

GRADO EN INGENIERÍA EN TECNOLOGÍA INDUSTRIAL

TRABAJO FIN DE GRADO

«HERRAMIENTA SOFTWARE PARA EL CÁLCULO DE GENERADORES SÍNCRONOS»

The screenshot shows a software interface titled "GENERADOR SÍNCRONO TRIFÁSICO". It is divided into several sections:

- TABLAS DE DATOS / SITUACIONES DE FUNCIONAMIENTO:** A navigation bar at the top.
- CARACTERÍSTICAS NOMINALES:** A form with input fields for:
 - Sn (kVA): 0
 - Frecuencia (Hz): 0
 - Un (kV): 0
 - n (rpm): 0
 - In (A): 0
 - Pares de Polos: 0
 - Valores máximos de Diseño: Ie_Máx (A): 0, n_Máx (rpm): 0, Ie_N (A): 0A "Confirmar Características Nominales" button is located below the fields.
- TABLA DE ENSAYOS:** A form with input fields for:
 - ENSAYO DE VACIO: Uo (kV): 0, Ie_o (A): 0, n_o (rpm): 0
 - ENSAYO DE CORTOCIRCUITO: Icc (A): 0, Ie_cc (A): 0, n_cc (rpm): 0A "Confirmar Datos Ensayos" button is located below the fields.
- CÁLCULO DE ENSAYOS:** A form with input fields for:
 - ENSAYO DE VACIO: f_vacio (Hz): 0, E0 (kV): 0
 - ENSAYO DE CORTOCIRCUITO: f_cc (Hz): 0, Xs (Ω): 0, Ls (H): 0
- Gráfico:** A graph titled "Curva característica de ensayo de vacío" showing E0 (kV) on the y-axis (0 to 1) and Ie (A) on the x-axis (0 to 1). A checkbox "Marcar situación de ensayo de vacío" is checked.



Estudiante: «Gondra Crespo, Jon»

Director/Directora: «Valverde Santiago, Víctor»

Curso: «2022-2023»

Fecha: «Bilbao, 12, septiembre, 2023»

INFORMACIÓN GENERAL REFERENTE AL TFG

- Alumno: Jon Gondra Crespo.
- Director: Valverde Santiago, Víctor.
- Departamento: Ingeniería Eléctrica

- Título del Trabajo Fin de Grado: Herramienta software para el cálculo de generadores síncronos.
- Resumen: El objetivo principal de este TFG es el desarrollo de una interfaz gráfica de usuario que nos permita calcular diferentes parámetros de funcionamiento del generador síncrono en diferentes casuísticas de funcionamiento: Vacío, cortocircuito y carga. Para todo ello, se ha hecho uso de la herramienta *APPDESIGNER*, que proporciona *MATLAB*, con el fin de realizar dicha aplicación. Además, se ha incorporado un estado del arte sobre los generadores síncronos.
- Palabras clave: generador síncrono, situación de funcionamiento, vacío, cortocircuito, carga, interfaz gráfica de usuario.

- Izenburua: Sorgailu sinkronoen kalkuluak egiteko erreminta softwarea.
- Laburpena: Lan honen helburu nagusia, sorgailu sinkronoaren huts, zirkuitu labur eta karga funtzionamendu kasuetan, parametro desberdinak kalkulatzeko interfaze grafikoa garatzea da. Horretarako, *MATLAB* hornitzen duen *APPDESIGNER* erreminta erabili da, aplikazioa egitearen helburuarekin. Gainera, sorgailu sinkronoen egungo egoeraren azterketa egingo da.
- Hitzgakoak: sorgailu sinkronoa, funtzionamendu egoera, huts, zirkulu labur, karga, erabiltzailearen interfaze grafikoa.

- Title: Software tool for synchronous generator calculations.
- Abstract: The main aim of this project is to develop a user graphic interface that allows us to calculate different parameters of the synchronous generator during empty, short-circuit and load functioning situations. In order to accomplish this application, a *MATLAB* tool called *APPDESIGNER* it is been used. In addition, a state-of-the-art of the synchronous generator will be made.
- Keywords: synchronous generator, functioning situation, empty, short-circuit, load, user graphic interface.

ÍNDICE

1	INTRODUCCIÓN	10
2	CONTEXTO	12
2.1	Sistema eléctrico de potencia	12
2.2	Clasificación de las máquinas eléctricas	14
2.3	Máquinas síncronas	15
2.4	Generador síncrono	17
3	OBJETIVOS Y ALCANCE	19
4	BENEFICIOS DEL PROYECTO	20
4.1	Beneficios técnicos	20
4.2	Beneficios económicos	21
4.3	Beneficios sociales	21
5	ESTADO DEL ARTE	22
5.1	Elementos constitutivos de una máquina síncrona	22
5.1.1	Partes Activas	23
5.1.2	Sistemas auxiliares	26
5.2	Principios de funcionamiento del generador síncrono	28
5.2.1	Situaciones de funcionamiento del Generador Síncrono	29
5.3	Regulación y control en sistema de red aislada	34
6	ANÁLISIS DE ALTERNATIVAS	36
6.1	Análisis de alternativas para el estudio del generador síncrono	36
6.1.1	Pruebas de campo	36
6.1.2	Modelos físicos a escala	36
6.1.3	Simulación software	37
6.1.4	Criterios de selección	37
6.1.5	Elección de la solución	38
6.2	Análisis para la elección de la herramienta software	39
6.2.1	Microsoft Excel	39
6.2.2	PTC Mathcad	39
6.2.3	Microsoft Visual Estudio	40
6.2.4	Matlab	40
6.2.5	Criterios de selección	41
6.2.6	Elección de la herramienta software	41
7	DESCRIPCIÓN DE LA SOLUCIÓN	44

7.1	Introducción a <i>Appdesigner</i>.....	44
7.2	Interfaz gráfica de usuario	46
7.2.1	Módulo 1: Datos nominales del generador síncrono	50
7.2.2	Módulo 2: Ensayos de la máquina síncrona.....	52
7.2.3	Módulo 3: Situaciones de funcionamiento.....	57
7.2.4	Mensajes de advertencia y error.....	68
8	EJEMPLOS DE APLICACIÓN.....	71
8.1	Ejemplo 1.....	71
8.2	Ejemplo 2.....	77
9	DESCRIPCIÓN DE LAS TAREAS.DIAGRAMA DE GANTT.....	82
10	PRESUPUESTO	84
11	CONCLUSIONES.....	86
12	REFERENCIAS.....	87
13	ANEXO.CÓDIGO COMPLETO DE LA INTERFAZ	88

ÍNDICE DE FIGURAS

FIGURA 1: ESQUEMA DE LAS PARTES FUNDAMENTALES DE UN SISTEMA ELÉCTRICO DE POTENCIA [1].	12
FIGURA 2: ESQUEMA GRÁFICO DEL SISTEMA ELÉCTRICO DE POTENCIA [8].	13
FIGURA 3: CLASIFICACIÓN DE MÁQUINAS ROTATIVAS [1].	14
FIGURA 4: LA MÁQUINA ELÉCTRICA COMO CONVERTIDOR DE ENERGÍA [2].	15
FIGURA 5: TIPO DE MÁQUINAS SÍNCRONAS [2].	16
FIGURA 6: PRESA DE ITAIPU - RÍO PARANÁ - POTENCIA INSTALADA DE 14 GW - 20 GENERADORES DE 18KV [10].	17
FIGURA 7: GENERADOR TRIFÁSICO DE 60 KVA EMPLEADO EN EL CONCORDE PARA SUMINISTRAR ENERGÍA ELÉCTRICA, APROVECHANDO PARTE DE LA POTENCIA DESARROLLADA POR UNO DE LOS TURBORREACTORES DEL AVIÓN [3].	17
FIGURA 8: ESQUEMA ELÉCTRICO Y MAGNÉTICO DE UNA MÁQUINA ROTATIVA [1].	22
FIGURA 9: ROTOR DE POLOS SALIENTES (HIDROALTERNADOR DE LA CENTRAL DE ITAIPÚ) [9].	23
FIGURA 10: ROTOR CILÍNDRICO [9].	24
FIGURA 11: ESTATOR DE UN GENERADOR SÍNCRONO (CENTRAL NUCLEAR NÓRDICA) [11].	24
FIGURA 12: DEVANADO DEL ESTATOR [9].	25
FIGURA 13: DEVANADOS DEL ROTOR CON BOBINADO AMORTIGUADOR [12].	25
FIGURA 14: SISTEMA DE EXCITACIÓN DE CORRIENTE CONTINUA [14].	26
FIGURA 15: SISTEMA DE EXCITACIÓN SIN ESCOBILLAS (BRUSHLESS) [13].	26
FIGURA 16: SISTEMA DE EXCITACIÓN ESTÁTICO O AUTOEXCITADO [13].	27
FIGURA 17: ESQUEMA MÁQUINA SÍNCRONA [13].	28
FIGURA 18: CURVA CARACTERÍSTICA DE VACÍO [1].	30
FIGURA 19: CURVA CARACTERÍSTICA DE CORTOCIRCUITO [1].	31
FIGURA 20: ESQUEMA MONOFÁSICO EQUIVALENTE EN CORTOCIRCUITO.	31
FIGURA 21: TIPOS DE ROTORES EN GENERADORES SÍNCRONOS [1].	32
FIGURA 22: DIAGRAMA VECTORIAL.	32
FIGURA 23: CIRCUITO EQUIVALENTE EN CARGA.	33
FIGURA 24: POSICIONAMIENTO DE LA REACCIÓN DE INDUCIDO [1].	34
FIGURA 25: CURVAS DE CARGA DEL GENERADOR [13].	34
FIGURA 26: SISTEMAS DE REGULACIÓN Y CONTROL [13].	35
FIGURA 27: ENTORNO SOFTWARE DE APPDESIGNER.	44
FIGURA 28: INTERFAZ. TABLA DE DATOS. RECOGIDA DE DATOS Y CÁLCULO DE PARÁMETROS DE LA MÁQUINA.	46
FIGURA 29: INTERFAZ. SITUACIONES DE FUNCIONAMIENTO. SIMULACIÓN.	47
FIGURA 30: DIAGRAMA DE FLUJO DE LA INTERFAZ.	49

FIGURA 31: MÓDULO 1. CARACTERÍSTICAS NOMINALES DEL GENERADOR.	50
FIGURA 32: VALORES MÁXIMOS DE DISEÑO.	50
FIGURA 33: ESQUEMA MÓDULO 1.	52
FIGURA 34: MÓDULO 2. RECOGIDA Y MUESTRA DE DATOS DE ENSAYOS DE LA MÁQUINA SÍNCRONA.	52
FIGURA 35: MÓDULO 2. GRÁFICA DE ENSAYO DE VACÍO.	53
FIGURA 36: ESQUEMA MÓDULO 2.	53
FIGURA 37: EJEMPLO GRÁFICA DEL ENSAYO DE VACÍO. CASO 1 Y 2.	56
FIGURA 38: MÓDULO 3. ÁREA DE SIMULACIÓN DE LA INTERFAZ GRÁFICA.	57
FIGURA 39: PARÁMETROS NOMINALES DE LAS SITUACIONES DE FUNCIONAMIENTO.	57
FIGURA 40: RADIO BUTTON GROUP. SITUACIONES DE FUNCIONAMIENTO.	58
FIGURA 41: SITUACIÓN DE VACÍO. INTERFAZ.	59
FIGURA 42: SITUACIÓN DE CORTOCIRCUITO. INTERFAZ.	59
FIGURA 43: SITUACIÓN DE CARGA. INTERFAZ.	59
FIGURA 44: PANEL. INTRODUCIR DATOS DE LA CARGA.	60
FIGURA 45: DROP DOWN MENU. TIPO DE CARGA.	60
FIGURA 46: OPCIONES DE ENTRADA DE PARÁMETROS DE REGULACIÓN.	60
FIGURA 47: PARÁMETROS DE REGULACIÓN. MANUAL.	61
FIGURA 48: PARÁMETROS DE REGULACIÓN. SELECTOR.	61
FIGURA 49: PANEL DE PUESTA EN MARCHA.	62
FIGURA 50: PUESTA EN MARCHA MANUAL. ACTIVADO.	62
FIGURA 51: PUESTA EN MARCHA AUTOMÁTICO. ACTIVADO.	63
FIGURA 52: PANEL DE SITUACIÓN DE VACÍO. SIMULACIÓN: ENSAYO DE VACÍO.	64
FIGURA 53: PANEL DE SITUACIÓN DE CORTOCIRCUITO. SIMULACIÓN: ENSAYO DE CORTO.	65
FIGURA 54: PANEL DE SITUACIÓN DE CARGA. SIMULACIÓN: ENSAYO DE CARGA.	68
FIGURA 55: ESQUEMA MÓDULO 3.	68
FIGURA 56: EJEMPLO. VALORES MÁXIMOS DE DISEÑO.	69
FIGURA 57: MENSAJE DE ADVERTENCIA. CASO 1.	69
FIGURA 58: MENSAJE DE ADVERTENCIA. CASO 2.	69
FIGURA 59: EJEMPLO DE APLICACIÓN 1. INTRODUCCIÓN DE DATOS NOMINALES.	72
FIGURA 60: EJEMPLO DE APLICACIÓN 1. INTRODUCCIÓN DE DATOS DE ENSAYO DE VACÍO.	72
FIGURA 61: EJEMPLO DE APLICACIÓN 1. PENDIENTE DE ENSAYO DE VACÍO.	73
FIGURA 62: EJEMPLO DE APLICACIÓN 1. CÁLCULOS DEL ENSAYO DE VACÍO.	73
FIGURA 63: EJEMPLO DE APLICACIÓN 1. INTRODUCCIÓN DE DATOS DE ENSAYO DE CORTOCIRCUITO.	74
FIGURA 64: EJEMPLO DE APLICACIÓN 1. CÁLCULO DEL ENSAYO DE CORTOCIRCUITO.	74
FIGURA 65: EJEMPLO DE APLICACIÓN 1. INTRODUCCIÓN DE DATOS DE REGULACIÓN.	75
FIGURA 66: EJEMPLO DE APLICACIÓN 1. SIMULACIÓN. CÁLCULO DE LA SITUACIÓN DE VACÍO.	75
FIGURA 67: EJEMPLO DE APLICACIÓN 1. SIMULACIÓN. CÁLCULO DE LA SITUACIÓN DE CORTOCIRCUITO.	76

FIGURA 68: EJEMPLO DE APLICACIÓN 2. INTRODUCCIÓN DE DATOS NOMINALES.	77
FIGURA 69: EJEMPLO DE APLICACIÓN 2. INTRODUCCIÓN DE DATOS DE ENSAYOS.	78
FIGURA 70: EJEMPLO DE APLICACIÓN 2. CÁLCULO DE ENSAYOS.	78
FIGURA 71: EJEMPLO DE APLICACIÓN 2. INTRODUCCIÓN DE PARÁMETROS DE REGULACIÓN Y DATOS DE LA CARGA.	80
FIGURA 72: EJEMPLO DE APLICACIÓN 2. SIMULACIÓN. CÁLCULO DE LA SITUACIÓN DE CARGA.	80
FIGURA 73: MENSAJE DE ADVERTENCIA. INTENSIDAD DE EXCITACIÓN SOBREPASADA.	80
FIGURA 74: MENSAJE DE ADVERTENCIA. VELOCIDAD DE GIRO SOBREPASADA.	81
FIGURA 75: DIAGRAMA DE GANTT.	83

ÍNDICE DE TABLAS

TABLA 1: MATRIZ DE PRIORIZACIÓN DE ALTERNATIVAS.	38
TABLA 2: MATRIZ DE PRIORIZACIÓN DE HERRAMIENTAS SOFTWARE.	42
TABLA 3: PARTIDA DE HORAS INTERNAS.	84
TABLA 4: PARTIDA DE AMORTIZACIONES.	84
TABLA 5: PARTIDA DE GASTOS.	84
TABLA 6: RESUMEN DE PRESUPUESTO.	85

1 INTRODUCCIÓN

En este presente documento se establece el desarrollo del Trabajo de Fin de Grado (TFG) titulado "Herramienta software para el cálculo de generadores síncronos".

Para que se pueda comprender de una manera más adecuada los conceptos de diseño de la interfaz gráfica, se realizará un contexto previo del funcionamiento y elementos que constituyen el sistema eléctrico de potencia. Además de ello, se comentarán otros apartados del proyecto esenciales como son los objetivos, alcance y benéficos (técnicos, sociales y económicos) del mismo, que se pretenden alcanzar mediante su realización.

Posteriormente, se llevará a cabo el estado del arte del proyecto, en el cual, se estudiará en detalle las características de la máquina síncrona y su funcionamiento a través de su respectivo circuito equivalente. Para ello, se analizarán diferentes situaciones de funcionamiento en las que puede trabajar y las curvas características asociadas a dichas situaciones. Se expondrán además las fórmulas que rigen su comportamiento, que serán utilizadas posteriormente en el desarrollo de la aplicación gráfica.

En cuanto al análisis de alternativas, se realizará un análisis de los diferentes softwares disponibles en el mercado para la realización de la interfaz gráfica, para los cuales se valorarán las ventajas y desventajas de cada uno en base a unos parámetros que se comentarán en su desarrollo posterior. Finalmente, se llegará a la elección del software más favorable para aplicación de la herramienta gráfica.

El objetivo principal es el desarrollo de una interfaz gráfica que nos permita calcular diferentes parámetros eléctricos de la máquina en diferentes situaciones de funcionamiento, además de mostrar la gráfica de la curva característica de vacío de la máquina. Todo ello, se explicará en detalle, aclarando como se ha realizado la programación de la herramienta, que datos de entrada son necesarios, como se han realizado los cálculos y que datos de salida se han mostrado al usuario. Para terminar con este punto, en el apartado de ejemplos de aplicación se emplearán algunos ejemplos, por medio de los cuales se podrá observar de una manera más práctica y visual el funcionamiento de la interfaz.

El presente documento también dispone de un apartado de presupuesto en el que se detallan los gastos del proyecto, ya sean, amortizaciones de licencias de software, o salarios de los ingenieros implicados en el mismo.

Por último, en el apartado de conclusiones, se analizan los resultados obtenidos tras la realización del estudio de los generadores síncronos y de la herramienta de cálculo, para posteriormente hacer una interpretación de los resultados, y realizar las respectivas conclusiones. Estas incluyen un resumen de la información tratada y el proceso de realización del proyecto.

2 CONTEXTO

Para una mejor comprensión de la herramienta gráfica, se realizará una breve descripción del sistema eléctrico de potencia, en el cual, se diferenciarán las partes clave de este sistema, desde que se produce la energía, hasta que se consume. Haciendo énfasis especialmente en el papel que juega la máquina síncrona en este sistema.

2.1 Sistema eléctrico de potencia

Se conoce con el nombre de Sistema Eléctrico de Potencia al conjunto de equipos e instalaciones que posibilitan el uso de la energía eléctrica a gran escala. Las partes fundamentales del Sistema Eléctrico de Potencia son la generación, el transporte, la distribución y el consumo (Figura 1). Aunque cada parte tiene sus propios niveles de tensión, todas ellas constituyen un sistema trifásico a frecuencia constante (50 Hz en Europa y 60 Hz en América) [1].

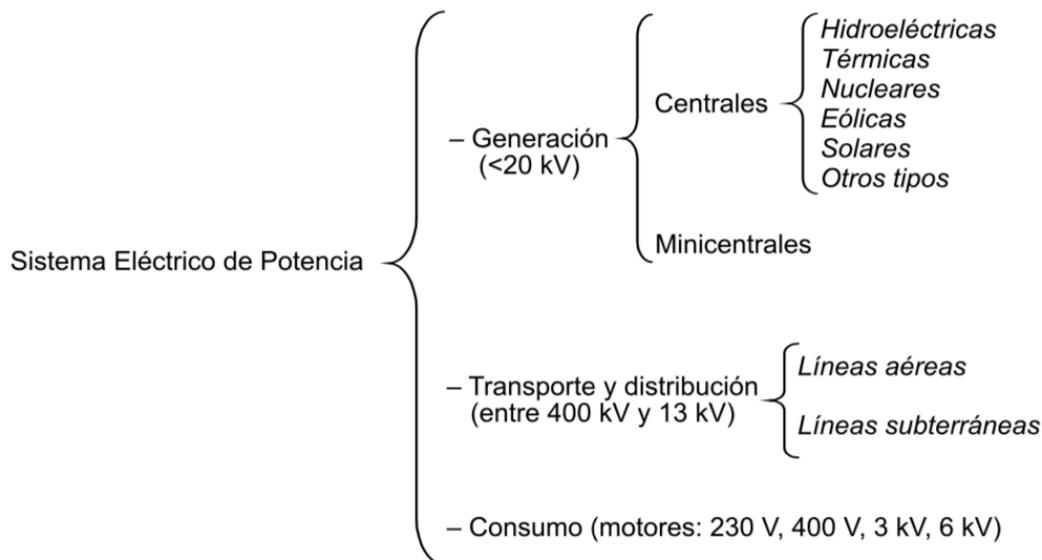


Figura 1: Esquema de las partes fundamentales de un Sistema eléctrico de Potencia [1].

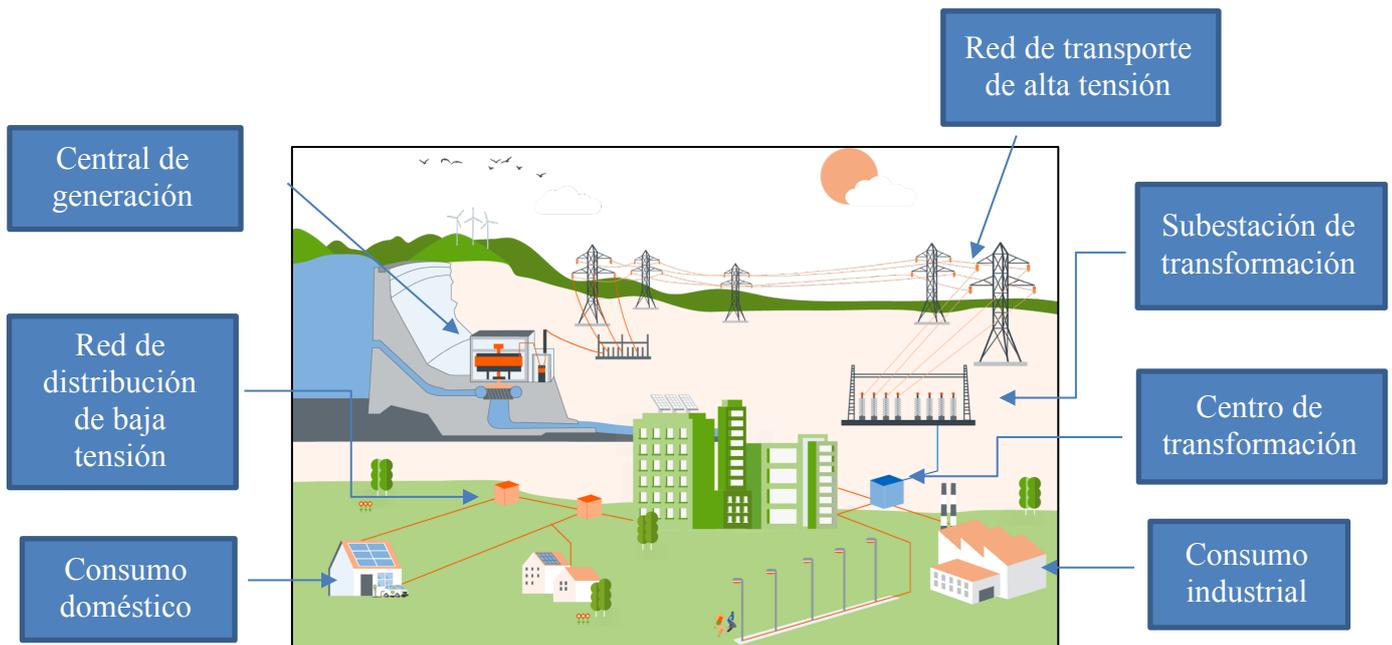


Figura 2: Esquema gráfico del sistema eléctrico de potencia [8].

Para describir el sistema eléctrico de potencia habría que empezar por donde tiene lugar la producción de energía, la generación. Esta tiene lugar en las centrales eléctricas, que en función del tipo de fuente primaria de energía que emplean, se pueden clasificar en hidroeléctricas (Figura 2), térmicas, nucleares, eólicas, solares, etc. Esa energía generada, es transmitida a los consumidores a través de las redes de transporte y distribución. Las redes de transporte transfieren la energía entre los centros de generación y los núcleos de reparto de carga (subestaciones). Mientras que las redes de distribución reparten la energía dentro de un núcleo de consumidores, constituidos principalmente por consumidores industriales y domésticos.

Finalmente, en los diferentes centros de consumo se recibe la energía generada en las centrales y se emplea en función de las características propias de cada centro (urbano, industrial, etc.). Este consumo puede realizarse a diferentes niveles de tensión de la red en función de la potencia demandada y los niveles de tensión presentes en la zona [1].

En consecuencia, para que el Sistema Eléctrico de Potencia pueda funcionar es necesario convertir diferentes formas de energía en energía eléctrica, transportar y distribuir esta energía hasta los centros de consumo y reconvertir la energía eléctrica a las formas de energía adecuadas para los diferentes tipos de consumo. Además, todo ello debe realizarse al nivel de tensión óptimo para cada etapa. Los elementos responsables de realizar las diferentes

conversiones de energía y adecuar el nivel de tensión de cada etapa a su valor óptimo son las máquinas eléctricas de corriente alterna [1].

2.2 Clasificación de las máquinas eléctricas

Las máquinas eléctricas se clasifican según su movilidad en máquinas estáticas y máquinas rotativas:

- **Máquinas estáticas:** son aquellas que, en funcionamiento, tienen todas sus partes fijas. La máquina estática fundamental es el transformador [1].
- **Máquinas rotativas:** son aquellas que, en funcionamiento, tienen una parte fija y otra móvil. La parte fija se denomina estator y la parte móvil rotor [1].

Las máquinas rotativas, a su vez, también se puede clasificar dependiendo de la función que desempeñan dentro del sistema eléctrico:

- **Generadores:** su función es transformar la energía mecánica procedente de diferentes fuentes de energía en energía eléctrica.
- **Motores:** su función es transformar la energía eléctrica procedente de red, en energía mecánica a través de un par de fuerzas que permite su giro, y, por consiguiente, la realización de un trabajo.

Además, tanto si desempeñan la función de generadores como si realizan la función de motores, las máquinas rotativas se clasifican en máquinas de corriente continua y máquinas de corriente alterna (síncronas o asíncronas), tal y como se indica en la Figura 3 [1].

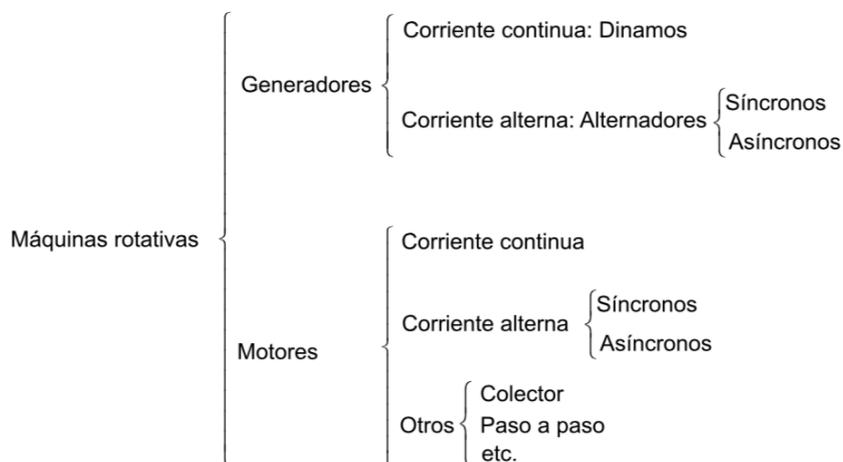


Figura 3: Clasificación de máquinas rotativas [1].

2.3 Máquinas síncronas

Las máquinas síncronas son convertidores electromagnéticos que, gracias a un campo magnético, transforman la energía mecánica, en forma de un par de fuerzas, en energía eléctrica de corriente alterna, cuando funcionan como generador, o la energía eléctrica en energía mecánica, cuando funcionan como motor. Es decir, son máquinas totalmente reversibles en su funcionamiento.

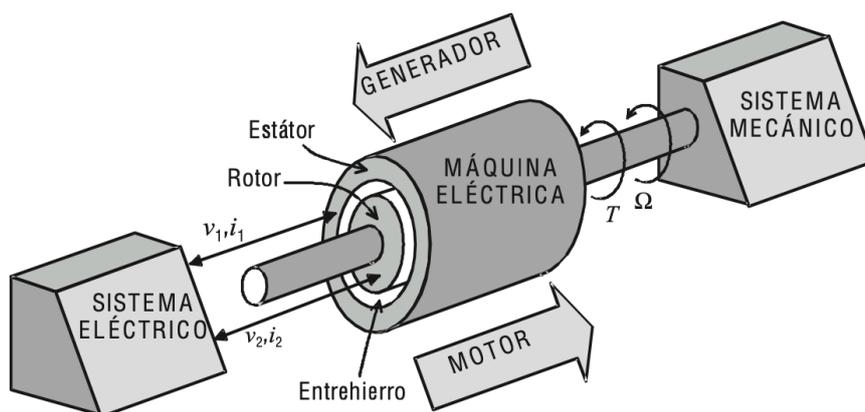


Figura 4: La máquina eléctrica como convertidor de energía [2].

Estas, a su vez, se caracterizan por girar a una velocidad denominada velocidad de sincronismo (n), que viene impuesto por la frecuencia de la red eléctrica (f) y por el número de pares de polos de la máquina (p).

$$n = \frac{60 \cdot f}{p} \quad (1)$$

El funcionamiento de estas consiste en un (bobinado) inductor alimentado por corriente continua, que suele colocarse en el rotor, y un (bobinado) inducido, normalmente trifásico, que suele colocarse en el estator. Cuando funcionan como generador (alternador), se introduce un par mecánico por el eje y al excitar el rotor con c.c, por medio de unos anillos (Figura 5), se induce una corriente alterna en el bobinado del estator. Asimismo, cuando funcionan como motor se excita el inducido (estator) con corriente alterna por medio de la red eléctrica a la que está conectado, además de excitar el rotor con c.c, apareciendo un par motor en el rotor que lo hace girar.

En el caso de funcionamiento como generadores síncronos, es la aplicación en la que más se emplean estas máquinas, aunque los motores síncronos también poseen diversas aplicaciones. Estos se utilizan en desempeños a baja velocidad, donde pueden trabajar a altas eficiencias y factores de potencia relativamente cercanos a la unidad. Además de funcionar como capacitores síncronos (motor síncrono funcionando en vacío), cuyo único propósito es el intercambio potencia reactiva con la red eléctrica, por medio de la regulación de la excitación del rotor en c.c. Lo que permite, de esta manera, regular el factor de potencia del mismo [3].

En función de la velocidad de giro a la que se ve sometido el rotor en sus diferentes aplicaciones, las características constructivas del rotor pueden ser de dos tipos (Figura 5):

- **Rotor de polos salientes:** son de gran diámetro y se utilizan mayoritariamente en centrales hidráulicas, con velocidades de giro bajas (entre 50 y 300 rpm) [3].
- **Rotor cilíndrico:** son de diámetro reducido (sin polos salientes) y se emplean especialmente en centrales térmicas, con velocidades de giro elevadas (entre 1800 y 3600 rpm) [3].

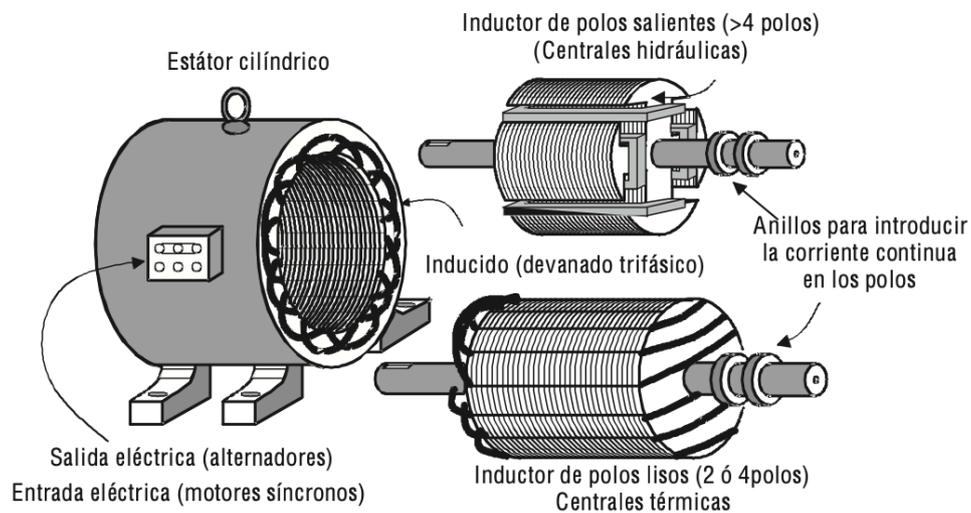


Figura 5: Tipo de máquinas síncronas [2].

2.4 Generador síncrono

El alternador o generador síncrono es la máquina generadora más importante dentro del sistema eléctrico de potencia, siendo esta, la máquina eléctrica que proporciona la mayor parte de la energía que se consume hoy en día. Es ampliamente utilizada en grandes centrales hidráulicas, térmicas y nucleares con potencias de hasta 1.000 MW (Figura 6). Por otro lado, también se emplean como grupos electrógenos acoplados a motores de combustión interna o incluso turborreactores que se utilizan como alimentación eléctrica en sistemas aislados, alejadas de las redes de distribución eléctrica, ya sean, en hospitales, aeropuertos, aviones (Figura 7) ... En caso de fallo en el suministro principal de la red eléctrica [2].



Figura 6: Presa de Itaipu - Río Paraná - Potencia Instalada de 14 GW - 20 Generadores de 18kV [10].

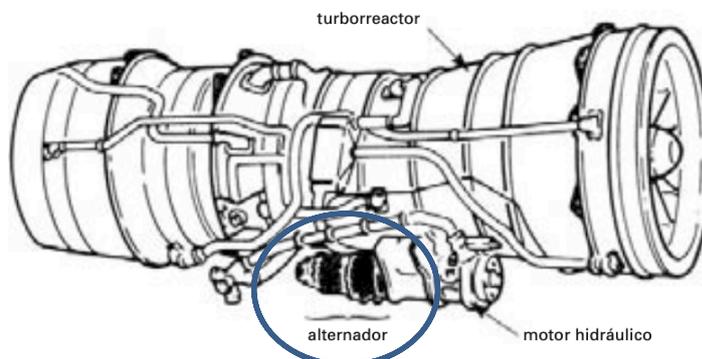


Figura 7: Generador trifásico de 60 kVA empleado en el Concorde para suministrar energía eléctrica, aprovechando parte de la potencia desarrollada por uno de los turborreactores del avión [3].

Una de sus principales características, a diferencia de otros generadores, como son los generadores asíncronos, es que estos primeros pueden regular tanto la potencia activa (kW) como la potencia reactiva (kVAR) que transmiten a la red eléctrica. Es decir, pueden ser regulados, por un lado, por medio de la excitación del rotor, lo que repercute directamente en la potencia reactiva. O, por otro lado, pueden ser regulados por medio de potencia mecánica que llega al eje, que se traduce en la velocidad de giro del rotor, repercutiendo en la potencia activa.

Otra característica de estas máquinas, es que el tamaño físico de una máquina de este tipo tiene una gran influencia en su eficiencia, salida de potencia, costo relativo y elevación de la temperatura [3].

La gran cantidad de energía generada hoy en día por las compañías eléctricas ha hecho que estas estén muy concienciadas con la eficiencia de sus generadores. Con respecto a esto, el tamaño del generador es realmente importante porque su eficiencia mejora automáticamente a medida que se incrementa la potencia del mismo. Esta mejora de la eficiencia con el tamaño es la razón por la cual los generadores síncronos de 1000 MW y más poseen eficiencias del orden del 99% [3].

En lo que se refiere a la velocidad de giro del rotor, los generadores de baja velocidad (rotor de polos salientes) siempre son más grandes que las máquinas de alta velocidad (rotor cilíndrico) de igual potencia. La magnitud de la baja velocidad simplifica el problema de enfriamiento. En el caso de los grandes generadores síncronos de baja velocidad instalados en plantas hidroeléctricas, estos son enfriados por aire o hidrógeno, dependiendo de las exigencias de refrigeración de los mismos [3].

3 OBJETIVOS Y ALCANCE

El objetivo primordial de este proyecto se basa en el diseño y desarrollo de una interfaz gráfica intuitiva y sencilla para el usuario a través de *AppDesigner*, herramienta para el diseño de aplicaciones que proporciona MATLAB, que nos permita definir los parámetros característicos de la máquina, así como las variables físicas (tensión, intensidad, potencia...) de la máquina en las diferentes situaciones de funcionamiento de vacío, cortocircuito y carga.

Además del objetivo principal, a lo largo del desarrollo de este proyecto también se establecen diversos objetivos secundarios:

- Profundizar en la comprensión de la máquina síncrona, más concretamente en el generador síncrono, como fuente de energía.
- Analizar el funcionamiento del generador síncrono ante diferentes parámetros de regulación de entrada y como afectan a los parámetros de salida.
- Adquirir un mayor conocimiento en el desarrollo software de diseño de interfaces de usuario, como es *AppDesigner*. Además de conseguir una mayor familiarización con el entorno MATLAB.
- Logro de una herramienta atractiva que otorgue al usuario agrado, sencillez, comodidad y velocidad en el estudio del comportamiento del generador. La interfaz gráfica nos permite observar de manera más práctica y rápida las diferentes situaciones de funcionamiento gracias a su gran capacidad computacional.
- Optimizar los recursos económicos, materiales y humanos en cuanto al diseño del generador, gracias a la simulación con la interfaz.
- Solución de problemas en las máquinas reales a través de pruebas de simulación con la interfaz gráfica.

4 BENEFICIOS DEL PROYECTO

En cuanto a los beneficios que se pueden obtener del desarrollo de este proyecto, podemos hablar de tres diferentes tipos, beneficios técnicos, beneficios económicos y beneficios sociales. En este proyecto, el beneficio técnico será el que tenga mayor relevancia dado la naturaleza del proyecto que se ha planteado.

4.1 Beneficios técnicos

Los generadores síncronos son una pieza clave en la generación de energía dentro del sistema eléctrico de potencia. A diferencia otras máquinas que se emplean para la generación, como lo son los generadores asíncronos, que necesitan consumir potencia reactiva, estos primeros, no solo no la consumen, sino que tienen la capacidad de regular la inyección de la misma, que transmiten a la red, por medio de la excitación del rotor con corriente continua (c.c).

Es el hecho de tener esa alta capacidad de regulación, pudiendo variar tanto la potencia reactiva mencionada anteriormente, como la potencia activa (como es normal en todo tipo de generadores), lo que le hace ser una máquina muy utilizada en grandes centrales de generación como son las centrales hidráulicas y térmicas, además de tener otras aplicaciones en redes aisladas como servicio auxiliar. Debido a la alta demanda en su uso como generador, visto anteriormente, es imprescindible poder asegurar su correcto funcionamiento en cualquiera de las situaciones de funcionamiento en las que se encuentre a lo largo de su vida útil.

Por lo tanto, ser capaces de identificar los posibles fallos en el funcionamiento del generador, incluso antes de que se produzcan, ahorraría incontables problemas a corto y largo plazo en la máquina. Además, poder comprobar la efectividad de diferentes soluciones adoptadas a dichos problemas que puedan surgir, es de vital importancia. La interfaz gráfica planteada nos permite lograr todo lo anterior, introduciendo los datos nominales, los datos de ensayos de vacío y cortocircuito, y los parámetros de la situación de funcionamiento del generador. De este modo, se podrá simular los parámetros físicos de salida en cualquiera de las posibles situaciones de funcionamiento de la máquina. Consecuentemente, en base a los datos de salida de la simulación, se podrá predecir futuros errores en la fase de diseño de la máquina o evaluar posibles soluciones a problemas actuales en un generador, asegurando así su correcto funcionamiento.

4.2 Beneficios económicos

A la hora de realizar pruebas para comprobar el correcto funcionamiento de máquina, la interfaz gráfica juega un papel muy importante, ya que, utilizar la simulación en vez de ensayos de campo, es mucho más rápido y menos costoso. Se pueden llevar a cabo infinidad de simulaciones de diferentes situaciones en cuestión de minutos, mientras que, en los ensayos de campo, a parte de tener que desplazarte donde se ubique la máquina, se requiere más protocolos de seguridad antes de la manipulación de la máquina real, que ralentizan todo el proceso y derivando en un aumento de los costes.

4.3 Beneficios sociales

En cuanto al papel que juega en la sociedad, la simulación por medio de la interfaz gráfica permite que no sea necesario la manipulación, tan frecuente, en máquinas de magnitudes masivas, como son los generadores síncronos, salvo para labores imprescindibles de mantenimiento. Permitiendo solventar problemáticas en el funcionamiento de la máquina sin apenas tener que intervenir con ensayos de campo. Por consiguiente, se mejora considerablemente la seguridad laboral de los operarios que la manipulan la máquina, a parte de otorgar fiabilidad de suministro a la red eléctrica con su correcto funcionamiento.

Esta herramienta gráfica también dispone de un objetivo de carácter didáctico, puesto que, puede ser empleada por alumnos de la escuela de forma que pueda suponer un apoyo en el momento del estudio del funcionamiento el generador. Todo ello, de forma más interactiva y práctica, para un mejor desempeño en el aprendizaje de los mismos.

5 ESTADO DEL ARTE

Antes de profundizar en el desarrollo de la interfaz gráfica, será necesario realizar una explicación detallada de la máquina síncrona, así como su desempeño como generador síncrono. Para ello, en primer lugar, se estudiarán los elementos más importantes de la máquina síncrona, que están constituidos por las partes activas y los sistemas auxiliares. En segundo lugar, se realizará un estudio de los principios de funcionamiento del generador, haciendo hincapié posteriormente en las situaciones de funcionamiento de vacío cortocircuito y carga, necesarios para comprender mejor los parámetros de la máquina y su situación de funcionamiento como generador. Por último, en cuanto, al generador, se realizará una explicación de como opera el sistema de regulación y control, incidiendo sobre los actuadores (variables de entrada) que permiten regular el sistema.

5.1 Elementos constitutivos de una máquina síncrona

Las maquinas eléctricas, en su totalidad, disponen de dos circuitos básicos, relacionados entre sí por medio de las leyes fundamentales del electromagnetismo, que permiten el funcionamiento de la máquina. Por un lado, tenemos el circuito magnético, cuya misión es permitir la circulación del flujo magnético creado en el bobinado inductor y recibido por el bobinado inducido. Por otro lado, tenemos el circuito eléctrico, que consta, asimismo, de dos circuitos (inductor e inducido), los cuales están magnéticamente unidos (por medio del entrehierro entre rotor y estator) y eléctricamente aislados. Ambos circuitos forman lo que se denomina las partes activas de la máquina.

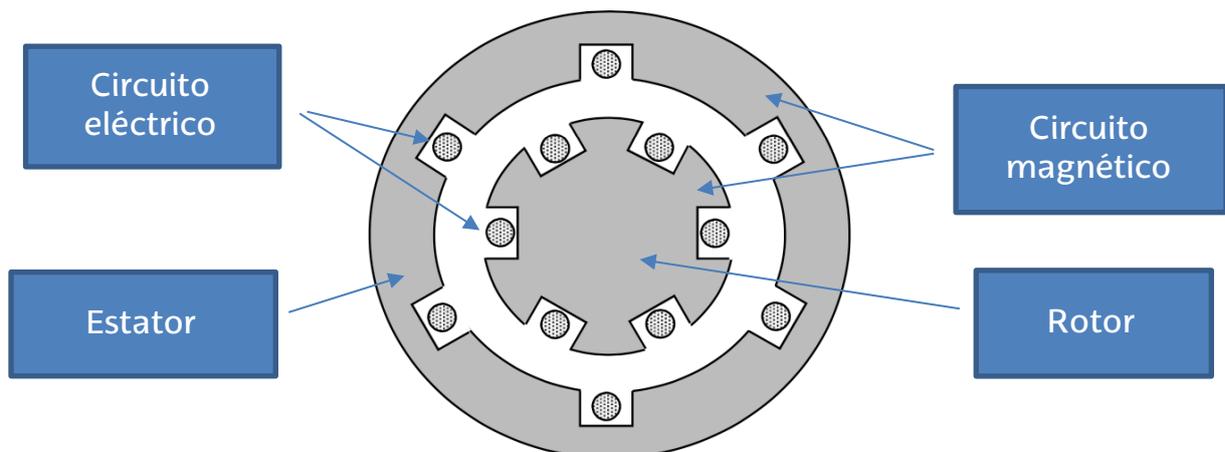


Figura 8: Esquema eléctrico y magnético de una máquina rotativa [1].

5.1.1 Partes Activas

Estas máquinas, para su funcionamiento, necesita disponer de una serie de elementos activos esenciales. Por lo tanto, se describirán cada uno de ellos en detalle a continuación.

5.1.1.1 Circuito Magnético

Dentro de toda máquina eléctrica, existe un responsable de crear el campo magnético y otro recibe la influencia de dicho campo magnético. Cada uno de ellos se denomina inductor e inducido respectivamente. En concreto, en las máquinas síncronas, a diferencia de las asíncronas, el inductor se encuentra en el rotor y el inducido se encuentra en el estator.

- **Rotor:** es una parte del circuito magnético de la máquina que tiene la capacidad de rotar sobre su propio eje. Además, es la parte de la maquina encargada de recibir la excitación externa en corriente continua (inductor). Existen, según la construcción del mismo, dos tipos de rotores.
 - **Rotor de polos salientes:** estos rotores están diseñados con una disposición de polos como se puede apreciar en la Figura 10. Cada uno de estos polos se construye a partir un material magnético alrededor del cual se enrolla el bobinado. Sin embargo, debido a la disposición de los polos en la periferia de gran diámetro del rotor y a las grandes fuerzas de inercia que se experimentan cuando el rotor gira, las velocidades del rotor están limitadas constructivamente a unas velocidades moderadas.



Figura 9: Rotor de Polos Salientes (Hidroalternador de la central de Itaipú) [9].

- **Rotor cilíndrico:** estos rotores están diseñados para girar a elevadas velocidades de giro, al contrario que los de polos salientes. Estos, se construyen a partir de un cilindro de material magnético, el cual posee unas ranuras por donde irán alojados los devanados, lo que le permite tener menor diámetro y una mayor longitud que el anterior (Figura 11).

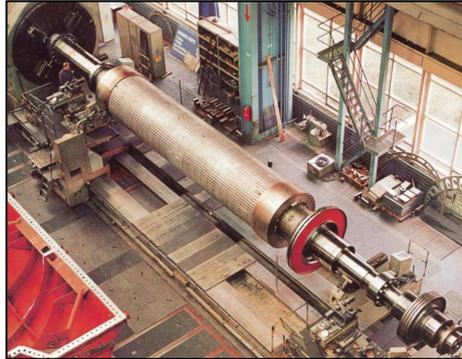


Figura 10: Rotor Cilíndrico [9].

- **Estator:** es la parte fija de la máquina encargada de alojar las bobinas inducidas donde se genera la corriente alterna. Está formada por un conjunto de chapas apiladas y aisladas entre sí, lo que permite disminuir las pérdidas en el hierro del circuito magnético y mejorar así el rendimiento de la máquina.

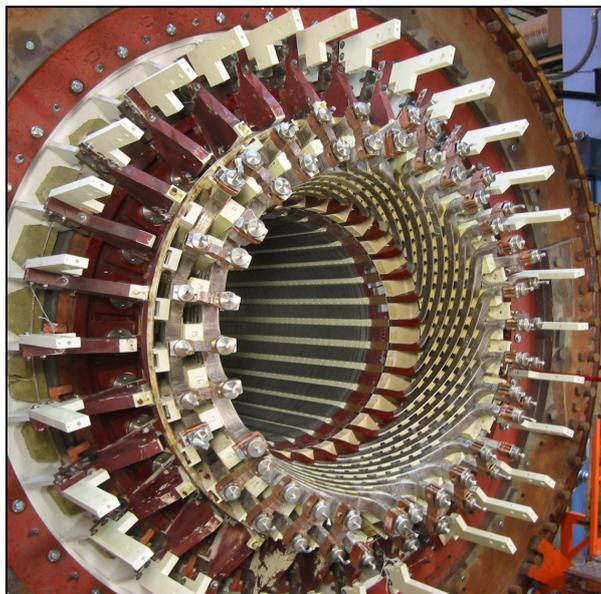


Figura 11: Estator de un generador síncrono (Central nuclear nórdica) [11].

5.1.1.2 Circuito eléctrico

- **Bobinado del estator:** Es el circuito eléctrico situado en el estator. Está constituido por espiras que se ubican a lo largo de las ranuras del estator y por donde se inducen tensiones alternas. Cuando el generador trabaja en carga, estas son recorridas por corrientes alternas [1].



Figura 12: Devanado del estator [9].

- **Bobinado del rotor:** es el circuito eléctrico situado en el rotor (Figura 14). Se trata de un bobinado recorrido por corriente continua que tiene por misión crear los polos inductores de la máquina [1].
- **Bobinado amortiguador:** es un conjunto de barras de aluminio y cobre ubicadas en los rotores de polos salientes, las cuales están insertadas en las extensiones polares con diferentes objetivos. Entre ellos están, el funcionamiento correcto en régimen desequilibrado (tensiones-intensidades), la limitación de los campos armónicos y la mejora del comportamiento en cortocircuito [12].



Figura 13: Devanados del rotor con bobinado amortiguador [12].

5.1.2 Sistemas auxiliares

Estas máquinas, para el funcionamiento óptimo y seguro de las partes activas que las componen, necesitan disponer de una serie de sistemas auxiliares.

- **Sistema de excitación:** es el conjunto de equipos que permiten suministrar una tensión e intensidad de corriente continua al devanado del rotor (el cual está en movimiento) [1]. Existen diferentes tipos de tecnologías de sistemas de excitación.
 - **Sistema de excitación de corriente continua:** consiste en una excitatriz principal y una excitatriz piloto (auxiliar) a base de máquinas de corriente continua. La corriente de excitación (I_X) se controla por medio de la corriente del devanado inductor (I_c), mediante un regulador automático de tensión (AVR, *Automatic Voltage Regulator*). Además, a la hora de transmitir la corriente continua de la excitatriz principal al rotor, es necesario un sistema de escobillas y anillos rozantes [14].

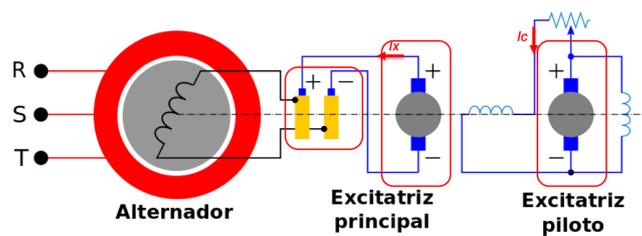


Figura 14: Sistema de excitación de corriente continua [14].

- **Sistema de excitación sin escobillas (*brushless*):** en este otro caso, la excitatriz principal es una máquina de corriente alterna invertida (estator CC y rotor CA) con un rectificador montado sobre el rotor. Asimismo, la excitatriz piloto es una máquina de corriente alterna de imanes permanentes [13]. Todo ello, permite evitar la utilización de escobillas y anillos rozantes, que son piezas sometidas a gran desgaste y requieren manteniendo frecuente. Por último, como en el caso anterior, la corriente de excitación se controla mediante un regulador (AVR).

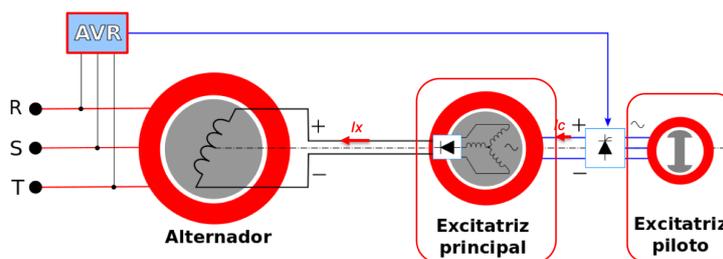


Figura 15: Sistema de excitación sin escobillas (*brushless*) [13].

- **Sistema de excitación estático o autoexcitado:** en este último caso, la excitación de corriente se obtiene por medio de un transformador conectado a la red eléctrica (a la salida del alternador). Esta corriente de excitación se controla por medio del puente rectificador (puente de tiristores controlados) que convierte la intensidad alterna de la red en continua. Es necesario una fuente externa de CC (auxiliar) para poder arrancar en ausencia de conexión con la red (*black start*) [13].

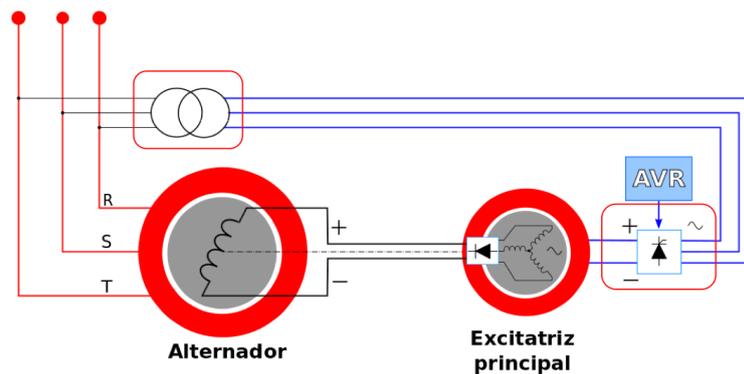


Figura 16: Sistema de excitación estático o autoexcitado [13].

- **Sistema de refrigeración:** el alternador, durante su funcionamiento, necesita disipar gran cantidad de calor debido a pérdidas por efecto Joule y pérdidas por corrientes de Foucault, entre otras, para evitar así el envejecimiento prematuro de los elementos que constituyen la máquina [1]. A mayor potencia nominal del generador, mayores son las necesidades de refrigeración. Por ejemplo, en pequeños generadores o grandes con velocidades lentas (centrales hidroeléctricas), suelen estar refrigerados por aire. Mientras tanto, en grandes generadores rápidos (centrales térmicas), es más habitual la refrigeración por hidrógeno [13], por lo que el agente refrigerante cambiará en función de la aplicación (si se necesita más refrigeración o no).
- **Sistema regulación y control:** su objetivo es optimizar el funcionamiento de la máquina, adecuándose en todo momento a la situación de operación en la que se desee trabajar. Este sistema debe incluir un sistema de captación de información (que permita conocer la situación de funcionamiento por medio de unos parámetros de entrada característicos de la máquina). De este modo, podrá regular la máquina, a través de diferentes actuadores (que se verán mas adelante en la regulación en su funcionamiento como red aislada), de la manera más adecuada.

5.2 Principios de funcionamiento del generador síncrono

Como se ha mencionado en apartados anteriores, la máquina síncrona, es una máquina rotativa totalmente reversible que tiene la capacidad de trabajar, tanto como motor, como generador. En cuanto al número de fases eléctricas de las máquinas, lo habitual son máquinas síncronas monofásicas, bifásicas o trifásicas. A largo de este apartado en adelante, el objeto de estudio será de máquinas trifásicas, al ser este tipo en las que está basada la interfaz gráfica, además de ser las más empleadas en centrales de generación.

En cuanto a la máquina como motor, funciona de la siguiente manera. El rotor es alimentado con corriente continua por sus devanados, a su vez, el estator, que está conectado a la red eléctrica, está recorrido por corrientes trifásicas (por sus devanados), cada una de igual magnitud y desfasadas entre ellas 120° . Todo ello, producirá según el teorema de Ferraris [4], un campo magnético giratorio de magnitud constante, que gira a la velocidad de sincronismo. Por último, los conductores de los devanados del rotor, por los que circula intensidad de c.c, se verán bajo la acción del campo magnético rotativo creado por el estator, por lo que según el teorema de Laplace [1], se inducirá una fuerza en ellos que hará girar al rotor.

En cuanto a la máquina como generador, se estudiará en más en detalle a continuación. Para comenzar, el generador o alternador síncrono se caracteriza por tener un bobinado inductor en el rotor y un bobinado inducido en el estator. Los polos del rotor se inducen haciendo pasar por sus devanados corriente continua, definiéndose así los polos norte y sur en función del sentido de circulación de la corriente (Figura 17).

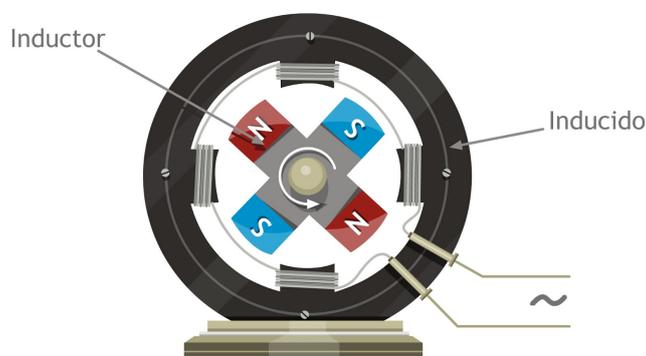


Figura 17: Esquema máquina síncrona [13].

Las líneas de campo magnético creado por la corriente, que fluye a través del bobinado del rotor, se cierran a través del rotor-entrehierro-estator (circuito magnético). Al mismo tiempo, se hacen girar el rotor por medio de una fuente externa de energía, por ejemplo, una turbina, de modo que se crea un campo magnético rotativo que se mueve respecto de los bobinados del estator. De esta manera, según la ley de Faraday [1], se inducirá una tensión o f.e.m en los conductores, que se mantienen estáticos comparados con el campo magnético giratorio.

En general, si el estator contiene N espiras, se inducirán en ellas una f.e.m total de valor eficaz:

$$E = 4,44 \cdot N \cdot \phi \cdot f$$

Por lo tanto, en la máquina se genera una inducción magnética sinusoidal gracias al giro del rotor. Es decir, el rotor crea un campo magnético fijo, pero al mismo el rotor gira, por lo que, las líneas de campo se desplazan con el. Generando de esta manera en el estator, un ciclo de tensión alterna de frecuencia f que varía en función de la velocidad de giro del rotor n (rpm) y el número de pares de polos p .

$$p \cdot n = 60 \cdot f \quad (1)$$

Al mismo tiempo, la distancia angular existente (grados geométricos) entre los polos norte y sur consecutivos no es arbitraria, sino que dependen de los grados eléctricos y el numero de pares de polos. En el caso del generador trifásico, se debe conseguir un sistema de tensiones trifásicas equilibradas, por lo tanto, los grados eléctricos deben ser siempre de 120° .

$$\theta_{\text{eléctricos}} = p \cdot \theta_{\text{geométricos}}$$

5.2.1 Situaciones de funcionamiento del Generador Síncrono

El generador síncrono puede funcionar de tres formas diferentes: en vacío, cortocircuito y carga. Pero, por lo general, el funcionamiento habitual es en carga, mientras que, las otras dos primeras situaciones se deben a periodos en los que o bien no hay carga que alimentar o simplemente hay una situación de falta, respectivamente. Asimismo, se realizan ensayos a la máquina en las situaciones de vacío y cortocircuito, de manera controlada, para poder obtener información adicional de parámetros de la máquina.

5.2.1.1 Situación de Vacío

El objetivo del ensayo de vacío es definir la relación entre la tensión de vacío (E_0) y la intensidad de excitación (I_e), de forma que se pueda definir la tensión en función del parámetro de excitación mencionado. El ensayo se realiza de la siguiente manera, se hace girar el rotor de la máquina a una velocidad de giro determinada constante a lo largo del ensayo, estando el estator sin ninguna carga conectada. De mientras, se empieza a excitar la máquina empezando con una intensidad nula e incrementándola a lo largo del ensayo, anotando los valores de tensión para cada valor de intensidad de excitación.

De este modo, se define la relación como se puede mostrar en la Figura 18, en la cual, se puede observar como la línea recta corresponde a una situación ideal en la que no existiera saturación del circuito magnético, que es el caso simplificado empleado para el desarrollo de la interfaz. Mientras tanto, la segunda curva, corresponde a una situación real en la que llegado un valor de I_e el circuito magnético queda sobreexcitado y no permite mayor flujo magnético. Por consiguiente, la tensión inducida ya deja de ser proporcional a la intensidad, teniendo que aumentar en gran medida I_e para conseguir un aumento significativo en la E_0 .

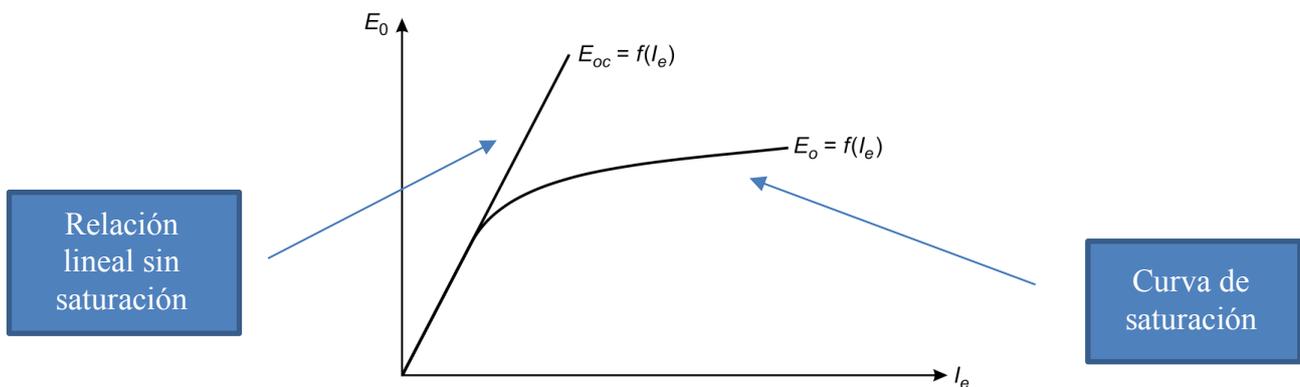


Figura 18: Curva característica de vacío [1].

5.2.1.2 Situación de Cortocircuito

El objetivo del ensayo de cortocircuito es definir la reactancia síncrona (X_s), es decir, un parámetro de la máquina necesario para conocer el funcionamiento de la misma. El ensayo se realiza de la siguiente manera, se hace girar el rotor de la máquina a una velocidad de giro determinada constante a lo largo del ensayo, además, se cortocircuitan los terminales de salida (bornes) del estator. De manera simultánea y análogamente al caso anterior, se van registrando los valores de la intensidad que recorren los devanados del estator por fase (I_{cc}) mientras se variando la I_e (Figura 19).

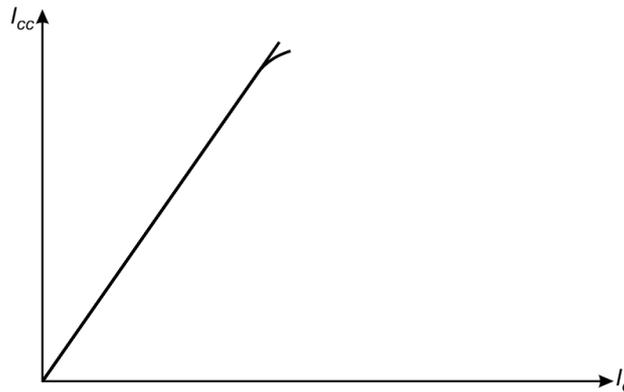


Figura 19: Curva característica de cortocircuito [1].

Teniendo en cuenta el circuito monofásico equivalente del generador en la situación de cortocircuito (Figura 20) se deduce el cálculo de X_s .

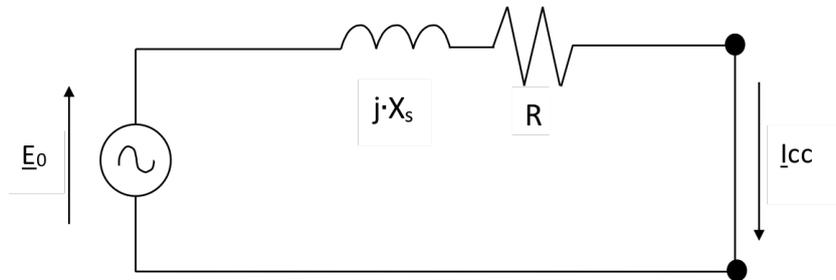


Figura 20: Esquema monofásico equivalente en cortocircuito.

En cuanto al cálculo de la reactancia síncrona (X_s), primero se calcula la intensidad de cortocircuito (I_{cc}).

$$I_{cc} = \frac{E_0}{\sqrt{(R^2 + X_s^2)}} \quad (2)$$

Asumiendo que en la práctica $X_s \gg R$, se realiza una simplificación de tal manera que la R (resistencia de los devanados por fase del rotor) se puede despreciar frente a X_s en el cálculo de I_{cc} [4].

Una vez obtenido la corriente de cortocircuito (I_{cc}) y con E_0 podemos calcular la X_s .

$$X_s = \frac{E_0}{I_{cc}} \quad (3)$$

La reactancia síncrona se puede expresar en función de una constante llamada inductancia síncrona (L_s) y la frecuencia (f) a través de la siguiente ecuación:

$$X_s = L_s \cdot 2\pi \cdot f \quad (4)$$

5.2.1.3 Situación de Carga

Como ya se ha mencionado en el apartado del contexto, según la forma constructiva existen dos tipos de rotores:

- Rotor cilíndrico o de entrehierro constante
- Rotor de polos salientes o de entrehierro variable

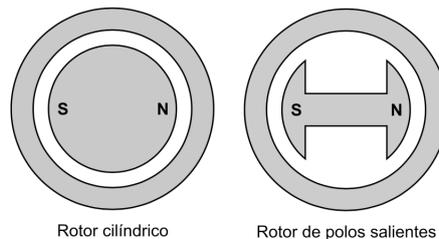


Figura 21: Tipos de rotores en generadores síncronos [1].

Para establecer el diagrama vectorial correspondiente al funcionamiento en carga es necesario distinguir el alternador si tiene rotor cilíndrico o de polos salientes. En el caso que concierne en cuanto a la resolución de esta interfaz gráfica, se asumirá que el rotor es cilíndrico.

La ecuación que rige el comportamiento del generador en carga es la siguiente:

$$\underline{E}_0 = \underline{V} + \underline{I} \cdot (R + j \cdot X_s) \quad (5)$$

\underline{E}_0 = Tensión interna del generador inducida por la corriente de excitación (I_e).

\underline{V} = Tensión por fase en bornes del generador (estator).

\underline{I} = Intensidad por fase que recorre los devanados del estator.

R = Resistencia por fase de los devanados del rotor.

X_s = Reactancia síncrona por fase.

De la ecuación vectorial (4) se puede deducir la Figura 22 de la situación en carga.

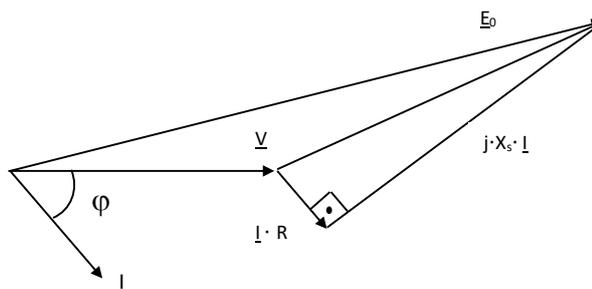


Figura 22: Diagrama vectorial.

Cabe mencionar que el circuito equivalente de la Figura 23 se obtiene de la premisa que la carga trifásica está conectada en estrella, siendo $Z \angle \varphi$ la carga por fase, indicado como un fasor (vector) de módulo Z y ángulo φ .

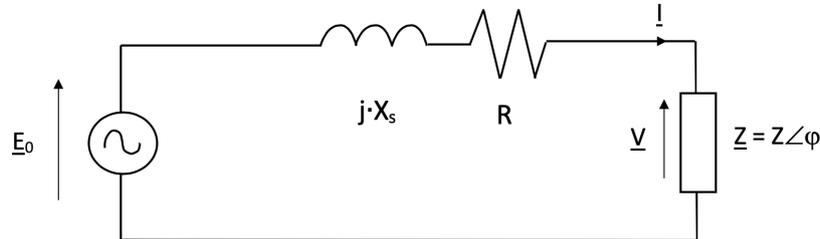


Figura 23: Circuito equivalente en carga.

En la situación de carga, la potencia consumida será dato para calcular la impedancia de la carga por fase. Pero se pueden definir la potencia activa (P) y reactiva (Q) de la siguiente manera mediante parámetros eléctricos:

$$P = \sqrt{3} \cdot U \cdot I \cdot \cos\varphi \quad (6)$$

$$Q = \sqrt{3} \cdot U \cdot I \cdot \sen\varphi \quad (7)$$

Además, estas dos últimas formulas se pueden agrupar en una sola de forma que la potencia aparente (S) se puede expresar como:

$$S = \sqrt{3} \cdot U \cdot I \quad (8)$$

Todo ello, viene de la relación entre las tres potencias que se refleja en esta ecuación:

$$S^2 = P^2 + Q^2$$

Por otro lado, cabe mencionar la relación existente de tensiones entre tensiones simples (V) y compuestas (U). Ya que, aunque en los circuitos equivalentes se habla de parámetros por fase, como ya se ha mencionado antes, estas máquinas suelen ser trifásicas. Por lo tanto, la tensión entre fase y neutro (V) está relacionada con la tensión entre fase y fase (U) a través de la siguiente ecuación.

$$U = \sqrt{3} \cdot V \quad (9)$$

5.3 Regulación y control en sistema de red aislada

En primer lugar, cabe mencionar que el generador puede funcionar solo, conectado a una carga trifásica equilibrada (red aislada o en isla), o simplemente ser un generador más, al igual que muchos otros, conectado a una red de potencia infinita (red eléctrica). En el caso que concierne al estudio, se supondrá el funcionamiento en isla, por lo que, la tensión de alimentación de la carga (V) es variable y el factor de potencia (f.d.p) de la carga es fijo.

Al mismo tiempo, es importante mencionar como el generador en su funcionamiento se comporta de manera completamente distinta dependiendo del carácter de la carga. Pudiendo ser esta, de carácter resistiva, inductiva o capacitiva. Es decir, si suponemos que la máquina tiene velocidad de giro (n) constante y, por tanto, el parámetro de la frecuencia (f) también constante. Cuando introducimos una carga de cualquier tipo se genera una reacción de inducido (F_{mm_i}) debido a la corriente que se induce en el estator, esta fuerza magnetomotriz, dependiendo del tipo de carga acoplada, se sumará o restará a la ya existente de excitación del rotor (F_{mm_e}). Por lo tanto, se puede afirmar que para cargas inductivas el efecto es desmagnetizante mientras que para cargas capacitivas el efecto es magnetizante.

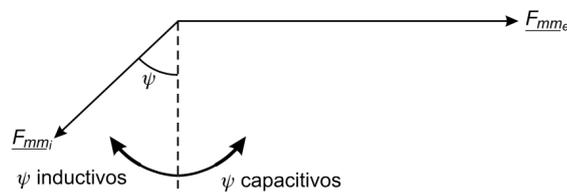


Figura 24: Posicionamiento de la reacción de inducido [1].

Estos efectos de las cargas sobre el comportamiento del generador se pueden apreciar en los parámetros eléctricos (Figura 25) donde para una intensidad de excitación (I_e) constante, en el caso de cargas capacitivas la tensión en bornes (V) se ve aumentado con respecto a la situación nominal (efecto magnetizante). Mientras tanto, con la carga inductiva (efecto desmagnetizante) o resistiva se produce el efecto contrario y la V es menor que la nominal (V_N).

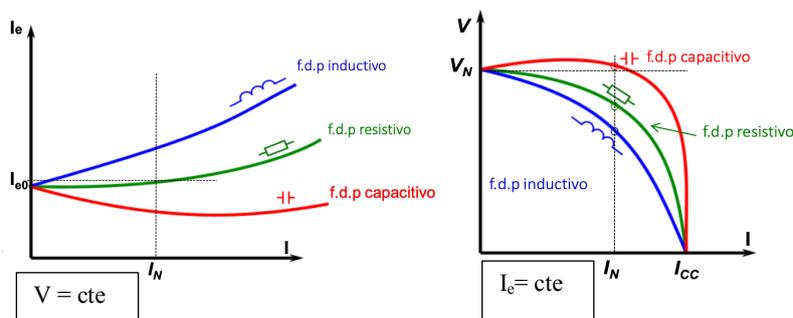


Figura 25: Curvas de carga del generador [13].

Como se ha podido apreciar, el generador para poder trabajar con unas condiciones prefijadas de tensión (V) y frecuencia (f) necesita disponer de un sistema de regulación y control. A continuación, se exponen los dos tipos de sistemas de regulación necesarios para su correcto funcionamiento.

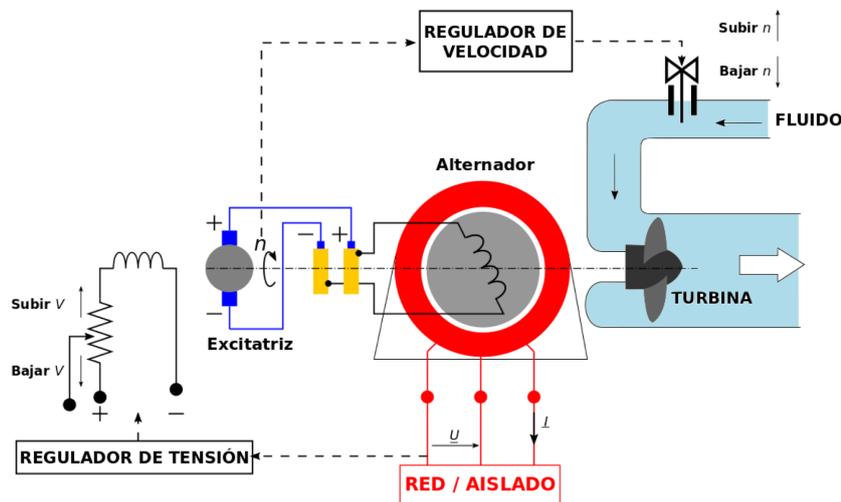


Figura 26: *Sistemas de Regulación y Control [13].*

En primer lugar, el *sistema de regulación de tensión*, está ligado al sistema de excitación del rotor (I_e), que, variando la excitación del mismo, se puede variar E_o y, por lo tanto, también la tensión en bornes del generador (V). De este modo, el sistema de control comprueba la consigna de tensión establecida y la compara con el parámetro de tensión actual, de modo que actúa en consecuencia, aumentando o disminuyendo I_e , para mantener la tensión consigna (prefijada) a la que trabaja la carga.

En segundo lugar, el *sistema de regulación de velocidad*, actúa cuando hay una variación en la potencia demandada por la carga, que influye inversamente proporcional a la velocidad del generador. De esta manera, el sistema de regulación y control actuará en el caso de una disminución de la velocidad del rotor (debido a un aumento de la potencia demandada), abriendo la admisión de agua a la turbina (Figura 26), para así, aumentar la velocidad del mismo hasta el valor inicial. Manteniendo de esta manera la velocidad constante, lo que conlleva directamente a la frecuencia constante deseada, representado a través de la ecuación (1).

6 ANÁLISIS DE ALTERNATIVAS

6.1 Análisis de alternativas para el estudio del generador síncrono

En este apartado se va a proceder a realizar el estudio de las diferentes alternativas que se disponen con respecto a la realización del proyecto. Cabe mencionar, que el principal objetivo del proyecto reside en el diseño de una interfaz gráfica para analizar el funcionamiento de los generadores síncronos, se analizarán otras opciones con las cuales, se puedan desempeñar la misma función. Entre estas alternativas, podemos hablar de pruebas de campo o la realización de modelos físicos a escalas. Finalmente, en base a unos criterios concretos que serán ponderados más adelante se llegará a una elección de manera cuantitativa por medio de una matriz de priorización.

6.1.1 Pruebas de campo

Las pruebas de campo consisten en la realización de mediciones de parámetros eléctricos en la propia máquina de manera física. Todo ello conlleva, por un lado, que las medidas obtenidas del generador, en diferentes situaciones de funcionamiento, sean completamente reales y que puedan diferir en cierto grado con valores teóricos calculados, lo cual daría una mayor precisión y facilidad para detectar funcionamientos anómalos en el generador. Por otro lado, este tipo de pruebas supone un gran impacto económico en cuanto al tiempo de desplazamiento de operarios para realizar las pruebas y al periodo de inactividad del generador durante dichas pruebas, que no permitiría hacer uso más rentable de la misma.

6.1.2 Modelos físicos a escala

Los modelos físicos a escala son una variante de las pruebas de campo, ya que, se realiza un modelo a escala de la máquina lo más asemejado posible a la realidad, donde se realizan las pruebas de funcionamiento de la máquina. Todo ello, permite obtener datos de los ensayos realizados al modelo, que no siempre corresponden exactamente con datos reales de la máquina. Eso dependerá de la calidad del modelo realizado, es decir, de si se ha hecho una buena correlación entre modelo y realidad, que para un mejor desempeño conllevará inevitablemente a una mayor inversión económica en el mismo. Adicionalmente, este método es poco polivalente, ya que, para cada generador (de diferentes características) que se desee analizar, es necesario hacer un modelo exclusivo para cada caso, lo que encarece aún más los costes de esta alternativa.

6.1.3 Simulación software

Esta alternativa consiste en el desarrollo de una interfaz gráfica de usuario que permita simular las diferentes situaciones de funcionamiento del generador de manera rápida e intuitiva para el usuario. Para ello, es necesario la introducción de unos parámetros de entrada como son parámetros nominales de la máquina, así como los ensayos de vacío y cortocircuito.

De esta forma, el software nos permitirá calcular en base a estos últimos datos y otros más que proporcionará el usuario antes de realizar la simulación, para dar como resultado los parámetros eléctricos que definen la situación de funcionamiento seleccionada. Aunque esta alternativa puede proporcionar una gran cantidad de datos de simulaciones de manera rápida y sencilla, hay que tener en cuenta que son datos calculados teóricos y no reales, pero si se pueden asemejar a ellos en gran medida.

6.1.4 Criterios de selección

En este apartado se establecen los criterios que se han considerado a la hora de decidir cual de las opciones a las tres alternativas mencionadas resulta la mejor para abordar el proyecto.

- **Coste económico:** es uno de los criterios más importantes a la hora de decidir, puesto que, va a determinar el presupuesto del proyecto. Dentro de este criterio se tendrá en cuenta gastos como desplazamientos de operarios, inversión en construcción de modelos a escala o gastos de licencias de software. Por lo tanto, la opción con los gastos más bajos será la opción que se valorará más positivamente.
- **Recursos humanos y materiales:** en este criterio se tiene en cuenta tanto elementos humanos como los bienes materiales necesarios para llevar a cabo el objetivo del proyecto en cada una de las alternativas. De esta manera, cuantos menos recursos tengamos que emplear para desempeñar el objetivo, será una opción más eficiente y consecuentemente, más viable.
- **Tiempo de ejecución:** el tiempo es un bien valioso, por lo que será importante analizar la rapidez con la que se puede realizar los ensayos en cada caso. Además, se tendrá en cuenta si el tiempo empleado en la realización de los ensayos repercute directamente en tiempo ineficiente de la máquina, es decir, tiempo que tiene que estar ensayándose, en vez de produciendo energía para la red eléctrica. Consecuentemente, se valorará positivamente tiempos de ejecución cuanto más pequeños posibles mejor.

- **Calidad de los resultados:** es uno aspecto a destacar debido a que para poder solucionar de manera más sencilla posibles defectos en el funcionamiento de la máquina es necesario que los datos de los ensayos sean lo más fiables posibles.

6.1.5 Elección de la solución

En este apartado se procederá a la elección de la alternativa más adecuada para el proyecto teniendo en cuenta los criterios de selección comentados en el apartado anterior, y en base a ellos se comprará con las otras alternativas.

Para llevar a cabo la comparación de las alternativas de manera cuantitativa se emplea la matriz de priorización. Esta matriz consiste, por un lado, en una serie de criterios de elección que se ponderan porcentualmente de 0 a 100% en función de su peso (importancia). En el caso del estudio se han aplicado los siguientes pesos:

- Coste económico: 35%.
- Recursos humanos y materiales: 15%.
- Tiempo de ejecución: 30%.
- Calidad de los resultados: 20%.

Criterios de selección	Peso	Alternativas valoradas sobre 10 puntos		
		Pruebas de campo	Modelos físicos a escala	Simulación software
Coste Económico	35%	3	6	8
Recursos humanos y materiales	15%	3	5	9
Tiempo de ejecución	30%	5	7	10
Calidad de los resultados	20%	10	8	6
TOTAL		5	6,55	8,35

Tabla 1: Matriz de priorización de alternativas.

Por otro lado, se evalúan de 0 a 10 puntos cada una de las alternativas para cada uno de los tres diferentes criterios. Siendo 0 que la alternativa seleccionada no cumple con el criterio en cuestión, mientras que, 10 significa que se adecua perfectamente al criterio descrito.

Analizado los resultados obtenidos de la matriz de priorización, se concluye que la simulación software es la opción más adecuada para el proyecto, dado que, en primer lugar, nos da un menor tiempo de ejecución de los ensayos, además de no repercutir negativamente durante la realización de los ensayos, no requiriendo la presencia física de la maquina para la

realización de los mismos. En segundo lugar, no necesita apenas recursos humanos ni materiales, únicamente un encargado de manejar la interfaz gráfica. Por último, a pesar de tener respecto a las otras dos alternativas, peor fiabilidad de resultados, esta flaqueza se compensa con las demás ventajas mencionadas.

En lo que respecta a las demás alternativas, se puede comentar como las pruebas de campo son la alternativa que mayor fiabilidad de resultados otorga, seguido del modelo a escala. Sin embargo, ambas opciones requieren de una gran inversión económica además de grandes recursos humanos y materiales. Por otro lado, el tiempo de ejecución es, de hecho, una de sus mayores desventajas respecto al software.

Los resultados finales de la matriz de alternativas son las siguientes:

- Pruebas de campo: 5.
- Modelo físico a escala: 6,55.
- **Simulación software: 8,35.**

6.2 Análisis para la elección de la herramienta software

Una vez realizado la elección de la simulación software para el desarrollo del proyecto, se va realizar un estudio las diferentes alternativas de herramientas software que se pueden emplear para el diseño de la interfaz gráfica.

Las diferentes alternativas de software se muestran a continuación:

6.2.1 Microsoft Excel

Es una herramienta de Microsoft dedicada al análisis de datos que permite la realización de cálculos en hojas de cálculo. Estas, están compuestas por una serie de filas y columnas que confluyen en celdas, las cuales, serán utilizadas para almacenar datos y realizar cálculos. Además, esta herramienta permite realizar gráficas dinámicas para mostrar los datos de manera más visual.

6.2.2 PTC Mathcad

Es una herramienta específica para hacer cálculos a través de un cuaderno digital de ingeniería que permite resolver problemas de manera sencilla y visual para el usuario, ya que, no es necesario tener conocimientos de programación y se asemeja mucho a un cálculo hecho a

mano, en papel. Asimismo, es un software intuitivo que nos da la opción de realizar análisis de datos mediante graficas en 2D y 3D, pudiendo así, documentar la información de manera más visual.

6.2.3 Microsoft Visual Estudio

Es un entorno de desarrollo integrado por Microsoft enfocado en el diseño de aplicaciones. Este software implementa una amplia gama de herramientas y características que permiten el desarrollo de interfaces gráficas de manera rápida y eficiente. Todo ello, se puede implementar a través de diferentes lenguajes de programación avanzados como son C#, F#, Visual Basic y Python. Lo cual, permite un gran abanico de posibilidades, siempre y cuando, se dispongan de conocimientos de lenguajes programación.

6.2.4 Matlab

Matlab es un entorno de cómputo numérico que, a través de un lenguaje de programación propio, expresa y trabaja con los datos matemáticamente a través de matrices y vectores. Es un software muy polivalente, ampliamente utilizado a nivel profesional y en investigación, que permite visualizar los datos calculados a través de gráficas. Este, tiene una infinidad de prestaciones disponibles que van, desde entornos para la creación de apps, ya sean, *Guide* y *Appdesigner*, hasta modelar sistemas de manera interactiva con *Simulink*.

6.2.4.1 Guide

Guide, es un entorno de programación visual ofrecido por Matlab que, a pesar estar en vías de desaparecer en futuras actualizaciones de *Matlab* [6], permite, de una manera intuitiva, crear aplicaciones de manera sencilla. Para ello, incluye las herramientas básicas para el diseño de estas interfaces de usuario.

6.2.4.2 Appdesigner

Appdesigner, es una versión mejorada de *Guide* incorporada en *Matlab* como entorno para el diseño interactivo de aplicaciones para usuarios. Esta nueva versión, incorpora muchas nuevas funcionalidades en cuanto a la programación de la misma con respecto a la versión anterior [6], y una interfaz más intuitiva y atractiva para el usuario.

6.2.5 Criterios de selección

En este apartado, se procederá a la mención y explicación de los criterios que se han tenido en cuenta a la hora de realizar la selección del software que se empleará en el desarrollo del proyecto.

- **Coste de licencia del software:** en este criterio, se tiene en cuenta el coste de la licencia anual de cada software. De este modo, se valorará positivamente las licencias más baratas.
- **Flexibilidad de diseño:** en este criterio se valora la capacidad que tiene el software a nivel de herramientas de diseño de las cuales dispone para proporcionar una mejor interfaz de usuario. Se tendrá en cuenta de manera favorable, aquel software que proporcione una mayor capacidad de diseño.
- **Intuitivo y sencillo para el usuario:** Es uno de los criterios más importantes a la hora de realizar una interfaz gráfica, y consiste en lograr una interfaz que sea agradable y fácil de usar para el usuario.
- **Facilidad de la programación:** Uno de los aspectos a tener en cuenta, en cuanto al papel del programador, es de si resulta sencillo la programación de la interfaz gráfica a través del software en cuestión.
- **Compatibilidad con distintos lenguajes de programación:** otro criterio que podría ser interesante es la opción de poder programar en diferentes lenguajes y no estar limitado únicamente a uno. Por lo que, daría mayor versatilidad al software, y en caso de no tener conocimientos de un lenguaje en concreto, poder programar con otro alternativo, y así agilizar la duración del proyecto.

6.2.6 Elección de la herramienta software

Una vez mencionados los criterios de selección de la herramienta software, se va proceder a la elección del software más adecuado.

Para realizar una selección de manera cuantitativa y sencilla entre las diferentes opciones se ha empleado la matriz de priorización. En ella, por un lado, se ponderan los criterios en función de su importancia (peso) a la hora de elegir, y, por otro lado, se evalúan con una puntuación de 0 a 10 puntos cada una de las alternativas para cada criterio concreto. Donde 10 indica que se

ajusta perfectamente al criterio en cuestión mientras que 0 significa que ocurre lo completamente opuesto a lo que se evalúa con el criterio seguido.

El peso de cada uno de los criterios es el siguiente:

- Coste de licencia del software: 20%
- Flexibilidad de diseño: 25%
- Intuitivo y sencillo para el usuario: 40%
- Facilidad de la programación: 10%
- Compatibilidad con distintos lenguajes de programación: 5%

Siendo lo que más se valora capacidad del software para el diseño de una aplicación intuitiva, fácil de manejar y atractiva visualmente.

Criterios de selección	Peso	Alternativas valoradas sobre 10 puntos				
		Microsoft Excel	PTC Mathcad	Microsoft Visual Estudio	Matlab - Guide	Matlab - Appdesigner
Coste de licencia del software	20%	10	8	4	3	3
Flexibilidad de diseño	25%	3	3	7	7	10
Intuitivo y sencillo para el usuario	40%	6	6	7	8	10
Facilidad de la programación	10%	8	8	3	6	7
Compatibilidad con distintos lenguajes de programación	5%	3	3	8	4	4
TOTAL		6,1	5,7	6,05	6,35	8

Tabla 2: Matriz de priorización de herramientas software.

De la matriz de priorización, se puede llegar a la conclusión que de entre todas las opciones de software, la que mejor se adapta a los requisitos para el desarrollo de la interfaz gráfica es *Matlab*, y más concretamente el entorno de *Appdesigner*. De forma que, a pesar de ser la que tiene un coste más alto por la licencia de la misma, tiene una gran flexibilidad a la hora de diseñar aplicaciones que permiten desarrollar una interfaz más atractiva, interactiva e intuitiva para el usuario. Por otro lado, sin tampoco demasiada relevancia, es el tema de la programación, que siendo un único lenguaje exclusivo de Matlab el que se utiliza, es necesario tener conocimiento del mismo. Aunque, existe la posibilidad de uso de Matlab con lenguajes como Python [7].

En lo que concierne a las demás alternativas, las opciones de *Microsoft Excell* y *PTC Mathcad* no ofrecen una posibilidad de diseño interfaz gráfica, a pesar de ser buenas opciones para realizar y mostrar cálculos. Por otro lado, las opciones de *Microsoft Visual Estudio* y *Matlab Guide* ofrecen la posibilidad de interfaz gráfica, pero no las mismas prestaciones de diseño de interfaz como pueden ser las de *Matlab-Appdesigner*.

Las puntuaciones de cada software han sido las siguientes:

- Microsoft Excel: 6,1.
- PTC Mathcad: 5,7.
- Microsoft Visual Estudio: 6,05.
- Matlab – Guide: 6,35.
- **Matlab – Appdesigner: 8.**

7 DESCRIPCIÓN DE LA SOLUCIÓN

Una vez decidido que la herramienta software a utilizar para el desarrollo de la interfaz gráfica será *Matlab*, más concretamente el entorno *Appdesigner* que ofrece este mismo software, se hará una descripción detallada del diseño de la interfaz. Además, se realizará una descripción del entorno empleado para la el diseño de la aplicación, en este caso, de *Appdesigner*. Por último, también se explicará a través de módulos, la funcionalidad de las diferentes partes de la aplicación.

7.1 Introducción a *Appdesigner*

Appdesigner es un entorno de *Matlab* para el diseño de aplicaciones como es el caso de nuestra interfaz gráfica de usuario.

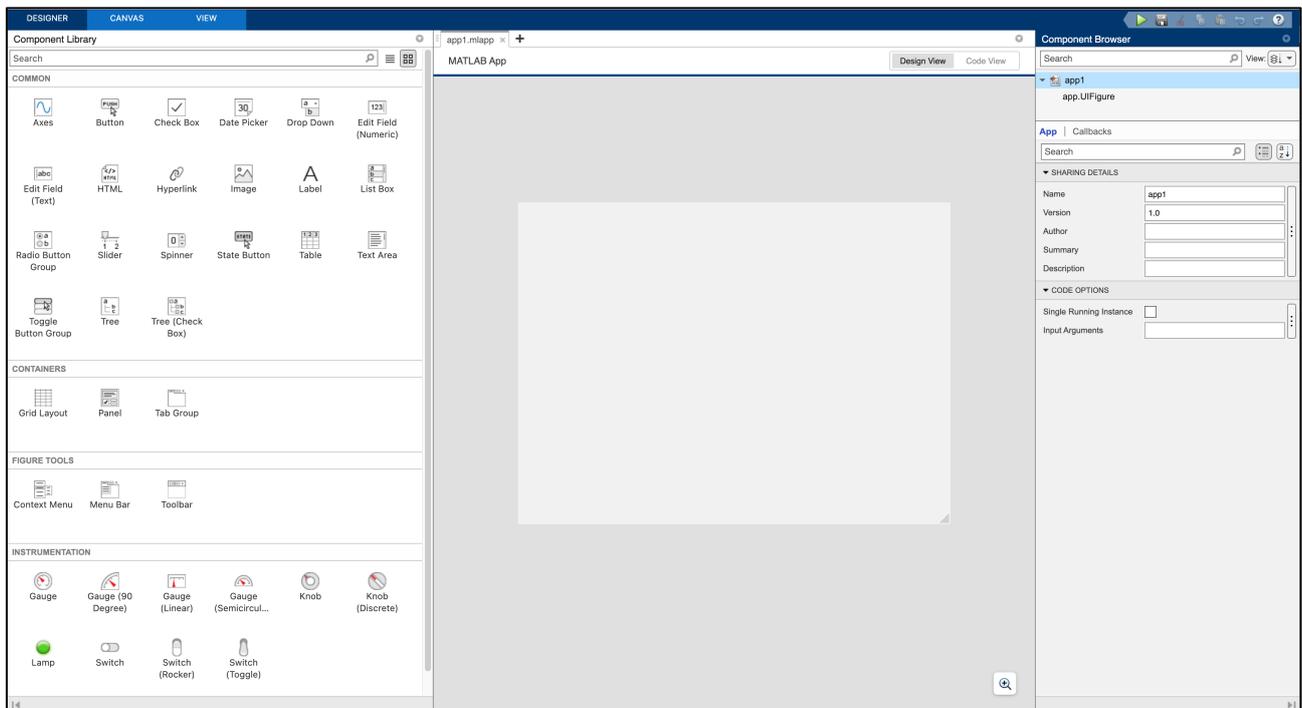


Figura 27: Entorno software de *Appdesigner*.

Como se muestra en la Figura 27, el entorno de *Appdesigner* consta de una parte central en forma de cuadrícula donde se realiza el diseño visual de la interfaz. Para ello, en el lado izquierdo, se dispone de una biblioteca de componentes o elementos interactivos que se pueden usar para crear nuestra interfaz. Estos mismos pueden ser probados ejecutando la aplicación en cualquier momento, una vez incorporados en la cuadrícula de diseño, dando así una mayor facilidad para corregir errores y mejorar la interfaz.

Entre los elementos interactivos usados para el desarrollo de la aplicación se encuentran los siguientes:

- **Label:** elemento de texto destinado a escribir títulos o información que no pueda ser modificado por el usuario.
- **Edit Field (Numeric):** elemento de texto, específicamente para trabajar con valores numéricos, que sirve tanto para entrada de datos como para mostrar datos para el usuario.
- **Button:** elemento que nos permite interactuar con la interfaz a través de un simple click del ratón, activando así la función asociada a dicho botón.
- **Check Box:** elemento que permite mantener activado o desactivado una determina funcionalidad durante un periodo ininterrumpido.
- **Radio Button Group:** elemento que permite seleccionar una opción de manera simultánea entre una serie de ellas.
- **Drop Down:** elemento desplegable en forma de lista con opciones que permite seleccionar cada una de ellas de manera individual.
- **Menu Bar:** elemento que permite seleccionar varias pestañas para una misma ventana de la interfaz.
- **Panel:** elemento en forma de cuadrícula que permite agrupar un conjunto de otros elementos para dotar de mayor claridad y orden a la interfaz.
- **Switch:** elemento que funciona como un interruptor, pudiendo dejar activado o desactivado una funcionalidad.
- **Lamp:** elemento que funciona como un indicador led, indicando activado y desactivado, o cualquier otra funcionalidad, por medio de diferentes colores de iluminación.
- **Knop:** elemento que funciona como un selector, pudiendo regular el valor de un parámetro de entrada a lo largo de un rango de valores continuo.
- **Axes:** elemento que funciona como una gráfica, permitiendo representar una serie de valores de manera visual y más atractiva.
- **Image:** elemento que permite añadir imágenes para dotar a la interfaz de mayor profesionalidad y atractivo visual.

7.2 Interfaz gráfica de usuario

La interfaz gráfica desarrollada a lo largo este proyecto nos permite calcular los parámetros característicos necesarios de la máquina síncrona para, posteriormente, poder simular el comportamiento de la misma como generador en las tres posibles situaciones de funcionamiento: vacío, cortocircuito y carga. Para todo ello, se ha elegido una distribución de la interfaz, en la cual, por un lado, en la primera pestaña llamada "TABLA DE DATOS", se recogerán los datos característicos de la máquina y se mostrarán los datos de los parámetros calculados. Por otro lado, en la segunda pestaña llamada "SITUACIONES DE FUNCIONAMIENTO", se realizarán las ya comentadas simulaciones o ensayos de funcionamiento a través de los parámetros de regulación y los parámetros ya calculados previamente. Cabe mencionar, que las simulaciones poseen dos métodos de ejecución (Figura 29), una sería de manera manual a través del botón "Puesta en marcha manual" que activaría la función de calcular los datos de salida en el momento que se pulsase el botón. La otra opción sería activar el botón "AutoUpdate" que permitiría realizar cálculos y mostrarlos simultáneamente a medida que se varían los parámetros de regulación por medio del selector.

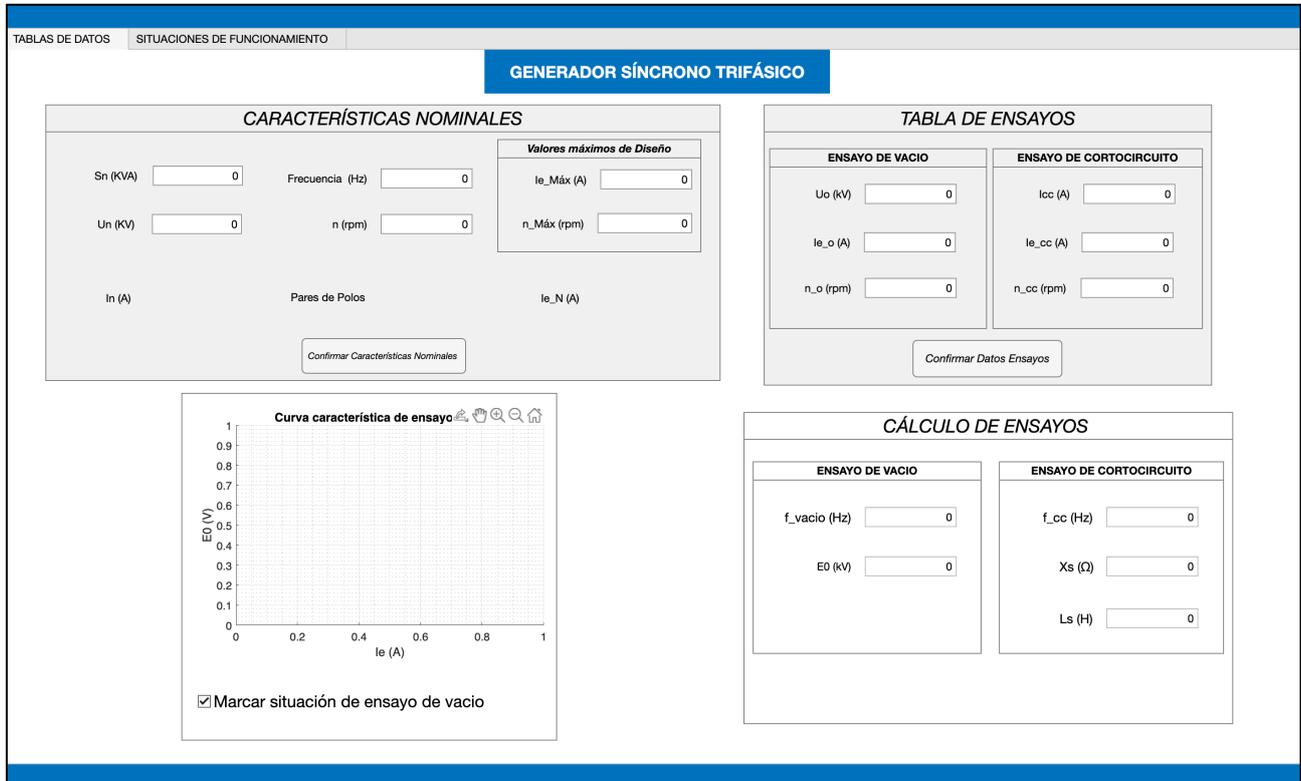


Figura 28: Interfaz. Tabla de datos. Recogida de datos y cálculo de parámetros de la máquina.

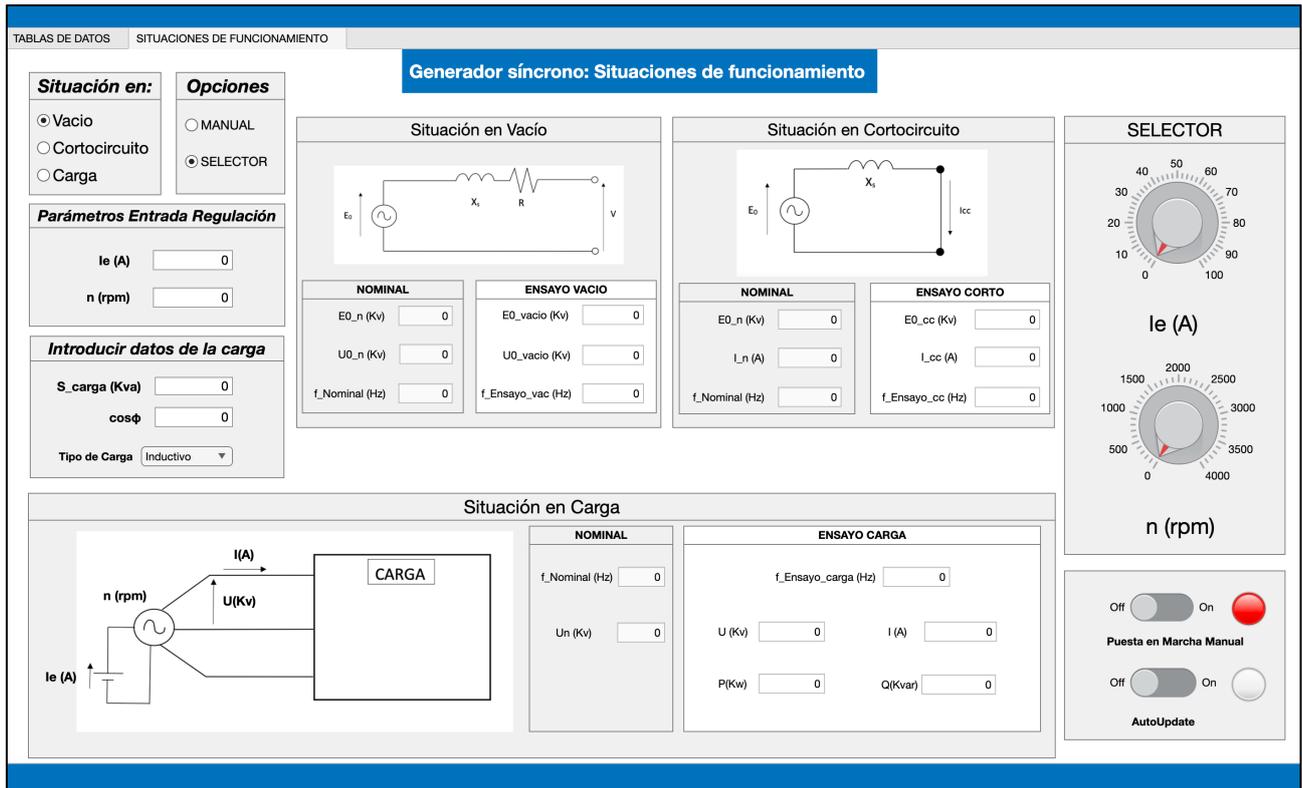


Figura 29: Interfaz. Situaciones de funcionamiento. Simulación.

En cuanto a los módulos en los que se ha dividido la interfaz podemos hablar de tres módulos bien diferenciados (Figura 30). Además, cabe aclarar que en cada módulo se ha diferenciado con color gris todos aquellos paneles que sirvan para introducir datos por el usuario o que muestran datos intermedios, que no son los valores de los parámetros finales de salida. Por otro lado, los paneles de color blanco muestran los valores de los parámetros de salida finales, que son los realmente importantes en el desarrollo de la interfaz.

En primer lugar, el módulo 1, (Figura 28) está compuesto por el apartado de "CARACTERÍSTICAS NOMINALES" donde se introducen los datos nominales de la máquina. Este módulo, incluye tanto parámetros de entrada como de salida nominales que se calcularán con los respectivos botones. Por un lado, los parámetros de entrada de la máquina consisten en la potencia aparente compleja (S_n), tensión compuesta (U_n), frecuencia (f) y velocidad de giro (n). Por otro lado, en base a diferentes datos, que se expondrán más adelante, se pueden calcular los de parámetros de salida. Entre estos, se encuentran, la intensidad por fase (I_n), los pares de polos de la máquina (p) y la intensidad de excitación (I_{e_n}).

En segundo lugar, el módulo 2, (Figura 28) está compuesto por tres paneles. Los cuales son, "TABLA DE ENSAYOS", "CÁLCULO DE ENSAYOS" y la gráfica de la curva característica de vacío. En cuanto al primer panel, es donde se introducen los datos de los ensayos de vacío y cortocircuito realizados a la máquina en cualquier condición de velocidad de giro considerada por el usuario, y por consiguiente de la frecuencia. Por lo tanto, para los ensayos de vacío y cortocircuito son necesario los siguientes parámetros de entrada. Entre ellos respectivamente se encuentran, la tensión compuesta en vacío (U_0), la intensidad de excitación en vacío (I_{e0}), la velocidad de giro en vacío (n_0), la intensidad de cortocircuito (I_{cc}), la intensidad de excitación en cortocircuito (I_{ecc}) y la velocidad de giro en cortocircuito (n_{cc}). En cuanto al segundo panel, es donde se muestran los cálculos que se realizan a partir de los datos anteriores. Entre los cálculos que se muestran, podemos encontrar respectivamente, la frecuencia del ensayo de vacío (f_{vacio}), la tensión interna por fase en vacío (E_0), la frecuencia del ensayo de cortocircuito (f_{cc}), la reactancia síncrona (X_s) y el coeficiente de autoinducción (L_s). Por último, en la gráfica se muestra la curva de vacío característica de la máquina suponiendo que no hay saturación en circuito magnético.

En tercer y ultimo lugar, el módulo 3, (Figura 29) está compuesto por una serie de paneles con diferentes funcionalidades que permiten simular diferentes parámetros en las tres condiciones de funcionamiento (vacío, cortocircuito y carga) en las que funciona el generador, a través de dos reguladores de los parámetros de entrada. Entre los parámetros de simulación, se pueden encontrar, en el caso de vacío: la tensión simple en vacío (E_{0_vacio}), la tensión compuesta en vacío (U_0) y la frecuencia del ensayo de vacío (f_{Ensayo_vacio}). En el caso de corto: la tensión simple en corto (E_{0_cc}), la intensidad por fase de corto (I_{cc}) y la frecuencia de ensayo de corto (f_{Ensayo_corto}). Finalmente, en el caso de carga, se encuentran la frecuencia del ensayo de carga (f_{Ensayo_carga}), la tensión compuesta (U), la intensidad por fase (I), la potencia activa (P) y la potencia reactiva (Q). Cabe mencionar, que en cada una de las situaciones se puede optar por funcionamiento manual o automático como se ha mencionado anteriormente.

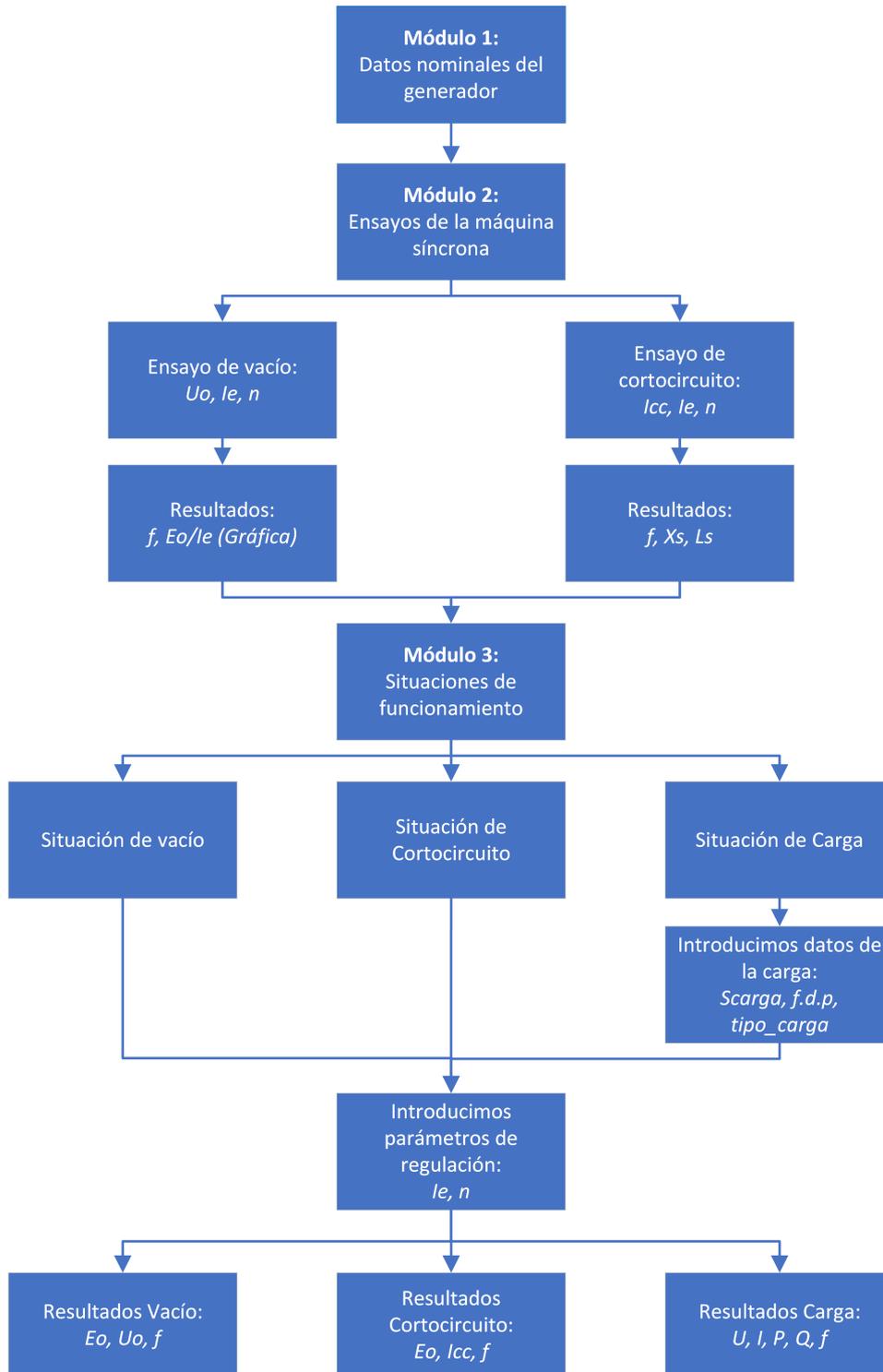


Figura 30: Diagrama de flujo de la interfaz.

7.2.1 Módulo 1: Datos nominales del generador síncrono.

CARACTERÍSTICAS NOMINALES		
Sn (KVA) <input type="text" value="0"/>	Frecuencia (Hz) <input type="text" value="0"/>	Valores máximos de Diseño I _e _Máx (A) <input type="text" value="0"/> n_Máx (rpm) <input type="text" value="0"/> I _e _N (A) <input type="text" value="0"/>
Un (KV) <input type="text" value="0"/>	n (rpm) <input type="text" value="0"/>	
In (A) <input type="text" value="0"/>	Pares de Polos <input type="text" value="0"/>	
<input type="button" value="Confirmar Características Nominales"/>		

Figura 31: Módulo 1. Características nominales del generador.

En este primer módulo podemos diferenciar 3 tipos de datos dependiendo de su funcionalidad. En primer lugar, se introducen los datos nominales de potencia aparente (S_n), tensión compuesta (U_n), frecuencia (f), y velocidad de giro (n) como datos de entrada. En segundo lugar, se introducen los datos de los parámetros de diseño (Figura 32) los cuales sirven para establecer los límites máximos de diseño de la máquina, que, en caso de ser sobrepasados, conducirán a una ventana emergente de aviso. En último lugar, están los parámetros calculados a partir de diversos datos de entrada como son la intensidad nominal por fase (I_n), los pares de polos (p) y la intensidad de excitación nominal (I_{e_n}), que se mostrarán una vez rellenados todos los campos y pulsando el botón "Confirmar Características Nominales". Para ello, se han utilizado las expresiones (1) y (8) para el cálculo de los diferentes parámetros.

Valores máximos de Diseño	
I _e _Máx (A)	<input type="text" value="0"/>
n_Máx (rpm)	<input type="text" value="0"/>

Figura 32: Valores máximos de Diseño.

Para calcular y mostrar los datos de I_n y pares de polos (p), cuando presionamos el botón de "Confirmar Características Nominales", el código del programa es el siguiente:

```
% Calculamos la In con los datos de la tabla de características
% para mostrarlo en pantalla;
```

```
SN = app.Input_Sn_CN.Value*10^3; % en kVA input y pasamos a VA;
```

```

UN = app.Input_Un_CN.Value*10^3; % en KV input y pasamos a V;
app.Output_In_CN.Visible = 'on'; % Habilitamos el Edit Field;
app.Output_Par_Polos_CN.Visible = 'on'; % Habilitamos el Edit
Field;
app.Output_In_CN.Value = (SN/(sqrt(3)*UN)); % Mostramos en
pantalla;

```

% Calculamos el número de pares de polos con los datos de frecuencia y velocidad
% de giro nominal;

```

fn = app.Input_fn_CN.Value;
n = app.Input_n_rpm_CN.Value;
app.Output_Par_Polos_CN.Value = (60*fn)/n;

```

Para calcular y mostrar el dato de I_{e_n} cuando presionamos el botón de “Confirmar datos de Ensayos” (necesitamos datos del ensayo de vacío), el código del programa es el siguiente:

```

% Calcular la Intensidad de excitación Nominal;
p = app.Output_Par_Polos_CN.Value;
fn = app.Input_fn_CN.Value;
n0 = app.Input_n_o_Ensayo.Value;
U0 = app.Input_Uo.Value*10^3; % En kV input y pasamos a V
E0_ens = U0/sqrt(3);
Ie_o = app.Input_Ie_o.Value;
m = E0_ens/Ie_o; % m=(E0/Ie_0) a frecuencia ensayo vacío;
f_ensayo_vacio = (p*n0)/60;
m1 = m*(fn/f_ensayo_vacio); % m1=(E0/Ie_0) a frecuencia nominal;
Un = app.Input_Un_CN.Value*10^3;
E0_n = Un/sqrt(3);
Ie_N = E0_n/m1;
app.Output_Ie_n_CN.Visible = 'on';
app.Output_Ie_n_CN.Value = Ie_N;

```

A continuación, se muestra un esquema (Figura 33) donde se establece la relación entre los datos de salida y los datos de entrada del módulo 1, necesarios para calcular dichos datos, que puede ayudar a mejorar la comprensión del código de programa descrito con anterioridad:

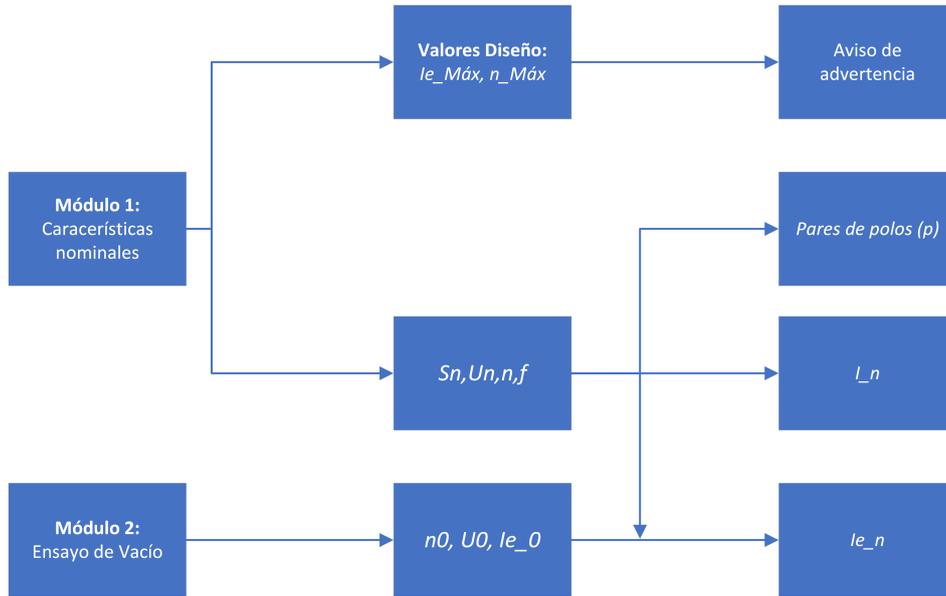


Figura 33: Esquema Módulo 1.

7.2.2 Módulo 2: Ensayos de la máquina síncrona.

Este segundo módulo, está compuesto de 3 diferentes paneles. En el primero de ellos, la "TABLA DE ENSAYOS", es donde se recogen tanto los datos de entrada de los ensayos de vacío como los de cortocircuito mencionados anteriormente (Figura 34). Por otro lado, una vez recogidos los datos, se realizan los cálculos pertinentes para así mostrarlos en los dos otros paneles restantes, una vez pulsado el botón de "Confirmar Datos Ensayos". Entre esos paneles se encuentran, el "CÁLCULO DE ENSAYOS" y la gráfica "Curva característica de ensayo de vacío" (Figura 35).

TABLA DE ENSAYOS		CÁLCULO DE ENSAYOS	
ENSAYO DE VACÍO Uo (kV) <input type="text" value="0"/> Ie_o (A) <input type="text" value="0"/> n_o (rpm) <input type="text" value="0"/>	ENSAYO DE CORTOCIRCUITO Icc (A) <input type="text" value="0"/> Ie_cc (A) <input type="text" value="0"/> n_cc (rpm) <input type="text" value="0"/>	ENSAYO DE VACÍO f_vacio (Hz) <input type="text" value="0"/> E0 (kV) <input type="text" value="0"/>	ENSAYO DE CORTOCIRCUITO f_cc (Hz) <input type="text" value="0"/> Xs (Ω) <input type="text" value="0"/> Ls (H) <input type="text" value="0"/>
<input type="button" value="Confirmar Datos Ensayos"/>			

Figura 34: Módulo 2. Recogida y muestra de datos de ensayos de la máquina síncrona.

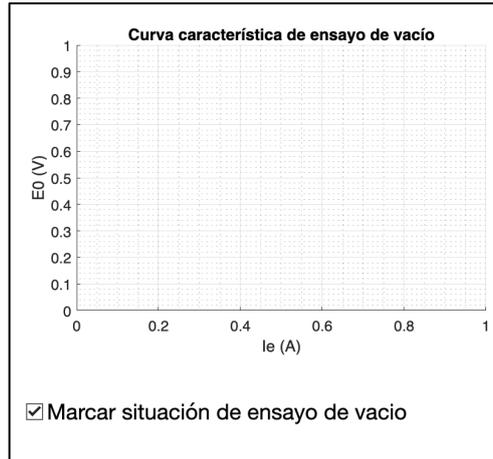


Figura 35: Módulo 2. Gráfica de ensayo de vacío.

En cuanto a la forma en la que se realizan los cálculos, se puede destacar como de cada ensayo, ya sea, de vacío o cortocircuito, se realizan una serie de cálculos de forma independiente de cada ensayo, como se muestra en la Figura 36, mostrando de esta manera en el panel de "CÁLCULO DE ENSAYOS" cada resultado correspondiente a cada ensayo por separado (Figura 36). Además, para calcular la gráfica (Figura 35) solo es necesario datos del ensayo de vacío, como su propio nombre indica. Todo ello, se ejecuta una vez pulsado el botón de "Confirmar Datos Ensayo".

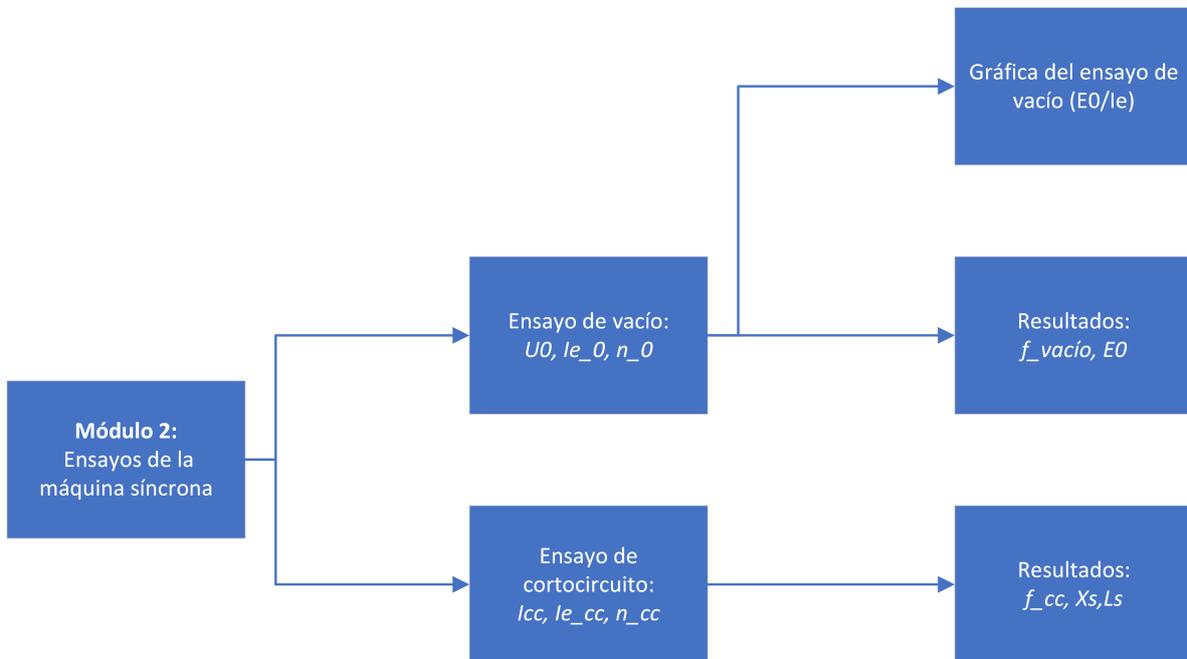


Figura 36: Esquema Módulo 2.

En lo que respecta a los cálculos realizados de cada ensayo a través del lenguaje de programación, se emplean las expresiones (1), (3) y (4), vistas anteriormente en la teoría.

Por un lado, en las primeras líneas de código se muestran como se recogen los datos necesarios para los cálculos de los respectivos parámetros de los ensayos, para posteriormente, mostrar esos resultados correspondientes de f_{cc} , X_s , L_s , $f_{vacío}$ y E_0 . Cabe mencionar, que cada ensayo se realiza a una frecuencia determinada, por lo que, para la realización de los cálculos será necesario hacer la conversión de los datos a otra frecuencia dependiendo del caso. Por otro lado, para calcular la pendiente de la curva del ensayo de vacío (E_0/I_e), también hay que tener en cuenta que los datos de partida se realizan a una frecuencia de ensayo de vacío. Se adjunta el código de programación asociado al botón "Confirmar Datos Ensayos" a continuación:

```
% --Mostrar Datos en el circuito Equivalente en condiciones nominales: Xs y E0--

%Recopilación de Datos;

U0 = app.Input_Uo.Value*10^3; %En kV input y pasamos a V;
E0_ens = U0/sqrt(3); % Calculamos tensión simple de ensayo;
Ie_o = app.Input_Ie_o.Value;
m = E0_ens/Ie_o; % m=(E0/Ie_0) a frecuencia ensayo vacío;
p = app.Output_Par_Polos_CN.Value; % Recuperamos el dato calculado
de pares de polos;
n0 = app.Input_n_o_Ensayo.Value;
f_ensayo_vacio = (p*n0)/60;

% Recopilar datos de Input necesarios para cálculos más abajo;

fn = app.Input_fn_CN.Value;
    % Datos a frecuencia de ensayo de CC;
Icc = app.Input_Icc.Value;
n_cc = app.Input_n_cc_Ensayo.Value;
Ie_cc = app.Input_Ie_cc_Ensayo.Value;

% Cálculo del ensayo de CC;
f_ensayo_cc = (p*n_cc)/60;
m1 = m*(fn/f_ensayo_vacio); % m1=(E0/Ie_0) a frecuencia nominal;
m2 = m1*(f_ensayo_cc/fn); % Pasar de pendiente m1(fn) a
m2(f_ensayo_cc)
E0cc = m2*Ie_cc; % A frecuencia de ensayo de cc;
Xs_cc = E0cc/Icc; % Reactancia síncrona a frecuencia de ensayo de
cc;
Ls = Xs_cc/(2*pi*f_ensayo_cc); % La inductancia síncrona es una
constante;
```

```
% Datos de salida de CÁLCULOS DE ENSAYOS;
```

```
app.Output_f_vacio.Value = f_ensayo_vacio;
app.Output_Eo.Value = app.Input_Un_CN.Value/sqrt(3);
app.Output_f_cc.Value = f_ensayo_cc;
app.Output_Xs.Value = Xs_cc;
app.Output_Ls.Value = Ls;
```

En el caso del código de programación de los resultados que se muestran en la gráfica, tiene algunas peculiaridades que habría que comentar. Es decir, dependiendo de si la frecuencia del ensayo de vacío es la nominal o no, se realizan una o dos curvas de vacío (Figura 37). En el primer caso, si la frecuencia del ensayo de vacío coincide con la frecuencia nominal, se mostrará una única función lineal del ensayo, a frecuencia del mismo. En el segundo caso, cuando la frecuencia del ensayo no corresponde con la nominal, se mostrará una función lineal para cada una de las dos frecuencias mencionadas, que tendrán diferentes pendientes cada una. Además, para ambos casos, existe la posibilidad de graficar el punto del ensayo de vacío a través del checkbox llamado "Marcar situación de ensayo de vacío", superponiendo el punto con la "curva" de vacío. Todo ello, válido para los dos casos mencionados anteriormente.

A continuación, se muestra el código del programa asociado a la gráfica:

```
% -----Calcular la curva de ensayo de vacío-----
x = 0:0.01:100;
y = m*x;

% Comprobamos si el ensayo se realiza a frecuencia nominal o
% distinta;

if fn == f_ensayo_vacio

fn_string = num2str(fn);

% Mostrar punto del ensayo vacío cuando activemos el checkbox;

Value_vacio = app.Situacion_Vacio_CheckBox.Value;

if Value_vacio == 1

plot(app.UIAxes,x,y,"b-",Ie_o,E0_ens,"r*");
legend(app.UIAxes,"fn = "+fn_string+" (Hz)","Situación de
Ensayo Vacío");

else

plot(app.UIAxes,x,y,"b-");
legend(app.UIAxes,"fn = "+fn_string+" (Hz)");

end
```

```
else
```

```
    % Calculamos la pendiente de  $m'=(E0/Ie_0)$  a frecuencia
    % nominal, ya una vez conocido  $m=(E0/Ie_0)$  a frecuencia de
    % ensayo de vacío;
```

```
% Aunque m1 está calculado previamente, es para que se entienda
mejor siguiendo con el paso anterior en el que se ha usado m;
```

```
m1 = m*(fn/f_ensayo_vacio);
y1 = m1*x;
```

```
fn_string = num2str(fn);
f_ensayo_string = num2str(f_ensayo_vacio);
```

```
Value2 = app.Situacion_Vacio_CheckBox.Value;
if Value2 == 1
```

```
    plot(app.UIAxes,x,y,"b-",x,y1,"g-",Ie_o,E0_ens,"r*");
    legend(app.UIAxes,"fensayo = "+f_ensayo_string+" (Hz)", ...
        "fn = "+fn_string+" (Hz)","Situación de Ensayo Vacío")
```

```
else
```

```
    plot(app.UIAxes,x,y,"b-",x,y1,"g-");
    legend(app.UIAxes,"fensayo = "+f_ensayo_string+" (Hz)","fn =
"+fn_string+" (Hz)");
```

```
end
```

```
end
```

Por último, se muestra un ejemplo de los dos casos en los que se grafican las curvas de vacío:

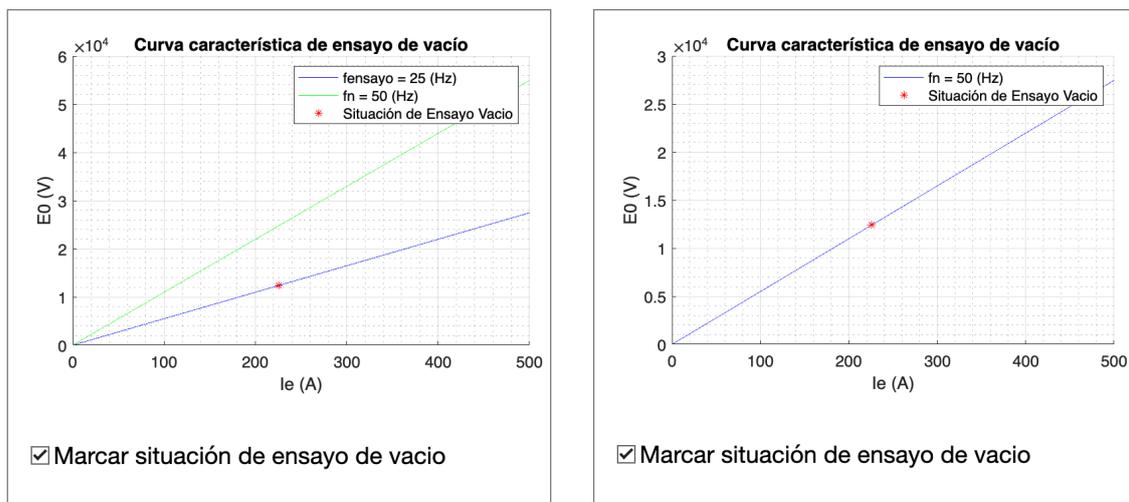


Figura 37: Ejemplo gráfica del ensayo de vacío. Caso 1 y 2.

7.2.3 Módulo 3: Situaciones de funcionamiento.

En este tercer módulo se agrupan tres diferentes situaciones de funcionamiento del generador: vacío, cortocircuito y carga. Las cuales se podrán simular de manera individual a través de los parámetros de regulación.

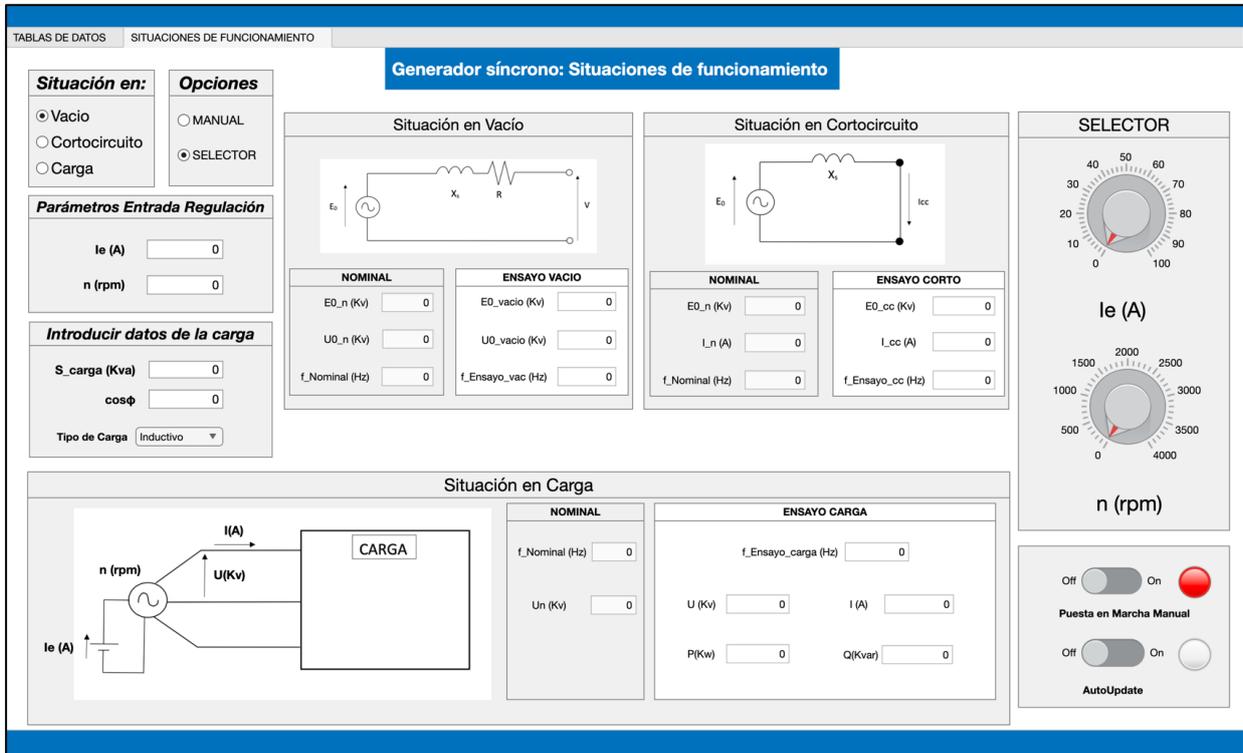


Figura 38: Módulo 3. Área de simulación de la interfaz gráfica.

Para entender mejor como funciona este módulo, primero debemos saber que algunos de los datos introducidos y calculados sobre la máquina, ya sean parámetros nominales o internos del circuito equivalente de los módulos anteriores, serán imprescindibles. De hecho, cuando abrimos la pestaña de situaciones de funcionamiento, los datos nominales de cada situación de funcionamiento quedan definidos con los datos de los módulos 1 y 2. Entre estos datos nominales podemos encontrar la tensión de vacío (E_0), la tensión compuesta (U_0), la frecuencia (f) y la intensidad por fase (I_n).

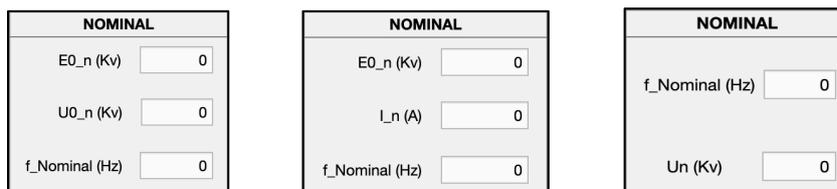


Figura 39: Parámetros nominales de las situaciones de funcionamiento.

Una vez definidos los parámetros nominales es hora de seleccionar el tipo de funcionamiento a través del *radio button group*:

Situación en:
<input checked="" type="radio"/> Vacío
<input type="radio"/> Cortocircuito
<input type="radio"/> Carga

Figura 40: Radio button group. Situaciones de funcionamiento.

Este *radio button group* lleva asociado un código de programación para que se habiliten los paneles correspondientes al uso que se le va a dar, dependiendo de la opción seleccionada, y deshabilitar los paneles que no se vayan a usar, de forma que sea más fácil e intuitivo para el usuario.

A continuación, se muestra el código del programa:

```
vacio = app.Vacio_Button.Value;
corto = app.Cortocircuito_Button.Value;
carga = app.Carga_Button.Value;

if vacio == 1
    % Dejamos como no visible la pestaña de "Datos de la carga";
    app.Introducir_Datos_Carga_Panel.Visible = "off";
    % Deshabilitamos el resto de paneles
    app.SituacionVacioPanel.Enable = "on";
    app.Situacion_Cortocircuito_Panel.Enable = "off";
    app.Situacion_Carga_Panel.Enable = "off";
elseif corto == 1
    % Dejamos como no visible la pestaña de "Datos de la carga";
    app.Introducir_Datos_Carga_Panel.Visible = "off";
    % Deshabilitamos el resto de paneles;
    app.SituacionVacioPanel.Enable = "off";
    app.Situacion_Cortocircuito_Panel.Enable = "on";
    app.Situacion_Carga_Panel.Enable = "off";

elseif carga == 1
    % Activamos la pestaña de "Datos de la carga";
    app.Introducir_Datos_Carga_Panel.Visible = "on";
    % Deshabilitamos el resto de paneles;
    app.SituacionVacioPanel.Enable = "off";
    app.Situacion_Cortocircuito_Panel.Enable = "off";
    app.Situacion_Carga_Panel.Enable = "on";
end
```

Se muestran más abajo los ejemplos de como se vería la interfaz en cada uno de los tres casos de selección de vacío, cortocircuito y carga, cuando se ejecuta el código mostrado anteriormente.

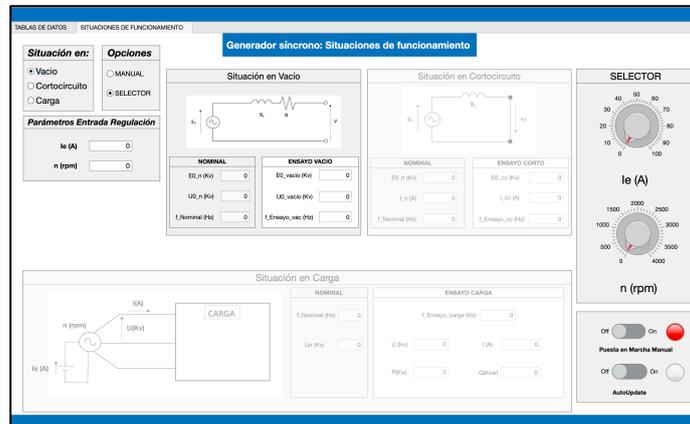


Figura 41: Situación de vacío. Interfaz.

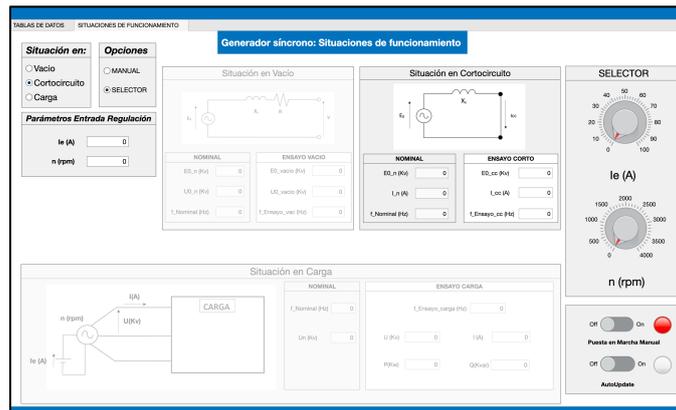


Figura 42: Situación de cortocircuito. Interfaz.

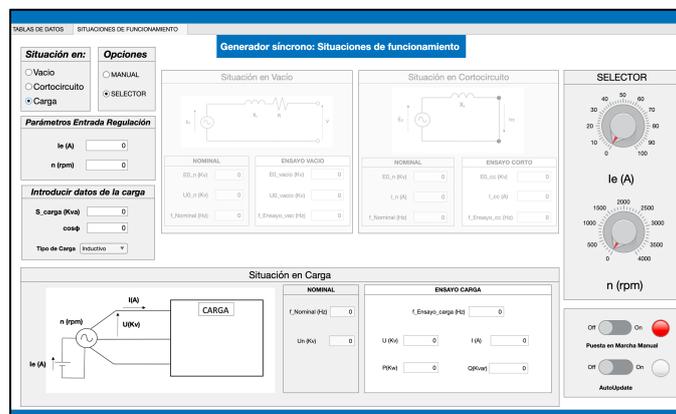
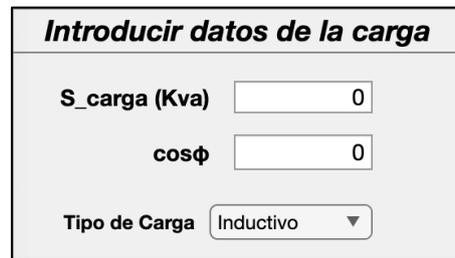


Figura 43: Situación de carga. Interfaz.

En caso de seleccionar la opción de carga habría que introducir los datos de la carga a través del panel "Introducir datos de la carga". Entre estos datos podemos encontrar, la potencia aparente consumida por la carga (S_{carga}), el factor de potencia ($\cos\phi$), y el tipo de carga.



Introducir datos de la carga

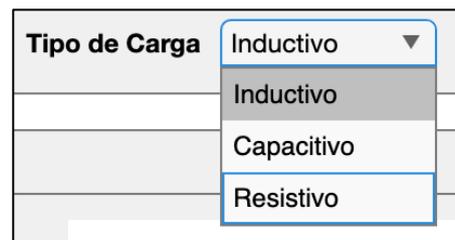
S_carga (Kva)

cosφ

Tipo de Carga

Figura 44: Panel. Introducir datos de la carga.

A continuación, se muestra el menú de lista de opciones para la selección del tipo de carga, que puede variar entre inductivo, capacitivo y resistivo.



Tipo de Carga

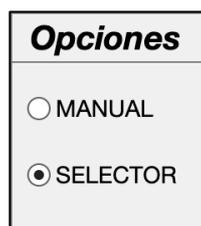
- Inductivo
- Inductivo
- Capacitivo
- Resistivo

Figura 45: Drop down menu. Tipo de carga.

7.2.3.1 Parámetros de regulación

Antes de realizar las simulaciones de las situaciones de funcionamiento es necesario definir en que condiciones de regulación se encuentra el generador. Para ello, hay que definir los parámetros de regulación, que consisten en la intensidad de excitación del rotor (I_e) y la velocidad de giro (n).

Para regular estos dos parámetros existen dos posibilidades, hacerlo de forma manual introduciendo los datos o con un selector. Para ello, hay que seleccionar la opción que deseemos en el siguiente panel.



Opciones

MANUAL

SELECTOR

Figura 46: Opciones de entrada de parámetros de regulación.

Una vez seleccionado la opción de manual o selector, se habilitará la opción correspondiente y se deshabilitará la otra opción restante.

A continuación, se muestra el código asociado a esta funcionalidad:

```

manual = app.Manual_Button.Value;
selector = app.Selector_Button.Value;

if manual == 1
    % Desactivamos el selector;
    app.SELECTORPanel.Enable = "off";
    % Activamos la opción editable de los cuadros de texto de los
    parámetros de regulación

    app.Input_regulacion_Ie.Editable = "on";
    app.Input_regulacion_n.Editable = "on";

elseif selector == 1
    % Activamos el selector;
    app.SELECTORPanel.Enable = "on";
    % Desactivamos la opción editable de los cuadros de texto de los
    parámetros de regulación
    app.Input_regulacion_Ie.Editable = "off";
    app.Input_regulacion_n.Editable = "off";

end

```

Las opciones de manual y selector corresponden a los paneles "Parámetros Entrada Regulación" y "Selector" respectivamente, como se muestran en las figuras 47 y 48.

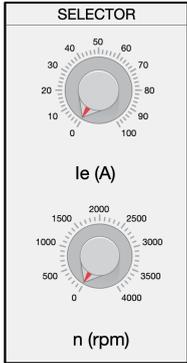


Parámetros Entrada Regulación

Ie (A)

n (rpm)

Figura 47: Parámetros de regulación. Manual.



SELECTOR

Ie (A)

n (rpm)

Figura 48: Parámetros de regulación. Selector.

7.2.3.2 Simulación de situaciones de funcionamiento

Habiendo seleccionado los parámetros de regulación se puede proceder a la simulación de cualquiera de las tres situaciones de funcionamiento de una en una. Para ello, se dispone de un panel de puesta en marcha (Figura 49) a través de la cual se puede arrancar la simulación de dos maneras, de forma manual y de forma automática.

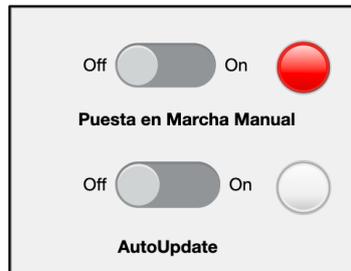


Figura 49: Panel de puesta en marcha.

En el caso de la opción de puesta en marcha manual, simplemente se activa el led de color verde (Figura 50), se calculan los datos necesarios internamente y se muestran en pantalla, en la situación de funcionamiento elegida anteriormente. Una vez que se desea desactivar, se acciona de nuevo el botón y se pone en "Off". De esta manera, se ponen todos los campos con valor inicial nulo y se activa de nuevo el led rojo. Aunque esta funcionalidad cumple perfectamente con el objetivo del proyecto, sería mucho más útil poder variar los parámetros de entrada y que automáticamente, sin tocar ningún botón, se vayan calculando los datos de salida para poder ver en tiempo real la simulación. Es por ese motivo que añade otra funcionalidad de puesta en marcha automática.

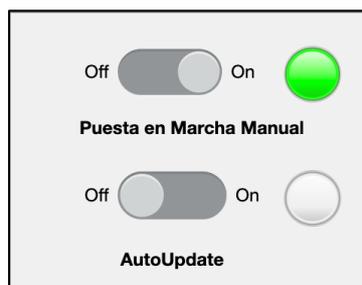


Figura 50: Puesta en marcha manual. Activado.

En este último caso de puesta en marcha automática, que va ligado al botón "AutoUpdate", permite simular en tiempo real, a través de la regulación de los parámetros de entrada, por medio de los reguladores del selector (Figura 48). De este modo, a medida que se hace girar los selectores, los valores de los parámetros de entrada cambian y, en consecuencia, también lo hacen los parámetros de salida calculados. Cabe mencionar, que cuando se activa la

funcionalidad, el led se ilumina de color azul y se desactiva la opción de marcha manual. Mientras tanto, cuando se desactiva, el led se vuelve gris y se habilita de nuevo la opción de marcha manual.



Figura 51: Puesta en marcha automático. Activado.

Para cada una de las tres situaciones de funcionamiento, el código asociado al cálculo de las misma se recoge dentro de la función "Calcular_Situaciones_Funcionamiento". Por lo tanto, para una mejor comprensión, se va a dividir para cada caso, con su respectivo código, una explicación de como se han realizado los cálculos. Cabe mencionar que para la realización de los siguientes cálculos se utilizan las expresiones (1), (2), (4), (5), (6), (7) y (9).

Para comenzar, se recogen los datos necesarios para los cálculos, entre los que están, los pares de polos (p), y los parámetros de regulación I_e y n .

```
p = app.Output_Par_Polos_CN.Value;
n = app.Input_regulacion_n.Value;

f_val_entrada = (p*n)/60; % Pasamos el dato de n(rpm) a f(Hz)
Ie_val_entrada = app.Input_regulacion_Ie.Value;

U0 = app.Input_Uo.Value*10^3; % En kV el input y pasamos a V;
E0_ens = U0/sqrt(3);
% Calculamos la relación (E0/Ie) a frecuencia ensayo de vacío
% en la variable m;

Ie_o = app.Input_Ie_o.Value;
m = E0_ens/Ie_o;
n0 = app.Input_n_o_Ensayo.Value;
f_ensayo_vacio = (p*n0)/60;
```

Una vez calculado la pendiente de la curva de vacío (E_0/I_e) para la frecuencia del ensayo de vacío, recalculamos la pendiente para la frecuencia impuesta por la velocidad de giro (n) de los

parámetros de regulación. Para más tarde, calcular la E_0 a través de la I_e , de los parámetros de regulación y la pendiente (E_0/I_e) calculada con anterioridad.

```
% Calculamos la relación (E0/Ie) a frecuencia de la variable de
% entrada en la variable m_val en función de m y las
% frecuencias respectivas;
```

```
m_val = m*(f_val_entrada/f_ensayo_vacio);
E0_val = m_val*Ie_val_entrada;
```

A continuación, se realizan los cálculos específicos de una de las tres situaciones elegidas que se muestran:

7.2.3.2.1 Situación de vacío

En primer lugar, en la situación de vacío se muestran los valores de E_0 y f calculados por medio de los parámetros de regulación elegidos, y se calcula la U_0 con la ecuación (9), para ser mostrada junto con los otros datos.

```
if app.Vacio_Button.Value == 1

U0_val=sqrt(3)*E0_val;
app.Output_U0_vacio.Value = U0_val/(10^3); %En Kv;
app.Output_E0_vacio.Value = E0_val/(10^3); %En Kv;
app.Output_f_Ensayo_vacio.Value = f_val_entrada;
```

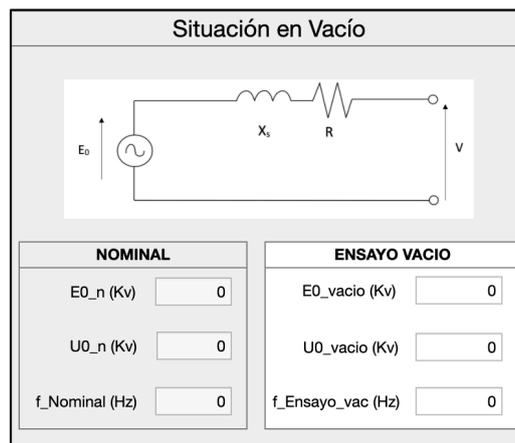


Figura 52: Panel de situación de vacío. Simulación: ensayo de vacío.

7.2.3.2.2 Situación de cortocircuito

En segundo lugar, en la situación de cortocircuito se reutiliza el valor de la inductancia síncrona (L_s) calculada previamente para calcular la reactancia síncrona (X_s), con la ecuación (4), a la frecuencia impuesta por el parámetro de regulación (n). Asimismo, se calcula la corriente de cortocircuito (I_{cc}) con la ecuación (2) a través los datos de E_o y X_s , ya calculados previamente. Por lo tanto, por ultimo se muestran los datos de E_o , I_{cc} y f en sus respectivos campos o casillas (Edit Field).

```
elseif app.Cortocircuito_Button.Value == 1

    Ls=app.Output_Ls.Value; % Dato de inductancia síncrona calculado
    previamente en tabla de datos;
    Xs=2*pi*f_val_entrada*Ls; % Reactancia síncrona a frecuencia del
    regulador n;
    Icc=E0_val/Xs;

    app.Output_E0_cc.Value = E0_val/(10^3); %En Kv;
    app.Output_I_cc.Value = Icc;
    app.Output_f_Ensayo_cc.Value = f_val_entrada;
```

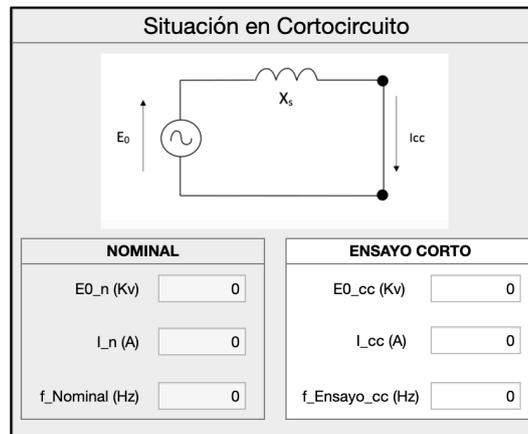


Figura 53: Panel de situación de cortocircuito. Simulación: ensayo de corto.

7.2.3.2.3 Situación de carga

En último lugar, en la situación de carga, una vez introducido los datos de la carga, como se ha comentado en apartados anteriores, se proceden con los cálculos y la muestra de datos, a partir de los parámetros de regulación.

Lo primero de todo, se recogen los datos ya calculados en etapas anteriores como (L_s) y se muestra la f (Figura 54) que se ha calculado al inicio con los parámetros de regulación. En cuanto al resto de datos, habrá que realizar otros cálculos específicos. Para ello, se recogen los datos de la carga: Un (suponiendo que alimentamos la carga a tensión nominal), $Scarga$ y $\cos\varphi$. Luego, en función del tipo de carga, se calcula el ángulo φ considerándolo positivo o negativo.

Con todo ello, se calcula la impedancia de la carga (Z_{carga}) con las ecuaciones (8) y (9), que servirá para calcular Z_{total} , incluyendo de esta manera el efecto de X_s . Por lo tanto, con los datos de los parámetros de entrada y los datos ya calculados, se pueden calcular ya los parámetros de salida del ensayo de carga con las ecuaciones (4), (6), (7), (8), que serán mostrados (Figura 54). Entre estos parámetros de salida se encuentran: la tensión de alimentación de la carga regulada con los parámetros de regulación (U), la intensidad por fase que alimenta la carga (I), la potencia activa consumida por la carga (P) y la potencia reactiva consumida por la carga (Q).

```

else
    % Cuando es situación de carga;

    Ls = app.Output_Ls.Value; % Dato de reactancia síncrona
calculado previamente en tabla de datos;
    app.Output_f_Ensayo_Carga.Value = f_val_entrada;

    -----%Datos necesarios para los cálculos-----

    Un = app.Input_Un_CN.Value*10^3; % Suponemos que alimentamos la
carga a tensión Nominal Un (Kv)
    S_carga = app.Input_Scarga.Value*10^3; % Scarga en (kVA)y
pasamos a VA
    Factor_Potencia = app.Input_Factor_Potencia.Value;
    % Por defecto ponemos que si cosphi=1 entonces es una
    % carga Resistiva pura

    if Factor_Potencia == 1
        app.TipodeCargaDropDown.Value = "Resistivo";
    end

```

```

% Calculamos el ángulo phi y le añadimos el signo en
% función de si es capacitiva (-) o inductiva (+)
Tipo_carga = app.TipodeCargaDropDown.Value;

if Tipo_carga == "Inductivo"

    phi = acos(Factor_Potencia); % phi en radianes (+)

elseif Tipo_carga == "Capacitivo"

    phi = (-1)*acos(Factor_Potencia); % phi en radianes (-)

else % Caso resistivo
    phi=0;
end

% Datos de la carga por fase para calcular Ztotal por
% fase;
I_carga = S_carga/(sqrt(3)*Un);
Z_carga = (Un/sqrt(3))/I_carga;
R_carga = Z_carga*cos(phi);
X_carga = Z_carga*sin(phi);

% Calculamos la Ztotal
w = 2*pi*f_val_entrada; % Frecuencia variable en rad/s
Xs = w*Ls;
Rtotal = R_carga;
Xtotal = X_carga+(Xs);
Ztotal = sqrt((Rtotal^2)+(Xtotal^2));
% Calculamos la intensidad real por fase;
I_carga_real = E0_val/Ztotal;
app.Output_I_Carga.Value = I_carga_real;
% Calculamos la tensión de la carga Ureal(Kv);
V = I_carga_real*Z_carga;
U = sqrt(3)*V;
app.Output_UKv_Carga.Value = U/(10^3);% Pasamos de V a KV

% Calculamos la potencia activa P(Kw);
P = sqrt(3)*U*I_carga_real*cos(phi);
app.Output_PKw_Carga.Value = P/(10^3); %En Kw;

% Calculamos la potencia reactiva Q(KVAr);

Q = sqrt(3)*U*I_carga_real*sin(phi);
app.Output_QKvar_Carga.Value = Q/(10^3);% En kVAr;

end

```

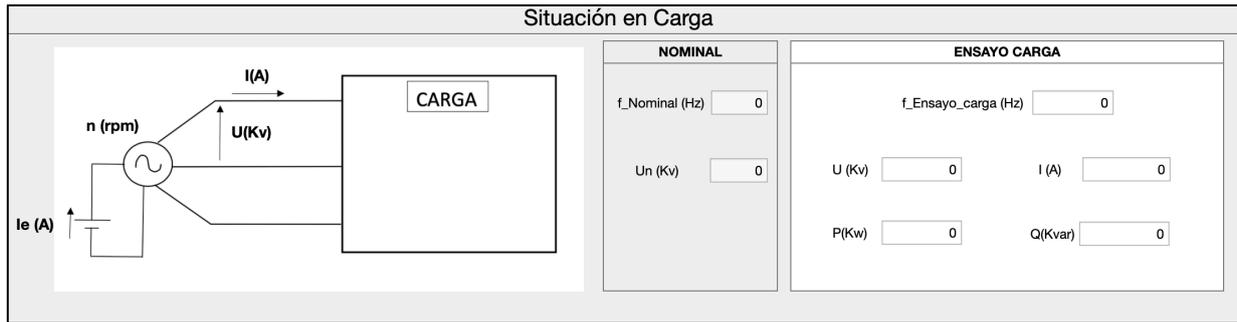


Figura 54: Panel de situación de carga. Simulación: ensayo de carga.

Para que se pueda entender mejor la funcionalidad del módulo 3, se muestra un esquema a continuación:

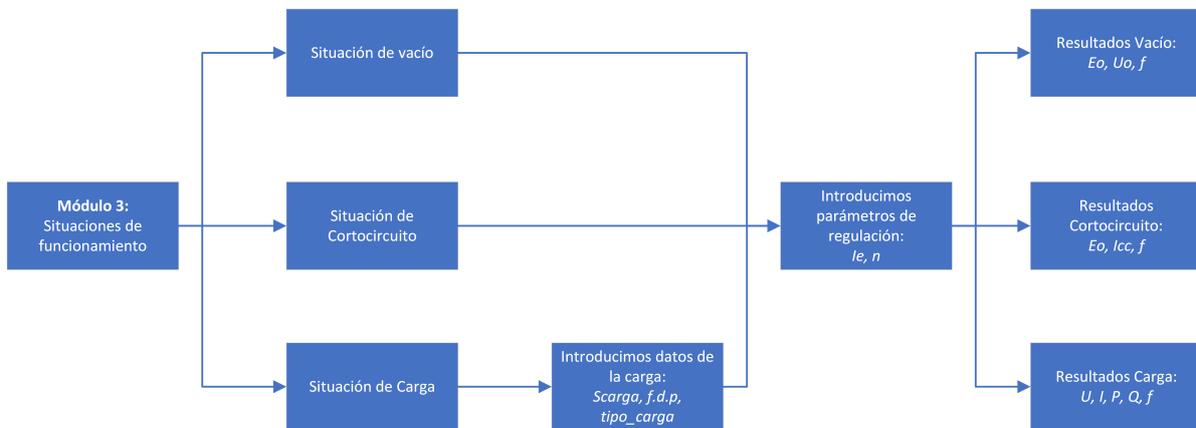


Figura 55: Esquema Módulo 3.

7.2.4 Mensajes de advertencia y error.

Como paso final, cuando se realiza la simulación, existe unas condiciones en las que muestran ventanas de advertencia. Estas condiciones están ligadas a los parámetros de regulación, que no pueden sobrepasar unos valores límite establecidos por diseño de la máquina, que son algo mayores que los valores nominales.

Estos mensajes de advertencia se activan cuando se realizan las simulaciones, activando cualquiera de los dos botones de marcha manual o automática, y sí se supera los valores máximos de diseño (Figura 56).

Valores máximos de Diseño	
I _e _Máx (A)	<input style="width: 80px;" type="text" value="80"/>
n_Máx (rpm)	<input style="width: 80px;" type="text" value="3000"/>

Figura 56: Ejemplo. Valores máximos de diseño.

Si se superan los valores máximos, hay dos casos diferenciados. El caso 1 sería cuando se supera la intensidad de excitación máxima ($I_{e_{máx}}$), mientras que el caso 2 sería cuando se supera la velocidad de giro máxima ($n_{máx}$). El mensaje de advertencia es personalizado para cada caso, sin embargo, en los dos casos el led cambia a color amarillo para indicar la advertencia.

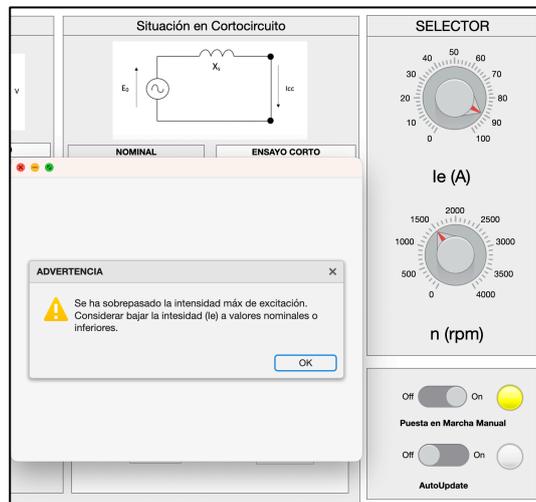


Figura 57: Mensaje de advertencia. Caso 1.

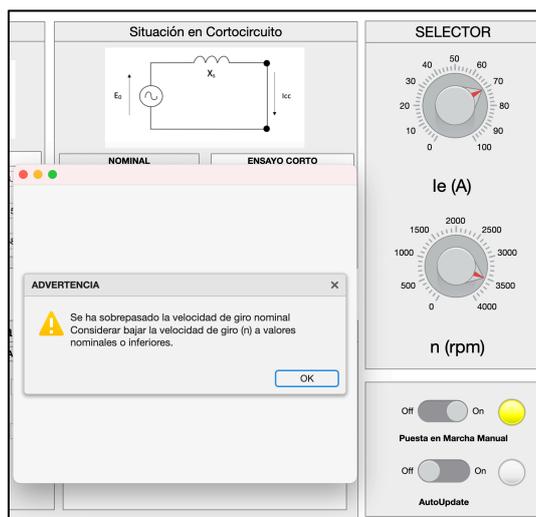


Figura 58: Mensaje de advertencia. Caso 2.

A continuación, se muestra el código de programación de los mensajes de advertencia:

```
value = app.Interruptor_Marcha_Manual.Value;

if strcmp(value, 'On') % Si el interruptor de marcha encendido;

    [error1,error2]=Evaluacion_Errores(app); %Llamamos a la función que nos
    devuelve el tipo de error

    if error1 == 1

        mensaje = {'Se ha sobrepasado la intensidad máx de
excitación.', 'Considerar bajar la intesidad (Ie) a valores nominales o
inferiores.'};

        titulo = 'ADVERTENCIA';
        uialert(uifigure,mensaje,titulo,'Icon','warning');

        app.LED_Marcha_Manual.Color = 'yellow'; % Encendemos el LED
en amarillo del botón manual;
        Calcular_Situaciones_Funcionamiento(app);

    elseif error2 == 1

        mensaje = {'Se ha sobrepasado la velocidad de giro
nominal', 'Considerar bajar la velocidad de giro (n) a valores nominales o
inferiores.'};

        titulo = 'ADVERTENCIA';
        uialert(uifigure,mensaje,titulo,'Icon','warning');

        app.LED_Marcha_Manual.Color = 'yellow'; % Encendemos el LED
en amarillo del botón manual;
        Calcular_Situaciones_Funcionamiento(app);

    else % Si no hay errores

        Calcular_Situaciones_Funcionamiento(app); % Llamamos a la función
para hacer los cálculos y presentarlos;
        app.LED_Marcha_Manual.Color = 'green'; % Encendemos el LED en
verde;
    end

    else % Si el interruptor de marcha manual está apagado;
        app.LED_Marcha_Manual.Color = 'red'; % Encender el LED rojo;
        Limpiar_Situaciones_Funcionamiento(app); % Llamamos a la función
para volver a la situación de inicio;
    end
```

8 EJEMPLOS DE APLICACIÓN

En este apartado se va a mostrar el funcionamiento de la interfaz gráfica mediante unos ejemplos prácticos. Para ello, se resolverán una serie de problemas que se adecuen a las condiciones de funcionamiento de la interfaz y se irán resolviendo paso a paso.

8.1 Ejemplo 1

Un generador síncrono trifásico de rotor cilíndrico de 45 kVA, 230 V, 1000 rpm, 50 Hz y conexión estrella.

Se realizan los ensayos de vacío y cortocircuito en las citadas condiciones nominales ($n_0 = n_{cc} = 1500$ rpm):

- **Ensayo de vacío:**
 - U_o : 230 V
 - I_e : 2,84 A
- **Ensayo de cortocircuito:**
 - I_{cc} : 118 A
 - I_e : 2,20 A

Despreciando, a efectos de cálculo, la influencia de la saturación y la resistencia por fase del estator (R). Se pide:

- **Reactancia síncrona (X_s) e inductancia síncrona (L_s).**
- **Tensión en bornes (U) cuando funciona en vacío a 1200 rpm con una excitación de 2,84 A.**
- **Intensidad del ensayo de cortocircuito (I_{cc}) realizado a 1200 rpm y una excitación de 1,5 A.**

a)

En primer lugar, antes de empezar con la resolución del problema, sería introducir datos nominales, además de los ensayos de vacío y cortocircuito, para poder calcular la pendiente (E_o/I_e) del ensayo de vacío y la reactancia síncrona (X_s) del ensayo de cortocircuito, necesarios para posteriores cálculos.

Para este caso en concreto, supondremos que unos valores máximos de diseño. Ya que, según los datos nominales de la máquina, suelen estar entorno a estos valores: $I_{e_{max}} = 100$ A y $n_{max} = 3000$ rpm.

➤ Datos Nominales.

Se introducen los datos nominales dentro del apartado de "Características Nominales" para más tarde pulsar el botón de confirmar datos que permite realizar los cálculos.

CARACTERÍSTICAS NOMINALES					
Sn (KVA)	<input type="text" value="45"/>	Frecuencia (Hz)	<input type="text" value="50"/>	Valores máximos de Diseño I _e _Máx (A) <input type="text" value="100"/> n_Máx (rpm) <input type="text" value="3000"/> I _e _N (A) <input type="text" value="2.48"/>	
Un (KV)	<input type="text" value="0.230"/>	n (rpm)	<input type="text" value="1000"/>		
I _n (A)	<input type="text" value="112.960"/>	Pares de Polos	<input type="text" value="3"/>		
<input type="button" value="Confirmar Características Nominales"/>					

Figura 59: Ejemplo de aplicación 1. Introducción de datos nominales.

Lo primero se calculan I_n y p de la siguiente forma:

$$I_n = \frac{S_n}{\sqrt{3} \cdot U_n} = \frac{45000}{\sqrt{3} \cdot 230} = 112,96 \text{ A}$$

$$p = \frac{60 \cdot f}{n} = \frac{60 \cdot 50}{1000} = 3 \text{ pares de polos}$$

Una vez calculados los estos primeros parámetros, se introducen los datos de los ensayos y se calculan las siguientes variables a través del botón "Confirmar Datos Ensayos".

➤ Ensayo de vacío.

Con los datos del ensayo de vacío se calculan la pendiente (E_o/I_e) y la intensidad de excitación nominal (I_{e_n}).

ENSAYO DE VACIO	
U _o (kV)	<input type="text" value="0.230"/>
I _{e_o} (A)	<input type="text" value="2.84"/>
n _o (rpm)	<input type="text" value="1000"/>

Figura 60: Ejemplo de aplicación 1. Introducción de datos de ensayo de vacío.

Primero, se calcula la pendiente del ensayo de vacío para posteriores cálculos.

$$m = \frac{E_{0\text{vacío}}}{I_{e\text{vacío}}} = \frac{\frac{U_0}{\sqrt{3}}}{I_{e\text{vacío}}} = \frac{\frac{230}{\sqrt{3}}}{2,84} = \frac{132,8}{2,48} = 46,757 \text{ V/A}$$

A continuación, se muestra la pendiente $m=46,757$ en la gráfica del ensayo de vacío:

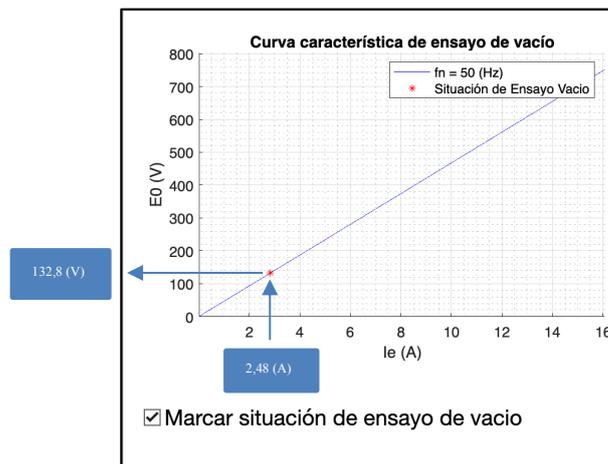


Figura 61: Ejemplo de aplicación 1. Pendiente de ensayo de vacío.

Como en este caso la frecuencia del ensayo de vacío coincide con la frecuencia nominal ($n_{\text{nominal}} = n_{\text{vacío}}$), la pendiente m no cambia porque no varía la frecuencia. Por lo tanto, $m=m1$, y como el ensayo de vacío se realiza a tensión nominal pues I_{e_n} coincide con $I_{e\text{vacío}}$.

$$f_{\text{vacío}} = \frac{p \cdot n_o}{60} = \frac{3 \cdot 1000}{60} = 50 \text{ Hz}$$

$$m1 = m \cdot \frac{f_{\text{vacío}}}{f_{\text{nominal}}} = 46,757 \cdot \frac{50}{50} = 46,757 \text{ V/A}$$

$$I_{e_n} = \frac{E0_{\text{nominal}}}{m1} = \frac{\frac{230}{\sqrt{3}}}{46,757} = 2,84 \text{ A}$$

ENSAYO DE VACIO	
f_vacio (Hz)	<input type="text" value="50"/>
E0 (kV)	<input type="text" value="0.1328"/>

Figura 62: Ejemplo de aplicación 1. Cálculos del ensayo de vacío.

En la figura 62 se muestran los cálculos de $f_{vacío}$ y $Eo_{vacío}$ que como se puede observar coinciden con los calculados con las ecuaciones anteriores.

➤ Ensayo de cortocircuito.

Con los datos del ensayo de cortocircuito y datos calculados anteriormente, se calcula la reactancia síncrona (X_s), la frecuencia del ensayo de corto (f_{cc}) y la inductancia síncrona (L_s).

ENSAYO DE CORTOCIRCUITO	
I _{cc} (A)	<input type="text" value="118"/>
I _{e_cc} (A)	<input type="text" value="2.2"/>
n _{cc} (rpm)	<input type="text" value="1000"/>

Figura 63: Ejemplo de aplicación 1. Introducción de datos de ensayo de cortocircuito.

$$f_{cc} = \frac{p \cdot n_{cc}}{60} = \frac{3 \cdot 1000}{60} = 50 \text{ Hz}$$

$$E0_{cc} = m \cdot I_{e_{cc}} = 46,757 \cdot 2,20 = 102,865 \text{ A}$$

$$X_s = \frac{E0_{cc}}{I_{cc}} = \frac{102,865}{118} = 0,872 \Omega$$

$$L_s = \frac{X_s}{2 \cdot \pi \cdot f_{cc}} = \frac{0,872}{2 \cdot \pi \cdot 50} = 0,002775 \text{ H}$$

ENSAYO DE CORTOCIRCUITO	
f _{cc} (Hz)	<input type="text" value="50"/>
X _s (Ω)	<input type="text" value="0.8717"/>
L _s (H)	<input type="text" value="0.002775"/>

Figura 64: Ejemplo de aplicación 1. Cálculo del ensayo de cortocircuito.

b)

Una vez habiendo calculado los pasos anteriores, podemos simular cualquier situación de funcionamiento. En este caso, se simula la situación de ensayo de vacío para $n = 1200 \text{ rpm}$ y $I_e = 2,84 \text{ A}$.

- Simulación de la situación de vacío.

Parámetros Entrada Regulación	
I_e (A)	<input type="text" value="2.84"/>
n (rpm)	<input type="text" value="1200"/>

Figura 65: Ejemplo de aplicación 1. Introducción de datos de regulación.

$$f = \frac{p \cdot n}{60} = \frac{3 \cdot 1200}{60} = 60 \text{ Hz}$$

$$m_{60 \text{ Hz}} = m_{50 \text{ Hz}} \cdot \frac{60}{50} = 46,757 \cdot \frac{60}{50} = 56,108 \text{ V/A}$$

$$E_{0_{60 \text{ Hz}}} = m_{60 \text{ Hz}} \cdot I_e = 56,108 \cdot 2,84 = 159,35 \text{ V} = 0,15935 \text{ kV}$$

$$U_0 = \sqrt{3} \cdot 159,35 = 276 \text{ V} = 0,276 \text{ kV}$$

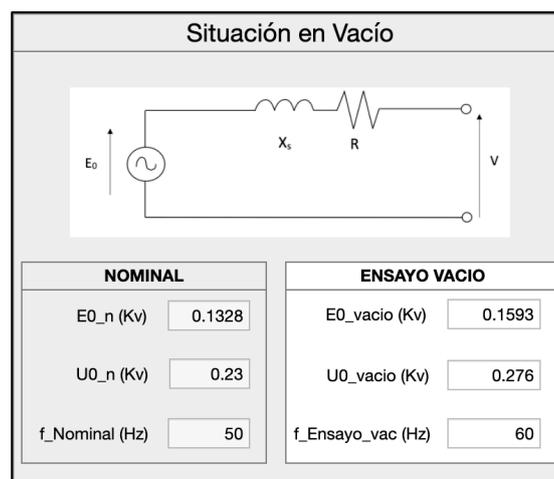


Figura 66: Ejemplo de aplicación 1. Simulación. Cálculo de la situación de vacío.

c)

Una vez habiendo calculado los pasos anteriores, podemos simular cualquier situación de funcionamiento. En este caso, se simula la situación de ensayo de cortocircuito para $n=1200$ rpm y $I_e=1,5$ A.

➤ Simulación de la situación de cortocircuito.

$$f = \frac{p \cdot n}{60} = \frac{3 \cdot 1200}{60} = 60 \text{ Hz}$$

$$m_{60 \text{ Hz}} = m_{50 \text{ Hz}} \cdot \frac{60}{50} = 46,757 \cdot \frac{60}{50} = 56,108 \text{ V/A}$$

$$E_{0_{60 \text{ Hz}}} = m_{60 \text{ Hz}} \cdot I_e = 56,108 \cdot 1,5 = 84,162 \text{ V} = 0,08416 \text{ kV}$$

$$I_{cc} = \frac{E_{0_{60 \text{ Hz}}}}{X_{S_{60 \text{ Hz}}}} = \frac{E_{0_{60 \text{ Hz}}}}{2 \cdot \pi \cdot f \cdot L_s} = \frac{84,162}{2 \cdot \pi \cdot 60 \cdot 0,002775} = 80,45 \text{ A}$$

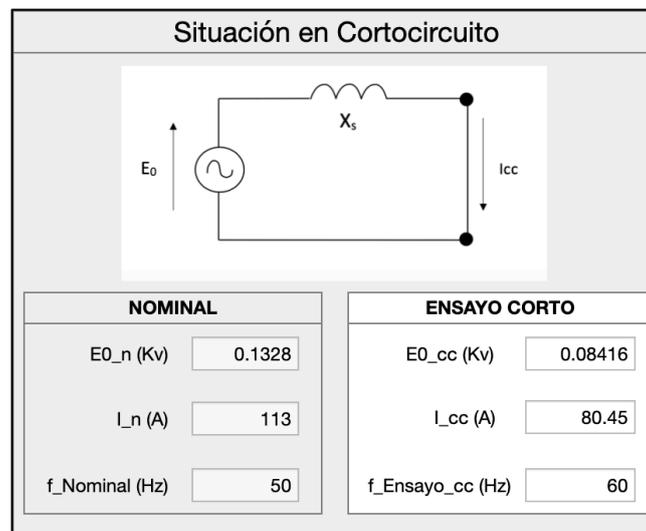


Figura 67: Ejemplo de aplicación 1. Simulación. Cálculo de la situación de cortocircuito.

8.2 Ejemplo 2

Un generador síncrono trifásico de rotor cilíndrico de 300 kVA, 6 kV, 1500 rpm, 50 Hz y conexión estrella.

Se realizan los ensayos de vacío y cortocircuito en las citadas condiciones nominales ($n_0 = n_{cc} = 1500 \text{ rpm}$):

- **Ensayo de vacío:**
 - U_o : 6 kV
 - I_e : 5 A
- **Ensayo de cortocircuito:**
 - I_{cc} : 28,87 A (Intensidad nominal)
 - I_e : 0,75 A
- **Datos de la carga:**
 - $S = 250 \text{ kVA}$
 - $\cos\phi = 0,8$
 - Tipo de carga = inductiva

a) Calcular la intensidad de excitación necesaria (I_e) para que la carga esté alimentada a tensión nominal ($U = 6 \text{ kV}$) y calcular todos los demás parámetros, en dichas condiciones.

➤ Simulación de la situación de carga.

Lo primero de todo, se introducen los datos nominales y los de los ensayos como en el ejemplo de aplicación 1. Luego se calculan los datos necesarios para proceder con la simulación de la situación en carga, dando como resultado los siguientes valores de las figuras:

CARACTERÍSTICAS NOMINALES

Sn (KVA) <input type="text" value="300"/>	Frecuencia (Hz) <input type="text" value="50"/>	Valores máximos de Diseño
Un (KV) <input type="text" value="6"/>	n (rpm) <input type="text" value="1500"/>	I _e _Máx (A) <input type="text" value="100"/>
I _n (A) <input type="text" value="28.87"/>	Pares de Polos <input type="text" value="2"/>	n_Máx (rpm) <input type="text" value="3000"/>
		I _e _N (A) <input type="text" value="5"/>

Figura 68: Ejemplo de aplicación 2. Introducción de datos nominales.

TABLA DE ENSAYOS	
ENSAYO DE VACIO	ENSAYO DE CORTOCIRCUITO
U _o (kV) <input type="text" value="6"/>	I _{cc} (A) <input type="text" value="28.87"/>
I _{e_o} (A) <input type="text" value="5"/>	I _{e_cc} (A) <input type="text" value="0.75"/>
n _o (rpm) <input type="text" value="1500"/>	n _{cc} (rpm) <input type="text" value="1500"/>
<input type="button" value="Confirmar Datos Ensayos"/>	

Figura 69: Ejemplo de aplicación 2. Introducción de datos de ensayos.

CÁLCULO DE ENSAYOS	
ENSAYO DE VACIO	ENSAYO DE CORTOCIRCUITO
f _{vacio} (Hz) <input type="text" value="50"/>	f _{cc} (Hz) <input type="text" value="50"/>
E ₀ (kV) <input type="text" value="3.464"/>	X _s (Ω) <input type="text" value="18"/>
	L _s (H) <input type="text" value="0.05729"/>

Figura 70: Ejemplo de aplicación 2. Cálculo de ensayos.

Una vez calculados los parámetros necesarios como se ha explicado en el ejemplo 1, se procede al cálculo de la situación en carga. De los datos de regulación solo es conocido la velocidad de giro, que es la nominal ($n = 1500$ rpm), para que sea $f = 50$ Hz. Por lo tanto, con la opción de puesta en marcha automática, se harán iteraciones con valores arbitrarios de I_e hasta encontrar cuando la tensión en bornes (U) es la $U_n = 6$ kV.

Los cálculos de las iteraciones que realiza el programa se harían de la siguiente manera:

$$\varphi = \arccos(0,8) = 36,87^\circ$$

$$I_{carga} = \frac{S_{carga}}{\sqrt{3} \cdot U_n} = \frac{250 \text{ kVA}}{\sqrt{3} \cdot 6 \text{ kV}} = 24,056 \text{ A}$$

$$Z_{carga} = \frac{U_n}{I_{carga}} = \frac{6000}{24,056} = 144 \Omega$$

$$R_{carga} = Z_{carga} \cdot \cos\varphi = 144 \cdot 0,8 = 115,2 \Omega$$

$$X_{carga} = Z_{carga} \cdot \sen\varphi = 144 \cdot 0,6 = 86,4 \Omega$$

Una vez definido la impedancia de la carga (Z_{carga}) a través de R_{carga} y X_{carga} , se añade la influencia de X_s para calcular la impedancia total del circuito equivalente (Z_{total}).

$$X_s = 2 \cdot \pi \cdot f \cdot L_s = 2 \cdot \pi \cdot 50 \cdot 0,05729 = 18 \Omega$$

$$X_{total} = X_{carga} + X_s = 86,4 \Omega + 18 \Omega = 104,4 \Omega$$

$$R_{total} = R_{carga} = 115,2 \Omega$$

$$Z_{total} = \sqrt{R_{total}^2 + X_{total}^2} = \sqrt{115,2^2 + 104,4^2} = 155,468 \Omega$$

Cuando tenemos la impedancia total, podemos calcular la intensidad real por fase (I_{carga}), suponiendo una intensidad de excitación, que después de ir probando con la interfaz gráfica, la que más se acerca es $I_{carga} = 5,4$ A.

$$E_{0carga} = m_{50 \text{ Hz}} \cdot I_{carga} = \frac{U_0}{I_{e \text{ vacío}}} \cdot I_{carga} = \frac{6000}{5} \cdot 5,4 = 3741,22 \text{ V}$$

$$I_{carga} = \frac{E_{0carga}}{Z_{total}} = \frac{3741,22}{155,468} = 24,064 \text{ A}$$

$$V = I_{carga} \cdot Z_{carga} = 24,064 \cdot 144 = 3465,22 \text{ V}$$

$$U = \sqrt{3} \cdot V = \sqrt{3} \cdot 3465,22 = 6001,93 \approx 6000 \text{ V} = 6 \text{ kV}$$

Habiendo comprobado que con $I_{carga} = 5,4$ A se logra la tensión nominal en bornes de la carga ($U = 6 \text{ kV}$), procedemos a calcular las potencias consumidas por la carga en estas condiciones.

$$P = \sqrt{3} \cdot U \cdot I_{carga} \cdot \cos\varphi = \sqrt{3} \cdot 6000 \cdot 24,064 \cdot 0,8 = 200064 \text{ W} = 200,064 \text{ kW}$$

$$Q = \sqrt{3} \cdot U \cdot I_{carga} \cdot \sen\varphi = \sqrt{3} \cdot 6000 \cdot 24,064 \cdot 0,6 = 150048,25 = 150,048 \text{ kW}$$

Todos estos cálculos se pueden comprobar que coinciden con los calculados por la interfaz gráfica si se observan las siguientes figuras:

Parámetros Entrada Regulación

le (A)

n (rpm)

Introducir datos de la carga

S_carga (Kva)

cosφ

Tipo de Carga

Figura 71: Ejemplo de aplicación 2. Introducción de parámetros de regulación y datos de la carga.

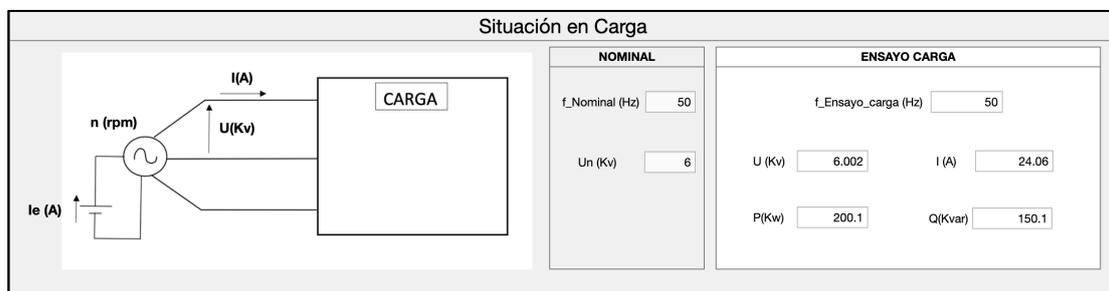


Figura 72: Ejemplo de aplicación 2. Simulación. Cálculo de la situación de carga.

Para mostrar funcionalidad de los mensajes de advertencia de la aplicación, se va a mostrar una situación en la que se supera tanto velocidad de giro como la intensidad de excitación por encima de los valores máximos de diseño de la máquina, que en este caso son: $I_{e_{max}}=100$ A y $n_{max}= 3000$ rpm.

Figura 73: Mensaje de Advertencia. Intensidad de excitación sobrepasada.

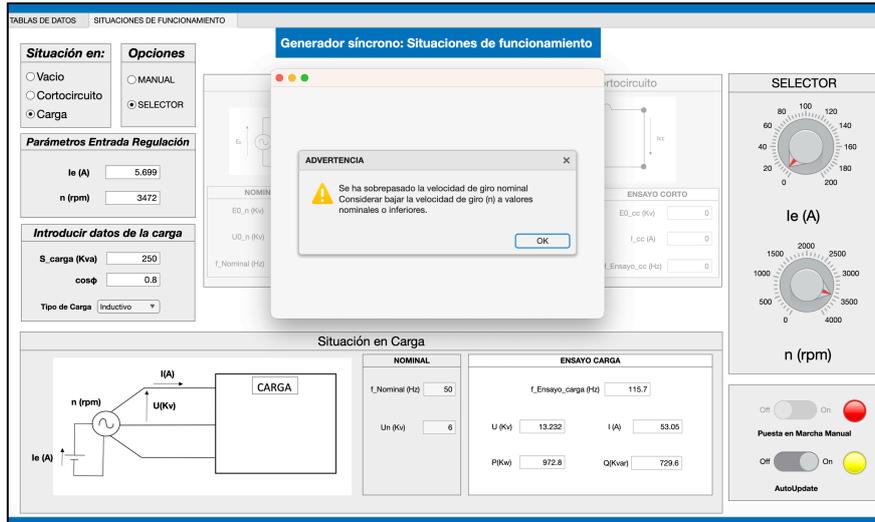


Figura 74: Mensaje de Advertencia. Velocidad de giro sobrepasada.

9 DESCRIPCIÓN DE LAS TAREAS. DIAGRAMA DE GANTT

A continuación, se describen las diferentes tareas llevadas a cabo para la realización del TFG. A lo largo del desarrollo del mismo, han tenido participación tanto un ingeniero junior (alumno encargado de la realización del TFG) y un ingeniero senior (encargado de la supervisión del TFG). Las tareas desarrolladas para la realización del documento han sido las siguientes:

- **Análisis del software:** El proyecto comienza el día 19 de octubre de 2022, tras la primera reunión entre el tutor y el alumno. En esta primera reunión se expone, por parte del alumno, la posibilidad de realizar una interfaz gráfica con una temática alternativa que no estaba planteada inicialmente para el TFG. Después de enfocar más concretamente el tema que se va a abordar y recibir la aprobación del tutor, el 20 de octubre, el alumno comienza a buscar diferentes opciones de programas de software para la implementación de la misma. Pero no es hasta el 15 de noviembre cuando se comienza familiarizar con el funcionamiento del programa seleccionado.
- **Búsqueda de información:** El 16 de noviembre de 2022 tiene lugar la siguiente reunión en donde el tutor indica al alumno que debe comenzar a recopilar información sobre el tema que aborda el TFG apoyándose en la bibliografía que le recomienda el tutor y en el resto disponible por medio de la universidad. Esta fase comienza el 17 y termina el 27 de enero de 2023.
- **Desarrollo de la interfaz gráfica:** Una vez acabada la reunión del 1 de febrero el alumno se termina de informar y le plantea al tutor las diferentes ideas que se le han ocurrido para como enfocar el desarrollo de la interfaz. Es durante este periodo de tiempo que comienza el 8 de febrero hasta 13 de marzo donde se van incorporando nuevas ideas a la interfaz o corrigiendo funcionalidades antiguas a medida que se va avanzando en el desarrollo.
- **Corrección de errores:** Una vez realizado el primer diseño de la interfaz en condiciones, tiene lugar la reunión el día 18 de abril para corregir posibles errores y plantear nuevas opciones a mejorar en la interfaz. Este proceso dura desde el 21 de abril hasta el 23 de mayo.
- **Redacción del documento:** El día 24 de mayo se comienza la redacción del documento del TFG, pero no es hasta el 31 de julio que se termina la redacción del mismo. Es decir, el proyecto se demora algo más de lo esperado debido a situaciones ajenas al mismo. Durante ese periodo, se realiza una reunión del día 13 de julio para el tutor pueda

supervisar el avance del alumno durante la redacción y resolver posibles dudas que le vayan surgiendo.

- Corrección del documento:** Una vez ya con un borrador del documento en vías de terminar la redacción final del mismo, se realiza una ultima reunión el 5 de septiembre para corregir posibles que el alumno haya podido cometer. Tras la reunión, se comienza una ultima etapa donde el alumno debe incorporar las ultimas modificaciones recomendadas por el tutor. Esta fase comienza el 6 de septiembre y termina el 12 de septiembre. Es para esta última fecha para la cual el proyecto se da por finalizado, es cuando también se realiza la ultima reunión del proyecto.

Se muestra a continuación, el diagrama de Gantt del proyecto donde establecen las tareas y las fechas de las mismas durante el desarrollo del trabajo.

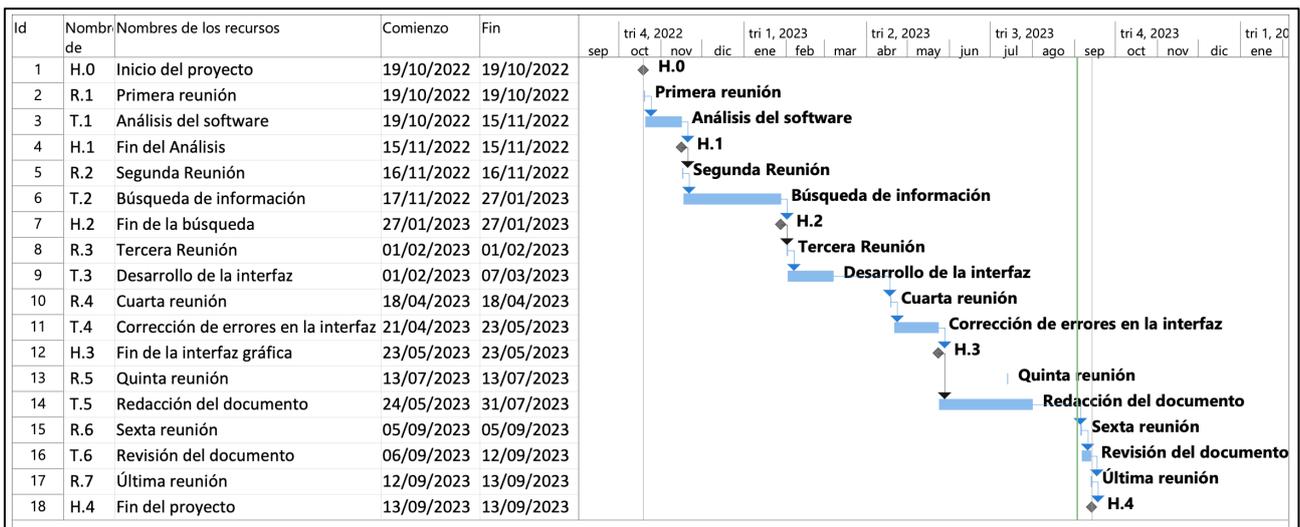


Figura 75: Diagrama de Gantt.

10 PRESUPUESTO

En cuanto al presupuesto del proyecto, este se divide en 3 partidas: horas internas, amortizaciones y gastos. En primer lugar, en las horas internas se incluyen las horas invertidas en este TFG por un ingeniero senior (director del TFG) y un ingeniero junior (alumno que realiza el TFG). En cuanto a la partida de amortizaciones, tenemos la licencia del software de *Matlab*, en la que se considera una licencia estándar anual, para uso a nivel empresarial, y la licencia de Office para el uso de los programas Word y Excel. Finalmente, en cuanto a los gastos se pueden incluir material de oficina empleado en el transcurso del proyecto. Adicionalmente, se ha considerado que los gastos indirectos suponen un 6% del presupuesto.

Horas internas			
Concepto	Nº de horas	€/ hora	Total
Ingeniero senior	40	50	2.000 €
Ingeniero junior	210	25	5.250 €
SUBTOTAL			7.250 €

Tabla 3: Partida de horas internas.

Amortizaciones				
Concepto	Precio producto	Vida útil (horas)	Utilización (horas)	Total
Ordenador	1.100 €	25000	250	11,00 €
Licencia de MATLAB	860 €	2920	150	44,18 €
Licencia de Office	69 €	2920	100	2,36 €
SUBTOTAL				57,54 €

Tabla 4: Partida de amortizaciones.

Gastos	
Concepto	Total
Material	50 €
SUBTOTAL	50 €

Tabla 5: Partida de gastos.

Resumen	
Horas internas	7.050,00 €
Amortizaciones	57,54 €
Gastos	50,00 €
SUBTOTAL	7.157,54 €
Costes indirectos (6%)	429,45 €
TOTAL	7.586,99 €

Tabla 6: Resumen de presupuesto.

11 CONCLUSIONES

La máquina síncrona es un elemento de gran importancia dentro del sistema eléctrico de potencia, más concretamente, en el papel que juega como generador dentro del mismo sistema. Aunque, en aplicaciones habituales para estos generadores como son en centrales hidráulicas, específicamente, en las de bombeo, tiene también mucha utilidad la capacidad reversible de la máquina. Puesto que, puede trabajar como motor e impulsar una bomba que transfiera el agua desde aguas abajo de la presa hasta aguas arriba, en periodos donde la demanda energética sea baja y la electricidad tenga precios bajos, es decir, por las noches. Acumulando así agua en el embalse y que este no se agote a medida que se va fluyendo por la turbina y generando electricidad por medio del generador síncrono.

Estos alternadores, están preparados para funcionar bajo unos parámetros nominales de diseño específicos de la máquina, aunque, también pueden trabajar con cierto margen superior a estos, en casos puntuales. Sin embargo, si se superan en gran medida o ligeramente durante un periodo de tiempo muy prolongado, pueden llevar al deterioro de los elementos que componen el generador. Además, pueden ser causa de daños a tanto seres humanos como al propio sistema eléctrico.

Es por este motivo, que se hace indispensable conocer mejor el estado de la máquina ante cualquier situación que pueda ocurrir con la misma. Es en este momento, cuando la interfaz gráfica puede desempeñar un papel esencial en todo esto. Este software, permite calcular los parámetros eléctricos necesarios para describir el comportamiento del generador, además permite simular, como si fuese un sistema de regulación, ante cualquiera de situaciones de funcionamiento. Permitiendo así, poder comparar los valores obtenidos en simulación con los valores nominales, y, por consiguiente, evaluar el buen funcionamiento del mismo en cualquiera de las condiciones, antes de que se produzcan los problemas técnicos.

12 REFERENCIAS

- [1] **J. Mazón, J. F. Miñambres, M. Á. Zorrozu, G. Buigues y V. Valverde**, Guía de autoaprendizaje de máquinas eléctricas, Madrid: PEARSON EDUCACIÓN, S.A., 2008.
- [2] **J. Fraile Mora**, *Máquinas eléctricas*, Madrid, McGraw-Hill, 2008.
- [3] **T. Wildi**, *Máquinas eléctricas y sistemas de potencia*, México, Pearson Educación, 2007.
- [4] **S. J. Chapman**, *Máquinas eléctricas*, México, McGraw-Hill, 2005.
- [5] **A. E. Fitzgerald, C. Kingsley Jr, S. D. Umans**, *Máquinas eléctricas*, México, McGraw-Hill, 2004.
- [6] «**Página web oficial mathworks**,» [En línea]. Available: <https://www.mathworks.com/products/matlab/app-designer/comparing-guide-and-app-designer.html>.
- [7] «**Página web oficial mathworks**,» [En línea]. Available: <https://la.mathworks.com/products/matlab/matlab-and-python.html#:~:text=MATLAB%20proporciona%20una%20integración,de%20producción%20y%20sistemas%20informáticos>.
- [8] «**Página web oficial Iberdrola**,» [En línea]. Available: <https://www.iberdrola.com/conocenos/nuestra-actividad/smart-grids/diferencia-alta-media-baja-tension-electrica>.
- [9] **M. A. Rodríguez Pozueta**, *Principios generales de las Máquinas Eléctricas*, Departamento de Ingeniería Eléctrica y Energética, España, Creative Commons, 2015.
- [10] «**Página web oficial GLOBAL ENERGY**,» [En línea]. Available: <https://globalenergyprize.org/en/2022/06/03/brasil-y-paraguay-comienzan-la-reparacion-de-itaipu/>.
- [11] «**Página web oficial Energiforsk**,» [En línea]. Available: <https://energiforsk.se/media/27015/nordic-nuclear-power-generator-stator-vibrations-energiforskrappport-2019-621.pdf>.
- [12] **J. A. Corral Hernández**, *Detección de roturas de devanado amortiguador en máquinas síncronas mediante análisis transitorio de la corriente*, Trabajo de Fin de Master, Master Ingeniería en Mantenimiento, Universidad Politécnica de Valencia.
- [13] **J. Mazón, I. López**, Apuntes de la asignatura *Integración de tecnologías de generación en el sistema eléctrico*, Grado en Ingeniería en Tecnología Industrial (Departamento Ingeniería eléctrica), Universidad del País Vasco, Curso 2022/2023.

13 ANEXO.CÓDIGO COMPLETO DE LA INTERFAZ

```

classdef Generador_Sincrono_TFG < matlab.apps.AppBase

    % Properties that correspond to app components
    properties (Access = public)
        UIFigure                matlab.ui.Figure
        TabGroup                 matlab.ui.container.TabGroup
        TABLASDEDATOSTab        matlab.ui.container.Tab
        Circuito_Monofasico_Equivalente_Panel  matlab.ui.container.Panel
        ENSAYODECORTOCIRCUITOPanel  matlab.ui.container.Panel
        Output_f_cc              matlab.ui.control.NumericEditField
        U0KvLabel_2              matlab.ui.control.Label
        Output_Ls                 matlab.ui.control.NumericEditField
        LsHLabel                  matlab.ui.control.Label
        Output_Xs                 matlab.ui.control.NumericEditField
        XsLabel                    matlab.ui.control.Label
        ENSAYODEVACIOPanel        matlab.ui.container.Panel
        Output_f_vacio            matlab.ui.control.NumericEditField
        U0KvLabel_1              matlab.ui.control.Label
        Output_Eo                 matlab.ui.control.NumericEditField
        E0kVEditFieldLabel        matlab.ui.control.Label
        Panel_2                   matlab.ui.container.Panel
        Situacion_Vacio_CheckBox  matlab.ui.control.CheckBox
        UIAxes                     matlab.ui.control.UIAxes
        CARACTERISTICAS_NOMINALES_Panel  matlab.ui.container.Panel
        ValoresmaximosdeDisenoPanel  matlab.ui.container.Panel
        Input_Ie_Max              matlab.ui.control.NumericEditField
        Ie_MxALabel                matlab.ui.control.Label
        Input_n_Max               matlab.ui.control.NumericEditField
        n_MxrpmLabel              matlab.ui.control.Label
        Input_fn_CN                matlab.ui.control.NumericEditField
        FrecuenciaHzLabel         matlab.ui.control.Label
        Input_n_rpm_CN            matlab.ui.control.NumericEditField
        Label_2                    matlab.ui.control.Label
        Input_Un_CN                matlab.ui.control.NumericEditField
        UnKVLabel_1                matlab.ui.control.Label
        Input_Sn_CN                matlab.ui.control.NumericEditField
        SnKVAEditFieldLabel        matlab.ui.control.Label
        Confirmar_CN_Button        matlab.ui.control.Button
        Output_In_CN              matlab.ui.control.NumericEditField
        InAEditFieldLabel          matlab.ui.control.Label
        Output_Par_Polos_CN        matlab.ui.control.NumericEditField
        ParesdePolosEditFieldLabel  matlab.ui.control.Label
        Output_Ie_n_CN            matlab.ui.control.NumericEditField
        Ie_NALabel                  matlab.ui.control.Label
        TABLA_ENSAYOS_Panel        matlab.ui.container.Panel
        ENSAYODEVACIOPanel_2      matlab.ui.container.Panel
        Input_n_o_Ensayo           matlab.ui.control.NumericEditField
        n_orpmEditFieldLabel        matlab.ui.control.Label
        Input_Ie_o                 matlab.ui.control.NumericEditField
        Ie_oALabel                  matlab.ui.control.Label
        Input_Uo                    matlab.ui.control.NumericEditField
        UokVEditFieldLabel          matlab.ui.control.Label
        ENSAYODECORTOCIRCUITOPanel_2  matlab.ui.container.Panel
        Input_Ie_cc_Ensayo         matlab.ui.control.NumericEditField
        Ie_ccAEditFieldLabel        matlab.ui.control.Label
        Input_n_cc_Ensayo          matlab.ui.control.NumericEditField
    end

```

n_ccrpmEditFieldLabel	matlab.ui.control.Label
Input_Icc	matlab.ui.control.NumericEditField
IccAEditFieldLabel	matlab.ui.control.Label
Confirmar_Datos_Ensayos_Button	matlab.ui.control.Button
GENERADOR_SINCRONO_TRIFASICO_Label	matlab.ui.control.Label
SITUACION_FUNCIONAMIENTO_Tab	matlab.ui.container.Tab
Parmetros_Entrada_Regulacion_Panel	matlab.ui.container.Panel
Input_regulacion_n	matlab.ui.control.NumericEditField
nrpmEditFieldLabel_2	matlab.ui.control.Label
Input_regulacion_Ie	matlab.ui.control.NumericEditField
IeAEditFieldLabel_2	matlab.ui.control.Label
Situacion_Carga_Panel	matlab.ui.container.Panel
nrpm_Label_carga	matlab.ui.control.Label
I_Label_Carga	matlab.ui.control.Label
UKv_Label_Carga	matlab.ui.control.Label
Ie_Label_Carga	matlab.ui.control.Label
ENSAYOCARGA_Panel	matlab.ui.container.Panel
Output_QKvar_Carga	matlab.ui.control.NumericEditField
QKvarEditFieldLabel	matlab.ui.control.Label
Output_PKw_Carga	matlab.ui.control.NumericEditField
PKwEditFieldLabel	matlab.ui.control.Label
Output_f_Ensayo_Carga	matlab.ui.control.NumericEditField
f_Ensayo_cargaHzEditFieldLabel	matlab.ui.control.Label
Output_I_Carga	matlab.ui.control.NumericEditField
IAEditFieldLabel	matlab.ui.control.Label
Output_UKv_Carga	matlab.ui.control.NumericEditField
UKvEditFieldLabel	matlab.ui.control.Label
NOMINAL_Panel	matlab.ui.container.Panel
f_Nominal_Carga	matlab.ui.control.NumericEditField
f_NominalHzEditField_3Label	matlab.ui.control.Label
UnKv_Nominal_Carga	matlab.ui.control.NumericEditField
UnKvEditFieldLabel	matlab.ui.control.Label
Esquema_Situacion_Carga	matlab.ui.control.Image
Situacion_Cortocircuito_Panel	matlab.ui.container.Panel
ENSAYOCORTOPanel	matlab.ui.container.Panel
Output_f_Ensayo_cc	matlab.ui.control.NumericEditField
f_Ensayo_ccHzEditFieldLabel	matlab.ui.control.Label
Output_I_cc	matlab.ui.control.NumericEditField
I_ccAEditFieldLabel	matlab.ui.control.Label
Output_E0_cc	matlab.ui.control.NumericEditField
E0_ccKvEditFieldLabel	matlab.ui.control.Label
NOMINALPanel_2	matlab.ui.container.Panel
f_Nominal_cc	matlab.ui.control.NumericEditField
f_NominalHzEditField_2Label	matlab.ui.control.Label
I_n_cc	matlab.ui.control.NumericEditField
I_nAEditFieldLabel	matlab.ui.control.Label
E0_n_cc	matlab.ui.control.NumericEditField
E0_nKvEditField_2Label	matlab.ui.control.Label
Esquema_Situ_cc	matlab.ui.control.Image
SituacionVacioPanel	matlab.ui.container.Panel
ENSAYOVACIOPanel	matlab.ui.container.Panel
Output_U0_vacio	matlab.ui.control.NumericEditField
U0_vacioKvEditFieldLabel	matlab.ui.control.Label
Output_f_Ensayo_vacio	matlab.ui.control.NumericEditField
f_Ensayo_vachHzEditFieldLabel	matlab.ui.control.Label
Output_E0_vacio	matlab.ui.control.NumericEditField
E0_vacioKvEditFieldLabel	matlab.ui.control.Label
NOMINALPanel	matlab.ui.container.Panel
f_Nominal_vacio	matlab.ui.control.NumericEditField

```

f_NominalHzEditFieldLabel      matlab.ui.control.Label
U0_n_vacio                      matlab.ui.control.NumericEditField
U0_nKvEditFieldLabel           matlab.ui.control.Label
E0_n_vacio                      matlab.ui.control.NumericEditField
E0_nKvEditFieldLabel           matlab.ui.control.Label
Esquema_Situ_vacio             matlab.ui.control.Image
Panel                           matlab.ui.container.Panel
LED_AutoUpdate                  matlab.ui.control.Lamp
Interruptor_AutoUpdate          matlab.ui.control.Switch
AutoUpdateSwitchLabel          matlab.ui.control.Label
Interruptor_Marcha_Manual       matlab.ui.control.Switch
PuestaenMarchaManualLabel      matlab.ui.control.Label
LED_Marcha_Manual              matlab.ui.control.Lamp
SELECTORPanel                   matlab.ui.container.Panel
Label_n                         matlab.ui.control.Label
Label_Ie                        matlab.ui.control.Label
Input_Select_Ie                 matlab.ui.control.Knob
KnobLabel                       matlab.ui.control.Label
Input_Select_n                  matlab.ui.control.Knob
Label                           matlab.ui.control.Label
OpcionesTipoCarga              matlab.ui.container.ButtonGroup
Selector_Button                 matlab.ui.control.RadioButton
Manual_Button                   matlab.ui.control.RadioButton
Introducir_Datos_Carga_Panel    matlab.ui.container.Panel
TipodeCargaDropDown            matlab.ui.control.DropDown
TipodeCargaDropDownLabel       matlab.ui.control.Label
Input_Factor_Potencia          matlab.ui.control.NumericEditField
CosLabel_2                     matlab.ui.control.Label
Input_Scarga                    matlab.ui.control.NumericEditField
PKWLabel                       matlab.ui.control.Label
Situaciones_Button_Group       matlab.ui.container.ButtonGroup
Carga_Button                    matlab.ui.control.RadioButton
Cortocircuito_Button           matlab.ui.control.RadioButton
Vacio_Button                    matlab.ui.control.RadioButton
Generador_Sincrono_Situacion_Funcionamiento_Label
matlab.ui.control.Label
end

properties (Access = private)
    autoUpdate % Opción de hacer cálculos automáticos sin usar boton de
confirmación de datos de entrada;

end

methods (Access = private)

function Calcular_Situaciones_Funcionamiento(app)

    p=app.Output_Par_Polos_CN.Value;
    n=app.Input_regulacion_n.Value;
    Ie=app.Input_regulacion_Ie.Value;

    f_val_entrada=(p*n)/60;
    Ie_val_entrada= Ie;

    U0=app.Input_Uo.Value*10^3; % En KV el input y pasamos a V;
    E0_ens=U0/sqrt(3);

```

```

% Calculamos la relación (E0/Ie) a frecuencia ensayo de vacio
% en la variable m;

Ie_o=app.Input_Ie_o.Value;
m=E0_ens/Ie_o;
n0=app.Input_n_o_Ensayo.Value;
f_ensayo_vacio=(p*n0)/60;

% Calculamos la relacion (E0/Ie) a frecuencia de la variable de
% entrada en la variable m_val en funcion de m y las
% frecuencias respectivas;

m_val= m*(f_val_entrada/f_ensayo_vacio);
E0_val= m_val*Ie_val_entrada;

if app.Vacio_Button.Value == 1

U0_val=sqrt(3)*E0_val;

app.Output_U0_vacio.Value=U0_val/(10^3); %En Kv;
app.Output_E0_vacio.Value=E0_val/(10^3); %En Kv;
app.Output_f_Ensayo_vacio.Value=f_val_entrada;

elseif app.Cortocircuito_Button.Value == 1

Ls=app.Output_Ls.Value; % Dato calculado previamente en tabla de
datos;
Xs=2*pi*f_val_entrada*Ls;% Reactancia sincrona a frecuencia del
regulador n;
Icc=E0_val/Xs;

app.Output_E0_cc.Value=E0_val/(10^3); %En Kv;
app.Output_I_cc.Value=Icc;
app.Output_f_Ensayo_cc.Value=f_val_entrada;

else
% Cuando es situacion de carga;

Ls=app.Output_Ls.Value; % Dato calculado previamente en tabla
de datos;
app.Output_f_Ensayo_Carga.Value=f_val_entrada;

%Datos necesarios para los calculos
Un=app.Input_Un_CN.Value*10^3; % Suponemos que alimentamos la
carga a tensión Nominal Un (Kv)
S_carga=app.Input_Scarga.Value*10^3; %Scarga en (kVA)--> VA
Factor_Potencia=app.Input_Factor_Potencia.Value;
% Por defecto ponemos que si cosphi=1 entonces es una
% carga Resitiva pura

if Factor_Potencia==1
app.TipodeCargaDropDown.Value="Resistivo";
end

% Calculamo el angulo phi y le añadimos el signo en
% funcion de si es capcitiva(-) o inductiva(+)
Tipo_carga=app.TipodeCargaDropDown.Value;

if Tipo_carga== "Inductivo"

```

```

        phi= acos(Factor_Potencia); % phi en radianes (+)

elseif Tipo_carga== "Capacitivo"

        phi= (-1)*acos(Factor_Potencia); % phi en radianes (-)

else % Caso resistivo
        phi=0;

end

% Datos de la carga por fase para calcular Ztotal por
% fase;
I_carga=S_carga/(sqrt(3)*Un);
Z_carga=(Un/sqrt(3))/I_carga;
R_carga=Z_carga*cos(phi);
X_carga=Z_carga*sin(phi);

% Calculamos la Ztotal
w=2*pi*f_val_entrada;% frecuencia variable en rad/s
Xs=w*Ls;
Rtotal=R_carga;
Xtotal=X_carga+(Xs);
Ztotal=sqrt((Rtotal^2)+(Xtotal^2));
% Calculamos la intensidad real por fase;
I_carga_real=E0_val/Z_carga;
app.Output_I_Carga.Value=I_carga_real;
% Calculamos la tensión de la carga Ureal(kV);
V=I_carga_real*Ztotal;
U=sqrt(3)*V;
app.Output_UKv_Carga.Value=U/(10^3);% Pasamos de V a kV

% Calculamos la potencia activa P(kW);
P=sqrt(3)*U*I_carga_real*cos(phi);
app.Output_PKw_Carga.Value=P/(10^3); %En kW;

% Calculamos la potencia reactiva Q(kVAr);

Q=sqrt(3)*U*I_carga_real*sin(phi);
app.Output_QKvar_Carga.Value=Q/(10^3);% En kVAr;

end

end

function Limpiar_Situaciones_Funcionamiento(app)

    if app.Vacio_Button.Value == 1

        app.Output_U0_vacio.Value=0;
        app.Output_E0_vacio.Value=0;
        app.Output_f_Ensayo_vacio.Value=0;

    elseif app.Cortocircuito_Button.Value == 1

```

```

app.Output_E0_cc.Value=0;
app.Output_I_cc.Value=0;
app.Output_f_Ensayo_cc.Value=0;

else

    app.Output_I_Carga.Value=0;
    app.Output_UKv_Carga.Value=0;
    app.Output_PKw_Carga.Value=0;
    app.Output_QKvar_Carga.Value=0;
    app.Output_f_Ensayo_Carga.Value=0;

end

end

function [error1,error2]= Evaluacion_Errores(app)

% Comprobamos previamente si puede haber algún error con
% los datos introducidos por el usuario, comparándolos con
% los valores máximos de los datos característicos de la
% máquina;

Ie_Max=app.Input_Ie_Max.Value; % Intensidad de excitación
máxima;

n_Max=app.Input_n_Max.Value;% rpms máxima de diseño;

Ie_regulacion=app.Input_regulacion_Ie.Value;
n_regulacion=app.Input_regulacion_n.Value;

% Evaluacion de errores posibles

error1=0;% Inicializamos a 0 sin errores
error2=0;% Inicializamos a 0 sin errores

if Ie_regulacion>Ie_Max

    error1=1;

elseif n_regulacion>n_Max

    error2=1;

end

end

end

% Callbacks that handle component events
methods (Access = private)

% Button pushed function: Confirmar_CN_Button
function Confirmar_CN_ButtonPushed(app, event)

```

```

%Calculamos la In con los datos de la tabla de características
%para mostrarlo en pantalla;

SN=app.Input_Sn_CN.Value*10^3; % en KVA input;
UN=app.Input_Un_CN.Value*10^3; % en KV input;
app.Output_In_CN.Visible='on';
app.Output_Par_Polos_CN.Visible='on';
app.Output_In_CN.Value =(SN/(sqrt(3)*UN));

% Calculamos el número de pares de polos;

fn=app.Input_fn_CN.Value;
n=app.Input_n_rpm_CN.Value;
app.Output_Par_Polos_CN.Value= (60*fn)/n;

end

% Button pushed function: Confirmar_Datos_Ensayos_Button
function Confirmar_Datos_Ensayos_ButtonPushed(app, event)

% --Mostrar Datos en el circuito Equivalente en condiciones
nominales: Xs y E0--

%Recopilación de Datos;

U0=app.Input_Uo.Value*10^3; %En Kv input y pasamos a v
E0_ens=U0/sqrt(3);
Ie_o=app.Input_Ie_o.Value;
m=E0_ens/Ie_o;% m=(E0/Ie_0) a frecuencia ensayo vacio;

% Calcular la frecuencia del ensayo con el dato de n0(rpm) y
% p(pares de polos);
p=app.Output_Par_Polos_CN.Value; % Recuperamos el dato calculado de
pares de polos
n0=app.Input_n_o_Ensayo.Value;
f_ensayo_vacio=(p*n0)/60;

% Recopilar datos de Input necesarios para más abajo;

Icc=app.Input_Icc.Value; % Dato a frecuencia de ensayo de CC;
n_cc=app.Input_n_cc_Ensayo.Value;
Ie_cc=app.Input_Ie_cc_Ensayo.Value;
fn=app.Input_fn_CN.Value;
f_ensayo_cc=(p*n_cc)/60;
m1= m*(fn/f_ensayo_vacio); % m1=(E0/Ie_0) a frecuencia nominal;

% Calcular la Intensidad de excitación Nominal;
Un=app.Input_Un_CN.Value*10^3;
E0_n=Un/sqrt(3);
Ie_N=E0_n/m1;
app.Output_Ie_n_CN.Visible='on';
app.Output_Ie_n_CN.Value=Ie_N;

% m2=(E0/Ie_0) a frecuencia de ensayo de cc;

m2= m1*(f_ensayo_cc/fn);
E0cc= m2*Ie_cc; % A frecuencia de ensayo de cc;
Xs_cc=E0cc/Icc; % Reactancia Sincrona a frecuencia de ensayo de cc;

```

```

Ls=Xs_cc/(2*pi*f_ensayo_cc); % La inductancia sincrona es una
constante;

% Datos de salida de CÁLCULOS DE ENAYOS;

app.Output_f_vacio.Value=f_ensayo_vacio;
app.Output_Eo.Value=app.Input_Un_CN.Value/sqrt(3);
app.Output_f_cc.Value=f_ensayo_cc;
app.Output_Xs.Value=Xs_cc;
app.Output_Ls.Value=Ls;

% Motrar datos en la segunda pestaña de Situaciones de
% funcionamiento nominales;

app.E0_n_vacio.Value=app.Input_Un_CN.Value/sqrt(3); % en Kv;
app.E0_n_cc.Value=app.Input_Un_CN.Value/sqrt(3); % en Kv;
app.U0_n_vacio.Value=app.Input_Un_CN.Value; % en Kv;
app.f_Nominal_vacio.Value=fn;
app.f_Nominal_cc.Value=fn;
app.f_Nominal_Carga.Value=fn;
app.I_n_cc.Value=app.Output_In_CN.Value;
app.UnKv_Nominal_Carga.Value =app.Input_Un_CN.Value;

% -----Calcular la curva de ensayo de vacío-----
x=0:0.01:500;
y=m*x;

% Comprobamos si el ensayo se realiza a frecuencia nominal o
% distinta;

if fn==f_ensayo_vacio

fn_string=num2str(fn);

%Mostrar punto del ensayo vacio cuando activemos el checkbox;

Value_vacio=app.Situacion_Vacio_CheckBox.Value;

if Value_vacio==1

plot(app.UIAxes,x,y,"b-",Ie_o,E0_ens,"r*");
legend(app.UIAxes,"fn = "+fn_string+" (Hz)","Situación de Ensayo
Vacío");

else
plot(app.UIAxes,x,y,"b-");
legend(app.UIAxes,"fn = "+fn_string+" (Hz)");
end

else

% Calculamos la pendiente de m'=(E0/Ie_0) a frecuencia
% nominal, ya una vez conocido m=(E0/Ie_0) a frecuencia de
% ensayo de vacio;

```

```

m1= m*(fn/f_ensayo_vacio);
y1=m1*x;

fn_string=num2str(fn);
f_ensayo_string=num2str(f_ensayo_vacio);

Value2=app.Situacion_Vacio_CheckBox.Value;
if Value2==1

plot(app.UIAxes,x,y,"b-",x,y1,"g-",Ie_o,E0_ens,"r*");
legend(app.UIAxes,"fensayo = "+f_ensayo_string+" (Hz)", ...
        "fn = "+fn_string+" (Hz)","Situación de Ensayo Vacio")

else
plot(app.UIAxes,x,y,"b-",x,y1,"g-");
legend(app.UIAxes,"fensayo = "+f_ensayo_string+" (Hz)","fn =
"+fn_string+" (Hz)");
end
end

end

% Selection changed function: Situaciones_Button_Group
function Situaciones_Button_GroupSelectionChanged(app, event)

vacio= app.Vacio_Button.Value;
corto= app.Cortocircuito_Button.Value;
carga= app.Carga_Button.Value;

if vacio == 1
    %Dejamos como no visible la pestaña de "Datos de la carga";
    app.Introducir_Datos_Carga_Panel.Visible="off";
    %Deshabilitamos el resto de paneles
    app.SituacionVacioPanel.Enable="on";
    app.Situacion_Cortocircuito_Panel.Enable="off";
    app.Situacion_Carga_Panel.Enable="off";
elseif corto==1
    % Dejamos como no visible la pestaña de "Datos de la carga";
    app.Introducir_Datos_Carga_Panel.Visible="off";
    %Deshabilitamos el resto de paneles;
    app.SituacionVacioPanel.Enable="off";
    app.Situacion_Cortocircuito_Panel.Enable="on";
    app.Situacion_Carga_Panel.Enable="off";

elseif carga==1
    % Activamos la pestaña de "Datos de la carga";
    app.Introducir_Datos_Carga_Panel.Visible="on";
    %Deshabilitamos el resto de paneles;
    app.SituacionVacioPanel.Enable="off";
    app.Situacion_Cortocircuito_Panel.Enable="off";
    app.Situacion_Carga_Panel.Enable="on";
end

end
end

```

```

% Selection changed function: OpcionesTipoCarga
function OpcionesTipoCargaSelectionChanged(app, event)
    %selectedButton = app.OpcionesTipoCarga.SelectedObject;
    manual=app.Manual_Button.Value;
    selector=app.Selector_Button.Value;

    if manual==1
        % Desactivamos el selector;
        app.SELECTORPanel.Enable="off";
        % Activamos la opcion editable de los cuadros de texto de los
parámetros de
        % regulación
        app.Input_regulacion_Ie.Editable="on";
        app.Input_regulacion_n.Editable="on";

    elseif selector==1
        % Activamos el selector;
        app.SELECTORPanel.Enable="on";
        % Desactivamos la opcion editable de los cuadros de texto de los
parámetros de
        % regulación
        app.Input_regulacion_Ie.Editable="off";
        app.Input_regulacion_n.Editable="off";

    end

end

% Value changed function: Interruptor_Marcha_Manual
function Interruptor_Marcha_ManualValueChanged(app, event)

    value = app.Interruptor_Marcha_Manual.Value;

    if strcmp(value, '0n') % Si el interruptor de marcha encendido;

        [error1,error2]=Evaluacion_Errores(app); %Llamamos a la funcion que nos
devuelve el tipo de error

        if error1==1

            mensaje={'Se ha sobrepasado la intensidad máx de
excitación.', 'Considerar bajar la intesidad (Ie) a valores nominales o
inferiores.'};
            titulo= 'ADVERTENCIA';
            uialert(uifigure,mensaje,titulo, 'Icon', 'warning');

            app.LED_Marcha_Manual.Color='yellow'; % Encendemos el LED en
amarillo del boton manual;
            Calcular_Situaciones_Funcionamiento(app);

        elseif error2==1

```

```

        mensaje={'Se ha sobrepasado la velocidad de giro
nominal','Considerar bajar la velocidad de giro (n) a valores nominales o
inferiores.'};
        titulo= 'ADVERTENCIA';
        uialert(uifigure,mensaje,titulo,'Icon','warning');

        app.LED_Marcha_Manual.Color='yellow'; % Encendemos el LED en
amarillo del boton manual;
        Calcular_Situaciones_Funcionamiento(app);

    else % Si no hay errores

        Calcular_Situaciones_Funcionamiento(app); % Llamamos a la función
para hacer los cálculos y presentarlos;
        app.LED_Marcha_Manual.Color='green'; % Encendemos el LED en verde;
    end

    else % Si el interruptor de marcha manual está apagado;
        app.LED_Marcha_Manual.Color='red'; % Encender el LED rojo;
        Limpiar_Situaciones_Funcionamiento(app);% Llamamos a la función
para volver a la situación de inicio;
    end

end

% Value changing function: Input_Select_Ie
function Input_Select_IeValueChanging(app, event)

    app.Input_regulacion_Ie.Value=event.Value;
    if app.autoUpdate

        [error1,error2]=Evaluacion_Errores(app); %Llamamos a la funcion
que nos devuelve el tipo de error

        if error1==1

            app.LED_AutoUpdate.Color='yellow'; % Encendemos el LED
en amarillo del Auto;
            mensaje={'Se ha sobrepasado la intensidad nominal de
excitación.','Considerar bajar la intesidad (Ie) a valores nominales o
inferiores.'};
            titulo= 'ADVERTENCIA';
            uialert(uifigure,mensaje,titulo,'Icon','warning');

        elseif error2==1

            app.LED_AutoUpdate.Color='yellow'; % Encendemos el LED
en amarillo del Auto;
            mensaje={'Se ha sobrepasado la velocidad de giro
nominal','Considerar bajar la velocidad de giro (n) a valores nominales o
inferiores.'};
            titulo= 'ADVERTENCIA';
            uialert(uifigure,mensaje,titulo,'Icon','warning');

```

```

else
    app.LED_AutoUpdate.Color = [0 0 1] ; % Encendemos el LED
en azul del Auto// Funcionamiento correcto
    app.Interruptor_Marcha_Manual.Enable = 'off' ;
    app.Interruptor_Marcha_Manual.Value = 'Off';
    app.LED_Marcha_Manual.Color= [1 0 0];
end
    Calcular_Situaciones_Funcionamiento(app);
end
end

% Value changing function: Input_Select_n
function Input_Select_nValueChanged(app, event)

    app.Input_regulacion_n.Value=event.Value;
    if app.autoUpdate

        [error1,error2]=Evaluacion_Errores(app); %Llamamos a la funcion
que nos devuelve el tipo de error

        if error1==1

en amarillo del Auto;
            app.LED_AutoUpdate.Color='yellow'; % Encendemos el LED
            mensaje={'Se ha sobrepasado la intensidad nominal de
excitación.', 'Considerar bajar la intensidad (Ie) a valores nominales o
inferiores.'};
            titulo= 'ADVERTENCIA';
            uialert(uifigure,mensaje,titulo,'Icon','warning');

        elseif error2==1

en amarillo del Auto;
            app.LED_AutoUpdate.Color='yellow'; % Encendemos el LED
            mensaje={'Se ha sobrepasado la velocidad de giro
nominal', 'Considerar bajar la velocidad de giro (n) a valores nominales o
inferiores.'};
            titulo= 'ADVERTENCIA';
            uialert(uifigure,mensaje,titulo,'Icon','warning');

        else
en azul del Auto// Funcionamiento correcto
            app.LED_AutoUpdate.Color = [0 0 1] ; % Encendemos el LED
            app.Interruptor_Marcha_Manual.Enable = 'off' ;
            app.Interruptor_Marcha_Manual.Value = 'Off';
            app.LED_Marcha_Manual.Color= [1 0 0];
            end
            Calcular_Situaciones_Funcionamiento(app);
        end
    end
end

% Value changed function: Interruptor_AutoUpdate
function Interruptor_AutoUpdateValueChanged(app, event)

if strcmp(app.Interruptor_AutoUpdate.Value, 'On')
    app.autoUpdate = 1 ;

```

```

[error1,error2]=Evaluacion_Errores(app); %Llamamos a la funcion
que nos devuelve el tipo de error

    if error1==1

        mensaje={'Se ha sobrepasado la intensidad nominal de
excitación.', 'Considerar bajar la intensidad (Ie) a valores nominales o
inferiores.'};
        titulo= 'ADVERTENCIA';
        uialert(uifigure,mensaje,titulo,'Icon','warning');

        app.LED_AutoUpdate.Color='yellow'; % Encendemos el LED en
amarillo del Auto;
        Calcular_Situaciones_Funcionamiento(app);

    elseif error2==1

        mensaje={'Se ha sobrepasado la velocidad de giro
nominal', 'Considerar bajar la velocidad de giro (n) a valores nominales o
inferiores.'};
        titulo= 'ADVERTENCIA';
        uialert(uifigure,mensaje,titulo,'Icon','warning');

        app.LED_AutoUpdate.Color='yellow'; % Encendemos el LED en
amarillo del Auto;
        Calcular_Situaciones_Funcionamiento(app);

    else % Si no hay errores

        app.LED_AutoUpdate.Color = [0 0 1] ; % Encendemos el LED en azul
del Auto// Funcionamiento correcto
        app.Interruptor_Marcha_Manual.Enable = 'off' ;
        app.Interruptor_Marcha_Manual.Value = 'Off';
        app.LED_Marcha_Manual.Color= [1 0 0];
        Calcular_Situaciones_Funcionamiento(app); % Llamamos a la
función para hacer los cálculos y presentarlos;

    end

else

    app.autoUpdate = 0 ;
    app.LED_AutoUpdate.Color = [0.94,0.94,0.94] ; % Si no está en
funcionamiento Auto cambiar color LED
    app.Interruptor_Marcha_Manual.Enable = 'on' ;
    Limpiar_Situaciones_Funcionamiento(app);% Llamamos a la funcion
para volver a la situación de inicio;

end

end

end

% Component initialization
methods (Access = private)

    % Create UIFigure and components

```

```
function createComponents(app)
end
end

% App creation and deletion
methods (Access = public)

% Construct app
function app = Generador_Sincrono_TFG

    % Create UIFigure and components
    createComponents(app)

    % Register the app with App Designer
    registerApp(app, app.UIFigure)

    if nargin == 0
        clear app
    end
end

% Code that executes before app deletion
function delete(app)

    % Delete UIFigure when app is deleted
    delete(app.UIFigure)
end
end
end
```