

UNIVERSIDAD POLITÉCNICA DE PUEBLA

MAESTRÍA EN INGENIERÍA EN
AUTOMATIZACIÓN DE LOS PROCESOS INDUSTRIALES



“CONTROL POR PASIVIDAD DE LA VELOCIDAD
DE UN MOTOR TRIFÁSICO DE IMANES PERMANENTES
IMPLEMENTADO EN UN FPGA.”

TESIS DE MAESTRÍA

ING. BRAULIO LOZANO HERNÁNDEZ

UNIVERSIDAD POLITÉCNICA DE PUEBLA

MAESTRÍA EN INGENIERÍA EN
AUTOMATIZACIÓN DE LOS PROCESOS INDUSTRIALES



“CONTROL POR PASIVIDAD DE LA VELOCIDAD
DE UN MOTOR TRIFÁSICO DE IMANES PERMANENTES
IMPLEMENTADO EN UN FPGA.”

TESIS DE MAESTRÍA

ING. BRAULIO LOZANO HERNÁNDEZ

DIRECTOR

CDR. JACOB JAVIER VÁSQUEZ SANJUAN
UNIVERSIDAD POLITÉCNICA DE PUEBLA

COASESOR

DR. JOSÉ DE JESÚS RANGEL MAGDALENO
INSTITUTO NACIONAL DE ASTROFÍSICA, ÓPTICA Y ELECTRÓNICA.

Índice general

| | |
|--|-----------|
| Índice de figuras | III |
| Índice de cuadros | VII |
| 1. Antecedentes | 1 |
| 1.1. Introducción | 1 |
| 1.2. Objetivos | 2 |
| 1.2.1. Objetivo general | 2 |
| 1.2.2. Objetivos particulares | 2 |
| 1.3. Justificación | 3 |
| 1.4. Alcances y limitaciones | 3 |
| 1.5. Estado del arte. | 4 |
| 1.5.1. Estado del arte del control de motores de imanes permanentes. | 5 |
| 1.6. Organización del trabajo | 6 |
| 2. Modelado del motor eléctrico de imanes permanentes (PMSM). | 7 |
| 2.1. Motor eléctrico síncrono de imanes permanentes (PMSM) de corriente alterna trifásico. | 8 |
| 2.1.1. Configuración básica de los motores síncronos de imanes permanentes. | 8 |
| 2.1.1.1. Imanes permanentes. | 9 |
| 2.2. Modelado del motor eléctrico trifásico de imanes permanentes (PMSM). | 11 |
| 2.3. Implementación del modelo del PMSM en simulink. | 14 |
| 3. Diseño del control por pasividad de un motor eléctrico de imanes permanentes (PMSM). | 17 |
| 3.1. Control por pasividad. | 18 |
| 3.2. Diseño del control por pasividad PMSM. | 18 |
| 3.3. Implementación del control por pasividad de PMSM en el programa simulink. | 21 |
| 4. Implementación del control por pasividad del PMSM en un FPGA. | 23 |
| 4.1. Dispositivo FPGA. | 24 |
| 4.1.1. Representación binaria en punto flotante | 25 |
| 4.1.2. Representación binaria en punto fijo. | 25 |
| 4.1.3. Operaciones aritméticas con palabras binarias con formato Q(m.f). | 27 |
| 4.2. Lenguaje de programación en VHDL | 27 |
| 4.2.1. Estructura de un programa VHDL. | 28 |
| 4.2.1.1. ENTITY en VHDL | 28 |
| 4.2.1.2. Arquitectura de VHDL | 29 |
| 4.3. Generación del código en alto nivel del control por pasividad de PMSM. | 30 |
| 4.4. Acondicionamiento en VHDL de las señales. | 41 |

| | |
|--|------------|
| 5. Análisis comparativo de resultados y conclusiones. | 45 |
| 5.1. Análisis de las cosimulaciones del control por pasividad implementado en simulink del PMSM. | 45 |
| 5.1.1. Simulación del control por pasividad en simulink | 46 |
| 5.1.2. Error del control por pasividad implementado en simulink | 49 |
| 5.2. Cosimulación del control por pasividad en VHDL del PMSM. | 51 |
| 5.2.1. Cosimulación del control por pasividad en VHDL | 51 |
| 5.2.2. Error del control por pasividad implementado en VHDL | 54 |
| 5.3. Comparación de la respuesta del control implementado en simulink y en Active-HDL. | 56 |
| 5.4. Análisis de las cosimulaciones del control en simulink del PMSM con inversor en PSIM | 57 |
| 5.4.1. Cosimulación del control por pasividad en simulink con inversor en PSIM para distintas frecuencias de la señal triangular | 57 |
| 5.4.2. Cosimulación del control por pasividad en simulink con inversor en PSIM a 20 kHz. | 59 |
| 5.5. Análisis de las cosimulaciones del control por pasividad implementado en VHDL del PMSM con inversor en PSIM. | 61 |
| 5.5.1. Cosimulación del control por pasividad en VHDL con inversor en PSIM con 20 kHz de frecuencia de la señal triangular con diferentes ganancias. | 61 |
| 5.6. Implementación del código VHDL en un dispositivo embebido FPGA. | 63 |
| 5.7. Conclusiones. | 65 |
| 5.8. Publicaciones realizadas | 66 |
| 5.9. Trabajos futuros. | 66 |
| Bibliografía | 67 |
| | |
| I ANEXOS | 71 |
| A. Implementación detallada del PMSM en Simulink. | 73 |
| B. Respuesta del control en Simulink para diferentes ganancias. | 81 |
| C. Proceso de interconexión entre Simulink y Active_HDL, Simulink y PSIM. | 85 |
| D. Respuesta del control en VHDL para diferentes ganancias. | 89 |
| E. Co-simulación de simulink y PSIM del control por pasividad. | 93 |
| F. Co-simulación de simulink y PSIM del control por pasividad con una frecuencia 20 kHz con diferentes ganancias. | 97 |
| G. Co-simulación del control en VHDL con PSIM a una frecuencia 20 kHz con diferentes ganancias. | 101 |

Índice de figuras

| | |
|---|----|
| 1.1. Motores empleados en accionamientos de velocidad variable. | 4 |
| 1.2. Diagrama a bloques de un sistema de control a lazo cerrado de un motor PMSM de AC trifásico. | 6 |
| 2.1. Configuración básica de PMSM. a)PMSM de imanes superficiales, b)PMSM de imanes interiores. | 8 |
| 2.2. Representación de motor PMSM en los ejes dq | 11 |
| 2.3. Transformada de Clark y Park en simulink | 14 |
| 2.4. Conversión Clark en simulink | 14 |
| 2.5. Bloque en simulink de PMSM | 15 |
| 2.6. Bloque en simulink de PMSM con parámetros incluidos. | 16 |
| 3.1. Bloque en simulink del control por pasividad de PMSM con parametros. | 21 |
| 3.2. Bloque en simulink del control por pasividad de PMSM con parámetros incluidos. 22 | |
| 4.1. Bloque en VHDL de la función compara. | 30 |
| 4.2. Bloque de la función suma en Active-HDL. | 31 |
| 4.3. Bloque de la función resta en Active-HDL. | 32 |
| 4.4. Bloque de la función multiplicación en Active-HDL. | 33 |
| 4.5. Bloque de constantes en Active-HDL. | 34 |
| 4.6. Diagrama del bloque de i_q^* en VHDL. | 38 |
| 4.7. Bloque de i_q^* en VHDL. | 38 |
| 4.8. Diagrama de control por pasividad en VHDL. | 38 |
| 4.9. Bloque de control por pasividad en VHDL. | 39 |
| 4.10. Diagrama de la transformada de Clark y Park en VHDL. | 40 |
| 4.11. Bloque de la transformada de Clark y Park en VHDL. | 40 |
| 4.12. Bloque de la inversa transformada de Clark y Park en VHDL. | 40 |
| 4.13. Bloque de ajustador de 16 bits a 35 bits en VHDL. | 41 |
| 4.14. Bloque de ajustador de 16 bits a 35 bits DE 6500 en VHDL. | 41 |
| 4.15. Bloque de inversor para simulink en PSIM. | 42 |
| 4.16. Bloque de inversor para VHDL en PSIM. | 42 |
| 4.17. Bloque del generador PWM en VHDL. | 42 |
| 4.18. Bloque de la máquina de estado en VHDL. | 43 |
| 4.19. Bloque del flip flop de 3 señales de 16 bits en VHDL. | 43 |
| 4.20. Bloque del flip flop de 1 señal de 16 bits en VHDL. | 43 |
| 4.21. Bloque del flip flop de 6 señales de 1 bits en VHDL. | 44 |
| 5.1. Diagrama de bloques en simulink del PMSM y control por pasividad. | 45 |
| 5.2. Respuesta del PMSM al control por pasividad implementado en simulink con $\gamma_1=10, \gamma_2=3$ para cambio de sentido a 500 rpm. | 46 |

| | | |
|-------|---|----|
| 5.3. | Respuesta del PMSM al control por pasividad implementado en simulink con distintos valores de γ_1 y γ_2 | 47 |
| 5.4. | Respuesta del PMSM al control por pasividad implementado en simulink con distintos valores de γ_1 y γ_2 con escala de tiempo menor. | 47 |
| 5.5. | Respuesta del PMSM al control por pasividad implementado en simulink con $\gamma_1=10, \gamma_2=3$ para cambio de sentido a 1000 rpm. | 48 |
| 5.6. | Respuesta del PMSM al control por pasividad implementado en simulink con $\gamma_1=10, \gamma_2=3$ para varias revoluciones. | 49 |
| 5.7. | Respuesta del PMSM al control por pasividad implementado en simulink con $\gamma_1=10, \gamma_2=3$ y el error. | 49 |
| 5.8. | Respuesta del PMSM al control por pasividad implementado en simulink con $\gamma_1=10, \gamma_2=3$ y el error con menor tiempo de simulación. | 50 |
| 5.9. | Respuesta del PMSM al control por pasividad implementado en simulink con $\gamma_1=10, \gamma_2=3$ y el error en un intervalo de tiempo de simulación. | 50 |
| 5.10. | Diagrama de bloques en simulink del PMSM y control por pasividad en HDL. | 51 |
| 5.11. | Respuesta del PMSM al control por pasividad en HDL. | 52 |
| 5.12. | Respuesta del PMSM al control por pasividad en VHDL con diferentes γ_1 y γ_2 | 52 |
| 5.13. | Respuesta del PMSM al control por pasividad en VHDL con diferentes γ_1 y γ_2 con menor tiempo de simulación. | 53 |
| 5.14. | Respuesta del PMSM al control por pasividad implementado en VHDL con $\gamma_1 = 10$ y $\gamma_2 = 3$ y el error. | 54 |
| 5.15. | Respuesta del PMSM al control por pasividad implementado en VHDL con $\gamma_1 = 10$ y $\gamma_2 = 3$ y el error con menor tiempo de simulación. | 54 |
| 5.16. | Respuesta del PMSM al control por pasividad implementado en VHDL con $\gamma_1 = 10$ y $\gamma_2 = 3$ y el error en un intervalo de tiempo de simulación. | 55 |
| 5.17. | Respuesta del PMSM al control por pasividad implementado en simulink y VHDL con $\gamma_1 = 10$ y $\gamma_2 = 3$ | 55 |
| 5.18. | Error del control por pasividad implementado en simulink y VHDL con $\gamma_1 = 10$ y $\gamma_2 = 3$ | 56 |
| 5.19. | Diagrama de bloques en simulink de PMSM, control por pasividad en simulink e inversor en PSIM. | 57 |
| 5.20. | Respuesta del PMSM al control por pasividad en simulink y con inversor en PSIM a distintas frecuencias de la señal triangular. | 58 |
| 5.21. | Respuesta del PMSM al control por pasividad en simulink y con inversor en PSIM a distintas frecuencias con menor tiempo de cosimulación. | 58 |
| 5.22. | Respuesta del PMSM al control por pasividad en simulink y con inversor en PSIM a una frecuencia de 20 kHz con distintas γ_1 y γ_2 | 59 |
| 5.23. | Respuesta del PMSM al control por pasividad en simulink y con inversor en PSIM a una frecuencia de 20 kHz con distintas γ_1 y γ_2 con menos tiempo de cosimulación. | 60 |
| 5.24. | Diagrama de bloques en simulink del PMSM y control por pasividad en HDL e inversor en PSIM. | 61 |
| 5.25. | Diagrama de bloques en simulink del PMSM y control por pasividad en HDL e inversor en PSIM con diferentes ganancias γ_1 y γ_2 | 62 |
| 5.26. | Diagrama de bloques en simulink del PMSM y control por pasividad en HDL e inversor en PSIM con diferentes ganancias γ_1 y γ_2 con menor tiempo de simulación. | 62 |
| A.1. | Transformacion de Clark para v_α | 74 |
| A.2. | Transformacion de Clark para v_β | 74 |
| A.3. | Transformacion de Park para v_d | 75 |

| | |
|--|----|
| A.4. Transformacion de Park para v_q | 75 |
| A.5. Bloque en Simulink para obtener I_d | 76 |
| A.6. Bloque en Simulink para obtener I_q | 76 |
| A.7. Bloque en Simulink para obtener W_r | 76 |
| A.8. Bloque en Simulink de inversa de Clark y Park para obtener I_a | 77 |
| A.9. Bloque en Simulink de inversa de Clark y Park para obtener I_b | 77 |
| A.10. Bloque en Simulink de inversa de Clark y Park para obtener I_c | 77 |
| A.11. Interior de bloque de PMSM1 en Simulink. | 78 |
| A.12. Bloque en Simulink de PMSM con parámetros incluidos. | 79 |
| | |
| B.1. Respuesta del PMSM al control por pasividad implementado en Simulink con $\gamma_1=1, \gamma_2=1$ para cambio de sentido a 1000 rpm. | 81 |
| B.2. Respuesta del PMSM al control por pasividad implementado en Simulink con $\gamma_1=3, \gamma_2=1$ para cambio de sentido a 1000 rpm. | 82 |
| B.3. Respuesta del PMSM al control por pasividad implementado en Simulink con $\gamma_1=10, \gamma_2=1$ para cambio de sentido a 1000 rpm. | 82 |
| B.4. Respuesta del PMSM al control por pasividad implementado en Simulink con $\gamma_1=1, \gamma_2=3$ para cambio de sentido a 1000 rpm. | 83 |
| B.5. Respuesta del PMSM al control por pasividad implementado en Simulink con $\gamma_1=1, \gamma_2=10$ para cambio de sentido a 1000 rpm. | 83 |
| | |
| C.1. Selección de Set Path... | 86 |
| C.2. Selección de Add Folder... | 86 |
| C.3. Selección de la carpeta Simulink | 87 |
| C.4. Selección de la carpeta PSIM9.0.3_trial | 87 |
| C.5. Active-HDL Toolbox. | 88 |
| C.6. S-function SimCoupler. | 88 |
| | |
| D.1. Respuesta del PMSM al control por pasividad en HDL con $\gamma_1=1$ y $\gamma_2=1$ | 89 |
| D.2. Respuesta del PMSM al control por pasividad en HDL con $\gamma_1=3$ y $\gamma_2=1$ | 90 |
| D.3. Respuesta del PMSM al control por pasividad en HDL con $\gamma_1=10$ y $\gamma_2=1$ | 90 |
| D.4. Respuesta del PMSM al control por pasividad en HDL con $\gamma_1=1$ y $\gamma_2=3$ | 91 |
| D.5. Respuesta del PMSM al control por pasividad en HDL con $\gamma_1=1$ y $\gamma_2=10$ | 91 |
| | |
| E.1. Respuesta del PMSM al control por pasividad en simulink e inversor en PMSM con un voltaje de 84 V y frecuencia de 1 kHz. | 93 |
| E.2. Respuesta del PMSM al control por pasividad en simulink e inversor en PMSM con un voltaje de 84 V y frecuencia de 2 kHz. | 94 |
| E.3. Respuesta del PMSM al control por pasividad en simulink e inversor en PMSM con un voltaje de 84 V y frecuencia de 5 kHz. | 94 |
| E.4. Respuesta del PMSM al control por pasividad en simulink e inversor en PMSM con un voltaje de 84 V y frecuencia de 10 kHz. | 95 |
| E.5. Respuesta del PMSM al control por pasividad en simulink e inversor en PMSM con un voltaje de 84 V y frecuencia de 20 kHz. | 95 |
| E.6. Respuesta del PMSM al control por pasividad en simulink e inversor en PMSM con un voltaje de 84 V y frecuencia de 50 kHz. | 96 |
| | |
| F.1. Respuesta del PMSM al control por pasividad en simulink e inversor en PSIM a 84 V, frecuencia de 20 kHz y $\gamma_1=1$ y $\gamma_2=1$ | 97 |
| F.2. Respuesta del PMSM al control por pasividad en simulink e inversor en PSIM a 84 V, frecuencia de 20 kHz y $\gamma_1=1$ y $\gamma_2=3$ | 98 |

| | | |
|------|--|-----|
| F.3. | Respuesta del PMSM al control por pasividad en simulink e inversor en PSIM a 84 V, frecuencia de 20 kHz y $\gamma_1=1$ y $\gamma_2=10$ | 98 |
| F.4. | Respuesta del PMSM al control por pasividad en simulink e inversor en PSIM a 84 V, frecuencia de 20 kHz y $\gamma_1=3$ y $\gamma_2=1$ | 99 |
| F.5. | Respuesta del PMSM al control por pasividad en simulink e inversor en PSIM a 84 V, frecuencia de 20 kHz y $\gamma_1=10$ y $\gamma_2=1$ | 99 |
| G.1. | Respuesta del PMSM al control por pasividad en VHDL e inversor en PSIM a 84 V, frecuencia de 20 kHz y $\gamma_1=1$ y $\gamma_2=1$ | 101 |
| G.2. | Respuesta del PMSM al control por pasividad en VHDL e inversor en PSIM a 84 V, frecuencia de 20 kHz y $\gamma_1=1$ y $\gamma_2=3$ | 102 |
| G.3. | Respuesta del PMSM al control por pasividad en VHDL e inversor en PSIM a 84 V, frecuencia de 20 kHz y $\gamma_1=1$ y $\gamma_2=10$ | 102 |
| G.4. | Respuesta del PMSM al control por pasividad en VHDL e inversor en PSIM a 84 V, frecuencia de 20 kHz y $\gamma_1=3$ y $\gamma_2=1$ | 103 |
| G.5. | Respuesta del PMSM al control por pasividad en VHDL e inversor en PSIM a 84 V, frecuencia de 20 kHz y $\gamma_1=10$ y $\gamma_2=1$ | 103 |

Índice de cuadros

| | |
|--|----|
| 2.1. Características de distintos imanes pemanentes. | 9 |
| 2.2. Parámetros de PMSM. | 15 |
| 4.1. Representación de punto flotante en la palabra binaria. | 25 |
| 4.2. Diferentes rangos para una palabra de N=8 bits. | 26 |
| 4.3. Multiplicación con diferentes formatos en binario con punto fijo. | 27 |
| 4.4. Magnitudes del formato L(15.20). | 31 |
| 4.5. G2: Valor real en bits y en punto fijo con formato L(15.20). | 35 |
| 4.6. CONS3; Valor real en bits y en punto fijo con formato L(15.20). | 36 |
| 5.1. Clasificación de las respuesta del PMSM al control por pasividad implementada en simulink. | 48 |
| 5.2. Clasificación de las respuesta del PMSM al control por pasividad implementada en Active DHL. | 53 |
| 5.3. Clasificación de las respuesta del PMSM al control por pasividad implementada en simulink con inversor en PSIM. | 59 |
| 5.4. Clasificación de las respuesta del PMSM al control por pasividad implementada en VHDL con inversor en PSIM. | 63 |
| 5.5. Resumen de uso FPGA NEXYS 2 al instalar el control por pasividad del PMSM. | 63 |
| 5.6. Resumen de uso FPGA VIRTEX 5 al instalar el control por pasividad del PMSM. | 64 |

Capítulo 1

Antecedentes

1.1. Introducción

En la actualidad la aplicación de la tecnología para la conservación del medio ambiente está en aumento, siendo un ejemplo la industria automotriz. Varias de las armadoras importantes a nivel mundial han iniciado la introducción de vehículos propulsados por motores híbridos y por motores eléctricos; Chevrolet tiene en este año en el mercado su automóvil **Spark** con su motor eléctrico de imanes permanentes [ChevS2015] y con autonomía extendida tiene al **Volt** [ChevV2015]; así mismo Cadillac, con su automóvil **ELR**, presenta un coupé de lujo de autonomía extendida, éste contiene bajo su capo un motor eléctrico y un motor de combustión interna que actuará como generador, éste con el fin de proporcionar energía eléctrica cuando las baterías estén bajas y así permitir varios cientos de kilómetros de conducción adicional [CadillacE2016]. La prestigiada armadora de motores como es McLaren surtirá de motores eléctricos a la **Formula E**, un campeonato aprobado por la FIA.

Uno de los motores eléctricos empleados en la industria automotriz es Motor Síncrono de Imanes Permanentes (PMSM por sus siglas en inglés) de corriente alterna (CA) [Ying2011] [Burgos2006], este motor tiene la ventaja de requerir menor mantenimiento comparado con un motor eléctrico con escobillas; el inconveniente de los motores PMSM de AC es que su dinámica es no lineal. Con los avances de las teorías de control no lineal, electrónica de potencia y los microprocesadores, es posible el control de velocidad y par de los motores PMSM de CA y ser una oferta viable y competitiva en la industria [Dufour2006].

1.2. Objetivos

1.2.1. Objetivo general

1. Diseñar el control por pasividad de la velocidad de un motor trifásico de imanes permanente y programar éste en VHDL.

1.2.2. Objetivos particulares

1. Sintonizar el controlador por pasividad de la velocidad de un motor trifásico de imanes permanentes.
2. Simular el control por pasividad de la velocidad de un motor trifásico de imanes permanentes en la plataforma Simulink.
3. Implementar el control por pasividad de la velocidad de un motor trifásico de imanes permanentes en VHDL.
4. Cosimular el control por pasividad de la velocidad de un motor trifásico de imanes permanentes implementado en VHDL con Simulink y Aldec Active-HDL Student Edition.

1.3. Justificación

La tendencia actual es el empleo de energías alternativas en diversos tipos de aplicaciones. Una de las áreas donde se está trabajando de manera intensa es en la creación de autos eléctricos por diversas compañías entre las que destacan Nissan, Volkswagen, Ford, Chevrolet, Cadillac, Honda y Minicooper entre otras [Llor2003] [CadillacE2016].

Diversos trabajos muestran que uno de los motores más eficientes es el de imanes permanentes, razón por lo cual es empleando en automóviles [SwKa2002] [ChevS2015] [ChevV2015]. En este trabajo se desarrollará un controlador aplicando la técnica de control por pasividad. El motor utilizado es un sistema MIMO (múltiples entradas múltiples salidas, por sus siglas en inglés) ya que requiere tres entradas y dos salidas y el método propuesto es el indicado para controlar el sistema [Cecati1999] [Duarte2003].

Disponiendo de las características actuales de los sistemas digitales; el control de campo orientado (FOC, por sus siglas en inglés) y control directo de par (DTC, por sus siglas en inglés) han sido implementados en DSP (procesador digital de señal, por sus siglas en inglés) con una frecuencia de conmutación de hasta 20 kHz, no es suficiente para trabajar con enfoque análogo; para dar solución a este problema, una opción es FPGA (arreglo de compuertas programables en campo, por sus siglas en inglés) que tiene como características el hacer varios procesos en paralelo y con esto disminuir el tiempo de muestreo; además de con programas de simulación se le permite al programador mejorar el rendimiento del sistema y disminuir errores [Sandre2014].

El lenguaje para programar los FPGA es VHDL (Lenguaje de descripción de hardware de alta velocidad, por sus siglas en inglés) en el que se programa el control basado en pasividad, permitiendo crear un sistema integrado que permitirá el control del motor de imanes permanentes en un sólo dispositivo [Sandre2014] [Dufour2006].

Al realizar la revisión bibliográfica no se encontró trabajo reportado como el que se desarrolló en esta tesis; y además que el código en VHDL es de arquitectura abierta y no depende del hardware y es rediseñable para mejorar su rendimiento [Sandre2014]; además que el control basado en pasividad es robusto [Nicklasson1997].

1.4. Alcances y limitaciones

En este trabajo se presenta la implementación en VHDL del control por pasividad de un motor eléctrico de imanes permanentes, usando el modelo matemático ya desarrollado por diferentes investigadores; el control propuesto es simulado en Matlab en su subprograma Simulink; y se generó el código en alto nivel para VHDL, el cual se simula.

Por lo anterior, los entregables de este trabajo serán:

- Simulación del Control por Pasividad de un motor eléctrico de imanes permanentes en *Matlab^R* en el subprograma *Simulink^R*
- Código en VHDL del Control por Pasividad de un motor eléctrico de imanes permanentes
- Cosimulación del Control por Pasividad de un motor eléctrico de imanes permanentes, implementando el control en VHDL, el modelo PMSM en Simulink e inversor en Psim.

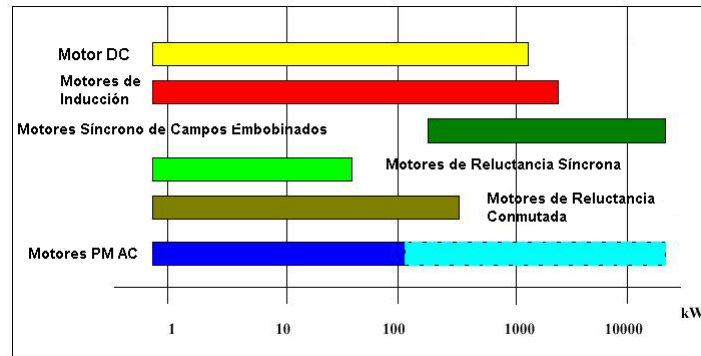


Figura 1.1: Motores empleados en accionamientos de velocidad variable.

1.5. Estado del arte.

El uso de motores eléctricos de velocidad variable ha evolucionado tanto que en la actualidad se tiene una amplia gama de soluciones, las aplicaciones que antes eran exclusivas para motores de DC en la actualidad la comparten con los motores PMSM; estos últimos son utilizados en diferentes aplicaciones como son: robots industriales, máquinas herramientas, ascensores, servomotores, vehículos eléctricos; donde la velocidad y par son variables [Zhong1997] [Gokdere1997].

Las principales características que presentan los motores PMSM son su alta relación potencia-peso, alta relación par-inercia, alta eficiencia, velocidad rápida de respuesta de par y los altos niveles de potencia que manejan [SwKa2002] y también la relación precio/prestaciones [Llor2003].

De la figura 1.1 se muestra la clasificación de los tipos de motores aplicados en diferentes accionamientos, en base a la potencia del motor y el tipo de motor [Llor2003], y se aprecia que los motores de imanes permanentes (PMSM) cubren el mayor rango de potencia.

Uno de los campos de aplicación de los PMSM es la tracción eléctrica, para un extenso rango de vehículos [Llor2003]. Aunado a esto; la preocupación por el cuidado del medio ambiente, motiva a considerar la viabilidad del cambio de motores de combustión interna a motores eléctricos en los vehículos. Uno de los estudios realizados indica que en el peor de los casos, una generación de energía eléctrica que emplea solamente carbón, generará emisiones contaminantes mucho menores que las que emite los vehículos que trabajan a gasolina, además que las plantas de electricidad están lejos de las zonas urbanas [California1989].

Es importante reiterar que los PMSM tienen una dinámica no lineal, por lo que requiere de la aplicación de las teorías de control de la misma dinámica. Por esto último, es necesario representar el motor síncrono de imanes permanentes mediante el modelo que lo describe. Y después aplicar la teoría de control por **pasividad**; las ecuaciones obtenidas para el control son: simples, robustas y garantizan la estabilidad global del sistema [Cecati1999].

Con el fin de tener los más recientes avances en conocimiento y en las herramientas usadas para el control de motores de imanes permanentes, se realizó una revisión de trabajos afines que han elaborado diferentes investigadores; y con base en el análisis del estado del arte, hasta el año 2013, se puede concluir que no hay trabajos como el desarrollado en esta tesis. De esta revisión se han encontrado los siguientes aportes:

1.5.1. Estado del arte del control de motores de imanes permanentes.

Se han realizado investigaciones en la aplicación de teorías de control en motores síncronos, de los cuales destacan los siguientes:

El control directo de par (DTC, por su siglas en inglés) tiene como característica principal el que a partir de referencias deseadas como es el par electromagnético y el flujo estatórico; se genera un estado de conmutación del inversor; este control lo desarrolla Bossoufi et al [Bossoufi2011].

Otra estrategia para diseñar el controlador del sistema es el analizar el conjunto convertidor-motor, con lo que se logra seleccionar el estado de conmutación y su duración, esta aproximación es conocida como *Control Híbrido* [Llor2003].

Un método también aplicado para el control de motores síncronos de imanes permanentes (PMSM) es el control de campo orientado (FOC, por sus siglas en inglés), que controla la velocidad del motor [Vindel2011].

El modelado recomendable para el diseño del control basado en pasividad es el obtenido aplicando la formulación basada en funciones de energía (metodología Euler-Lagrange) [Duran2004].

Una herramienta utilizada actualmente para analizar la actuación del control propuesto, sin realizar la implementación física, es la simulación del sistema en programas mas especializados; el uso de esta herramienta esta ayudando a disminuir costos. En el caso particular del control de motores de imanes permanentes, se han realizado simulaciones en programa SIMNON del control directo de par y por pasividad [Ramirez2011]. También se ha publicado, el control de velocidad por pasividad de un motor de imanes permanentes con su respectiva simulación; demostrando la estabilidad del control aplicado y el desempeño en la simulación en el programa SIMNON. [Ramirez2011].

Badre Bossoufi presenta la simulación del control directo de par implementado en un FPGA (arreglo de compuertas programables en campo, por su siglas en inglés) [Bossoufi2011], y su implementación del control directo de torque de un PMSM en un FPGA [Bossoufi2011a]; utilizando la ventaja de estos dispositivos, que es hacer los procesos en paralelo, para realizar las operaciones de algoritmos complejos en periodos de tiempo cortos. En esta aplicación se utiliza el concepto de modular y reusable; además de usar *Simulink System Generator* para los bloques de Active HDL, y con esto generar el código para implementarlo en el FPGA, demostrando que la implementación del control esta bien realizada en este dispositivo y se comporta como era esperado .

Lo siguiente a realizar en la investigación del control por pasividad de un motor de imanes permanentes es la implementación de este control en dispositivos físicos, uno de estos dispositivos es el FPGA; y antes de aplicarlo al motor físicamente se debe de realizar una simulación; que es el objetivo de este trabajo .

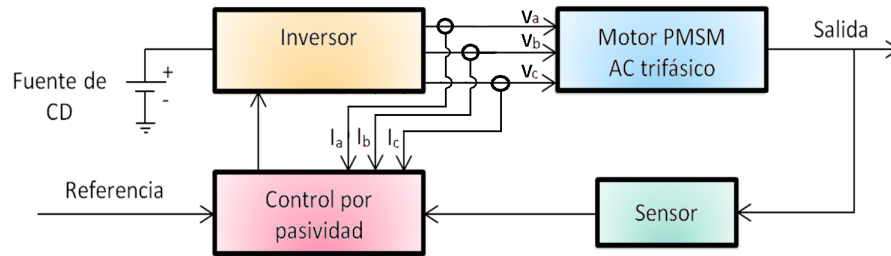


Figura 1.2: Diagrama a bloques de un sistema de control a lazo cerrado de un motor PMSM de AC trifásico.

1.6. Organización del trabajo

En la figura 1.2 se muestra un diagrama a bloques del sistema de control de lazo cerrado de un motor eléctrico de imanes permanentes de corriente alterna, de acuerdo con el diagrama se diseñará el controlador; cada bloque se desarrolla en las secciones siguientes, exceptuando el sensor.

La forma en que se define la estructura de este trabajo de tesis es la siguiente:

Capítulo 1. Contiene el estado del arte que consiste en una antología de diferentes fuentes bibliográficas que contiene información relacionada con el control de motores eléctricos de imanes permanentes (PMSM).

Capítulo 2. Se describe el modelo matemático que se ha obtenido por diferentes investigadores del motor eléctrico de imanes permanentes(PMSM).

Capítulo 3. En este capítulo se puntualiza el control por pasividad y se aplica al modelo del motor eléctrico de imanes permanentes(PMSM), además de realizar la simulación del control propuesto.

Capítulo 4. En este capítulo se realiza la generación de los códigos de alto nivel necesarios para implementarlos en el FPGA y se harán las simulaciones.

Capítulo 5. Se presenta los resultados obtenidos de las simulaciones realizadas, además de las conclusiones y los posibles investigaciones futuras.

Capítulo 2

Modelado del motor eléctrico de imanes permanentes (PMSM).

En este capítulo se plantea el modelo matemático del PMSM, que varios autores presentan, y se realiza la representación de éste en su forma de ecuaciones diferenciales; para con ellas en el siguiente capítulo hacer el tratamiento matemático para obtener la ley de control por pasividad. Además, en la segunda sección de este capítulo se crean los bloques en simulink de cada una de estas ecuaciones, con las operaciones que implican cada una. Con el fin de obtener en simulink la representación de motor de imanes permanentes.

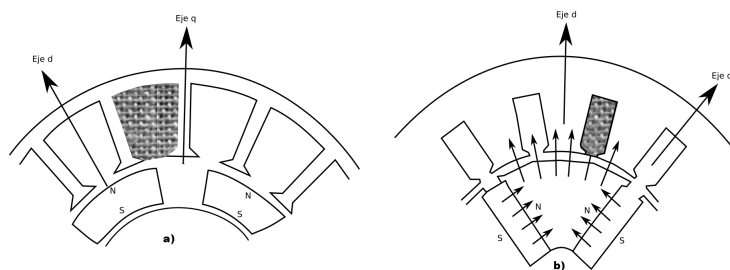


Figura 2.1: Configuración básica de PMSM. a)PMSM de imanes superficiales, b)PMSM de imanes interiores.

2.1. Motor eléctrico síncrono de imanes permanentes (PMSM) de corriente alterna trifásico.

La integración de imanes permanentes de alta energía y la evolución de las herramientas de diseño asistido (CAD), han hecho que los motores síncronos de imanes permanentes (PMSM) amplíen sus aplicaciones en el mercado. Los motores síncronos tienen la característica principal de que su velocidad es proporcional a la frecuencia del voltaje aplicado a su estator. Además que la velocidad con que gira el rotor es igual a la del campo magnético giratorio generado por el estator, de aquí que son llamados síncronos [Fitzgerald2003].

Una de las principales características de los PMSM es que el rotor no posee bobinas de excitación, las que son reemplazadas por imanes permanentes. Además los motores de imanes permanentes tienen una alta eficiencia, alcanzando 95%. Y también la razón de potencia-peso es mayor que otros motores eléctricos. Otra ventaja que tienen estos motores es que son menos susceptibles a sufrir recalentamientos, debido a que las bobinas están enrolladas en el estator y hace que su refrigeración sea simple; los motores de baja potencia son enfriados por aire [Rodríguez2002].

A continuación se analizará las características de construcción y de operación de los PMSM.

2.1.1. Configuración básica de los motores síncronos de imanes permanentes.

En el mercado se tienen dos configuraciones básicas de estos motores, estas están en función de la disposición de los imanes permanentes en el rotor [Llor2003]: montaje superficial e imanes interiores. Los cuales se esquematizan en la figura 2.1.

Se representa en la figura 2.1 un PMSM. a) es un motor de imanes permanentes superficiales de 4 polos y el espacio interpolar está relleno de material no-ferromagnético, y en b) tiene como característica tener incrustado en el interior del rotor ferromagnético los imanes permanentes. Con esta arquitectura se obtiene una robustez mecánica, necesaria para aplicaciones de altas velocidades; además de producir una reluctancia en la dirección del eje en cuadratura con el flujo de los imanes haciendo que sea menor que en el eje directo, logrando así ser un componente de par reluctante [Llor2003].

A continuación hablaremos de los imanes permanentes, sus características y porque han hecho que evolucionen los motores eléctricos de imanes permanentes.

Tabla 2.1: Características de distintos imanes permanentes.

| Familia | Tc [°C] | Br [kG] | Hc [kOe] | Bhmax [MGoe] | Temp. Oper. [°C] |
|-----------------|-------------------|-------------------|--------------------|------------------------|----------------------------|
| Alnicos | | | | | |
| Alnico 2 | 700-850 | 7.2 | 0.54 | 1.6 | 300 |
| Alnico 5 | 700-850 | 13.5 | 0.75 | 7.5 | 300 |
| Alnico 8 | 700-850 | 7.5 | 1.9 | 5 | 300 |
| Ferritas | | | | | |
| Ceramic 8 | 450 | 3.8 | 2.9 | 4 | 150 |
| REPMs | | | | | |
| 1-5 | 650 | 5-10 | 5-25 | 6-26 | 250 |
| 2-17 | 800 | 9-12 | 6-25 | 20-31 | 350 |
| 2-14-1 | 350 | 8-12 | 8-24 | 12-14 | 150 |

2.1.1.1. Imanes permanentes.

Un componente que ha hecho que los motores de imanes permanentes tengan un incremento importante es la aplicación de materiales con elevado nivel de magnetismo remanente. Los primeros materiales usados para los imanes eran de ferrita, tienen bajo costo y excelente liberalidad en la desmagnetización; pero el bajo magnetismo remanente limitaba sus aplicaciones. En la actualidad, los imanes son fabricados con tierras raras; como el de Cobalto-Samario ($SmCo_5$ ó Sm_2Co_{17}) o de Neodimio-Boro ($NdFeB$). El imán de Cobalto-Samario tiene la mejor combinación de características sólo que es caro. Los magnetos fabricados con Neodimio tienen un magnetismo remanente alto y una linealidad en la curva de desmagnetización; pero con el inconveniente que la intensidad de campo decrece con el aumento de la temperatura [Llor2003].

Profundizando un poco más, a continuación se definen algunas características esenciales. El primero, la densidad de flujo remanente **Br**, que es la medida de la fuerza de atracción o repulsión que ejerce el imán; éste es medido en **Tesla** o **kiloGauss** (1 Tesla = 10 kGauss). La segunda característica de los imanes permanentes es la **coercitividad**, es la capacidad de resistir la desmagnetización, su medida es la fuerza coercitiva **Hc** y sus unidad de medida son los kiloOersted. La siguiente característica es la **temperatura de Curie TC**, es aquella donde el imán pierde sus propiedades magnéticas. Por último, está el producto de energía máximo **BHmax**, que esta representado por el rectángulo más grande que puede inscribirse en el cuadrante B+ H- de la curva de magnetización del imán [Rodríguez2002].

En la Tabla 2.1 se muestran los valores característicos de algunas familias de imanes permanentes; en esta se destacan los "Alnicos", las "Ferritas" y los "REPMs". Los "Alnicos" (aleación de Aluminio, Níquel y Cobalto), estos tiene baja coercitividad. Las "Ferritas" (materiales cerámicos) tienen valores de coercitividad y remanencia medios, la aplicación de estos es en motores industriales por su bajo costo, sólo que son de peso excesivo. Y los "REPMs" (Rare Earth Permanet Magnet) ó imanes de "Tierras Raras", son los aplicados a vehículos eléctricos; estos son magnéticamente fuertes y permiten la fabricación de motores livianos; dentro de este tipo de imanes, se clasifican en dos familias: los en base a Samario y los en base a Neodimio. Los imanes de Samario-Cobalto más importantes son del tipo "1-5" y "2-17"; y los de Neodimio-Fierro-Boro lo conforma el tipo "2-14-1". Estos números son conformados en relación al peso de sus elementos[Rodríguez2002].

El motor de imanes permanentes alimentado con CA trifásico genera un campo magnético giratorio de magnitud constante [Kundur1994]. La velocidad del campo magnético giratorio está dado por la ecuación 2.1 [Duran2004].

$$\eta_s = \frac{120f}{P} \quad (2.1)$$

donde:

f = Frecuencia eléctrica (Hz)

η_s = Velocidad del campo magnético (rpm)

P = Número de polos

En estado estacionario gira a esta misma velocidad el rotor del motor [Duran2004].

También podemos interpretar de la ecuación 2.1, que la velocidad depende de la frecuencia del voltaje aplicado a los devanados del estator y del número de polos, y no de la carga mecánica que es aplicada al motor [Duran2004].

Por construcción, los motores de imanes permanentes tienen fijo el número de polos; por lo que, si se mantiene la misma frecuencia del voltaje de la armadura, la velocidad del motor se mantendrá constante en cualquier carga hasta la potencia máxima [Duran2004].

Los motores síncronos de bajo número de pares de polos son de altas velocidades y los de mayor número de pares de polos son para velocidades bajas [Duran2004].

La relación que existe entre el desplazamiento angular eléctrico θ_e y el desplazamiento angular mecánico θ_m , está expresado en la ecuación 2.2 [Duran2004].

$$\theta_r = P\theta_m \quad (2.2)$$

donde:

θ_r = Desplazamiento angular eléctrico del motor.

θ_m = Desplazamiento angular mecánico del motor.

P = Número de pares de polos.

La ecuación 2.2 es importante debido a que generalmente el valor que se puede medir es el desplazamiento mecánico.

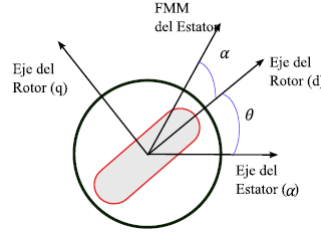


Figura 2.2: Representación de motor PMSM en los ejes dq .

2.2. Modelado del motor eléctrico trifásico de imanes permanentes (PMSM).

El modelo del motor eléctrico de imanes permanentes referenciado en un sistema trifásico estático es complicado; Sólo que, aplicando la transformación de Clark a un sistema de referencia de dos ejes, se simplifica [Ramirez2011sp] y se obtiene la ecuación (2.3). Las dos transformaciones usadas son α y β , y estos son ortogonales entre si.

En la figura 2.2 se muestra la relación de ejes en dq ; de donde se obtiene la transformación de Park asociada al rotor; es un sistema de referencia de dos ejes que giran a la velocidad del rotor, y se obtiene la ecuación que se muestra en (2.4) [Ramirez2011sp].

$$\begin{bmatrix} v_\alpha \\ v_\beta \end{bmatrix} = \frac{2}{3} \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} v_a \\ v_b \\ v_c \end{bmatrix} \quad (2.3)$$

$$\begin{bmatrix} v_d \\ v_q \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} v_\alpha \\ v_\beta \end{bmatrix} \quad (2.4)$$

Donde:

v_d y v_q son voltajes en el eje de referencia dq [V].

v_a , v_b y v_c son voltajes de la línea trifásica [V].

θ desplazamiento angular de los ejes del estator y del rotor [rad].

El modelo del motor eléctrico de imanes permanentes (PMSM) en el sistema de referencia dq esta expresado en las dos ecuaciones diferenciales (2.5) y (2.6), que a continuación se presentan.

$$L_d \frac{di_d}{dt} = \omega_r L_q i_q - R i_d + v_d \quad (2.5)$$

$$L_q \frac{di_q}{dt} = -\omega_r (L_d i_d + \Psi_{PM}) - R i_q + v_q \quad (2.6)$$

Donde:

v_d y v_q son voltajes en el eje de referencia dq [V].

i_d y i_q son corrientes en el eje de referencia dq [A].

v_a , v_b y v_c son voltajes de la línea trifásica [V].

L_d y L_q son la inductancias [H].

R es la resistencia del estator [Ω].

Ψ_{PM} es la magnitud del flujo de los imanes permanentes [Wb].

ω_r es la velocidad eléctrica del motor [rpm].

En cuanto al modelado de la parte mecánica del motor, la ecuación (2.7) lo representa. La carga eléctrica está representado por la ecuación (2.8); para el caso de máximo torque se cumple $L_d = L_q = L_0$, y el par de carga queda sólo en función de la corriente, como se muestra en la ecuación (2.9) [Ramirez2011sp].

$$\frac{J}{P} \frac{d\omega_r}{dt} = \tau_e - T_L - T_f \quad (2.7)$$

$$\tau_e = \frac{3}{2} P [i_d i_q (L_d - L_q) + i_q \Psi_{PM}] \quad (2.8)$$

$$\tau_e = \frac{3}{2} P \Psi_{PM} i_q \quad (2.9)$$

Donde:

τ_e = El par eléctrico del motor [Nm].

T_L = El par de carga [Nm]

T_f = El par de fricción [Nm].

J = El momento de inercia [kgm²].

P = El número de par de polos.

Sustituyendo la ecuación (2.9) en (2.7), y conociendo $\omega_r = P\omega_m$, la relación entre la velocidad eléctrica y mecánica del motor; se obtiene la ecuación (2.10).

$$\frac{2}{3} \frac{J}{P^2} \frac{d\omega_r}{dt} = \Psi_{PM} i_q - \frac{2}{3P} (T_L + \frac{B\omega_r}{P}) \quad (2.10)$$

Con las ecuaciones (2.5), (2.6) y (2.10); el sistema se representa en variables de estado, y se obtiene el modelo 2.11.

$$\begin{bmatrix} L_d \frac{di_d}{dt} \\ L_q \frac{di_q}{dt} \\ \frac{2}{3} \frac{J}{P^2} \frac{d\omega_r}{dt} \end{bmatrix} = \begin{bmatrix} \omega_r L_q i_q - R i_d + v_d \\ -\omega_r (L_d i_d + \Psi_{PM}) - R i_q + v_q \\ \Psi_{PM} i_q - \frac{2}{3P} (T_L + \frac{B\omega_r}{P}) \end{bmatrix} \quad (2.11)$$

Donde:

v_d y v_q son voltajes en el eje de referencia dq [V]

i_d y i_q son corrientes en el eje de referencia dq [A]

L_d y L_q son la inductancias [H].

R es la resistencia del estator [Ω].

Ψ_{PM} es la magnitud del flujo de los imanes permanentes[Wb].

ω_r es la velocidad eléctrica del motor [$\frac{rad}{s}$].

T_L es el par de carga [Nm].

J es el momento de inercia [Nm].

B es la fricción viscosa [Ns]

P es el número de par de polos.

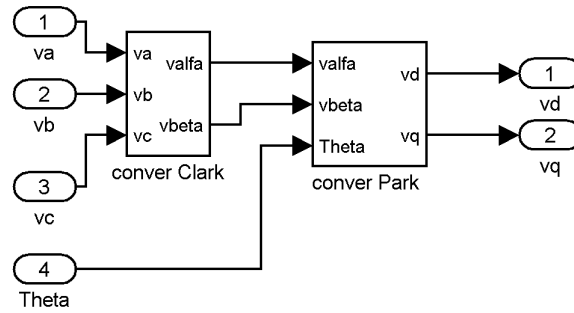


Figura 2.3: Transformada de Clark y Park en simulink

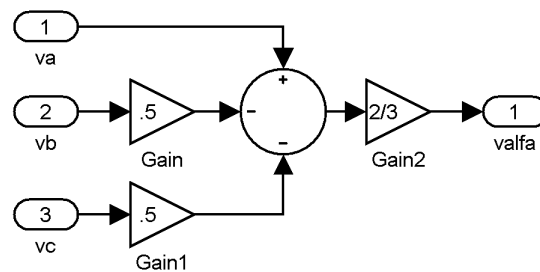


Figura 2.4: Conversión Clark en simulink

2.3. Implementación del modelo del PMSM en simulink.

En la actualidad se cuenta con diferentes herramientas para realizar diseños de elementos, equipos, entre otros; una de las herramientas que se ha desarrollado para realizar la simulación de sistemas es Matlab, que tiene integrado simulink.

Simulink es un software que tiene prestaciones tales que en bloques se pueden representar los modelos matemáticos que se han obtenido de sistemas específicos.

En la figura 2.3 se muestra la primera de las relaciones matemáticas que se requiere implementar en simulink es la transformación de Clark y Park, (ec. 2.3), donde las entradas son los voltajes v_a, v_b, v_c y θ y las salidas son v_d y v_q .

De la figura 2.4 se observa como se obtiene v_{alfa} , representado en la ecuación 2.3; este está dentro del bloque *conver Clark*.

Para representar el modelo del motor se emplea la ecuación 2.11, donde las entradas son $v_a, v_b, v_c, \psi_{PM}, P, TL, B, L_d, L_q, R$ y J y las salidas I_a, I_b, I_c y W_r . El resultado se muestra en la figura 2.5.

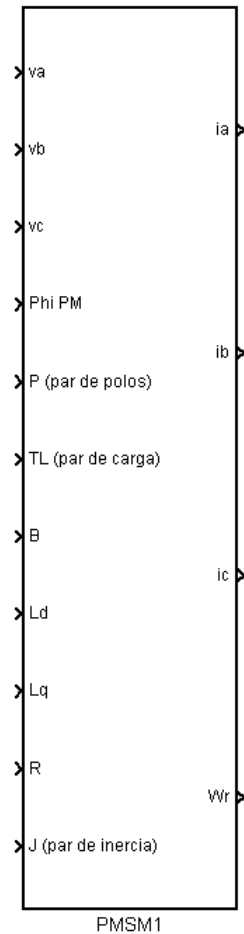


Figura 2.5: Bloque en simulink de PMSM

Tabla 2.2: Parámetros de PMSM.

| Parámetro | Valor |
|--|---------------------------|
| Voltaje de entrada VDC | 150 V |
| Par de polos | 3 |
| flujo permanente magnético ψ_{PM} | 0.2547 Wb |
| Constante de inercia J | 0.00141 kg/m ² |
| Resistencia R | 0.86 Ω |
| Inductancia Ld | 0.0065 H |
| Inductancia Lq | 0.0065 H |
| Fricción viscosa B | 0.00038 Ns |

En la tabla 2.2 se muestran los parámetros que se utilizarán para el PMSM en este proyecto, estos son los valores que tiene cada motor por construcción; estos son agregados al bloque del motor en simulink y se obtiene el bloque del PMSM que se muestra en la figura 2.6 , y solo tiene las entradas de v_a , v_b , v_c y con las salidas I_a , I_b , I_c y W_r .

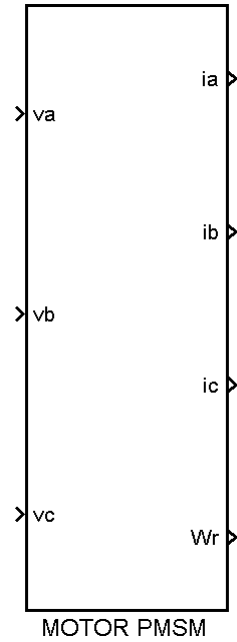


Figura 2.6: Bloque en simulink de PMSM con parámetros incluidos.

La implementación detallada del PMSM en simulink se describe en el anexo A.

Capítulo 3

Diseño del control por pasividad de un motor eléctrico de imanes permanentes (PMSM).

La obtención de la ley de control por pasividad se realiza en este capítulo, utilizando el modelo matemático del PMSM obtenido en el capítulo anteriormente. En la última sección, en simulink se implementa las ecuaciones que representan la ley de control, con el fin de tener un bloque que contenga la ley de control y se pueda implementar para la simulación de la respuesta de éste.

3.1. Control por pasividad.

El control basado en pasividad emplea el moldeo de la energía del sistema e inyecta el amortiguamiento requerido para lograr el objetivo de control [Duran2004].

Aplicando el control por pasividad en un sistema, se logra obtener un punto de equilibrio. Adicional a esto, se inyecta amortiguamiento al sistema para que sea estable y aumente la velocidad de convergencia [Duran2004].

3.2. Diseño del control por pasividad PMSM.

Para la aplicación de pasividad es necesario expresar el sistema de la forma de la ecuación 3.1 [Ortega1998]. Las variables de estados son i_d , i_q y ω_r .

$$\mathbf{A}\dot{\mathbf{X}} = \mathbf{J}(\omega_r)\mathbf{X} - \mathbf{R}\mathbf{X} + \mathbf{B}\mathbf{u} + \varepsilon \quad (3.1)$$

Realizando las conversiones necesarias de la ecuación 2.11 obtenemos:

$$\begin{aligned} \begin{bmatrix} L_d & 0 & 0 \\ 0 & L_q & 0 \\ 0 & 0 & \frac{2}{3} \frac{J}{P^2} \end{bmatrix} \begin{bmatrix} \frac{di_d}{dt} \\ \frac{di_q}{dt} \\ \frac{d\omega_r}{dt} \end{bmatrix} &= \begin{bmatrix} 0 & \omega_r L_q & 0 \\ \omega_r L_d & 0 & -\Psi_{PM} \\ 0 & \Psi_{PM} & 0 \end{bmatrix} \begin{bmatrix} i_d \\ i_q \\ \omega_r \end{bmatrix} \\ &- \begin{bmatrix} R & 0 & 0 \\ 0 & R & 0 \\ 0 & 0 & \frac{2B}{3P^2} \end{bmatrix} \begin{bmatrix} i_d \\ i_q \\ \omega_r \end{bmatrix} \\ &+ \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} v_d \\ v_q \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -\frac{2}{3P} \end{bmatrix} T_L \end{aligned} \quad (3.2)$$

Para que el sistema sea controlable por pasividad, se deben cumplir las siguientes condiciones

$$\mathbf{J}^T(\omega_r) = -\mathbf{J}(\omega_r) \quad (3.3)$$

$$\mathbf{R}^T(u) = \mathbf{R}(u) \quad (3.4)$$

Esto se cumple si $L_d = L_q$ y como la Matriz \mathbf{R} es una matriz diagonal, su transpuesta es la misma matriz.

Para aplicar la técnica de pasividad es necesario crear una copia del sistema y se define como:

$$\mathbf{A}\dot{\mathbf{X}}^* = \mathbf{J}(\omega_r^*)\mathbf{X}^* - \mathbf{R}\mathbf{X}^* + \mathbf{B}\mathbf{u}^* + \varepsilon \quad (3.5)$$

Restando la ecuación 3.5 a la 3.1, nos queda:

$$\mathbf{A}\dot{\mathbf{X}} - \mathbf{A}\dot{\mathbf{X}}^* = \mathbf{J}(\omega)\mathbf{X} - \mathbf{R}\mathbf{X} + \mathbf{B}\mathbf{u} + \varepsilon - \mathbf{J}(\omega_r^*)\mathbf{X}^* + \mathbf{R}\mathbf{X}^* - \mathbf{B}\mathbf{u}^* - \varepsilon \quad (3.6)$$

$$\mathbf{A}\dot{\mathbf{e}} = \mathbf{J}(\omega)\mathbf{X} - \mathbf{R}\mathbf{e} + \mathbf{B}\mathbf{e}_u - \mathbf{J}(\omega_r^*)\mathbf{X}^* \quad (3.7)$$

Donde:

$$\mathbf{e} = \mathbf{X} - \mathbf{X}^*$$

$$\mathbf{e}_u = \mathbf{u} - \mathbf{u}^*$$

Realizando operaciones algebraicas a la ecuación 3.7

$$\mathbf{A}\dot{\mathbf{e}} = \mathbf{J}(\omega)\mathbf{X} - \mathbf{J}(\omega_r^*)\mathbf{X}^* - \mathbf{R}\mathbf{e} + \mathbf{B}\mathbf{e}_u \quad (3.8)$$

Aplicando linealización a $\mathbf{J}(\omega)$ por series de Taylor, alrededor de ω_r^* , se obtiene:

$$\mathbf{A}\dot{\mathbf{e}} = \frac{\partial \mathbf{J}}{\partial dt} \Big|_{\omega_r^*} \mathbf{e}_{\omega_r} \mathbf{X} - \mathbf{R}\mathbf{e} + \mathbf{B}\mathbf{e}_u + \mathbf{J}(\omega_r^*)\mathbf{e} \quad (3.9)$$

Para demostrar que el sistema es estable, que esta representada por 3.9; se va a usar el segundo principio de Lyapunov. Para ello se propone una función candidata de Lyapunov, la cual es definida positiva.

$$\mathbf{V}(\mathbf{e}) = \frac{1}{2} \mathbf{e}^T \mathbf{A}\mathbf{e} \quad (3.10)$$

Es definida positiva, ya que la matriz \mathbf{A} es diagonal y sus coeficientes son mayores a cero. Derivando con respecto al tiempo se obtiene la ecuación 3.11

$$\dot{\mathbf{V}}(\mathbf{e}) = \mathbf{e}^T \mathbf{A}\dot{\mathbf{e}} = \mathbf{e}^T \left[\frac{\partial \mathbf{J}}{\partial dt} \Big|_{\omega_r^*} \mathbf{e}_{\omega_r} \mathbf{X} - \mathbf{R}\mathbf{e} + \mathbf{B}\mathbf{e}_u + \mathbf{J}(\omega_r^*)\mathbf{e} \right] \quad (3.11)$$

Para la ecuación 3.11 se selecciona a \mathbf{e}_u como:

$$\mathbf{e}_u = -\gamma \mathbf{B}^T \mathbf{e} - \mathbf{B}^T \frac{\partial \mathbf{J}}{\partial dt} \Big|_{\omega_r^*} \mathbf{e}_{\omega_r} \mathbf{X} \quad (3.12)$$

sustituyendo 3.12 en 3.11 se obtiene:

$$\dot{\mathbf{V}}(\mathbf{e}) = -\mathbf{e}^T [\mathbf{R} + \mathbf{B}\gamma \mathbf{B}^T] \mathbf{e} \quad (3.13)$$

Para que $\dot{\mathbf{V}}$ sea definida negativa es necesario que se cumpla la siguiente desigualdad $[\mathbf{R} + \mathbf{B}\gamma \mathbf{B}^T] \geq 0$. La condición anterior se satisface si los elementos de la matriz diagonal γ son mayores a cero. Con esta consideración \mathbf{e}_u queda de la siguiente forma:

$$\mathbf{e}_u = - \begin{bmatrix} \gamma_1 & 0 \\ 0 & \gamma_2 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} i_d - i_d^* \\ i_q - i_q^* \\ \omega_r - \omega_r^* \end{bmatrix} - \begin{bmatrix} L_q i_q (\omega_r - \omega_r^*) \\ L_d i_d (\omega_r - \omega_r^*) \end{bmatrix} \quad (3.14)$$

Para encontrar la ley de control se parte del hecho que $\mathbf{e}_u = \mathbf{u} - \mathbf{u}^*$, despejando: $\mathbf{u} = \mathbf{e}_u + \mathbf{u}^*$ de donde se obtiene la ecuación 3.15.

$$\mathbf{u} = \begin{bmatrix} v_d \\ v_q \end{bmatrix} = \begin{bmatrix} v_d^* - \gamma_1 (i_d - i_d^*) - L_q i_q (\omega_r - \omega_r^*) \\ v_q^* - \gamma_2 (i_q - i_q^*) - L_d i_d (\omega_r - \omega_r^*) \end{bmatrix} \quad (3.15)$$

Los valores deseados v_d^* , v_q^* , i_d^* y i_q^* se obtienen a partir del punto de equilibrio del sistema, cuando $\dot{\mathbf{X}} = 0$. Esto es:

$$\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \omega_r^* L_q i_q^* - R i_d^* + v_d^* \\ -\omega_r^* (L_d i_d^* + \Psi_{PM}) - R i_q^* + v_q^* \\ \Psi_{PM} i_q^* - \frac{2}{3P} (T_L + \frac{B\omega_r^*}{P}) \end{bmatrix} \quad (3.16)$$

Despejando y separando las ecuaciones de la ecuación 3.16.

$$v_d^* = Ri_d^* - \omega_r^* L_q i_q^* \quad (3.17)$$

$$v_q^* = Ri_q^* + \omega_r^* (L_d i_d^* + \Psi_{PM}) \quad (3.18)$$

$$i_q^* = \frac{2}{3P} \frac{1}{\Psi_{PM}} (T_L + \frac{B\omega_r^*}{P}) \quad (3.19)$$

El máximo torque se obtiene cuando $i_d^* = 0$, sustituyendo esto en la ecuación 3.17, 3.18; quedando de la siguiente forma:

$$v_d^* = -\omega_r^* L_q i_q^* \quad (3.20)$$

$$v_q^* = Ri_q^* + \omega_r^* \Psi_{PM} \quad (3.21)$$

$$i_q^* = \frac{2}{3P} \frac{1}{\Psi_{PM}} (T_L + \frac{B\omega_r^*}{P}) \quad (3.22)$$

Y la ley de control queda de la siguiente forma:

$$\mathbf{u} = \begin{bmatrix} v_d \\ v_q \end{bmatrix} = \begin{bmatrix} v_d^* - \gamma_1(i_d) - L_q i_q(\omega_r - \omega_r^*) \\ v_q^* - \gamma_2(i_q - i_q^*) + L_d i_d(\omega_r - \omega_r^*) \end{bmatrix} \quad (3.23)$$

Es importante hacer notar que la ley de control, dada por la ecuación 3.23, tiene dos salidas que son v_d y v_q y nuestro PMSM es un motor trifásico, por lo que se requiere aplicar la transformada inversa de Clark y Park para obtener v_a , v_b y v_c .

Con respecto a las corrientes, en el motor se miden i_a , i_b y i_c ; las entradas al control son las corrientes i_d e i_q , que se muestran en la ecuación 3.23, por lo que se aplicará la transformada inversa de Clark y Park.

La transformada directa e inversa de Clark y Park se incluirán dentro del bloque de control, con el fin de tener como entradas las corrientes i_a , i_b y i_c y las salidas v_a , v_b y v_c .

De las ecuaciones 3.20, 3.21 y 3.22, ω_r es la velocidad angular del motor expresada en revoluciones por minuto del motor y ω_r^* es la de referencia.

Es muy importante mencionar que γ_1 y γ_2 de la ecuación 3.23, son las ganancias y con ellas se modifican la forma en que el control llegará al punto de equilibrio, el cual están en función de las revoluciones que se desea tener en el motor PMSM; estas ganancias deben tener valores mayores que 0.

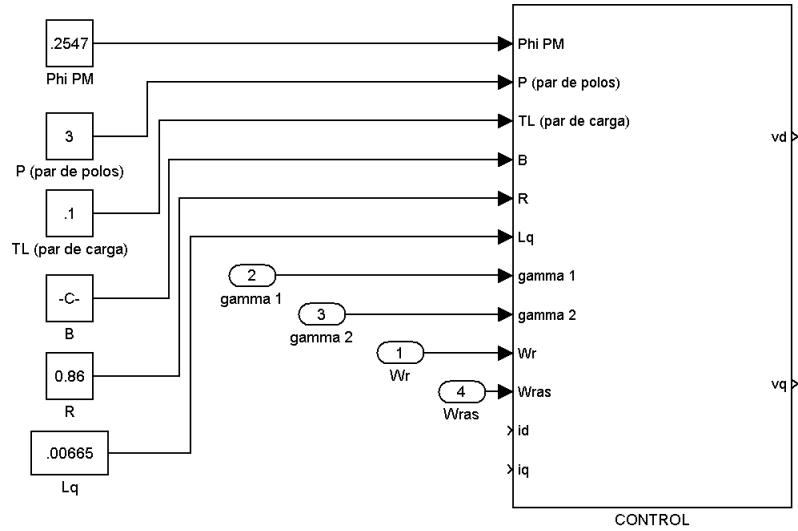


Figura 3.1: Bloque en simulink del control por pasividad de PMSM con parametros.

3.3. Implementación del control por pasividad de PMSM en el programa simulink.

En la plataforma de simulink se crea el bloque de control por pasividad del PMSM empleando las relaciones matemáticas de las ecuaciones 3.20, 3.21, 3.22 y 3.23; donde las entradas son i_a , i_b , i_c , ω_r , ω_r^* , γ_1 , γ_2 y Θ , esta última variable es empleada para las transformadas de Clark y Park; y las salidas v_a , v_b y v_c , como se muestra en la figura 3.1. Para formar este bloque es necesario crear los bloques de la ley de control de la ecuaciones 3.23, 3.20, 3.21 y 3.22.

En la figura 3.1 se muestra el bloque del control por pasividad del PMSM, en éste se incluyen los parámetros correspondientes a la ecuación 3.23; además de los parámetros del motor que están expresados en la tabla 2.2.

Se muestra en la figura 3.2 los bloques descritos anteriormente y se agregan los correspondientes a las transformadas directa e inversa de Clark y Park y se tiene el bloque de control por pasividad de PMSM.

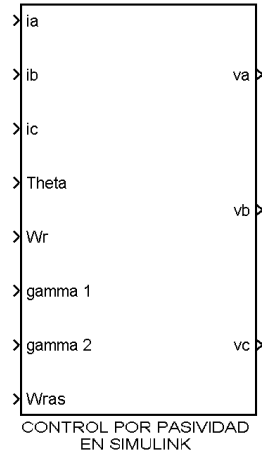


Figura 3.2: Bloque en simulink del control por pasividad de PMSM con parámetros incluidos.

Capítulo 4

Implementación del control por pasividad del PMSM en un FPGA.

Los dispositivos digitales utilizan un lenguaje de programación particular para que realicen las operaciones requeridas en las ecuaciones que se van a aplicar al sistema, además de convertir las señales que ingresan a datos que pueda procesar y finalmente proveer de señales adecuadas para saber el resultado de sus operaciones.

En este capítulo se desarrollará el código de la ecuación de control por pasividad para implementarlo en un FPGA, este es desarrollado en VHDL.

4.1. Dispositivo FPGA.

Durante los últimos 30 años se ha tenido un gran interés por la aplicación del control digital en la electrónica de potencia y el control de aplicaciones como es el de los motores PMSM, esto es debido a los grandes avances que ha tenido la tecnología. La flexibilidad de los nuevos dispositivos, reduce el tiempo de diseño y la posibilidad de implementar complejos tratamientos en tiempo real. [Monmasson2011]

Los campos de aplicación de los FPGA se han incrementado notablemente, entre los cuales se tiene: procesamiento de señales, telecomunicaciones, vídeo, sistemas de control eléctrico, sistemas de control embebido [Bossoufi2011]; en este último, está aplicado este trabajo.

Otra razón de usar estos dispositivos es la posibilidad de interconexión entre diversos componentes como: convertidores analógico-digital, retenedor de orden cero para interconectarse con los dispositivos de medición analógico, etc. Al realizar una retroalimentación al sistema se reduce el ancho de banda. Con los avances que se tienen tanto en el control analógico y digital, los procesos digitales son casi-instantáneos en sus algoritmos de control. Por lo que se inicia una nueva categoría de controladores: los cuasi-analógicos. Los FPGA son buenos candidatos a esta categoría, debido a que pueden acelerar significativamente el tiempo de procesamiento de un algoritmo por su capacidad para implementar una arquitectura específica que encaja con las características más destacadas del algoritmo a implementar [Monmasson2011]. Además, el control implementado en FPGA responde eficientemente a cambios presentes y futuros [Bossoufi2011].

Existen dos casos derivados del beneficio de la rapidez del control basado en FPGA: El primer caso, aplicación de alta demanda donde se usa el control basado en FPGA con una arquitectura de hardware personalizada y el segundo caso, la restricción de la frecuencia de conmutación donde es relevante el aprovechamiento del tiempo entre cada toma de muestra, agregando una solución al sistema en cada periodo.

Los FPGA son una amplia familia de componentes lógicos programables, estos son definidos como una matriz de lógicos programables (CLBs), vinculados entre si por una red completamente reconfigurable. Las celdas de memoria controlan la interconexión de los bloques lógicos de tal modo que cumplan con los requerimientos específicos de la aplicación [Monmasson2011].

La tecnología flash y/o volátil que utiliza el FPGA conserva la configuración mientras esta apagado y en cuanto se energiza esta listo para ejecutar el programa anteriormente cargado en el dispositivo. Una arquitectura genérica del FPGA basada en SRAM está compuesto de una matriz de CLBs, que consiste de un conjunto de celdas lógicas; la matriz de CLBs esta enlazada por un anillo de bloques configurables de entrada/salida (IOBs). Estos están interconectados por una red programable [Monmasson2011].

Los usos frecuentes de los FPGA son para implementar funciones complejas. Esto debido a los avances de integración a gran escala y desarrollo de herramientas; Estas herramienta están basadas en el lenguaje de descripción de hardware (HDLs), como el lenguaje de descripción de hardware de alta velocidad (VHDL)ó Verilog. Al sintetizar VHDL se obtiene la descripción que representa exactamente los operadores y las variables de la arquitectura final [Monmasson2011].

Tabla 4.1: Representación de punto flotante en la palabra binaria.

| Componente | Palabra de 32 bits | | Palabra de 64 bits | |
|------------|---------------------|-------------------|---------------------|-------------------|
| | ‡ de bits asignados | Posición asignada | ‡ de bits asignados | Posición asignada |
| Signo | 1 | 31 | 1 | 63 |
| Exponente | 8 | 23 - 30 | 11 | 52 - 62 |
| Mantisa | 23 | 0 - 22 | 52 | 0 - 51 |

Los FPGAs son dispositivos que utilizan señales digitales para realizar su procesamiento, algunas de estas señales son obtenidas de dispositivos analógicos y es necesario representarlas en un formato específico, punto fijo o punto flotante.

4.1.1. Representación binaria en punto flotante

La arquitectura de **punto flotante** está basada en la notación exponencial (científica). Esta notación tiene una exactitud suficiente, solo que necesita de un redondeo. La notación exponencial de un número diferente de cero se muestra en la ecuación (4.1).

$$x = \pm S * B^E \tag{4.1}$$

Donde:

\pm = signo del número.

S = mantisa, $0 < S < 10$.

B = Base en que se está trabajando.

E = Exponente, con su signo.

En la representación digital se necesita ubicar cada una de estas partes y debe normalizarse, para ello se aplica la norma **IEEE 754**. La longitud de palabra utilizada es de 32 bits para una precisión simple y 64 bits para una precisión doble. Con la palabra de 32 bits se representan los número desde 10^{-38} hasta 10^{38} y con 64 bits 10^{-308} hasta 10^{308} . En la tabla 4.1 se muestra la ubicación en la palabra digital de los componentes de la notación científica.

4.1.2. Representación binaria en punto fijo.

La arquitectura de **punto fijo** se basa en la ubicación de un punto virtual dentro de la longitud de la palabra; la cual la divide en dos, la parte entera y la parte fraccionaria.

La representación utilizada para esta arquitectura es de complemento 2, por lo que, el bit mas significativo corresponde al signo. Los dígitos que representa el número entero se ubican a la izquierda del punto virtual, y los de la parte fraccionaria a la derecha hasta el bit menos significativo.

Tabla 4.2: Diferentes rangos para una palabra de N=8 bits.

| Formato | Máximo número | | Resolución |
|---------|---------------|-----------|------------|
| | Negativo | Positivo | |
| A(1.7) | -1 | 0.9921875 | 0.0078125 |
| A(2.6) | -2 | 1.984375 | 0.015625 |
| A(3.5) | -4 | 3.96875 | 0.03125 |
| A(4.4) | -8 | 7.9375 | 0.0625 |
| A(5.3) | -16 | 15.975 | 0.125 |
| A(6.2) | -32 | 31.75 | 0.25 |
| A(7.1) | -64 | 63.5 | 0.5 |
| A(8.0) | -128 | 127 | 1 |

El formato que utilizaremos para representar a la palabra es el siguiente:

$$Q(m.f) \quad (4.2)$$

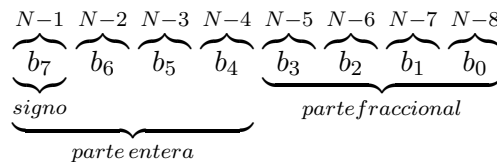
Donde:

Q = Palabra digital.

m = número de dígitos de la parte entera de la palabra.

f = número de dígitos de la parte fraccional de la palabra.

Para una palabra de longitud N=8 bits y un formato de A(4.4), se tiene que el bit mas significativo es el signo, la parte entera son 4 dígitos y 4 dígitos de la parte fraccional; como se muestra a continuación:



El rango que se tiene con esta representación se calcula con la siguientes expresiones:

En los positivos:

$$2^{m-1} - \frac{1}{2^f} \quad (4.3)$$

Y para los negativos:

$$- 2^{m-1} \quad (4.4)$$

Para el ejemplo de N=8, con formato A(4.4) su rango es A(-8,7.9375).

La resolución en este formato se obtiene con la expresión (4.5), siendo para este mismo ejemplo de 0.0625.

$$\frac{1}{2^f} \quad (4.5)$$

Como se muestra en la tabla 4.2, en el formato de punto fijo con una misma longitud de palabra se puede representar magnitudes mayores sacrificando la precisión.

Tabla 4.3: Multiplicación con diferentes formatos en binario con punto fijo.

| Multiplicación en punto fijo. | | |
|--------------------------------|---|--|
| Formato | Q(4.0) | Q(2.2) |
| Representación | $\underbrace{a_3 a_2 a_1 a_0}_{\text{enteros}}$ | $\underbrace{a_3 a_2}_{\text{enteros}} \underbrace{a_1 a_0}_{\text{fraccion}}$ |
| Resultado de multiplicación | $c_7 c_6 c_5 c_4 c_3 c_2 c_1 c_0$ | $c_7 c_6 c_5 c_4 c_3 c_2 c_1 c_0$ |
| Resultado truncado y recortado | $\underbrace{c_3 c_2 c_1 c_0}_{\text{entero}}$ | $\underbrace{c_5 c_4}_{\text{entero}} \underbrace{c_3 c_2}_{\text{fraccion}}$ |

Es importante saber que el formato Q(m.f) lo determina el programador y el equipo seguirá manejando el valor de la palabra completa; esto es, con la palabra de N=8 bits el rango que alcanza en complemento 2 en el bloque **DSP** (Digital signal Processing) es de (-128,127) y para el programador que usa el formato Q(4.4) es de (-8,7.9375).

4.1.3. Operaciones aritméticas con palabras binarias con formato Q(m.f).

Las operaciones básicas que se realizan con palabras binarias con formato Q(m.f) son la suma y la multiplicación. El **DSP** está programado para hacer operaciones con palabras representadas en Complemento 2, y su reglas son las siguientes:

La suma de dos palabras con N número de bits el resultado esta contenido en una palabra de N bits, esto es: $A_N + B_N = C_N$

En la multiplicación de dos palabras, A_M y B_N el resultado estará contenido en una palabra de C_{M+N} .

Es importante mencionar que el formato para hacer las operaciones en binario, la palabra debe estar con el mismo formato; si los formatos de las palabras son diferentes, se debe de realizar los ajustes necesarios para dejarlo con el mismo formato.

El resultado obtenido por la multiplicación es del doble de longitud que la palabra de los multiplicandos, por lo que es necesario ajustar el resultado. La forma en que se realiza es truncando la palabra, es simple, se truncan de la parte de los bits menos significativos el número de bits que conforman la parte fraccional de la palabra y con los bits sobrantes se inicia a contar desde el bit menos significativo hasta llegar a la longitud de la palabra. En la tabla 4.3 se muestra dos ejemplos con formato distinto.

4.2. Lenguaje de programación en VHDL

Los circuitos integrados de alta velocidad como PLD, CPLD, PAL y FPGA utilizan para la descripción de hardware el lenguaje de programación VHDL (Hardware Description Language); este lenguaje permite describir la estructura, flujo de datos y comportamiento de dispositivos lógicos. En este se puede implementar distintos circuitos lógicos, tanto combinatoriales como secuenciales.

VHDL se desarrolló para el modelado y simulación lógica dirigida por eventos de sistemas digitales, también es usado para síntesis automática de circuitos. Este lenguaje tiene una sintaxis amplia y flexible, permitiendo el modelado preciso en distintos estilos.

Algunas de las ventajas del uso de VHDL son:

- VHDL permite diseñar, modelar y comprobar un sistema desde un alto nivel de abstracción bajando hasta el nivel de definición estructural de compuerta.
- El cumplir con un estándar (IEEE Std 1076-1987) disminuye problemas de comunicación y compatibilidad, tanto en la industria como en la investigación.
- VHDL permite un diseño Top-Down, modelando el comportamiento en bloques de alto nivel y analizarlos con la simulación.
- VHDL permite dividir o fraccionar un diseño hardware y su descripción VHDL en unidades más pequeñas, es decir permite la Modularidad.

Existen dos formas de describir un circuito, en uno se describe el circuito indicando los componentes que lo conforman y su interconexión; la otra es describiendo que hace o como funciona el circuito, es decir su comportamiento.

4.2.1. Estructura de un programa VHDL.

La programación en VHDL tiene dos partes en su estructura: la declaración de la entidad y la declaración de la arquitectura.

4.2.1.1. ENTITY en VHDL

La *entidad* es la abstracción del hardware, como si fuera una caja negra; aquí se especifica las entradas y salidas de la interfaz. La entidad se declara con la palabra *ENTITY* y su topología es:

```
ENTITY nombre de la entidad IS
PORT(
    puertos de entrada : IN tipo de dato
    puertos bidireccionales : INOUT tipo de dato
    puerto de salida :OUT tipo de dato
    puerto de salida : BUFFER tipo de dato
END nombre de la entidad
```

Los identificadores son solamente nombres o etiquetas que son usados para que el programador identifique que contiene, puede ser una función, un proceso, una señal; un identificador es como ejemplo el nombre de la entidad. Estos cumplen unas reglas simples para ser compilados sin error, mismas que son enunciadas a continuación [maxinez2002] :

- El primer caracter siempre es una letra mayúscula o minúscula.
- El segundo caracter no puede ser un guion bajo.
- Dos guiones juntos no son permitidos.
- Un identificador no puede utilizar símbolos.

En los sistemas digitales son usados los bits, que representan individualmente dos valores lógicos (0 y 1); en la práctica para la programación en VHDL se utilizan las palabras binarias y son conocidas como **vectores de bit**, estas son consideradas en grupo y no como bits independientes [maxinez2002]; su representación de este vector para una palabra de 4 bits es la siguiente:

$$\text{vector_}A = [A3, A2, A1, A0]$$

El **vector de bits** se utiliza en las entradas y salidas del sistema, así como en sus operaciones; este define el valor máximo que puede usarse en las operaciones en VHDL.

4.2.1.2. Arquitectura de VHDL

La arquitectura es la unidad de diseño que describe el comportamiento interno del circuito, es decir, es aquí donde se describe el funcionamiento de una ENTITY [maxinez2002], su sintáxis es la siguiente:

```
ARCHITECTURE (nombre de la arquitectura) OF (nombre de la entidad) IS
  (área de la declaración de señales internas y externas)
BEGIN
  (área de la descripción del proceso)
END (nombre de la arquitectura)
```

En esta declaración se tienen dos etiquetas, una que es el nombre de la arquitectura y el nombre de la entidad a la cual esta asociada; y en el área de descripción del proceso, es donde se desarrolla el algoritmo del proceso.

Es importante anexar al código en VHDL librerías que requiere para hacer las operaciones, una de ellas es *ieee* y los que se usa de esta comúnmente son: *ieee.std_logic_1164.all* y *ieee.numeric_std.all*.

Un ejemplo muy sencillo es de comparar dos datos y obtener como resultado si son iguales o no lo son; esto es:

```
si a = b entonces c = 1
si a ≠ b entonces c = 0
```

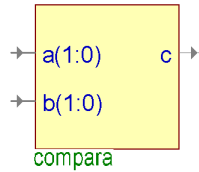


Figura 4.1: Bloque en VHDL de la función compara.

El código en VHDL puede ser de la siguiente forma:

```
library ieee;
use ieee.std_logic_1164.all;
entity comp is
port (a, b : IN std_logic_vector ( 1 downto 0 );
      c : OUT bit );
end comp;

architecture funcional of comp is
begin
  compara: process (a,b)
  begin
    If a = b then
      c <= '1';
    else
      c <= '0';
    end if;
  end process compara;
end funcional;
```

La figura 4.1 se genera con el código anterior. En el que se observa que las entradas son vectores y la salida es un bit; además que es evidente que es una caja negra en lo que no se ve que operación realiza, por lo que el nombre debe tener relación con la función que hace este bloque.

En ingeniería se requiere de aplicar varias veces la misma función y esta se puede hacer en un diagrama de flujo; solamente se debe de manejar el mismo tipo de dato que ingrese al bloque.

4.3. Generación del código en alto nivel del control por pasividad de PMSM.

La generación del código en alto nivel del control por pasividad para un PMSM, requiere de usar como herramienta un programa en el que se genere, sintetize el algoritmo y sea compatible con el dispositivo que contiene el FPGA; el programa que se utilizará es **Aldec Active-HDL Student Edition**, además de generar el código en VHDL es compatible con Simulink, esto significa que es posible realizar cosimulación entre Simulink y Active-HDL.

Tabla 4.4: Magnitudes del formato L(15.20).

| Formato | Máximo número | | Resolución |
|----------|---------------|-----------------------------|------------------------|
| | Negativo | Positivo | |
| L(15.20) | -16 384 | 16 383.99999904632568359375 | 0.00000095367431640625 |

La cosimulación que se realiza entre simulink y Aldec Active-HDL permite comparar la respuesta del control por pasividad del PMSM generado en simulink con el construido en VHDL. Para elaborar el código en VHDL es necesario definir la longitud de la palabra que va a ser usada y su formato; además, el número estará representado en complemento a 2 y en punto fijo.

En la tabla 4.4 se visualiza las magnitudes que se tiene con la longitud de la palabra con la cual se trabajará que es de 35 bits; con un formato de L(15,20); que se obtiene aplicando las ecuaciones 4.3, 4.4 y 4.5.

Con la definición del formato de la palabra que va ha ser usado, se genera el código de las funciones básicas como son: la suma, la resta y la multiplicación; como se mencionó en la sección 4.1.3 la suma y la resta generan en el resultado la misma longitud de palabra que de la entrada.

El código de la función suma y resta se muestran a continuación:

Suma

```
library IEEE;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;

entity SUMA14_20 is
    port (H1: IN std_logic_vector(34 downto 0);
          I1: IN std_logic_vector(34 downto 0);
          K1: OUT std_logic_vector(34 downto 0) );
end SUMA14_20;

architecture simple of SUMA14_20 is
begin
    K1<=std_logic_vector(signed(H1)+signed(I1));
end simple;
```

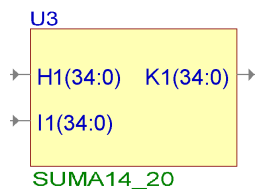


Figura 4.2: Bloque de la función suma en Active-HDL.

En la figura 4.2 se muestra el bloque que se genera para esta función; está indica el valor la longitud de la entrada, que es de 35 bits.



Figura 4.3: Bloque de la función resta en Active-HDL.

Resta

```
library IEEE;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;
entity RESTA is
  port ( H1: IN std_logic_vector(34 downto 0);
         RES: IN std_logic_vector(34 downto 0);
         K1: OUT std_logic_vector(34 downto 0) );
end RESTA;
architecture simple of RESTA is
begin
  K1<=std_logic_vector(signed(H1)-(signed(RES)));
end simple;
```

En la figura 4.3 se muestra el bloque en Active-HDL que representa a este código, el cual será utilizado en los diagramas de otras funciones.

Es importante destacar que la palabra SIGNED le indica al programa que tome en cuenta el bit de signo para realizar las operaciones.

La función de multiplicación, como en la sección 4.1.3, da como resultado una palabra con el doble de longitud y debe ser recortada al formato que se está manejando, en este caso L(15.20); el código se describe a continuación, dentro de la sección del proceso se hace el truncado de la palabra de 70 bits a 35 bit.

Multiplicación

```
library IEEE;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;
entity MULTI14_20 is
  port ( A1: IN std_logic_vector(34 downto 0);
         B1: IN std_logic_vector(34 downto 0);
         D1: OUT std_logic_vector(34 downto 0) );
end MULTI14_20;
architecture simple of MULTI14_20 is
  signal aux: std_logic_vector(69 downto 0);
begin
  aux <= std_logic_vector(signed(A1)*signed(B1));
  D1(34 downto 0) <= aux(54 downto 20);
end simple;
```

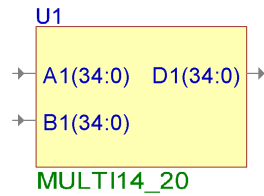


Figura 4.4: Bloque de la función multiplicación en Active-HDL.

Se muestra en la figura 4.4 el bloque se genera en Active HDL; donde se puede observar que las entradas son de la misma longitud que la salida y la operación de truncado no se realiza.

Otro bloque que se usará como base para realizar el diagrama a bloques del control en Aldec Active-HDL es el de las constantes que se requieren en las ecuaciones 3.20, 3.21, 3.22 y de la ley de control 3.23, las cuales están en la tabla 2.2; se genera este bloque para tener los valores cargados directamente en el FPGA y no estar ingresando varias veces estos valores; además se agregarán otras constantes para hacer operaciones mas sencillas. El código se muestra a continuación.

```

library IEEE;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;
entity CONSTANTES14_20 is
  port (
    G1 : OUT std_logic_vector(34 downto 0); -- GAMMA 1
    G2 : OUT std_logic_vector(34 downto 0); -- GAMMA 2
    R : OUT std_logic_vector(34 downto 0); -- Resistencia
    Lq : OUT std_logic_vector(34 downto 0); -- Inductancia
    M1 : OUT std_logic_vector(34 downto 0); -- menos uno
    PI01: OUT std_logic_vector(34 downto 0); -- $pi*0,1$
    PHI : OUT std_logic_vector(34 downto 0); -- PHI PM
    VAR1: OUT std_logic_vector(34 downto 0); -- $2/3$
    VAR2: OUT std_logic_vector(34 downto 0); -- $Tl/(Phi * P)$
    VAR3: OUT std_logic_vector(34 downto 0); -- $B/(P*P*Phi)$
    CONS1: OUT std_logic_vector(34 downto 0); -- 0,5
    CONS2: OUT std_logic_vector(34 downto 0); -- sqtr$(3)/2$
    CONS3: OUT std_logic_vector(34 downto 0)); -- $2*PI/3$
end CONSTANTES14_20;
architecture simple of CONSTANTES14_20 is
  begin
    R <= "00000000000000011011100001010001111"; -- 0,86
    G1 <= "000001111101000000000000000000000000"; -- 1000
    G2 <= "000000000000011000000000000000000000"; -- 3
    Lq <= "00000000000000000000000001101100111101"; -- 0,000665
    M1 <= "111111111111111000000000000000000000"; -- menos uno
    PHI <= "00000000000000001000001001100110101"; -- 0,2547
    PI01 <= "00000000000000001010000011011001011"; -- 0,31415844 $PI*0,1$
    VAR1 <= "000000000000000010101010101010101011"; -- 0,666666985
    VAR3 <= "0000000000000000000000000000010101101"; -- 0,000164986
    VAR2 <= "00000000000000000100001100000001110"; -- 0,130872726
    CONS1 <= "000000000000000010000000000000000000"; -- 0,5
  
```

```

CONS2 <= "000000000000000011011101101100111101"; -- 0,866024971
CONS3 <= "000000000000001000011000001010100100"; -- 2,0943965911865234375
end simple;

```

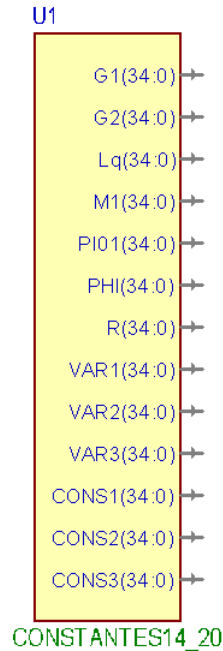


Figura 4.5: Bloque de constantes en Active-HDL.

En la figura 4.5 se muestra el bloque que se genera con el código anterior; éste tiene la particularidad de no tener entradas, y sólo salidas; con la indicación del nombre de la señal y la longitud de la palabra.

En el código de las constantes observamos en la arquitectura los valores, como ejemplo el G2 tiene su valor en un vector de bits de 0 00 00 00 00 00 00 11 00 00 00 00 00 00 00 00 00 00; y éste valor para el dispositivo va a ser de 3 145 728 y con el uso del punto fijo y el formato L(15.20) el valor es de 3; en la tabla 4.5, se visualiza y se muestra la obtención del valor representado.

Otro ejemplo de los diferentes valores que toma un vector de bits, es la constante denominada **CONS3**; que representa a $\frac{2 * \pi}{3}$ que es 2.0943951023931954923084289221863; y es representado por el vector 0 00 00 00 00 00 00 10 00 01 10 00 00 10 10 10 01 00 que representa el valor real en bits 2 196 132 y en punto fijo y con formato el L(15.20) representa 2,0943965911865234375, esto por el máximo en su rango y por la resolución que se tiene; estos valores se ejemplifican en la tabla 4.6.

Tabla 4.5: G2: Valor real en bits y en punto fijo con formato L(15.20).

| Posición en el vector | Vector de bits | Valor real en bits | | Valor en punto fijo con formato L(15.20) | | |
|-----------------------|----------------|-----------------------|----------------|--|-----------------------|------------------------|
| | | Valor por su posición | Valor a sumar | | Valor por su posición | Valor a sumar |
| 34 | 0 | 2^{34} | 17 179 869 184 | 0 | | signo |
| 33 | 0 | 2^{33} | 8 589 934 592 | 0 | 2^{13} | 8 192 |
| 32 | 0 | 2^{32} | 4 294 967 296 | 0 | 2^{12} | 4 093 |
| 31 | 0 | 2^{31} | 2 147 483 648 | 0 | 2^{11} | 2 048 |
| 30 | 0 | 2^{30} | 1 073 741 824 | 0 | 2^{10} | 1 024 |
| 29 | 0 | 2^{29} | 536 870 912 | 0 | 2^9 | 512 |
| 28 | 0 | 2^{28} | 268 435 456 | 0 | 2^8 | 256 |
| 27 | 0 | 2^{27} | 134 217 728 | 0 | 2^7 | 128 |
| 26 | 0 | 2^{26} | 67 108 864 | 0 | 2^6 | 64 |
| 25 | 0 | 2^{25} | 33 554 432 | 0 | 2^5 | 32 |
| 24 | 0 | 2^{24} | 33 554 432 | 0 | 2^4 | 16 |
| 23 | 0 | 2^{23} | 8 388 608 | 0 | 2^3 | 8 |
| 22 | 0 | 2^{22} | 4 194 304 | 0 | 2^2 | 4 |
| 21 | 1 | 2^{21} | 2 097 152 | 2 097 152 | 2^1 | 2 |
| 20 | 1 | 2^{20} | 1 048 576 | 1 048 576 | 2^0 | 1 |
| 19 | 0 | 2^{19} | 254 288 | 0 | 2^{-1} | 0.5 |
| 18 | 0 | 2^{18} | 262 144 | 0 | 2^{-2} | 0.25 |
| 17 | 0 | 2^{17} | 131072 | 0 | 2^{-3} | 0.125 |
| 16 | 0 | 2^{16} | 65 536 | 0 | 2^{-4} | 0.0625 |
| 15 | 0 | 2^{15} | 32 768 | 0 | 2^{-5} | 0.0125 |
| 14 | 0 | 2^{14} | 16 384 | 0 | 2^{-6} | 0.015625 |
| 13 | 0 | 2^{13} | 8 192 | 0 | 2^{-7} | 0.0078125 |
| 12 | 0 | 2^{12} | 4 093 | 0 | 2^{-8} | 0.00390625 |
| 11 | 0 | 2^{11} | 2 048 | 0 | 2^{-9} | 0.001953125 |
| 10 | 0 | 2^{10} | 1 024 | 0 | 2^{-10} | 0.0009765625 |
| 9 | 0 | 2^9 | 512 | 0 | 2^{-11} | 0.00048828125 |
| 8 | 0 | 2^8 | 256 | 0 | 2^{-12} | 0.000244140625 |
| 7 | 0 | 2^7 | 128 | 0 | 2^{-13} | 0.0001220703125 |
| 6 | 0 | 2^6 | 64 | 0 | 2^{-14} | 0.00006103515625 |
| 5 | 0 | 2^5 | 32 | 0 | 2^{-15} | 0.000030517578125 |
| 4 | 0 | 2^4 | 16 | 0 | 2^{-16} | 0.0000152587890625 |
| 3 | 0 | 2^3 | 8 | 0 | 2^{-17} | 0.00000762939453125 |
| 2 | 0 | 2^2 | 4 | 0 | 2^{-18} | 0.000003814697265625 |
| 1 | 0 | 2^1 | 2 | 0 | 2^{-19} | 0.0000019073486328125 |
| 0 | 0 | 2^0 | 1 | 0 | 2^{-20} | 0.00000095367431640625 |
| Valor que corresponde | | | 3 145 728 | | | 3 |

Tabla 4.6: CONS3; Valor real en bits y en punto fijo con formato L(15.20).

| Posición en el vector | Vector de bits | Valor real en bits | | Valor en punto fijo con formato L(15.20) | |
|-----------------------|----------------|-----------------------|----------------|--|----------------------------------|
| | | Valor por su posición | Valor a sumar | Valor por su posición | Valor a sumar |
| 34 | 0 | 2^{34} | 17 179 869 184 | 0 | signo |
| 33 | 0 | 2^{33} | 8 589 934 592 | 0 | 2^{13} 8 192 |
| 32 | 0 | 2^{32} | 4 294 967 296 | 0 | 2^{12} 4 093 |
| 31 | 0 | 2^{31} | 2 147 483 648 | 0 | 2^{11} 2 048 |
| 30 | 0 | 2^{30} | 1 073 741 824 | 0 | 2^{10} 1 024 |
| 29 | 0 | 2^{29} | 5 368 700 912 | 0 | 2^9 512 |
| 28 | 0 | 2^{28} | 268 435 456 | 0 | 2^8 256 |
| 27 | 0 | 2^{27} | 134 217 728 | 0 | 2^7 128 |
| 26 | 0 | 2^{27} | 67 108 864 | 0 | 2^6 64 |
| 25 | 0 | 2^{25} | 33 554 432 | 0 | 2^5 32 |
| 24 | 0 | 2^{24} | 33 554 432 | 0 | 2^4 16 |
| 23 | 0 | 2^{23} | 8 388 608 | 0 | 2^3 8 |
| 22 | 0 | 2^{22} | 4 194 304 | 0 | 2^2 4 |
| 21 | 1 | 2^{21} | 2 097 152 | 2 097 152 | 2^1 2 |
| 20 | 0 | 2^{20} | 1 048 576 | 0 | 2^0 1 |
| 19 | 0 | 2^{19} | 254 288 | 0 | 2^{-1} 0.5 |
| 18 | 0 | 2^{18} | 262 144 | 0 | 2^{-2} 0.25 |
| 17 | 0 | 2^{17} | 131072 | 0 | 2^{-3} 0.125 |
| 16 | 1 | 2^{16} | 65 536 | 65 536 | 2^{-4} 0.0625 |
| 15 | 1 | 2^{15} | 32 768 | 32 768 | 2^{-5} 0.0125 |
| 14 | 0 | 2^{14} | 16 384 | 0 | 2^{-6} 0.015625 |
| 13 | 0 | 2^{13} | 8 192 | 0 | 2^{-7} 0.0078125 |
| 12 | 0 | 2^{12} | 4 093 | 0 | 2^{-8} 0.00390625 |
| 11 | 0 | 2^{11} | 2 048 | 0 | 2^{-9} 0.001953125 |
| 10 | 0 | 2^{10} | 1 024 | 0 | 2^{-10} 0.0009765625 |
| 9 | 1 | 2^9 | 512 | 512 | 2^{-11} 0.00048828125 |
| 8 | 0 | 2^8 | 256 | 0 | 2^{-12} 0.000244140625 |
| 7 | 1 | 2^7 | 128 | 128 | 2^{-13} 0.0001220703125 |
| 6 | 0 | 2^6 | 64 | 0 | 2^{-14} 0.00006103515625 |
| 5 | 1 | 2^5 | 32 | 32 | 2^{-15} 0.000030517578125 |
| 4 | 0 | 2^4 | 16 | 0 | 2^{-16} 0.0000152587890625 |
| 3 | 0 | 2^3 | 8 | 0 | 2^{-17} 0.00000762939453125 |
| 2 | 1 | 2^2 | 4 | 4 | 2^{-18} 0.000003814697265625 |
| 1 | 0 | 2^1 | 2 | 2 | 2^{-19} 0.0000019073486328125 |
| 0 | 0 | 2^0 | 1 | 0 | 2^{-20} 0.00000095367431640625 |
| Valor que corresponde | | | 2 196 134 | | 2,0943965911865234375 |

Las constantes que están agregados en el código del bloque de **CONSTATES14_20** a continuación se mencionan.

$$PI01 = \pi(0,1) \quad (4.6)$$

$$M1 = -1 \quad (4.7)$$

$$VAR1 = \frac{2}{3} \quad (4.8)$$

$$VAR2 = \frac{T_L}{\Psi_{PM}(P)} \quad (4.9)$$

$$VAR3 = \frac{B}{\Psi_{PM}(P^2)} \quad (4.10)$$

$$CONS1 = 0,5 \quad (4.11)$$

$$CONS2 = \frac{\sqrt{3}}{2} \quad (4.12)$$

$$CONS3 = \frac{2\pi}{3} \quad (4.13)$$

Las ecuaciones 3.20, 3.21, 3.22 y 3.23 las reescribimos para este capítulo, asignando nuevos números de ecuación.

$$v_d^* = -\omega_r^* L_q i_q^* \quad (4.14)$$

$$v_q^* = R i_q^* + \omega_r^* \Psi_{PM} \quad (4.15)$$

$$i_q^* = \frac{2}{3P} \frac{1}{\Psi_{PM}} \left(T_L + \frac{B \omega_r^*}{P} \right) \quad (4.16)$$

$$\mathbf{u} = \begin{bmatrix} v_d \\ v_q \end{bmatrix} = \begin{bmatrix} v_d^* - \gamma_1(i_d) \\ v_q^* - \gamma_2(i_q - i_q^*) \end{bmatrix} \quad (4.17)$$

Reescribiendo la ecuación 4.16 se obtiene:

$$i_q^* = \frac{2}{3} \left\{ \frac{T_L}{\Psi_{PM}} + \left[\frac{B}{\Psi_{PM}(P^2)} \right] \omega_r^* \right\} \quad (4.18)$$

Sustituyendo las ecuaciones 4.7, 4.8, 4.9 y 4.10 en 4.14, 4.15 y 4.18 obtenemos las siguientes ecuaciones:

$$v_d^* = M1(\omega_r^*) L_q i_q^* \quad (4.19)$$

$$v_q^* = R i_q^* + \omega_r^* \Psi_{PM} \quad (4.20)$$

$$i_q^* = VAR1 [VAR2 + (VAR3) \omega_r^*] \quad (4.21)$$

Se observa que v_d^* y v_q^* son dependientes de i_q^* ; por lo que primero generaremos el bloque de i_q^* con las funciones básicas que se han creado y con la ecuación 4.21. En la figura 4.6 se muestra el diagrama creado y en la figura 4.7 el bloque i_{qas} .

Con este bloque y con las ecuaciones 4.19, 4.20 y la ley de control 4.17 se crea el el diagrama de control en VHDL como se muestra en la figura 4.8; y el bloque de **bloque de control por pasividad en VHDL** en el programa Aldec Active-HDL se muestra en la figura 4.9.

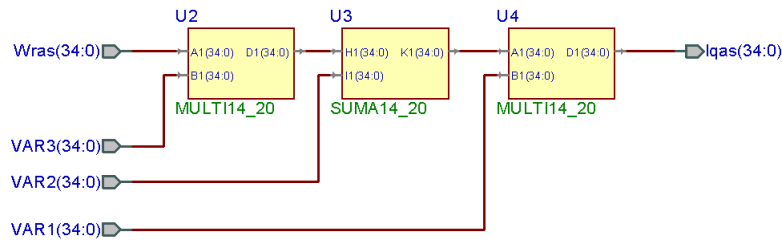


Figura 4.6: Diagrama del bloque de i_q^* en VHDL.

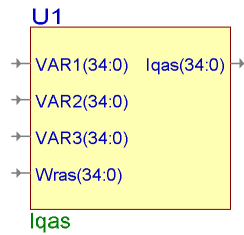


Figura 4.7: Bloque de i_q^* en VHDL.

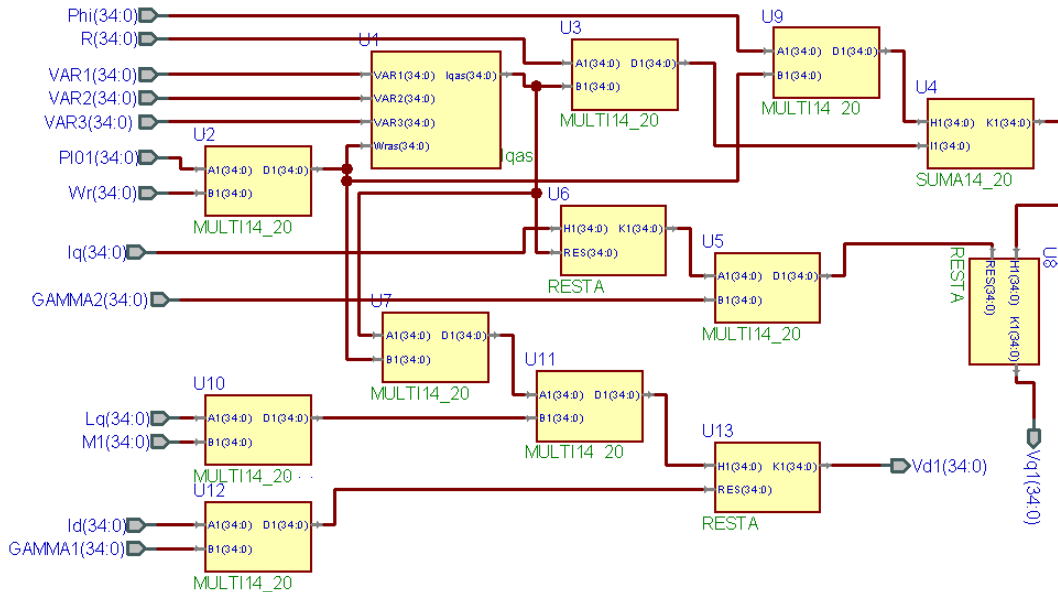


Figura 4.8: Diagrama de control por pasividad en VHDL.

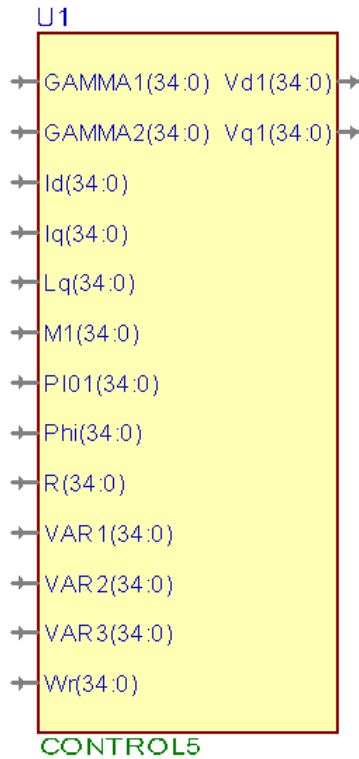


Figura 4.9: Bloque de control por pasividad en VHDL.

Como observamos en la figura 4.9 el control tiene como entradas a las corrientes i_d , i_q y ω_r como señales de entrada variable y las demás entradas son constantes, y sus salidas son v_d y v_q ; del motor se miden las corrientes i_a , i_b y i_c , por lo que es necesario aplicar la transformada de Clark y Park para obtener a i_d y i_q ; y la entrada del motor son los voltajes v_a , v_b y v_c por consiguiente se debe aplicar la inversa de la transformada de Clark y Park a la salida del control por pasividad.

Reescribimos las ecuaciones 2.3 y 2.4, las cuales se enuncian a continuación, y con estas se crea el diagrama de la transformada de Clark y Park que se muestra en la figura 4.10 y queda representado por el bloque de la figura 4.11 y la inversa de la transformada de Clark y Park en la figura 4.12.

$$\begin{bmatrix} v_\alpha \\ v_\beta \end{bmatrix} = \frac{2}{3} \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} v_a \\ v_b \\ v_c \end{bmatrix} \quad (4.22)$$

$$\begin{bmatrix} v_d \\ v_q \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} v_\alpha \\ v_\beta \end{bmatrix} \quad (4.23)$$

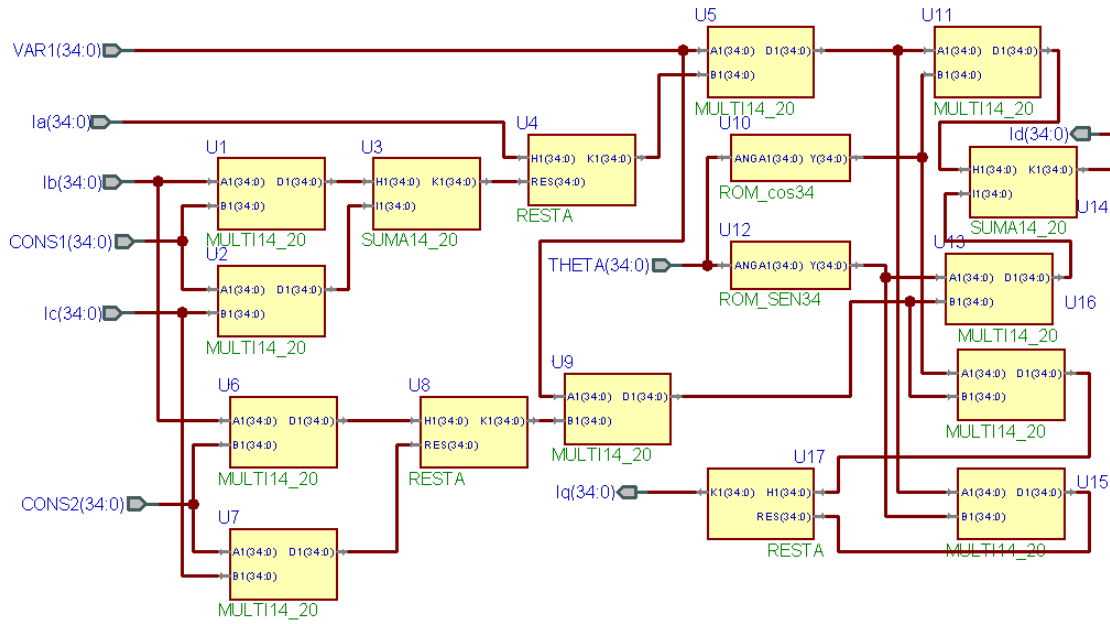


Figura 4.10: Diagrama de la transformada de Clark y Park en VHDL.

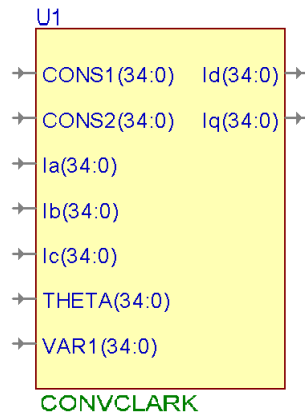


Figura 4.11: Bloque de la transformada de Clark y Park en VHDL.

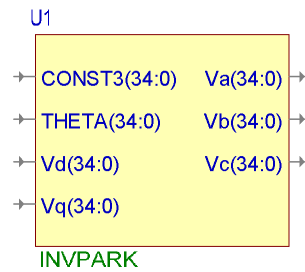


Figura 4.12: Bloque de la inversa transformada de Clark y Park en VHDL.

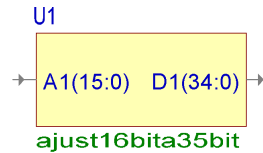


Figura 4.13: Bloque de ajustador de 16 bits a 35 bits en VHDL.

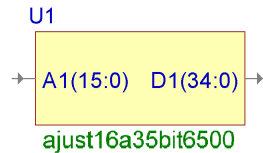


Figura 4.14: Bloque de ajustador de 16 bits a 35 bits DE 6500 en VHDL.

4.4. Acondicionamiento en VHDL de las señales.

El control por pasividad de PMSM generado en la sección 4.3 requiere de más bloques en VHDL para su funcionamiento, debido a que las señales que requiere el bloque de control son medidas por equipos separados del FPGA; Varios de los sensores que son utilizados para obtener las mediciones de corriente tiene una salida digital de una longitud menor y con escala distinta; como por ejemplo los medidores de corriente que convierte una medida analógica en digital.

Se considera para la medida de la corriente un **ADC** (por su siglas en ingles), de 16 bits; se establece que la máxima corriente a medir es de 150 A que corresponde al voltaje de 10 V; para que el control por pasividad procese esta lectura debe ser convertida a la longitud de la palabra de 35 bits, En la figura 4.13 se muestra el bloque que se implementa para realizar este ajuste, un convertidor.

De la misma manera en la figura 4.14 se muestra el bloque para las revoluciones deseadas.

Además, quien hace la conversión de la corriente directa a corriente alterna es un inversor de voltaje; para éste en particular se utiliza **PSIM**, programa para sistemas electrónicos de potencia. **Simulink** tiene la ventaja de realizar la cosimulación con **PSIM**.

En la figura 4.15 se muestra el diseño del inversor para el control por pasividad en simulink.

Los inversores comerciales requieren de señales digitales PWM (modulación por ancho de pulso, por sus siglas en inglés); en la figura 4.15 son mostrados los elementos de entrada del inversor que son integrados a la programación en VHDL y se muestra en la figura 4.16 como queda el inversor para la simulación.

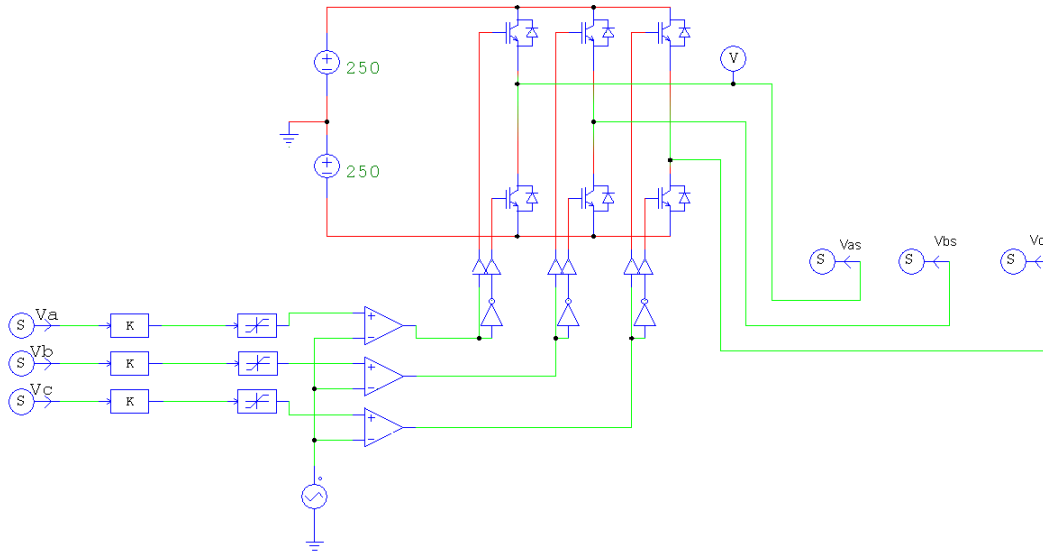


Figura 4.15: Bloque de inversor para simulink en PSIM.

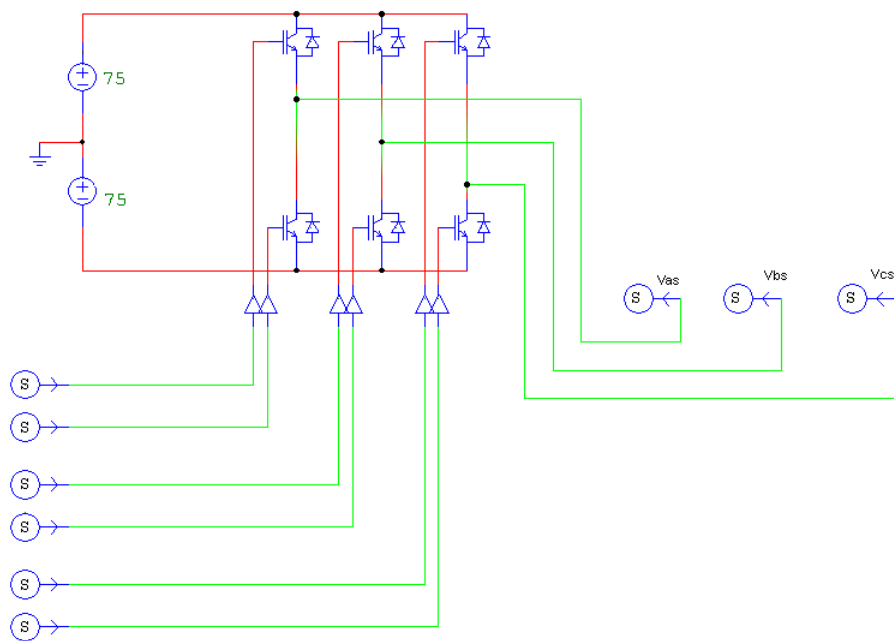


Figura 4.16: Bloque de inversor para VHDL en PSIM.

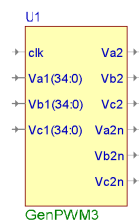


Figura 4.17: Bloque del generador PWM en VHDL.

Se muestra en la figura 4.17 el bloque desarrollado para generar las señales **PWM**.

Para tener en la salida del control el resultado ya procesado se utiliza una máquina de estado, la cual, hace que la entrada y el resultado se sincronicen; esto es que el dato que obtuvo en la entrada se procese y la salida sea el resultado; y no un resultado no procesado o del proceso anterior, provocando un mal cálculo. Se crean cuatro bloques para hacer la máquina de estado. los cuales se muestran a continuación. El bloque de la figura 4.18 da la señal para activar los demás bloques y ellos dejen pasar la información, en la figura 4.19 muestra el bloque que ingresa tres señales con longitud de palabra de 16 bits; el bloque que se visualiza en la figura 4.20 trasmite una palabra de 16 bits, y en la figura 4.21 el bloque retrasmite las seis señales en un solo bit que son generados por el bloque generador de señal **PWM**.

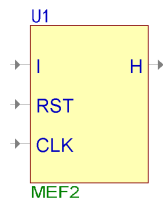


Figura 4.18: Bloque de la máquina de estado en VHDL.

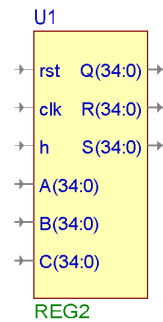


Figura 4.19: Bloque del flip flop de 3 señales de 16 bits en VHDL.

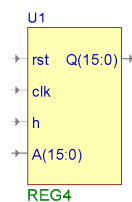


Figura 4.20: Bloque del flip flop de 1 señal de 16 bits en VHDL.

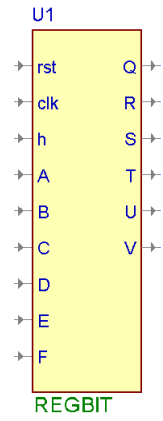


Figura 4.21: Bloque del flip flop de 6 señales de 1 bits en VHDL.

Capítulo 5

Análisis comparativo de resultados y conclusiones.

5.1. Análisis de las cosimulaciones del control por pasividad implementado en simulink del PMSM.

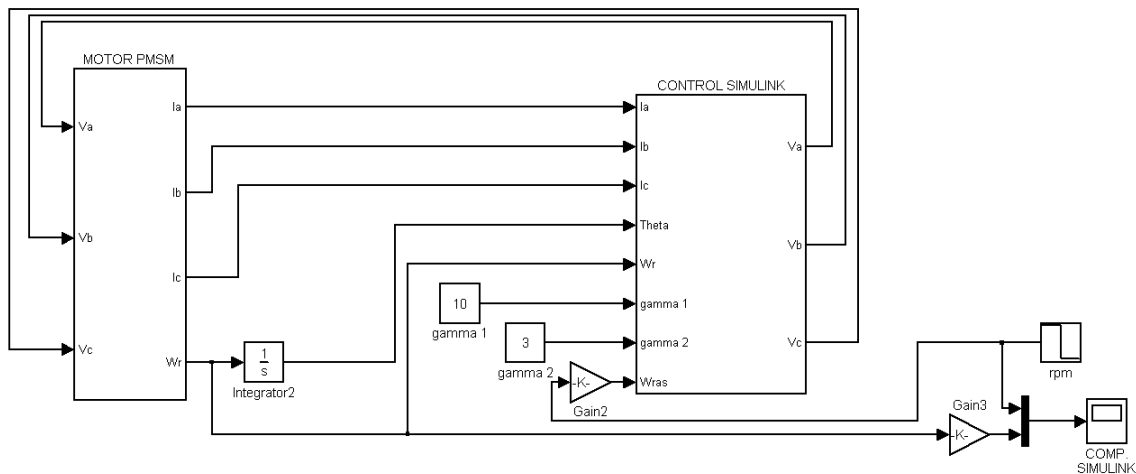


Figura 5.1: Diagrama de bloques en simulink del PMSM y control por pasividad.

En el programa simulink se implementan el bloque del motor de imanes permanentes y el bloque del control por pasividad en simulink; el primero se obtuvo en el capítulo 2, mostrado en la figura 2.6; y el segundo en el capítulo 3, que se muestra en la figura 3.2. Con estos dos bloques en la plataforma de simulink se interconectan en lazo cerrado, y establecida la velocidad del motor se realiza la simulación. Se muestra en la figura 5.1 el diagrama que se crea para hacer la simulación .

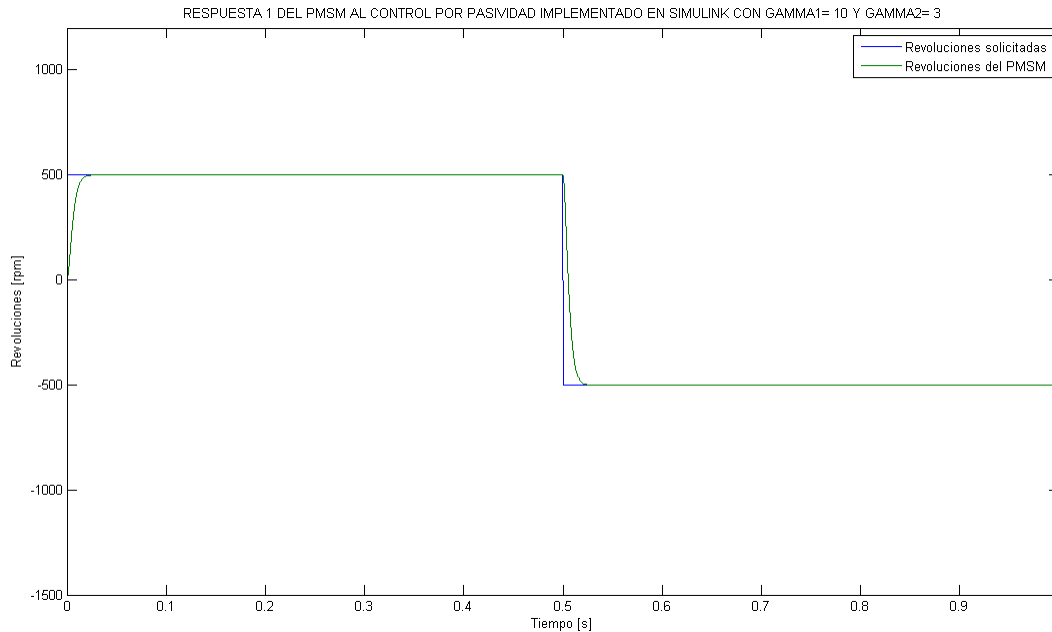


Figura 5.2: Respuesta del PMSM al control por pasividad implementado en simulink con $\gamma_1=10, \gamma_2=3$ para cambio de sentido a 500 rpm.

5.1.1. Simulación del control por pasividad en simulink

Se muestra en la figura 5.2 realizando la simulación correspondiente con las ganancias $\gamma_1=10, \gamma_2=3$, se solicita que el motor gire a 500 rpm los primeros 0.5 s y después a -500 rpm hasta 1 s. El comportamiento del PMSM.

En la figura 5.2 se observa que el control por pasividad implementado en simulink lleva a las revoluciones del PMSM solicitadas y las mantiene; tanto en un sentido como en sentido contrario.

Se realizan más simulaciones con diferentes valores de las ganancias γ_1 y γ_2 , los resultados son mostrados en el anexo B. En la figura 5.3 se muestra el concentrado de las respuestas del PMSM a los diferentes valores propuestos; para visualizar mejor la respuesta, sólo se simula 0.2 segundos, y las respuestas al control por pasividad se muestra en la figura 5.4.

Las distintas respuestas que se observan en la figura 5.4 del PMSM al **control por pasividad implementado en simulink** se clasifican según Kuo [Kuo1996] como se muestra en la tabla 5.1; por lo que el control por pasividad tiene la versatilidad de ofrecer un rango amplio de control sobre la respuesta del motor y con ellos poder elegir el óptimo para una cierta aplicación.

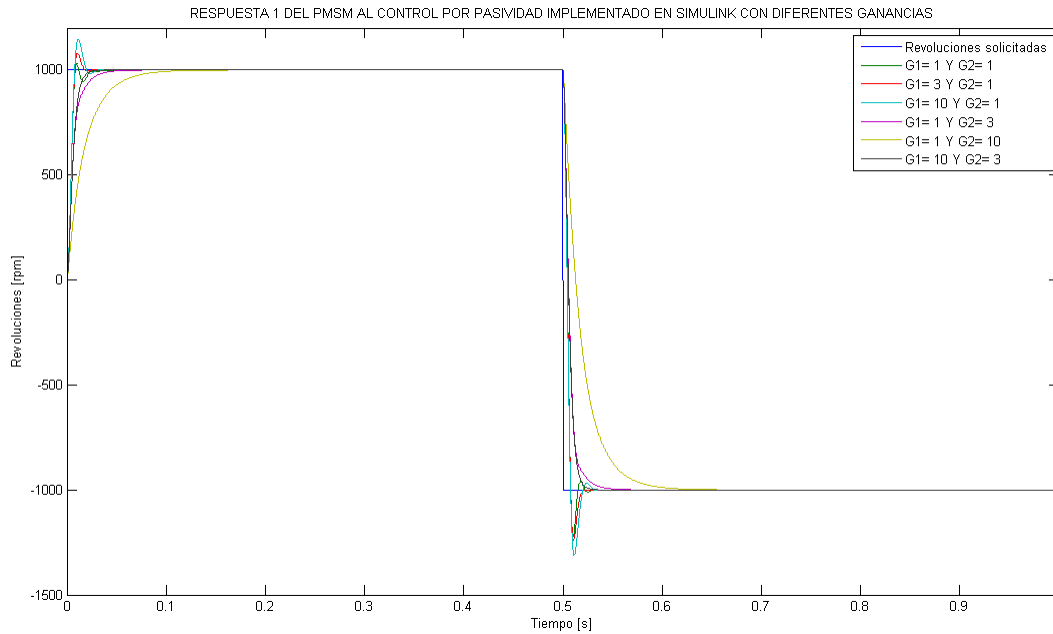


Figura 5.3: Resposta do PMSM ao controle por passividade implementado em simulink com distintos valores de γ_1 y γ_2 .

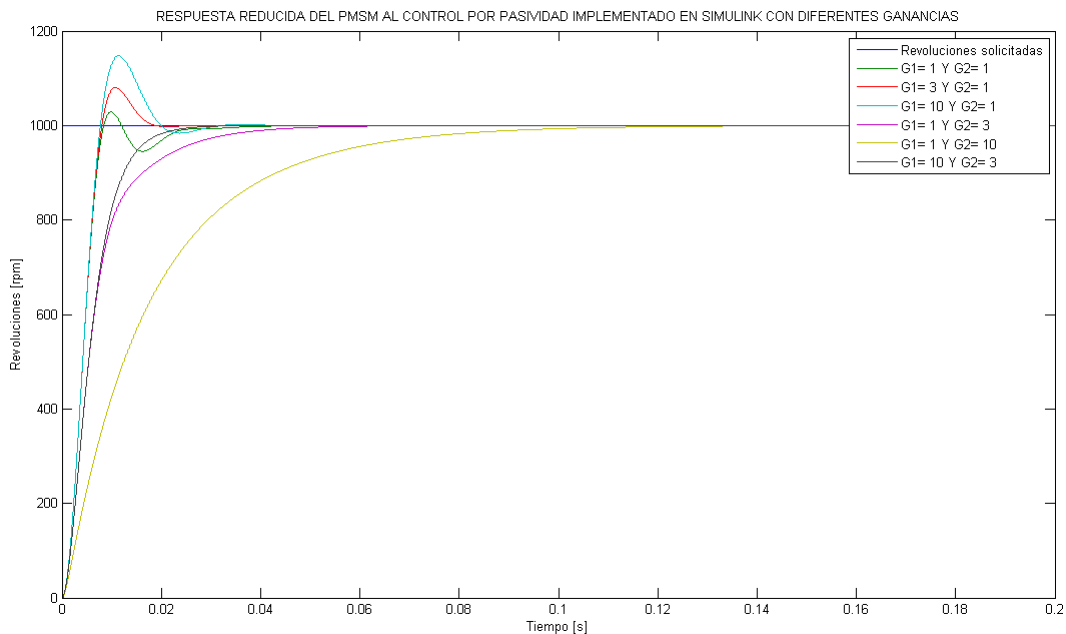


Figura 5.4: Resposta do PMSM ao controle por passividade implementado em simulink com distintos valores de γ_1 y γ_2 con escala de tiempo menor.

Tabla 5.1: Clasificación de las respuesta del PMSM al control por pasividad implementada en simulink.

| γ_1 | γ_2 | Tipo de respuesta. |
|------------|------------|-------------------------|
| 1 | 1 | Sobreamortiguado |
| 3 | 1 | Sobreamortiguado |
| 10 | 1 | Sobreamortiguado |
| 1 | 3 | Amortiguamiento crítico |
| 1 | 10 | Bajo amortiguamiento |
| 10 | 3 | Amortiguamiento crítico |

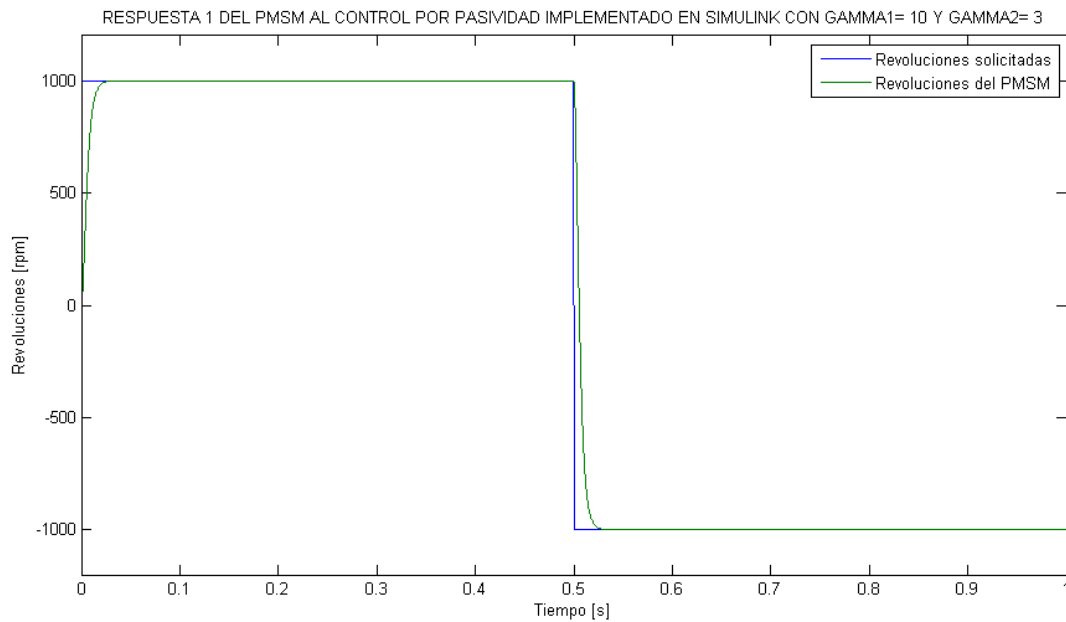


Figura 5.5: Respuesta del PMSM al control por pasividad implementado en simulink con $\gamma_1=10, \gamma_2=3$ para cambio de sentido a 1000 rpm.

Se seleccionan las ganancias $\gamma_1 = 10$ y $\gamma_2 = 3$ para hacer las simulaciones posteriores; Se muestra en la figura 5.5 el resultado de la simulación para una velocidad angular de 1000 rpm y con cambio de sentido de giro.

En la figura 5.6 se muestra la respuesta de motor PMSM para distintas velocidades angulares, con el control implementado en simulink.

La respuesta que se tiene del PMSM al **control por pasividad implementado en simulink** para distintas revoluciones que se solicitan es aceptable y realiza el seguimiento a las revoluciones solicitadas.

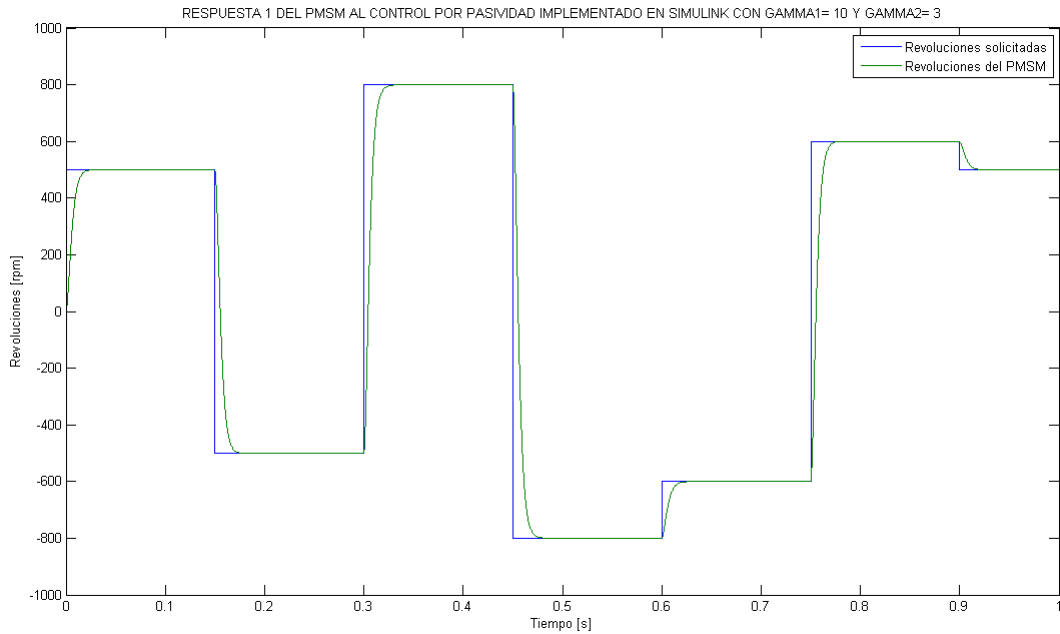


Figura 5.6: Respuesta del PMSM al control por pasividad implementado en simulink con $\gamma_1=10, \gamma_2=3$ para varias revoluciones.

5.1.2. Error del control por pasividad implementado en simulink

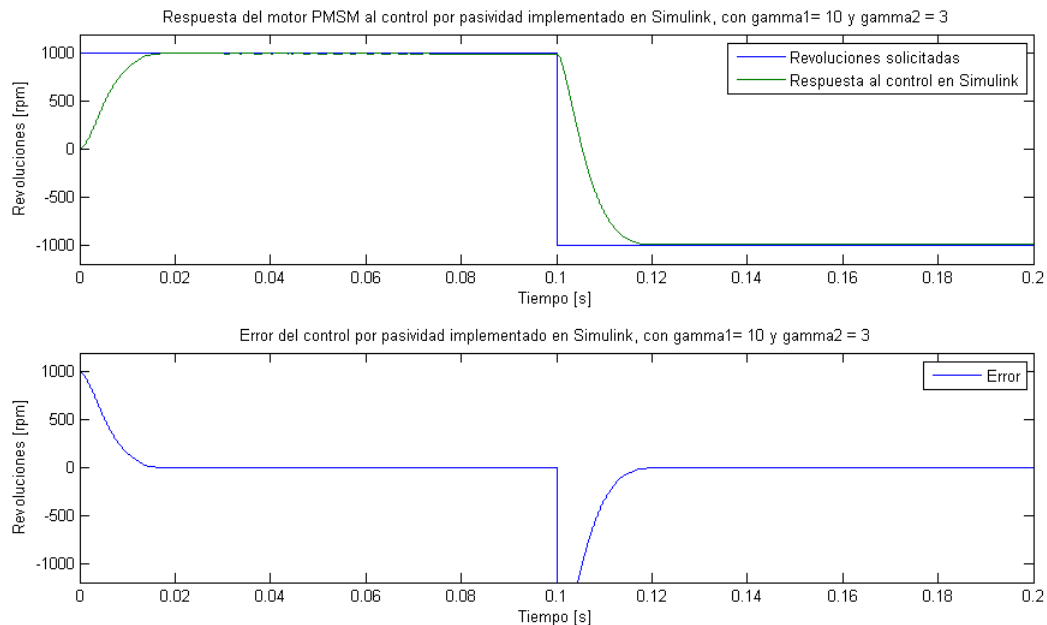


Figura 5.7: Respuesta del PMSM al control por pasividad implementado en simulink con $\gamma_1=10, \gamma_2=3$ y el error.

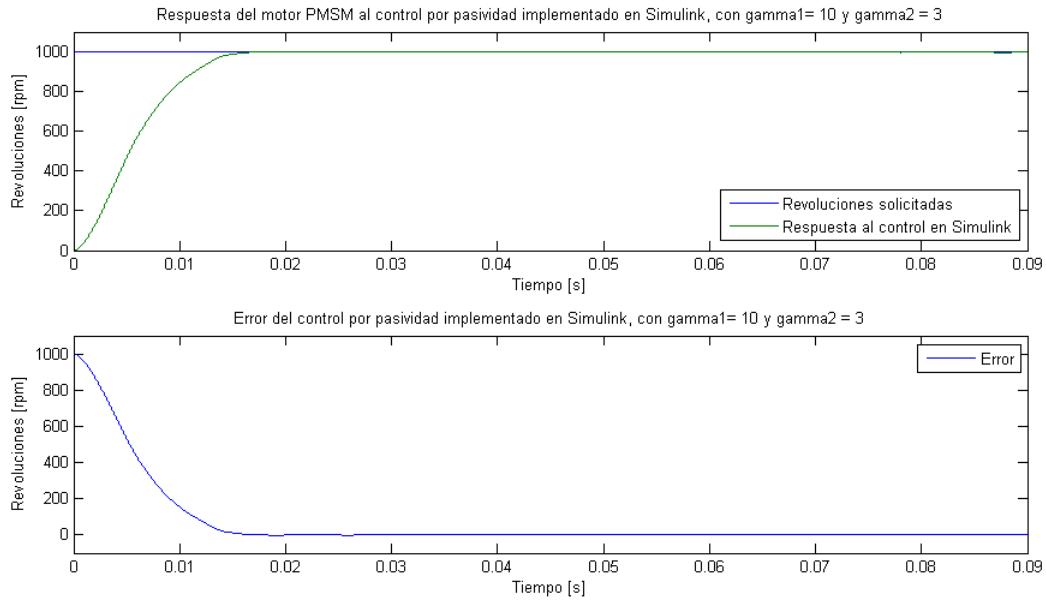


Figura 5.8: Respuesta del PMSM al control por pasividad implementado en simulink con $\gamma_1=10, \gamma_2=3$ y el error con menor tiempo de simulación.

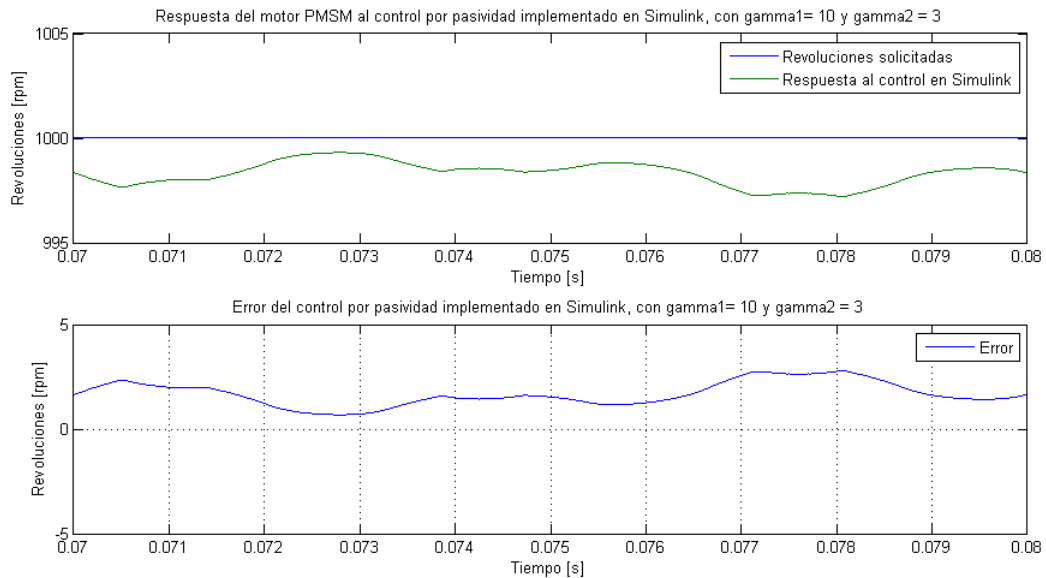


Figura 5.9: Respuesta del PMSM al control por pasividad implementado en simulink con $\gamma_1=10, \gamma_2=3$ y el error en un intervalo de tiempo de simulación.

Al realizar la simulación del **control por pasividad** implementado en simulink también se requiere de obtener el error; el cual, calcula la diferencia entre las revoluciones solicitadas y las revoluciones que alcanza el motor PMSM. En la figura 5.7 se muestra el error que tiene el control implementado en simulink.

Se muestra en la figura 5.8 el resultados anterior con una reducción del tiempo de simulación.

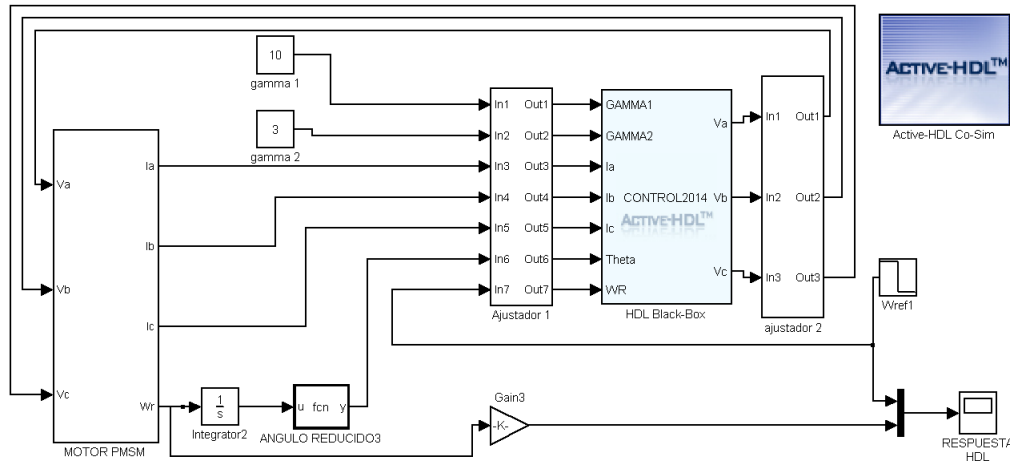


Figura 5.10: Diagrama de bloques en simulink del PMSM y control por pasividad en HDL.

En la figura 5.9 se muestra el error con mayor detalle, sólo se simula de 0.07 s a 0.08 s.

Con respecto a la figura 5.9, en un tiempo de 0.0781 s de simulación el motor PMSM tiene 997.228 rpm, siendo un error de 2.7792 rpm lo que corresponde al 0.277 por ciento de error.

5.2. Cosimulación del control por pasividad en VHDL del PMSM.

El programa simulink tiene la ventaja de poder hacer interconexión con otros programas para realizar una cosimulación; para hacer la conexión entre los programas se debe de agregar los bloques necesarios a la librería de simulink, este proceso se describe en el anexo C. Realizado esta conexión se procede a realizar las cosimulaciones requeridas.

En la plataforma de simulink se implementa el bloque del motor de imanes permanentes y el bloque control por pasividad implementado en HDL; el primero se obtuvo en el capítulo 2, mostrado en la figura 2.6; y el segundo en el capítulo 4, que se muestra en el diagrama de la figura 4.8. Se hace la interconexión entre los bloques de simulink y Active HDL, se realiza la simulación para obtener la respuesta que tiene el motor al ser controlado por pasividad implementado en Active HDL. Además, se adiciona el bloque que hace posible la cosimulación que tiene como título *ACTIVE-HDLTM* y los bloques de ajustador 1 y 2 que son para llevar la señales al formato de la palabra que se está trabajando en **Active HDL** y para regresarlo al formato de simulink. Además se quedan también como entradas a este control las ganancias γ_1 y γ_2 . En la figura 5.10 se muestra el diagrama completo de este sistema.

5.2.1. Cosimulación del control por pasividad en VHDL

Se realiza la cosimulación con las ganancias $\gamma_1=10$ y $\gamma_2=3$, y se le solicita al controlador que el motor gire primero a 1000 rpm a a los 0.1 s cambie de de sentido de giro. En la figura 5.11 se muestra la respuesta.

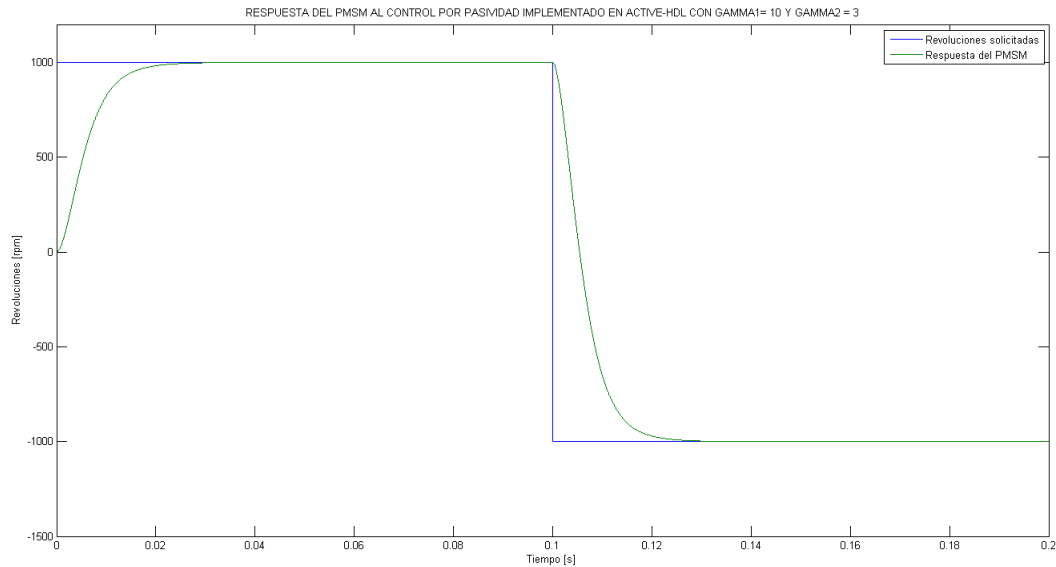


Figura 5.11: Respuesta del PMSM al control por pasividad en HDL.

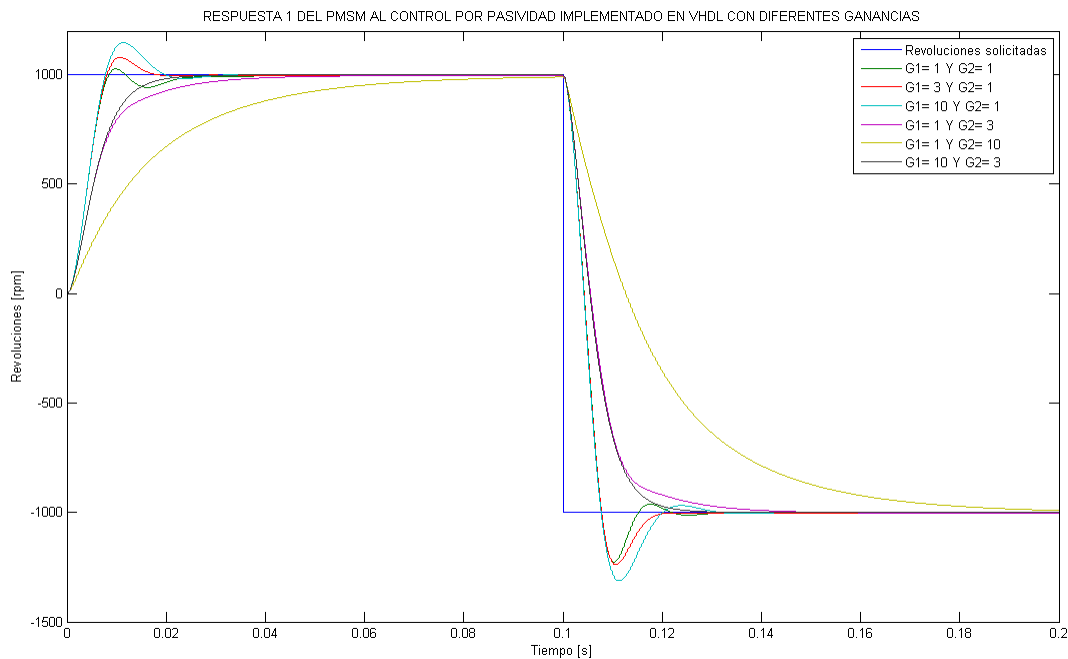


Figura 5.12: Respuesta del PMSM al control por pasividad en VHDL con diferentes γ_1 y γ_2 .

Analizando las figuras 5.5 y 5.11, respuestas que tiene el motor PMSM al control por pasividad implementado en simulink y VHDL respectivamente, se observa que ambos tienen un comportamiento similar.

La respuesta que tiene el motor PMSM a diferentes valores de las ganancias γ_1 y γ_2 se presentan en el anexo D.

En la figura 5.12 se presentan el concentrado de estas respuestas con diferentes ganancias, y se muestra la figura 5.13 las respuestas con un tiempo de simulación de 0,1 s para visualizarlas

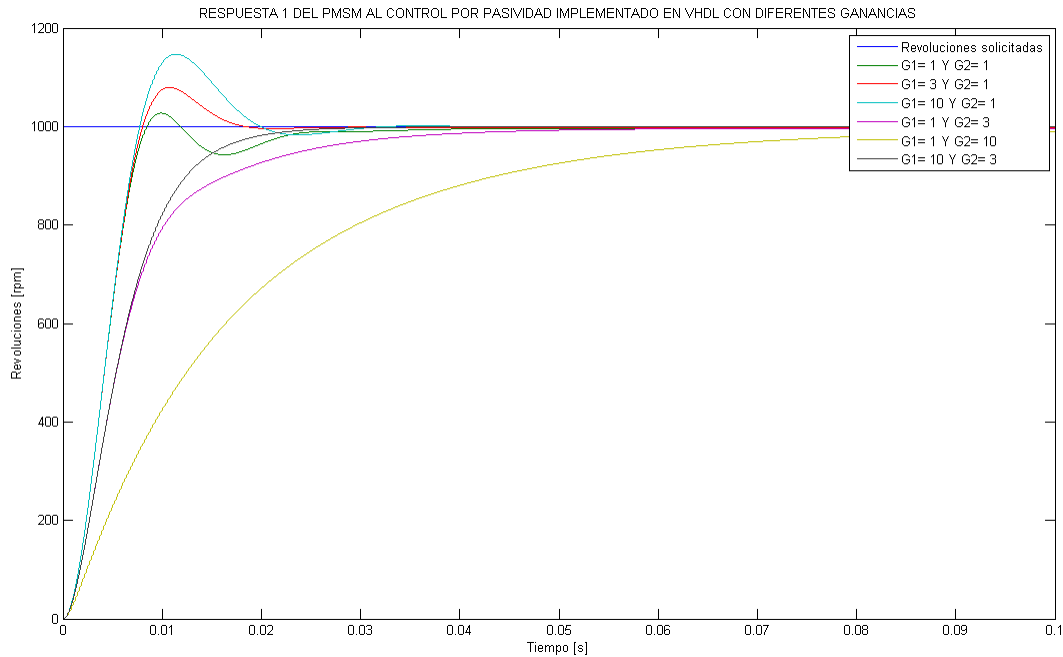


Figura 5.13: Respuesta del PMSM al control por pasividad en VHDL con diferentes γ_1 y γ_2 con menor tiempo de simulación.

Tabla 5.2: Clasificación de las respuesta del PMSM al control por pasividad implementada en Active DHL.

| γ_1 | γ_2 | Tipo de respuesta. |
|------------|------------|-------------------------|
| 1 | 1 | Sobreamortiguado |
| 3 | 1 | Sobreamortiguado |
| 10 | 1 | Sobreamortiguado |
| 1 | 3 | Amortiguamiento crítico |
| 1 | 10 | Bajo amortiguamiento |
| 10 | 3 | Amortiguamiento crítico |

a mayor detalle.

En la tabla 5.2 se muestra clasificación según Kuo [Kuo1996] de la respuesta que se tiene al control por pasividad implementado en **VDHL** para diferentes ganancias.

Realizando una comparación de los datos recopilados de la tabla 5.1 y la tabla 5.2 se observa que la respuesta que tiene el motor PMSM al **control por pasividad** implementado en simulink y en VHDL, respectivamente, tiene el mismo comportamiento.

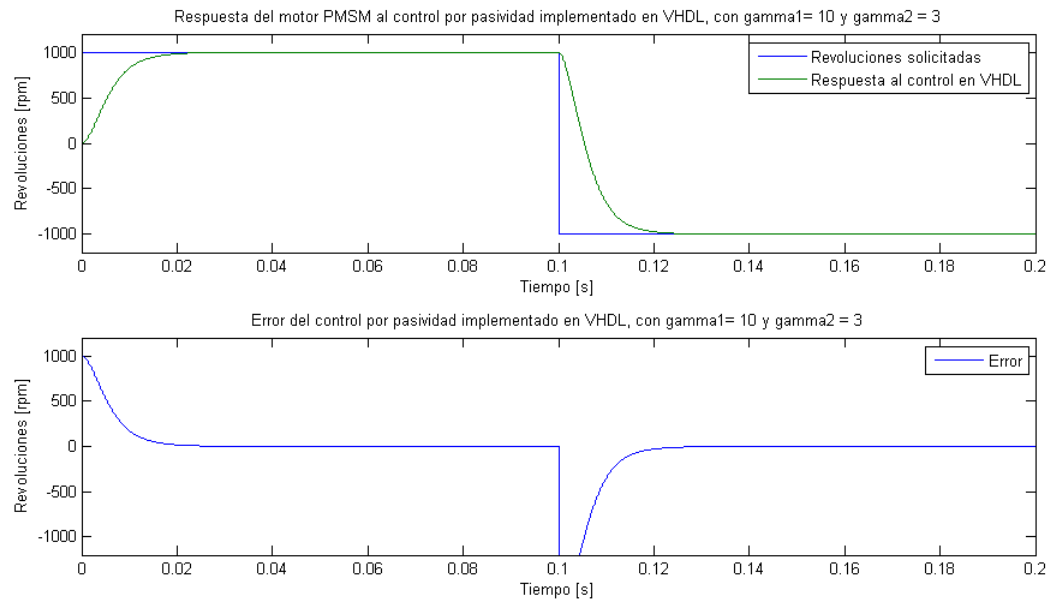


Figura 5.14: Respuesta del PMSM al control por pasividad implementado en VHDL con $\gamma_1 = 10$ y $\gamma_2 = 3$ y el error.

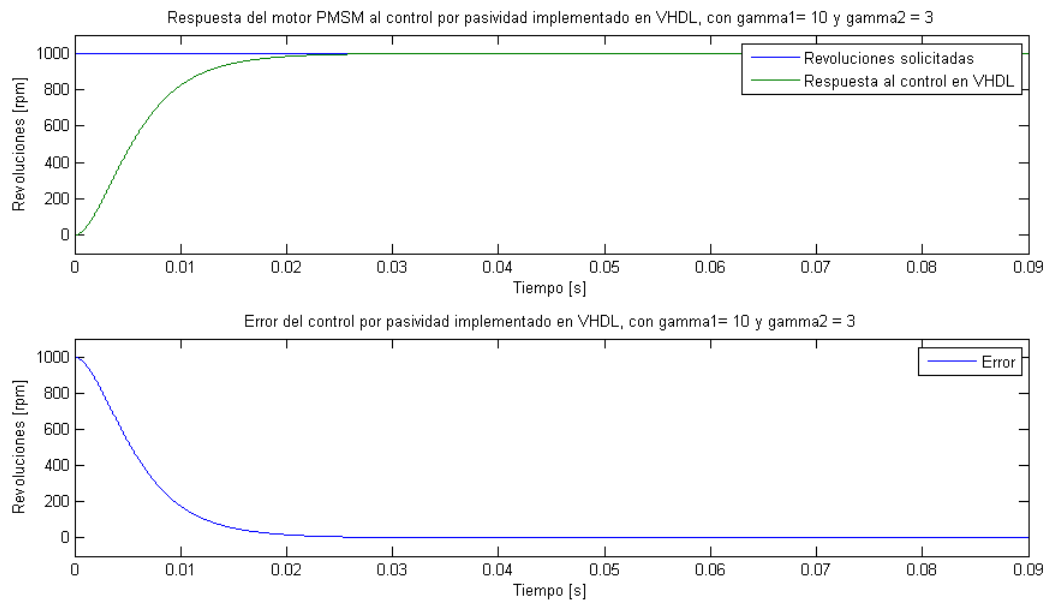


Figura 5.15: Respuesta del PMSM al control por pasividad implementado en VHDL con $\gamma_1 = 10$ y $\gamma_2 = 3$ y el error con menor tiempo de simulación.

5.2.2. Error del control por pasividad implementado en VHDL

En la figura 5.14 es mostrado gráficamente la cosimulación del **control por pasividad** entre simulink y **Active-DHL**, se puede visualizar el error. y en la figura 5.15 se muestra una simulación con reducción de tiempo.

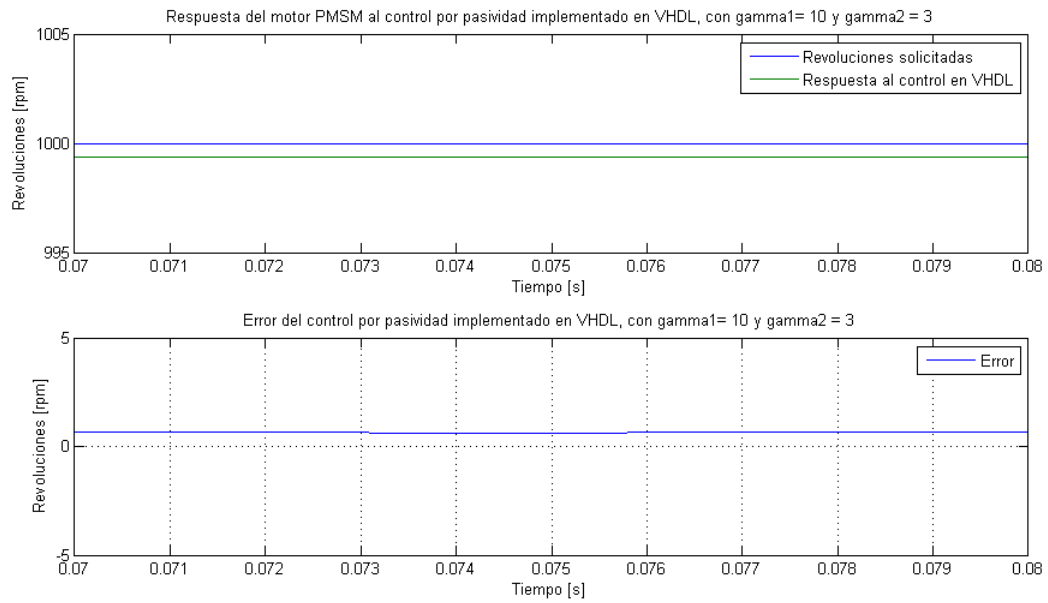


Figura 5.16: Respuesta del PMSM al control por pasividad implementado en VHDL con $\gamma_1 = 10$ y $\gamma_2 = 3$ y el error en un intervalo de tiempo de simulación.

En la figura 5.16 es mostrado el error para una cosimulación entre 0.07 s y 0.08 s.

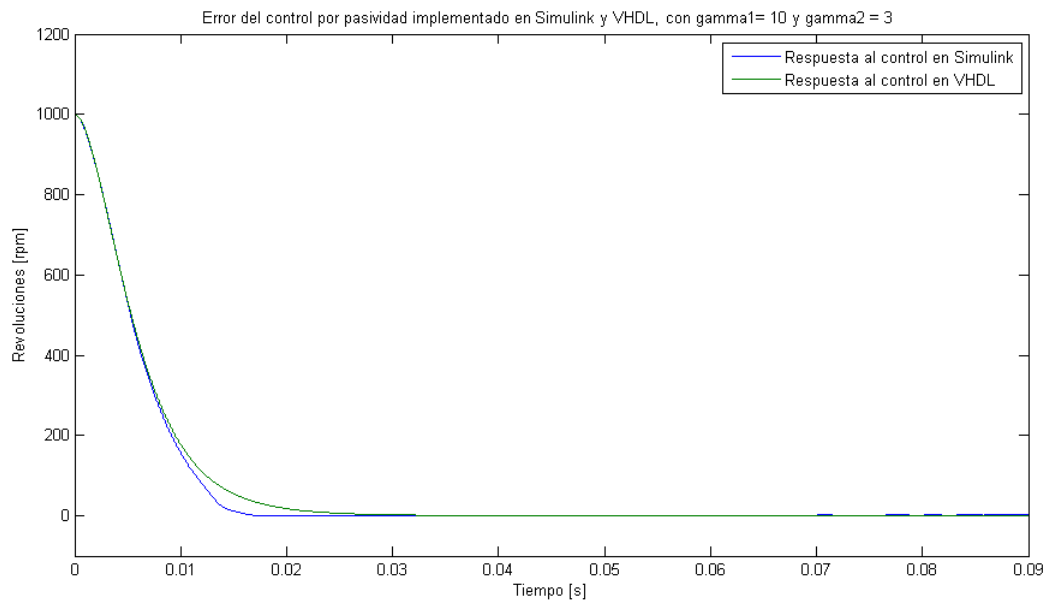


Figura 5.17: Respuesta del PMSM al control por pasividad implementado en simulink y VHDL con $\gamma_1 = 10$ y $\gamma_2 = 3$.

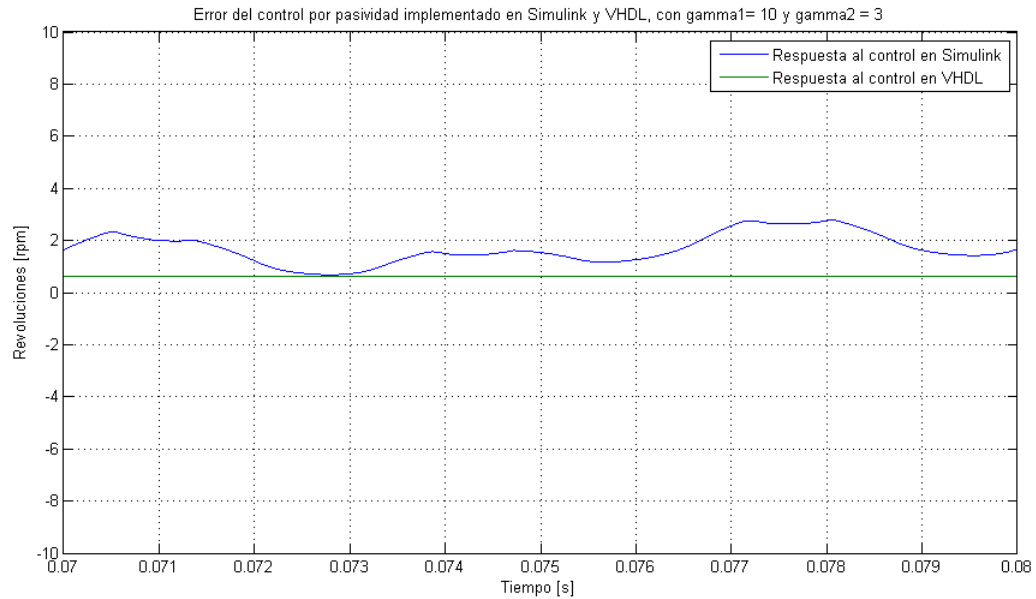


Figura 5.18: Error del control por pasividad implementado en simulink y VHDL con $\gamma_1 = 10$ y $\gamma_2 = 3$.

5.3. Comparación de la respuesta del control implementado en simulink y en Active-HDL.

Es importante realizar una comparación del como es implementado el control por pasividad; en primer lugar, por ser uno de ellos el entregable de este trabajo, por lo que es necesario saber la diferencia de error que presenta cada uno, con los mismos requerimientos y los mismos recursos; y en segundo lugar, que tan diferente es el comportamiento del motor PMSM para cada tipo de implementación.

El error obtenido de las respuestas del motor PMSM al **control por pasividad** implementado en simulink y en **Active-DHL**, con las mismas ganancias $\gamma_1 = 10$ y $\gamma_2 = 3$, se mostraron en las figuras 5.8 y 5.14 respectivamente; estos dos errores son mostrados en la figura 5.17 y en la figura 5.18 se muestra la diferencia de las revoluciones solicitadas y las revoluciones alcanzadas por el motor PMSM por las dos implementaciones del **control por pasividad**.

De la figura 5.18 se inferen los siguientes resultados:

- La respuesta del control por pasividad implementada en VHDL tiene menor error que la implementada en simulink.
- La variación del error es menor en la implementación del control por pasividad en VHDL.
- La implementación realizada en VHDL es correcta por tener un mismo comportamiento que la de simulink

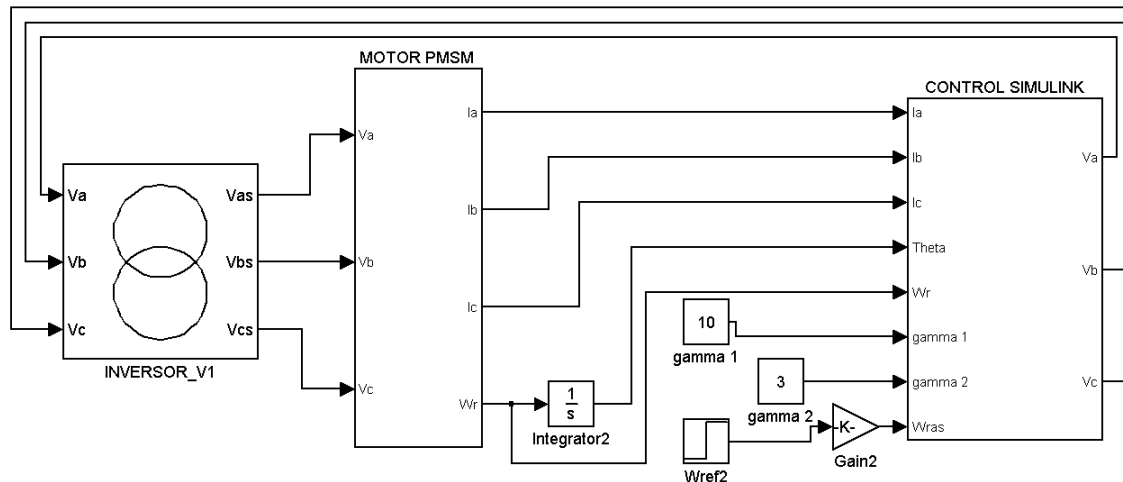


Figura 5.19: Diagrama de bloques en simulink de PMSM, control por pasividad en simulink e inversor en PSIM.

5.4. Análisis de las cosimulaciones del control en simulink del PMSM con inversor en PSIM

En la figura 5.19 se muestra el diagrama de bloques en simulink de PMSM, control por pasividad en simulink e inversor en PSIM, que son todos los elementos que son usados para la implementación del sistema; al diagrama mostrado en la figura 5.1 que fue para obtener la respuesta del control por pasividad de un PMSM se le agrega un bloque más que es el inversor creado en programa PSIM.

En el inversor los ajustes que se le realizan son: el voltaje que se obtienen de las baterías y la frecuencia de la señal triangular; el voltaje se fija en 84 V que serian suministrados por 7 baterías de 12 V conectadas en serie. La señal triangular es utilizada para generar la señal **PWM** para activar los interruptores de potencia para generar el voltaje trifásico para alimentar al PMSM.

5.4.1. Cosimulación del control por pasividad en simulink con inversor en PSIM para distintas frecuencias de la señal triangular

Se realizan las cosimulaciones para diferentes frecuencias de la señal triangular; con el voltaje de 84 V y con las ganancias $\gamma_1 = 10$ y $\gamma_2 = 3$. Los resultados de cada una de estas frecuencias se muestran en el anexo E; las respuestas de los distintos valores de la frecuencias cosimuladas se muestran el figura 5.20.

Se reduce la escala tiempo para observar a más detalle como se comportan las respuestas a diferentes frecuencias, esto esta mostrado en la figura 5.21.

Observando las respuestas que tiene el PMSM a las distintas frecuencias de la señal triangular es seleccionada la frecuencia de 20 kHz para realizar las siguientes pruebas.

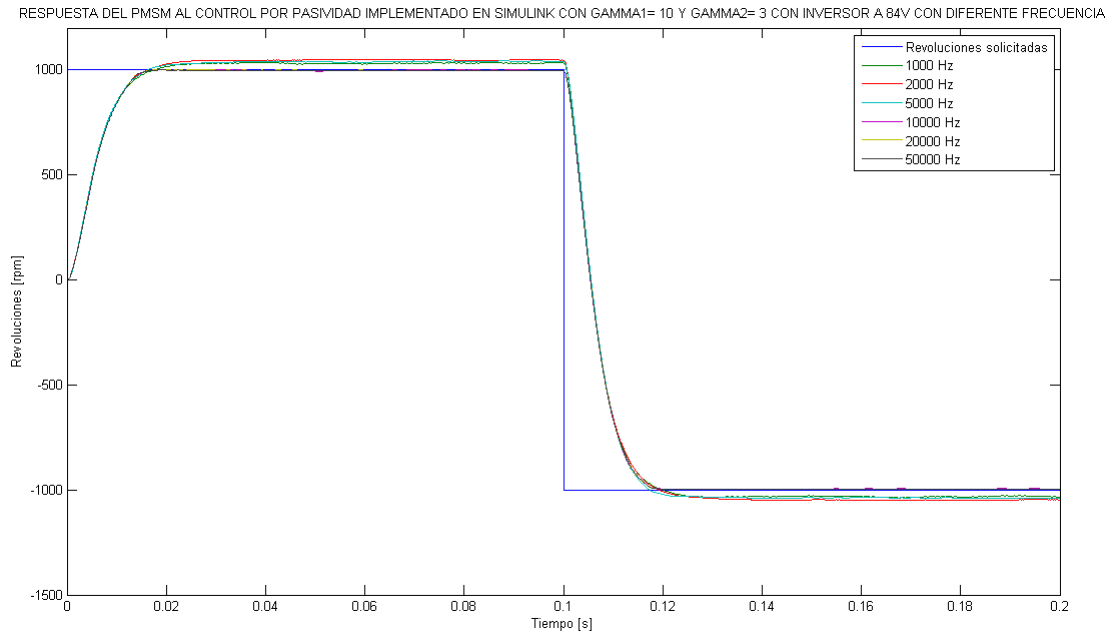


Figura 5.20: Respuesta del PMSM al control por pasividad en simulink y con inversor en PSIM a distintas frecuencias de la señal triangular.

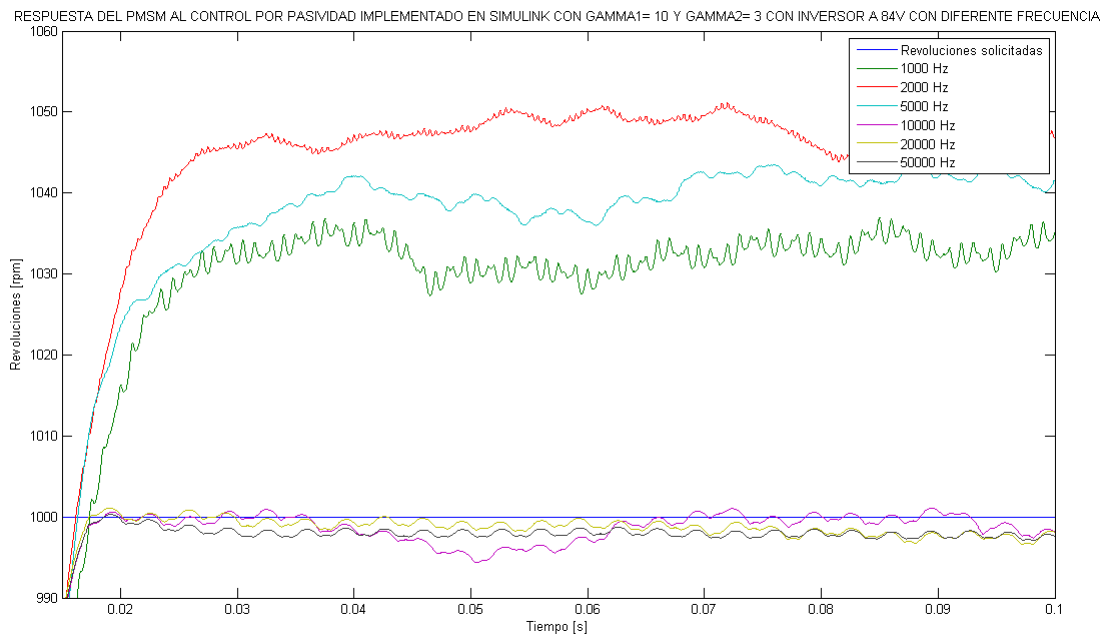


Figura 5.21: Respuesta del PMSM al control por pasividad en simulink y con inversor en PSIM a distintas frecuencias con menor tiempo de cosimulación.

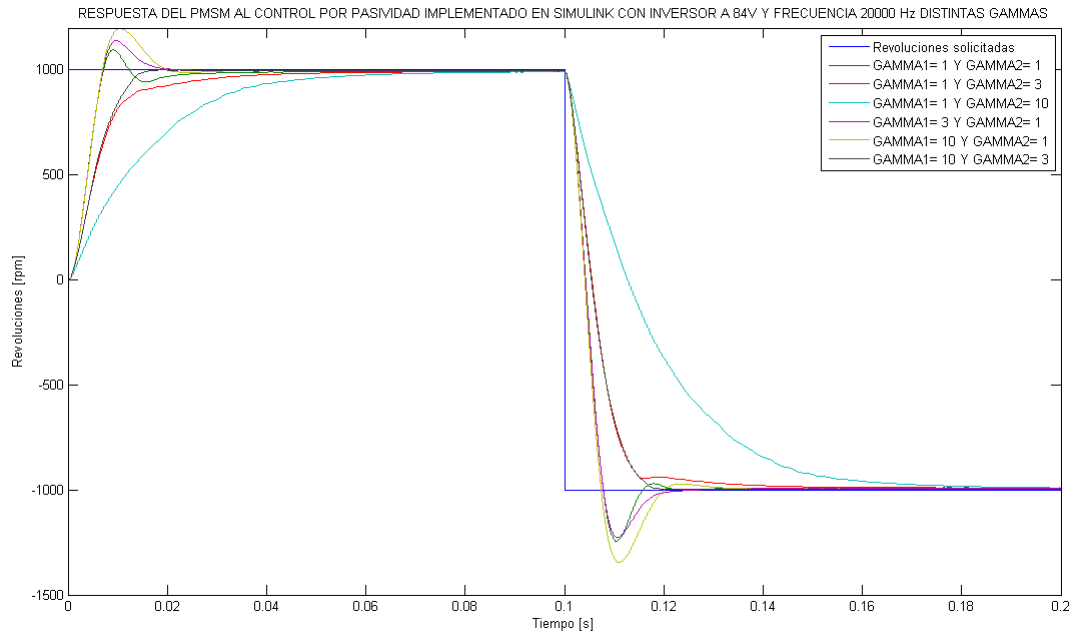


Figura 5.22: Respuesta del PMSM al control por pasividad en simulink y con inversor en PSIM a una frecuencia de 20 kHz con distintas γ_1 y γ_2 .

Tabla 5.3: Clasificación de las respuesta del PMSM al control por pasividad implementada en simulink con inversor en PSIM.

| γ_1 | γ_2 | Tipo de respuesta. |
|------------|------------|-------------------------|
| 1 | 1 | Sobreamortiguado |
| 3 | 1 | Sobreamortiguado |
| 10 | 1 | Sobreamortiguado |
| 1 | 3 | Amortiguamiento crítico |
| 1 | 10 | Bajo amortiguamiento |
| 10 | 3 | Amortiguamiento crítico |

5.4.2. Cosimulación del control por pasividad en simulink con inversor en PSIM a 20 kHz.

En la figura 5.22 se muestra el comportamiento que tiene la respuesta del motor PMSM en la cosimulación con diferente ganancias γ_1 y γ_2 , las respuestas individuales se muestran en el anexo F.

Para observar mejor las respuesta se reduce el tiempo de cosimulación y se muestra en la figura 5.23; con esta gráfica se crea la tabla 5.3 bajo el criterio que plantea Kuo [Kuo1996].

En las gráficas mostradas en la figura 5.4 y en la figura 5.23 se observa que el comportamiento de las respuestas del motor PMSM, las cuales mantienen las mismas tendencias para cada par de valores de las ganancias γ_1 y γ_2 .

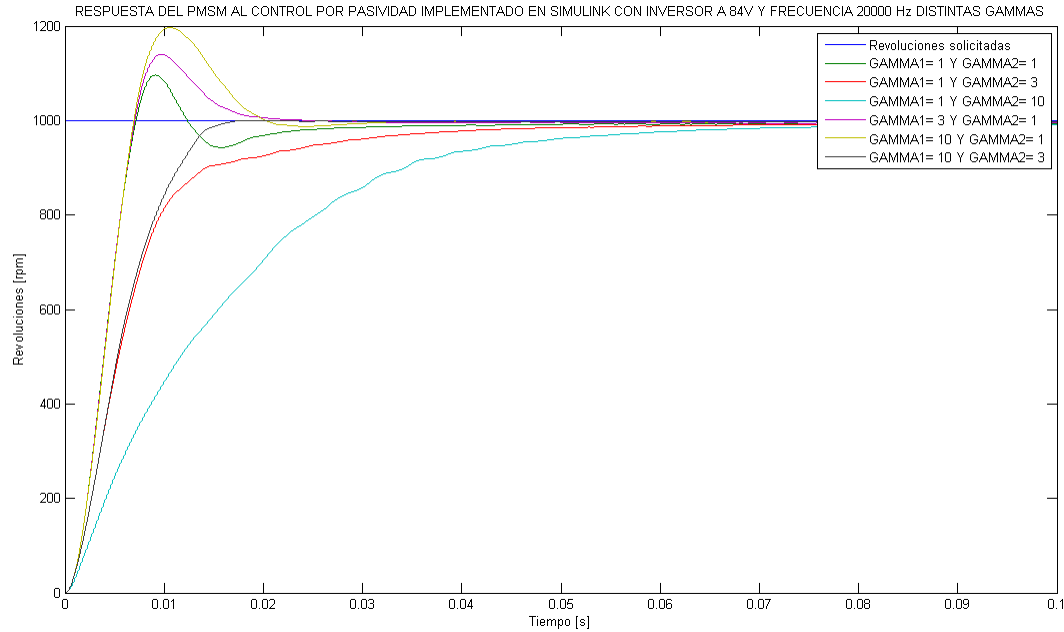


Figura 5.23: Respuesta del PMSM al control por pasividad en simulink y con inversor en PSIM a una frecuencia de 20 kHz con distintas γ_1 y γ_2 con menos tiempo de cosimulación.

Realizando una comparación de la información que presenta la tabla 5.1 y la tabla 5.3, el comportamiento de las respuestas del motor PMSM no cambia con respecto a las mismas ganancias γ_1 y γ_2 ; demostrando que al ingresar al sistema el inversor el **control por pasividad** sigue siendo robusto.

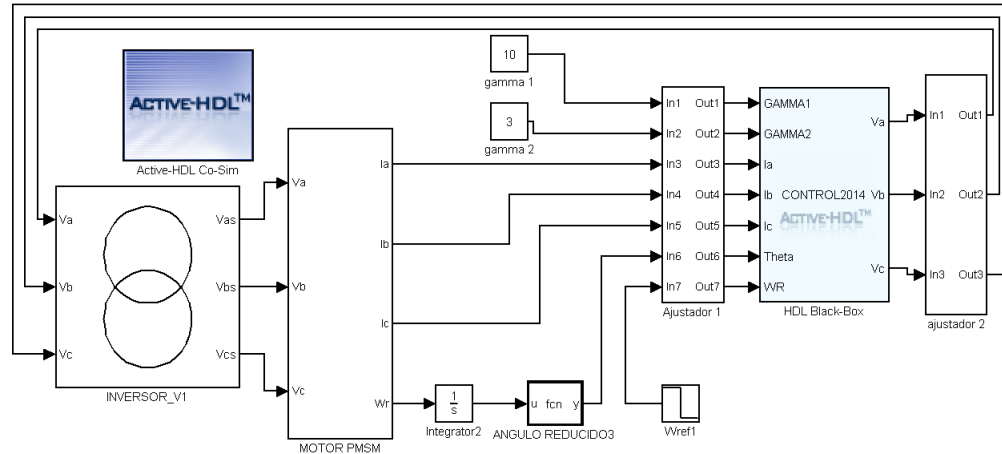


Figura 5.24: Diagrama de bloques en simulink del PMSM y control por pasividad en HDL e inversor en PSIM.

5.5. Análisis de las cosimulaciones del control por pasividad implementado en VHDL del PMSM con inversor en PSIM.

En la figura 5.24 se muestra el diagrama para realizar la cosimulación del control por pasividad implementado en VHDL del PMSM con inversor en PSIM; es obtenido de los diagramas de bloques en simulink del PMSM y control por pasividad en HDL que se muestra en la figura 5.10 y se le agrega el bloque del inversor.

Con el inversor en PSIM con un voltaje de 84 V y una frecuencia de 20 kHz, se realizan las pruebas con el control en VHDL y el motor en simulink con diferentes ganancias γ_1 y γ_2 ; la respuesta de cada una de ellas se muestra en el anexo G.

5.5.1. Cosimulación del control por pasividad en VHDL con inversor en PSIM con 20 kHz de frecuencia de la señal triangular con diferentes ganancias.

Las respuestas que se obtienen con diferentes ganancias con el control en VHDL, con el inversor en PSIM y el motor en PMSM en simulink se muestra en la figura 5.25.

Se realiza una reducción en el tiempo de simulación para observar el comportamiento de las respuestas del motor PMSM; el cual, se muestra en la figura 5.26.

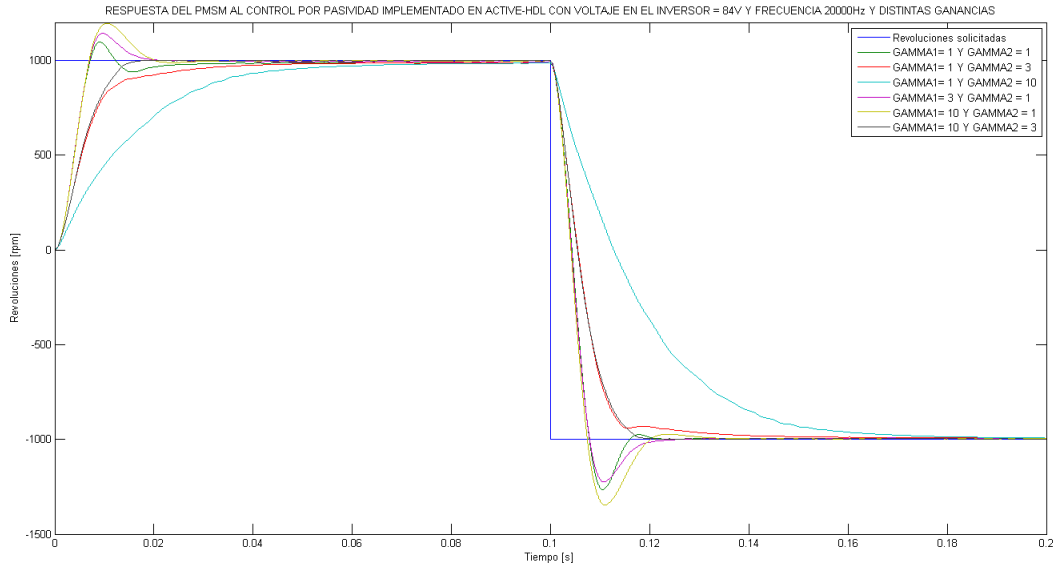


Figura 5.25: Diagrama de bloques en simulink del PMSM y control por pasividad en HDL e inversor en PSIM con diferentes ganancias γ_1 y γ_2 .

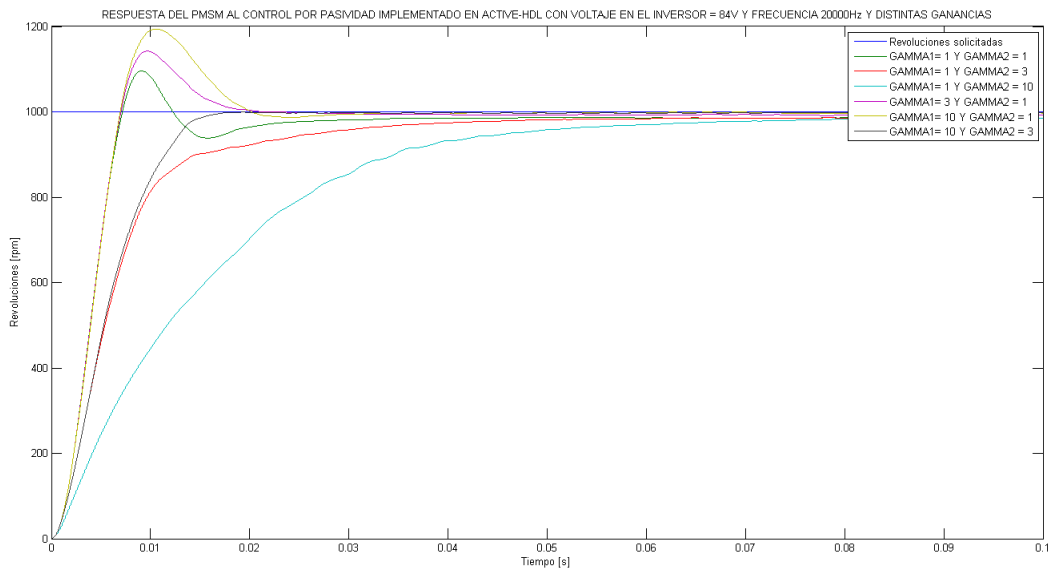


Figura 5.26: Diagrama de bloques en simulink del PMSM y control por pasividad en HDL e inversor en PSIM con diferentes ganancias γ_1 y γ_2 con menor tiempo de simulación.

El comportamiento de las respuestas del motor PMSM para diferentes ganancias γ_1 y γ_2 , que se muestran en las gráficas de la figuras 5.13 y 5.26, mantienen el mismo patrón.

Las respuestas que se observa en la figura 5.26 se clasifican según el autor Kuo [Kuo1996] como se muestra en la tabla 5.4.

Al comparar el tipo de respuesta que tiene el motor PMSM con diferentes ganancias γ_1 y γ_2 que se muestran en las tablas 5.2 y 5.4, sin inversor y con inversor respectivamente, se observa que el comportamiento de cada una mantiene las mismas características que plantea el autor Kuo [Kuo1996].

Tabla 5.4: Clasificación de las respuesta del PMSM al control por pasividad implementada en VHDL con inversor en PSIM.

| γ_1 | γ_2 | Tipo de respuesta. |
|------------|------------|-------------------------|
| 1 | 1 | Sobreamortiguado |
| 3 | 1 | Sobreamortiguado |
| 10 | 1 | Sobreamortiguado |
| 1 | 3 | Amortiguamiento crítico |
| 1 | 10 | Bajo amortiguamiento |
| 10 | 3 | Amortiguamiento crítico |

Tabla 5.5: Resumen de uso FPGA NEXYS 2 al instalar el control por pasividad del PMSM.

| Dispositivo | Nexys 3 | | | |
|--|-----------------|-------------|---------------|-----------------|
| No. Xilinx | xc3s500e-4fg320 | | | |
| Descripción de lógica utilizada | Usados | Disponibles | % Utilización | Observación |
| Número de bloques de Flip Flop | 312 | 9 312 | 3 | |
| Número de LUTs de 4 entradas | 15 642 | 9 312 | 167 | sobre-utilizado |
| Número de bloques | 8 305 | 4 645 | 178 | sobre-utilizado |
| Número de bloques con operaciones lógicas | 8 305 | 8 305 | 100 | |
| Número de bloques con operaciones no lógicas | 0 | 0 | 0 | |
| Número total de LUTs de 4 entradas | 16 403 | 9 312 | 176 | sobre-utilizado |
| Número de LUTs usados para operaciones lógicas | 15 642 | | | |
| Número de LUTs usados para direccionar | 761 | | | |
| Número de conexiones del IOBs | 9 | 232 | 3 | |
| Números de BUFGMUXs | 1 | 24 | 4 | |
| Número de multiplacadores MULT18X18SIOS | 18 | 20 | 90 | |

5.6. Implementación del código VHDL en un dispositivo embebido FPGA.

El programa Xilinx se realiza la compilación y sintetización del código creado para el control por pasividad del PMSM; en éste software se selecciona los dispositivos a utilizar, los cuales para este trabajo son: Nexys 3 y Virtex 5.

En la tabla 5.5 se muestra los recursos usados en el dispositivo Nexys 3 al implementar el control por pasividad; es importante resaltar que la estructura de este equipo no tiene los suficientes elementos para ejecutar el control diseñado; requiere un 76 % más LUTs, es decir 7091.

Tabla 5.6: Resumen de uso FPGA VIRTEX 5 al instalar el control por pasividad del PMSM.

| Dispositivo | Virtex 5 | | | |
|---|-------------------|-------------|---------------|-------------|
| No. Xilinx | xc5clx50t-2ff1136 | | | |
| Descripción de lógica utilizada | Usados | Disponibles | % Utilización | Observación |
| Número de bloques de registro | 321 | 28 800 | 1 | |
| Número usados como Flip Flops | 306 | | | |
| Número usado como Latch-thrus | 15 | | | |
| Número de bloques LUTs | 6 318 | 28 800 | 21 | |
| Número usadas como lógicas | 5 870 | 28 800 | | |
| Número usado con O6 salidas | 5 023 | | | |
| Número usado con O5 salidas | 341 | | | |
| Número usado con O5 y O6 | 506 | | | |
| Número usado como route thru exclusivo | 448 | | | |
| Número LUTs utilizado para ruteo | 782 | | | |
| Número usado O6 salidas solamente | 782 | | | |
| Número de bloques ocupados | 1 933 | 7 200 | 26 | |
| Número de pares LUTs Flip Flop usados | 6 320 | 7 200 | 26 | |
| Número de un Flip Flop sin usar del par útil | 5 999 | 6 320 | 94 | |
| Número de un LUT sin usar | 2 | 6 320 | 1 | |
| Número de pares LUT-FF totalmente usados | 319 | 6 320 | 5 | |
| Número de juegos de solo control | 2 | 6 320 | 5 | |
| Número de bloques de registros perdidos de restricciones de juegos de control | 6 | 28 800 | 1 | |
| Número de conexiones del IOBs | 9 | 480 | 1 | |
| Número de LOCed IOBs | 9 | 9 | 100 | |
| Número de BUFG/BUFGCTRLs | 1 | 32 | 3 | |
| Número de BUFGs | 1 | | | |
| Número de DSP48Es | 24 | 48 | 50 | |
| Conexión media de la red no cronometrada | 3.68 | | | |

En la tabla 5.6 se observa los recursos utilizados por el control por pasividad en el dispositivo Virtex 5; en este equipo se utiliza el 21 % de los bloques LUTs y el 50 % de los DSP48Es ó multiplicadores, además el porcentaje de bloques usados es de 26 y los pares de LUTs Flip Flop es el 26 %.

Para aplicar el código generado para el control por pasividad del PMSM, es necesario contar con un dispositivo de gama media.

5.7. Conclusiones.

Los resultados de las simulaciones realizadas en la plataforma de simulink en cosimulación con los programas Active HDL y PSIM brindan la información para obtener las siguientes conclusiones.

Se realiza el diseño y la implementación del control por pasividad utilizando la plataforma **simulink** para realizar la cosimulación con los programa **Active HDL** y **PSIM**, en donde se prueba el control por pasividad implementado en VHDL y se obtiene las distintas respuestas que clasifica Kuo [Kuo1996], demostrando una implementación exitosa.

En la sintonización de las ganancias del control por pasividad se logro obtener toda la gama de respuesta que propone Kuo [Kuo1996]; por lo que, este control tiene la flexibilidad de proporcionar distintas respuestas. Para una respuesta de amortiguamiento crítico la sintonización de las ganancias quedaron con los valores $\gamma_1=3$ y $\gamma_2=10$, con las que el motor alcanza las revoluciones requeridas en un tiempo menor 0.04 s.

El control por pasividad implementado en simulink realiza el control de las revoluciones del motor PMSM, primero llevando las revoluciones del motor PMSM a las revoluciones solicitadas; segundo, manteniendo estable en el tiempo las revoluciones del motor PMSM.

El control por pasividad implementado en simulink muestra ser robusto al solicitarle diferentes revoluciones del motor PMSM y este los alcanza y lo mantiene estable; con el mismo tipo de respuesta en cada cambio.

Las respuestas de la velocidad del motor PMSM obtenidas en la simulación por el control por pasividad implementado en simulink y en VHDL presentan el mismo comportamiento, comprobando que ambos programas producen el mismo resultado.

Al ingresar el inversor de voltaje a la simulación, con un voltaje de entrada de 84 V y una frecuencia de 20 kHz en la señal triangular, el control por pasividad implementado en simulink mantiene las mismas características de la respuesta del motor PMSM para los distintos valores de las ganancias γ_1 y γ_2 .

En la simulación del control por pasividad implementada en VHDL al ingresar el inversor de voltaje con los mismos parámetros utilizados en la simulación con el control implementado en simulink, se obtiene el mismo comportamiento de las respuestas del motor PMSM con diferentes ganancias γ_1 y γ_2 comparado con la simulación del control por pasividad implementada en VHDL sin inversor.

El código generado en VHDL del control por pasividad para el PMSM es implementado en un FPGA sin problema, como es mostrado en la tabla 5.6; es importante enfatizar que se requiere un mínimo de 15 642 LUTs de cuatro entradas para un FPGA Nexys 2, como se muestra en la tabla 5.5 o su equivalente en el dispositivo correspondiente.

5.8. Publicaciones realizadas

Las publicaciones realizadas como resultado de este trabajo son las siguiente:

- **Braulio Lozano Hernández**, Jacob Vásquez Sanjuan, J. Rangel-Magdaleno, Hugo Ramirez Leyva, " Simulación en Simulink-PSIM del control por pasividad de un motor síncrono de imanes permanentes (PMSM)", IX Semana Nacional de Ingeniería Electrónica, pp. 44-49 Tepic, Nayarit, México. 23-25 Octubre 2013.
- Jacob Vásquez-Sanjuan, **Braulio Lozano Hernández**, J. Rangel-Magdaleno, Ernesto Castellanos-Velasco, " Comparación del control por pasividad y directo de par de un motor de imanes permanentes ", XVI Congreso Latinoamericano de Control Automático, CLCA, Octubre 14-14, Cancún, Quintana Roo, México, 2014.

5.9. Trabajos futuros.

Entre los trabajos futuros que se requieren realizar, estan los siguiente:

- Realizar el código para el encoder motor que se va a utilizar.
- Realizar pruebas con un PMSM, y ajustar las ganancias γ_1 y γ_2

Bibliografía

- [Bose2002] Bimal K. Bose. *Modern Power Electronics and AC Drives*. Prentice-Hall, 2002.
- [Bossoufi2011] B. Bossoufi, M. Karim, A. Lagrioui, and S. Ionita. Fpga-based implementation with simulation of structue direct torque control of a pmsm. In *Multimedia Computing and Systems (ICMCS), 2011 International Conference on*, pages 1–6, April 2011.
- [Bossoufi2011a] B. Bossoufi, M. Karim, I. Silviu, and A. Lagrioui. Fpga-based implementation by direct torque control of a pmsm machine. In *Compatibility and Power Electronics (CPE), 2011 7th International Conference-Workshop*, pages 464–469, June 2011.
- [Burgos2006] R.P. Burgos, P. Kshirsagar, A. Lidozzi, F. Wang, and D. Boroyevich. Mathematical model and control design for sensorless vector control of permanent magnet synchronous machines. In *Computers in Power Electronics, 2006. COMPEL '06. IEEE Workshops on*, pages 76–82, July 2006.
- [CadillacE2016] Cadillac. <http://www.cadillac.com/coupes/elr-coupe.html>, 18:00 hrs 24 of January.
- [California1989] California. Air Resources Board. Office of Strategic Planning. *The Air Pollution-transportation Linkage: How Vehicle Travel on Our Roads and Highways Affects the Air We Breathe : a Report from the State of California, Air Resources Board, Office of Strategic Planning*. Air Resources Board, Office of Strategic Planning, 1989.
- [Carrillo2006] Enrique L. Carrillo Arroyo. *Modelig and simulation of permanent magnet synchronous motor drive system*. PhD thesis, University of Puerto Rico Mayagüez Campus, 2006.
- [Cecati1999] C. Cecati and N. Rotondale. Torque and speed regulation of induction motors using the passivity theory approach. *Industrial Electronics, IEEE Transactions on*, 46(1):119–127, Feb 1999.
- [Chapman2005] S.J. Chapman and C. de Robina Cordera. *Máquinas eléctricas*. McGraw-Hill Interamericana de España S.L., 2005.
- [ChevS2015] Chevrolet. <http://www.chevrolet.com.mx/spark-ev-vehiculo-electrico.html>, 18:00 hrs 24 of January.
- [ChevV2015] Chevrolet. <http://es.chevrolet.com/volt-electric-car.html>, 18:00 hrs 24 of January.

- [Chou2012] Hsin Hung Chou, Ying Shieh Kung, Nguyen Vu Quynh, and Stone Cheng. Optimized fpga design, verification and implementation of a neuro-fuzzy controller for pmsm drives. *Mathematics and Computers in Simulation*, 2012.
- [Duarte2003] Manuel A. Duarte-mermoud and Juan C. Travieso. Control of induction motors: an adaptive passivity mimo perspective, 2003.
- [Dufour2006] C. Dufour, S. Abourida, and J. Be?langer. Real-time simulation of permanent magnet motor drive on fpga chip for high-bandwidth controller tests and validation. In *Industrial Electronics, 2006 IEEE International Symposium on*, volume 3, pages 2591–2596, July 2006.
- [Duran2004] Miguel Angel Duran Fonseca. *Diseño de un controlador no lineal basado en pasividad de un motor síncrono*. PhD thesis, Centro Nacional de Investigación y Desarrollo Tecnológico, 2004.
- [Fitzgerald2003] A.E. Fitzgerald, J. Charles Kingsley, and S.D. Umans. *Electric Machinery. A.E. Fitzgerald, Charles Kingsley, JR., Stephen D. Umans*. McGraw-Hill series in electrical engineering: Power and energy. McGraw-Hill, 2003.
- [Gokdere1997] L.U. Gokdere and M.A. Simaan. A passivity-based method for induction motor control. *Industrial Electronics, IEEE Transactions on*, 44(5):688–695, Oct 1997.
- [Kerk1999] D.W. Schlegel R.J. Kerkman, G.L. Skibinski. Ac drives: Year 2000 (y2k) and beyond. In *In Proc. of APEC99 Conference. Dallas, Texas*, 1999.
- [Kundur1994] P. Kundur, N.J. Balu, and M.G. Lauby. *Power system stability and control*. EPRI power system engineering series. McGraw-Hill, 1994.
- [Kuo1996] B.C. Kuo. *Sistemas de control automático*. Prentice Hall Hispanoamericana, 1996.
- [Langarica2010] Diego Langarica Córdoba. *Control de un Motor Brushless para Aplicación a Vehículos Eléctricos*. PhD thesis, Centro Nacional de Investigación y Desarrollo Tecnológico, 2010.
- [Llor2003] A.M. LLor. *Control directo de par a frecuencia de modulación constante de motores síncronos de imanes permanentes: Commande directe de couple a fréquence de modulation constante des moteurs synchrones a aimants permanents*. 2003.
- [Monmasson2011] E. Monmasson, L. Idkhajine, and M. W Naouar. Fpga-based controllers. *Industrial Electronics Magazine, IEEE*, 5(1):14–26, March 2011.
- [Nicklasson1997] Per J. Nicklasson, Romeo Ortega, and Gerardo Espinosa-Perez. Passivity-Based Control of a Class of Blondel-Park Transformable Electric Machines. *Modeling, Identification and Control*, 18(4):273–305, 1997.
- [Ortega1998] Ortega. *Passivity-based control of Euler-Lagrange systems: mechanical, electrical and electromechanical applications*. Springer, 1998.
- [Pacheco1994] Maria del Carmen Pacheco Arteaga. *Técnicas de Control Basadas en Pasividad aplicadas a un Motor de Inducción*. PhD thesis, Universidad Autonoma de Nuevo Leon, 1994.

- [Palafox2009] Gerardo Israel Palafox Alvarado. *Diseño y construcción de un vehículo eléctrico con variador de velocidad mediante un convertidor CD-CD*. PhD thesis, Universidad Tecnológica de la Mixteca, 2009.
- [Ramirez2011] F.H. Ramírez Leyva; E. Peralta Sánchez; J. J. Vásquez San Juan. Simulación en simon del control directo de par de un motor de imanes permanentes. page 13, 2011.
- [Ramirez2011sp] F.H. Ramírez Leyva; J. J. Vásquez San Juan; E. Peralta Sánchez. Control de velocidad por pasividad de un motor de imanes permanentes y su simulación en simnon. page 11, 2011.
- [Rodríguez2002] Matías . Rodríguez Arnal. *Estimación de posición y control simplificado de corriente para motores BLDC, usando tecnología DSP*. PhD thesis, Pontificia Universidad Católica de Chile. Escuela de Ingeniería., 2002.
- [Sandre2014] O. Sandre-Hernandez, J.J. Rangel-Magdaleno, and R. Morales-Caporal. Simulink-hdl cosimulation of direct torque control of a pm synchronous machine based fpga. In *Electrical Engineering, Computing Science and Automatic Control (CCE), 2014 11th International Conference on*, pages 1–6, Sept 2014.
- [SwKa2002] Dariusz Swierczynski and Marian P Kazmierkowski. Direct torque control of permanent magnet synchronous motor (pmsm) using space vector modulation (dvc-svm)-simulation and experimental results. In *IECON 02 [Industrial Electronics Society, IEEE 2002 28th Annual Conference of the]*, volume 1, pages 751–755. IEEE, 2002.
- [Taoutaou2000] D. Taoutaou and R. Castro-Linares. *A controller-observer scheme for induction motors based on passivity feedback equivalence and sliding modes*. Int. J. Adapt. Control Signal Process., 2000.
- [Vindel2011] David Vindel Muñoz. *Design, Simulation and Implementation of a PMSM Drive System*. PhD thesis, Chalmers University of Technology, 2011.
- [Wikamowski2011] B.M. Wikamowski and J.D. Irwin. *The Industrial Electronics Handbook: Power electronics and motor drives*. The Industrial Electronics Handbook. CRC PressINC, 2011.
- [Ying2011] Ying-Shieh Kung, Nguyen Vu Quynh, Chung-Chun Huang, and Liang-Chiao Huang. Simulink/modelsim co-simulation of sensorless pmsm speed controller. In *Industrial Electronics and Applications (ISIEA), 2011 IEEE Symposium on*, pages 24–29, Sept 2011.
- [Zhong1997] Li Zhong, Mr F Rahman, W Yetal Hu, and KW Lim. Analysis of direct torque control in permanent magnet synchronous motor drives. *Power Electronics, IEEE Transactions on*, 12(3):528–536, 1997.
- [maxinez2002] D.G. Maxinez and J. Alcalá. *VHDL: el arte de programar sistemas digitales*. Compañía Editorial Continental, 2002.
- [ortega2005] Romeo Ortega and G Espinoza-Pérez. Passivity-based control with simultaneous energy-shaping and damping injection: The induction motor case study. In *Proceedings of the 16th IFAC World Congress*, pages 477–482. Elsevier Science, 2005.

[shriwastava2014] Mr RG Shriwastava, MB Diagavane, and SR Vaishnav. Design and testing of pmsm drive for automotive industry. *system*, 3(3), 2014.

Parte I
ANEXOS

Anexo A

Implementación detallada del PMSM en Simulink.

Las ecuaciones 2.11 representan al PMSM, mostradas en el capítulo 2; se presentan ahora con el número de ecuación **ecu-A-1**, para visualizar rápidamente los bloques que a continuación se describen.

$$\begin{bmatrix} L_d \frac{di_d}{dt} \\ L_q \frac{di_q}{dt} \\ \frac{2}{3} \frac{J}{P^2} \frac{d\omega_r}{dt} \end{bmatrix} = \begin{bmatrix} \omega_r L_q i_q - R i_d + v_d \\ -\omega_r (L_d i_d + \Psi_{PM}) - R i_q + v_q \\ \Psi_{PM} i_q - \frac{2}{3P} (T_L + \frac{B\omega_r}{P}) \end{bmatrix} \quad (\text{ecu-A-1})$$

Donde:

v_d y v_q son voltajes en el eje de referencia dq [V]

i_d y i_q son corrientes en el eje de referencia dq . [A]

L_d y L_q son la inductancias.

R es la resistencia del estator [Ω].

Ψ_{PM} es la magnitud del flujo de los imanes permanentes [Wb].

ω_r es la velocidad eléctrica del motor [$\frac{rad}{s}$].

T_L es el par de carga [Nm].

J es el momento de inercia [Nm].

B es la fricción viscosa [Ns]

P es el número de par de polos.

En esta ecuación las entradas son v_d , v_q , L_d , L_q , R , Ψ_{PM} , T_L , J , B y P , y las salidas son i_d , i_q y ω_r .

Para obtener a v_d y v_q que son voltajes en el eje de referencia dq , se utiliza la transformada de Clark y Park; las ecuaciones son 2.3 y 2.4, del capítulo 2; y se vuelven a escribir con el número **ecu-A-2** y **ecu-A-3**.

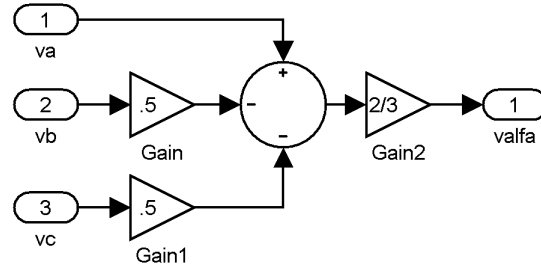


Figura A.1: Transformacion de Clark para v_α .

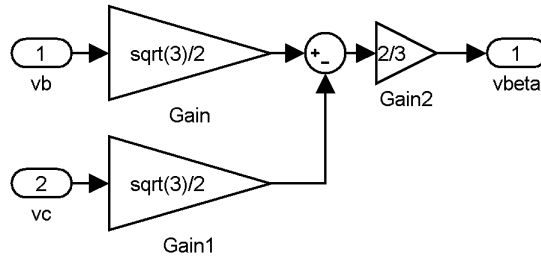


Figura A.2: Transformacion de Clark para v_β .

$$\begin{bmatrix} v_\alpha \\ v_\beta \end{bmatrix} = \frac{2}{3} \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} v_a \\ v_b \\ v_c \end{bmatrix} \quad (\text{ecu-A-2})$$

$$\begin{bmatrix} v_d \\ v_q \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} v_\alpha \\ v_\beta \end{bmatrix} \quad (\text{ecu-A-3})$$

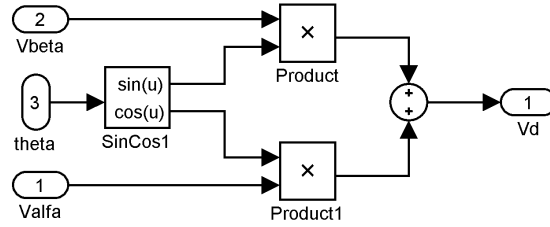
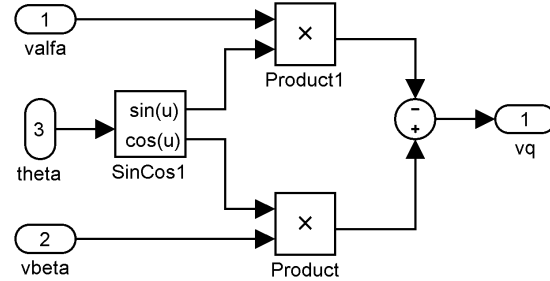
Donde:

v_d y v_q son voltajes en el eje de referencia dq .

v_a , v_b y v_c son voltajes de la linea trifásica.

θ desplazamiento angular de los ejes del estator y del rotor..

En las figuras A.1 y A.2 se visualiza la implementación en Simulink de las ecuaciones **ecu-A-2**, estas para obtener a v_α y v_β respectivamente. Estas transformaciones están en el bloque coner Clark de la figura A.11.

Figura A.3: Transformacion de Park para v_d .Figura A.4: Transformacion de Park para v_q .

En las figuras A.3 y A.4 se muestran la implementación de las ecuaciones **ecu-A-3** para obtener a v_d y v_q respectivamente; las cuales son el bloque conver Park de la figura A.11.

Con los valores de v_d y v_q y las constantes del motor de la tabla 2.2 del capítulo 2 se construye el diagrama de bloques en Simulink para obtener i_d , i_q y ω_r , aplicando las ecuaciones **ecu-A-1**; las relaciones matemáticas se implementaron como se muestran en las figuras A.5, A.6 y A.7; Cada una de estos bloques está representado dentro del bloque del PMSM de la figura A.11.

Con los valores de i_d y i_q y aplicando la inversa de la transformada de Clark y Park obtenemos i_a , i_b y i_c ; su implementación en Simulink de cada una de estas inversas se muestran en las figuras A.8, A.9 y A.10, respectivamente; estos bloques están representados en la figura A.11.

Finalmente se crea el subsistema del bloque de la figura A.11 y queda el bloque del motor PMSM que se muestra en la figura A.12.

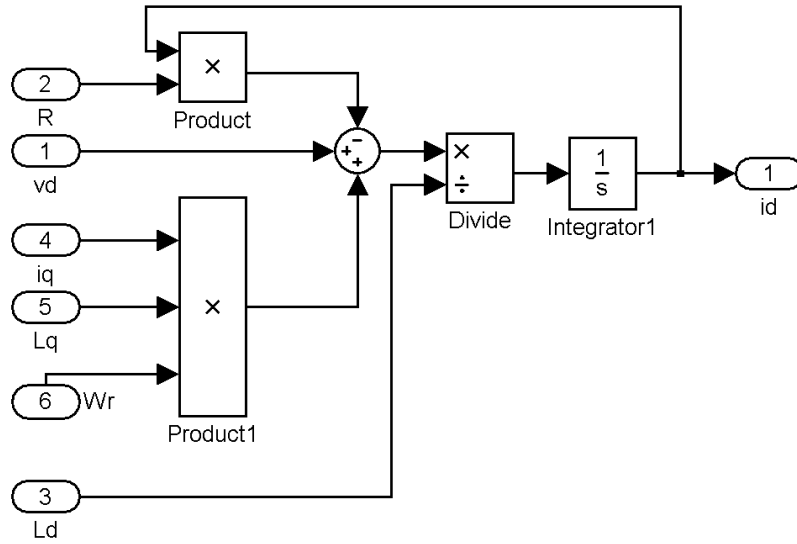


Figura A.5: Bloque en Simulink para obtener I_d

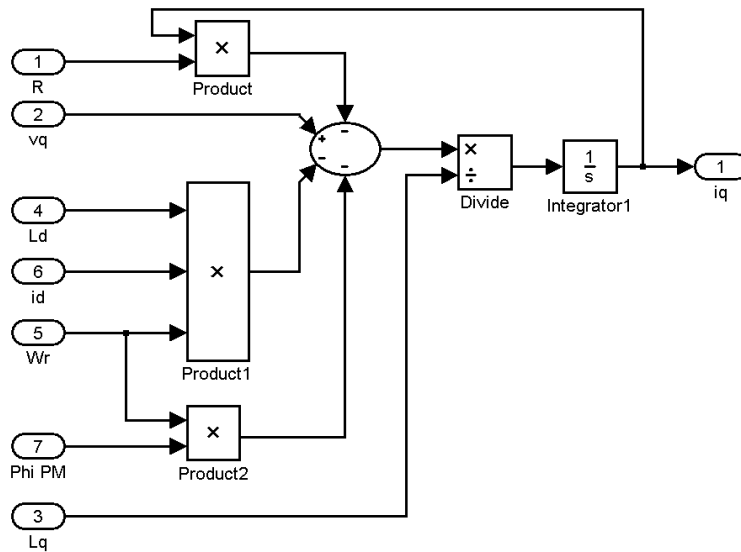


Figura A.6: Bloque en Simulink para obtener I_q

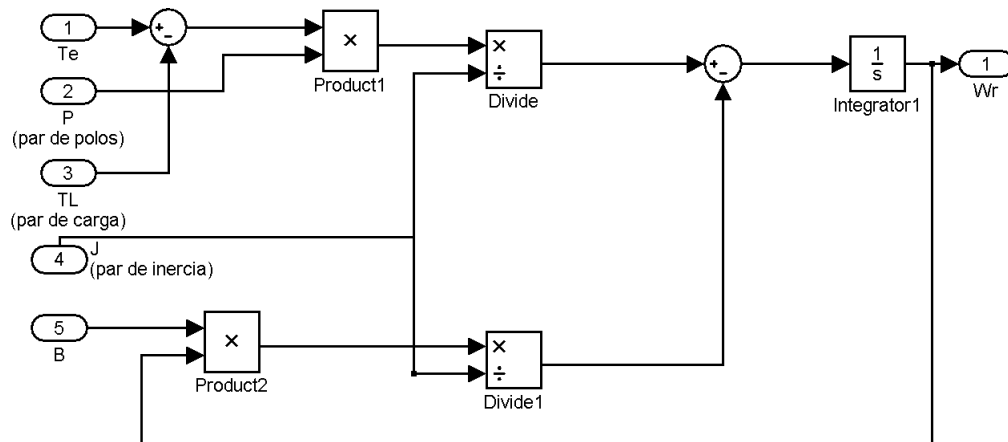


Figura A.7: Bloque en Simulink para obtener W_r

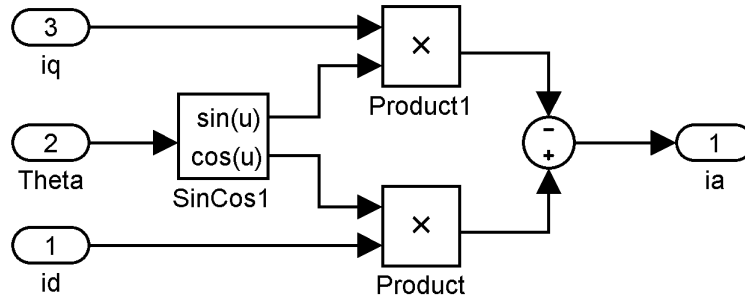


Figura A.8: Bloque en Simulink de inversa de Clark y Park para obtener I_a

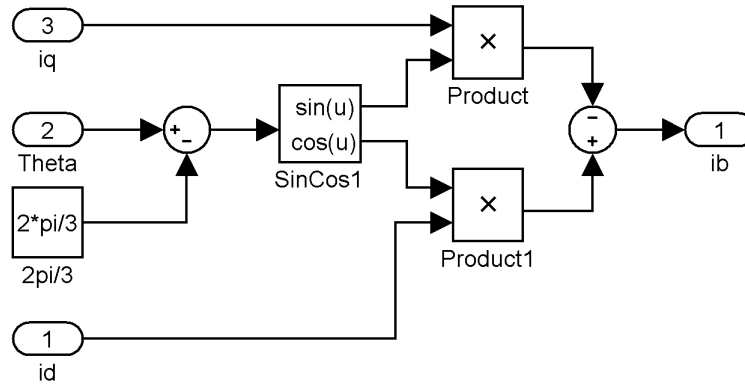


Figura A.9: Bloque en Simulink de inversa de Clark y Park para obtener I_b

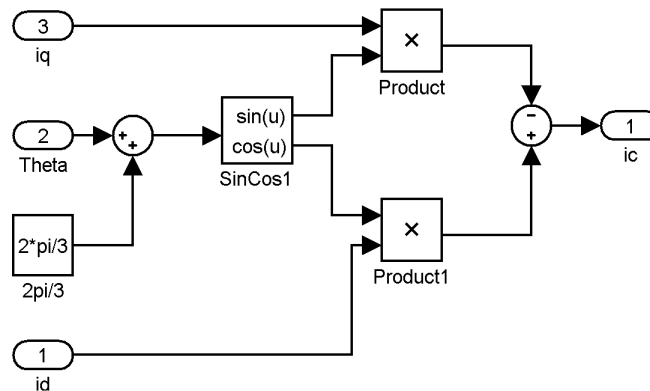


Figura A.10: Bloque en Simulink de inversa de Clark y Park para obtener I_c

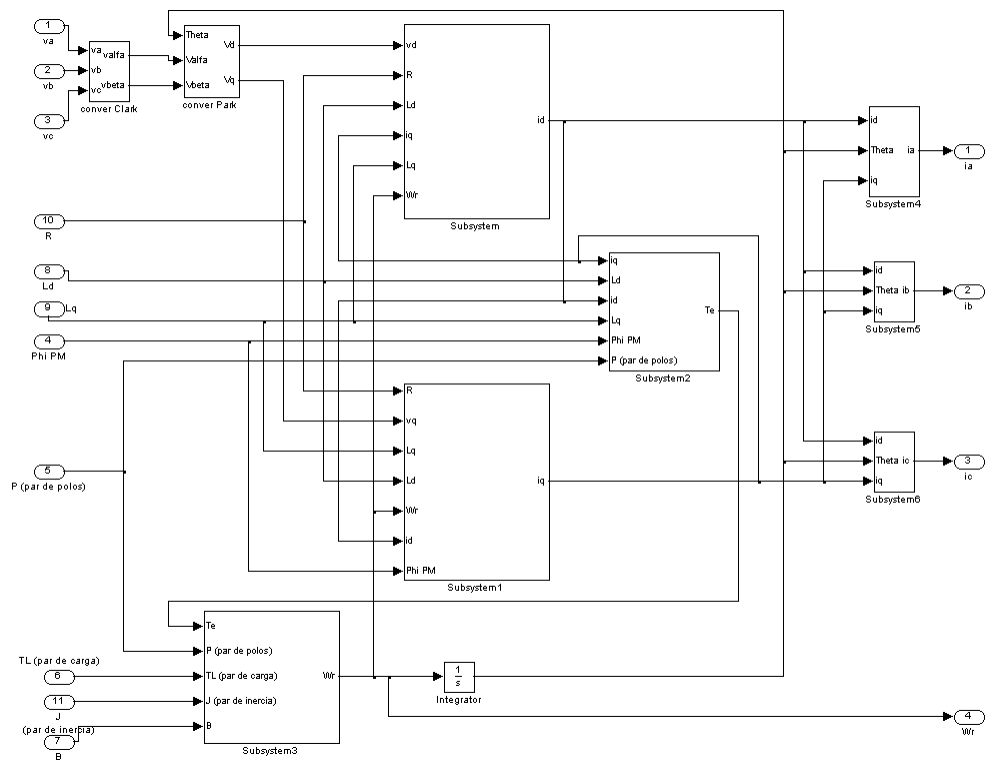


Figura A.11: Interior de bloque de PMSM1 en Simulink.

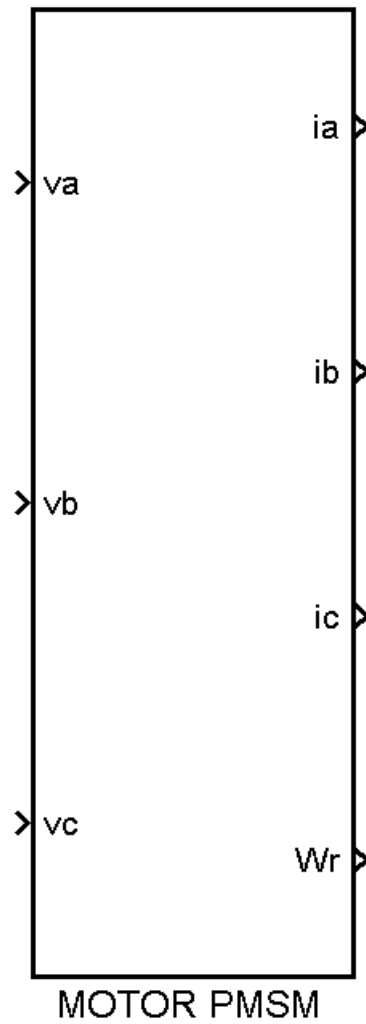


Figura A.12: Bloque en Simulink de PMSM con parámetros incluidos.

Anexo B

Respuesta del control en Simulink para diferentes ganancias.

La respuesta que tiene el PMSM al control por pasividad implementado en Simulink para las ganancias γ_1 y γ_2 varia con forme a los valores de estas; Se realiza simulaciones con distintos par de valores y se obtiene las siguientes gráficas que se muestran a continuación:

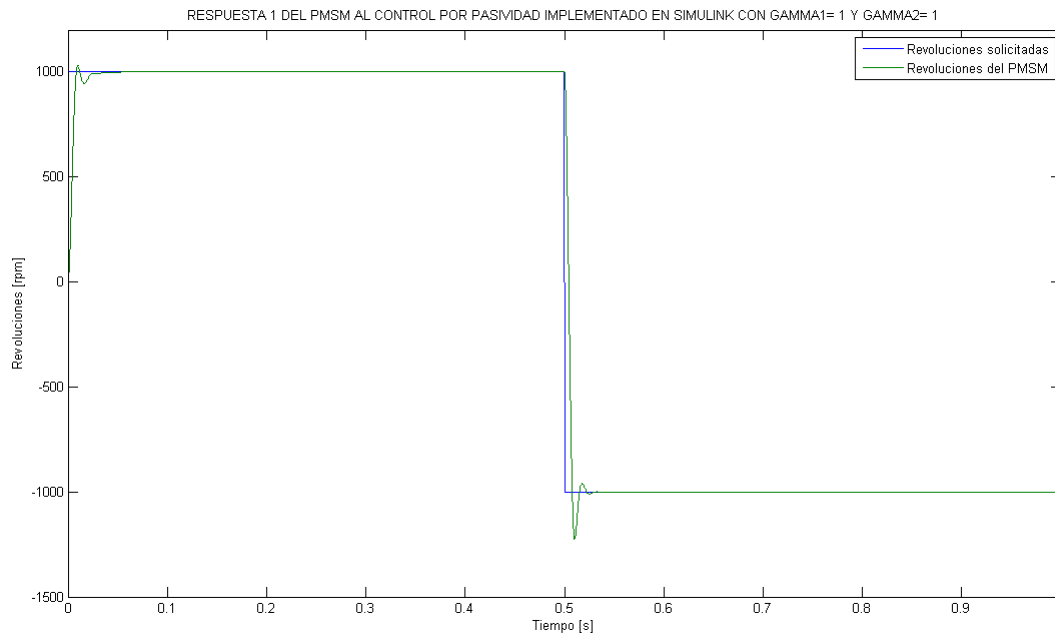


Figura B.1: Respuesta del PMSM al control por pasividad implementado en Simulink con $\gamma_1=1, \gamma_2=1$ para cambio de sentido a 1000 rpm.

En la figura B.1 se muestra la respuesta con $\gamma_1=1$ y $\gamma_2=1$, se observa que es una condición de sobreamortiguamiento [Kuo1996], por lo que tenemos sobre giros del motor.

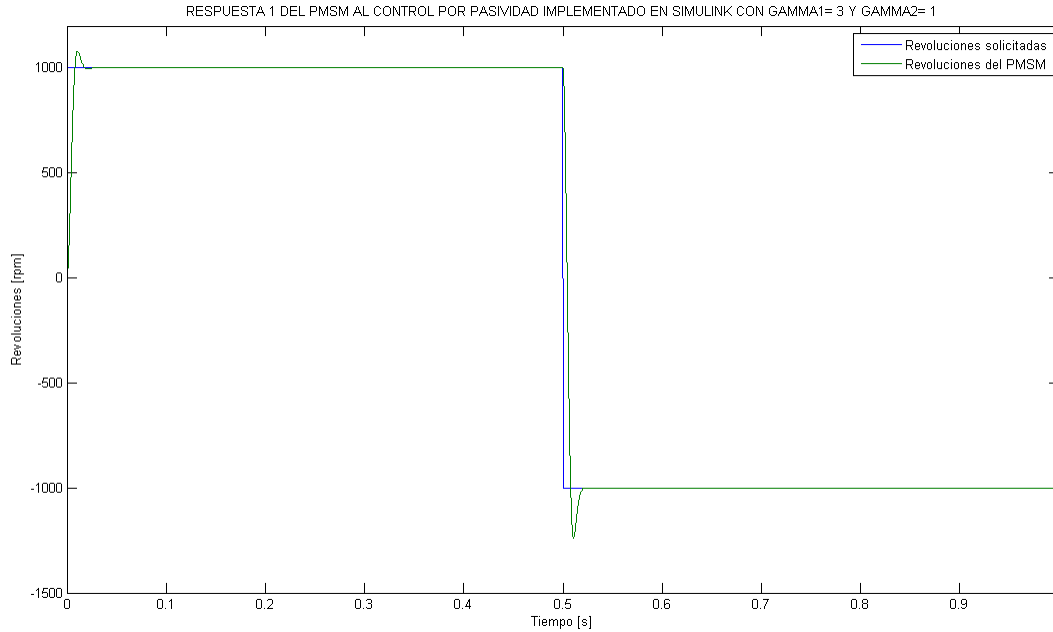


Figura B.2: Respuesta del PMSM al control por pasividad implementado en Simulink con $\gamma_1=3, \gamma_2=1$ para cambio de sentido a 1000 rpm.

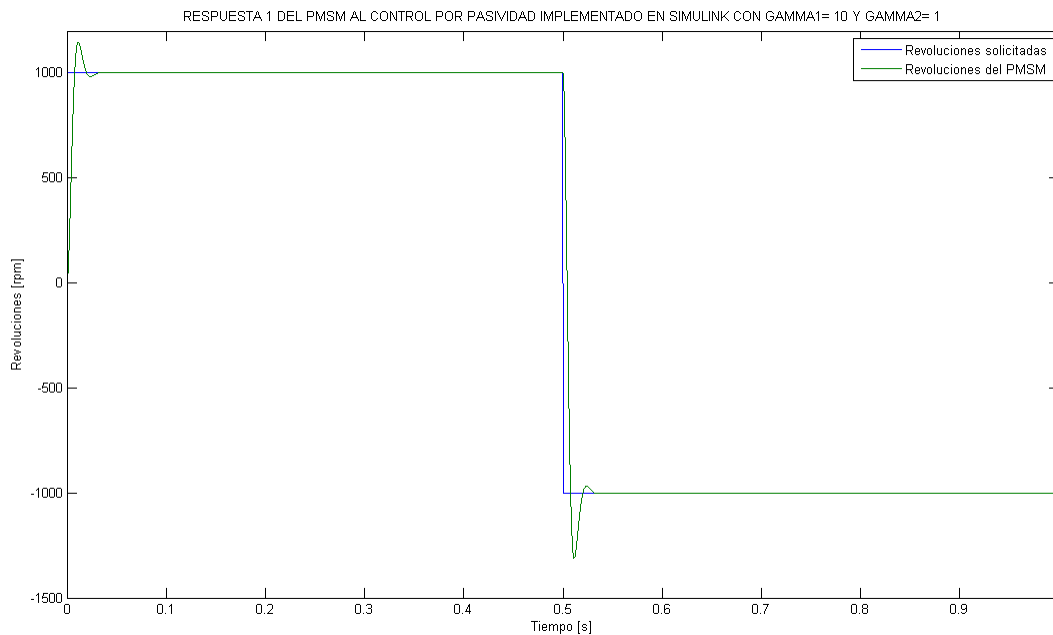


Figura B.3: Respuesta del PMSM al control por pasividad implementado en Simulink con $\gamma_1=10, \gamma_2=1$ para cambio de sentido a 1000 rpm.

En la figura B.2 se observa otra respuesta del PMSM que se obtiene con los valores de $\gamma_1=3$ y $\gamma_2=1$, se presenta un sobreamortiguamiento [Kuo1996].

En la figura B.3, se observa la respuesta de PMSM con las ganancias $\gamma_1=10, \gamma_2=1$, también se puede clasificar como una respuesta sobreamortiguamiento [Kuo1996].

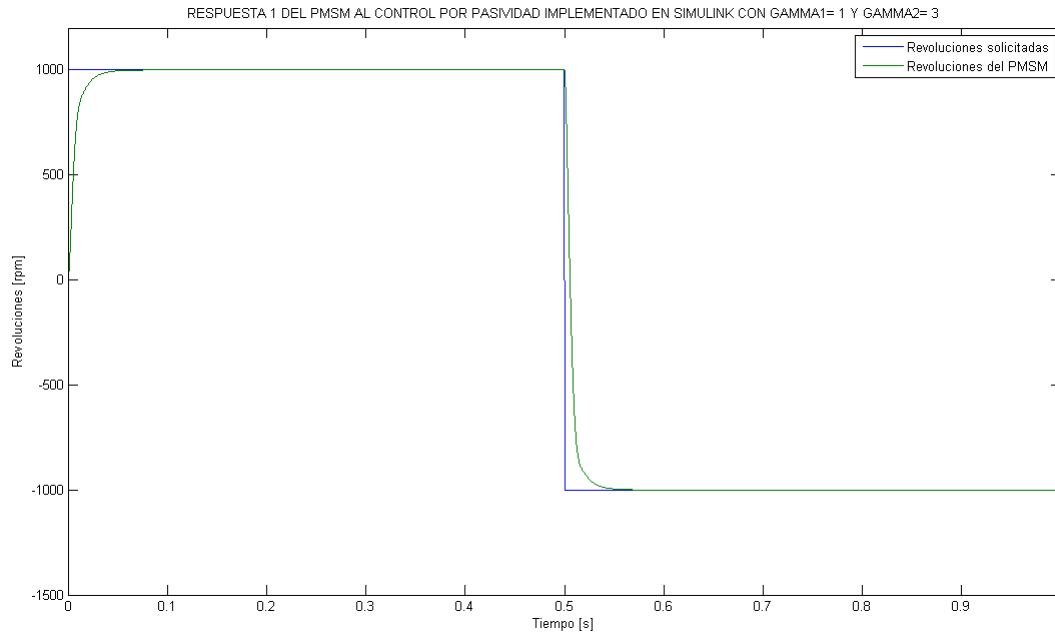


Figura B.4: Respuesta del PMSM al control por pasividad implementado en Simulink con $\gamma_1=1, \gamma_2=3$ para cambio de sentido a 1000 rpm.

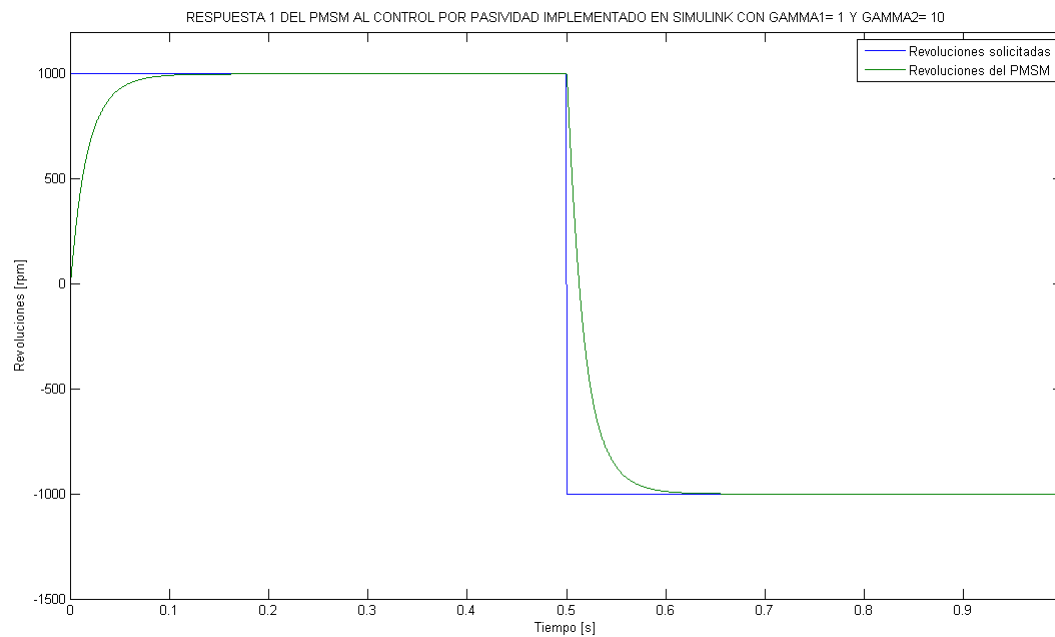


Figura B.5: Respuesta del PMSM al control por pasividad implementado en Simulink con $\gamma_1=1, \gamma_2=10$ para cambio de sentido a 1000 rpm.

En la figura B.4 se muestra la respuesta que se obtiene con los valores de $\gamma_1=1, \gamma_2=3$, se observa que es un amortiguamiento crítico [Kuo1996].

En la figura B.5 se observa la respuesta del PMSM con las ganancias $\gamma_1=1, \gamma_2=10$, esta se clasifica como de bajo amortiguamiento [Kuo1996].

Anexo C

Proceso de interconexión entre Simulink y Active_HDL, Simulink y PSIM.

El proceso para incluir en la librería de Simulink los bloques de **Active HDL** y **PSIM** se describe a continuación:

1. En la barra de menú principal se selecciona **File**.
2. En el menú desplegable se selecciona **Set Path ...**, figura C.1.
3. La ventana emergente selecciona **Add Folder...**, figura C.2
4. En la ventana emergente se busca en los *archivos de programa* la carpeta que contiene al programa adicionar
 - Para el programa de **Active HDL**, se selecciona la carpeta *Acdel*, la subcarpeta *Active-HDL* y se selecciona la subcarpeta *Simulink*, figura C.3.
 - Para el programa de **PSIM**, se selecciona la carpeta *Archivos de programa*, la subcarpeta *Powersim*, y se selecciona la subcarpeta *PSIM9.0.3_trial*, figura C.4
5. Se comprueba si se agregaron los bloques para realizar la cosimulación, en la ventana emergente de *Simulink Library Browser*.
 - Para el programa **Active HDL**, se agregó la subcarpeta *Active-HDL Toolbox*; la cual se visualiza como se muestra en la figura C.5.
 - En el caso del programa **PSIM**, el bloque que se agrega esta la subcarpeta *S-function SimCoupler*, como se observa en al figura C.6

En caso de no funcionar este procedimiento, buscar en la ruta C:\Aldec\Active-HDL Student Edition\Simulink (esta ruta es donde esta ubicado las carpetas del programa **Active-HDL**) el archivo *aldec_ahdl.mdl* y ejecutarlo; para el caso de **Active-HDL**. Y para **PSIM** la ruta probable es C:\Program Files\Powersim\PSIM9.0.3_Trial, en esta ruta encontrar y ejecutar el archivo *SetSimPath.exe*.

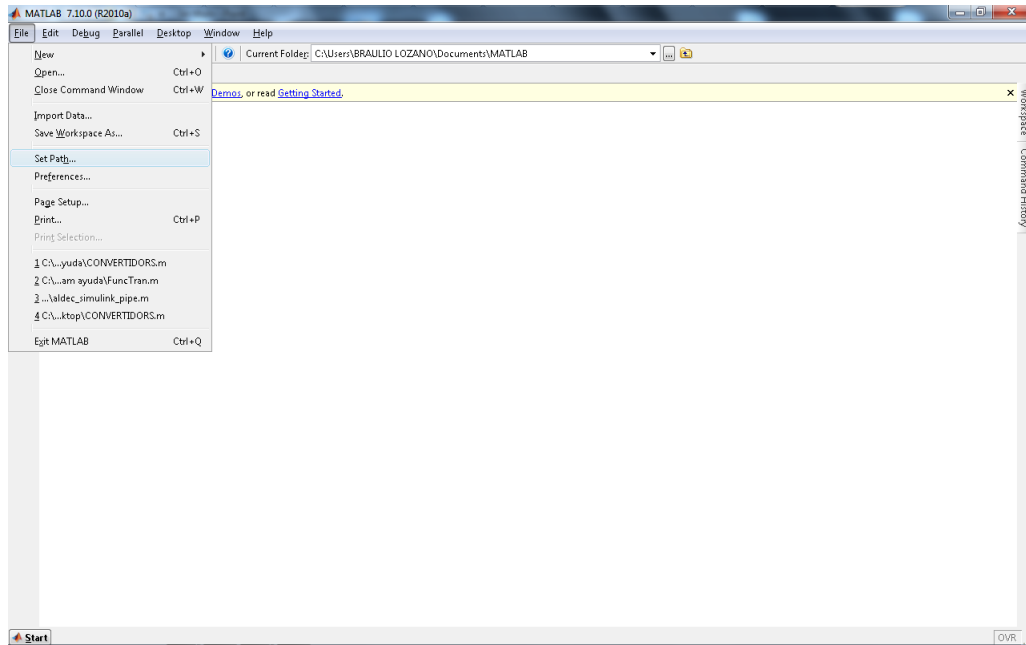


Figura C.1: Selección de Set Path...

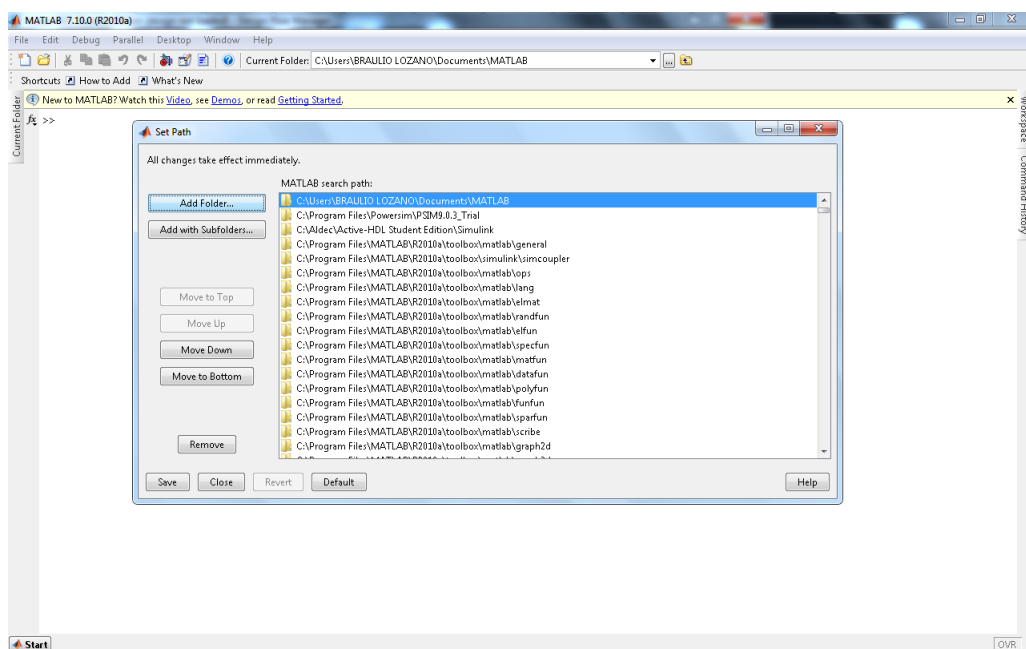


Figura C.2: Selección de Add Folder...

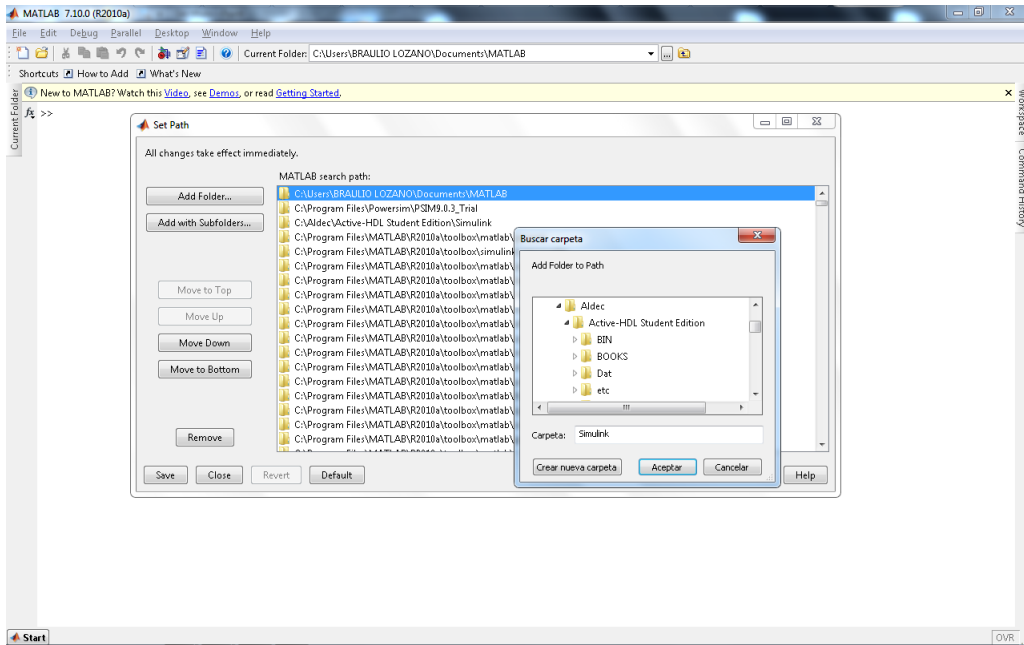


Figura C.3: Selección de la carpeta Simulink

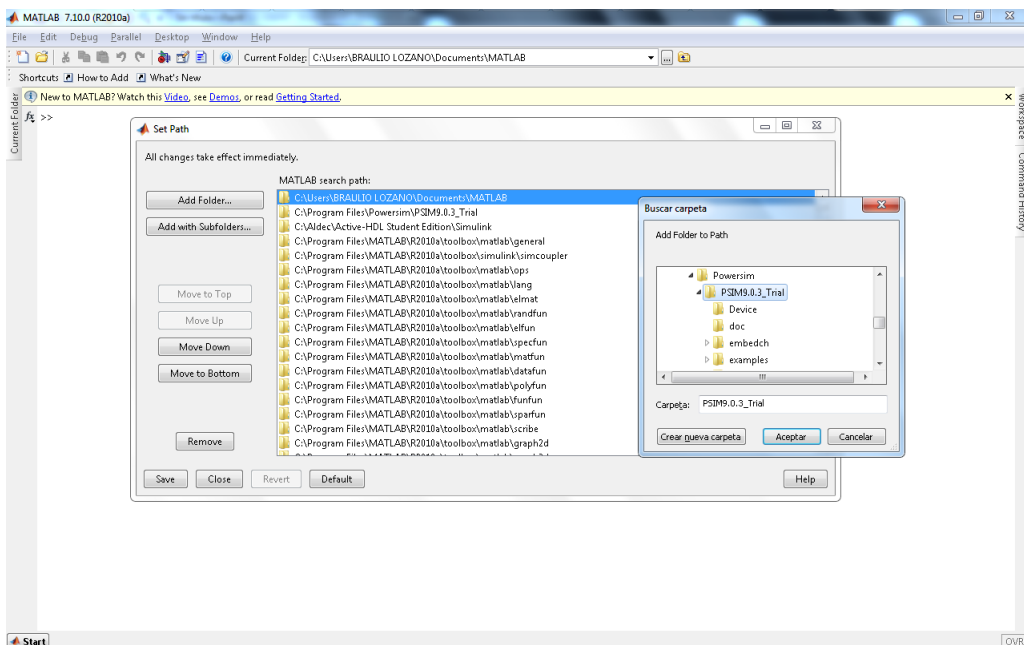


Figura C.4: Selección de la carpeta PSIM9.0.3_trial

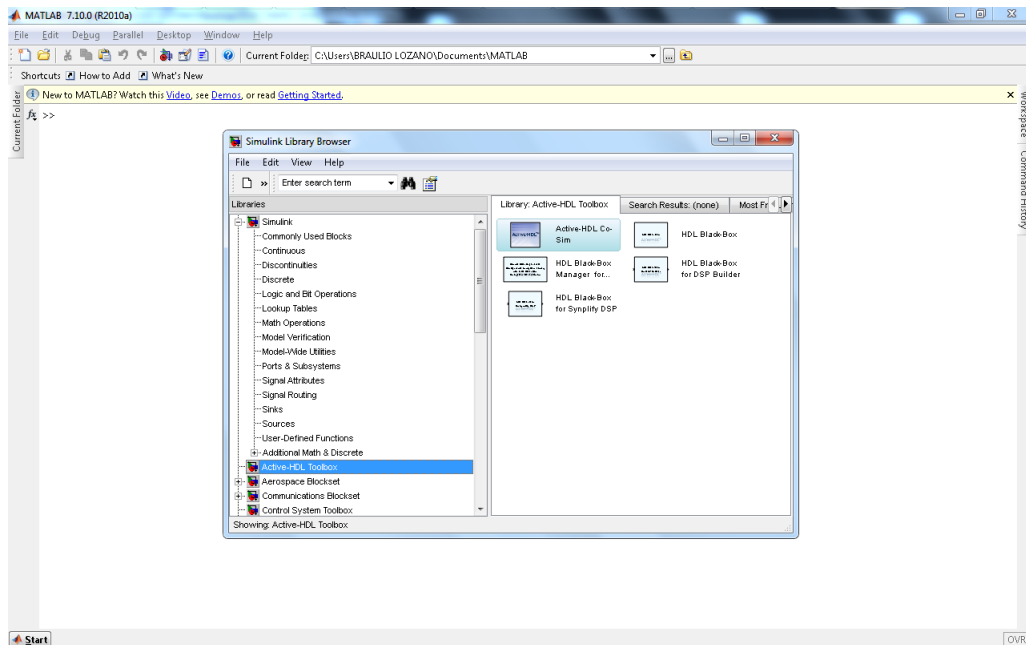


Figura C.5: Active-HDL Toolbox.

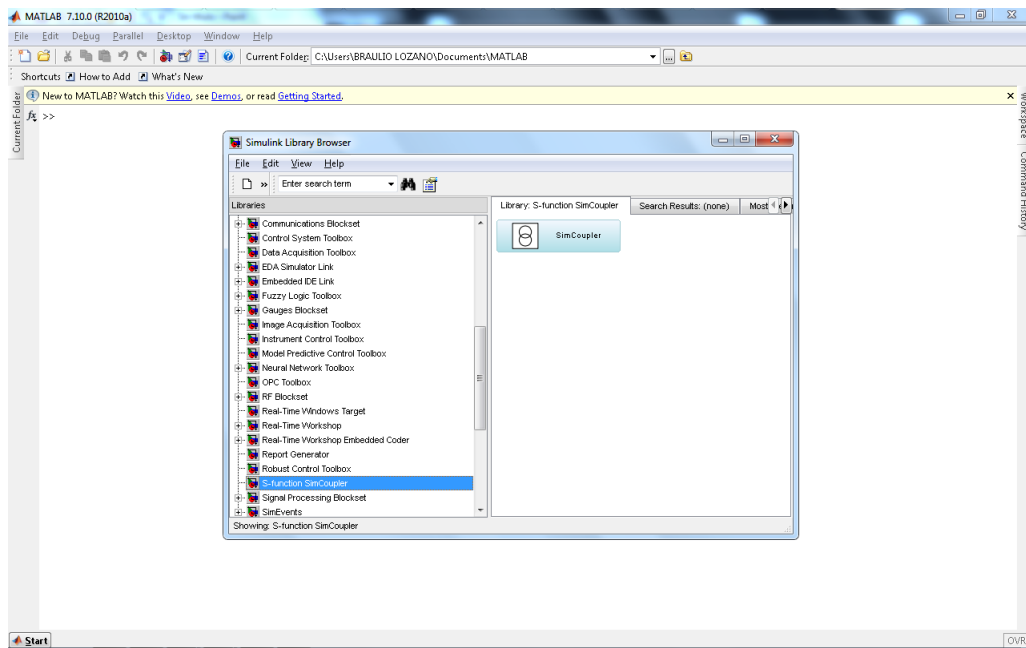


Figura C.6: S-function SimCoupler.

Anexo D

Respuesta del control en VHDL para diferentes ganancias.

Realizando las simulaciones con el diagrama a bloques que se muestra en la figura 5.10, haciendo la variación de las ganancias γ_1 y γ_2 se obtienen las siguientes gráficas.

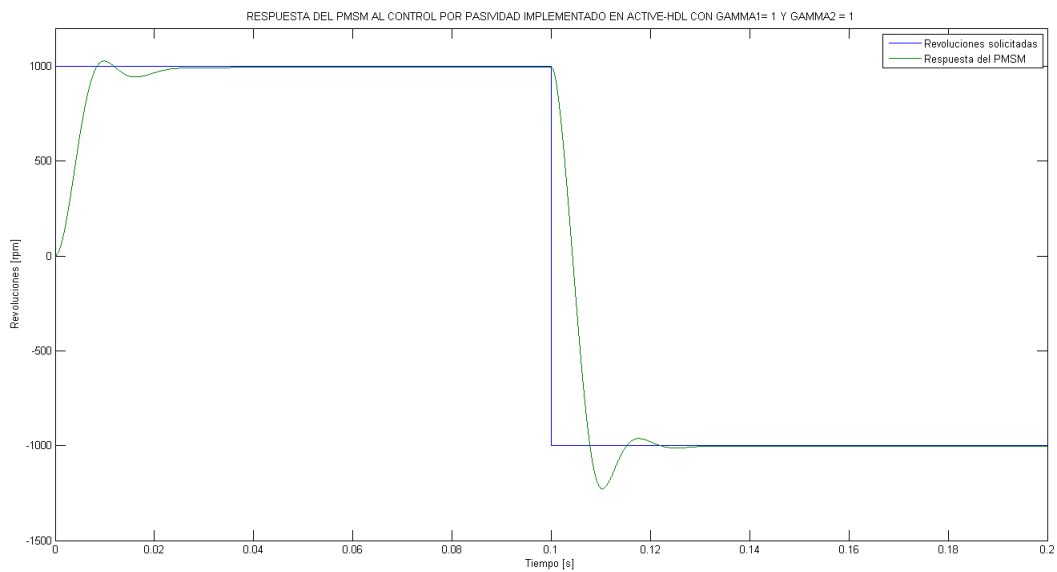


Figura D.1: Respuesta del PMSM al control por pasividad en HDL con $\gamma_1=1$ y $\gamma_2=1$.

En la figura D.1 se muestra la respuesta del PMSM para las ganancias $\gamma_1=1$ y $\gamma_2=1$.

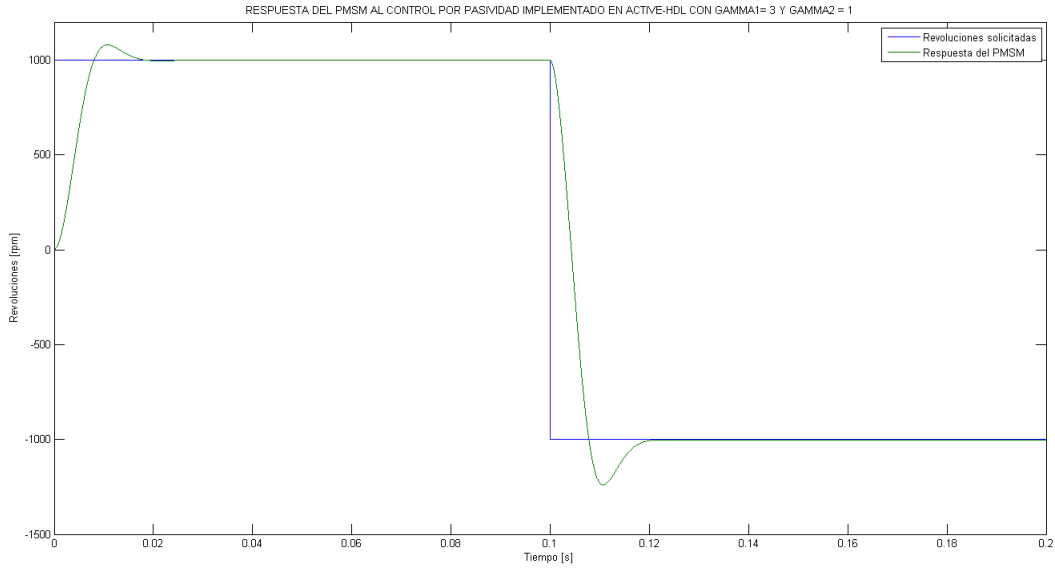


Figura D.2: Resposta do PMSM ao controle por passividade em HDL com $\gamma_1=3$ e $\gamma_2=1$.

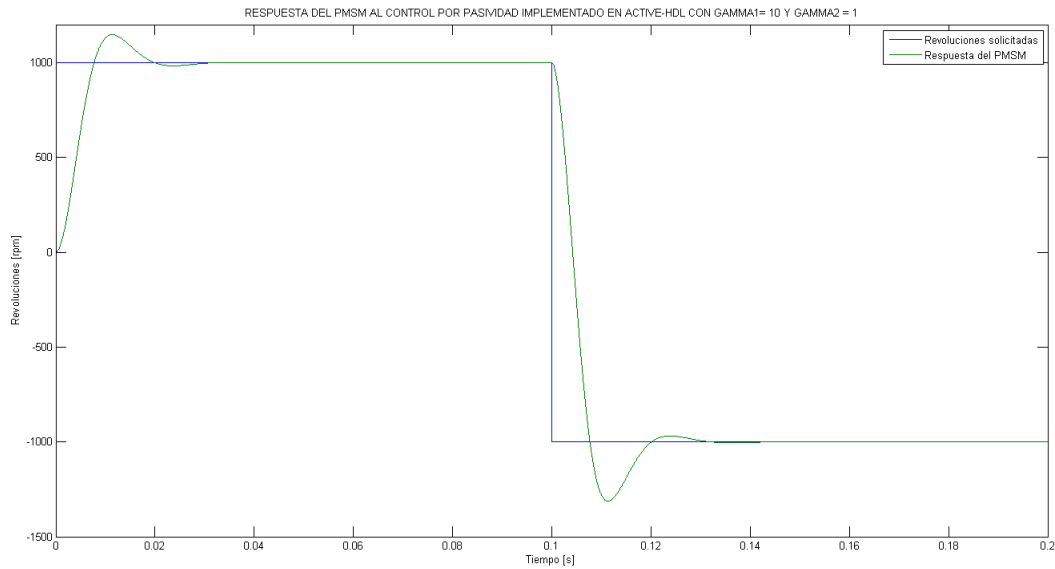


Figura D.3: Resposta do PMSM ao controle por passividade em HDL com $\gamma_1=10$ e $\gamma_2=1$.

En la figura D.2 se muestra la respuesta del PMSM que se obtiene con las ganancias $\gamma_1=3$ y $\gamma_2=1$.

En la figura D.3 es mostrada la respuesta que tiene el PMSM para las ganancias $\gamma_1=10$ y $\gamma_2=1$.

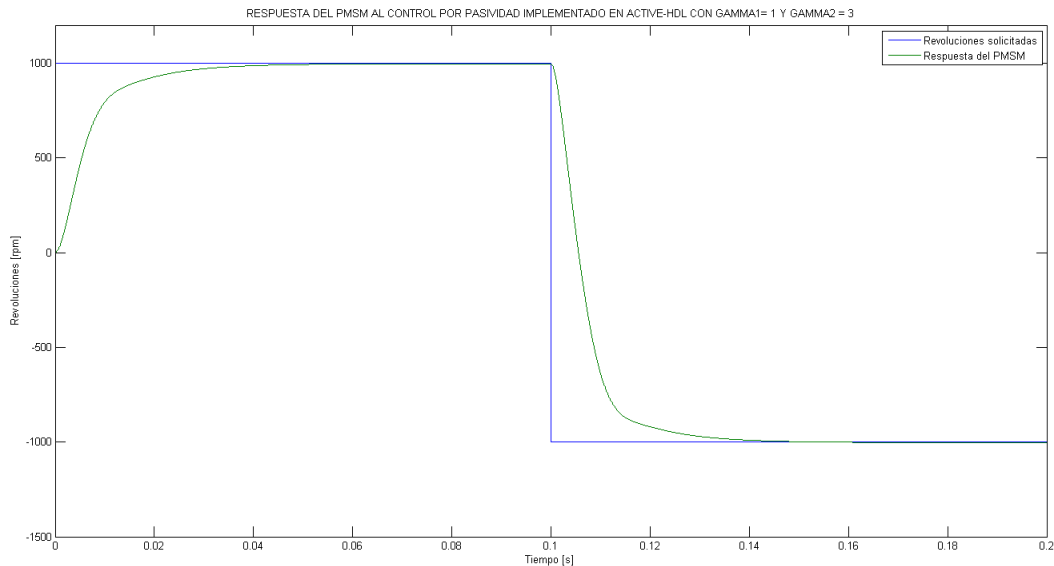


Figura D.4: Respuesta del PMSM al control por pasividad en HDL con $\gamma_1=1$ y $\gamma_2=3$.

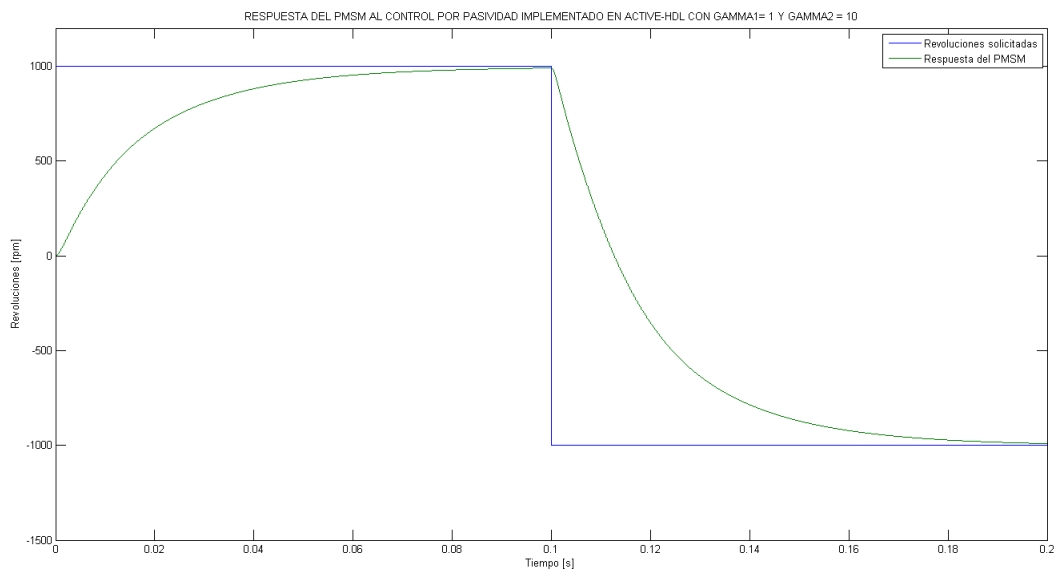


Figura D.5: Respuesta del PMSM al control por pasividad en HDL con $\gamma_1=1$ y $\gamma_2=10$.

En la figura D.4 se observa la respuesta del PMSM con las ganancias $\gamma_1=1$ y $\gamma_2=3$.

En la figura D.5 se muestra la respuesta del PMSM al control por pasividad implementado en HDL con ganancias $\gamma_1=1$ y $\gamma_2=10$.

Anexo E

Co-simulación de simulink y PSIM del control por pasividad.

Las cosimulaciones que se realizan para distintas frecuencias de la señal triangular se hacen con un voltaje de 84 V y con las ganancias $\gamma_1=10$ y $\gamma_2=3$; y es solicitado al PMSM cambio de sentido de 1000 rpm a -1000 rpm.

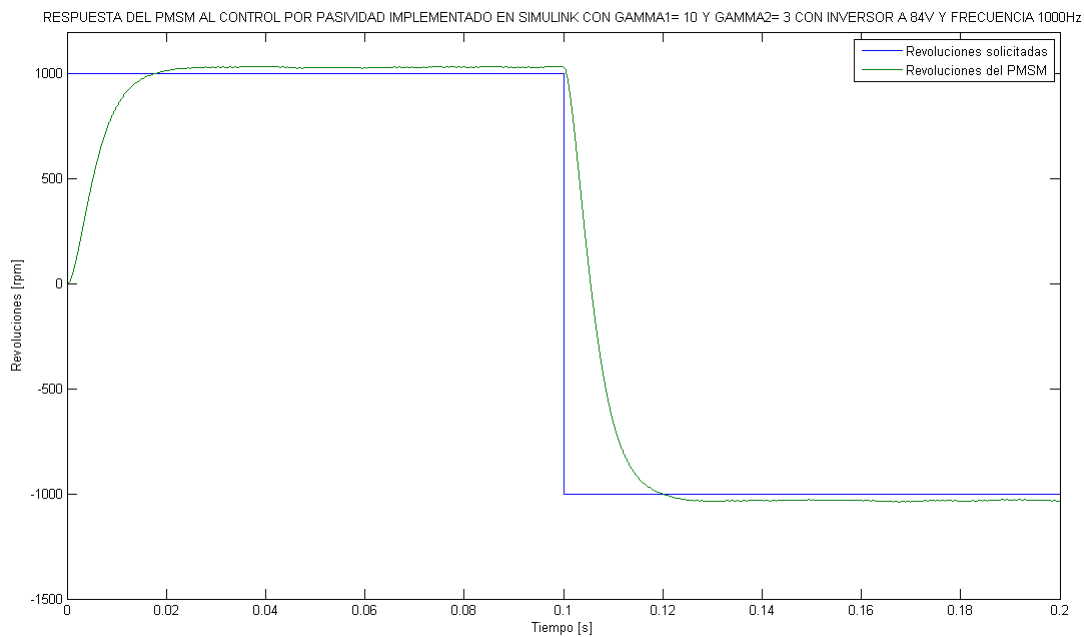


Figura E.1: Respuesta del PMSM al control por pasividad en simulink e inversor en PMSM con un voltaje de 84 V y frecuencia de 1 kHz.

En la figura E.1 se observa la respuesta para la frecuencia de 1000 Hz.

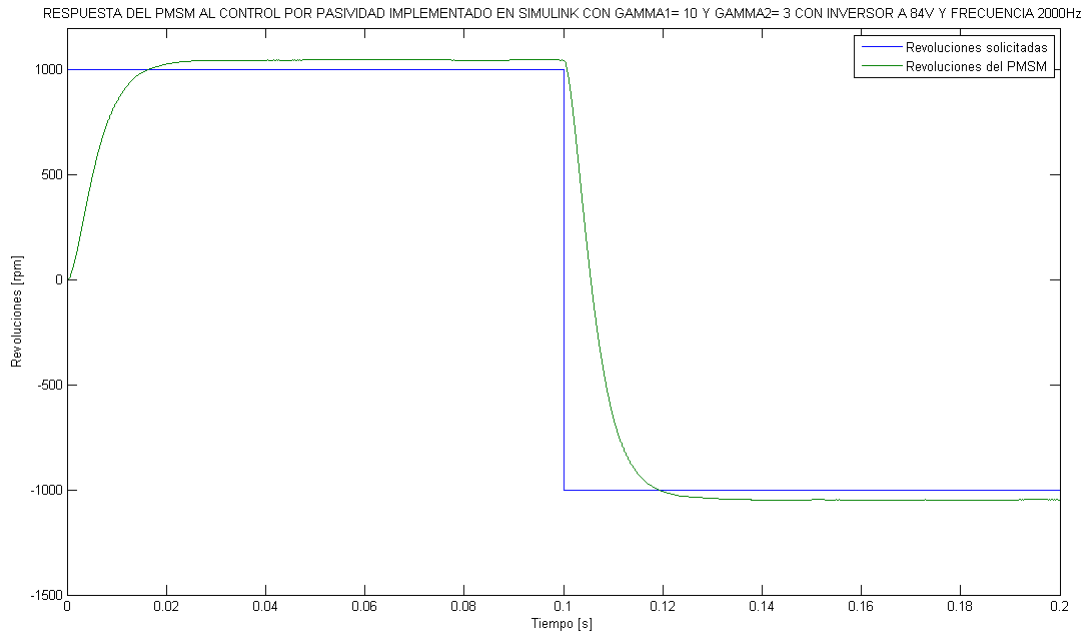


Figura E.2: Respuesta del PMSM al control por pasividad en simulink e inversor en PMSM con un voltaje de 84 V y frecuencia de 2 kHz.

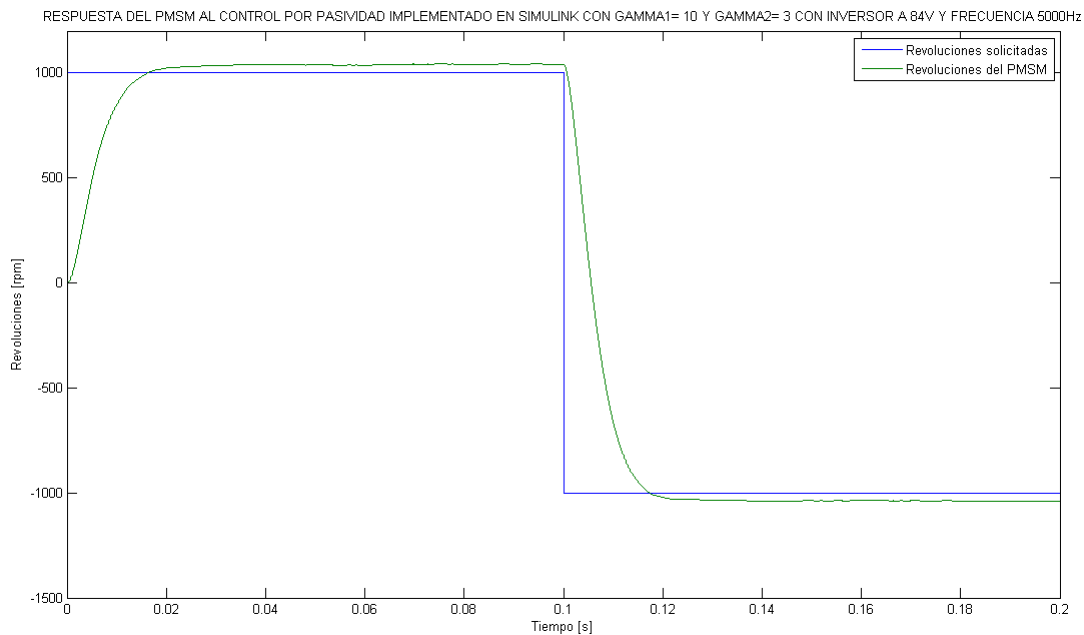


Figura E.3: Respuesta del PMSM al control por pasividad en simulink e inversor en PMSM con un voltaje de 84 V y frecuencia de 5 kHz.

En la figura E.2 se muestra la respuesta que tiene el motor PMSM con una frecuencia de 2000 Hz.

En la figura E.3 se observa la respuesta que tiene el PMSM a una frecuencia de 5000 Hz de la señal triangular.

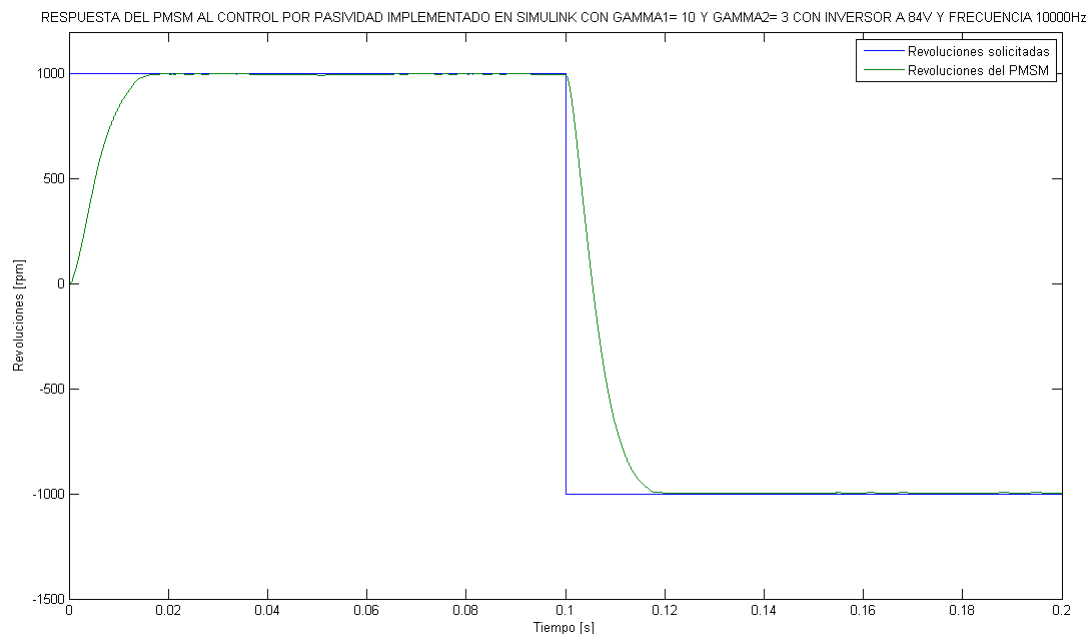


Figura E.4: Respuesta del PMSM al control por pasividad en simulink e inversor en PMSM con un voltaje de 84 V y frecuencia de 10 kHz.

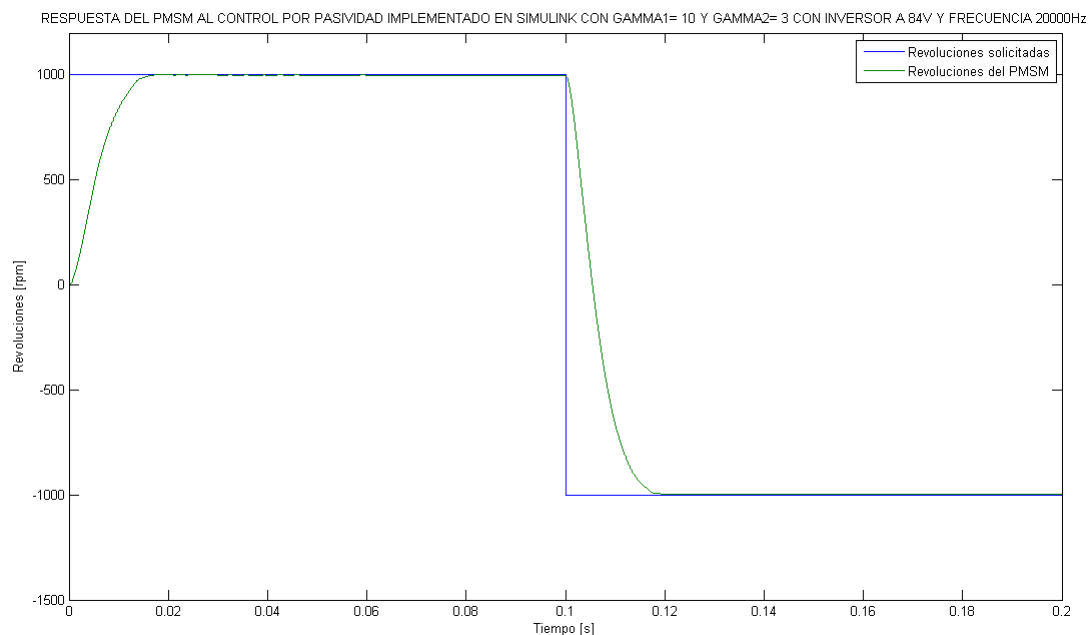


Figura E.5: Respuesta del PMSM al control por pasividad en simulink e inversor en PMSM con un voltaje de 84 V y frecuencia de 20 kHz.

En la figura E.4 se muestra la respuesta que tiene el motor PMSM con una frecuencia de 10 000 Hz.

Se observa en la figura E.5 la respuesta que tiene el motor PMSM a una señal triangular con una frecuencia de 20 000 Hz.

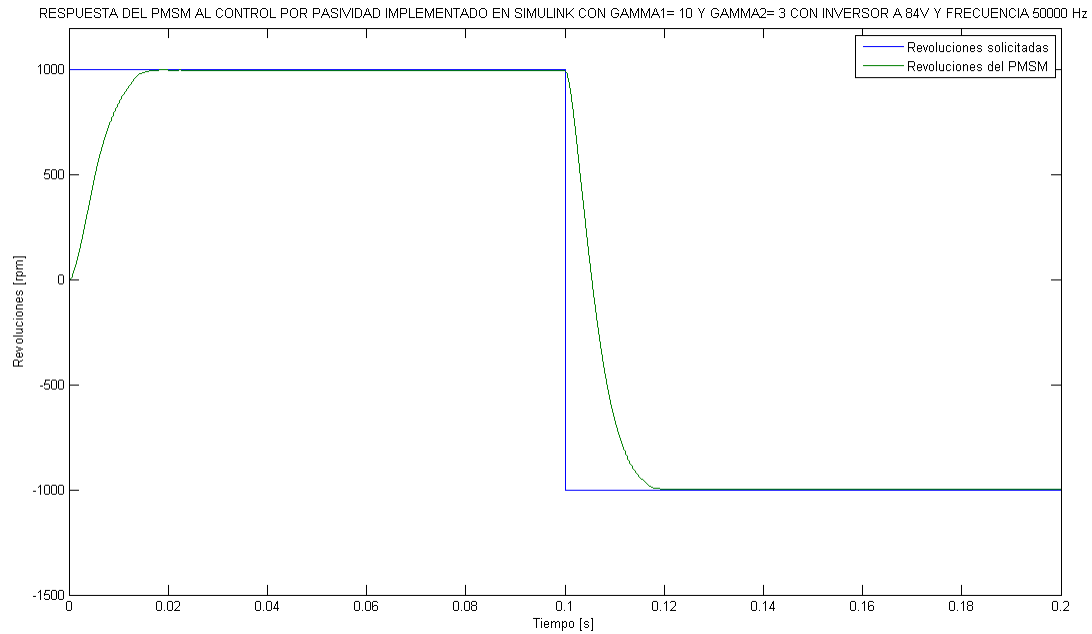


Figura E.6: Respuesta del PMSM al control por pasividad en simulink e inversor en PMSM con un voltaje de 84 V y frecuencia de 50 kHz.

En la figura E.6 se muestra la respuesta del PMSM a una señal triangular con frecuencia de 50 000 Hz.

Anexo F

Co-simulación de simulink y PSIM del control por pasividad con una frecuencia 20 kHz con diferentes ganancias.

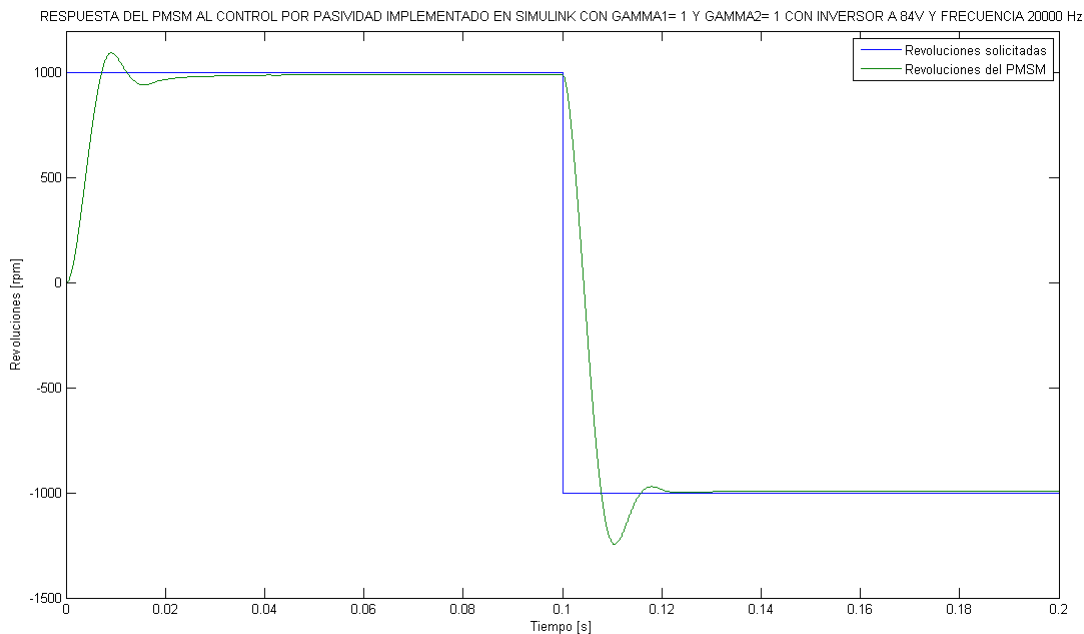


Figura F.1: Respuesta del PMSM al control por pasividad en simulink e inversor en PSIM a 84 V, frecuencia de 20 kHz y $\gamma_1 = 1$ y $\gamma_2 = 1$.

En la figura F.1 se muestra la respuesta que tiene el motor PMSM al implementar el inversor en PSIM con un voltaje de 84 V y una frecuencia de 20 kHz, para las ganancias con los valores de $\gamma_1 = 1$ y $\gamma_2 = 1$.

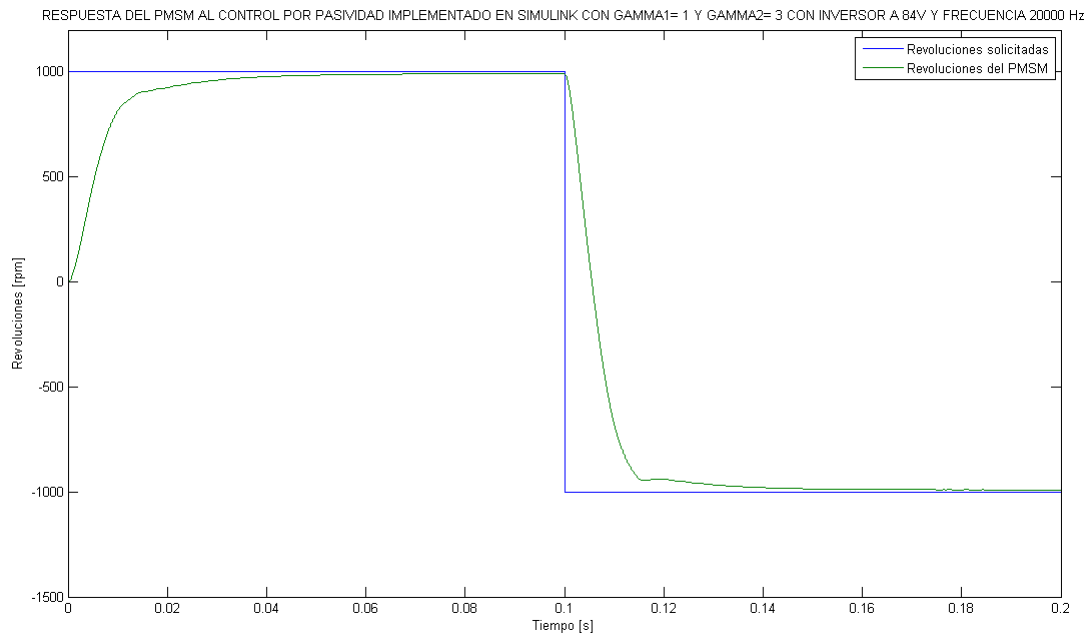


Figura F.2: Respuesta del PMSM al control por pasividad en simulink e inversor en PSIM a 84 V, frecuencia de 20 kHz y $\gamma_1 = 1$ y $\gamma_2 = 3$.

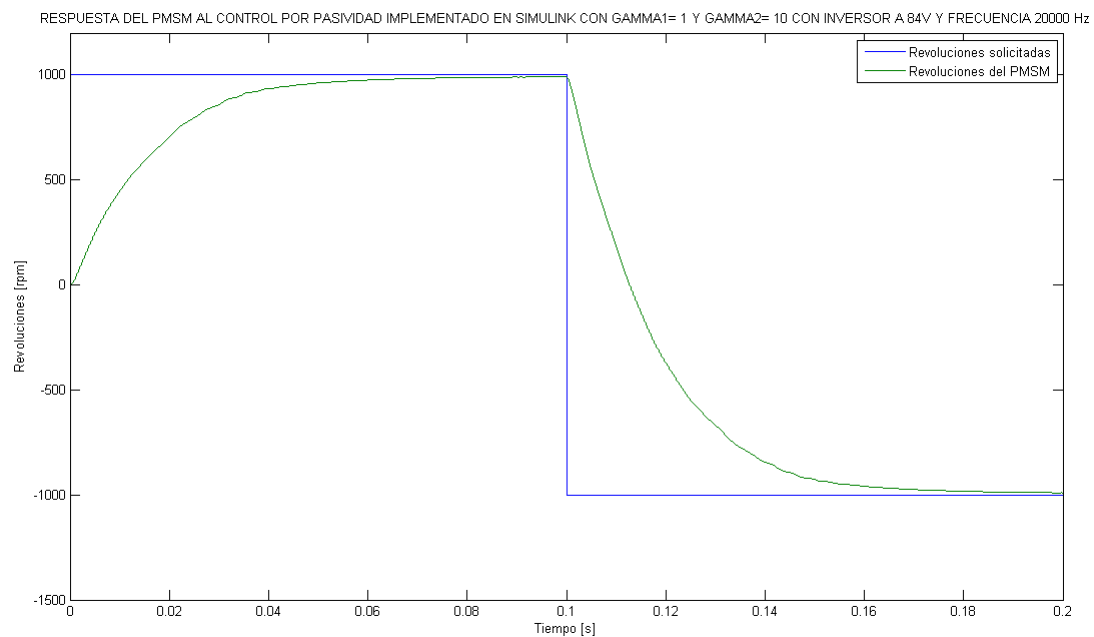


Figura F.3: Respuesta del PMSM al control por pasividad en simulink e inversor en PSIM a 84 V, frecuencia de 20 kHz y $\gamma_1 = 1$ y $\gamma_2 = 10$.

Se muestra en la figura F.2 la respuesta que se obtienen con ganancias $\gamma_1 = 1$ y $\gamma_2 = 3$.

Se observa en la figura F.3 la respuesta que se obtiene del motor PMSM con las ganancias $\gamma_1 = 1$ y $\gamma_2 = 10$.

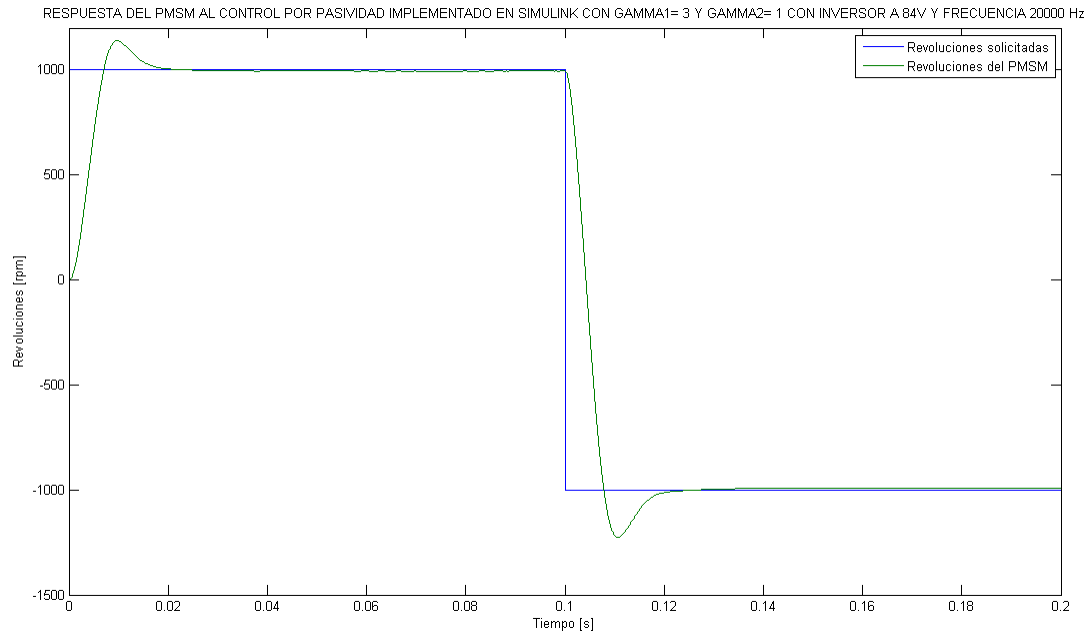


Figura F.4: Respuesta del PMSM al control por pasividad en simulink e inversor en PSIM a 84 V, frecuencia de 20 kHz y $\gamma_1=3$ y $\gamma_2=1$.

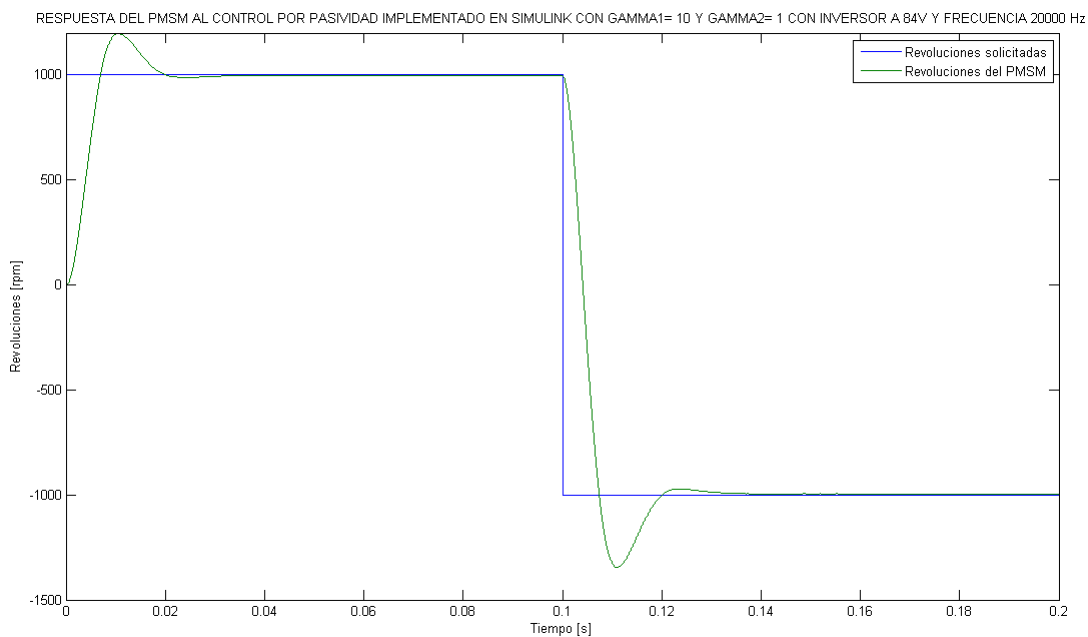


Figura F.5: Respuesta del PMSM al control por pasividad en simulink e inversor en PSIM a 84 V, frecuencia de 20 kHz y $\gamma_1=10$ y $\gamma_2=1$.

Se observa en la figura F.4 la respuesta del motor PMSM con las ganancias $\gamma_1=3$ y $\gamma_2=1$.

La respuesta que se muestra en la figura F.5 es la que se obtiene con las ganancias $\gamma_1=10$ y $\gamma_2=1$.

Anexo G

Co-simulación del control en VHDL con PSIM a una frecuencia 20 kHz con diferentes ganancias.

La respuesta que tiene el motor PMSM al control por pasividad implementado en **HVLD** con el inversor en PSIM, con un voltaje de 84 V y una frecuencia de 20 kHz.

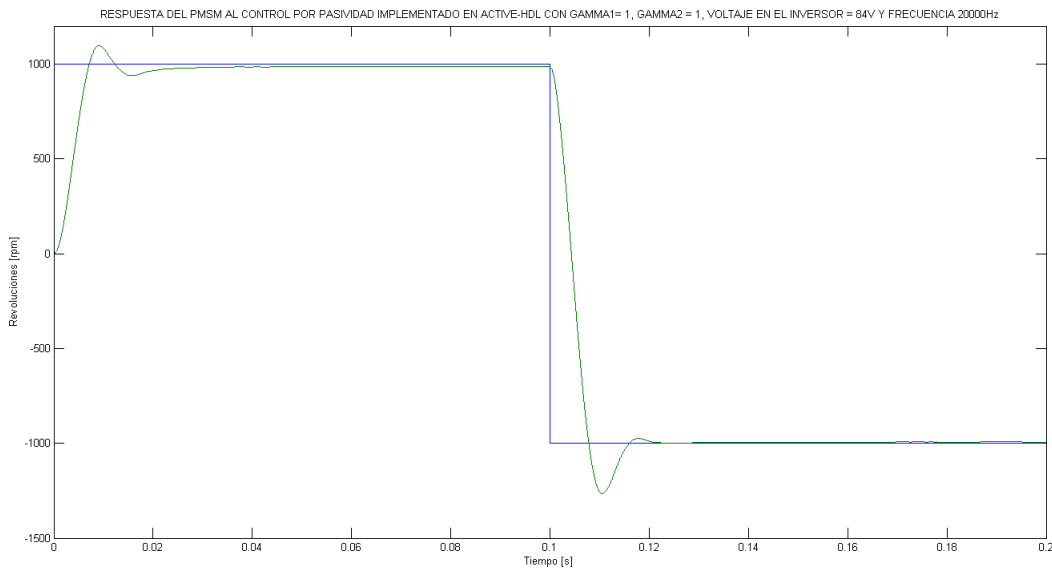


Figura G.1: Respuesta del PMSM al control por pasividad en VHDL e inversor en PSIM a 84 V, frecuencia de 20 kHz y $\gamma_1 = 1$ y $\gamma_2 = 1$.

Se muestra en la figura G.1 la respuesta que tiene el motor PMSM al control por pasividad implementado en **HVLD** con el inversor en PSIM, con un voltaje de 84 V y una frecuencia de 20 kHz, para las ganancias con los valores de $\gamma_1 = 1$ y $\gamma_2 = 1$.

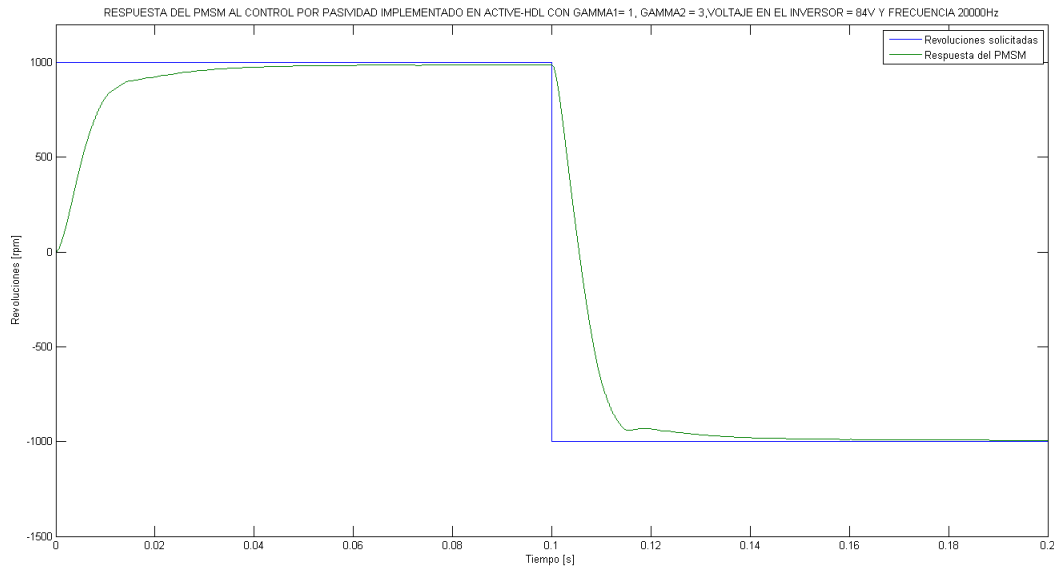


Figura G.2: Respuesta del PMSM al control por pasividad en VHDL e inversor en PSIM a 84 V, frecuencia de 20 kHz y $\gamma_1 = 1$ y $\gamma_2 = 3$.

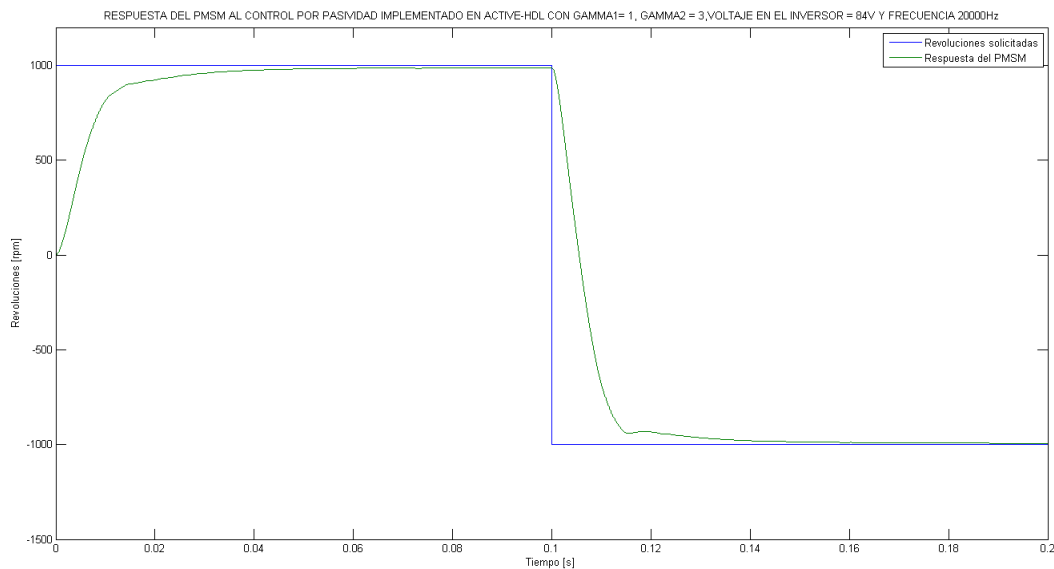


Figura G.3: Respuesta del PMSM al control por pasividad en VHDL e inversor en PSIM a 84 V, frecuencia de 20 kHz y $\gamma_1 = 1$ y $\gamma_2 = 10$.

Se observa en la figura G.2 la respuesta que se obtiene del motor PMSM con las ganancias $\gamma_1 = 1$ y $\gamma_2 = 3$.

La respuesta que muestra en la figura G.3 se obtienen con ganancias $\gamma_1 = 1$ y $\gamma_2 = 10$.

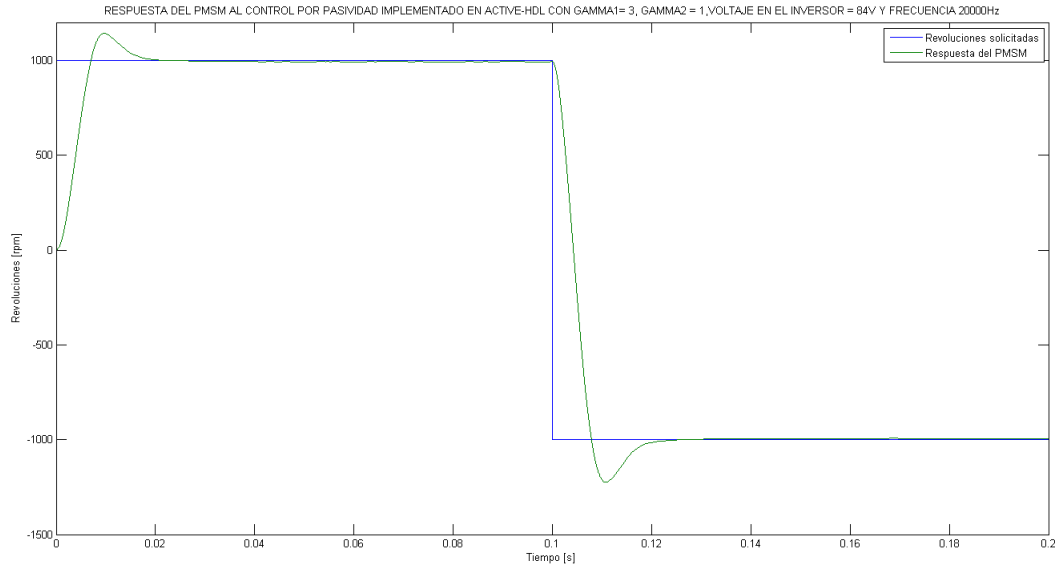


Figura G.4: Respuesta del PMSM al control por pasividad en VHDL e inversor en PSIM a 84 V, frecuencia de 20 kHz y $\gamma_1 = 3$ y $\gamma_2 = 1$.

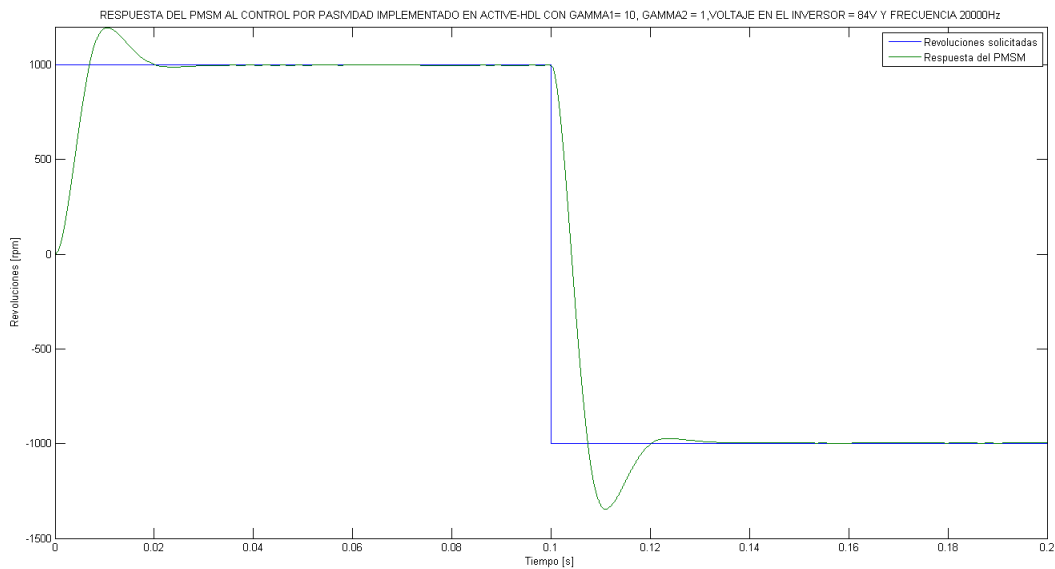


Figura G.5: Respuesta del PMSM al control por pasividad en VHDL e inversor en PSIM a 84 V, frecuencia de 20 kHz y $\gamma_1 = 10$ y $\gamma_2 = 1$.

Se observa en la figura G.4 la respuesta del motor PMSM con las ganancias $\gamma_1 = 3$ y $\gamma_2 = 1$.

La respuesta que se muestra en la figura G.5 es la que se obtiene con las ganancias $\gamma_1 = 10$ y $\gamma_2 = 1$.