ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

CO-SIMULACIÓN PARA EL CONTROL DE UNA PMSM UTILIZANDO SOFTWARE DE ANÁLISIS MEDIANTE ELEMENTOS FINITOS

TESIS DE GRADO PREVIO A LA OBTENCIÓN DEL TÍTULO DE MAGÍSTER EN AUTOMATIZACIÓN Y CONTROL ELECTRÓNICO INDUSTRIAL

HUGO GIOVANNI LOYA RIVERA

giomh_1107@hotmail.com

DIRECTOR: DR. ING. PAUL MARCELO POZO PALMA

marcelo.pozo@epn.edu.ec

Quito, noviembre 2018

AVAL

Certifico que el presente trabajo fue desarrollado por Hugo Giovanni Loya Rivera, bajo mi supervisión.

DR. ING. PAUL MARCELO POZO PALMA
DIRECTOR DEL TRABAJO DE TITULACIÓN

DECLARACIÓN DE AUTORÍA

Yo, Hugo Giovanni Loya Rivera declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

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

HUGO GIOVANNI LOYA RIVERA

DEDICATORIA

A Dios

Por haberme permitido llegar hasta este punto y haberme dado salud para lograr mis objetivos, además de su infinita bondad y amor.

A mis padres Eduardo y Victoria mis hermanos Alexander y Jonathan

Por ser el pilar fundamental en todo lo que soy, en toda mi educación, tanto académica, como de la vida, por su incondicional apoyo perfectamente mantenido a través del tiempo.

A mi futura esposa Verito

Por su gran amor y apoyo incondicional en todo momento.

Todo este trabajo ha sido posible gracias a ellos.

Hugo Giovanni

AGRADECIMIENTO

A mis profesores por compartir sus conocimientos y enseñarme a cumplir mis objetivos con perseverancia y tenacidad.

Y un agradecimiento especial al Dr.-Ing. Marcelo Pozo por su valiosa ayuda y guía para la elaboración del presente trabajo.

ÍNDICE DE CONTENIDO

AVAL I
DECLARACIÓN DE AUTORÍA II
DEDICATORIA III
AGRADECIMIENTO IV
ÍNDICE DE CONTENIDO V
RESUMEN VIII
ABSTRACT IX
1 INTRODUCCIÓN 1
1.1 Objetivo General 2
1.2 Objetivos Específicos 2
1.3 Alcance 2
1.4 Marco Teórico 3
1.4.1 Análisis mediante el método de elementos finitos 3
1.4.2 Sistema de accionamiento de un motor sincrónico de imanes permanentes 10
1.4.3 Motor sincrónico de imanes permanentes 10
1.4.4 Sensores de posición 16
1.4.5 Inversor trifásico 20
1.4.6 Diagrama vectorial, modelo matemático y circuito equivalente de un motor síncrono de imanes permanentes. 21
1.4.7 Control por frecuencia 31
1.4.8 Control de velocidad 34
2 METODOLOGÍA 36
2.1 Modelado de una PMSM 37
2.1.1 Desarrollo del modelo de la PMSM utilizando Matlab/Simulink 37
2.1.2 Desarrollo del modelo de la PMSM utilizando RMxprt Desing 37
2.1.3 Desarrollo del modelo de la PMSM utilizando Maxwell 2D 47
2.1.4 Implementación del circuito eléctrico de potencia en Ansys Simplorer. 51
2.1.5 Co-simulación entre los softwares de Ansys con Simplorer 54
2.1.6 Co-simulación Simplorer-Matlab 57

2.1.7 Co-simulación RMxprt-Simplorer-Matlab (Without Finite Elements Analysis) ……..60
2.1.8 Co-simulación Maxwell 2D-Simplorer-Matlab (With Finite Element Analysis) ……..61
2.2 Diseño de controladores 62
2.2.1 Control escalar V/F 62
2.2.2 Control de velocidad.....63
2.2.3 Cálculo de las contantes del controlador proporcional-integral.....64
2.2.4 Variables de conexión Matlab-Simplorer 66
3 RESULTADOS Y DISCUSIÓN 68
3.1 Resultados 68
3.2 Resultados obtenidos de la simulación del modelo de la PMSM implementado en Matlab/Simulink. 68
3.2.1 Modelo con variación de velocidad y torque de carga en lazo abierto. 68
3.2.2 Modelo con variación de velocidad a 450 RPMs, T_L=4N*m a t=0 segundos, T_L=5N*m a t=0.35 segundos.....69
3.2.3 Modelo con variación de velocidad a 900 RPMs, T_L=4N*m a t=0 segundos, T_L=5N*m a t=0.35 segundos.....72
3.3 Resultados obtenidos mediante co-simulación Ansys RMxprt-Simplorer-Matlab.....75
3.3.1 Modelo con variación de velocidad a 450 RPMs, T_L=4N*m a t=0 segundos, T_L=5N*m a t=0.35 segundos.....76
3.3.2 Modelo con variación de velocidad a 900 RPMs, T_L=4N*m a t=0 segundos, T_L=5N*m a t=0.35 segundos.....79
3.4 Resultados obtenidos mediante co-simulación Ansys Maxwell2D-Simplorer-Matlab. 82
3.4.1 Modelo con variación de velocidad a 450 RPMs, T_L=4N*m a t=0 segundos, T_L=5N*m a t=0.35 segundos.....82
3.4.2 Modelo con variación de velocidad a 900 RPMs, T_L=4N*m a t=0 segundos, T_L=5N*m a t=0.35 segundos.....85
3.5 Comparación de resultados de los tres modelos simulados. 88
3.5.1 Comparación de la respuesta de velocidad a 450 RPMs, T_L=4N*m a t=0 segundos, T_L=5N*m a t=0.35 segundos de los tres modelos simulados. 88
3.5.2 Comparación de la respuesta de velocidad a 900 RPMs, T_L=4N*m a t=0 segundos, T_L=5N*m a t=0.35 segundos de los tres modelos simulados. 91
4 CONCLUSIONES.....95
5 REFERENCIAS BIBLIOGRÁFICAS 96
6 ANEXOS 99
ANEXO I 100

ANEXO II 114
ORDEN DE EMPASTADO 115

RESUMEN

La máquina síncrona de imanes permanentes PMSM corresponde a una de las más utilizadas en la actualidad por sus características de dinámica, eficiencia y desempeño. Es por esta razón que los diseñadores buscan formular nuevos modelos de máquinas y sistemas de accionamiento que puedan ser simulados antes de ser llevados a su construcción.

En este contexto, el objetivo de este trabajo es emplear el método de análisis mediante elementos finitos para simular y verificar el comportamiento de un motor síncrono de imanes permanentes, circuito de potencia y del controlador escalar V/F trabajando en conjunto. Las técnicas de simulación utilizadas son: modelado con dispositivos de Matlab, modelado con RMxprt y modelado con Maxwell 2D, siendo las dos últimas las que realizan un modelado de las características y condiciones constructivas del motor. Cabe indicar que el modelado de mayor complejidad y a la ves el que permite obtener resultados que más se aproximan a la realidad es el realizado con Maxwell 2D ya que emplea un modelado utilizando la técnica de elementos finitos. Los resultados obtenidos de las simulaciones serán comparados y analizados para validar el presente trabajo.

PALABRAS CLAVE: PMSM, Escalar, Simulación, Ansys, Matlab, Maxwell, RMxprt.

ABSTRACT

The synchronous machine of permanent magnets PMSM corresponds to one of the most used nowadays due to its characteristics of dynamics, efficiency and performance. It is for this reason that designers seek to formulate new models of machines and drive systems that can be simulated before being taken to their construction. In this context, the objective of this work is to use the method of finite element analysis to simulate and verify the behavior of a permanent magnet synchronous motor, power circuit and the scalar V/F controller working together. The simulation techniques used are: modeling with Matlab devices, modeling with RMxprt and modeling with Maxwell 2D, the last two being those that perform a modeling of the characteristics and constructive conditions of the engine. It should be noted that the modeling of greater complexity and at the same time the one that allows obtaining results that are closer to reality is the one made with Maxwell 2D since it uses modeling using the finite element technique. The results obtained from the simulations will be compared and analyzed to validate the present work.

KEYWORDS: PMSM, Scalar, Ansys, Maxwell, Matlab, RMxprt, Simulation.

1 INTRODUCCIÓN

En la actualidad, las máquinas síncronas de imanes permanentes (por sus siglas en inglés, Permanent Magnet Synchronous Motor) PMSM son ampliamente utilizadas en aplicaciones de baja y mediana potencia como por ejemplo en equipos periféricos de computadora, robótica móvil, variadores de velocidad ajustables, vehículos eléctricos, generación eólica [1], entre otras.

El aumento de la demanda de las PMSM, ha exigido la necesidad de utilizar nuevas herramientas de simulación que sean capaces de emular del comportamiento del motor y su accionamiento. Las simulaciones han ayudado al proceso de desarrollo de nuevos sistemas, incluidos los motores, al reducir el costo y el tiempo de fabricación.

Existe una gran variedad de softwares como por ejemplo Comsol [2], Flux [3], Ansys [4] que permiten realizar simulaciones de máquinas eléctricas utilizando la técnica de análisis mediante elementos finitos. Este tipo de análisis permite crear modelos de máquinas en base a la parametrización de dimensiones físicas, tipos de materiales de construcción, condiciones de contorno, etc. y obtener resultados del comportamiento que se apegan más a la realidad, en comparación a los modelos convencionales implementados utilizando ecuaciones diferenciales.

Con base a lo anterior, se han desarrollado diversos estudios donde se utiliza ésta técnica como herramienta de simulación para determinar el comportamiento real de motores de inducción [5], motores BLDC [6], motores de imanes permanentes [7], al aplicar diversas técnicas de control, simulación de fallas y otros tipos de análisis.

En esta propuesta se plantea la implementación de tres simulaciones de un motor síncrono de imanes permanentes con su respectivo circuito de potencia y circuito de control, utilizando los softwares Matlab/Simulink, Simplorer, Ansys RMxprt y Ansys Maxwell 2D.

Matlab/Simulink permite implementar la simulación del motor, circuito de potencia, circuito de control, así como el controlador escalar V/F utilizando librerías propias del software.

La co-simulación Ansys RMxprt-Simplorer-Matlab implementa una simulación en base a parametrización de la máquina mediante características constructivas y condiciones de contorno. Este modelo se interconecta con Simplorer en donde se modela el circuito de potencia y a la vez transfiere datos con Simulink que posee el controlador escalar para el control de velocidad y torque.

La co-simulación Maxwell 2D-Simplorer-Matlab implementa una simulación en base a parametrización de la máquina mediante características constructivas y condiciones de contorno y además de ello utiliza análisis mediante elementos finitos para obtener una

respuesta más realista del sistema implementado. Este modelo se interconecta con Simplorer en donde se modela el circuito de potencia y con Simulink en donde se implementa el controlador escalar para el control de velocidad y torque.
Los resultados de las tres simulaciones serán comparados con el objetivo de determinar la eficiencia del análisis mediante elementos finitos al aplicar varias velocidades de referencia y torque de carga.

1.1 Objetivo General

El objetivo general del presente proyecto es realizar la co-simulación para el control de una pmsm utilizando software de análisis mediante elementos finitos.

1.2 Objetivos Específicos

Los objetivos específicos de este estudio son:

- Realizar la revisión, estudio y recopilación de información de papers y de trabajos enfocados en la creación del modelo de una PMSM mediante software de análisis por elementos finitos.
- Diseñar e implementar el modelo de una PMSM mediante software de análisis por elementos finitos.
- Diseñar e implementar el circuito de control y de potencia para la simulación de una PMSM.
- Implementar la guía para la realización de la co-simulación de los subsistemas diseñados.
- Realizar pruebas de la co-simulación implementada que permitan validar el trabajo realizado mediante el análisis de resultados.

1.3 Alcance

El presente trabajo se trata de un estudio en donde se han realizado las diversas actividades de investigación e implementación descritas a continuación:

Se ha realizado la revisión, estudio y recopilación de información de papers y de trabajos enfocados en la misma línea de investigación para la creación del modelo de una PMSM

mediante software de análisis por elementos finitos y, de esta manera contar con fuentes de información que permitan obtener la co-simulación con el circuito de potencia y el controlador escalar a ser diseñado en MATLAB.

Se ha diseñado e implementado el modelo de una PMSM mediante el software elementos finitos que permite visualizar de manera gráfica la interacción de los efectos electromagnéticos y mecánicos mediante animaciones y gráficas de análisis de resultados.

Se ha diseñado e implementado el circuito de control en Matlab utilizando técnicas de modulación PWM cuyas señales se interconectarán con el software de modelado del circuito de potencia de una PMSM.

Se ha diseñado e implementado la co-simulación de todo el sistema entre un software de elementos finitos y Matlab-Simulink. Para esto se ha investigado el funcionamiento de los softwares a ser interconectados con el objetivo de que la simulación entregue los resultados esperados.

Se han realizado pruebas de la co-simulación con el objetivo de obtener resultados que permitan validar el presente trabajo y se los ha comparado con los resultados obtenidos en la simulación de un modelo de una PMSM obtenido de las librerías de Matlab.

1.4 Marco Teórico

1.4.1 Análisis mediante el método de elementos finitos

El método de elementos finitos (FEM) o análisis de elementos finitos (FEA) es una técnica numérica computarizada que permite calcular los parámetros y soluciones de los dispositivos electromagnéticos dentro de un conjunto de condiciones de contorno especificadas. Se puede usar para calcular diferentes magnitudes mecánicas y electromagnéticas como por ejemplo la densidad del flujo, los enlaces de flujo, la inductancia, el torque, la fuerza electromotriz inducida, etc.

En el método FEA, el dispositivo electromagnético implementado se descompone o se discretiza en muchos elementos pequeños (por ejemplo, triángulos en 2D) para poder resolver el conjunto de ecuaciones algebraicas de Maxwell sobre una región finita del espacio. También ofrece una gran flexibilidad para el análisis de diversas formas geométricas, propiedades de materiales y condiciones de contorno en diversas regiones de la máquina en estudio [8].

Además, FEA proporciona información detallada sobre los efectos no lineales de la máquina en función de su geometría y propiedades del material. Este enfoque de

modelado permite realizar una descripción precisa de la máquina eléctrica ya que el circuito magnético es modelado por una malla de elementos pequeños.

1.4.1.1 Softwares de análisis mediante el método de elementos finitos

En general, un software de análisis por elementos finitos debe cumplir tres fases para obtener la solución del modelo del motor implementado. Estas fases son:

- Pre-procesamiento en donde se define el modelo de elementos finitos y sus condiciones de contorno.
- Solución del análisis, en donde se ejecuta la solución del modelo de elementos finitos implementado.
- Post-procesamiento en donde se utiliza herramientas de visualización de datos y gráficas para el análisis de resultados.

En el desarrollo del presente proyecto se investiga tres softwares que emplean el método de elementos finitos para analizar un sistema multifísico como es un sistema electromagnético.

Los softwares revisados son: Ansys Maxwell, Comsol Multiphysics y Flux. Cada uno de estos softwares posee excelentes características que permiten la creación de modelos de sistemas electromagnéticos mediante elementos finitos. De la investigación realizada se presenta las características generales de cada uno para llegar a la selección del software que permita la elaboración del presente proyecto.

1.4.1.1.1 Ansys Maxwell

Ansys Maxwell es un paquete de software de simulación multidominio que permite modelar y diseñar sistemas electromecánicos. Se puede generar modelos de componentes de FEA de alta fidelidad para ser incluidos en Simplorer de modo que se pueda verificar que el rendimiento electromagnético cumpla con los requisitos generales de diseño del sistema. Dentro del paquete de software descrito, se incluye otra herramienta denominada Simplorer la cual emplea un enfoque de diseño esquemático que permite representar y combinar modelos eléctricos, magnéticos, mecánicos, hidráulicos, térmicos y de otros dominios permitiendo simular de forma rápida y precisa el comportamiento de circuitos y sistemas. Las múltiples técnicas que ofrece Simplorer permiten crear diseños de sistemas multidominio, analógico, digital, Ansys Maxwell utiliza una interfaz gráfica con un conjunto de menús, submenús y comandos que permiten realizar el diseño geométrico del motor, selección del tipo de material de construcción, operaciones de mallado y el tipo de excitación eléctrica para cada componente. También permite establecer las condiciones del contorno dentro de las cuales se desarrollará la simulación. La figura 1.1 permite

visualizar la ventana de Ansys Maxwell en la cual se muestra cada una de las secciones que conforman la interfaz de diseño.

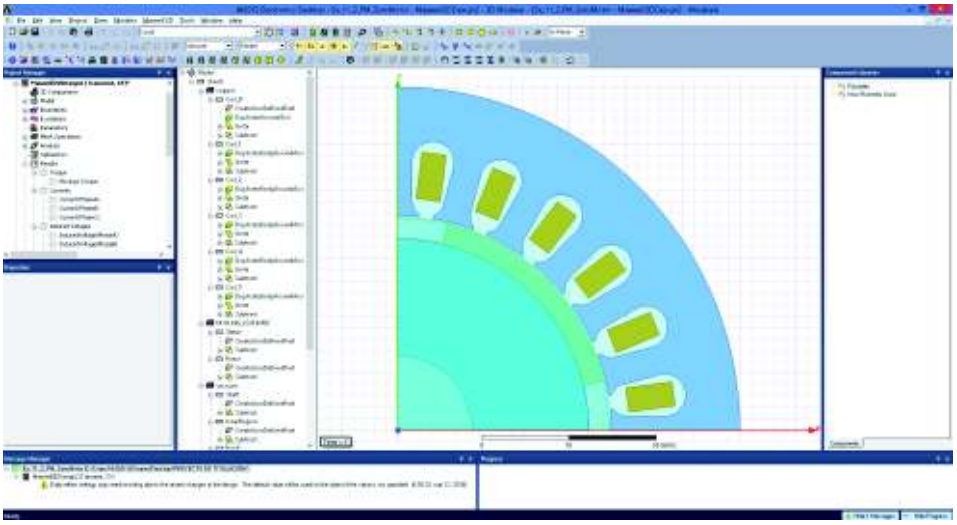


Figura 1.1 Interfaz de trabajo del Software Ansys Maxwell 2D

1.4.1.1.2 Comsol Multiphysics

COMSOL Multiphysics es un software de propósito general basado en el método de elementos finitos que permite modelar aplicaciones de ingeniería, simular diseños y procesos basados en electromagnetismo, mecánica estructural, acústica, flujo de fluidos, transferencia de calor, entre otros. Los módulos complementarios y los productos LiveLink que permite la interconexión con Matlab, se conectan a través de un entorno de software integrado. Comsol maneja una interfaz gráfica en la cual se tiene herramientas y comandos para la elaboración del modelo que se requiere realizar el análisis. La figura 1.2 permite visualizar la interfaz de trabajo de Comsol Multiphysics.

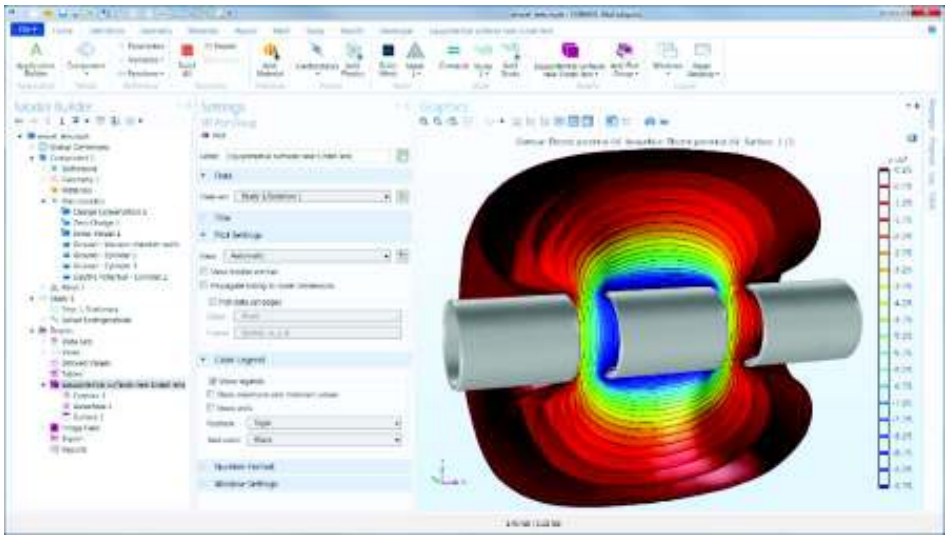


Figura 1.2 Interfaz de trabajo del software Comsol Multiphysics

1.4.1.1.3 Flux

El software de simulación Flux es una herramienta que permite realizar análisis, diseño, simulaciones y optimización de aplicaciones electromagnéticas y térmicas con capacidades de análisis multiparamétrico integradas. La interfaz abierta trata con diferentes dominios de simulación y permite realizar acoplamientos multifísicos. Al igual que Ansys y Comsol, el software Flux permite implementar la conexión con Matlab añadiendo la licencia específica para su utilización. La interfaz de trabajo de Flux como se visualiza en la figura 1.3, cuenta con una distribución de ventanas acorde a la necesidad del programador. Además, contiene una distribución de menús, submenús y comandos que permiten crear el modelo de elementos finitos requerido.

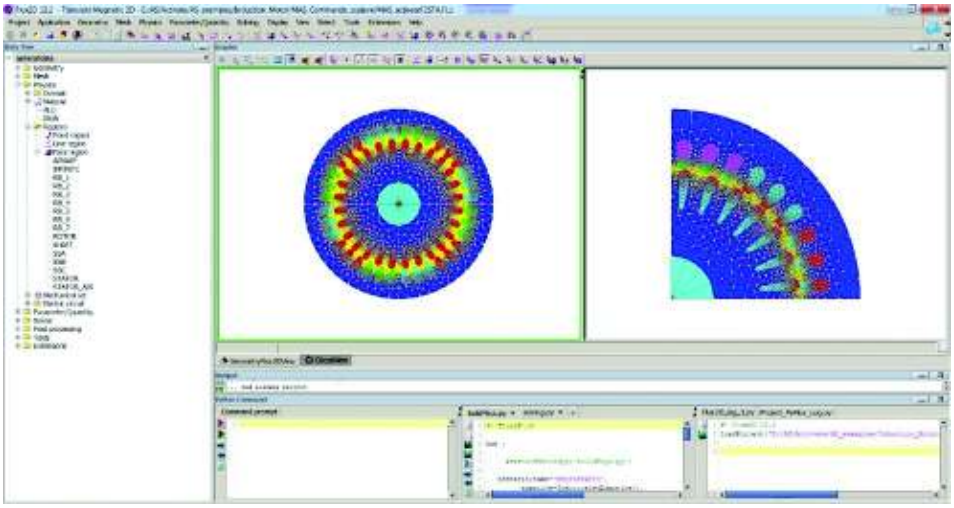


Figura 1.3 Interfaz de trabajo del software Flux

A continuación, se presenta un cuadro comparativo de las características de cada uno de los softwares descritos.

Tabla 1.1 Características de los softwares de F.E.A.

Ansys Maxwell	**Comsol Multiphysics**	**Flux**
Interfaz gráfica amigable	Interfaz gráfica amigable	Interfaz gráfica amigable
Exige elevados recursos computacionales.	Exige elevados recursos computacionales.	Exige elevados recursos computacionales.
Existe variedad de información en la web.	NO existe variedad de información en la web.	Existe poca información en la web.
Permite co-simulación con Matlab	Permite co-simulación con Matlab	Permite co-simulación con Matlab

Posee software de análisis rápido de los modelos creados. Ejemplo: RMxprt	NO posee software de análisis rápido de los modelos creados.	NO posee software de análisis rápido de los modelos creados.

Al revisar cada una de las características mencionadas en la tabla 1.1, se procede a seleccionar el software Ansys Maxwell, el cual se adapta a las necesidades que se requiere para el desarrollo del presente proyecto.

1.4.1.2 Ansys Maxwell

El software Ansys Maxwell 2D permite obtener soluciones del comportamiento de las variables eléctricas y mecánicas como son voltajes inducidos, potencia eléctrica, potencia mecánica, velocidad angular, campo electromagnético, líneas de flujo magnético, etc. para un modelo creado con materiales apropiadamente seleccionados, condiciones de fuente y contorno aplicando las ecuaciones de Maxwell sobre una región finita del espacio.
Hay dos modos con los cuales se puede describir la geometría de un elemento eléctrico utilizando Maxwell 2D, los cuales son:

- Modo cartesiano (XY)
- Modo simétrico alrededor del eje Z (RZ)

El modelo cartesiano (XY) representa la sección transversal de un dispositivo que se extiende en la dirección del eje Z. Por lo tanto, el modelo geométrico creado se extiende perpendicularmente al plano XY en el que está siendo modelado como se visualiza en la figura 1.4.

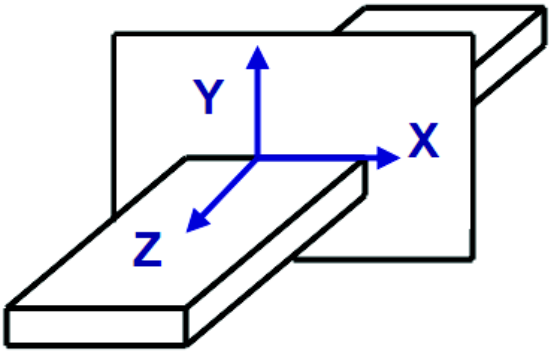


Figura 1.4 Creación del modelo cartesiano XY en Ansys Maxwell 2D [4]

El modelo simétrico alrededor del eje Z (RZ) representa una sección transversal de un dispositivo que se gira 360° alrededor de un eje de simetría (eje z). La sección transversal para este caso se dibuja en el plano RZ como se visualiza en la figura 1.5.

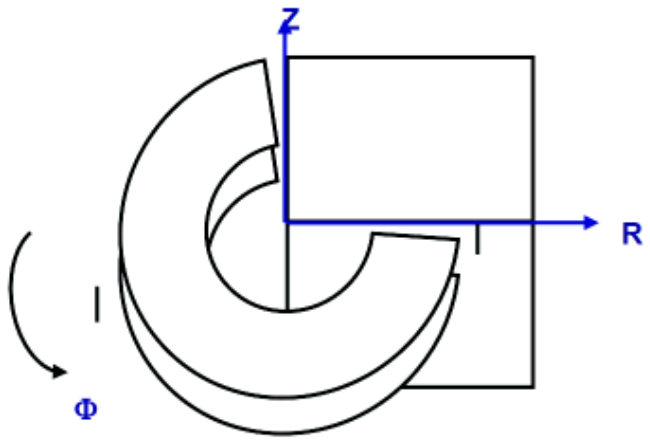


Figura 1.5 Creación del modelo simétrico alrededor del eje Z (RZ) en Maxwell 2D [4]

Las diferentes ecuaciones diferenciales utilizadas por el software Maxwell 2D para la solución de los modelos implementados se las detalla a continuación:

Ley de Gauss para el campo eléctrico

$$\nabla \cdot D = \rho$$

Ecuación 1.1 Ley de Gauss para el campo eléctrico

En donde:

∇ es la divergencia.

D es la densidad de flujo eléctrico.

ρ es la densidad de carga en el medio interior a la superficie cerrada.

Ley de Gauss para el campo magnético

$$\nabla \cdot B = 0$$

Ecuación 1.2 Ley de Gauss para el campo magnético

En donde:

B es la densidad de flujo magnético o llamada también inducción magnética.

Ley de la inducción de Faraday-Lenz

$$\nabla x E = -\frac{\partial B}{\partial t}$$

Ecuación 1.3 Ley de la inducción de Faraday-Lenz

En donde:

E es el campo eléctrico.

∂B derivada de la inducción magnética.

Ley de Ampere

$$\nabla x H = J + \frac{\partial D}{\partial t}$$

Ecuación 1.4 Ley de Ampere

En donde:

H es el vector intensidad de campo magnético.

J es la densidad de corriente eléctrica.

Con el objetivo de resolver el conjunto de ecuaciones algebraicas generadas, el software Ansys Maxwell elabora un mallado adaptativo y automático en forma de pequeños triángulos para el modelo elaborado. El conjunto de todos los triángulos se conoce como la malla de elementos finitos del modelo o simplemente la malla. Un ejemplo de mallado automático desarrollado en Ansys Maxwell 2D se visualiza en la figura 1.6.

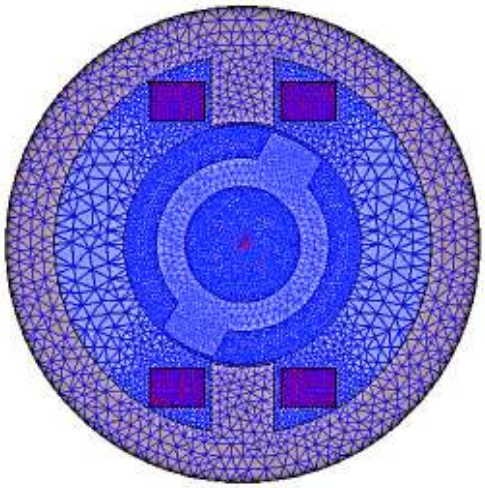


Figura 1.6 Creación del mallado para un modelo de motor en Ansys Maxwell 2D [4]

1.4.2 Sistema de accionamiento de un motor sincrónico de imanes permanentes

El sistema de accionamiento y control de un motor sincrónico de imanes permanentes está compuesto varios componentes fundamentales los cuales son: motor síncrono de imanes permanentes (PMSM), inversor trifásico y fuente DC, controlador escalar (V/F) y la realimentación de variables eléctricas (Ej.: Sensor de posición). Estos componentes se interconectan de tal manera que se pueda controlar la velocidad y torque de la PMSM. El esquema utilizado en el desarrollo del presente proyecto se muestra en el esquema de la figura 1.7.

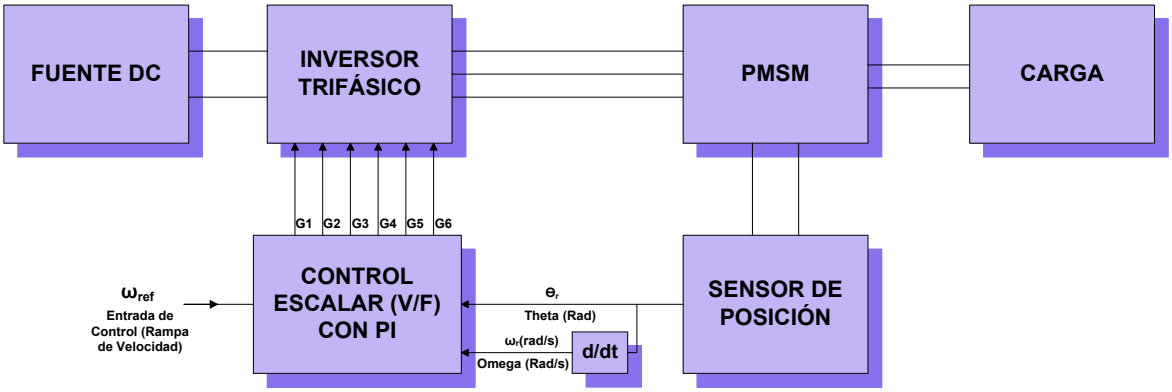


Figura 1.7 Sistema de accionamiento de una PMSM

Cada uno de los componentes mostrados en la figura 1.7 desempeña un papel importante como se explica a continuación:

1.4.3 Motor sincrónico de imanes permanentes

El motor síncrono de imanes permanentes (PMSM) es una clase de motor que utiliza imanes permanentes en lugar de electroimanes en el rotor para producir el campo magnético. El estator está compuesto por un conjunto de devanados similar al de un motor de inducción trifásico [9].

Por el tipo de construcción, el motor de imanes permanentes posee mayores ventajas para ser utilizado en diversas aplicaciones domésticas, industriales, médicas, aeroespaciales, etc.

Entre las principales características de un motor de imanes permanentes se puede mencionar [10]:

- Complejidad media de construcción.
- Alta confiabilidad sin desgaste de escobillas.

- Alta eficiencia
- Se impulsa por inversores y controladores multifase.
- Es muy apropiado para el control de posición.

1.4.3.1 Materiales de los imanes permanentes

A partir de los años 70 y 80, se ha desarrollado materiales magnéticos de diversas características constructivas con lo cual los motores de excitación con imanes permanentes han ido desplazando a los motores trifásicos de inducción por sus características de dinámica superior, tamaño, peso y eficiencia [11].

La selección adecuada de las propiedades constructivas del imán permanente influye directamente en el rendimiento del motor, por lo cual se requiere un conocimiento adecuado para la comprensión del funcionamiento de este tipo de motores.

Los primeros materiales magnéticos se los fabricaba con acero endurecido que se magnetizaban fácilmente, pero podían contener muy poca energía magnética y eran fácil de desmagnetizar. Con el trascurso del tiempo, nuevos materiales magnéticos fueron apareciendo como aluminio-níquel y aleaciones de cobalto (Alnico), ferrita de estroncio o ferrita de bario (ferrita), samario cobalto (SmCo) y neodimio-hierro-boro (NdFeB).

Los imanes de samario-cobalto y de neodimio-hierro-boro son considerados como imanes de tierras raras. Los imanes samario-cobalto poseen niveles de densidad de flujo más altos pero son muy costosos. Los imanes neodimio-hierro-boro son los imanes que más se utilizan en la actualidad para la construcción de motores de imanes permanentes.

Para la selección de un tipo de imán es importante analizar la curva de comportamiento de la densidad de flujo vs el campo de magnetización [12]. La figura 1.8 permite visualizar las características mencionadas para los diferentes tipos de materiales magnéticos.

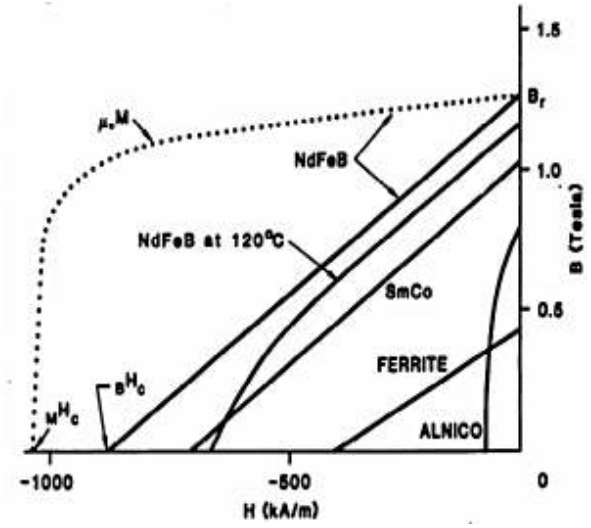


Figure 1.8 Densidad de flujo versus campo de magnetización [12]

1.4.3.2 Clasificación de los motores de imanes permanentes

De manera general, los motores de imanes permanentes se clasifican:

- Por la dirección del flujo de campo
- Por la distribución de densidad de flujo
- Motores de imanes permanentes de campo radial

1.4.3.2.1 Por la dirección del flujo de campo

Los motores de imanes permanentes se clasifican según la dirección del flujo de campo en motores de campo radial y motores de campo axial.

Los motores de campo radial se caracterizan porque las líneas de campo se distribuyen a lo largo del radio de motor (dirección radial) y la excitación es producida por los imanes permanentes. Este tipo de topología es la más utilizada y algunas de sus diversas configuraciones se visualizan en la figura 1.9.

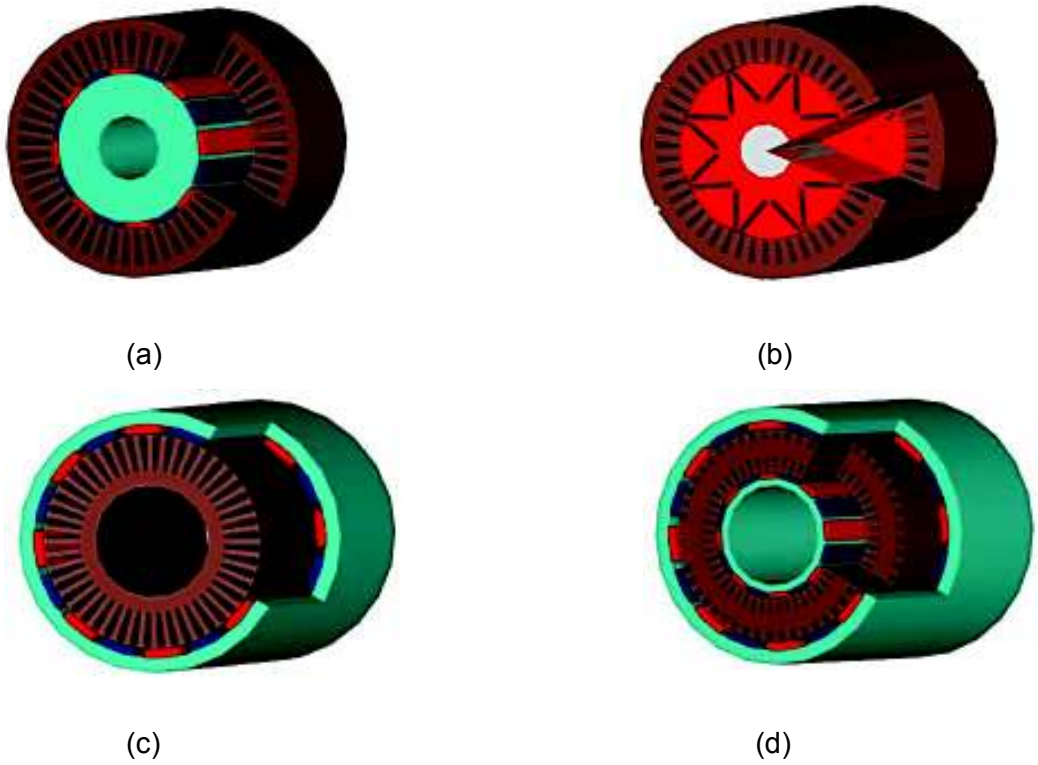


Figure 1.9 Máquinas eléctricas de flujo radial. a) Estator externo e imanes sobre el rotor. b) Estator externo e imanes en el interior del rotor. c) Estator interno. d) Motores de doble rotor [13]

Los motores de campo axial son aquellos en los que el flujo magnético es producido por los imanes permanentes en dirección perpendicular al radio del motor o dirección del eje de rotación.

Este tipo de topología no ha sido muy común, pero se han aparecido en ciertas aplicaciones donde no se puede usar los motores de flujo radial.

Las topologías más comunes de los motores de flujo axial se pueden visualizar en la figura 1.10, las cuales se clasifican según la posición del rotor y estator.

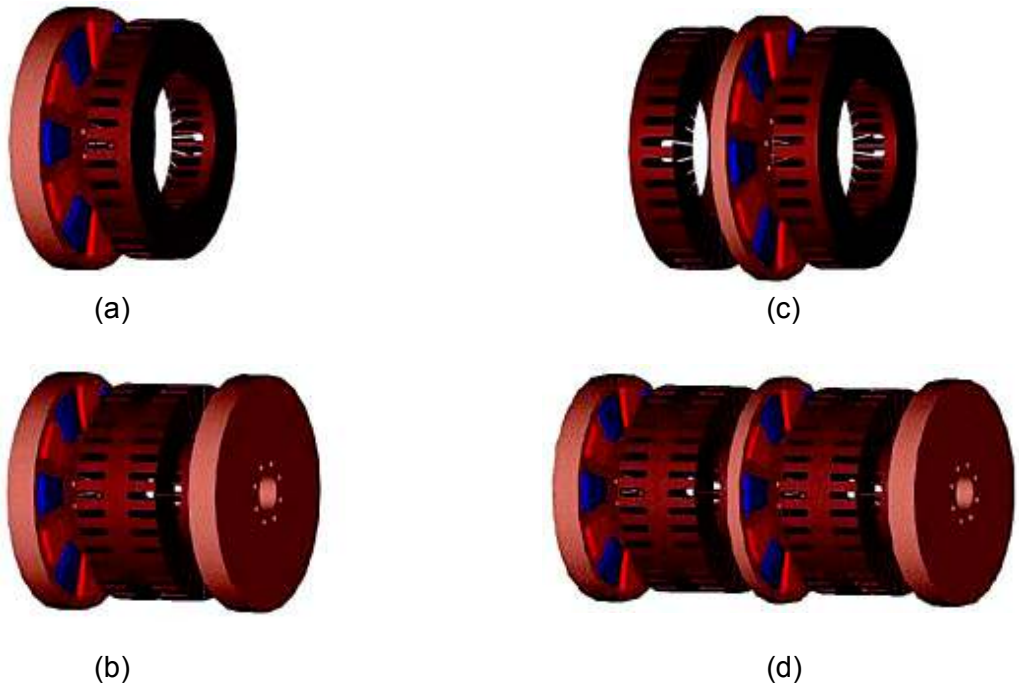


Figure 1.10 Máquinas eléctricas de flujo axial. a) Un rotor y un estator. b) Dos rotores y un estator. c) Dos estatores y un rotor. d) Estructura de múltiples etapas, dos bloques de estator y tres bloques de rotor [13]

1.4.3.2.2 Por la distribución de densidad de flujo

Los motores de imanes permanentes se clasifican en función de la distribución de la densidad de flujo y la forma de excitación en:

- Motores sincrónicos de imanes permanentes (PMSM), y
- Motores Brushless de imanes permanentes (BLDC).

Los motores sincrónicos de imanes permanentes se alimentan de fuerzas electromotrices sinusoidales ya poseen un devanado trifásico con bobinados distribuidos en su estator y son capaces de generar fuerzas electromotrices sinusoidales. Es por esta razón que sus principales características son:

- Distribución sinusoidal del flujo magnético en el entrehierro
- Formas de onda de corriente sinusoidal
- Distribución sinusoidal de los conductores del estator.

Las formas de onda para este tipo de motores se visualizan en la figura 1.11.

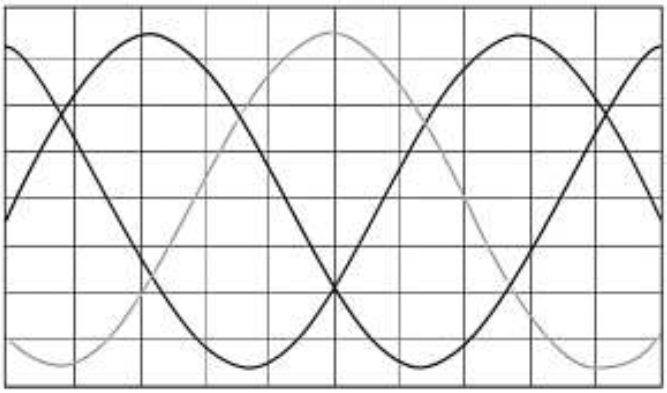


Figure 1.11 Formas de onda sinusoidales para un motor de imanes permanentes [11]

Los motores BLDC poseen un bobinado distribuido de forma de polos salientes o entrantes y está diseñado para generar fuerzas electromotrices trapezoidales. Las características principales de este tipo de motores son:

- Distribución rectangular del flujo magnético en el espacio de aire
- Forma de onda de corriente rectangular
- Devanados concentrados del estator

Las formas de onda para este tipo de motores se visualizan en la figura 1.12.

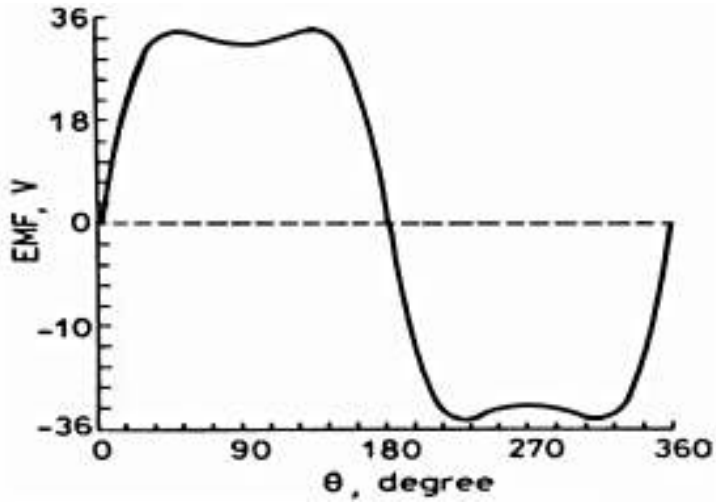


Figure 1.12 Formas de onda trapezoidales para un motor BLDC [11]

1.4.3.2.3 Motores de imanes permanentes de campo radial

Existen dos maneras de ubicar los imanes permanentes en el rotor de un motor de campo radial. Dependiendo de la ubicación de los imanes, se los puede clasificar en motores de imanes permanentes superficiales y motores de imanes permanentes interiores.

Los motores de imanes permanentes superficiales poseen un rotor con imanes permanentes pegados a su superficie mediante adhesivos. Esta característica permite que sean fáciles de construir, y los polos especialmente sesgados se magnetizan fácilmente. Además, el montaje de los imanes en la superficie permite minimizar el torque de engranaje. La figura 1.13 permite visualizar un ejemplo de un motor con imanes permanentes superficiales.

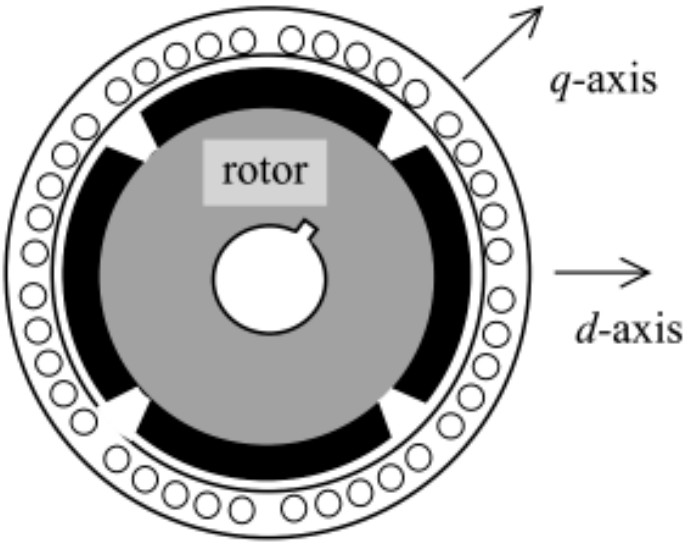


Figure 1.13 Sección de un motor síncrono con imanes permanentes pegados a la superficie del rotor [11]

La configuración de motores con imanes permanentes superficiales se la utiliza en aplicaciones de baja velocidad debido a la limitación de que los imanes se desprendan durante las operaciones de alta velocidad.

Para este tipo de motores, las inductancias en ambos ejes son prácticamente iguales [14], por lo tanto, para un motor con imanes permanentes en su superficie se debe considerar que:

$$L_d = L_q$$

Dado que el material del imán permanente posee una permeabilidad relativa cercana a la unidad, el espacio libre efectivo es uniforme y grande (aproximadamente 8mm). La construcción del núcleo puede ser hecha de hierro sólido o mediante láminas de hierro

unidas para hacer más simple su fabricación [15]. Los imanes colocados en la superficie del rotor producen una densidad de flujo dirigida radialmente a través del espacio de aire. Esta densidad de flujo reacciona con las corrientes de los devanados colocados en las ranuras del estator para producir torque.

Para el caso de los motores de imanes permanentes interiores, cada imán está ubicado dentro del rotor. Este tipo de configuración permite que la PMSM pueda funcionar a alta velocidad.

En este tipo de motores se considera que la inductancia del eje q es mayor que la inductancia del eje d, debido a que la parte del imán permanente es equivalente al aire en el cálculo del circuito magnético [14].

$$L_q > L_d$$

En la figura 1.14 se muestra dos configuraciones de máquinas síncronas de imanes permanentes ubicados en el interior del rotor.

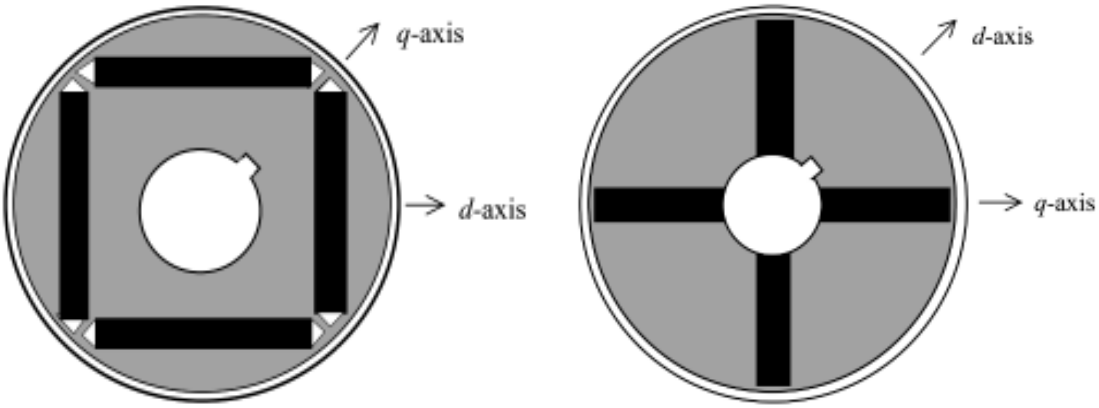


Figure 1.14 a) PMSM con imanes tangenciales internos, b) PMSM con imanes radiales internos [11]

1.4.4 Sensores de posición

Los motores sincrónicos de imanes permanentes requieren de sensores que permitan medir la posición del eje del rotor. Para medir con exactitud dicho parámetro, existen diversos tipos de sensores entre los principales podemos mencionar: el potenciómetro, el LVDT (Linear Variable Differential Transformer), encoders ópticos y resolvers. Los sensores más comúnmente utilizados son los encoders y los resolvers los cuales se los analiza a continuación.

1.4.4.1 Encoder óptico

El encoder óptico es el sensor de posición más utilizado en la actualidad, el cual consiste en un disco giratorio, un emisor de luz y un sensor foto detector que se encuentran acoplados al eje giratorio del motor. El disco giratorio está conformado por sectores opacos y transparentes que permiten el paso o la interrupción de la luz hacia el foto detector cuando el rotor gira, generando pulsos o señales digitales de salida. En la figura 1.15 se puede visualizar los componentes que conforman un encoder óptico.

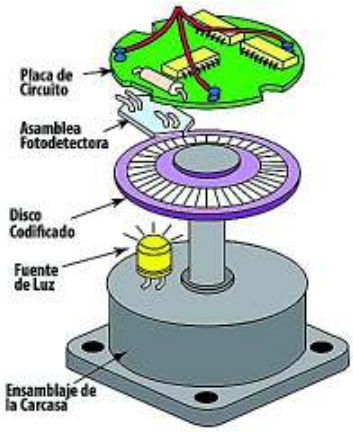


Figure 1.15 Componentes de un encoder óptico [24]

El encoder óptico ofrece la ventaja de la interfaz digital ya que los datos pueden ser adquiridos directamente por las entradas digitales de un microcontrolador, PLC, etc.

Hay dos tipos de encoder ópticos:

- Encoder incremental

El encoder incremental posee una buena precisión y es fácil de acoplar al eje del motor, pero presenta perdida de información cuando el motor se encuentra detenido. Además, para obtener una medida más precisa se requiere que el motor se detenga en el mismo punto de partida lo cual no ocurre en la mayoría de veces. El encoder incremental más común utiliza dos canales de salida denominados A y B con sectores posicionados a 90° grados de fase para detectar la posición y el sentido de giro.

La fase relativa entre A y B permite conocer el sentido de giro ya que, si A se adelanta a B por ejemplo, el disco está girando en sentido de las agujas del reloj y viceversa.

El número de pulsos y la fase relativa entre las dos señales permiten medir la posición y sentido de rotación.

Algunos encoders de cuadratura incluyen un tercer canal de salida denominado Z el cual suministra un pulso por cada revolución para determinar de manera precisa una posición de referencia. La figura 1.16 permite visualizar las tres señales emitidas por el disco giratorio del encoder con lo cual se puede determinar la posición y sentido de giro del eje del rotor.

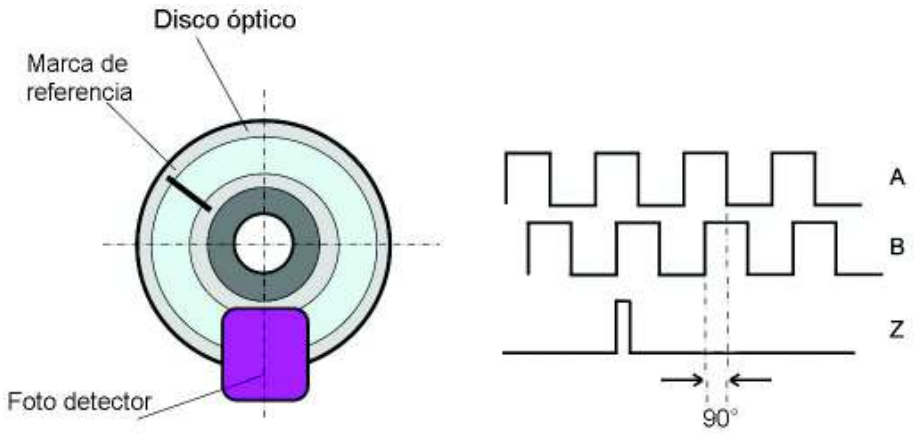


Figure 1.16 Pulsos generados por un encoder óptico incremental [16]

- Encoder absoluto

El encoder absoluto determina la posición exacta del rotor durante el encendido sin necesidad de una marca de referencia, además puede rotar indefinidamente y la posición puede medirse incluso si el motor se detiene. Los encoders absolutos poseen una salida digital y su resolución se define por el número de bits en la salida como se visualiza en la figura 1.17.

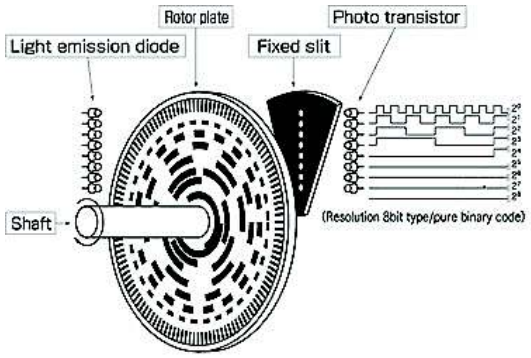


Figure 1.17 Estructura simplificada de un encoder absoluto de 8 bits de resolución [17]

Se los utiliza en aplicaciones donde un dispositivo está inactivo durante largos períodos de tiempo o se mueve a baja velocidad. También se recomiendan en sistemas que requieren retener información de posición cuando ocurre un corte de energía.

1.4.4.2 Resolver de Posición

El resolver de posición o llamado también transformador rotativo, trabaja bajo el principio de un transformador. El devanado primario (U_{ref}) es colocado en el rotor y dependiendo del ángulo de avance del eje del rotor (Θ), la tensión inducida en los devanados secundarios colocados en el estator y desplazados 90° (U_{sin} y U_{cos}) sería diferente. La posición se la calcula utilizando estos dos voltajes.

La figura 1.18 muestra los componentes de un resolver de posición.

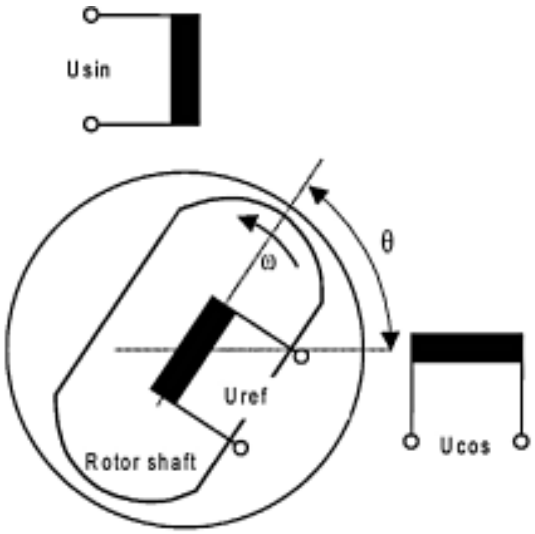


Figure 1.18 Estructura de un resolver [18]

La frecuencia de los voltajes U_{sin} y U_{cos} es igual a la frecuencia del voltaje de referencia U_{ref}. y sus amplitudes varían acorde a la variación del ángulo del eje del rotor Θ. Considerando que U_{ref} y U_{cos} se encuentran alineados, se genera un voltaje total en uno de los devanados de salida y un voltaje de cero en el otro devanado de salida y viceversa como se visualiza en la figura 1.19.

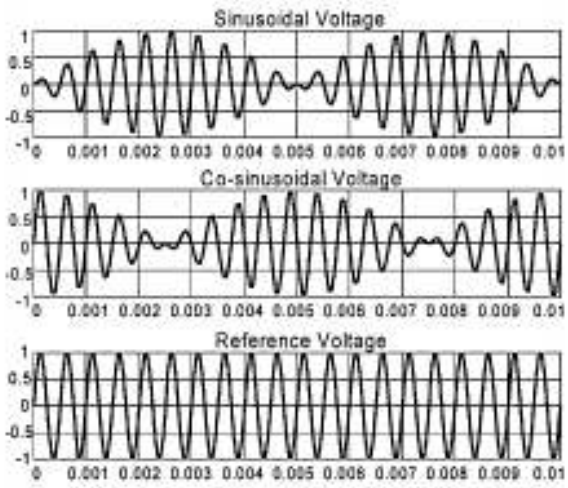


Figure 1.19 Señales de salida y de referencia de un resolver [18]

Finalmente, el valor del ángulo de giro del eje del rotor se lo obtiene aplicando la tangente inversa de la relación de los voltajes de salida U_{sin} y U_{cos} como se visualiza en la ecuación 1.5.

$$\theta = \tan^{-1}\left(\frac{U_{sin}}{U_{cos}}\right)$$

Ecuación 1.5 Ecuación para calcular el ángulo de giro en un resolver.

1.4.5 Inversor trifásico

El motor sincrónico de imanes permanentes se alimentará desde un inversor conectado a una fuente de voltaje continua DC. El control se lo realiza regulando el flujo de corriente a través del estator del motor. Por lo tanto, el controlador a ser implementado generará señales digitales para el accionamiento de las compuertas de los IGBTs que constituyen el inversor.

1.4.5.1 Inversor

Los inversores son circuitos que convierten un voltaje de corriente continua que proviene generalmente de un rectificador trifásico o de una batería a un voltaje de corriente alterna, de frecuencia y magnitud variables. Son utilizados en variadores de velocidad ajustables ya que poseen una forma de onda definida en los terminales de salida que puede ser variable o constante dependiendo de la aplicación [15]. El esquema de conexión de un inversor trifásico alimentado por un rectificador trifásico se visualiza en la figura 1.20.

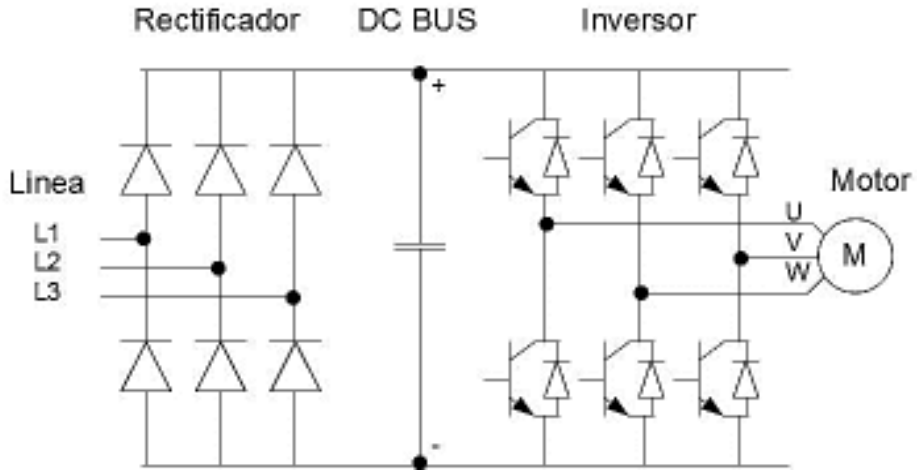


Figure 1.20 Inversor trifásico alimentado por un rectificador trifásico

Los inversores trifásicos están compuestos por seis semiconductores de potencia conectados a una fuente de voltaje de corriente continua como se visualizó en la figura 1.20.

En la actualidad, existen diversos tipos de dispositivos de potencia que deben ser seleccionados de acuerdo a los requisitos de operación del inversor como son tiristores, transistores de unión bipolar (BJT), transistores de efecto de campo MOS (MOSFET), transistores bipolares de puerta aislada (IGBT) y tiristores de apagado de compuerta (GTO). Las capacidades de los dispositivos utilizados como interruptores en función de la potencia requerida y frecuencia de switcheo permiten ubicarlos dentro de un campo de aplicación como se visualiza en la figura 1.21.

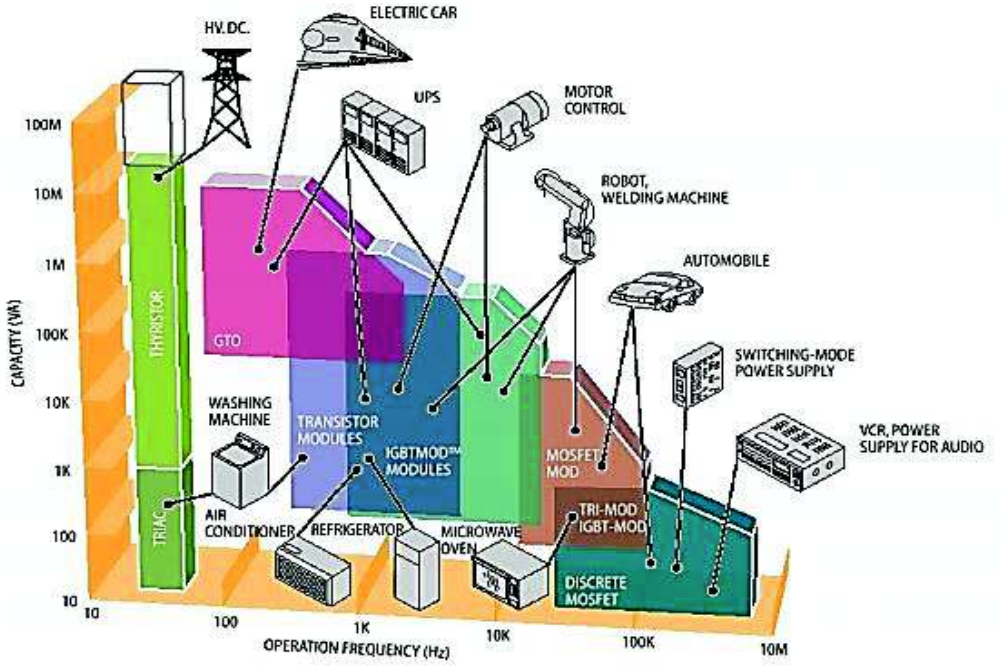


Figure 1.21 Comparación de las características de operación de los dispositivos de potencia y sus aplicaciones [27]

Los dispositivos empleados en la implementación de este proyecto son los IGBTs los cuales tienen una amplia aceptación para el control de motores y otras aplicaciones en el rango de baja y media potencia. Al usar cargas inductivas se debe colocar diodos en paralelo con cada uno de los IGBTs para así poder disipar la energía encerrando la corriente almacenada sobre la misma carga.

1.4.6 Diagrama vectorial, modelo matemático y circuito equivalente de un motor síncrono de imanes permanentes.

Para obtener el modelo matemático de la máquina síncrona de imanes permanentes se inicia con el modelado de la máquina síncrona de excitación independiente de la cual se obtendrán las ecuaciones generales [26].

Para realizar el modelado se deberán tomar en cuenta las siguientes consideraciones:

- Para el modelado se asume una máquina de dos polos. Para una máquina de un número mayor de polos se deberá considerar el número de pares de polos en las ecuaciones finales obtenidas.
- La máquina síncrona de excitación independiente posee devanados de amortiguamiento que serán tomados en cuenta para el modelado.
- El estator se encuentra constituido por un bobinado trifásico distribuido simétricamente.
- La influencia de la temperatura en los parámetros de la máquina y en la des magnetización, así como las pérdidas en el núcleo no son tomados en cuenta.
- Los parámetros del rotor están relacionados hacia el lado de estator.
- Solo se considera la onda fundamental para el cálculo de las inductancias en el campo del entrehierro.

El diagrama de una máquina síncrona de excitación independiente se muestra en la figura 1.22.

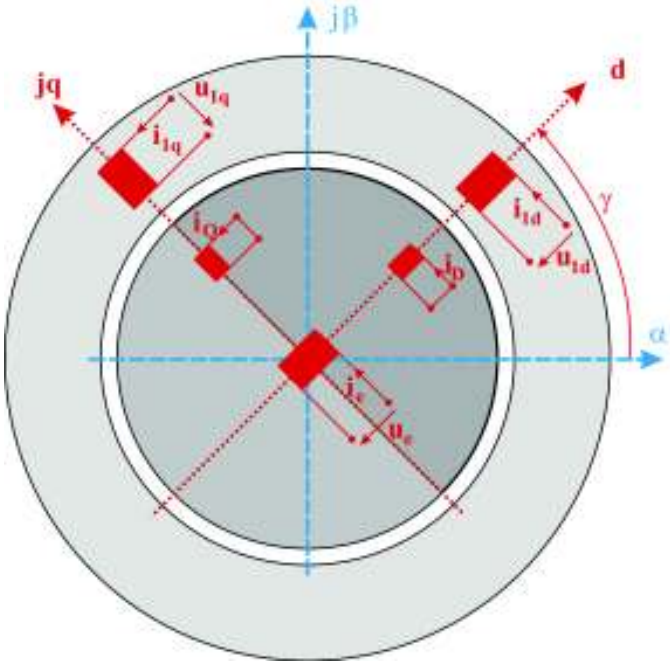


Figure 1.22 Diagrama de una máquina síncrona de excitación independiente [28]

Las ecuaciones que describen el modelo matemático de una máquina síncrona de excitación independiente se detallan a continuación.

$$u_{1d} = R_1 * i_{1d} + \frac{d\psi_{1d}}{dt} - \frac{d\gamma}{dt} * \psi_{1q}$$

Ecuación 1.6 Ecuación de voltaje de la máquina síncrona de excitación independiente para el *eje d* [28]

En donde:

u_{1d} es el voltaje del estator en el eje d.

R_1 es la resistencia del estator.

i_{1d} es la corriente del estator en el eje d.

$\frac{d\psi_{1d}}{dt}$ es la derivada del flujo del estator del eje d.

$\frac{d\gamma}{dt}$ es la derivada de la posición del rotor.

ψ_{1q} es el flujo del estator del eje q.

$$u_{1q} = R_1 * i_{1q} + \frac{d\psi_{1q}}{dt} + \frac{d\gamma}{dt} * \psi_{1d}$$

Ecuación 1.7 Ecuación de voltaje de la máquina síncrona de excitación independiente para el *eje q* [28]

En donde:

u_{1q} es el voltaje del estator en el eje q.

R_1 es la resistencia del estator.

i_{1q} es la corriente del estator en el eje q.

$\frac{d\psi_{1q}}{dt}$ es la derivada del flujo del estator del eje q.

$\frac{d\gamma}{dt}$ es la derivada de la posición del rotor.

ψ_{1d} es el flujo del estator del eje d.

$$0 = R_Q * i_Q + \frac{d\psi_Q}{dt}$$

Ecuación 1.8 Ecuación de voltaje del bobinado de amortiguamiento del eje *q* [28]

En donde:

R_Q es la resistencia del bobinado de amortiguamiento del rotor para el eje q.

i_Q es la corriente del bobinado de amortiguamiento del rotor para el eje q.

$\frac{d\psi_Q}{dt}$ es la derivada del flujo del bobinado de amortiguamiento del rotor para el eje q.

$$0 = R_D * i_D + \frac{d\psi_D}{dt}$$

Ecuación 1.9 Ecuación de voltaje del bobinado de amortiguamiento del eje *d* [28]

En donde:

R_D es la resistencia del bobinado de amortiguamiento del rotor para el eje d.

i_D es la corriente del bobinado de amortiguamiento del rotor para el eje d.

$\frac{d\psi_D}{dt}$ es la derivada del flujo del bobinado de amortiguamiento del rotor para el eje d.

$$u_e = R_e * i_e + \frac{d\psi_e}{dt}$$

Ecuación 1.10 Ecuación de voltaje del bobinado de campo del rotor [28]

En donde:

u_e es el voltaje del bobinado de campo del rotor.

R_e es la resistencia del bobinado de campo del rotor.

i_e es la corriente del bobinado de campo del rotor.

$\frac{d\psi_e}{dt}$ es la derivada del flujo del bobinado de campo del rotor.

Las ecuaciones que relacionan los enlaces de flujo, respectivamente y con la interacción magnética de cada uno de los devanados de la máquina síncrona de excitación independiente se las determina a partir de los circuitos equivalentes mostrados en la figura 1.23 tanto en el eje d como el eje q del sistema de coordenadas del rotor.

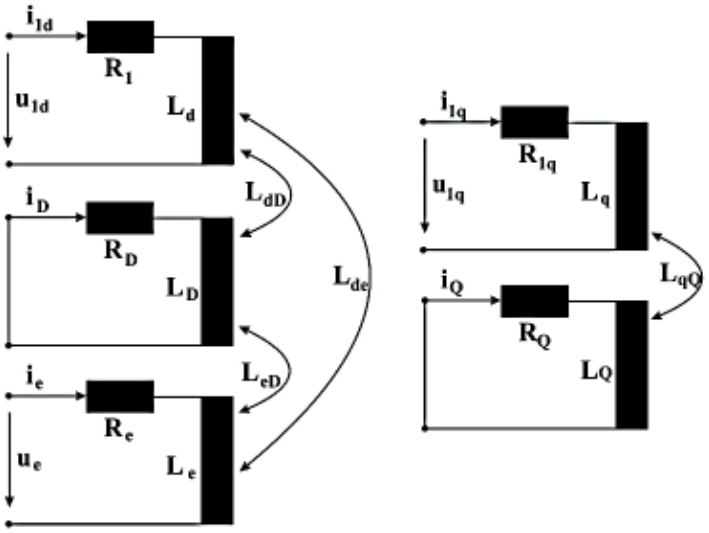


Figure 1.23 Circuitos equivalentes de los devanados de la máquina síncrona de excitación independiente [28]

A partir de los circuitos mostrados en la figura 1.23 se obtiene las ecuaciones de las diferentes componentes de flujo.

$$\psi_{1d} = L_d * i_{1d} + L_{dD} * i_D + L_{de} * i_e$$

Ecuación 1.11 Ecuación para determinar la componente de flujo del eje d [28]

En donde:

ψ_{1d} es el flujo del estator del eje d.

L_d es la inductancia del bobinado del estator del eje d.

i_{1d} es la corriente del estator en el eje d.

L_{dD} es la inductancia mutua del bobinado de amortiguamiento del rotor y bobinado del estator para el eje d.

i_D es la corriente del bobinado de amortiguamiento del rotor en el eje d.

L_{de} es la inductancia mutua del bobinado del estator y bobinado del rotor en el eje d.

i_e es la corriente del bobinado de campo del rotor.

$$\psi_{1q} = L_q * i_{1q} + L_{qQ} * i_Q$$

Ecuación 1.12 Ecuación para determinar la componente de flujo del eje q [28]

En donde:

ψ_{1q} es el flujo del estator del eje q.

L_q es la inductancia del bobinado del estator del eje q.

i_{1q} es la corriente del estator en el eje q.

L_{qQ} es la inductancia mutua del bobinado de amortiguamiento del rotor y bobinado del estator para el eje q.

i_Q es la corriente del bobinado de amortiguamiento del rotor en el eje q.

$$\psi_D = L_D * i_D + L_{dD} * i_{1d} + L_{eD} * i_e$$

Ecuación 1.13 Ecuación del flujo proporcionado por el bobinado de amortiguamiento en el eje d [28]

En donde:

ψ_D es el flujo del bobinado de amortiguamiento del rotor del eje d.

L_D es la inductancia del bobinado de amortiguamiento del rotor del eje d.

i_{D} es la corriente del bobinado de amortiguamiento del rotor del eje d.

L_{dD} es la inductancia mutua del bobinado de amortiguamiento del rotor y bobinado del estator para el eje d.

i_{1d} es la corriente del estator en el eje d.

L_{eD} es la inductancia mutua del bobinado de campo del rotor y bobinado de amortiguamiento del rotor en el eje d.

i_e es la corriente del bobinado de campo del rotor.

$$\psi_Q = L_Q * i_Q + L_q * i_{1q}$$

Ecuación 1.14 Ecuación del flujo proporcionado por el bobinado de amortiguamiento en el eje q [28]

En donde:

ψ_Q es el flujo del bobinado de amortiguamiento del rotor del eje q.

L_Q es la inductancia del bobinado de amortiguamiento del rotor del eje q.

i_{Q} es la corriente del bobinado de amortiguamiento del rotor del eje q.

L_q es la inductancia del bobinado del estator para el eje q.

i_{1q} es la corriente del estator en el eje q.

$$\psi_e = L_e * i_e + L_{de} * i_{1d} + L_{eD} * i_D$$

Ecuación 1.15 Ecuación del flujo proporcionado por el bobinado de campo [28]

En donde:

ψ_e es el flujo del bobinado de campo del rotor.

L_e es la inductancia del bobinado de campo del rotor.

i_{e} es la corriente del bobinado de campo del rotor.

L_{de} es la inductancia mutua del bobinado de campo del rotor y bobinado del estator para el eje d.

i_{1d} es la corriente del estator en el eje d.

L_{eD} es la inductancia mutua del bobinado de campo del rotor y bobinado de amortiguamiento del rotor en el eje d.

i_D es la corriente del bobinado de amortiguamiento del rotor para el eje d.

El torque electromagnético de la maquina síncrona de excitación independiente está dado por la ecuación 1.16 [28].

$$M_e = \frac{3}{2} * p * I_m\{\psi_{1d} * i_{1q} - \psi_{1q} * i_{1d}\}$$

Ecuación 1.16 Ecuación de torque electromagnético de la máquina síncrona de excitación independiente [28]

En donde:

M_e es el torque electromagnético.

p es el número de par de polos.

I_m es la corriente total resultante obtenida a partir de i_{1d} e i_{1q}.

Para el caso de un motor síncrono de imanes permanentes, el modelo matemático se reduce debido a que este tipo de motor no cuenta con devanados de amortiguamiento y el devanado de campo no existe. El diagrama de un motor de imanes permanentes de polos superficiales se muestra en la figura 1.24.

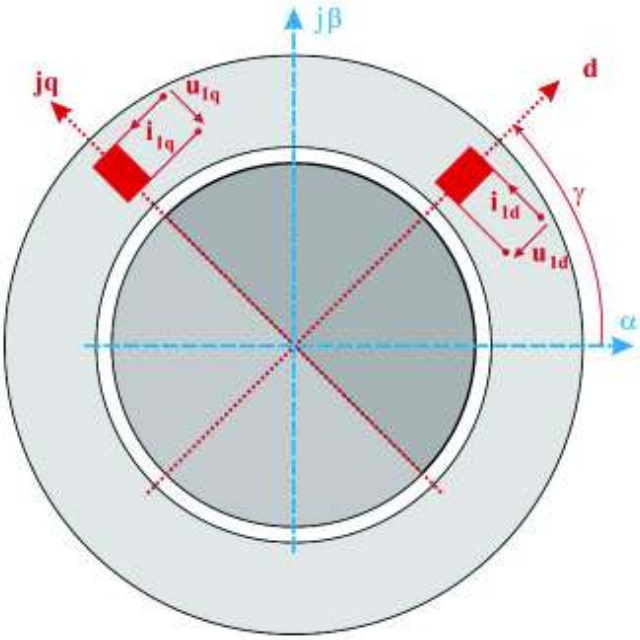


Figure 1.24 Diagrama vectorial de un motor síncrono de imanes permanentes de polos superficiales [28]

Para este tipo de motor, las ecuaciones se reducen debido a que se cumple la condición:

$$L_D = L_Q = L_{dD} = L_{dQ} = L_{eD} = L_{eQ} = 0$$

$$R_D = R_Q = R_e = 0$$

De esta manera las ecuaciones que permiten modelar el motor síncrono de imanes permanentes son:

$$u_{1d} = R_1 * i_{1d} + L_1 * \frac{di_{1d}}{dt} - \frac{d\gamma}{dt} * L_1 * i_{1q}$$

Ecuación 1.17 Ecuación de voltaje de la máquina síncrona de imanes permanentes en el eje d [28]

En donde:

u_{1d} es el voltaje del estator en el eje d.

R_1 es la resistencia del estator.

i_{1d} es la corriente del estator en el eje d.

L_1 es la inductancia del bobinado del estator.

$\frac{di_{1d}}{dt}$ es la derivada de la corriente del estator del eje d.

$\frac{d\gamma}{dt}$ es la derivada de la posición del rotor.

i_{1q} es la corriente del estator en el eje q.

$$u_{1q} = R_1 * i_{1q} + L_1 * \frac{di_{1q}}{dt} + \frac{d\gamma}{dt} * L_1 * i_{1d} + \frac{d\gamma}{dt} * \psi_{d0}$$

Ecuación 1.18 Ecuación de voltaje de la máquina síncrona de imanes permanentes en el eje q [28]

En donde:

u_{1q} es el voltaje del estator en el eje q.

R_1 es la resistencia del estator.

i_{1q} es la corriente del estator en el eje q.

L_1 es la inductancia del bobinado del estator.

$\frac{di_{1q}}{dt}$ es la derivada de la corriente del estator del eje q.

$\frac{d\gamma}{dt}$ es la derivada de la posición del rotor.

i_{1q} es la corriente del estator en el eje q.

i_{1d} es la corriente del estator en el eje d.

ψ_{d0} es el flujo producido por los imanes permanentes.

$$\psi_{1d} = L_1 * i_{1d} + \psi_{d0}$$

Ecuación 1.19 Ecuación para determinar la componente de flujo del estator del eje d [28]

En donde:

ψ_{1d} es el flujo del bobinado del estator en el eje d.

L_1 es la inductancia del bobinado del estator.

i_{1d} es la corriente del estator en el eje d.

ψ_{d0} es el flujo producido por los imanes permanentes.

$$\psi_{1q} = L_1 * i_{1q}$$

Ecuación 1.20 Ecuación para determinar la componente de flujo del estator del eje q [28]

En donde:

ψ_{1q} es el flujo del bobinado del estator en el eje q.

L_1 es la inductancia del bobinado del estator.

$i_{1\mathrm{q}}$ es la corriente del estator en el eje q.

El torque electromagnético de la maquina síncrona de imanes permanentes está dado por la ecuación 1.21.

$$M_e = \frac{3}{2} * p * \left((L_1 * i_{1d} + \psi_{d0}) * i_{1q} - L_1 * i_{1d} * i_{1q}\right)$$

Ecuación 1.21 Ecuación de torque electromagnético de la máquina síncrona de imanes permanentes [28]

Si los imanes permanentes se encuentran ubicados en la superficie del rotor, las inductancias el eje *q* y el eje *d* son aproximadamente iguales (L_d=L_q). En estado estable, la ecuación de torque puede ser simplificada como se muestra en la ecuación 1.22.

$$M_e = \frac{3}{2} * p * \psi_{d0} * i_{1q}$$

Ecuación 1.22 Ecuación de torque electromagnético en estado estable con L_d=L_q [28]

Como puede notarse en la ecuación 1.22, el eje *d* de la corriente no tiene ningún efecto en el torque y la corriente mínima del estator es alcanzada a un torque constante cuando i_d=0.

$$i_{dref} = 0$$

Ecuación 1.23 Ecuación para aplicar el control en la región de torque constante [28]

Finalmente, el diagrama vectorial y el circuito equivalente de un motor de imanes permanentes de polos salientes se muestran en las figuras 1.25 y 1.26 respectivamente.

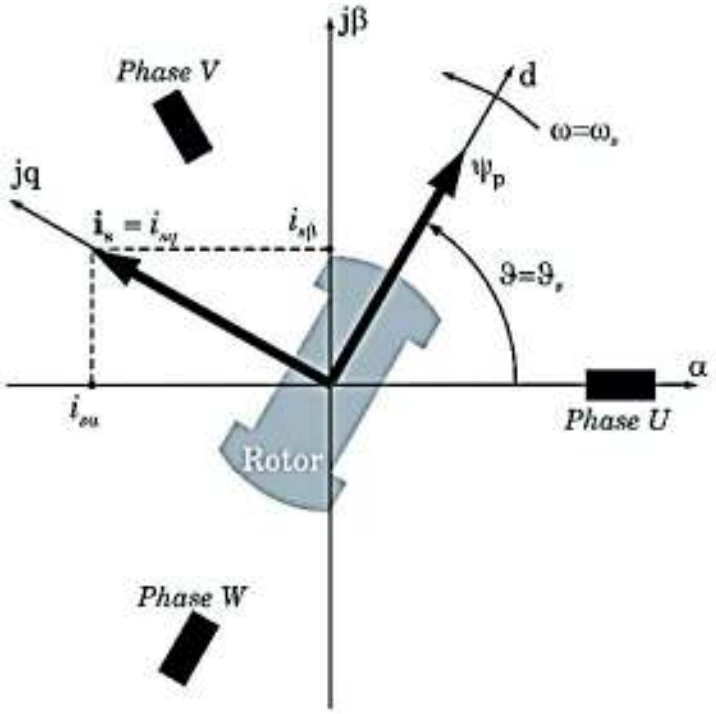


Figure 1.25 Diagrama vectorial de la PMSM de polos salientes con Id =0 [26]

Con la finalidad de obtener un circuito equivalente común para los ejes longitudinal y transversal se aceptará que la misma inductancia es válida para ambos. La ecuación 1.24 determina el voltaje del estator [26].

$$u_s = R_s * i_s + L_s * \frac{di_s}{dt} + j\omega\psi_P$$

Ecuación 1.24 Ecuación de voltaje del estator de un motor síncrono de imanes permanentes.

En donde:

u_s es el voltaje del estator.

R_s es la resistencia del estator.

i_s es la corriente del estator.

$L_s$es la inductancia del bobinado del estator.

$\frac{di_s}{dt}$ es la derivada de la corriente del estator.

ω es la velocidad el rotor.

ψ_P es el flujo generado por los imanes permanentes.

A partir de la ecuación 1.24 se obtiene el circuito equivalente simplificado para un motor síncrono de imanes permanentes el cual se muestra en la figura 1.26.

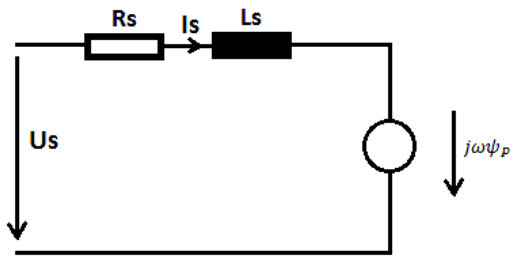


Figure 1.26 Circuito equivalente del PMSM [26]

1.4.7 Control por frecuencia

En base a la expresión de la velocidad sincrónica de un motor de imanes permanentes (Ecuación 1.25), se observa que la velocidad puede variar en forma proporcional al cambio de la frecuencia del voltaje aplicado al estator del motor, por tal razón el uso de un inversor de frecuencia variable permitirá controlar la velocidad del motor.

$$N_S = \frac{120 * f}{p}$$

Ecuación 1.25 Ecuación para determinar la velocidad sincrónica

En donde:

f, es la frecuencia del voltaje aplicado al motor.

p, es el número de polos del rotor.

Al variar la frecuencia el flujo mutuo máximo también se ve afectado ya que varía inversamente proporcional a la frecuencia. Si el motor trabaja a flujo distinto que el nominal, esto conlleva el aumento de pérdidas y por esta razón es necesario mantenerlo constante. Para conseguir esto se varía el voltaje de alimentación de manera proporcional a la frecuencia en todo el rango de velocidad del motor para mantener constante la relación entre la magnitud y la frecuencia del voltaje aplicado. Este método se denomina control escalar V/F.

A partir de las ecuaciones 1.6, 1.7 que relacionan los voltajes y corrientes del eje d y q, reescribiendo las ecuaciones en forma escalar en el dominio del tiempo se obtiene:

$$u_{1d} = R_1 * i_{1d} + \frac{d\psi_{1d}}{dt} - \omega_e * \psi_{1q}$$

Ecuación 1.26 Ecuación de voltaje de la máquina síncrona de excitación independiente para el *eje d*

$$u_{1q} = R_1 * i_{1q} + \frac{d\psi_{1q}}{dt} + \omega_e * \psi_{1d}$$

Ecuación 1.27 Ecuación de voltaje de la máquina síncrona de excitación independiente para el *eje q*

En régimen de estado estable, la variación del enlace de flujo es cero, y para una mayor simplificación se asume que la resistencia del devanado del estator es despreciable. Teniendo en cuenta estas simplificaciones y la ecuación de vinculación de flujo 1.19 y 1.20 que relacionan las componentes de flujo del eje d, eje q y flujo de los imanes permanentes, las ecuaciones 1.26 y 1.27 se convierten en :

$$u_{1d} = -\omega_e * L_q * i_q$$

Ecuación 1.28 Ecuación voltaje-velocidad del eje d

$$u_{1q} = \omega_e * L_d * i_d + \omega_e * \psi_{d0}$$

Ecuación 1.29 Ecuación voltaje-velocidad del eje q

En este punto, transformando la velocidad en frecuencia se reescribe la ecuación 1.28 y 1.29 de tal manera que relaciona voltaje/frecuencia:

$$\frac{u_{1d}}{f} = -2\pi * (L_{1q} * i_{1q})$$

Ecuación 1.30 Ecuación del control V/F aplicado al eje d

$$\frac{u_{1q}}{f} = 2\pi * (L_{1d} * i_{1d} + \psi_{d0})$$

Ecuación 1.31 Ecuación del control V/F aplicado al eje q

Partiendo de las ecuaciones del circuito equivalente, de manera general el control V/F de un motor síncrono de imanes permanentes está dado por las siguientes ecuaciones:

$$\frac{u_s}{f} = 2\pi(L_s * i_s + \psi_P)$$

$$V_s \approx K * (L_s * i_s + \psi_P) * f$$

$$V_s \approx K * (\psi_S + \psi_P) * f$$

$$\frac{V_s}{f} = CONSTANTE$$

Ecuación 1.32 Ecuaciones del control escalar V/F

En el método de control escalar V/F, la frecuencia del flujo magnético del estator se establece de acuerdo con la velocidad deseada del rotor mientras que la magnitud del voltaje del estator se ajusta para mantener constante la relación entre estos dos parámetros. En este método no se utiliza ningún control sobre los ángulos de los voltajes o sobre los vectores de corriente, de ahí el nombre de control escalar. La relación V/F se calcula a partir de los valores nominales de los parámetros de voltaje y frecuencia de la PMSM. Manteniendo una relación V/F constante entre la amplitud y la frecuencia de las formas de onda de voltaje trifásico, el flujo de estator del PMSM se puede mantener relativamente constante en estado estacionario. Sin embargo, en la práctica, un perfil V/F típico no es constante en todo el rango de velocidad del motor como se muestra en la figura 1.27.

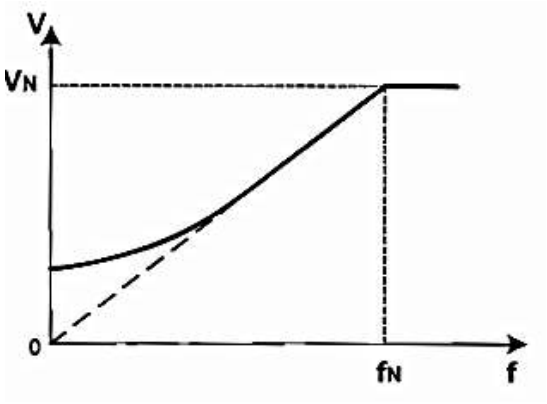


Figure 1.27 Perfil V/F real para el control de una PMSM

Cuando se trabaja a bajas velocidad (bajas frecuencias) la caída de voltaje en la resistencia del estator se vuelve considerable por lo cual el flujo mutuo tiende a disminuir y con esto el torque máximo. Para poder compensar la reducción de torque máximo se incrementa la relación V/F aumentando el voltaje aplicado, por tal motivo el crecimiento del voltaje con respecto a la frecuencia ya no es lineal como se observa en la figura 1.27. Esta compensación se la denomina compensación IxR.

1.4.8 Control de velocidad

El control de velocidad en lazo cerrado permite que un sistema sea robusto ante perturbaciones, diferentes perfiles de torque de carga y ante cambios de referencia de velocidad. Los sistemas de control de velocidad de lazo cerrado requieren de componentes de retroalimentación (sensores) que permitan medir la velocidad del motor en todo momento.

1.4.8.1 Implementación de lazo de control de velocidad

Para realizar el control de velocidad en primer lugar se requiere de un sensor de velocidad con el cual se mide la velocidad a la cual está girando el motor. Esta señal irá a un comparador que tiene como entradas la velocidad de referencia y la velocidad medida para obtener el error que deberá ser controlado por medio del regulador PI. De esta manera se genera una señal de control para integrarla al perfil V/F.

Para velocidades superiores a la velocidad nominal, el motor operaría en la región de debilitamiento del flujo que para el desarrollo del presente proyecto no se lo toma en cuenta.

El comparador calcula la diferencia entre la velocidad de referencia (Ω_{ref}) y la velocidad medida por el sensor (Ω), produciendo un error el cual se ingresa al controlador PI.

Los controladores PI están compuestos por dos partes, una proporcional y una integral que generan una acción de control.

Control Proporcional, el controlador proporcional realiza la multiplicación del error e(t) por una constante proporcional K_P con el objetivo de llevar el error a cero, sin embargo en estado estable siempre está presente un error entre la salida del sistema y el valor de referencia. La ecuación 1.33 representa el controlador proporcional aplicado al error.

$$u(t) = K_P * e(t)$$

Ecuación 1.33 Ecuación del control proporcional

Control Integral, el controlador integral tiene como objetivo eliminar el error presente en estado estable provocado por la acción proporcional. El error e(t) en estado estacionario es integrado en un periodo determinado, multiplicado por la constante K_P y dividido por una constante de tiempo integral T_i. La ecuación 1.34 representa el controlador integral aplicado al error.

$$u(t) = \frac{K_P}{T_i} \int e(t) * dt$$

Ecuación 1.34 Ecuación del control integral

El esquema general de un controlador PI se muestra en la figura 1.28.

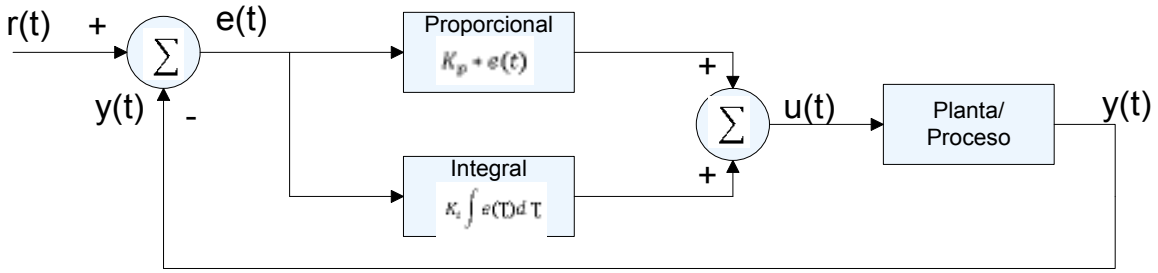


Figure 1.28 Diagrama de bloques de una planta con controlador Proporcional-Integral

Las variables que intervienen en el esquema general mostrado en la figura 1.26 son:

- r(t), representa la señal de referencia.
- e(t), representa el error obtenido de la diferencia entre la señal de referencia y la señal medida por el sensor.
- Kp, representa la ganancia proporcional de controlador PI.
- Ki, representa la ganancia integral del controlador PI.
- u(t), representa la acción de control generada por el controlador PI para ser ingresada a la planta o proceso.
- y(t), representa la salida del proceso.

2 METODOLOGÍA

El desarrollo del presente proyecto es una investigación aplicada donde se revisa la teoría de análisis mediante elementos finitos y su aplicación al diseño y simulación de máquinas eléctricas. Para deducir si el método de análisis mediante elementos finitos permite obtener una respuesta más real del comportamiento del motor síncrono de imanes permanentes (PMSM), se realiza en primer lugar el modelado del motor utilizando las librerías de Matlab (análisis mediante modelo matemático aproximado), posteriormente se utiliza la herramienta RMXprt Desing (análisis sin elementos finitos), y finalmente Ansys Maxwell 2D (análisis con elementos finitos) con el objetivo de comparar los resultados de los 3 modelos implementados.

El control de velocidad y de torque de la máquina síncrona de imanes permanentes se basa en el método de Control escalar V/F, el cual es aplicable para controlar diversos tipos de motores eléctricos.

El método de control escalar V/F es aplicable en accionamientos con máquinas de inducción que no tenga alta dinámica, es decir es un control en lazo abierto. Para aplicaciones que exige un alto rendimiento dinámico, se recomienda el control vectorial [20].

La metodología empleada en el desarrollo del presente proyecto comprende 4 fases: teórica, diseño, simulación e implementación y una fase final que comprende las pruebas, validación y comparación de los resultados de funcionamiento de cada uno de modelos diseñados.

La fase teórica constituye la investigación para la creación del modelo de una PMSM mediante software de análisis por elementos finitos, del circuito de potencia y del controlador escalar. Se investigará la manera de realizar la co-simulación entre los softwares utilizados en el desarrollo del proyecto.

Utilizando la información y aprendizaje de la fase teórica, se procederá con la fase de diseño en el cual se construirá el modelo de una PMSM, se realizará el diseño e implementación del circuito de potencia y se realizará el diseño e implementación del control escalar V/F de velocidad y torque de una PMSM.

Realizadas las dos primeras fases se procederá con la fase de implementación y simulación en donde se obtendrán 3 modelos, el primero que corresponde a un modelo creado exclusivamente con el software Matlab/Simulink, el segundo mediante co-simulación entre RMxprt-Simplorer-Matlab, y el tercero mediante co-simulación Maxwell 2D-Simplorer-Matlab para la simulación del sistema de accionamiento completo de una PMSM.

Finalmente, se realizarán las simulaciones de los sistemas desarrollados como parte de la fase de validación, análisis de resultados y pruebas de funcionamiento. Los resultados de los tres modelos serán comparados con el objetivo de obtener datos medibles de la eficiencia en el análisis mediante la técnica de elementos finitos.

2.1 Modelado de una PMSM

2.1.1 Desarrollo del modelo de la PMSM utilizando Matlab/Simulink

El modelo de una PMSM es desarrollado utilizando el bloque *Permanent Magnet Synchronous Machine* que se encuentra ubicado dentro de las librerías de Matlab/simulink. Para acceder al bloque mencionado, se sigue la siguiente ruta: Simscape /SimPowerSystems/Specialized Technology Machines.

Una vez que el bloque ha sido añadido al proyecto de Simulink, se procede a configurar cada uno de los parámetros eléctricos del motor como se detalla en la tabla 2.1.

Tabla 2.1 Parámetros de configuración del bloque PMSM en Matlab/Simulink.

Máquina síncrona de imanes permanentes	
Numero de fases	3
Tipo de onda de alimentación	Sinusoidal
Tipo de rotor	Polos salientes
Variable mecánica de entrada	Torque
Resistencia de fase del estator (ohm)	2.01924
Inductancia Lq y Lq (H)	0.00396762
Inercia	0.00149257
Numero de par de polos	2

Configurando cada uno de los parámetros mencionados en la tabla 2.1, el modelo de la PMSM se encuentra listo para ser conectado con el resto de componentes configurados en Simulink.

2.1.2 Desarrollo del modelo de la PMSM utilizando RMxprt Desing

El desarrollo del modelo de la PMSM se lo realiza en primera instancia utilizando la herramienta de software RMxprt (por sus siglas en inglés, Rotating Machine Expert), que se despliega seleccionando el comando *Insert RMxprt Design* del menú *Project* de la ventana principal del software Ansys Electronics Desktop.

Ansys RMxprt es una herramienta de diseño que utiliza la teoría clásica del motor, teoría analítica y los métodos equivalentes de circuitos magnéticos para crear el modelo y dimensionamiento inicial del motor. RMxprt además permite configurar automáticamente un proyecto completo de Maxwell (2-D/3-D) que incluye geometría, materiales y condiciones de contorno para el respectivo análisis mediante elementos finitos. La configuración incluye las simetrías y excitaciones apropiadas con la topología del circuito de acoplamiento para el análisis de transientes electromagnéticos. [21] [22]

2.1.2.1 Ventana de selección de tipo de máquina

La herramienta de diseño RMxprt cuenta con modelos de máquinas rotativas AC y DC en los cuales se modifican los parámetros necesarios para obtener el modelo de máquina deseado. La figura 2.1 muestra el listado de máquinas que pueden ser elegidas en la ventana de selección del tipo de máquina. Para el desarrollo del presente trabajo se selecciona el tipo de máquina: *Adjust-Speed Synchronous Machine* que permite configurar una máquina sincrónica de imanes permanentes.

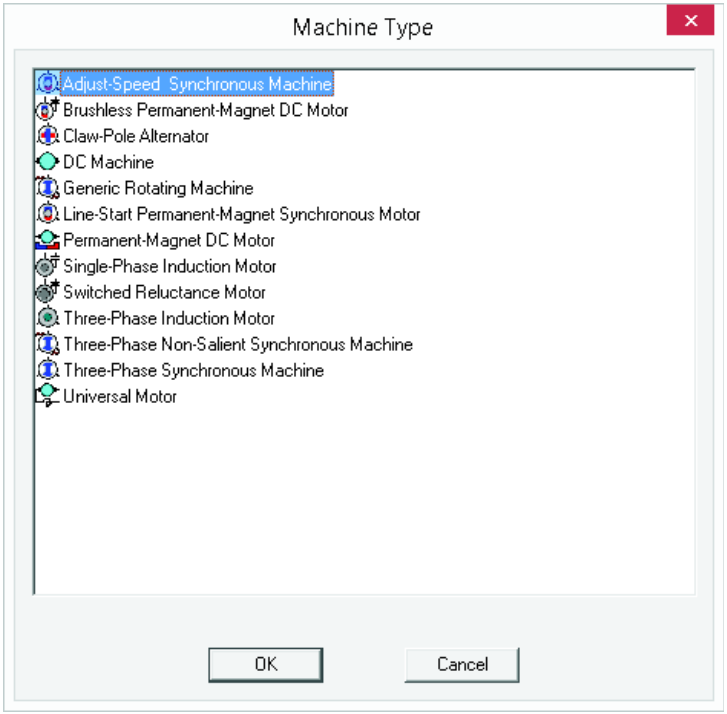


Figura 2.1 Ventana de selección del tipo de máquina

Una vez seleccionado el tipo de máquina se despliega una interfaz que incluye menús y submenús con varias funciones cada uno, las cuales se explican a continuación:

2.1.2.2 Ventana de Proyecto

La figura 2.2 muestra los menús y sub-menús que permiten realizar el diseño de cada uno de los componentes del motor.

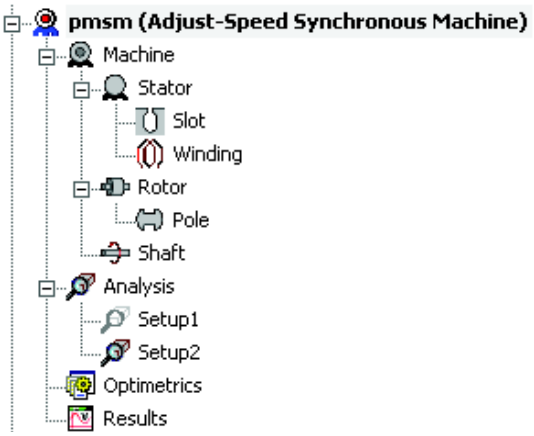


Figura 2.2 Ventana de diseño y configuración del motor utilizando RMxprt

El menú *Machine* permite realizar la configuración del estator, rotor, y eje del motor seleccionado.

El menú *Analysis* permite configurar los parámetros con los cuales se realizará la simulación.

El menú *Optometrics*, permite crear un conjunto de condiciones con el objetivo de optimizar el diseño del motor.

El menú *Results*, permite visualizar datos, así como gráficas del comportamiento del motor diseñado.

2.1.2.3 Ventana de configuración de la máquina

La ventana mostrada en la figura 2.3 permite configurar los parámetros generales de la máquina seleccionada.

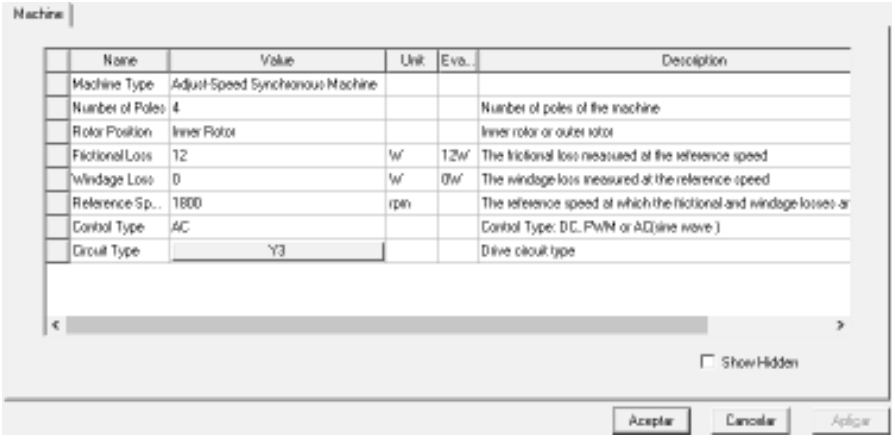


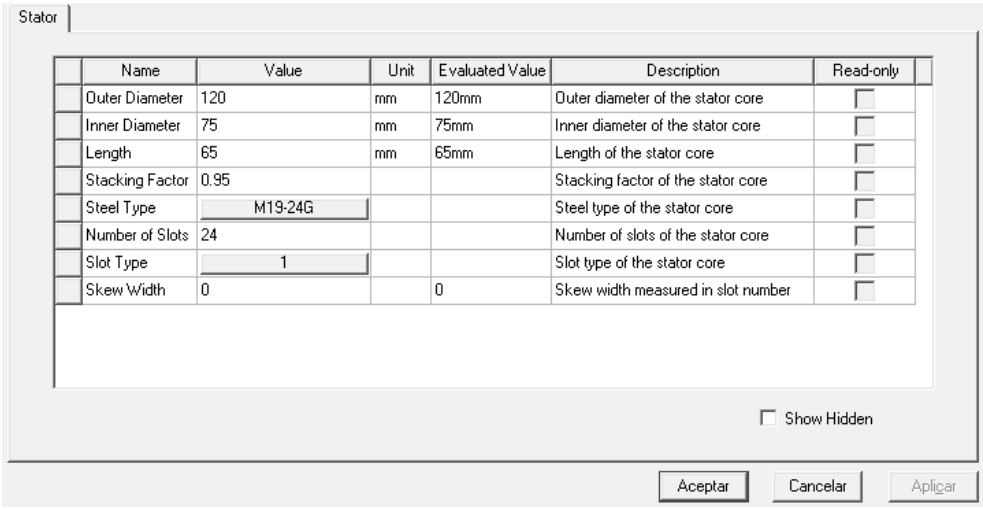
Figura 2.3 Ventana de configuración de la máquina

Para el desarrollo del presente proyecto, se toma como punto de partida una máquina de imanes permanentes que viene incluida en el software RMxprt la cual posee las siguientes características generales:

- Tipo de máquina: Adjust-Speed Synchronous Machine
- Número de polos: 4
- Posición del rotor: interno.
- Perdidas por fricción: 12Watts
- Velocidad de referencia: 1800 r.p.m.
- Tipo de control: AC (Ondas senoidales)
- Tipo de conexión: Y, trifásico.

2.1.2.4 Ventana de diseño del estator

El estator es un conjunto de láminas con ranuras en donde se acoplan los devanados de las fases de alimentación del motor. En la figura 2.4 se visualiza la ventana de diseño y configuración de los parámetros del estator.



Name	Value	Unit	Evaluated Value	Description	Read-only
Outer Diameter	120	mm	120mm	Outer diameter of the stator core	
Inner Diameter	75	mm	75mm	Inner diameter of the stator core	
Length	65	mm	65mm	Length of the stator core	
Stacking Factor	0.95			Stacking factor of the stator core	
Steel Type	M19-24G			Steel type of the stator core	
Number of Slots	24			Number of slots of the stator core	
Slot Type	1			Slot type of the stator core	
Skew Width	0		0	Skew width measured in slot number	

Figura 2.4 Ventana de diseño y configuración del estator

Los parámetros de diseño del estator de la máquina previamente seleccionada son:

- Diámetro exterior del estator: 120mm.
- Diámetro interno del estator: 75mm.
- Tamaño/longitud del estator: 65mm
- Tipo de acero de construcción del estator: M19-24G. Al configurar este parámetro se despliega una lista de materiales de los cuales se selecciona para asignar el material de construcción del estator. La ventana de selección del tipo de acero se visualiza en la figura 2.5.

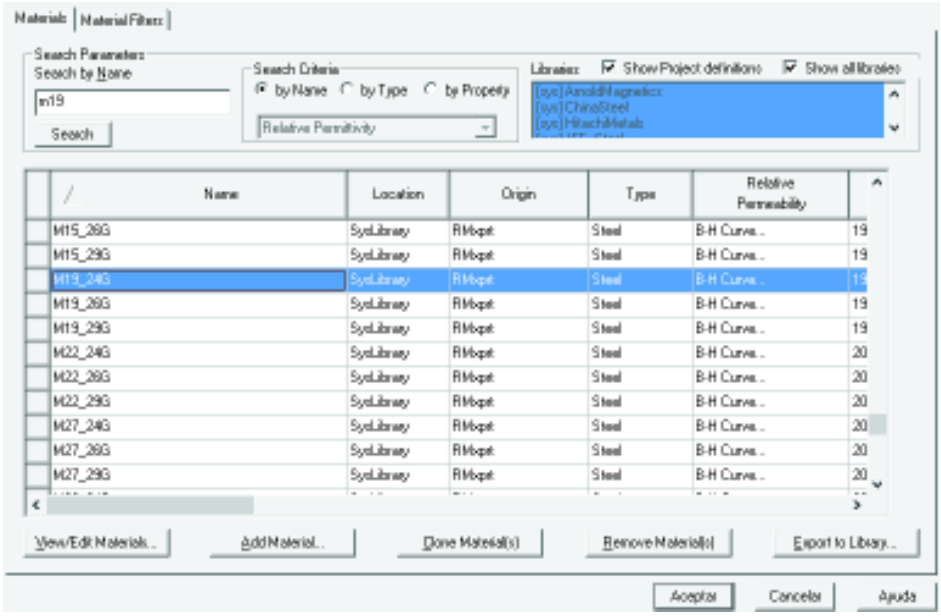


Figura 2.5 Ventana de selección del tipo de acero de construcción del estator

- Número de slots: 24
- Tipo de slot: 1. Esta opción permite escoger la forma de las ranuras del estator. Existen 4 tipos de slots para el diseño del estator como se observa en la figura 2.6.

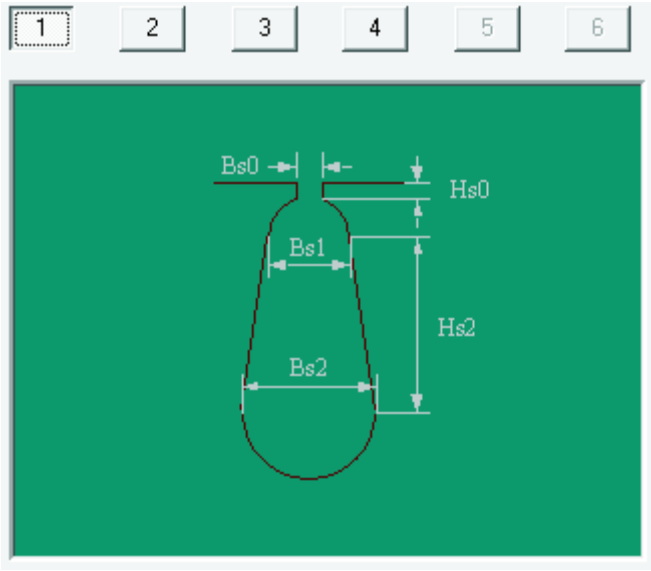


Figura 2.6 Ventana de selección del tipo de slot

La configuración de los parámetros del slot seleccionado se realizará en el submenú denominado *Slot.*

2.1.2.4.1 Slot

La ventana mostrada en la figura 2.7 permite configurar las dimensiones del slot seleccionado. El software permite crear un auto diseño de las dimensiones del slot seleccionando la opción *Auto Design* e introduciendo los parámetros Hs0 y Bs0.

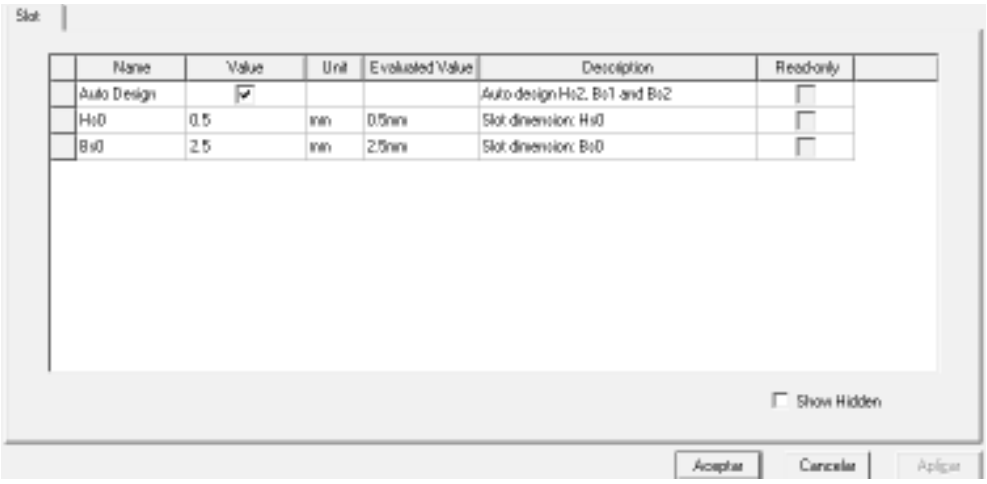


Figura 2.7 Ventana de configuración del tipo de slot seleccionado

2.1.2.5 Ventana de diseño del rotor

El rotor del motor seleccionado posee imanes permanentes ubicados en la superficie del rotor. El campo magnético de las bobinas del estator reacciona al campo magnético generado por los imanes del rotor, dando como resultado una fuerza que causa movimiento giratorio.

La ventana mostrada en la figura 2.8 permite realizar el diseño y configuración de los parámetros del rotor.

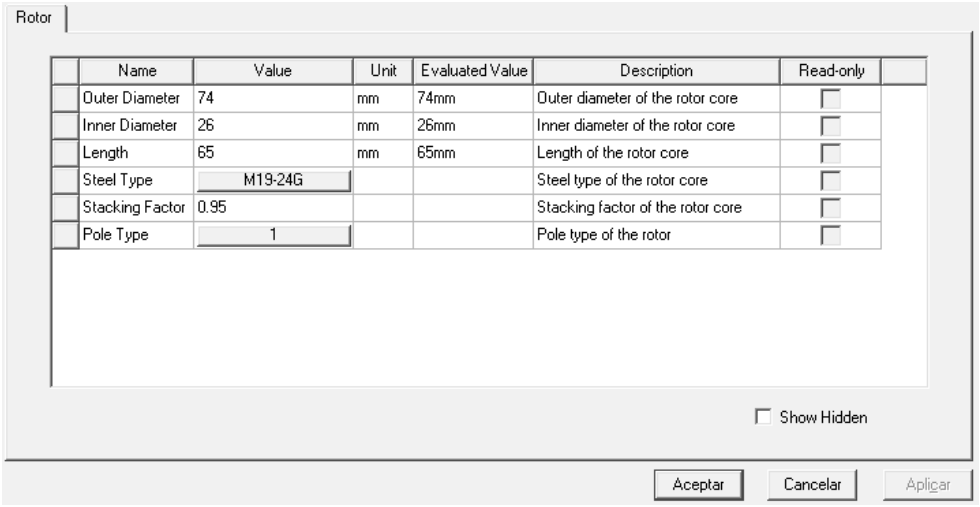


Figura 2.8 Ventana de diseño y configuración del rotor

Los parámetros de diseño del rotor de la máquina previamente seleccionada son:

- Diámetro exterior del rotor: 74mm.
- Diámetro interno del rotor: 26mm.
- Tamaño/longitud del estator: 65mm

- Tipo de acero de construcción del rotor: M19-24G. Al configurar este parámetro se despliega una lista de materiales de los cuales se seleccionó el material de construcción del rotor. En la figura 2.9 se visualiza la lista de materiales que viene incluida como biblioteca en RMxprt.

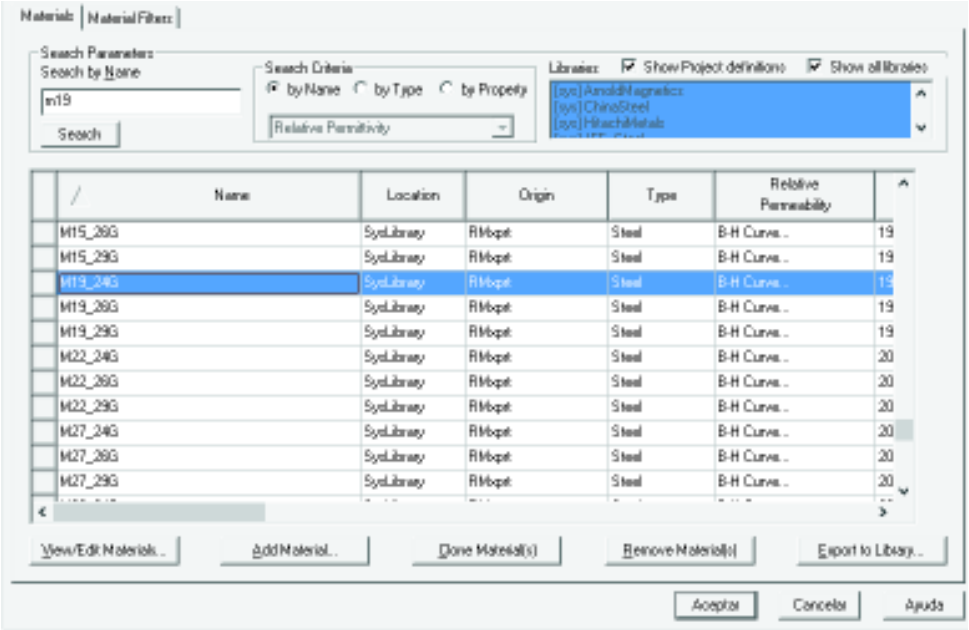


Figura 2.9 Ventana de selección del tipo de acero de construcción del rotor

- Tipo de polo: 1. Esta opción permite escoger la forma de los polos superficiales del rotor. Existen 4 tipos de formas como se muestra en la figura 2.10, de los cuales se encuentra seleccionada la opción número 1.

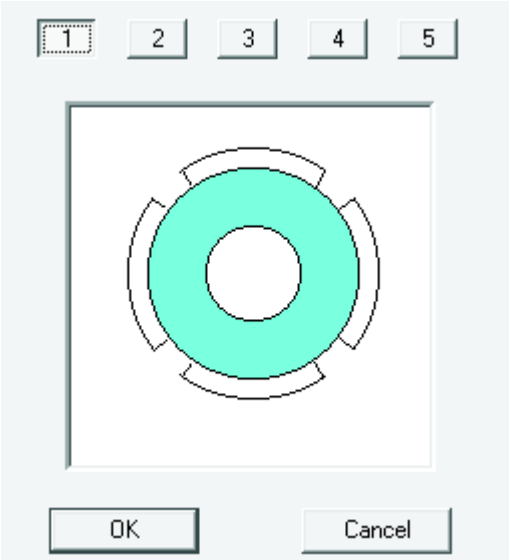


Figura 2.10 Ventana de selección de la forma de los polos del rotor

La configuración de los parámetros del polo seleccionado se realizará en el submenú denominado *Pole.*

2.1.2.5.1 Pole

La ventana mostrada en la figura 2.11 permite configurar las dimensiones del polo seleccionado.

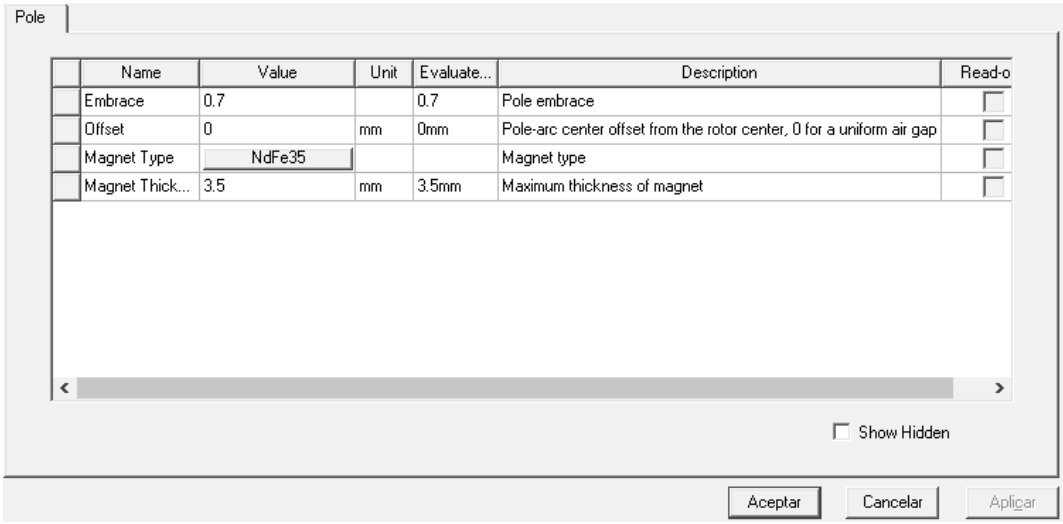


Figura 2.11 Ventana de configuración de los polos del rotor

Para la selección del material magnético que se empleará en la simulación de los polos, se utiliza la librería *Materials* de RMxprt. El material magnético que se encuentra seleccionado es NdFe35 como se verifica en la figura 2.12.

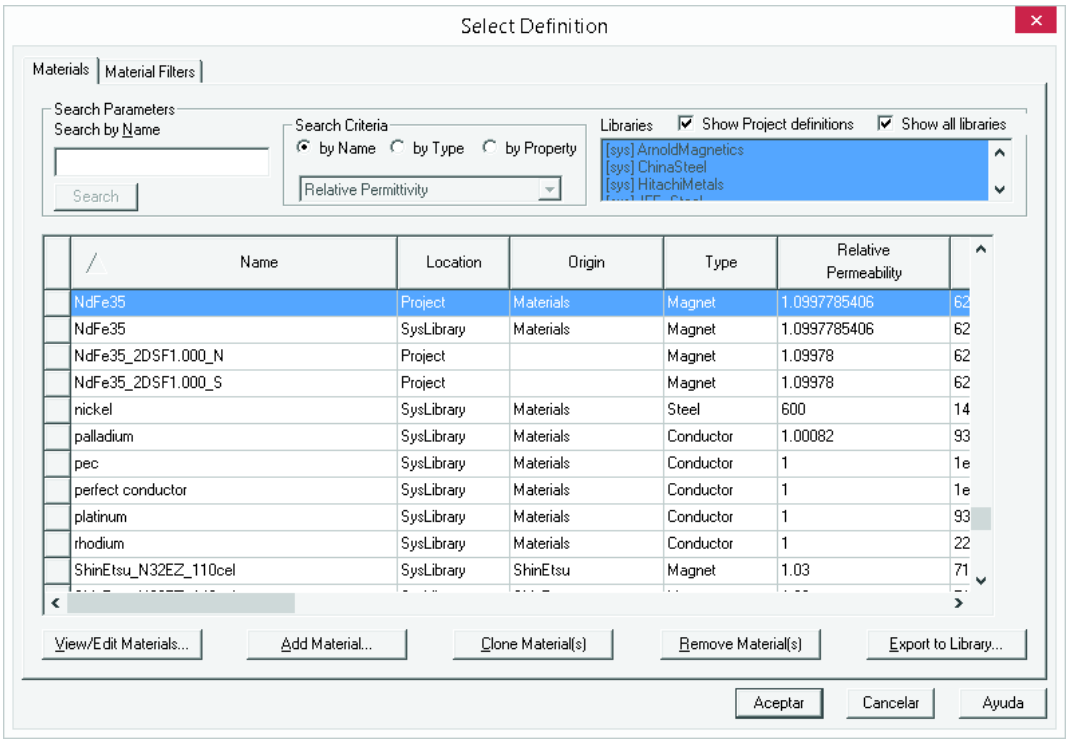


Figura 2.12 Ventana de selección del material magnético

2.1.2.6 Ventana de configuración del análisis

La ventana de configuración de análisis mostrada en la figura 2.13, permite definir los parámetros de entrada/salida nominal del motor con los cuales se realizará el análisis.

El campo *Operation Type,* permite definir el funcionamiento de la máquina como motor o como generador.

El campo *Rated Output Power*, describe la potencia nominal desarrollada en el eje del motor.

El campo *Rated Voltage*, representa el voltaje RMS línea-línea nominal.

El campo *Rated Speed* define la velocidad nominal del motor a la cual se realiza el análisis.

El campo *Operating Temperature* determina la temperatura de operación del motor.

General | ASSM

	Name	Value	Unit	Evaluated V...	Description	Read-o
	Name	Setup1				☐
	Enabled	☑				☐
	Operation Type	Motor			Motor or generator	☐
	Load Type	Const Power			Mechanical load type	☑
	Rated Output ...	743	W	743W	Rated mechanical or electrical output power	☐
	Rated Voltage	220	V	220V	Applied or output rated line-to-line AC (RMS) or DC voltage	☐
	Rated Speed	1800	rpm	1800rpm	Given rated speed	☐
	Operating Tem...	75	cel	75cel	Operating temperature	☐

☐ Show Hidden

Aceptar | Cancelar | Aplicar

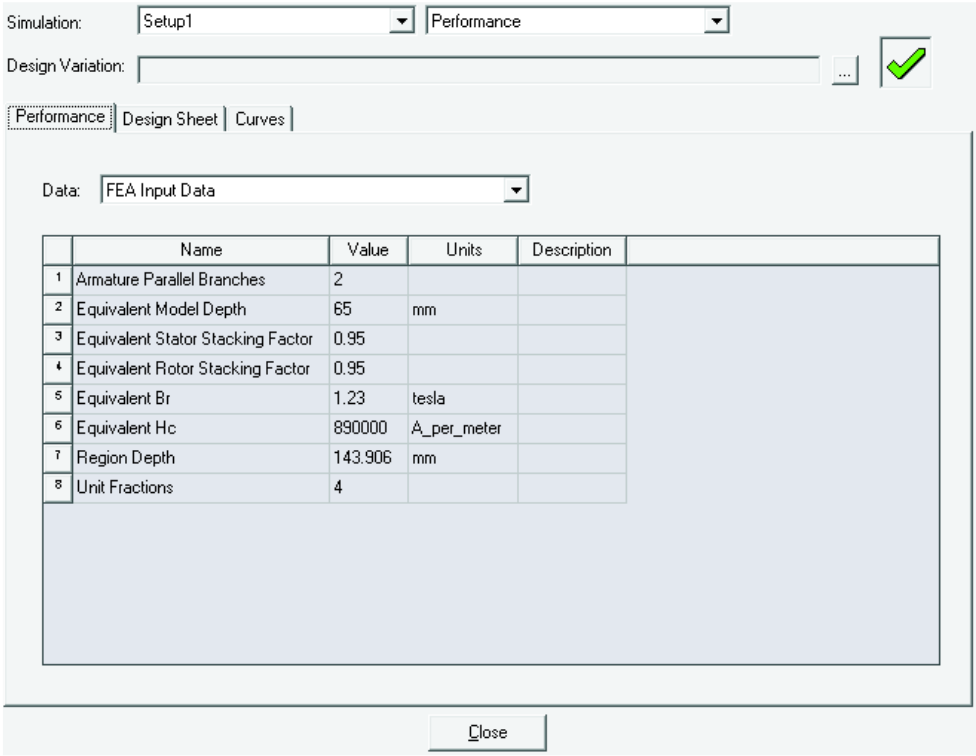
Figura 2.13 Ventana de configuración del análisis

El tipo de operación seleccionado para la simulación de la máquina en este caso es *Motor*. La potencia nominal de salida es 743 W, el voltaje nominal aplicado es 220 V y velocidad nominal es 1800rpm.

2.1.2.7 Ventana de visualización de resultados

Para visualizar los resultados del motor diseñado, se inicia con el análisis mediante la ejecución del comando *Analize all* que se encuentra en el menú de la ventana de RMxprt. Los resultados obtenidos se visualizan en la ventana *Solutions,* la cual está dividida en tres pestañas: *Performance, Design Sheet* y *Curves* que se explican a continuación:

Perfomance, contiene un menú desplegable que permite visualizar varias tablas con los valores detallados del modelo de motor diseñado. La figura 2.14 permite visualizar la ventana *Perfomance*.



	Name	Value	Units	Description
1	Armature Parallel Branches	2		
2	Equivalent Model Depth	65	mm	
3	Equivalent Stator Stacking Factor	0.95		
4	Equivalent Rotor Stacking Factor	0.95		
5	Equivalent Br	1.23	tesla	
6	Equivalent Hc	890000	A_per_meter	
7	Region Depth	143.906	mm	
8	Unit Fractions	4		

Figura 2.14 Ventana del análisis, Perfomance

Desing Sheet, contiene la información completa de todo el diseño de la máquina realizado. La figura 2.15 muestra la ventana *Design Sheet.*

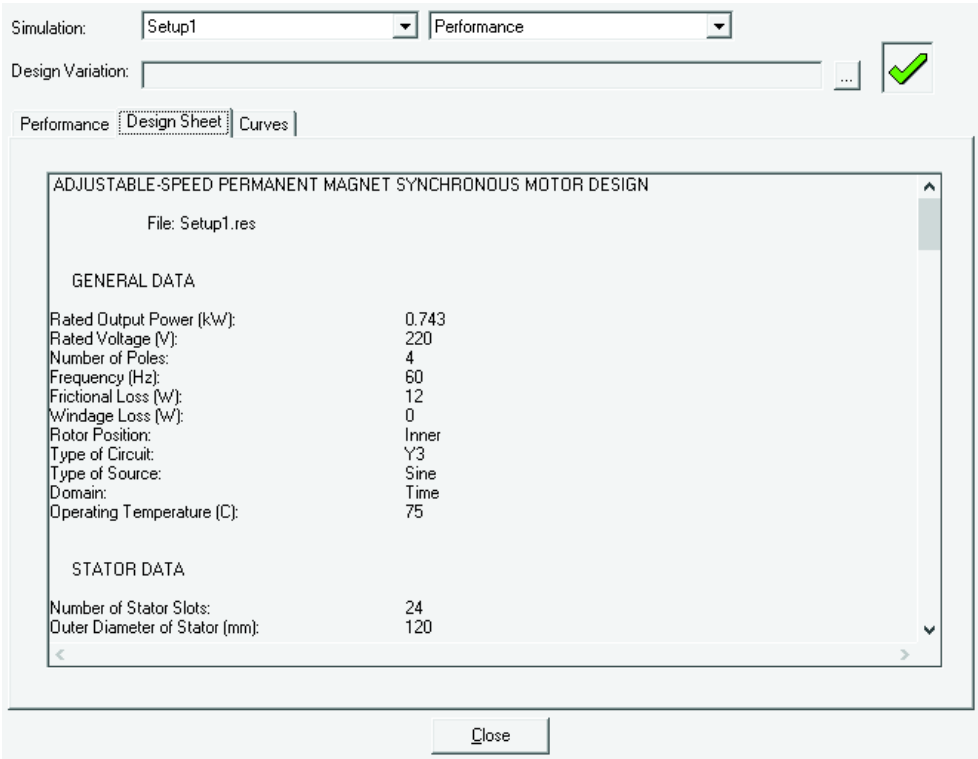


Figura 2.15 Ventana del análisis, Desing Sheet

Curves, permite desplegar gráficas de varios parámetros dependiendo del tipo de máquina. El menú *Curves* cuenta con una barra que permite desplegar y seleccionar la gráfica que se desea como se visualiza en la figura 2.16.

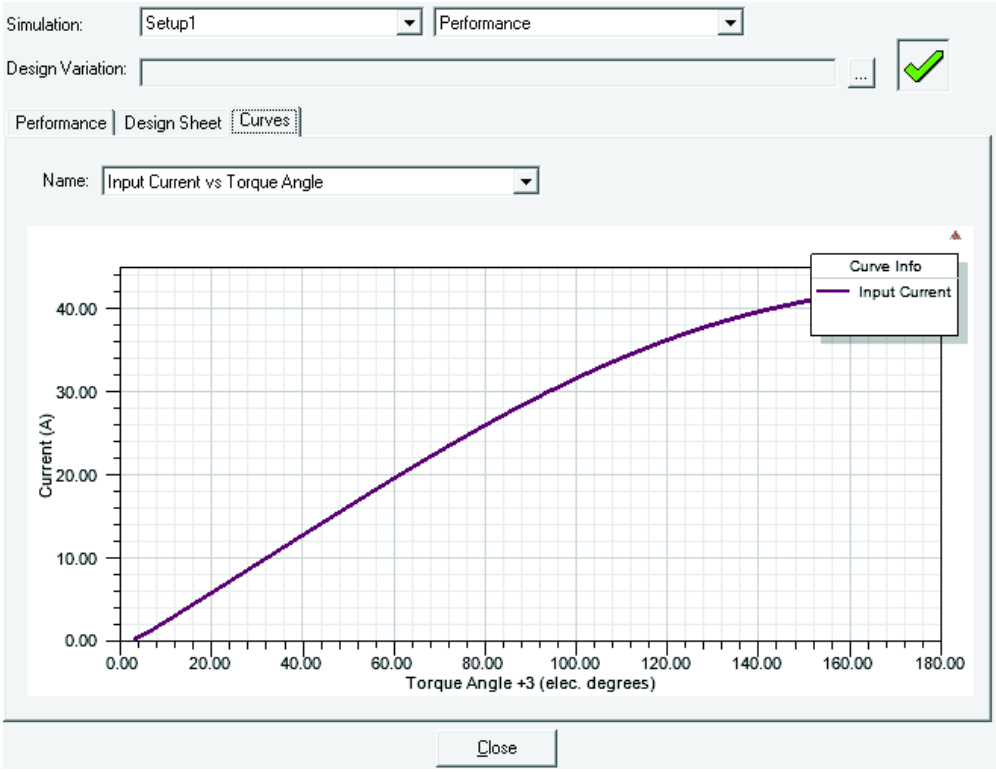


Figura 2.16 Ventana del análisis, Curves

2.1.3 Desarrollo del modelo de la PMSM utilizando Maxwell 2D

El software ANSYS Maxwell proporciona diversos métodos y herramientas para crear la geometría de los diferentes tipos de máquinas AC o DC. Una vez creada la geometría se configura cada uno de los parámetros del modelo para aplicar el método de análisis por elementos finitos (F.E.A.).

La creación de la geometría de las máquinas se las realiza mediante el uso de las utilidades propias del software ANSYS Maxwell o mediante la importación de archivos de fuentes externas en donde se posea la geometría deseada como el caso de ANSYS RMxprt que trabaja con plantillas.

El método elegido para el desarrollo de la geometría es el uso de ANSYS RMxprt, el cual incluye una plantilla del motor síncrono de imanes permanentes como se muestra en la Fig. 2.17.

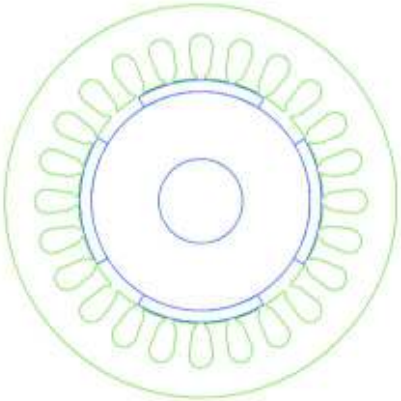


Figura 2.17 Plantilla de un motor síncrono de imanes permanentes en RMxprt

Una vez que la geometría de la máquina ha sido creada, modificando la plantilla de RMxprt, se puede configurar un nuevo proyecto completo de Maxwell 2D.

Para crear el nuevo proyecto de Maxwell, se ejecuta el comando *Create Maxwell Desing* y se escoge la solución configurada, en este caso Setup1 como se visualiza en la figura 2.18.

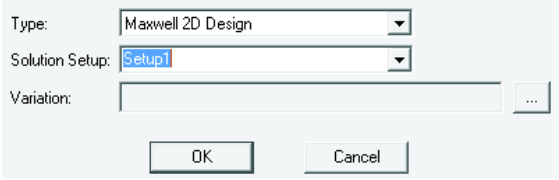


Figura 2.18 Comando de creación de un Proyecto en Maxwell desde RMxprt

Una vez que se posea el modelo de la máquina en Maxwell 2D se debe verificar cada uno de los componentes importados, y se procede con un pre-análisis para validar el modelo creado como se visualiza en la figura 2.19.

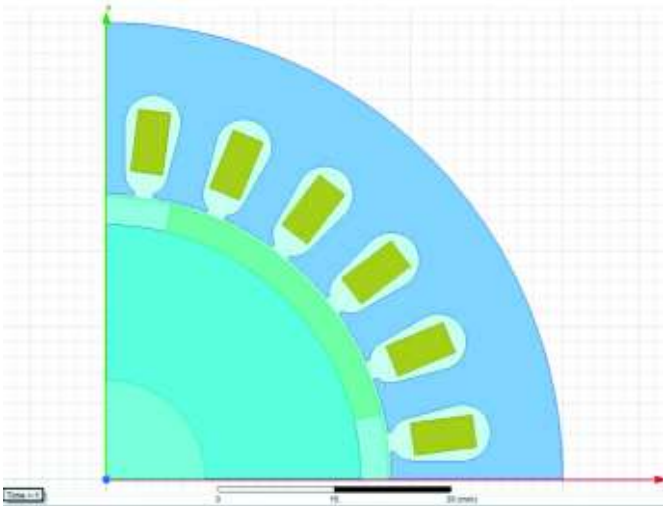


Figura 2.19 Modelo del motor de imanes permanentes creado en Maxwell 2D

2.1.3.1 Análisis del modelo creado en Maxwell 2D mediante elementos finitos.

2.1.3.1.1 Mesh plot

El mallado del modelo 2D de la PMSM se realiza de manera automática mediante la ejecución de los comandos: *Edit-Select All, View-Set solution context, Maxwell2D-Fields-Plot Mesh-Done*. La figura 2.20 muestra la configuración del mallado realizado.

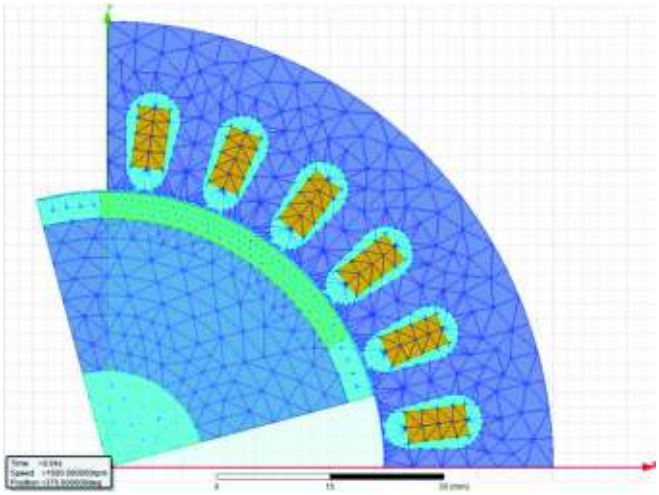


Figura 2.20 Mallado del modelo de una PMSM en Maxwell 2D

La configuración del tamaño del mallado se ejecuta de manera automática dependiendo del tamaño de las superficies, pero también se la puede configurar acorde a las necesidades del diseñador para lo cual se ejecuta los pasos a continuación:

En el menú de *Project Manager*, desplegar el submenú *Mesh Operations*. Una vez realizada esta acción se escoge la superficie que se desea modificar. Para desplegar se procede a dar doble click. El campo *Maximun Length of Elements* es el que permite configurar el tamaño del mallado de cada elemento como se visualiza en la figura 2.21.

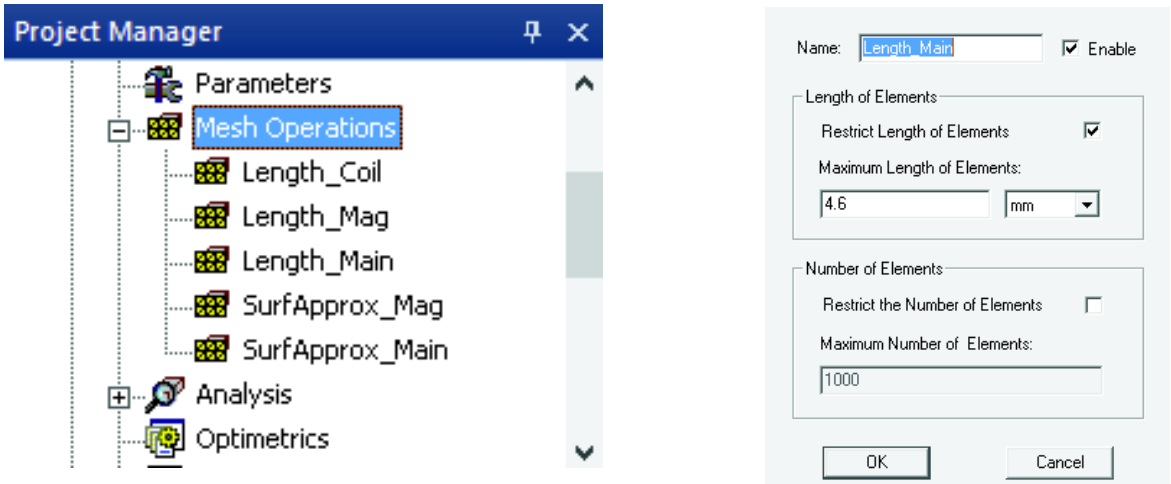


Figura 2.21 Configuración manual del mallado del modelo de una PMSM en Maxwell 2D

Una vez que se ha terminado de configurar el mallado, el análisis se ejecuta mediante los comandos: *Maxwell2D-Analyze All.*

2.1.3.1.2 Curvas de resultados en Maxwell 2D

Una vez ejecutado el análisis del modelo de la PMSM en Maxwell, se puede verificar el comportamiento de cada una de las variables eléctricas involucradas. Las gráficas obtenidas del análisis se presentan de manera interactiva como se visualiza en la figura 2.22.

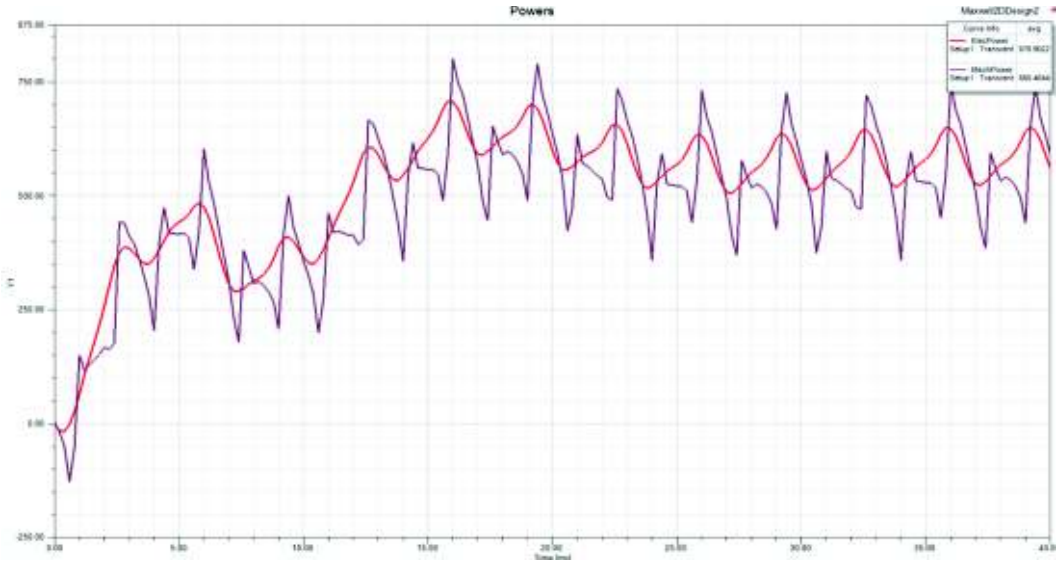


Figura 2.22 Gráficas obtenidas de la simulación en Maxwell 2D

2.1.3.2 Diagrama de distribución de variables magnéticas.

El software Ansys Maxwell 2D permite visualizar el comportamiento y distribución de las variables magnéticas que se crean al simular el modelo de la PMSM. Para empezar, se procede a ejecutar el comando *Set Solution Context* del menú *View* con lo cual se presenta la ventana mostrada en la figura 2.23.

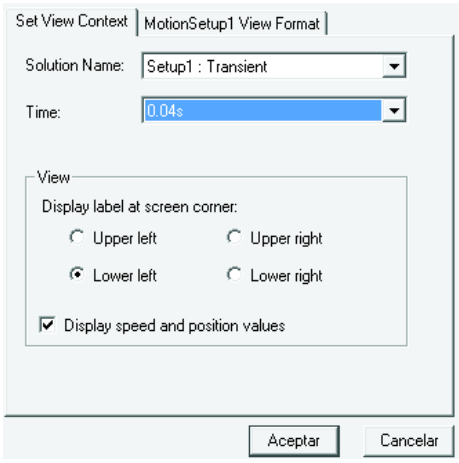


Figura 2.23 Ventana Set Solution Context

La ventana *Set Solution Context* mostrada en la figura 2.23 permite definir el tiempo de inicio para visualizar el comportamiento de las variables magnéticas.

Una vez definido el parámetro anterior, se procede a ejecutar el comando *Select All* del menú *Edit* el cual permite seleccionar todos los componentes involucrados en el modelo. Una vez que se han seleccionado todos los componentes del modelo 2D, se procede con la ejecución del comando *Fields-Fields-A-Flux_Lines* del menú *Maxwell 2D.* Los resultados obtenidos se muestran en base a una escala de valores y colores como se visualiza en la figura 2.24.

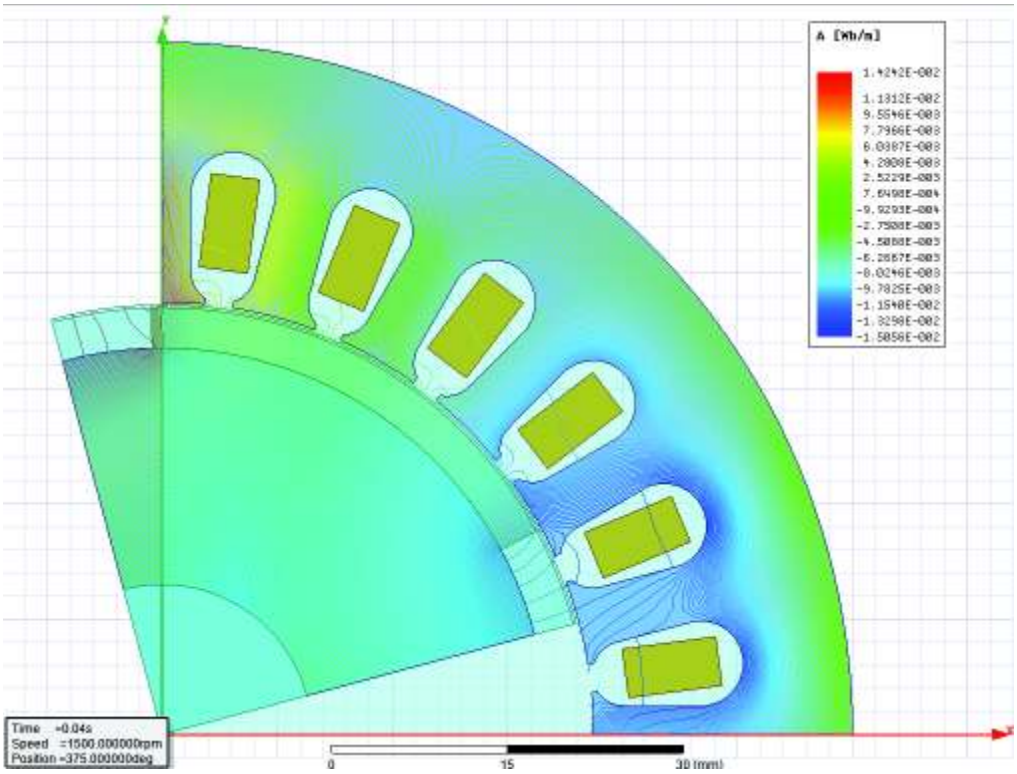


Figura 2.24 Líneas de flujo magnético del modelo 2D (Wb/m)

Una vez realizadas las simulaciones que permiten verificar el comportamiento del modelo implementado, el siguiente paso consiste en implementar el circuito de potencia para el accionamiento del modelo de la PMSM mediante Simplorer.

2.1.4 Implementación del circuito eléctrico de potencia en Ansys Simplorer.

Ansys Simplorer es una herramienta de software que permite diseñar y configurar circuitos eléctricos de potencia y control para el accionamiento de los modelos de máquinas creados tanto en RMxprt como en Maxwell 2D.

El circuito de potencia implementado se muestra en la figura 2.25, el cual consta de seis IGBTs, seis diodos y dos fuentes de voltaje DC.

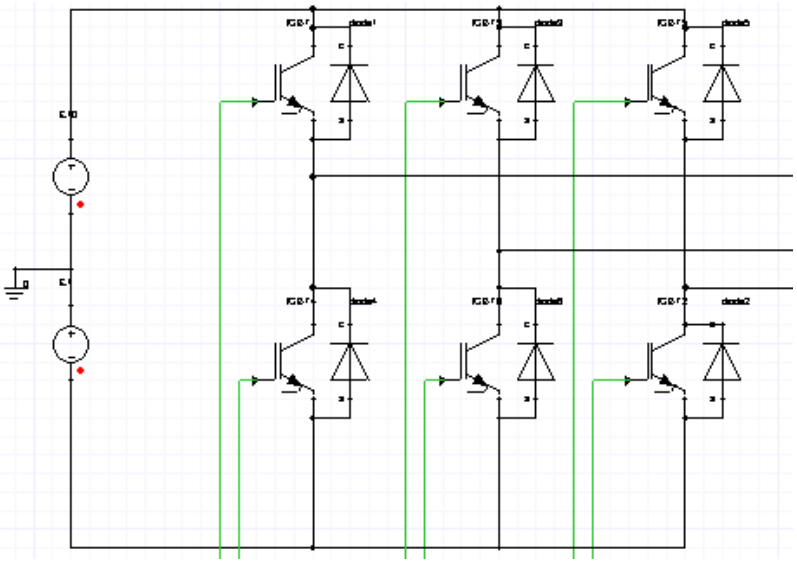


Figura 2.25 Inversor trifásico implementado en Simplorer

Simplorer cuenta con una biblioteca completa de dispositivos eléctricos y electrónicos, así como elementos de medición de variables eléctricas. Los componentes incluidos se despliegan en la lista mostrada en la figura 2.26.

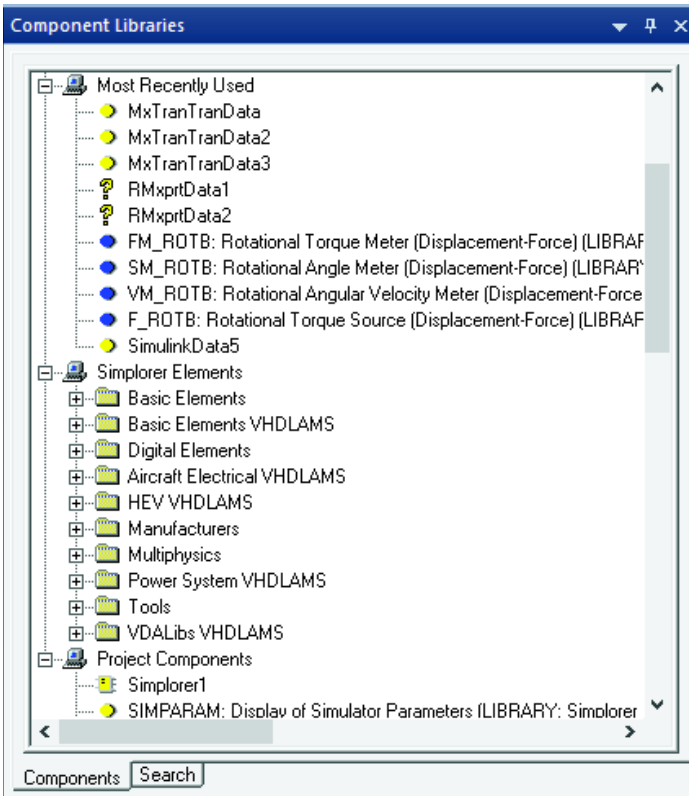


Figura 2.26 Bibliotecas de dispositivos eléctricos y electrónicos incluidos en Simplorer

Las compuertas de los IGBTs son activadas mediante señales de control entregadas desde Matlab a través del acoplamiento entre el software Simplorer-Matlab que se explica más adelante. Para la medición de variables eléctricas se utiliza los instrumentos virtuales propios de Simplorer mostrados en la figura 2.27. Las señales adquiridas por los instrumentos virtuales serán enviadas hacia el controlador diseñado en Matlab.

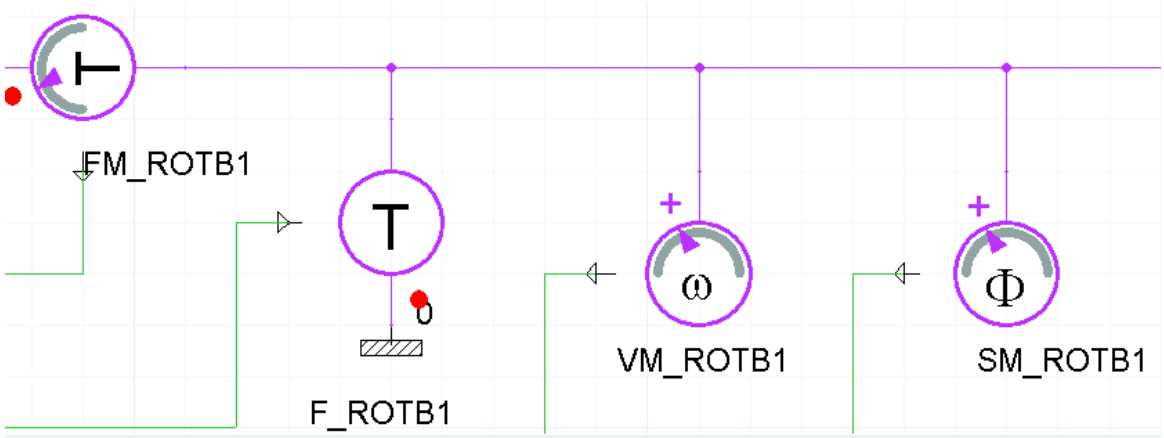


Figura 2.27 Instrumentos virtuales incluidos en Simplorer

Para extraer los datos de las variables adquiridas mediante los instrumentos virtuales, se despliega la ventana de configuración de cada instrumento y se selecciona *Show Pin* en la pestaña Output/Display como se visualiza en la figura 2.28. Realizada esta acción, se procede a cablear cada una de las señales con el componente de acoplamiento con Simulink.

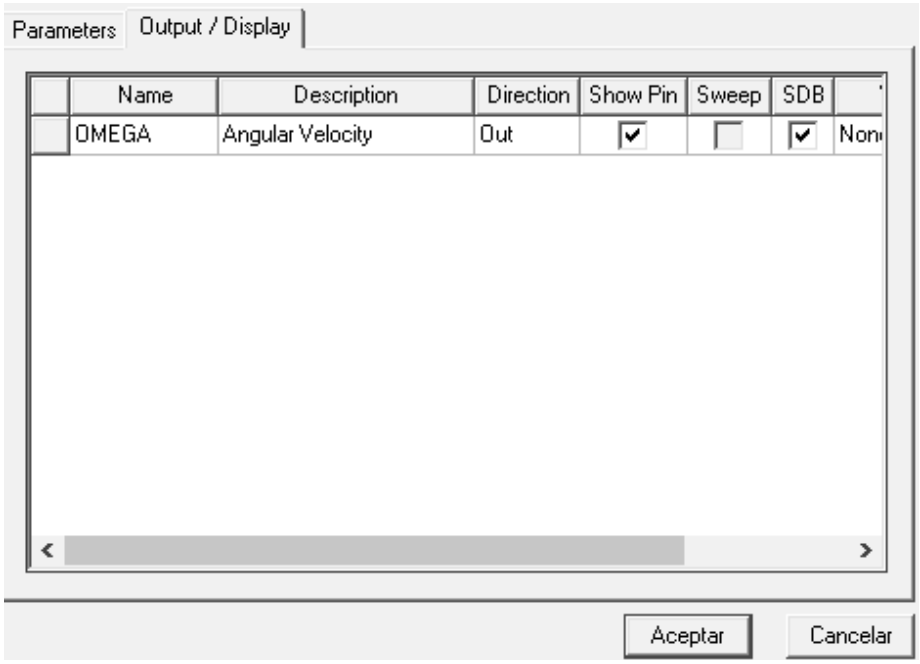


Figura 2.28 Ventana de configuración de instrumentos virtuales incluidos en Simplorer

2.1.5 Co-simulación entre los softwares de Ansys con Simplorer

2.1.5.1 Co-simulación RMxprt-Simplorer

Para realizar la co-simulación entre los softwares Ansys RMxprt y Simplorer se utiliza el enlace denominado *RMxprt dynamic coupling*, el cual permite ejecutar ambos programas de manera interactiva. Para activar esta función, en el menú de *Simplorer Circuit*, seleccionar *SubCircuit* y *Add RMxprt Dynamic component*, con lo cual se despliega la ventana mostrada en la figura 2.29.

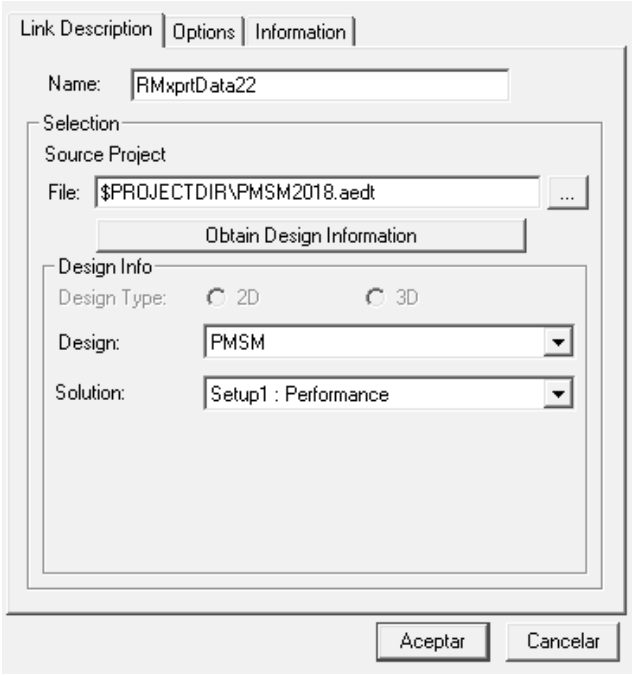


Figura 2.29 Ventana de enlace RMxprt dynamic coupling

La ventana mostrada en la figura 2.29 permite seleccionar el diseño y la solución que fueron configurados utilizando la herramienta RMxprt Design. Realizado el paso anterior, se procede a colocar el modelo dinámico del motor en el diagrama esquemático de Simplorer. El bloque se visualiza nombrado como RMx1 como se muestra en la figura 2.30.

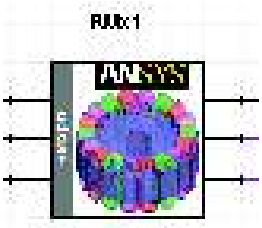


Figura 2.30 Modelo RMx1 añadido al diagrama esquemático en Simplorer

Finalmente se procede a realizar las conexiones del modelo RMx1 con el resto de elementos eléctricos configurados en Simplorer.

2.1.5.2 Co-simulación Maxwell 2D-Simplorer

Para realizar la co-simulación entre los softwares Maxwell 2D y Simplorer se utiliza el enlace denominado *Maxwell Transient-Transient coupling* que viene incluido en Simplorer, y que permite ejecutar ambos programas de manera interactiva.

Para implementar la co-simulación, el primer paso consiste en configurar el tipo de alimentación del motor como *Externa* como se visualiza en la figura 2.31. La configuración de este parámetro se realiza dentro del submenú *Excitations-Phase A, Phase B y Phase C* del modelo creado en Maxwell 2D.

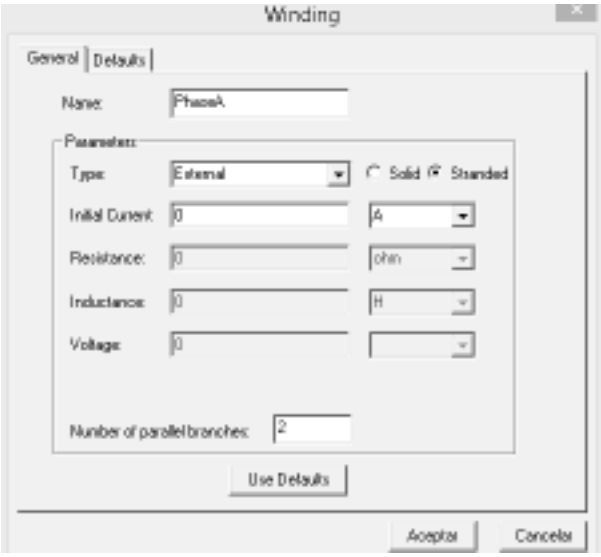


Figura 2.31 Selección de alimentación del motor tipo externa

Como segundo paso se deberá seleccionar el campo *Enable transient-transient link with Simplorer* en las configuraciones de diseño de Maxwell 2D, con el objetivo de habilitar la co-simulación entre los dos softwares tal como se visualiza en la figura 2.32.

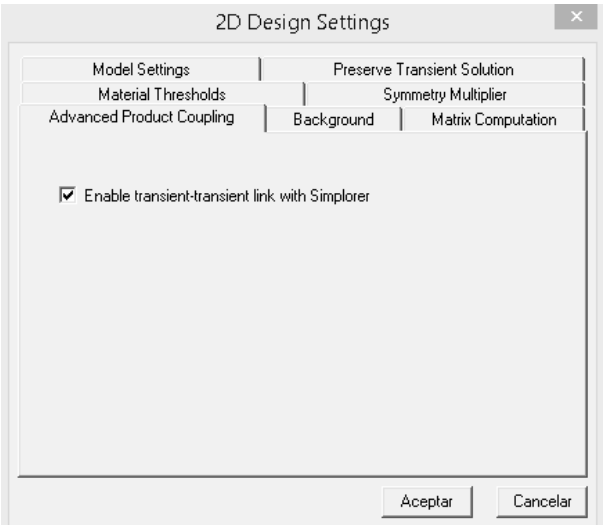


Figura 2.32 Habilitación para co-simulación entre Maxwell 2D y Simplorer

Una vez terminadas las configuraciones para la co-simulación mediante Maxwell 2D, se procede a configurar utilizando Simplorer. Para establecer el link de co-simulación, en la ventana de Simplorer de debe seleccionar el menú *Simplorer Circuit-SubCircuit-Maxwell Component-Add Transient Cosimulation* con lo cual se despliega la ventana de la figura 2.33. La ventana de la figura 2.33 permite seleccionar el modelo y la solución implementados en Maxwell 2D que serán utilizados en la co-simulación.

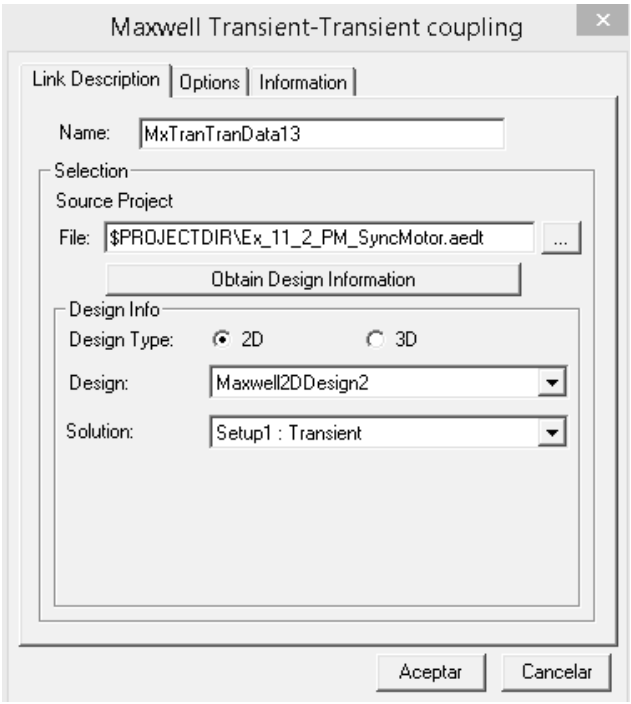


Figura 2.33 Ventana para acoplamiento Maxwell 2D y Simplorer

Realizado los pasos anteriores, se procede a colocar el modelo FEA de la PMSM en el diagrama esquemático de Simplorer. El bloque se visualiza nombrado como FEA1 como se muestra en la figura 2.34.

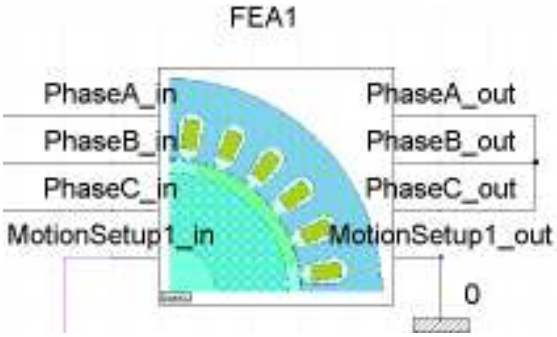


Figura 2.34 Modelo FEA1 añadido al diagrama esquemático en Simplorer

Finalmente se procede a realizar las conexiones del bloque FEA1 con el resto de elementos eléctricos implementados en Simplorer.

2.1.6 Co-simulación Simplorer-Matlab

La configuración del enlace entre Simplorer y Matlab tiene como objetivo el intercambio de información entre los dos softwares mencionados. Por un lado, Simplorer envía a Matlab la información adquirida de las señales eléctricas de los sensores y por otro lado Matlab envía hacia Simplorer la información de las señales que controlan las compuertas de los seis IGBTs del inversor trifásico.

Para realizar la configuración de la co-simulación en Simplorer, se procede a seleccionar el comando *Subcircuit-Add-Simulink Component* con lo cual se despliega la ventana en donde se añaden las variables a ser utilizadas para el envío y recepción de los datos de Matlab/Simulink.

Los campos se configuran utilizando nombres para las variables y el sentido de dirección del flujo de datos como se muestra en la ventana de la figura 2.35.

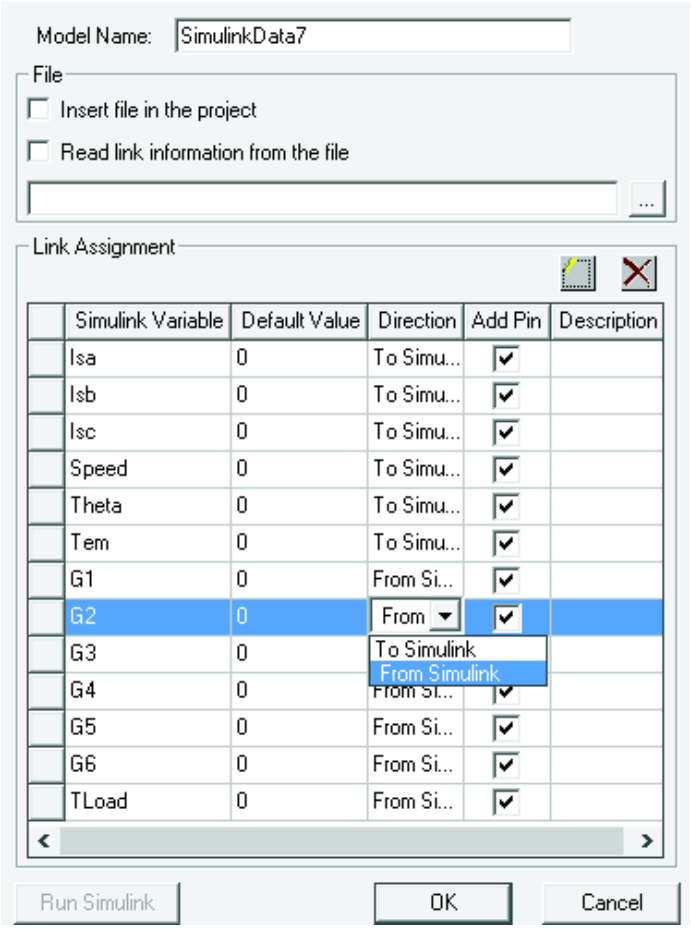


Figura 2.35 Ventana de configuración de variables para co-simulación Simplorer-Matlab

Una vez realizada la configuración, se procede a colocar el icono de conexión de la interface entre Simplorer y Matlab el cual cuenta con pines membretados de acuerdo a la configuración realizada como se visualiza en la figura 2.36.

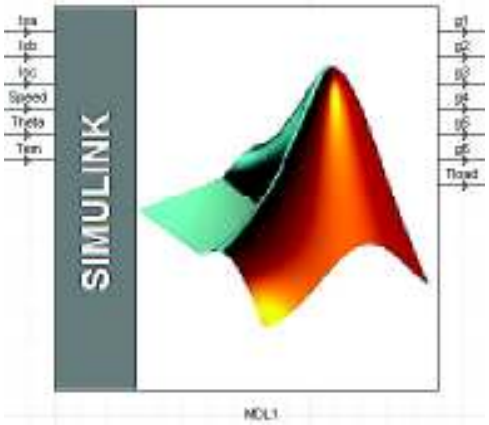


Figura 2.36 Icono del enlace Simplorer-Matlab con nombres de variables y sentido de flujo de datos

Los pines del icono nombrado como MDL1 se conectan con cada uno de los elementos eléctricos involucrados en el intercambio de información entre Simplorer y Matlab/Simulink. El siguiente paso consiste en realizar la configuración del enlace para la co-simulación desde Matlab/Simulink. Para llevar a cabo esta tarea se utiliza el bloque denominado *S-Function* que se encuentra incluido en las librerías de Simulink.

Se procede a colocar el bloque *S-Function* en el proyecto de Matlab y al dar doble clic se debe reemplazar el nombre de la función por 'AnsoftSFunction' como se visualiza en la figura 2.37. Clic en OK y la ventana de asignación de enlace de las variables de Simplorer aparece.

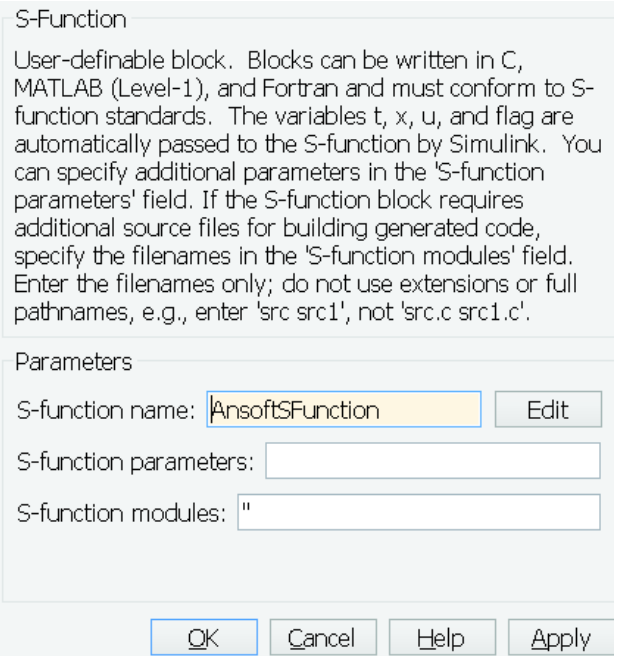


Figura 2.37 Edición del bloque S-Function

Para finalizar el acoplamiento entre Simplorer y Simulink se procede con la asignación de las variables eléctricas tanto de entrada como de salida como se visualiza en la ventana de la figura 2.38. El botón denominado *Link* permite enlazar cada una de las variables ingresadas.

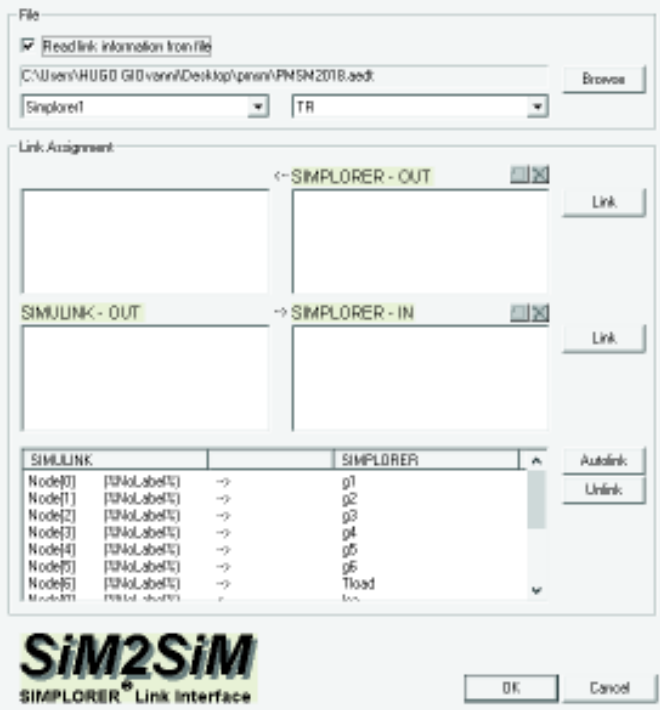


Figura 2.38 Asignación de variables de entrada-salida

Una vez asignadas las variables, se procede a ubicar el bloque *S-Function* en el diagrama de Simulink. Es importante mencionar que el orden de entrada y salida de variables es el configurado en el icono de enlace añadido a Simplorer. *AnsoftSFunction* no muestra los nombres de las variables como se visualiza en la figura 2.39.

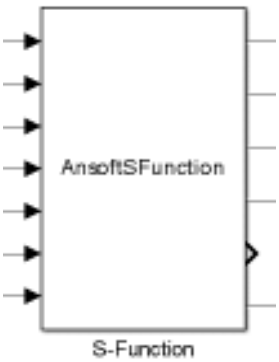


Figura 2.39 Icono del enlace Matlab-Simplorer con sentido de flujo de datos

Finalmente se procede a realizar las conexiones del bloque *AnsoftSFunction* con el resto de los bloques implementados en Maltab/Simulink.

2.1.7 Co-simulación RMxprt-Simplorer-Matlab (Without Finite Elements Analysis)

Para realizar la co-simulación entre los softwares RMxprt, Simplorer y Matlab se deberá establecer el tiempo de muestreo y tiempo total de simulación de cada uno. En Simulink, el tiempo de muestreo se establece en continuo y el tiempo total de la simulación es 0.5s. Para configurar el tiempo de muestreo se utiliza el bloque *powergui* que se encuentra dentro de las librerías de Matlab/Simulink.

Por otro lado, Simplorer permite configurar el tiempo de muestreo y el tiempo total de simulación utilizando la ventana *Transient Analysis Setup* como se visualiza en la figura 2.40.

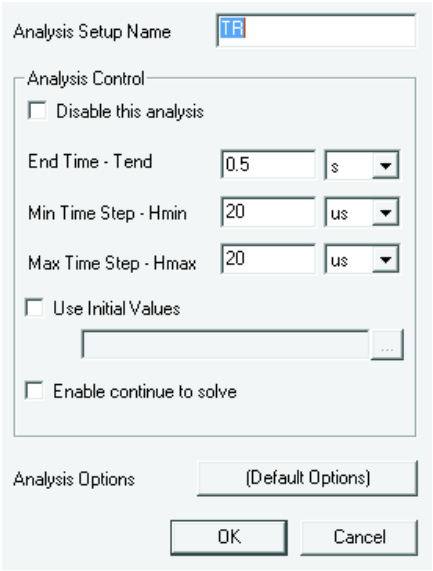


Figura 2.40 Ventana de configuración de tiempos de simulación en Simplorer

Los parámetros de la figura 2.40 que deberán ser configurados son:

Tiempo total de simulación (End Time-T end) que para el caso de este proyecto es igual a 0.5s. El paso de tiempo mínimo (Hmin) y paso de tiempo máximo (Hmax) son configurados en 20us. Los valores establecidos en Simplorer configurarán de manera automática el enlace con RMxprt.

Una vez establecidos los tiempos de simulación de Simplorer y Simulink, la co-simulación es inicializada desde el software Matlab-Simulink utilizando el comando *RUN*. La co-simulación RMxprt-Simplorer-Matlab comienza a ejecutarse visualizando barras de incremento sincronizadas en cada uno de los softwares tal como se muestra en la figura 2.41.

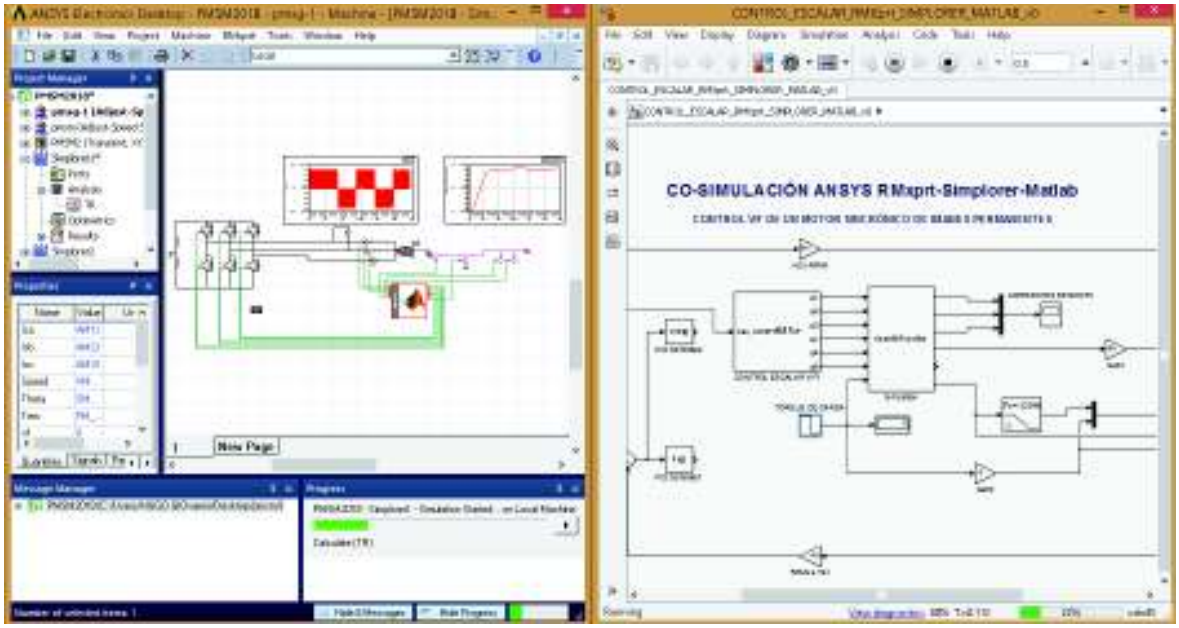


Figura 2.41 Desarrollo de la co-simulación RMxprt/Simplorer-Matlab

2.1.8 Co-simulación Maxwell 2D-Simplorer-Matlab (With Finite Element Analysis)

Para realizar la co-simulación entre Maxwell 2D-Simplorer-Matlab se deberá establecer el tiempo de muestreo y tiempo total de simulación de cada uno de los tres softwares. La configuración de los tiempos de simulación en Maxwell 2D se realiza en el menú de *Maxwell2DDesign1*, submenú *Analysis*. Se procede a establecer los valores de los campos Stop time y Time step como se visualiza en la figura 2.42.

Figura 2.42 Ventana de configuración de tiempos de simulación en Maxwell 2D

Los parámetros de la figura 2.42 que deberán ser configurados son:

Tiempo total de simulación (Stop Time) que para el caso de este proyecto es igual a 0.5s. El paso de tiempo es configurado en 10us.

En el caso de Simplorer, el Tiempo total de simulación (End Time-T end) se establece en 0.5s. El paso de tiempo mínimo (Hmin) y paso de tiempo máximo (Hmax) son configurados en 20us.

Finalmente, para Matlab el tiempo de muestreo se establece en continuo y el tiempo total de simulación será igual a 0.5s.

Una vez establecidos los tiempos de simulación de Maxwell 2D, Simplorer y Simulink, la co-simulación es inicializada desde el software Matlab-Simulink utilizando el comando *RUN*. La co-simulación Maxwell 2D-Simplorer-Matlab comienza a ejecutarse visualizando barras de incremento sincronizadas en cada uno de los softwares tal como se muestra en la figura 2.43.

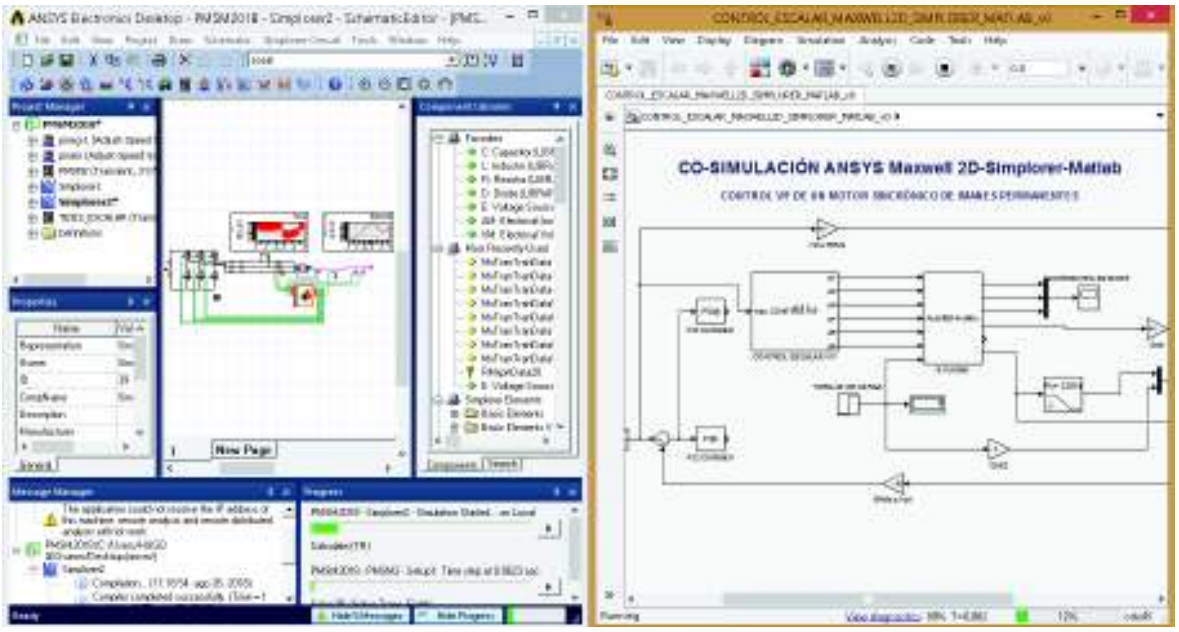


Figura 2.43 Desarrollo de la co-simulación Maxwell 2D-Simplorer-Matlab

2.2 Diseño de controladores

2.2.1 Control escalar V/F

El método elegido para el control de velocidad y torque del motor sincrónico de imanes permanentes es el control escalar V/F (Voltios/Hercios). La figura 2.44 muestra el esquema del control escalar V/F interconectado con un controlador Proporcional-Integral para mejorar la respuesta en estado transitorio y estable.

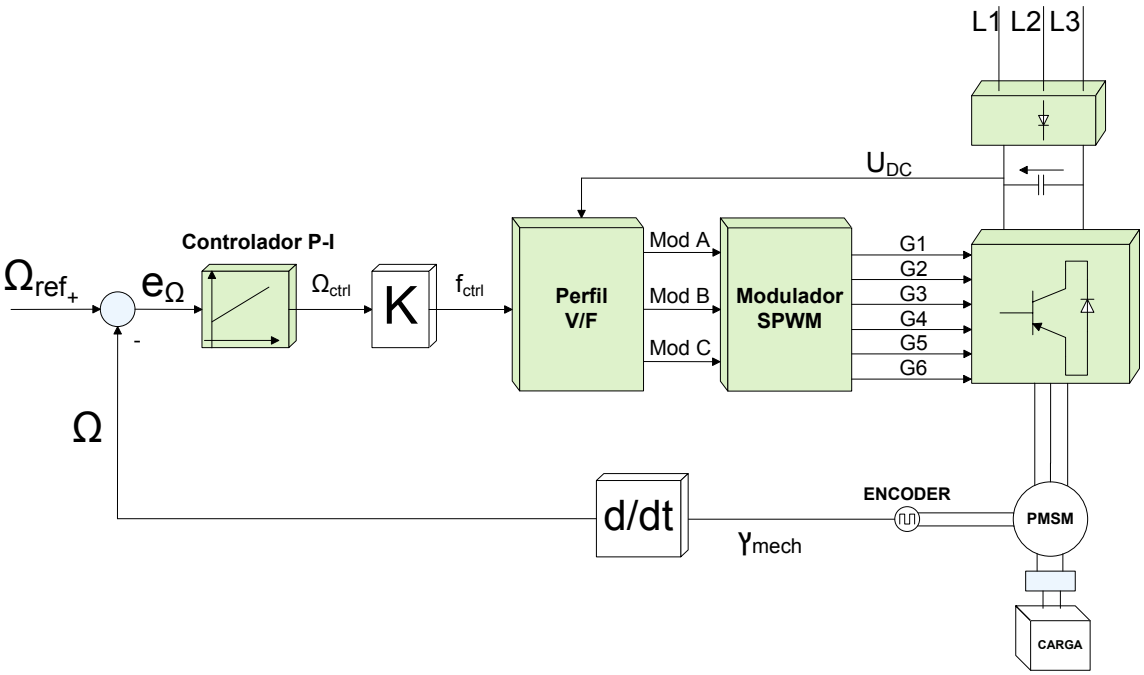


Figura 2.44 Esquema del control escalar V/F

El esquema de la figura 2.44 permite apreciar el control V/F aplicado en donde se compara la velocidad de referencia con la velocidad real del motor. El error obtenido se introduce en un regulador PI (proporcional-integral) del cual se obtiene la salida de control que será transformada a un señal de frecuencia que ingresa al perfil V/F. El bloque denominado perfil V/F mantiene constante la relación entre el voltage y la frecuencia de alimentación del estator y genera tres ondas modulantes desfasadas 120 grados las cuales ingresan al bloque modulador SPWM para generar las señales de control de los seis IGBTs.

2.2.2 Control de velocidad

El control de velocidad se lo implementa utilizando un controlador Proporcional-Integral como el que se muestra en la figura 2.45.

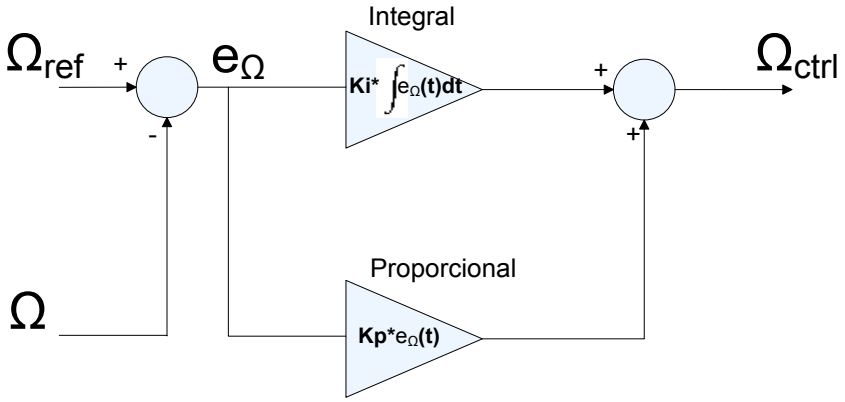


Figura 2.45 Esquema del controlador Proporcional-Integral

En donde:

Ω_{ref} es la referencia de velocidad.

Ω es la velocidad medida del rotor.

e_{Ω} es el error de velocidad calculado.

Ω_{ctrl} es la señal de control.

Kp es la ganancia proporcional.

Ki es la ganancia integral.

2.2.3 Cálculo de las contantes del controlador proporcional-integral

Cuando es posible determinar el modelo matemático de una planta se puede aplicar técnicas de diseño para determinar los valores de las contantes del controlador PI que cumplan las especificaciones tanto en estado transitorio como en estado estable del sistema en lazo cerrado. Sin embargo, cuando no es posible obtener el modelo matemático de la planta se debe recurrir a técnicas experimentales para poder identificarla.

La técnica empleada para la sintonización de controlador PI de velocidad es mediante Ziegler-Nicholls la cual establece un conjunto de reglas para obtener los valores de las contantes K_P y K_i las cuales se detallan a continuación:

Para sistemas con carácter oscilante, se pone el regulador en bucle cerrado utilizando únicamente ganancia proporcional.

Se establece en cero el valor de la constante K_i.

Partiendo de un valor bajo de ganancia proporcional K_P, se incrementa la ganancia hasta que el sistema presente una oscilación sostenida. Este valor de ganancia se denomina K_{CR}.

Una vez obtenida la ganancia crítica se procede a medir el periodo de la oscilación al cual se denomina T_{osc}.

Una vez determinado K_{CR} y T_{OSC} se aplica las oscilaciones de la tabla 2.2 para determinar las ganancias del controlador PI.

Tabla 2.2 Regla de Ziegler-Nicholls

CONTROLADOR	KP	Ki=1/Ti	Kd=Td
P	$0.5*K_{CR}$	0	0
PI	$0.45*K_{CR}$	$1.2/ T_{osc}$	0
PID	$0.6*K_{CR}$	$2/ T_{osc}$	T_{osc} /8

La figura 2.46 permite visualizar la oscilación sostenida obtenida al establecer una ganancia de K_P=7.815 y K_i=0

Figura 2.46 Oscilación sostenida obtenida con K_P=7.815 y K_i=0.

A partir de la figura 2.46 se puede determinar que:

- $K_{CR} = 7.815$
- $T_{OSC} = 0.0916 - 0.0557 = 0.0359$

Para determinar los valores de la ganancia proporcional e integral del controlador se emplea las fórmulas de la tabla 2.2.

$$K_P = 0.45 * 7.815 = 3.51675$$

$$K_i = \frac{1.2}{T_{OSC}} = \frac{1.2}{0.0359} = 33.426$$

Por lo tanto, los parámetros del controlador calculados mediante el método de Ziegler-Nicholls que permiten reducir el error en estado estacionario, reduciendo el tiempo de subida y disminuyendo el sobre pico del sistema son:

Ganancia Proporcional K_P = 3.51675

Ganancia Integral K_i = 33.426

Las ganancias calculadas son establecidas en el bloque *PI Controller* de Simulink en la pestaña *Main*, *Controller parameters, Proportional (P)=3.51675* e *Integral (I)=33.426* como se visualiza en la figura 2.47.

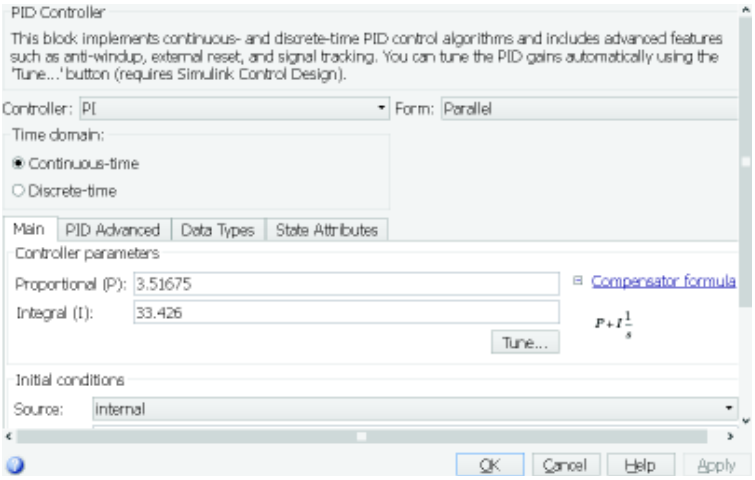


Figura 2.47 Configuración de ganancias en bloque PID Controller de Simulink

2.2.4 Variables de conexión Matlab-Simplorer

El intercambio de datos entre Matlab Simulink y Simplorer se realiza a través de la función "AnsoftSFunction". La función tiene como variables de entrada-salida las detalladas en la tabla 2.3.

Tabla 2.3 Variables de interacción Simplorer-Matlab.

Tipo de variable	
De Matlab a Simplorer	**De Simplorer a Matlab**
G1	Ia
G2	Ib
G3	Ic
G4	Speed
G5	Theta
G6	Tem
T_load	

Los datos de las variables que envía Simplorer a Simulink sirven como entradas para el funcionamiento del controlador diseñado, y las variables enviadas de Simulink a Simplorer permiten realizar la activación y desactivación de las compuertas de los IGBTs que controlan el modelo del motor en Simplorer.

Una vez ejecutada las simulaciones correspondientes, se procede con la adquisición de los datos de cada modelo para lo cual se implementó un archivo .m de Matlab cuyo código se detalla en el anexo 2.

El código descrito en el anexo 2 también permite crear un archivo en Excel que contiene todos los datos de la simulación realizada.

Este archivo será utilizado para crear gráficas que permitan realizar el análisis de cada variable como se visualiza en la figura 2.48.

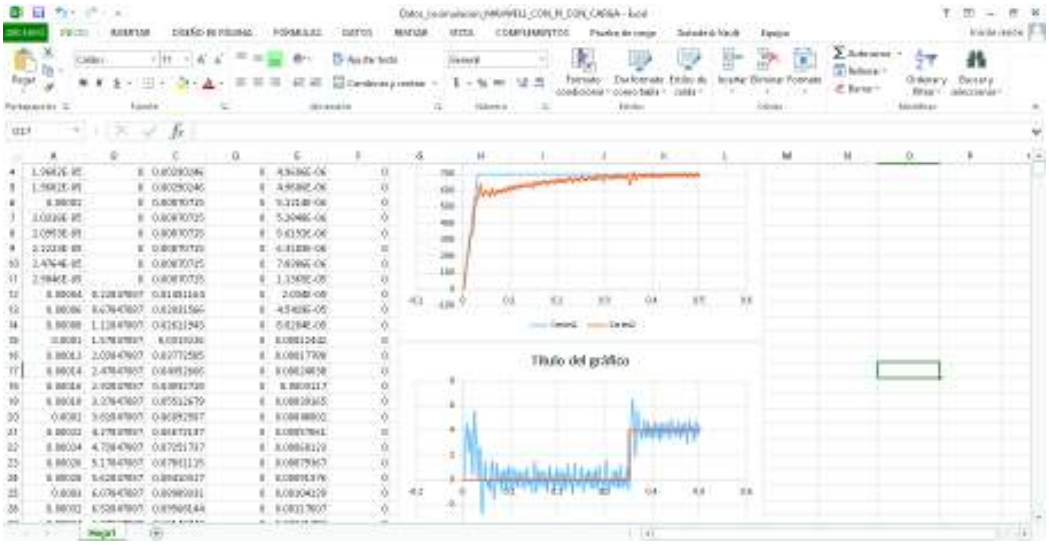


Figura 2.48 Ventana de Excel con los datos obtenidos

La revisión y comparación de resultados se realizará en el capítulo número 3 junto al análisis del comportamiento del controlador y de cada uno de los modelos de la PMSM implementados.

3 RESULTADOS Y DISCUSIÓN

3.1 Resultados

En el presente capítulo se muestran los resultados de las simulaciones realizadas con el modelo creado mediante librerías de Matlab, el modelo creado mediante Ansys RMxprt-Simplorer-Matlab (modelo sin F.E.A.), y el modelo creado en Ansys Maxwell 2D-Simplorer-Matlab (modelo con F.E.A) en los siguientes escenarios:

- Modelo con variación de velocidad a 450 RPMs, torque de carga T_L=4N*m en el instante t=0 segundos y sobrecarga T_L=5N*m en el instante t=0.35 segundos.
- Modelo con variación de velocidad a 900 RPMs, torque de carga T_L=4N*m en el instante t=0 segundos y sobrecarga T_L=5N*m en el instante t=0.35 segundos.

3.2 Resultados obtenidos de la simulación del modelo de la PMSM implementado en Matlab/Simulink.

El diagrama de bloques de la figura 3.1 corresponde al modelo de control de la PMSM creado con las librerías de Matlab-Simulink, el cual permite simular y obtener las gráficas de comportamiento bajo variación de velocidad con carga y sobrecarga.

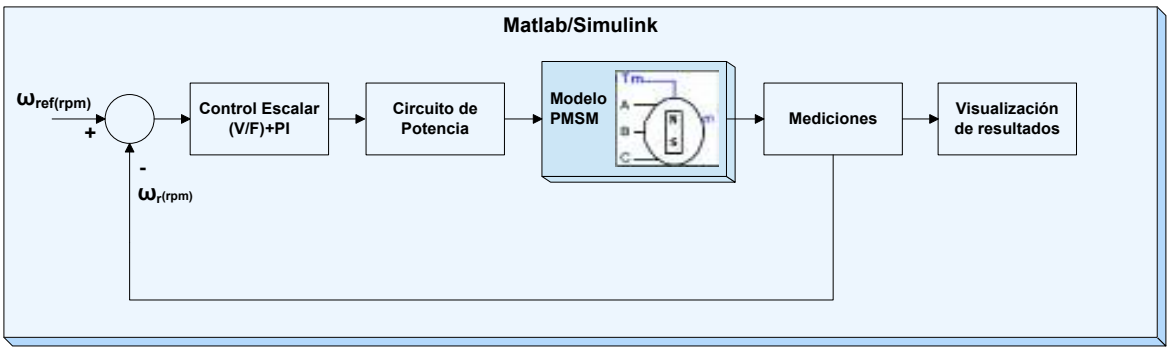


Figura 3.1 Diagrama de bloques del modelo de control de la PMSM en Matlab-Simulink

3.2.1 Modelo con variación de velocidad y torque de carga en lazo abierto.

Los resultados obtenidos de la simulación del modelo de la PMSM en lazo abierto aplicando un torque de carga de 4N*m en el instante t=0 segundos y sobrecarga de 5N*m en el instante t=0.35 segundos se muestra en la figura 3.2.

Como se visualiza en la figura 3.2, se especifica una velocidad referencia de 1800 rpm. En el instante del arranque se puede visualizar un sobre pico de velocidad de 2474 rpm y un torque pico de 61.57 N*m. En el instante t=0.35s, se aplica un torque de sobrecarga de 5N*m y se puede visualizar una caída de velocidad hasta las 1783 rpm y sobre pico de

torque de 5.726N*m. El tiempo de estabilización posterior al arranque es de 0.2 segundos y al aplicar el torque de carga el tiempo de estabilización se da en t = 0.45segundos.

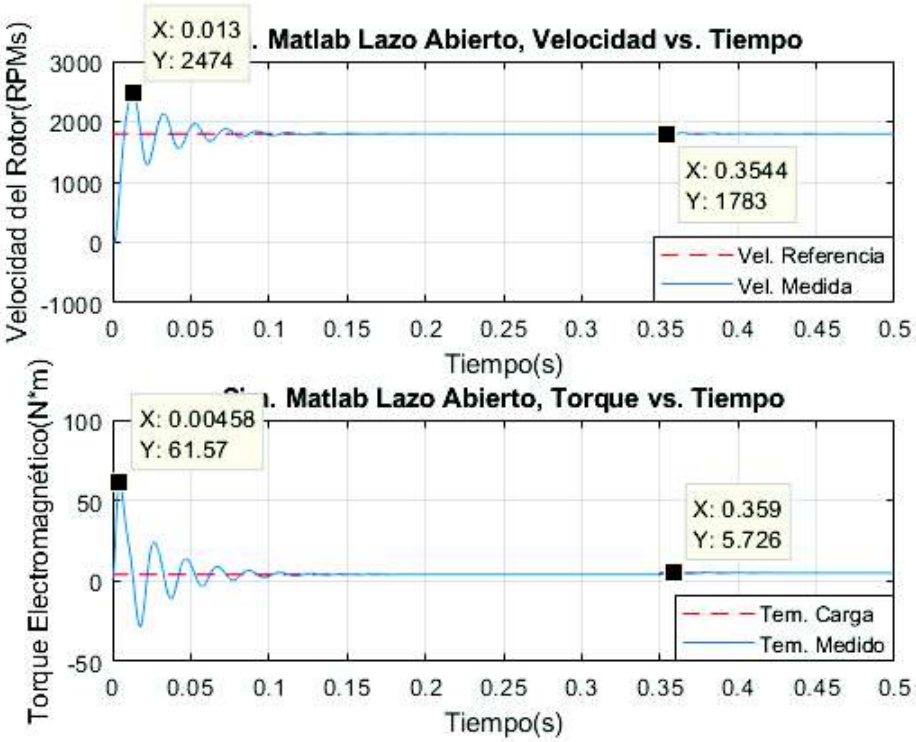


Figura 3.2 Respuesta de velocidad y torque electromagnético de la PMSM en lazo abierto

Como se puede visualizar en la figura 3.2, la respuesta de velocidad y de torque no es adecuada por lo cual se utiliza un controlador Proporcional-Integral para mejorar la respuesta del estado transitorio y estable de la PMSM como se visualiza en las siguientes secciones.

3.2.2 Modelo con variación de velocidad a 450 RPMs, T_L=4N*m a t=0 segundos, T_L=5N*m a t=0.35 segundos.

Los resultados de la simulación del modelo de la PMSM con el controlador proporcional-integral diseñado en Matlab-Simulink se muestran en la sección 3.2.2.1. Las condiciones de velocidad bajo las cuales se realizó la simulación están dadas por el vector de velocidad V= [450] rpm y vector de torque de carga T_L= [4 5] N*m.

3.2.2.1 Respuesta de velocidad y torque electromagnético

En la figura 3.3 se visualiza la respuesta de la velocidad del rotor y torque electromagnético ante el perfil de velocidad dado por el vector V= [450] rpm y T_L= [4 5] N*m. Los puntos de interés: Punto 1 Y Punto 2 marcados muestran el comportamiento del controlador PI al alcanzar la referencia de velocidad de 450 [rpm].

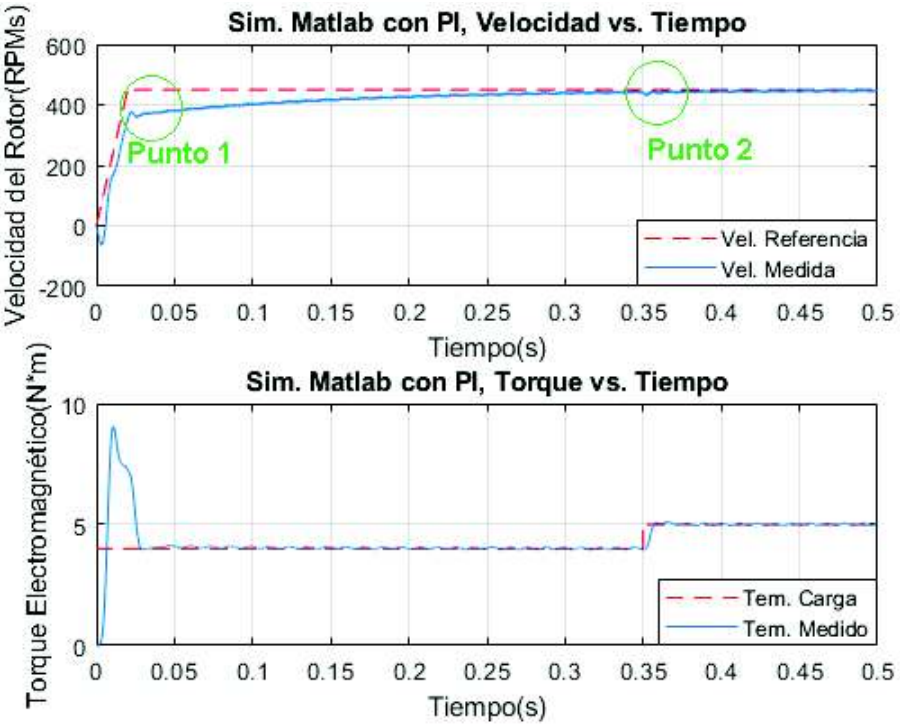


Figura 3.3 Respuesta de velocidad y torque, V_{ref}= 450 RPMs, T_L=[4 5] N*m

La figura 3.4 y 3.5 amplifica la figura 3.3 para el análisis detallado del punto 1 y punto 2 con el objetivo de determinar el valor del error en estado transitorio y estado estable de la velocidad medida respecto a la velocidad de referencia.

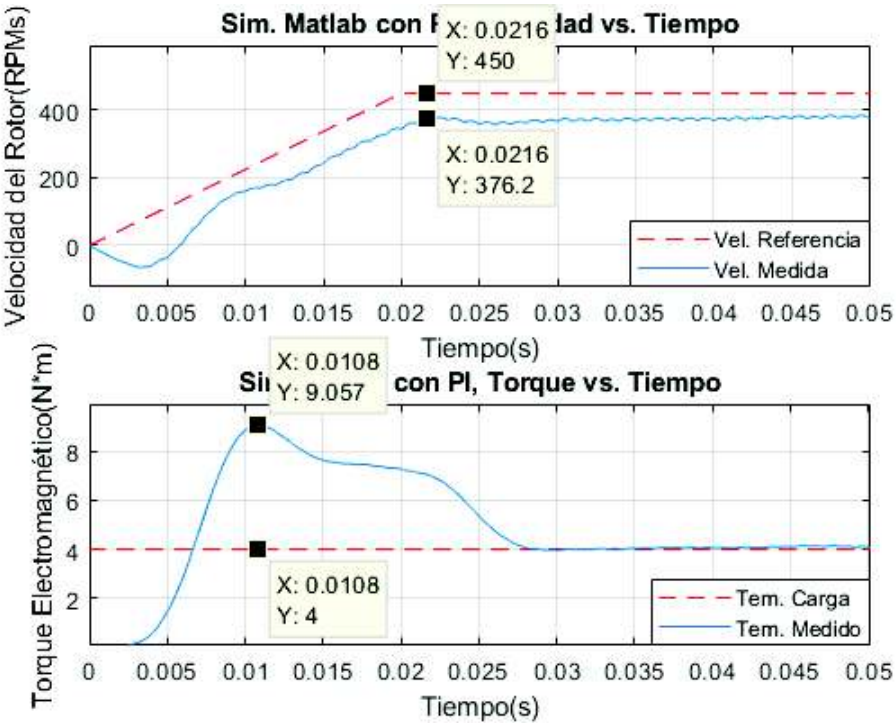


Figura 3.4 Respuesta de velocidad y torque en el punto de interés N° 1

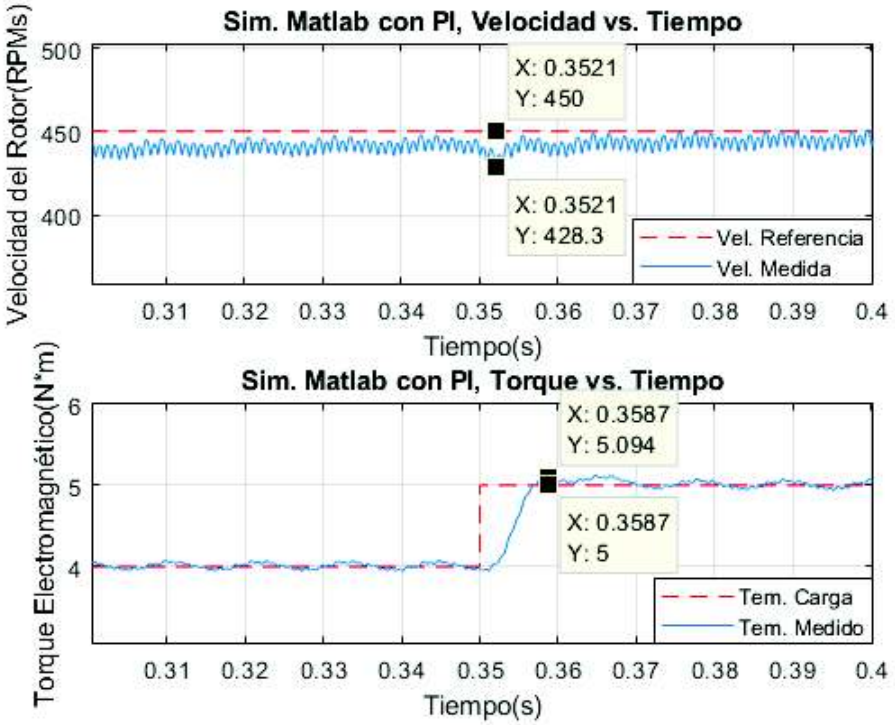


Figura 3.5 Respuesta de velocidad y torque en el punto de interés N° 2

El cálculo del porcentaje de error en estado transitorio generado en el punto de interés 1 y 2 se realiza mediante la ecuación 3.1.

$$\%error = \frac{|Valor\ de\ referencia - Valor\ medido|}{Valor\ de\ referencia} * 100$$

Ecuación 3.1 Ecuación para calcular el porcentaje de error relativo

La tabla 3.1 muestra el porcentaje de error generado en estado transitorio de la velocidad medida respecto a la velocidad de referencia en el momento de arranque con T_L=4N*m y al aplicar una sobrecarga T_L=5N*m. Los errores calculados corresponden a la mejor respuesta obtenida después de realizar la sintonización del controlador PI. El torque electromagnético pico obtenido en el momento del arranque es 9.057N*m.

Tabla 3.1 Cálculo del error en estado transitorio de la velocidad del rotor

Velocidad de referencia (rpm)	T_L (N*m)	Instante t(s)	Velocidad medida (rpm)	Error máximo (%)
450	4	0	376.2	16.4%
450	5	0.35	428.3	4.82%

El tiempo de estabilización de la velocidad luego de aplicar el torque de sobrecarga es de 0.1 segundos. La figura 3.6 permite visualiza la respuesta del controlador al alcanzar el estado estable.

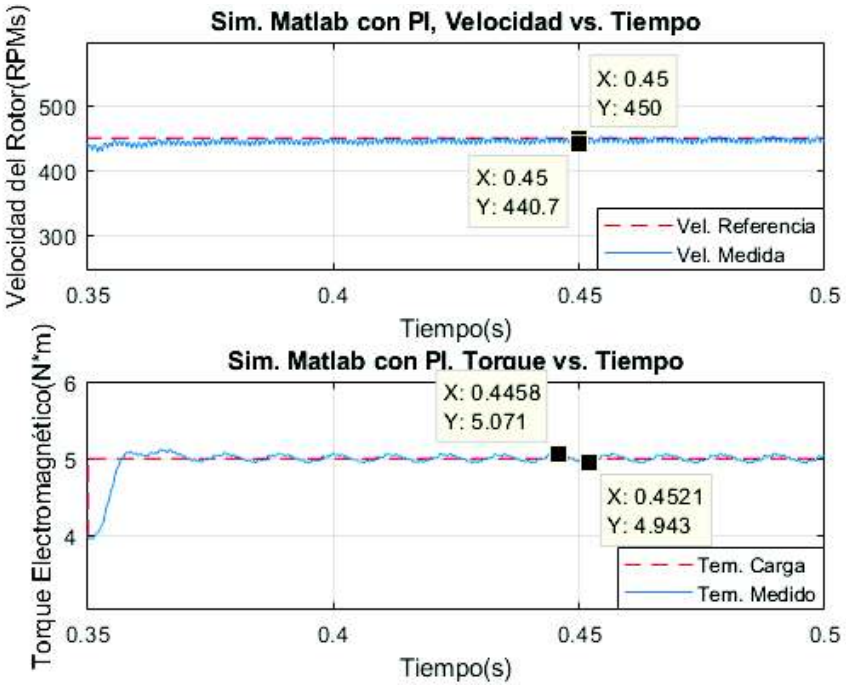


Figura 3.6 Respuesta de velocidad y torque en estado estable

El cálculo del error en estado estable una vez transcurrido el tiempo de estabilización se realiza mediante la ecuación 3.1. La tabla 3.2 muestra el porcentaje de error generado en estado estable de la velocidad medida respecto a la velocidad de referencia. El error obtenido es 2.07% y corresponde a la mejor respuesta obtenida después de realizar la sintonización del controlador PI. El tiempo de estabilización determinado en la figura 3.6 es de 0.45 segundos.

Tabla 3.2 Cálculo del error en estado estable de la velocidad del rotor

Velocidad de referencia (rpm)	Velocidad medida (rpm)	Error máximo (%)
450	440.7	2.07%

La figura 3.6 permite también visualizar el torque electromagnético generado en estado estable el cual oscila entre 5.071N*m y 4.943 N*m que equivale a un rizado de 0.128N*m.

3.2.3 Modelo con variación de velocidad a 900 RPMs, T_L=4N*m a t=0 segundos, T_L=5N*m a t=0.35 segundos.

Los resultados de la simulación del modelo de la PMSM con el controlador proporcional-integral diseñado en Matlab-Simulink se muestran en la sección 3.2.3.1. Las condiciones de velocidad bajo las cuales se realizó la simulación están dadas por el vector de velocidad V= [900] rpm y vector de torque de carga T_L= [4 5] N*m.

3.2.3.1 Respuesta de velocidad y torque electromagnético

En la figura 3.7 se visualiza la respuesta de la velocidad del rotor y torque electromagnético ante el perfil de velocidad dado por el vector V= [900] rpm y T_L= [4 5] N*m.

Los puntos de interés: Punto 1 Y Punto 2 marcados muestran el comportamiento del controlador PI al alcanzar la referencia de velocidad de 900 [rpm].

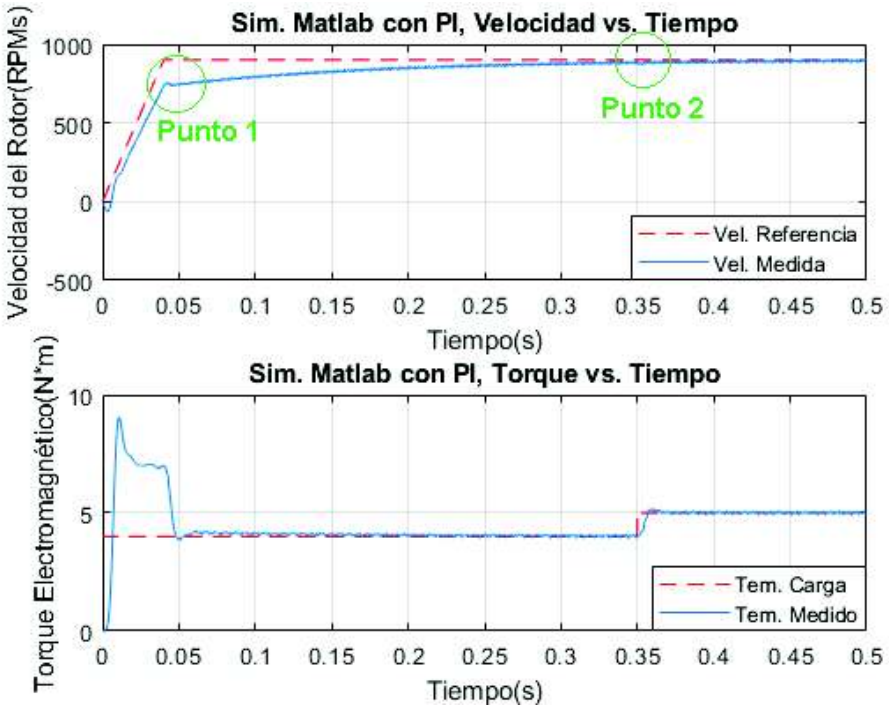


Figura 3.7 Respuesta de velocidad y torque, V_{ref}= 900 RPMs, T_L=[4 5] N*m

La figura 3.8 y 3.9 amplifica la figura 3.7 para el análisis detallado del punto 1 y punto 2 con el objetivo de determinar el valor del error en estado transitorio y estado estable de la velocidad medida respecto a la velocidad de referencia.

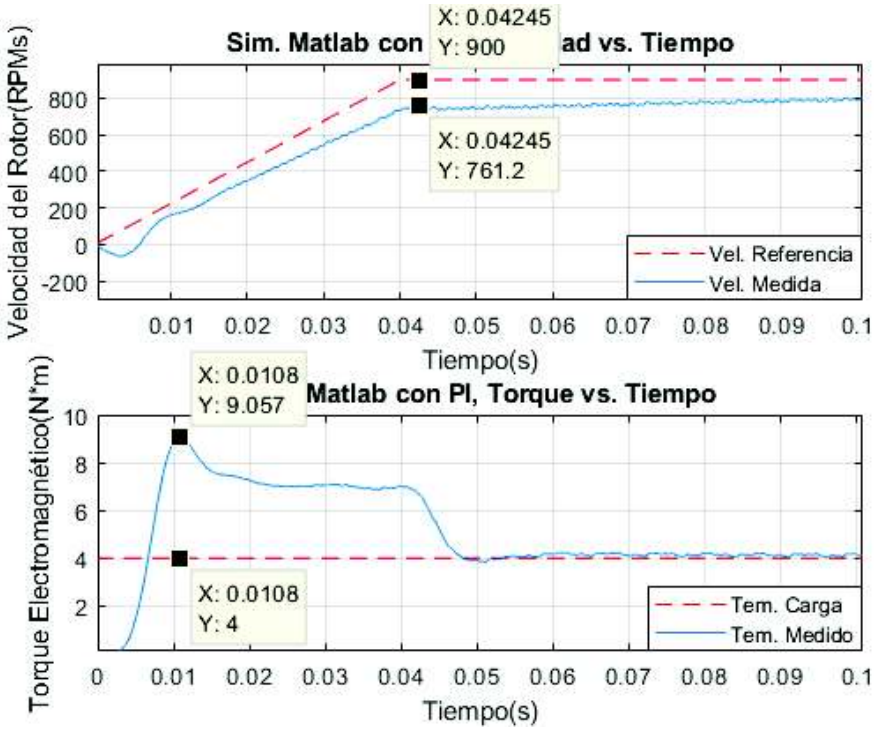


Figura 3.8 Respuesta de velocidad y torque en el punto de interés N° 1

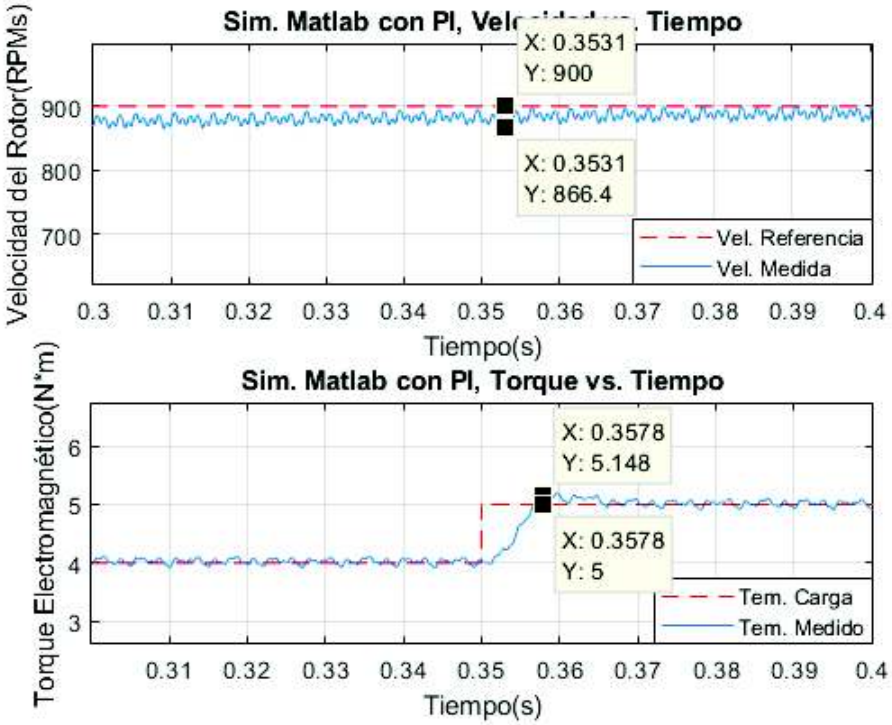


Figura 3.9 Respuesta de velocidad y torque en el punto de interés N° 2

El cálculo del error en estado transitorio generado en el punto de interés 1 y 2 se realiza mediante la ecuación 3.1.

La tabla 3.3 muestra el porcentaje de error generado en estado transitorio de la velocidad medida respecto a la velocidad de referencia en el momento de arranque con T_L=4N*m y al aplicar una sobrecarga T_L=5N*m. Los errores calculados corresponden a la mejor respuesta obtenida después de realizar la sintonización del controlador PI. El torque electromagnético pico obtenido en el momento del arranque es 9.057N*m.

Tabla 3.3 Cálculo del error en estado transitorio de la velocidad del rotor

Velocidad de referencia (rpm)	T_L (N*m)	Instante t(s)	Velocidad medida (rpm)	Error máximo (%)
900	4	0	761.2	15.42%
900	5	0.35	866.4	3.73%

El tiempo de estabilización de la velocidad luego de aplicar el torque de sobrecarga es de 0.1 segundos.

La figura 3.10 permite visualizar la respuesta del controlador al alcanzar el estado estable a la velocidad de referencia de 900PRMs con un torque de sobrecarga aplicado T_L=5N*m.

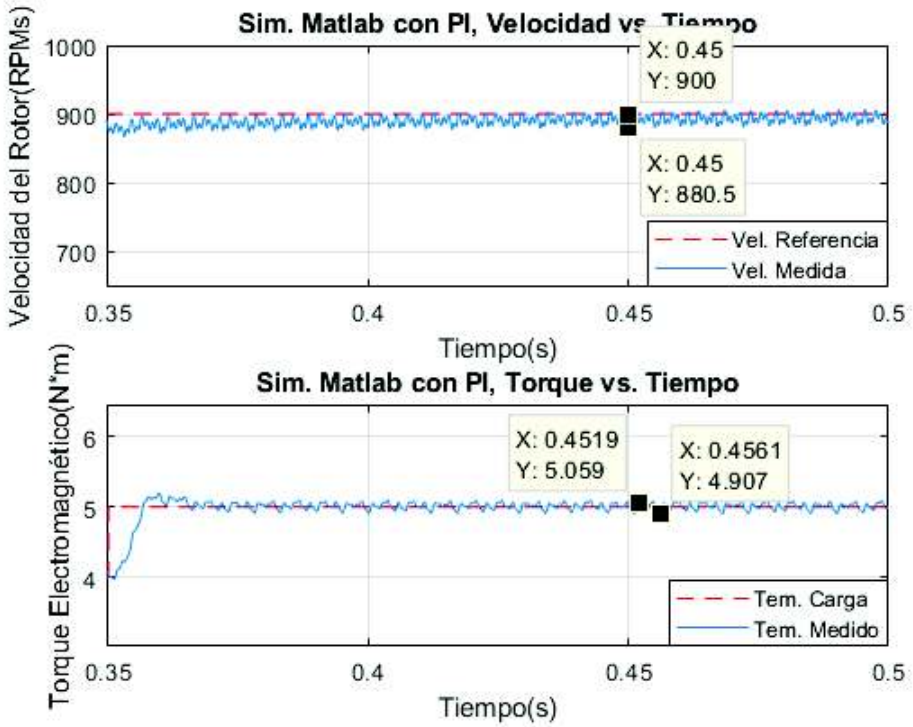


Figura 3.10 Respuesta de velocidad y torque en estado estable

El cálculo del error en estado estable una vez transcurrido el tiempo de estabilización se realiza mediante la ecuación 3.1. La tabla 3.4 muestra el porcentaje de error generado en estado estable de la velocidad medida respecto a la velocidad de referencia. El error obtenido es 2.17% y corresponde a la mejor respuesta obtenida después de realizar la sintonización del controlador PI. El tiempo de estabilización determinado en la figura 3.10 es de 0.45 segundos.

Tabla 3.4 Cálculo del error en estado estable de la velocidad del rotor

Velocidad de referencia (rpm)	Velocidad medida (rpm)	Error máximo (%)
900	880.5	2.17%

La figura 3.10 permite también visualizar el torque electromagnético generado en estado estable el cual oscila entre 5.059 N*m y 4.907 N*m que equivale a un rizado de 0.152N*m.

3.3 Resultados obtenidos mediante co-simulación Ansys RMxprt-Simplorer-Matlab.

El modelo implementado en la figura 3.11 enlaza la PMSM creada en Ansys RMxprt con el circuito de potencia constituido por el inversor y la fuente DC creados en Simplorer. Las señales de realimentación y de control serán gestionadas por medio del vínculo de co-simulación generado entre Simplorer y Matlab Simulink

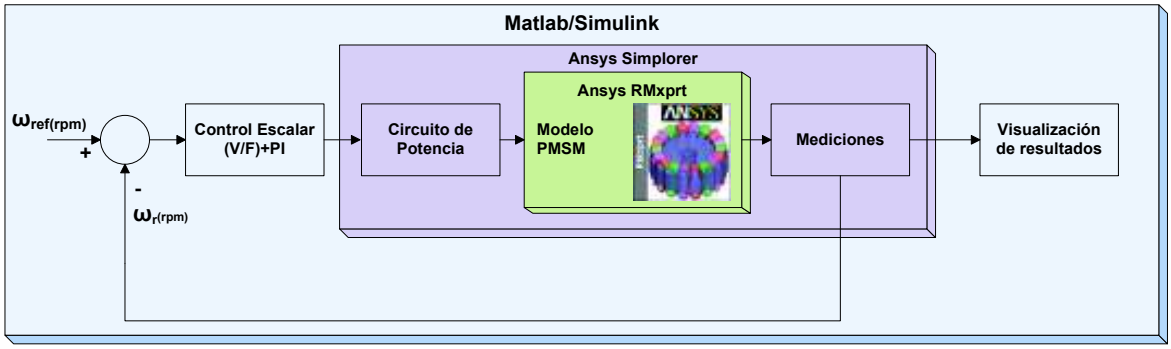


Figura 3.11 Diagrama de bloques de co-simulación Ansys RMxprt-Simplorer-Matlab

3.3.1 Modelo con variación de velocidad a 450 RPMs, T_L=4N*m a t=0 segundos, T_L=5N*m a t=0.35 segundos.

Los resultados de la simulación del modelo de la PMSM con el controlador proporcional-integral diseñado en Matlab-Simulink se muestran en la sección 3.3.1.1. Las condiciones de velocidad bajo las cuales se realizó la simulación están dadas por el vector de velocidad V= [450] rpm y vector de torque de carga T_L= [4 5] N*m.

3.3.1.1 Respuesta de velocidad y torque electromagnético

En la figura 3.12 se visualiza la respuesta de la velocidad del rotor y torque electromagnético ante el perfil de velocidad dado por el vector V= [450] rpm y T_L= [4 5] N*m. Los puntos de interés: Punto 1 Y Punto 2 marcados muestran el comportamiento del controlador PI al alcanzar la referencia de velocidad de 450 [rpm].

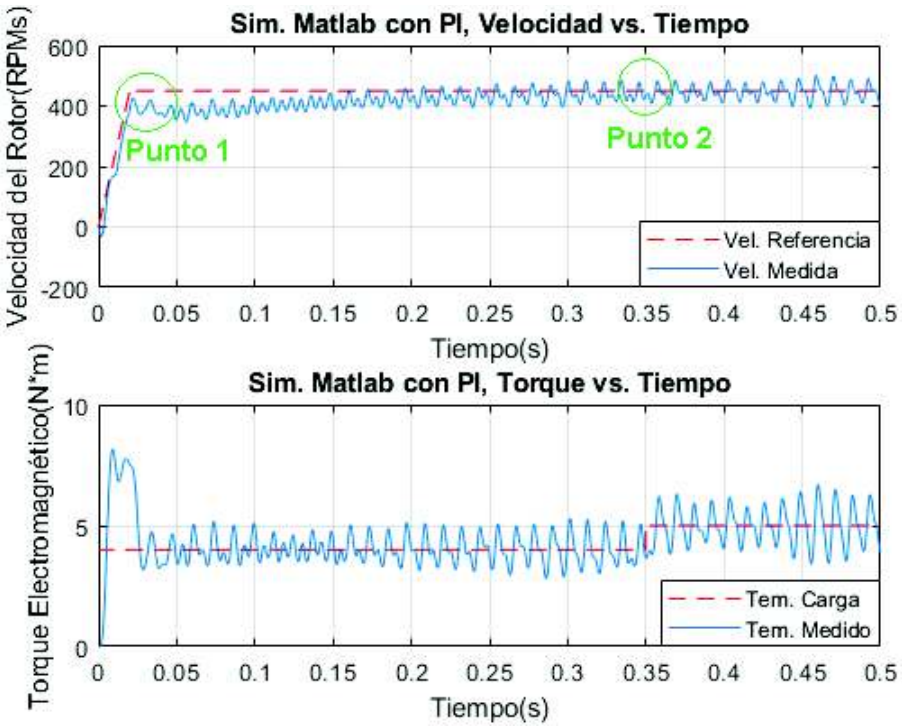


Figura 3.12 Respuesta de velocidad y torque, V_{ref}= 450 RPMs, T_L=[4 5] N*m

La figura 3.13 y 3.14 amplifica la figura 3.12 para el análisis detallado del punto 1 y punto 2 con el objetivo de determinar el valor del error en estado transitorio y estado estable de la velocidad medida respecto a la velocidad de referencia.

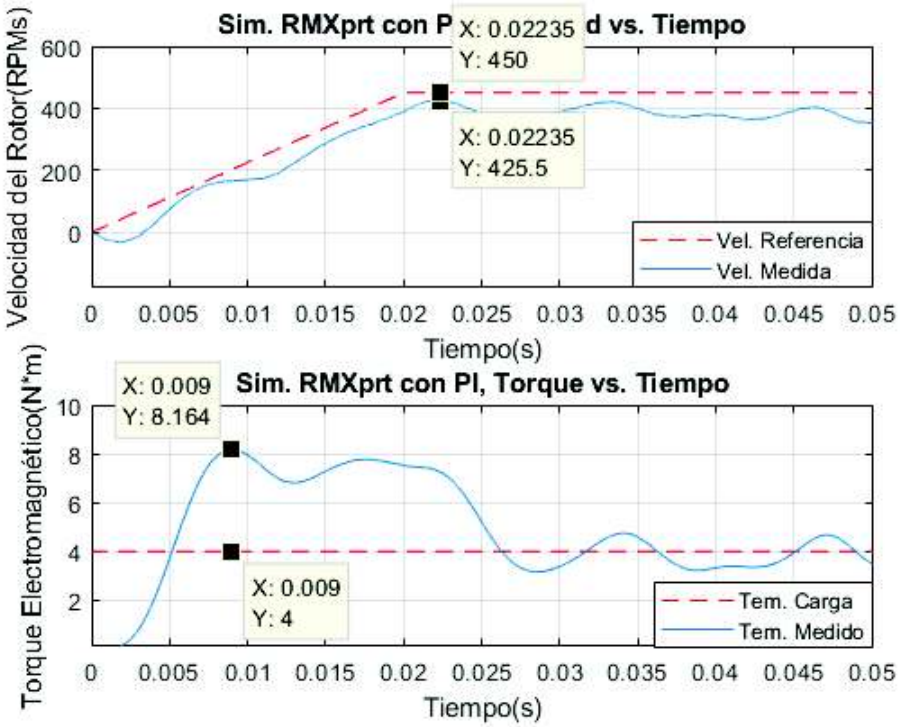


Figura 3.13 Respuesta de velocidad y torque en el punto de interés N° 1

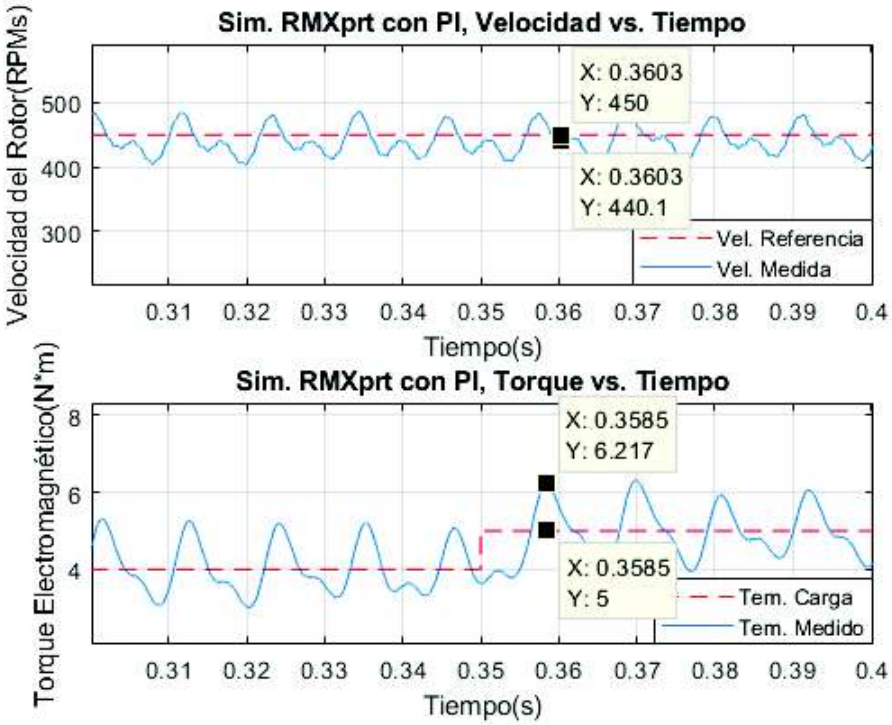


Figura 3.14 Respuesta de velocidad y torque en el punto de interés N° 2

El cálculo del error en estado transitorio generado en el punto de interés 1 y 2 se realiza mediante la ecuación 3.1. La tabla 3.5 muestra el porcentaje de error generado en estado transitorio de la velocidad medida respecto a la velocidad de referencia en el momento de arranque con T_L=4N*m y al aplicar una sobrecarga T_L=5N*m. Los errores calculados corresponden a la mejor respuesta obtenida después de realizar la sintonización del controlador PI. El torque electromagnético pico obtenido en el momento del arranque es 8.164 N*m.

Tabla 3.5 Cálculo del error en estado transitorio de la velocidad del rotor

Velocidad de referencia (rpm)	T_L (N*m)	Instante t(s)	Velocidad medida (rpm)	Error máximo (%)
450	4	0	425.5	5.44%
450	5	0.35	440.1	2.2%

El tiempo de estabilización de la velocidad luego de aplicar el torque de sobrecarga es de 0.1 segundos.

La figura 3.15 permite visualizar la respuesta del controlador al alcanzar el estado estable a la velocidad de referencia de 450 RPMs con un torque de sobrecarga aplicado T_L=5N*m.

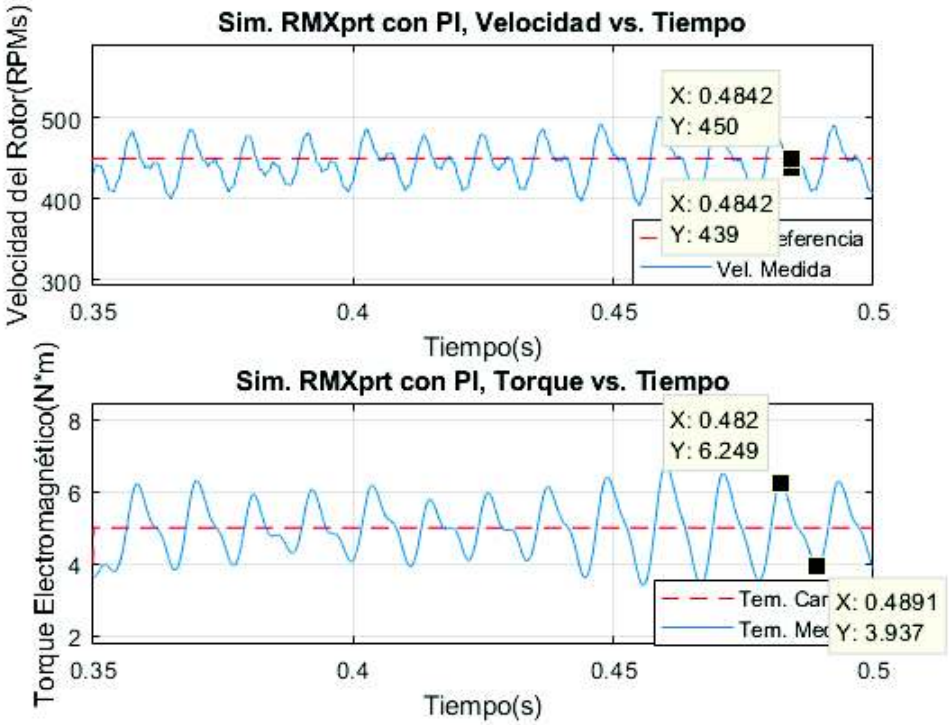


Figura 3.15 Respuesta de velocidad y torque en estado estable

El cálculo del error en estado estable una vez transcurrido el tiempo de estabilización se realiza mediante la ecuación 3.1. La tabla 3.6 muestra el porcentaje de error generado en estado estable de la velocidad medida respecto a la velocidad de referencia. El error

obtenido es 2.44% y corresponde a la mejor respuesta obtenida después de realizar la sintonización del controlador PI. El tiempo de estabilización determinado en la figura 3.15 es de 0.45 segundos.

Tabla 3.6 Cálculo del error en estado estable de la velocidad del rotor

Velocidad de referencia (rpm)	Velocidad medida (rpm)	Error máximo (%)
450	439	2.44%

La figura 3.15 permite también visualizar el torque electromagnético generado en estado estable el cual oscila entre 6.249 N*m y 3.937 N*m que equivale a un rizado de 2.312N*m.

3.3.2 Modelo con variación de velocidad a 900 RPMs, T_L=4N*m a t=0 segundos, T_L=5N*m a t=0.35 segundos.

Los resultados de la simulación del modelo de la PMSM con el controlador proporcional-integral diseñado en Matlab-Simulink se muestran en la sección 3.3.2.1. Las condiciones de velocidad bajo las cuales se realizó la simulación están dadas por el vector de velocidad V= [900] rpm y vector de torque de carga T_L= [4 5] N*m.

3.3.2.1 Respuesta de velocidad y torque electromagnético

En la figura 3.16 se visualiza la respuesta de la velocidad del rotor y torque electromagnético ante el perfil de velocidad dado por el vector V= [900] rpm y T_L= [4 5] N*m. Los puntos de interés: Punto 1 Y Punto 2 marcados muestran el comportamiento del controlador PI al alcanzar la referencia de velocidad de 900 [rpm].

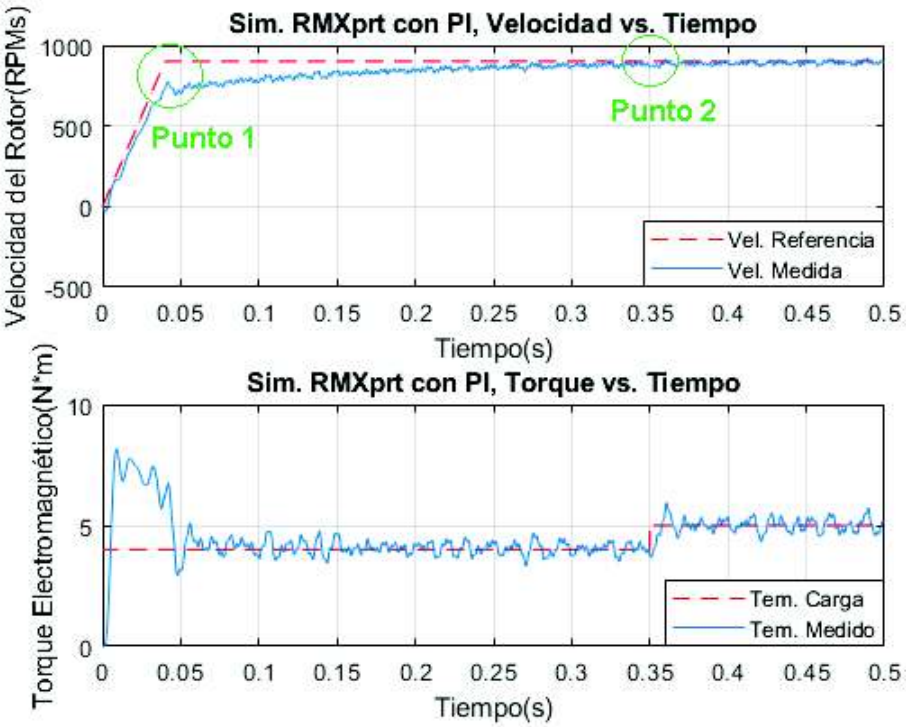


Figura 3.16 Respuesta de velocidad y torque, V_{ref}= 900 RPMs, T_L=[4 5] N*m

La figura 3.17 y 3.18 amplifica la figura 3.16 para el análisis detallado del punto 1 y punto 2 con el objetivo de determinar el valor del error en estado transitorio y estado estable de la velocidad medida respecto a la velocidad de referencia.

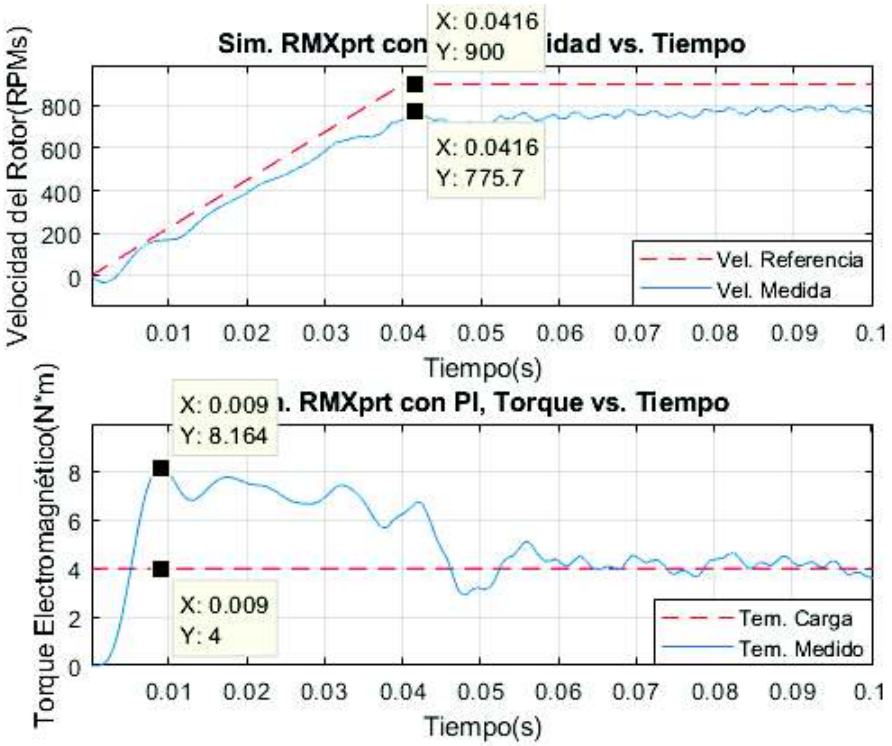


Figura 3.17 Respuesta de velocidad y torque en el punto de interés N° 1

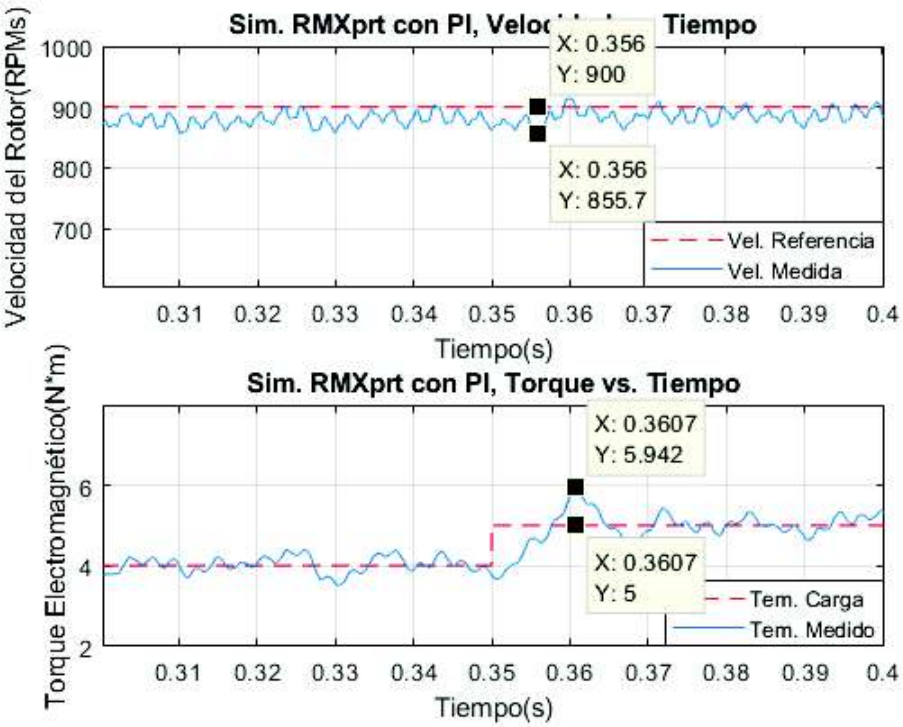


Figura 3.18 Respuesta de velocidad y torque en el punto de interés N° 2

El cálculo del error en estado transitorio generado en el punto de interés 1 y 2 se realiza mediante la ecuación 3.1. La tabla 3.7 muestra el porcentaje de error generado en estado transitorio de la velocidad medida respecto a la velocidad de referencia en el momento de arranque con T_L=4N*m y al aplicar una sobrecarga T_L=5N*m. Los errores calculados corresponden a la mejor respuesta obtenida después de realizar la sintonización del controlador PI. El torque electromagnético pico obtenido en el momento del arranque es 8.164 N*m.

Tabla 3.7 Cálculo del error en estado transitorio de la velocidad del rotor

Velocidad de referencia (rpm)	T_L (N*m)	Instante t(s)	Velocidad medida (rpm)	Error máximo (%)
900	4	0	775.7	13.81%
900	5	0.35	855.7	4.92%

El tiempo de estabilización de la velocidad luego de aplicar el torque de sobrecarga es de 0.1 segundos.

La figura 3.19 permite visualizar la respuesta del controlador al alcanzar el estado estable a la velocidad de referencia de 900 RPMs con un torque de sobrecarga aplicado T_L=5N*m.

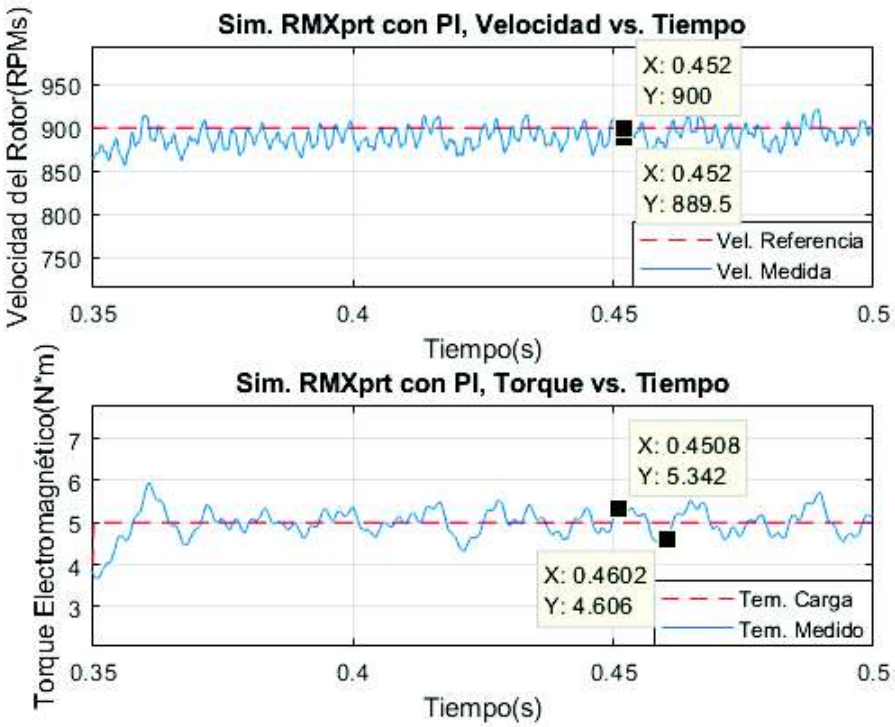


Figura 3.19 Respuesta de velocidad y torque en estado estable

El cálculo del error en estado estable una vez transcurrido el tiempo de estabilización se realiza mediante la ecuación 3.1. La tabla 3.8 muestra el porcentaje de error generado en estado estable de la velocidad medida respecto a la velocidad de referencia. El error obtenido es 1.17% y corresponde a la mejor respuesta obtenida después de realizar la

sintonización del controlador PI. El tiempo de estabilización determinado en la figura 3.19 es de 0.45 segundos.

Tabla 3.8 Cálculo del error en estado estable de la velocidad del rotor

Velocidad de referencia (rpm)	Velocidad medida (rpm)	Error máximo (%)
900	889.5	1.17%

La figura 3.19 permite también visualizar el torque electromagnético generado en estado estable el cual oscila entre 5.342 N*m y 4.606 N*m que equivale a un rizado de 0.736N*m.

3.4 Resultados obtenidos mediante co-simulación Ansys Maxwell2D-Simplorer-Matlab.

El modelo implementado para la co-simulación entre Maxwell 2D-Simplorer-Simulink es el mostrado en la figura 3.20.

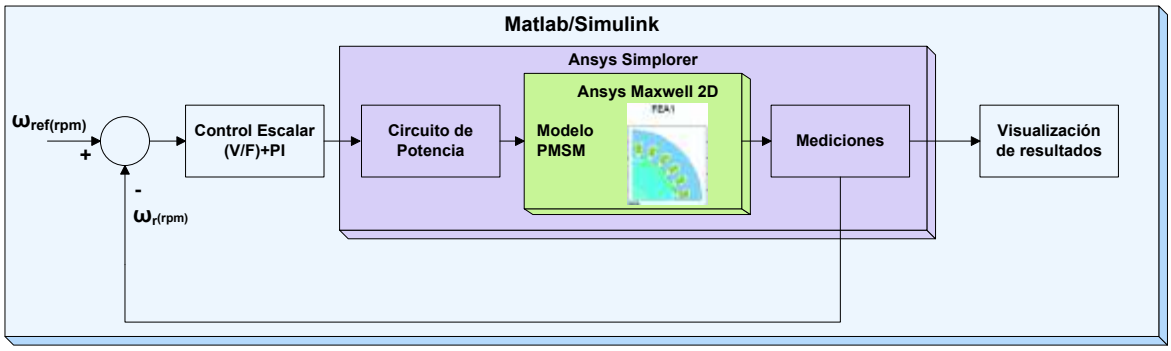


Figura 3.20 Diagrama de bloques de co-simulación Ansys Maxwell 2D-Simplorer-Matlab

El modelo de la figura 3.20 enlaza la PMSM creada en Ansys Maxwell 2D con el circuito de potencia creado en Simplorer. Para llevar a cabo el enlace se establece el vínculo de co-simulación entre Maxwell 2D y Simplorer. Las señales de realimentación y de control serán gestionadas por medio del vínculo de co-simulación generado entre Simplorer y Simulink.

3.4.1 Modelo con variación de velocidad a 450 RPMs, T_L=4N*m a t=0 segundos, T_L=5N*m a t=0.35 segundos.

Los resultados de la simulación del modelo de la PMSM con el controlador proporcional-integral diseñado en Matlab-Simulink se muestran en la sección 3.4.1.1. Las condiciones de velocidad bajo las cuales se realizó la simulación están dadas por el vector de velocidad V= [450] rpm y vector de torque de carga T_L= [4 5] N*m.

3.4.1.1 Respuesta de velocidad y torque electromagnético

En la figura 3.21 se visualiza la respuesta de la velocidad del rotor y torque electromagnético ante el perfil de velocidad dado por el vector V= [450] rpm y T_L= [4 5] N*m. Los puntos de interés: Punto 1 Y Punto 2 marcados muestran el comportamiento del controlador PI al alcanzar la referencia de velocidad de 450 [rpm].

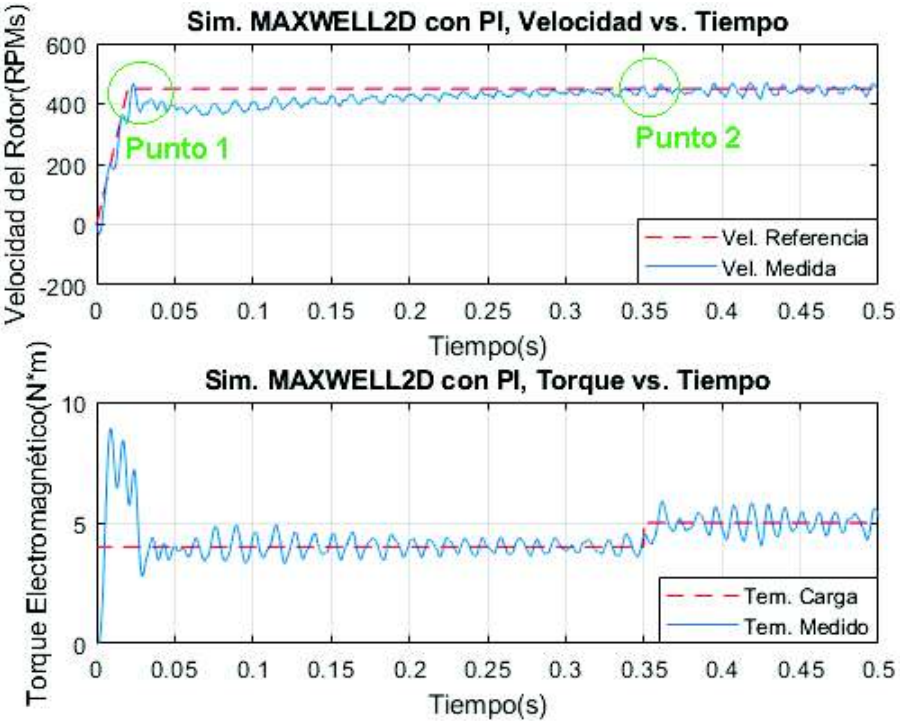


Figura 3.21 Respuesta de velocidad y torque, V_{ref}= 450 RPMs, T_L=[4 5] N*m

La figura 3.22 y 3.23 amplifica la figura 3.21 para el análisis detallado del punto 1 y punto 2 con el objetivo de determinar el valor del error en estado transitorio y estado estable de la velocidad medida respecto a la velocidad de referencia.

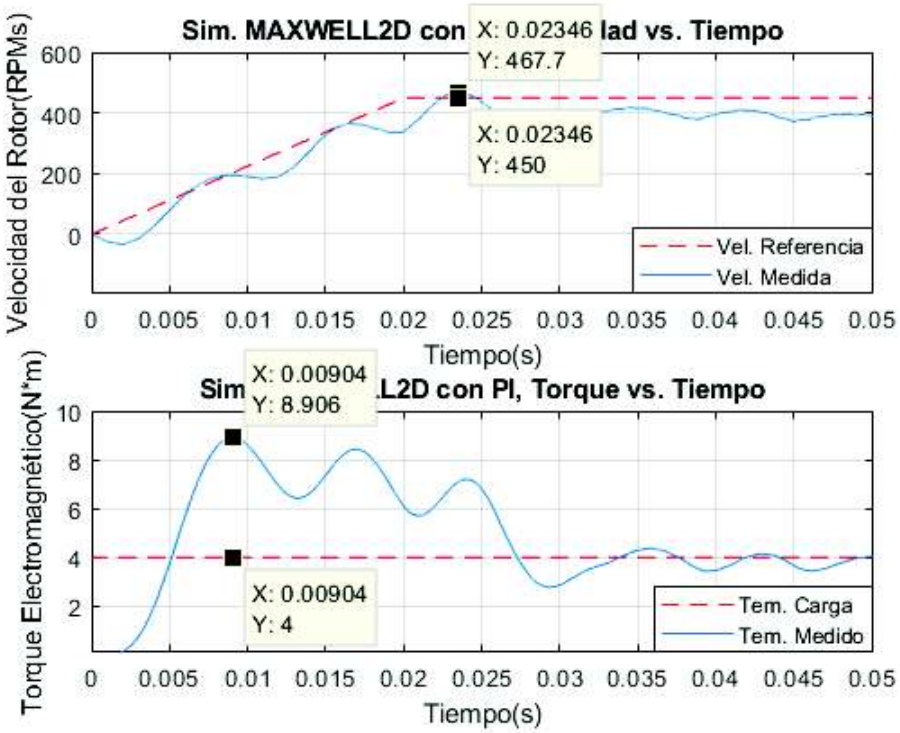


Figura 3.22 Respuesta de velocidad y torque en el punto de interés N° 1

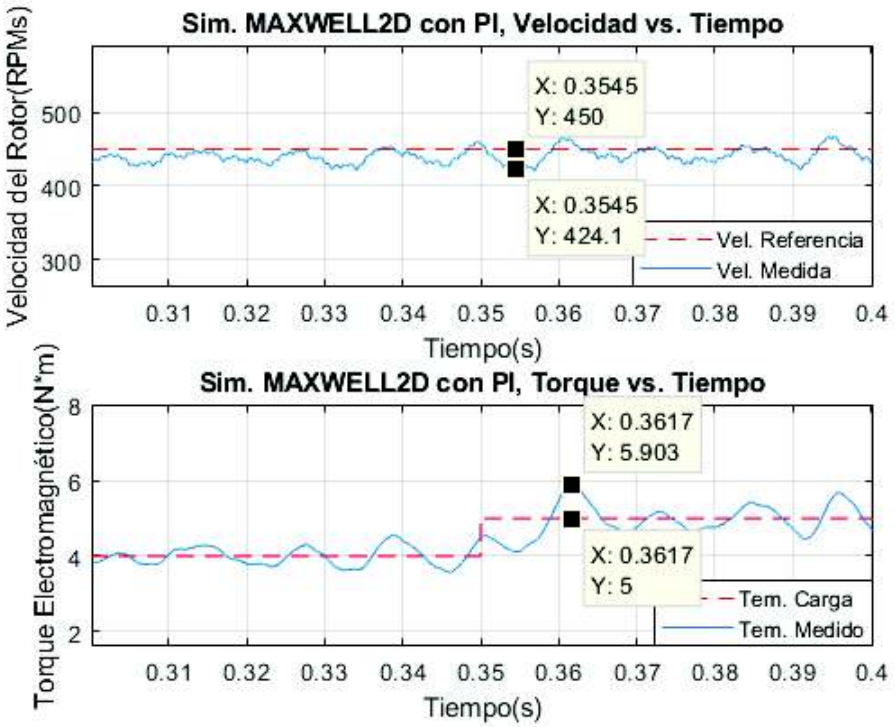


Figura 3.23 Respuesta de velocidad y torque en el punto de interés N° 2

El cálculo del error en estado transitorio generado en el punto de interés 1 y 2 se realiza mediante la ecuación 3.1. La tabla 3.9 muestra el porcentaje de error generado en estado transitorio de la velocidad medida respecto a la velocidad de referencia en el momento de arranque con T_L=4N*m y al aplicar una sobrecarga T_L=5N*m. Los errores calculados corresponden a la mejor respuesta obtenida después de realizar la sintonización del controlador PI. El torque electromagnético pico obtenido en el momento del arranque es 8.906 N*m.

Tabla 3.9 Cálculo del error en estado transitorio de la velocidad del rotor

Velocidad de referencia (rpm)	T_L (N*m)	Instante t(s)	Velocidad medida (rpm)	Error máximo (%)
450	4	0	467.7	3.93%
450	5	0.35	424.1	5.75%

El tiempo de estabilización de la velocidad luego de aplicar el torque de sobrecarga es de 0.1 segundos.

La figura 3.24 permite visualizar la respuesta del controlador al alcanzar el estado estable a la velocidad de referencia de 450 RPMs con un torque de sobrecarga aplicado T_L=5N*m.

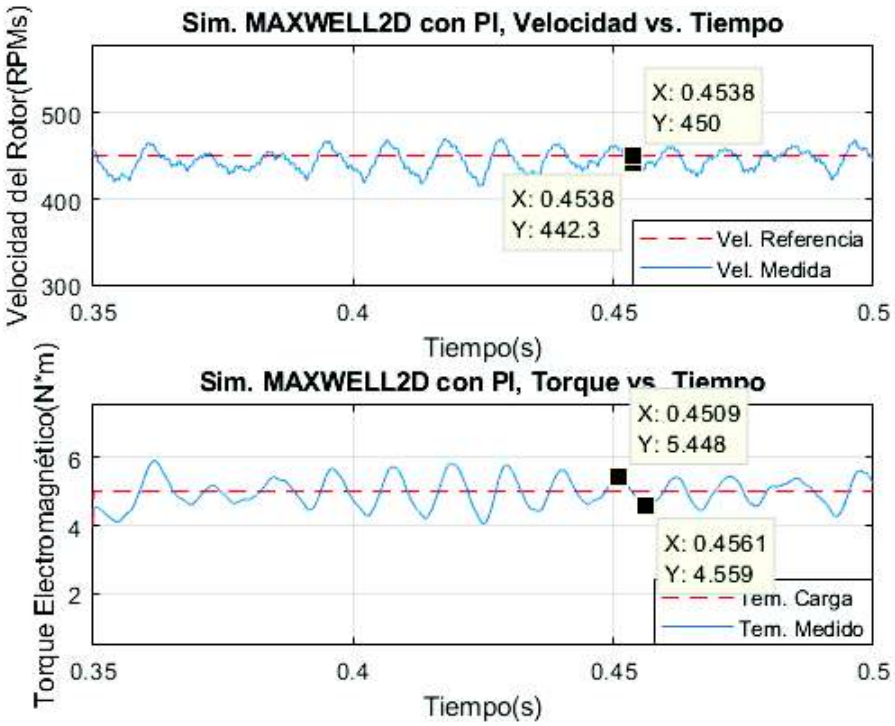


Figura 3.24 Respuesta de velocidad y torque en estado estable

El cálculo del error en estado estable una vez transcurrido el tiempo de estabilización se realiza mediante la ecuación 3.1. La tabla 3.10 muestra el porcentaje de error generado en estado estable de la velocidad medida respecto a la velocidad de referencia. El error obtenido es 1.71% y corresponde a la mejor respuesta obtenida después de realizar la sintonización del controlador PI. El tiempo de estabilización determinado en la figura 3.24 es de 0.45 segundos.

Tabla 3.10 Cálculo del error en estado estable de la velocidad del rotor

Velocidad de referencia (rpm)	Velocidad medida (rpm)	Error máximo (%)
450	442.3	1.71%

La figura 3.24 permite también visualizar el torque electromagnético generado en estado estable el cual oscila entre 5.448 N*m y 4.559 N*m que equivale a un rizado de 0.889 N*m.

3.4.2 Modelo con variación de velocidad a 900 RPMs, T_L=4N*m a t=0 segundos, T_L=5N*m a t=0.35 segundos.

Los resultados de la simulación del modelo de la PMSM con el controlador proporcional-integral diseñado en Matlab-Simulink se muestran en la sección 3.4.2.1. Las condiciones de velocidad bajo las cuales se realizó la simulación están dadas por el vector de velocidad V= [900] rpm y vector de torque de carga T_L= [4 5] N*m.

3.4.2.1 Respuesta de velocidad y torque electromagnético

En la figura 3.25 se visualiza la respuesta de la velocidad del rotor y torque electromagnético ante el perfil de velocidad dado por el vector V= [900] rpm y T_L= [4 5] N*m.

Los puntos de interés: Punto 1 Y Punto 2 marcados muestran el comportamiento del controlador PI al alcanzar la referencia de velocidad de 900 [rpm].

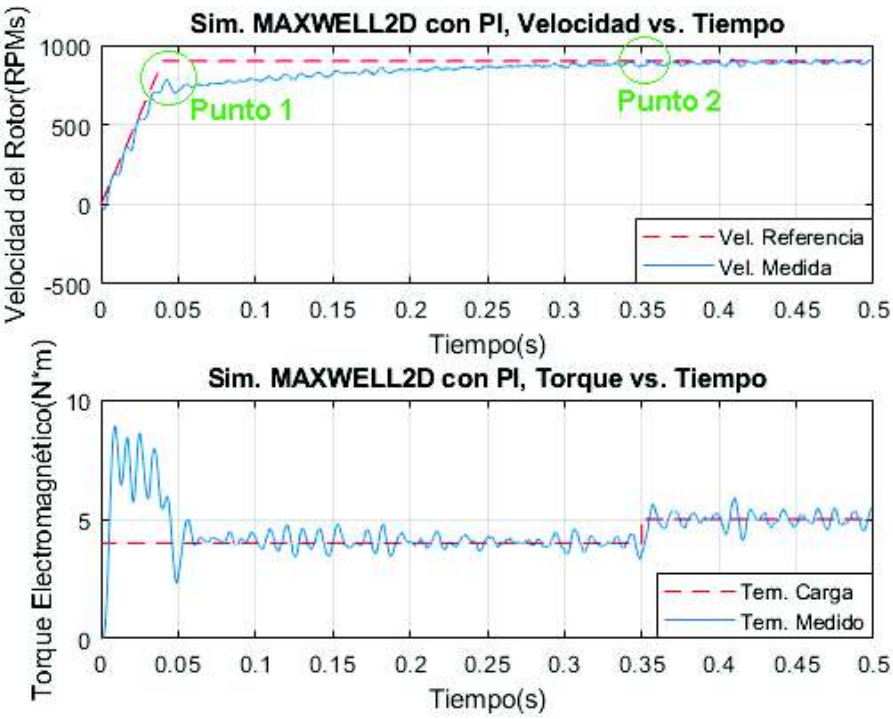


Figura 3.25 Respuesta de velocidad y torque, V_{ref}= 900 RPMs, T_L=[4 5] N*m

La figura 3.26 y 3.27 amplifica la figura 3.25 para el análisis detallado del punto 1 y punto 2 con el objetivo de determinar el valor del error en estado transitorio y estado estable de la velocidad medida respecto a la velocidad de referencia.

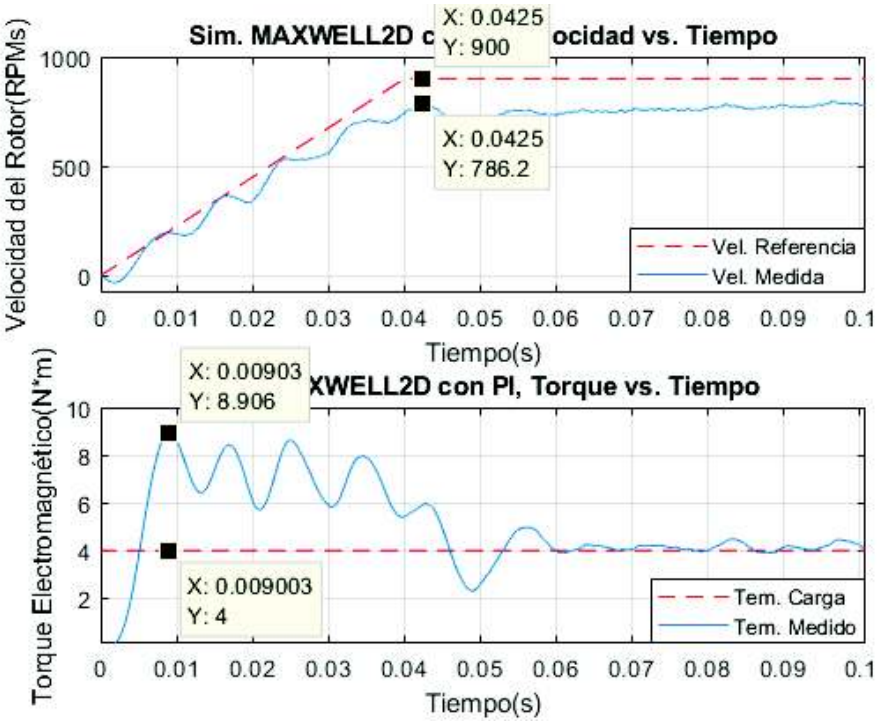


Figura 3.26 Respuesta de velocidad y torque en el punto de interés N° 1

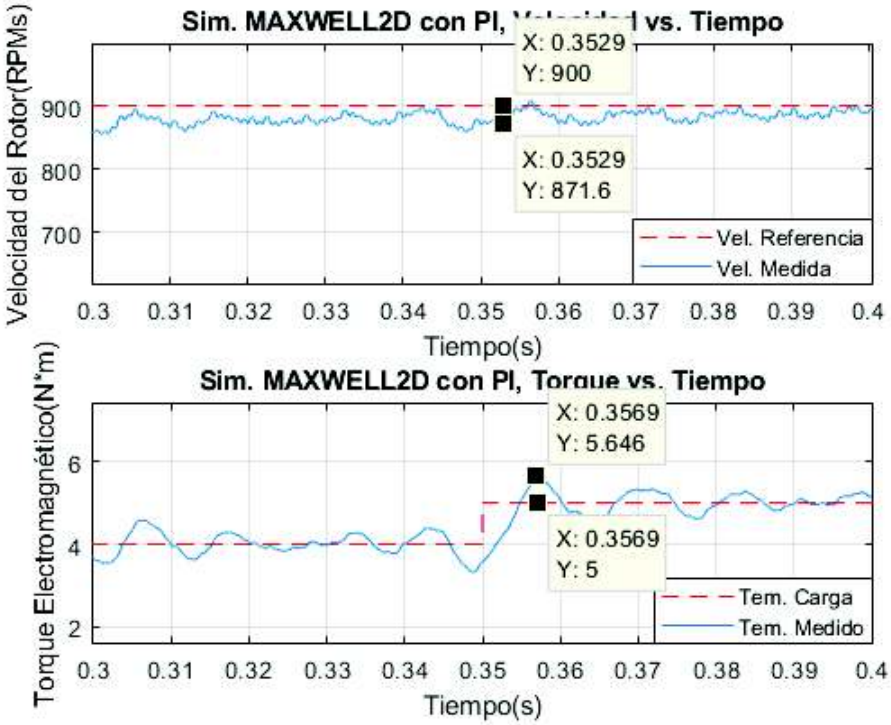


Figura 3.27 Respuesta de velocidad y torque en el punto de interés N° 2

El cálculo del error en estado transitorio generado en el punto de interés 1 y 2 se realiza mediante la ecuación 3.1. La tabla 3.11 muestra el porcentaje de error generado en estado transitorio de la velocidad medida respecto a la velocidad de referencia en el momento de arranque con T_L=4N*m y al aplicar una sobrecarga T_L=5N*m. Los errores calculados corresponden a la mejor respuesta obtenida después de realizar la sintonización del controlador PI. El torque electromagnético pico obtenido en el momento del arranque es 8.906 N*m.

Tabla 3.11 Cálculo del error en estado transitorio de la velocidad del rotor

Velocidad de referencia (rpm)	T_L (N*m)	Instante t(s)	Velocidad medida (rpm)	Error máximo (%)
900	4	0	786.2	12.64%
900	5	0.35	871.6	3.15%

El tiempo de estabilización de la velocidad luego de aplicar el torque de sobrecarga es de 0.1 segundos.

La figura 3.28 permite visualizar la respuesta del controlador al alcanzar el estado estable a la velocidad de referencia de 900 RPMs con un torque de sobrecarga aplicado T_L=5N*m. El cálculo del error en estado estable una vez transcurrido el tiempo de estabilización se realiza mediante la ecuación 3.1

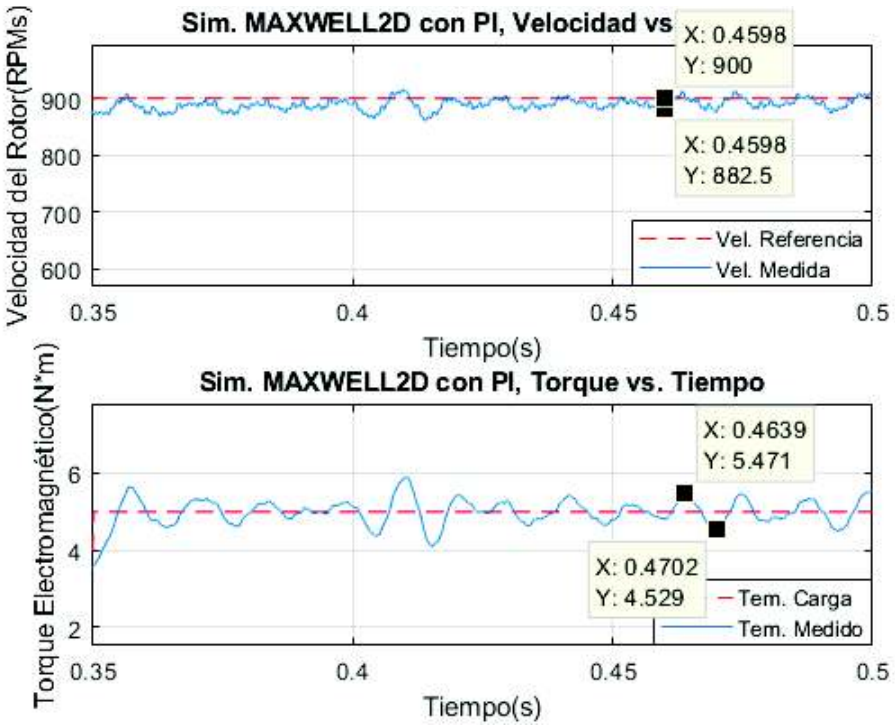


Figura 3.28 Respuesta de velocidad y torque en estado estable

La tabla 3.12 muestra el porcentaje de error generado en estado estable de la velocidad medida respecto a la velocidad de referencia. El error obtenido es 1.94% y corresponde a la mejor respuesta obtenida después de realizar la sintonización del controlador PI. El tiempo de estabilización determinado en la figura 3.28 es de 0.45 segundos.

Tabla 3.12 Cálculo del error en estado estable de la velocidad del rotor

Velocidad de referencia (rpm)	Velocidad medida (rpm)	Error máximo (%)
900	882.5	1.94%

La figura 3.28 permite también visualizar el torque electromagnético generado en estado estable el cual oscila entre 5.471 N*m y 4.529 N*m que equivale a un rizado de 0.942N*m.

3.5 Comparación de resultados de los tres modelos simulados.

A continuación, se realiza una comparación de los resultados obtenidos en los tres modelos simulados. Las gráficas de comparación de velocidades y de torque electromagnético permiten visualizar en comportamiento de cada modelo.

3.5.1 Comparación de la respuesta de velocidad a 450 RPMs, T_L=4N*m a t=0 segundos, T_L=5N*m a t=0.35 segundos de los tres modelos simulados.

La figura 3.29 permite visualizar los resultados de cada uno de los modelos implementados ante el perfil de velocidad dado por el vector de velocidad V= [450] rpm y T_L= [4 5] N*m.

La línea de color rojo corresponde a la velocidad de referencia.

La línea de color verde corresponde a la respuesta de velocidad del modelo implementado con librerías de Matlab Simulink. La línea de color magenta corresponde a la respuesta de

velocidad de co-simulación RMxprt-Simplorer-Matlab. La línea de color azul corresponde a la respuesta de velocidad de co-simulación Maxwell 2D-Simplorer-Matlab.

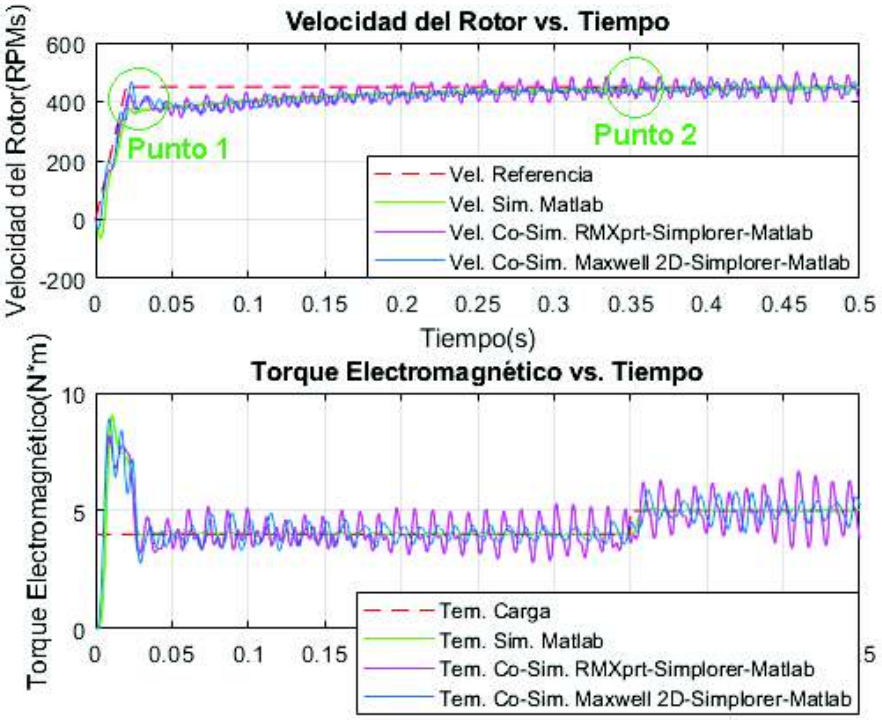


Figura 3.29 Respuestas de velocidad y torque, V_{ref}= 450 RPMs, T_L=[4 5] N*m

La figura 3.30 y 3.31 amplifica la figura 3.29 para el análisis detallado del punto 1 y punto 2 con el objetivo de determinar el valor del error en estado transitorio y estado estable de la velocidad medida respecto a la velocidad de referencia.

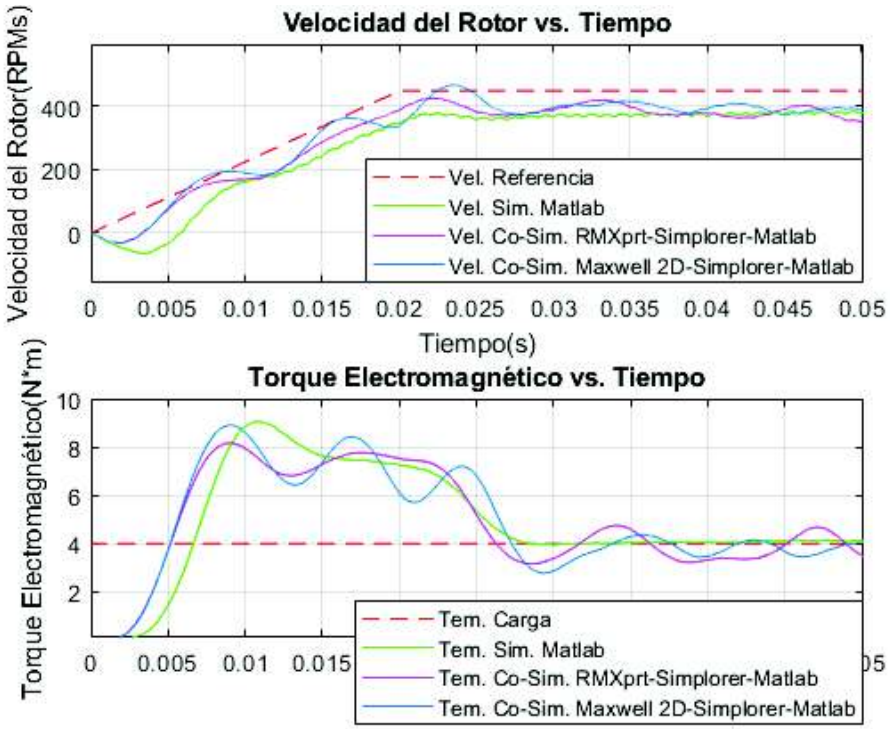


Figura 3.30 Comparación de la respuesta de velocidad y torque en el punto de interés 1

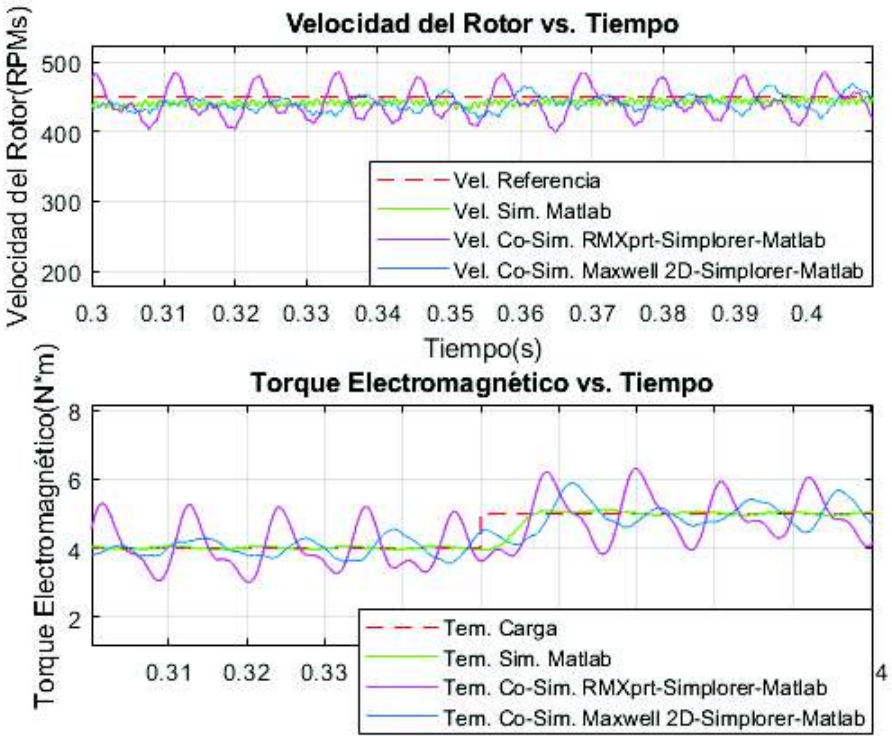


Figura 3.31 Comparación de la respuesta de velocidad y torque en el punto de interés 2

El cálculo del error en estado transitorio y estado estable generado en cada uno de los puntos de interés se realiza mediante la ecuación 3.1.

La tabla 3.13 indica el error porcentual en estado transitorio respecto a la línea de velocidad de referencia en el instante t=0.35 segundos al aplicar una sobrecarga T_L=5N*m. El error obtenido se encuentra dentro de un rango de 2% a 6% lo que significa que la respuesta de velocidad se asemeja en un alto porcentaje entre los tres modelos simulados.

Tabla 3.13 Cálculo del error en estado transitorio de la velocidad del rotor

Velocidad de referencia (rpm)	**T_L (N*m)**	**Error en estado transitorio Simulación del modelo Matlab (%)**	**Error en estado transitorio Co-simulación RMxprt-Simplorer-Matlab (%)**	**Error en estado transitorio Co-simulación Maxwell 2D-Simplorer-Matlab (%)**
450	5	4.82%	2.2%	5.75%

La tabla 3.14 indica el error porcentual en estado estable respecto a la línea de velocidad de referencia. El error obtenido se encuentra dentro de un rango de 1.5 a 2.5% lo que significa que la respuesta de velocidad se asemeja en un alto porcentaje entre los tres modelos simulados.

Tabla 3.14 Cálculo del error en estado estable de la velocidad del rotor

Velocidad de referencia (rpm)	T_L (N*m)	Error en estado estable Simulación del modelo Matlab (%)	Error en estado estable Co-simulación RMxprt-Simplorer-Matlab (%)	Error en estado estable Co-simulación Maxwell 2D-Simplorer-Matlab (%)
450	5	2.07%	2.44%	1.71%

La tabla 3.15 indica el rizado del torque electromagnético en estado estable respecto a la línea de torque de referencia. El valor de rizado obtenido de la simulación de los tres modelos se encuentra dentro de un rango de 0.128 [N*m] a 2.312 [N*m]. El rizado de menor valor corresponde a la simulación del modelo creado íntegramente con las librerías de Matlab-Simulink. Para los dos modelos restantes, el valor de rizado de torque se debe en gran parte a las características mecánicas y constructivas tomadas en cuenta en el momento del diseño del motor utilizando el software Ansys. Para mejorar esta respuesta se deberá realizar un análisis más riguroso en cuanto a las características de construcción del motor antes de ser llevado a la simulación. Este análisis se encuentra fuera del alcance del presente trabajo pero es posible realizarlo mediante las herramientas de análisis paramétrico del software Ansys Maxwell.

Tabla 3.15 Comparación del rizado de torque de los modelos implementados

Velocidad de referencia (rpm)	T_L (N*m)	Torque de rizado Simulación del modelo Matlab (N*m)	Torque de rizado Co-simulación RMxprt-Simplorer-Matlab (N*m)	Torque de rizado Co-simulación Maxwell 2D-Simplorer-Matlab (N*m)
450	5	0.128	2.312	0.889

Cabe señalar que el controlador PI de velocidad en cada uno de los modelos posee las mismas constantes proporcional e integral con el objetivo de comparar sus respuestas.

3.5.2 Comparación de la respuesta de velocidad a 900 RPMs, T_L=4N*m a t=0 segundos, T_L=5N*m a t=0.35 segundos de los tres modelos simulados.

La figura 3.32 permite visualizar los resultados de cada uno de los modelos implementados ante el perfil de velocidad dado por el vector de velocidad V= [900] rpm y T_L= [4 5] N*m.

La línea de color rojo corresponde a la velocidad de referencia. La línea de color verde corresponde a la respuesta de velocidad del modelo implementado con librerías de Matlab

Simulink. La línea de color magenta corresponde a la respuesta de velocidad de co-simulación RMxprt-Simplorer-Matlab. La línea de color azul corresponde a la respuesta de velocidad de co-simulación Maxwell 2D-Simplorer-Matlab.

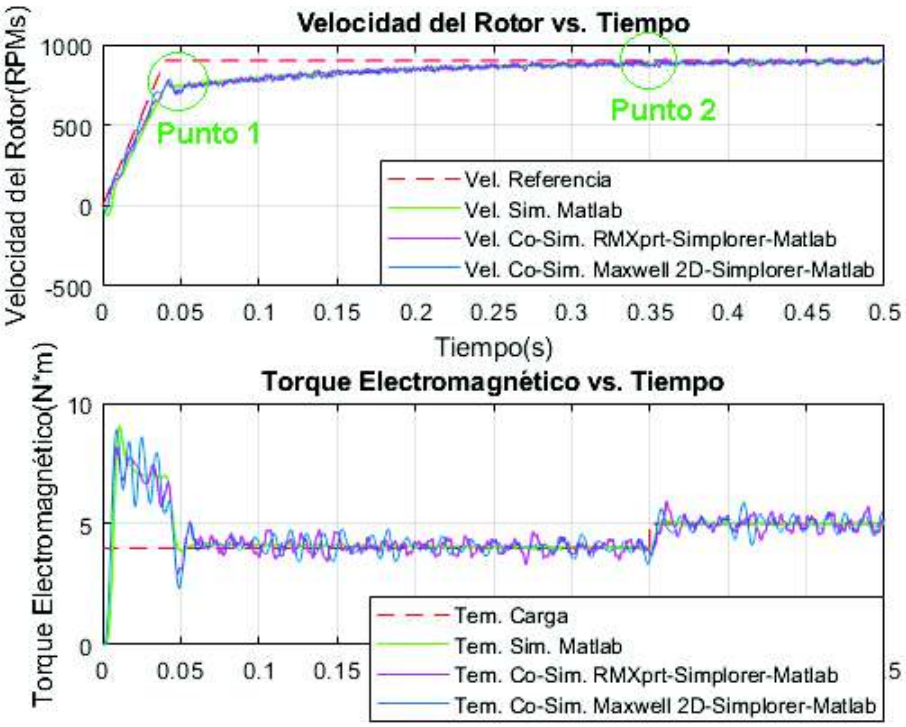


Figura 3.32 Respuestas de velocidad y torque, Vref= 900 RPMs, TL=[4 5] N*m

La figura 3.33 y 3.34 amplifica la figura 3.32 para el análisis detallado del punto 1 y punto 2 con el objetivo de determinar el valor del error en estado transitorio y estado estable de la velocidad medida respecto a la velocidad de referencia.

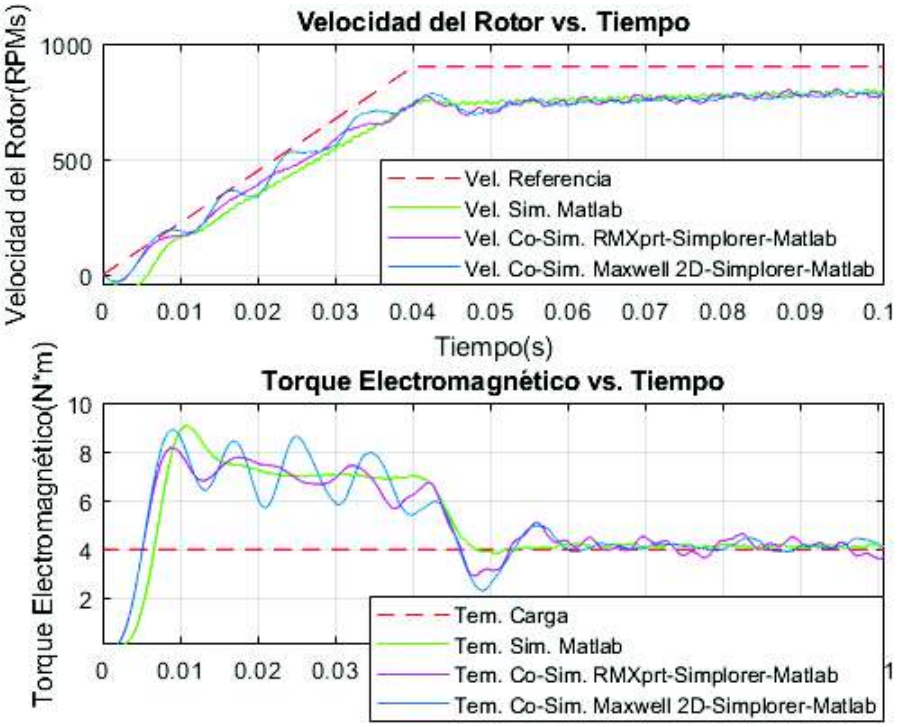


Figura 3.33 Comparación de la respuesta de velocidad y torque en el punto de interés 1

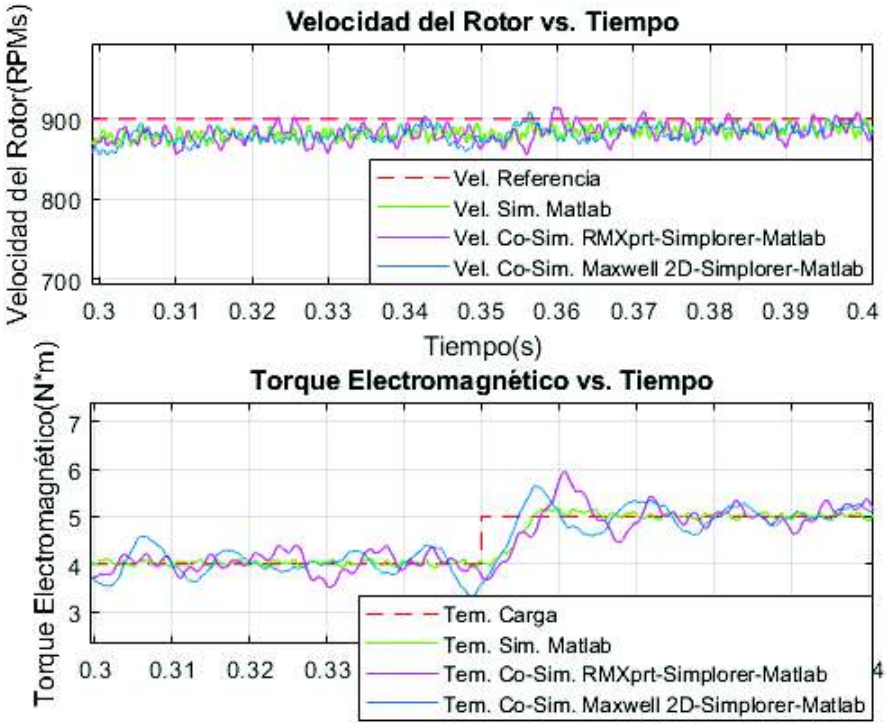


Figura 3.34 Comparación de la respuesta de velocidad y torque en el punto de interés 2

El cálculo del error en estado transitorio y estado estable generado en cada uno de los puntos de interés se realiza mediante la ecuación 3.1.

La tabla 3.16 indica el error porcentual del error en estado transitorio respecto a la línea de velocidad de referencia en el instante t=0.35 segundos al aplicar una sobrecarga T_L=5N*m. El error obtenido se encuentra dentro de un rango de 3% a 5% lo que significa que la respuesta de velocidad se asemeja en un alto porcentaje entre los tres modelos simulados.

Tabla 3.16 Cálculo del error en estado transitorio de la velocidad del rotor

Velocidad de referencia (rpm)	T_L (N*m)	Error en estado transitorio Simulación del modelo Matlab (%)	Error en estado transitorio Co-simulación RMxprt-Simplorer-Matlab (%)	Error en estado transitorio Co-simulación Maxwell 2D-Simplorer-Matlab (%)
900	5	3.73%	4.92%	3.15%

La tabla 3.17 indica el error porcentual del error en estado estable respecto a la línea de velocidad de referencia. El error obtenido se encuentra dentro de un rango de 1.1 a 2.2% lo que significa que la respuesta de velocidad se asemeja en un alto porcentaje entre los tres modelos simulados.

Tabla 3.17 Cálculo del error en estado estable de la velocidad del rotor

Velocidad de referencia (rpm)	T_L (N*m)	Error en estado estable Simulación del modelo Matlab (%)	Error en estado estable Co-simulación RMxprt-Simplorer-Matlab (%)	Error en estado estable Co-simulación Maxwell 2D-Simplorer-Matlab (%)
900	5	2.17%	1.17%	1.94%

La tabla 3.18 indica el rizado del torque electromagnético en estado estable respecto a la línea de torque de referencia. El valor de rizado obtenido de la simulación de los tres modelos se encuentra dentro de un rango de 0.152 [N*m] a 0.942 [N*m]. El rizado de menor valor corresponde a la simulación del modelo creado íntegramente con las librerías de Matlab-Simulink. Para los dos modelos restantes, el valor de rizado de torque se debe en gran parte a las características mecánicas y constructivas tomadas en cuenta en el momento del diseño del motor utilizando el software Ansys. Para mejorar esta respuesta se deberá realizar un análisis más riguroso en cuanto a las características de construcción del motor antes de ser llevado a la simulación. Este análisis se encuentra fuera del alcance del presente trabajo pero es posible realizarlo mediante las herramientas de análisis paramétrico del software Ansys Maxwell.

Tabla 3.18 Comparación del rizado de torque de los modelos implementados

Velocidad de referencia (rpm)	T_L (N*m)	Torque de rizado Simulación del modelo Matlab (N*m)	Torque de rizado Co-simulación RMxprt-Simplorer-Matlab (N*m)	Torque de rizado Co-simulación Maxwell 2D-Simplorer-Matlab (N*m)
900	5	0.152	0.736	0.942

Cabe señalar que el controlador PI de velocidad utilizado en cada uno de los modelos posee las mismas constantes proporcional e integral con el objetivo de comparar las respuestas obtenidas.

4 CONCLUSIONES

El análisis mediante la técnica de elementos finitos permite tener una mejor perspectiva del funcionamiento de un motor síncrono de imanes permanentes debido a que se integran las características no lineales que permiten tener una aproximación más real del modelo implementado en comparación a los modelos clásicos creados mediante ecuaciones diferenciales.

Una máquina de imanes permanentes requiere indispensablemente el uso de un variador de velocidad ya que no puede ser conectado directamente a una fuente de energía debido a que el campo se encuentra polarizado en todo momento y si el motor no posee protecciones puede ser gravemente afectado.

El control escalar V/F es una técnica ampliamente utilizada en el control de motores eléctricos pero no presenta resultados óptimos en el control de un motor síncrono de imanes permanentes por lo cual no es recomendable utilizarlo en este tipo de motores.

El control escalar V/F aplicado a un motor síncrono de imanes permanentes presenta una respuesta deficiente en el control de la velocidad y torque por lo cual es aconsejable utilizar una técnica de control vectorial como Control por orientación de campo (FOC).

La sintonización del controlador proporcional-integral asociado al control escalar V/F permite obtener una respuesta aceptable de velocidad y torque de un motor síncrono de imanes permanentes sin embargo el controlador provoca que el tiempo de estabilización sea de mayor valor.

Los tiempos de simulación son importantes en la sincronización de los softwares empleados lo cual refleja en el tiempo que se demora en llevar a cabo la simulación de los modelos. El establecimiento incorrecto de los tiempos provoca que se generen resultados erróneos.

La co-simulación RMxprt-Simplorer-Matlab es la más utilizada para el desarrollo de la simulación del modelo del motor de imanes permanentes debido a que el tiempo de cálculo es corto en comparación al tiempo de cálculo desarrollado con el modelo de maxwell 2D, y además es posible modificar los parámetros del motor de manera rápida para realizar nuevas simulaciones que permitirán al diseñador crear un modelo eficiente.

Para mejorar la respuesta transitoria y estable de los modelos implementados utilizado el software de análisis por elementos finitos (Ansys Maxwell), se recomienda realizar un análisis más riguroso de las características mecánicas y físicas de construcción del motor hasta obtener el modelo que presente la mejor respuesta ante la simulación.

5 REFERENCIAS BIBLIOGRÁFICAS

[1] X. Yang, J. Hudgins, and D. Patterson, Permanent magnet generator design and control for large wind turbines, presented at Power Electronics and Machines in Wind Applications (PEMWA), July 2012.

[2] Introduction to COMSOL Multiphysics, Version 5.2, 2015, [Online]. Available: http: //cdn.comsol.com/documentation/5.2.0.166/ IntroductionToCOMSOLMultiphysics.pdf

[3] Altair Hyperworks, Application Sheet Electric Field Simulation with Flux, 2018, [Online]. Available: https://hyperworks.fr/ResourceLibrary

[4] Anysy Inc. Electromagnetic Products, User´s guide Maxwell 2D, Edición: Rev 6.0, Software Version: 15, 2012.

[5] C. Apostoaia, Co-Simulation Platform for AC Drives Control Systems, 2012

[6] K. Buhr, Simulation of a Brushless DC Motor in Ansys-Maxwell 3D, Czech Technical University, Prague, 2016

[7] M. Fitouri, Y. BenSalem and M. Naceur, Analysis and Co-Simulation of Permanent Magnet Sychronous Motor with short-circuit fault by Finite Element Method, 13 th International Multi-Conference on Systems, Signal and Devices, 2016

[8] M. Ebrahimi, J. Faiz, M. Javan-Roshtkhari, Static eccentricity fault diagnosis in permanent magnet synchronous motor using time stepping finite element method. IEEE Transactions on Magnetics, 44(11), 4297–4300, 2008.

[9] A. Mishra and J. Makwana and P. Agarwal and SP. Srivastava, Modeling and implementation of vector control for PM synchronous motor drive. International Conference on Advances in Engineering, Science and Management, pp. 582-585, 2012

[10] C.M. Apostoaia, Multi-domain system models integration for faults detection in induction motor drives, IEEE International Conference on Electro/Information Technology, pp. 388–393, 2014.

[11] M. Rashid, Power Electronics Handbook, University of West Florida, Pensacola, Florida, 2001

[12] M. Aydin, Axial Flux Mounted Permanent Magnet Disk Motors For Smooth Torque Traction Drive Application, in Electrical and Computer Engineering, vol. PhD: University of Wisconsin 2004, pp. 453.

[13] M. Simón, Diseño de un motor de flujo axial de imanes permanentes, Universidad de Sevilla, Sevilla, 2017

[14] R. Krishnan, Electric Motor Drives Modeling, Analysis, and Control, Pearson Education, 2001.

[15] B. Bose, Power Electronics and Variable Frequency Drives, 1 ed: Wiley, John & Sons, 1996.

[16] Zettlex – accurate position sensors for harsh conditions, "www.zettlex.com", June 2018. [Online]. Available: https://www.zettlex.com/articles/incremental-encoders-absolute-encoders/

[17] Encoders, "www.logicbus.com", June 2018. [Online]. Available: http://www.logicbus.com.mx/info_encoders.php

[18] Freescale-Semiconductor, "AN1943 56F80x Resolver Driver and Hardware Interface," [Online]. Available: http://www.freescale.com/files/product/doc/AN1942.pdf, 2005.

[19] J. Pyrhönen, T. Jokinen, and V. Hrabovcova, Design of Rotating Electrical Machines, John Wiley & Sons, Ltd. ISBN: 978-0-470-69516-6, 2008.

[20] Ch. Perera, F. Blaabjerg, J. Pedersen, and P. Thogersen. A sensorless, stable v/f control method for permanent-magnet synchronous motor drives. IEEE Transaction on industry applications, 39(3):83-89, 2003.

[21] Ansoft ANSYS, "www.ansys.com," Ansoft, 2016. [Online]. Available: http://www.ansys.com/Products/Electronics/ANSYS-RMxprt.

[22] ANSYS, Inc, "www.scribid.com," June 2015. [Online]. Available: https://www.scribd.com/doc/129666336/RMxprt-Manual-pdf.

[23] B. Prasad, B. Ram, Inter-turn fault analysis of synchronous generator using finite element method (fem). International Journal of Innovative Technology and Exploring Engineering (IJITEE), 3(7), 170–176, 2013.

[24] Encoder products company, "www.encoder.com", June 2016. [Online]. Available:http://encoder.com/blog/encoder-basics/que-es-un-encoder/

[25] J. Aguilar, M. Olid , Electrónica de Potencia, Creative Commons, Universidad de Jaén, España, 2005.

[26] N. Quang, J. Dittrich, Vector Control of Three-Phase AC Machines, Springer, ISBN: 978-3-540-79028-0, 2008.

[27] Comparación de dispositivos electrónica de potencia, "www.dispositivoselectrnicadepotencia.blogspot.com," Marzo 2012. [Online]. Available: http://dispositivoselectrnicadepotencia.blogspot.com/

[28] Pozo M., Apuntes de clase de control de máquinas eléctricas, Quito: Escuela Politécnica Nacional, 2018

6 ANEXOS

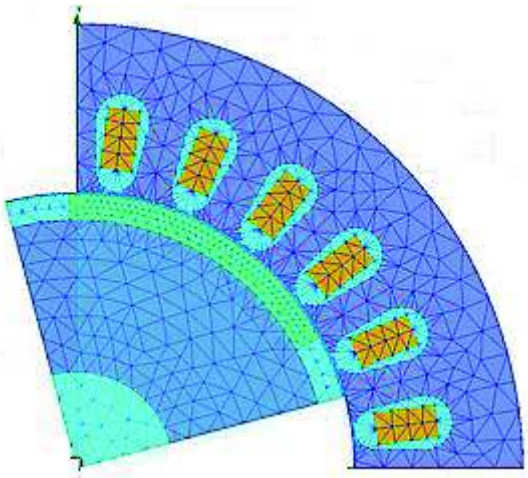
ANEXO I. Manual de usuario.

ANEXO II. Código generado en Matlab para la obtención de datos de la simulación.

ANEXO I

MANUAL DE USUARIO

CO-SIMULACIÓN PARA EL CONTROL DE UNA PMSM UTILIZANDO SOFTWARE DE ANÁLISIS MEDIANTE ELEMENTOS FINITOS



ÍNDICE

CAPITULO 0: INDICACIONES PREVIAS

CAPITULO 1: DESCRIPCIÓN
1.1 CARACTERÍSTICAS GENERALES
1.2 DICCIONARIO DE TÉRMINOS

CAPITULO 2: INSTALACIÓN DE SOFTWARES
2.1 ANSYS ELECTRONICS DESKTOP

CAPITULO 3: FUNCIONAMIENTO
3.1 SIMULACION DE LA PMSM EN MATLAB
3.2 CO-SIMULACIÓN RMxprt-SIMPLORER-MATLAB
3.3 CO-SIMULACIÓN MAXWELL 2D-SIMPLORER-MATLAB

CAPITULO 4: PROBLEMAS Y SOLUCIONES

CAPÍTULO 0 – INDICACIONES PREVIAS

El objeto del presente manual es dotar al usuario de la información necesaria para llevar a cabo la co-simulación del modelo de una máquina de imanes permanentes PMSM implementado mediante elementos finitos, circuito de potencia-control implementado en Simplorer y controlador escalar (V/F) implementado en Matlab/Simulink.

Como requisito indispensable para la simulación del sistema se requiere realizar la instalación previa del Software Matlab/Simulink versión 2014 A o superior.

Los requisitos mínimos para el correcto funcionamiento de Ansys Electronics Desktop son:

- Windows 7-64 bits o superior.
- Procesador con 4 núcleos o más.
- Tarjeta gráfica de 4GB o superior.
- Tarjeta RAM de 16GB o superior.
- Disco duro de 300 GB o superior. Preferiblemente disco de estado sólido.
- Monitor de 17 pulgadas o superior.

CAPÍTULO 1 – DESCRIPCIÓN

La co-simulación entre las herramientas de software Ansys Electronics y Matlab/Simulink descritas en el presente manual permiten visualizar el comportamiento del sistema implementado (PMSM - Circuito de potencia y control - Controlador Escalar V/F) para un perfil de velocidad y carga seteado por el usuario.

1.1 CARACTERÍSTICAS GENERALES

El modelo de la máquina síncrona de imanes permanentes se lo implementa utilizando las herramientas de diseño del software Ansys Electronics que comprende: RMxprt, Maxwell 2D y Simplorer. El modelo se crea utilizando características y dimensiones físicas, selección de materiales constructivos y condiciones de contorno para aplicar la técnica de elementos finitos que permitan visualizar un comportamiento realista del funcionamiento del motor.

1.2 DICCIONARIO DE TÉRMINOS MÁS USUALES

PMSM	Máquina síncrona de imanes permanentes.
V/F	Control escalar Voltios/Hercios.

SENSOR Elemento electrónico que permite la adquisición de variables electromecánicas.

VARIABLES Representación de una señal eléctrica.

CO-SIMULACIÓN Funcionamiento conjunto de dos o más softwares para la simulación de un sistema multifísico.

F.E.A. Análisis por elementos finitos.

CAPITULO 2 – INSTALACIÓN DE SOFTWARES

2.1 ANSYS ELECTRONICS DESKTOP

Para realizar la instalación del software Ansys Electronics Desktop se deberán seguir los pasos que se mencionan a continuación.

1. Instalar ANSYS Electromagnetics Suite 18.0 de 64 bits incluido en la imagen de disco proporcionada. Cuando solicite la información de la licencia, se debe seleccionar *" I have a new license file"*. Presionar *Next* y buscar *"ansoftd_SSQ.lic"* de la carpeta *Crack*. Seleccionar *Open* y *Next.*

2. Una vez finalizada la instalación, reemplazar la carpeta creada por defecto en *C:\Program Files\AnsysEM* por la carpeta del mismo nombre que se encuentra ubicada en el interior de la carpeta Electronics. Se debe asegurar que *"ansoftd_SSQ.lic"* es la única licencia en la carpeta *\AnsysEM\admin*. Se deberá eliminar los archivos .lic adicionales si existieran.

3. Ejecutar el archivo *"SolidSQUADLoaderEnabler.reg"* y confirmar para añadir a los registros de Windows.

4. Reiniciar la PC.

5. El software se encuentra listo para usarse.

NOTA: El software RMxprt y Simplorer viene incluidos dentro del paquete de Ansys Electronics Desktop instalado.

CAPITULO 3 – FUNCIONAMIENTO

3.1 SIMULACIÓN DE LA PMSM EN MATLAB

Para llevar a cabo la simulación del sistema implementado en Matlab se procede de la siguiente manera:

1.- Abrir el archivo *CONTROL_ESCALAR_MATLAB_2014_v0.slx* con lo cual se muestra la ventana de la figura 1.

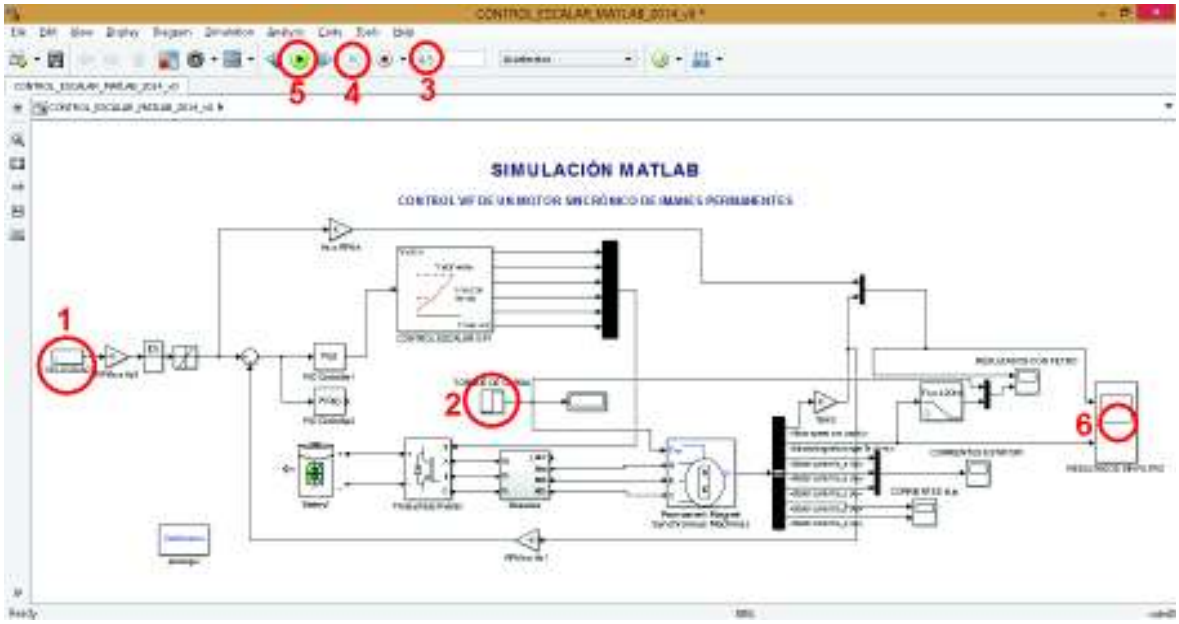


Figura 1 Ventana de simulación del modelo implementado en Matlab/Simulink.

2.- Colocar los valores de velocidad deseados y tiempos de ejecución a manera de vectores dentro del bloque marcado con el número 1 como se visualiza en la figura 2.

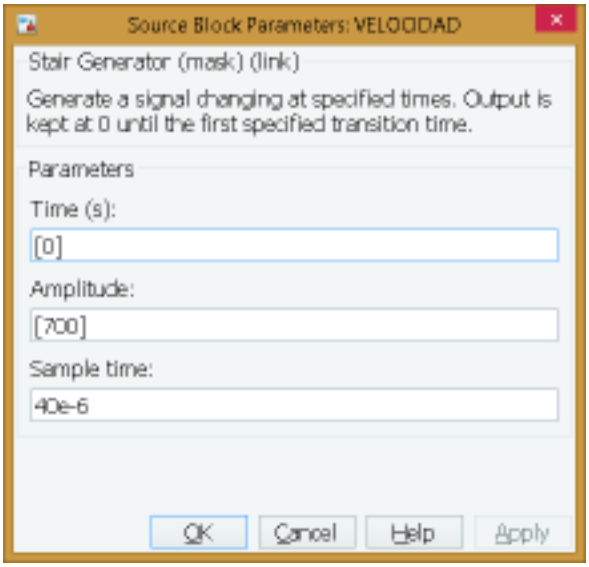


Figura 2 Ventana de configuración del perfil de velocidad en Matlab/Simulink.

3.- Colocar el torque de carga deseado a manera de vector dentro del bloque marcado con el número 2 como se visualiza en la figura 3.

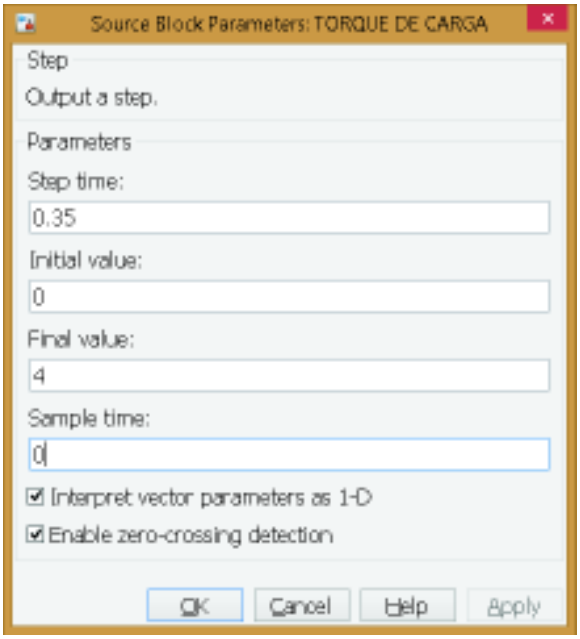


Figura 3 Ventana de configuración de torque de carga en Matlab/Simulink.

4.- Colocar el valor del tiempo total de simulación en el campo marcado con el número 3.

5.- Para empezar o detener la simulación utilizar los botones marcados con el número 5 (RUN) y 4 (STOP) respectivamente.

6.- Una vez terminada la simulación, para visualizar los resultados mediante el osciloscopio dar doble clic sobre el bloque marcado con el número 6 con lo cual se muestra la ventana de la figura 4.

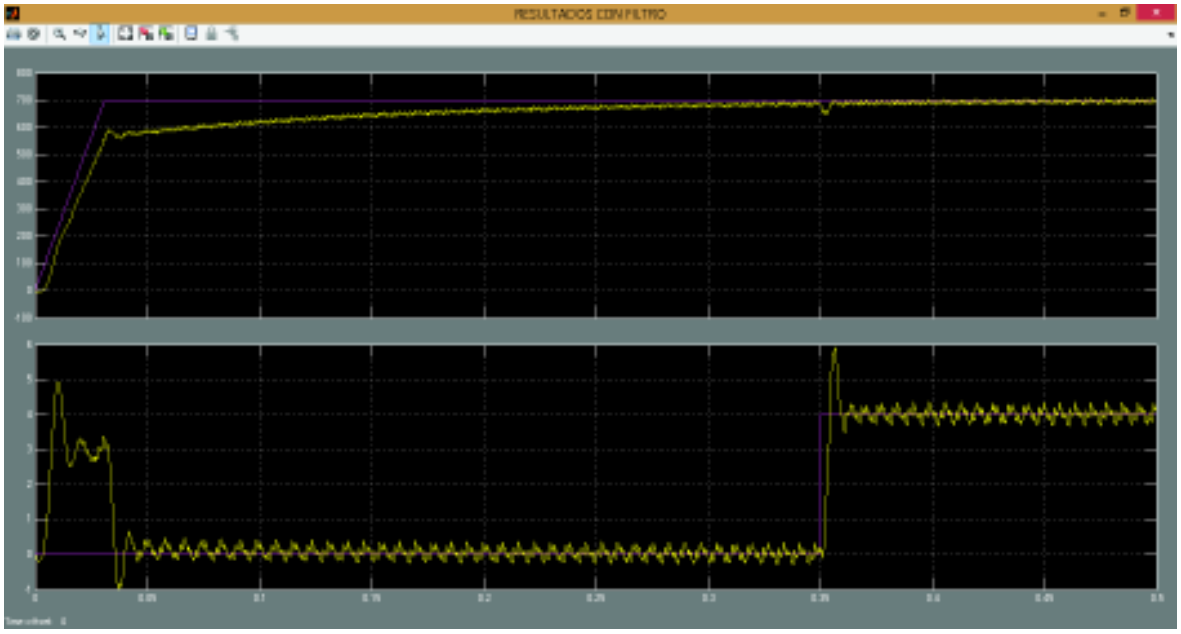


Figura 4 Ventana de visualización de resultados.

7.-En Matlab, establecer el nombre del archivo a guardar en el campo xlswrite (XXXXXXX.xlsx',data,1) del archivo EXTRACCION_*DATOS.m*.

8.- Ejecutar el archivo EXTRACCION_*DATOS.m* para descargar los datos del osciloscopio a un archivo editable en Microsoft Excel.

3.2 CO-SIMULACIÓN RMxprt-SIMPLORER-MATLAB

Para llevar a cabo la co-simulación RMXprt-Simplorer-Matlab se deberá seguir los siguientes pasos:

1.- Abrir el archivo *CONTROL_ESCALAR_RMXprt_SIMPLORER_MATLAB_v0.slx.* de Matlab/Simulink con lo cual se muestra la ventana de la figura 5.

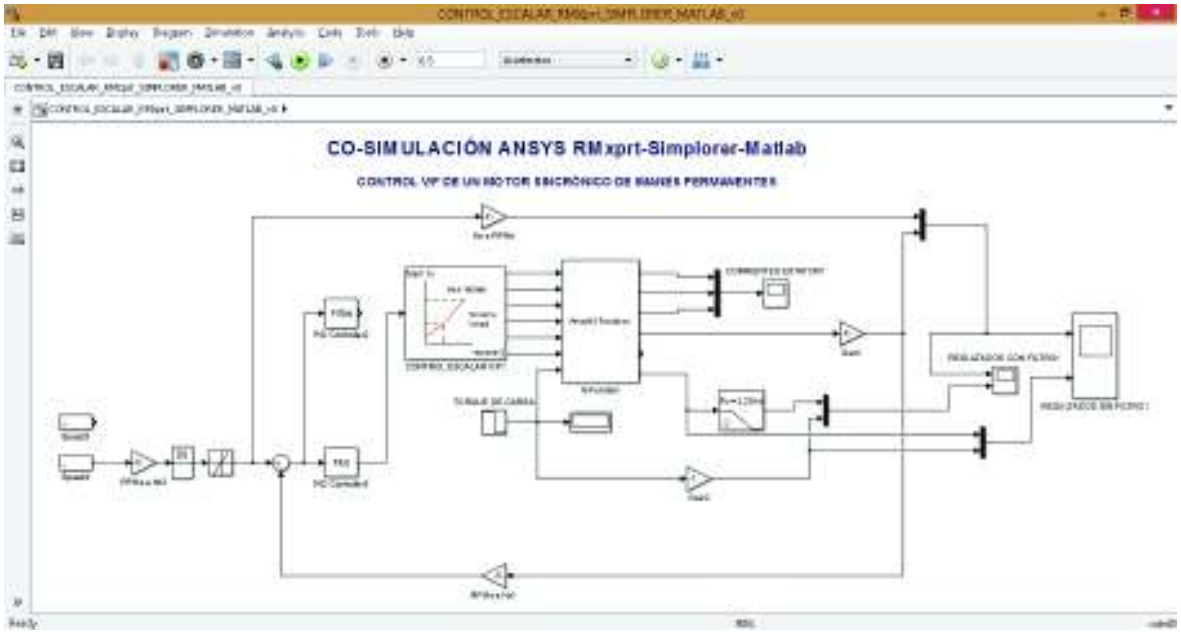


Figura 5 Ventana de simulación en Matlab del modelo de motor creado en Simplorer y RMxprt.

2.- Abrir el software Ansys electronics desktop 2017. Se presenta la ventana mostrada en la figura 6.

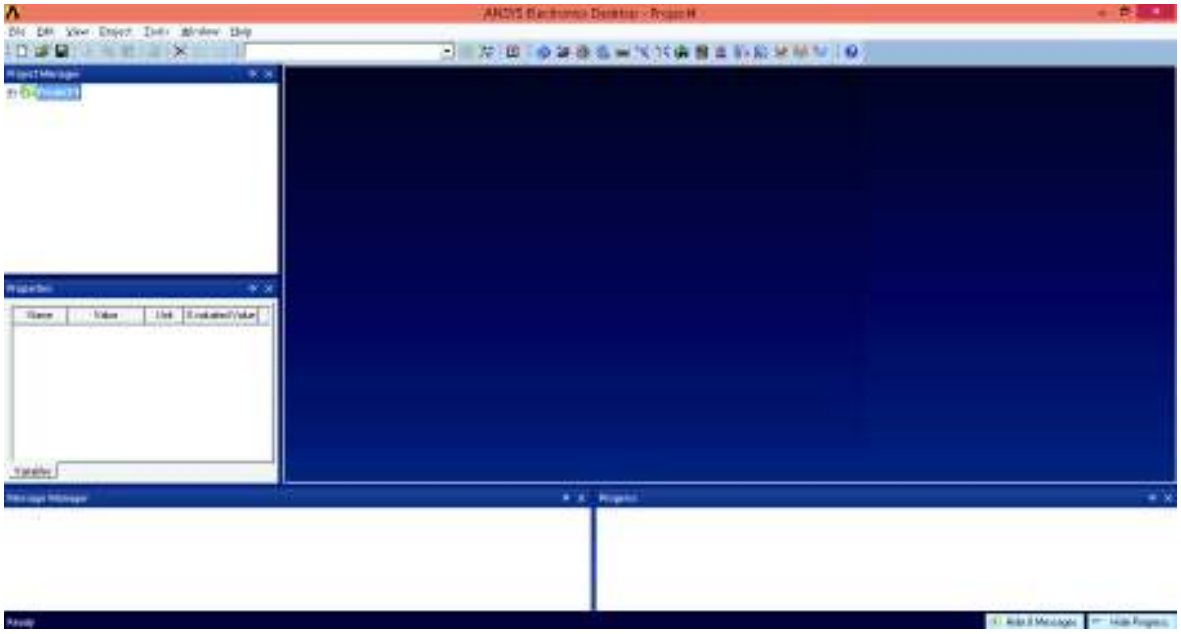


Figura 6 Ventana de Ansys electronics desktop 2017

3.- En la pestaña *file* seleccionar *open* y buscar el archivo PMSM2018. Esperar que termine de cargar todos los archivos.

4.- Seleccionar la pestaña Simplorer 1 de la ventana *Project Manager* como se visualiza en la figura 7.

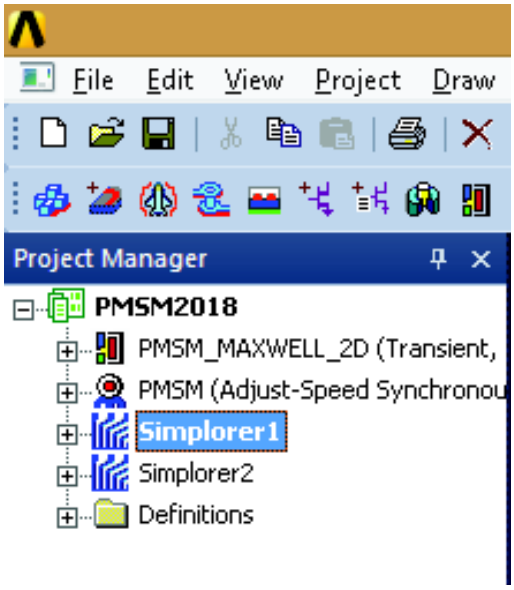


Figura 7 Ventana Project Manager.

5.- Una vez seleccionado *Simplorer 1* (Punto marcado con el número 1), se deberá establecer los valores de tiempo en los puntos marcados como 3, 4, 5 en la ventana *Properties* los cuales se despliegan al seleccionar el menú *TR* marcado con el número 2 como se visualiza en la figura 8.

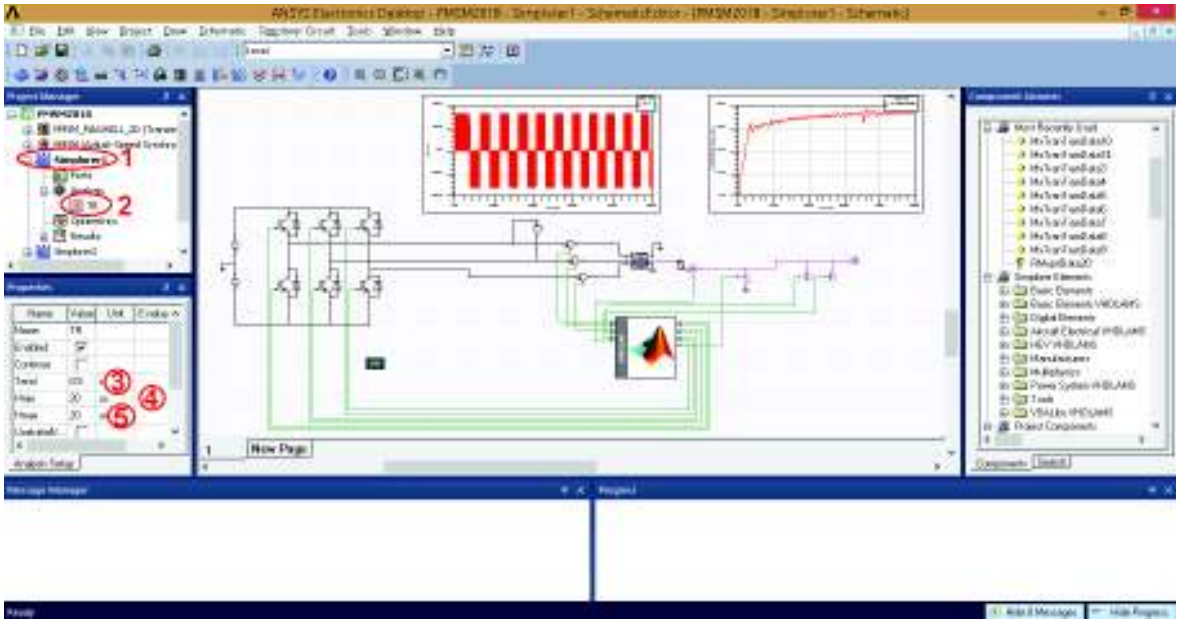


Figura 8 Ventana de Simplorer con el modelo implementado mediante RMxprt.

6.- Una vez que los dos archivos fueron abiertos, se muestra la imagen mostrada en la figura 9.

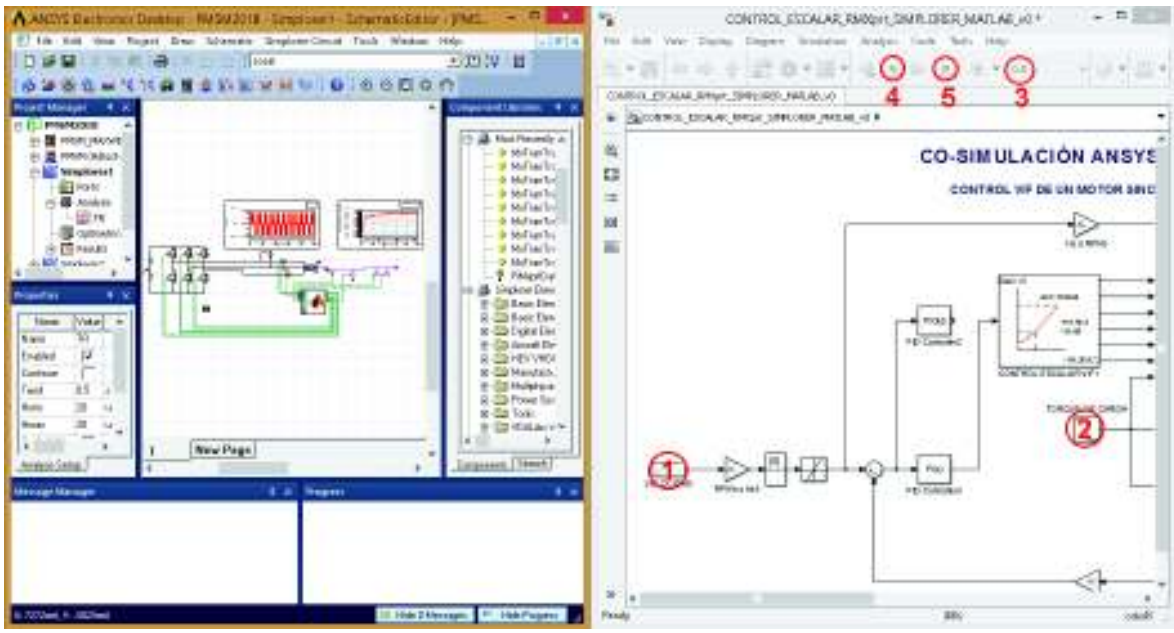


Figura 9 Ventanas de co-simulación RMxprt-Simplorer-Matlab.

7.- Tomando como referencia la figura 9, colocar los valores de velocidad deseados (perfil de velocidad) en el bloque marcado con el número 1.

8.- Colocar el torque de carga deseado en el bloque marcado con el número2.

9.- Colocar el valor del tiempo total de simulación en el campo marcado con el número 3.

10.- Para empezar o detener la simulación utilizar los botones marcados con el número 4 (RUN) y 5 (STOP) respectivamente.

11.-Una vez terminada la simulación, para visualizar los resultados mediante el osciloscopio dar doble clic sobre el bloque RESULTADOS.

12.-En Matlab, establecer el nombre del archivo a guardar en el campo xlswrite (XXXXXXX.xlsx',data,1) del archivo EXTRACCION_*DATOS.m*.

13.- Ejecutar el archivo EXTRACCION_*DATOS.m* para descargar los datos del osciloscopio a un archivo editable en Microsoft Excel.

3.3 CO-SIMULACIÓN MAXWELL 2D-SIMPLORER-MATLAB

Para llevar a cabo la co-simulación Maxwell 2D-Simplorer-Matlab se deberá seguir los siguientes pasos:

1.- Abrir el archivo *CONTROL_ESCALAR_MAXWELL2D_SIMPLORER_MATLAB_v0.slx.* de Matlab/Simulink con lo cual se despliega la ventana de la figura 10.

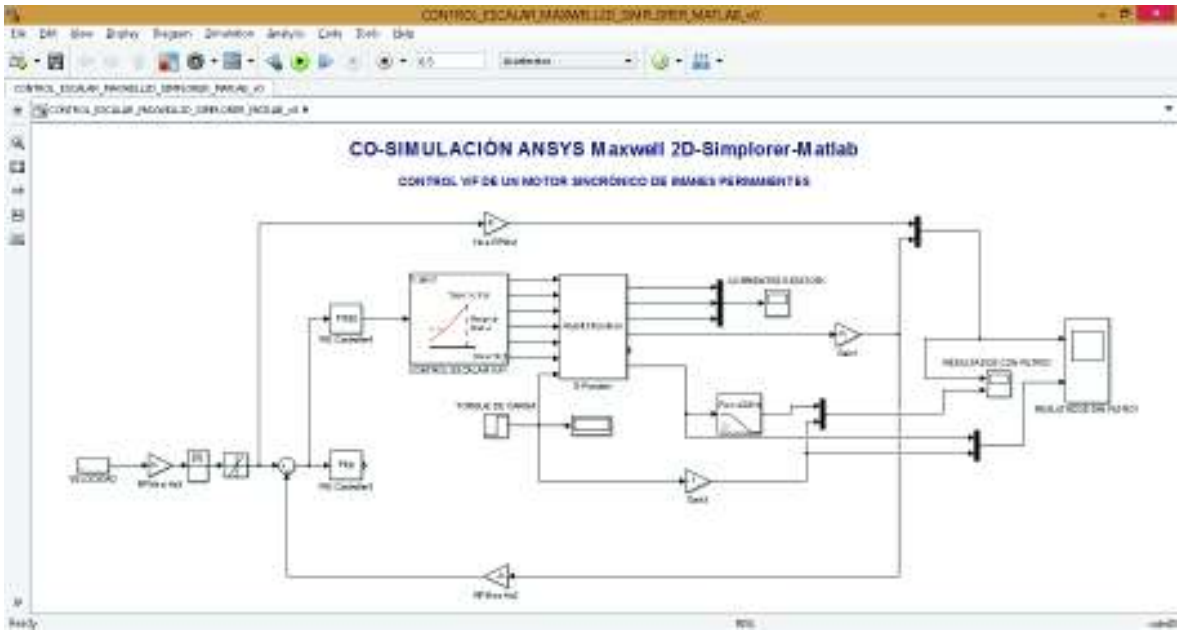


Figura 10 Ventana de simulación en Matlab del modelo de motor creado en Simplorer y Maxwell 2D.

2.- Abrir el software Ansys electronics desktop 2017. Se presenta la ventana mostrada en la figura 11.

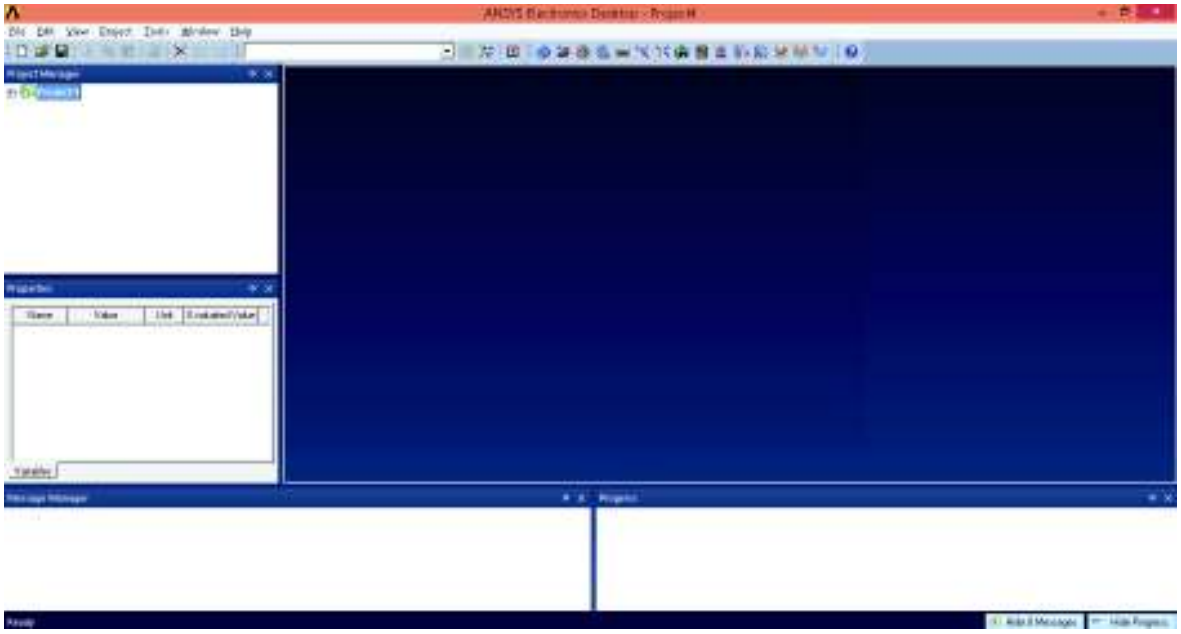


Figura 11 Ventana de Ansys electronics desktop 2017

3.- En la pestaña *file* seleccionar *open* y buscar el archivo PMSM2018. Esperar que termine de cargar todos los archivos.

4.- Seleccionar la pestaña Simplorer 2 de la ventana *Project Manager* como se visualiza en la figura 12.

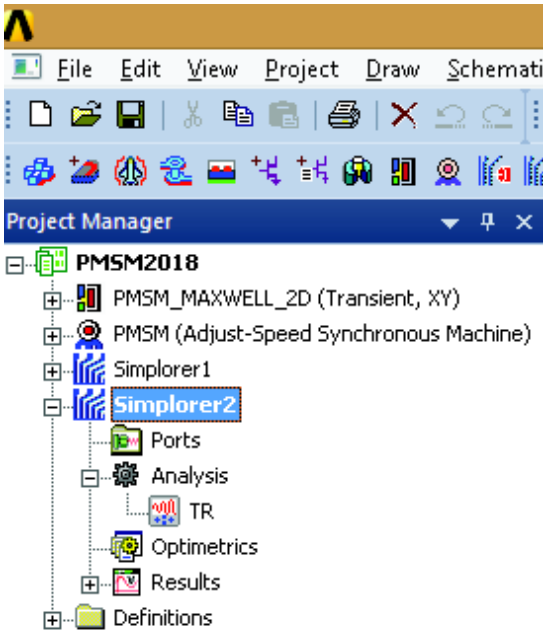


Figura 12 Ventana Project Manager.

5.- Una vez seleccionado *Simplorer 2* (Punto marcado con el número 1), se deberá establecer los valores de tiempo en los puntos marcados como 3, 4, 5 en la ventana *Properties* los cuales se despliegan al seleccionar el menú *TR* marcado con el número 2 como se visualiza en la figura 13.

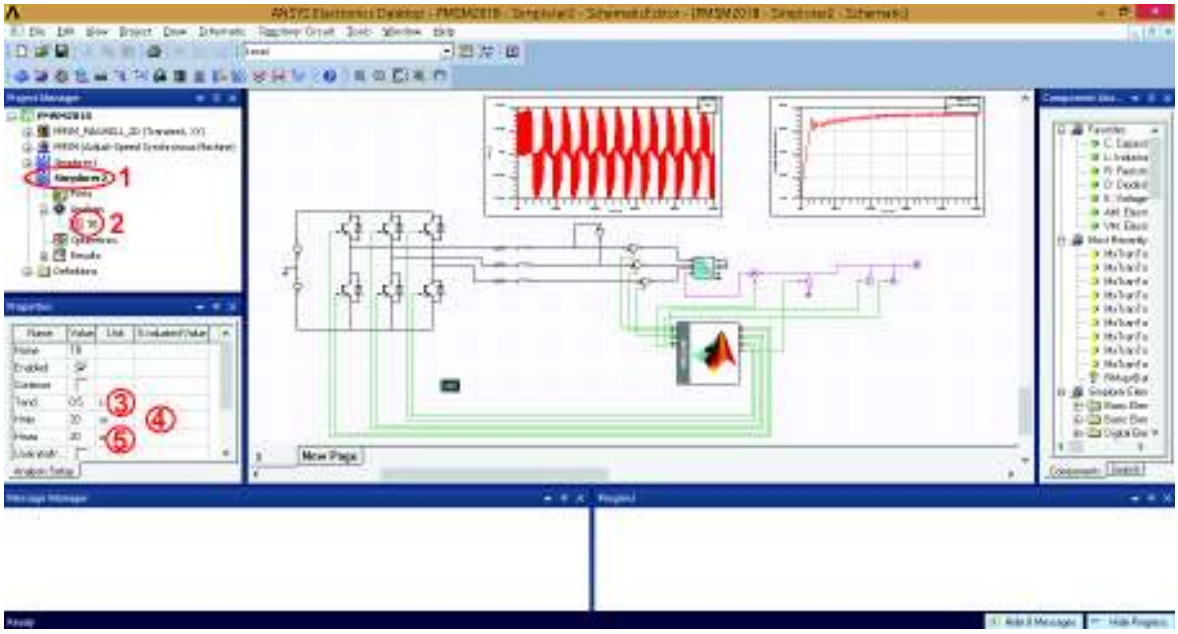


Figura 13 Ventana de Simplorer con el modelo implementado mediante Maxwell 2D.

6.- Una vez que los dos archivos fueron abiertos, se muestra la imagen mostrada en la figura 14.

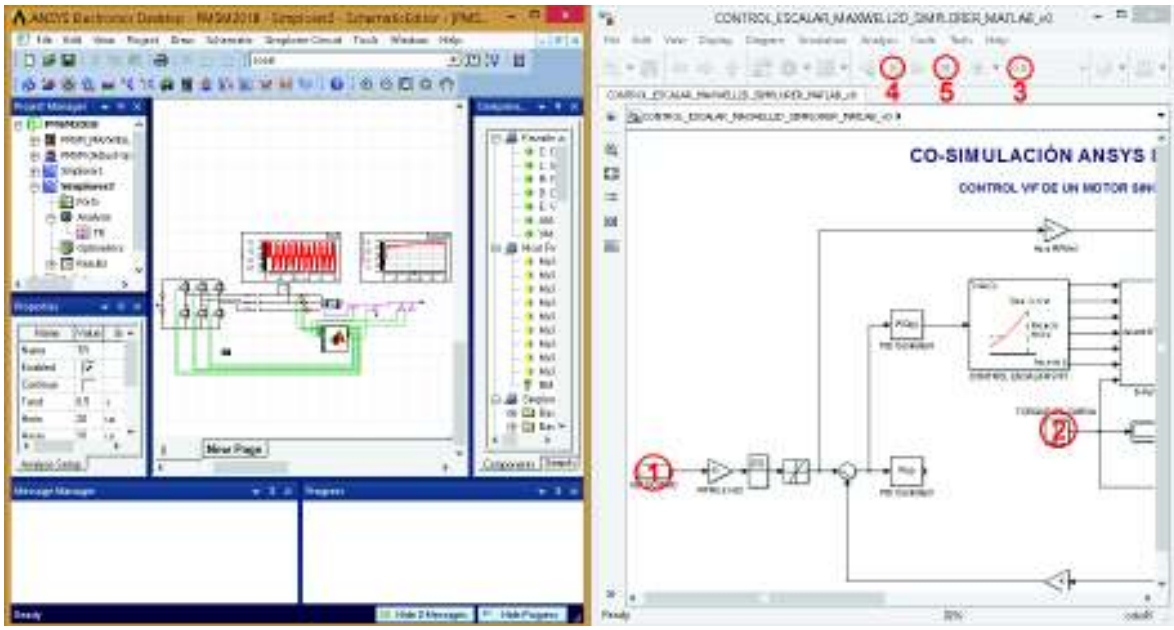


Figura 14 Ventanas de co-simulación Maxwell 2D-Simplorer-Matlab.

7.- Tomando como referencia la figura 14, establecer los valores de velocidad deseados (perfil de velocidad) en el bloque marcado con el número 1.

8.- Colocar el torque de carga deseado en el bloque marcado con el número2.

9.- Colocar el valor del tiempo total de simulación en el campo marcado con el número 3.

10.- Para empezar o detener la simulación utilizar los botones marcados con el número 4 (RUN) y 5 (STOP) respectivamente.

11.-Una vez terminada la simulación, para visualizar los resultados mediante el osciloscopio dar doble clic sobre el bloque RESULTADOS.

12.-En Matlab, establecer el nombre del archivo a guardar en el campo xlswrite (XXXXXXX.xlsx',data,1) del archivo EXTRACCION_*DATOS.m*.

13.- Ejecutar el archivo EXTRACCION_*DATOS.m* para descargar los datos del osciloscopio a un archivo editable en Microsoft Excel.

CAPITULO 4 - PROBLEMAS Y SOLUCIONES

Los problemas que se han podido encontrar en el desarrollo de una simulación se detallan a continuación.

INCIDENCIA 1 – SIMULACIÓN DETENIDA INESPERADAMENTE

POSIBLES MOTIVOS:

- Falla en la conexión de red.

REPARACIÓN:

- Detener la simulación.
- Desconectar el cable de red o módem wi-fi.
- Volver a realizar los pasos para comenzar la co-simulación.

INCIDENCIA 2 – NO ARRANCA LA SIMULACIÓN AL SER EJECUTADA DESDE MATLAB

POSIBLES MOTIVOS:

- Conflictos entre los links de co-simulación.

REPARACIÓN:

- Cerrar la ventana de los softwares Ansys electronics desktop y Matlab.
- Volver a realizar los pasos para comenzar la co-simulación.

INCIDENCIA 3 – LOS RESULTADOS OBTENIDOS SON ERRÓNEOS

POSIBLES MOTIVOS:

- Tiempos de simulación mal establecidos.

REPARACIÓN:

- Verificar y editar para establecer adecuadamente los tiempos de simulación.
- Volver a realizar los pasos para comenzar la co-simulación.

ANEXO II

El anexo II detalla el código para extraer los datos obtenidos en las simulaciones realizadas.

El almacenamiento de datos genera en un archivo de Excel.

```
%*****OBTENCIÓN DE DATOS DE LA SIMULACIÓN****************
data(:,1)=ESCALAR_MATLAB_FILTRO.time; %obtener tiempos
data(:,2)=ESCALAR_MATLAB_FILTRO.signals(1).values(:,1);%
data(:,3)=ESCALAR_MATLAB_FILTRO.signals(1).values(:,2);%
data(:,5)=ESCALAR_MATLAB_FILTRO.signals(2).values(:,1);%
data(:,6)=ESCALAR_MATLAB_FILTRO.signals(2).values(:,2);%
xlswrite('Datos_cosimulacion_MATLAB_PI_CON_CARGA.xlsx',data,1);
```

ORDEN DE EMPASTADO