0x8307;  
#define PWM_B_out PWM_CTRL=0x8C07;  
#define PWM_C_out PWM_CTRL=0xB007;  
#define PWM_A_off PWM_CNFG=0x0710; PWMC1_SetRatio16(PWM_AA,0);  
#define PWM_B_off PWM_CNFG=0x0720; PWMC1_SetRatio16(PWM_BB,0);  
#define PWM_C_off PWM_CNFG=0x0740; PWMC1_SetRatio16(PWM_CC,0);  
  
#define PWM_AA 1  
#define PWM_BB 3
```

```
#define PWM_CC 5
#define NUM_MEASUREMENTS 10

int ciclo,aa,n,sector;
int A,B,C,Pos;
word ct,ct1;
extern int set_point;
unsigned int EvnCnt=0;
byte Values[3];
volatile int hola;
extern long int c_t;
float torque;
word conversion;
float i_bus;
void conmutacion(void);
```

### E.3. Registros para comunicación

```
void main(void){
  \*Inicializa el modulo COP*\
  ioctl(COP, COP_INIT, NULL);
  \*Inicializa el modulo SYS*\
  ioctl(SYS, SYS_INIT, NULL);
  \*Inicializa el modulo SPI*\
  SPI_init();
  \*Inicializa el modulo GPIO*\
  GpioInit();

  Cpu_Delay100US(10000);

  \*Este comando lee el estado de Registro 0 del MC33927 y lo almacena en el registro
  mudtStatusRegister0*\
  MC33927_GetSR0(&mudtStatusRegister0);
```

```
\*Este comando lee el estado de Registro 1 del MC33927 y lo almacena en el registro mudtStatusRegister1*\
```

```
MC33927_GetSR1(&mudtStatusRegister1);
```

```
\*Este comando lee el estado de Registro 2 del MC33927 y lo almacena en el registro mudtStatusRegister2*\
```

```
MC33927_GetSR2(&mudtStatusRegister2);
```

```
\*Este comando lee el estado de Registro 3 (tiempo muerto) del MC33927 y lo almacena en la variable mbytDeadtime*\
```

```
mbytDeadtime = MC33927_GetSR3();
```

```
\***** Modo de Configuración *****\
```

```
mudtModeCommands.B.Lock = 0; \*0: Sigue escribiendo, 1: Termina la configuración hasta el reseteo*\
```

```
mudtModeCommands.B.DesatFaultMode = 1; \* 0; Cuando la desaturación o el error de fase en cualquier MOSFET desactiva todos los MOSFETs; 1: Ignora las fallas *\
```

```
\* Este comando controla el modo de falla de desaturación, en modo de encendido y desbloqueo de bits *\
```

```
MC33927_ModeCommand(&);
```

```
mudtModeCommands \* Configuración de Interrupciones *\
```

```
mudtMaskInterrupts.B.TLim = 0;
```

```
mudtMaskInterrupts.B.Desat = 0;
```

```
mudtMaskInterrupts.B.LowVLS = 0;
```

```
mudtMaskInterrupts.B.Overcurrent = 1;
```

```
mudtMaskInterrupts.B.PhaseError = 0;
```

```
mudtMaskInterrupts.B.FramingError = 0;
```

```
mudtMaskInterrupts.B.WriteError = 0;
```

```
mudtMaskInterrupts.B.ResetEvent = 0;
```

```
\*Este comando controla el enmascaramiento de los registros *\
```

```
MC33927_MaskInterrupts(&mudtMaskInterrupts);
```



```
FMSTR_Poll();  
}  
}
```

## E.4. Detección de Posición del Motor

```
void posicion()  
{  
if (A==0&&B==8&&C==0){  
n=1; }  
if (A==4&&B==8&&C==0){  
n=2; }  
if (A==4&&B==0&&C==0){  
n=3; }  
if (A==4&&B==0&&C==64){  
n=4; }  
if (A==0&&B==0&&C==64){  
n=5; }  
if (A==0&&B==8&&C==64){  
n=6; }  
  
sector=n+1;  
if (sector>6){  
sector=1; }  
ciclo=sector;  
conmutacion();  
}
```

## E.5. Conmutación

```
void conmutacion(void){  
aa=Bit1_GetRawVal();
```

```
if(aa==0){
if(ciclo>6){
ciclo=1; }
if(ciclo==1){
\*La señal entra en A, sale en C y B esta desactivada*\
PWMC1_SetRatio16(PWM_AA,ciclo_v);
PWM_C_out;
PWM_B_off;
\*Activa Salida*\
PWMC1_Load();
PWMC1_OutputPadEnable();
}
if(ciclo==2){
\*La señal entra en A, sale en B y C esta desactivada*\
PWMC1_SetRatio16(PWM_AA,ciclo_v);
PWM_B_out;
PWM_C_off;
\*Activa Salida*\
PWMC1_Load();
PWMC1_OutputPadEnable();
}

if(ciclo==3){
\*La señal entra en C, sale en B y A esta desactivada*\
PWMC1_SetRatio16(PWM_CC,ciclo_v);
PWM_B_out;
PWM_A_off;
\*Activa Salida*\
PWMC1_Load();
PWMC1_OutputPadEnable();
}

if(ciclo==4){
```

```
\*La señal entra en C, sale en A y B esta desactivada*\
PWM_C1_SetRatio16(PWM_CC,ciclo_v);
PWM_A_out;
PWM_B_off;
\*Activa Salida*\
PWM_C1_Load();
PWM_C1_OutputPadEnable();
}

if(ciclo==5){
\*La señal entra en B, sale en A y C esta desactivada*\
PWM_C1_SetRatio16(PWM_BB,ciclo_v);
PWM_A_out;
PWM_C_off;
\*Activa Salida*\
PWM_C1_Load();
PWM_C1_OutputPadEnable();
}

if(ciclo==6){
\*La señal entra en B, sale en C y A esta desactivada*\
PWM_C1_SetRatio16(PWM_BB,ciclo_v);
PWM_C_out;
PWM_A_off;
\*Activa Salida*\
PWM_C1_Load();
PWM_C1_OutputPadEnable();
}

}
else{
PWM_C1_OutputPadDisable();
set_point=0;
ct=0;
}
```



```
}  
}  
  
static void GpioInit(void){  
#ifdef GPIO_A  
ioctl(GPIO_A, GPIO_INIT, NULL);  
#endif  
#ifdef GPIO_B  
ioctl(GPIO_B, GPIO_INIT, NULL);  
#endif  
#ifdef GPIO_C  
ioctl(GPIO_C, GPIO_INIT, NULL);  
#endif  
#ifdef GPIO_D  
ioctl(GPIO_D, GPIO_INIT, NULL);  
#endif  
#ifdef GPIO_E  
ioctl(GPIO_E, GPIO_INIT, NULL);  
#endif  
#ifdef GPIO_F  
ioctl(GPIO_F, GPIO_INIT, NULL);  
#endif  
}
```

Los eventos, es todo aquel código que se genera para que cada uno de los componentes que requieren de un tiempo de muestro cumpla con su función, el código para Events es el siguiente:

```
\**** Modulo de Eventos ****\  
#include "CPU.h "  
#include "Events.h"  
  
\* Se declaran variables *\  
extern word ct;
```

```
bool up,down;
long int c_t,c_tv;
long int ciclo_v, ciclo_vant;
long int pulsos_totales;
int pulsos, set_point;

extern float tiempo,RPM;
extern word conversion;
extern unsigned int EvnCnt;
extern float i_bus;
int i_bus1;

float kp=2,ki=5,kd=.05,error,error1,error2,error3,T=0.020,P,I,D;
float kp_v=0.5, ki_v=0.01, kd_v=.06,error_v,error1_v,error2_v,error3_v,T_v=0.030,P_v,I_v,D_v;
extern float torque;
int RPM1;
```

## E.6. Calculo de velocidad

```
void velocidad(void)
{
pulsos_totales++;
pulsos++;

if(pulsos==6){
FC321_GetTimeReal(&tiempo);
FC321_Reset();
RPM=6/6.0001*(1/tiempo)*60;
pulsos=0;
}
}

#pragma interrupt called
```

```
void Hall_C_OnInterrupt(void)
{
    velocidad();
}
```

```
#pragma interrupt called
void Hall_B_OnInterrupt(void)
{
    velocidad();
}
```

```
#pragma interrupt called
void Hall_A_OnInterrupt(void)
{
    velocidad();
}
```

```
#pragma interrupt called
void DOWN_OnInterrupt(void)
{
    set_point=set_point-500;
    if(set_point<0){
        set_point=0;
    }
}
```

```
#pragma interrupt called
void UP_OnInterrupt(void)
{
    set_point=set_point+2000;
}
```

## E.7. Control PID para Velocidad

```
#pragma interrupt called
void TI1_OnInterrupt(void)
{
error=set_point-RPM;
P=kp*error;
I=ki*T*(set_point-(RPM+RPM1)/2);
D=(kd/(6*T))*((error-error3)+3*(error1-error2));

c_t=P+I+D+c_tv;

if(c_t<0){
c_t=0;
}

c_tv=c_t;
RPM1=RPM;
error3=error2;
error2=error1;
error1=error;
```

## E.8. Control PI para Corriente

```
c_t=torque;
error_v=torque-i_bus;
P_v=kp_v*error_v;
I_v=ki_v*T_v*(torque-(i_bus+i_bus1)/2);

ciclo_v=P_v+I_v+ciclo_vant;

ciclo_vant=ciclo_v;
```

```
i_bus1=i_bus;  
}
```

```
void AD1_OnEnd(void) {  
AD1_GetValue(&conversion);  
i_bus=conversion*0.0002442;  
}
```