



UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Escola d'Enginyeria de Barcelona Est

TRABAJO DE FINAL DE GRADO

Grado en Ingeniería Electrónica Industrial y Automática

**SUPERVISIÓN, CONTROL Y SIMULACIÓN DE UNA PLANTA
DE PRODUCCIÓN DE ACEITE DE OLIVA**



Memoria Técnica

Autor: Sergio Casas Gallegos
Director: Javier Francisco Gámiz Caro
Convocatoria: Junio 2019

Resumen

En el presente proyecto se ha diseñado y automatizado satisfactoriamente una planta de producción de aceite de oliva para su correcto control y monitorización posterior. El proceso contempla el ciclo completo de la producción de aceite, desde la entrada de aceituna hasta el almacenaje del aceite en el depósito, considerando todos los subprocesos y maquinaria efectiva en ellos.

Para la realización de la planta y el proceso que se ha automatizado, se ha buscado y recopilado información sobre la producción de aceite de oliva virgen para poder dotar a este proyecto con un acabado profesional.

El Controlador Lógico Programable (PLC) contiene la parte de programación lógica y por medio del Sistema de Control y Adquisición de Datos (SCADA) se hace el control y obtención de datos del proceso. Para el control del proceso se ha utilizado una comunicación PLC-SCADA tipo DDE.

Todo esto, se ha podido llevar a cabo gracias al entorno de programación de sistemas de automatización en un controlador virtual de Rockwell Automation y el software de visualización de pantallas humano-máquina (HMI) de Wonderware.

Resum

En el present projecte s'ha dissenyat i automatitzat satisfactòriament una planta de producció d'oli d'oliva pel seu correcte control i monitorització posterior. El procés contempla el cicle complet de la producció d'oli, des de l'entrada d'oliva fins l'emmagatzematge de l'oli en el depòsit, fent consideració de tots el subprocessos i maquinaria efectiva en ells.

Per a la realització de la planta i el procés que s'ha automatitzat, s'ha fet recerca i recollit informació sobre la producció d'oli d'oliva verge per poder dotar aquest projecte amb un acabat professional.

El controlador Lògic Programable (PLC) conté la part de programació lògica i per mitja del Sistema de Control i Adquisició de Dades (SCADA) es fa el control i l'obtenció de dades del procés. Pel control del procés s'ha utilitzat una comunicació PLC-SCADA tipus DDE.

Tot això, s'ha pogut dur a terme gràcies a l'entorn de programació de sistemes d'automatització en un controlador virtual de Rockwell Automation i el programari de visualització de pantalles humà-maquina (HMI) de Wonderware.

Abstract

In the present project, an olive oil production plant has been successfully designed and automated for proper control and subsequent monitoring. The process covers the complete cycle of oil production, from the olive inlet to the storage of the oil in the tank, considering all the subprocesses and effective machinery in them.

For the completion of the industrial plant and the process that has been automated, information of the virgin olive oil production has been searched and gathered in order to provide the project with a professional finish.

The Programmable Logic Controller (PLC) contains the logical programming part and the Control and Data Acquisition System (SCADA) offers the control and data attainment. DDE communication type is established for controlling the process PLC-SCADA.

All this has been achieved thanks to the automation system programming in a virtual environment by Rockwell Automation and human-machine interfaces (HMI) software by Wonderware.

Agradecimientos

Sin ningún orden especial de mérito en la mención de estos agradecimientos.

Gracias a mi familia por el inmenso apoyo y los ánimos recibidos durante toda la carrera.

A aquellos amigos que han sabido darme palabras de ánimo en momentos concretos.

Al profesorado, tanto de la EEBE como de la EUETIB, durante toda esta etapa de formación universitaria.

Finalmente a la universidad por darme la oportunidad de ser participe en el programa Erasmus+, una de las experiencias más enriquecedoras tanto a nivel académico como personal.

Índice

RESUMEN	I
RESUM	II
ABSTRACT	III
AGRADECIMIENTOS	IV
1. INTRODUCCIÓN	2
1.2. Objetivos.....	2
1.3. Motivación.....	2
1.4. Justificación	3
1.5. Alcance.....	3
1.6. Antecedentes	4
1.6.1. Evolución de la automatización	4
1.6.2. Estado del arte.....	5
2. ANÁLISIS DEL PROBLEMA	7
2.1. Descripción del proceso a automatizar	7
2.1.1. Recepción y limpieza de aceituna	8
2.1.2. Molienda.....	8
2.1.3. Termobatido	8
2.1.4. Extracción de aceite.....	8
2.1.5. Tamizado y centrifugado	10
2.1.6. Almacenaje	10
2.2. Modelo del proceso	10
2.3. Sistema de control.....	11
2.3.1. Equipos	12
2.3.2. Fases del proceso.....	14
2.4. Especificaciones funcionales.....	14
2.4.1. Requerimientos funcionales.....	15
2.4.2. Requerimientos de diseño	17
2.5. Metodología de desarrollo.....	18
2.6. Planificación de tareas	20
2.7. Recursos.....	21

3.	DISEÑO E IMPLEMENTACIÓN DE LA SOLUCIÓN	23
3.1.	Arquitectura del sistema de control	23
3.1.1.	Hardware del sistema.....	23
3.1.2.	Software del sistema	23
3.1.3.	Vista global de las comunicaciones	24
3.1.4.	Comunicaciones SCADA-Controlador-Proceso	25
3.2.	Codificación de los elementos	30
3.3.	Descomposición del problema de control.....	31
3.4.	Definición del fichero de intercambio Controlador-SCADA	32
3.5.	Simulación del proceso	44
3.5.1.	Diseño	44
3.5.2.	Bloques del sistema.....	45
3.5.3.	Estructura de simulación	48
3.6.	Programa del controlador	51
3.6.1.	Estructura del programa.....	51
3.6.2.	Definición de tipos de datos.....	51
3.6.3.	Lógica de control de los elementos.....	52
3.6.4.	Lógica de control de los sistemas.....	55
3.6.5.	Secuencias de control.....	57
3.7.	Programa del software SCADA.....	58
3.7.1.	Sistemas SCADA.....	58
3.7.2.	Árbol de navegación.....	58
3.7.3.	Definición de los tipos de datos	59
3.7.4.	Diseño de las pantallas de la aplicación	60
3.7.5.	Diseño de los comandos.....	65
3.7.6.	Scripts	67
3.7.7.	Diseño de la interfaz de alarmas del sistema.....	69
3.7.8.	Gestión de usuarios	71
3.7.9.	Gráficos de históricos y tendencias.....	72
4.	PRUEBAS Y RESULTADOS	77
4.1.	Prueba de conexión controlador-SCADA.....	77
4.2.	Prueba de las entradas y salidas físicas	78
4.3.	Pruebas de funcionalidad.....	79
5.	NORMATIVA	86

5.1. Implementación al programa del PLC.....	86
5.2. Implementación al programa SCADA	88
5.2.1. Guía ISA-S5.5.....	89
5.2.2. Guía GEDIS.....	92
6. ANÁLISIS DE IMPACTO AMBIENTAL.....	94
CONCLUSIONES	95
BIBLIOGRAFÍA	96



CAPÍTULO 1: INTRODUCCIÓN



1. Introducción

En este capítulo se verán los diferentes objetivos del proyecto de una manera detallada. Además del alcance del proyecto, quedarán reflejados tanto los motivos como la justificación de la elección del tema.

1.2. Objetivos

Los objetivos propuestos tienen diferente índole, por ello, se pueden dividir en tres tipos diferentes: objetivo general, objetivo principal y objetivos secundarios.

El objetivo general se basa en hacer uso de la capacidad de analizar un problema de automatización y llegar a diseñar e implementar una solución basada en la integración de Controladores Lógicos Programables (PLC) y Sistemas de Supervisión, Control y Adquisición de Datos (SCADA), a partir de contenidos teóricos y prácticos avanzados.

El objetivo principal es el perfecto y correcto funcionamiento de todo el proceso y planta creados, de inicio a fin, tanto en modo remoto manual como en modo automático. Desde la entrada de aceitunas hasta el almacenaje del aceite en el depósito, haciendo el control y supervisión de todos los parámetros reales fundamentales del manejo de varias máquinas puestas en línea, correcto funcionamiento de los sensores, actuadores y dispositivos de control de los que se haga uso.

Los objetivos secundarios son orientados al desarrollo de sistemas ya creados, más exigentes y complicados como la implementación de sistemas PID para el control de variables (flujo de agua en mezcla, temperatura,...), sistemas osciladores, creación de Add-Ons, selección de maquinaria del mismo tipo, aleatoriedad en la extracción de aceite, control de usuarios, control máximo de todos los parámetros, de entre otros...

1.3. Motivación

Una de las principales motivaciones para realizar este Trabajo de Fin de Grado ha sido el interés y predisposición hacia la programación de PLCs y creación de SCADAs debido a la extensa utilización en ingeniería y mundo laboral. Llegar a comprender lo importante que es la automatización a días de hoy y como afecta a procesos diarios que vemos comunes que comparándolos antiguamente eran procesos manuales, más costosos y menos productivos.

A la hora de escoger el proceso fue algo difícil y dubitativo, de todas las opciones de la lluvia de ideas de procesos posibles para automatizar, el proceso de producción de aceite parecía la mejor idea.

1.4. Justificación

Actualmente estamos en un momento que la automatización forma parte de todos, o casi todos, los procesos ya que las empresas buscan y realizan esta transformación para estar a la vanguardia para poder competir en el mercado, ofrecer su producto de mejor calidad y más barato que las otras empresas a cambio de un coste de producción inferior.

La implementación y mantenimiento de la automatización de procesos ayuda a mejorar la seguridad, incrementa la productividad, reduce los costes de producción a la larga, reduce la cantidad de trabajos pesados y establece un estándar de calidad en el producto.

Gracias a lo mencionado anteriormente y viendo la gran variedad de oferta en aceites y la competencia entre diferentes marcas, se ha encontrado acertado diseñar un proceso automático de producción de aceite de oliva de tal manera que contemple la máxima realidad posible.

En definitiva, se trata de automatizar un proceso de producción de aceite de oliva para optimizar la productividad, el funcionamiento, la calidad y la supervisión.

1.5. Alcance

El ámbito de alcance del proyecto está compuesto por un lado, el diseño, la programación e implementación de todos los elementos en un PLC según requisitos definidos, así como ejecución de una simulación de la planta a automatizar con el PLC, de la manera más realista posible y realizando un conjunto de pruebas teniendo en cuenta todos los elementos que la componen.

Por otro lado, el diseño, la programación e implementación de todos los elementos en SCADA, que será utilizado para supervisar el proceso actual se lleva a cabo en la planta gracias a la comunicación directa con el PLC. El sistema diseñado permitirá cambiar entre sistemas manual o automático siempre que se den las condiciones necesarias.

1.6. Antecedentes

1.6.1. Evolución de la automatización

Siglo XIX y anteriores

En la revolución industrial se introdujeron los primeros motores y junto con la máquina de vapor se crearon nuevos sistemas de control automáticos, como reguladores de temperatura y reguladores de presión. En el 1801, una patente de un telar automático que utiliza tarjetas perforadas creada por Joseph Marie Jacquard revoluciona la industria textil.

Siglo XX

Las salas de control se hicieron comunes en la década de 1920 y hasta principios de 1930 el control de procesos era solo encender/apagar. En la década de 1930 comenzaron a introducirse los controladores, con capacidad de realizar cambios calculados como respuestas a las desviaciones de una cifra de control.

En 1947, se creó el primer transistor por los físicos John Bardeen, Walter Brattain y William Shockley. Años más tarde, en 1959 aparece la primera herramienta controlada por un motor.

En 1968 se abre paso en la historia los PLC (Controlador Lógico Programable) con el controlador industrial modular creado por Dick Morley permitiendo el paso a la automatización de procesos electromecánicos propios de las líneas de montaje. En cuanto a máquinas, en 1978, AMK inventa la programación CNC (Control Numérico Computarizado) que permitiría su control remoto.

Transcurrieron nueve años, cuando por casualidad, un cliente hizo un encargo de un sistema de control Beckhoff con la peculiaridad que fuera equipado con un disco duro con el propósito de recoger datos. Lo que se hizo fue integrar un PC en el sistema y a partir de ese momento se supo que el PC integrado podía servir para mucho más, cosa que provocó una revolución en la automatización industrial.

A lo largo de 1997 llegó el empuje de integración, la tecnología de la automatización evolucionaba en un control cada vez más descentralizado e inteligente con componentes que se comunicaban entre ellos industrialmente (Ethernet). Además se empezó a fabricar virtualmente y a desarrollar productos digitales con la tecnología de la automatización.

La máxima revolución llegó en 2004 con la llegada del microchip y se implantó la funcionalidad del PLC en esta estructura de pequeña dimensión. A partir de entonces y desde 2010 se han creado gran variedad de autómatas. Estos autómatas son compactos y sencillos para todo tipo de aplicaciones (desde industriales hasta domésticas), con totalidad de control, modulares, prestaciones similares a las de un pequeño ordenador y con posibilidad de ampliación.

1.6.2. Estado del arte

Se sabe que la automatización es el motivo por la cual ha habido un gran salto evolutivo en cuanto a la productividad y la industria está en medio de una transformación digital acelerada exponencial por tecnologías en pleno crecimiento. Esta automatización industrial unida al *Internet of Things* (IoT) se define con el término *Industrial Internet of Things* (IIoT), que se identifica con el uso de las tecnologías IP (Internet Protocol) para conectar los procesos y dispositivos en una red o conjunto de redes.

Este hecho ha posibilitado un cambio de mentalidad sobre la automatización tradicional. La principal diferencia reside en el volumen de la información captada, registrada y el modo en el que se usa para potenciar la efectividad de las operaciones y procesos. Esto permite que partes separadas de una línea de producción o proceso de fabricación se comuniquen entre sí en tiempo real, haciendo que el proceso sea fácil de monitorizar y controlar. Por esa razón, hoy en día cuándo se habla de automatización e intercambio de datos en las tecnologías de fabricación se hace referencia a los conceptos de *Industria 4.0* (cuarta revolución industrial), *Industrial IoT* o *Factoría Digital*.

CAPÍTULO 2: ANÁLISIS DEL PROBLEMA



2. Análisis del problema

En este capítulo se describe el principio básico del proceso a automatizar desglosándolo en fases más pequeñas, así como los diferentes equipos que modelan el sistema de control, requisitos para el funcionamiento y diseño.

2.1. Descripción del proceso a automatizar

Se ha dividido el proceso de producción de aceite de oliva en 6 subprocesos que siguen un orden lógico:

1. Recepción y limpieza de aceituna
2. Molienda
3. Termobatido
4. Extracción del aceite
5. Tamizado y centrifugado
6. Almacenaje

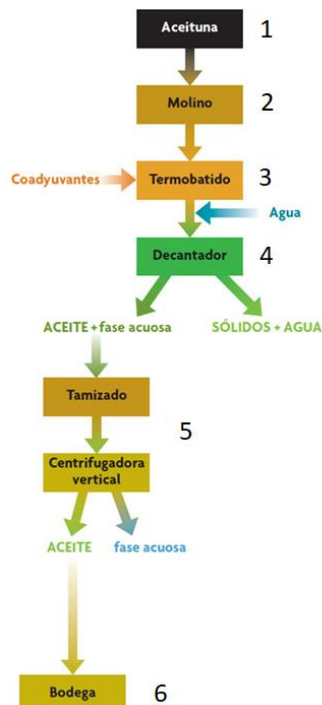


Figura 1. Fases del proceso de elaboración del aceite. [2]

2.1.1. Recepción y limpieza de aceituna

La recepción se realiza a través de la descarga de camiones o tractores con las aceitunas recién cogidas de los olivares. Primeramente, se comprueba el estado del fruto y se pesa para saber la cantidad de aceituna total a descargar. A continuación, las aceitunas son enviadas a la tolva de recepción conducidas por una cintra transportadora estas son lavadas y limpiadas con el fin de eliminar las hojas, piedras, tallos, tierra, así como abonos o productos fitosanitarios que puedan acompañar el fruto y son solubles al agua. La tolva es la encargada de almacenar la cantidad inicial de aceituna lavada, expulsar el agua sobrante del lavado y administrar la entrada del fruto a la prensa.

2.1.2. Molienda

Desde la recepción en la tolva a la molienda no han de transcurrir más de 48 horas ya que el fruto comenzaría a perder propiedades, por consecuente también el aceite. La molienda se hace a través de una prensa hidráulica donde se obtiene una pasta de la cual se extrae la materia oleosa a una presión de 300-400 bar. Hay que evitar tiempos excesivos de molturación puesto que se origina un exceso de aireación en la pasta con la consiguiente pérdida de aromas e iniciándose las correspondientes reacciones de oxidación.

2.1.3. Termobatido

Este subproceso se realiza en la batidora a 27 °C, de duración no superior a los 45 minutos, con la masa proveniente de la molienda. Tiene como misión favorecer la separación de fases (aceite, agua y alpechines) unificando y agrandando las pequeñas partículas de aceites contenidas en la masa. La batidora está dotada con palas de movimiento circular y eje horizontal, giran a velocidad lenta, con aportación de calor, mediante la doble cámara por la que circula agua caliente. En el termobatido se le añade a la masa coadyuvante (microtalco y enzimas) como máximo de 1,2 % de esta y tienen como objetivo facilitar la extracción del aceite. Finalmente, se le añade agua no caliente en pequeñas cantidades y sin sobrepasar el 10 % de la masa con el mismo propósito.

2.1.4. Extracción de aceite

Acabado el proceso de batido pasa al decánter, que es un centrifugador helicoidal horizontal y camisa maciza que sirve para la separación en continuo de sustancias sólidas contenidas en aceite como son pulpa, hueso molido, pellejos, etc. de los componentes líquidos. El decánter separa en fases según la densidad de la masa, por un lado cerca del eje la fase ligera, por otro la mezcla de

aceite y agua y alpeorujos junto a la pared del bol o cilindro (estos últimos por su parte posterior), normalmente se obtiene un 20-25% en aceite de la cantidad de masa entrada.

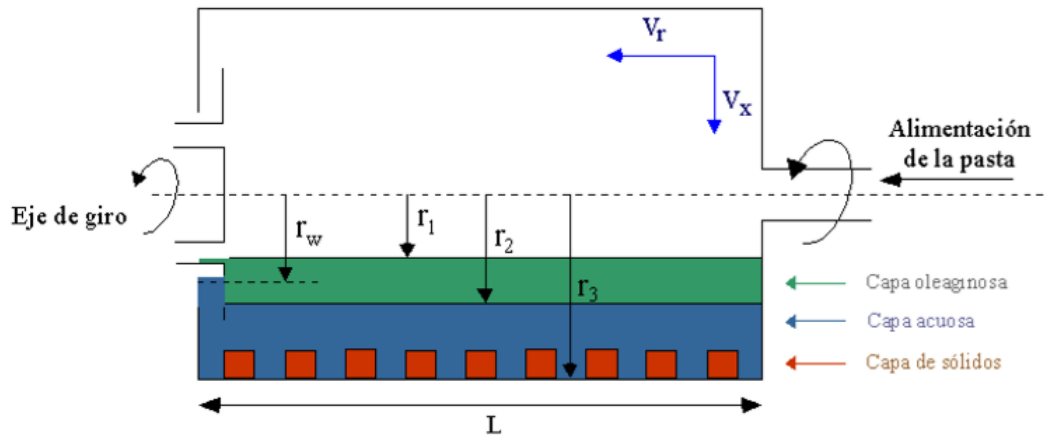


Figura 2. Esquema de centrifugación de pasta de aceituna de un decánter. [Cortesía: Universidad de Jaén]

Según el proceso de elaboración el sistema cambia sustancialmente según se realice con el sistema a dos o tres fases, el proyecto se inclina a la elaboración en dos fases, seguidamente se detallan las diferencias básicas de entre los dos métodos.

- **Sistema de elaboración a dos fases**, el decánter separa por un lado el aceite y por otro los sólidos y alpechín o agua de vegetación conjuntamente, con lo que se obtiene un subproducto que se denomina orujo de dos fases.
- **Sistema de elaboración a tres fases**, el decánter separa por un lado el aceite y por otro los sólidos con un cierto contenido de alpechín y agua de proceso denominado orujo de tres fases.

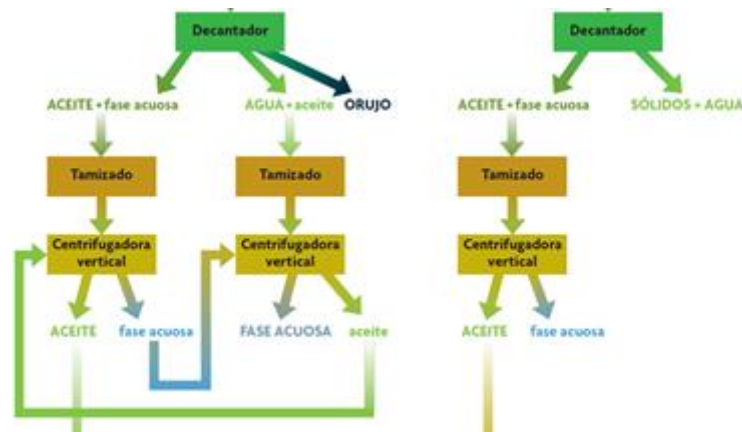


Figura 3. Diferenciación del sistema a dos fases (a la derecha) y tres fases (a la izquierda). [2]

2.1.5. Tamizado y centrifugado

El aceite obtenido en el decánter se recoge en un vibrofiltro, que elimina los sólidos que pueda llevar en suspensión antes de entrar en la centrifugadora vertical. El principio de la centrifugadora se basa en la diferencia de densidad entre el aceite y las impurezas, así pues, se elimina la humedad, los sólidos finos y las impurezas del aceite.

2.1.6. Almacenaje

Finalmente se procede al almacenaje en depósitos de acero inoxidable de tipo alimentario a temperatura constante de 12 °C durante un mínimo de 45 días, con el fin que el aceite de oliva virgen se estabilice y reposen las impurezas que todavía quedan en él.

2.2. Modelo del proceso

Se definen las condiciones del modelo del proceso de las diferentes partes para tratarlas en detalle, estas condiciones serán posteriormente implementadas en el modelo de simulación de la planta.

- **Entrada de aceituna:** la entrada máxima de aceituna es de 100 kg para reducir los tiempos de espera de los procesos.
- **Motobombas de agua:** cada una bombea agua a presión a razón de 5 l/s.
- **Nivel de tolva:** la capacidad de la tolva coincide con la entrada máxima de aceituna (100 kg), se ha de tener en cuenta que la entrada no es inmediata ya que primero pasa por la cinta transportadora donde es efectiva la limpieza del fruto y cabe la posibilidad de no ser efectiva toda la cantidad entrada. El vaciado de la tolva se realiza a razón de 3 kg/s, ergo, a capacidad máxima tardaría 34 s en vaciarse.
- **Masa obtenida de la prensa:** se considera que la masa obtenida será siempre igual a la cantidad de aceituna entrada.
- **Masa obtenida de la batidora:** en la batidora se le añade el 11,2 % a la masa total de aceituna debido al añadido de coadyuvante de talco 1 %, al coadyuvante de enzimas 0,2 % y de agua al final del proceso 10 %.

- **Nivel de batidora:** la batidora tiene una capacidad de 100 kg. Para el llenado, la válvula que limita el caudal de entrada es de 5 kg/s, con flujo infinito de entrada la batidora tardaría 20 s en llenarse. La válvula de salida vacía la batidora a razón de 1,5 kg/s, ergo, a capacidad máxima tardaría 67 s en vaciarse.
- **Aceite obtenido del decánter:** primeramente se debe considerar que el aceite obtenido suele ser de un 20 - 25 % del valor de la masa. Se tendrá en cuenta que dos masas idénticas contienen diferentes valores de aceite y en este caso se hará que la obtención de aceite sea aleatoria de entre el 20 % a 25 % de la masa por posibles diferencias que estas puedan contener.
- **Nivel de depósito:** el depósito tiene una capacidad de 200 l. Para el llenado, la válvula que limita el caudal de entrada es de 2,5 l/s máximo, con flujo infinito de entrada el depósito tardaría 80 s en llenarse, la válvula manual de salida vacía el depósito a razón de 1 l/s, ergo, a capacidad máxima tardaría 200 s en vaciarse.

2.3. Sistema de control

El sistema de control está basado en un sistema simple, centralizado con un SCADA con monitoreo de estación tipo monopuesto, determinista que quiere decir que siempre sabremos lo que tardan los datos en llegar a su destino y no redundante con un solo servidor principal, es decir, sin servidores en los alrededores en espera que asuman la función del servidor principal y realizar los trabajos hasta que el servidor primario se recupere.

Para controlar el sistema cada elemento tendrá sus propios parámetros y se controlarán de forma individual con cada uno de sus respectivos estados y órdenes.

El sistema tendrá dos modos, el modo manual o automático que no podrán estar los dos activos a la vez.

De cara al control manual, el sistema deberá estar en manual y por ello, todos los elementos en este modo. Además, si se quiere manejar un elemento en cuestión deberá estar libre de alarmas.

Respecto al control automático, el sistema deberá estar en automático y por ello, todos los elementos en este modo. Al ser un proceso secuencial la activación de la alarma comporta el paro del proceso hasta que esta se arregle.

En ambos casos, se puede hacer uso de las funciones de servicio y fuera de servicio para los diferentes elementos del sistema para su comprobación en caso de no haber una alarma activa. Si alguno de los elementos funciona incorrectamente o se producen errores en los indicadores, el elemento en cuestión se parará en su proceso para evitar así daños en el sistema. Se incluye un reset en los elementos para inicializarlos de nuevo en caso de aparición de alarmas, y una vez debidamente solucionadas deberán ser reconocidas por el operario para continuar con los procesos.

2.3.1. Equipos

Con la finalidad de facilitar la identificación de los elementos del sistema de forma completa, es necesario utilizar un criterio lógico común, no ambiguo para la codificación y nomenclatura de los diferentes elementos que intervienen en el proceso de producción de aceite de oliva.

Con ello se minimizarán los tiempos de comprensión pudiendo trabajar de una manera más cómoda, rápida y eficaz. A continuación, se detallan los elementos con su respectiva asignación y codificación:

Tabla 1. Tabla de elementos del proceso.

Elemento	Codificación y tag	Unidades
Cinta	CT	1
Tolva	TOLV	1
Prensa	PR	1
Batidora	BAT	1
Decánter	DEC	1
Centrifugadora	CH	1
Sensores	SI	1
	DP	1
	LI_Número	6
Motobombas	EPM_Número	2
	VPM_Número	2
Válvulas todo/nada	V_Número	2
Válvulas reguladoras de caudal	FCV_Número	3
Caudalímetros	FIT_Número	4

En consecuencia podemos identificar fácilmente los elementos como se observa a continuación:

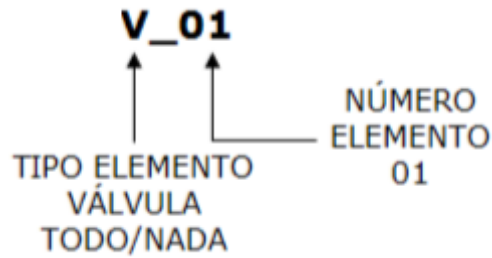


Figura 4. Ejemplo de identificación de elemento válvula número uno.

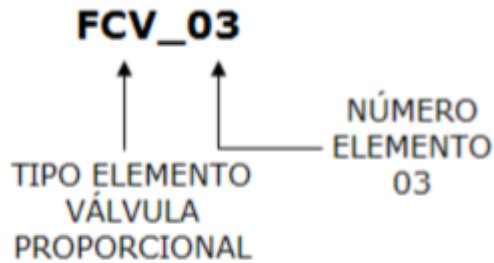


Figura 5. Ejemplo de identificación de elemento válvula proporcional número tres.

2.3.2. Fases del proceso

Para entender mejor el proceso y verlo esquemáticamente se ha creado un diagrama de flujo que comprende todas las fases del proceso.

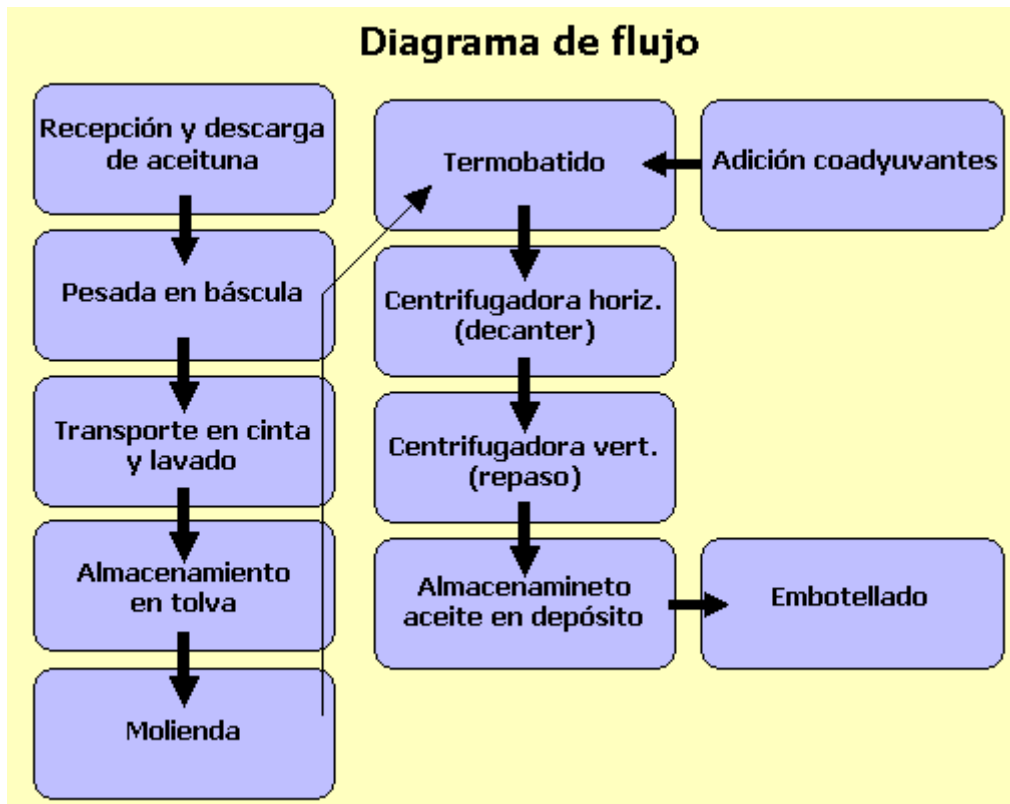


Figura 6. Diagrama de flujo del proceso de producción de aceite de oliva en una almazara.

2.4. Especificaciones funcionales

Una especificación funcional es un documento que detalla los requisitos, normalmente hechas por un cliente, para una aplicación, en este caso aplicaciones de controlador y SCADA. En estas se detalla de manera clara lo que se quiere de una aplicación y se proporcionen los detalles de cuál es su funcionamiento, como se usa o como se ve.

Es más fácil cambiar la funcionalidad en un documento MS Word, o volver a preparar las pantallas HMI que re-programar una aplicación completa SCADA en caso de el cliente o usuario final no esté satisfecho con la totalidad del proyecto.

Partiendo de las especificaciones funcionales, el trabajo del ingeniero es detallar de una forma concisa, completa y estructurada tanto los requisitos funcionales como los de diseño de la solución a desarrollar.

2.4.1. Requerimientos funcionales

Al ser un proyecto que no contempla ningún cliente y está realizado desde la nulidad, se ha cogido como especificación funcional el conjunto de las fases y el modelo del proceso de una planta de producción de aceite real. A partir de este conjunto, se ha elaborado los requisitos funcionales que se encuentran recogidos en la siguiente tabla:

Tabla 2. Tabla de especificaciones de requisitos funcionales.

Requisitos Funcionales (RQF)	
ID	Descripción
RQ_F01	Inicialmente el proceso estará en modo manual y en modo paro.
RQ_F02	Inicialmente todas las alarmas estarán desactivadas.
RQ_F03	Cada elemento ha de tener estados de paro, marcha, servicio, fuera de servicio, manual, automático o alarma.
RQ_F04	Un elemento no podrá estar en dos estados contrarios.
RQ_F05	La marcha de un elemento en automático solo será posible siempre que esté sin alarmas, en servicio y en modo automático.
RQ_F06	La marcha de un elemento en manual solo será posible siempre que esté sin alarmas, en servicio y en modo manual.
RQ_F07	Las alarmas se quedaran enclavadas hasta que no se ordene su reseteo.

RQ_F08	Si un elemento se encuentra en estado de alarma, este dejará de funcionar y se pondrá inmediatamente en el estado fuera de servicio.
RQ_F09	Si se da la orden de apertura, o cierre, a una electroválvula y no se abre, o cierra, en 10 s se activará una alarma de posición.
RQ_F10	Si se da la orden de apertura, o cierre, a una válvula y no se abre, o cierra, se activará una alarma de posición.
RQ_F11	En modo automático las electroválvulas reguladoras se abrirán a su máximo caudal permitido.
RQ_F12	En los procesos que se traslade un flujo másico o líquido se dispondrán de caudalímetros.
RQ_F13	Se contabilizarán las horas de marcha y totales de toda la maquinaria utilizada en el proceso.
RQ_F14	Los depósitos que contengan masa o líquido tendrán sensores de nivel que activarán una alarma en caso de que rebose.
RQ_F15	El depósito de almacenamiento dispondrá de una válvula manual de vaciado.
RQ_F16	El nivel del depósito de almacenamiento se actualizará al final de cada proceso.
RQ_F17	El rango de consigna de los valores de entrada aceituna reside entre: [0 kg, 100kg].
RQ_F18	El rango de consigna de los valores de la válvula que regula el flujo másico del motor de impulsión vertical PR-BAT reside entre: [0 kg/s, 5 kg/s].
RQ_F19	El rango de consigna de los valores de la válvula que regula el flujo líquido del motor de impulsión vertical CH-DP reside entre: [0 l/s, 2,5 l/s].

2.4.2. Requerimientos de diseño

Al igual que los requisitos funcionales, los requisitos de diseño han sido elaborados a partir del conjunto de las fases y el modelo del proceso de producción de aceite de oliva real. Los requisitos de diseño están recogidos en la siguiente tabla:

Tabla 3. Tabla de especificaciones de requisitos de diseño.

Requisitos de Diseño (RQD)	
ID	Descripción
RQ_D01	Todos los elementos integrados están representados en el SCADA y HMI.
RQ_D02	La aplicación dispone de una pantalla principal donde es visible todo el proceso.
RQ_D03	Desde la pantalla principal es posible la selección del modo manual y automático, así como el servicio o fuera de servicio del sistema.
RQ_D04	Las pantallas disponen de marco superior con fecha y hora visible en todo momento.
RQ_D05	La aplicación dispone de una pantalla de alarmas con su respectivo histórico de alarmas y los botones pertenecientes para el reconocimiento de estas.
RQ_D06	La pantalla principal dispone de un sumario de alarmas críticas y estado.
RQ_D07	Los estados de alarmas son visibles visualmente en la maquinaria empleada.
RQ_D08	La aplicación dispone de históricos de datos y a tiempo real.
RQ_D09	Es posible identificar el proceso así como el autor del trabajo de forma clara.

RQ_D10	La aplicación dispone de pantallas tipo pop-up, que evitan quitar visibilidad al proceso de la pantalla principal, estas son debidamente identificadas.
RQ_D11	Cada elemento tiene una ventana tipo pop-up independiente donde se muestra sus estados, órdenes a dar y se abre al hacer clic sobre el elemento gráfico en cuestión.
RQ_D12	La aplicación dispone de un control de seguridad con usuario y contraseña. Inicialmente puede ser desactivada para mejorar la rapidez de las demostraciones y simulaciones.
RQ_D13	Las pantallas principales y secundarias disponen de marco superior y de los botones necesarios para enlazar las diferentes pantallas sinópticas.
RQ_D14	Los colores siguen un criterio de diseño lógico. El color verde representa el estado de marcha, el amarillo de indeterminado, el rojo de alarma, el azul de standby y el gris de paro.
RQ_D15	Se tiene en cuenta el contraste de colores para una fácil comprensión.
RQ_D16	Las representaciones simbólicas contienen animaciones graficas que indican los estados y siguen un criterio de diseño estándar lógico.
RQ_D17	Los tags deberán seguir una identificación intuitiva y de sentido común, siguiendo correctamente el fichero de intercambio y acordes con el controlador.

2.5. Metodología de desarrollo

En la automatización de una almazara hay que tener en cuenta que intervienen muchos elementos y factores. Para desarrollar el proyecto de forma satisfactoria se seguirán los siguientes pasos:

1. **Búsqueda y estudio de proceso de producción de aceite de oliva**, ya que el proyecto se ha de asimilar lo máximo posible a la realidad y darle un acabado profesional, supone un estudio exhaustivo de cada fase del proceso, de las operaciones que se realizan, que maquinas intervienen y su finalidad.
2. **Descripción del proceso y elementos que componen el sistema**, acabado el estudio será necesario plantear todos los elementos que intervendrán en el automatismo así como describir el proceso a realizar.
3. **Diseño del controlador**, se trata de definir los controles necesarios de cada elemento existente dotándolos de sus características y respectivas entradas, salidas, consignas, estados y alarmas.
4. **Diseño de la HMI y del sistema SCADA**, se trata de determinar las representaciones necesarias de cada elemento existente dotándolos de sus características y respectivas variables reales, consignas, estados y alarmas.
5. **Comunicaciones controlador-SCADA**, la comunicación entre softwares se ha tomado como punto de control o checkpoint para comprobar el controlador responde a las órdenes del SCADA empleando las entradas y salidas, de la misma manera que este último representa correctamente todos los estados y cambios en las variables reales.
6. **Diseño de la planta**, se precisa la estructura de simulación de la planta indicando los comportamientos de tal forma que sean válidos y poder observar el funcionamiento en el conjunto de simulaciones posterior.
7. **Simulaciones y verificación**, es el conjunto de pruebas que comprueban que el automatismo diseñado funciona correctamente y el comportamiento de este es el esperado.

Con esta metodología se pretende optimizar el proceso de depuración de fallos. Por un lado, comprobando los funcionamientos y comportamientos del sistema automatizado, y por otro, haciendo localizable los fallos fácilmente, verificando los elementos uno a uno.

2.6. Planificación de tareas

La planificación de tareas para la realización y confección de este proyecto se muestra en la siguiente tabla con su respectivo diagrama de Gantt.

Tabla 4. Tabla de planificación de las tareas del proyecto para el desarrollo del diagrama Gantt.

ACTIVIDAD	INICIO	FIN	DÍAS
Matriculación del proyecto	11/02/2019	14/02/2019	3
Búsqueda de información sobre el proceso a automatizar	01/03/2019	08/03/2019	7
Diseño y planificación de fases del proyecto	09/03/2019	16/04/2019	38
Análisis de los elementos que intervienen en el sistema	17/04/2019	24/04/2019	7
Diseño del fichero de intercambio, requerimientos funcionales y diseño	25/04/2019	06/05/2019	11
Creación de tags, tipos de elemento y definición de I/O	14/05/2019	21/05/2019	7
Programación de elementos y subrutinas en modo manual	22/05/2019	29/05/2019	7
Diseño del sistema SCADA	07/06/2019	19/06/2019	12
Establecimiento de comunicaciones	19/06/2019	24/06/2019	5
Verificación de subrutinas en modo manual	25/06/2019	30/06/2019	5
Programación de elementos y subrutinas en modo automático	01/07/2019	05/07/2019	4

Creación de la planta de simulación	06/07/2019	06/08/2019	31
Simulación y verificación de subrutinas en modo automático y planta simulada	07/08/2019	14/08/2019	7
Corrección de errores	15/08/2019	01/09/2019	17
Implementación de mejoras en SCADA y controlador	02/09/2019	05/09/2019	3
Simulación y verificación de subrutinas y planta simulada definitiva	06/09/2019	08/09/2019	2
Redacción de memoria, anexos y memoria económica	09/09/2019	22/09/2019	13
Montaje final del proyecto	23/09/2019	30/09/2019	7
Depósito y entrega del proyecto	30/09/2019	02/10/2019	2

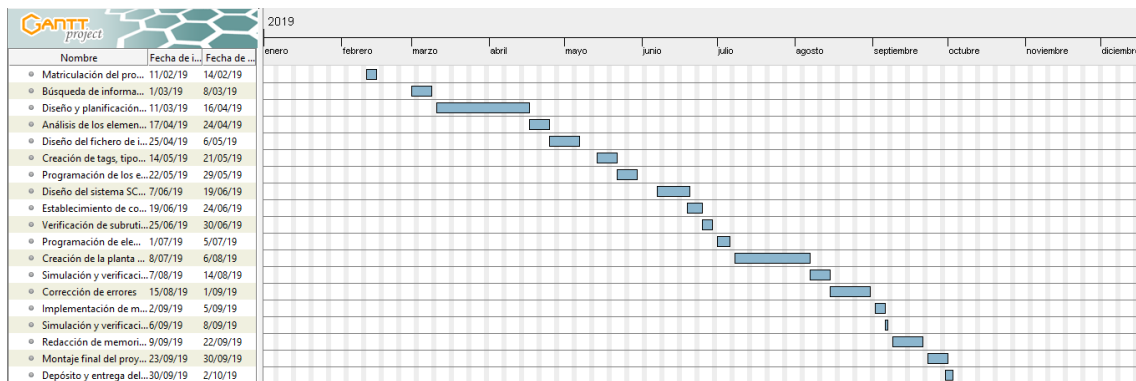


Figura 7. Diagrama Gantt del proyecto.

2.7. Recursos

Los recursos utilizados en el desarrollo son un PC con conexión a internet para poder consultar los manuales de usuario e información de los softwares, el software VMware Workstation 15 Player y una máquina virtual con el software RSLogix 5000 Enterprise, RSLogix Emulate 5000, RSLinx Classic, InTouch WindowMaker.

CAPÍTULO 3: DISEÑO E IMPLEMENTACIÓN DE LA SOLUCIÓN



3. Diseño e implementación de la solución

En el presente capítulo se detalla la arquitectura implementada en el sistema de control y la descomposición del problema en otros diferentes más sencillos para llegar a una solución final correcta y adecuada al planteamiento inicial. También se hace referencia a la codificación utilizada, se define la interfaz de control y se detalla el fichero de intercambio que permite relacionar los controles de las interfaces con la lógica del PLC.

3.1. Arquitectura del sistema de control

3.1.1. Hardware del sistema

Este apartado describe el hardware mínimo necesario que requiere el sistema para hacer funcionar el proyecto:

- **1x Standard x86 bits compatible Personal Computer**
 - Sistema Operativo: Windows 10 (x64) o versiones posteriores
 - Procesador: Intel® Core™ i3 CPU@ 2.00 GHz o superior
 - Memoria: 8 GB RAM o superior
 - Gráficos: Intel® HD Graphics 520 o superior
 - Versión DirectX: DirectX 11
 - Red: Ethernet para comunicación entre PLC-SCADA
 - Almacenamiento: 42 GB libres para máquina virtual

3.1.2. Software del sistema

El software utilizado ha sido variado, en este caso en particular se ha utilizado:

- **VMware Workstation 15 Player.**
Software utilizado para emular una máquina virtual que encapsula un sistema operativo Windows 7 (x64) y sus aplicaciones.
- **RSLogix 5000 Enterprise Series v20.01.00 (CPR 9 Service Release 5)**
Software de Rockwell Allen-Bradley dedicado al diseño, configuración y programación de PLCs.

- **RSLogix Emulate 5000 Chassis Monitor v20.01.00 (CPR 9 Service Release 5)**
Software de Rockwell Allen-Bradley dedicado a la creación de emuladores de PLCs y virtualizaciones de la CPU del PLC y tarjetas I/O.
 - **2x PLC 1789-SIM 32 Point Input/Output Simulator**
Estos PLCs son simulaciones que tienen una cantidad de 32 entradas y 32 salidas configurados con un ciclo de scan (muestreo) de 50 ms.

	Assembly Instance	Size
Input	1	2
Output	2	1
Configuration	16	0

Figura 8. Configuración de los parámetros del PLC en RSLogix 5000.

- **RSLinx Classic v3.70.00 (CPR 9 Service Release 7)**
Software de Rockwell Allen-Bradley utilizado para establecer comunicación entre controlador y sistema SCADA mediante protocolo DDE y permite una vista rápida de la topología de la red.
- **InTouch WindowMaker v11.1.19 (CPR 2 Service Release 1)**
Software de Wonderware para Schneider Electric que permite la creación de pantallas HMI para la supervisión y el control del proceso.

3.1.3. Vista global de las comunicaciones

Para obtener una vista global de las comunicaciones primero se ha de definir que son los protocolos de comunicación. Básicamente, un protocolo de comunicación es un sistema de reglas que permiten que dos o mas entidades de un sistema de comunicación se informen entre sí para transmitir información por medio de cualquier tipo de variación de una magnitud física.

Los protocolos de comunicación de la red se pueden dividir en varias categorías, una de las clasificaciones más estudiadas es la del modelo OSI (Open System Interconnection).

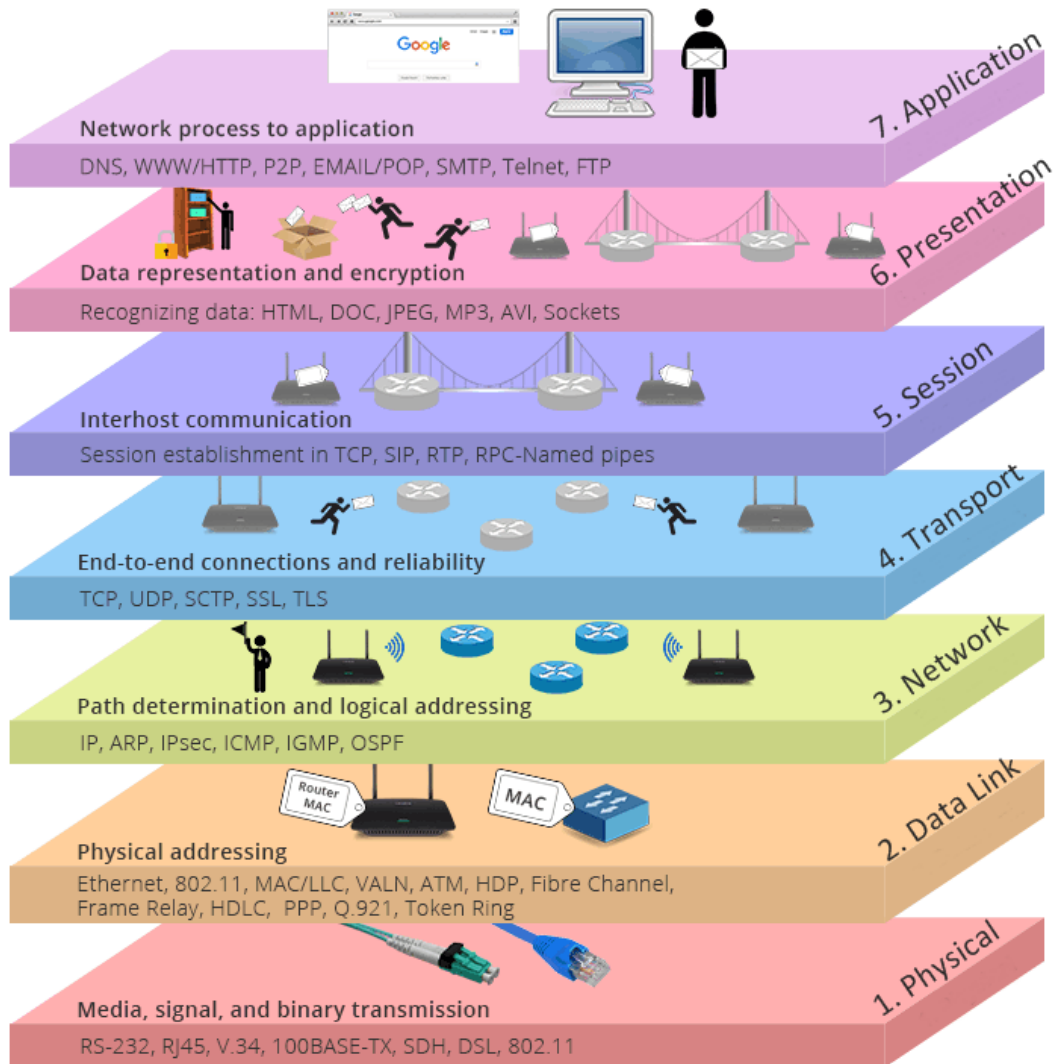


Figura 9. Modelo OSI para los protocolos de la red. [Cortesía: @xxxamin1314 en medium.com]

3.1.4. Comunicaciones SCADA-Controlador-Proceso

Para las simulaciones y la comunicación SCADA-PLC se ha utilizado una tecnología de comunicación DDE (Dynamic Data Exchange), situada en la segunda capa OSI, mediante Ethernet.

Esta tecnología de comunicación es utilizada entre varias aplicaciones bajo los sistemas operativos Microsoft Windows y OS/2. La función principal del DDE es permitir que las aplicaciones de Windows envíen comandos al servidor de aplicaciones y recibir respuestas consiguiendo así que

compartan datos. Por ejemplo, un valor de una variable en SCADA que está vinculada directamente a una aplicación diferente como es RSLogix 5000, cuando el valor de la variable cambie, se actualizará automáticamente viéndose reflejada en ambos sitios.

La comunicación de datos se establece mediante un modelo simple de tres segmentos. Cada programa es conocido por DDE por su nombre de aplicación (application name), que a la vez organizan la información en grupos conocidos como temas o tópicos (topics) y finalmente, estos últimos organizan la información en una categoría más pequeña de datos individuales llamados objetos o elementos (items).

Para realizar la comunicación, en primer lugar dentro de InTouch se accede a *Tools/configure/Access Names* y se crea un *Access Name* tipo DDE.

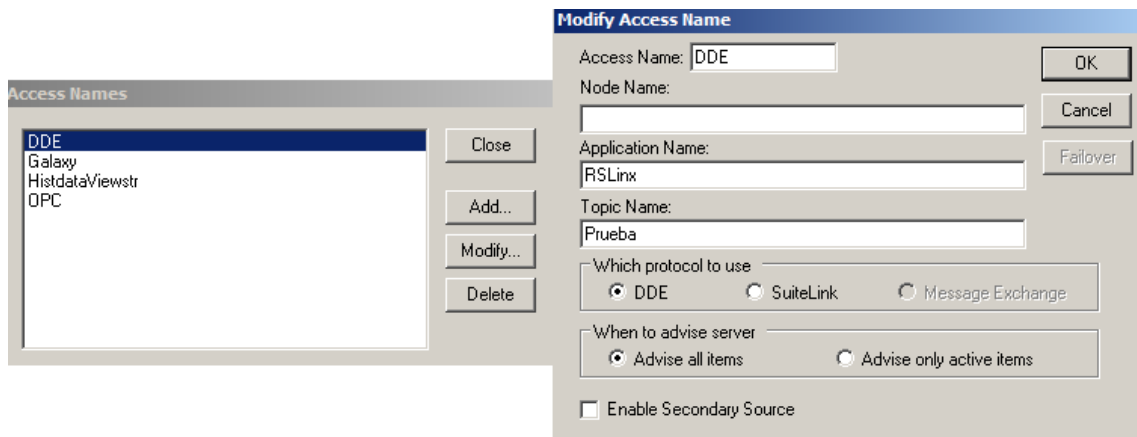


Figura 10. Creación del Access Name tipo DDE.

Con el DDE configurado en el SCADA, por a lo que a InTouch respecta, solo es necesario configurar los tags existentes para que se puedan comunicar.

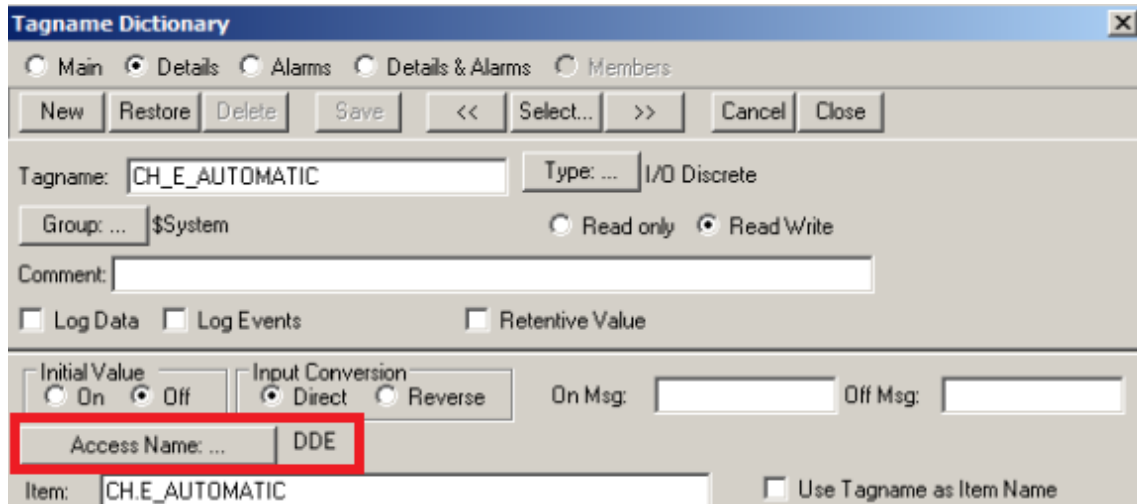


Figura 11. Ejemplo de configuración de un tag para un Access Name.

En el entorno de InTouch no se admiten tags con puntos en su nomenclatura y se sustituirán manualmente por guiones bajos. Seguidamente dentro de RSLogix Emulate se crea un módulo de emulación y dos módulos de PLCs simulados con 32 I/O cada uno.

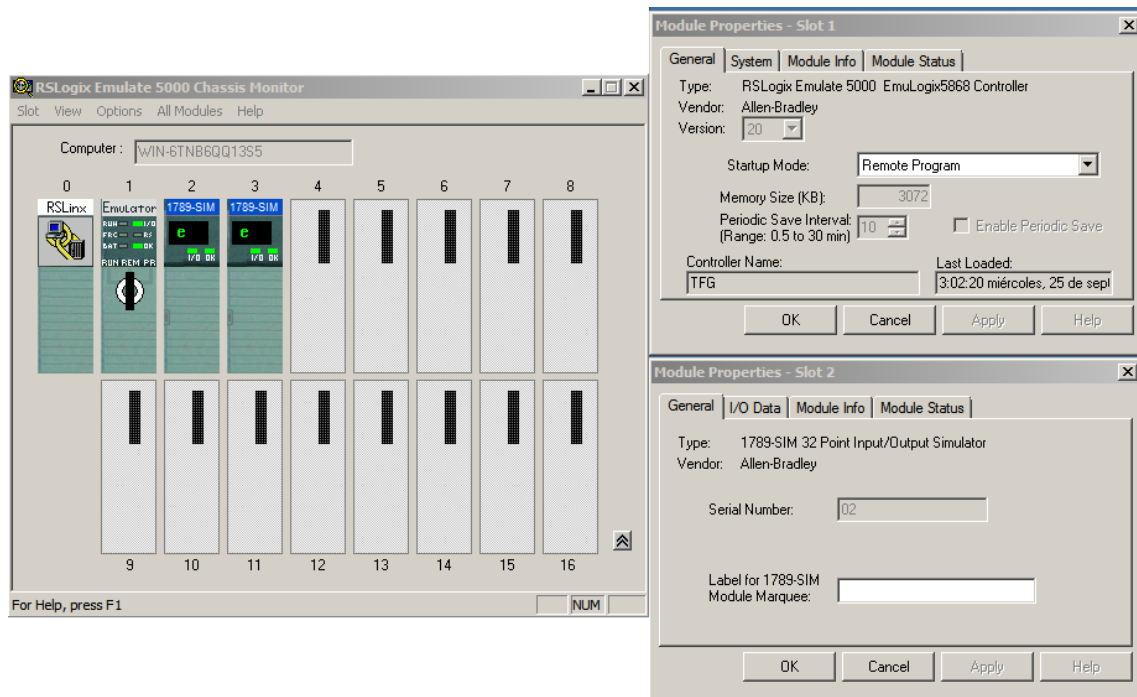


Figura 12. Inserción del módulo de emulación y PLCs simulados en el chasis de la virtualización.

Finalmente, se configura el driver *Virtual Backplane* con los parámetros de serie y el protocolo DDE en RSLinx creando un *Topic Configuration* nuevo y vinculándolo a la rutina de la CPU emulada junto con el DDE previamente creado.

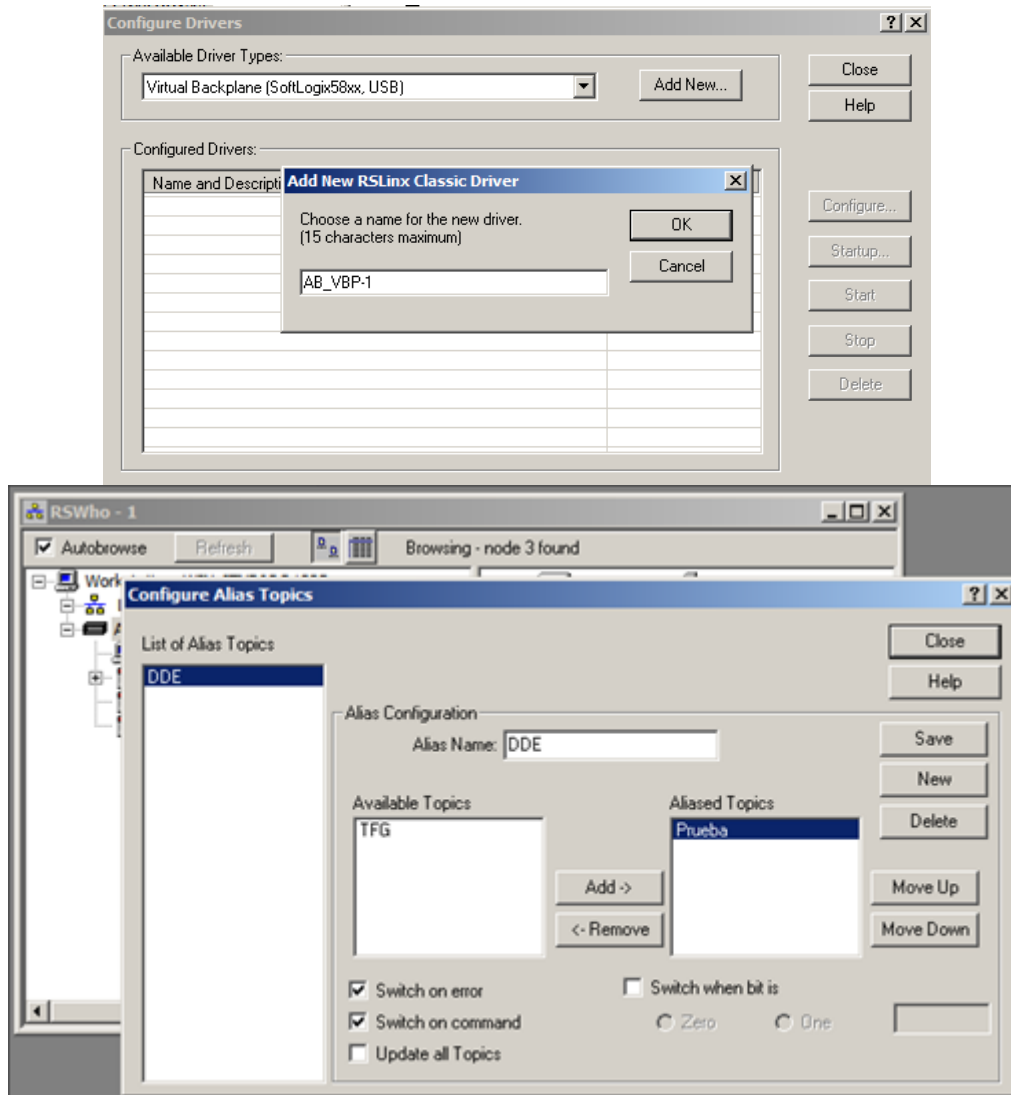


Figura 13. Configuración del driver virtual [arriba] y vinculación del topic al DDE previamente configurado [abajo].

Con todas las comunicaciones configuradas, se puede cargar el programa al PLC virtual (download), establecer su modo de operación online y ejecutarlo (run) mediante RSLogix junto con la ejecución de la aplicación SCADA (runtime) para comprobar el correcto funcionamiento e interacción con el PLC virtual.

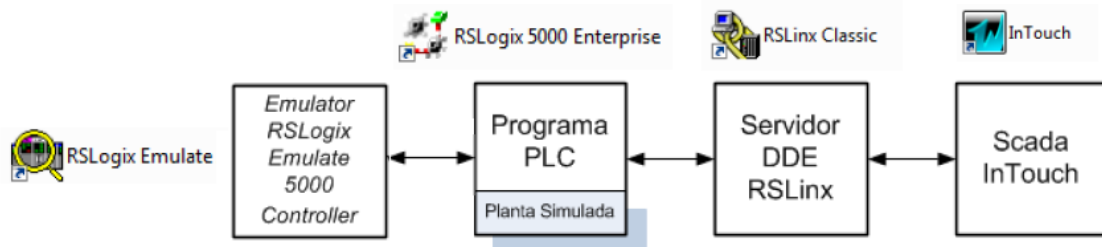


Figura 14. Esquema de comunicación global del proyecto.

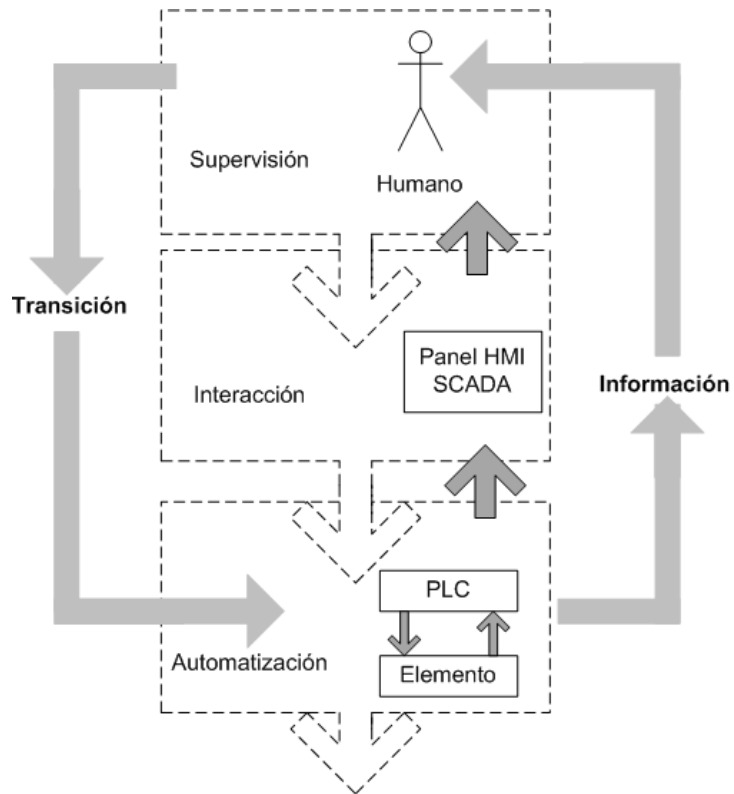


Figura 15. Comunicación controlador-SCADA-proceso.

3.2. Codificación de los elementos

Cuando se habla de tags, simplemente se hace referencia a la etiqueta pudiendo ser una señal, un elemento o máquina que hay dentro del sistema que se pretende automatizar. La codificación de los elementos sigue un estándar de lógica común: *NombreElemento_Número* y se puede encontrar en la siguiente tabla:

Tabla 5. Listado de elementos que intervienen en el proceso con su respectivo tag de codificación.

Listado de Elementos			
Ítem	Tag elemento	Tipo elemento	Descripción
1	CT	M	Cinta
2	EPM_01	M	Motor centrifugo de bombeo de agua externa 1
3	EPM_02	M	Motor centrifugo de bombeo de agua externa 2
4	VPM_01	M	Motor bomba de impulsión vertical de prensa hacia batidora
5	VPM_02	M	Motor bomba de impulsión vertical de centrifugadora hacia depósito
6	TOLV	M	Tolva
7	SI	S	Sensor input cinta
8	LI_01	S	Sensor tolva high+
9	LI_02	S	Sensor tolva high
10	LI_03	S	Sensor tolva low
11	LI_04	S	Sensor batidora high+
12	LI_05	S	Sensor batidora high
13	LI_06	S	Sensor batidora low
14	DP	S	Sensor depósito
15	V_01	V	Válvula todo/nada input decánter
16	V_02	V	Válvula todo/nada output decánter
17	FCV_01	FCV	Input aceitunas (válvula reguladora de caudal)
18	FCV_03	FCV	Válvula reguladora de caudal hacia batidora
19	FCV_04	FCV	Válvula reguladora de caudal hacia depósito
20	FIT_01	IT	Caudalímetro input aceitunas
21	FIT_02	IT	Caudalímetro agua para lavado
22	FIT_03	IT	Caudalímetro de prensa hacia batidora
23	FIT_04	IT	Caudalímetro de centrifugadora hacia depósito
24	PR	PR	Prensa
25	BAT	BAT	Batidora
26	DEC	DEC	Decánter
27	CH	CH	Centrifugadora

Un elemento puede tener muchas señales ya sea de órdenes, de estados, de entradas o salidas, etc. y para diferenciarlas se ha escogido un sistema de codificación natural, estas siguen el modelo *Elemento.TipoSeñal_NombreSeñal*. Dónde el tipo de señal es una inicial que indica si se trata de un estado (E), una orden (C) exceptuando alguna señal, una entrada física (X) o una salida física (Y).

La siguiente tabla demuestra cómo se ha realizado dicha codificación:

Tabla 6. Codificación de las diferentes señales que puede tener un elemento.

Tag Señal	Codificación
Estado	<i>Elemento.E_NombreSeñal.</i>
Orden	<i>Elemento.C_NombreSeñal.</i>
Señal Entrada	<i>Elemento.XNombreSeñal.</i>
Señal Salida	<i>Elemento.YNombreSeñal.</i>
Temporizador	<i>Elemento.T_NombreSeñal.</i>
Contador	<i>Elemento.C_NombreSeñal.</i>

Por ejemplo, FCV_03.E_MANUAL hace referencia al tipo de elemento válvula proporcional número 3 en estado manual. Así como, DEC.C_OSERVICE se refiere al tipo de elemento decánter en orden de fuera de servicio, o VPM_01.YSTART, que es el tipo de elemento motobomba con señal de salida de arranque.

3.3. Descomposición del problema de control

Por simplicidad, todos los elementos serán programados por separado y aquellos que sean múltiples serán indexados para que cada uno trabaje con la misma rutina, evitando así hacer rutinas dos veces de algo que tiene el mismo comportamiento.

Por otro lado, se ha dividido el sistema de control en cuatro procesos más pequeños cada uno con un solo elemento máquina (PR, BAT, DEC, CH).

-El primero comprende desde la entrada de aceituna hasta el molido en prensa (tramo Input-PR), en orden, formado por los elementos: FCV_01, FIT_01, SI, EPM_01, EPM_02, FIT_02, CT, TOLV, PR.

-El segundo abarca desde la salida de masa de la prensa hasta el termobatido en la batidora (tramo PR-BAT), en orden, contiene los elementos: VPM_01, FCV_03, FIT_03, BAT.

-El tercer incluye de la salida de la pasta de la batidora hasta la entrada del aceite en la centrifugadora sin incluirla (tramo BAT-DEC), en orden, reúne los elementos: V_01, DEC, V_02.

-El cuarto y último implica desde centrifugadora hasta el almacenaje en el depósito (tramo CH-DP), en orden, implica los elementos: CH, VPM_02, FCV_04, FIT_04, DP.

3.4. Definición del fichero de intercambio Controlador-SCADA

El fichero de intercambio es aquel archivo que contiene los tags de las señales de cada elemento e indica las relaciones de entradas/salidas del controlador PLC-SCADA. De esta manera es fácil reconocer qué es cada tag y su función.

 UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH	TFG
SUPERVISIÓN, CONTROL Y SIMULACIÓN DE UNA PLANTA DE PRODUCCIÓN DE ACEITE DE OLIVA	Autor: Sergio Casas Gallegos
Titulación: Grado en Ingeniería Electrónica Industrial y Automática	Profesor: Javier Gámez

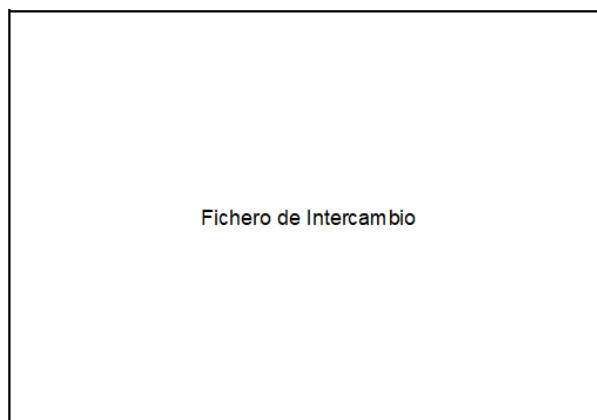


Figura 16. Portada del fichero de intercambio.

Tabla 7. Elementos tipo motor con listado e información de todas las señales correspondientes.

Tipo Motor: VPM, EPM, CT, TOLV				PLC			SCADA		
Ítem	Descripción	Tag señal	Tipo	E	E/S	S	E	E/S	S
1	Estado con alarmas	E_ALARM	BOOL			1	1		
2	Estado automático	E_AUTOMATIC	BOOL			1	1		
3	Estado fuera de servicio	E_OSERVICE	BOOL			1	1		
4	Estado manual	E_MANUAL	BOOL			1	1		
5	Estado de servicio	E_SERVICE	BOOL			1	1		
6	Estado en marcha	E_RUNNING	BOOL			1	1		
7	Orden de paro en automático	C_STOP_AUTO	BOOL	1					1
8	Orden de paro	C_STOP	BOOL	1					1
9	Orden de automático	C_AUTOMATIC	BOOL	1					1
10	Orden de manual	C_MANUAL	BOOL	1					1
11	Orden de fuera de servicio	C_OSERVICE	BOOL	1					1
12	Orden de reset	C_RESET	BOOL	1					1
13	Orden de servicio	C_SERVICE	BOOL	1					1
14	Orden de marcha	C_START	BOOL	1					1
15	Orden de marcha automático	C_START_AUTO	BOOL	1					1
16	Señal de entrada de alarma	XALARM	BOOL	1					
17	Señal de entrada de marcha	XRUNNING	BOOL	1					
18	Señal de salida de reset de alarma	YRESET	BOOL			1			
19	Señal de salida de marcha	YSTART	BOOL			1			
20	Señal de salida de paro	YSTOP	BOOL			1			
21	Tiempo de reset	T_RESET	TIMER						
22	Tiempo en funcionamiento	T_START	TIMER						

Tabla 8. Elementos tipo sensor con listado e información de todas las señales correspondientes.

Tipo Sensor: SI, DP, LI				PLC			SCADA		
Ítem	Descripción	Tag señal	Tipo	E	E/S	S	E	E/S	S
1	Señal de entrada de detección	XDETECTA	BOOL	1					
2	Estado de detección	E_DETECTA	BOOL			1	1		

Tabla 9. Elementos tipo válvula con listado e información de todas las señales correspondientes.

Tipo Válvula todo/nada: V				PLC			SCADA		
Ítem	Descripción	Tag señal	Tipo	E	E/S	S	E	E/S	S
1	Orden de automático	C_AUTOMATIC	BOOL	1					1
2	Orden de cierre	C_CLOSE	BOOL	1					1
3	Orden de cierre en automático	C_CLOSE_AUTO	BOOL	1					1
4	Orden de manual	C_MANUAL	BOOL	1					1
5	Orden de apertura	C_OPEN	BOOL	1					1
6	Orden de apertura en automático	C_OPEN_AUTO	BOOL	1					1
7	Orden de reset	C_RESET	BOOL	1					1
8	Orden de fuera de servicio	C_OSERVICE	BOOL	1					1
9	Consigna de posición	C_POSITION	BOOL	1					1
10	Orden de servicio	C_SERVICE	BOOL	1					1
11	Estado con alarmas	E_ALARM	BOOL			1	1		
12	Estado en automático	E_AUTOMATIC	BOOL			1	1		
13	Estado de cierre	E_CLOSED	BOOL			1	1		
14	Estado manual	E_MANUAL	BOOL			1	1		
15	Estado de apertura	E_OPEN	BOOL			1	1		
16	Estado de fuera de servicio	E_OSERVICE	BOOL			1	1		
17	Estado de servicio	E_SERVICE	BOOL			1	1		
18	Alarma posición	A_POSITION	BOOL			1	1		
19	Estado de final de carrera abierto	E_FCO	BOOL			1	1		
20	Estado de final de carrera cerrado	E_FCT	BOOL			1	1		
21	Señal de entrada de apertura	XOPEN	BOOL	1					
22	Señal de entrada de cierre	XCLOSED	BOOL	1					
23	Señal de entrada de alarma	XALARM	BOOL	1					
24	Señal de salida de apertura	YOPEN	BOOL			1			
25	Señal de salida de cierre	YCLOSE	BOOL			1			
26	Señal de salida de reset	YRESET	BOOL			1			
27	Temporizador de cierre	T_CLOSE	TIMER						
28	Temporizador de apertura	T_OPEN	TIMER						
29	Temporizador de reset	T_RESET	TIMER						

Tabla 10. Elemento tipo válvula reguladora con listado e información de todas las señales correspondientes.

Tipo Válvula Reguladora: FCV				PLC			SCADA		
Ítem	Descripción	Tag señal	Tipo	E	E/S	S	E	E/S	S
1	Alarma posición	A_POSITION	BOOL			1	1		
2	Estado con alarmas	E_ALARM	BOOL			1	1		
3	Estado en automático	E_AUTOMATIC	BOOL			1	1		
4	Estado de fuera de servicio	E_OSERVICE	BOOL			1	1		
5	Estado manual	E_MANUAL	BOOL			1	1		
6	Estado en servicio	E_SERVICE	BOOL			1	1		
7	Consigna de posición en automático	C_POSITION_AUTO	REAL			1			
8	Consigna de posición	C_POSITION	REAL	1					1
9	Orden de automático	C_AUTOMATIC	BOOL	1					1
10	Orden de fuera de servicio	C_OSERVICE	BOOL	1					1
11	Orden de manual	C_MANUAL	BOOL	1					1
12	Orden de servicio	C_SERVICE	BOOL	1					1
13	Orden fuera de servicio	C_RESET	BOOL	1					1
14	Señal de salida de posición	YPOSITION	REAL			1			
15	Temporizador de cierre	T_CLOSE	TIMER						
16	Temporizador de apertura	T_OPEN	TIMER						

Tabla 11. Elementos tipo medidor con listado e información de todas las señales correspondientes.

Tipo Medidor: FIT, TIT, LIT				PLC			SCADA		
Ítem	Descripción	Tag señal	Tipo	E	E/S	S	E	E/S	S
1	Estado de alarma del medidor	E_ALARM	BOOL			1	1		
2	Medida	MINS	REAL			1	1		
3	Señal de entrada del medidor	XMINS	REAL	1					
4	Señal de entrada de alarma del medidor	XALARM	BOOL	1					

Tabla 12. Elemento tipo prensa con listado e información de todas las señales correspondientes.

Tipo Prensa: PR				PLC			SCADA		
Ítem	Descripción	Tag señal	Tipo	E	E/S	S	E	E/S	S
1	Estado de presión	E_PRESSIO	REAL			1	1		
2	Estado disminución de presión	E_DESINFL	BOOL			1	1		
3	Estado de aumento de presión	E_INFL	BOOL			1	1		
4	Estado con alarmas	E_ALARM	BOOL			1	1		
5	Estado automático	E_AUTOMATIC	BOOL			1	1		
6	Estado de servicio	E_OSERVICE	BOOL			1	1		
7	Estado manual	E_MANUAL	BOOL			1	1		
8	Estado de marcha	E_RUNNING	BOOL			1	1		
9	Estado de servicio	E_SERVICE	BOOL			1	1		
10	Horas de marcha	RHOURS	DINT			1	1		
11	Horas totales	THOURS	DINT			1	1		
12	Orden de presión en automático	C_PRESSIO_AUTO	BOOL			1			
13	Orden de presión	C_PRESSIO	REAL	1					1
14	Orden de automático	C_AUTOMATIC	BOOL	1					1
15	Orden de marcha	C_RUNNING	BOOL	1					1
16	Orden de manual	C_MANUAL	BOOL	1					1
17	Orden fuera de servicio	C_OSERVICE	BOOL	1					1
18	Orden de reset	C_RESET	BOOL	1					1
19	Orden de servicio	C_SERVICE	BOOL	1					1
20	Orden de aumento de presión	C_INFL	BOOL	1					1
21	Orden de disminución	C_DESINFL	BOOL	1					1
22	Orden de paro	C_STOP	BOOL	1					1
23	Orden de marcha automático	C_START_AUTO	BOOL			1			
24	Orden de paro automático	C_STOP_AUTO	BOOL			1			
25	Señal de entrada de marcha	XRUNNING	BOOL	1					
26	Señal de entrada de alarma	XALARM	BOOL	1					
27	Señal de entrada analógica de potencia eléctrica	XPOWER	REAL			1	1		
28	Señal de entrada de disminución de presión	XDESINFL	BOOL	1					
29	Señal de entrada de aumento de presión	XINFL	BOOL	1					
30	Señal de salida de marcha	YSTART	BOOL			1			
31	Señal de salida de reset	YRESET	BOOL			1			
32	Señal de salida de presión	YPRESSIO	REAL			1			
33	Señal de salida de disminución de presión	YDESINFL	BOOL			1			

34	Señal de salida de aumento de presión	YINFL	BOOL			1			
35	Temporizador de horas de marcha	T_HRUNNING	TIMER						
36	Temporizador de horas totales	T_HTOTAL	TIMER						
37	Contador de horas de marcha	C_HRUNNING	COUNTER						
38	Contador de horas totales	C_HTOTAL	COUNTER						
39	Potencia eléctrica	POWER	REAL	1					
40	Cantidad input	C_INPUT	REAL			1	1		
41	Cantidad convertir	C_CONVERSE	REAL			1	1		
42	Cantidad output	C_OUTPUT	REAL			1	1		

Tabla 13. Elemento tipo batidora con listado e información de todas las señales correspondientes.

Tipo Batidora: BAT				PLC			SCADA		
Ítem	Descripción	Tag señal	Tipo	E	E/S	S	E	E/S	S
1	Estado con alarmas	E_ALARM	BOOL			1	1		
2	Estado automático	E_AUTOMATIC	BOOL			1	1		
3	Estado fuera de servicio	E_OSERVICE	BOOL			1	1		
4	Estado manual	E_MANUAL	BOOL			1	1		
5	Estado de marcha	E_RUNNING	BOOL			1	1		
6	Estado de servicio	E_SERVICE	BOOL			1	1		
7	Estado de preparación	E_READY	BOOL			1	1		
8	Estado de temporizador OK	E_TEMPOK	BOOL			1	1		
9	Estado de coadyuvante OK	E_HELPEROK	BOOL			1	1		
10	Potencia eléctrica	POWER	REAL			1	1		
11	Horas de marcha	RHOURS	DINT			1			
12	Horas totales	THOURS	DINT			1			
13	Orden de paro	C_STOP	BOOL	1					1
14	Orden de marcha	C_START	BOOL	1					1
15	Orden de automático	C_AUTOMATIC	BOOL	1					1
16	Orden de manual	C_MANUAL	BOOL	1					1
17	Orden de servicio	C_SERVICE	BOOL	1					1
18	Orden de reset	C_RESET	BOOL	1					1
19	Orden de fuera de servicio	C_OSERVICE	BOOL	1					1
20	Orden de coadyuvante	C_HELPER	BOOL	1					1
21	Orden de temporizador	C_TEMP	BOOL	1					1
22	Orden de paro automático	C_STOP_AUTO	BOOL			1			
23	Orden de temporizador automático	C_TEMP_AUTO	BOOL			1			

24	Orden de marcha en automático	C_START_AUTO	BOOL			1			
25	Orden de coadyuvante en automático	C_HELPER_AUTO	BOOL			1			
26	Señal de entrada analógica de potencia eléctrica	XPOWER	BOOL	1					
27	Señal de entrada de marcha	XRUNNING	BOOL	1					
28	Señal de entrada de alarma	XALARM	BOOL	1					
29	Señal de entrada de temporizador	XTEMP	BOOL	1					
30	Señal de entrada de coadyuvante	XHELPER	BOOL	1					
31	Señal de salida de marcha	YSTART	BOOL			1			
32	Señal de salida de reset	YRESET	BOOL			1			
33	Señal de salida de temporizador	YTEMP	BOOL			1			
34	Señal de salida de coadyuvante	YHELPER	BOOL			1			
35	Temporizador de horas de marcha	T_HRUNNING	TIMER						
36	Temporizador de horas totales	T_HTOTAL	TIMER						
37	Contador de horas de marcha	C_HRUNNING	COUNTER						
38	Contador de horas totales	C_HTOTAL	COUNTER						
39	Cantidad input	C_INPUT	REAL			1	1		
40	Cantidad convertir	C_CONVERSE	REAL			1	1		
41	Cantidad output	C_OUTPUT	REAL			1	1		

(espacio deliberadamente en blanco)

Tabla 14. Elemento tipo decánter con listado e información de todas las señales correspondientes.

Tipo Decánter: DEC				PLC			SCADA		
Ítem	Descripción	Tag señal	Tipo	E	E/S	S	E	E/S	S
1	Estado con alarmas	E_ALARM	BOOL			1	1		
2	Estado automático	E_AUTOMATIC	BOOL			1	1		
3	Estado fuera de servicio	E_OSERVICE	BOOL			1	1		
4	Estado manual	E_MANUAL	BOOL			1	1		
5	Estado de marcha	E_RUNNING	BOOL			1	1		
6	Estado de servicio	E_SERVICE	BOOL			1	1		
7	Estado de inyección de masa OK	E_INJECTOK	BOOL			1	1		
8	Estado preparado para funcionar	E_READY	BOOL			1	1		
9	Potencia eléctrica	POWER	REAL			1	1		
10	Horas de marcha	RHOURS	DINT			1	1		
11	Horas totales	THOURS	DINT			1	1		
12	Orden de paro	C_STOP	BOOL	1					1
13	Orden de marcha	C_START	BOOL	1					1
14	Orden de automático	C_AUTOMATIC	BOOL	1					1
15	Orden de manual	C_MANUAL	BOOL	1					1
16	Orden de servicio	C_OSERVICE	BOOL	1					1
17	Orden de reset	C_RESET	BOOL	1					1
18	Orden de fuera de servicio	C_SERVICE	BOOL	1					1
19	Orden de inyección de masa	C_INJECT	BOOL	1					1
20	Orden de paro automático	C_STOP_AUTO	BOOL			1			
21	Orden de inyección de masa automático	C_INJECT_AUTO	BOOL			1			
22	Orden de marcha automático	C_START_AUTO	BOOL			1			
23	Señal de entrada analógica de potencia eléctrica	XPOWER	REAL	1					
24	Señal de entrada de marcha	XRUNNING	BOOL	1					
25	Señal de entrada de alarma	XALARM	BOOL	1					
26	Señal de entrada de inyección de masa	XINJECT	BOOL	1					
27	Señal de salida de marcha	YSTART	BOOL			1			
28	Señal de salida de reset	YRESET	BOOL			1			
29	Señal de salida de inyección de masa	YINJECT	BOOL			1			
30	Temporizador de horas de marcha	T_HRUNNING	TIMER						
31	Temporizador de horas totales	T_HTOTAL	TIMER						
32	Contador de horas de marcha	C_HRUNNING	COUNTER						

33	Contador de horas totales	C_HTOTAL	COUNTER						
34	Cantidad input	C_INPUT	REAL			1	1		
35	Cantidad convertir	C_CONVERSE	REAL			1	1		
36	Cantidad output	C_OUTPUT	REAL			1	1		

Tabla 15. Elemento tipo centrifugadora con listado e información de todas las señales correspondientes.

Tipo Centrifugadora: CH				PLC			SCADA		
Ítem	Descripción	Tag señal	Tipo	E	E/S	S	E	E/S	S
1	Estado con alarmas	E_ALARM	BOOL			1	1		
2	Estado de decanter OK	E_DECANTOK	BOOL			1	1		
3	Estado automático	E_AUTOMATIC	BOOL			1	1		
4	Estado fuera de servicio	E_OSERVICE	BOOL			1	1		
5	Estado manual	E_MANUAL	BOOL			1	1		
6	Estado de marcha	E_RUNNING	BOOL			1	1		
7	Estado de servicio	E_SERVICE	BOOL			1	1		
8	Estado preparado para funcionar	E_READY	BOOL			1	1		
9	Potencia eléctrica	POWER	REAL			1	1		
10	Horas de marcha	RHOURS	DINT			1	1		
11	Horas totales	THOURS	DINT			1	1		
12	Orden de paro	C_STOP	BOOL	1					1
13	Orden de marcha	C_START	BOOL	1					1
14	Orden de automático	C_AUTOMATIC	BOOL	1					1
15	Orden de manual	C_MANUAL	BOOL	1					1
16	Orden de servicio	C_OSERVICE	BOOL	1					1
17	Orden de reset	C_RESET	BOOL	1					1
18	Orden de fuera de servicio	C_SERVICE	BOOL	1					1
19	Orden de decánter	C_DECANT	BOOL	1					1
20	Orden de paro automático	C_STOP_AUTO	BOOL			1			
21	Orden de decánter automático	C_DECANT_AUTO	BOOL			1			
22	Orden de marcha automático	C_START_AUTO	BOOL			1			
23	Señal de entrada analógica de potencia eléctrica	XPOWER	REAL	1					
24	Señal de entrada de marcha	XRUNNING	BOOL	1					
25	Señal de entrada de alarma	XALARM	BOOL	1					
26	Señal de entrada de decánter	XDECANT	BOOL	1					
27	Señal de salida de marcha	YSTART	BOOL			1			
28	Señal de salida de reset	YRESET	BOOL			1			

29	Señal de salida de decánter	YDECANT	BOOL			1			
30	Temporizador de horas de marcha	T_HRUNNING	TIMER						
31	Temporizador de horas totales	T_HTOTAL	TIMER						
32	Contador de horas de marcha	C_HRUNNING	COUNTER						
33	Contador de horas totales	C_HTOTAL	COUNTER						

(espacio deliberadamente en blanco)

A continuación se detallan los inputs digitales (DI) y analógicos (AI) de las señales de campo que van relacionadas con los inputs de los módulos correspondientes (módulos 2I y 3I).

Tabla 16. Señales de entradas digitales y analógicas de campo del módulo virtual y controlador.

DI0	Module		DI1	Module	
DI0.1	2I.00	CT.XRUNNING	DI1.1	2I.08	BAT.XRUNNING
DI0.2	2I.01	CT.XALARM	DI1.2	2I.09	BAT.XALARM
DI0.3	2I.02	TOLV.XRUNNING	DI1.3	2I.10	BAT.XHELPER
DI0.4	2I.03	TOLV.XALARM	DI1.4	2I.11	BAT.XTEMP
DI0.5	2I.04	PR.XRUNNING	DI1.5	2I.12	DEC.XRUNNING
DI0.6	2I.05	PR.XALARM	DI1.6	2I.13	DEC.XALARM
DI0.7	2I.06	PR.XDESINFL	DI1.7	2I.14	DEC.XINJECT
DI0.8	2I.07	PR.XINFL	DI1.8	2I.15	
DI2	Module		DI3	Module	
DI2.1	2I.16	CH.XRUNNING	DI3.1	2I.24	EPM_01.XRUNNING
DI2.2	2I.17	CH.XALARM	DI3.2	2I.25	EPM_01.XALARM
DI2.3	2I.18	CH.XDECANT	DI3.3	2I.26	EPM_02.XRUNNING
DI2.4	2I.19	FIT_01.XALARM	DI3.4	2I.27	EPM_02.XALARM
DI2.5	2I.20	FIT_02.XALARM	DI3.5	2I.28	VPM_01.XRUNNING
DI2.6	2I.21	FIT_03.XALARM	DI3.6	2I.29	VPM_01.XALARM
DI2.7	2I.22	FIT_04.XALARM	DI3.7	2I.30	VPM_02.XRUNNING
DI2.8	2I.23	VACIADO MAN	DI3.8	2I.31	VPM_02.XALARM
DI4	Module		DI5	Module	
DI4.1	3I.00	V_01.XALARM	DI5.1	3I.08	LI_01.XDETECTA
DI4.2	3I.01	V_01.XOPEN	DI5.2	3I.09	LI_02.XDETECTA
DI4.3	3I.02	V_01.XCLOSE	DI5.3	3I.10	LI_03.XDETECTA
DI4.4	3I.03	V_02.XALARM	DI5.4	3I.11	LI_04.XDETECTA
DI4.5	3I.04	V_02.XOPEN	DI5.5	3I.12	LI_05.XDETECTA
DI4.6	3I.05	V_02.XCLOSE	DI5.6	3I.13	LI_06.XDETECTA
DI4.7	3I.06		DI5.7	3I.14	S.XDETECTA
DI4.8	3I.07		DI5.8	3I.15	DP.XDETECTA
AI0			AI1		
AI0.1		FIT_01.XMINS	AI1.1		AUXLV_TOLVA
AI0.2		FIT_02.XMINS	AI1.2		AUXLV_BAT
AI0.3		FIT_03.XMINS	AI1.3		AUXLV_DP
AI0.4		FIT_04.XMINS	AI1.4		
AI0.5		PR.XPOWER	AI1.5		
AI0.6		BAT.XPOWER	AI1.6		
AI0.7		DEC.XPOWER	AI1.7		
AI0.8		CH.XPOWER	AI1.8		

Consecutivamente, los outputs digitales (DO) y analógicos (AO) de las señales de campo que van relacionadas con los outputs de los módulos correspondientes (módulos 20 y 30).

Tabla 17. Señales de salidas digitales y analógicas de campo del módulo virtual y controlador.

DO0	Module		DO1	Module	
DO0.1	20.00	CT.YSTART	DO1.1	20.08	PR.YSTART
DO0.2	20.01	CT.YSTOP	DO1.2	20.09	PR.YRESET
DO0.3	20.02	CT.YRESET	DO1.3	20.10	PR.YINFL
DO0.4	20.03	TOLV.YSTART	DO1.4	20.11	PR.YDESINFL
DO0.5	20.04	TOLV.YSTOP	DO1.5	20.12	BAT.YSTART
DO0.6	20.05	TOLV.YRESET	DO1.6	20.13	BAT.YRESET
DO0.7	20.06		DO1.7	20.14	BAT.YHELPER
DO0.8	20.07		DO1.8	20.15	BAT.YTEMP
DO2	Module		DO3	Module	
DO2.1	20.16	DEC.YSTART	DO3.1	20.24	EPM_01.YSTART
DO2.2	20.17	DEC.YRESET	DO3.2	20.25	EPM_01.YSTOP
DO2.3	20.18	DEC.YINJECT	DO3.3	20.26	EPM_02.YSTART
DO2.4	20.19	CH.YSTART	DO3.4	20.27	EPM_02.YSTOP
DO2.5	20.20	CH.YRESET	DO3.5	20.28	VPM_01.YSTART
DO2.6	20.21	CH.YDECANT	DO3.6	20.29	VPM_01.YSTOP
DO2.7	20.22		DO3.7	20.30	VPM_02.YSTART
DO2.8	20.23		DO3.8	20.31	VPM_02.YSTOP
DO4	Module		DO5	Module	
DO4.1	30.00	V_01.YOPEN	DO5.1	30.08	EPM_01.YRESET
DO4.2	30.01	V_01.YCLOSE	DO5.2	30.09	EPM_02.YRESET
DO4.3	30.02	V_01.YRESET	DO5.3	30.10	VPM_01.YRESET
DO4.4	30.03	V_02.YOPEN	DO5.4	30.11	VPM_02.YRESET
DO4.5	30.04	V_02.YCLOSE	DO5.5	30.12	
DO4.6	30.05	V_02.YRESET	DO5.6	30.13	
DO4.7	30.06		DO5.7	30.14	
DO4.8	30.07		DO5.8	30.15	
AO0					
AO0.1		FCV_01.YPOSITION			
AO0.2		FCV_02.YPOSITION			
AO0.3		FCV_03.YPOSITION			
AO0.4		FCV_04.YPOSITION			
AO0.5					
AO0.6					
AO0.7					
AO0.8					

3.5. Simulación del proceso

La simulación será embebida ya que no se dispone de una planta real, es decir, integrada en el diseño y programación del PLC para poder realizar todas las funciones dedicadas y específicamente programadas. Cabe destacar que para poder hacer la simulación se ha de definir tanto los componentes de los elementos como el comportamiento lógico de estos.

Debido a que los tiempos de espera y los tiempos para diferentes procesos que constituyen la producción de aceite de oliva son largos, estos han sido disminuidos considerablemente en la simulación de procesos para ser debidamente mostrados en un periodo de tiempo razonable.

3.5.1. Diseño

En la programación Controladores Lógicos Programables, como en cualquier otra programación, existen diferentes tipos de lenguaje de programación. La siguiente imagen presenta los cinco diferentes tipos:

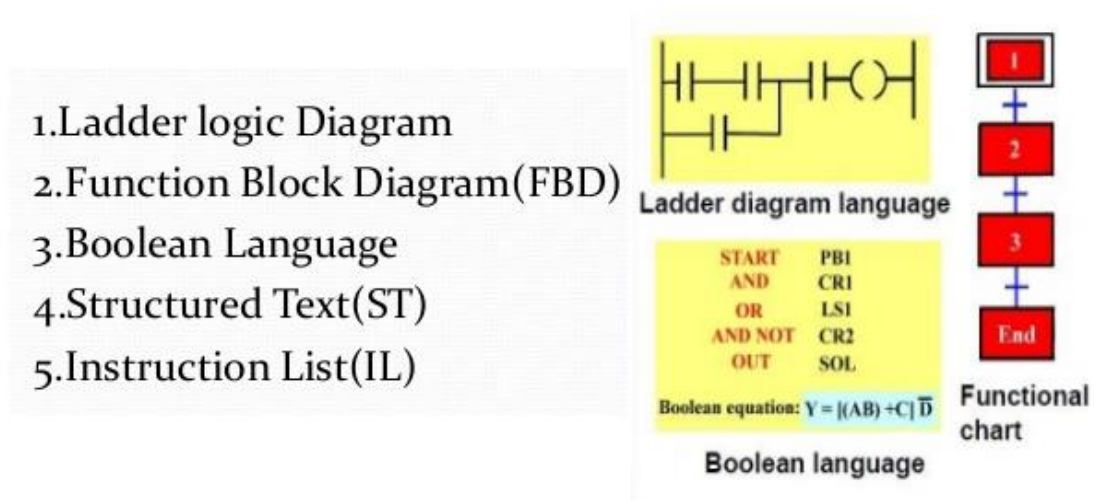


Figura 17. Lenguajes de programación de PLCs [Cortesía: Syed Atif Chishti]

Para facilitar la comprensión, todos los diseños del controlador y simulación serán tipo *Ladder Diagram* (tipo escalera), exceptuando algún Add-On que se programará tipo *Function Block Diagram* (tipo diagrama de bloques).

3.5.2. Bloques del sistema

Dentro de los bloques del sistema se encuentran los denominados Add-On, que son elementos o funciones encapsuladas con instrucciones lógicas particulares. En ellos se definen los parámetros y el comportamiento lógico de las señales de los componentes o funciones.

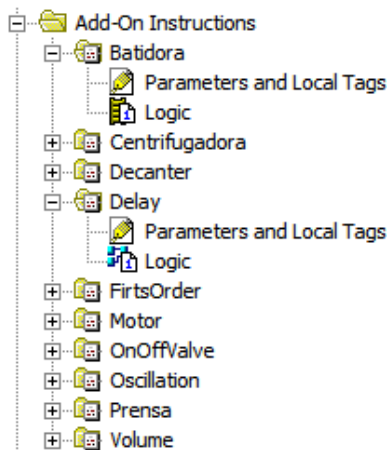


Figura 18. Add-Ons de la planta programados, ejemplos en Ladder y FBD.

Como se ha visto en la imagen anterior existen elementos como Batidora, Centrifugadora, Decanter, Motor, OnOffValve o Prensa, pero también existen diferentes funciones como Delay, FirstOrder, Oscillation o Volume.

La siguiente imagen muestra los parámetros que definen al elemento batidora y el comportamiento lógico.

Name	Usage	Default	Force Mask	Style	Data Type	Description
AI	Output	0.0		Float	REAL	
AI06	InOut	??		Float	REAL	
AlarmCondition...	Input	0		Decimal	BOOL	
DI11	InOut	??		Decimal	BOOL	XRUNNING
DI12	InOut	??		Decimal	BOOL	XALARM
DI13	InOut	??		Decimal	BOOL	XHELPER
DI14	InOut	??		Decimal	BOOL	XTEMP
DO15	Input	0		Decimal	BOOL	YSTART
DO16	Input	0		Decimal	BOOL	YRESET
DO17	Input	0		Decimal	BOOL	YHELPER
DO18	Input	0		Decimal	BOOL	YTEMP
EnableIn	Input	1		Decimal	BOOL	Enable Input - Sys...
EnableOut	Output	0		Decimal	BOOL	Enable Output - S...
Oscillation	Local	{...}	{...}		Oscillation	
POWER	Local	0.0		Float	REAL	
T_Helper	Local	{...}	{...}		TIMER	
T_Reset	Local	{...}	{...}		TIMER	
T_Run	Local	{...}	{...}		TIMER	
T_Temp	Local	{...}	{...}		TIMER	

Figura 19. Parámetros del elemento de simulación (Add-On) batidora.

A rasgos generales y poniendo en ejemplo el comportamiento lógico de la batidora (figuras 20 y 21) se ha de destacar:

- En la línea cero la inicialización de los tiempos de añadido de coadyuvante, espera a temperatura, de marcha y reset.
- De las líneas cinco a diez la secuencia simulada de los procesos dentro de la batidora con las diferentes condiciones necesarias que se requieren.
- En la línea doce las condiciones requeridas para hacer activar la alarma.

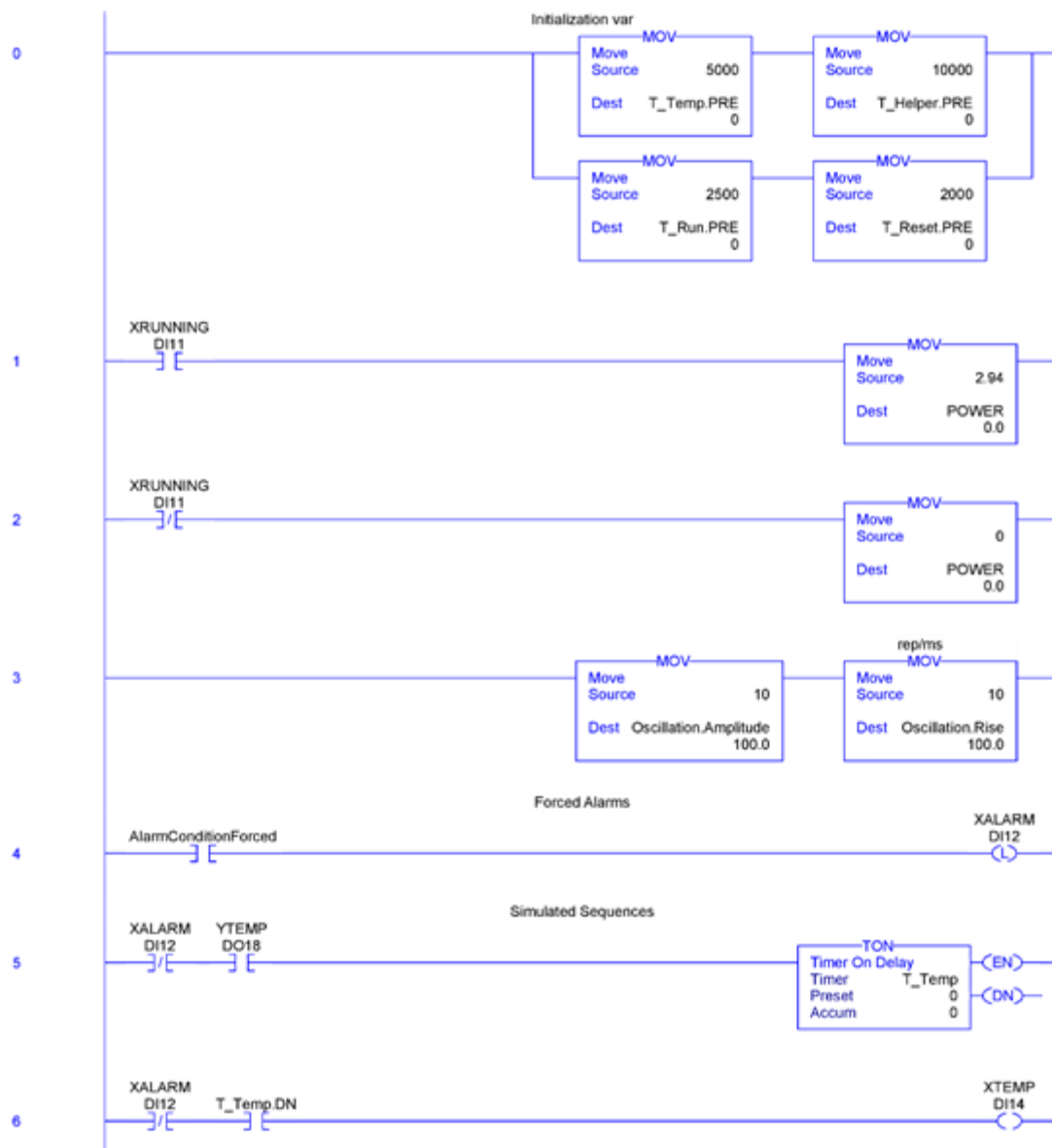


Figura 20. Lógica de control del Add-On batidora. [Parte 1/2]

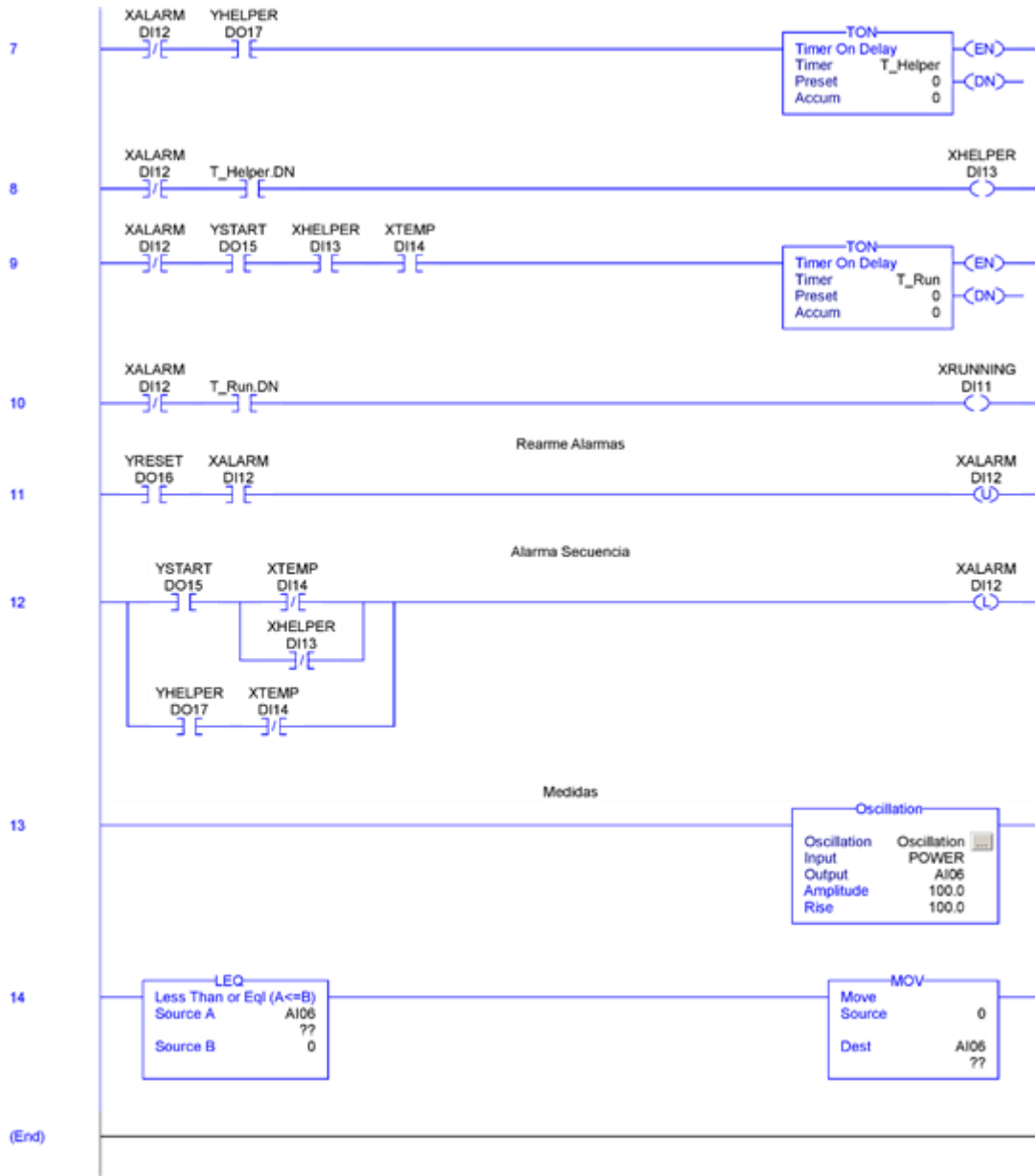


Figura 21. Lógica de control del Add-On batidora. [Parte 2/2]

Antes se ha comentado que también hay otras partes programadas en estructura de bloques, se han programado de esta manera por facilidad y para incluirlos dentro de la planta para darle realidad a la simulación. El bloque del delay (retraso) es un sencillo ejemplo (figura 22) de input y output definidos con un deadtime (tiempo muerto) intencionado para retrasar la salida de la entrada.

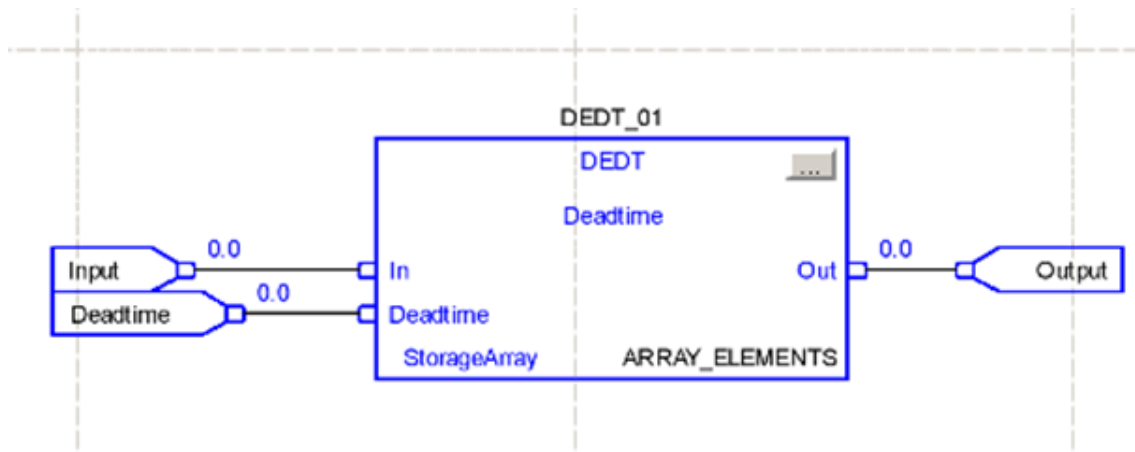


Figura 22. Lógica de control del Add-On delay.

La creación de los bloques del sistema depende de cada Add-On programado y toda la documentación de estos se encuentra disponible en el Anexo A.A3.

3.5.3. Estructura de simulación

La simulación de la planta es el bloque que conecta el comportamiento lógico con las entradas y salidas del sistema.

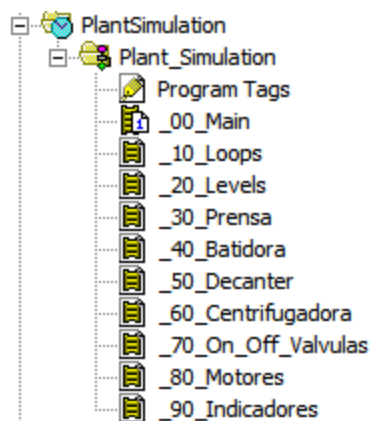


Figura 23. Estructura de la simulación de la planta.

Por un lado, se conectan las entradas y las salidas definidas del comportamiento lógico con las entradas y salidas del sistema, siguiendo con el mismo ejemplo de la batidora tomado en el apartado de los bloques del sistema Add-Ons tenemos (figura 24):

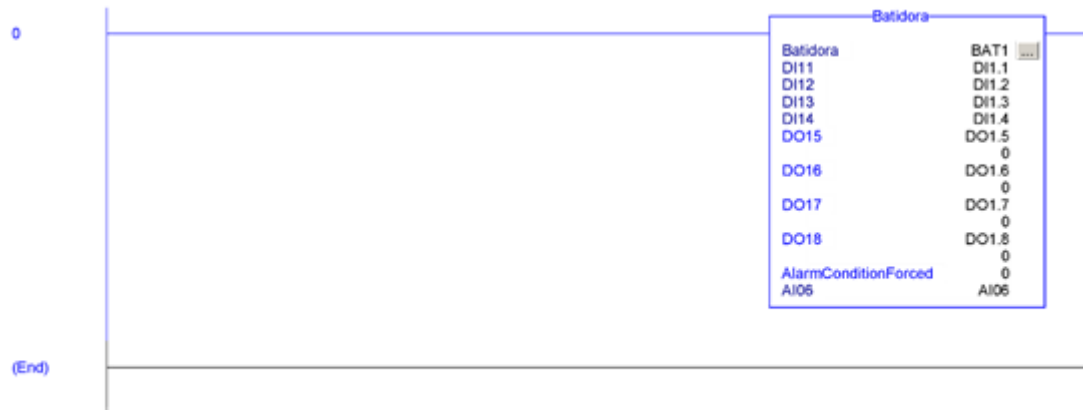


Figura 24. Conexión de las entradas y salidas de la lógica de la batidora con las entradas y salidas del elemento batidora en la simulación de la planta.

Por otro lado, dentro de la simulación de la planta, también se han de describir los procesos internos de los niveles de los depósitos y de las válvulas reguladoras de caudal. Un ejemplo (figura 25) es el del llenado y vaciado de la tolva.

- En la línea cero con las condiciones adecuadas la tolva se llena o vacía progresivamente gracias al retardo, el sistema de primer orden, la oscilación y mover (conjunto de instrucciones implementadas de delay, firstorder, oscillation y move).
- Las líneas uno a cinco describen las condiciones de entrada y salida de la tolva y las adecua al llenado o vaciado progresivo.

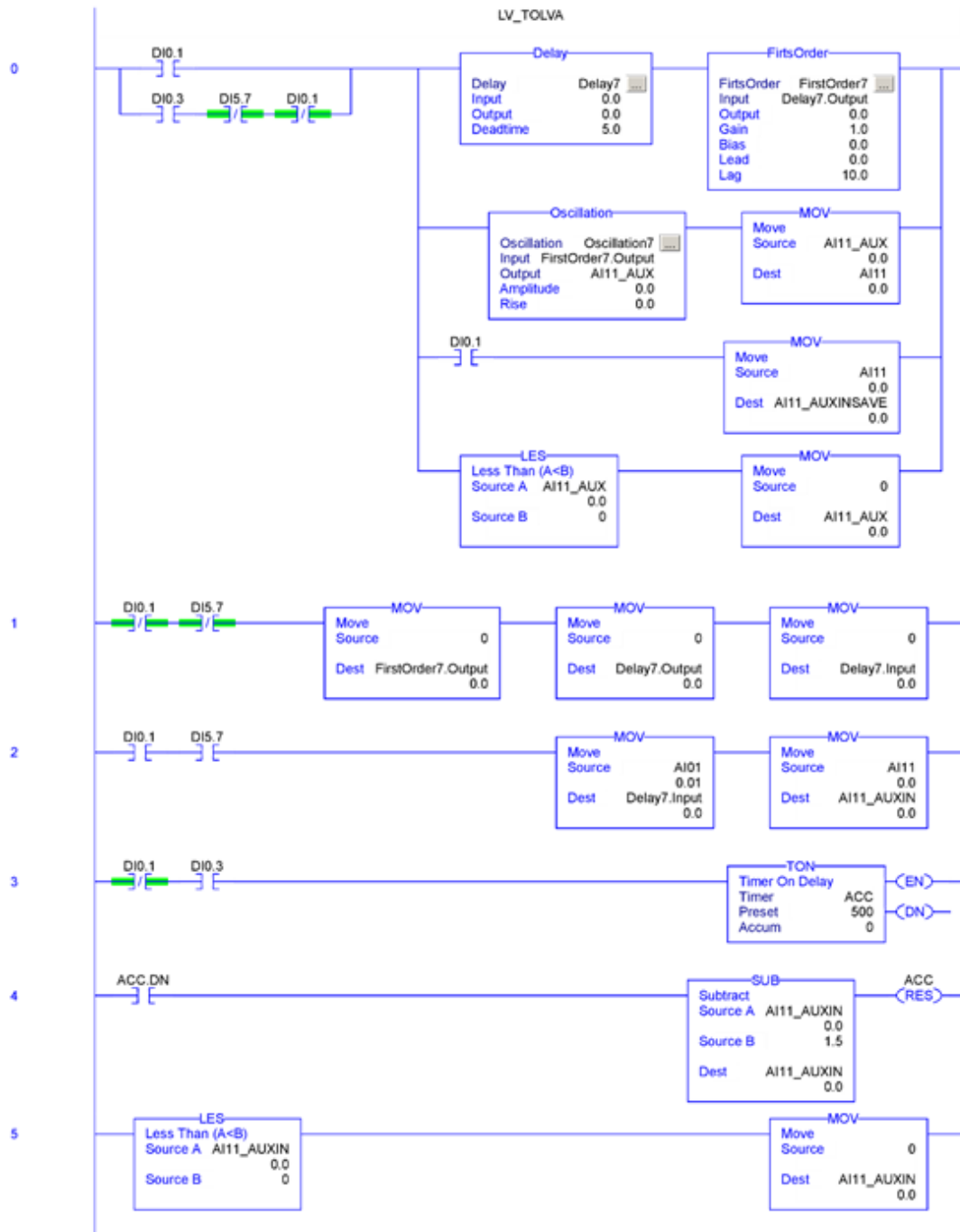


Figura 25. Comportamiento del nivel de la tolva dentro de la simulación de la planta.

Al igual que los bloques del comportamiento lógico, toda la documentación de la estructura de simulación está disponible en el Anexo A.A2.

3.6. Programa del controlador

3.6.1. Estructura del programa

El programa del controlador consta de una rutina principal que llama a todas las subrutinas y con posibilidad de enviar y retornar parámetros si la subrutina lo requiere. Esta tarea se ejecuta de manera continua según ciclo de scan (muestreo). También tiene los inputs y outputs del sistema, así como los elementos que intervienen en el proceso de manera indexada y la rutina en automático.

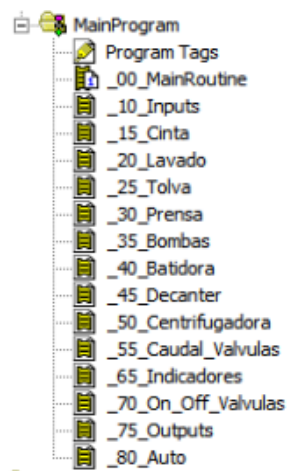


Figura 26. Estructura del controlador.

3.6.2. Definición de tipos de datos

Una de las características del controlador y del software es la posibilidad de crear UDDTs (User Defined Data Types), los tipos de dato definidos por el usuario permiten estructurar mejor el programa, hacer modificaciones rápidas, simplificar código y ahorrar horas en el proceso de programación.

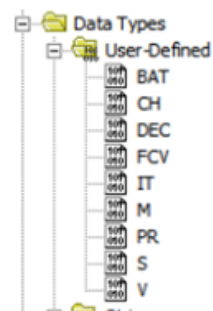


Figura 27. User Defined Data Types creados en el proyecto.

Cada elemento (válvula, válvula de control, motor, máquina...) tiene su propia UDDT con todos sus tags posibles definidos acorde al fichero de intercambio. Esto permitirá indexar tags a infinitos elementos deseables.

Name:

Description:

Members: Data Type Size: ?? byte(s)

Name	Data Type	Style	Description	External Access
E_ALARM	BOOL	Decimal	Estado alarma	Read/Write
E_AUTOMATIC	BOOL	Decimal	Estado auto	Read/Write
E_OSERVICE	BOOL	Decimal	Estado fuera de servicio	Read/Write
E_MANUAL	BOOL	Decimal	Estado manual	Read/Write
E_RUNNING	BOOL	Decimal	Estado marcha	Read/Write
E_SERVICE	BOOL	Decimal	Estado servicio	Read/Write
E_INJECTOK	BOOL	Decimal	Estado inject ok	Read/Write
POWER	REAL	Float	Potencia	Read/Write
RHOURS	DINT	Decimal	Horas marcha	Read/Write
THOURS	DINT	Decimal	Horas totales	Read/Write
C_STOP	BOOL	Decimal	Orden stop	Read/Write
C_START	BOOL	Decimal	Orden start	Read/Write
C_AUTOMATIC	BOOL	Decimal	Orden auto	Read/Write
C_MANUAL	BOOL	Decimal	Orden manual	Read/Write
C_OSERVICE	BOOL	Decimal	Orden fuera de servicio	Read/Write
C_RESET	BOOL	Decimal	Orden reset	Read/Write
C_SERVICE	BOOL	Decimal	Orden servicio	Read/Write
C_INJECT	BOOL	Decimal	Orden inyect	Read/Write

Figura 28. Tags de las señales definidos para la UDDT batidora.

Las programación relacionada con de los tipos de datos UDDTs está disponible en el Anexo A.A4.

3.6.3. Lógica de control de los elementos

La creación de UDDTs ha permitido crear subrutinas indexadas, de tal manera que la rutina principal, cuando salta a una subrutina, le transmite el elemento que ha de actuar y cuando acaba devuelve todas las acciones de control efectuadas en el elemento en cuestión.

Si se ve en detalle cualquier subrutina, se hace notable que todas presentan algunas similitudes, esto es debido a que los elementos han tener un estado en manual y un estado en automático, así como un estado en servicio, otro en fuera de servicio (figura 29). Además, una señal física de salida de inicio y paro, junto una señal física de entrada que activa el estado de marcha.

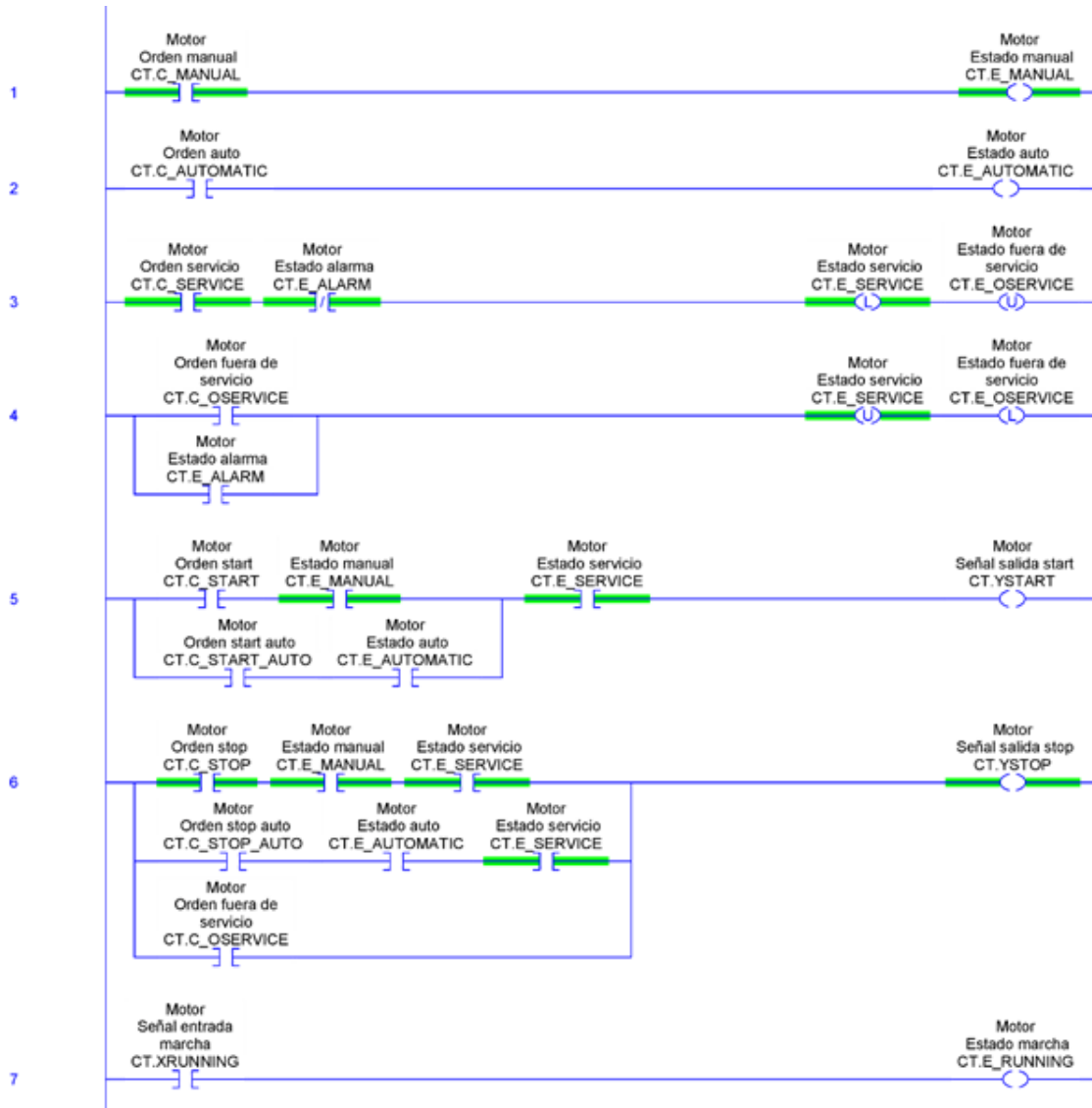


Figura 29. Código de programa de la rutina de cinta. [Parte 1/2]

Por último, también han de presentar un reset y una alarma (figura 30). Para evitar falsas pulsaciones en el reset o falsas alarmas se han implementado de temporizadores con preset de tres segundos, que activarán el estado en cuestión en caso el temporizador llegue al valor prefijado.

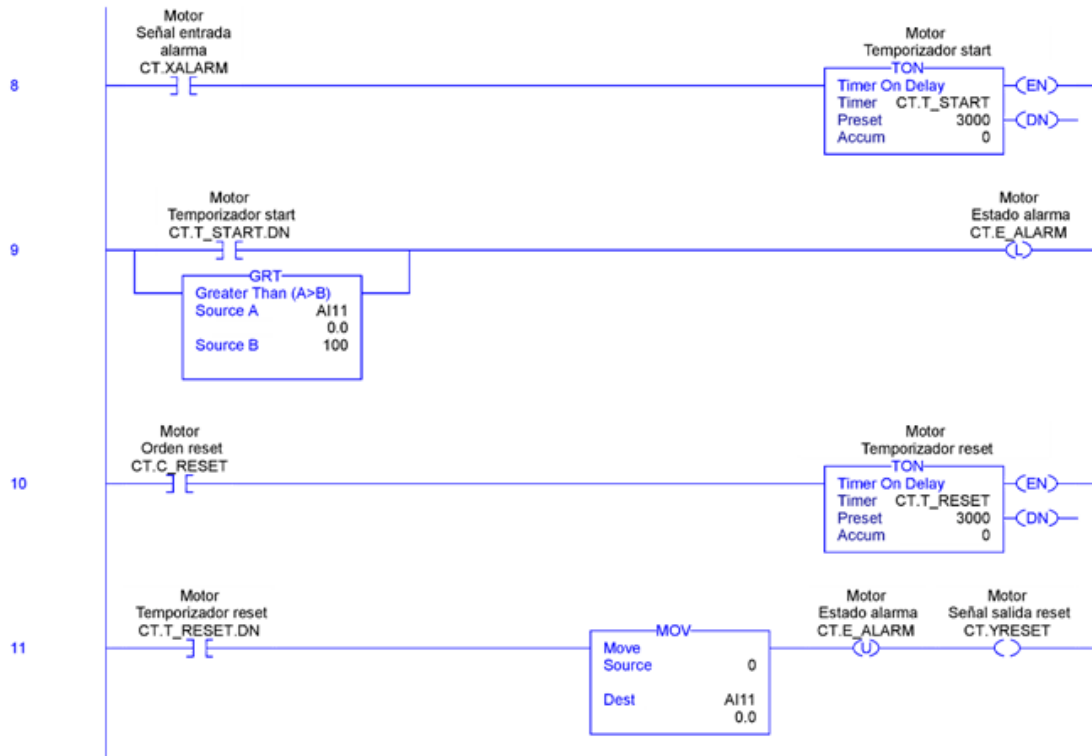


Figura 30. Código de programa de la rutina de cinta. [Parte 2/2]

3.6.4. Lógica de control de los sistemas

La lógica de control del sistema está disponible en la Main Routine, como se puede observar, va recorriendo los elementos con la función JSR (Jump to Subroutine), opcionalmente con entrada y salida de elementos indexados.

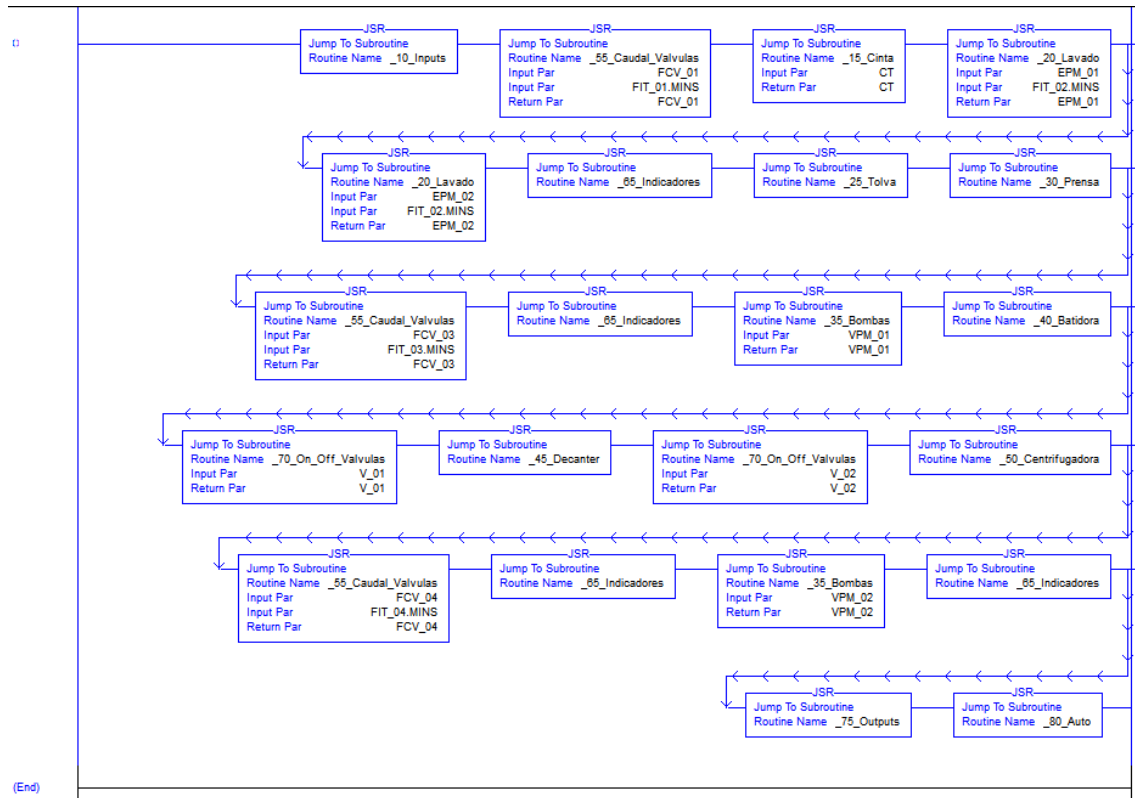


Figura 31. Rutina principal del controlador.

Ahora bien, en el momento de indexar en la lógica de control de los elementos se deberá especificar el salto a la subrutina con la función SBR (Subroutine) y la regresión a la Main Routine con la función RET (Return from Subroutine) como se muestra en la siguiente imagen (figura 32).

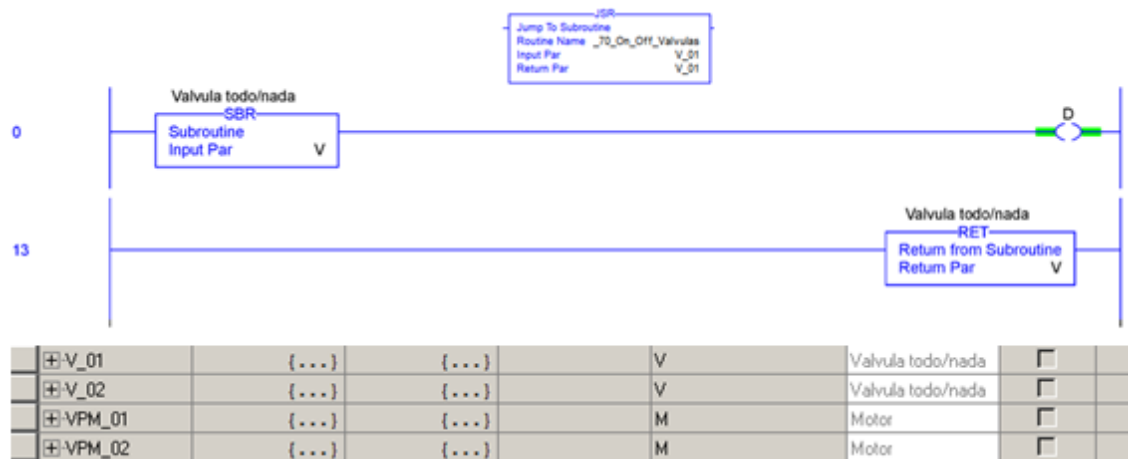


Figura 32. Ejemplo de indexación de las válvulas V_01 [mostrada en imagen] y V_02. La línea sin numeración pertenece a la rutina principal, las líneas numeradas a la rutina específica de válvulas y definiciones pertenecen a la parte de tags del controlador.

La documentación relacionada con la lógica de control de los elementos y sistemas se encuentra disponible en el Anexo A.A1.

3.6.5. Secuencias de control

En líneas generales para que una orden tenga efecto sobre una salida se comprobará que los estados del elemento o sistema son adecuados y en caso de ser en modo automático todos los elementos deberán estar en el correspondiente estado. Cada elemento está programado de tal forma que no es posible que esté en dos estados contrarios al mismo tiempo.

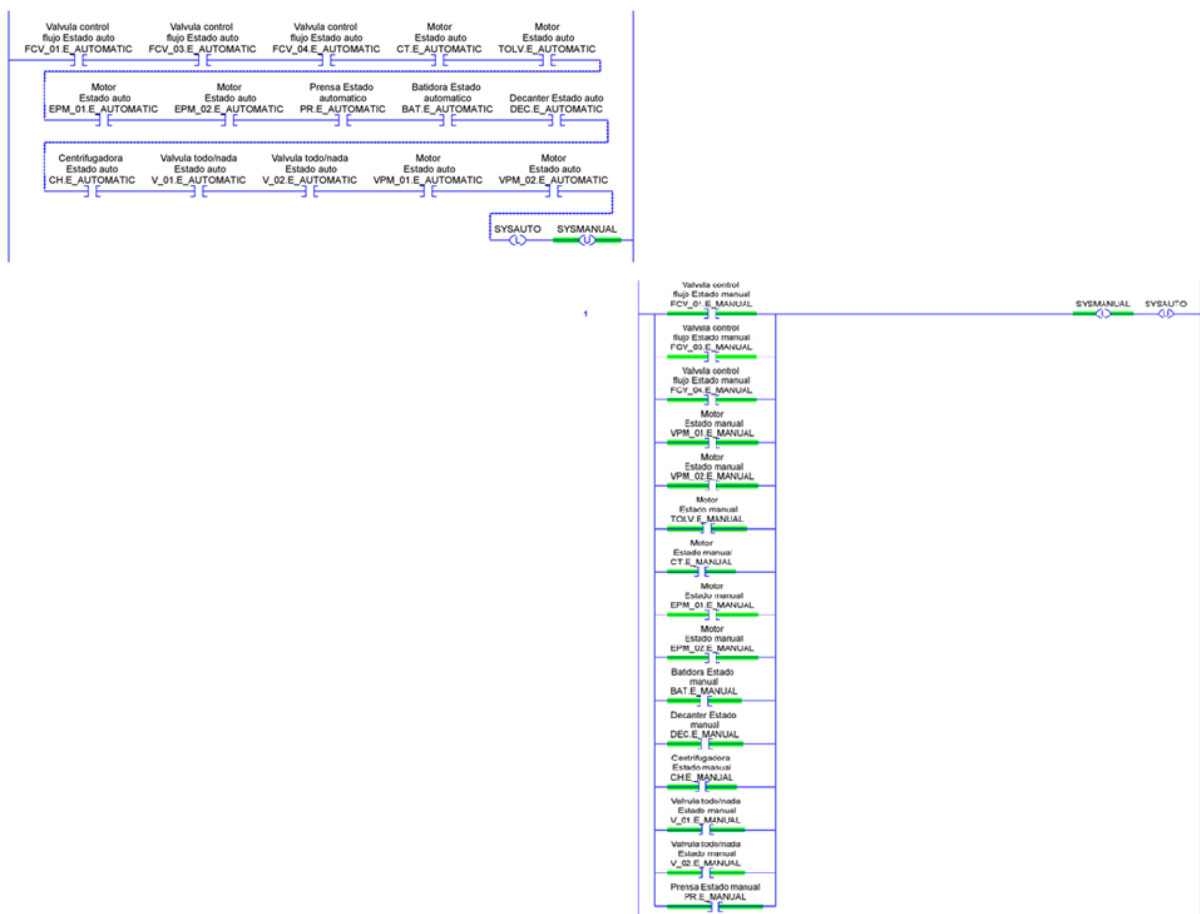


Figura 33. Secuencia de control del estado automático en la rutina de automático

Para poder actuar sobre la salida particular de un elemento éste deberá estar en estado de servicio y sin estado de alarma activo.

Por seguridad los estados y las salidas físicas digitales de cada elemento serán latcheados (mantenidos) y deslatcheados para conservar su valor si la conexión con el SCADA se interrumpe.

3.7. Programa del software SCADA

3.7.1. Sistemas SCADA

Los sistemas SCADA (Supervisory Control and Data Acquisition) son aplicaciones de software diseñadas con la finalidad de controlar y supervisar procesos a distancia basadas en adquirir datos de procesos remotos a partir de una HMI (Human-Machine Interface), un panel de control diseñado para conseguir una comunicación interactiva entre operador y proceso, con la función de transmitir ordenes, visualizar gráficamente los resultados y obtener una situación del proceso en tiempo real.

3.7.2. Árbol de navegación

El árbol de navegación es un esquema gráfico que permite ver la estructura de una aplicación de forma simple y concisa.

El sistema ha sido creado de tal manera que desde la pantalla principal es posible acceder a las diferentes pantallas de la aplicación ya sean tipo *pop-up* o *replace* para hacer una navegación dinámica y fácil.

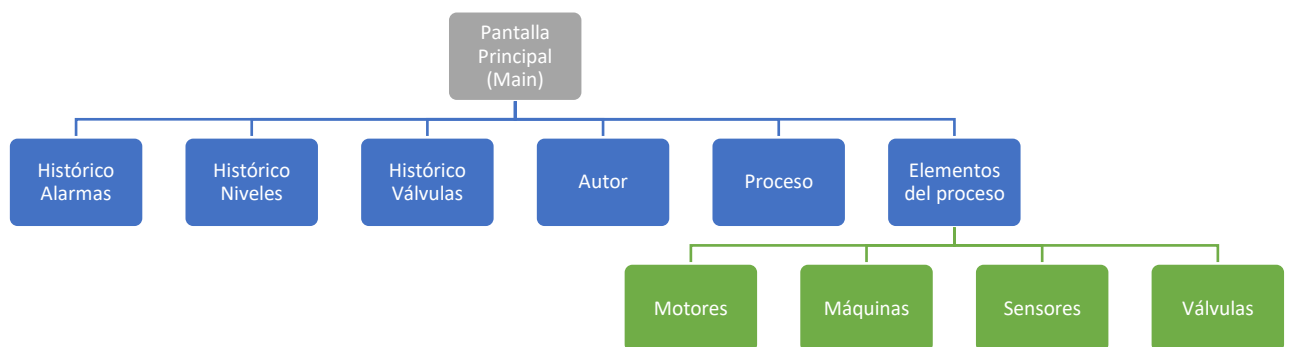


Figura 34. Esquema del árbol de navegación del SCADA.

3.7.3. Definición de los tipos de datos

Siguiendo la misma lógica que en la programación del PLC y junto con el fichero de intercambio, se ha asignado para cada señal de los elementos un tag en concreto. Como se ha mencionado anteriormente, el entorno de InTouch no se admiten tags con puntos en su nomenclatura y se sustituirán manualmente por guiones bajos.

Las tags que el SCADA comparte con el controlador han de ser tipo *I/O Real* para señales analógicas que muestren un rango de valores. Los tags para señales digitales de valor cero o uno han de ser tipo *I/O Discrete*. Es importante para la comunicación PLC-SCADA sé que utilice el *Access Name* DDE de comunicación previamente creado.

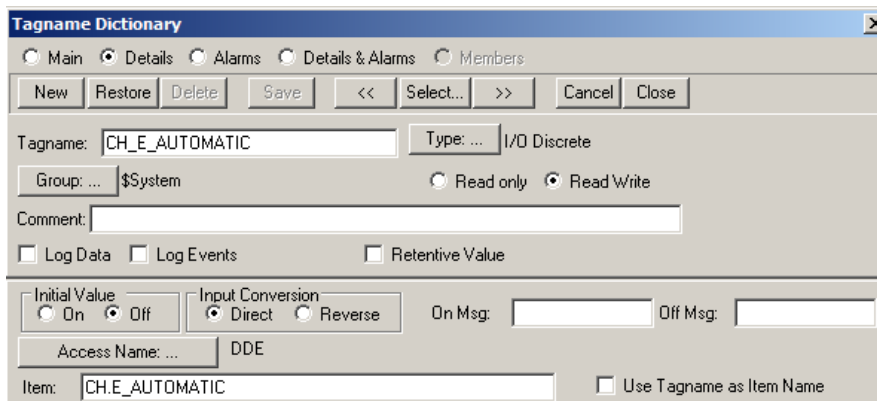


Figura 35. Ejemplo de señal I/O Discrete.

Los tags de los que se requiera generar un histórico deberán tener habilitado el campo *Log Data* en el *Tagname Dictionary* así como haber habilitado el guardado de históricos del software.

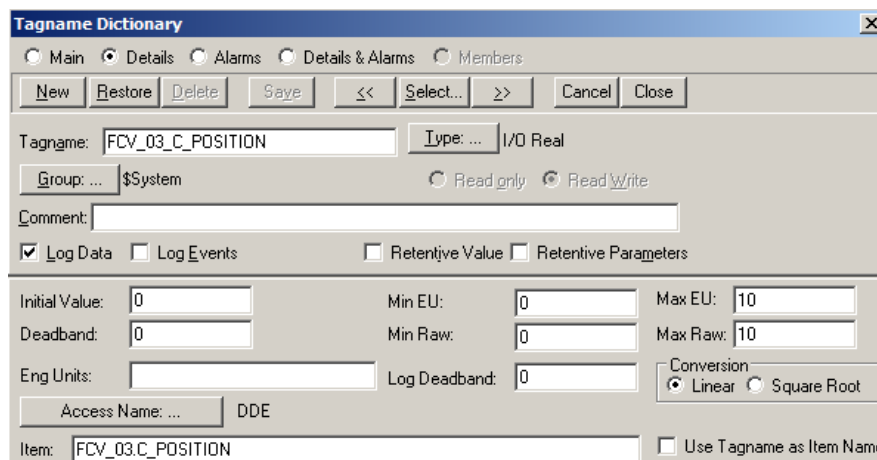


Figura 36. Ejemplo de señal I/O Real.

Los tags *I/O Discrete* de alarmas que requieran un estado de alarma y el reconocimiento por condición, tendrán que ser indicados en el *Tagname Dictionary* en el apartado de *Alarms*.

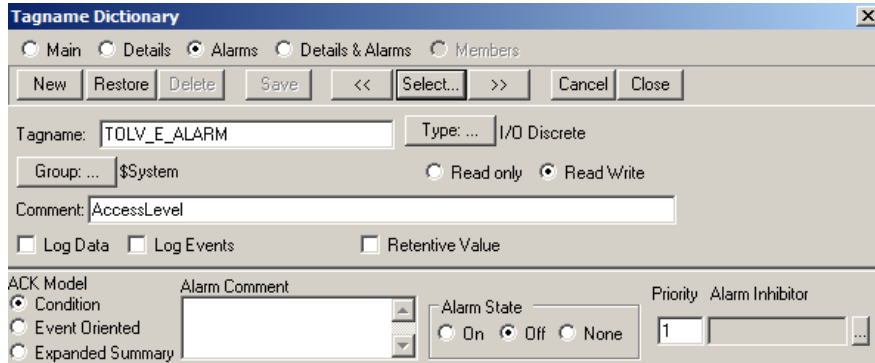


Figura 37. Ejemplo de señal *I/O Discrete* con condición de alarma.

Haciendo una breve observación al *Tagname Directory* se comprueba que el programa contiene un total de 371 tags.

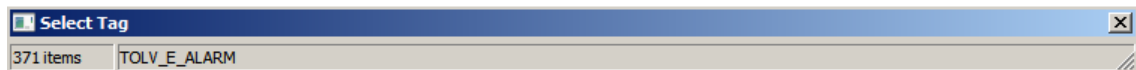


Figura 38. Tags totales.

3.7.4. Diseño de las pantallas de la aplicación

Para el diseño de las pantallas de la aplicación se ha seguido el criterio lógico propio y la experiencia adquirida en otros proyectos. Se ha buscado un equilibrio en el cual las pantallas y controles sean fáciles e intuitivos y a la vez proporcionen el máximo de información posible sin llegar a ahogar en información al usuario.

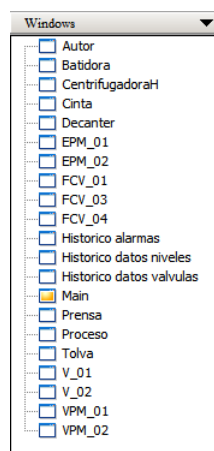


Figura 39. Ventanas programadas en SCADA (Main seleccionada y principal).

Al iniciarse el programa se muestra la pantalla principal dónde se haya una visión general del proceso, permite acceder a todas las posibles interfaces gracias a la barra de navegación superior, permite el uso de usuarios y muestra un sumario con todas las alarmas principales activas.

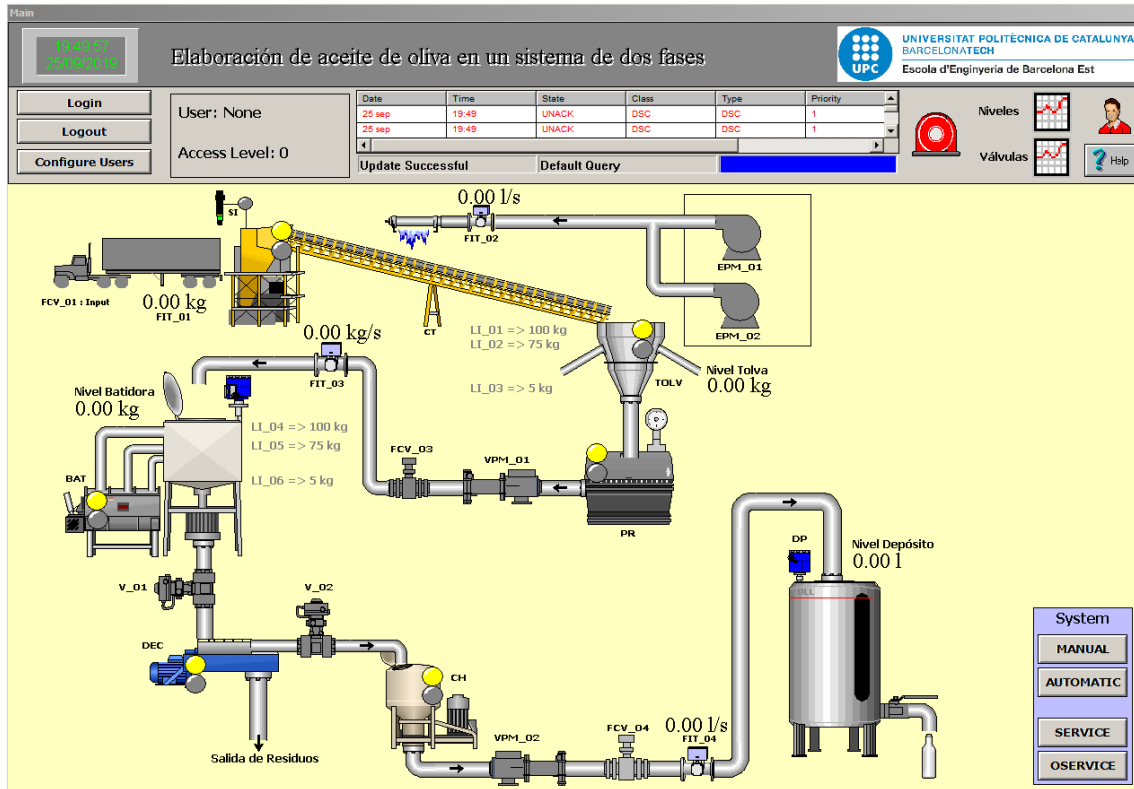


Figura 40. Pantalla principal SCADA.

Cabe destacar que desde cualquier ventana y/o pantalla se puede acceder a cualquier otra en un solo paso, así como, todas las ventanas creadas son movibles por el usuario y en todas las pantallas aparecen fecha, hora, usuario, configuración de usuario y *login/logout*.

Como interfaces secundarias existen tres diferentes. El histórico de alarmas como primera interfaz secundaria, que muestra un informe de los estados de éstas. Se disponen de botones específicos para que el operario pueda reconocer cada alarma rearmándola una vez se ha solucionado el problema que la hizo saltar.

Historico alarmas

20:02:44
25/09/2016

Elaboración de aceite de oliva en un sistema de dos fases

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Escola d'Enginyeria de Barcelona Est

Login
Logout
Configure Users

User: None
Access Level: 0

Histórico alarmas

General View

Date	Time	State	Class	Type	Priority	Name	Group	Provider	Value
25 sep	19:49	UNACK	DSC	DSC	1	BAT_E_ALARM	System	Intouch	OFF
25 sep	19:49	UNACK	DSC	DSC	1	PR_E_ALARM	System	Intouch	OFF
25 sep	19:49	UNACK	DSC	DSC	1	DEC_E_ALARM	System	Intouch	OFF
25 sep	19:49	UNACK	DSC	DSC	1	CH_E_ALARM	System	Intouch	OFF
25 sep	19:49	UNACK	DSC	DSC	1	V_02_E_ALARM	System	Intouch	OFF
25 sep	19:49	UNACK	DSC	DSC	1	V_01_E_ALARM	System	Intouch	OFF
25 sep	19:49	UNACK	DSC	DSC	1	EPM_02_E_AL...	System	Intouch	OFF
25 sep	19:49	UNACK	DSC	DSC	1	EPM_01_E_AL...	System	Intouch	OFF
25 sep	19:49	UNACK	DSC	DSC	1	VPM_02_E_AL...	System	Intouch	OFF
25 sep	19:49	UNACK	DSC	DSC	1	FIT_04_E_ALA...	System	Intouch	OFF
25 sep	19:49	UNACK	DSC	DSC	1	FIT_03_E_ALA...	System	Intouch	OFF
25 sep	19:49	UNACK	DSC	DSC	1	FIT_02_E_ALA...	System	Intouch	OFF
25 sep	19:49	UNACK	DSC	DSC	1	FIT_01_E_ALA...	System	Intouch	OFF
25 sep	19:49	UNACK	DSC	DSC	1	VPM_01_E_AL...	System	Intouch	OFF
25 sep	19:49	UNACK	DSC	DSC	1	FCV_01_E_ALA...	System	Intouch	OFF
25 sep	19:49	UNACK	DSC	DSC	1	FCV_04_E_ALA...	System	Intouch	OFF
25 sep	19:49	UNACK	DSC	DSC	1	FCV_03_E_ALA...	System	Intouch	OFF
25 sep	19:49	UNACK	DSC	DSC	1	TQLV_E_ALARM	System	Intouch	OFF
25 sep	19:49	UNACK	DSC	DSC	1	CT_E_ALARM	System	Intouch	OFF

Update Successful

Default Query

Válvulas
Reconocer FCV_01
Reconocer FCV_03
Reconocer FCV_04
Reconocer V_01
Reconocer V_02

Motores
Reconocer Cínta
Reconocer Tolva
Reconocer EPM_01
Reconocer EPM_02
Reconocer VPM_01
Reconocer VPM_02

Maquinaria
Reconocer Prensa
Reconocer Batidora
Reconocer Decanter
Reconocer Centrifug.

Figura 41. Interfaz secundaria de histórico de alarmas.

Como segunda interfaz secundaria está el histórico de datos de nivel, que contiene información sobre las tendencias históricas de los niveles de la tolva, batidora y depósito. Se pueden visualizar, consultar valores anteriores y se da la posibilidad de exportarlos a un archivo .csv para procesar los datos con otro software si es necesario. Por último, se ha implementado un medidor de nivel actual del depósito con su respectiva válvula de vaciado y las potencias de trabajo de las máquinas a tiempo real.

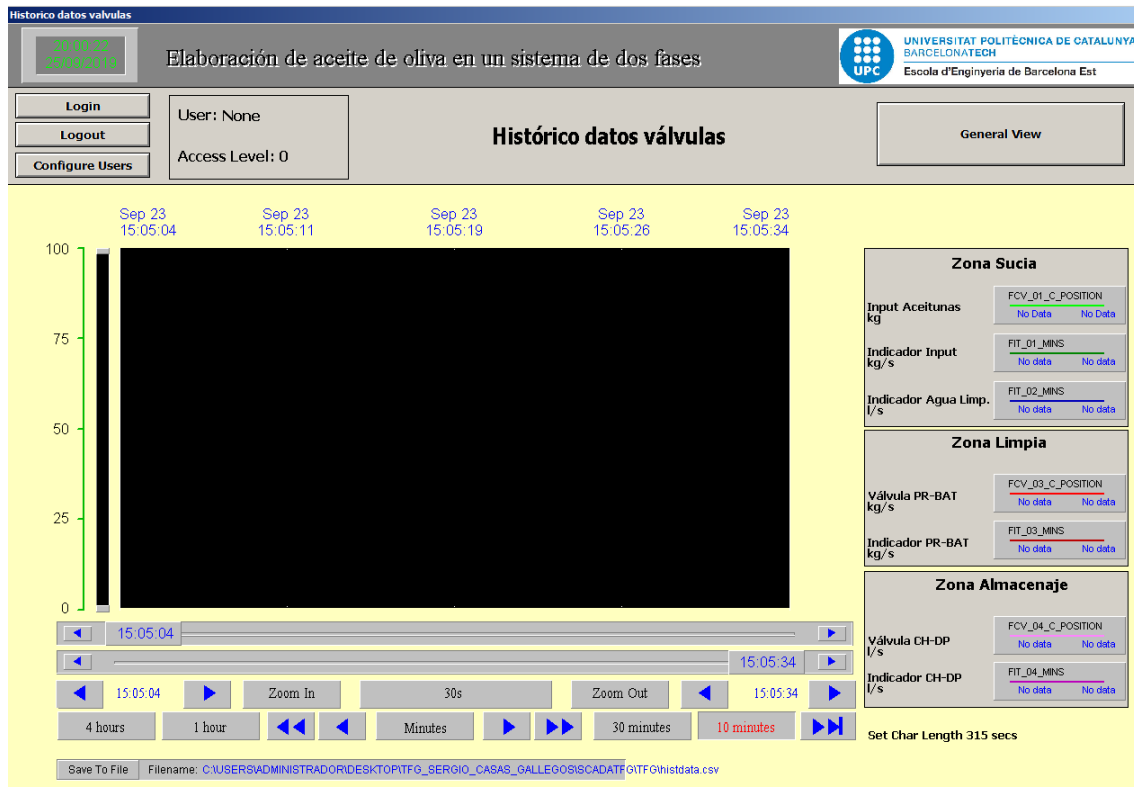


Figura 42. Interfaz secundaria de históricos de datos de las válvulas.

La tercera interfaz secundaria es el histórico de datos de válvulas, que contiene información sobre las tendencias históricas de las diferentes válvulas del proceso con sus respectivos medidores. Se pueden visualizar, consultar valores anteriores y se da la posibilidad de exportarlos a un archivo .csv para procesar los datos con otro software si es necesario



Figura 43. Interfaz secundaria de históricos de datos de las de niveles y control de depósito.

3.7.5. Diseño de los comandos

Los comandos son las órdenes que permiten accionar interactuar con los diferentes elementos de tal manera que con la debida conexión con el controlador permiten poner en marcha, paro, en servicio o fuera de servicio, automático, manual, reset al elemento que se quiere manejar en cuestión.

Los estados son las situaciones temporales de los elementos a partir de la orden o comando recibido. Los estados activos se remarcan en negro y los inactivos en gris, dos estados contrarios nunca podrán estar activos a la vez. Por criterio lógico propio se ha creado una misma interfaz de tipo *pop-up*, si no muy similar, para todos los elementos del proceso.

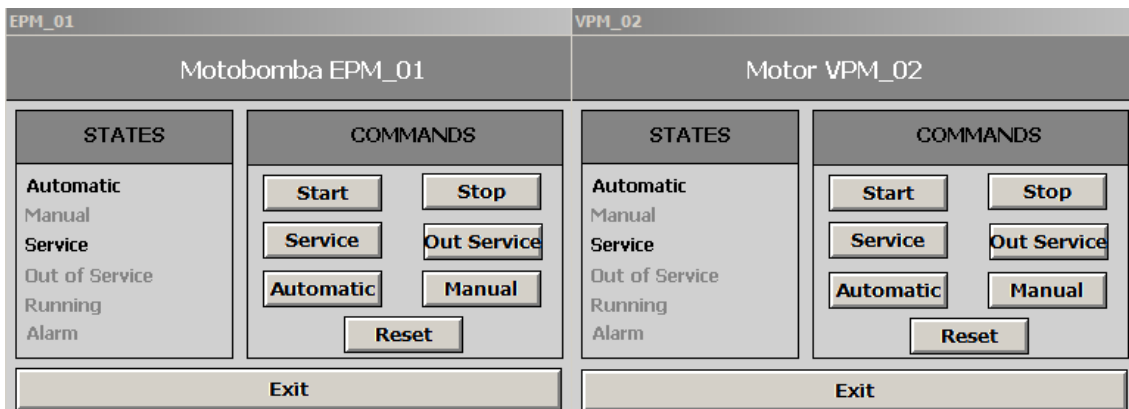


Figura 44. Diseño de la pantalla de comandos para los motores.

Para aquellos elementos que necesitan regulación se les ha añadido una entrada de posición de *setpoint*.

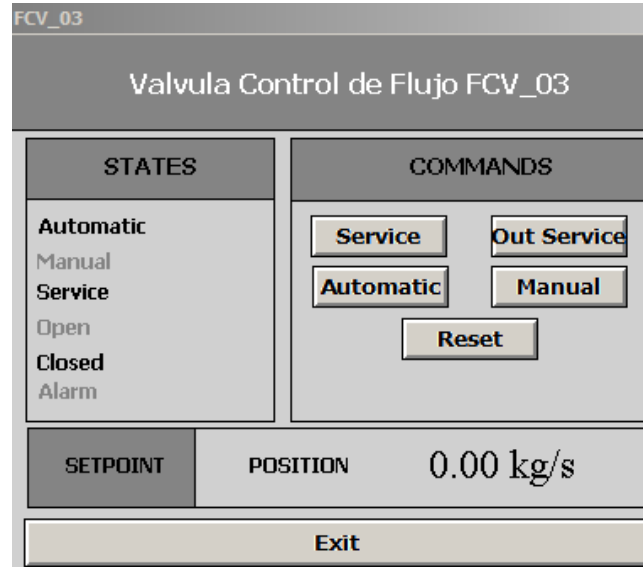


Figura 45. Diseño de la pantalla de comandos para las válvulas de flujo.

Los procesos de las maquinarias más complejas que necesitan uno o más procesos extras también los comandos y los estados así como la información necesaria para su control también han sido añadidos, tal y que:

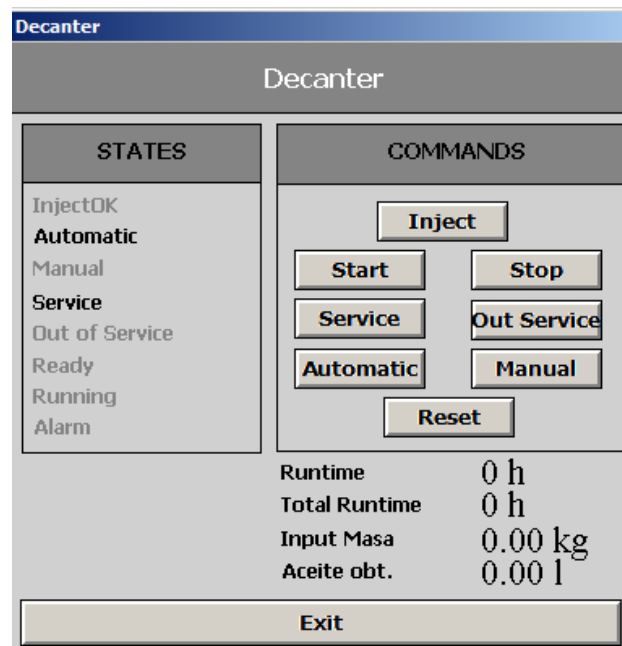


Figura 46. Diseño de la pantalla de comandos para el decánter.

Los comandos del sistema han sido implementados en la pantalla principal con la finalidad de no dificultar el proceso.

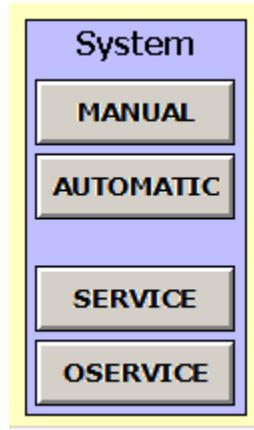


Figura 47. Diseño de comandos del sistema.

3.7.6. Scripts

Las scripts son una característica avanzada de un sistema y se utilizan cuando se necesita personalizar la integración de comportamientos, configuraciones o cualquier escenario poco común que no corresponde a la configuración ordinaria y no se podría lograr.

Para enviar las órdenes desde el SCADA al controlador se utilizan scripts en los botones de comando que envían un '1' lógico al bit de señal de orden y un '0' lógico al bit de la señal de la orden contraria.

Los botones de reset envían un '1' lógico a la señal de orden de reset mientras estén pulsados. Para evitar falsas pulsaciones se deberá mantener apretado el botón durante 3 s para resetear el elemento (función programada desde el controlador).

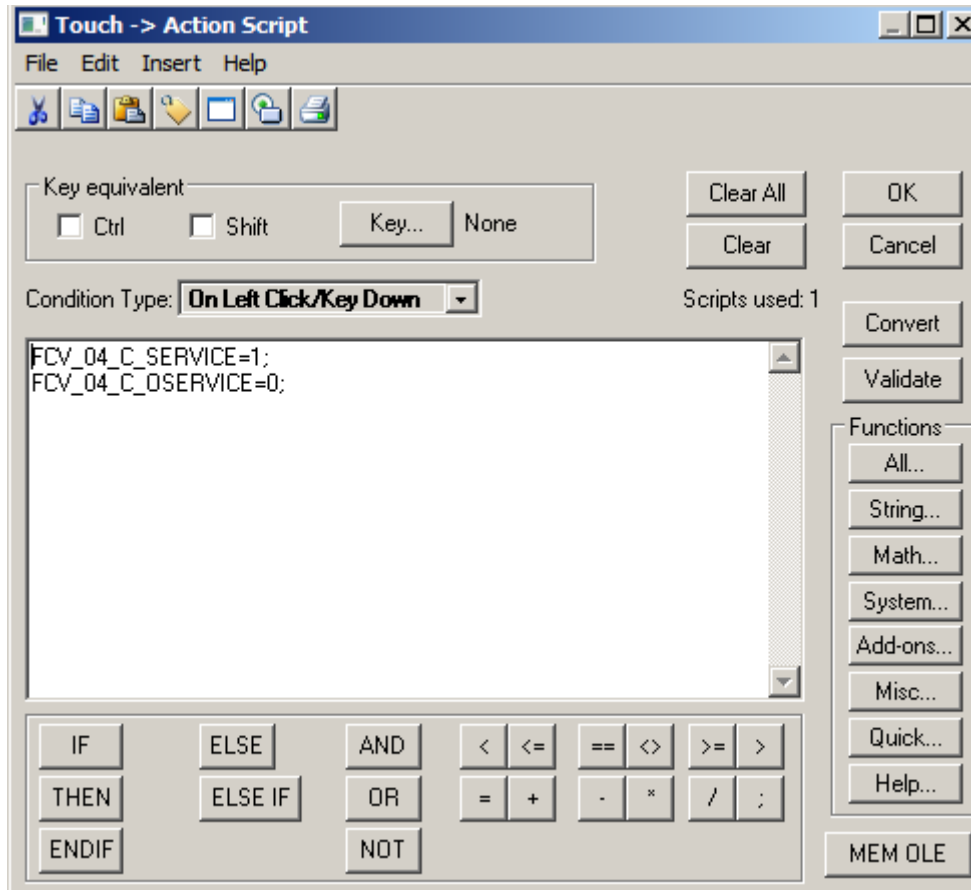


Figura 48. Ejemplo de script de dar orden de servicio al pulsar el botón correspondiente de la válvula de flujo número 4.

La siguiente script está contenida en todos los elementos interactivos para que la pantalla *pop-up* pueda mostrarse (figura 49).

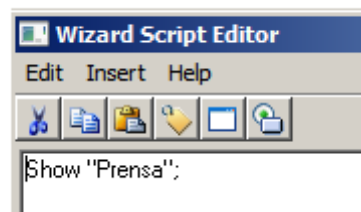


Figura 49. Ejemplo de script para la interacción de elementos.

Por último referente al tema de scripts, en los botones del sistema se han de detallar absolutamente todos los elementos del sistema con la función que se desee configurar.

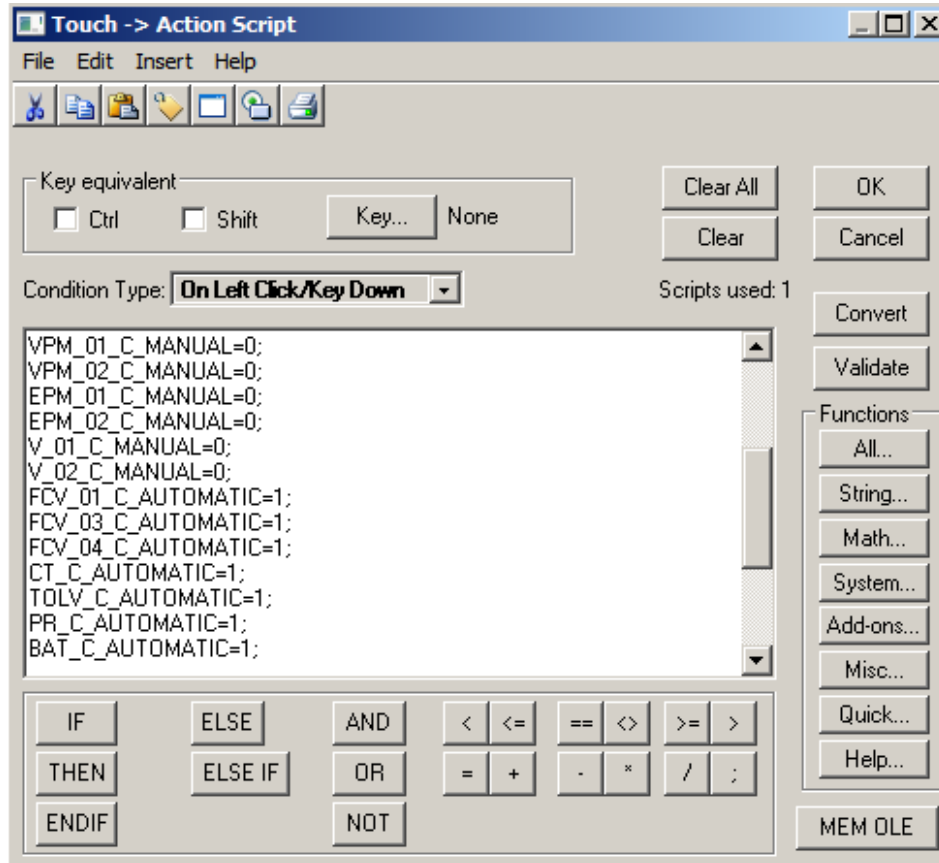


Figura 50. Ejemplo de script dar orden de automático a todo el sistema al pulsar el botón correspondiente en la pantalla principal.

3.7.7. Diseño de la interfaz de alarmas del sistema

Como todo sistema de control y supervisión se ha de disponer de un sistema de alarmas y un histórico que se pueda consultar en todo momento. El diseño es simplista y permite ver claramente que alarmas están activas cada momento.

De cada alarma se puede visualizar en orden, fecha y hora en la que se ha producido, estado, clase, tipo y prioridad, nombre del tag, grupo y proveedor y valor. Así mismo, en el margen inferior de la pantalla el usuario dispone de una botonera para poder reconocer de forma individual cada una de las alarmas.

20:40:36
26/09/2013

Elaboración de aceite de oliva en un sistema de dos fases

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Escola d'Enginyeria de Barcelona Est

Login

Logout

Configure Users

User: None

Access Level: 0

Histórico alarmas

General View

Date	Time	State	Class	Type	Priority	Name	Group	Provider	Value
26 sep	20:40	ACK	DSC	DSC	1	BAT_E_ALARM	\$System	Untouch	OFF
26 sep	20:40	UNACK	DSC	DSC	1	BAT_E_ALARM	\$System	Untouch	OFF
26 sep	20:39	ACK_RTN	DSC	DSC	1	PR_E_ALARM	\$System	Untouch	ON
26 sep	20:39	ACK_RTN	DSC	DSC	1	BAT_E_ALARM	\$System	Untouch	ON
26 sep	20:37	UNACK	DSC	DSC	1	BAT_E_ALARM	\$System	Untouch	OFF
26 sep	20:37	UNACK	DSC	DSC	1	PR_E_ALARM	\$System	Untouch	OFF
26 sep	20:37	UNACK	DSC	DSC	1	DEC_E_ALARM	\$System	Untouch	OFF
26 sep	20:37	UNACK	DSC	DSC	1	OH_E_ALARM	\$System	Untouch	OFF
26 sep	20:37	UNACK	DSC	DSC	1	V_02_E_ALARM	\$System	Untouch	OFF
26 sep	20:37	UNACK	DSC	DSC	1	V_01_E_ALARM	\$System	Untouch	OFF
26 sep	20:37	UNACK	DSC	DSC	1	EPM_02_E_AL...	\$System	Untouch	OFF
26 sep	20:37	UNACK	DSC	DSC	1	EPM_01_E_AL...	\$System	Untouch	OFF
26 sep	20:37	UNACK	DSC	DSC	1	VPM_02_E_AL...	\$System	Untouch	OFF
26 sep	20:37	UNACK	DSC	DSC	1	FIT_04_E_AL...	\$System	Untouch	OFF
26 sep	20:37	UNACK	DSC	DSC	1	FIT_03_E_AL...	\$System	Untouch	OFF
26 sep	20:37	UNACK	DSC	DSC	1	FIT_02_E_AL...	\$System	Untouch	OFF
26 sep	20:37	UNACK	DSC	DSC	1	FIT_01_E_AL...	\$System	Untouch	OFF
26 sep	20:37	UNACK	DSC	DSC	1	VPM_01_E_AL...	\$System	Untouch	OFF
26 sep	20:37	UNACK	DSC	DSC	1	FCV_01_E_AL...	\$System	Untouch	OFF
26 sep	20:37	UNACK	DSC	DSC	1	FCV_04_E_AL...	\$System	Untouch	OFF
26 sep	20:37	UNACK	DSC	DSC	1	FCV_03_E_AL...	\$System	Untouch	OFF
26 sep	20:37	UNACK	DSC	DSC	1	TOLV_E_ALARM	\$System	Untouch	OFF
26 sep	20:37	UNACK	DSC	DSC	1	CT_E_ALARM	\$System	Untouch	OFF

Update Successful

Default Query

Válvulas

Reconocer FCV_01

Reconocer FCV_03

Reconocer FCV_04

Reconocer V_01

Reconocer V_02

Motores

Reconocer EPM_01

Reconocer EPM_02

Reconocer VPM_01

Reconocer VPM_02

Maquinaria

Reconocer Cinta

Reconocer Tolva

Reconocer Prensa

Reconocer Batidora

Reconocer Decanter

Reconocer Centrifug.

Figura 51. Ejemplo de funcionamiento de la pantalla de histórico de alarmas.

En el ejemplo (figura 51) se puede comprobar que en un inicio, todas las alarmas están desactivadas de color rojo, con valor OFF y estado UNACK (unacknowledged), seguidamente hay dos alarmas por parte de la prensa y la batidora, es fácil identificarlas porque están de color azul, con valor ON y estado ACK_RTN (acknowledged alarm return to normal).

De las dos alarmas que han saltado, solo se ha resuelto la de la batidora ya que ha vuelto a su estado inicial y posteriormente ha sido reconocida por el usuario mediante su correspondiente botón, se puede ver claramente ya que está en color negro, con valor OFF y estado ACK (acknowledged).

En este proyecto tan solo se considera una única señal de alarma, no se ha contemplado la posibilidad de tener varios tipos de alarma para un mismo elemento con la finalidad de identificar con certeza dónde se ha producido el fallo.

3.7.8. Gestión de usuarios

La gestión de usuarios permite el control de acceso al sistema a diferentes niveles, es natural proteger el control del SCADA a personas ajenas sin autorización y es necesario facilitar el usuario/contraseña para poder manipular el sistema. Esta función es importante en todos los sistemas SCADA ya que permite diferenciar el nivel de cada usuario y las funcionalidades a las que se les permiten acceder o modificar.

Para ello se ha implementado un sistema de seguridad propio de InTouch con dos usuarios. Un primer usuario tipo administrador que permite acceder a cualquier punto del sistema SCADA y podrá modificar y configurar los usuarios. Un segundo usuario tipo operador que está limitado exclusivamente al funcionamiento y control de proceso.

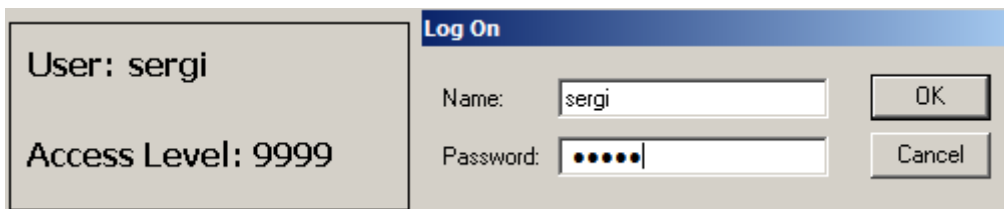


Figura 52. Ejemplo de la gestión de usuarios del sistema, en sesión el usuario Sergi con nivel de acceso máximo.

En las siguientes figuras y por orden de aparición se muestra el tipo de seguridad escogido (figura 53), los botones de login/logout (figura 54) y configuración de usuarios con sus respectivas scripts (figura 55).

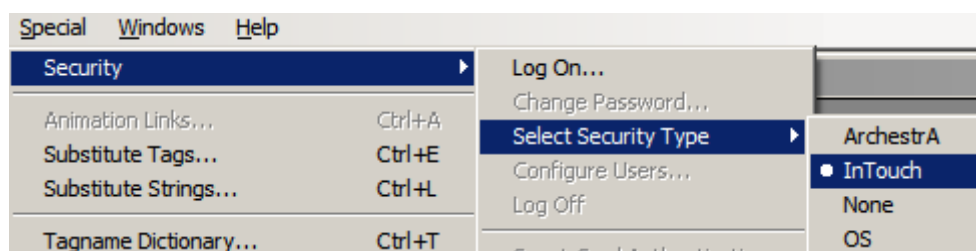


Figura 53. Elección del sistema de seguridad de la aplicación SCADA.

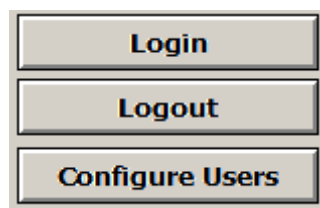


Figura 54. Botones de login, logout y configuración de usuarios implementados.

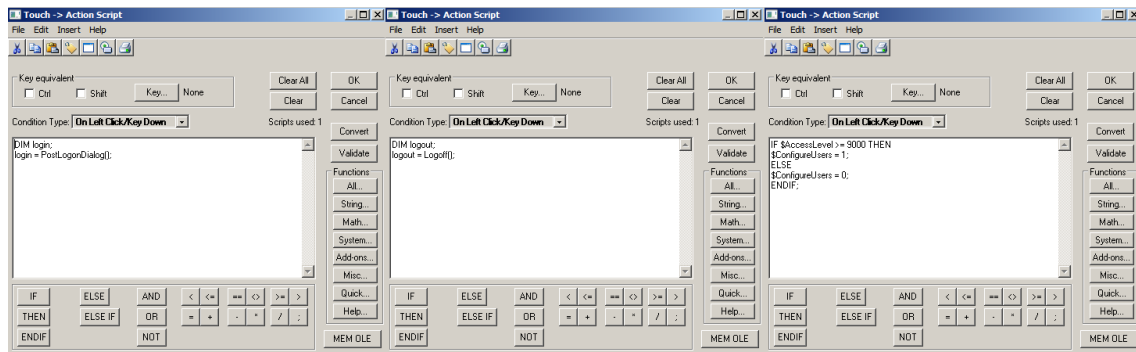


Figura 55. Scripts utilizadas para programar los botones, en este orden, login, logout y configure users.

3.7.9. Gráficos de históricos y tendencias

Para poder utilizar gráficos de históricos y tendencias primero de todo se ha de habilitar el *Historical Logging* dentro de InTouch así como tener activado en los tags necesarios el campo *Log Data*.

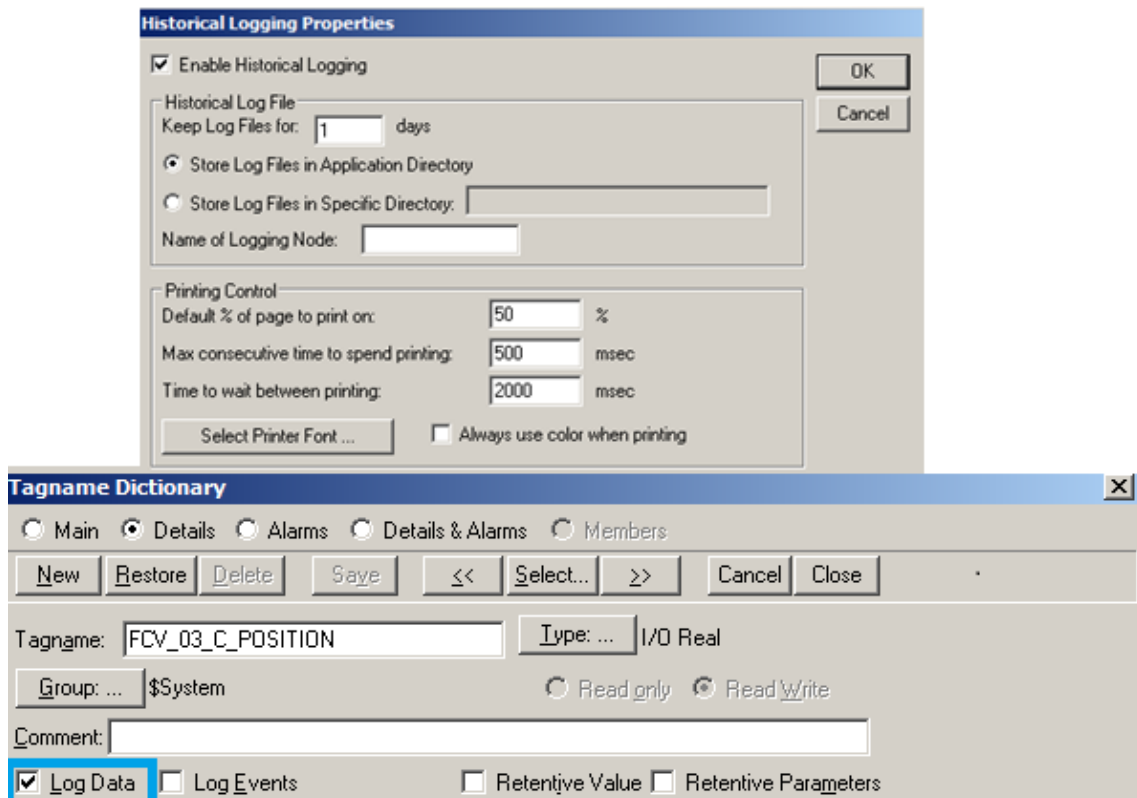


Figura 56. Habilitación del guardado histórico [arriba] y ejemplo de señal I/O Real con guardado de datos de la variable en histórico [abajo].

Se dispone de dos gráficos de históricos y tendencias a tiempo real con botonera inferior para buscar valores históricos de las variables. Uno para la visualización de datos de las distintas válvulas y otro para la visualización de datos de los diferentes niveles de los depósitos y potencias de trabajo de las maquinas.

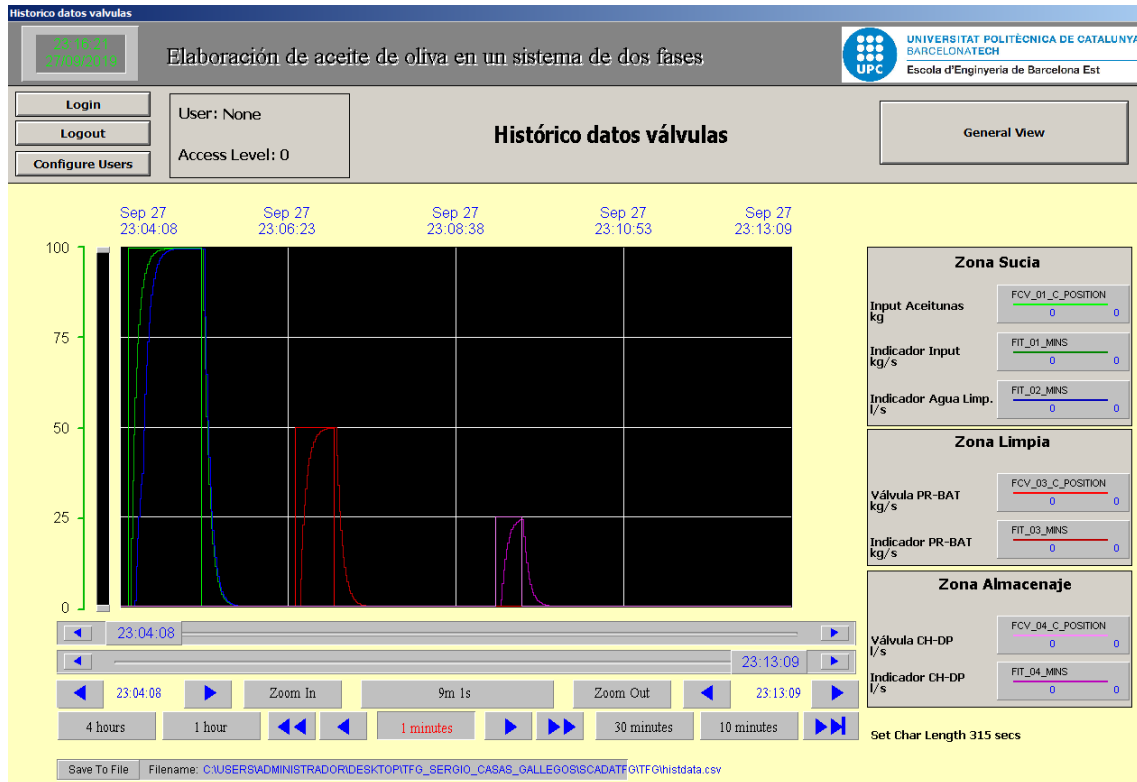


Figura 57. Ejemplo de la captura de valores de las válvulas en el histórico de datos.

También se ha de tener presente que los tiempos de simulación se han reducido en comparación a los tiempos reales con el propósito de hacer demostraciones y testeos.

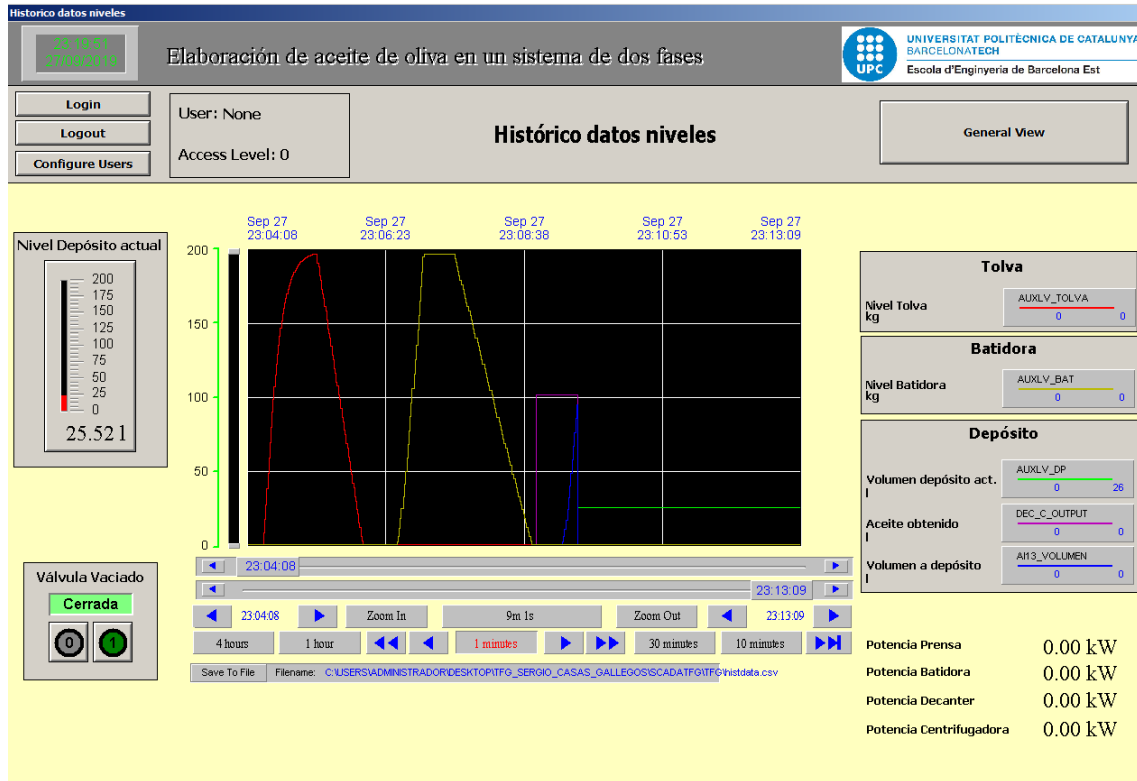


Figura 58. Ejemplo de la captura de valores de los niveles en el histórico de datos.

CAPÍTULO 4: PRUEBAS Y RESULTADOS



4. Pruebas y resultados

El objetivo de este capítulo es documentar de los resultados de las pruebas simuladas y debidamente realizadas a la solución de automatización del proyecto.

4.1. Prueba de conexión controlador-SCADA

Las pruebas controlador-SCADA comprueba la correcta comunicación creada entre PLC y SCADA utilizando el protocolo DDE mediante el programa RSLinx Classic. Para tal efecto, a partir del topic creado en el apartado 3.1.4 se comprueba toda lista de ítems servidos.

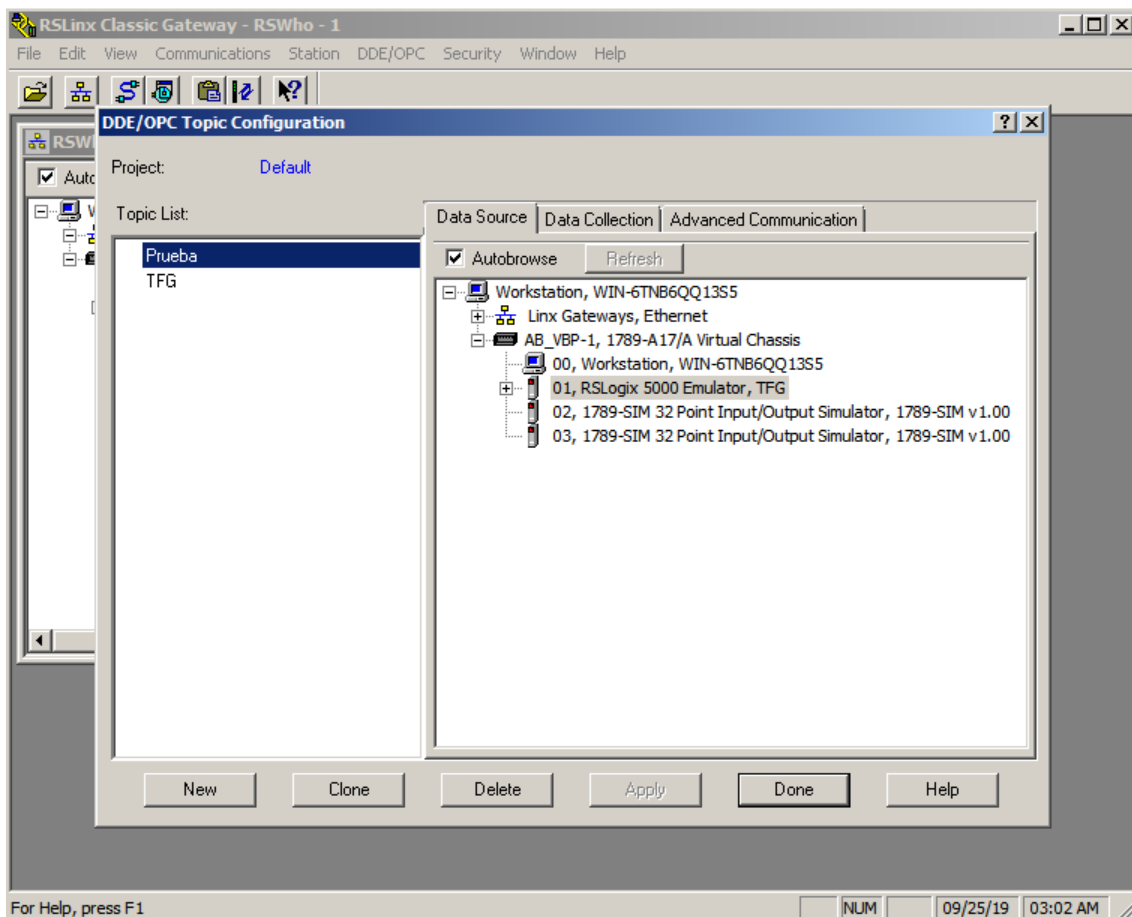


Figura 59. Vinculación del topic configurado con el emulador RSLogix 5000.

Se concluye que el resultado de esta prueba es satisfactorio como muestra la siguiente imagen (figura 60).

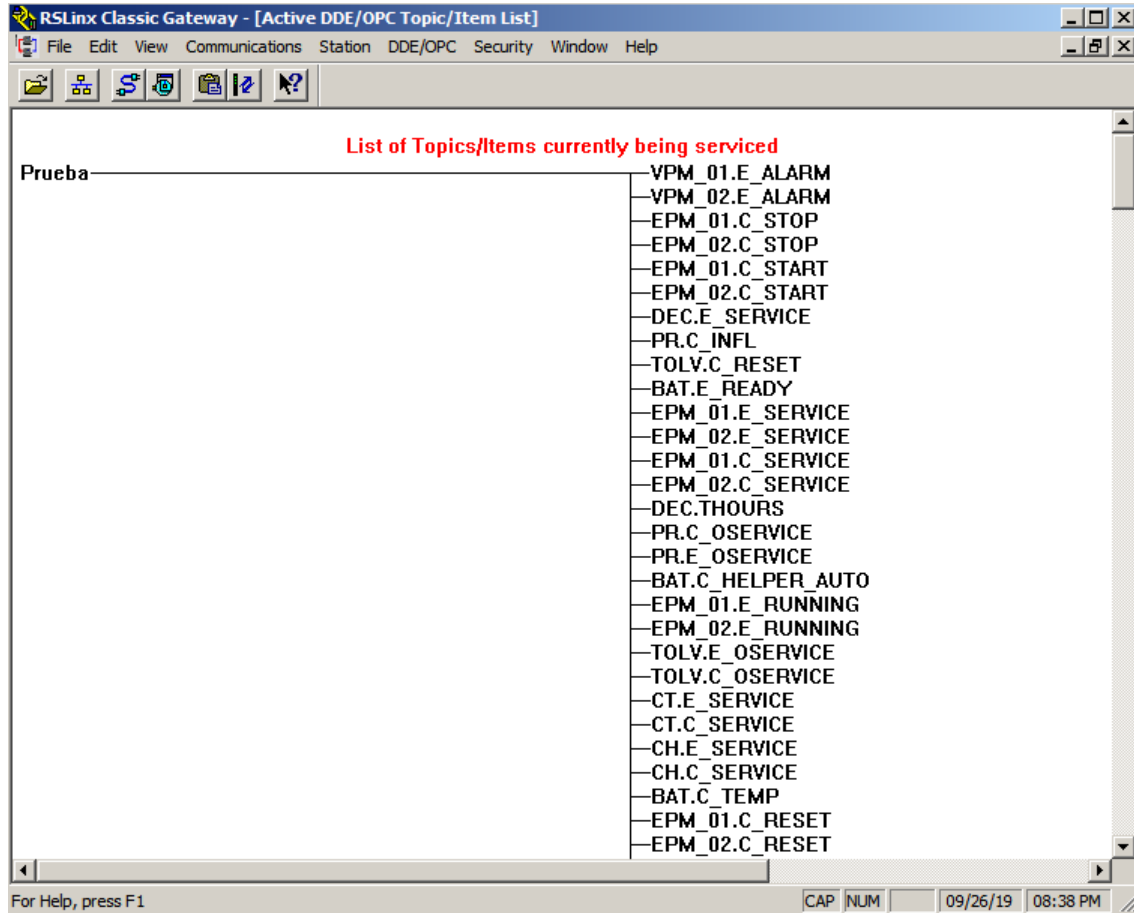


Figura 60. Prueba de comunicación con del listado de ítems que enlazan con el topic configurado.

4.2. Prueba de las entradas y salidas físicas

Con resultado satisfactorio de las pruebas de comunicación se procede a continuar con las pruebas de las entradas y salidas físicas. En ellas se revisa que el controlador es capaz de leer todas las señales de entrada requeridas al igual que es capaz de escribir en las señales adecuadas en cada momento. Las señales I/O se encuentran identificadas en el fichero de intercambio.

El resultado de esta prueba también es satisfactorio, es aconsejable realizar esta prueba con las tablas I/O de los módulos que se encuentran en el fichero de intercambio al final del apartado 3.4, con el motivo de saber en qué variables de entrada se está actuando y qué variables de salida están resultando.

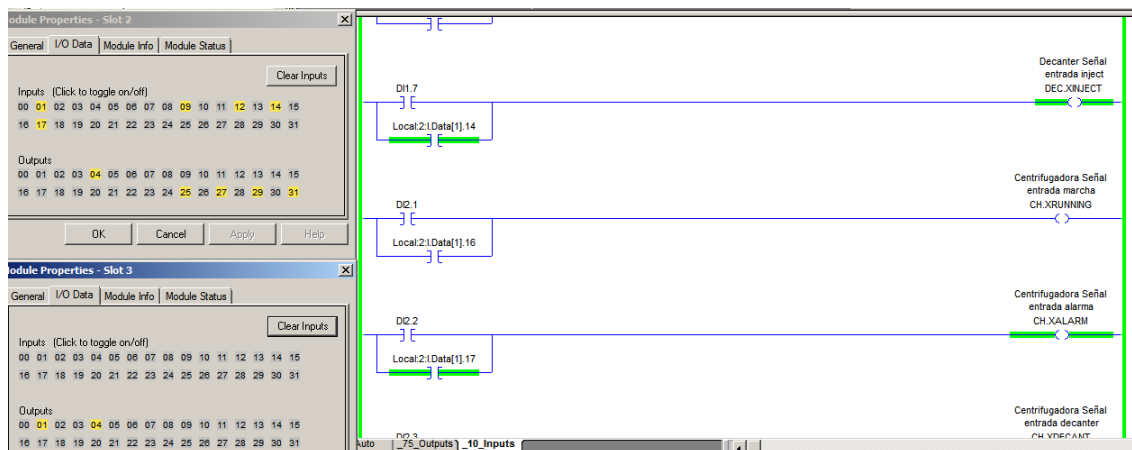


Figura 61. Prueba de I/O con los módulos, en el ejemplo se comprueba los inputs de las alarmas de las máquinas con sus respectivos salidas y el funcionamiento del decánter en modo manual.

4.3. Pruebas de funcionalidad

En las pruebas de funcionalidad se verifica el correcto funcionamiento del sistema en modo manual y automático, paralelamente también se comprueba si se han cumplido los requerimientos funcionales descritos en el apartado 2.4.1 y de diseño descritos en el apartado 2.4.2

Tabla 18. Tabla con los resultados de las pruebas de funcionamiento en modo manual y automático.

Descripción	Resultado	Fecha	Observaciones
Correcto funcionamiento en modo manual.	PASS	20/09/2019	
Correcto funcionamiento en modo automático.	PASS	20/09/2019	

Tabla 19. Tabla con los resultados de las pruebas de funcionalidad.

Req.	Descripción	Resultado	Fecha	Observaciones
RQ_F01	Inicialmente el proceso estará en modo manual y en modo paro.	PASS	27/09/2019	V_01 y V_02 están cerradas al principio del proceso.
RQ_F02	Inicialmente todas las alarmas estarán desactivadas.	PASS	27/09/2019	
RQ_F03	Cada elemento ha de tener estados de paro, marcha, servicio, fuera de servicio, manual, automático o alarma.	PASS	27/09/2019	
RQ_F04	Un elemento no podrá estar en dos estados contrarios.	PASS	27/09/2019	
RQ_F05	La marcha de un elemento en automático solo será posible siempre que esté sin alarmas, en servicio y en modo automático.	PASS	27/09/2019	
RQ_F06	La marcha de un elemento en manual solo será posible siempre que esté sin alarmas, en servicio y en modo manual.	PASS	27/09/2019	
RQ_F07	Las alarmas se quedaran enclavadas hasta que no se ordene su reseteo.	PASS	27/09/2019	
RQ_F08	Si un elemento se encuentra en estado de alarma, este dejará de funcionar y se pondrá inmediatamente en el estado fuera de servicio.	OK	27/09/2019	Los elementos dejan de funcionar pero el estado de fuera de servicio ha de activarlo el operario (programado desde controlador)
RQ_F09	Si se da la orden de apertura, o cierre, a una electroválvula y no se abre, o cierra, en 10 s se activará una alarma de posición.	FAIL	27/09/2019	Pasado el tiempo, con las condiciones necesarias no se activa la alarma de posición.

RQ_F10	Si se da la orden de apertura, o cierre, a una válvula y no se abre, o cierra, se activará una alarma de posición.	PASS	27/09/2019
RQ_F11	En modo automático las electroválvulas reguladoras se abrirán a su máximo caudal permitido.	PASS	27/09/2019
RQ_F12	En los procesos que se traslade un flujo másico o líquido se dispondrán de caudalímetros.	PASS	27/09/2019
RQ_F13	Se contabilizarán las horas de marcha y totales de toda la maquinaria utilizada en el proceso.	PASS	27/09/2019
RQ_F14	Los depósitos que contengan masa o líquido tendrán sensores de nivel que activarán una alarma en caso de que rebose.	PASS	27/09/2019
RQ_F15	El depósito de almacenamiento dispondrá de una válvula manual de vaciado.	PASS	27/09/2019
RQ_F16	El nivel del depósito de almacenamiento se actualizará al final de cada proceso.	PASS	27/09/2019
RQ_F17	El rango de consigna de los valores de entrada aceituna reside entre: [0 kg, 100kg].	PASS	27/09/2019
RQ_F18	El rango de consigna de los valores de la válvula que regula el flujo másico del motor de impulsión vertical PR-BAT reside entre: [0 kg/s, 5 kg/s].	PASS	27/09/2019
RQ_F19	El rango de consigna de los valores de la válvula que regula el flujo líquido del motor de impulsión vertical CH-DP reside entre: [0 l/s, 2,5 l/s].	PASS	27/09/2019

RQ_D01	Todos los elementos integrados están representados en el SCADA y HMI.	PASS	27/09/2019
RQ_D02	La aplicación dispone de una pantalla principal donde es visible todo el proceso.	PASS	27/09/2019
RQ_D03	Desde la pantalla principal es posible la selección del modo manual y automático, así como el servicio o fuera de servicio del sistema.	PASS	27/09/2019
RQ_D04	Las pantallas disponen de marco superior con fecha y hora visible en todo momento.	PASS	27/09/2019
RQ_D05	La aplicación dispone de una pantalla de alarmas con su respectivo histórico de alarmas y los botones pertenecientes para el reconocimiento de estas.	PASS	27/09/2019
RQ_D06	La pantalla principal dispone de un sumario de alarmas críticas y estado.	PASS	27/09/2019
RQ_D07	Los estados de alarmas son visibles visualmente en la maquinaria empleada.	PASS	27/09/2019
RQ_D08	La aplicación dispone de históricos de datos y a tiempo real.	PASS	27/09/2019
RQ_D09	Es posible identificar el proceso así como el autor del trabajo de forma clara.	PASS	27/09/2019
RQ_D10	La aplicación dispone de pantallas tipo pop-up, que evitan quitar visibilidad al proceso de la pantalla principal, estas son debidamente identificadas.	PASS	27/09/2019

RQ_D11	Cada elemento tiene una ventana tipo pop-up independiente donde se muestra sus estados, órdenes a dar y se abre al hacer clic sobre el elemento gráfico en cuestión.	PASS	27/09/2019	
RQ_D12	La aplicación dispone de un control de seguridad con usuario y contraseña. Inicialmente puede ser desactivada para mejorar la rapidez de las demostraciones y simulaciones.	OK	27/09/2019	Disposición de un control de seguridad de usuario y contraseña pero sin uso.
RQ_D13	Las pantallas principales y secundarias disponen de marco superior y de los botones necesarios para enlazar las diferentes pantallas sinópticas.	PASS	27/09/2019	
RQ_D14	Los colores siguen un criterio de diseño lógico. El color verde representa el estado de marcha, el amarillo de indeterminado, el rojo de alarma, el azul de standby y el gris de paro.	OK	27/09/2019	En las máquinas se ha implementado un sistema doble de luces para representar los cuatro estados diferentes ya que con una luz sola solo se podría representar dos estados.
RQ_D15	Se tiene en cuenta el contraste de colores para una fácil comprensión.	OK	27/09/2019	Se ha de vigilar el contraste entre decánter y luz azul de standby.
RQ_D16	Las representaciones simbólicas contienen animaciones graficas que indican los estados y siguen un criterio de diseño estándar lógico.	PASS	27/09/2019	

RQ_D17	Los tags siguen una identificación intuitiva y de sentido común, siguiendo correctamente el fichero de intercambio y acordes con el controlador.	PASS	27/09/2019
---------------	--	-------------	------------

(espacio deliberadamente en blanco)

CAPÍTULO 5: NORMATIVA



5. Normativa

Como cualquier dispositivo que vaya a ser comercializado y/o programado requieren de una serie de estándares de obligado cumplimiento. La normativa que se utiliza se centrará pues en las directrices de programación y desarrollo de software para evitar conflictos y malas interpretaciones, de diseño de la planta, del PLC o SCADA al ser un proyecto de índole de programación.

5.1. Implementación al programa del PLC

El estándar internacional IEC 61131 es un conjunto de normas relativas a los controladores lógicos programables tanto a nivel de software como de hardware, con el fin de normalizar los autómatas programables y sus periféricos correspondientes, tales como interfaces hombre maquina (HMIs), equipos de ensayo (TEs) o equipos de programación y depuración (PADTs). Este estándar se define ocho documentos diferentes:

- **1. Información general:** contiene definiciones y características funcionales típicas, las cuales distinguen a los sistemas basados en PLC de otros sistemas. Describe el procesamiento cíclico de la ejecución de los algoritmos de control, resaltando la imagen almacenada de los estados de las señales de entrada y salida físicas de un proceso industrial.
- **2. Especificaciones y ensayos de los equipos:** presenta los requerimientos eléctricos, mecánicos y funcionales de los dispositivos y las pruebas de calidad que deben cumplir, además de las condiciones ambientales que deben soportar.
- **3. Lenguajes de programación:** se describen la sintaxis y la semántica de cinco lenguajes de programación para PLC; Diagrama Escalera, Lista de Instrucciones, Texto Estructurado, Diagrama de Bloques de Funciones y Diagrama de Funciones Secuenciales.
- **4. Guías de usuario:** proporciona una ayuda para los usuarios en todas las fases del proyecto de automatización.
- **5. Especificación del servicio de mensajería:** define la comunicación de datos entre controladores programables y otros dispositivos.
- **6. Seguridad funcional:** tiene por objetivo adaptar los requerimientos de los estándares IEC 61508 e IEC 62061 de seguridad funcional de los sistemas de seguridad programable y seguridad en maquinarias a los PLC.

- **7. Programación de control difuso:** tiene por objetivo estandarizar entre desarrolladores y usuarios la integración de aplicaciones de control relativa basadas en el apartado 3 del estándar.
- **8. Directrices para la aplicación e implementación de lenguajes de programación:** ofrece interpretaciones para cuestiones no contempladas en el estándar, como instrucciones de uso para el usuario final y asistencia en la programación de algoritmos de control.

Desde el punto de vista de la programación, utilizar este estándar aportará:

Opciones ya que el estándar contempla la programación de los PLC con diferentes lenguajes se ha podido utilizar como lenguaje principal el tipo *ladder* y la posibilidad de programar otras tareas en FBD.

Celeridad debido a que los elementos comúnmente utilizados ya existen. Son un ejemplo las funciones básicas de programación tipo ADD, MOV, ABS, SQRT, SIN, COS entre otras, o diferentes tipos de datos como BOOL, INTEGER, REAL, BYTE, STRING entre otros.

Flexibilidad puesto que se programará en un lenguaje de programación fiel al hardware y no al software.

Facilidad al poder solucionar cualquier problema que surja o pueda surgir y poder evitar posibles errores de interpretación.

5.2. Implementación al programa SCADA

El objetivo principal al coger un sistema SCADA es que este sea fácilmente manipulable y entendible. Para ello, se ha de seguir ciertos estándares comunes cuyo objetivo es poder asegurar la máxima compatibilidad de símbolos en el proceso visual del usuario para obtener la máxima información posible de manera eficiente.

De los siguientes estándares que se han seguido, se hará especial énfasis en el primero y último:

- **ISA S5.5**, Graphic Symbols for Process Displays.
- **ISA 101**, Human-Machine Interfaces.
- **ISO 3864**, Safety colors and safety signs.
- **ISO 7000**, Graphical symbols for use on equipment.
- **ISO 11064**, Ergonomic design of control centres.
- **IEC 60617**, Graphical symbols for diagrams.
- **IEC 60447**, Man-machine interface (MMI) – Actuating principles.
- **Guía GEDIS**, Guía de Creación de guía ergonómica para el diseño de interfaz de supervisión por Pere Ponsa, Marta Díaz, Andreu Catalá (UPC).

5.2.1. Guía ISA-S5.5

El propósito de ISA-S5.5 es establecer un sistema de codificación de tags, símbolos gráficos y colores en las aplicaciones SCADA que pretendan facilitar la rápida comprensión a los usuarios de la información que se transmite, entre PLC-SCADA, a través del muestreo y se establece uniformidad en cualquier tipo de proceso diferente.

Es por ello que la codificación y los colores de los elementos se han escogido de tal manera que siguen los rasgos de la normativa ISA-S5.5. Así como también se ha utilizado una simbología de los elementos SCADA adecuada al estándar.

En las siguientes figuras se encuentran en detalle la codificación e identificación de los elementos (figura 63) así como el código de colores utilizado en el proyecto (figura 62).

Color	Generic meaning	Element association
Black	Background	
Red	Emergency	A) Stop B) Highest Priority Alarm C) Closed D) Off
Yellow	Caution	A) Abnormal Condition B) Second Priority Alarm
Green	Safe	A) Normal Operation B) Start C) Open D) On
Cyan (Light Blue)	Static & Significant	A) Process Equipment in Service B) Major Labels
Blue	Nonessential	A) Standby Process Equipment B) Labels, Tags, etc.
Magenta (Purple)	Radiation	A) Radiation Alarms B) Questionable Values
White	Dynamic Data	A) Measurements & State Information B) System Messages C) Trend D) Active Sequential Step

Figura 62. Tabla de identificación de estados según color.

De la tabla, y en este caso en particular se han utilizado los colores rojo para alarmas, verde para el estado de marcha, gris para el estado de paro, amarillo para el estado indeterminado y azul para el estado de espera.

Table B-1: Identification letters

First letter (4)		Succeeding letters (3)			
Measured or initiating variable	Modifier	Readout or passive function	Output function	Modifier	
A	Analysis (5, 19)	Alarm			
B	Burner, Combustion	User's Choice (1)	User's Choice (1)	User's Choice (1)	
C	User's Choice (1)		Control (13)		
D	User's Choice (1)	Differential (4)			
E	Voltage	Sensor (Primary Element)			
F	Flow Rate	Ratio (Fraction) (4)			
G	User's Choice (1)	Glass, Viewing Device (9)			
H	Hand			High (7, 15, 16)	
I	Current (Electrical)	Indicate (10)			
J	Power	Scan (7)			
K	Time, Time Schedule	Time Rate of Change (4, 21)	Control Station (22)		
L	Level			Low (7, 15, 16)	
M	User's Choice (1)	Momentary (4)		Middle, Intermediate (7, 15)	
N	User's Choice (1)		User's Choice (1)	User's Choice (1)	
O	User's Choice (1)	Orifice, Restriction			
P	Pressure, Vacuum	Point (Test) Connection			
Q	Quantity	Integrate, Totalize (4)			
R	Radiation		Record (17)		
S	Speed, Frequency	Safety (8)		Switch (13)	
T	Temperature			Transmit (18)	
U	Multivariable (6)		Multifunction (12)	Multifunction (12)	Multifunction (12)
V	Vibration, Mechanical Analysis			Valve, Damper, Louver (13)	
W	Weight, Force		Well		
X	Unclassified (2)	X Axis	Unclassified (2)	Unclassified (2)	Unclassified (2)
Y	Event, State or Presence (20)	Y Axis		Relay, Compute, Convert (13, 14, 18)	
Z	Position Dimension	Z Axis		Driver, Actuator, Unclassified Final Control Element	

Figura 63. Tabla de significado de los tags según letra.

Por otro lado, si hacemos un repaso a los nombres de los tags según las letras se puede fácilmente identificar sensores o válvulas, son un ejemplo:

FIT – Flow Rate Indicator Transmitter

FCV – Flow Rate Control Valve

LI – Level Indicator

Esta identificación no es posible con la maquinaria, debido a que los tags se han escogido de tal manera que facilitan la comprensión y disminuyen el tiempo de programación del controlador.

(espacio deliberadamente en blanco)

5.2.2. Guía GEDIS

Se ha tenido en cuenta la guía GEDIS (Guía Ergonómica de Diseño de Interfaces de Supervisión), que tiene como finalidad la evaluación de diez indicadores que permiten al usuario valorar la interfaz de supervisión. Por otro lado, muestra un método de diseño especializado en sistemas de control supervisor industrial basado en niveles donde se van concretando los diseños de los distintos tipos de pantalla y contenidos.

Cada uno de los indicadores se mide en una escala de 1 (muy pobre) a 5 (muy detallado), al igual que la evaluación global, y tiene un peso idéntico de cara a la evaluación global.

Tabla 20. Evaluación de los indicadores.

Núm.	Indicador	Descripción	Punt.	Punt.
			Experto	Usuario
1	Estructura	Organización jerárquica de las pantallas.	4	4
2	Distribución de pantallas	Plantillas de los diferentes tipos de pantalla.	2	4
3	Navegación	Modos de navegación entre pantallas.	4	4
4	Uso de color	Asociación de funcionalidades en el ámbito del control de procesos.	3	4
5	Uso de fuentes e información textual	Abanico de fuentes y asociación de funcionalidades.	2	2
6	Estatus de los equipos y eventos de proceso	Símbolos e iconos gráficos para representar el estado de la planta y los cambios	4	4

		de estado.		
7	Información y valores de proceso	Presentación de los datos analógicos/digitales en los gráficos.	3	4
8	Gráficos de tendencias y tablas	Presentación y agrupación de valores en gráficos de tendencias (históricos) y tablas.	4	4
9	Comandos y entradas de datos	Modo de entrada de datos a la interfaz.	3	3
10	Alarmas	Características principales del subsistema de alarmas.	3	4
	Evaluación global:		3,2	3,7
	Media:			3,5

Se han hecho dos tipos de evaluaciones de forma numérica cualitativa debido a que estas dependen subjetivamente de la persona evaluadora, una por un experto y otra por un usuario de campo.

Las evaluaciones globales tienen por puntuación de un 3,2 realizada por el experto y 3,7 realizada por el operario. Con este resultado se llega a la conclusión que es posible proponer medidas de mejora para acercarse a los cinco puntos en el futuro. Con una evaluación posterior y llegando a mejorar los aspectos en los que aquellos indicadores son actualmente bajos.

6. Análisis de impacto ambiental

Tanto en simulaciones como en plantas reales existe la posibilidad de averías, accidentes, etc. en los elementos que componen este sistema de producción de aceite de oliva. Por ello se ha dispuesto de alarmas de control y funcionamiento en todas las máquinas, así como sistemas de control que permiten el paro inmediato de los elementos.

Fuera del ámbito de la maquinaria y en funcionamiento normal de una planta real han de existir análisis y controles de calidad que verifiquen los parámetros de calidad, que al trabajar con simulación no se contemplan por este mismo hecho.

Conclusiones

Finalizado el proyecto, se puede concluir que se ha cumplido satisfactoriamente el objetivo principal de diseñar e implementar una solución del proceso de producción de aceite de oliva, basada en la integración de Controladores Lógicos Programables (PLC) y Sistemas de Supervisión, Control y Adquisición de Datos (SCADA) definido al principio de esta memoria.

Se ha conseguido también el objetivo principal de hacer funcionar todo el proceso y planta creados desde cero. De principio a fin, en modo manual y automático, supervisando los parámetros reales y los elementos fundamentales que actúan en el proceso. En cuanto a los objetivos secundarios se han podido efectuar algunos como la creación de sistemas osciladores, Add-Ons, gestión de usuario, dar aleatoriedad en el proceso de extracción de aceite entre otros.

Dejando a un lado los objetivos, puedo decir indudablemente, que la creación de un proyecto de tal índole ha significado la asimilación de muchos conceptos nuevos y la consolidación de muchos otros. Se ha de considerar la gran cantidad de trabajo, la dedicación que requiere la automatización de procesos, la cantidad de complicaciones y dificultades que se encuentra uno al realizar un proyecto de tal dimensión. Pero también se ha de considerar la gran emoción al completar un proyecto así y el valor añadido cuando funciona correctamente.

Trabajo Futuro

Como trabajo futuro, primeramente se pueden implementar mejoras y algunos de los objetivos secundarios que no se han podido realizar, o incluir más parámetros de control para aumentar el nivel de calidad y realidad de esta planta de producción de aceite de oliva.

Por otra parte, sería interesante implementar el proyecto en una placa electrónica con los I/O del controlador real, o incluso llegar a poder implementarlo en una mini-almazara para ver el proceso de forma física y no simulada.

Bibliografía

- [1] Hermoso, Manuel; González, Jesús; Uceda, Marino; García-Ortiz, Angel; Morales, Juan; Frías, Luisa; Fernández, Ángel. *Elaboración de aceite de oliva de calidad. Obtención por el sistema de dos fases*, [en línea]. 3ª edición. Edita: Junta de Andalucía. Comunidad Europea. ISBN.: 84-89802-37-8. Recurso electrónico disponible en: https://www.juntadeandalucia.es/export/drupaljda/1337166142Elaboracixn_de_aceite.pdf
- [2] Coell, Cati. *Un paseo por la industria del oro líquido*, [en línea]. Mayo 2006. Recurso electrónico disponible en: <https://metode.es/revistas-metode/monograficos/un-paseo-por-la-industria-del-oro-liquido.html>
- [3] Provedo. *Curso de elaboración de aceite de oliva virgen y almazaras*, [en línea]. Recurso electrónico disponible en: <http://www.variedadesdeolivo.com/cursos/Curso%20de%20olivicultura.%20Elaboracion%20a ceite%20oliva%20y%20almazara.pdf>
- [4] Oleodiel. *Proceso productivo del aceite de oliva virgen extra*, [en línea]. Recurso electrónico disponible en: <https://oleodiel.com/proceso-productivo-del-aceite-de-oliva-virgen-extra-aove/>
- [5] Sy , Helmut. *Automatización industrial: historia, características y tipos*, [en línea]. Recurso electrónico disponible en: <https://www.lifeder.com/automatizacion-industrial/#Historia>
- [6] Geinfor. *¿Qué es la Industria 4.0?*, [en línea]. Recurso electrónico disponible en: <https://geinfor.com/blog/industria-40/>
- [7] Anónimo. *Cronología de la evolución de la automatización*, [en línea]. Recurso disponible en: <https://www.timetoast.com/timelines/la-evolucion-de-la-automatizacion>
- [8] Ponsa, Pere; Díaz, Marta; Catalá, Andreu. *Creación de guía ergonómica para el diseño de interfaz de supervisión*, [en línea]. UPC. Recurso electrónico disponible en: <https://aipo.es/articulos/4/4.pdf>
- [9] Departamento de Ciencias de la Computación e IA (DECSAI). *Especificación de requerimientos*, [en línea]. Universidad de Granada. Recurso electrónico disponible en: <elvex.ugr.es/idbis/db/docs/design/2-requirements.pdf>
- [10] Organización Internacional de Normalización (ISO). *Búsqueda de estándares ISO*, [en línea]. Recurso electrónico disponible en: <https://www.iso.org/obp/ui#home>
- [11] Organización Internacional de Normalización (ISO). *Búsqueda del estándar ISO 11064*, [en línea]. Recurso electrónico disponible en: <https://www.iso.org/obp/ui/#iso:std:iso:11064:-1:ed-1:v1:en>
- [12] Comision Electrotecnica Internacional (IEC). *Búsqueda del estándar IEC 61131*, [en línea]. Recurso electrónico disponible en: <https://webstore.iec.ch/webstore/webstore.nsf/mysearchajax?Openform&key=iec%2061131&sorting=&start=1&onglet=1>
- [13] Gámiz, Javier. Lluva, Marc. *Tutorial de comunicaciones mediante DDE*. ISA 295202. UPC 2018.
- [14] Gámiz, Javier. Lluva, Marc. *Sesiones teórico-prácticas de la asignatura. Sesiones 1-11*. ISA 295202. UPC 2018.
- [15] Gámiz, Javier. *Fichero de intercambio. Proyecto PBL*. ISA 295202. UPC 2018.



Manuales de usuario

- [16] Rockwell Automation. *Guía de Inicio rápido para controladores Logix5000*. Rockwell Automation. Rev. Marzo 2004.
- [17] Rockwell Automation. *Procedimientos comunes de los controladores Logix5000*. Rockwell Automation. Rev. Agosto 2002.
- [18] Rockwell Automation. *Instrucciones generales de controladores Logix5000*. Rockwell Automation. Rev. Mayo 2002.
- [19] Rockwell Automation. *Manual del usuario del sistema ControlLogix*. Rockwell Automation. Rev. Agosto 2002.
- [20] Rockwell Automation. *Manual del usuario de los controladores CompactLogix1769*. Rockwell Automation. Rev. Febrero 2013.
- [21] Rockwell Automation. *PhaseManager User Manual*. Rockwell Automation. Rev. Abril 2010.
- [22] Rockwell Automation. *Logix5000 Controllers Add-On Instructions*. Rockwell Automation. Rev. Julio 2008.
- [23] Rockwell Automation. *Emulate 5000, Processor Emulation*. Rockwell Automation. Rev. Abril 2005.
- [24] AVEVA. *Wonderware InTouch HMI SmartSymbols Guide*. AVEVA group. Rev. Julio 2008.
- [25] AVEVA. *InTouch HMI Concepts and Capabilities Guide*. AVEVA group. Rev. Noviembre 2008.