



**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**

**FACULTAD DE CIENCIAS**

**CARRERA DE INGENIERÍA QUÍMICA**

**“MODELACIÓN, SIMULACIÓN Y CONTROL DE UN  
EVAPORADOR DE DOBLE EFECTO COMO HERRAMIENTA  
DIDÁCTICA PARA CONTROL DE PROCESOS.”**

**Trabajo de Titulación**

Tipo: Propuesta Tecnológica

Presentado para optar al grado académico de:

**INGENIERO QUÍMICO**

**AUTOR: JERRY LEODAN MENA NAGUA**

**DIRECTOR: Ing. DARÍO FERNANDO GUAMÁN LOZADA MSc.**

Riobamba - Ecuador

2021

**©2021, Jerry Leodan Mena Nagua**

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento, siempre y cuando se reconozca el Derecho de Autor.

Yo, JERRY LEODAN MENA NAGUA, declaro que el presente trabajo de titulación es de mi autoría y que los resultados del mismo son auténticos. Los textos en el documento que provienen de otra fuente están debidamente citados y referenciados.

Como autor, asumo la responsabilidad legal y académica de los contenidos de este trabajo de titulación. El patrimonio intelectual pertenece a la Escuela Superior Politécnica de Chimborazo.

Riobamba, 31 de agosto del 2021



**Jerry Leodan Mena Nagua**

**190048675-2**

**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**  
**FACULTAD DE CIENCIAS**  
**CARRERA DE INGENIERÍA QUÍMICA**

El Tribunal de Trabajo de Titulación certifica que: El Trabajo de Titulación: Tipo: Propuesta Tecnológica, “**MODELACIÓN, SIMULACIÓN Y CONTROL DE UN EVAPORADOR DE DOBLE EFECTO COMO HERRAMIENTA DIDÁCTICA PARA CONTROL DE PROCESOS**”, realizado por el señor: **JERRY LEODAN MENA NAGUA**, ha sido minuciosamente revisado por los Miembros del Tribunal del Trabajo de Titulación, el mismo que cumple con los requisitos científicos, técnicos, legales, en tal virtud el Tribunal Autoriza su presentación.

	<b>FIRMA</b>	<b>FECHA</b>
Ing. Mónica Lilian Andrade Avalos MSc. <b>PRESIDENTE DEL TRIBUNAL</b>	.....	2021-09-09
Ing. Darío Fernando Guamán Lozada MSc. <b>DIRECTOR DEL TRABAJO DE TITULACIÓN</b>	.....	2021-09-09
Ing. Adrián Alejandro Rodríguez Pinos MSc. <b>MIEMBRO DE TRIBUNAL</b>	.....	2021-09-09

## **DEDICATORIA**

Dedico el presente trabajo de titulación a Dios, y de manera especial a mis padres Marizol y Bayron ejemplos de disciplina, humildad y respeto, quienes me han apoyado incondicionalmente durante el transcurso de mi vida y me han dado valor para no dejar de lado mi carrera profesional, a mi hermano Stalin y a cada uno de mis familiares quienes con sus oraciones y palabras alentadores me motivaron a seguir adelante.

Jerry

## **AGRADECIMIENTO**

El más sincero agradecimiento en primer lugar a Dios y a la Virgen Santísima del Cisne por acompañarme siempre en cada uno de los pasos que doy, cubriéndome siempre con su manto misericordioso, por ser mi luz en este mundo lleno de oscuridad.

A mis padres y familiares por apoyarme e instruirme en valores los cuales me han llevado a culminar con éxito una de mis tantas metas propuestas.

A mis amigos por brindarme su amistad desinteresada y ayudarme a seguir con mi carrera.

A la Escuela Superior Politécnica de Chimborazo, al Ing. Darío Guamán e Ing. Adrián Rodríguez quienes me brindaron los conocimientos necesarios y me direccionaron por el camino correcto para la culminación del presente proyecto.

Jerry

## TABLA DE CONTENIDOS

ÍNDICE DE TABLAS.....	ix
ÍNDICE DE FIGURAS.....	x
ÍNDICE DE GRÁFICOS.....	xii
ÍNDICE DE ANEXOS.....	xiii
ÍNDICE DE ABREVIATURAS.....	xiv
RESUMEN .....	xvi
ABSTRACT.....	xvii

### CAPÍTULO I

1.	INTRODUCCIÓN.....	1
1.1.	Antecedentes .....	1
1.2.	Planteamiento del problema.....	2
1.3.	Justificación del proyecto.....	3
1.4.	Objetivos.....	3
1.4.1.	<i>Objetivo General</i> .....	3
1.4.2.	<i>Objetivos Específicos</i> .....	3
1.5.	Alcance.....	4

### CAPÍTULO II

2.	MARCO TEÓRICO.....	5
2.1.	Estado del Arte .....	5
2.1.1.	<i>Simulación dinámica en evaporadores de múltiple efecto</i> .....	5
2.1.2.	<i>Simulación ingenieril con MATLAB</i> .....	6
2.2.	Bases Teóricas.....	7
2.2.1.	<i>Evaporación</i> .....	7
2.2.1.1.	<i>Evaporadores de efecto múltiple</i> .....	7
2.2.2.	<i>Simulación dinámica</i> .....	9
2.2.2.1.	<i>Simulación de procesos</i> .....	9
2.2.3.	<i>Control automático de procesos</i> .....	10
2.2.3.1.	<i>Generalidades de los sistemas de control</i> .....	10
2.2.3.2.	<i>Tipos de sistemas de control</i> .....	12
2.2.3.3.	<i>Estrategias de control</i> .....	12

2.2.3.4.	<i>Sintonización de controladores</i> .....	14
2.2.4.	<b>Software usado</b> .....	15
2.2.4.1.	<i>Matlab (MATrix LABoratory)</i> .....	15
2.2.4.2.	<i>Herramienta de MATLAB: Simulink</i> .....	15
2.2.4.3.	<i>Herramienta de MATLAB: App Designer</i> .....	16
2.2.4.4.	<i>Herramienta Simulink Compiler</i> .....	21
2.2.4.5.	<i>Herramienta Matlab Compiler</i> .....	21
2.3.	<b>Marco Legal</b> .....	22

### CAPÍTULO III

3.	<b>METODOLOGÍA</b> .....	23
3.1.	<b>Etapas de la metodología</b> .....	23
3.2.	<b>Establecer parámetros de trabajo del proceso</b> .....	24
3.2.1.	<i>Información del proceso</i> .....	24
3.2.2.	<i>Valores numéricos para usar dentro del proceso</i> .....	25
3.3.	<b>Modelar las ecuaciones necesarias para la simulación</b> .....	26
3.3.1.	<i>Balance de masa</i> .....	26
3.3.1.1.	<i>Balance de masa en el primer efecto:</i> .....	26
3.3.1.2.	<i>Balance de componentes (solute) en el primer efecto:</i> .....	27
3.3.1.3.	<i>Balance de masa en el segundo efecto:</i> .....	27
3.3.1.4.	<i>Balance de componentes (solute) en el efecto:</i> .....	27
3.3.2.	<i>Balance de energía</i> .....	28
3.3.2.1.	<i>Balance de energía en el primer efecto:</i> .....	28
3.3.2.2.	<i>Balance de energía en el segundo efecto:</i> .....	30
3.3.2.3.	<i>La tasa de vapor para el primer efecto obedece a:</i> .....	31
3.3.2.4.	<i>La tasa de vapor para el segundo efecto obedece a:</i> .....	32
3.4.	<b>Simular el comportamiento del sistema dinámico en Simulink</b> .....	32
3.4.1.	<i>Pasos para seguir en simulink</i> .....	32
3.5.	<b>Diseñar el sistema de control a implementarse</b> .....	35
3.5.1.	<i>Objetivos de control</i> .....	35
3.5.2.	<i>Esquema del sistema de control a implementarse</i> .....	35
3.5.2.1.	<i>Elemento final de control</i> .....	36
3.6.	<b>Desarrollar el sistema de control del proceso en simulink</b> .....	36
3.7.	<b>Construir la interfaz gráfica</b> .....	38
3.7.1.	<i>Interfaz de usuario: inicio</i> .....	40



3.7.2.	<i>Interfaz de usuario: Principal</i> .....	41
3.7.3.	<i>Interfaz de usuario: info</i> .....	48
3.8.	<b>Empaquetar la interfaz gráfica diseñada</b> .....	48

## CAPÍTULO IV

4.	<b>GESTIÓN DEL PROYECTO</b> .....	51
4.1.	<b>Recursos y materiales</b> .....	51
4.1.1.	<i>Presupuesto</i> .....	51
4.1.2.	<i>Recursos materiales</i> .....	51
4.2.	<b>Cronograma</b> .....	52
4.3.	<b>Resultados</b> .....	53
4.3.1.	<i>Informe de resultados de la simulación</i> .....	53
4.3.2.	<i>Informe de resultados de la implementación del sistema de control</i> .....	54
4.3.2.1.	<i>Controladores usados dentro del sistema de control</i> .....	55
4.3.2.2.	<i>Análisis de las gráficas resultantes de la implementación del sistema de control</i> ..	55
4.3.3.	<i>Informe de resultados de la aplicación generada</i> .....	59
4.3.4.	<i>Validación de la modelación programada</i> .....	60
4.3.4.1.	<i>Concentración del producto (<math>X_p</math>)</i> .....	61
4.3.4.2.	<i>Acumulación en el primer efecto (<math>M1</math>)</i> .....	62
	<b>CONCLUSIONES</b> .....	64
	<b>RECOMENDACIONES</b> .....	65
	<b>GLOSARIO</b>	
	<b>BIBLIOGRAFÍA</b>	
	<b>ANEXOS</b>	

## ÍNDICE DE TABLAS

<b>Tabla 1 – 2:</b> Opciones existentes en la barra de herramientas presentes en App Designer.....	17
<b>Tabla 2 – 2:</b> Opciones existentes en la biblioteca de componentes presentes en App Designer	18
<b>Tabla 3 – 2:</b> Opción de editor existentes en la barra de herramientas presentes en App Designer .....	20
<b>Tabla 4 – 2:</b> Norma legal sobre utilización del software .....	22
<b>Tabla 1 – 3:</b> Valores de estado estacionario y de los parámetros físicos .....	25
<b>Tabla 2 – 3:</b> Pares de control seleccionados dentro del proceso .....	35
<b>Tabla 1 – 4:</b> Presupuesto general del proyecto de titulación.....	51
<b>Tabla 2 – 4:</b> Cronograma para la realización del proyecto de titulación.....	52
<b>Tabla 3 – 4:</b> Valores obtenidos para las constantes del controlador. ....	55
<b>Tabla 4 – 4:</b> Resultados y error porcentual (Xp).....	62
<b>Tabla 5 – 4:</b> Resultados y error porcentual (M1). ....	63

## ÍNDICE DE FIGURAS

<b>Figura 1 – 2.</b> (a) Alimentación directa, (b) alimentación inversa, (c) alimentación mixta, (d) alimentación paralela.....	8
<b>Figura 2 – 2.</b> Sistema de control en lazo abierto.....	12
<b>Figura 3 – 2.</b> Sistema de control en lazo cerrado.....	12
<b>Figura 4 – 2.</b> Diagrama de bloques de un lazo de control feedback.....	13
<b>Figura 5 – 2.</b> Sintonía de controladores.....	14
<b>Figura 6 – 2.</b> Ventana de inicio de App Designer.....	16
<b>Figura 7 – 2.</b> Ventana design view de App Designer.....	17
<b>Figura 8 – 2.</b> Ventana code view de App Designer.....	20
<b>Figura 1 – 3.</b> Diagrama del proceso que va a implementarse.....	24
<b>Figura 2 – 3.</b> Formato del ingreso de la ecuación en Simulink.....	33
<b>Figura 3 – 3.</b> Subsistemas creados dentro del entorno Simulink.....	33
<b>Figura 4 – 3.</b> Máscaras de subsistema para los procesos creados dentro del entorno Simulink.....	34
<b>Figura 5 – 3.</b> Script main de Matlab para definir las variables.....	34
<b>Figura 6 – 3.</b> Esquema del Sistema de control a implementarse.....	36
<b>Figura 7 – 3.</b> Configuración del bloque Step dentro de simulink.....	36
<b>Figura 8 – 3.</b> Configuración del bloque PID Controller dentro de Simulink.....	37
<b>Figura 9 – 3.</b> Esquema del modelo de simulink implementado.....	38
<b>Figura 10 – 3.</b> Esquema de la perturbación implementada.....	38
<b>Figura 11– 3.</b> Pasos para crear una estructura dentro de matlab.....	39
<b>Figura 12 – 3.</b> Exportar los resultados como señales de salida al workspace de matlab.....	39
<b>Figura 13 – 3.</b> Vista preliminar de la interfaz inicio y de los uicontrols utilizados.....	40
<b>Figura 14 – 3.</b> Líneas de código para comunicar las interfaces inicio e interfaz.....	40
<b>Figura 15 – 3.</b> Vista preliminar de la plantilla interfaz y de los uicontrols utilizados.....	41
<b>Figura 16 – 3.</b> Código que se ejecuta después de la creación de los componentes.....	42
<b>Figura 17 – 3.</b> Código que almacena la información que se cargará por defecto.....	42
<b>Figura 18 – 3.</b> Esquema de cómo se procede a llenar cada una de las tablas.....	43
<b>Figura 19 – 3.</b> Líneas de código ingresado para que aparezcan los valores por defecto.....	43
<b>Figura 20 – 3.</b> Líneas de código para configurar los parámetros de entrada.....	44
<b>Figura 21 – 3.</b> Línea de código para configurar la entrada simInp para el despliegue de la interfaz.....	44
<b>Figura 22 – 3.</b> Líneas de código para ejecutar el modelo de simulink dentro de la interfaz.....	45
<b>Figura 23 – 3.</b> Línea de código para extraer los resultados de la simulación del modelo de simulink dentro de la interfaz.....	46

<b>Figura 24 – 3.</b> Línea de código para graficar los valores del set point.....	47
<b>Figura 25 – 3.</b> Línea de código para graficar los valores del set point.....	47
<b>Figura 26 – 3.</b> Línea de código para desplegar la información contenida en la interfaz de usuario .....	48
<b>Figura 27 – 3.</b> Vista preliminar de la plantilla info.mlapp y de los uicontrols utilizados .....	48
<b>Figura 28 – 3.</b> Localización de share en la ventana de herramientas de App designer .....	49
<b>Figura 29 – 3.</b> Distribución de la ventana de App Compiler .....	49
<b>Figura 30 – 3.</b> Resultado exitoso del empaquetamiento .....	50
<b>Figura 1 – 4.</b> Esquema de la simulación completa del evaporador de doble efecto.....	53
<b>Figura 2 – 4.</b> Esquema de la simulación detallada del evaporador de doble efecto .....	53
<b>Figura 3 – 4.</b> Esquema del sistema de control que se va a implementar .....	54
<b>Figura 4 – 4.</b> Esquema de los recursos implementados dentro del sistema de control .....	54
<b>Figura 5 – 4.</b> Representación gráfica de los resultados de salida del elemento final de control	57
<b>Figura 6 – 4.</b> Representación gráfica del resultado del sistema de control para la variable controlada Xp .....	58
<b>Figura 7 – 4.</b> Representación gráfica del resultado del sistema de control para la variable controlada M1.....	58
<b>Figura 8 – 4.</b> Representación gráfica del resultado del sistema de control para la variable controlada M2.....	59
<b>Figura 9 – 4.</b> Resultado bibliográfico de la concentración en el primer efecto.....	61
<b>Figura 10 – 4.</b> Simulación establecida para comprobar el funcionamiento de la aplicación .....	61
<b>Figura 11 – 4.</b> Resultado bibliográfico de la acumulación en el primer efecto.....	62
<b>Figura 12 – 4.</b> Segunda simulación establecida para comprobar el funcionamiento de la aplicación (acumulación en el primer efecto).....	62

## ÍNDICE DE GRÁFICOS

<b>Gráfico 1 – 3.</b> Metodología propuesta para el trabajo de titulación .....	23
<b>Gráfico 1 – 4.</b> Estructura de la aplicación final (.exe).....	59

## **ÍNDICE DE ANEXOS**

**ANEXO A. MANUAL DE USUARIO DE LA APLICACIÓN DEL SISTEMA DE CONTROL  
DE UN EVAPORADOR DE DOBLE EFECTO**

**ANEXO B. CÓDIGO DE APP DESIGNER IMPLEMENTADO PARA EL CORRECTO  
FUNCIONAMIENTO DE LA INTERFAZ DISEÑADA**

## ÍNDICE DE ABREVIATURAS

<b>A</b>	Acumulación
<b>M1</b>	Acumulación masa Tanque 1
<b>M2</b>	Acumulación masa Tanque 2
<b>A1</b>	Área de transferencia de calor en el Tanque 1
<b>A2</b>	Área de transferencia de calor en el Tanque 2
<b><math>\lambda_2</math></b>	Calor latente de vaporización ( $H1 - hc2$ )
<b><math>\lambda_2</math></b>	Calor latente de vaporización ( $HS - hS$ )
<b>F</b>	Caudal de alimentación
<b>S</b>	Caudal de vapor
<b>U1</b>	Coefficiente global de transferencia de calor para el Tanque 1
<b>U2</b>	Coefficiente global de transferencia de calor para el Tanque 2
<b>X</b>	Composición
<b>Xf</b>	Composición del alimento
<b>Xp</b>	Composición del tanque 1
<b>X2</b>	Composición del tanque 2
<b>Xf</b>	Composición tanque 2
<b>MPC</b>	control predictivo del modelo
<b>IHMPC</b>	controlador predictivo del modelo de horizonte infinito
<b>PID</b>	controlador proporcional, integral, derivativo
<b>f</b>	Cualquier función lineal o no lineal de las variables
<b>Ec.</b>	Ecuación
<b>F0</b>	El flujo de alimentación de entrada
<b>S0</b>	El flujo de vapor de entrada
<b>EPE</b>	Elevación Punto Ebullición
<b>H</b>	Entalpía del vapor
<b>H1</b>	Entalpía del vapor en función de T1
<b>HS</b>	Entalpía del vapor en función de TS
<b>h</b>	Entalpía en función de T1 y XP
<b>h2</b>	Entalpía en función de T2 y X2
<b>Hf</b>	Entalpía en función de Tf y Xf
<b>E</b>	Entrada
<b>V1</b>	Flujo de vapor del tanque 1
<b>V2</b>	Flujo de vapor del tanque 2
<b>hc1</b>	hS

<b>T<sub>s</sub></b>	La temperatura del vapor
<b>MSE</b>	Multiple stage evaporator – Evaporador de múltiples etapas
<b>MVR</b>	Recompresión mecánica de vapor
<b>C1</b>	S Taza de vapor primer efecto
<b>S</b>	Salida
<b>SP</b>	Set point
<b>P1</b>	Tasa de flujo del producto líquido del tanque 1
<b>P2</b>	Tasa de flujo del producto líquido del tanque 2
<b>T</b>	Temperatura
<b>T<sub>f</sub></b>	Temperatura de alimentación
<b>T<sub>S</sub></b>	Temperatura de entrada tanque 1
<b>T1</b>	Temperatura en el tanque 1
<b>T2</b>	Temperatura en el tanque 2
<b>t</b>	Tiempo
<b>C2</b>	V1 Taza de vapor segundo efecto
<b>(x)<sub>t</sub></b>	Variable de estado
<b>(u)<sub>t</sub></b>	Variable de estado



## RESUMEN

El presente trabajo de investigación modela, simula y controla un evaporador de doble efecto como herramienta didáctica para la cátedra de control de procesos a través de software. El modelamiento matemático del comportamiento del evaporador de doble efecto se realizó mediante balances de masa y energía para el primer y segundo efecto, obteniendo las respectivas ecuaciones diferenciales, que son ingresadas en un modelo en Simulink. Adicional a ello, para cumplir con los objetivos de control planteados, se requirió controlar las variables concentración del producto ( $X_p$ ), acumulación en el primer efecto y segundo efecto ( $M1 - M2$ ) para que se mantengan por debajo o en el punto de ajuste. La creación de la interfaz didáctica se desarrolló a partir de componentes básicos de la aplicación de Matlab “App Designer”, incluyendo comandos dentro de las líneas de código para que exista interacción entre el modelo de Simulink y la interfaz creada; dicho comando es procesado por Simulink Compiler y se empaquetó como aplicación independiente gracias a Matlab Compiler. Como resultado se obtuvo una interfaz gráfica que se ejecuta de manera independiente, en donde se pueden definir los diferentes valores de ajuste (SET-POINTS). El modelo matemático utilizado para la simulación se validó con el artículo “Simulation and Control of a Commercial Double Effect Evaporator: Tomato Juice” expuesto por Yadav & Jana (2010) cuyos resultados arrojan valores de consigna de: 0.2609 kg soluto/kg solución para la concentración del producto y 2268 kg para la acumulación en el primer efecto. Comparando dichos resultados con el presente trabajo se obtuvieron: 0.26067 kg soluto/kg, y 2268.067 kg respectivamente evidenciando un error relativo porcentual de 0.088% y -0.003% respectivamente, siendo valores de error aceptables. Se recomienda ampliar el estudio de la creación de aplicaciones independientes para profundizar el estudio del tema.

**Palabras clave:** <INGENIERÍA Y TECNOLOGÍA QUÍMICA>, <EVAPORADOR>, <MÚLTIPLE EFECTO>, <SIMULINK (SOFTWARE)>, <SIMULINK COMPILER (SOFTWARE)>, <APP DESIGNER (SOFTWARE)>, <MATLAB COMPILER (SOFTWARE)>.



1629-DBRA-UTP-2021

## ABSTRACT

The present research work models, simulates and controls a double effect evaporator as a didactic tool for the chair of process control through software. The mathematical modeling of the behavior of the double effect evaporator was carried out by means of mass and energy balances for the first and second effect, obtaining the respective differential equations, which are entered in a model in Simulink. In addition, to meet the set control objectives, it was necessary to control the variables concentration of the product ( $X_p$ ), accumulation in the first effect and second effect ( $M1 - M2$ ) so that they remain below or at the set point. The creation of the didactic interface was developed from the basic components of the Matlab application "App Designer", including commands within the code lines so that there is interaction between the Simulink model and the interface created; This command is processed by the Simulink Compiler and was packaged as a separate application thanks to Matlab Compiler. As a result, a graphical interface that operates independently was obtained, where the different adjustment values (SET-POINTS) can be defined. The mathematical model used for the simulation was validated with the article "Simulation and Control of a Commercial Double Effect Evaporator: Tomato Juice" presented by Yadav & Jana (2010) whose results show setpoint values of: 0.2609 kg solute / kg solution for the product concentration and 2268 kg for accumulation in the first effect. Comparing these results with the present work, the following data were obtained: 0.26067 kg solute / kg, and 2268.07 kg respectively, showing a relative percentage error of 0.088% and -0.003% respectively, being acceptable error values. It is recommended to broaden the study of the creation of independent applications to deepen the study of the subject.

**Key words:** <ENGINEERING AND CHEMICAL TECHNOLOGY>, <EVAPORATOR>, <MULTIPLE EFFECT>, <SIMULINK (SOFTWARE)>, <SIMULINK COMPILER (SOFTWARE)>, <APP DESIGNER (SOFTWARE)>, <MATLAB COMPILER (SOFTWARE)>.

# CAPÍTULO I

## 1. INTRODUCCIÓN

### 1.1. Antecedentes

La evolución del control desde una tecnología antigua hasta un campo moderno es una prueba fehaciente del crecimiento de la sociedad tecnológica moderna. Su estudio permite conocer cómo las teorías, tecnologías y las aplicaciones pueden interactuar en el desarrollo de una disciplina (Aëström & Kumar, 2014: p.3).

En pleno auge de la segunda guerra mundial alrededor de 1940, el control era una herramienta ampliamente utilizada para sistemas eléctricos, control de procesos, las telecomunicaciones y el manejo de navíos, pero no fue hasta 1960 años luego de la guerra que el control se instauró como una disciplina en donde la teoría del servomecanismo era la base teórica y la computación analógica se utilizaba tanto para la implementación de controladores como para herramienta de simulación. Sin embargo la época de oro empezó luego de los 60 ya que se profundizó en el estudio del control y simulación dinámica de distintos procesos (Aëström et al., 2014: p.4).

El conocimiento de la simulación dinámica permite comprender el funcionamiento de los distintos simuladores usados en la industria para el diseño, control, capacitación y soporte operativo de procesos, plataformas como Matlab / Simulink en donde se han diseñado modelos simples de simulación dinámica tales como columnas de destilación, reactores químicos, procesos de pH, microelectrónica, y como área de simulación emergente los procesos biológicos, son ampliamente utilizados en la investigación y la docencia (Komulainen et al., 2012: p.1).

En investigaciones actuales se ha profundizado el estudio de la simulación dinámica, tal es el caso de Luyben (2018, p.1) en su investigación titulada “Dynamic simulation of multi-effect evaporators”, en donde da las pautas para que la simulación sea exitosa, además menciona aspectos importantes como que:

*“La evaporación se utiliza para eliminar un componente muy volátil de una mezcla de productos químicos cuando el vapor generado en el flash de una etapa contiene cantidades muy pequeñas de los componentes menos volátiles. Cuando varios efectos están conectados en serie, el vapor generado en un recipiente de mayor temperatura aguas arriba se utiliza para generar vapor hacia el siguiente recipiente de menor temperatura aguas abajo. La configuración de efectos múltiples reduce en gran medida los requisitos de consumo de energía”* (Luyben, 2018, p.1).

Resumiendo lo planteado y acorde con los resultados de Komulainen et al. (2012: p.10) se plantea que la simulación dinámica es una herramienta didáctica muy útil para una amplia gama de cursos

de ingeniería química. El software de simulación permite demostrar y probar diferentes tipos de procesos y equipos, algoritmos de control y procedimientos de seguridad.

## **1.2. Planteamiento del problema**

Para que cualquier proceso dentro de la industria fluya de manera correcta es indispensable realizar un control, para ello es imperante identificar las distintas variables involucradas en el proceso, así como señalar el tipo de sistema de control a emplear sin dejar de lado la acción de control implicada dentro del ya mencionado sistema de control.

El análisis dinámico muestra que los controladores son realmente necesarios y pueden conducir a un mejor rendimiento del sistema en lo que respecta a la facilidad de funcionamiento y los logros económicos (Kumar, 2014, p.24).

La evaporación es un proceso importante dentro de la industria química, su objetivo es lograr una concentración evaporando un solvente generalmente agua para conseguir que el producto de salida contenga los parámetros adecuados en cuanto a sólidos. Cuando se conectan varios efectos en serie, el vapor generado en un recipiente de temperatura más alta corriente arriba se utiliza para generar vapor al siguiente recipiente de temperatura más baja corriente abajo. (Luyben, 2018, p.1).

Dentro de la formación profesional es considerable recalcar que el evaporador de doble efecto es usado en diferentes investigaciones por ejemplo Ben-ali (2018, p.1) en su investigación titulada como *Modeling of a double effect evaporator: Bond graph approach*, en donde el modelado del evaporador de doble efecto se lleva a cabo utilizando el enfoque del gráfico de enlace, este permite la descripción de sistemas interdisciplinarios que implican fenómenos energéticos, termodinámicos, hidráulicos y térmicos, otra investigación de relevancia propone un sistema de cristalización de evaporación por recompresión mecánica de vapor (MVR) de doble efecto conectado en paralelo, que combina el evaporador de película descendente con el evaporador de circulación forzada dicha investigación se muestra en su artículo denominado “*Design and evaluation of a parallel-connected double-effect mechanical vapor recompression evaporation crystallization system*” (Jiang et al., 2020: p.1), dichos estudios muestran la diversificación del evaporador de doble efecto dentro de los análisis investigativos en el campo de la ingeniería química.

En definitiva, el presente trabajo de titulación perfila a convertirse en una referencia para los estudiantes que buscan indagar en temas de carácter tecnológico ya que tópicos como modelado, simulación y control deben formar parte de los conocimientos adquiridos por los estudiantes de pregrado.

### **1.3. Justificación del proyecto**

“El estudio de la dinámica y el control de los procesos es un componente esencial de cualquier programa de estudios de ingeniería química” (Moodley, 2020, p.1).

La implementación de nuevas tecnologías enmarcadas en satisfacer necesidades en el ámbito educativo está creciendo exponencialmente ya que de esta manera se ha logrado transformar los métodos de aprendizaje habituales en métodos nuevos e innovadores mediante la aplicación de prácticas virtuales las cuales cubren la necesidad de poner en funcionamiento un laboratorio físico cuyo coste de ejecución es muy alto.

El requerimiento de vincular la teoría y práctica para entablar una concordancia en el desarrollo del aprendizaje influye de manera significativa en encontrar herramientas tecnológicas que permitan el desarrollo de prácticas de laboratorio de forma interactiva las cuales involucren los distintos cálculos requeridos en el diseño de un sistema de control.

El uso de un instrumento virtual de control de un evaporador de doble efecto permite desarrollar prácticas de laboratorios en concordancia con las exigencias tecnológicas actuales mediante el uso de un software de interfaz gráfica logrando que las distintas sustentaciones teóricas impartidas por los docentes sean entendidas de una manera más clara y el estudiante pueda observar cómo funcionan realmente los sistemas de control.

Dado que el presente proyecto se destina como herramienta en el contexto enseñanza – aprendizaje, se pretende sea de libre acceso para todas las personas que requieran profundizar en temas de control de procesos, se enmarca en la didáctica del perfeccionamiento académico de los estudiantes de la cátedra de control de procesos.

Para cubrir las cuestiones antes mencionadas, el presente proyecto de titulación propone el diseño de un instrumento virtual de control de un evaporador de doble efecto mediante el uso de un software, que sirva como herramienta clave en la enseñanza de la cátedra de control de procesos.

### **1.4. Objetivos**

#### ***1.4.1. Objetivo General***

Modelar, simular y controlar un evaporador de doble efecto como herramienta didáctica para control de procesos.

#### ***1.4.2. Objetivos Específicos***

- Modelar matemáticamente el comportamiento del evaporador de doble efecto.
- Programar la simulación y el control del evaporador de doble efecto mediante software.

- Realizar el manual de uso de la herramienta y validar la modelación programada.

### **1.5. Alcance**

En términos generales el presente trabajo de titulación utiliza recursos tecnológicos enfocados en el desarrollo de una interfaz gráfica mediante el uso de un modelo matemático que represente el comportamiento de un evaporador de doble efecto, así como la implementación de un sistema de control para el equipo antes mencionado. Como resultado se obtiene una aplicación de MATLAB también denominada interfaz gráfica de usuario la cual integra botones y gráficos que representan el funcionamiento del controlador en el equipo y permiten visualizar de manera clara las gráficas obtenidas de la compilación del programa, dicha interfaz se empaquetó como una aplicación que se puede utilizar fuera de Matlab para que sea más accesible para los estudiantes.

## CAPÍTULO II

### 2. MARCO TEÓRICO

#### 2.1. Estado del Arte

Diferentes investigaciones actuales tienen como premisa principal el estudio del diseño simultáneo del proceso y su control, no solo se centran en alcanzar fines económicos, sino que requieren alcanzar multiobjetivos los cuales se basan en distintos criterios e índices de controlabilidad y esto dependerá del compromiso existente entre cada uno de los propósitos. Teniendo en cuenta la complejidad inherente de los problemas de optimización multiobjetivo, el desafío es garantizar que el diseño óptimo del proceso y el sistema de control estén lo más cerca posible.

##### *2.1.1. Simulación dinámica en evaporadores de múltiple efecto*

El aumento de la demanda de energía, su elevado coste y los problemas de calentamiento global en todo el mundo exigen que las industrias como las fábricas de papel que consumen mucha energía mejoren su eficiencia energética, los evaporadores multietapa también conocidos como evaporadores de múltiple efecto son la unidad que más energía consume y requieren una sólida comprensión del comportamiento en estado estacionario y no estacionario para garantizar el ahorro de energía, para mejorar la dinámica del sistema de variación de la concentración de producto y de la temperatura de vapor en el flujo de alimentación, es necesario implementar una estrategia de control robusta.

Un controlador proporcional, integral, derivativo (PID) puede ser una buena opción para regular rápidamente la concentración de producto a través de la variación del caudal de licor de alimentación (Verma et al., 2017: pp.1-9).

La comprensión del rendimiento dinámico del Multiple stage evaporator – Evaporador de múltiples etapas (MSE) a través de un modelado dinámico detallado ha demostrado ser bastante útil para seguir eficientemente la trayectoria y controlar con facilidad las concentraciones de producto a través de la implementación de una estrategia Cascada – PID (Verma et al., 2018: pp.11-12).

Sousa & Fonseca (2019: p.11) en su investigación “Aplicación del control adaptativo feedforward - feedback en un proceso de evaporación de efecto múltiple” además de realizar el modelado, se evaluó la viabilidad de los lazos de control adaptativos feedforward y feedback para el contenido de sólidos en la salida de líquido de un evaporador de múltiple efecto. Aunque se identificó un comportamiento oscilatorio, los resultados mostraron que el PID adaptativo en el lazo de

realimentación mejoró el rendimiento del control, reduciendo el error general y el tiempo de establecimiento para la mayoría de los pasos de perturbación.

En relación con las implicaciones Nunis da Silva et al., (2018, p.2) en su investigación titulada “Aplicación del control predictivo de modelos a un cristalizador continuo de múltiples efectos” evalúan la viabilidad y las posibles ventajas de controlar y optimizar un proceso de cristalizador evaporativo de efectos múltiples mediante un control predictivo del modelo (MPC) lineal de horizonte infinito con control de zonas y objetivos de entrada óptimos. Además, se demuestra que el controlador predictivo del modelo de horizonte infinito IHMPC es más ventajoso que la estructura clásica de control PID para la optimización del proceso estudiado.

### ***2.1.2. Simulación ingenieril con MATLAB***

En los últimos años la diversificación del aprendizaje mediante el uso de diversas plataformas que contribuyen a la formación de los estudiantes ha constituido una herramienta de gran importancia en el campo educativo.

Tal es el caso de un estudio reciente titulado Enfoque de aprendizaje basado en la experiencia para la cinética química de Sucre-Rosales et al., (2020: p.1) en donde usan MATLAB y Python para resolver numéricamente el sistema de ecuaciones diferenciales asociado al modelo planteado y grafican los resultados mostrando el comportamiento del sistema.

Según Ibrahim (2011, p.2) los paquetes de simulación generales pueden utilizarse para simular todo tipo de problemas de ingeniería de todas las ramas de la ingeniería. Algunos ejemplos de paquetes de simulación general son Excel, MATLAB, MatriXx, etc. Las ventajas y desventajas de los paquetes de simulación general son:

- Coste moderado
- No se requieren conocimientos de programación
- Rapidez para simular y obtener resultados
- Se requiere un modelo del sistema que se va a simular y éste puede ser difícil de obtener
- El código fuente no está disponible y, por tanto, no es posible modificar/actualizar el paquete de simulación

Recapitulando, una de las tantas principales ventajas de utilizar MATLAB con Simulink es que el software es barato y está disponible en la mayoría de las instituciones de ingeniería (Moodley, 2020, p.2).



## 2.2. Bases Teóricas

### 2.2.1. Evaporación

La evaporación es una operación unitaria cuyo propósito es concentrar una disolución compuesta por un soluto no volátil y un solvente volátil, generalmente se usa agua como disolvente en la mayor parte de los procesos evaporativos. “La evaporación se realiza vaporizando una parte del disolvente para producir una disolución concentrada” (McCabe et al., 1991: p.482).

La evaporación se utiliza con frecuencia en diferentes tipos de industria, un ejemplo claro se evidencia en la industria farmacéutica, en la cual dicha operación unitaria se usa a menudo para crear una solución concentrada, que luego se puede secar para crear un producto en polvo (Hackett, 2018, p.5).

En la industria de procesamiento de alimentos, los productos alimenticios se concentran para aumentar su vida útil, reducir el volumen, el peso, los costos de almacenamiento, los costos de transporte y aumentar el valor, es así como los concentrados de jugo de frutas y la pasta de tomate son dos ejemplos de alimentos de temporada que gracias a la evaporación se conservan para el consumo durante todo el año. La menor actividad del agua es una indicación general de estabilidad microbiológica, esta es una de las razones por la cual se conservan los alimentos concentrados (Hackett, 2018, p.5).

#### 2.2.1.1 Evaporadores de efecto múltiple

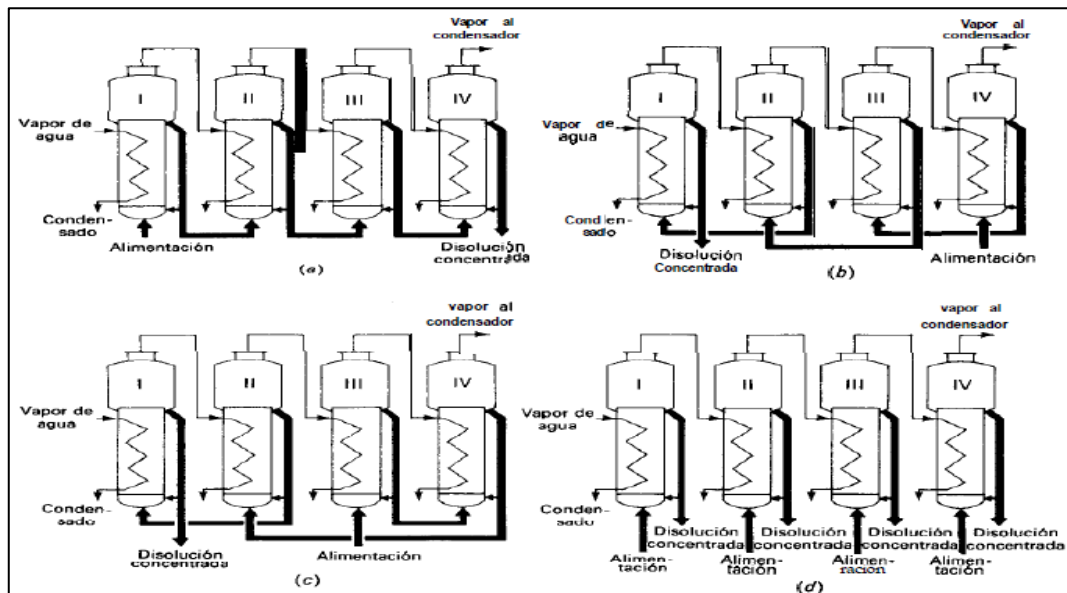
Los evaporadores pueden estar compuestos por uno o más efectos. En un evaporador de efecto múltiple, el vapor de un cuerpo calienta un segundo cuerpo a una temperatura de ebullición más baja. El primer efecto se calienta directamente con vapor, y los cuerpos adicionales se ordenan en función de la temperatura (o presión) de ebullición descendente (Hackett, 2018, p.2).

“El acoplamiento de los cuerpos o efectos del evaporador en un sistema de múltiple efecto es una cuestión de tuberías de interconexión y no de la estructura de las unidades individuales”(McCabe et al., 1991: p.502).

Si el vapor de agua del primer efecto de un evaporador de efecto múltiple se puede introducir en la cámara de vapor de un segundo efecto que opera a un punto de ebullición más bajo, el calor latente en el vapor de agua se puede reutilizar. La reducción de la presión de vapor del segundo efecto en relación con el primer efecto reduce el punto de ebullición del segundo efecto. Esta disposición de reutilizar el calor latente del vapor se denomina evaporación multiefecto o efecto múltiple. La economía de vapor de un evaporador se puede mejorar agregando más efectos. (Hackett, 2018, p.3).

## Métodos de alimentación en evaporadores de múltiple efecto

- Alimentación directa.- “Consiste en introducir mediante una bomba la disolución diluida en el primer efecto y hacerla circular a través de los demás efectos, al igual que en el primer efecto se necesita una bomba para extraer el producto del último efecto” (McCabe et al., 1991: p.506). Generalmente la alimentación directa es mejor que la alimentación a contracorriente cuando la alimentación es fría.
- Alimentación a contracorriente.- “También conocida como alimentación inversa en donde la alimentación diluida a evaporar entra en el último efecto y se bombea a través de los efectos sucesivos hasta obtener el producto concentrado por el primer efecto” (McCabe et al., 1991: p.506). El líquido que se va a concentrar y el vapor vivo utilizado como medio calefactor circulan en sentido contrario. La alimentación a contracorriente se usa cuando la alimentación es viscosa.
- Alimentación mixta. - Es un método en donde circula una parte del sistema en alimentación directa y la otra parte inversa o contracorriente. Este modo de operación elimina ciertas bombas requeridas para la alimentación inversa y permite que la evaporación final se de a una temperatura más alta (McCabe et al., 1991: p.507).
- Alimentación paralela.- “La alimentación se introduce directamente en cada efecto, no hay transporte de líquido entre los efectos, se usa generalmente en evaporadores con cristalización” (McCabe et al., 1991: p.507).



**Figura 1 – 2.** (a) Alimentación directa, (b) alimentación inversa, (c) alimentación mixta, (d) alimentación paralela.

Fuente: (Hackett, 2018, p.3).

### 2.2.2. Simulación dinámica

La simulación dinámica es un instrumento ideal para capacitar a los operadores para operar un proceso integrado y su sistema de control. No solo puede generar los mismos resultados con una escasa manipulación algebraica o sin ésta, sino que también puede producir las respuestas de los sistemas de control en función del tiempo para varias señales de entrada (Smith & Corripio, 2014: p.627).

- Simulación de sistemas lineales a través de funciones de transferencia. - El beneficio que se obtiene de este método es que no se consideran las condiciones iniciales, porque la función de transferencia asume que las variables son desviaciones de las condiciones iniciales del estado estacionario, es decir, la condición inicial es cero. Sin embargo, este método requiere el desarrollo de la función de transferencia, y en el caso de un sistema no lineal, se requiere la linealización de las ecuaciones del modelo.
- Simulación de sistemas de control a partir de modelos mecánicos básicos. - Este método es más fundamental y apropiado modelar sistemas no lineales. “La principal dificultad es que el modelador debe tener cuidado de utilizar las condiciones iniciales correctas para todas las variables” (Smith & Corripio, 2014: p.627). Al simular un sistema de control para un proceso continuo, también hay que asegurarse que las condiciones iniciales representen un estado estacionario.

#### 2.2.2.1 Simulación de procesos

Se puede resolver aspectos básicos de la simulación dinámica aplicados a sistemas continuos mediante la resolución numérica de las ecuaciones diferenciales que representan el modelo, de esta manera se ahorra la necesidad de desarrollar las funciones de transferencia del proceso y linealizar el modelo del proceso si éste es no lineal. De hecho, tanto los modelos de procesos lineales como los no lineales se pueden simular con igual facilidad. Los programas de simulación actuales ofrecen una variedad de métodos muy poderosos para la solución numérica de ecuaciones diferenciales, lo único que se requiere es un modelo en el cual cada ecuación diferencial esté en el formato de primer orden (Smith & Corripio, 2014: p.634):

$$\frac{d(x)t}{dt} = f[(x)t, (u)t, t]$$

Donde:

- $(x)t$  = variable de estado
- $(u)t$  = variable de estado
- $t$  = tiempo
- $f$  = cualquier función lineal o no lineal de las variables

Para facilitar aún más la simulación de procesos dinámicos se usan programas de simulación por bloques los cuales proporcionan un bloque integrador diseñado para integrar numéricamente ecuaciones diferenciales con la forma de la ecuación presentada anteriormente, además de la derivada de entrada, el bloque de integración requiere la condición inicial de la variable de estado estacionario este estado estacionario usualmente se obtiene a partir de las condiciones de diseño del proceso. Sin embargo, debemos comprender que el requerimiento de estado estacionario impone una restricción en los valores que pueden asumir las condiciones de diseño (Smith & Corripio, 2014: p.634).

### **2.2.3. Control automático de procesos**

#### **2.2.3.1 Generalidades de los sistemas de control**

El control automático del proceso es responsable de mantener las variables del proceso (temperatura, presión, flujo, composición, etc.) en los valores operativos requeridos. Cabe recalcar que los procesos generalmente son dinámicos por naturaleza, los cambios ocurren de manera espontánea y de no tomarse las medidas adecuadas para responder, las variables significativas del proceso (variables relacionadas con la seguridad, la productividad y la calidad del producto) no cumplirá con las condiciones requeridas (Smith & Corripio, 2014: p.1).

Para entender los sistemas de control es necesario tener claro algunas definiciones, tales como:

- Variable controlada. - “Cantidad o condición que se va a medir y controlar. Por lo general, la variable controlada es considerada la señal de salida del sistema” (Ogata, 2010, p.3). Se considera que esta variable es la que debe controlar o mantenerse en un valor deseado (Smith & Corripio, 2014: p.4).
- Variable manipulada.- “La señal de control o variable manipulada es la cantidad o condición que el controlador modifica para afectar el valor de la variable controlada” (Ogata, 2010, p.3). En resumen esta variable se utiliza para mantener la variable controlada en su valor de referencia o punto de ajuste/consigna (set point) (Smith & Corripio, 2014: p.4).
- La referencia o set point (SP).- También conocido como punto de consigna prefijado es el valor requerido para la variable controlada (Smith & Corripio, 2014: p.4).
- Controlar.- “Medir el valor de la variable controlada del sistema y aplicar la variable manipulada al sistema para corregir o limitar la desviación del valor medido respecto del valor deseado” (Ogata, 2010, p.3).
- Perturbaciones.- “Son señales que tienden a afectar negativamente el valor de la salida de un sistema” (Ogata, 2010, p.3). Cualquier señal que haga que la variable controlada se aleje del punto de consigna establecido se llama perturbación, siempre existirán interferencias (perturbaciones) en los procesos, si estas no existieran, las condiciones de diseño

predominarían y no habría necesidad de dar seguimiento constantemente al proceso (Smith & Corripio, 2014: p.4).

En adición a las definiciones mencionadas para comprender un sistema de control y que este tenga un funcionamiento adecuado necesariamente debe existir tres componentes básicos, estos son según Smith & Corripio (2014: p.3):

- Sensor-transmisor. - Denominados también como elemento primario y secundario. La señal del transmisor informa al controlador el valor numérico de la variable controlada, es decir se convierte en una señal de referencia la cual es ampliamente conocida como señal de medida.
- Controlador. – al controlador se le conoce como el cerebro del sistema de control, utiliza la señal de medida proveniente del sensor/transmisor para compararla con el valor del punto de consigna prefijado (set point) y así establecer la denominada señal de error o desviación, el controlador procesa su señal de salida o señal de control de acuerdo con la señal de error y la direcciona hacia el elemento final de control para decirle como proceder (Roca, 2014, p.9).
- Elemento final de control.- La señal de salida del controlador es dirigida al elemento final de control que hará las correcciones correspondientes en la variable manipulada que ingresa al proceso de manera que se modifique a la variable controlada, acercándose o igualando al punto de consigna prefijado llamado set point (Roca, 2014, p.8). Por lo general “un elemento final de control es una válvula de control, pero también se usan bombas de velocidad variable, bandas transportadoras, motores y calentadores eléctricos” (Smith & Corripio, 2014: p.3).

Los componentes básicos que se mencionan son importantes ya que ejecutan las operaciones primordiales que deben existir en los sistemas de control y es imperativo que estén en un lazo. Dichas operaciones se conceptualizan según Smith & Corripio, (2014: p.3) como:

- Medición. - “La medición de la variable por controlar se realiza mediante el acoplamiento de un sensor y un transmisor. En ciertos casos, la señal del sensor se puede enviar directamente al controlador sin la necesidad de un transmisor”
- Decisión. - “En función de la medición, el controlador resuelve qué hacer para mantener la variable en el valor deseado”
- Acción. - Acorde a la decisión del controlador, el sistema determina que acciones adoptar. Esta acción suele ser ejecutada por el elemento final de control. Las acciones tomadas deben influir sobre la medición, de lo contrario habrá fallas importantes en el diseño y no se podrá lograr el control, puesto que se considera que existe una condición de lazo abierto

De la misma forma Smith & Corripio, (2014: p.5) manifiesta que se debe tener claro las distintas señales que se utilizan para transmitir información, por lo general son tres:

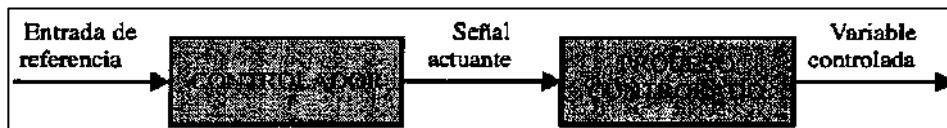
- “La señal neumática, o presión de aire, varía normalmente entre 3 y 15 psig”
- “La señal eléctrica varía regularmente entre 4 mA y 20 mA”
- “La señal digital también denominada discreta (ceros y unos)”

Resulta lógico mencionar que el objetivo de todo sistema de control es ajustar la variable manipulada para que la variable controlada mantenga su valor en el punto de consigna a pesar de las interferencias (perturbaciones).

### 2.2.3.2 Tipos de sistemas de control

Es conveniente mencionar los tipos de sistemas de control más comunes de entre los cuales se detallan los siguientes:

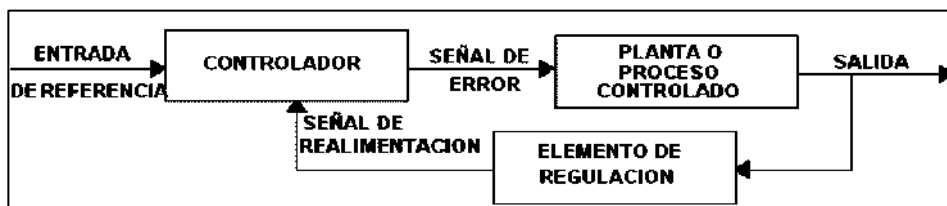
- **Sistemas de control en lazo abierto.**- Son sistemas en los cuales la salida no tiene efecto sobre la acción de control, es decir no se mide la salida ni se realimenta para compararla con la entrada (Ogata, 2010, p.8). Los elementos de un sistema de control en lazo abierto no son más que el controlador y el proceso controlado, al controlador ingresa una señal de referencia cuya salida actúa como una señal de control la cual mantiene el proceso controlado de tal forma que a la salida exista una variable acorde con estándares establecidos (Kuo, 1996, p.9).



**Figura 2 – 2.** Sistema de control en lazo abierto

Fuente: (Kuo, 1996, p.9).

- **Sistemas de control en lazo cerrado.**- En estos sistemas, la señal de salida debe realimentarse y compararse con la señal de referencia para obtener una señal de error de actuación o acción de control proporcional a la diferencia entre la entrada y la salida del sistema, con la finalidad de reducir los errores y llevar la salida del sistema al valor esperado (Ogata, 2010, p.7).



**Figura 3 – 2.** Sistema de control en lazo cerrado

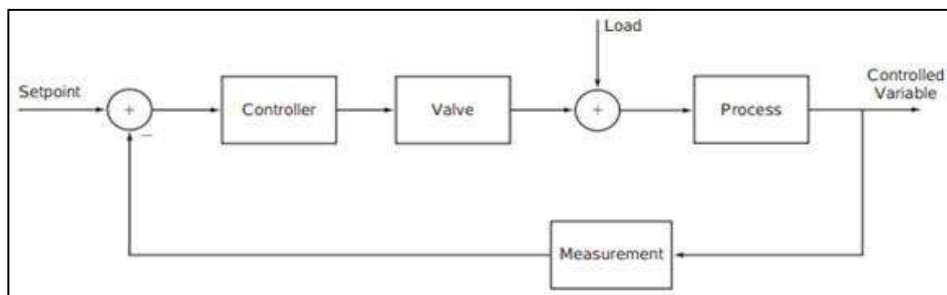
Fuente: (Perez et al., 2008, p.11).

### 2.2.3.3 Estrategias de control

Las estrategias de control más usadas dentro de los sistemas de control son detallan a continuación:

- Control feedback (retroalimentación). - Su funcionamiento toma lugar cuando el controlador nota que la variable de salida es muy distante del valor de referencia, ante esto se envía una señal al elemento final de control para que tome la acción pertinente, no siempre se llega a alcanzar el valor instantáneamente, este valor se encuentra por encima o por debajo del valor de referencia. Esta operación de prueba y error continúa hasta que la variable controlada alcanza y permanece en la referencia.

El lazo de control por retroalimentación no sabe, ni le interesa qué perturbación ha entrado en el proceso, tan sólo trata de mantener la variable controlada en su referencia y al hacerlo compensa todas las perturbaciones (Smith & Corripio, 2014: p.8). El control por realimentación utiliza una estructura de lazo con realimentación negativa para llevar una medición a un valor deseado, o punto de consigna. En la figura 4 - 2 se muestra un diagrama de bloques de un lazo de control de procesos típico, cuyos elementos clave son el controlador, la válvula, el proceso y la medición.



**Figura 4 – 2.** Diagrama de bloques de un lazo de control feedback

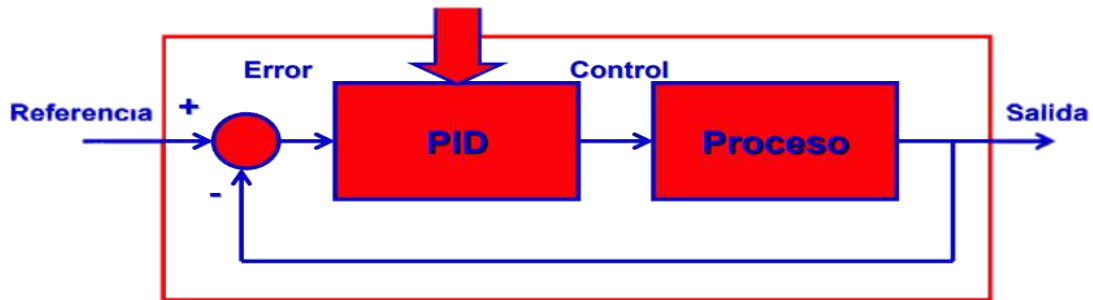
Fuente: (Couper, Penney, Fair, y Walas, 2010, p.31).

- Control feedforward. - También denominado control por acción precalculada, tiene como objetivo medir las perturbaciones y compensarlas antes de que la variable controlada se desvíe de la referencia, dicha desviación podría ser mínima si el control se aplica correctamente. Si cualquier otra perturbación entra al proceso, esta estrategia no compensará su efecto, para evitar esta desviación, es necesario agregar una compensación por retroalimentación al control feedforward (Smith & Corripio, 2014: pp.8-9).

La implementación de una estrategia de control siempre debe tener como premisa principal analizar aspectos relacionados a la economía del proceso, así como también tener en cuenta las distintas variables que se desean controlar con la finalidad de mantener en funcionamiento la estrategia de control que se implemente.

#### 2.2.3.4 Sintonización de controladores

La sintonización de controladores se conceptualiza como la determinación de los parámetros de control de acuerdo con algunas especificaciones también denominados criterios de sintonía. La sintonía es necesaria en dos casos, el primero radica en la puesta en marcha de un sistema de control y el segundo caso es cuando el usuario observa un deterioro del comportamiento del sistema de control (García, 2006, p.4).



**Figura 5 – 2.** Sintonía de controladores

Fuente: (García, 2006, p.3).

Según García, (2006, p.5) existen tres posibilidades cuando se sintonizan los parámetros de control, la primera es el caso más favorable, la segunda permite elegir de un conjunto de juegos de control y la tercera es el caso menos favorable, dichas posibilidades se enlistan a continuación:

- Exista un único juego de parámetros de control tal que se cumplan estas especificaciones
- Existan varios juegos de parámetros de control tal que cumplan estas especificaciones
- No exista un juego de parámetros de control tal que se cumplan estas especificaciones.

#### *Métodos de sintonía de controladores*

- Métodos de prueba y error. - Modificaciones continuas de los parámetros de control hasta alcanzar las especificaciones del sistema en lazo cerrado. Su uso es posible off-line (en simulación), pero no se recomienda usar este método en la planta real (por el coste en tiempo y el coste en la producción) (García, 2006, p.5).
- Métodos basados en experimentos.- Permite estimar ciertas características dinámicas del proceso con un experimento a su vez permite calcular los parámetros del regulador mediante tablas o fórmulas deducidas en función de las características dinámicas estimadas (De la Fuente, 1999, p.76).
- Métodos analíticos basados en modelos (De la Fuente, 1999, p.76).



- Métodos de sintonía automática. - Casi todos los reguladores comerciales utilizan algún tipo de método de ajuste automático. En algunos casos, existe una verdadera función adaptativa (De la Fuente, 1999, p.123).

Respuesta salto

Método del relé

Identificación de la respuesta en lazo cerrado

Control Adaptativo

#### **2.2.4. Software usado**

##### *2.2.4.1 Matlab (MATrix LABORatory)*

Es un software interactivo con muchas herramientas incorporadas para resolver problemas y desarrollar ilustraciones gráficas; el usuario introduce una expresión y el software responde de manera inmediata con un resultado. Es posible escribir scripts que son esencialmente grupos de comandos que se ejecutan secuencialmente. MATLAB se caracteriza por ser un software matemático y gráfico con capacidades numéricas, gráficas y de programación. Al ser una plataforma optimizada incluye un entorno de desarrollo integrado, así como construcciones de programación tanto procedimentales como orientadas a objetos. Tiene funciones incorporadas para realizar muchas operaciones, y hay cajas de herramientas que pueden añadirse para aumentar estas funciones (Attaway, 2019, pp.3-4).

##### *2.2.4.2 Herramienta de MATLAB: Simulink*

Simulink es una extensión gráfica de MATLAB diseñado para modelar y simular sistemas, emplea un lenguaje y un entorno de programación visual: las funciones, ya sean incorporadas o definidas por el usuario, se representan como bloques y los datos que pasan entre las funciones se representan como líneas. Además de la amplia gama de funciones, también se dispone de entradas y salidas. Al estar integrado dentro de MATLAB, los datos pueden transferirse fácilmente entre los programas (Dunn et al., 2006: p.487).

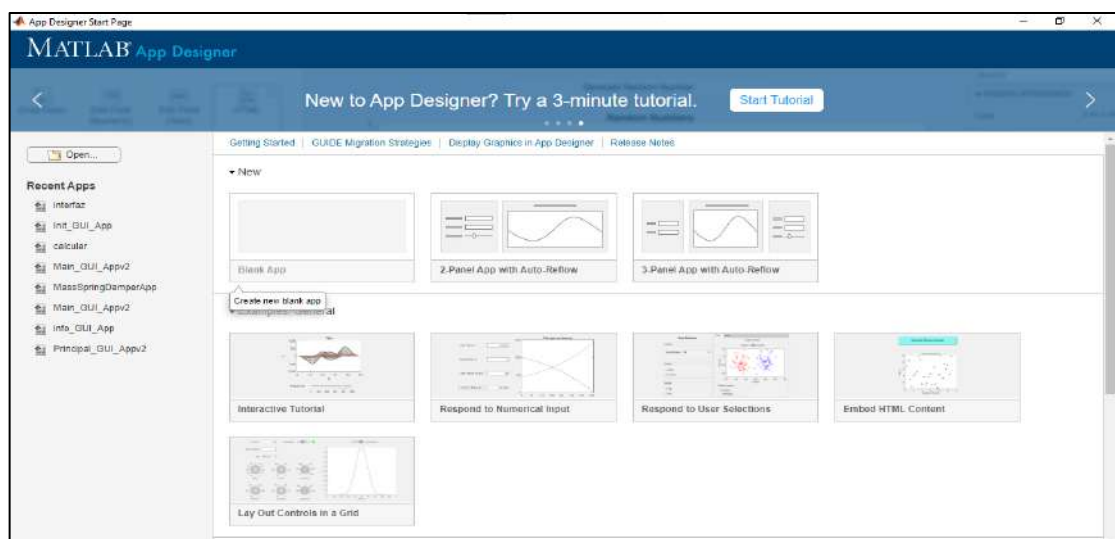
La simulación de sistemas dinámicos ha demostrado ser inmensamente útil a la hora de diseñar controles, ya que ahorra tiempo y dinero que de otro modo se gastaría en la creación de prototipos de un sistema físico. La principal ventaja de Simulink es que es un entorno de programación visual que permite al usuario crear rápidamente un prototipo de simulación y ver inmediatamente los resultados; el flujo de datos entre bloques se especifica mediante líneas que conectan la salida de un bloque con el siguiente. Durante una simulación, los datos se procesan y transfieren sólo en

pasos de tiempo de simulación discretos, la selección de un paso de tiempo viene determinada por la dinámica más rápida del sistema simulado (Dunn et al., 2006: p.487).

### 2.2.4.3 Herramienta de MATLAB: App Designer

App Designer es una herramienta integrada al editor de MATLAB que posee diversos controles interactivos que permiten crear un diseño integral de cualquier aplicación a partir de una interfaz gráfica y programar su comportamiento a partir de la creación de un código, adicional a ello cuenta con opciones que permiten reordenar los componentes dentro de la interfaz de manera que se acople a los cambios de tamaño de pantalla, una más de sus funciones consiste en crear una comunicación con MATLAB Compiler para distribuir las apps como un ejecutable (Mathworks Inc., 2020).

Para poder tener acceso a la herramienta App Designer se ingresa a la ventana de comandos el comando `appdesigner`, al ejecutarlo se tiene acceso a la ventana de inicio el cual da paso al entorno design view y code view.

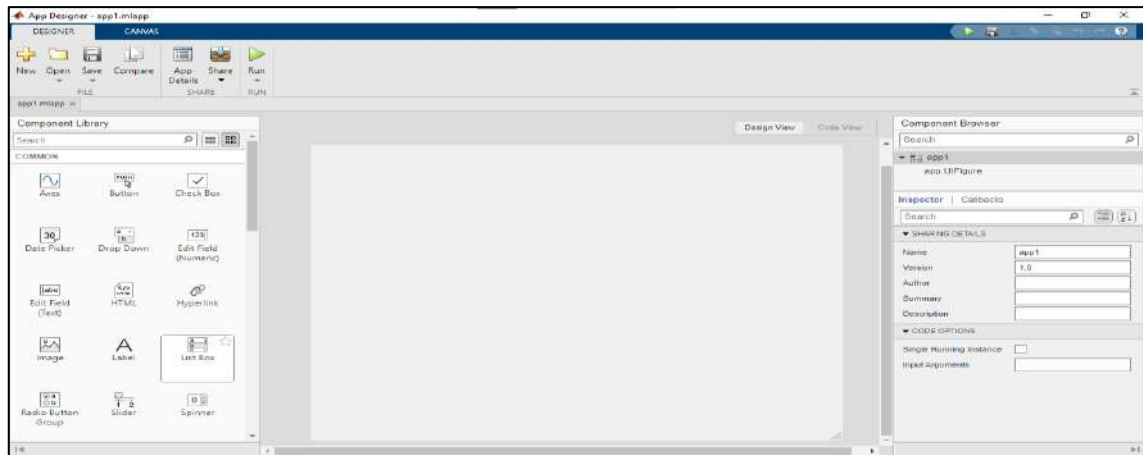


**Figura 6 – 2.** Ventana de inicio de App Designer.

**Elaborado por:** Mena, Jerry. 2021.

### *Entorno Design view*

Dentro de la ventana design view se evidencia dos partes la primera conocida como barra de herramientas en donde se encuentra la parte de designer y canvas, y la parte del entorno de trabajo en donde se encuentran la librería de componentes, el inspector de componentes las propiedades y los callbacks de los componentes (Espinosa, 2018, p.17).



**Figura 7 – 2.** Ventana design view de App Designer.

Elaborado por: Mena, Jerry. 2021.

Dentro de la barra de herramientas de la ventana Design View se pueden evidenciar los siguientes componentes:

**Tabla 1 - 2.** Opciones existentes en la barra de herramientas presentes en App Designer

DESIGN		CANVAS			
<b>New</b>	Crea una nueva aplicación	<b>Save</b>	Guarda la aplicación	<b>Show grid</b>	Muestra la cuadrícula
<b>Open</b>	Abre una aplicación ya creada	<b>Convert</b>	Convierte la app a una app con comportamiento auto reflectante	<b>Interval</b>	Permite modificar el tamaño de la cuadrícula
<b>Save</b>	Guarda la aplicación	<b>Align</b>	Alinea los componentes dentro de la selección	<b>Snap to grid</b>	Ajusta la posición de los elementos a la cuadrícula
<b>App Details</b>	Se puede describir detalles de la app	<b>Same Size</b>	Equipara las dimensiones de los componentes creados	<b>Show alignment hints</b>	Ayuda en la alineación de componentes.
<b>Package App</b>	Compila la aplicación para que sea usada por otros usuarios de MATLAB	<b>Grouping</b>	Agrupar los elementos dentro de la selección	<b>Show resizing hints</b>	Ajuste de dimensiones de los componentes.
<b>Deploy App</b>	Crea una aplicación independiente	<b>Reorder</b>	Reordena los componentes	<b>Zoom</b>	Zoom in, zoom out y restablecer el tamaño
<b>Run</b>	Ejecuta la app creada	<b>Space</b>	Distribuye los elementos dentro de la selección	<b>Run</b>	Ejecuta la app creada


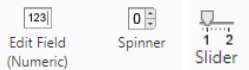
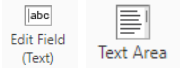



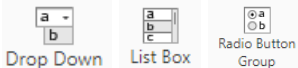






Fuente: (Espinosa, 2018, pp.20-21).


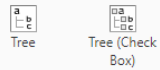

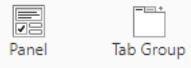
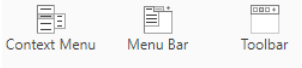


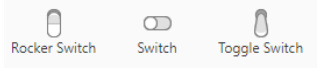

Elaborado por: Mena, Jerry. 2021.

Dentro del entorno de trabajo existen diversas opciones las cuales se detallan a continuación:

- Component Library (Biblioteca de componentes).- Es una lista de los diferentes componentes disponibles para el diseño de la aplicación, para poder utilizarlos se arrastran los componentes a la ventana de design view (Espinosa, 2018, p.18). Estos se encuentran clasificados en 5 partes y se muestran en la tabla 2-2.

**Tabla 2 - 2.** Opciones existentes en la biblioteca de componentes presentes en App Designer

<b>COMMON</b>	<b>Representación de datos</b>	UIAxes	
	<b>Obtención de input numérica</b>	Edit Field (Numeric) Spinner Slider	
	<b>Obtención de input de tipo texto.</b>	Edit Field (Text) Text Area	
	<b>Representación de datos en tablas</b>	Table	
	<b>Ejecución de acciones</b>	Button	
	<b>Selección de estados mutuamente excluyentes</b>	Check box State Button	
	<b>Selección entre dos o más estados</b>	Drop down List box Radiobutton group	
	<b>Seleccionador de hora</b>	Date Picker	
	<b>Incluir un archivo HTML</b>	HTML	
	<b>Abre un archivo externo (página web, código MATLAB)</b>	Hyperlink	
	<b>Agregar una imagen</b>	Image	
	<b>Describir mediante texto</b>	Label	
	<b>Seleccionar opciones de una lista</b>	List box	

	<b>Seleccionar una opción de un listado de botones</b>	Toggle button group	
	<b>Despliega una lista de opciones</b>	Tree Tree(Checkbox)	
<b>CONTAINERS</b>	<b>Organizar componentes</b>	Grid Layout	
	<b>Caracterización de componentes</b>	Panel Tab Group	
<b>FIGURE TOOLS</b>	<b>Introducción de menús desplegables</b>	Context Menu Menu Bar Toolbar	
<b>INSTRUMENTATION</b>	<b>Obtención de input numérica</b>	90 Degree Gauge Linear Gauge Semicircular gauge	
	<b>Señalización del estado visualmente</b>	Lamp	
	<b>Selección de estados mutuamente excluyentes</b>	Rocker switch Switch Toggle Switch	
<b>AEROSPACE</b>	<b>Visualización de medidas en pantalla</b>	Medidores	

Fuente: (Espinosa, 2018, pp.24-28).

Elaborado por: Mena, Jerry. 2021.

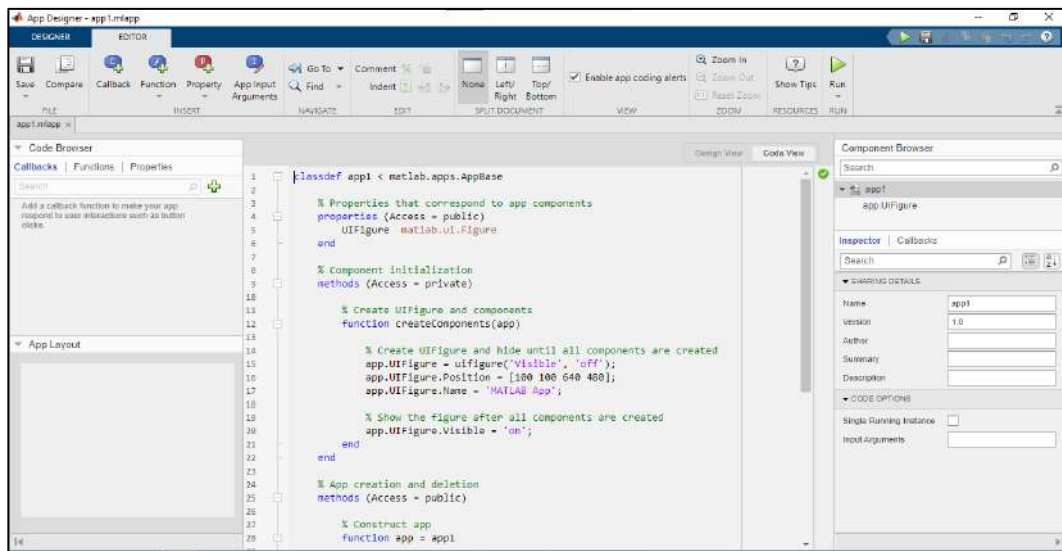
- Component browser.- Es una ventana que contiene las propiedades de los objetos creados en la aplicación, una vez seleccionado el componente, se desplegarán el nombre, características y variables asociadas al objeto creado, el cual podrá ser modificado (Espinosa, 2018, p.23).

## Entorno Code view

Permite ver y administrar el código de la aplicación, comparte las mismas características en la parte del designer de la barra de herramientas y el component browser del entorno de trabajo con el entorno design view (Espinosa, 2018, p.28).

La estructura del código generado consta de las siguientes partes según (Espinosa, 2018, p.28):

- La primera parte del código define las propiedades de la aplicación. Inicialmente, esta área contiene características correspondientes a aplicaciones y componentes
- La segunda parte del código contiene funciones. Si se agregan callbacks y otras funciones aparecen en esta área
- La tercera parte del código inicializa y crea la aplicación y sus componentes



**Figura 8 – 2.** Ventana code view de App Designer.

Elaborado por: Mena, Jerry. 2021.

La barra de herramientas contiene la parte de designer (que como se menciona en el párrafo anterior es idéntica a la que se muestra en el entorno de design view) y la parte de editor que se detallará a continuación:

**Tabla 3 - 2.** Opción de editor existentes en la barra de herramientas presentes en App Designer

EDITOR	
Save	Guarda la aplicación
Callback	Crea un callback
Function	Crea una función de utilidad
Property	Crea una propiedad
App Input Arguments	Permite agregar argumentos de entrada

Go To	Localiza la línea o la función indicada
Find	Encuentra palabras dentro de la parte de code view
Comment	Crea o elimina comentarios
Split Document	Permite tener visualizaciones de code view
Enable app coding alerts	Activa la aparición de warnings y errores que surgen del código
Zoom	Zoom in, zoom out y regresar a la vista predeterminada
Show Tips	Permite visualizar los consejos en la vista de código
Run	Ejecuta la aplicación

**Fuente:** (Borrero, 2011, pp.60-61).

**Elaborado por:** Mena, Jerry. 2021.

Dentro del entorno de trabajo se pueden evidenciar las siguientes partes:

- Code Browser.- Parte de la ventana code view en donde se encuentran los diferentes callbacks, funciones de utilidad y propiedades creadas en la aplicación (Espinosa, 2018, p.21).
- App Layout.- Pestaña que se encuentra bajo la ventana code browser, brinda una vista preliminar de la distribución de los diferentes elementos que conforman la aplicación (Espinosa, 2018, p.22).

#### 2.2.4.4 Herramienta Simulink Compiler

Simulink Compiler es una herramienta integrada de MATLAB que permite compartir simulaciones de Simulink como archivos ejecutables independientes. Un archivo ejecutable independiente puede ser una aplicación de simulación completa utilizando gráficos MATLAB y una interfaz de usuario diseñada por MATLAB App Designer. (Mathworks, 2020).

#### 2.2.4.5 Herramienta Matlab Compiler

MATLAB Compiler le permite crear aplicaciones independientes a partir de programas MATLAB y compartirlas con otros usuarios de forma gratuita. Para crear una aplicación independiente a partir de la simulación de Simulink, utilice el compilador de Simulink. La aplicación independiente puede ser una aplicación con todas las funciones que utiliza gráficos MATLAB y una interfaz de usuario diseñada por MATLAB App Designer o GUIDE, o puede ser un archivo ejecutable de línea de comandos (Mathworks, 2009).

## 2.3. Marco Legal

**Tabla 4 - 2.** Norma legal sobre utilización del software

Normativa	Definición
Código orgánico de la economía social de los conocimientos creatividad e innovación 2016 del software del código cerrado y bases de datos	Artículo 132.- Adaptaciones necesarias para la utilización de software “Sin perjuicio de los derechos morales del autor, el titular de los derechos sobre el software, o el propietario u otro usuario legítimo de un ejemplar del software, podrá realizar las adaptaciones necesarias para la utilización de este, de acuerdo con sus necesidades, siempre que ello no implique su utilización con fines comerciales” (Asamblea Nacional del Ecuador, 2016, p.29).
Constitución de la República del Ecuador Tecnología y producción	<p>Título II. Derechos. Sección Tercera: Comunicación e Información</p> <ul style="list-style-type: none"> <li>• Artículo 16, numeral n°2. “Todas las personas , en forma individual o colectiva, tienen derecho al acceso universal a las tecnologías de información y comunicación” (Asamblea Nacional del Ecuador, 2008, p.25).</li> </ul> <p>Título VI: Régimen de desarrollo. Capítulo VI: Trabajo y producción. Sección Cuarta Democratización de los factores de producción</p> <ul style="list-style-type: none"> <li>• Artículo 334, numeral n°3. “Impulsar y apoyar el desarrollo y la difusión de conocimientos y tecnologías orientados a los procesos de producción” (Asamblea Nacional del Ecuador, 2008, p.156).</li> </ul>

Elaborado por: Mena, Jerry. 2021.

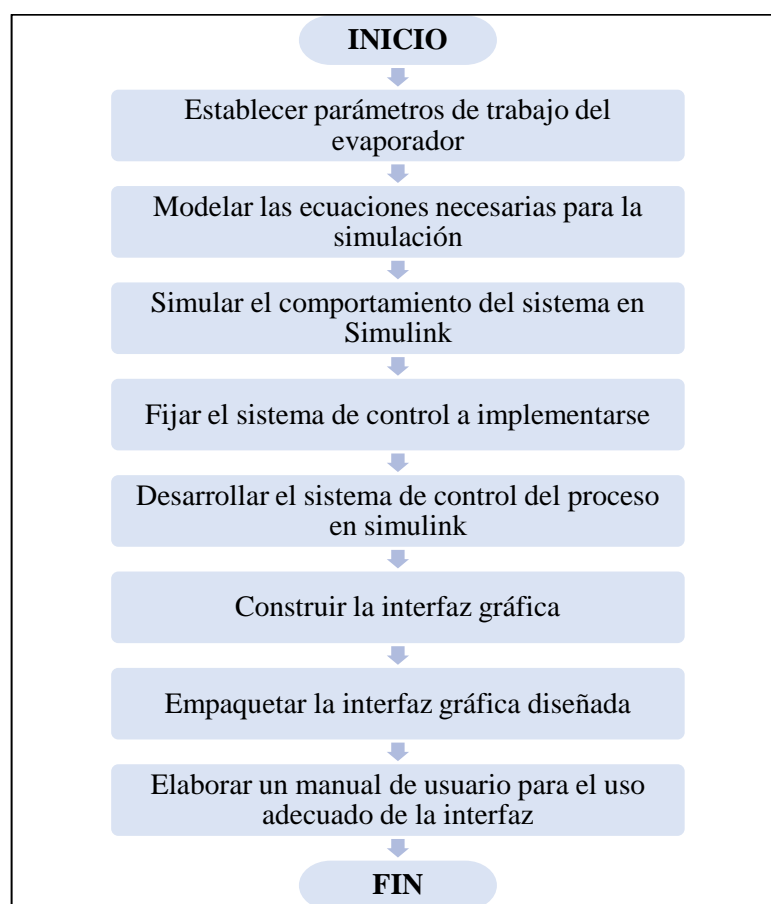


## CAPÍTULO III

### 3. METODOLOGÍA

#### 3.1. Etapas de la metodología

El procedimiento para la construcción de la interfaz gráfica parte de la revisión bibliográfica realizada en el Capítulo II, en el gráfico 1-3 se resume por etapas.



**Gráfico 1 – 3.** Metodología propuesta para el trabajo de titulación

Elaborado por: Mena, Jerry. 2021.

Esta propuesta tecnológica tiene como finalidad desarrollar una interfaz gráfica que integre conocimientos básicos de los controladores, de manera que los usuarios que ejecuten el software tengan una idea clara de cómo un sistema de control influye en el desarrollo correcto del proceso, involucrando variables claves como el set point denominado también punto de consigna o punto de referencia.

### 3.2. Establecer parámetros de trabajo del proceso

Los datos de referencia para establecer los parámetros de trabajo provienen del artículo científico publicado en la revista Chemical Product and Process Modeling titulado como “Simulation and Control of a Commercial Double Effect Evaporator: Tomato Juice” expuesto por Yadav & Jana (2010).

#### 3.2.1. Información del proceso

Se va a trabajar con un evaporador de doble efecto con disposición de alimentación hacia atrás; los dos efectos están numerados de izquierda a derecha como Tanque 1 y Tanque 2 respectivamente. La alimentación con caudal  $F$ , composición  $X_f$  y temperatura  $T_f$  entra al Tanque 2, y el vapor con caudal  $S$  y temperatura  $T_s$  entra en el Tanque 1. Las retenciones de masa (acumulación) en los dos tanques se definen como  $M_1$  y  $M_2$ .  $V_1$  y  $V_2$  son las tasas de flujo de vapor de la parte superior de dos tanques con temperatura  $T_1$  y  $T_2$ , respectivamente.  $P_1$  y  $P_2$  son las tasas de flujo de producto de los dos efectos con composición  $X_p$  y  $X_2$ , y temperatura  $T_1$  y  $T_2$ , respectivamente (Jana, 2011, p.323).

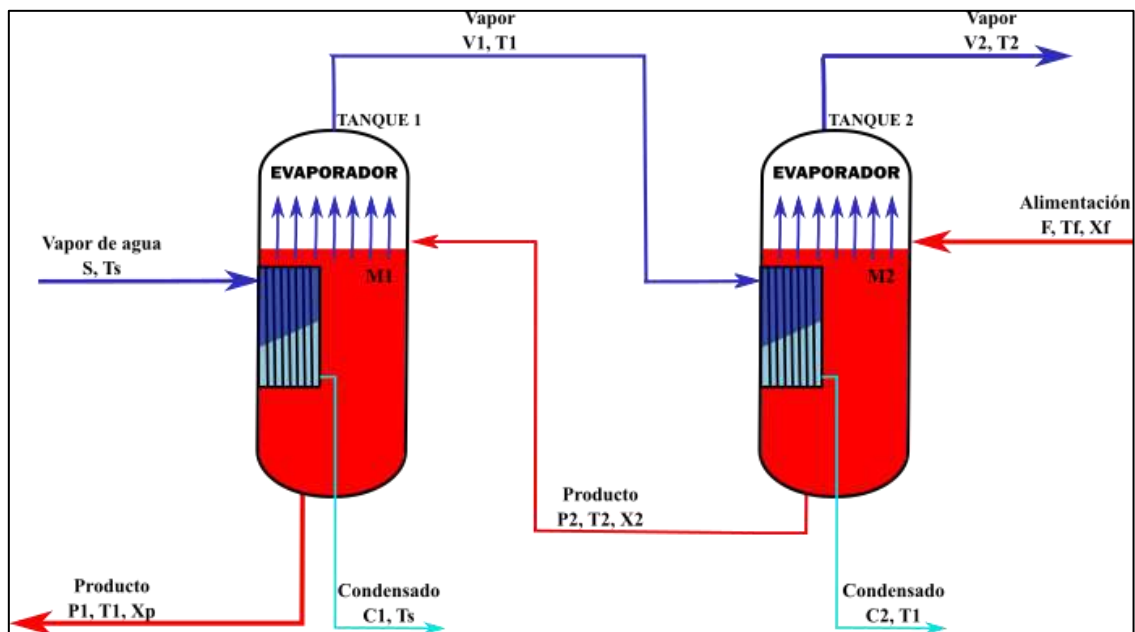


Figura 1 – 3. Diagrama del proceso que va a implementarse

Elaborado por: Mena, Jerry. 2021.

### 3.2.2. Valores numéricos para usar dentro del proceso

Por lo general, es imprescindible contar con los valores de los parámetros del sistema físico y la solución del modelo del evaporador en estado estacionario para la simulación de modelo dinámico y las configuraciones de control. Los parámetros físicos del evaporador utilizados se encuentran en la Tabla 1 - 3; estos valores se han tomado de un estudio anterior que trabajan con un evaporador de pasta de tomate a escala comercial ( Runyon et al., 1991: p.194; citados en Yadav & Jana (2010)).

**Tabla 1 - 3.** Valores de estado estacionario y de los parámetros físicos

Termino	Abreviación (unidad)	Valor	Valor en segundos
Acumulación masa Tanque 1	$M_1$ (kg)	2268	2268
Acumulación masa Tanque 2	$M_2$ (kg)	2268	2268
El flujo de alimentación de entrada	$F_0$ (kg/h)	26103	7,25
El flujo de vapor de entrada	$S_0$ (kg/h)	11023	3,06
Tasa de flujo del producto líquido del tanque 1	$P_1$ (kg/h)	5006	1,39
Tasa de flujo del producto líquido del tanque 2	$P_2$ (kg/h)	14887	4,14
Flujo de vapor del tanque 1	$V_1$ (kg/h)	9932	2,76
Flujo de vapor del tanque 2	$V_2$ (kg/h)	11165	3,10
Composición del alimento	$X_f$ (kg/kg)	0.05	0.05
Composición del tanque 1	$X_p$ (kg/kg)	0.2607	0.2607
Composición del tanque 2	$X_2$ (kg/kg)	0.0874	0.0874
La temperatura del vapor	$T_s$ (°C)	115.7	115.7
Temperatura de alimentación	$T_f$ (°C)	85	85
Temperatura en el tanque 1	$T_1$ (°C)	74.7	74.7
Temperatura en el tanque 2	$T_2$ (°C)	52	52
Área de transferencia de calor en el Tanque 1	$A_1$ (m <sup>2</sup> )	102	102
Área de transferencia de calor en el Tanque 2	$A_2$ (m <sup>2</sup> )	412	412
Coefficiente global de transferencia de calor para el Tanque 1	$U_1$ (kJ/h.m <sup>2</sup> .°C)	5826	1,62
Coefficiente global de transferencia de calor para el Tanque 2	$U_2$ (kJ/h.m <sup>2</sup> .°C)	2453	0,68

**Fuente:** (Runyon et al., 1991: p.194; citados en Yadav & Jana (2010)).

**Elaborado por:** Mena, Jerry. 2021

Teniendo en cuenta que el producto que se requiere evaporar es jugo de tomate y se considera como una solución binaria de agua y sólidos solubles inertes en sentido químico (Miranda & Simpson, 2005: p.205), se establece correlaciones empíricas las cuales permiten tener mayor precisión en los datos experimentales de una mezcla concreta, dadas las condiciones temperatura y presión (Mulero et al., 2004: pp.95-104), a continuación se detallan dichas correlaciones:

- La entalpía del producto (jugo de tomate) se representa como (Heldman & Singh, 1981; citados en Runyon et al., 1991: p.189):

$$h(T, X) = (4.177 - 2.506X)T$$

- La entalpía del vapor es el disolvente puro y se calcula utilizando una ecuación de regresión polinómica de valores de las tablas de vapor de (Runyon et al., 1991: p.189):

$$H(T) = (2495 + 1.958T - 0.002128T^2)T$$

- Para las corrientes de condensado, la entalpía líquida de disolvente puro también se obtiene de las tablas de vapor de (Runyon et al., 1991: p.189):

$$h(T) = 4.177T$$

- Así, el calor latente de la vaporización tiene la siguiente forma (Runyon et al., 1991: p.189):

$$\lambda(T) = H(T) - h(T) = 2495 + 1.958T - 0.002128T^2$$

### 3.3. Modelar las ecuaciones necesarias para la simulación

El primer paso para un diseño correcto de un sistema de control es alcanzar una comprensión profunda de las relaciones existentes entre las variables del sistema y la tecnología de proceso a controlar y a partir de los balances de masa y energía se obtienen las ecuaciones necesarias para realizar la simulación dinámica.

#### 3.3.1. Balance de masa

Dentro de este inciso se presentan los balances de masa correspondientes a todo el proceso, tanto el balance global como el balance de componentes para el primer y segundo efecto.

##### 3.3.1.1 Balance de masa en el primer efecto:

$$E = S + A$$

$$A = E - S$$

Ec. (1)

$$\frac{dM_1}{dt} = P_2 - P_1 - V_1$$

Donde:

E: Entrada

S: Salida

A: Acumulación

3.3.1.2 Balance de componentes (soluto) en el primer efecto:

Ec. (2)

$$\frac{d(M_1 X_P)}{dt} = P_2 X_2 - P_1 X_P$$

Ec. (3)

$$M_1 \frac{dX_P}{dt} + X_P \frac{dM_1}{dt} = P_2 X_2 - P_1 X_P$$

Ec. (4)

$$\frac{dX_P}{dt} = \frac{1}{M_1} \left[ P_2 X_2 - P_1 X_P - X_P \frac{dM_1}{dt} \right]$$

Ec. (1) en Ec. (4)

Ec. (5)

$$\frac{dX_P}{dt} = \frac{1}{M_1} [P_2 X_2 - P_1 X_P - X_P (P_2 - P_1 - V_1)]$$

Ec. (6)

$$\frac{dX_P}{dt} = \frac{1}{M_1} (P_2 X_2 - P_1 X_P - X_P P_2 + X_P P_1 + X_P V_1)$$

Ec. (7)

$$\frac{dX_P}{dt} = \frac{P_2 (X_2 - X_P) + X_P V_1}{M_1}$$

3.3.1.3 Balance de masa en el segundo efecto:

$$E = S + A$$

$$A = E - S$$

Ec. (8)

$$\frac{dM_2}{dt} = F - P_2 - V_2$$

3.3.1.4 Balance de componentes (soluto) en el efecto:

Ec. (9)

$$\frac{d(M_2 X_2)}{dt} = F X_f - P_2 X_2$$

Ec. (10)

$$M_2 \frac{dX_2}{dt} + X_2 \frac{dM_2}{dt} = F X_f - P_2 X_2$$

Ec. (11)

$$\frac{dX_2}{dt} = \frac{1}{M_2} \left[ FX_f - P_2 X_2 - X_2 \frac{dM_2}{dt} \right]$$

Ec. (8) en Ec. (11)

Ec. (12)

$$M_2 \frac{dX_2}{dt} = \frac{1}{M_2} [FX_f - P_2 X_2 - X_2 (F - P_2 - V_2)]$$

Ec. (13)

$$M_2 \frac{dX_2}{dt} = \frac{1}{M_2} (FX_f - P_2 X_2 - X_2 F + X_2 P_2 + X_2 V_2)$$

Ec. (14)

$$\frac{dX_2}{dt} = \frac{F(X_f - X_2) + X_2 V_2}{M_2}$$

### 3.3.2. Balance de energía

Asumiendo que el calor generado es causado por la condensación total de la corriente de vapor, se tiene:

Ec. (15)

$$Q = \lambda S$$

Para el balance de energía se utilizan las variables asignadas a continuación:

- h = Entalpía en función de T1 y XP
- h2 = Entalpía en función de T2 y X2
- HS = Entalpía del vapor en función de TS
- hc1 = Entalpía del condensado
- H1 = Entalpía del vapor en función de T1
- hc1 = hS
- $\lambda$  = Calor latente de vaporización (HS - hS)
- C1 = S Taza de vapor primer efecto
- hf = Entalpía en función de Tf y Xf
- $\lambda 2$  = Calor latente de vaporización (H1 - hc2)
- C2 = V1 Taza de vapor segundo efecto

#### 3.3.2.1 Balance de energía en el primer efecto:

$$E = S + A$$

$$A = E - S$$

Ec. (16)

$$\frac{d(M_1 h)}{dt} = P_2 h_2 + S H_S - P_1 h - C_1 h_{c1} - V_1 H_1$$

Ec. (17)

$$M_1 \frac{dh}{dt} + h \frac{dM_1}{dt} = P_2 h_2 + S \lambda - P_1 h - V_1 H_1$$

Ec. (18)

$$M_1 \frac{dh}{dt} = P_2 h_2 + S \lambda - P_1 h - V_1 H_1 - h \frac{dM_1}{dt}$$

Ec. (19)

$$\frac{dh}{dt} = \frac{1}{M_1} \left( P_2 h_2 + S \lambda - P_1 h - V_1 H_1 - h \frac{dM_1}{dt} \right)$$

$$h(T_1, X_p) = (4.177 - 2.506 X_p) T_1$$

expresando constantes a=4.177 y b=2.506 se tiene

$$h(T, X) = (a - b X_p) T_1$$

Ec. (20)

$$\frac{d(a - b X_p) T_1}{dt} = \frac{1}{M_1} \left( P_2 h_2 + S \lambda - P_1 h - V_1 H_1 - h \frac{dM_1}{dt} \right)$$

Ec. (21)

$$\frac{d(a) T_1}{dt} - \frac{d(b X_p) T_1}{dt} = \frac{1}{M_1} \left( P_2 h_2 + S \lambda - P_1 h - V_1 H_1 - h \frac{dM_1}{dt} \right)$$

Ec. (22)

$$a \frac{dT_1}{dt} - b \frac{dX_p T_1}{dt} = \frac{1}{M_1} \left( P_2 h_2 + S \lambda - P_1 h - V_1 H_1 - h \frac{dM_1}{dt} \right)$$

Ec. (23)

$$a \frac{dT_1}{dt} - b \left( X_p \frac{dT_1}{dt} + T_1 \frac{dX_p}{dt} \right) = \frac{1}{M_1} \left( P_2 h_2 + S \lambda - P_1 h - V_1 H_1 - h \frac{dM_1}{dt} \right)$$

Ec. (24)

$$a \frac{dT_1}{dt} - b X_p \frac{dT_1}{dt} + b T_1 \frac{dX_p}{dt} = \frac{1}{M_1} \left( P_2 h_2 + S \lambda - P_1 h - V_1 H_1 - h \frac{dM_1}{dt} \right)$$

Ec. (25)

$$(a - b X_p) \frac{dT_1}{dt} + b T_1 \frac{dX_p}{dt} = \frac{1}{M_1} \left( P_2 h_2 + S \lambda - P_1 h - V_1 H_1 - h \frac{dM_1}{dt} \right)$$

Ec. (26)

$$(a - b X_p) \frac{dT_1}{dt} = \left[ \frac{1}{M_1} \left( P_2 h_2 + S \lambda - P_1 h - V_1 H_1 - h \frac{dM_1}{dt} \right) \right] - b T_1 \frac{dX_p}{dt}$$

Ec. (27)

$$\frac{dT_1}{dt} = \frac{1}{(a - b X_p)} \left\{ \left[ \frac{1}{M_1} \left( P_2 h_2 + S \lambda - P_1 h - V_1 H_1 - h \frac{dM_1}{dt} \right) \right] - b T_1 \frac{dX_p}{dt} \right\}$$

Siguiendo las asunciones involucradas en la formulación del modelo se puede prescindir del término  $bT_1 \frac{dX}{dt}$ , el cual se cataloga como EPE (Elevación Punto Ebullición).

Ec. (28)

$$\frac{dT_1}{dt} = \frac{1}{(a - bX_p)} \left[ \frac{1}{M_1} \left( P_2 h_2 + S\lambda - P_1 h - V_1 H_1 - h \frac{dM_1}{dt} \right) \right]$$

Para facilitar el ingreso de las entalpías dentro de Matlab se detallan en función de que variables se obtienen.

Ec. (29)

$$\frac{dT_1}{dt} = \frac{1}{(a - bX_p)} \left[ \frac{1}{M_1} \left( P_2 h_{2(T_2, X_2)} + S\lambda_{(T_3)} - P_1 h_{(T_1, X_p)} - V_1 H_{1(T_1)} - h_{(T_1, X_p)} \frac{dM_1}{dt} \right) \right]$$

Ec. (30)

$$\frac{dT_1}{dt} = \frac{1}{(4.177 - 2.506X_p)} \left\{ \left[ \frac{1}{M_1} \left( P_2 h_{2(T_2, X_2)} + S\lambda_{(T_3)} - P_1 h_{(T_1, X_p)} - V_1 H_{1(T_1)} - h_{(T_1, X_p)} \frac{dM_1}{dt} \right) \right] + bT_1 \frac{dX_p}{dt} \right\}$$

3.3.2.2 *Balance de energía en el segundo efecto:*

$$E = S + A$$

$$A = E - S$$

Ec. (31)

$$\frac{d(M_2 h_2)}{dt} = F h_f + V_1 H_1 - P_2 h_2 - C_2 h_{c2} - V_2 H_2$$

Ec. (32)

$$M_2 \frac{dh_2}{dt} + h_2 \frac{dM_2}{dt} = F h_f + V_1 \lambda_2 - P_2 h_2 - V_2 H_2$$

Ec. (33)

$$M_2 \frac{dh_2}{dt} = F h_f + V_1 \lambda_2 - P_2 h_2 - V_2 H_2 - h_2 \frac{dM_2}{dt}$$

Ec. (34)

$$\frac{dh_2}{dt} = \frac{1}{M_2} \left( F h_f + V_1 \lambda_2 - P_2 h_2 - V_2 H_2 - h_2 \frac{dM_2}{dt} \right)$$

$$h_2(T_2, X_2) = (4.177 - 2.506X_2)T_2$$

expresando constantes como a=4.177 y b=2.506 se tiene

$$h(T, X) = (a - bX_2)T_2$$

Ec. (35)

$$\frac{d(a - bX_2)T_2}{dt} = \frac{1}{M_2} \left( F h_f + V_1 \lambda_2 - P_2 h_2 - V_2 H_2 - h_2 \frac{dM_2}{dt} \right)$$

Ec. (36)



$$\frac{d(a)T_2}{dt} - \frac{d(bX_2)T_2}{dt} = \frac{1}{M_2} \left( Fh_f + V_1\lambda_2 - P_2h_2 - V_2H_2 - h_2 \frac{dM_2}{dt} \right)$$

Ec. (37)

$$a \frac{dT_2}{dt} - b \frac{dX_2T_2}{dt} = \frac{1}{M_2} \left( Fh_f + V_1\lambda_2 - P_2h_2 - V_2H_2 - h_2 \frac{dM_2}{dt} \right)$$

Ec. (38)

$$a \frac{dT_2}{dt} - b \left( X_2 \frac{dT_2}{dt} + T_2 \frac{dX_2}{dt} \right) = \frac{1}{M_2} \left( Fh_f + V_1\lambda_2 - P_2h_2 - V_2H_2 - h_2 \frac{dM_2}{dt} \right)$$

Ec. (39)

$$a \frac{dT_2}{dt} - bX_2 \frac{dT_2}{dt} + bT_2 \frac{dX_2}{dt} = \frac{1}{M_2} \left( Fh_f + V_1\lambda_2 - P_2h_2 - V_2H_2 - h_2 \frac{dM_2}{dt} \right)$$

Ec. (40)

$$(a - bX_2) \frac{dT_2}{dt} + bT_2 \frac{dX_2}{dt} = \frac{1}{M_2} \left( Fh_f + V_1\lambda_2 - P_2h_2 - V_2H_2 - h_2 \frac{dM_2}{dt} \right)$$

Ec. (41)

$$(a - bX_2) \frac{dT_2}{dt} = \left[ \frac{1}{M_2} \left( Fh_f + V_1\lambda_2 - P_2h_2 - V_2H_2 - h_2 \frac{dM_2}{dt} \right) \right] - bT_2 \frac{dX_2}{dt}$$

Ec. (42)

$$\frac{dT_2}{dt} = \frac{1}{(a - bX_2)} \left\{ \left[ \frac{1}{M_2} \left( Fh_f + V_1\lambda_2 - P_2h_2 - V_2H_2 - h_2 \frac{dM_2}{dt} \right) \right] - bT_2 \frac{dX_2}{dt} \right\}$$

Al igual que en la sección 3.3.2.1 se prescinde del término  $bT_1 \frac{dX}{dt}$ , el cual se denomina EPE.

Ec. (43)

$$\frac{dT_2}{dt} = \frac{1}{(a - bX_2)} \left[ \frac{1}{M_2} \left( Fh_f + V_1\lambda_2 - P_2h_2 - V_2H_2 - h_2 \frac{dM_2}{dt} \right) \right]$$

Para facilitar el ingreso de las entalpías dentro de Matlab se detallan en función de que variables se obtienen.

Ec. (44)

$$\frac{dT_2}{dt} = \frac{1}{(a - bX_2)} \left[ \frac{1}{M_2} \left( Fh_{f(T_f, X_f)} + V_1\lambda_{2(T_1)} - P_2h_{2(T_2, X_2)} - V_2H_{2(T_2)} - h_{2(T_2, X_2)} \frac{dM_2}{dt} \right) \right]$$

Ec. (45)

$$\frac{dT_2}{dt} = \frac{1}{(4.177 - 2.506X_2)} \left\{ \left[ \frac{1}{M_2} \left( Fh_{f(T_f, X_f)} + V_1\lambda_{2(T_1)} - P_2h_{2(T_2, X_2)} - V_2H_{2(T_2)} - h_{2(T_2, X_2)} \frac{dM_2}{dt} \right) \right] + bT_2 \frac{dX_2}{dt} \right\}$$

3.3.2.3 La tasa de vapor para el primer efecto obedece a:

Para encontrar la tasa de vapor correspondiente al primer efecto se necesita colocar la ecuación del balance de energía en estado estacionario y resolverla.

Ec. (46)

$$\frac{dT_1}{dt} = \frac{1}{(4.177 - 2.506X_p)} \left\{ \left[ \frac{1}{M_1} \left( P_2 h_{2(T_2, X_2)} + S \lambda_{(T_s)} - P_1 h_{(T_1, X_p)} - V_1 H_{1(T_1)} - h_{(T_1, X_p)} \frac{dM_1}{dt} \right) \right] + b T_1 \frac{dX}{dt} \right\}$$

Ec. (47)

$$0 = \frac{1}{(4.177 - 2.506X_p)} \left[ \frac{1}{M_1} \left( P_2 h_{2(T_2, X_2)} + S \lambda_{(T_s)} - P_1 h_{(T_1, X_p)} - V_1 H_{1(T_1)} \right) \right]$$

Despejando  $V_1 H_{1(T_1)}$  se tiene:

Ec. (48)

$$V_1 H_{1(T_1)} = P_2 h_{2(T_2, X_2)} + S \lambda_{(T_s)} - P_1 h_{(T_1, X_p)}$$

Cómo  $V_1 H_{1(T_1)} = Q_{v1}$ , se puede despejar  $V_1$  de forma que:

Ec. (49)

$$V_1 = \frac{Q_{v1}}{H_{1(T_1)}}$$

3.3.2.4 La tasa de vapor para el segundo efecto obedece a:

Siguiendo el mismo proceso de resolución de la tasa de vapor del primer efecto se tiene:

Ec. (50)

$$\frac{dT_2}{dt} = \frac{1}{(4.177 - 2.506X_2)} \left\{ \left[ \frac{1}{M_2} \left( F h_{f(T_f, X_f)} + V_1 \lambda_{2(T_1)} - P_2 h_{2(T_2, X_2)} - V_2 H_{2(T_2)} - h_{2(T_2, X_2)} \frac{dM_2}{dt} \right) \right] + b T_2 \frac{dX}{dt} \right\}$$

Ec. (51)

$$0 = \frac{1}{(4.177 - 2.506X_2)} \left[ \frac{1}{M_2} \left( F h_{f(T_f, X_f)} + V_1 \lambda_{2(T_1)} - P_2 h_{2(T_2, X_2)} - V_2 H_{2(T_2)} \right) \right]$$

Despejando  $V_2 H_{2(T_2)}$  se tiene:

Ec. (52)

$$V_2 H_{2(T_2)} = F h_{f(T_f, X_f)} + V_1 \lambda_{2(T_1)} - P_2 h_{2(T_2, X_2)}$$

Cómo  $V_2 H_{2(T_2)} = Q_{v2}$ , se puede despejar  $V_2$  de forma que:

Ec. (53)

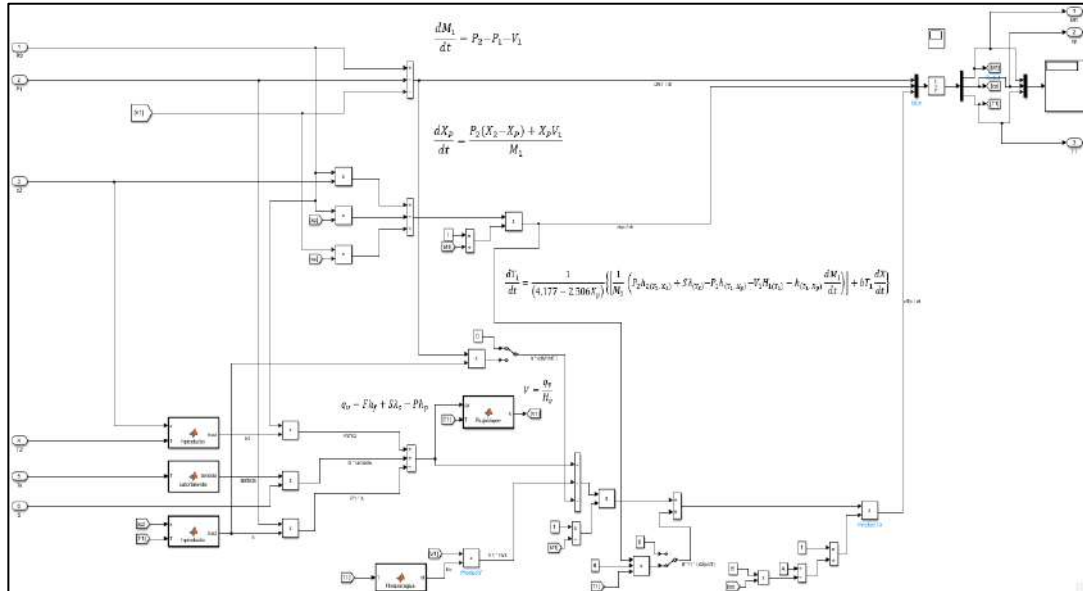
$$V_2 = \frac{Q_{v2}}{H_{2(T_2)}}$$

### 3.4. Simular el comportamiento del sistema dinámico en Simulink

#### 3.4.1. Pasos para seguir en simulink

Luego de haber modelado las ecuaciones diferenciales, es posible realizar la simulación en Simulink, se siguen los siguientes pasos:

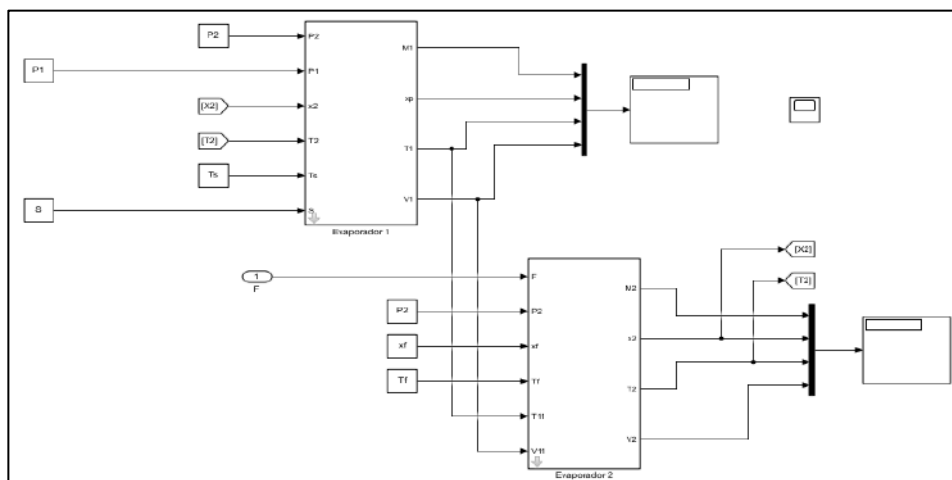
- Ingresar las ecuaciones procedentes de la sección 3.3 para cada evaporador por separado, así los efectos usan un solo integrador, dentro de estos se ingresa los valores iniciales provenientes del estado estacionario los cuales se extraen de bibliografía.



**Figura 2 – 3.** Formato del ingreso de la ecuación en Simulink

Elaborado por: Mena, Jerry. 2021.

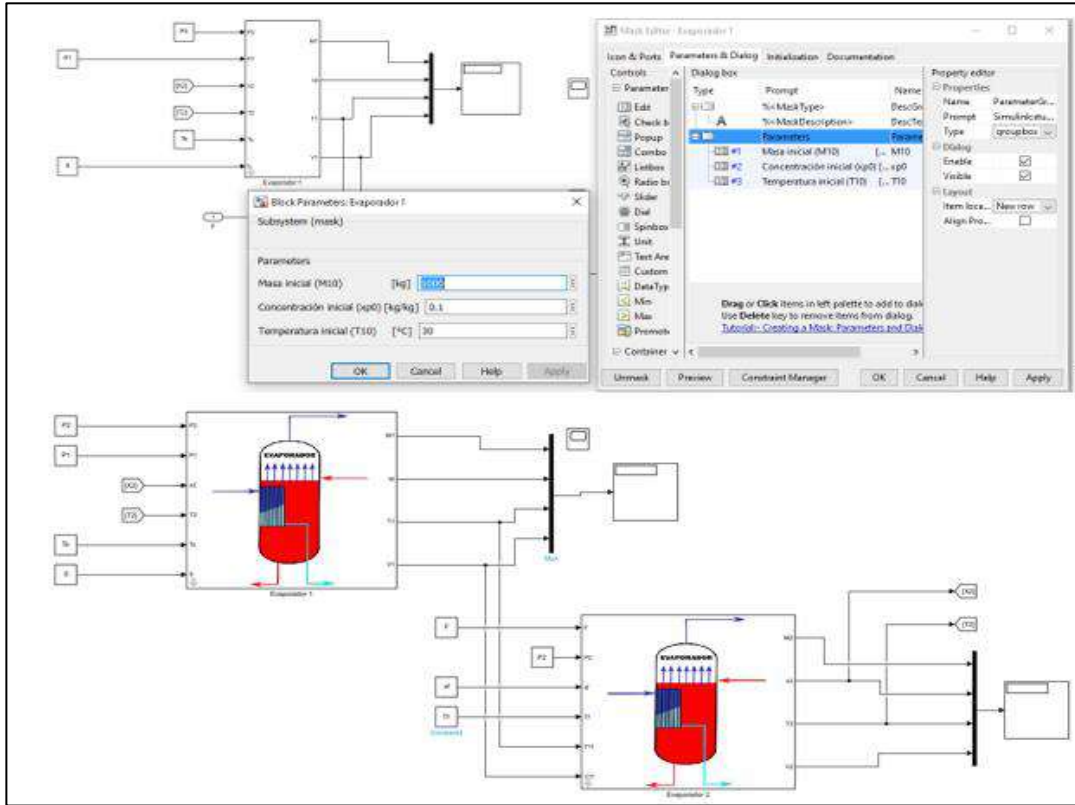
- Una vez ingresadas las ecuaciones de los dos evaporadores se las coloca dentro subsistemas para mantener un orden y así se facilite la forma de interconexión de un evaporador con el otro, como el proceso es en contracorriente la entrada del flujo de vapor para el segundo efecto será la salida de vapor proveniente del primero, lo mismo sucede con el flujo de alimentación.



**Figura 3 – 3.** Subsistemas creados dentro del entorno Simulink

Elaborado por: Mena, Jerry. 2021.

- Para la entrada de datos se crea una máscara en cada subsistema esto permite modificar los parámetros de entrada, mientras que los valores dentro del workspace no cambian, adicional a ello usando un software libre para diagramación de objetos denominado Inkscape fue posible añadir una imagen a las máscaras de subsistema creadas.



**Figura 4 – 3.** Máscaras de subsistema para los procesos creados dentro del entorno Simulink

Elaborado por: Mena, Jerry. 2021.

- Dentro del workspace se definen las variables y en el espacio de Matlab se crea un script en el cual se puedan ingresar las variables, a su vez se introducen las carpetas que se va a utilizar dentro del path de Matlab que permitirá añadir imágenes en las máscaras de subsistema.

```

Editor - C:\Users\Mena\Desktop\ModeloCompleto\main.m*
main.m*  main.m  +
1  clear; clc;
2  addpath('func/');
3  addpath('src/');
4  addpath('test/');
5  addpath('ui/');
6  addpath('imagenes/');
7
8  A = 4.187;      % Valores provenientes de la entalpia
9  B = 2.9309;   % Valores provenientes de la entalpia
10
11  F = 7.05;     % Va...
Command Window

```

**Figura 5 – 3.** Script main de Matlab para definir las variables

Elaborado por: Mena, Jerry. 2021.

### 3.5. Diseñar el sistema de control a implementarse

#### 3.5.1. *Objetivos de control*

Dentro del sistema en cuestión es importante establecer ciertos objetivos para que el proceso fluya de manera correcta, estos son:

- Mantener la concentración de sólidos del producto de salida en su valor deseado.
- Controlar la retención de masa líquida (acumulación) para evitar el desbordamiento o la desecación de los tubos del evaporador.

Para cumplir con los objetivos de control, la concentración final del producto ( $X_p$ ) y la retención de masa en ambos efectos (M1 y M2) son variables a las cuales se aplica un control, mientras que el caudal de vapor (S), la corriente de producto del primer y segundo efecto (P1 y P2), son variables manipuladas las cuales se seleccionan identificando las que tienen un impacto directo en la variable controlada y a su vez un impacto mínimo en el resto del sistema. Se considera como elemento final de control a una válvula de control.

**Tabla 2 - 3.** Pares de control seleccionados dentro del proceso

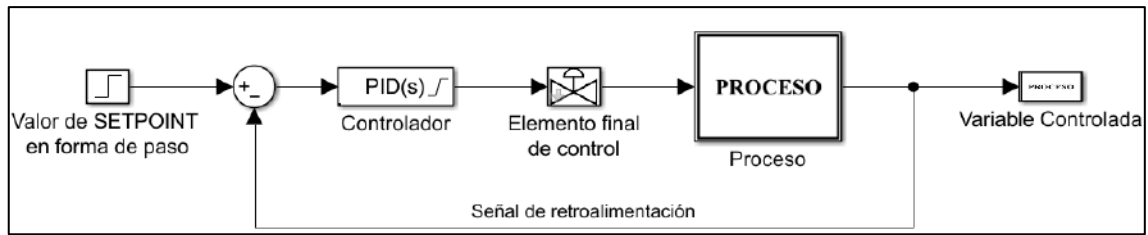
Variable controlada	Variable manipulada
$X_p$	S
M1	P1
M2	P2

Elaborado por: Mena, Jerry. 2021.

#### 3.5.2. *Esquema del sistema de control a implementarse*

Acorde con el inciso 2.2.3 de la sección 2 es posible esquematizar el sistema de control que va a implementarse el cual consta de un valor de referencia en la entrada denominado set point, junto a él existe un símbolo de sumatoria que genera un valor de diferencia entre el valor de set point ingresado y la señal de retroalimentación proveniente de la salida del proceso.

Esta señal resultante alimenta al controlador (que no necesariamente es un PID) y la señal de salida del controlador va directo al elemento final de control en este caso será una válvula (la cual estrangula el flujo), esta señal regula el caudal que va a pasar por la válvula hacia el proceso, finalmente a la salida del proceso se obtiene una señal conocida como variable controlada la cual arroja valores exactos al valor del set point o en los casos menos favorables un valor muy cercano al valor establecido.



**Figura 6 – 3.** Esquema del Sistema de control a implementarse

Elaborado por: Mena, Jerry. 2021.

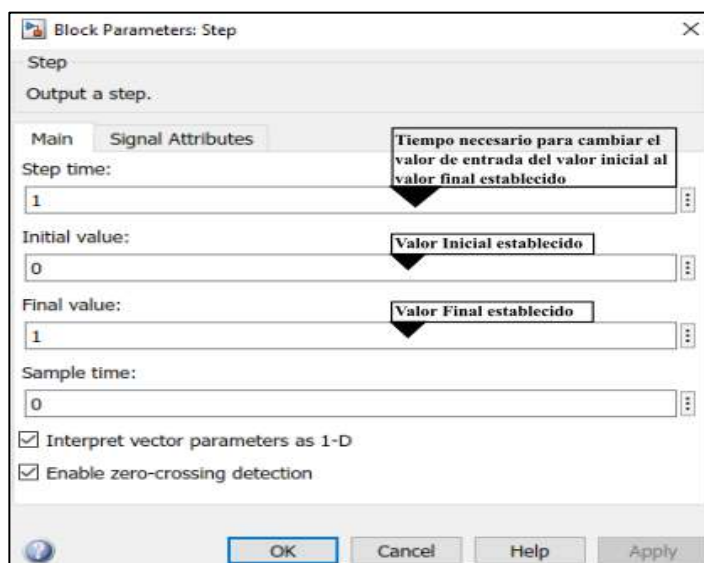
### 3.5.2.1 Elemento final de control

Como elemento final de control se empleará una válvula la cual captará la señal proveniente del controlador y establecerá la cantidad de flujo que ingresa al proceso para mantener a la variable controlada en un valor muy cercano al set Point, y se utiliza la fórmula tanto para el flujo del líquido a través de la válvula como para el flujo de vapor, el tipo de cierre de la válvula de control será un cierre lineal debido a que la capacidad de la válvula tendrá un crecimiento lineal con respecto a la apertura de la válvula.

## 3.6. Desarrollar el sistema de control del proceso en simulink

Una vez simulado el comportamiento dinámico gracias a la ayuda de Matlab y simulink es posible desarrollar el sistema de control:

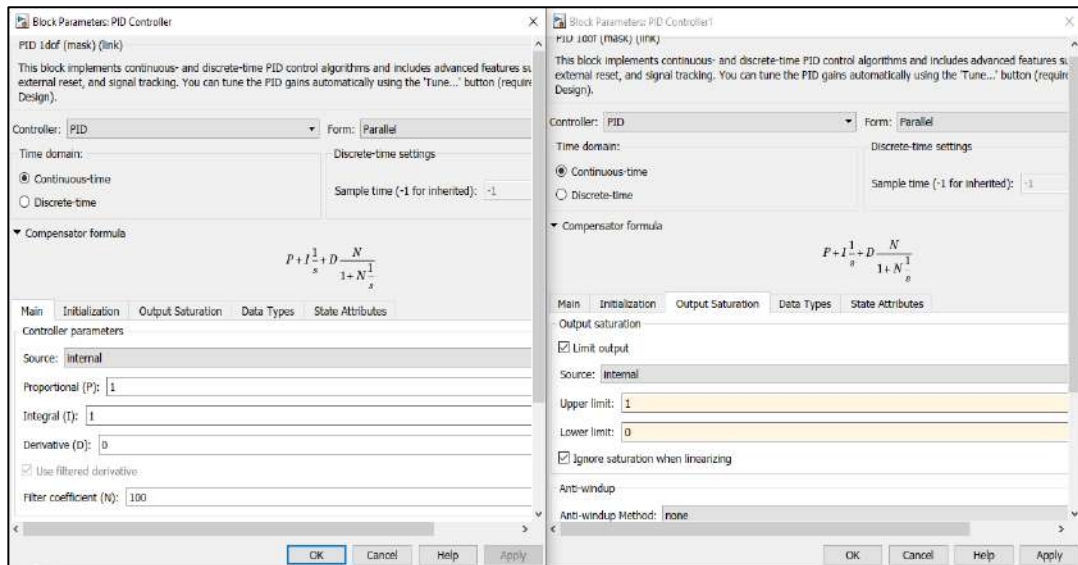
- Ingresar como parámetro de entrada el valor deseado de la variable a controlar o el valor del set point en forma de paso (bloque step) dentro de la biblioteca de simulink.



**Figura 7 – 3.** Configuración del bloque Step dentro de simulink

Elaborado por: Mena, Jerry. 2021.

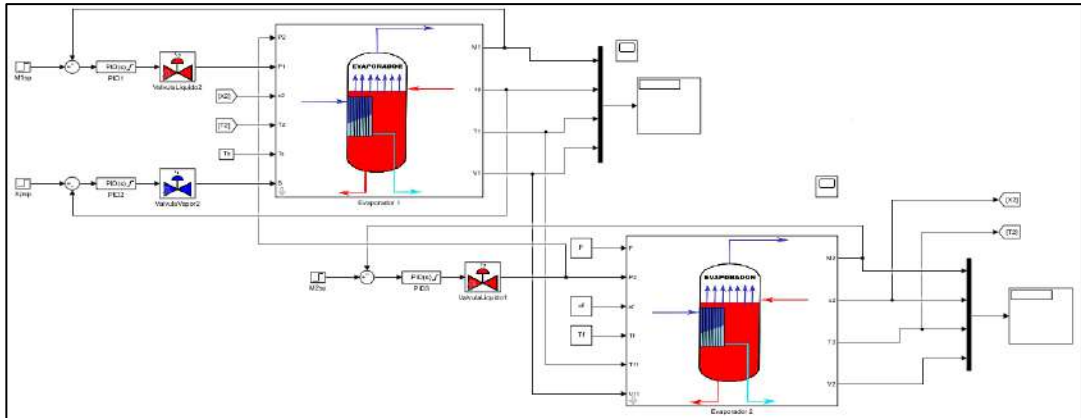
- Colocar el bloque sum y ajustar los signos, donde el signo positivo se conecte con la entrada del set point y el signo negativo con la señal de retroalimentación.
- Situar el bloque denominado PID Controller el cual se usa para realizar el sistema de control, es válido mencionar que la entrada para este bloque es la diferencia entre la señal de referencia y la salida del sistema, denominada también como señal de error; al hacer doble clic al bloque se puede ajustar los parámetros correspondientes a P, I, D y la saturación de apertura la cual se encuentra entre 0 (válvula cerrada) y 1 (válvula completamente abierta).



**Figura 8 – 3.** Configuración del bloque PID Controller dentro de Simulink

Elaborado por: Mena, Jerry. 2021.

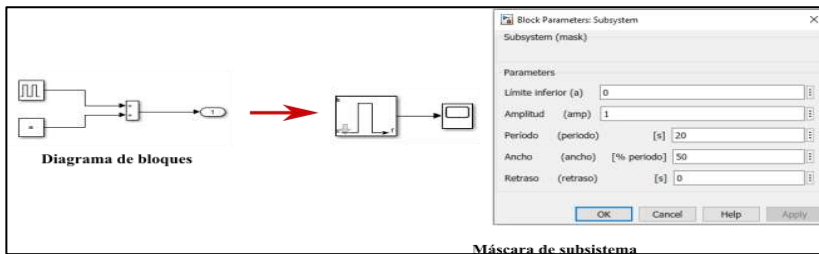
- La señal de salida del bloque PID Controller se conecta al elemento final de control que en este caso será una válvula; se ingresa la ecuación correspondiente al flujo de líquido a través de la válvula dentro de un bloque matlab function, el mismo procedimiento se aplica para una válvula que permita el paso del flujo de vapor, adicional a se crea una máscara de subsistema para cada una de las válvulas, que permite ajustar cada uno de los parámetros involucrados dentro de las ecuaciones tanto para líquido como para vapor.
- La señal resultante del elemento final de control está conectada al proceso, esta señal servirá como un indicador de la variable manipulada, correspondiente a los pares de control enlistados en la tabla 2 - 3, la señal de salida controlada debe ser retroalimentada al bloque sum del sistema de control, estas señales de salida se conectan a un mux el cual va directo a un display para obtener los valores de salida, después se coloca un floating scope con una disposición de 4 gráficas dentro del mismo bloque.



**Figura 9 – 3.** Esquema del modelo de simulink implementado

Elaborado por: Mena, Jerry. 2021.

- Para tener una visión de cómo las perturbaciones afectan a todo el proceso se ingresan las entradas  $F$ ,  $X_f$  y  $T_f$  en forma de pulsos, en un subsistema donde se involucra al bloque pulse generator y adicionar una constante que sirva como el valor para el límite inferior, dentro del bloque pulse generator existen cuatro parámetros el retraso es el tiempo que se tardará hasta dar el primer salto, la amplitud es el límite superior al que llegará el pulso, el periodo es el valor en el que durará el pulso y el ancho es el porcentaje del periodo en el que se dividirá el pulso.



**Figura 10 – 3.** Esquema de la perturbación implementada

Elaborado por: Mena, Jerry. 2021.

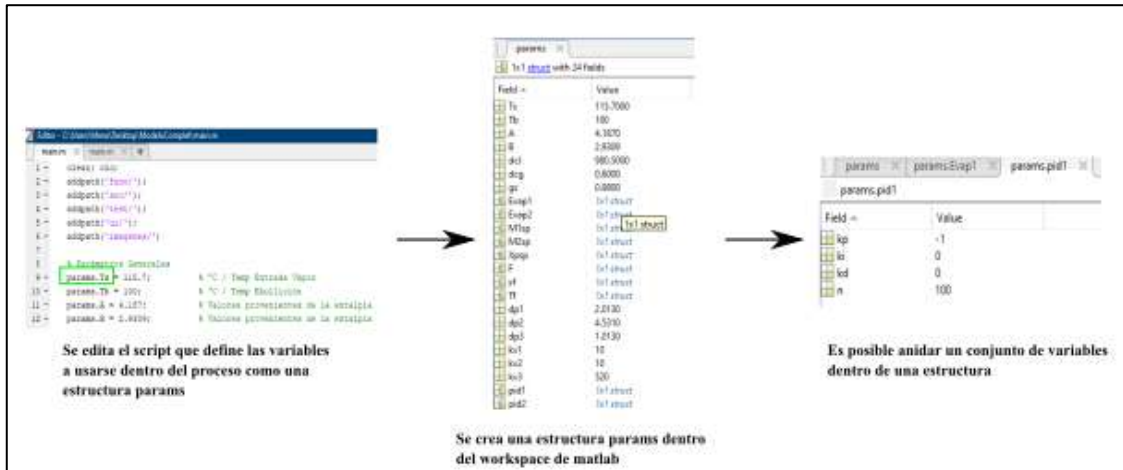
### 3.7. Construir la interfaz gráfica

Para establecer una comunicación entre el modelo de simulink y la interfaz diseñada en App Designer se ingresan algunos comandos dentro de las líneas de código de App Designer esto permite crear el prototipo del modelo, además hay que agregar las entradas en forma de un solo elemento que contenga una lista de parámetros para evitar ingresar uno por uno, también se define las señales que se requieren obtener como salidas.

Para ingresar las entradas en forma de un solo elemento que contenga una lista de parámetros ejecuta en el workspace de Matlab una estructura la cual se denominó params que es un



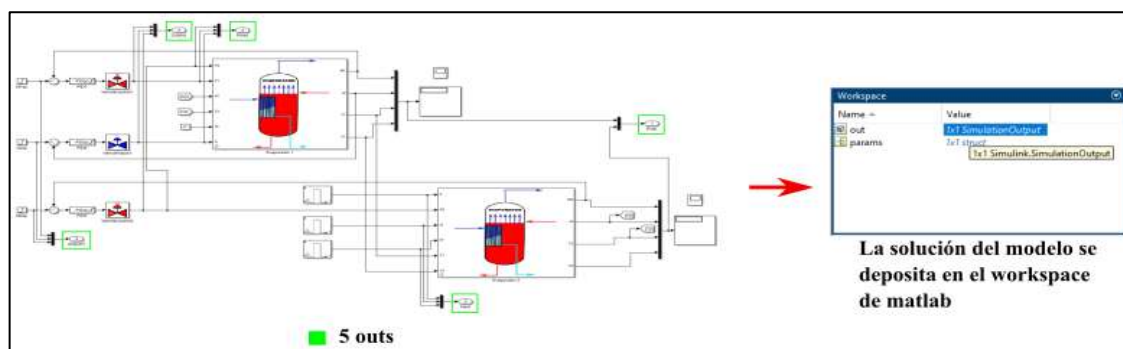
almacenador de datos, dentro de este se pueden definir características o métodos específicos para cada modelo, en esta estructura se pueden anidar campos o elementos para que sea leída por el modelo de simulink se ingresa en cada una de las entradas del modelo los datos de la estructura creada correspondiente para cada entrada.



**Figura 11– 3.** Pasos para crear una estructura dentro de matlab

Elaborado por: Mena, Jerry. 2021.

Para definir las señales que se quieren obtener como salidas sencillamente se añade el bloque out que se encuentra en la librería de simulink. Para evitar crear un sinnúmero de bloques y conexiones innecesarias se coloca un mux dentro del cual se pueden modificar el número de puertos de entrada. Finalmente, la salida de este mux se conectará al bloque out, así todos los resultados que se obtienen de la simulación se almacenarán dentro de una estructura out la cual se encuentra en el workspace de Matlab, estos valores son los que utilizará la aplicación para crear las distintas gráficas que el usuario desee.



**Figura 12 – 3.** Exportar los resultados como señales de salida al workspace de matlab

Elaborado por: Mena, Jerry. 2021.

Para construir la aplicación final, dentro del proyecto se crearon tres interfaces gráficas las cuales contienen cierto tipo de información y realizan diferentes acciones, los pasos para la construcción de la interfaz se detallan a continuación:

### 3.7.1. Interfaz de usuario: inicio

Contiene información general de la aplicación, tiene como funcionalidad establecer la comunicación entre la presente interfaz de inicio y la interfaz con todos los cálculos de la simulación mediante la implementación de un Push button.

- Crear una nueva plantilla de app designer en blanco (inicio.mlapp)
- Colocar los uicontrols de acuerdo con el diseño que se quiera mostrar, en la figura se muestran todos los uicontrols utilizados



**Figura 13 – 3.** Vista preliminar de la interfaz inicio y de los uicontrols utilizados

Elaborado por: Mena, Jerry. 2021.

- Para que cuando se presione el botón COMENZAR se comunique la interfaz inicio con la segunda interfaz se digita las siguientes líneas de código.

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

    % Button pushed function: COMENZARButton
    function COMENZARButtonPushed(app, event)
        interfaz
    end
end
```

**Figura 14 – 3.** Líneas de código para comunicar las interfaces inicio e interfaz

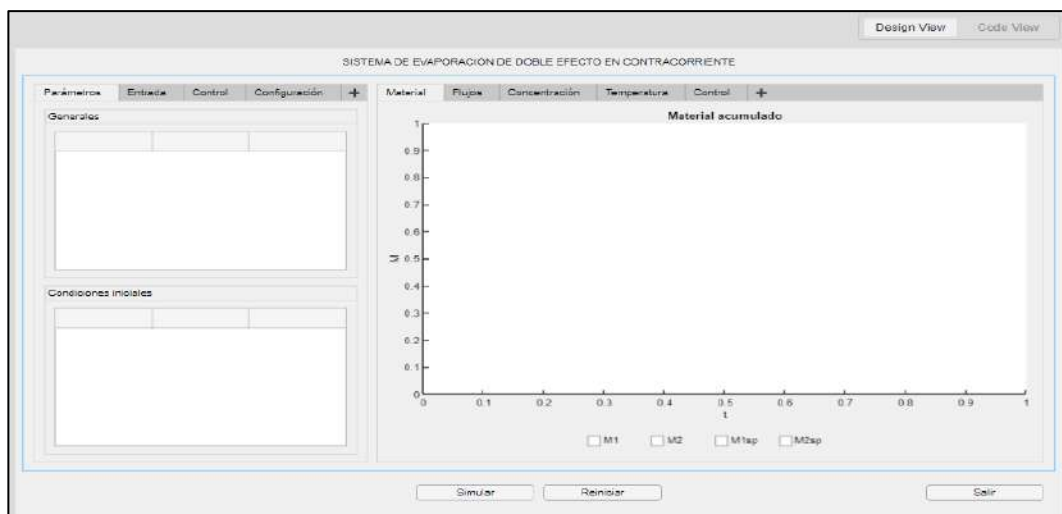
Elaborado por: Mena, Jerry. 2021.

### 3.7.2. Interfaz de usuario: Principal

Contiene la información que arroja como resultados el modelo de simulink, cabe recalcar que para que el código que se muestra en las figuras de la sección 3.7.2 funcionen se necesita tener instalado simulink compiler y el soporte de MATLAB para el compilador MinGW-w64 C / C ++, dichos elementos se instalan a través de Add-Ons de Matlab.

- Crear una nueva plantilla de app designer en blanco (interfaz.mlapp)
- Para el diseño que se quiere mostrar se insertan los uicontrols dentro de la plantilla, con la aplicación de layouts (cuadriculas) se organiza todo el contenido y para mantener un orden se usó tab group. La ventana queda dividida en 3 partes, en la primera parte se encuentra el título del tema desarrollado, en la tercera parte se encuentran botones como simular, reiniciar y salir que me permiten realizar algunas acciones.

En la segunda parte se encuentra el contenido de la aplicación, esta va a estar subdividida en dos partes bien marcadas, dentro de la primera subdivisión se encuentran 4 pestañas distintas estas son parámetros, entradas, control y configuración dentro de cada pestañas se ingresó varios paneles los cuales contienen tablas con valores editables que ingresarán como parámetros del modelo mientras que en la segunda subdivisión a diferencia de la primera existirán 6 pestañas distintas estas son: material, flujos, concentración, temperatura, control e información, dentro de cada pestaña se encuentra una gráfica (axes) que mostrará los resultados de la aplicación y para hacer más entendible la aplicación se colocan check box que al hacer clic sobre alguno de ellos se muestre únicamente ese resultado dando la posibilidad al usuario de elegir qué resultados desea observar, la vista preliminar de la interfaz se muestra en la figura 13-3.



**Figura 15 – 3.** Vista preliminar de la plantilla interfaz y de los uicontrols utilizados

Elaborado por: Mena, Jerry. 2021.

- Para configurar la app despliegan todas las propiedades que participarán en la interfaz, la propiedad params almacenará toda la información del caso por defecto y cuando se modifique el caso guardará los datos para ejecutar una nueva simulación, la propiedad results guardará y clasificará los resultados que provienen de simulink, la propiedad simulation funcionará dentro del caso SimOut, la propiedad config guardará los parámetros de configuración de la interfaz tales como tiempo de inicio, tiempo de parada y tamaño del salto mientras que la propiedad flags guardará el estado activado o desactivado de todos los check box que se encuentran incluidos dentro de la ventana principal de la interfaz.

```

properties (Access = private)
    params % Description
    results
    simulation
    config
    flags
end

```

**Figura 16 – 3.** Código que se ejecuta después de la creación de los componentes

Elaborado por: Mena, Jerry. 2021.

- Una vez definidas las propiedades que participaran dentro del código de la interfaz se carga el caso por defecto que se mostrará al inicializar la aplicación y se procede a copiar los datos del script denominado main, después se crea una función subsidiaria denominada AppFillTables la cual permite rellenar las tablas con la información del caso por defecto.

```

methods (Access = private)
    function defaultCase(app)
        % Overall params
        app.params.Ts = 115.7; % °C
        app.params.Tb = 100; % °C
        app.params.A = 4.187;
        app.params.B = 2.9309;

        % Evaporator's parameters
        app.params.Evap1.M0 = 1000; % kg
        app.params.Evap1.C0 = 0.1; % kg solut/kg sol.
        app.params.Evap1.T0 = 30; % °C

        app.params.Evap2.M0 = 1000; % kg
        app.params.Evap2.C0 = 0.1; % kg solut/kg sol.
        app.params.Evap2.T0 = 30; % °C

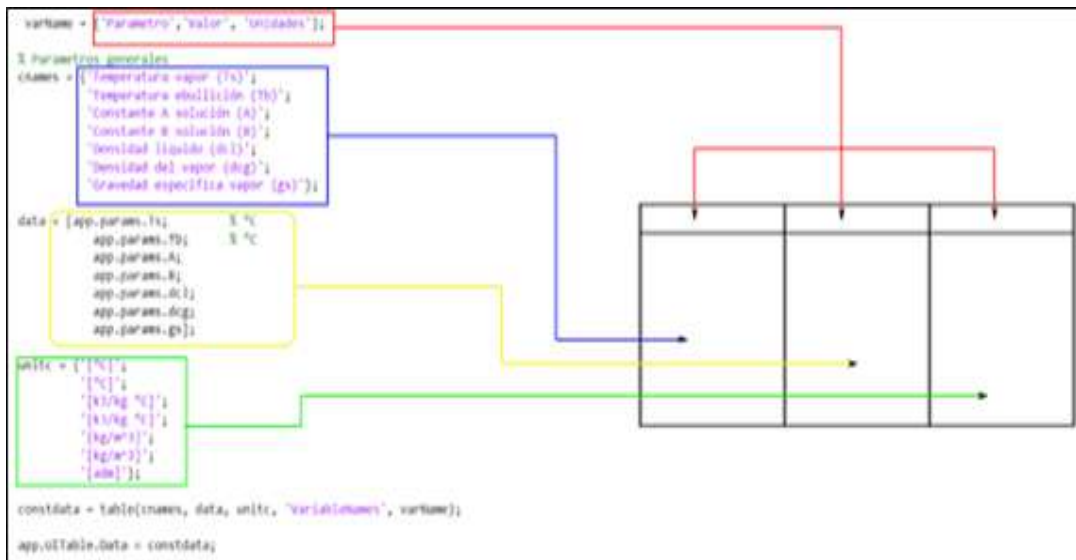
        app.fillTables;
    end

```

**Figura 17 – 3.** Código que almacena la información que se cargará por defecto

Elaborado por: Mena, Jerry. 2021.

- Para poder construir las tablas se desarrollan diferentes variables, una que contenga los nombres de las columnas, luego una con los valores, otra con las unidades y finalmente una que me ingrese todas las variables antes mencionadas dentro de la tabla que va a contener esos valores, el código utilizado se replica a todas las tablas que se encuentran dentro de la interfaz ya que siguen el mismo procedimiento, además todas las tablas contienen la misma estructura parámetro, valor, unidad no se modifican y se declaran por una sola vez.



**Figura 18 – 3.** Esquema de cómo se procede a llenar cada una de las tablas

Elaborado por: Mena, Jerry. 2021.

- Antes de configurar cada uno de los componentes se implementa un callback o función de llamada (startupFcn) que me permita rellenar cada uno de los componentes dentro de la interfaz con los valores establecidos para el caso por defecto.

```

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

    % Code that executes after component creation
    function startupFcn(app)
        % Cargar datos caso por defecto
        app.defaultCase;
    end
end

```

**Figura 19 – 3.** Líneas de código ingresado para que aparezcan los valores por defecto

Elaborado por: Mena, Jerry. 2021.

- Para establecer que los parámetros de entrada cambien en función de lo que el usuario requiera se introduce la siguiente línea de código, la cual consta de una función (getData) dentro de

esta existe una línea de código con la cual se apunta a la tabla que está almacenada dentro de este objeto de control, todos estos valores se asignan por indexación típica.

```
function getData(app)
    % Función para capturar datos desde los campos.

    data = app.UITable.Data.Valor;
    app.params.Ts = data(1);
    app.params.Tb = data(2);
    app.params.A = data(3);
    app.params.B = data(4);
    app.params.dcl = data(5);
    app.params.dcg = data(6);
    app.params.gs = data(7);

    data = app.UITable2.Data.Valor;
    app.params.Evap1.M0 = data(1);
```

**Figura 20 – 3.** Líneas de código para configurar los parámetros de entrada

Elaborado por: Mena, Jerry. 2021.

- Para que la simulación trabaje, se ejecuta una función que establece la comunicación entre el modelo de simulink y appdesigner se digita una línea de código en la cual se pueda cargar el modelo de simulink y los parámetros de entrada gracias a la programación establecida anteriormente solo basta cargar los valores que se encuentran dentro de app.params, para saber el tiempo requerido para la ejecución de la simulación así como la cantidad de pasos que se tomará para resolver la simulación se incluyen las variables startTime, stopTime y FixedStep, entonces el despliegue de la interfaz se configura el modelo de acelerador rápido (RapidAccelerator) para generar el código en C++ y se pueda compilar el modelo cuando se cree el ejecutable de la aplicación, la última línea de código mostrada en la figura 18-3 únicamente guarda el modelo de simulación configurado.

```
function configSimulation(app)

    simInp = Simulink.SimulationInput('DoubleEffectEvaporator');

    simInp = simInp.setVariable('params', app.params);

    simInp = simInp.setModelParameter('startTime', num2str(app.config.startTime));
    simInp = simInp.setModelParameter('stopTime', num2str(app.config.stopTime));
    simInp = simInp.setModelParameter('FixedStep', num2str(app.config.stepTime));

    simInp = simulink.compiler.configureForDeployment(simInp);

    app.simulation = simInp;

end
```

**Figura 21 – 3.** Línea de código para configurar la entrada simInp para el despliegue de la interfaz

Elaborado por: Mena, Jerry. 2021.

- Para estructurar y ver cómo se va a ir ejecutando la simulación se establece una función llamada run simulation la cual simplemente va a ir paso por paso llamando todas las funciones para ejecutar y luego al final presentar los resultados, esta función se ejecuta cuando se presiona el botón resolver o simular, en resumen lo primero que hace es configurar la simulación para eso verifica si la información de entrada obtuvo algún cambio o modificación, modifica params y luego configura la simulación, seguido se corre la simulación aquí se ejecuta la simulación, con la función toggle la cual bloquea el botón, después se procede a guardar los resultados y únicamente se necesita guardar ciertos resultados esto se consigue creando una estructura results acompañada de cada una de las salidas localizadas en el modelo de simulink, de manera lógica a continuación se creó una línea de código principal para dibujar resultados esta dependerá de algunas funciones subsidiarias.

```

function runSimulation(app)
    % 1. Configurar la simulación
    app.getData;
    app.configSimulation;

    % 2. Correr la simulación
    app.toggleSimButton('off', 'Simulando...');
    simOut = sim(app.simulation);
    app.toggleSimButton('on', 'Simular');

    % 3. Guardar los resultados
    app.results.time = simOut.tout;
    app.results.evap = simOut.yout.signals(1).values;
    app.results.control = simOut.yout.signals(2).values;
    app.results.flows = simOut.yout.signals(3).values;
    app.results.setpoint = simOut.yout.signals(4).values;
    app.results.input = simOut.yout.signals(5).values;

    %         disp(app.results);

    % 5. Dibujar las gráficas
    app.drawResults;
end

function toggleSimButton(app, state, label)
    app.ResolverButton.Enable = state;
    app.ResolverButton.Text = label;
end

```

**Figura 22 – 3.** Líneas de código para ejecutar el modelo de simulink dentro de la interfaz

Elaborado por: Mena, Jerry. 2021.

- Las funciones subsidiarias que acompañan a la función principal para dibujar los resultados dependerán de funciones inmersas en la función drawResults las cuales a su vez dependerán de funciones internas las cuales se detallarán en los siguientes ítems.

```
function drawResults(app)

    % Dibujar material acumulado
    app.drawMassAccumulated;

    % Dibujar flujos
    app.drawFlows;

    % Dibujar concentraciones
    app.drawConcentrations;

    % Dibujar temperaturas
    app.drawTemperatures;

    % Dibujar lineas de control
    app.drawControl;

end
```

**Figura 23 – 3.** Línea de código para extraer los resultados de la simulación del modelo de simulink dentro de la interfaz

**Elaborado por:** Mena, Jerry. 2021.

- Para poder graficar los valores los cuales van a ir relacionados con los estados activados de los checkbox, primero se guardan los estados dentro de una estructura denominada hflags, luego debe existir una línea de código que me permita limpiar la gráfica y una línea de código que me permita dibujar más de una línea (hold on), también se agrega una línea de código que me permita agregar una leyenda que va a indicar el color y el nombre de los datos que le corresponde y para dibujar las líneas se utiliza la sentencia condicional if en la cual se establece si se activa el checkbox de ser así se dibuja la gráfica caso contrario no, se sigue el mismo procedimiento para masa acumulada, flujos, concentraciones y temperatura para las gráficas de control al no contar con check box únicamente se coloca líneas de código para limpiar las gráficas y para luego dibujar las nuevas líneas.



```

function drawMassAccumulated(app)
% Flags from checkbox
app.flags.hflags = [app.M1CheckBox.Value,...
                    app.M2CheckBox.Value,...
                    app.m1spCheckBox.Value,...
                    app.m2spCheckBox.Value];

% Clean axes
cla(app.UIAxes4);

% Hold on
hold(app.UIAxes4, 'on');

% Legend names' vector
legname = {};

% Draw Lines
if app.flags.hflags(1) == 1 % M1
    plot(app.UIAxes4, app.results.time, app.results.evap(:,1));
    legname(end+1) = 'M1';
end
if app.flags.hflags(2) == 1 % M2
    plot(app.UIAxes4, app.results.time, app.results.evap(:,5));
    legname(end+1) = 'M2';
end
if app.flags.hflags(3) == 1 % m1sp
    plot(app.UIAxes4, app.results.time, app.results.setpoint(:,1), '--', 'Color', [0.6350 0.0780 0.1840]);
    legname(end+1) = 'm1sp';
end
if app.flags.hflags(4) == 1 % m2sp
    plot(app.UIAxes4, app.results.time, app.results.setpoint(:,3), '--', 'Color', [0.6350 0.0780 0.50]);
    legname(end+1) = 'm2sp';
end

% Add legend
legend(app.UIAxes4, legname);

ylabel(app.UIAxes4, 'M [kg]');

```

**Figura 24 – 3.** Línea de código para graficar los valores del set point

Elaborado por: Mena, Jerry. 2021.

- Para completar las funcionalidades de la interfaz se utiliza callbacks o funciones de llamada tanto para los botones como para cada uno de los checkbox estos deben tener comunicación directa con las funciones que realizan diversas acciones y se detallaron anteriormente.

```

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

% Code that executes after component creation
function startupFcn(app)
    % Cargar datos caso por defecto
    app.defaultCase;
end

% Button pushed function: SimularButton
function SimularButtonPushed(app, event)
    app.runSimulation;
end

% Value changed function: M1CheckBox
function M1CheckBoxValueChanged(app, event)
    app.drawMassAccumulated;
end

% Value changed function: M2CheckBox
function M2CheckBoxValueChanged(app, event)
    app.drawMassAccumulated;
end

% Value changed function: M1spCheckBox
function M1spCheckBoxValueChanged(app, event)
    app.drawMassAccumulated;
end

```

**Figura 25 – 3.** Línea de código para graficar los valores del set point

Elaborado por: Mena, Jerry. 2021.

- Para desplegar la información contenida en la interfaz de usuario info secrea un callback para el botón InfoButton y colocar la siguiente línea de código.

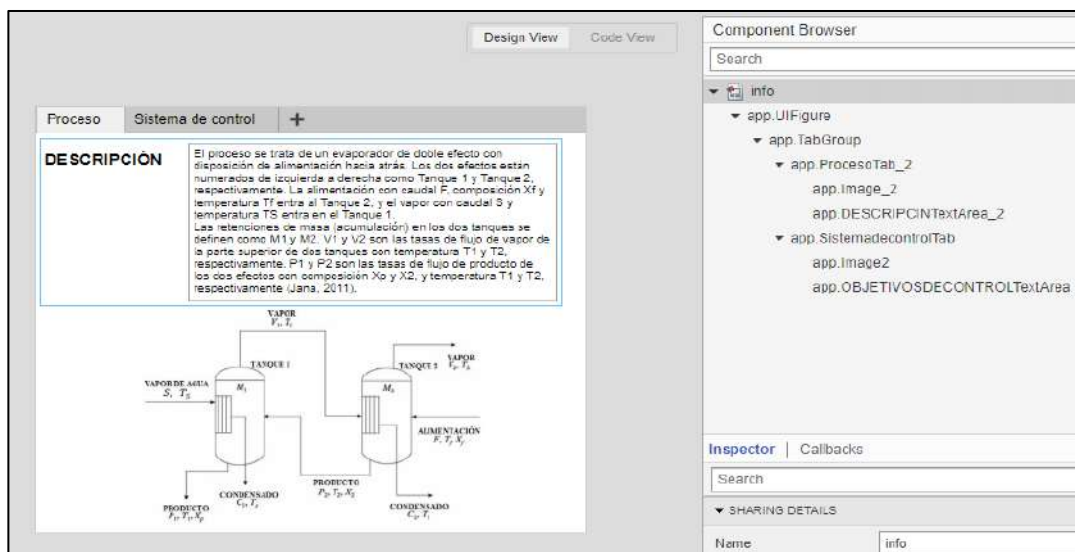
```
% Button pushed function: InfoButton
function InfoButtonPushed(app, event)
    info
end
```

**Figura 26 – 3.** Línea de código para desplegar la información contenida en la interfaz de usuario

Elaborado por: Mena, Jerry. 2021.

### 3.7.3. Interfaz de usuario: info

- Crear una nueva plantilla de app designer en blanco (info.mlapp)
- Ubicar los uicontrols acorde al diseño que se quiera mostrar, en la figura 23-3 se muestran todos los uicontrols utilizados

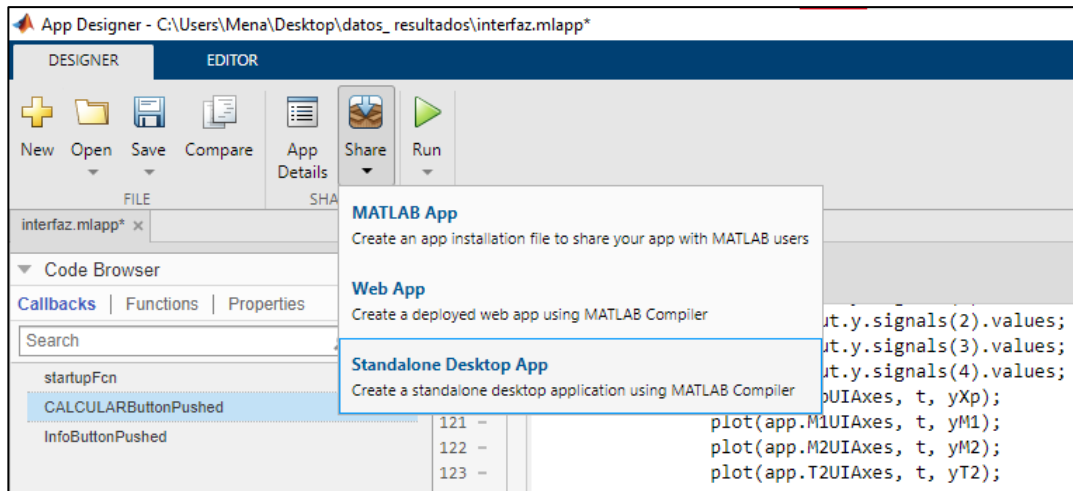


**Figura 27 – 3.** Vista preliminar de la plantilla info.mlapp y de los uicontrols utilizados

Elaborado por: Mena, Jerry. 2021.

### 3.8. Empaquetar la interfaz gráfica diseñada

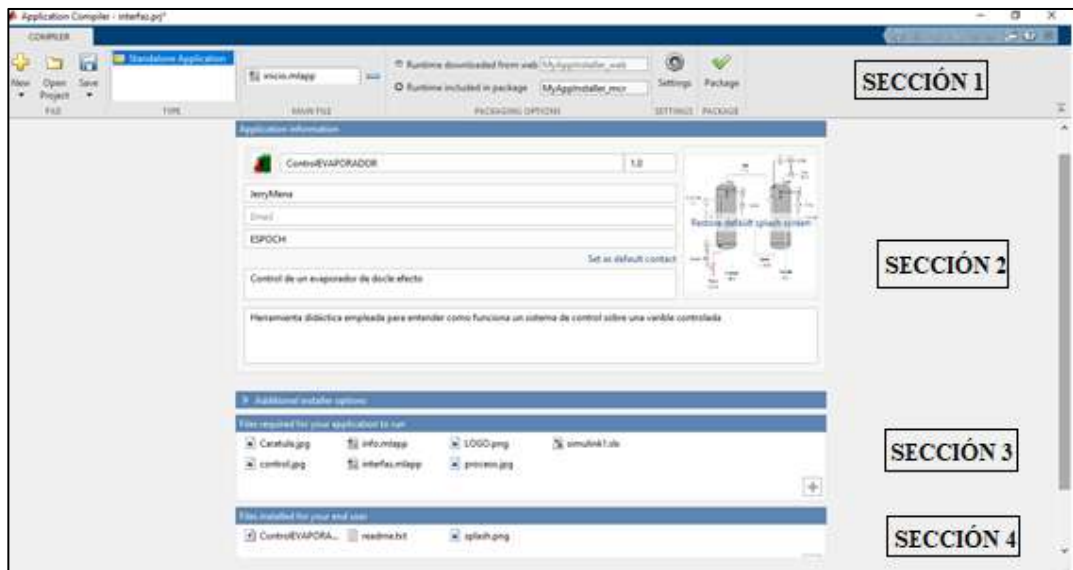
- Para empaquetar la interfaz gráfica y crear una aplicación independiente (ejecutable.exe), se localiza en la barra de herramientas el elemento share y desplegar las opciones, se selecciona la opción Standalone Desktop App.



**Figura 28 – 3.** Localización de share en la ventana de herramientas de App designer

Elaborado por: Mena, Jerry. 2021.

- Una vez abierta la pestaña de Matlab compiler se encuentra varias secciones las cuales son imprescindibles para el empaquetamiento, para iniciar con el empaquetamiento es preciso llenar los espacios en las secciones que se describen en la figura 28-3:



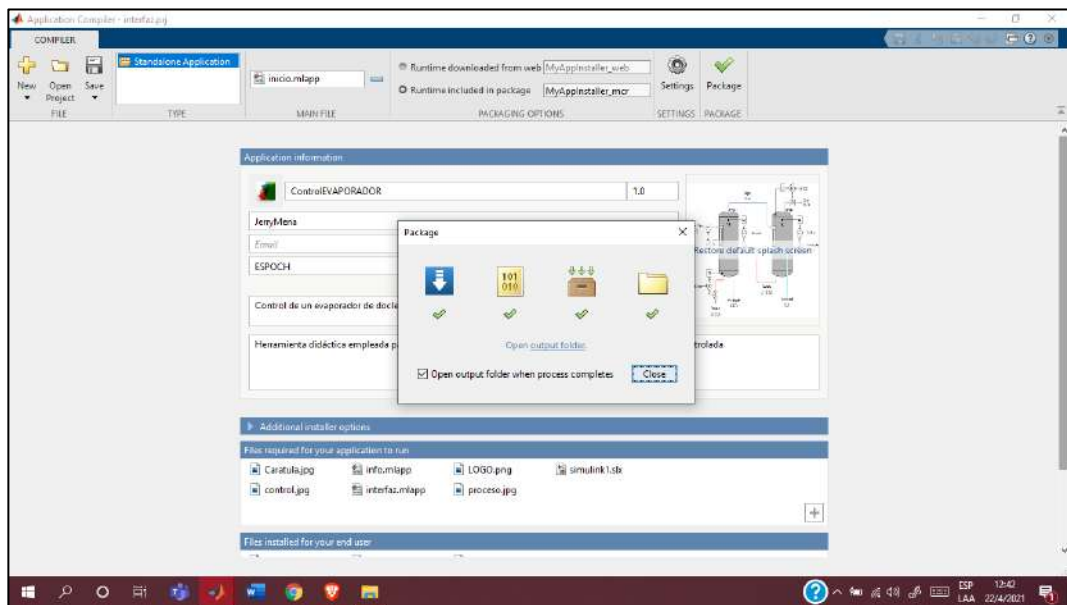
**Figura 29 – 3.** Distribución de la ventana de App Compiler

Elaborado por: Mena, Jerry. 2021.

- Sección 1: Dentro de esta sección se ubica la interfaz de usuario que va a iniciar la aplicación en este caso se denomina inicio este procedimiento se realiza en la parte de main file, luego de esto se procede a seleccionar si el paquete de runtime; para que la aplicación se ejecute sin la necesidad de instalar Matlab venga incluido en el empaquetamiento o pueda ser instalado desde la web, la ventana de settings contiene aspectos relacionados a donde se

guardará la aplicación después de haberse empaquetado y el botón Package se pulsará una vez llenas todas las secciones.

- Sección 2: Aquí se incluyen datos referentes a el desarrollador de la aplicación, así como el nombre y una breve descripción de lo que contendrá esta.
- Sección 3: Se llena automáticamente al seleccionar la interfaz de usuario en la parte de main file, esta sección contiene todos los elementos para que la creación de la aplicación externa se genere con éxito.
- Sección 4: Incluye elementos que deberán ser ejecutados por el usuario (se llena automáticamente).
- Para determinar que el empaquetamiento se realizó de manera exitosa se efectua checks en cada uno de los siguientes iconos de lo contrario existe la certeza de que al ejecutar la aplicación de manera externa se presenten problemas y en el peor de los casos no exista una ejecución del programa.



**Figura 30 – 3.** Resultado exitoso del empaquetamiento

Elaborado por: Mena, Jerry. 2021.

- El resultado del empaquetamiento genera tres carpetas las cuales constan de las siguientes partes:  
For\_redistribution: contiene el runtime y el instalador del programa  
For\_redistribution\_files\_only: abarca el ejecutable y archivos de lectura  
For\_testing: consta del ejecutable e informes provenientes del empaquetamiento

## CAPÍTULO IV

### 4. GESTIÓN DEL PROYECTO

#### 4.1. Recursos y materiales

##### 4.1.1. Presupuesto

El monto de inversión del trabajo de titulación se establece dentro del siguiente esquema de presupuesto:

**Tabla 1 - 4.** Presupuesto general del proyecto de titulación.

PRESUPUESTO			
CONCEPTO	MONTO (\$)	FUENTE DE FINANCIAMIENTO	
		INTERNA	EXTERNA
Investigación y aprendizaje (internet)	150	x	
Impresión del documento final	30	x	
Imprevistos	20	x	
Empastados	50	x	
<b>TOTAL</b>	250		

Elaborado por: Mena, Jerry. 2021.

##### 4.1.2. Recursos materiales

El proyecto de titulación se desarrolla mediante el uso de los siguientes softwares:

Productos de MATLAB utilizados (versión R2021a)

- Simulink
- Herramienta App designer
- Simulink Compiler
- Soporte de MATLAB para el compilador MinGW-w64 C / C ++

A su vez, todos los procedimientos mencionados se realizaron en un ordenador con procesador Intel® Core™ i3-4030U CPU @ 1.90GHz 1.90 GHz, con 4,00 GB de memoria RAM, sistema operativo de 64 bits y Windows 10 Pro.

## 4.2. Cronograma

El tiempo establecido para la realización del proyecto se establece dentro del siguiente cronograma:

**Tabla 2 – 4.** Cronograma para la realización del proyecto de titulación.

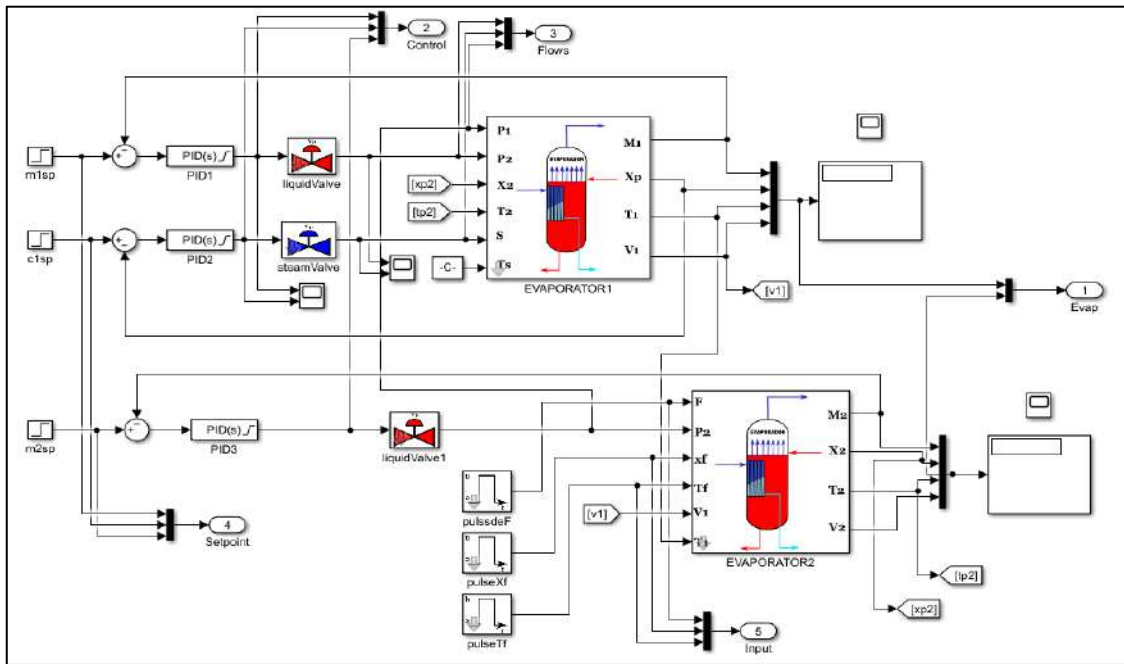
ACTIVIDAD	TIEMPO															
	1° mes				2° mes				3° mes				4° mes			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Revisión bibliográfica	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
Elaboración del anteproyecto			x	x												
Determinación de variables y modelado del equipo					x	x										
Modelado matemático del sistema de control para el proceso							x	x	x	x	x					
Simulación del trabajo													x	x		
Creación de una aplicación formato.exe													x	x		
Elaboración y corrección de borradores, tipiado del trabajo final	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
Empastado y presentación del trabajo final													x	x	x	
Revisión del documento final													x	x	x	
Defensa del trabajo																x

Elaborado por: Mena, Jerry. 2021.

### 4.3. Resultados

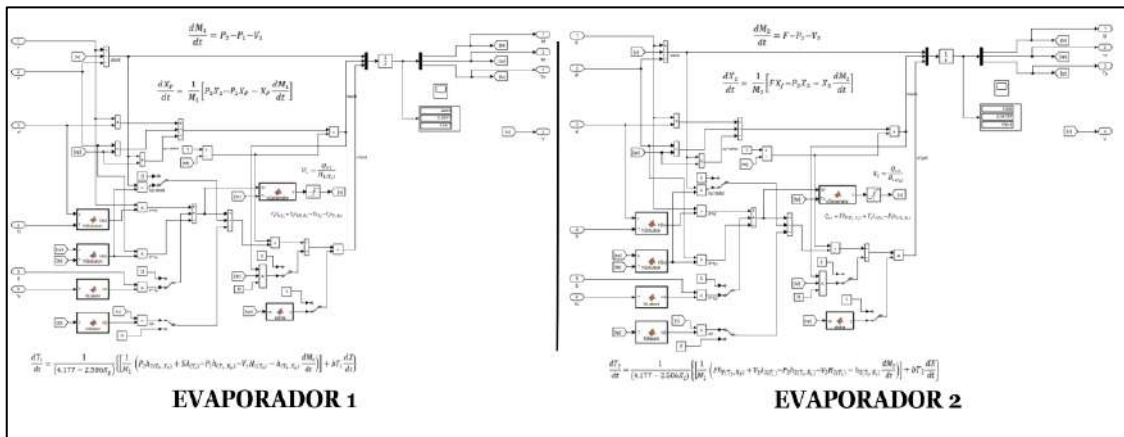
#### 4.3.1. Informe de resultados de la simulación

La figura 1-4 engloba de forma esquemática la simulación del evaporador de doble efecto mediante varios subsistemas los cuales arrojan como resultados el valor de la variable M1, Xp, T1, V1, M2, X2, T2, V2, P1, S Y P2 estos valores son útiles para implementar el sistema de control.



**Figura 1 – 4.** Esquema de la simulación completa del evaporador de doble efecto

Elaborado por: Mena, Jerry. 2021.



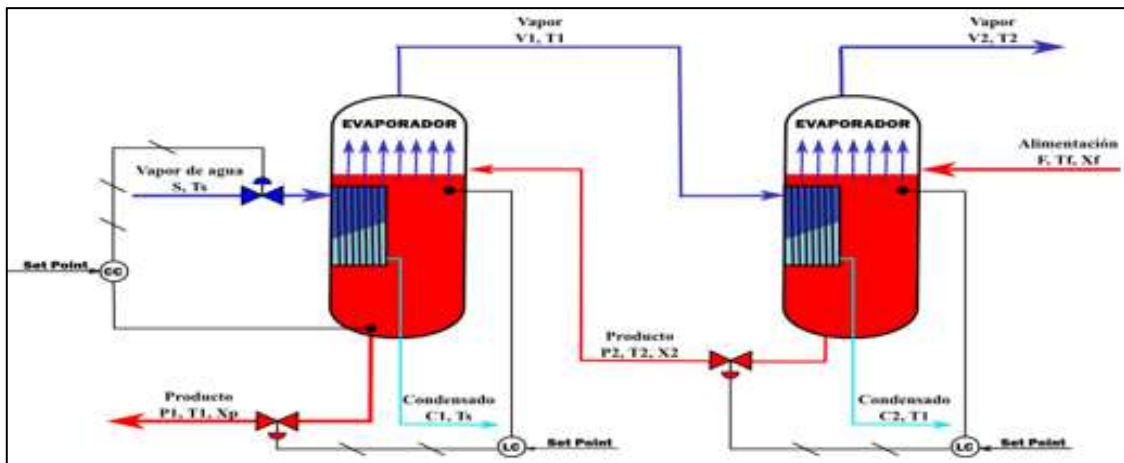
**Figura 2 – 4.** Esquema de la simulación detallada del evaporador de doble efecto

Elaborado por: Mena, Jerry. 2021.

Para la resolución de las ecuaciones diferenciales que se ingresaron directamente al modelo de simulink fue necesario apuntar por un método de aproximación iterativa a la solución de ecuaciones diferenciales ordinarias. Siguiendo las especificaciones de Yadav & Jana (2010. p.4), se dispuso la configuración del solver a paso fijo (Fixed-step) de tipo ode4 (Runge-Kutta) con un tamaño de paso 0.1, todos los parámetros de entrada obedecen a los valores expuestos por Yadav & Jana (2010) en el artículo “Simulation and Control of a Commercial Double Effect Evaporator: Tomato Juice”.

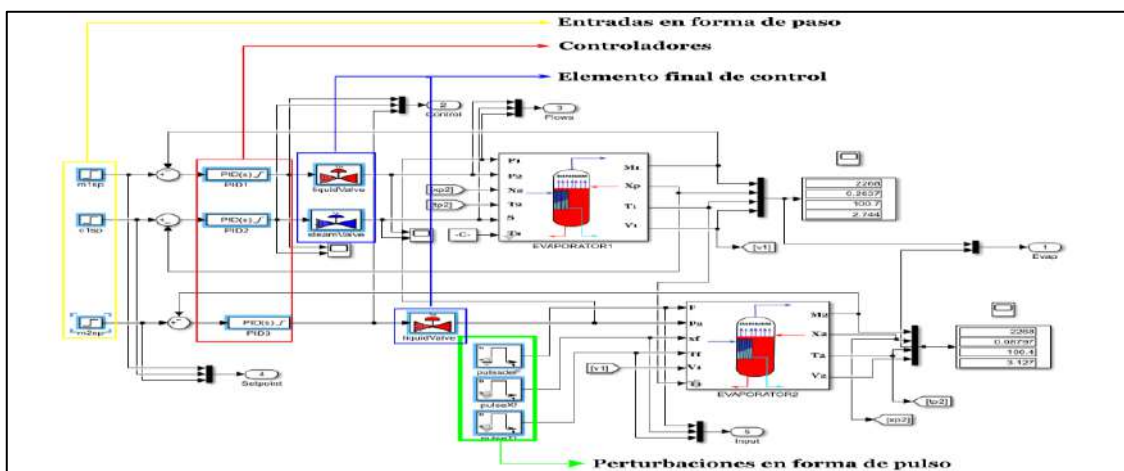
#### 4.3.2. Informe de resultados de la implementación del sistema de control

De la simulación realizada e implementando un modelo sencillo de control el cual se sintetiza en la figura 3-4 y 4-4, se obtuvieron los siguientes resultados:



**Figura 3 – 4.** Esquema del sistema de control que se va a implementar

Elaborado por: Mena, Jerry. 2021.



**Figura 4 – 4.** Esquema de los recursos implementados dentro del sistema de control

Elaborado por: Mena, Jerry. 2021.



#### 4.3.2.1 Controladores usados dentro del sistema de control

El controlador usado dentro del sistema de control fue el controlador con acción de control proporcional (P) con este se obtuvo una menor diferencia respecto al valor del punto de consigna o set point; el método de sintonización del controlador fue realizado con la técnica de ajuste de prueba y error, que consiste en colocar valores de manera empírica y verificar con que valor en las constantes se ajusta mejor el controlador.

El ingreso del valor seteado responde a la acción directa o inversa que se ejecutará al aplicar el sistema de control esta indica que:

- Si  $K_p < 0$  existirá una acción directa del controlador
- Si  $K_p > 0$  existirá una acción inversa del controlador

Para el sistema en cuestión los valores obtenidos para las constantes del controlador son las siguientes:

**Tabla 3 – 4.** Valores obtenidos para las constantes del controlador.

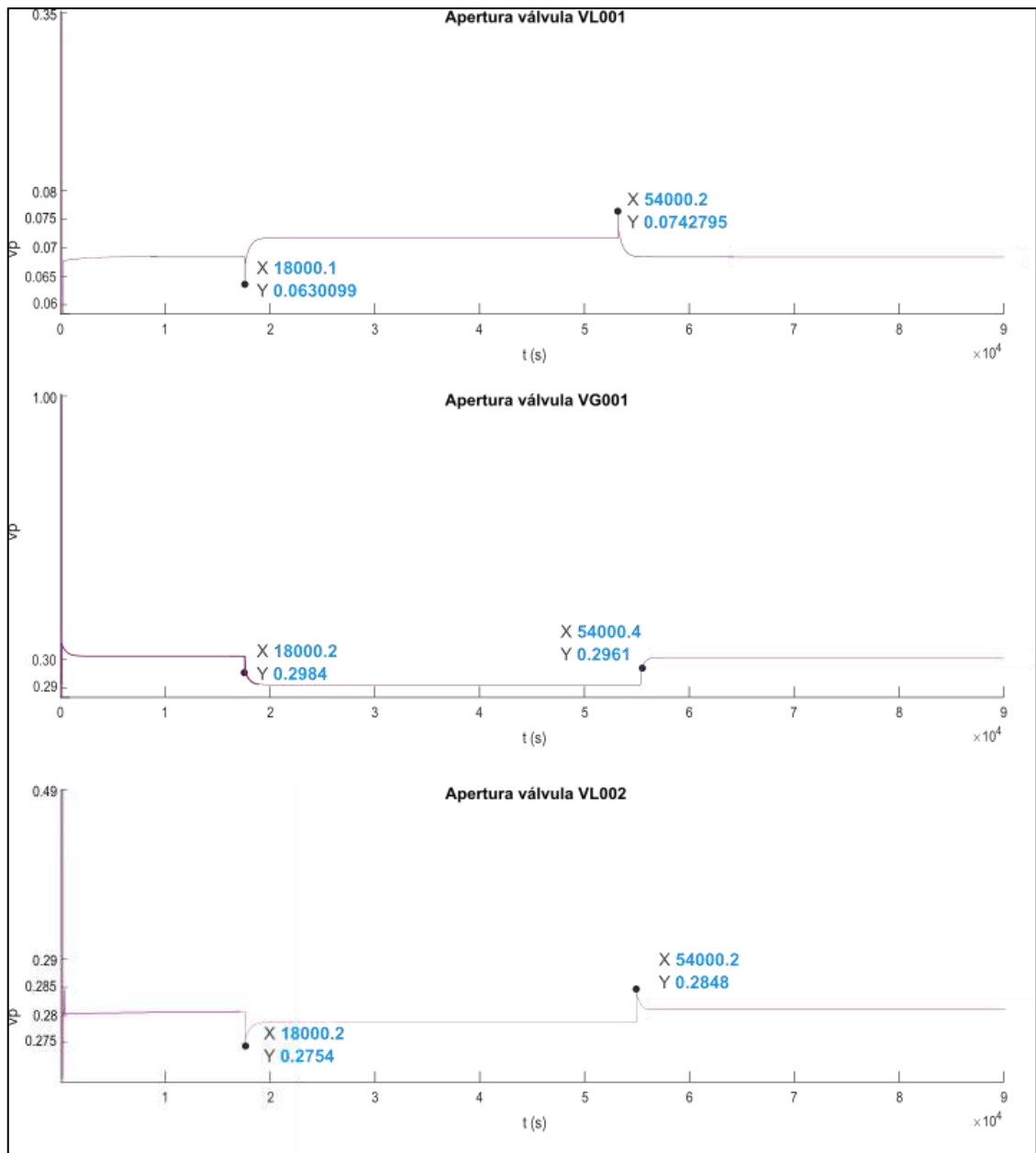
	Tipo de Controlador	Valor seteado	Razón de colocación
<b>CONTROLADOR 1</b>	Proporcional	-1	Dado que todos los controladores usados responden a la acción de control proporcional las características para su colocación fueron las siguientes: <ul style="list-style-type: none"><li>• Tiempo de subida: Decrece</li><li>• Sobre impulso: Crece</li><li>• Tiempo de estabilización: Cambio menor</li><li>• Error en estado estable: Decrece</li></ul>
<b>CONTROLADOR 2</b>	Proporcional	10000	
<b>CONTROLADOR 3</b>	Proporcional	-1	

Elaborado por: Mena, Jerry. 2021.

#### 4.3.2.2 Análisis de las gráficas resultantes de la implementación del sistema de control

En concordancia con lo mencionado con anterioridad, a continuación, se realiza un análisis de resultados para cada una de las gráficas obtenidas de la implementación del sistema de control.

Dentro de la configuración de los controladores se establece la saturación de salida la cual según Smith & Corripio (2014, p.219) está en un intervalo de 0 a 1 y se obtienen los siguientes resultados los cuales corresponden a la válvula de líquido, válvula de vapor de agua y válvula de líquido 2.

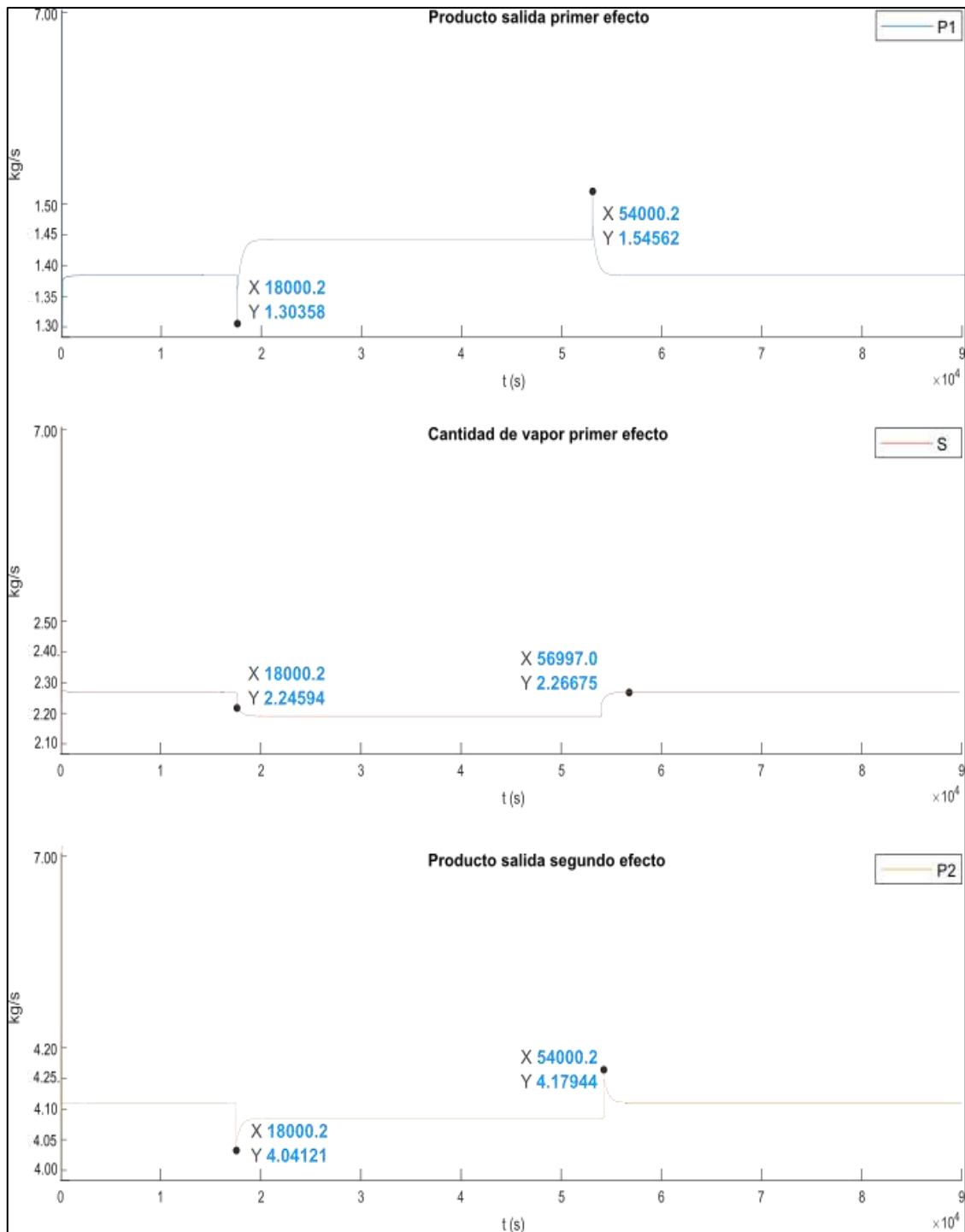


**Figura 5 – 4.** Representación gráfica de los resultados de salida de la señal del controlador

Elaborado por: Mena, Jerry, 2021.

La figura 5-4 denota la posición de cada una de las válvulas, dicha señal se utiliza como señal de entrada al elemento final de control la cual contribuye a determinar la apertura de la válvula que no es más que el grado de restricción al flujo.

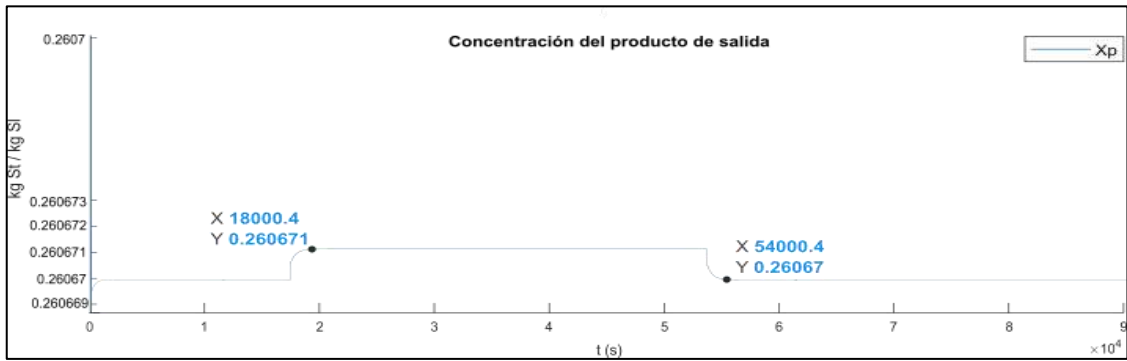
A partir de lo mencionado en el párrafo anterior es posible obtener el valor numérico de cada una de las variables manipuladas las cuales se evidencian en la figura 6-5 y se tienen los siguientes resultados:



**Figura 5 – 4.** Representación gráfica de los resultados de salida del elemento final de control

Elaborado por: Mena, Jerry, 2021.

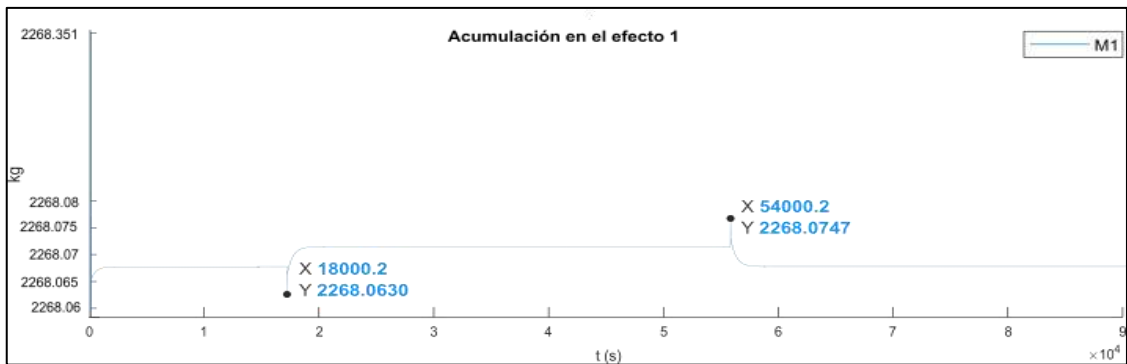
Consecutivamente siguiendo con el análisis de resultados para cumplir con el objetivo de control el cual busca mantener la concentración del producto en un valor deseado es necesario implementar el sistema de control que involucre al valor deseado o set point y la variable que se requiere controlar, adicional a ello en la figura 5-4 se evidencia el valor correspondiente a la señal de la variable  $X_p$ .



**Figura 6 – 4.** Representación gráfica del resultado del sistema de control para la variable controlada Xp

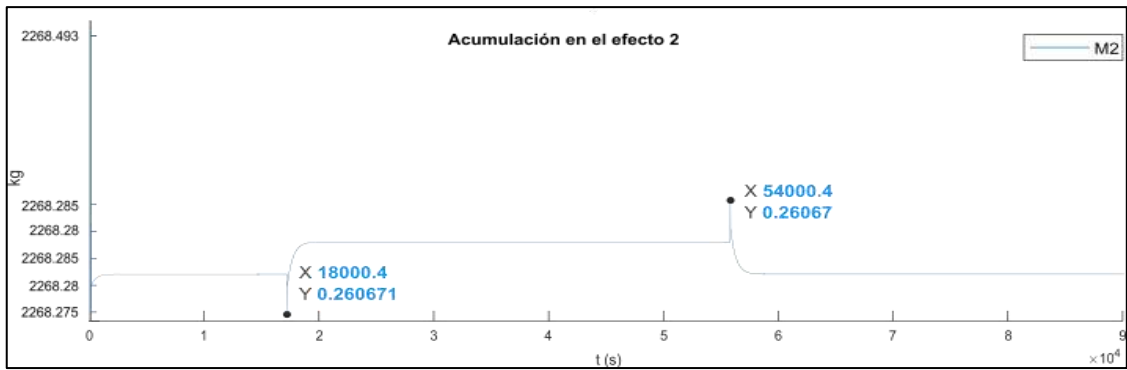
Elaborado por: Mena, Jerry, 2021.

El segundo objetivo de control establece que, para evitar el desbordamiento o la desecación de los tubos del evaporador, es necesario controlar la retención de masa líquida también llamada acumulación, para lo cual se diseñaron dos sistemas de control por separado estos involucran la señal del set point de manera independiente en cada sistema de control, las variables controladas M1 y M2 correspondientes al primero y segundo sistema de control respectivamente. Las figuras 12-4 y 13-4 que se presentan a continuación esquematizan los sistemas de control empleados.



**Figura 7 – 4.** Representación gráfica del resultado del sistema de control para la variable controlada M1

Elaborado por: Mena, Jerry, 2021.



**Figura 8 – 4.** Representación gráfica del resultado del sistema de control para la variable controlada M2

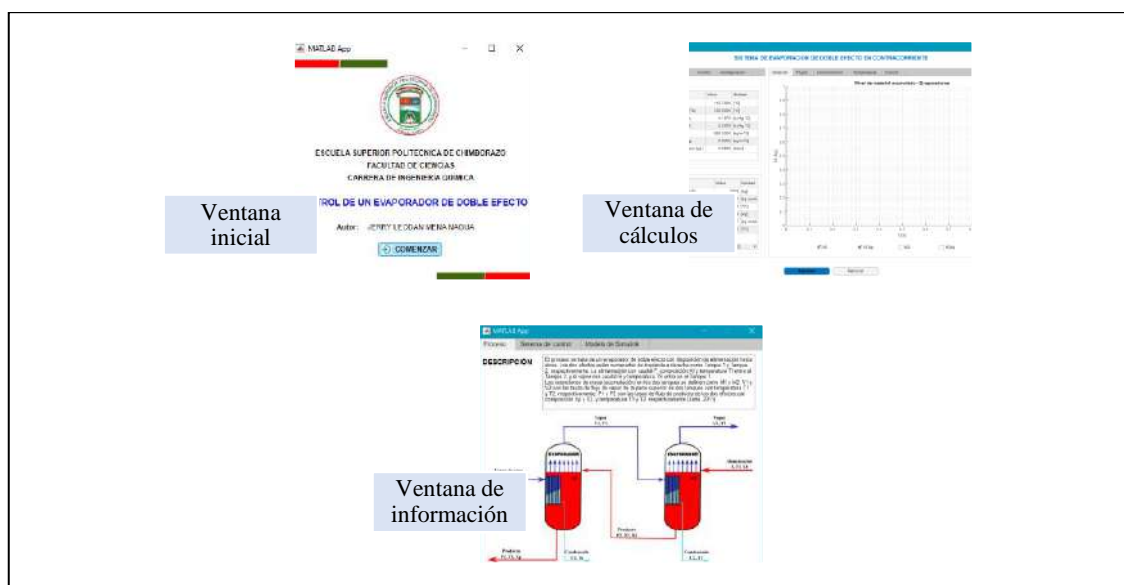
Elaborado por: Mena, Jerry, 2021.

#### 4.3.3. Informe de resultados de la aplicación generada

Dentro de este apartado se realiza la ejecución de la aplicación y se obtiene la representación de como varían las variables controladas dentro del proceso ( $X_p$ , M1, M2) respecto a los valores establecidos en el set point, esto determinará qué tan preciso es el control implementado, después se señala la diversificación de la aplicación al permitir ingresar los valores de las perturbaciones como pulsos y los parámetros de entrada del set point como un bloque paso (step).

Para ejecutar la aplicación se introducen diferentes parámetros de entrada para verificar como actúa el sistema de control, finalmente se ingresa un tiempo determinado que establecerá la duración de la simulación en curso.

A continuación, en el gráfico 1-4 se tiene una representación de la estructura de la aplicación.



**Gráfico 1 – 4.** Estructura de la aplicación final (.exe)

Elaborado por: Mena, Jerry, 2021.

Seguidamente se explicará a detalle todas las ventanas que conforman la aplicación las cuales se presentan en el gráfico 1-4.

- Al momento de ejecutar la aplicación se despliega una ventana que contiene información relacionada al autor, el nombre del proyecto y un botón de comenzar el cual nos redirige hacia la ventana principal del proyecto.
- La ventana principal del proyecto establece la comunicación con la parte simulada proveniente del modelo diseñado en simulink y nos permite mostrar los resultados al presionar el botón resolver.
- Como instrumento adicional existe una tercera ventana la cual se ejecuta al presionar el botón información referente al proceso ubicada en la ventana principal, esta ventana adicional está conformada por cuatro pestañas la primera proporciona información relacionada al proceso, la segunda expone información referente al sistema de control, la tercera exhibe las perturbaciones inmersas en el sistema y la cuarta expone detalles generales sobre el modelo de simulink.

#### **4.3.4. Validación de la modelación programada**

Una manera clara de determinar qué tan exacta fue la modelación programada es validando el modelo simulado con los parámetros existentes en bibliografía, para ello surge la necesidad de buscar un estudio que reúna las mismas características que la del equipo empleado en el presente trabajo, por tal razón el trabajo que se toma en cuenta para la validación, es un artículo publicado en la revista Chemical Product and Process Modeling titulado “Simulation and Control of a Commercial Double Effect Evaporator: Tomato Juice” expuesto por Yadav & Jana (2010).

Con los valores de entrada proporcionados por Yadav & Jana, (2010, p.4) ingresados en la presente simulación, un efecto de un cambio de entrada de pulso en la concentración de alimento (cambio de 0.05 a 0.0525 kg St / kg Sl al tiempo = 5 horas (18000 s) y luego de 0.0525 a 0.05 kg St / kg Sl en el tiempo = 15 horas (54000)) y obedeciendo al tiempo de simulación (25 horas = 90000 segundos), se obtiene una diferencia entre los valores reales y los valores de set point, dicha diferencia radica primero en la longitud del eje de las x puesto que el tiempo establecido en la simulación realizada se encuentra en segundos y los valores de la gráfica tomada de la referencia bibliográfica se encuentra en horas adicional a ello se suman los cambios a las ecuaciones de vapor para la presente investigación.

También se realiza un efecto de un cambio de entrada de pulso en la temperatura de alimentación (cambio de 85 a 95 C al tiempo = 5 horas (18000 s) y luego de 95 a 85 C en el tiempo = 15 horas (54000)) y obedeciendo al tiempo de simulación (25 horas = 90000 segundos), se obtiene una diferencia entre los valores reales y los valores de set point, dichos cambios se deben a que dentro

del modelado se usaron diferentes ecuaciones diferenciales para definir la cantidad de vapor a la entrada de cada uno de los efectos.

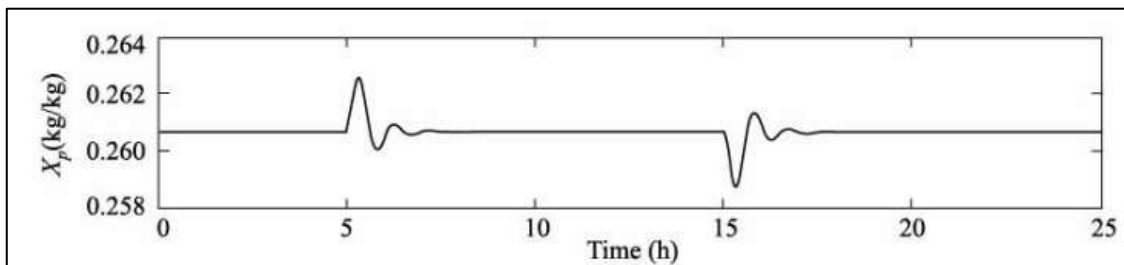
Los resultados de mayor relevancia en la simulación como la concentración del producto ( $X_p$ ) y la acumulación en el primer efecto (M1) serán los resultados que se valide, para el cálculo del error porcentual se utilizará la siguiente fórmula:

$$e = \frac{V. real - V. calculado}{V. real} \times 100$$

V.real: Valor resultante del artículo científico para la validación

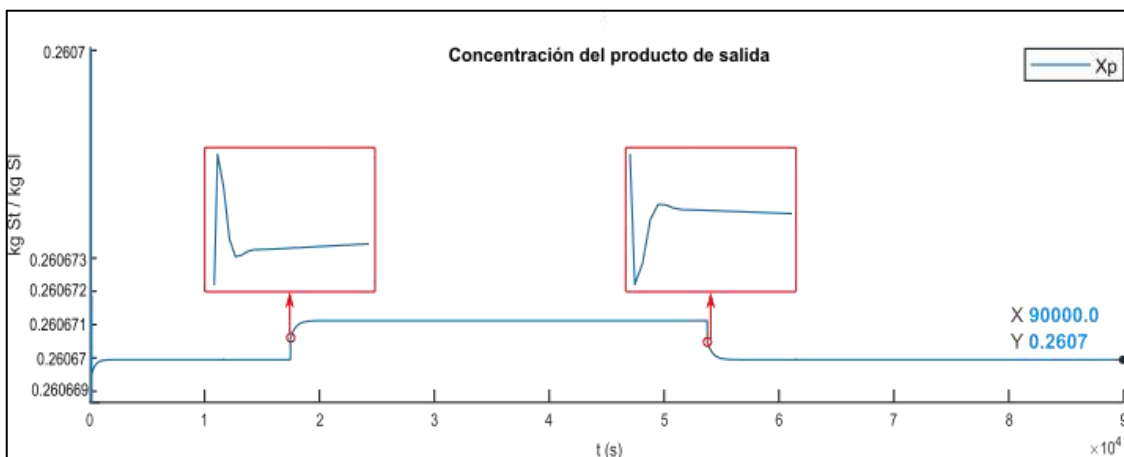
V.calculado: Valor resultante de la aplicación generada

#### 4.3.4.1 Concentración del producto ( $X_p$ )



**Figura 9 – 4.** Resultado bibliográfico de la concentración en el primer efecto

Elaborado por: (Yadav & Jana, 2010, p.).



**Figura 10 – 4.** Simulación establecida para comprobar el funcionamiento de la aplicación

Elaborado por: Mena, Jerry, 2021.

A partir de los resultados obtenidos es posible calcular el error porcentual a partir de la siguiente fórmula:

$$e = \frac{V. real - V. calculado}{V. real} \times 100$$

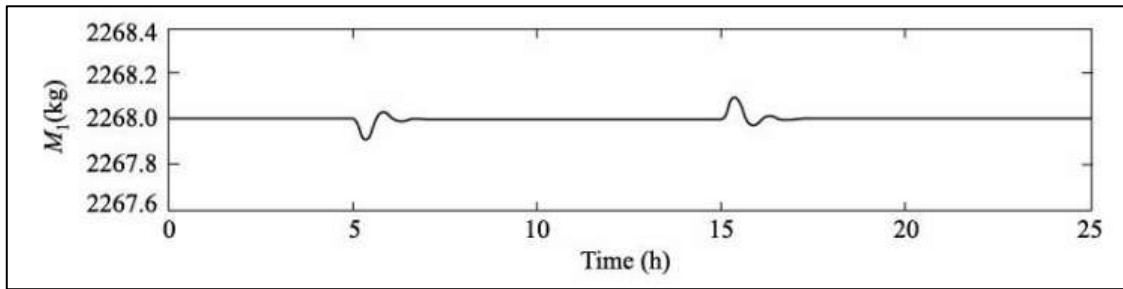
**Tabla 4 - 4.** Resultados y error porcentual ( $X_p$ ).

	Concentración del producto ( $X_p$ )
Valor real	0,2609
Valor calculado	0,2607
Error porcentual	0,0008
e (%)	0,08

Elaborado por: Mena, Jerry. 2021.

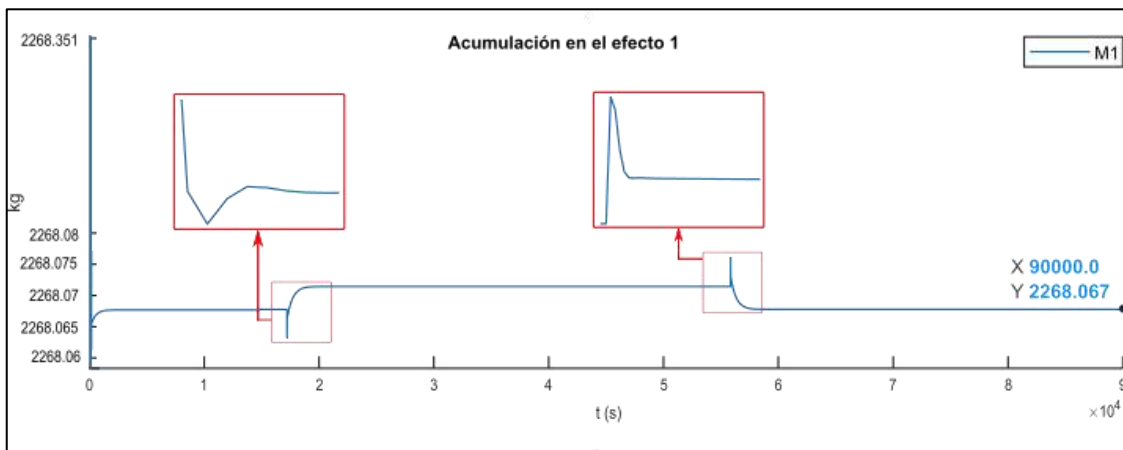
El error porcentual obtenido es 0.08%, como se muestra en la Tabla 4-4, es menor que el error total permisible, por ende, se da por aceptada la simulación de los resultados obtenidos por la interfaz desarrollada a partir del modelado matemático en Simulink.

#### 4.3.4.2 Acumulación en el primer efecto ( $M1$ )



**Figura 11 – 4.** Resultado bibliográfico de la acumulación en el primer efecto

Elaborado por: (Yadav & Jana, 2010, p.).



**Figura 12 – 4.** Segunda simulación establecida para comprobar el funcionamiento de la aplicación (acumulación en el primer efecto)

Elaborado por: Mena, Jerry, 2021.

$$e = \frac{V. real - V. calculado}{V. real} \times 100$$



**Tabla 5 – 4.** Resultados y error porcentual (M1).

	<b>Acumulación en el primer efecto (M1)</b>
<b>Valor real</b>	2268
<b>Valor calculado</b>	2268,067
<b>Error porcentual</b>	-0.000029
<b>e (%)</b>	-0,0029

Elaborado por: Mena, Jerry. 2021.

El porcentaje de error obtenido como se muestra en la Tabla 5-4 es -0.0029%, que es menor que el error total permisible, por lo que los resultados obtenidos de la simulación a través de la interfaz desarrollada por modelado matemático en Simulink se considera aceptable.

## CONCLUSIONES

- Teniendo en consideración que el primer paso para un diseño correcto del sistema de control es alcanzar una comprensión profunda de las relaciones existentes entre las variables del sistema, se modeló matemáticamente el comportamiento del evaporador de doble efecto mediante ecuaciones de balance de masa global y de componentes para el primer y segundo efecto, así como los respectivos balances de energía para el primer y segundo efecto respectivamente, cabe señalar que dichos cálculos se encuentran expuestos en la sección 3.3 gracias a la versatilidad del software utilizado fue posible ingresar directamente las ecuaciones diferenciales de balance de masa y energía, dichas resoluciones fueron posibles gracias al solver que incluye el software utilizado.
- A través del uso de Simulink un producto de MATLAB se programó la simulación para el evaporador de doble efecto para ello se incluyó parámetros físicos detallados en el apartado 3.2.2, de la mano de la simulación se programó el sistema de control en concordancia con los objetivos de control planteados, de modo que para mantener la concentración de sólidos del producto en su valor deseado se debe controlar la variable  $X_p$  y para evitar desbordamiento o la desecación de los tubos del evaporador fue preciso controlar el nivel de la acumulación M1 y M2.
- Consecuentemente luego de haber construido la aplicación se realizó el manual de usuario y se procedió a ejecutar la aplicación externa en diferentes ordenadores los cuales no poseían el software usado para la creación de la interfaz.
- Finalmente se validó la simulación programada con el artículo “Simulation and Control of a Commercial Double Effect Evaporator: Tomato Juice” expuesto por Yadav & Jana (2010) cuyos resultados arrojan valores de consigna de: 0.2609 kg soluto/kg solución para la concentración del producto y 2268 kg para la acumulación en el primer efecto. Comparando dichos resultados con el presente trabajo se obtuvieron: 0.26067 kg soluto/kg, y 2268.067 kg respectivamente evidenciando un error relativo porcentual de 0.088% y -0.0029% respectivamente, siendo valores de error aceptables.

## RECOMENDACIONES

- Implementar un sistema de distribución previo a los evaporadores que me permita definir la presión que llega a la tubería para conocer la caída de presión existente.
- Extender el uso de simulink compiler como el producto de Matlab que proporciona una comunicación entre el modelo de simulink creado y una aplicación desarrollada en app designer.
- Evitar el uso de GUIDE para crear aplicaciones independientes que partan de un modelo de simulink ya que esta herramienta se encuentra en desuso e impide empaquetar la aplicación como aplicación independiente.
- Estudiar el uso de nuevos softwares de código abierto que admita crear simulaciones informáticas discretas.
- Colocar en el manual de usuario datos de algún otro producto de alimentación para que el estudiante pueda diversificar el uso de la herramienta.
- Establecer las funciones de transferencia y en base a ello elaborar un nuevo modelo de simulink de manera que me permita comparar la simulación de sistemas de control a partir de modelos mecánicos básicos y la simulación de sistemas lineales a través de funciones de transferencia.

## GLOSARIO

**Balances de materia:** La ley de conservación de la materia establece que la materia no se puede crear ni destruir. La conservación de la materia exige que los materiales que entran en un proceso, o bien se acumulan o salen del proceso, de forma que no puede haber pérdida ni ganancia (McCabe, W. L., Smith, J. C., Harriott, 1991, p.20).

**Balances de energía:** A un proceso, o a una parte de este, separado de los alrededores por un límite imaginario, se le puede aplicar un balance de energía. En un balance de energía es preciso incluir todas las formas de energía (McCabe, W. L., Smith, J. C., Harriott, 1991, p.20).

**Callbacks:** Se define como una función que se ejecuta en respuesta a determinadas operaciones predefinidas por el usuario, como hacer clic en un objeto gráfico o cerrar la ventana gráfica (Mathworks 2020a).

**Empaquetar:** Agrupar dentro de una extensión exe distintos archivos que se conjugan en una sola aplicación, dispuesta como un único archivo de instalación que puede ser compartido de manera fácil y sencilla con otros usuarios (Mathworks Inc. , 2009).

**Perturbaciones:** Son señales que tienden a afectar negativamente el valor de la salida de un sistema (Ogata, 2010, p.3).

**Señal de referencia (set point):** Es el valor en el cual se quiere mantener la variable controlada (valor deseado) (Dulhoste ,2013, p.11).

**Variable controlada:** Cantidad o condición que se va a medir y controlar. Por lo general, la variable controlada es considerada la señal de salida del sistema (Ogata, 2010, p.3).

## BIBLIOGRAFÍA

**AËŠSTRÖM, Karl J. y KUMAR, P.R.**, Control: A perspective. *Automatica*, 2014. vol. 50, no. 1, pp. 3-43.

**ASAMBLEA NACIONAL DEL ECUADOR**, 2008. Constitución de la República del Ecuador. *Registro Oficial*.

**ASAMBLEA NACIONAL DEL ECUADOR**, 2016. CÓDIGO ORGÁNICO DE LA ECONOMÍA SOCIAL DE LOS CONOCIMIENTOS, CREATIVIDAD E INNOVACIÓN. .

**ATTAWAY, Stormy**, 2019. Introduction to MATLAB. *MATLAB*. S.l.: s.n., pp. 3-35. ISBN 9780128154793.

**BEN-ALI, Samia**, Modeling of a double effect evaporator : Bond graph. *Chemical Engineering Research and Design* [en línea], 2018. pp. 1-14. [Consulta: 12 agosto 2020]. Disponible en: <https://doi.org/10.1016/j.cherd.2018.07.007>.

**BORRERO, Maria**, *Herramienta Software para el Control Remoto de una Fuente de Alimentación mediante una Interfaz Gráfica (Trabajo de titulación) (Pregrado)* [en línea]. S.l.: Universidad de Sevilla. 2011. [Consulta: 3 marzo 2021]. Disponible en: <http://bibing.us.es/proyectos/abreproy/11986/fichero/CAPÍTULO+3%252FCAPÍTULO+3.pdf>.

**COUPER, James R., PENNEY, Roy, FAIR, James R. y WALAS, Stanley M.**, *Chemical Process Equipment Selection and Design*. S.l.: 2010. s.n. ISBN 9780123969590.

**DE LA FUENTE, Jesus**, 1999. Sintonía de Reguladores PID. *Universidad de Valladolid* [en línea]. [Consulta: 24 abril 2021]. Disponible en: <http://www.isa.cie.uva.es/~maria/pids.pdf>.

**DULHOSTE, Jean Francois**, 2013. Introducción al Control de Procesos para Ingenieros. [en línea]. [Consulta: 20 julio 2021]. Disponible en: [http://webdelprofesor.ula.ve/ingenieria/djean/index\\_archivos/Documentos/Teoria\\_Control.pdf](http://webdelprofesor.ula.ve/ingenieria/djean/index_archivos/Documentos/Teoria_Control.pdf).

**DUNN, Stanley M., CONSTANTINIDES, Alkis y MOGHE, Prabhas V.**, 2006. Numerical Methods in Biomedical Engineering Appendix B : Introduction to Simulink. *Library* [en línea]. S.l.: s.n., pp. 487-516. [Consulta: 2 marzo 2021]. ISBN 9780121860318. Disponible en: <https://www.sciencedirect.com/science/article/pii/B9780121860318500121>.

**ESPINOSA, Guillermo Cid**, *Programación de interfaz gráfica en app designer para el control vectorial de motores de imanes permanentes* [en línea]. S.l.: UNIVERSIDAD POLITÉCNICA DE MADRID. 2018. [Consulta: 20 abril 2021]. Disponible en: [http://oa.upm.es/53343/1/TFG\\_GUILLERMO\\_CID\\_ESPINOSA.pdf](http://oa.upm.es/53343/1/TFG_GUILLERMO_CID_ESPINOSA.pdf).

**GARCÍA MORILLA, Fernando**, 2006. CONTROLADORES PID (AJUSTE EMPÍRICO). [en línea]. S.l.: [Consulta: 24 abril 2021]. Disponible en: [http://www.dia.uned.es/~fmorilla/MaterialDidactico/ajuste\\_empirico.pdf](http://www.dia.uned.es/~fmorilla/MaterialDidactico/ajuste_empirico.pdf).

**HACKETT, Bryan W.**, The essentials of continuous evaporation. *Chemical Engineering Progress*, 2018. vol. 114, no. 5, pp. 24-28.

**IBRAHIM, Dogan**, Engineering simulation with MATLAB: Improving teaching and learning effectiveness. *Procedia Computer Science* [en línea], 2011. vol. 3, pp. 853-858. [Consulta: 2 febrero 2021]. Disponible en: <http://dx.doi.org/10.1016/j.procs.2010.12.140>.

**JIANG, Hua, ZHANG, Ziyao y GONG, Wuqi**, Design and evaluation of a parallel-connected double-effect mechanical vapor recompression evaporation crystallization system. *Applied Thermal Engineering*, 2020. vol. 179.

**KOMULAINEN, Tiina M., ENEMARK-RASMUSSEN, Rasmus, SIN, Gürkan, FLETCHER, John P. y CAMERON, David**, Experiences on dynamic simulation software in chemical engineering education. *Education for Chemical Engineers* [en línea], 2012. vol. 7, no. 4, pp. e153-e162. [Consulta: 26 enero 2021]. Disponible en: <http://dx.doi.org/10.1016/j.ece.2012.07.003>.

**KUMAR, Prabir**, *PROCESS DYNAMICS AND CONTROL*. Delhi: 2014. Asoke K. Ghosh, PHI Learning Private Limited, Rimjhim House, 111, Patparganj Industrial Estate, Delhi-110092. ISBN 9788120348462.

**KUO, Benjamin**, *Sistemas de Control Automático (G. Aranda, Trad.)*. México: 1996. Prentice-Hall Hispanoamericana, S.A. ISBN 968-880-723-0.

**LUYBEN, William L.**, Chemical Engineering & Processing : Process Intensification Dynamic simulation of multi-effect evaporators. , 2018. vol. 131, no. April, pp. 106-115.

**MATHWORKS**, 2009a. Empaquetar aplicaciones en App Designer. [en línea]. [Consulta: 20

abril 2021]. Disponible en: [https://la.mathworks.com/help/matlab/creating\\_guis/package-apps-in-app-designer.html](https://la.mathworks.com/help/matlab/creating_guis/package-apps-in-app-designer.html).

**MATHWORKS**, 2009b. MATLAB Compiler. [en línea]. [Consulta: 20 abril 2021]. Disponible en: <https://la.mathworks.com/products/compiler.html>.

**MATHWORKS**, 2020a. Callbacks: respuesta programada a una acción del usuario. [en línea]. [Consulta: 20 abril 2021]. Disponible en: [https://la.mathworks.com/help/matlab/creating\\_plots/callbacks-programmed-response-to-user-action.html](https://la.mathworks.com/help/matlab/creating_plots/callbacks-programmed-response-to-user-action.html).

**MATHWORKS**, 2020b. Simulink Compiler. [en línea]. [Consulta: 20 abril 2021]. Disponible en: <https://la.mathworks.com/products/simulink-compiler.html>.

**MATHWORKS INC.**, 2020. Desarrollar apps mediante App Designer. *Mathworks* [en línea]. [Consulta: 19 abril 2021]. Disponible en: [https://la.mathworks.com/help/matlab/app-designer.html?searchHighlight=app designer&s\\_tid=srchtitle](https://la.mathworks.com/help/matlab/app-designer.html?searchHighlight=app%20designer&s_tid=srchtitle).

**MCCABE, W. L., SMITH, J. C., HARRIOTT, P.**, *Operaciones unitarias en ingeniería química*. S.l.: 1991. s.n. ISBN 84-481-1918-5.

**MIRANDA, V. y SIMPSON, R.**, Modelling and simulation of an industrial multiple effect evaporator : tomato concentrate. , 2005. vol. 66, pp. 203-210.

**MOODLEY, Kuvenshan**, Improvement of the learning and assessment of the practical component of a Process Dynamics and Control course for fourth year chemical engineering students. *Education for Chemical Engineers* [en línea], 2020. vol. 31, pp. 1-10. [Consulta: 1 octubre 2020]. Disponible en: <https://doi.org/10.1016/j.ece.2020.02.002>.

**MULERO, A., CACHADIÑA, I. y CUADROS, F.**, Validez y Precisión de Correlaciones Empíricas Sencillas para el Cálculo de Entalpías de Exceso. *Información tecnológica* [en línea], 2004. vol. 15, no. 4. Disponible en: [http://www.scielo.cl/scielo.php?script=sci\\_arttext&pid=S0718-07642004000400014&lng=en&nrm=iso&tlng=en](http://www.scielo.cl/scielo.php?script=sci_arttext&pid=S0718-07642004000400014&lng=en&nrm=iso&tlng=en).

**NUNIS DA SILVA, André Luiz, SECKLER, Marcelo Martins y ODLOAK, Darci**, Application of Model Predictive Control to a Continuous Multiple-Effect Crystallizer. *Chemical*

*Engineering and Technology*, 2018. vol. 41, no. 7, pp. 1406-1416.

**OGATA, Katsuhiko**, *Ingeniería de control moderna*. Madrid: 2010. Prentice-Hall. ISBN 9788483226605.

**PEREZ, Mario Alberto, PEREZ HIDALGO, Analía y PEREZ BERENGUER, Elisa**, *Introducción a los sistemas de control y modelo matemático para sistemas lineales invariantes en el tiempo*. S.l.: Universidad Nacional de San Juan. 2008.

**ROCA, Alfredo**, *Control automático de procesos industriales*. España: 2014. s.n. ISBN 9788499698748.

**RUNYON, C.H., RUMSEY, T.R. y MCCARTHY, K.L.**, Dynamic simulation of a nonlinear model of a double effect evaporator. *Journal of Food Engineering*, 1991. vol. 14, no. 3, pp. 185-201.

**SMITH, Carlos y CORRIPIO, Armando**, *Control automático de procesos*. S.l.: 2014. s.n. ISBN 9786070507212.

**SOUSA, Felipe Matheus Mota y FONSECA, Rodolpho Rodrigues**, Application of Adaptive Feedforward-Feedback Control on Multiple Effect Evaporator Process. *Chemical Product and Process Modeling*, 2019. vol. 14, no. 2, pp. 1-13.

**SUCRE-ROSALES, Estefanía, FERNÁNDEZ-TERÁN, Ricardo, CARVAJAL, David, ECHEVARRÍA, Lorenzo y HERNÁNDEZ, Florencio E.**, Experience-Based Learning Approach to Chemical Kinetics: Learning from the COVID-19 Pandemic. *Journal of Chemical Education*, 2020. vol. 97, no. 9, pp. 2598-2605.

**VERMA, Om Prakash, MANIK, Gaurav y MOHAMMED, Toufiq Haji**, Energy management in multi stage evaporator through a steady and dynamic state analysis. *Korean Journal of Chemical Engineering*, 2017. vol. 34, no. 10, pp. 2570-2583.

**VERMA, Om Prakash, MOHAMMED, Toufiq Hazi, MANGAL, Shubham y MANIK, Gaurav**, Modeling, simulation and control of the dynamics of a Heptads' effect evaporator system used in the Kraft recovery processes. *Transactions of the Institute of Measurement and Control*, 2018. vol. 40, no. 7, pp. 2278-2290.

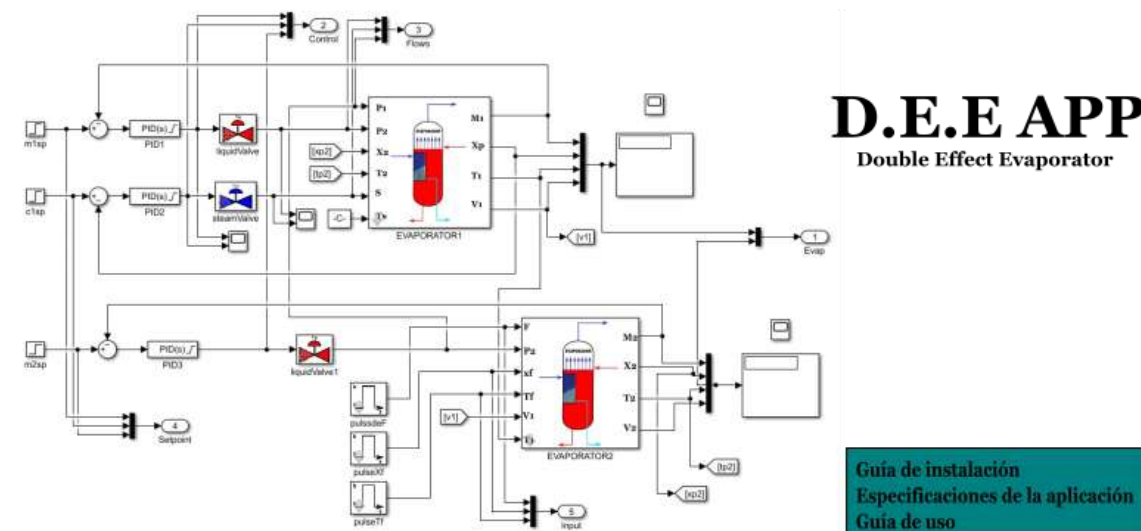


**YADAV, Praveen y JANA, Amiya K.**, Simulation and control of a commercial double effect evaporator: Tomato juice. *Chemical Product and Process Modeling*, 2010. vol. 5, no. 1.

## ANEXOS

### ANEXO A. MANUAL DE USUARIO DE LA APLICACIÓN DEL SISTEMA DE CONTROL DE UN EVAPORADOR DE DOBLE EFECTO

# MANUAL DE USUARIO



ESPOCH - Escuela de Ingeniería Química

Autor: Jerry Leodan Mena Nagua

Email: jerry2800@hotmail.com




Director del trabajo de integración curricular: Ing. Darío Fernando Guaman Lozada

Miembro del trabajo de integración curricular: Ing. Adrián Alejandro Rodríguez Pinos



## Información preliminar para el uso de la aplicación

D.E.E APP (Double Effect Evaporator) se concibe como una aplicación formato .exe que permite entender cómo funciona un sistema de control sobre una variable controlada, las perturbaciones que afectan a todo el proceso se pueden ingresar como entrada escalón y entrada pulso, dicho sistema de control se efectúa sobre un evaporador de doble efecto en contracorriente, las condiciones iniciales de los evaporadores se pueden ingresar como parámetros editables y los parámetros de configuración del controlador tales como constantes de controladores, aperturas de la válvulas, set points o puntos de consigna y perturbaciones en forma de escalón y pulso son valores que también se editan, si así se requiere.

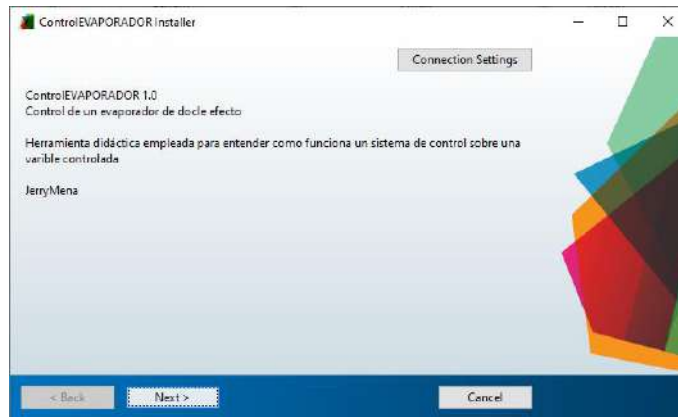
<b>Antes de empezar a leer el Manual NECESITAS SABER ESTO</b>	<b>Íconos para considerar dentro del documento</b>
	<b>Para tener en cuenta !</b>  Hace hincapié en información de reelevancia para el usuario.
	<b>Advertencia</b>  Situaciones que se deben evitar a toda costa por el usuario.
	<b>Información relevante</b>  Notas informativas de la aplicación generada.

## Guía de instalación de la aplicación

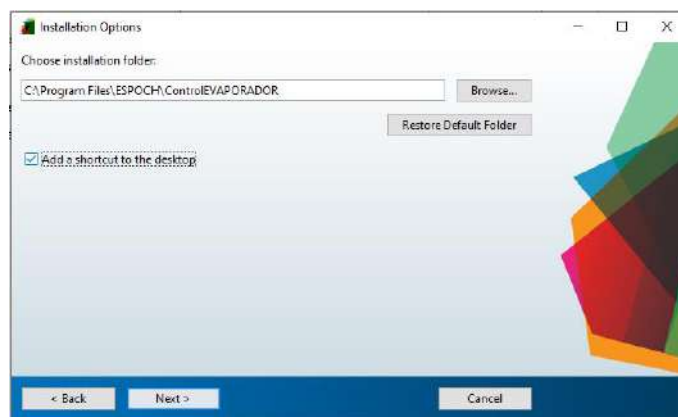
Se recomienda contar como mínimo con un ordenador de 1 GB de espacio libre, procesador Intel® Core™ i3-4030U, con 4 GB de memoria RAM.

Para instalar la aplicación se siguen los siguientes pasos:

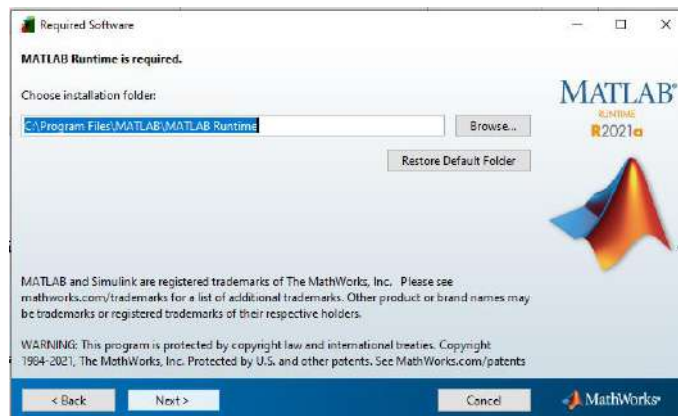
- En la carpeta For\_redistribucion se va a encontrar con un instalador el cual contiene un runtime indispensable para que se ejecute la aplicación, para empezar la ejecución se debe hacer doble clic sobre el icono, es posible que aparezca una ventana pidiendo permisos de administrador para realizar cambios en el equipo se le debe conceder.
- Una vez realizado el paso anterior aparecerá la siguiente ventana y se debe dar clic en Next, dicha ventana en algunos casos suele tardar en aparecer.



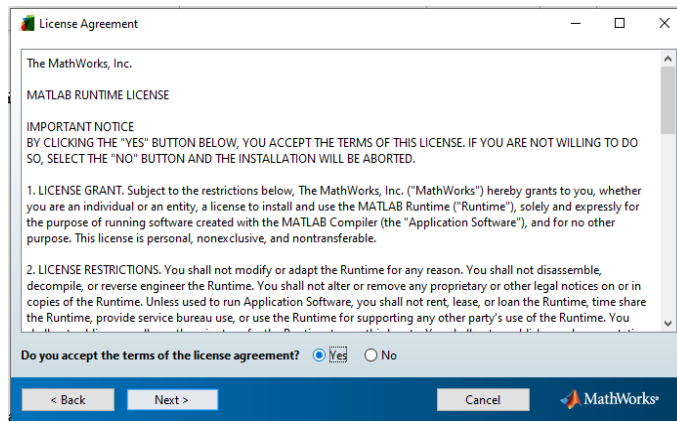
- En la siguiente ventana se debe marcar la opción de agregar un icono en el escritorio y darle clic en Next



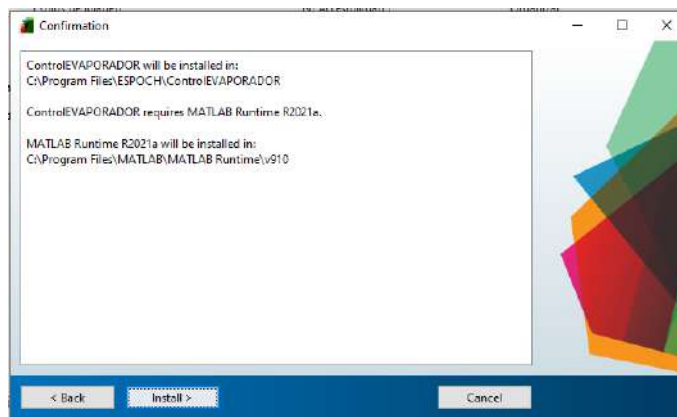
- La siguiente ventana me informa del runtime que se va a instalar juntamente con la aplicación, se debe dar clic en Next



- La ventana que aparece a continuación es el acuerdo de aceptación de términos y condiciones se debe marcar la opción Yes y darle clic en Next



- La siguiente ventana muestra los programas que van a ser instalados se debe dar clic en install



- La instalación de la aplicación puede tomar algún tiempo, aproximadamente 15 minutos, dicho tiempo puede variar según el ordenador, para finalizar la instalación de la aplicación se debe dar clic en el botón finish



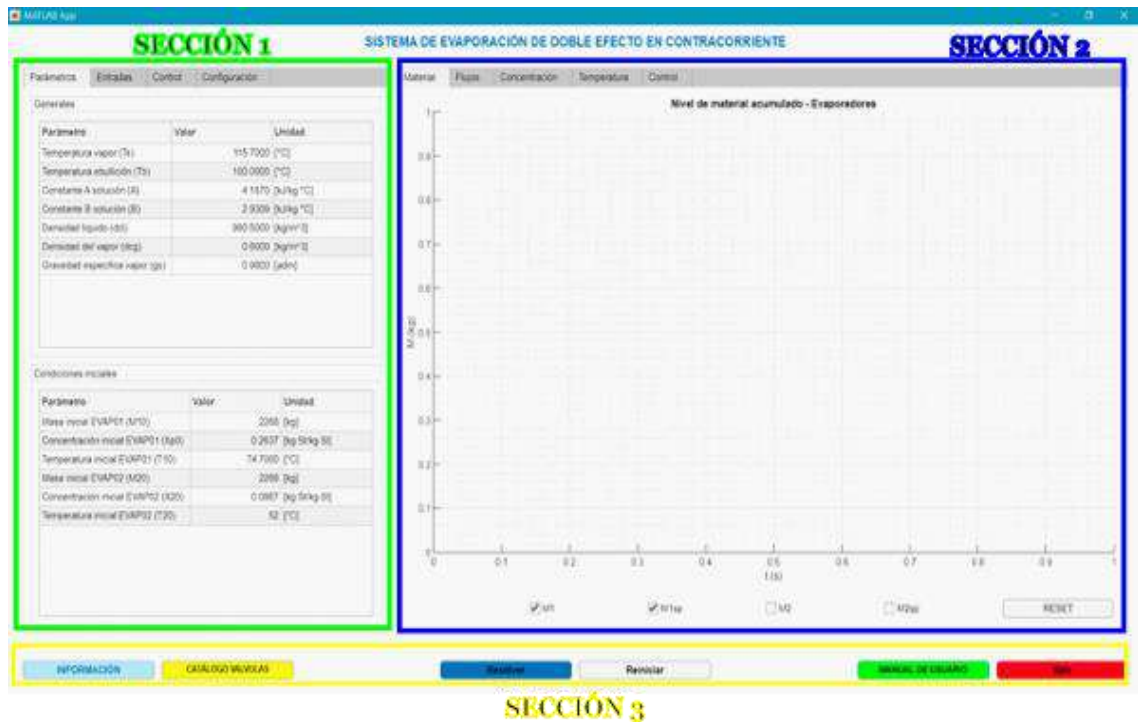
- Una vez culminada la instalación aparecerá un icono en el escritorio, al hacer doble clic se puede acceder a la aplicación, la inicialización toma algunos minutos.

## Especificaciones de la aplicación

1. Al hacer doble clic sobre el ícono de la aplicación se desplegará una ventana de inicio con información de la aplicación y del autor, para pasar a la siguiente ventana se debe hacer clic en el botón comenzar.



2. La siguiente ventana contiene el cálculo de la aplicación, pasado el punto uno dicha ventana suele aparecer luego de un tiempo, esto dependerá de la capacidad de procesamiento del ordenador, la ventana consta de dos secciones en las cuales existen paneles los cuales de lado izquierdo contienen espacios para parámetros modificables y al lado derecho contiene gráficas las cuales se grafican de acuerdo a los estados activados o desactivados de los checkbox, es preciso destacar una tercera sección la cual está compuesta de diversos botones:



**Sección 1:** Aparecen valores por defecto los cuales deben ser borrados para colocar los valores de entrada o set point deseados. Existen cuatro apartados dentro de esta sección:

- **Parámetros.** – Dentro de este apartado es posible editar distintos parámetros dentro de dos diferentes paneles, en el primer panel se puede editar:

Ts: Temperatura de vapor de entrada al proceso (C).

Tb: Temperatura de ebullición la cual corresponderá a la etapa de evaporación (C).

Constantes A y B correspondientes para el presente caso por defecto a una solución de pasta de tomate.

dcl – dcg: Densidad del líquido y vapor para el presente caso por defecto agua y vapor de agua.

Gs: gravedad específica de vapor.

Mientras que en el segundo panel se pueden editar:

M10 – M20: la masa inicial tanto para el evaporador 1 y 2

Xp0 – X20: la concentración del producto Xp y X2

T10 – T20: las temperaturas T1 y T2 correspondientes al evaporador 1 y 2 respectivamente.

Parámetros			
Entradas			
Control			
Configuración			
<b>Generales</b>			
Parámetro	Valor	Unidad	
Temperatura vapor (Ts)	115.7000	[°C]	
Temperatura ebullición (Tb)	100.0000	[°C]	
Constante A solución (A)	4.1870	[kJ/kg °C]	
Constante B solución (B)	2.9309	[kJ/kg °C]	
Densidad líquido (dcl)	980.5000	[kg/m³]	
Densidad del vapor (dcg)	0.6000	[kg/m³]	
Gravedad específica vapor (gs)	0.9800	[adn]	
<b>Condiciones iniciales</b>			
Parámetro	Valor	Unidad	
Masa inicial EVAP01 (M10)	2268	[kg]	
Concentración inicial EVAP01 (Xp0)	0.2637	[kg St/kg]	
Temperatura inicial EVAP01 (T10)	74.7000	[°C]	
Masa inicial EVAP02 (M20)	2268	[kg]	
Concentración inicial EVAP02 (X20)	0.0867	[kg St/kg]	
Temperatura inicial EVAP02 (T20)	52	[°C]	



**Para tener en cuenta !**

Las condiciones iniciales de operación deben corresponder a los valores resultantes de un modelo estacionario reportado en bibliografía

- **Entrada.** – Dentro de este panel se ingresarán las perturbaciones que afectarán al sistema se debe editar el límite inferior, periodo, amplitud, ancho y retraso.

Flujo de entrada F; Composición de entrada Xf; Temperatura Tf

Parámetro	Valor	Unidad
Limite inferior F (Fa)	7.2450	[kg/s]
Periodo (p)	3600	[s]
Amplitud (amp)	0	[kg/s]
Ancho de pulso (w)	50	[%]
Retrazo (d)	0	[s]

Parámetro	Valor	Unidad
Limite inferior xF (xFa)	0.0500	[kg solut/kg sol]
Periodo (p)	3600	[s]
Amplitud (amp)	0	[kg solut/kg sol]
Ancho de pulso (w)	50	[%]
Retrazo (d)	0	[s]

Parámetro	Valor	Unidad
Limite inferior TF (TFa)	85	[°C]
Periodo (p)	3600	[s]
Amplitud (amp)	0	[°C]
Ancho de pulso (w)	50	[%]
Retrazo (d)	0	[s]



### Para tener en cuenta !

El limite inferior debe corresponder al valor inicial del salto, la amplitud es el rango de crecimiento del salto, el periodo es el tiempo total que dura el salto en el impulso, el ancho se establecera mediante una regla de tres simple para lo cual el valor colocado en el periodo correspondera al 100% y la resta del punto en que la perturbacion vuelve al valor inicial menos el valor del retraso resolviendo dicha incognita saldria el porcentaje que debiera ser colocado en este inciso y para el retraso es preciso colocar el tiempo requerido para realizar el primer cambio

- Control. – Existen tres subpartados el primero denominado controladores en el cual es posible editar factores como constante  $k_p$ ,  $k_i$ ,  $k_d$  y  $N$ , el segundo subpartado contiene elementos tales como tiempo de salto, valor inicial y valor final y el tercer subpartado consta de parámetros relacionados a la válvula tales como coeficiente de válvula y caída de presión.

Parámetros	Entradas	Control	Configuración
Controladores	Setpoints	Válvulas	

**Subpartado 1**

Parámetro	Valor	Unidad
Constante $K_p$	-1	[adm]
Constante $K_i$	0	[adm]
Constante $K_d$	0	[adm]

**Subpartado 2**

Parámetro	Valor	Unidad
Tiempo de salto (t)	0	[s]
Valor inicial (vi)	2266	[kg]
Valor final (vf)	2266	[kg]

**Subpartado 3**

Parámetro	Valor	Unidad
Coefficiente de válvula (Kv)	53.0050	[m <sup>3</sup> /h bar <sup>0.5</sup> ]
Caída de presión (ΔP)	7.0130	[bar]



### Para tener en cuenta !

Dentro del subpartado 2 si se requiere que los valores del punto de consigna por alguna razón se mantengan constantes se debe mantener el valor inicial y final como el mismo y colocar el tiempo de salto igual a cero.

En relación al subpartado 3 para ingresar valores de apertura se debe necesariamente hacer un clic sobre el botón denominado catálogo y realizar el cambio por los Kv establecidos en el catálogo.

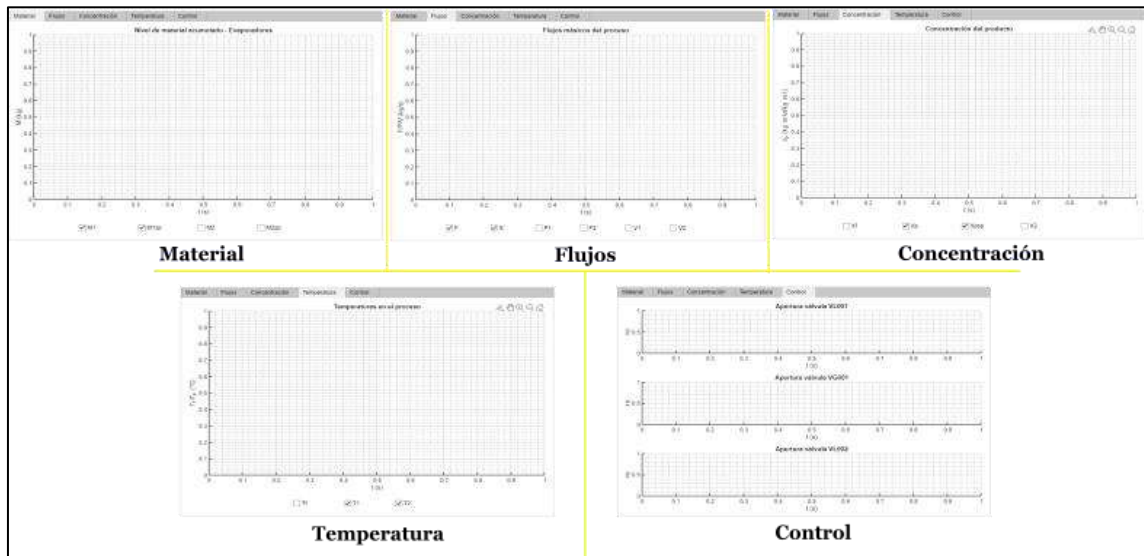


- Configuración. – Engloba las características editables referentes al tiempo y a la cantidad de pasos necesarios para resolver la simulación.

Parámetros	Entradas	Control	Configuración
Panel			
Parámetro	Valor	Unidad	
Tiempo inicial (t0)	0	[s]	
Tiempo final (tf)	90000	[s]	
Salto de tiempo (st)	0.3000	[s]	

**Sección 2:** Consta de apartados en los cuales existen gráficas las cuales se activan al activar el estado de los checkbox, las variables que se pueden dibujar son:

- Material. – M1, M1sp, M2, M2sp
- Flujos. – F, S, P1, P2, V1, V2
- Concentración. – Xf, Xp, Xpsp, X2
- Temperatura. – Tf, T1, T2
- Control. – Apertura de la válvula VL001, VG001 y VL002.

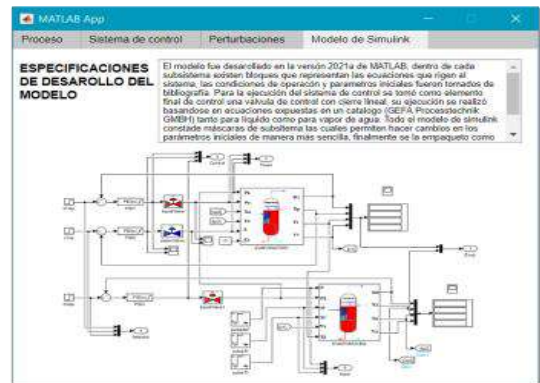
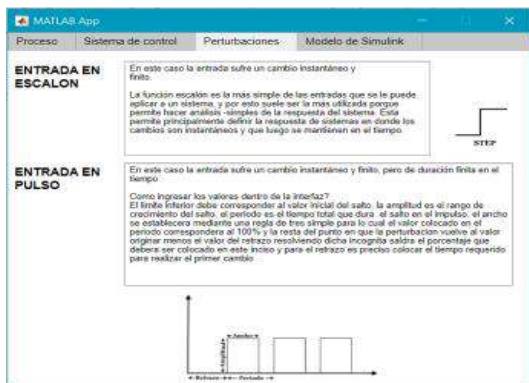
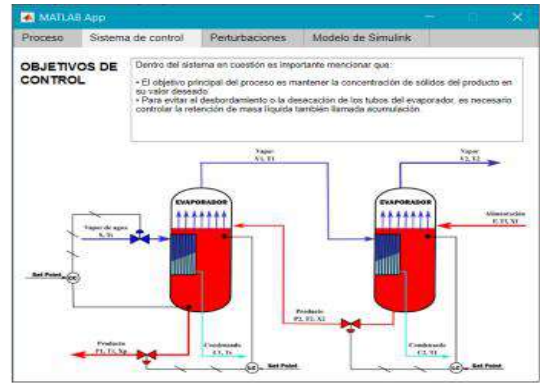
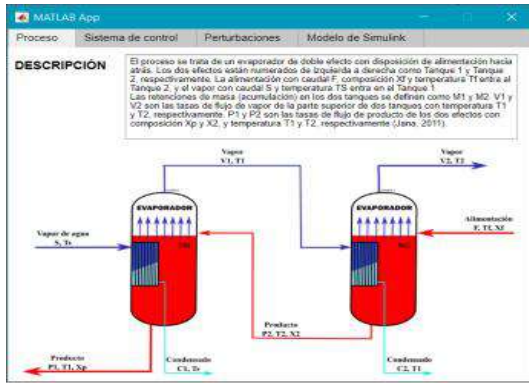


**Para tener en cuenta !**

Los resultados de cada una de las gráficas se activan de acuerdo a los estados de los checkbox estos determinarán que se muestra en las gráficas.

**Sección 3:** Denominada la sección de acciones ya que al dar clic sobre cualquiera de los botones que se encuentran en esta sección se ejecuta una acción.

En esta sección se encuentra el botón de información referente al proceso, al dar clic se desplegará una ventana que consta de varias pestañas en las que se encuentra información referente al proceso, a los objetivos de control, a las perturbaciones y al modelo de simulink.



También se encuentra el botón resolver el cual al presionarlo dará paso a los resultados de la simulación

Resolver

**Para tener en cuenta !**

Antes de dar clic en el botón resolver se recomienda pulsar el botón de información; cuando la simulación se encuentra en curso el botón cambiará a “simulando”.

**Advertencia**

Se debe dar clic una única vez al botón resolver, debido a que más de un clic en el botón producirá un error.

El botón reiniciar dará paso a los parámetros cargados en el modelo por defecto, el catálogo de válvulas y el manual de usuario darán paso a un pdf que contiene su información y finalmente se encuentra el botón salir el cual me posibilita salir de la aplicación.

CATÁLOGO VALVULAS    MANUAL DE USUARIO    Reiniciar    Salir

**Información relevante**

Es importante mencionar que los valores que aparecen por defecto son valores tomados de bibliografía, al empezar a cambiar valores en la sección se debe necesariamente ajustar los valores de los parámetros de control de la sección 5 para alcanzar la mínima diferencia entre el valor real y el valor ingresado (set-point).

## Guía de uso de la ventana principal de la aplicación

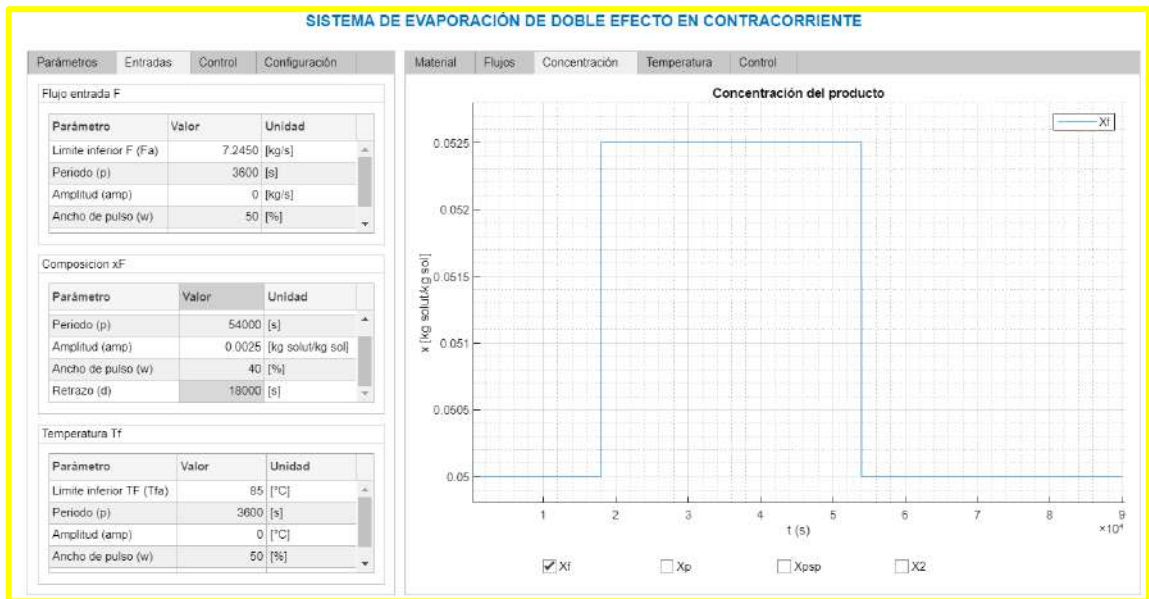
Para demostrar cómo se debe manipular la aplicación, se ha cargado un modelo por defecto el cual reporta los valores necesarios para las condiciones iniciales y parámetros generales y tiempo de simulación (25h - 90000s).

**Análisis 1:** Efectuar un efecto de un cambio de entrada de pulso en la concentración inicial del alimento (cambio de 0.05 a 0.0525 kg St / kg SI al tiempo = 5 horas (18000 s) y luego de 0.0525 a 0.05 kg St / kg SI en el tiempo = 15 horas (54000 s)).

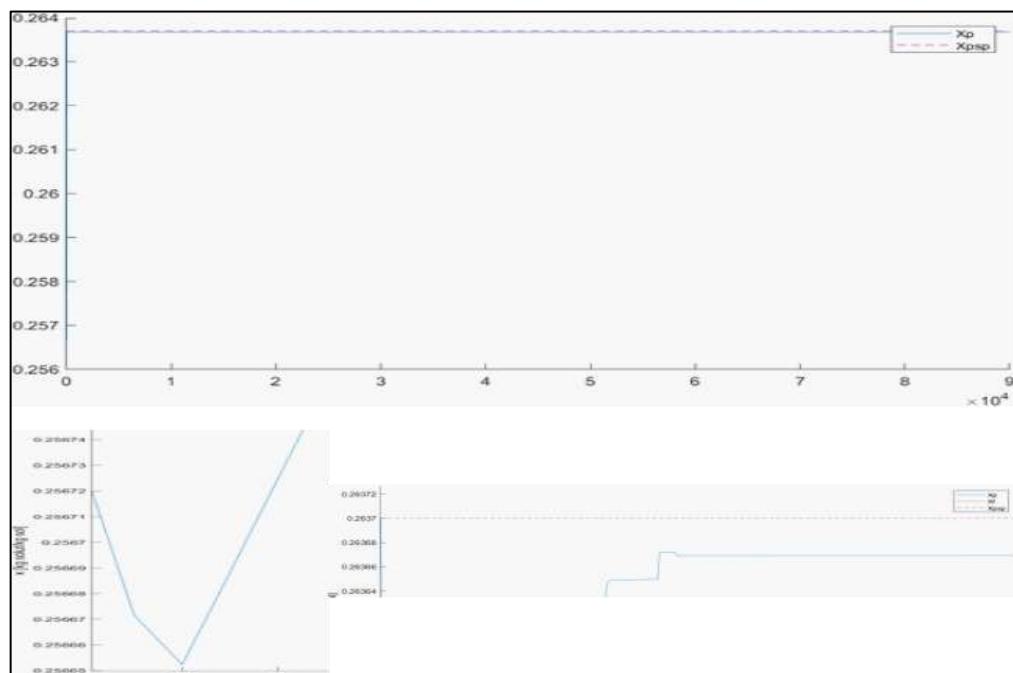
Para ello se debe seguir la siguiente metodología:

- Dar clic en la ventana de información y saber cómo funciona el proceso:  
Identificar variables de entrada y de salida.  
Conocer cuáles son los objetivos de control  
Conocer como ingresar las perturbaciones al sistema
- Ubicarse en la sección 1, dentro del panel entradas y localizar la concentración inicial del alimento (xF)
- Siguiendo las indicaciones de como ingresar el pulso modificar los parámetros de modo que:  
El límite inferior corresponde a 0.05 según el enunciado expuesto en análisis 1.  
El retraso según el enunciado es 18000s ya que corresponde al tiempo requerido para realizar el cambio.  
La amplitud es el rango de crecimiento del salto, es decir, cómo el enunciado nos dice que existirá un cambio de 0.05 a 0.0525 se debe ingresar un valor de 0.0025 dentro de este apartado.  
El periodo es el tiempo total que dura el salto en el impulso, en este caso será 54000 ya que el nuevo cambio según el enunciado se hará a ese tiempo.  
El ancho del pulso se establecerá mediante una regla de tres simple para lo cual el valor equivalente al tiempo que dura toda la simulación corresponderá al 100% mientras que el valor colocado en el periodo menos el valor del retraso determinará el valor del porcentaje que deberá ser colocado en este inciso  
Periodo – Retraso: (54000 – 18000 = 36000) segundos  
Regla de tres:  $\frac{36000 \times 100\%}{90000} = 40\%$  **ANCHO DEL PULSO**
- Una vez modificados esos valores se presiona el botón resolver y se procede a analizar las gráficas

Entrada de la concentración inicial en forma de pulso provocando que las variables inmiscuidas dentro del proceso presenten cambios.



A continuación, se presenta la primera variable controlada la cual corresponde a la concentración del producto  $X_p$  para su control se debe manipular a la variable  $S$  (cantidad de vapor de agua) su funcionamiento radica en que existe una válvula que regula el paso de la cantidad de vapor y se va abriendo o cerrando hasta que la señal de retroalimentación alcance o se asemeje al punto de consigna (el grado de apertura viene dado por el controlador), la gráfica resultante refleja picos de descenso al segundo 12.8 reportándose un valor de 0.25665, a partir de ese punto el valor crece hasta 0.2635 al segundo 30.5 dicho valor se mantiene constante hasta el segundo 38.1 cuando cambia a un valor de 0.26367 y al pasar al segundo 40.31 el proceso se estabiliza hasta un valor de 0.263669 hasta los 18000 segundos donde el valor sube hasta 0.2636701 y en el valor de 54000 segundos vuelve a 0.263669 manteniéndose constante denotando un mínimo error equivalente a 0.000031.

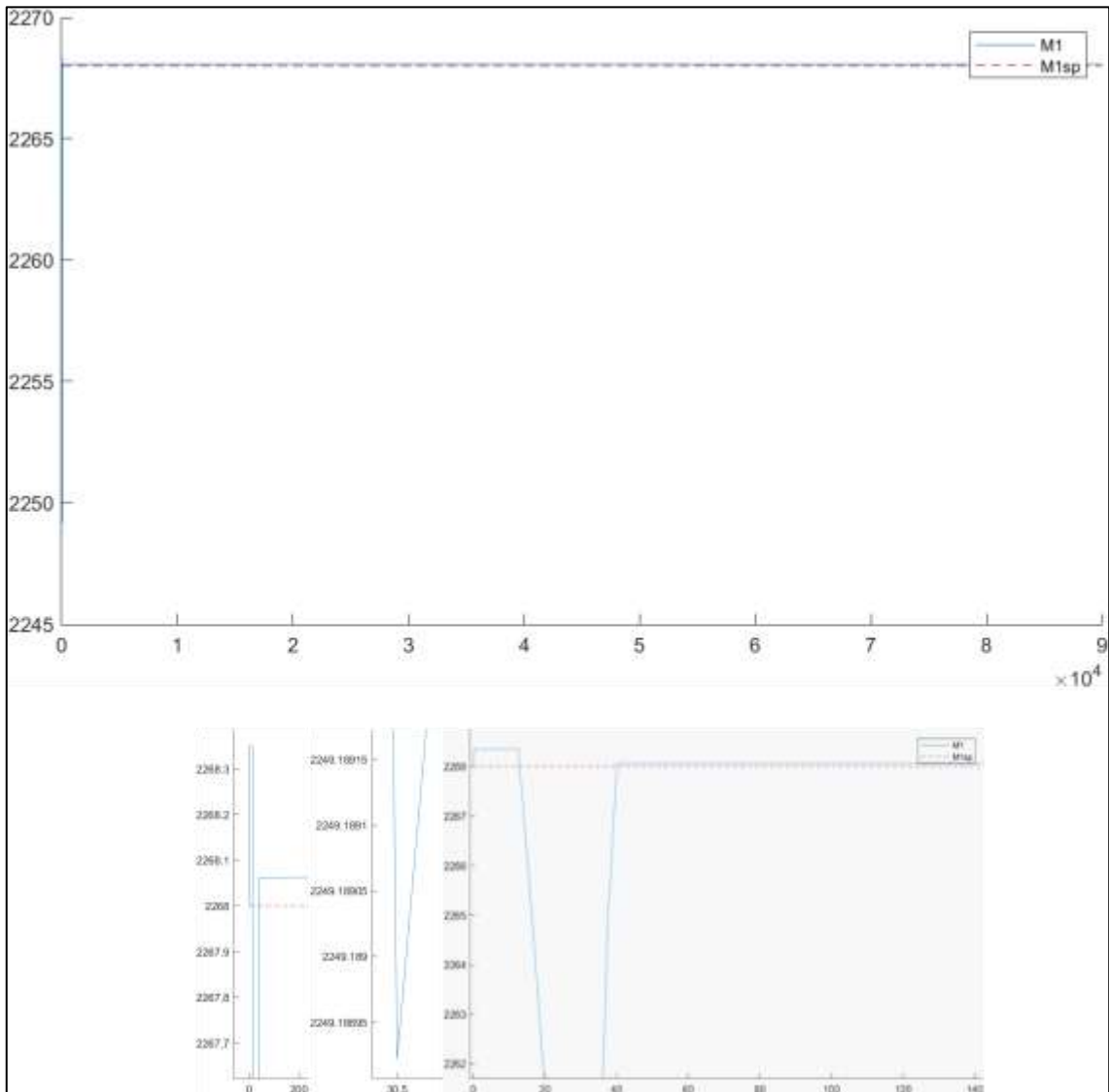


APERTURA	Apertura válvula VL001		Segund	Valor
			0.5	0.351
			12.9	0
			40.2	0.0624
			40.5	0.0605
			40.8	0.06118
			1000	0.0665
			18000	0.0695
			54000	0.0665

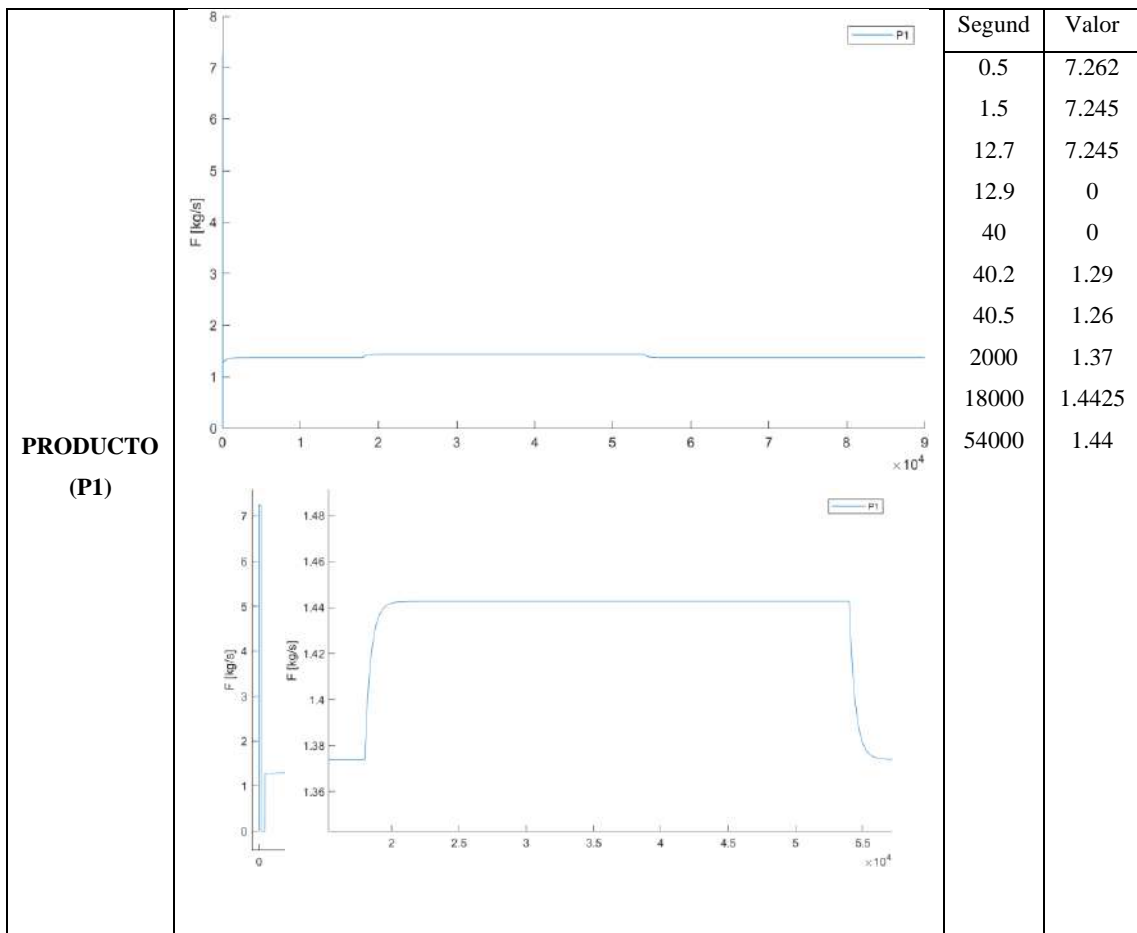
VAPOR (S)	Flujos máscos del proceso		Segund	Valor
			30.5	7.526
			38.1	2.106
			38.4	2.115
			40	2.1161
			40.3	2.3225
			40.4	2.3249
			40.7	2.3133
			3000	2.2729
			18000	2.245
		54000	2.2729	

La segunda variable controlada la cual corresponde a la acumulación del producto M1 para su control se debe manipular a la variable P1 (cantidad producto de salida del efecto 1) su funcionamiento radica en que existe una válvula que regula el paso de la cantidad de producto y se va abriendo o cerrando hasta que la señal de retroalimentación alcance o se asemeje al punto de consigna (el grado de apertura viene dado por el controlador), la gráfica resultante refleja un aumento en el material acumulado en 2268.35 en 0.4 segundos se mantiene constante hasta el segundo 12.7, se produce un descenso hasta el segundo 30.5 llegando alcanzar un valor de 2249.1889, inmediatamente se produce un ascenso hasta el segundo 40.3 tomando un valor de 2268.0625 luego desciende hasta 2268.0605 en el segundo 40.5 alcanza un valor de 2268.0625 y

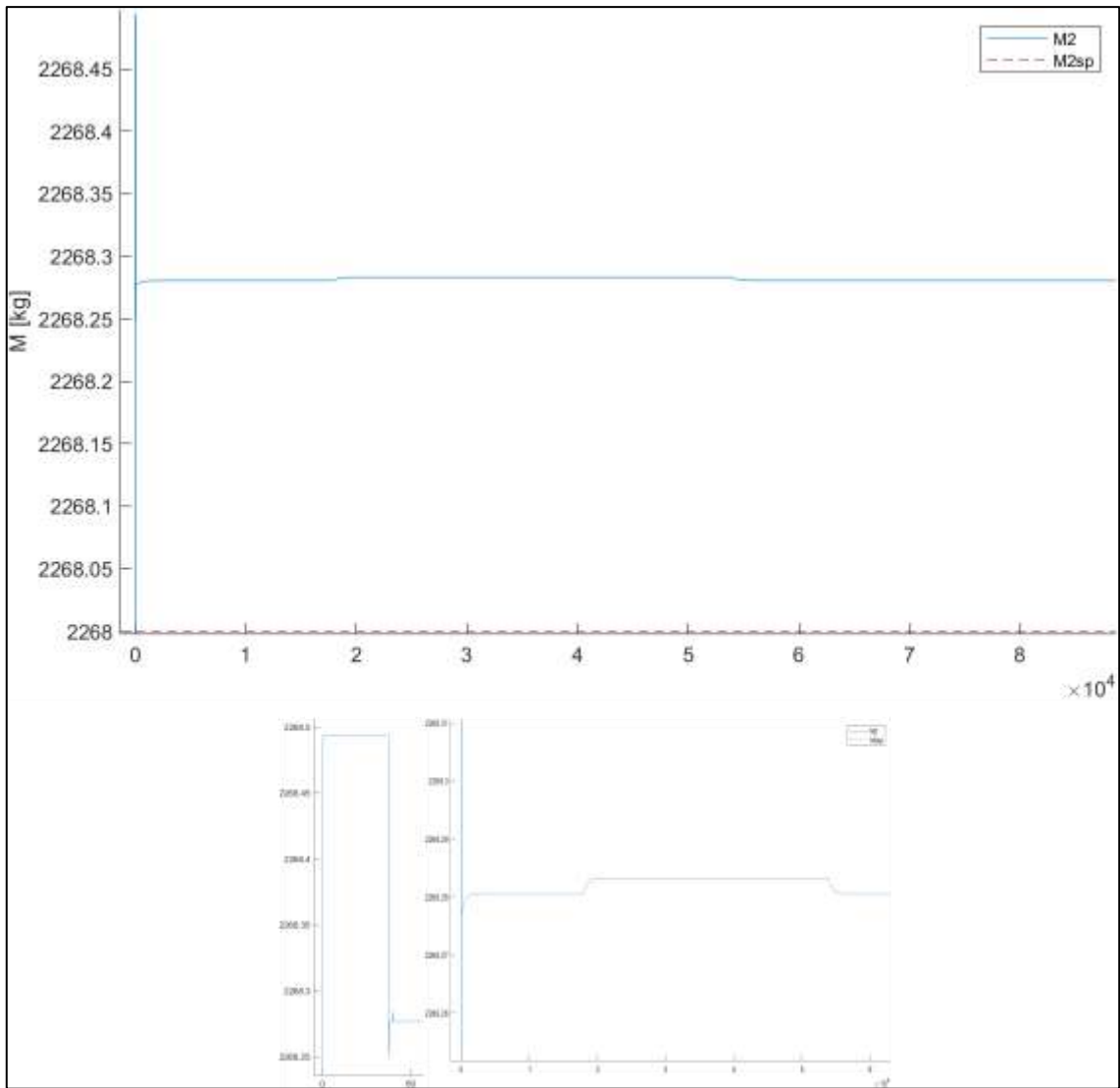
al segundo 18000 toma una valor de 2268.069 hasta el segundo 54000 donde el valor equivalente a 2268.066 se mantiene constante durante todo el proceso.



APERTURA		Segund	Valor
		0.3	1
30.4	1		
38	0.28		
40	0.28		
1800	0.302		
18000	0.2985		
54000	0.302		

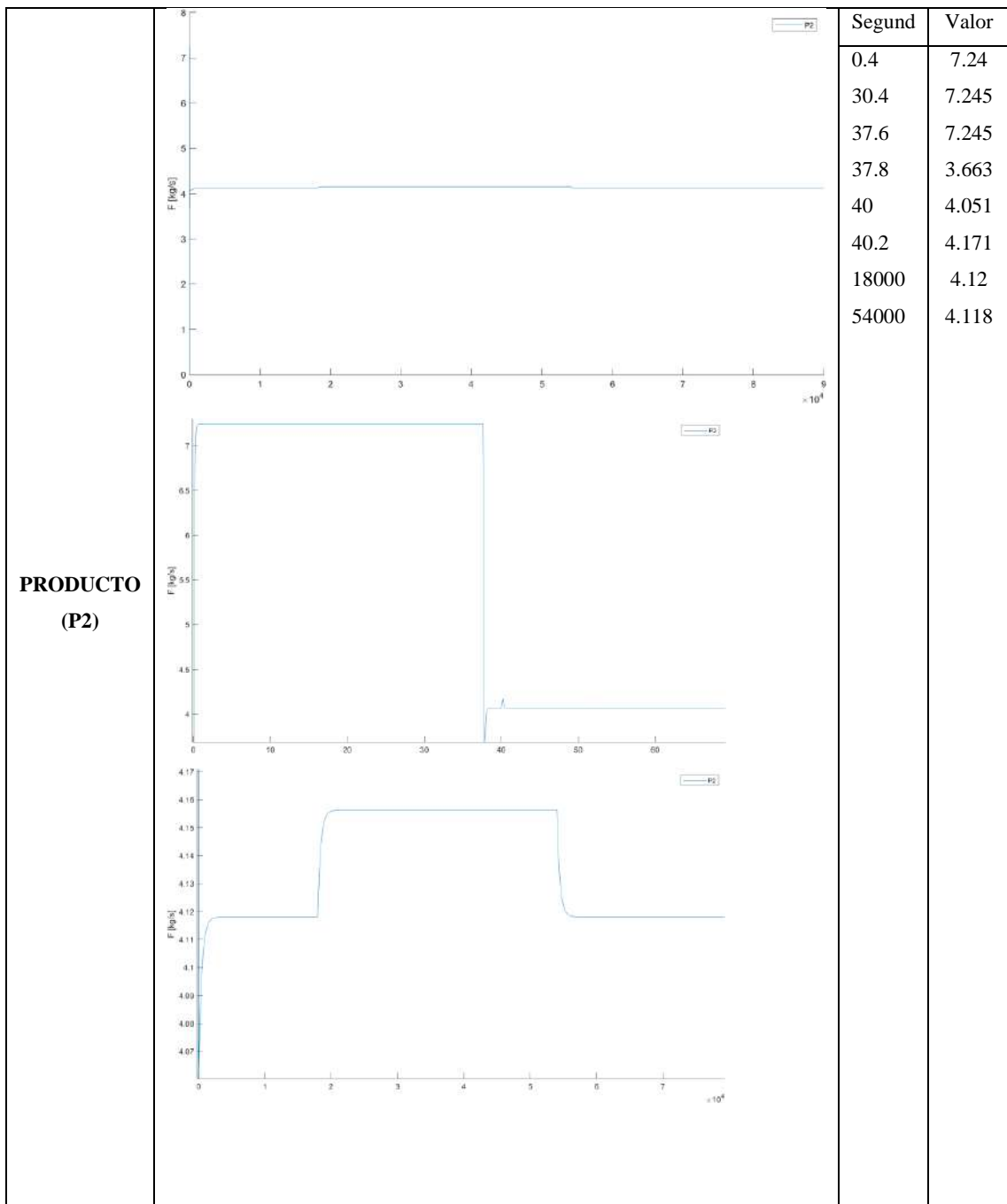


La tercera variable controlada la cual corresponde a la acumulación del producto M2 para su control se debe manipular a la variable P2 (cantidad producto de salida del efecto 2) su funcionamiento radica en que existe una válvula que regula el paso de la cantidad de producto y se va abriendo o cerrando hasta que la señal de retroalimentación alcance o se asemeje al punto de consigna (el grado de apertura viene dado por el controlador), la gráfica resultante refleja un aumento en el material acumulado en 2268.49 en 0.4 segundos se mantiene constante hasta el segundo 37.6, se produce un descenso hasta el segundo 37.8 llegando alcanzar un valor de 2268.25, inmediatamente se produce un ascenso hasta el segundo 38.2 tomando un valor de 2268.27 se mantiene constante hasta el segundo 40 y se presenta un pico el cual establece su máximo valor en 2268.284 luego desciende hasta 2268.277 en el segundo 40.5, al segundo 3500 alcanza un valor de 2268.2805 y se mantiene constante hasta el segundo 18000 en ese punto la gráfica vuelve a subir hasta un valor de 2268.2832 y vuelve a mantenerse constante hasta el segundo 54000 en donde se presenta un descenso de la gráfica hasta un valor de 2268.2805 y se mantiene durante todo el proceso.



<b>APERTURA</b>			Segund	Valor
			0.4	0.4936
			30.4	0.4936
			37.6	0.25
			38.2	0.2765
			40	0.2765
			40.2	0.284
			18000	0.283
		54000	0.2805	





**Análisis 2:** Efectuar un efecto de un cambio de entrada de pulso en la temperatura de alimentación (cambio de 85 a 95 C al tiempo = 5 horas (18000 s) y luego de 95 a 85 C en el tiempo = 15 horas (54000 s)).

Para ello se debe seguir la siguiente metodología:

- Dar clic en la ventana de información y saber cómo funciona el proceso:  
Identificar variables de entrada y de salida.

Conocer cuáles son los objetivos de control

Conocer como ingresar las perturbaciones al sistema

- Ubicarse en la sección 1, dentro del panel entradas y localizar la concentración inicial del alimento (Tf)
- Siguiendo las indicaciones de como ingresar el pulso modificar los parámetros de modo que:  
El límite inferior corresponde a 85 según el enunciado expuesto en análisis 2.  
El retraso según el enunciado es 18000s ya que corresponde al tiempo requerido para realizar el cambio.

La amplitud es el rango de crecimiento del salto, es decir, cómo el enunciado nos dice que existirá un cambio de 85 a 95 se debe ingresar un valor de 10 dentro de este apartado.

El periodo es el tiempo total que dura el salto en el impulso, en este caso será 54000 ya que el nuevo cambio según el enunciado se hará a ese tiempo.

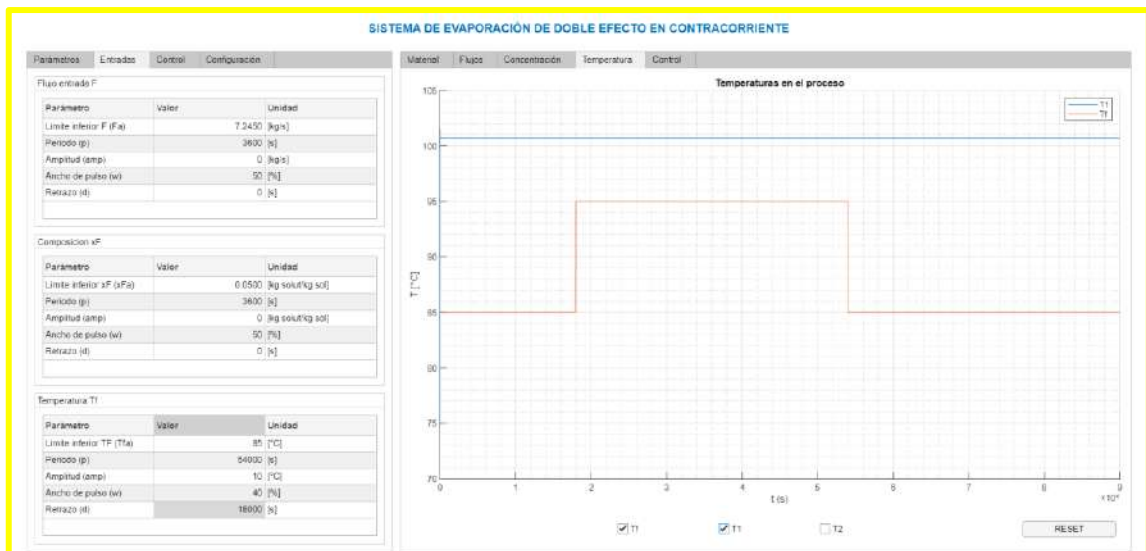
El ancho del pulso se establecerá mediante una regla de tres simple para lo cual el valor equivalente al tiempo que dura toda la simulación corresponderá al 100% mientras que el valor colocado en el periodo menos el valor del retraso determinará el valor del porcentaje que deberá ser colocado en este inciso

Periodo – Retraso: (54000 – 18000 = 36000) segundos

Regla de tres:  $\frac{36000 \times 100\%}{90000} = 40\%$  **ANCHO DEL PULSO**

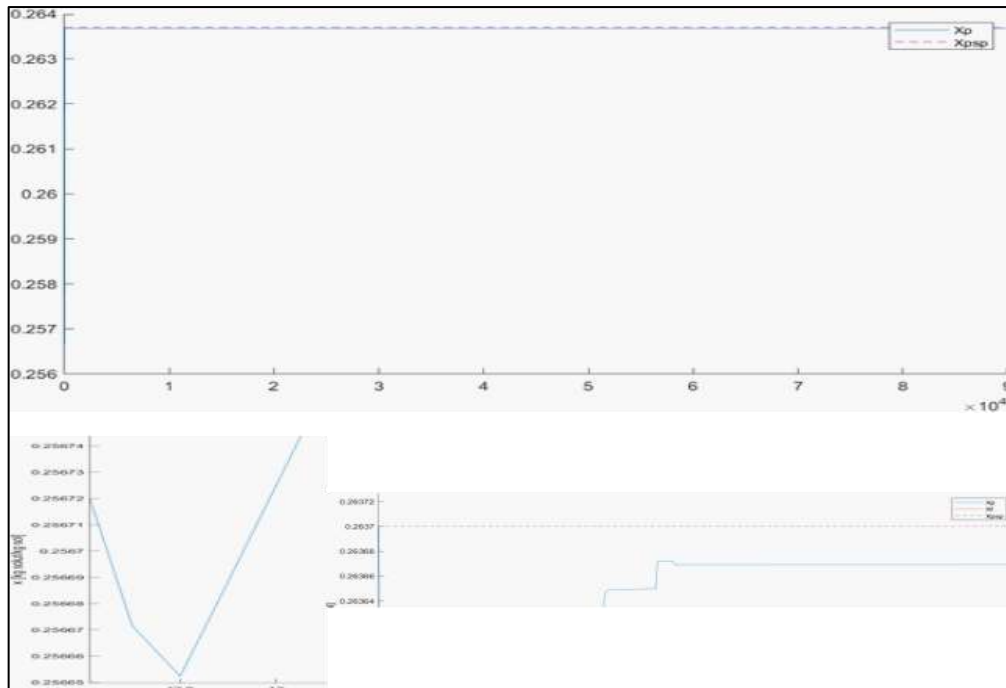
- Una vez modificados esos valores se presiona el botón resolver y se procede a analizar las gráficas

Entrada de la concentración inicial en forma de pulso provocando que las variables inmiscuidas dentro del proceso presenten cambios.

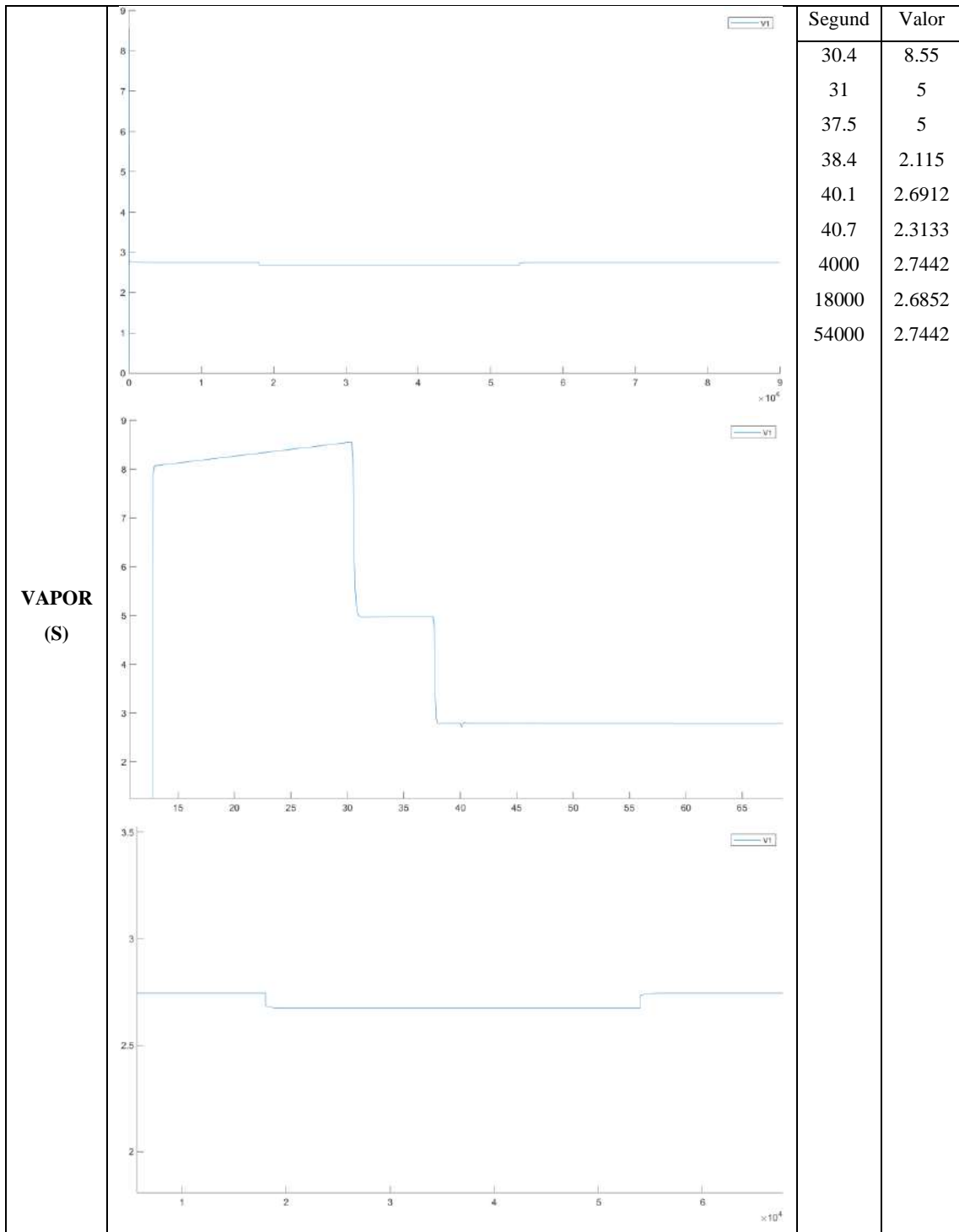


A continuación, se presenta la primera variable controlada la cual corresponde a la concentración del producto X<sub>p</sub> para su control se debe manipular a la variable S (cantidad de vapor de agua) su

funcionamiento radica en que existe una válvula que regula el paso de la cantidad de vapor y se va abriendo o cerrando hasta que la señal de retroalimentación alcance o se asemeje al punto de consigna (el grado de apertura viene dado por el controlador), la gráfica resultante refleja picos de descenso al segundo 12.8 reportándose un valor de 0.25665, a partir de ese punto el valor crece hasta 0.26365 al segundo 30.8 dicho valor se mantiene constante hasta el segundo 38 cuando cambia a un valor de 0.26367 el proceso se estabiliza hasta los 18000 segundos donde el valor sube hasta un valor de 0.2636705 y en el valor de 54000 segundos vuelve a 0.26367 manteniéndose constante denotando un mínimo error equivalente a 0.00003.

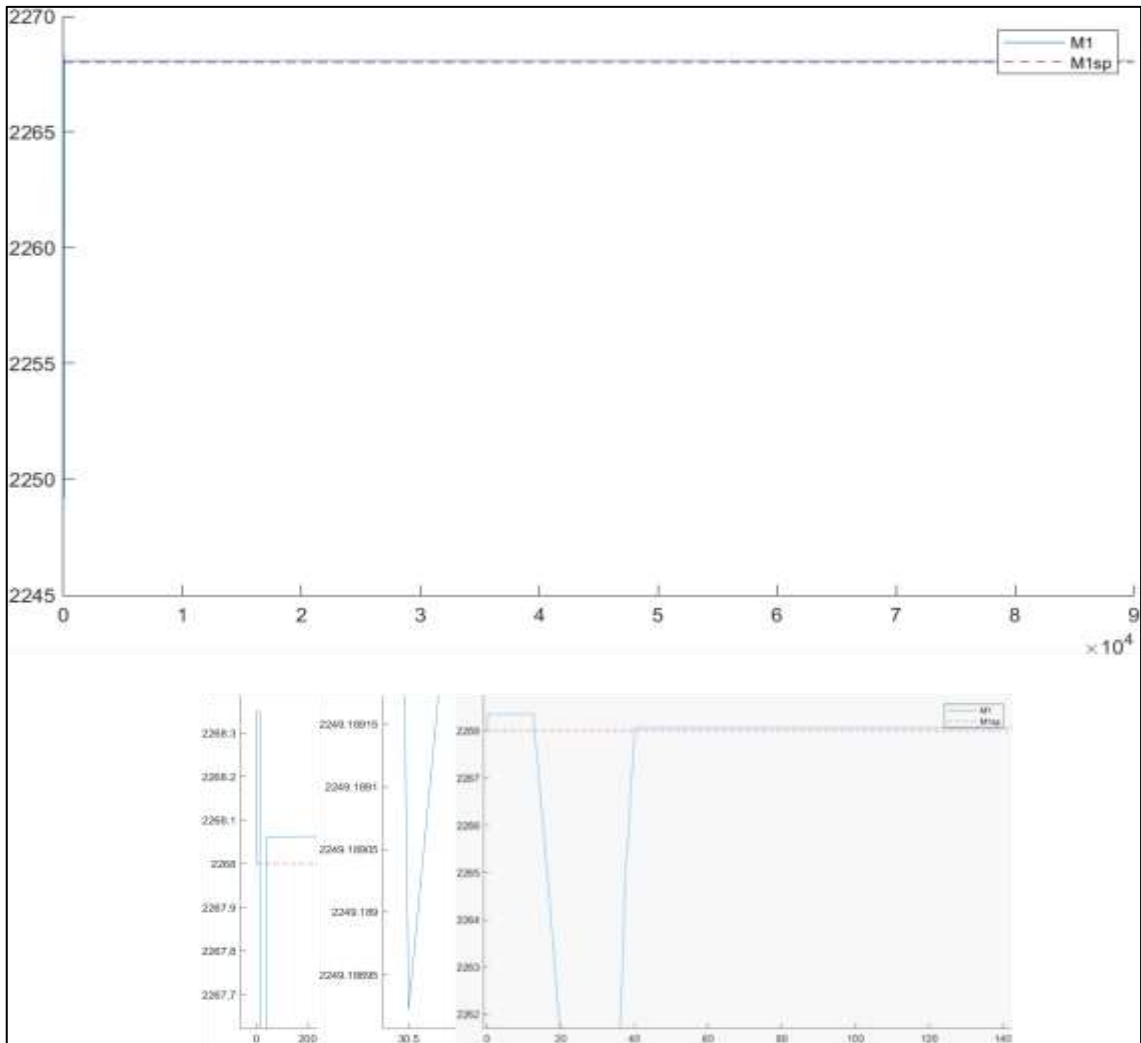


APERTURA		Segund	Valor
		0.5	0.351
12.9	0		
40.2	0.0624		
40.5	0.0605		
40.8	0.06118		
1000	0.0665		
18000	0.0695		
54000	0.0665		

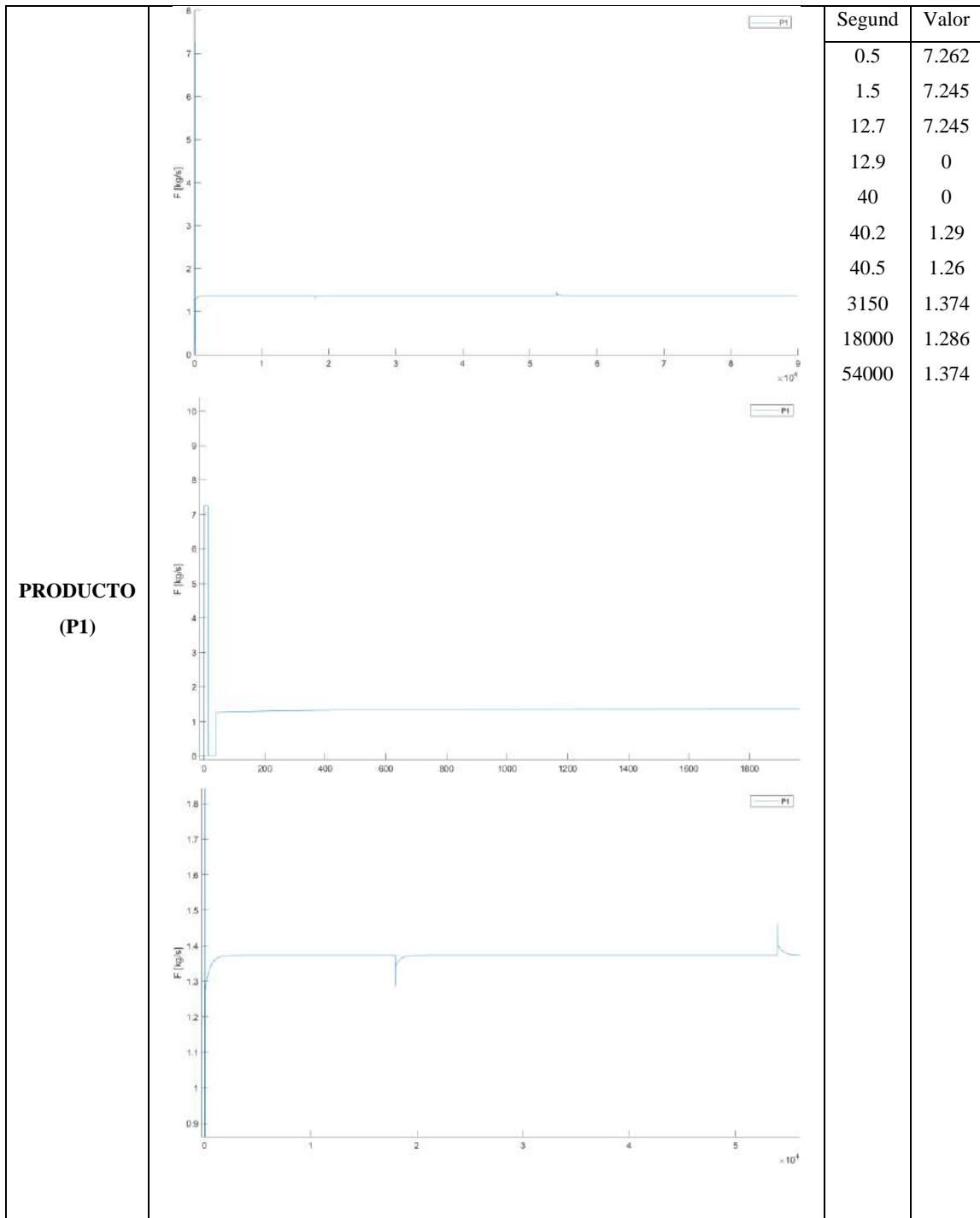


La segunda variable controlada la cual corresponde a la acumulación del producto M1 para su control se debe manipular a la variable P1 (cantidad producto de salida del efecto 1) su funcionamiento radica en que existe una válvula que regula el paso de la cantidad de producto y se va abriendo o cerrando hasta que la señal de retroalimentación alcance o se asemeje al punto de consigna (el grado de apertura viene dado por el controlador), la gráfica resultante refleja un aumento en el material acumulado en 2268.35 en 0.4 segundos se mantiene constante hasta el segundo 12.7, se produce un descenso hasta el segundo 30.5 llegando alcanzar un valor de

2249.1889, inmediatamente se produce un ascenso hasta el segundo 40.3 tomando un valor de 2268.0625 luego desciende hasta 2268.0605 en el segundo 40.5 alcanza un valor de 2268.0625, al segundo 5000 toma un valor de 2268.066 constante hasta el segundo 18000 donde desciende y toma un valor de 2268.062 y empieza a ascender hasta un valor de 2268.065 constante hasta el segundo 54000 donde el valor equivalente a 2268.067 se mantiene constante durante todo el proceso.

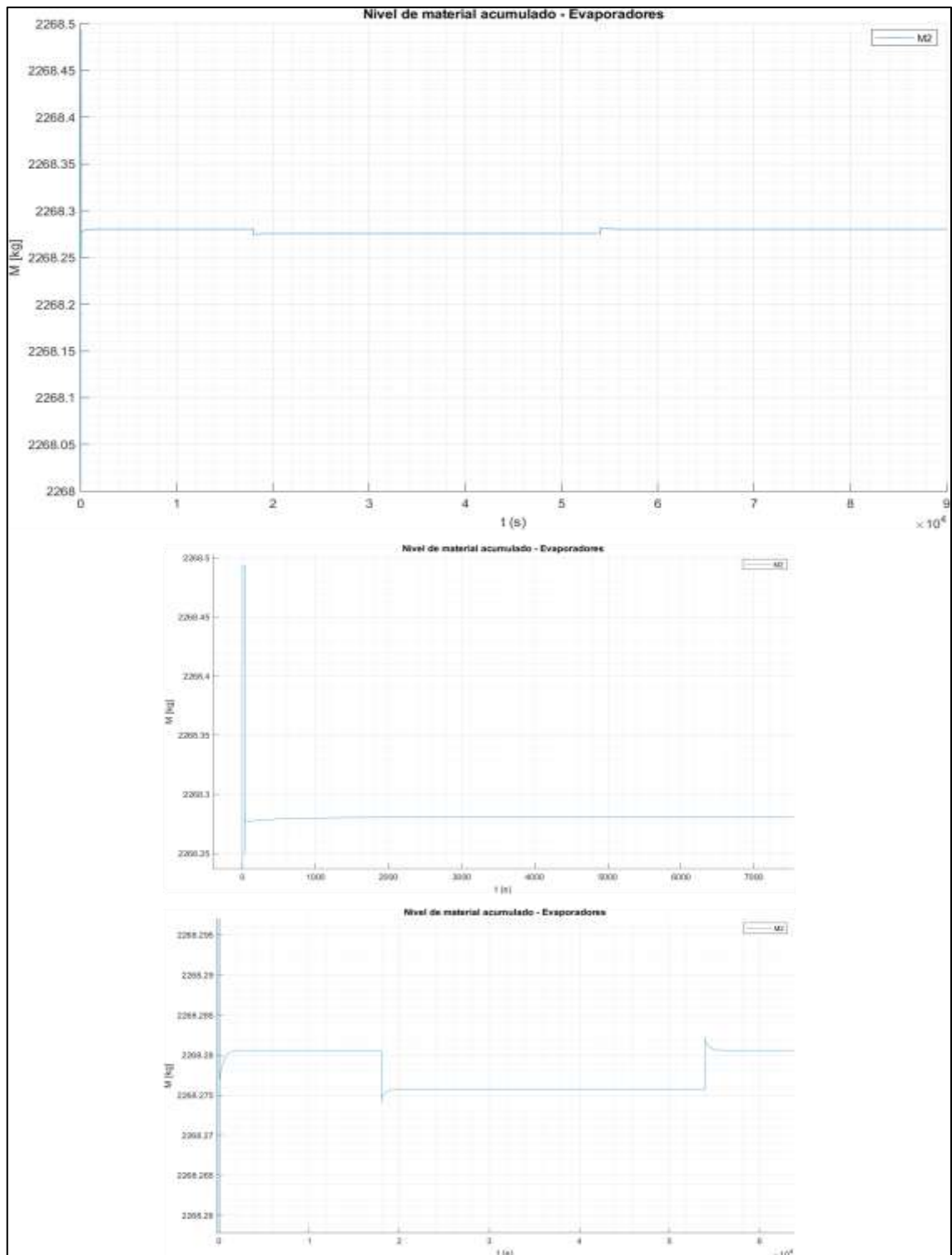


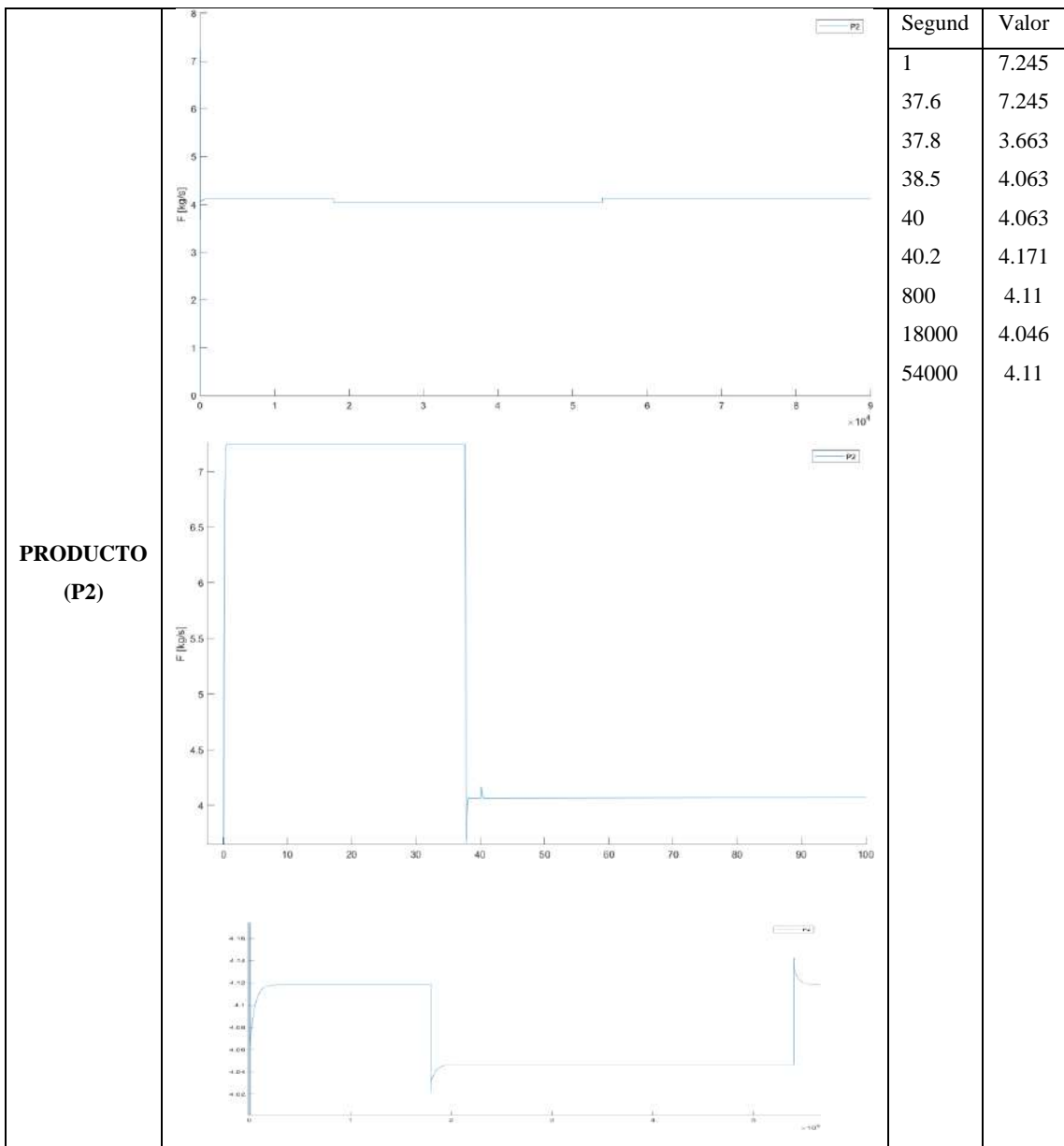
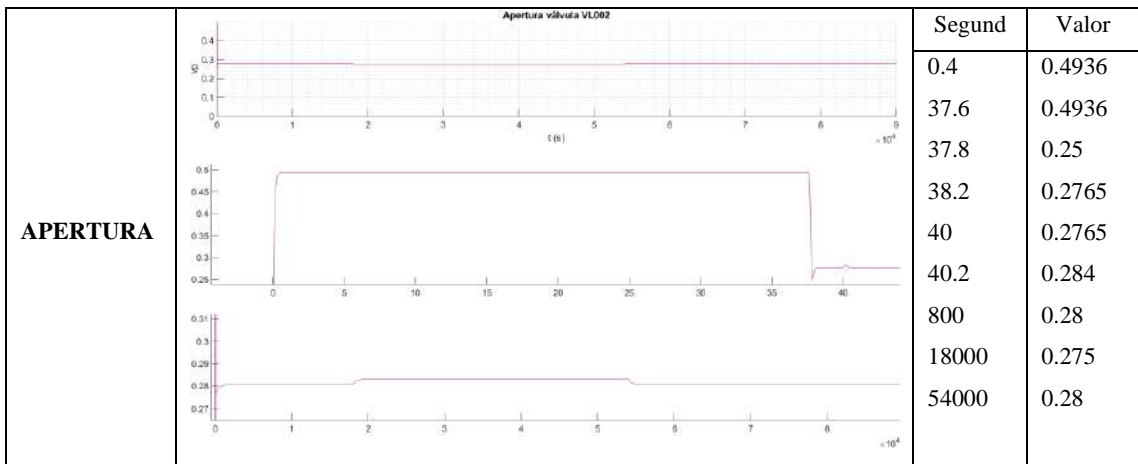
<b>APERTURA</b>	Apertura válvula VG001		Segund	Valor
			0.3	1
			30.4	1
			38.1	0.28
			40	0.28
			40.3	0.302
			1800	0.2985
			18000	0.302
			54000	0.2895



La tercera variable controlada la cual corresponde a la acumulación del producto M2 para su control se debe manipular a la variable P2 (cantidad producto de salida del efecto 2) su funcionamiento radica en que existe una válvula que regula el paso de la cantidad de producto y se va abriendo o cerrando hasta que la señal de retroalimentación alcance o se asemeje al punto de consigna (el grado de apertura viene dado por el controlador), la gráfica resultante refleja un aumento en el material acumulado en 2268.49 en 0.4 segundos se mantiene constante hasta el segundo 37.6, se produce un descenso hasta el segundo 37.8 llegando alcanzar un valor de 2268.25, inmediatamente se produce un ascenso hasta el segundo 38.2 tomando un valor de

2268.27 se mantiene constante hasta el segundo 40 y se presenta un pico el cual establece su máximo valor en 2268.284 luego desciende hasta 2268.277 en el segundo 40.5, al segundo 2500 alcanza un valor de 2268.2805 y se mantiene constante hasta el segundo 18000 en ese punto la gráfica baja hasta el valor de 2268.274 y vuelve a subir hasta un valor de 2268.275 y vuelve a mantenerse constante hasta el segundo 54000 en donde se presenta un ascenso de la gráfica hasta un valor de 2268.2823 baja hasta un valor de 2268.2805 y se mantiene durante todo el proceso.

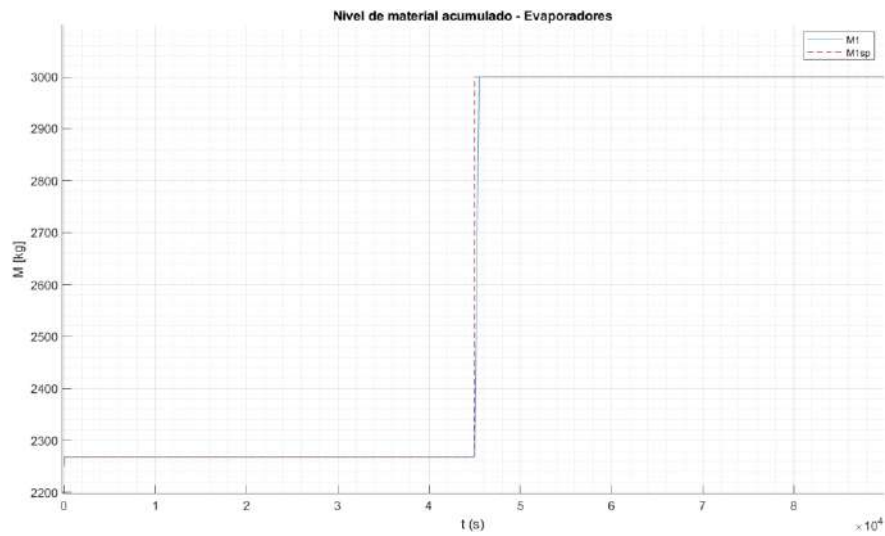




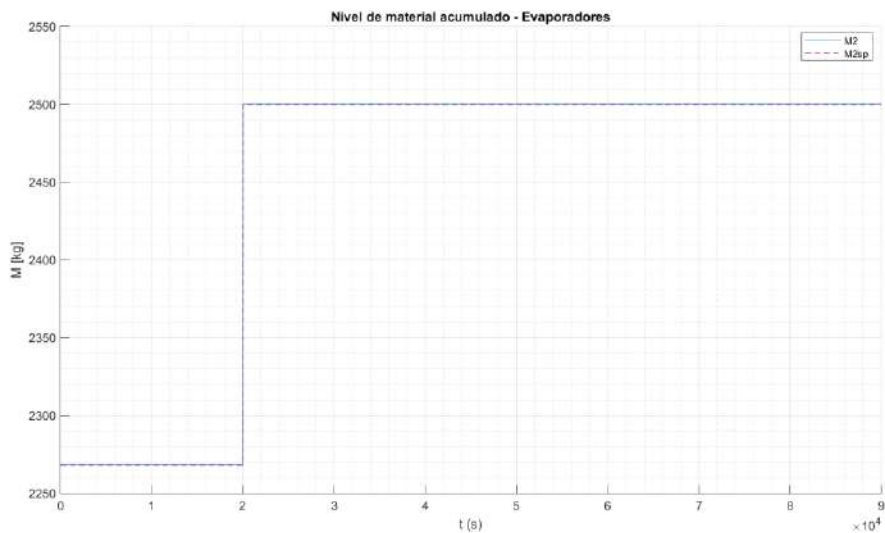


**Análisis 3:** Efectuar un cambio en los puntos de consigna y ver como se ajustan los controladores al set point.

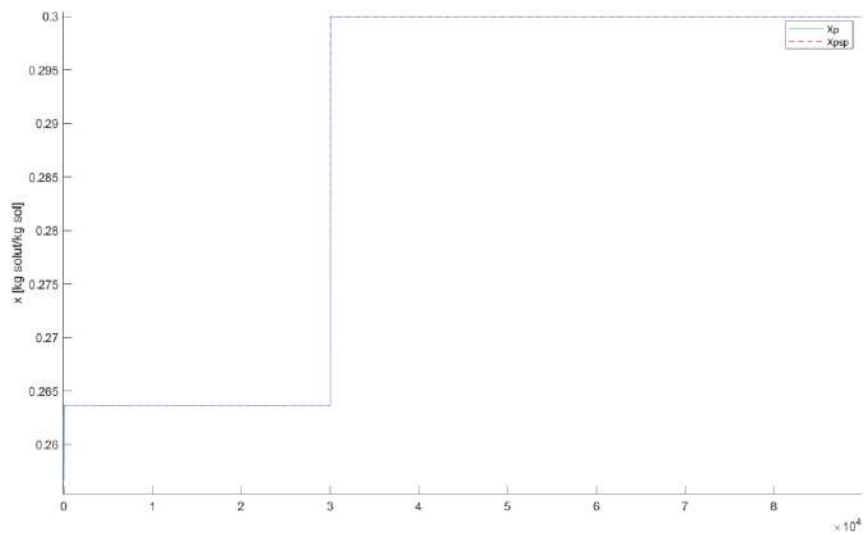
Cambio de la acumulación M1 de 2268 a 3000 en un tiempo de 12.5 horas (45000 segundos).



Cambio de la acumulación M2 a 2500 en un tiempo de 5.5 horas (20000 segundos).



Cambio de la concentración Xp de 0.2637 a 0.3 en un tiempo de 8.3 horas (30000 segundos).



## Parámetros que se pueden editar dentro de la aplicación

	Nombre del parámetro		Rango o Valor Sugerido		Unidades
<b>Parámetros Generales</b>	Temperatura vapor	(Ts)	115.7	Depende de la sustancia utilizada	°C
	Temperatura ebullición	(Tb)	100 (Agua)	Depende de la sustancia utilizada	°C
	Constante A solución	(A)	4.177	Responde a $h = 4.187(1 - 0.7X) T$	kJ/kg °C
	Constante B solución	(B)	2.506	Responde a $h = 4.187(1 - 0.7X) T$	kJ/kg °C
	Densidad líquida	(dcl)	980.5 (Agua)	Depende de la temperatura	kg/m <sup>3</sup>
	Densidad vapor	(dcp)	0.6 (Vapor de Agua)	Depende de la temperatura	kg/m <sup>3</sup>
	Gravedad específica vapor	(gs)	0.98 (Densidad relativa)	Depende de la densidad del líquido	adm
<b>Condiciones iniciales</b>	Masa inicial EVAP01	(M10)	A partir de 1000, el valor por defecto será 2268	(Depende del artículo de referencia)	kg
	Concentración inicial EVAP01	(Xp0)	A partir de 0,1, el valor por defecto será 0.2637	(Depende del artículo de referencia)	kg St/kg Sl
	Temperatura inicial EVAP01	(T10)	A partir de 30, el valor por defecto será 74.7	(Depende del artículo de referencia)	°C
	Masa inicial EVAP02	(M20)	A partir de 1000, el valor por defecto será 2268	(Depende del artículo de referencia)	kg
	Concentración inicial EVAP02	(X20)	A partir de 0,1, el valor por defecto será 0.2637	(Depende del artículo de referencia)	kg St/kg Sl
	Temperatura inicial EVAP02	(T20)	A partir de 30, el valor por defecto será 74.7	(Depende del artículo de referencia)	°C
<b>Entradas (Perturbaciones)</b>	<b>Flujo entrada (F)</b>	Límite inferior	(Fa)	A partir de 4,14, el valor por defecto será 7,245	kg/s
		Periodo	(p)	3600 - Valor final de la simulación	s
		Amplitud	(amp)	A partir de 0, no colocar valores exorbitantes	kg/s

				Ancho de pulso (w)	Responde a la fórmula de los análisis expuestos anteriormente	%			
				Retraso (d)	A partir de 0, no colocar valores exorbitantes	s			
				Límite inferior (xF)	0.16 – 0.36	Según la norma INEN 1 025	kg/s		
				Periodo (p)	3600 - Valor final de la simulación	s			
				Composición (xF)	Amplitud (amp)	A partir de 0, no colocar valores exorbitantes	kg/s		
					Ancho de pulso (w)	Responde a la fórmula de los análisis expuestos anteriormente	%		
					Retraso (d)	A partir de 0, no colocar valores exorbitantes	s		
				Temperatura (Tf)	Límite inferior (Fa)	A partir de 25, el valor por defecto será 85	kg/s		
					Periodo (p)	3600 - Valor final de la simulación	s		
					Amplitud (amp)	A partir de 0, no colocar valores exorbitantes	kg/s		
					Ancho de pulso (w)	Responde a la fórmula de los análisis expuestos anteriormente	%		
					Retraso (d)	A partir de 0, no colocar valores exorbitantes	s		
				Control	CONTROLADORES	PID 1	Constante Kp	-1	adm
							Constante Ki	0	adm
Constante Kd	0	adm							
PID 2	Constante N	0	Se debe aplicar el método			adm			
	Constante Kp	-1	heurístico de prueba y error para			adm			
	Constante Ki	0	colocar estos			adm			
	Constante Kd	0	valores			adm			
	Constante N	0				adm			
PID 3	Constante Kp	-1				adm			
	Constante Ki	0				adm			
	Constante Kd	0				adm			
SET - POINTS	Setpoint M1	Tiempo de salto	0 - tiempo final simulación			s			
		Valor inicial	A partir de 1000, el valor por defecto será 2268			kg			
		Valor final	A partir de 1000, el valor por defecto será 2269	kg					
	Setpoint Xp	Tiempo de salto	0 - tiempo final simulación	s					
		Valor inicial	0.16 – 0.36	Según la norma INEN 1 025	kg St/kg Sl				
		Valor final	0.16 – 0.36	Según la norma INEN 1 025	kg St/kg Sl				
	Setpoint M2	Tiempo de salto	0 - tiempo final simulación	s					
		Valor inicial	A partir de 1000, el valor por defecto será 2268	kg					

		Valor final	A partir de 1000, el valor por defecto será 2269	kg
<b>VÁLVULAS</b>	<b>Válvula VL001</b>	Coficiente de la válvula	Se debe buscar en el catálogo un valor que permita generar un buen control del sistema	$m^3/h.bar^{0,5}$
		Caída de presión	Establecer la caída de presión idónea para que la válvula funcione de la manera correcta y no afecte al proceso	bar
	<b>Válvula VG001</b>	Coficiente de la válvula	Se debe buscar en el catálogo un valor que permita generar un buen control del sistema	$m^3/h.bar^{0,5}$
		Caída de presión	Establecer la caída de presión idónea para que la válvula funcione de la manera correcta y no afecte al proceso	bar
	<b>Valvula VL002</b>	Coficiente de la válvula	Se debe buscar en el catálogo un valor que permita generar un buen control del sistema	$m^3/h.bar^{0,5}$
		Caída de presión	Establecer la caída de presión idónea para que la válvula funcione de la manera correcta y no afecte al proceso	bar
<b>Configuración</b>	Tiempo inicial		Colocar 0	s
	Tiempo final		Tiempo de simulación total que se quiere llegar	s
	Salto de tiempo		Pasos para resolver las ecuaciones diferenciales, se recomienda no editar el valor	s

### Enlace de video informativo

<https://drive.google.com/drive/folders/1-PFLgmK9QhmgvSmc-BnL2UAfoXWjiO2c?usp=sharing>

## ANEXO B. CÓDIGO DE APP DESIGNER IMPLEMENTADO PARA EL CORRECTO FUNCIONAMIENTO DE LA INTERFAZ DISEÑADA

```
properties (Access = private)
params % Description
results
simulation
config
flags
end
methods (Access = private)
function defaultCase(app)
% Overall params
app.params.Ts = 115.7; % °C
app.params.Tb = 100; % °C
app.params.A = 4.187;
app.params.B = 2.9309;
% Evaporator's parameters
app.params.Evap1.M0 = 2268; % kg
app.params.Evap1.C0 = 0.2637; % kg solut/kg sol.
app.params.Evap1.T0 = 74.7; % °C
app.params.Evap2.M0 = 2268; % kg
app.params.Evap2.C0 = 0.0867; % kg solut/kg sol.
app.params.Evap2.T0 = 52; % °C
% Perturbations
app.params.F.a = 7.245; % kg/s
app.params.F.per = 3600; % s
app.params.F.w = 50; % %
app.params.F.amp = 0; %
app.params.F.d = 0; % s
app.params.xf.a = 0.05; % kg solut/ kg sol
app.params.xf.per = 3600; % s
app.params.xf.w = 50; % %
app.params.xf.amp = 0; %
app.params.xf.d = 0; % s
app.params.Tf.a = 85; % °C
app.params.Tf.per = 3600; % s
app.params.Tf.w = 50; % %
app.params.Tf.amp = 0; %
app.params.Tf.d = 0; % s
% Set points
app.params.m1sp.t = 0; % s
app.params.m1sp.vi = 2268; % kg
app.params.m1sp.vf = 2268; % kg
app.params.m2sp.t = 0; % s
app.params.m2sp.vi = 2268; % kg
app.params.m2sp.vf = 2268; % kg
app.params.c1sp.t = 0; % s
app.params.c1sp.vi = 0.2637; % kg
app.params.c1sp.vf = 0.2637; % kg
% Valves' parameters
app.params.dp1 = 2.013; % bar
```

```

app.params.dp2 = 4.531; % bar
app.params.dp3 = 1.013; % bar
app.params.kv1 = 53; % m3/h bar^0.5 liquid valve
app.params.kv2 = 520; % m3/h bar^0.5 gas valve
app.params.kv3 = 53; % m3/h bar^0.5 liquid valve
app.params.dcl = 980.5; % kg/m^3
app.params.dcg = 0.6; % kg/m^3
app.params.gs = 0.98; % no def
%PIDs
app.params.pid1.kp = -1;
app.params.pid1.ki = 0;
app.params.pid1.kd = 0;
app.params.pid1.n = 100;
app.params.pid2.kp = 10000;
app.params.pid2.ki = 0;
app.params.pid2.kd = 0;
app.params.pid2.n = 100;
app.params.pid3.kp = -1;
app.params.pid3.ki = 0;
app.params.pid3.kd = 0;
app.params.pid3.n = 100;
% Configuración
app.config.startTime = 0;
app.config.stopTime = 90000;
app.config.stepTime = 0.3;
app.fillTables;
end
function fillTables(app)
varName = {'Parametro','Valor', 'Unidades'};
% Parametros generales
cnames = {'Temperatura vapor (Ts)';
'Temperatura ebullición (Tb)';
'Constante A solución (A)';
'Constante B solución (B)';
'Densidad liquido (dcl)';
'Densidad del vapor (dcg)';
'Gravedad especifica vapor (gs)'};
data = [app.params.Ts; % °C
app.params.Tb; % °C
app.params.A;
app.params.B;
app.params.dcl;
app.params.dcg;
app.params.gs];
unitc = {'[°C]';
'[°C]';
'[kJ/kg °C]';
'[kJ/kg °C]';
'[kg/m^3]';
'[kg/m^3]';
'[adm]'};
constdata = table(cnames, data, unitc, 'VariableNames', varName);
app.UITable.Data = constdata;
% Parametros de evaporador

```

```

cnames = {'Masa inicial EVAP01 (M10)';
'Concentración inicial EVAP01 (Xp0)';
'Temperatura inicial EVAP01 (T10)';
'Masa inicial EVAP02 (M20)';
'Concentración inicial EVAP02 (X20)';
'Temperatura inicial EVAP02 (T20)'};
data = [app.params.Evap1.M0; % °C
app.params.Evap1.C0; % °C
app.params.Evap1.T0;
app.params.Evap2.M0; % °C
app.params.Evap2.C0; % °C
app.params.Evap2.T0];
unitc = {'[kg]';
'[kg St/kg Sl]';
'[°C]';
'[kg]';
'[kg St/kg Sl]';
'[°C]'};
data = table(cnames, data, unitc, 'VariableNames', varName);
app.UITable2.Data = data;
% Perturbations
% Flujo de entrada F
cnames = {'Limite inferior F (Fa)';
'Periodo (p)';
'Amplitud (amp)';
'Ancho de pulso (w)';
'Retrazo (d)'};
data = [app.params.F.a; % [kg/s]
app.params.F.per; % [s]
app.params.F.amp; % [kg/s]
app.params.F.w; % [%]
app.params.F.d]; % [s]
unitc = {'[kg/s]';
'[s]';
'[kg/s]';
'[%]';
'[s]'};
data = table(cnames, data, unitc, 'VariableNames', varName);
app.UITable3.Data = data;
% Composición de entrada F
cnames = {'Limite inferior xF (xFa)';
'Periodo (p)';
'Amplitud (amp)';
'Ancho de pulso (w)';
'Retrazo (d)'};
data = [app.params.xf.a; % kg/s
app.params.xf.per; % [s]
app.params.xf.amp; % [kg/s]
app.params.xf.w; % [%]
app.params.xf.d]; % [s]
unitc = {'[kg solut/kg sol]';
'[s]';
'[kg solut/kg sol]';
'[%]'};

```

```

'[s]');
data = table(cnames, data, unitc, 'VariableNames', varName);
app.UITable4.Data = data;
% Temperatura de flujo de entrada
cnames = {'Limite inferior TF (Tfa)';
'Periodo (p)';
'Amplitud (amp)';
'Ancho de pulso (w)';
'Retrazo (d)'};
data = [app.params.Tf.a; % kg/s
app.params.Tf.per; % [s]
app.params.Tf.amp; % [kg/s]
app.params.Tf.w; % [%]
app.params.Tf.d]; % [s]
unitc = {'[°C]';
'[s]';
'[°C]';
'[%]';
'[s]'};
data = table(cnames, data, unitc, 'VariableNames', varName);

```

```

app.UITable5.Data = data;
% CONTROL
% PID 1
cnames = {'Constante Kp';
'Constante Ki';
'Constante Kd';
'Constante N'};
data = [app.params.pid1.kp; % kg/s
app.params.pid1.ki; % [s]
app.params.pid1.kd; % [kg/s]
app.params.pid1.n]; % [s]
unitc = {'[adm]';
'[adm]';
'[adm]';
'[adm]'};
data = table(cnames, data, unitc, 'VariableNames', varName);

```

```

app.UITable6.Data = data;
% PID 2
cnames = {'Constante Kp';
'Constante Ki';
'Constante Kd';
'Constante N'};
data = [app.params.pid2.kp; % kg/s
app.params.pid2.ki; % [s]
app.params.pid2.kd; % [kg/s]
app.params.pid2.n]; % [s]
unitc = {'[adm]';
'[adm]';
'[adm]';
'[adm]'};

```



```
data = table(cnames, data, unitc, 'VariableNames', varName);
```

```
app.UITable7.Data = data;
```

```
% PID 3
```

```
cnames = {'Constante Kp';
```

```
'Constante Ki';
```

```
'Constante Kd';
```

```
'Constante N'};
```

```
data = [app.params.pid3.kp; % kg/s
```

```
app.params.pid3.ki; % [s]
```

```
app.params.pid3.kd; % [kg/s]
```

```
app.params.pid3.n]; % [s]
```

```
unitc = {'[adm]';
```

```
'[adm]';
```

```
'[adm]';
```

```
'[adm]'};
```

```
data = table(cnames, data, unitc, 'VariableNames', varName);
```

```
app.UITable8.Data = data;
```

```
% SETPOINT
```

```
% Masa M1
```

```
cnames = {'Tiempo de salto (t)';
```

```
'Valor inicial (vi)';
```

```
'Valor final (vf)'};
```

```
data = [app.params.m1sp.t; % [s]
```

```
app.params.m1sp.vi; % [kg]
```

```
app.params.m1sp.vf]; % [kg]
```

```
unitc = {'[s]';
```

```
'[kg]';
```

```
'[kg]'};
```

```
data = table(cnames, data, unitc, 'VariableNames', varName);
```

```
app.UITable9.Data = data;
```

```
% Masa xp
```

```
cnames = {'Tiempo de salto (t)';
```

```
'Valor inicial (vi)';
```

```
'Valor final (vf)'};
```

```
data = [app.params.c1sp.t; % [s]
```

```
app.params.c1sp.vi; % [kg solut/kg sol]
```

```
app.params.c1sp.vf]; % [kg solut/kg sol]
```

```
unitc = {'[s]';
```

```
'[kg solut/kg sol]';
```

```
'[kg solut/kg sol]'};
```

```
data = table(cnames, data, unitc, 'VariableNames', varName);
```

```
app.UITable10.Data = data;
```

```
% Masa M2
```

```
cnames = {'Tiempo de salto (t)';
```

```
'Valor inicial (vi)';
```

```
'Valor final (vf)'};
```

```

data = [app.params.m2sp.t; % [s]
app.params.m2sp.vi; % [kg]
app.params.m1sp.vf]; % [kg]
unitc = {'[s]';
'[kg]';
'[kg]'};
data = table(cnames, data, unitc, 'VariableNames', varName);

```

```

app.UITable11.Data = data;
% Valve VL001
cnames = {'Coeficiente de válvula (Kv)';
'Caida de presión (dP)'};
data = [app.params.kv1; % [m^3/h bar^0.5]
app.params.dp1]; % [bar]
unitc = {'[m^3/h bar^0.5]';
'[bar]'};
data = table(cnames, data, unitc, 'VariableNames', varName);

```

```

app.UITable9_2.Data = data;
% Valve VL002
cnames = {'Coeficiente de válvula (Kv)';
'Caida de presión (dP)'};
data = [app.params.kv2; % [m^3/h bar^0.5]
app.params.dp2]; % [bar]
unitc = {'[m^3/h bar^0.5]';
'[bar]'};
data = table(cnames, data, unitc, 'VariableNames', varName);

```

```

app.UITable10_2.Data = data;
% Valve VG001
cnames = {'Coeficiente de válvula (Kv)';
'Caida de presión (dP)'};
data = [app.params.kv3; % [m^3/h bar^0.5]
app.params.dp3]; % [bar]
unitc = {'[m^3/h bar^0.5]';
'[bar]'};
data = table(cnames, data, unitc, 'VariableNames', varName);

```

```

app.UITable11_2.Data = data;
% Configuración
cnames = {'Tiempo inicial (t0)';
'Tiempo final (tf)';
'Salto de tiempo (st)'};
data = [app.config.startTime; % [m^3/h bar^0.5]
app.config.stopTime;
app.config.stepTime]; % [bar]
unitc = {'[s]';
'[s]';
'[s]'};
data = table(cnames, data, unitc, 'VariableNames', varName);

```

```

app.UITable12.Data = data;
end
function getData(app)
% Función para capturar datos desde los campos.
data = app.UITable.Data.Valor;
app.params.Ts = data(1);
app.params.Tb = data(2);
app.params.A = data(3);
app.params.B = data(4);
app.params.dcl = data(5);
app.params.dcg = data(6);
app.params.gs = data(7);
data = app.UITable2.Data.Valor;
app.params.Evap1.M0 = data(1);
app.params.Evap1.C0 = data(2);
app.params.Evap1.T0 = data(3);
app.params.Evap2.M0 = data(4);
app.params.Evap2.C0 = data(5);
app.params.Evap2.T0 = data(6);
data = app.UITable3.Data.Valor;
app.params.F.a = data(1);
app.params.F.per = data(2);
app.params.F.amp = data(3);
app.params.F.w = data(4);
app.params.F.d = data(5);
data = app.UITable4.Data.Valor;
app.params.xf.a = data(1);
app.params.xf.per = data(2);
app.params.xf.amp = data(3);
app.params.xf.w = data(4);
app.params.xf.d = data(5);
data = app.UITable5.Data.Valor;
app.params.Tf.a = data(1);
app.params.Tf.per = data(2);
app.params.Tf.amp = data(3);
app.params.Tf.w = data(4);
app.params.Tf.d = data(5);
data = app.UITable6.Data.Valor;
app.params.pid1.kp = data(1);
app.params.pid1.ki = data(2);
app.params.pid1.kd = data(3);
app.params.pid1.n = data(4);
data = app.UITable7.Data.Valor;
app.params.pid2.kp = data(1);
app.params.pid2.ki = data(2);
app.params.pid2.kd = data(3);
app.params.pid2.n = data(4);
data = app.UITable8.Data.Valor;
app.params.pid3.kp = data(1);
app.params.pid3.ki = data(2);
app.params.pid3.kd = data(3);
app.params.pid3.n = data(4);

```

```

data = app.UITable9.Data.Valor;
app.params.m1sp.t = data(1);
app.params.m1sp.vi = data(2);
app.params.m1sp.vf = data(3);
data = app.UITable10.Data.Valor;
app.params.c1sp.t = data(1);
app.params.c1sp.vi = data(2);
app.params.c1sp.vf = data(3);
data = app.UITable11.Data.Valor;
app.params.m2sp.t = data(1);
app.params.m2sp.vi = data(2);
app.params.m2sp.vf = data(3);
data = app.UITable9_2.Data.Valor;
app.params.kv1 = data(1);
app.params.dp1 = data(2);
data = app.UITable10_2.Data.Valor;
app.params.kv2 = data(1);
app.params.dp2 = data(2);
data = app.UITable11_2.Data.Valor;
app.params.kv3 = data(1);
app.params.dp3 = data(2);
data = app.UITable12.Data.Valor;
app.config.startTime = data(1);
app.config.stopTime = data(2);
app.config.stepTime = data(3);
% disp(app.params);
end

function configSimulation(app)
simInp = Simulink.SimulationInput('DoubleEffectEvaporator');

simInp = simInp.setVariable('params', app.params);
simInp = simInp.setModelParameter('startTime',
num2str(app.config.startTime));
simInp = simInp.setModelParameter('stopTime',
num2str(app.config.stopTime));
simInp = simInp.setModelParameter('FixedStep',
num2str(app.config.stepTime));
simInp = simulink.compiler.configureForDeployment(simInp);
app.simulation = simInp;
end
function drawMassAccumulated(app)
% Flags from checkbox
app.flags.hflags = [app.M1CheckBox.Value,...
app.M2CheckBox.Value,...
app.M1spCheckBox.Value,...
app.M2spCheckBox.Value];
% Clean axes
cla(app.UIAxes4);
% Hold on
hold(app.UIAxes4, 'on');
% Legend names' vector

```

```

legname = {};
% Draw Lines
if app.flags.hflags(1) == 1 % M1
plot(app.UIAxes4, app.results.time, app.results.evap(:,1));
legname{end+1} = 'M1';
end
if app.flags.hflags(2) == 1 % M2
plot(app.UIAxes4, app.results.time, app.results.evap(:,5));
legname{end+1} = 'M2';
end
if app.flags.hflags(3) == 1 % m1sp
plot(app.UIAxes4, app.results.time, app.results.setpoint(:,1), '--',
'Color', [0.6350 0.0780 0.1840]);
legname{end+1} = 'M1sp';
end
if app.flags.hflags(4) == 1 % m2sp
plot(app.UIAxes4, app.results.time, app.results.setpoint(:,3), '--',
'Color', [0.6350 0.0780 0.50]);
legname{end+1} = 'M2sp';
end
% Add legend
legend(app.UIAxes4, legname);

ylabel(app.UIAxes4, 'M [kg]');
end
function drawFlows(app)
% Flags from checkbox
app.flags.hflags = [app.FCheckBox.Value,...
app.P1CheckBox.Value,...
app.V1CheckBox.Value,...
app.P2CheckBox.Value,...
app.V2CheckBox.Value,...
app.SCheckBox.Value];
% Clean axes
cla(app.UIAxes5);
% Hold on
hold(app.UIAxes5, 'on');
% Legend names' vector
legname = {};
% Draw Lines
if app.flags.hflags(1) == 1 % F
plot(app.UIAxes5, app.results.time, app.results.input(:,1));
legname{end+1} = 'F';
end
if app.flags.hflags(2) == 1 % P1
plot(app.UIAxes5, app.results.time, app.results.flows(:,1));
legname{end+1} = 'P1';
end
if app.flags.hflags(3) == 1 % V1
plot(app.UIAxes5, app.results.time, app.results.evap(:,4));
legname{end+1} = 'V1';
end
if app.flags.hflags(4) == 1 % P2

```

```

plot(app.UIAxes5, app.results.time, app.results.flows(:,3));
legname{end+1} = 'P2';
end
if app.flags.hflags(5) == 1 % V2
plot(app.UIAxes5, app.results.time, app.results.evap(:,8));
legname{end+1} = 'V2';
end
if app.flags.hflags(6) == 1 % S1
plot(app.UIAxes5, app.results.time, app.results.flows(:,2));
legname{end+1} = 'S';
end
% Add legend
legend(app.UIAxes5, legname);

ylabel(app.UIAxes5, 'F [kg/s]');
end
function drawConcentrations(app)
% Flags from checkbox
app.flags.hflags = [app.XpCheckBox.Value, ...
app.X2CheckBox.Value, ...
app.XfCheckBox.Value, ...
app.XpspCheckBox.Value];
% Clean axes
cla(app.UIAxes6);
% Hold on
hold(app.UIAxes6, 'on');
% Legend names' vector
legname = {};
% Draw Lines
if app.flags.hflags(1) == 1 % xp1
plot(app.UIAxes6, app.results.time, app.results.evap(:, 2));
legname{end+1} = 'Xp';
end
if app.flags.hflags(2) == 1 % xp2
plot(app.UIAxes6, app.results.time, app.results.evap(:, 6));
legname{end+1} = 'X2';
end
if app.flags.hflags(3) == 1 % xf
plot(app.UIAxes6, app.results.time, app.results.input(:, 2));
legname{end+1} = 'Xf';
end
if app.flags.hflags(4) == 1 % xpsp
plot(app.UIAxes6, app.results.time, app.results.setpoint(:,2), '--',
'Color', [0.6350 0.0780 0.50]);
legname{end+1} = 'Xpsp';
end
% Add legend
legend(app.UIAxes6, legname);

ylabel(app.UIAxes6, 'x [kg solut/kg sol]');
end
function drawTemperatures(app)

```

```

% Flags from checkbox
app.flags.hflags = [app.T1CheckBox.Value,...
app.T2CheckBox.Value,...
app.TfCheckBox.Value];
% Clean axes
cla(app.UIAxes7);
% Hold on
hold(app.UIAxes7, 'on');
% Legend names' vector
legname = {};
% Draw Lines
if app.flags.hflags(1) == 1 % T1
plot(app.UIAxes7, app.results.time, app.results.evap(:,3));
legname{end+1} = 'T1';
end
if app.flags.hflags(2) == 1 % T2
plot(app.UIAxes7, app.results.time, app.results.evap(:,7));
legname{end+1} = 'T2';
end
if app.flags.hflags(3) == 1 % Tf
plot(app.UIAxes7, app.results.time, app.results.input(:,3));
legname{end+1} = 'Tf';
end
% Add legend
legend(app.UIAxes7, legname);

ylabel(app.UIAxes7, 'T [°C]');
end
function drawControl(app)
% Clean axes
cla(app.UIAxes);
cla(app.UIAxes2);
cla(app.UIAxes3);
% Draw Lines
plot(app.UIAxes, app.results.time, app.results.control(:,1), '-k',
'Color', [0.6350 0.0780 0.50]);
plot(app.UIAxes2, app.results.time, app.results.control(:,2), '-k',
'Color', [0.6350 0.0780 0.50]);

plot(app.UIAxes3, app.results.time, app.results.control(:,3), '-k',
'Color', [0.6350 0.0780 0.50]);

end

function drawResults(app)
% Dibujar material acumulado
app.drawMassAccumulated;
% Dibujar flujos
app.drawFlows;
% Dibujar concentraciones

```

```

app.drawConcentrations;
% Dibujar temperaturas
app.drawTemperatures;
% Dibujar líneas de control
app.drawControl;
end
function runSimulation(app)
% 1. Configurar la simulación
app.getData;
app.configSimulation;
% 2. Correr la simulación
app.toggleSimButton('off', 'Simulando...');
simOut = sim(app.simulation);
app.toggleSimButton('on', 'Simular');
% 3. Guardar los resultados
app.results.time = simOut.tout;
app.results.evap = simOut.yout.signals(1).values;
app.results.control = simOut.yout.signals(2).values;
app.results.flows = simOut.yout.signals(3).values;
app.results.setpoint = simOut.yout.signals(4).values;
app.results.input = simOut.yout.signals(5).values;
% disp(app.results);
% 5. Dibujar las gráficas
app.drawResults;
end
function toggleSimButton(app, state, label)
app.ResolverButton.Enable = state;
app.ResolverButton.Text = label;
end

end

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

% Code that executes after component creation
function startupFcn(app)
% Cargar datos caso por defecto
app.defaultCase;
end

% Button pushed function: ResolverButton
function ResolverButtonPushed(app, event)
app.runSimulation;
end

% Value changed function: M1CheckBox
function M1CheckBoxValueChanged(app, event)
app.drawMassAccumulated;

```



end

```
% Value changed function: M2CheckBox
function M2CheckBoxValueChanged(app, event)
app.drawMassAccumulated;
end
```

```
% Value changed function: M1spCheckBox
function M1spCheckBoxValueChanged(app, event)
app.drawMassAccumulated;
end
```

```
% Value changed function: M2spCheckBox
function M2spCheckBoxValueChanged(app, event)
app.drawMassAccumulated;
end
```

```
% Value changed function: FCheckBox
function FCheckBoxValueChanged(app, event)
app.drawFlows;
end
```

```
% Value changed function: P1CheckBox
function P1CheckBoxValueChanged(app, event)
app.drawFlows;
end
```

```
% Value changed function: V1CheckBox
function V1CheckBoxValueChanged(app, event)
app.drawFlows;
end
```

```
% Value changed function: P2CheckBox
function P2CheckBoxValueChanged(app, event)
    app.drawFlows;
end
```

```
% Value changed function: V2CheckBox
function V2CheckBoxValueChanged(app, event)
    app.drawFlows;
end
```

```
% Value changed function: SCheckBox
function SCheckBoxValueChanged(app, event)
    app.drawFlows;
```

```
end
```

```
% Value changed function: XpCheckBox  
function XpCheckBoxValueChanged(app, event)  
    app.drawConcentrations;  
end
```

```
% Value changed function: X2CheckBox  
function X2CheckBoxValueChanged(app, event)  
    app.drawConcentrations;  
end
```

```
% Value changed function: XfCheckBox  
function XfCheckBoxValueChanged(app, event)  
    app.drawConcentrations;  
end
```

```
% Value changed function: T1CheckBox  
function T1CheckBoxValueChanged(app, event)  
    app.drawTemperatures;  
end
```

```
% Value changed function: T2CheckBox  
function T2CheckBoxValueChanged(app, event)  
    app.drawTemperatures;  
end
```

```
% Value changed function: TfCheckBox  
function TfCheckBoxValueChanged(app, event)  
    app.drawTemperatures;  
end
```

```
% Button pushed function: SalirButton  
function SalirButtonPushed(app, event)  
    app.delete;  
end
```

```
% Button pushed function: ReiniciarButton  
function ReiniciarButtonPushed(app, event)  
    app.defaultCase;  
end
```

```
% Value changed function: XpspCheckBox  
function XpspCheckBoxValueChanged(app, event)  
    app.drawConcentrations;
```

```
end
% Button pushed function: INFORMACINButton
function INFORMACINButtonPushed(app, event)
    info
end
end
```



**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**

**DIRECCIÓN DE BIBLIOTECAS Y RECURSOS DEL APRENDIZAJE  
UNIDAD DE PROCESOS TÉCNICOS Y ANÁLISIS BIBLIOGRÁFICO Y DOCUMENTAL**

**REVISIÓN DE NORMAS TÉCNICAS, RESUMEN Y BIBLIOGRAFÍA**

**Fecha de entrega:** 07 / 09 / 2021

<b>INFORMACIÓN DEL AUTOR/A (S)</b>
<b>Nombres – Apellidos:</b> <i>Jerry Leodan Mena Nagua</i>
<b>INFORMACIÓN INSTITUCIONAL</b>
<b>Facultad:</b> <i>Ciencias</i>
<b>Carrera:</b> <i>Ingeniería Química</i>
<b>Título a optar:</b> <i>Ingeniero Químico</i>
<b>f. Analista de Biblioteca responsable:</b> <i>Ing. Leonardo Medina Ñuste MSc.</i>



1629-DBRA-UTP-2021