





**Ministerio de Educación Superior  
Instituto Superior Minero Metalúrgico de Moa  
“Dr. Antonio Núñez Jiménez”  
Facultad Metalurgia – Electromecánica**

---

Trabajo de diploma

## Trabajo presentado con el fin de obtener el título de Ingeniero Eléctrico

Título: Interfaz para comunicación de sistema SCADA de accionamientos con variadores de velocidad.

Autor: Dannys Fonseca Alpajón.

Tutor: Dr. Luís Delfín Rojas Purón.

## **Pensamiento**

En lo tocante a la ciencia, la autoridad de un millar no es superior al humilde razonamiento de una sola persona.

Albert Einstein.

## Dedicatoria

A mi Tutor,  
A mis Padres,

Y a todas aquellas personas que de alguna forma nos ayudaron en la realización de esta Tesis.

## **Agradecimiento**

Al taller de instrumentación de la Empresa “Comandante Pedro Sotto Alba MOA NÍQUEL.SA”. Al colectivo de desarrollo “EMPLENI”, y otros. En especial a los especialistas de control automático.

## **Resumen**

En este trabajo se implementa el montaje físico de una maqueta del control de proceso en una planta real para que el laboratorio de accionamientos de la carrera de Ingeniería Eléctrica realice prácticas, además de observar un proceso industrial controlado gráficamente a escala menor.

Para ello se realiza un flujo tecnológico y se procede a la creación de una interfase de programación mediante lazos de control con lógica difusa obteniéndose una regulación continua del proceso.

## **Summary**

In this work the physical assembly of a scale model of the process control is implemented in a real plant so that the laboratory of workings of the career of Electric Engineering carries out practical, besides observing an industrial process controlled graphically to smaller scale.

For he is carried out it a technological flow and you proceeds to the creation of a programming interface by means of control knots with diffuse logic being obtained a continuous regulation of the process.

## Índice general

<a href="#">Introducción general.</a>	1
<a href="#">Capítulo I: Accionamientos con variadores de velocidad.</a>	
<a href="#">1.1 Generalidades.</a>	
<a href="#">1.2 Características de los variadores de velocidad.</a>	
<a href="#">1.3 Tipos de variadores de velocidad.</a>	
<a href="#">1.3.1 Variadores mecánicos.</a>	
<a href="#">1.3.2 Variadores hidráulicos.</a>	
<a href="#">1.3.3 Variadores eléctrico-electrónicos.</a>	
<a href="#">1.3.4 Variadores de velocidad eléctrico-electrónicos.</a>	
<a href="#">1.3.5 Variadores para motores de CC.</a>	
<a href="#">1.3.6 Variadores por corrientes de Eddy.</a>	
<a href="#">1.3.7 Variadores de deslizamiento.</a>	
<a href="#">1.3.8 Variadores para motores de CA.</a>	
<a href="#">1.4 velocidad como una forma de controlar un proceso.</a>	
<a href="#">1.4.1 Regulación en lazo abierto.</a>	
<a href="#">1.4.2 Regulación en lazo cerrado.</a>	
<a href="#">1.4.3 Tipos de sistemas de control.</a>	
<a href="#">1.4.3.1 Regulador ON - OFF</a>	
<a href="#">1.4.3.2 Regulador PID.</a>	
<a href="#">1.4.3.3 Control multivariable.</a>	
<a href="#">1.4.3.4 Control difuso.</a>	
<a href="#">1.5 Conclusiones parciales.</a>	
<a href="#">Capítulo II: Fundamentos sobre adquisición de datos.</a>	
<a href="#">2.1 Generalidades.</a>	
<a href="#">2.2 Sistemas SCADA.</a>	
<a href="#">2.2.1 Prestaciones.</a>	
<a href="#">2.2.2 Requisitos.</a>	
<a href="#">2.2.3 Componentes del sistema.</a>	
<a href="#">2.3 Fundamentos sobre adquisición de datos del programa.</a>	
<a href="#">2.3.1 Proceso de adquisición de datos.</a>	
<a href="#">2.4 Fundamentos sobre adquisición de datos del PLC.</a>	
<a href="#">2.4.1 Programa para la realización de la programación en el PLC.</a>	
<a href="#">2.4.1.1 Funcionalidad del software.</a>	
<a href="#">2.5 Conclusiones parciales.</a>	
<a href="#">Capítulo III: Técnicas para interfaces con los accionamientos eléctricos.</a>	
<a href="#">3.1 Generalidades.</a>	
<a href="#">3.2 Interfaz de usuario.</a>	
<a href="#">3.2.1 Datos técnicos de comunicación.</a>	
<a href="#">3.2.2 Funciones principales de la interfaz de usuario.</a>	
<a href="#">3.2.3 Tipos de interfaces de usuario.</a>	
<a href="#">3.3 Interfaces de Citect para comunicarse con el autómatas PLC.</a>	
<a href="#">3.4 Interfaz PLC- variador de velocidad.</a>	
<a href="#">3.5 Conclusiones parciales.</a>	
<a href="#">Capítulo IV: Programa supervisor en accionamientos.</a>	
<a href="#">4.1 Generalidades.</a>	

<b><u>4.2 Flujo tecnológico del proceso.</u></b>	
<u>4.2.1 Señales que intervienen en el proceso.</u>	
<u>4.2.2 Red del sistema autómeta.</u>	
<b><u>4.3 Programación del SCADA.</u></b>	
<u>4.3.1 Comunicación con el Citect SCADA.</u>	
<u>4.3.2 Visualización gráfica del proceso.</u>	
<u>4.3.2.1 Insertación de las diferentes variables tag que intervienen en la visualización gráfica del proceso en la planta.</u>	
<b><u>4.4 Programación del PLC.</u></b>	
<u>4.4.1 Programa STEP 7.</u>	
<u>4.4.2 Declaración de bloques programados en el PLC.</u>	
<b><u>4.5 Conclusiones parciales.</u></b>	
<b><u>Capítulo V: Valoración técnica económica.</u></b>	
<b><u>5.1 Generalidades.</u></b>	
<b><u>5.2 Costos de la instalación.</u></b>	
<b><u>5.3 Impacto social.</u></b>	
<b><u>Conclusiones y recomendaciones.</u></b>	
<b><u>Bibliografía.</u></b>	
<b><u>Anexos.</u></b>	

## Introducción general:

La automatización industrial se ha convertido hoy en día en un área en pleno desarrollo y de gran importancia para todos los sectores industriales, ya que permite mejorar la productividad de las empresas, la calidad de sus productos, la flexibilidad y la seguridad laboral. Por ende la comunicación industrial, necesaria en todos los ámbitos de la empresa, es uno de los campos de desarrollo más activos de la automatización.

Las tendencias actuales más importantes en comunicaciones están básicamente focalizadas en el desarrollo del control automático. Este desde su surgimiento ha desempeñado una función vital en el avance de la ingeniería y la ciencia, en la actualidad el desarrollo del mismo se ha vuelto una parte importante e integral de los procesos modernos industriales. La amplia y progresiva implantación de los accionamientos eléctricos en la industria y las posibilidades de automatización y control y de mejora de la eficiencia de los procesos productivos constituyen la base del presente estudio.

Nuestro trabajo está orientado al estudio, análisis del funcionamiento y control de las máquinas y los accionamientos eléctricos, de manera específica el diseño de una interfase para el monitoreo y supervisión de accionamientos eléctricos con variadores de velocidad, asistidos por PLC en condiciones de los laboratorios docentes para el estudio de los Accionamientos Eléctricos en el Instituto Superior Minero Metalúrgico de Moa. Siendo el objetivo primordial elaborar una interfase [PC – PLC – Tarjeta – Variador velocidad] para el monitoreo, control y la supervisión de accionamientos eléctricos con variadores de velocidad a partir de varios criterios de diseño del software SCADA.

Los sistemas SCADA utilizan la computadora y tecnologías de comunicación para automatizar el monitoreo y control de procesos industriales. Estos sistemas son partes integrales de la mayoría de los ambientes industriales complejos o muy geográficamente dispersos, ya que pueden recoger la información de una gran cantidad de fuentes muy rápidamente, mejorando la eficacia del proceso de monitoreo y control, proporcionando la información oportuna para poder tomar decisiones operacionales apropiadas.

## **Problema**

La ausencia de una interfase para el monitoreo, control y supervisión de accionamientos eléctricos con variadores de velocidad, asistidos por PLC en condiciones de los laboratorios docentes de Accionamientos Eléctricos y Electrónica de Potencia en el ISMM de Moa.

## **Hipótesis**

Con la identificación de las variables y los diferentes protocolos de comunicación que se tienen a disposición en los laboratorios de la carrera de Ing. Eléctrica, en accionamientos con variadores de velocidad asistidos por PLC, es posible elaborar una tarjeta que permita la supervisión y control de estos sistemas autómatas.

## **Objetivos**

Elaborar una interfase [PC – PLC – Tarjeta – Variador velocidad] para el monitoreo, control y la supervisión de accionamientos eléctricos con variadores de velocidad a partir de varios criterios de diseño SCADA (Supervisión Control y Adquisición de Datos) conocidos.

## **Resultados Esperados**

- Elaborar la matriz de variables que participan en el comportamiento de los accionamientos con variadores de velocidad.
- Ofrecer una interfase para el monitoreo, control y supervisión de los accionamientos con variadores de velocidad a partir de varios criterios utilizados internacionalmente.
- Brindar una base experimental bien ordenada que permita elaborar un sistema para la supervisión de los accionamientos.
- Desarrollar un caso de estudio para la validación de los accionamientos a través de un sistema hecho desde CITEC.

# CAPÍTULO I

## Capítulo I: Accionamientos con variadores de velocidad.

### 1.1 GENERALIDADES.

Una gran parte de los equipos utilizados en la industria moderna funcionan a velocidades variables, como por ejemplo los trenes laminadores, los mecanismos de elevación, las máquinas-herramientas, etc. En los mismos se requiere un control preciso de la velocidad para lograr una adecuada productividad, una buena terminación del producto elaborado, o garantizar la seguridad de personas y bienes.

El alto grado de automatización de los sistemas productivos en las industrias, ha conllevado a un avanzado desarrollo en el control de los procesos, destacándose en éstos el monitoreo y control de los accionamientos eléctricos, así como controlar la velocidad giratoria de las maquinarias, especialmente de los motores eléctricos.

El control de procesos y el ahorro de la energía son dos de las principales razones para el empleo de variadores de velocidad, logrando así procesos más rápidos, más confiables, más eficientes y con mejor relación costo/beneficio.

## 1.2 CARACTERÍSTICAS DE LOS VARIADORES DE VELOCIDAD.

El **Variador de Velocidad** (VSD, *Variable Speed Drive*) es en un sentido amplio un dispositivo o conjunto de dispositivos mecánicos, hidráulicos, eléctricos o electrónicos empleados para controlar la velocidad giratoria de maquinaria, especialmente de motores. También es conocido como Accionamiento de Velocidad Variable (ASD, *Adjustable-Speed Drive*).

La maquinaria industrial generalmente es accionada a través de motores eléctricos, a velocidades constantes o variables, pero con valores precisos. No obstante, los motores eléctricos generalmente operan a velocidad constante o casi-constante, y con valores que dependen de la alimentación y de las características propias del motor, los cuales no se pueden modificar fácilmente.

Para lograr regular la velocidad de los motores, se emplea un controlador especial que recibe el nombre de variador de velocidad. Los variadores de velocidad se emplean en una amplia gama de aplicaciones industriales, como ventiladores, equipos de bombeo, bandas transportadoras industriales, elevadores, tornos y fresadoras, etc.

Un variador de velocidad puede consistir en la combinación de un motor eléctrico y el controlador que se emplea para regular la velocidad del mismo. La combinación de un motor de velocidad constante y de un dispositivo mecánico que permita cambiar la velocidad de forma continua (sin ser un motor paso a paso) también puede ser designado como variador de velocidad.

## 1.3 TIPOS DE VARIADORES DE VELOCIDAD.

En términos generales, puede decirse que existen tres tipos básicos de variadores de velocidad: mecánicos, hidráulicos y eléctrico-electrónicos. Dentro de cada tipo pueden encontrarse más subtipos, que se detallarán a continuación. Cabe aclarar que los variadores más antiguos fueron los mecánicos, que se emplearon originalmente para controlar la velocidad de las ruedas hidráulicas de molinos, así como la velocidad de las máquinas de vapores.

Los variadores de velocidad mecánicos e hidráulicos generalmente son conocidos como transmisiones cuando se emplean en vehículos, equipo agroindustrial u otro tipo de maquinaria.

### 1.3.1 Variadores mecánicos.

- Variadores de paso ajustable: estos dispositivos emplean poleas y bandas en las cuales el diámetro de una o más poleas puede ser modificado.
- Variadores de tracción: transmiten potencia a través de rodillos metálicos. La relación de velocidades de entrada/salida se ajusta moviendo los rodillos para cambiar las áreas de contacto entre ellos y así la relación de transmisión.



Fig.1 Variador de paso ajustable.



Fig.2 Variador de tracción.

### 1.3.2 Variadores hidráulicos.

- Variador hidrostático: consta de una bomba hidráulica y un motor hidráulico (ambos de desplazamiento positivo). Una revolución de la bomba o el motor corresponde a una cantidad bien definida de volumen del fluido manejado. De esta forma la velocidad puede ser controlada mediante la regulación de una válvula de control, o bien, cambiando el desplazamiento de la bomba o el motor.
- Variador hidrodinámico: emplea aceite hidráulico para transmitir par mecánico entre un impulsor de entrada (sobre un eje de velocidad constante) y un rotor de salida (sobre un eje de velocidad ajustable). También llamado acoplador hidráulico de llenado variable.

- Variador hidrovicoso: consta de uno o más discos conectados con un eje de entrada, los cuales estará en contacto físico (pero no conectados mecánicamente) con uno o más discos conectados al eje de salida. El par mecánico (torque) se transmite desde el eje de entrada al de salida a través de la película de aceite entre los discos. De esta forma, el par transmitido es proporcional a la presión ejercida por el cilindro hidráulico que presiona los discos.

### **1.3.3 Variadores eléctrico-electrónicos.**

Existen cuatro categorías de variadores de velocidad eléctrico-electrónicos.

- variadores para motores de CC
- variadores de velocidad por corrientes de Eddy
- variadores de deslizamiento
- variadores para motores de CA conocidos como variadores de frecuencia.

### **1.3.4 Variadores de velocidad eléctrico-electrónicos.**

Los variadores eléctrico-electrónicos incluyen tanto el controlador como el motor eléctrico, sin embargo es práctica común emplear el término variador únicamente al controlador eléctrico.

Los primeros variadores de esta categoría emplearon la tecnología de los tubos de vacío. Con los años después se han ido incorporando dispositivos de estado sólido, lo cual ha reducido significativamente el volumen y costo, mejorando la eficiencia y confiabilidad de los dispositivos.

### **1.3.5 Variadores para motores de CC.**

Estos variadores permiten controlar la velocidad de motores de Corriente continua, serie, derivación, compuesto y de imanes permanentes. Para el caso de cualquiera de las máquinas anteriores se cumple la siguiente expresión:

$$\text{Donde } V_t = K \cdot FM \cdot Nm$$

$V_t$  Voltaje terminal (V).

$K$  Constante de la máquina.

$FM$  Flujo magnético producido por el campo (Wb)

$Nm$  Velocidad mecánica (rpm).

Despejando la velocidad mecánica, se obtiene: 
$$Nm = \frac{V_t}{K \cdot FM}$$

Entonces, de (2) puede observarse que la velocidad mecánica de un motor de CC es directamente proporcional al voltaje terminal (VT) e inversamente proporcional al flujo magnético (FM), el cual a su vez depende de la corriente de campo (IF). Aprovechando esta situación es que este tipo de variadores puede controlar la velocidad de un motor de CC: controlando su voltaje terminal, o bien, manipulando el valor de la corriente de campo.

### **1.3.6 Variadores por corrientes de Eddy.**

Un variador de velocidad por corrientes de Eddy consta de un motor de velocidad fija y un embrague de corrientes de Eddy. El embrague contiene un rotor de velocidad fija (acoplado al motor) y un rotor de velocidad variable, separados por un pequeño entrehierro. Se cuenta, además, con una bobina de campo, cuya corriente puede ser regulada, la cual produce un campo magnético que determinará el par mecánico transmitido del rotor de entrada al rotor de salida. De esta forma, a mayor intensidad de campo magnético, mayor par y velocidad transmitidos, y a menor campo magnético menores serán el par y la velocidad en el rotor de salida. El control de la velocidad de salida de este tipo de variadores generalmente se realiza por medio de lazo cerrado, utilizando como elemento de retroalimentación un tacómetro de CA.

### **1.3.7 Variadores de deslizamiento.**

Este tipo de variadores se aplica únicamente para los motores de inducción de rotor devanado. En cualquier motor de inducción, la velocidad mecánica ( $n_m$ ) puede determinarse mediante la siguiente expresión:

$$N_m = \frac{120 \cdot f \cdot (1 - s)}{P}$$

Donde  $s$  es el deslizamiento del motor, cuyo valor oscila entre 0 y 1. De esta forma, a mayor deslizamiento, menor velocidad mecánica del motor. El deslizamiento puede incrementarse al aumentar la resistencia del devanado del rotor, o bien, al reducir el voltaje en el devanado del rotor. De esta forma es que puede conseguirse el control de la velocidad en los motores de inducción de rotor devanado. Sin embargo, este tipo de variadores es de menor eficiencia que otros, razón por la cual en la actualidad tiene muy poca aplicación.

La figura 3 presenta el esquema convencional de un controlador de deslizamiento para un motor de inducción, utilizado en variadores de velocidad sin gran exigencia de prestaciones. Tradicionalmente, el error de velocidad ( $\omega_e$ ) es la entrada del regulador PI, que establece la frecuencia de deslizamiento del motor ( $\omega_r$ ).

La frecuencia estática resulta de la adición de la frecuencia de deslizamiento a la velocidad del rotor ( $\omega$ ). La tensión del estator ( $U_s$ ) es establecida de acuerdo con una ley predefinida ( $U_s/\omega_s$  aproximadamente constante), de forma que el flujo del motor sea mantenido en su valor nominal. La frecuencia de deslizamiento es limitada,

estableciendo (indirectamente) un límite para el par máximo disponible en el motor y para la corriente del estator.

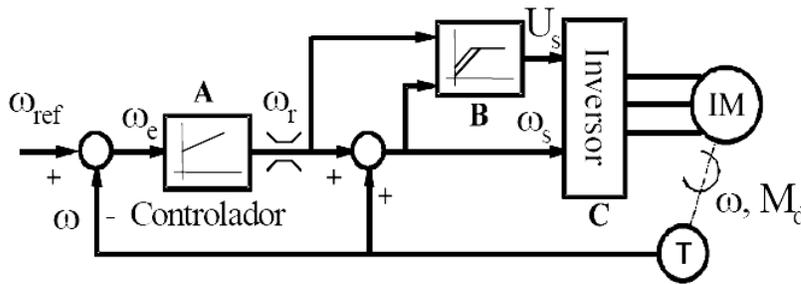


Fig.3 Esquema convencional de un controlador de deslizamiento.

### 1.3.8 Variadores para motores de CA.

Los variadores de frecuencia ( AFD, *Adjustable Frequency Drive*; o bien VFD *Variable Frequency Drive*) permiten controlar la velocidad tanto de motores de inducción (asíncronos de jaula de ardilla o de rotor devanado), como de los motores síncronos mediante el ajuste de la frecuencia de alimentación al motor.

- Para el caso de un motor síncrono, la velocidad se determina mediante la siguiente expresión:

$$N_s = \frac{120 \cdot f}{P}$$

- Cuando se trata de motores de inducción, se tiene:

Donde: 
$$N_m = \frac{120 \cdot f \cdot (1 - s)}{P}$$

$N_s$  = velocidad síncrona (rpm)

$N_m$  = velocidad mecánica (rpm)

$f$  = frecuencia de alimentación (Hz)

$s$  = deslizamiento (adimensional)

$P$  = número de polos.

Como puede verse en las expresiones (4) y (5), la frecuencia y la velocidad son directamente proporcionales, de tal manera que al aumentar la frecuencia de alimentación al motor, se incrementará la velocidad de la flecha, y al reducir el valor de la frecuencia disminuirá la velocidad del eje. Por ello es que este tipo de variadores manipula la frecuencia de alimentación al motor a fin de obtener el control de la velocidad de la máquina

Estos variadores mantienen la razón Voltaje/ Frecuencia (V/Hz) constante entre los valores mínimo y máximos de la frecuencia de operación, con la finalidad de evitar la saturación magnética del núcleo del motor y además porque el hecho de operar el motor a un voltaje constante por encima de una frecuencia dada (reduciendo la relación V/Hz) disminuye el par del motor y la capacidad del mismo para proporcionar potencia constante de salida.

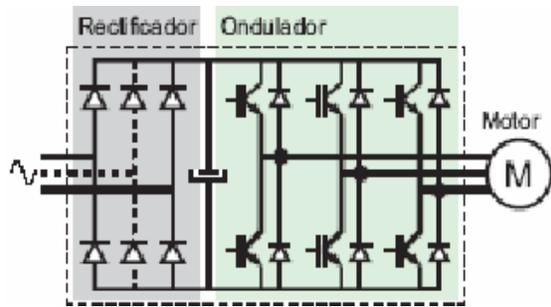


Fig.4 Esquema eléctrico de un Convertidor de frecuencia.

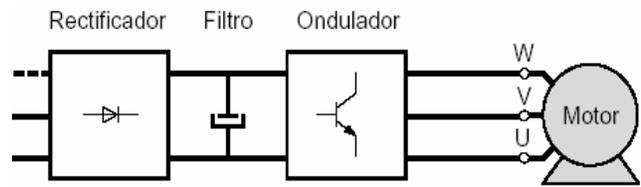


Fig.5 Diagrama de bloques de un Convertidor de frecuencia.

## 1.4 VELOCIDAD COMO UNA FORMA DE CONTROLAR UN PROCESO.

Se entiende como procesos continuos aquellos cuya magnitud a regular varia de forma continua en el tiempo, pudiendo pasar por infinitos valores. Por ejemplo, una regulación del caudal de agua puede ser muy importante en una central hidroeléctrica, para poder ajustar el suministro de energía a la demanda. En un salto de agua, la potencia se regula por el caudal (es presión por caudal), siendo prácticamente constante la presión (la altura del salto de agua es constante).

El mundo real es analógico (continuo), como también lo son la mayoría de los procesos industriales, hasta el punto que resulta difícil encontrar uno en el que no exista regulación de alguna magnitud. Por ejemplo, se regula posición en cada uno de los carros de una máquina de control numérico, en la regulación de una válvula de tubería que pueden ocupar múltiples funciones u otros.

Una larga lista puede citarse igualmente para otras magnitudes como velocidad, presión, fuerza, caudal, potencia, nivel, temperatura, tensión, intensidad. En todo lo anterior no se ha mencionado la tecnología de control, así por ejemplo, una válvula puede regular caudal de forma mecánica, neumática, electrónica..., o varias formas combinadas. Difícilmente podríamos hacer una clasificación de los reguladores, que en muchas ocasiones ni siquiera existen físicamente porque son un mero algoritmo que forma parte del programa en un controlador digital.

El problema principal de la regulación es que depende fundamentalmente del proceso. Podemos conocer qué valores debemos ajustar en unas circunstancias concretas, pero tarde o temprano el proceso es alterado, bien porque se necesita atender una demanda a gusto del cliente, bien para introducir mejoras como aumentar la velocidad de producción, o bien porque aparecen perturbaciones debidas a irregularidades, desgastes, cambios de temperatura o quién sabe qué otros factores.

Entre las diversas ventajas en el control del proceso proporcionadas por el empleo de variadores de velocidad se destacan:

- Operaciones más suaves.
- Control de la aceleración.
- Distintas velocidades de operación para cada fase del proceso.
- Compensación de variables en procesos variables.
- Permitir operaciones lentas para fines de ajuste o prueba.
- Ajuste de la tasa de producción.
- Permitir el posicionamiento de alta precisión.
- Control del Par motor (torque).

### 1.4.1 Regulación en lazo abierto.

Un ejemplo cotidiano lo tenemos en la ducha de una vivienda. Al girar determinado ángulo las llaves de agua fría y caliente, la temperatura crece lentamente hasta llegar a un equilibrio, pero no existe otro sensor de temperatura que nosotros mismos. Para que el sistema dejase de funcionar en lazo abierto, sería necesario medir la temperatura y compararla con otra de consigna o de referencia, considerada agradable al contacto.

Si la temperatura medida es mayor que la de consigna, la llave de agua caliente tendría que cerrarse parcialmente de forma automática y la de agua fría se abriría parcialmente, esto sería necesario para reducir la temperatura a la vez que se mantiene el caudal. Posiblemente, la corrección sería excesiva y tendría que hacerse en sentido inverso, hasta que después de varias aproximaciones quedaría estabilizada la temperatura y el caudal. Si lo pensamos, esto es lo que hacemos manualmente, asumiendo la función del sensor de temperatura y del regulador de apertura de las llaves de paso. Si en lugar de agua para la ducha se tratase de un disolvente industrial, la regulación en lazo abierto sería inadmisibles, puesto que debe mantenerse la temperatura óptima sin oscilaciones, de ello dependerá su eficacia. Este es el problema que más dificulta una regulación: El tiempo muerto. Hay tiempo muerto siempre que el proceso responde con retraso, por ejemplo, al abrir la llave de agua caliente, el aumento de temperatura se manifiesta después de que el agua ha recorrido la tubería y al abrir la llave de agua fría, la temperatura desciende solo cuando se ha consumido el agua caliente que ya estuviera en circulación. Una regulación rápida, precisa y sin oscilaciones, solo es posible cuando se puede predecir cómo responderá el proceso, de forma que se corrijan las desviaciones antes de que aparezcan, o de lo contrario ya será demasiado tarde. Hay diversas formas de predecir la respuesta de un proceso, siendo la más exacta (y más difícil) la que se basa en su modelo matemático, pero la realidad es tan compleja que difícilmente se puede tener en cuenta todos los factores como son los desgastes, rozamientos, calentamientos, etc. La otra forma es utilizar un regulador que pueda ajustarse al proceso concreto que deberá controlar y su ajuste se conoce como sintonía de parámetros.

Dicha sintonía es experimental y existen técnicas que van desde la prueba y error hasta el análisis matemático basado en datos extraídos experimentalmente del proceso. La fig. 6 contiene un ejemplo con los que experimentar una regulación en lazo abierto. Lo importante es que se comprenda la problemática de la regulación en general y porqué es tan importante el proceso y las variaciones que experimenta.

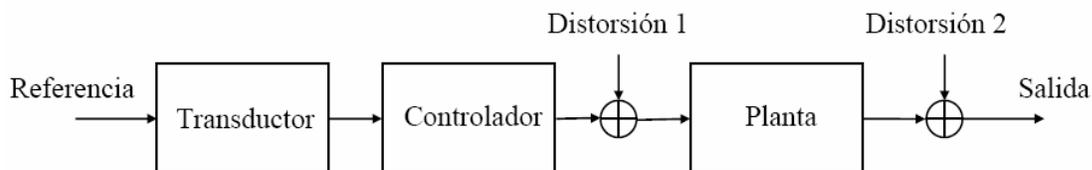


Fig. 6 Diagrama en bloque de un regulador lazo abierto.

- Sistemas de control extraordinariamente sencillos.
- Necesita una perfecta calibración de la planta.
- No puede compensar posibles distorsiones en el sistema.

#### **1.4.2 Regulación en lazo cerrado.**

Además de la dificultad de regulación en lazo abierto, el control en lazo cerrado es imprescindible cuando se da alguna de las siguientes circunstancias:

- Vigilar un proceso es especialmente duro en algunos casos y requiere una atención que el hombre puede perder fácilmente por cansancio o despiste, con los consiguientes riesgos que ello pueda ocasionar al trabajador y al proceso.
- A veces la regulación manual requiere mucha experiencia y esta es difícil de sustituir.
- Una producción a gran escala exige grandes instalaciones que el hombre no puede manejar. Cualquier desviación produce grandes pérdidas tanto por exceso de consumo de materias primas o energía como por disminución de la calidad del producto.
- Algunos procesos solo se pueden llevar a cabo por control automático, etc., y en general, todo proceso que no sea autorregulable.

El primer paso para poder regular es poder medir, por eso la evolución del control ha exigido el desarrollo de sensores y de las técnicas de medida y tratamiento de la señal. Actualmente se tiende a incluir dentro del propio sensor toda la electrónica que realiza el tratamiento de la señal.

En este sistema la señal de salida se toma como referencia, mediante el empleo de uno o varios sensores, e influye en la acción tomada por el sistema de control.

La señal de error actuante, que es la diferencia entre la señal de entrada y la de retroalimentación (que puede ser la señal de salida o una función de la señal de salida y sus derivadas), entra al controlador para reducir el error y llevar la salida a un valor deseado.

Esta retroalimentación se logra a través de la acción de un operador (control manual) o por medio de instrumentos (control automático). Los sistemas de control de lazo cerrado se llaman comúnmente sistemas de control por retroalimentación. Generalmente se dice que existe retroalimentación en un sistema cuando existe una secuencia cerrada de relaciones de causa y efecto ente las variables del sistema.

### **Características de la retroalimentación:**

- Aumento de la exactitud.
- Reducción de la sensibilidad de la salida, correspondiente a una determinada entrada, ante variaciones en las condiciones del sistema.
- Efectos reducidos de la no linealidad y de la distorsión.
- Tendencia a la oscilación o a la inestabilidad.

### **Ventajas sobre el control en lazo abierto:**

- Corrección de las perturbaciones.
- Permite estabilizar procesos inestables.
- Tolerancia a variaciones en los parámetros.
- Buen comportamiento, en aquellos casos en que la estructura del modelo no representa perfectamente la realidad del proceso o los parámetros del modelo no se pueden medir con absoluta precisión.

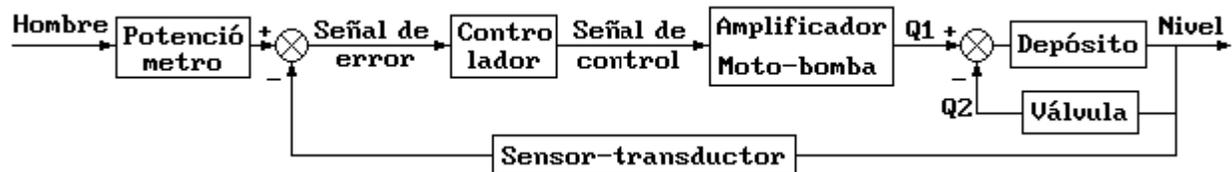


Fig. 7 Diagrama en bloque de un regulador en lazo cerrado.

Los diagramas de bloques se utilizan para representar gráficamente la relación que existe entre los componentes de un sistema, lo que veremos con un ejemplo de regulación de nivel en lazo cerrado. En el siguiente esquema, un comparador se encarga de hacer la diferencia entre la consigna del potenciómetro y la señal de realimentación o medida del sensor. Esta diferencia se denomina señal de error y es la única información que utiliza el controlador para regular el proceso. La ley de regulación más simple es la que iguala la salida del controlador (señal de control) con la señal de error, o bien un valor proporcional a la señal de error. El amplificador o variador, junto con el motor y bomba, convierten la señal de control en un caudal  $Q1$  que es proporcional a dicha señal. A medida que el nivel aumenta crece el caudal de salida  $Q2$  y la diferencia entre caudales disminuye. Al aumentar el nivel, medido por un sensor, hace que aumente la señal de realimentación y la señal de error disminuye. Cuando la medida y la consigna llegan a ser iguales, el error es 0 y el controlador transmite una señal de control que también será 0.

Esta ley de regulación proporcional al error tiene un defecto, el equilibrio se alcanza cuando  $Q_1$  y  $Q_2$  son iguales. Como estos caudales no son 0, el error tampoco será 0 en ningún momento, ya que tiene una relación directa con  $Q_1$  (son proporcionales). Por lo tanto, existe una falta de precisión que no se resuelve con una ley proporcional. La ley de regulación debe permitir que el error se anule (se necesita precisión), debe anticiparse a los tiempos muertos que existan en el sistema, los cuales son la principal causa de inestabilidades (oscilaciones en torno al punto de equilibrio), debe hacer que el sistema alcance el punto de consigna en el menor tiempo posible, debe contrarrestar las perturbaciones que afecten a la regulación, debe seguir fielmente a la señal de consigna aunque ésta sea variable... demasiadas especificaciones a cumplir con tan escasa información: el error. Sin embargo, ¿es posible!

### 1.4.3 Tipos de sistemas de control.

#### 1.4.3.1 Regulador ON - OFF

Es la regulación más simple y económica, interesante en numerosas aplicaciones en las que puede admitirse una oscilación continua entre dos límites, siempre y cuando se trate de procesos de evolución lenta. Como ejemplos podemos citar la regulación de nivel, de presión o de temperatura, todos ellos problemas relativamente sencillos de lógica digital que no tratamos en este tema. Numerosos reguladores incorporan esta función básica, que además ofrece la máxima rapidez de respuesta y en ocasiones se recurre a este tipo de control cuando el error es grande, y se pasa de forma automática a otro tipo de regulación cuando el error se aproxima a cero.

En la figura 8 se puede ver un diagrama de bloques y una representación de su funcionamiento: Gracias a la existencia de una histéresis ( $h$ ), el número de conmutaciones se reduce notablemente. Sin histéresis, el accionador se activaría y desactivaría con demasiada frecuencia (gráfica con línea fina). La histéresis es como una oposición a experimentar cualquier cambio y generalmente será un efecto perjudicial.

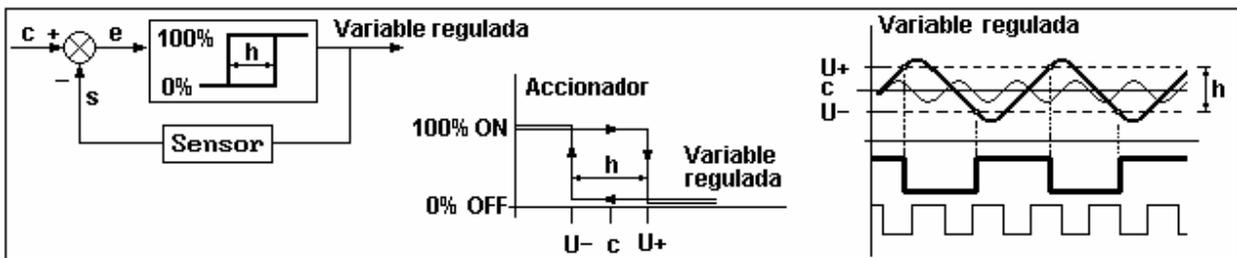


Fig. 8 Diagrama en bloque del regulador on-off y una representación de su funcionamiento.

### 1.4.3.2 Regulador PID.

El algoritmo de control más ampliamente extendido es el PID, pero existen muchos otros métodos que pueden dar un control de mayor calidad en ciertas situaciones donde el PID no responde a la perfección. El PID da buenos resultados en la inmensa mayoría de casos y tal vez es por esta razón que goza de tanta popularidad frente a otros reguladores teóricamente mejores.

Un PID (Proporcional Integral Derivativo) es un mecanismo de control por realimentación que se utiliza en sistemas de control industriales. Un controlador PID corrige el error entre un valor medido y el valor que se quiere obtener calculándolo y luego sacando una acción correctora que puede ajustar al proceso acorde. El algoritmo de cálculo del control PID se da en tres parámetros distintos: el proporcional, el integral, y el derivativo. El valor Proporcional determina la reacción del error actual. El Integral genera una corrección proporcional a la integral del error, esto nos asegura que aplicando un esfuerzo de control suficiente, el error de seguimiento se reduce a cero. El Derivativo determina la reacción del tiempo en el que el error se produce. La suma de estas tres acciones es usada para ajustar al proceso vía un elemento de control como la posición de una válvula de control o la energía suministrada a un calentador. Nótese que el uso del PID para control no garantiza control óptimo del sistema o la estabilidad del mismo.

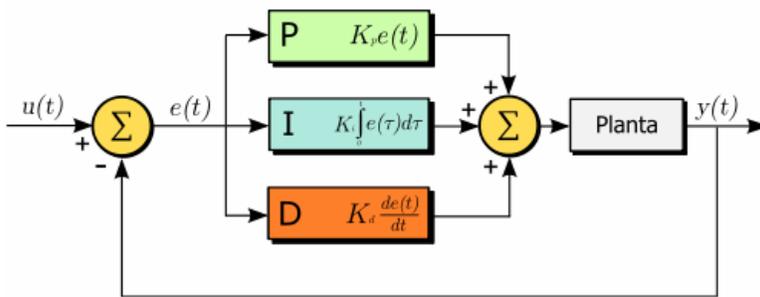


Fig. 9 Diagrama en bloque de un regulador PID.

Las tres componentes de un controlador PID son: parte **P**roporcional, acción **I**ntegral y acción **D**erivativa. El peso de la influencia que cada una de estas partes tiene en la suma final, viene dado por la constante proporcional, el tiempo integral y el tiempo derivativo, respectivamente. Se pretenderá lograr que el bucle de control corrija eficazmente y en el mínimo tiempo posible los efectos de las perturbaciones.

## Proporcional

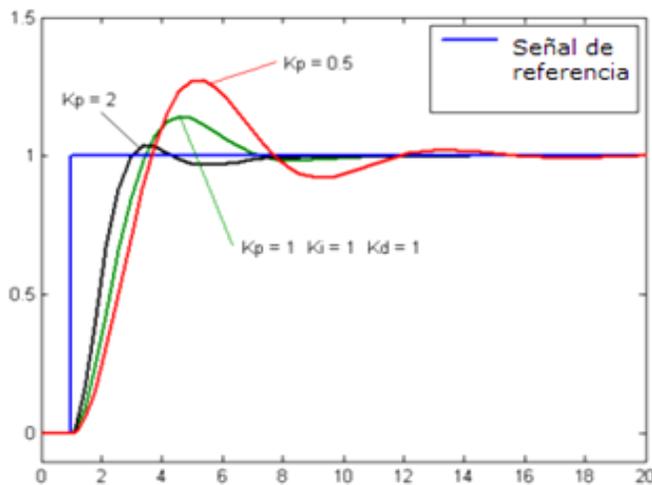


Fig. 10 Estabilidad proporcional.

La parte proporcional consiste en el producto entre la señal de error y la constante proporcional como para que hagan que el error en estado estacionario sea casi nulo, pero en la mayoría de los casos, estos valores solo serán óptimos en una determinada porción del rango total de control, siendo distintos los valores óptimos para cada porción del rango.. La parte proporcional no considera el tiempo, por lo tanto, la mejor manera de solucionar el error permanente y hacer que el sistema contenga alguna componente que tenga en cuenta la variación respecto al tiempo, es incluyendo y configurando las acciones integral y derivativa.

La fórmula del proporcional esta dada por:  $P_{sal} = K_p e(t)$

## Integral

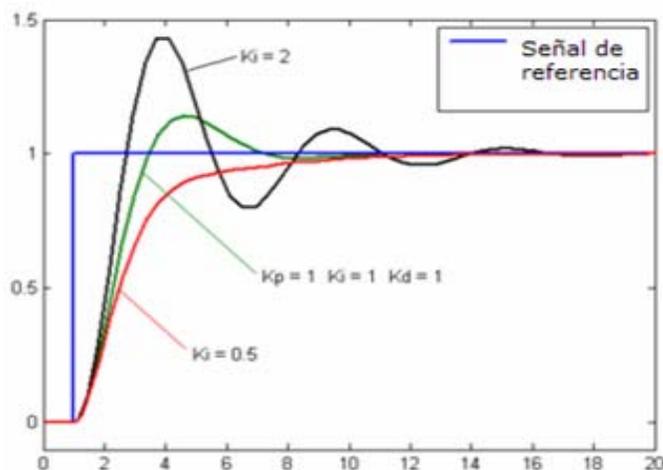


Fig. 11 Corrección de la velocidad con que se repite la acción proporcional.

El modo de control Integral tiene como propósito disminuir y eliminar el error en estado estacionario, provocado por el modo proporcional. El control integral actúa cuando hay una desviación entre la variable y el punto de consigna, integrando esta desviación en el tiempo y sumándola a la acción proporcional. El *error* es integrado, lo cual tiene la función de promediario o sumarlo por un periodo de tiempo determinado; Luego es multiplicado por una constante **I**. **I** representa la constante de integración. Posteriormente, la respuesta integral es adicionada al modo Proporcional para formar el control P + I con el propósito de obtener una respuesta estable del sistema sin error estacionario.

El modo integral presenta un desfaseamiento en la respuesta de 90° que sumados a los 180° de la retroalimentación (negativa) acercan al proceso a tener un retraso de 270°, luego entonces solo será necesario que el tiempo muerto contribuya con 90° de retardo para provocar la oscilación del proceso. El control integral se utiliza para obviar el inconveniente del offset (desviación permanente de la variable con respecto al punto de consigna) de la banda proporcional.

La formula del integral esta dada por:  $I_{sal} = K_i \int_0^t e(\tau) d\tau$

## Derivativo

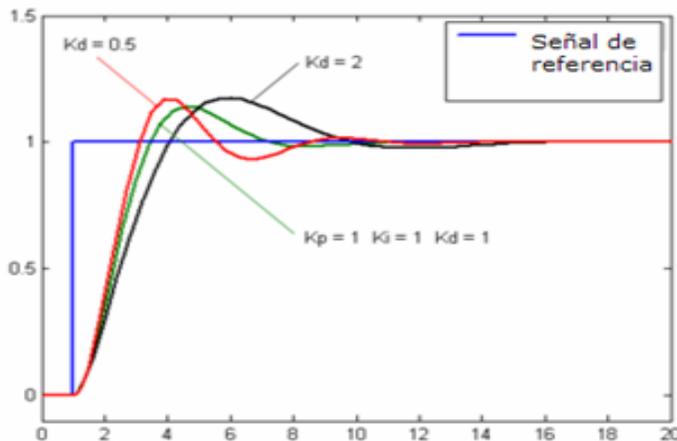


Fig. 12 Corrección del error.

La acción derivativa se manifiesta cuando hay un cambio en el valor absoluto del error; (si el error es constante, solamente actúan los modos proporcional e integral).

El *error* es la desviación existente entre el punto de medida y el valor consigna, o "*Set Point*".

La función de la acción derivativa es mantener el error al mínimo corrigiéndolo proporcionalmente con la misma velocidad que se produce; de esta manera evita que el error se incremente.

Se deriva con respecto al tiempo y se multiplica por una constante **D** y luego se suma a las señales anteriores (P+I). Es importante adaptar la respuesta de control a los

cambios en el sistema ya que una mayor derivativa corresponde a un cambio más rápido y el controlador puede responder acordeamente.

La fórmula del derivativo esta dada por:  $D_{sal} = K_d \frac{de}{dt}$

Cuando el tiempo de acción derivada es grande, hay inestabilidad en el proceso. Cuando el tiempo de acción derivada es pequeño la variable oscila demasiado con relación al punto de consigna. Suele ser poco utilizada debido a la sensibilidad al ruido que manifiesta y a las complicaciones que ello conlleva.

El tiempo óptimo de acción derivativa es el que retorna la variable al punto de consigna con las mínimas oscilaciones.

### **1.4.3.3 Control multivariable.**

Se aplica cuando existen dos o más variables que están acopladas, o lo que es lo mismo, cuando la variación de una ejerce una variación en otra. Los controladores multivariable son más comunes en las industrias aeronáutica, energética y petroquímica.

Los sistemas de control multivariables son agrupados en varios tipos básicos:

- Según el número de entradas y salidas del sistema, se denominan:
  - De una entrada y una salida o **SISO** (single input, single output).
  - De una entrada y múltiples salidas o **SIMO** (single input, múltiple output).
  - De múltiples entradas y una salida o **MISO** (múltiple input, single output).
  - De múltiples entradas y múltiples salidas o **MIMO** (múltiple input, múltiple output).
  
- Según la ecuación que define el sistema, se denomina:
  - Lineal, si la ecuación diferencial que lo define es lineal.
  - No lineal, si la ecuación diferencial que lo define es no lineal.
  
- Según sea la respuesta del sistema (valor de la salida) respecto a la variación de la entrada del sistema:
  - El sistema se considera estable cuando ante una variación muy rápida de la entrada se produce una respuesta acotada de la salida.
  - El sistema se considera inestable cuando ante una entrada igual a la anteriormente se produce una respuesta no acotada de la salida.

#### 1.4.3.4 Control difuso.

Otro tipo de control que está ganando adeptos es el control por lógica difusa (fuzzy control). Se trata de un control que se basa en la experiencia adquirida y actuar como lo haría una persona, es decir, con reglas empíricas. Este tipo de control no lineal está dando muy buenos resultados en procesos no lineales y de difícil modelización. Si el problema de control es la falta de experiencia debido a que el proceso no se conoce a la perfección o la evolución de la variable a regular es complicada, se puede programar un control con lógica difusa con la ayuda de una red neuronal, es decir, la red neuronal aprendería del sistema lo suficiente para informar al control con lógica difusa de cuáles son las reglas a usar en cada momento para obtener un buen control.

Un controlador difuso consta de las siguientes partes:

1. la normalización de la entrada.
2. la fusificación.
3. el mecanismo de inferencia.
  - Base de reglas
4. la defusificación.
5. la desnormalización.

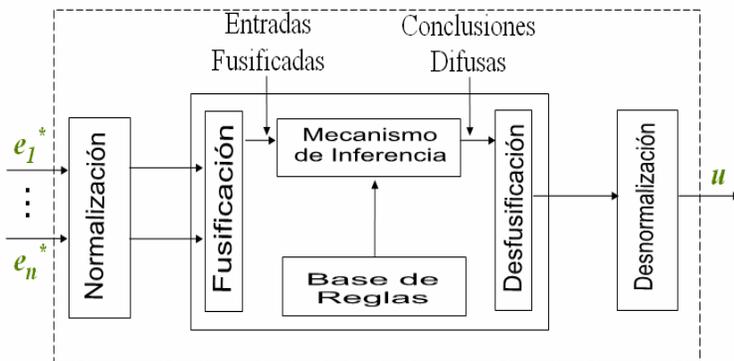


Fig. 13 Diagrama en bloque de un controlador Fuzzy

## 1.5 CONCLUSIONES PARCIALES.

- Con el presente trabajo se pretende desarrollar e implantar herramientas y medios dirigidos a desarrollar de manera práctica el aprendizaje de los estudiantes sobre el conocimiento de los accionamientos eléctricos y uso de los diferentes tipos de variadores de velocidad para motores eléctricos en condiciones de laboratorio del ISMM.
- Debe siempre tenerse presente que la utilización de variadores de velocidad está encaminado a controlar este parámetro en un motor o en un proceso; por tanto, su selección nunca podrá ser aislada.
- Finalmente digamos que muchas veces el criterio de selección entre el uso de los distintos sistemas de regulación de velocidad pasa fundamentalmente por una consideración de tipo económico/beneficios. De ahí que su selección y utilización está dada de forma que pueda optimizarse el uso de los equipos y el consumo de energía.

## CAPÍTULO II

### Capítulo II: Fundamentos sobre adquisición de datos.

#### 2.1 GENERALIDADES.

El acceso a la información es la clave a la hora de garantizar la seguridad y productividad en los procesos de producción. El software de Citect garantiza este acceso con **CitectScada** para las operaciones de monitorización y control.

**CitectScada** es uno de los principales paquetes de software de automatización del mundo. Permite a los usuarios incrementar la tasa de retorno de sus inversiones (ROI) gracias a un sistema de monitorización y control altamente escalable y extraordinariamente fiable, que permite reducir los costes de operación y mejorar la productividad y la calidad del producto. Las herramientas de configuración son fáciles de usar y las potentes prestaciones que incorpora CitectScada permiten a los usuarios e integradores de sistemas desarrollar y poner en servicio rápidamente sistemas de cualquier tamaño, para cualquier tipo de industria. El software dispone de más de 100 drivers de conectividad y de un sistema de desarrollo gratuito. Los gráficos dinámicos de las pantallas de operador de CitectScada aseguran que los datos relativos al proceso están disponibles para los operadores en todo momento, de forma clara e inequívoca. Incluso en el idioma preferido, la característica multi-lenguaje de Citect permite, con una sola pulsación, cambiar el idioma de presentación de toda la información.

## 2.2 SISTEMAS SCADA.

El término SCADA (supervisión, control y adquisición de datos) usualmente se refiere a un sistema central que monitoriza y controla un sitio completo o una parte de un sitio que nos interesa controlar (el control puede ser sobre máquinas en general, depósitos, bombas, etc.) o finalmente un sistema que se extiende sobre una gran distancia (kilómetros / millas).

La mayor parte del control del sitio es en realidad realizada automáticamente por una UTR (Unidad Terminal Remota), por un PLC (Controlador Lógico Programable) y más actualmente por un PAC (Controlador Automático Programable). Las funciones de control del servidor están casi siempre restringidas a reajustes básicos del sitio o capacidades de nivel de supervisión. Por ejemplo un PLC puede controlar el flujo de agua a través de un variador de frecuencia, pero un sistema SCADA puede permitirle a un operador cambiar el punto de consigna (set point) de control para el flujo, y permitirá grabar y mostrar cualquier condición de alarma como la pérdida de un flujo o una alta temperatura. La realimentación del lazo de control es cerrada a través del UTR o el PLC; el sistema SCADA monitorea el desempeño general de dicho lazo y también puede mostrar gráficas con históricos, tablas con alarmas y eventos, permisos y accesos de los usuarios.

Una de las soluciones en el control SCADA es utilizando un programa para monitorizar, controlar y automatizar señales analógicas y digitales capturadas a través de tarjetas de adquisición de datos. Uno de los programas más utilizados para este fin es el CitectSCADA.

Control de supervisión = medir para corregir.

Registro de datos = obtención y tratamiento de datos.

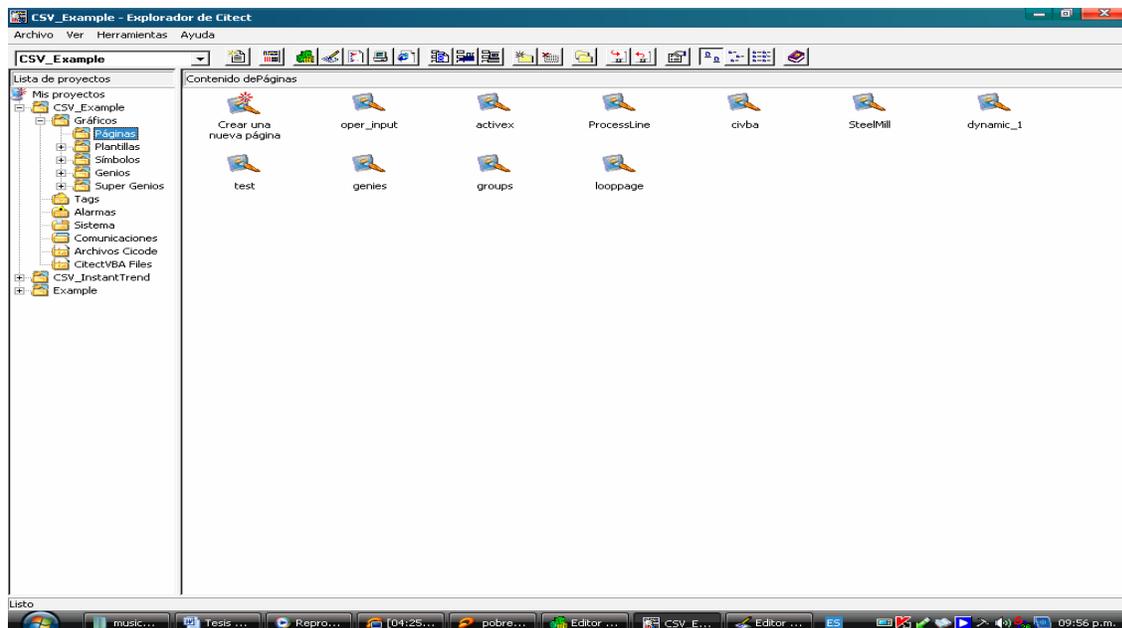


Fig. 14 Ventana Explorador CitectSCADA.

Es una aplicación software especialmente diseñada para funcionar sobre ordenadores en el control de producción, proporcionando comunicación con los dispositivos de campo (controladores autónomos...) y controlando el proceso de forma automática desde la pantalla del ordenador. También provee de toda la información que se genera en el proceso productivo a diversos usuarios, tanto del mismo nivel como de otros supervisores dentro de la empresa (supervisión, control calidad, control de producción, almacenamiento de datos, etc.).

En esta ocasión hablamos de un sistema cuya función principal es la de capturar información y proporcionar una interfase para control de equipos específicos como controladores autómatas programables (PLCs), Unidades de terminales remotas (RTUs),etc.

CitectSCADA es una solución HMI/SCADA (interfase máquina humano/SCADA) que permite implementar sistemas de control fiables, herramientas de configuración fáciles de usar y características que permiten desplegar soluciones para automatización de plantas de diversos tamaños. Las características del software son:

- Visualización gráfica del proceso.
- Gestión de alarmas.
- Opciones de clustering.
- Tendencias históricas y en tiempo real.
- Informes integrados.
- Control del proceso estadístico. Esta información del proceso es útil para controlar por ejemplo la calidad del producto, usando paquetes estadísticos para supervisión y análisis de datos.
- Lenguajes de programación multi-threaded CitectVBA y Cicode.
- Herramientas de análisis poderosas. El sistema puede integrarse fácilmente con sistemas comerciales de gestión empresarial.

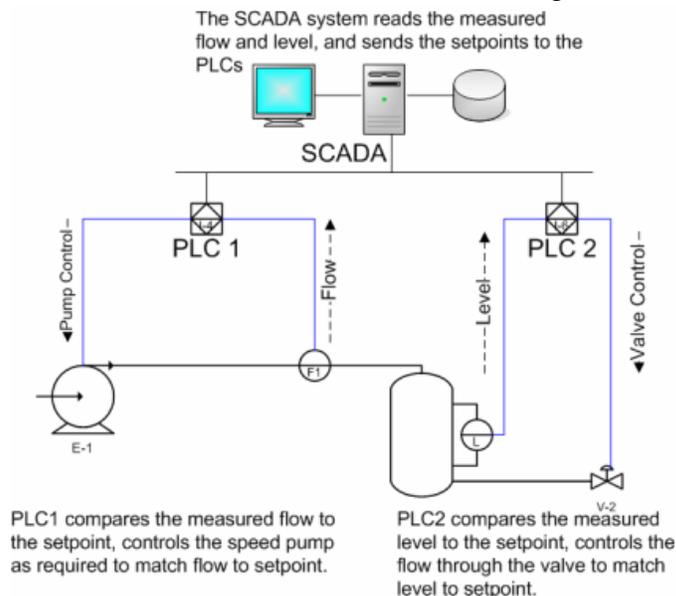


Fig. 15 Ejemplo de la aplicación del sistema SCADA como lazo cerrado en áreas industriales.

### **Estas áreas pueden ser:**

- Monitorizar procesos físicos o de transporte en sistemas industriales, para controlar la distribución de energía eléctrica.
- Gestión de la producción (facilita la programación de la fabricación).
- Mantenimiento (proporciona magnitudes de interés tales para evaluar y determinar modos de fallo, índices de Fiabilidad, entre otros).
- Control de Calidad (proporciona de manera automatizada los datos necesarios para calcular índices de estabilidad de la producción y tolerancias).
- Administración (actualmente pueden enlazarse estos datos del SCADA con un servidor ERP (Enterprise Resource Planning o sistema de planificación de recursos empresariales), e integrarse como un módulo más).
- Tratamiento histórico de información (mediante su incorporación en bases de datos).

#### **2.2.1 Prestaciones.**

Un paquete SCADA debe estar en disposición de ofrecer las siguientes prestaciones:

- Posibilidad de crear paneles de alarma, que exigen la presencia del operador para reconocer una parada o situación de alarma, con registro de incidencias.
- Generación de históricos de señal de planta, que pueden ser volcados para su proceso sobre una hoja de cálculo.
- Ejecución de programas, que modifican la ley de control, o incluso anular o modificar las tareas asociadas al autómatas, bajo ciertas condiciones.
- Bullet, Posibilidad de programación numérica, que permite realizar cálculos aritméticos de elevada resolución sobre la CPU del ordenador.

Algunos SCADA ofrecen librerías de funciones para lenguajes de uso general que permiten personalizar de manera muy amplia la aplicación que desee realizarse con dicho SCADA.

#### **2.2.2 Requisitos.**

Un SCADA debe cumplir varios objetivos para que su instalación sea perfectamente aprovechada:

- Deben ser sistemas de arquitectura abierta, capaces de crecer o adaptarse según las necesidades cambiantes de la empresa.
- Deben comunicarse con total facilidad y de forma transparente al usuario con el equipo de planta y con el resto de la empresa (redes locales y de gestión).
- Deben ser programas sencillos de instalar, sin excesivas exigencias de hardware, y fáciles de utilizar, con interfaces amigables con el usuario.

### 2.2.3 Componentes del sistema.

Los tres componentes de un sistema SCADA son:

1. Múltiples Unidades de Terminal Remota (también conocida como UTR, RTU o Estaciones Externas).
2. Estación Maestra y Computador con HMI.
3. Infraestructura de Comunicación.

#### **Unidad de Terminal Remota (UTR)**

La UTR se conecta al equipo físicamente y lee los datos de estado como los estados abierto/cerrado desde una válvula o un interruptor, lee las medidas como presión, flujo, temperatura, voltaje o corriente. Por el equipo el UTR puede enviar señales que pueden controlarlo: abrirlo, cerrarlo, intercambiar la válvula o configurar la velocidad de la bomba, ponerla en marcha, pararla.

La UTR puede leer el estado de los datos digitales o medidas de datos analógicos y envía comandos digitales de salida o puntos de ajuste analógicos.

Una de las partes más importantes de la implementación de SCADA son las alarmas. Una alarma es un punto de estado digital que tiene cada valor NORMAL o ALARMA. La alarma se puede crear en cada paso que los requerimientos lo necesiten.

#### **Estación Maestra.**

El termino "Estación Maestra" se refiere a los servidores y el software responsable para comunicarse con el equipo del campo (UTRs, PLCs, etc) en estos se encuentra el software HMI corriendo para las estaciones de trabajo en el cuarto de control, o en cualquier otro lado. En un sistema SCADA pequeño, la estación maestra puede estar en un solo computador, A gran escala, en los sistemas SCADA la estación maestra puede incluir muchos servidores, aplicaciones de software distribuido.

El sistema SCADA usualmente presenta la información al personal operativo de manera gráfica, en forma de un diagrama de representación. Esto significa que el operador puede ver un esquema que representa la planta que está siendo controlada. Por ejemplo un dibujo de una bomba conectada a la tubería puede mostrar al operador cuanto fluido esta siendo bombeado desde la bomba a través de la tubería en un momento dado o bien el nivel de líquido de un tanque o si la válvula esta abierta o cerrada. Los diagramas de representación pueden consistir en gráficos de líneas y símbolos esquemáticos para representar los elementos del proceso, o pueden consistir en fotografías digitales de los equipos sobre los cuales se animan las secuencias.

## 2.3 FUNDAMENTOS SOBRE ADQUISICIÓN DE DATOS DEL PROGRAMA.

La **Adquisición de Datos del programa**, consiste en la toma de muestras del mundo real (sistema analógico) para generar datos que puedan ser manipulados por un ordenador (sistema digital). Consiste, en tomar un conjunto de variables físicas, convertirlas en tensiones eléctricas y digitalizarlas de manera que se puedan procesar en una computadora. Se requiere una etapa de acondicionamiento, que adecua la señal a niveles compatibles con el elemento que hace la transformación a señal digital. El elemento que hace dicha transformación es el módulo o tarjeta de Adquisición de Datos (**DAQ**).

### 2.3.1 Proceso de adquisición de datos.

#### **Definiciones:**

**Dato:** Representación simbólica (numérica, alfabética...), atributo o característica de un valor. No tiene sentido en sí mismo, pero convenientemente tratado (procesado) se puede utilizar en la relación de cálculos o toma de decisiones.

**Adquisición:** Recogida de un conjunto de variables físicas, conversión en voltaje y digitalización de manera que se puedan procesar en un ordenador.

**Sistema:** Conjunto organizado de dispositivos que interactúan entre sí ofreciendo prestaciones más completas y de más alto nivel. Una vez que las señales eléctricas se transformaron en digitales, se envían a través del bus de datos a la memoria del PC. Una vez los datos están en memoria pueden procesarse con una aplicación adecuada, archivarlas en el disco duro, visualizarlas en la pantalla, etc...

**Bit de resolución:** Número de bits que el convertidor analógico a digital (ADC) utiliza para representar una señal.

**Rango:** Valores máximo y mínimo entre los que el sensor, instrumento o dispositivo funcionan bajo unas especificaciones.

Los componentes de los sistemas de adquisición de datos, poseen sensores adecuados que convierten cualquier parámetro de medición de una señal eléctrica, que se adquiere por el hardware de adquisición de datos. Los datos adquiridos se visualizan utilizando el software suministrado. Los controles y visualizaciones se pueden desarrollar utilizando varios lenguajes de programación de propósito general. Los lenguajes especializados de programación utilizados para la adquisición de datos incluyen EPICS, utilizada en la construcción de grandes sistemas de adquisición de datos, LabVIEW, que ofrece un entorno gráfico de programación optimizado para la adquisición de datos, y MATLAB. Estos entornos de adquisición proporcionan un lenguaje de programación además de bibliotecas y herramientas para la adquisición de datos y posterior análisis.

De la misma manera que se toma una señal eléctrica y se transforma en una digital para enviarla al ordenador, se puede también tomar una señal digital o binaria y convertirla en una eléctrica. En este caso el elemento que hace la transformación es una tarjeta o módulo de Adquisición de Datos de salida, o tarjeta de control. La señal dentro de la memoria del PC la genera un programa adecuado a las aplicaciones que quiere el usuario y, luego de procesarla, es recibida por mecanismos que ejecutan movimientos mecánicos, a través de servomecanismos, que también son del tipo transductores.

## 2.4 FUNDAMENTOS SOBRE ADQUISICIÓN DE DATOS DEL PLC.

El hardware que se utiliza para la adquisición de datos es el PLC S7-300 de Siemens con sus módulos de tarjetas analógicas y digitales.



Fig. 16 CPU 315-2DP.

Esta CPU es por tanto suficiente como elemento de formación, disponiendo de las siguientes características:

- 16K de instrucciones. 48K de memoria de trabajo 80K de espacio disponible.
- 1024 Bytes de Entradas/Salidas Digitales.
- 128 Bytes de Entradas/Salidas Analógicas.
- 0,3 ms / 1K instrucciones.
- 64 contadores.
- 128 temporizadores.
- 2048 bits de Marcas.

**DAQ hardware** son por lo general las interfaces entre la señal y un PC. Podría ser en forma de módulos que pueden ser conectados a la computadora de los puertos (paralelo, serie, USB, ethernet, etc) o ranuras de las tarjetas conectadas a (PCI, ISA) en la placa madre. Por lo general, el espacio en la parte posterior de una tarjeta PCI es demasiado pequeño para todas las conexiones necesarias, de modo que una ruptura de caja externa es obligatorio. El cable entre este recuadro y el PC es caro debido a los numerosos cables y el blindaje necesario. Las tarjetas DAQ a menudo contienen múltiples componentes (multiplexores, ADC, DAC, TTL-IO, temporizadores de alta velocidad, memoria RAM). Estos son accesibles a través de un bus por un micro controlador, que puede ejecutar pequeños programas.

Las tarjetas a utilizar en nuestro trabajo para la adquisición y salidas de datos del PLC son las siguientes.

## Módulo de entradas digitales SM 321; DI 16 x AC 120/230 V (6ES7321-1FH00-0AA0)

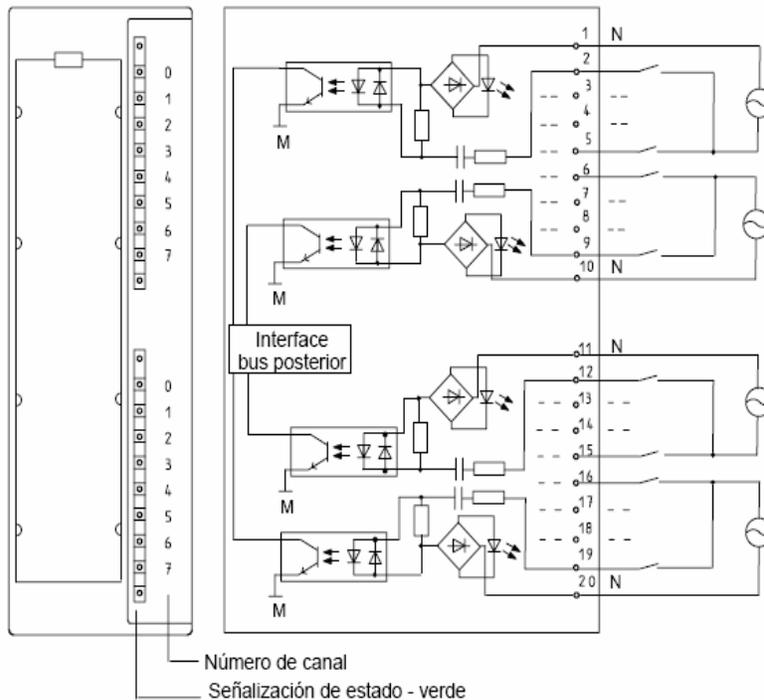


Fig. 17 Esquema de conexiones y principio del módulo SM 321; DI 16XAC 120/230 V.

### Estados, alarmas y diagnóstico.

- Señalización de estado.....un led verde por canal.
- Alarmas.....ninguna.
- Funciones de diagnóstico.....ninguna.

### Datos para selección de un sensor.

#### Tensión de entrada.

- Valor nominal.....120/230 V a.c.
- Para señal "1".....79 a 264 V
- Para señal "0".....0 a 40 V
- Margen de frecuencia 47 a 63Hz

#### Corriente de entrada.

- Con señal "1"
  - 120 V, 60 Hz.....typ. 8,0mA
  - 230 V, 50 Hz.....typ. 16,0mA

### Retardo de entrada.

- De "0" a "1" .....máx. 25ms
- De "1" a "0" .....máx. 25ms

Característica de entrada según CEI61131, tipo 1

Conexión de BERO a 2 hilos.....posible

- Intensidad de reposo admisible.....máx. 2mA

### Módulo de salidas digitales SM 322; DO 16 x AC 120/230 V/1 A; (6ES7322-1FH00-0AA0)

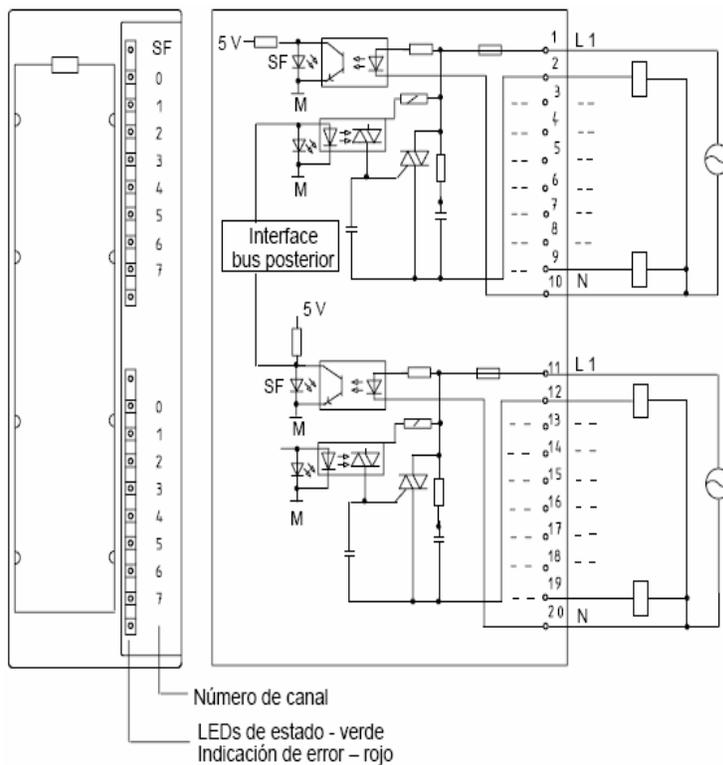


Fig. 18 Esquema de conexiones y principio del modulo SM 322; DO 16XAC 120/230 V 1A.

### Estados, alarmas y diagnostico.

Señalización de estado.....un led verde por canal.

#### Alarmas.

- Alarma de diagnóstico.....no.

#### Funciones de diagnostico.

- Indicador fallo colectivo.....Led rojo (SF)  
(fusible o sin L1/N)

## Datos para la selección de un actuador.

### Tensión de salida.

- Con señal "1" .....mín. L+(-0,8 V)

### Corriente de salida.

- Con señal "1"
  - Valor nominal.....1A
  - Margen admisible para 0°C a 40°C.....10 mA a 1A
  - Margen admisible para 0°C a 60°C.....10 mA a 0,5A
  - Corriente de choque admisible (por grupo).....máx. 20A (con 2 semiondas)
- Con señal "0" (intensidad residual).....máx. 2mA

Tensión de bloqueo.....máx. 60V

Paso por cero

Tamaño del arrancador del motor.....tamaño máx. 4 según NEMA.

Carga de lámparas.....máx. 50W

### Conexión en paralelo de 2 salidas.

- Para mando redundante de una carga..... (solo salidas del mismo grupo)
- Para elevación de potencia.....no.

Activación de una entrada digital.....posible.

### Frecuencia de conmutación.

- Para carga óhmica.....máx. 10Hz
- Para carga inductiva, según CEI947-5-1,AC 15
- Para carga de lámparas.....máx. 1Hz

## Módulo de entradas analógicas SM 331; AI 8X16 Bit.

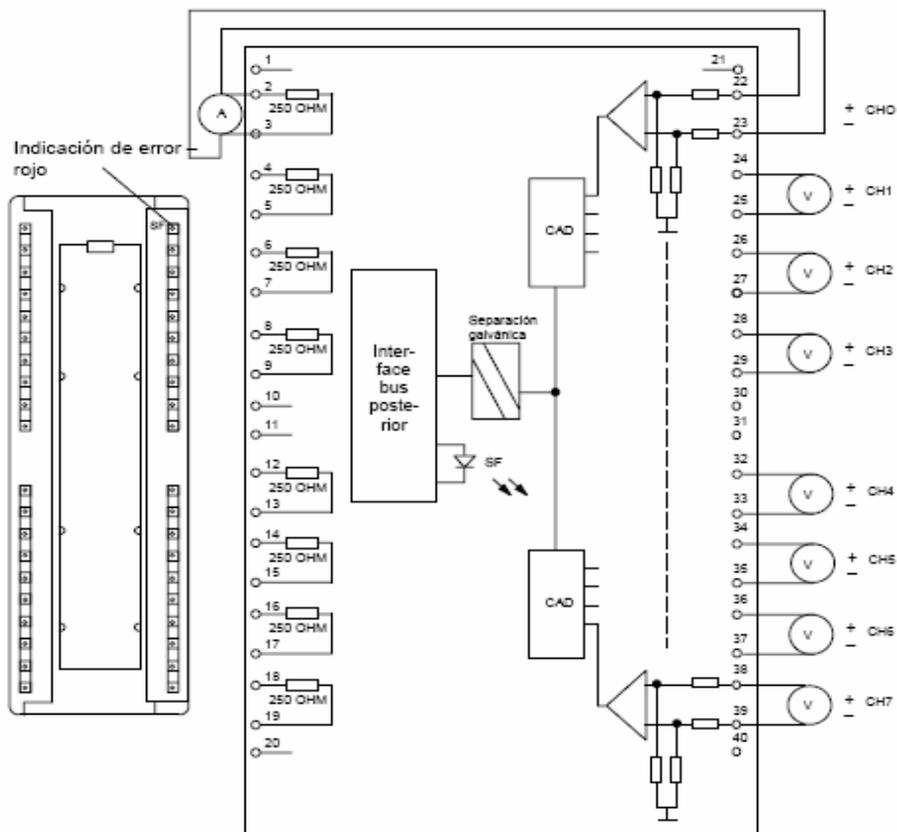


Fig. 19 Esquema de conexiones y principio del módulo de entradas analógicas SM 331; AI 8X16 Bit.

### Datos específicos del módulo.

Soporta operación sincronizada.....no.  
 Cantidad de entradas.....8  
 Longitud de cable.  
 ➤ Con pantalla.....máx. 200m

### Tensiones, intensidades y potenciales.

#### Separación galvánica.

➤ Entre canales y bus posterior.....sí.

#### Diferencia de potencial admisible.

➤ Entre las entradas ( $U_{cm}$ ).....50V c.c, 35Vc.c.

Disipación del módulo.....típ. 0,6 W

**Módulo de salidas analógicas SM 332; AO 4 x 12 Bit  
(6ES7332-5HD01-0AB0)**

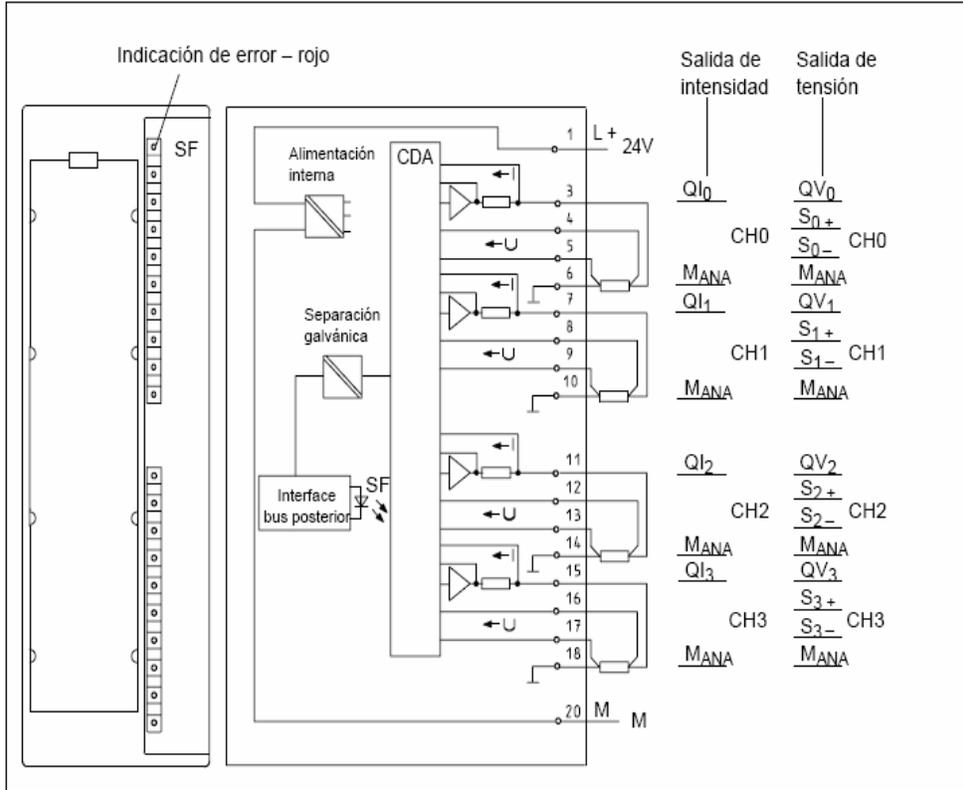


Fig. 20 Esquema de conexiones y principio del módulo salidas analógicas SM 332; AO 4X12 Bit.

**Datos específicos del módulo.**

Soporta operación sincronizada.....no.  
 Cantidad de salidas.....4

**Longitud de cable**

- Comportamiento de las salidas no parametrizadas.....Entregan el último valor de salida vigente antes de la parametrización.

Cantidad de salidas.....4

**Longitud del cable.**

- Con pantalla.....máx. 200m

**Tensiones, intensidades y potenciales.**

**Tensión de carga nominal L+ 24 Vc.c.**

- Protección contra inversión de polaridad.....si.

## Separación galvánica.

- Entre canales y bus posterior.....si.
- Entre canales y tensión de alimentación.....si.
- Entre los canales.....no.

Disipación del módulo.....típ. 3W

## Tiempo de conversión (por canal)

- En operación estándar.....máx. 0.8ms

## Estados, alarmas y diagnóstico.

- Alarma de diagnóstico.....parametrizable

## Funciones de diagnóstico

- Indicador fallo colectivo.....Led rojo (SF)

### [2.4.1 Programa para la realización de la programación en el PLC.](#)

Existen en la actualidad autómatas de diferentes tipos de fabricantes y de diversos software de comunicación. Uno de los fabricantes más reconocidos en este campo de los autómatas programables es SIEMENS con su respectivo software SIMATIC, en nuestro caso SIMATIC S7 para comunicarse con los PLC de dicha firma.

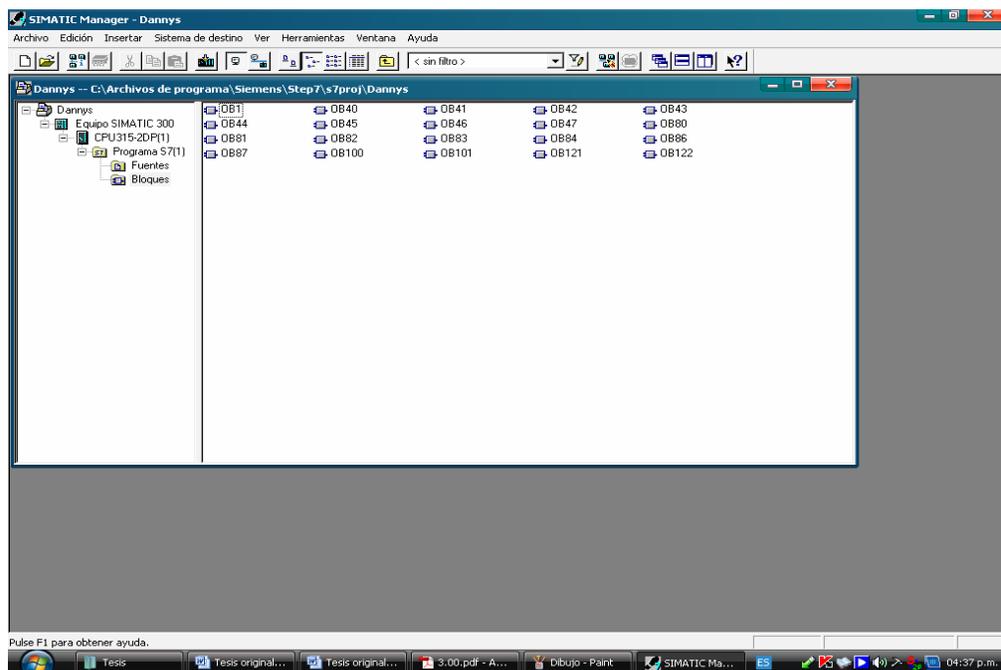


Fig. 21 Ventana SIMATIC S7 con los bloques de programación declarados.

## Lenguajes de programación en STEP 7.

Lenguaje	STEP7
Diagrama de funciones	FBD/FUP Diagrama de bloque de funciones
Diagrama de contactos	LAD/KOP Lógica de escalera
Lista de instrucciones	STL/AWL Lista de estado
Texto estructurado	SCL Lenguaje de control de estructura
Gráfico secuencial	GRAPH7
Diagrama de bloques	CFC Continuous Function Chart
Diagrama de gráfico de estados	HIGRAPH7

Tabla 1 Tipos de lenguajes soportados.

### 2.4.1.1 Funcionalidad del software:

El entorno de programación permite desarrollar proyectos de control y automatización con las siguientes funcionalidades.

- Crear la estructura de directorios necesaria para la gestión de cada proyecto.
- Configurar el hardware del equipo.
- Desarrollar programas off-line.
- Cargar, descargar programas y datos en el PLC/API (on-line).
- Depurar el programa en fase de ejecución sin necesidad de interrumpir la ejecución del programa (on-line).
- Realizar funciones de test o estado de variables y zonas de la memoria.
- Realizar funciones de forzado de variables.
- Elaborar documentos para la fase de mantenimiento posterior.
- Analizar el estado y comportamiento de la CPU (pilas, tiempos) y de los módulos de señal asociados.

### **Bloque en el Programa de Usuario**

STEP 7 permite estructurar el programa. Esto aporta las siguientes ventajas:

- los programas se pueden dividir en bloques.
- Se pueden estandarizar secciones individuales del programa.
- Se simplifica la organización del programa.
- las modificaciones del programa pueden realizarse más fácilmente.
- Se simplifica el test del programa, ya que puede ejecutarse por partes.
- Se simplifica la puesta en servicio.

Bloques de organización para la realización de la programación en el PLC.

TIPO DE ERROR	EJEMPLO	OB DE ERROR
Error por Tiempo	Exceso del tiempo ciclo máximo	OB 80
Error en fuente de Alimentación	Fallo de la batería Tampón	OB 81
Interrupción de Diagnostico	Rotura de cable en la entrada de un módulo de diagnosis	OB 82
Interrupción al Insertar/Eliminar un Módulo	Inserción/Eliminación de un módulo	OB 83
Fallo Hardware de la CPU	Error en la interfase a la red MPI, al bus interno de comunicaciones (C-Bus) o a la interfase de la periferia distribuida de I/O	OB 84
Error de Clase Prioritaria	Llamada a un módulo OB no cargado	OB 85
Error de bastidor (solo S7-400)	Fallo del módulo montado en el S7-400	OB 86
Error de Comunicaciones	Detección de un cable falso	OB 87

Tabla 2 Bloques declarados para las fallas de origen PLC.

## PARTE DEL OPERANDO

La parte del operando contiene todas las instrucciones necesarias para la ejecución de la operación. Las características de los operandos incluyen la clase del operando.

El parámetro del operando indica la dirección del operando.

- E para entradas.
- A para salidas.
- M para marcas.
- L para datos locales (Bloque interno de variables).
- T para temporizadores.
- Z para contadores.
- OB para bloques de organización.
- FB para bloques de funciones.
- FC para funciones.
- DB para bloques de datos.
- SFB para bloques de función del sistema.
- SFC para bloques de función.
- L#.. Para constantes de 32-Bit etc.

## **2.5 CONCLUSIONES PARCIALES.**

Desde lo del desarrollo industrial la supervisión de la producción ha sido una de las ideas que más han se ha trabajado por los diferentes departamento de una empresa. La supervisión inicialmente era sinónimo del control de la producción, se inicio realizando formación técnica al personal que trabaja en las diferentes industrias. Posteriormente cuando nos acercamos a los tiempos modernos esta similitud entre supervisión y control se distancia, sobre todo con la llegada de la automatización.

En la industria ahora el control de la producción en primer plano lo realizan las máquinas enviando al hombre al concepto actual de supervisión. El hombre ya no controla de forma directa la máquina sino que supervisa que esta no entre en error, en fallo o en estado crítico en ese momento es cuando se pasa a lo que ya definimos como el control manual, en ese momento el hombre realiza realmente el control directo sobre la máquina.

## CAPÍTULO III

### Capítulo III: Técnicas para interfaces con los accionamientos eléctricos.

#### 3.1 GENERALIDADES.

Desde el inicio mismo del desarrollo del Control Automatizado, uno de los principios básicos de la automatización de sistemas y procesos radica en la comunicación que debe existir entre el hombre y una máquina, un equipo o una computadora, ésta interfase hombre – máquina forma una parte importante e integral de los procesos modernos industriales.

Para ello fueron desarrollándose diferentes técnicas para la interconexión de equipos y para el control de procesos mediante buses, de modo que la forma en que el usuario se comunica con una aplicación para solicitar los recursos del sistema constituye la interfaz del mismo.

La interfaz en los procesos de automatización y control de sistemas es particularmente importante para establecer una comunicación amigable y lo más cómoda posible entre el usuario de la computadora y la aplicación o proceso, permitiendo de manera simple la toma de información del sistema e interactuar con el mismo.

### 3.2 INTERFAZ DE USUARIO.

La industria de HMI nació esencialmente de la necesidad de estandarizar la manera de monitorear y de controlar múltiples sistemas remotos, PLCs y otros mecanismos de control. Aunque un PLC realiza automáticamente un control pre-programado sobre un proceso, normalmente se distribuyen a lo largo de toda la planta, haciendo difícil recoger los datos de manera manual, los sistemas SCADA lo hacen de manera automática. Históricamente los PLC no tienen una manera estándar de presentar la información al operador. La obtención de los datos por el sistema SCADA parte desde el PLC o desde otros controladores y se realiza por medio de algún tipo de red, posteriormente esta información es combinada y formateada. Un HMI puede tener también vínculos con una base de datos para proporcionar las tendencias, los datos de diagnóstico y manejo de la información

Las interfaces de usuario son aquellas que incluyen cosas como menús, ventanas, teclado, ratón, los "beeps" y algunos otros sonidos que la computadora hace, en general, todos aquellos canales por los cuales se permite la comunicación entre el hombre y la computadora. La mejor interacción hombre-máquina a través de una adecuada interfaz (Interfaz de Usuario), que le brinde tanto comodidad, como eficiencia.

Dentro de las Interfaces de Usuario se puede distinguir básicamente dos tipos:

- Una interfaz de hardware, a nivel de los dispositivos utilizados para ingresar, procesar y entregar los datos: teclado, ratón y pantalla visualizadora.
- Una interfaz de software, destinada a entregar información acerca de los procesos y herramientas de control, a través de lo que el usuario observa habitualmente en la pantalla.

Interfaces de hardware utilizadas en nuestro trabajo.

#### **Tarjeta CP 5613 A2**



Fig. 22 Tarjeta PCI (Universal Key 5V/3,3V) con microprocesador para la conexión de PCs y PGs/PCs SIMATIC a PROFIBUS, hasta 12 Mbits/s.

## Servicios de comunicación:

- Maestro PROFIBUS DP según IEC 61158/EN 50170
- Comunicación PG/OP con STEP 5 y STEP 7
- Comunicación S7.
- Comunicación compatible con S5 (SEND/RECEIVE) sobre la base de la interfaz FDL.
- PROFIBUS FMS según IEC 61158/EN 50170
- Amplias posibilidades de diagnóstico para instalación, puesta en marcha y funcionamiento del módulo.
- Alto rendimiento gracias a acceso directo por Dual Port-RAM.
- Mecanismo de filtro de eventos para descargar la CPU central.
- Modo multiprotocolo y funcionamiento paralelo de máx. 4 CPs.
- Posibilidad de uso para aplicaciones de control de movimiento gracias al soporte del modo de equidistancia.
- Los servidores OPC y las herramientas de configuración están incluidos en el alcance de suministro del software de comunicación respectivo.

## Adaptador PC



Fig. 23 Adaptador PC (RS 232) a MPI (Interface multi punto).

Para poder programar una estación SIMATIC S7-300 desde el PC o la PG, es necesaria una interface MPI. La comunicación de la interface tiene hasta 32 partes, las cuales se han de programar. Se utilizan para operar y vigilar con HMI y para el intercambio de datos entre SIMATIC S7 y CPUs.

### 3.2.1 Datos técnicos de comunicación:

<b>NORMALIZACIÓN</b>	<b>ESPECIFICACIÓN DE SIEMENS</b>
Equipos	32 como máximo
Método de acceso	Token (paso de testigo)
Velocidad de transferencia	19,2Kbit/s, 187,5Kbit/s ó 12Mbit/s
Soporte de transmisión	Cable bifilar apantallado, fibra óptica (vidrio o plástico)
Extensión de la red	Longitud de segmento 50m, vía
Topología	Eléctrica: Línea Óptica: Arbol, estrella, anillo
Servicios de comunicación	Funciones PG/OP Funciones S7 Funciones básicas S7 Comunicación por datos globales
Formato de datos	8 bit
Modo de transmisión	serie asíncrono
Alimentación	+24 V ± 25 % del autómatas o fuente externa.
Conector	SUB-D 9-pines
Protocolos	PC <-> S7

Tabla 3 Características técnicas de la comunicación del PLC

### 3.2.2 Funciones principales de la interfaz de usuario.

Sus principales funciones son las siguientes:

- Puesta en marcha y apagado.
- Control de las funciones manipulables del equipo.
- Manipulación de archivos y directorios.
- Herramientas de desarrollo de aplicaciones.
- Comunicación con otros sistemas.
- Información de estado.
- Configuración de la propia interfaz y entorno.
- Intercambio de datos entre aplicaciones.
- Control de acceso.
- Sistema de ayuda interactivo.

### 3.2.3 Tipos de interfaces de usuario.

#### **Según la forma de interactuar del usuario.**

Atendiendo a como el usuario puede interactuar con una interfaz, nos encontramos con varios tipos de interfaces de Usuario:

- Interfaces alfanuméricas (intérpretes de mandatos) que solo presentan texto.
- Interfaces gráficas de usuario (GUI, *Graphics User Interfaces*), las que permiten comunicarse con el ordenador de una forma muy rápida e intuitiva representando gráficamente los elementos de control y medida.
- Interfaces táctiles, que representan gráficamente un "panel de control" en una pantalla sensible que permite interactuar con el dedo de forma similar a si se accionara un control físico.

En nuestro trabajo implementaremos la Interfaces gráficas de usuario (GUI, *Graphics User Interfaces*) para el control y visualización del proceso a gran escala, la cual es posible con la Tarjeta CP 5613 A2 y el Adaptador PC.

#### **Según su construcción:**

Pueden ser de hardware o de software:

- Interfaces hardware.- Se trata de un conjunto de controles o dispositivos que permiten la interacción hombre-máquina, de modo que permiten introducir o leer datos del equipo, mediante pulsadores, reguladores e instrumentos.
- Interfaces software.- Son programas o parte de ellos, que permiten expresar nuestros deseos al ordenador o visualizar su respuesta.

#### **Valoración**

El principal objetivo de una interfaz de usuario es que éste se pueda comunicar a través de ella con algún tipo de dispositivo, conseguida esta comunicación, el segundo objetivo que se debería perseguir es el de que dicha comunicación se pueda desarrollar de la forma más fácil y cómoda posible para el usuario. El diseño de la interfaz es crítico para el manejo del equipo.

#### **Interfaz Humano – Máquina.**

Una interfaz Hombre - Máquina o HMI ("Human Machine Interface") es el aparato que presenta los datos a un operador (humano) y a través del cual éste controla el proceso.

Los sistemas HMI podemos pensarlos como una "ventana de un proceso. Esta ventana puede estar en dispositivos especiales como paneles de operador o en un ordenador. Los sistemas HMI en ordenadores se los conoce también como software HMI o de monitoreo y control de supervisión. Las señales del proceso son conducidas al HMI por medio de dispositivos como tarjetas de entrada/salida en el ordenador, PLC's (Controladores lógicos programables), PACs (controlador de automatización

programable), RTU (Unidades remotas de I/O) o DRIVER's (Variadores de velocidad de motores). Todos estos dispositivos deben tener una comunicación que entienda el HMI.

### 3.3 INTERFACES DE CITECT PARA COMUNICARSE CON EL AUTÓMATA PLC.

#### **Comunicaciones**

Los PLC intercambian datos con otros dispositivos. Típicamente un PLC puede tener integrado puertos de comunicaciones seriales que pueden cumplir con distintos estándares de acuerdo al fabricante. Estos puertos pueden ser de los siguientes tipos:

- RS-232
- RS-485
- RS-422
- Ethernet

Sobre estos tipos de puertos de hardware las comunicaciones se establecen utilizando algún tipo de protocolo o lenguaje de comunicaciones. En esencia un protocolo de comunicaciones define la manera como los datos son empaquetados para su transmisión y como son codificados. De estos protocolos los más conocidos son:

- Modbus
- Bus CAN
- Profibus
- Devicenet
- Controlnet
- Ethernet I/P

El puerto Utilizado en nuestro trabajo por la interfase Citect SCADA con el PLC es el MPI-DP/ RS232 la cual dispone de velocidad ajustable para realizar la programación.

#### **INTERFACE MULTIPUNTO (M.P.I.)**

Todas las CPUs (312, 313, 314, 315 y 315 -2DP) lo incorporan desde fábrica. Con éste puerto se puede comunicar fácilmente a distancias reducidas sin requerir módulos adicionales, por ejemplo hacia equipos de M+V (manejo + visualización), unidades de programación y otros autómatas S7-300 o S7- 400 para probar programas o consultar valores de estado.

Se pueden enviar datos a 4 distintos aparatos al mismo tiempo y utilizando siempre el mismo puerto a una velocidad de 187,5 Kbits / seg o 187,5 Kbaudios. Para pequeñas redes de comunicación o pequeños volúmenes de datos la CPU ofrece el servicio de Datos Globales, que permite intercambiar cíclicamente cantidades de datos en paquetes de hasta 22 bytes como máximo.

Distancia máxima entre dos estaciones o nudos de red de MPI adyacentes: 50 metros (sin repetidores); 1100 metros (con dos repetidores); 9100 metros (con más de 10 repetidores en serie); por encima de los 500 Klm. (cable de fibra óptica, con módulos de conexión ópticas)

Capacidad de expansión: los componentes comprobadores de campo son usados para configurar la comunicación de interface multipunto: cables LAN, conectores LAN y repetidores RS485, desde el PROFIBUS y la línea de productos de entradas/salidas distribuidas. Estos componentes permiten una óptima utilización de la configuración.

### 3.4 INTERFAZ PLC- VARIADOR DE VELOCIDAD.

Por lo general todos los autómatas de la industria incorporan interfaces de comunicación para su enlace a una red ya sea global o autónoma. Esto permite realizar diferentes conexiones en el control de dicha planta o proceso a monitorear.

Existen diferentes tipos de módulos en la gama del PLC S7-300. Unos de los más utilizados en la industria son los módulos de funciones FM que por su alta confiabilidad y modo de operación permiten realizar un mejor ajuste y control. Uno de estos son los FM 355 (*módulo de control de lazo cerrado*).

#### **Módulo de control de lazo cerrado FM 355; 6ES7355-0VH10-0AE0**

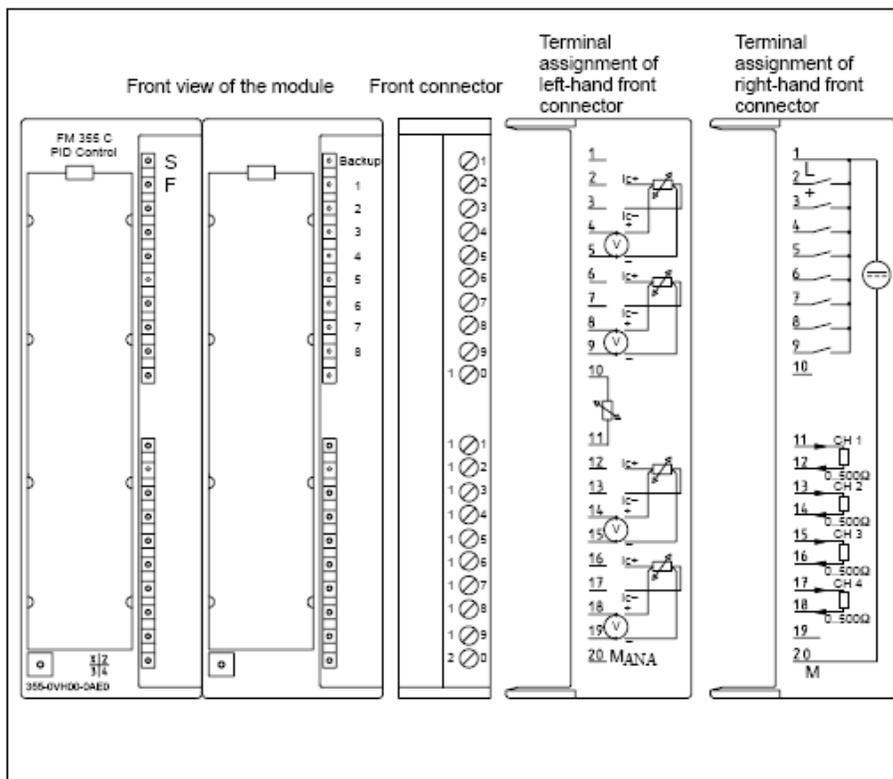


Fig. 24 Esquema de conexiones y principio del módulo de lazo cerrado FM 355; 6ES7355-0VH10-0AE0

Tarjeta de regulación FM 355 C, 4 canales, lazo continuo, 4 EA + 8 ED + 4 SA. Este módulo posee gran ventaja sobre los demás porque contiene un Modo de respaldo. Esta función garantiza que el módulo de regulación también siga funcionando si falla o se detiene la CPU. Para el modo de respaldo es posible ajustar una consigna de seguridad específica.

El FM 355 (4 canales) es un módulo de regulación universal que está disponible en dos versiones distintas:

- FM 355C Como regulador continuo para controlar actuadores analógicos.
- FM 355S como regulador paso a paso o a impulsos para actuadores controlados digitalmente.

### **Estructuras de regulación.**

Los módulos de regulación contienen varias estructuras de regulación en gran parte preprogramadas:

- Regulación de referencia fija.
- Regulación en cascada.
- Regulación de relación.
- Regulación de 3 componentes.

Se pueden interconectar hasta 4 reguladores a una estructura de regulación.

### **Modos de operación.**

Además del Automático y de respaldo, los módulos tienen los modos de operación siguientes:

- Modo manual.
- Modo Seguimiento.
- Modo de seguridad.

### 3.5 CONCLUSIONES PARCIALES.

➤ Interfase de usuario:

- El programa **CitectScada**: que es el que utilizaremos para las operaciones de monitorización y control de nuestro sistema garantiza una interfase que le permite al usuario interactuar con los diferentes elementos del proceso de forma simple, censar, monitorear y supervisar la información de entrada y salida del sistema.

## Capítulo IV: Programa supervisor en accionamientos.

### 4.1 GENERALIDADES.

Para la realización de cualquier proyecto de automatización tanto a pequeña como a gran escala es preciso conocer de manera previa el algoritmo lógico en que se implementará el sistema de control, así como el desarrollo de cada fase del proceso.

En el presente capítulo presentamos una reseña del diagrama secuencial del flujo tecnológico, así como la secuencia lógica de las acciones en la implementación del sistema de control, además hacemos referencia al criterio de selección – utilización y programación del software a utilizar (CitectScada) para la toma de muestras (adquisición de datos del mundo real), al mismo tiempo que describimos la arquitectura del modelo de control basado en la filosofía de escaneo del sistema del PLC.

## 4.2 FLUJO TECNOLÓGICO DEL PROCESO.

En nuestro flujo tecnológico implementaremos el control de llenado y vaciado de los tanques de almacenamiento de la planta en proceso de automatización, así como las respectivas variables que intervienen en dicho proceso. Estas variables se regularan mediante el control de lazo cerrado, control PID, control multivariable y control difuso.

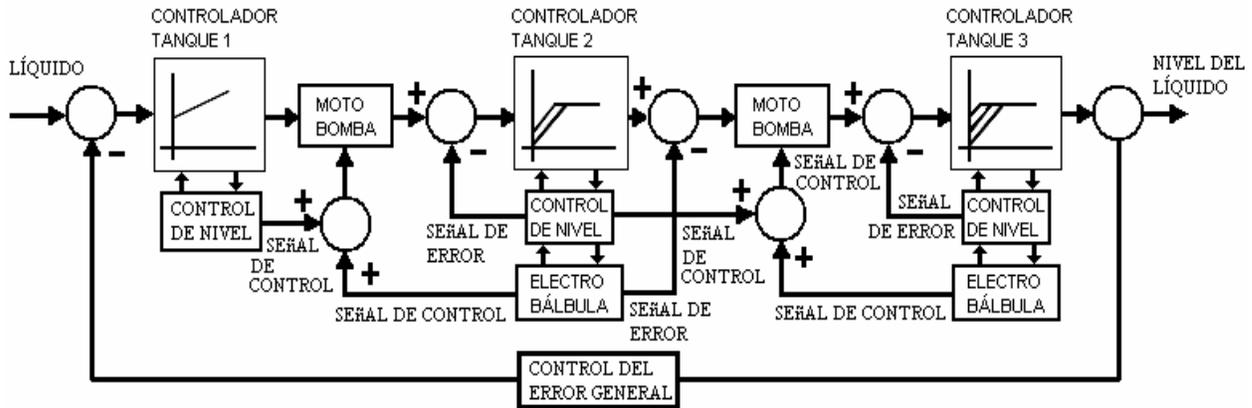


Fig. 25 Diagrama en bloques del controlador con lógica difusa del flujo tecnológico.

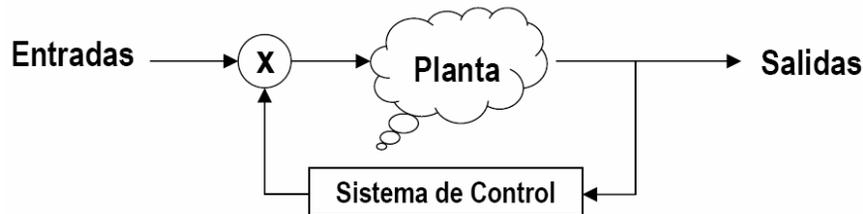


Fig. 24 Elementos básicos de un proceso controlado.

### **Fundamento teórico:**

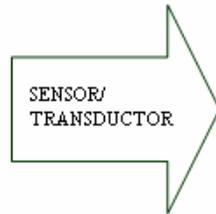
En este diagrama se sintetiza el control PID y el control multivariable asociados a una red neuronal controlada por lógica difusa. El control de las diferentes variables se integra como un todo en el control de nivel del proceso en la planta a controlar. Se integra el control como un todo para garantizar alta confiabilidad del sistema así como la detección de fallas en lugares muy precisos. Es necesario que el regulador PIC actúe sobre las variables para que el control con lógica difusa corrija los errores del sistema.

## 4.2.1 Señales que intervienen en el proceso.

Selección de los sensores

Fenómenos físicos.

- Presión
- Nivel
- Flujo



Señales eléctricas.

- Corriente
- Corriente
- Corriente

### **Clasificación de las señales.**

Las señales analógicas son aquellas que surgen de modo continuo en el proceso asociadas al mundo real las cuales son tomadas por los sensores o transductores para su correspondiente análisis.

Las señales discretas aparecen en el proceso una vez que son procesadas las analógicas. Estas son ejecutadas en su totalidad por el mecanismo de control automático.

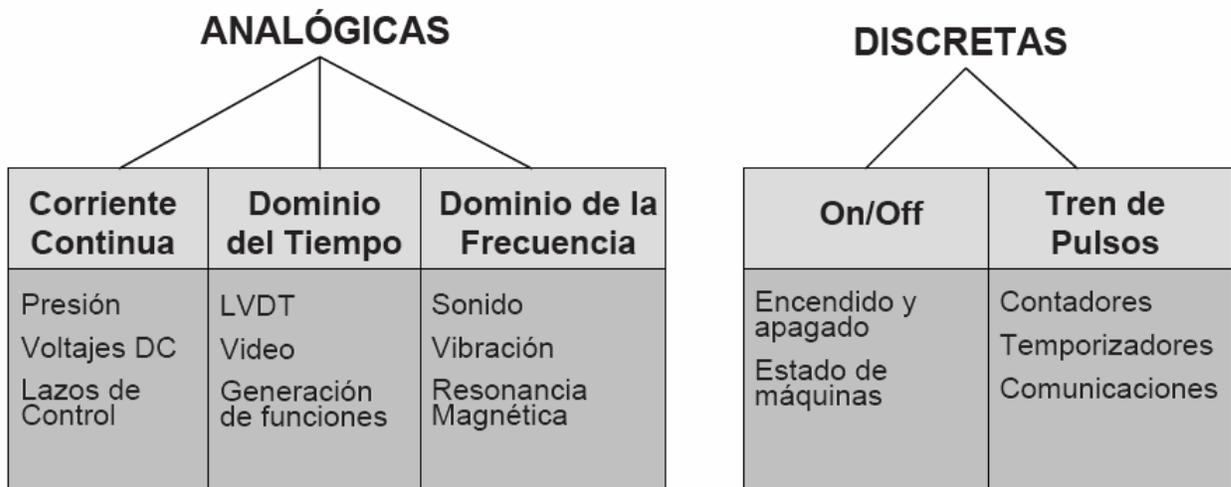


Fig. 26 Tipos de señales asociadas al proceso.

### **Selección Del Hardware.**

Tarjetas DAQ:



- Bajo Costo, Alta Velocidad, Alta Flexibilidad.
- Las señales necesitan acondicionamiento.

PLC y PAC: 

- Resistente, Confiable, varias formas de comunicación con PC, incluyen acondicionamiento.
- Baja velocidad, poca capacidad de análisis, requieren driver.

Instrumentos: 

- Dirigidos, alto desempeño.
- Poco flexibles, costosos, necesitan hardware y software para conexión con PC.

#### 4.2.2 Red del sistema automática.

Montaje de la red por puerto serie RS-485, protocolo profibus para la comunicación del sistema SCADA.

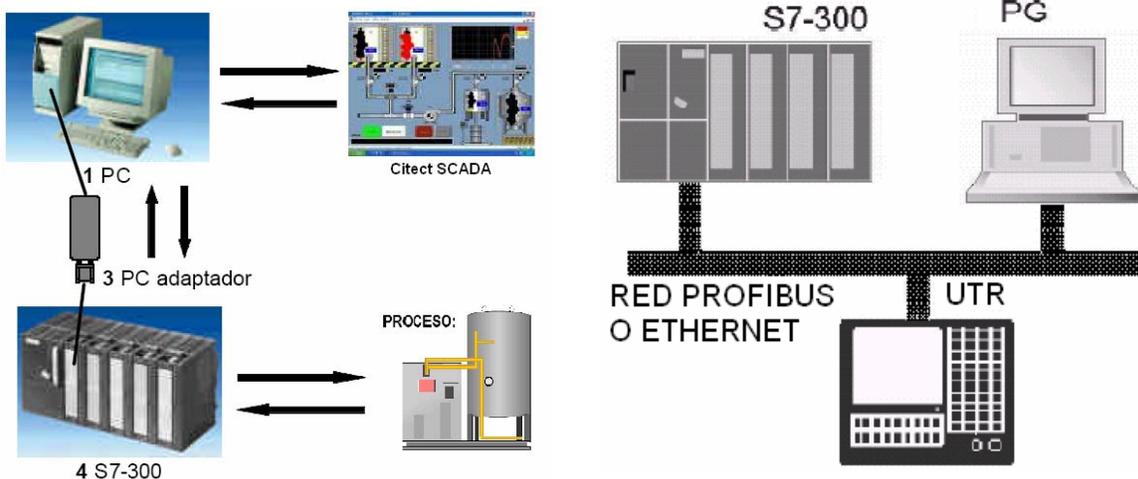


Fig. 27 Comunicación del sistema SCADA mediante la red MPI.

Las comunicaciones seriales entre los autómatas permite el intercambio de datos entre estos así como el procesamiento en un PC en tiempo real o el almacenamiento de datos para realizar un historial y su correspondiente estudio.

## 4.3 PROGRAMACIÓN DEL SCADA.

El SCADA utilizado en el proyecto de automatización del proceso es el Citect, uno de los más reconocidos a nivel mundial.

1. Se instala (si no se utiliza un servidor Web) y ejecuta en PC.
2. Se crea una aplicación gráfica que reproduce en la pantalla el proceso que se va a automatizar.
3. El programa SCADA se utiliza para crear una aplicación mediante herramientas. También soporta el uso de lenguajes de programación (según SCADA).
4. Las herramientas gráficas se utilizan para configurar una interfaz del usuario en base a objetos gráficos que son una imagen del proceso.
5. Es capaz de manejar funciones como alarmas, registros, tendencias (en tiempo real e históricas), y permite el almacenamiento y la exportación de los mismos a otros sistemas.

### **Características de un Sistema SCADA.**

1. Alto número de canales (y diversidad de dispositivos).
2. Registro en base de datos.
3. Gestión de Alarmas y Eventos.
4. Seguridad.
5. Trabajo en red.

### 4.3.1 Comunicación con el Citect SCADA.

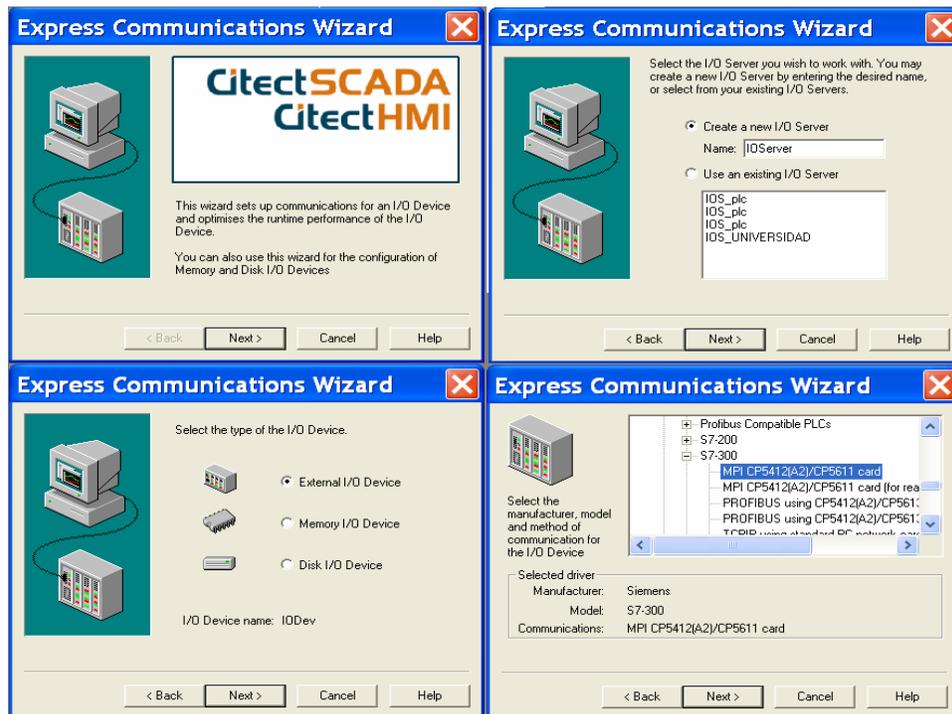


Fig. 28 Configuración de la comunicación para el sistema SCADA.

### 4.3.2 Visualización gráfica del proceso.

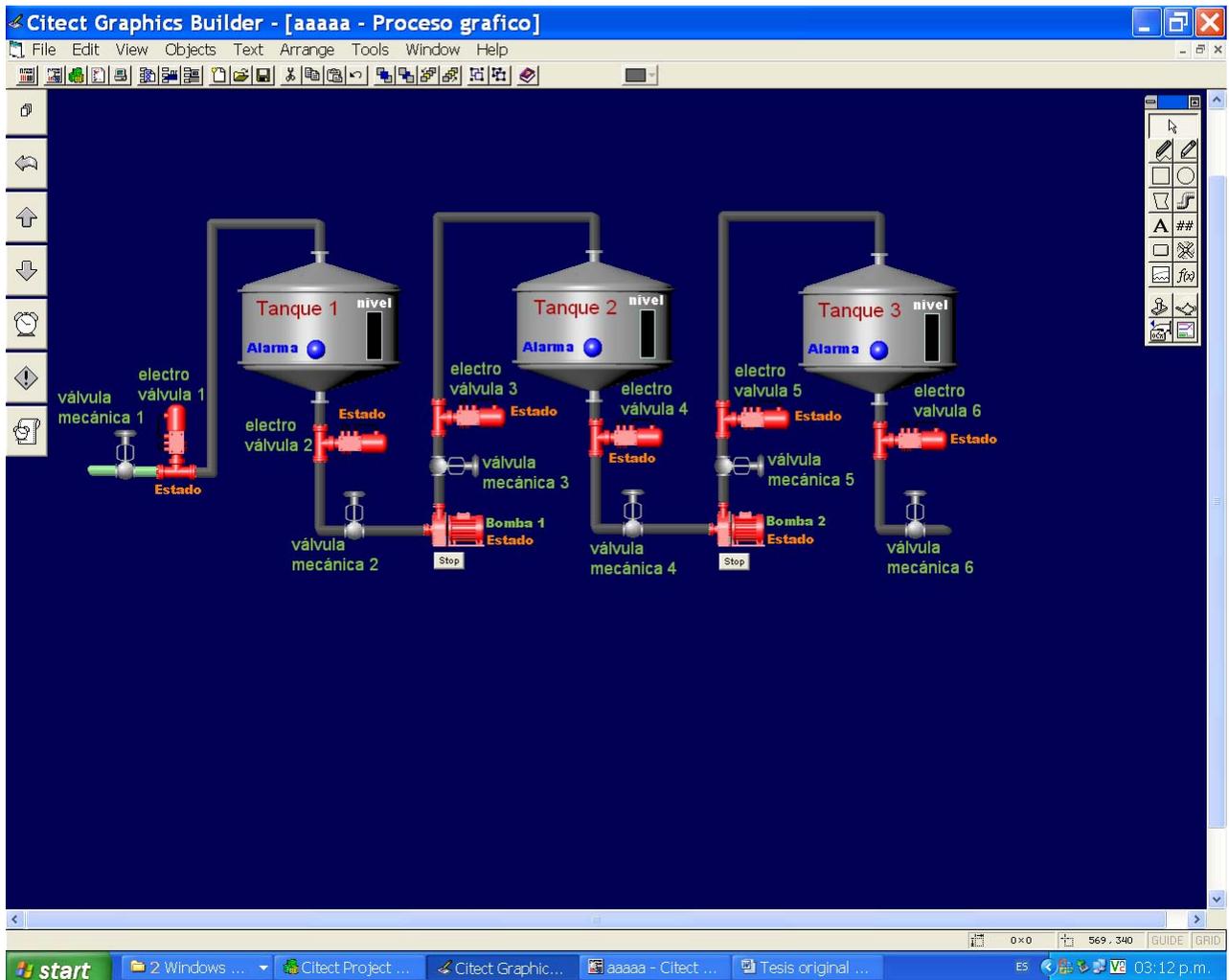


Fig. 29 Ventana gráfica del proceso.

La visualización gráfica del proceso permite ver la planta completa a escala muy pequeña y los diferentes estados de la instrumentación de control, además de crear la HMI (Interfase Humano Máquina) para el operador de la planta u otro usuario autorizado.

#### 4.3.2.1 Insertación de las diferentes variables tag que intervienen en la visualización gráfica del proceso en la planta.

Las variables tag son las variables por la cual se rige el sistema gráfico de monitoreo y control. Estas variables son definidas en cuanto a la cantidad de elementos que se quiera visualizar de forma gráfica en el proceso.

## Declaración de variables tag.

Etiquetas Variables

Nom de etiq variable	BIT_1	Nombre de dispos. E/S	DISK_PLC
Nombre del Grupo		Tipo de datos	DIGITAL
Dirección	D1	Plena escala de crudo	
Plena escala de crudo		Escala cero de crudo	
Escala cero de crudo		Formato	
Unidades de ing.			
Banda muerta			
Comentario	Alarm bit 1		

Agregar Reemplazar Eliminar Ayuda Vinculado: No

Fig. 30 Ventana para declarar las variables tag.

- Nombre de etiqueta variable: Es el nombre del elemento a visualizar.
- Nombre del dispositivo de entrada y salida: Dispositivo mediante el cual se establece la comunicación.
- Dirección: Dirección de almacenamiento de la variable en el PLC.
- Unidades de ingeniería: Unidades de medidas de las diferentes variables físicas u eléctricas, ejemplo Amper(A), voltaje (V), kilogramos (Kg.). etc.
- Comentario: Descripción específica del funcionamiento como variable.
- Tipo de datos: Dato físico que corresponde a la variable.
- Formato: Cantidad de dígitos mostrados en el display que sensa dicha variable, ejemplo #, ##, ### etc.

### Tag 1

- Nombre de etiqueta variable: Conducto 1.
- Nombre del dispositivo de entrada y salida: IODev\_plc
- Dirección: E0.0
- Unidades de ingeniería: No
- Comentario: Tubería de entrada para el tanque 1.
- Tipo de datos: Digital.
- Formato: # (un dígito "0", "1")

### Tag 2

- Nombre de etiqueta variable: Conducto 2.
- Nombre del dispositivo de entrada y salida: IODev\_plc
- Dirección: E0.1
- Unidades de ingeniería: No
- Comentario: Tubería de salida para el tanque 1, entrada para tanque 2.
- Tipo de datos: Digital.
- Formato: # (un dígito "0", "1")

### Tag 3

- Nombre de etiqueta variable: Conducto 3.
- Nombre del dispositivo de entrada y salida: IODev\_plc
- Dirección: E0.2
- Unidades de ingeniería: No
- Comentario: Tubería de salida para el tanque 2, entrada para tanque 3.
- Tipo de datos: Digital.
- Formato: # (un dígito "0", "1")

### Tag 4

- Nombre de etiqueta variable: electro válvula 1.
- Nombre del dispositivo de entrada y salida: IODev\_plc
- Dirección: A7.4
- Unidades de ingeniería: No
- Comentario: Abrir o cerrar entrada tanque 1.
- Tipo de datos: Digital.
- Formato: # (un dígito "0", "1")

### Tag 5

- Nombre de etiqueta variable: electro válvula 2.
- Nombre del dispositivo de entrada y salida: IODev\_plc
- Dirección: A4.6
- Unidades de ingeniería: No
- Comentario: Abrir o cerrar salida tanque 1.
- Tipo de datos: Digital.
- Formato: # (un dígito "0", "1")

### Tag 6

- Nombre de etiqueta variable: electro válvula 3
- Nombre del dispositivo de entrada y salida: IODev\_plc
- Dirección: A4.7
- Unidades de ingeniería: No
- Comentario: Abrir o cerrar entrada tanque 2.
- Tipo de datos: Digital.
- Formato: # (un dígito "0", "1")

### Tag 7

- Nombre de etiqueta variable: electro válvula 4
- Nombre del dispositivo de entrada y salida: IODev\_plc
- Dirección: A5.3
- Unidades de ingeniería: No
- Comentario: Abrir o cerrar salida tanque 2.
- Tipo de datos: Digital.
- Formato: # (un dígito "0", "1")

#### Tag 8

- Nombre de etiqueta variable: electro válvula 5
- Nombre del dispositivo de entrada y salida: IODev\_plc
- Dirección: A5.4
- Unidades de ingeniería: No
- Comentario: Abrir o cerrar entrada tanque 3.
- Tipo de datos: Digital.
- Formato: # (un dígito "0", "1")

#### Tag 9

- Nombre de etiqueta variable: electro válvula 6
- Nombre del dispositivo de entrada y salida: IODev\_plc
- Dirección: A7.3
- Unidades de ingeniería: No
- Comentario: Abrir o cerrar salida tanque 3.
- Tipo de datos: Digital.
- Formato: # (un dígito "0", "1")

#### Tag 10

- Nombre de etiqueta variable: Alarma tanque 1
- Nombre del dispositivo de entrada y salida: IODev\_plc
- Dirección: A7.2
- Unidades de ingeniería: No
- Comentario: Alarma por sobre nivel de tanque 1.
- Tipo de datos: Digital.
- Formato: # (un dígito "0", "1")

#### Tag 11

- Nombre de etiqueta variable: Alarma tanque 2.
- Nombre del dispositivo de entrada y salida: IODev\_plc
- Dirección: A5.2
- Unidades de ingeniería: No
- Comentario: Alarma por sobre nivel de tanque 2.
- Tipo de datos: Digital.
- Formato: # (un dígito "0", "1")

#### Tag 12

- Nombre de etiqueta variable: Alarma tanque 3.
- Nombre del dispositivo de entrada y salida: IODev\_plc
- Dirección: A6.4
- Unidades de ingeniería: No
- Comentario: Alarma por sobre nivel de tanque 3.
- Tipo de datos: Digital.
- Formato: #(un dígito "0", "1")

### Tag 13

- Nombre de etiqueta variable: Nivel bajo tanque 1
- Nombre del dispositivo de entrada y salida: IODev\_plc
- Dirección: E0.3
- Unidades de ingeniería: Por ciento (%)
- Comentario: Medición de nivel en porcentaje.
- Tipo de datos: Real(0-100)
- Formato: # # # (tres dígito)

### Tag 14

- Nombre de etiqueta variable: Nivel medio tanque 1
- Nombre del dispositivo de entrada y salida: IODev\_plc
- Dirección: E0.4
- Unidades de ingeniería: Por ciento (%)
- Comentario: Medición de nivel en porcentaje.
- Tipo de datos: Real(0-100)
- Formato: # # # (tres dígito)

### Tag 15

- Nombre de etiqueta variable: Nivel alto tanque 1
- Nombre del dispositivo de entrada y salida: IODev\_plc
- Dirección: E0.5
- Unidades de ingeniería: Por ciento (%)
- Comentario: Medición de nivel en porcentaje.
- Tipo de datos: Real(0-100)
- Formato: # # # (tres dígito)

### Tag 16

- Nombre de etiqueta variable: Nivel bajo tanque 2
- Nombre del dispositivo de entrada y salida: IODev\_plc
- Dirección: E1.0
- Unidades de ingeniería: Por ciento (%)
- Comentario: Medición de nivel en porcentaje.
- Tipo de datos: Real(0-100)
- Formato: # # # (tres dígito)

### Tag 17

- Nombre de etiqueta variable: Nivel medio tanque 2
- Nombre del dispositivo de entrada y salida: IODev\_plc
- Dirección: E1.1
- Unidades de ingeniería: Por ciento (%)
- Comentario: Medición de nivel en porcentaje.
- Tipo de datos: Real(0-100)
- Formato: # # # (tres dígito)

#### Tag 18

- Nombre de etiqueta variable: Nivel alto tanque 2
- Nombre del dispositivo de entrada y salida: IODev\_plc
- Dirección: E1.2
- Unidades de ingeniería: Por ciento (%)
- Comentario: Medición de nivel en porcentaje.
- Tipo de datos: Real(0-100)
- Formato: # # # (tres dígito)

#### Tag 19

- Nombre de etiqueta variable: Nivel bajo tanque 3
- Nombre del dispositivo de entrada y salida: IODev\_plc
- Dirección: E1.5
- Unidades de ingeniería: Por ciento (%)
- Comentario: Medición de nivel en porcentaje.
- Tipo de datos: Real(0-100)
- Formato: # # # (tres dígito)

#### Tag 20

- Nombre de etiqueta variable: Nivel medio tanque 3
- Nombre del dispositivo de entrada y salida: IODev\_plc
- Dirección: E1.6
- Unidades de ingeniería: Por ciento (%)
- Comentario: Medición de nivel en porcentaje.
- Tipo de datos: Real(0-100)
- Formato: # # # (tres dígito)

#### Tag 21

- Nombre de etiqueta variable: Nivel alto tanque 3
- Nombre del dispositivo de entrada y salida: IODev\_plc
- Dirección: E1.7
- Unidades de ingeniería: Por ciento (%)
- Comentario: Medición de nivel en porcentaje.
- Tipo de datos: Real(0-100)
- Formato: # # # (tres dígito)

#### Tag 22

- Nombre de etiqueta variable: Stop bomba 1
- Nombre del dispositivo de entrada y salida: IODev\_plc
- Dirección: M100.0
- Unidades de ingeniería:
- Comentario: Parada de la bomba.
- Tipo de datos: Digital.
- Formato: #(un dígito "0", "1")

## Tag 23

- Nombre de etiqueta variable: Stop bomba 2
- Nombre del dispositivo de entrada y salida: IODev\_plc
- Dirección: M100.2
- Unidades de ingeniería:
- Comentario: Parada de la bomba.
- Tipo de datos: Digital.
- Formato: #(un dígito "0", "1")

## 4.4 PROGRAMACIÓN DEL PLC.

El PLC cuyas siglas en inglés significan Controlador Lógico Programable es el dispositivo encargado de llevar todas las funciones de control del proceso, sin el sería imposible la automatización de dicha planta. Este se programa mediante un software según el fabricante diseñado para estas aplicaciones. Nuestra automatización se realizara con un dispositivo de la marca SIEMENS del tipo S7-300 con versión de programa Step7v5.4SP1. Para programar el PLC es necesario configurarle la comunicación para la interfaz con el PC además de los diferentes tipos de módulos a usar en la entrada o salida de datos con la periferia.

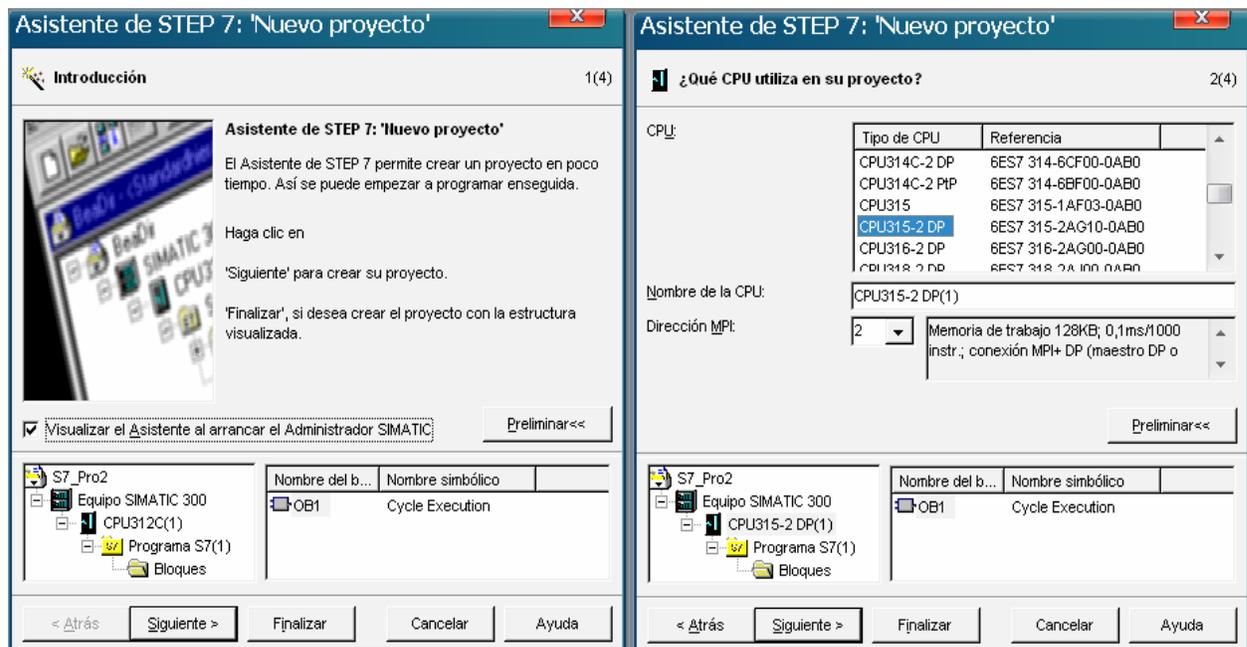


Fig. 31 Configuración del proyecto.

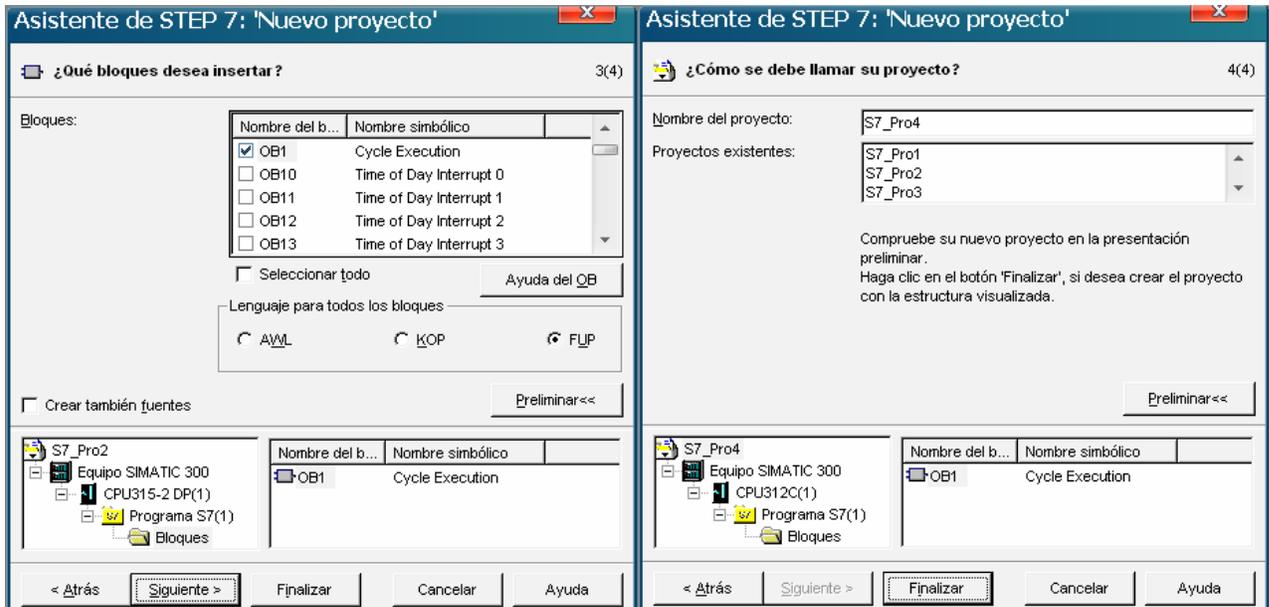


Fig. 32 Configuración del proyecto.

Una vez configurado nuestro proyecto procedemos a configurar la interface de programación.

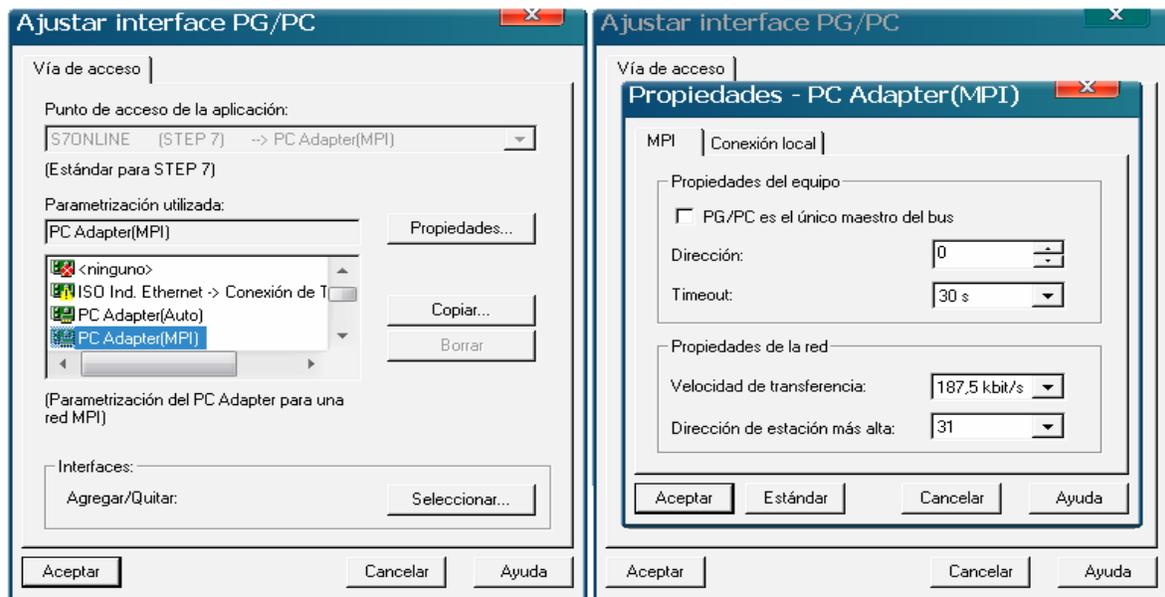


Fig. 33 Configuración de la interfase de programación.

Ya configurada la interfase de programación procedemos al programa como tal.

#### 4.4.1 Programa STEP 7.

##### AUTÓMATA SIMATIC S7-300

El autómata SIMATIC S7-300 es un módulo de control de gama baja para controles pequeños y medianos. Para una óptima adaptación en las tareas a automatizar existe una ampliación en los grupos de trabajo. El Autómata S7 contiene en el bastidor, una CPU y grupos de entradas y salidas. Normalmente contiene también procesadores de comunicación y módulos de funciones para tareas especiales, como la regulación de un motor. El programa almacenado en el Autómata (PLC) controla y regula, con el programa S7, una máquina o un proceso. El grupo de E/A se comunica con el programa S7 a través de las direcciones de entrada y salida. La programación del sistema se realiza con el Software STEP 7.

Para realizar la programación del PLC debemos seleccionar el regulador más óptimo para el control de proceso.

##### **Selección del regulador**

Es prácticamente imposible influenciar las propiedades de los sistemas o procesos regulados porque resultan de datos mecánicos y de ingeniería química. La elección del tipo de regulador más apropiado al sistema o proceso regulado y su adaptación al comportamiento temporal de este último son los únicos medios de obtener un buen resultado de regulación. Para nuestro trabajo hemos seleccionado un regulador con lógica fuzzy asociado al PID.

##### **Funciones básicas.**

Un regulador realizado por medio de los FB se compone de una serie de funciones parciales parametrizables por el usuario. Además del regulador con su algoritmo PID, los FB incluyen funciones de acondicionamiento de valor de consigna, de valor real así como de la magnitud manipulada calculada.

#### 4.4.2 Declaración de bloques programados en el PLC.

BLOQUES DE ORGANIZACIÓN (OB).

##### **Modo de funcionamiento del OB1:**

El OB 1 presenta la prioridad más baja de todos los OBs sometidos a vigilancia en lo que respecta a su tiempo de ejecución. Con excepción del OB 90, todos los demás OBs pueden interrumpir la ejecución del OB 1. Los eventos siguientes dan lugar a que el sistema operativo llame al OB 1:

- Final de la ejecución del arranque.
- Final de la ejecución del OB 1 (durante el ciclo anterior).

## Datos locales de OB1.

Variable	Tipo de datos	Descripción
OB1_EV_CLASS	BYTE	Clase de evento e identificadores: B#16#11: activo
OB1_SCAN_1	BYTE	B#16#01: Conclusión del re arranque completo (arranque en caliente).
		B#16#02: Conclusión del re arranque.
		B#16#03: Conclusión del ciclo libre.
		B#16#04: Conclusión del □ arranque en frío.
		B#16#05: Primer ciclo del OB1 de la nueva CPU maestra tras la conmutación del maestro de reserva y el STOP de la antigua CPU maestra.
OB1_PRIORITY	BYTE	Prioridad: 1
OB1_OB_NUMBR	BYTE	Número de OB (01).
OB1_RESERVED_1	BYTE	Reservado.
OB1_RESERVED_2	BYTE	Reservado.
OB1_PREV_CYCLE	INT	Tiempo de ejecución del ciclo anterior (ms).
OB1_MIN_CYCLE	INT	Tiempo de ciclo mínimo (ms) desde el último arranque.
OB1_MAX_CYCLE	INT	Tiempo de ciclo máximo (ms) desde el último arranque.
OB1_DATE_TIME	DATE_AND_TIME	Fecha y hora a las que se solicitó el OB.

Tabla 4. La tabla siguiente contiene las variables temporales (TEMP) del OB 1.

### **Modo de funcionamiento del OB80:**

El sistema operativo de la CPU llamará al OB 80 cuando en la ejecución de un OB surja uno de los siguientes errores: se ha excedido el tiempo de ciclo, error de acuse en el procesamiento de un OB, adelanto de la hora (salto horario) para el arranque de un OB, regreso a RUN después de CiR. Si surge por ejemplo un evento de arranque de un OB de alarma cíclica antes de haber concluido la ejecución anterior de este mismo OB, el sistema operativo llama entonces al OB 80. Si el OB 80 no ha sido programado, la CPU pasa al estado operativo STOP.

Con ayuda de las SFCs 39 a 42 puede bloquearse o retardarse y habilitarse de nuevo el OB de error de tiempo.

Datos locales del OB 80 de error de tiempo.

Variable	Tipo de datos	Descripción
OB80_EV_CLASS	BYTE	Clase de evento e identificadores: B#16#35
OB80_FLT_ID	BYTE	Código de error (valores posibles: B#16#01, B#16#02, B#16#05, B#16#06, B#16#07, B#16#08, B#16#09, B#16#0A, B#16#0B)
OB80_PRIORITY	BYTE	Prioridad: el OB se ejecuta en RUN con la prioridad 26 y, en caso de rebase por exceso del búfer de petición, con la prioridad 28
OB80_OB_NUMBR	BYTE	Número de OB (80).
OB80_RESERVED_1	BYTE	Reservado.
OB80_RESERVED_2	BYTE	Reservado.
OB80_ERROR_INFO	WORD	Información de error: depende del código de error.
OB80_ERR_EV_CLASS	BYTE	Clase del evento que ha activado el error.
OB80_ERR_EV_NUM	BYTE	Número del evento que ha activado el error.
OB80_OB_PRIORITY	BYTE	Información de error: depende del código de error.
OB80_OB_NUM	BYTE	Información de error: depende del código de error.
OB80_DATE_TIME	DATE_AND_TIME	Fecha y hora a las que se solicitó el OB.

Tabla 5. La tabla siguiente contiene las variables temporales (TEMP) del OB de error de tiempo.

### **Modo de funcionamiento del OB84:**

El sistema operativo de la CPU llama al OB 84 en los siguientes casos:

- Tras la detección y eliminación de errores de memoria
- En WinAC RTX: error en el sistema operativo del PC.

Si no ha programado el OB 84, la CPU no cambiará al estado operativo STOP.

Puede bloquear o retardar el OB de error de hardware de la CPU con ayuda de las SFCs 39 hasta 42 y volver a habilitarlo.

Datos locales del OB 84 de avería de hardware.

Variable	Tipo de datos	Descripción
OB84_EV_CLASS	BYTE	Clase de evento e identificadores: B#16#38: Evento saliente. B#16#35, B#16#39:: Evento entrante.
OB84_FLT_ID	BYTE	Código de error (B#16#82, B#16#83, B#16#85, B#16#86, B#16#87).
OB84_PRIORITY	BYTE	Prioridad; parametrizable con STEP 7 (HW Config).
OB84_OB_NUMBR	BYTE	Número de OB (84)
OB84_RESERVED_1	BYTE	Reservado
OB84_RESERVED_2	BYTE	Reservado
OB84_RESERVED_3	WORD	Reservado
OB84_RESERVED_4	DWORD	Reservado
OB84_DATE_TIME	DATE_AND_TIME	Fecha y hora a las que se solicitó el OB

La siguiente tabla muestra qué evento ha provocado el inicio del OB 84.

OB84_EV_CLASS	OB84_FLT_ID	Evento de arranque del OB 84
B#16#35	B#16#82	Error de memoria del sistema operativo detectado y eliminado
B#16#35	B#16#83	Acumulación de errores de memoria detectados y eliminados
B#16#35	B#16#85	Error en el sistema operativo del PC
B#16#39	B#16#86	Potencia de un acoplamiento H-Sync
B#16#35	B#16#87	Error de memoria de varios bits detectado y corregido
B#16#35	B#16#83	Acumulación de errores de memoria detectados y corregidos

Tabla 6. La tabla siguiente contiene las variables temporales (TEMP) del OB de avería de CPU.

### **Modo de funcionamiento del OB87:**

El sistema operativo de la CPU llama al OB 87 cuando se produce un evento activado por un error de comunicación.

Si no ha programado el OB 87 y aparece un evento de arranque para el OB 87, la CPU se comportará como sigue:

- Una CPU S7-300 cambiará al estado operativo STOP.
- Una CPU S7-400 no cambiará al estado operativo STOP.

Datos locales del OB 87 de error de comunicación.

Variable	Tipo de datos	Descripción
OB87_EV_CLASS	BYTE	Clase de evento e identificadores: B#16#35
OB87_FLT_ID	BYTE	Código de error
		(valores posibles: B#16#D2, B#16#D3, B#16#D4, B#16#D5, B#16#E1, B#16#E2, B#16#E3, B#16#E4, B#16#E5, B#16#E6)
OB87_PRIORITY	BYTE	Prioridad; parametrizable con STEP 7 (HW Config)
OB87_OB_NUMBR	BYTE	Número de OB (87)
OB87_RESERVED_1	BYTE	Reservado
OB87_RESERVED_2	BYTE	Reservado
OB87_RESERVED_3	WORD	En función del código de error
OB87_RESERVED_4	DWORD	En función del código de error
OB87_DATE_TIME	DATE_AND_TIME	Fecha y hora a las que se solicitó el OB

Tabla 7. Variables temporales (TEMP) del OB de error de comunicación.

### **Modo de funcionamiento del OB121:**

El sistema operativo de la CPU llama al OB 121 cuando aparece un evento activado por un error durante la ejecución del programa. Así por ejemplo, si dentro del programa se llama a un bloque que no fue cargado en la CPU, el OB 121 será invocado.

### **Modo de funcionamiento del OB 121 de error de programación.**

El OB 121 funciona en la misma prioridad que el bloque que ha sido interrumpido. Si no está programado el OB 121, la CPU pasará de RUN a STOP.

S7 dispone de las SFCs siguientes que permiten enmascarar y desenmascarar eventos de arranque del OB 121, mientras se ejecuta el programa:

- La SFC 36"MSK\_FLT" enmascara determinados códigos de error.
- La SFC 37"DMSK\_FLT" desenmascara los códigos de error enmascarados por la SFC 36.
- La SFC 38"READ\_ERR" lee el registro de estado de eventos.

Variable	Tipo de datos	Descripción
OB121_EV_CLASS	BYTE	Clase de evento e identificadores: B#16#25
OB121_SW_FLT	BYTE	Código de error (valores posibles: B#16#21, B#16#22, B#16#23, B#16#24, B#16#25, B#16#26, B#16#27, B#16#28, B#16#29, B#16#30, B#16#31, B#16#32, B#16#33, B#16#34, B#16#35, B#16#3A, B#16#3C, B#16#3D, B#16#3E, B#16#3F)
OB121_PRIORITY	BYTE	Prioridad: Prioridad del OB en el que se ha producido el error.
OB121_OB_NUMBR	BYTE	Número de OB (121)
OB121_BLK_TYPE	BYTE	Tipo de bloque en el que ha aparecido el error (en S7-300 no se consigna aquí un valor vigente): B#16#88: OB, B#16#8A: DB, B#16#8C: FC, B#16#8E: FB
OB121_RESERVED_1	BYTE	Reservado
OB121_FLT_REG	WORD	Origen del error (en función del código de error), eje.
		Registro en el que ha aparecido el error de conversión.
		Dirección errónea (error de lectura/escritura).
		Número erróneo de un temporizador, de un contador o de un bloque.
		Área de memoria errónea.
OB121_BLK_NUM	WORD	Número del bloque con el comando MC7 causante del error (en los S7-300 aquí no se registran valores válidos).
OB121_PRG_ADDR	WORD	Dirección relativa del comando MC7 causante del error (en S7-300 no se consigna aquí un valor vigente).
OB121_DATE_TIME	DATE_AND_TIME	Fecha y hora a las que se solicitó el OB.

Tabla 8. La tabla siguiente contiene las variables temporales (TEMP) del OB de error de programación.

### **Modo de funcionamiento del OB122:**

El sistema operativo de la CPU llama al OB 122 cuando aparece un error al acceder a datos de un módulo. Cuando por ejemplo, la CPU reconoce un error de lectura al acceder a datos de un módulo de señales, el sistema operativo llama entonces al OB 122.

### **Modo de funcionamiento del OB 122 de error de acceso a la periferia.**

El OB 122 funciona en la misma prioridad que el bloque interrumpido. Si el OB 122 no está programado, la CPU cambiará el estado operativo de RUN a STOP.

Para enmascarar y desenmascarar los eventos de arranque del OB 122 mientras se procesa su programa, S7 dispone de las siguientes SFCs:

- La SFC 36"MSK\_FLT" enmascara determinados códigos de error.
- La SFC 37"DMSK\_FLT" desenmascara los códigos de error que fueron enmascarados por la SFC 36.
- La SFC 38"READ\_ERR" lee el registro de estado de eventos.

Variable	Tipo de datos	Descripción
OB122_EV_CLASS	BYTE	Clase de evento e identificadores: B#16#29
OB122_SW_FLT	BYTE	Código de error.
		B#16#42: error de acceso a la periferia, en lectura.
		B#16#43: error de acceso a la periferia, en escritura.
OB122_PRIORITY	BYTE	Prioridad:
		Prioridad del OB en el que ha aparecido el error.
OB122_OB_NUMBR	BYTE	Número de OB (122)
OB122_BLK_TYPE	BYTE	Tipo de bloque en el que ha aparecido el error (B#16#88: OB, B#16#8C: FC, B#16#8E: FB) (en los S7-300 aquí no se registran valores válidos)
OB122_MEM_AREA	BYTE	Área de memoria y tipo de acceso:
		<ul style="list-style-type: none"> <li>• Bits 7 a 4: tipo de acceso.</li> </ul>
		- 0: Acceso a bit.
		- 1: Acceso a byte.
		- 2: Acceso a palabra.
		- 3: Acceso a palabra doble.
		<ul style="list-style-type: none"> <li>• Bits 3 a 0: área de memoria.</li> </ul>
		- 0: Área de periferia.
- 1: Imagen del proceso de las entradas.		
- 2: Imagen del proceso de las salidas.		
OB122_MEM_ADDR	WORD	Dirección en la memoria en la que ha aparecido el error.
OB122_BLK_NUM	WORD	Número del bloque con el comando MC7 causante del error (en los S7-300 aquí no se registran valores válidos).
OB122_PRG_ADDR	WORD	Dirección relativa del comando MC7 causante del error (en los S7-300 aquí no se registran valores válidos).
OB122_DATE_TIME	DATE_AND_TIME	Fecha y hora a las que se solicitó el OB.

Tabla 9. La tabla siguiente contiene las variables temporales (TEMP) del OB de error de acceso a la periferia.

## BLOQUES DE DATOS (DB)

Los Bloques de Datos (DBs) pueden ser utilizados en el programa para salvar información en la CPU. Tienen una capacidad de memoria de hasta 8 KBytes (8192 Bytes). Existen dos tipos de Bloques de Datos. DBs Globales, en los cuales todos los OBs, FBs y FCs pueden guardar o leer datos y DBs de Instancia, los cuales se encuentran asignados a un FB en particular. En los DBs, se pueden almacenar diferentes tipos de datos (ejemplo BOOL o WORD) de manera aleatoria. La estructura de un DB se genera a través de la herramienta **Editor KOP/ AWL//FUP: Programar Bloques** de STEP7.

Los DBs de instancia declarados para almacenar los datos de los bloques de funciones en la CPU son DB50 y DB30.

Estos bloques permiten el control del proceso estadístico. Esta información del proceso es útil para controlar por ejemplo la calidad del producto, usando paquetes estadísticos para supervisión y análisis de datos.

	Dirección	Declaración	Nombre	Tipo	Valor inicial	Valor actual	Comentario
1	0.0	in	COM_RST	BOOL	FALSE	FALSE	complete restart
2	0.1	in	MAN_ON	BOOL	TRUE	TRUE	manual value on
3	0.2	in	PVPER_ON	BOOL	FALSE	FALSE	process variable peripherie on
4	0.3	in	P_SEL	BOOL	TRUE	TRUE	proportional action on
5	0.4	in	I_SEL	BOOL	TRUE	TRUE	integral action on
6	0.5	in	INT_HOLD	BOOL	FALSE	FALSE	integral action hold
7	0.6	in	I_JTL_ON	BOOL	FALSE	FALSE	initialization of the integral action
8	0.7	in	D_SEL	BOOL	FALSE	TRUE	derivative action on
9	2.0	in	CYCLE	TIME	T#1S	T#1S	sample time
10	6.0	in	SP_INT	REAL	0.000000e+...	0.000000e...	internal setpoint
11	10.0	in	PV_IN	REAL	0.000000e+...	0.000000e...	process variable in
12	14.0	in	PV_PER	WORD	W#16#0	W#16#0	process variable peripherie
13	16.0	in	MAN	REAL	0.000000e+...	0.000000e...	manual value
14	20.0	in	GAIN	REAL	2.000000e+...	2.000000e...	proportional gain
15	24.0	in	TI	TIME	T#20S	T#20S	reset time
16	28.0	in	TD	TIME	T#10S	T#10S	derivative time
17	32.0	in	TM_LAG	TIME	T#2S	T#2S	time lag of the derivative action
18	36.0	in	DEADB_W	REAL	0.000000e+...	0.000000e...	dead band width
19	40.0	in	LMN_HLM	REAL	1.000000e+...	1.000000e...	manipulated value high limit
20	44.0	in	LMN_LLM	REAL	0.000000e+...	0.000000e...	manipulated value low limit
21	48.0	in	PV_FAC	REAL	1.000000e+...	1.000000e...	process variable factor
22	52.0	in	PV_OFF	REAL	0.000000e+...	0.000000e...	process variable offset
23	56.0	in	LMN_FAC	REAL	1.000000e+...	1.000000e...	manipulated value factor
24	60.0	in	LMN_OFF	REAL	0.000000e+...	0.000000e...	manipulated value offset
25	64.0	in	I_JTLVAL	REAL	0.000000e+...	0.000000e...	initialization value of the integral action
26	68.0	in	DISV	REAL	0.000000e+...	0.000000e...	disturbance variable
27	72.0	out	LMN	REAL	0.000000e+...	0.000000e...	manipulated value
28	76.0	out	LMN_PER	WORD	W#16#0	W#16#0	manipulated value peripherie
29	78.0	out	QLMN_HLM	BOOL	FALSE	FALSE	high limit of manipulated value reached
30	78.1	out	QLMN_LLM	BOOL	FALSE	FALSE	low limit of manipulated value reached
31	80.0	out	LMN_P	REAL	0.000000e+...	0.000000e...	proportionality component
32	84.0	out	LMN_I	REAL	0.000000e+...	0.000000e...	integral component
33	88.0	out	LMN_D	REAL	0.000000e+...	0.000000e...	derivative component
34	92.0	out	PV	REAL	0.000000e+...	0.000000e...	process variable
35	96.0	out	ER	REAL	0.000000e+...	0.000000e...	error signal
36	100.0	stat	sInvtAlt	REAL	0.000000e+...	0.000000e...	
37	104.0	stat	sInteilAlt	REAL	0.000000e+...	0.000000e...	
38	108.0	stat	sRestInt	REAL	0.000000e+...	0.000000e...	
39	112.0	stat	sRestDif	REAL	0.000000e+...	0.000000e...	
40	116.0	stat	sRueck	REAL	0.000000e+...	0.000000e...	

Fig. 34 Bloque de instancia del regulador 1.

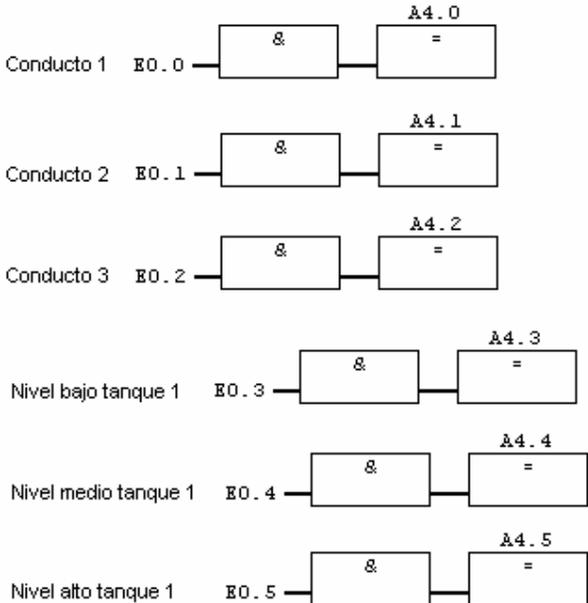
	Dirección	Declaración	Nombre	Tipo	Valor inicial	Valor actual	Comentario
1	0.0	in	COM_RST	BOOL	FALSE	FALSE	complete restart
2	0.1	in	MAN_ON	BOOL	TRUE	TRUE	manual value on
3	0.2	in	PVPER_ON	BOOL	FALSE	FALSE	process variable peripherie on
4	0.3	in	P_SEL	BOOL	TRUE	TRUE	proportional action on
5	0.4	in	I_SEL	BOOL	TRUE	TRUE	integral action on
6	0.5	in	INT_HOLD	BOOL	FALSE	FALSE	integral action hold
7	0.6	in	I_TL_ON	BOOL	FALSE	FALSE	initialization of the integral action
8	0.7	in	D_SEL	BOOL	FALSE	FALSE	derivative action on
9	2.0	in	CYCLE	TIME	T#1S	T#1S	sample time
10	6.0	in	SP_INT	REAL	0.000000e+...	0.000000e...	internal setpoint
11	10.0	in	PV_IN	REAL	0.000000e+...	0.000000e...	process variable in
12	14.0	in	PV_PER	WORD	W#16#0	W#16#0	process variable peripherie
13	16.0	in	MAN	REAL	0.000000e+...	0.000000e...	manual value
14	20.0	in	GAIN	REAL	2.000000e+...	2.000000e...	proportional gain
15	24.0	in	TI	TIME	T#20S	T#20S	reset time
16	28.0	in	TD	TIME	T#10S	T#10S	derivative time
17	32.0	in	TM_LAG	TIME	T#2S	T#2S	time lag of the derivative action
18	36.0	in	DEADB_WV	REAL	0.000000e+...	0.000000e...	dead band width
19	40.0	in	LMN_HLM	REAL	1.000000e+...	1.000000e...	manipulated value high limit
20	44.0	in	LMN_LLM	REAL	0.000000e+...	0.000000e...	manipulated value low limit
21	48.0	in	PV_FAC	REAL	1.000000e+...	1.000000e...	process variable factor
22	52.0	in	PV_OFF	REAL	0.000000e+...	0.000000e...	process variable offset
23	56.0	in	LMN_FAC	REAL	1.000000e+...	1.000000e...	manipulated value factor
24	60.0	in	LMN_OFF	REAL	0.000000e+...	0.000000e...	manipulated value offset
25	64.0	in	I_TLVAL	REAL	0.000000e+...	0.000000e...	initialization value of the integral action
26	68.0	in	DISV	REAL	0.000000e+...	0.000000e...	disturbance variable
27	72.0	out	LMN	REAL	0.000000e+...	0.000000e...	manipulated value
28	76.0	out	LMN_PER	WORD	W#16#0	W#16#0	manipulated value peripherie
29	78.0	out	QLMN_HLM	BOOL	FALSE	FALSE	high limit of manipulated value reached
30	78.1	out	QLMN_LLM	BOOL	FALSE	FALSE	low limit of manipulated value reached
31	80.0	out	LMN_P	REAL	0.000000e+...	0.000000e...	proportionality component
32	84.0	out	LMN_I	REAL	0.000000e+...	0.000000e...	integral component
33	88.0	out	LMN_D	REAL	0.000000e+...	0.000000e...	derivative component
34	92.0	out	PV	REAL	0.000000e+...	0.000000e...	process variable
35	96.0	out	ER	REAL	0.000000e+...	0.000000e...	error signal
36	100.0	stat	sInVAIt	REAL	0.000000e+...	0.000000e...	
37	104.0	stat	sInteIlAIt	REAL	0.000000e+...	0.000000e...	
38	108.0	stat	sRestInt	REAL	0.000000e+...	0.000000e...	
39	112.0	stat	sRestDif	REAL	0.000000e+...	0.000000e...	
40	116.0	stat	sRueck	REAL	0.000000e+...	0.000000e...	
41	120.0	stat	sLmn	REAL	0.000000e+...	0.000000e...	
42	124.0	stat	sbArwH...	BOOL	FALSE	FALSE	
43	124.1	stat	sbArwLL...	BOOL	FALSE	FALSE	
44	124.2	stat	sblImOn	BOOL	TRUE	TRUE	

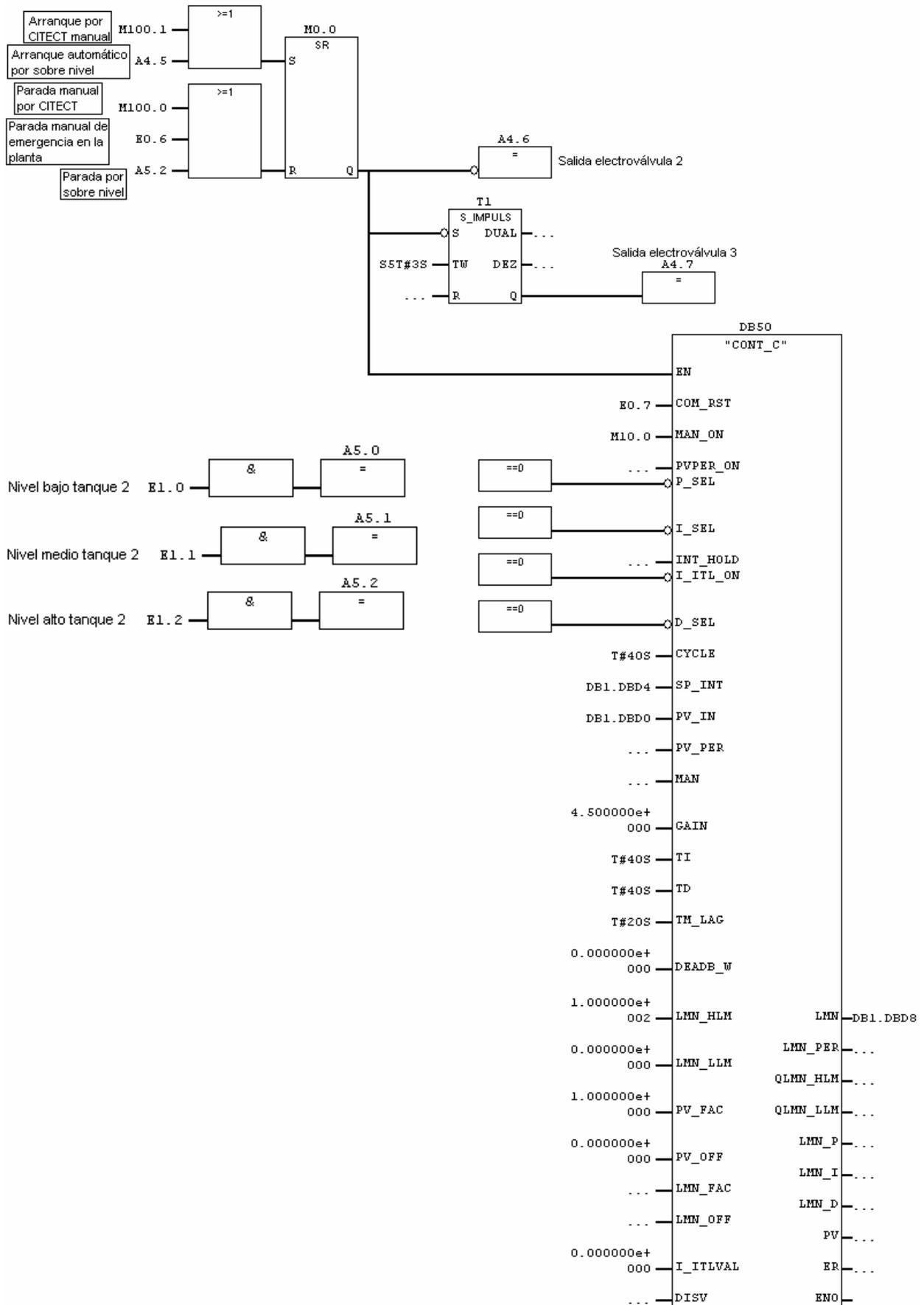
Fig. 35 Bloque de instancia del regulador 2.

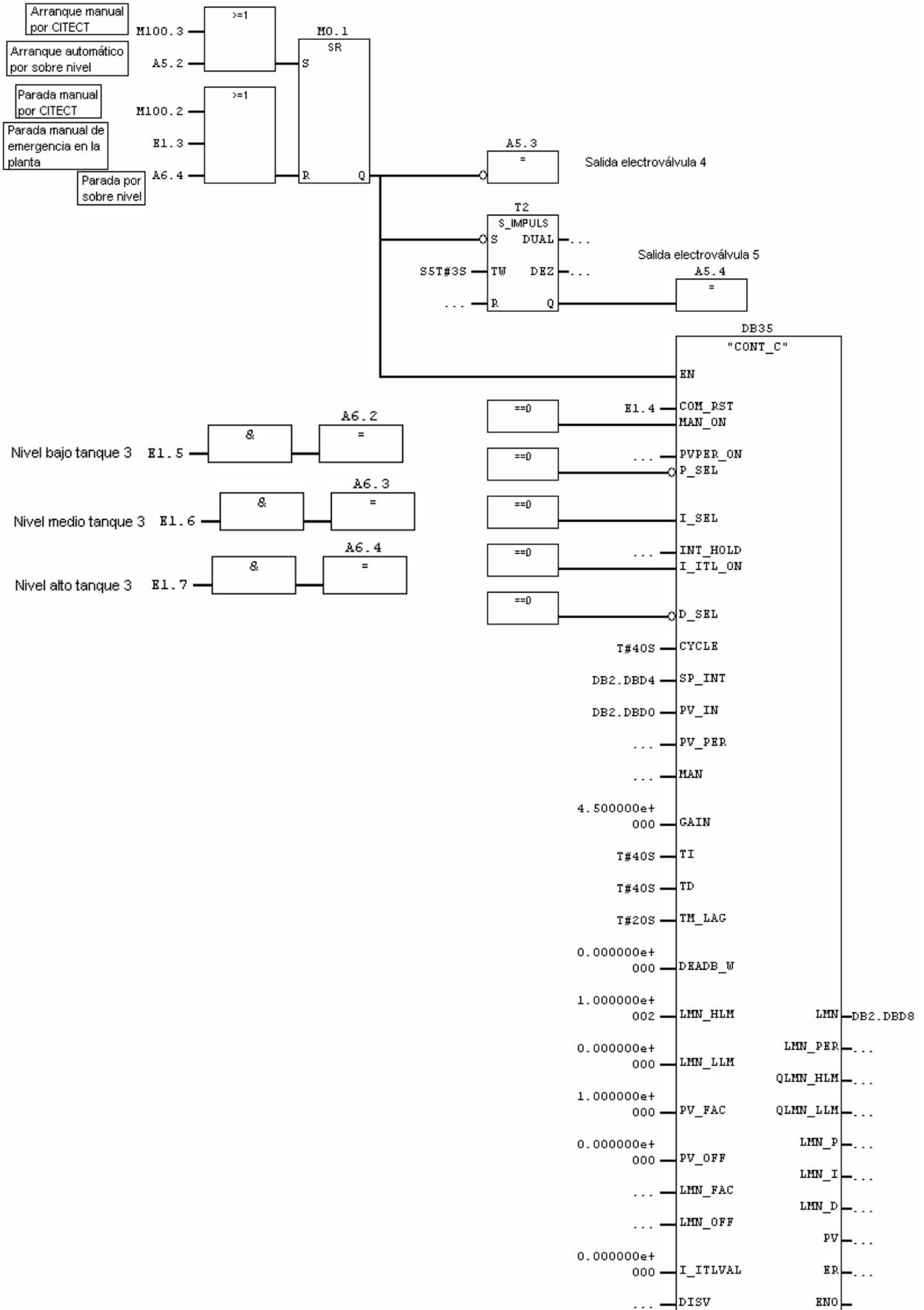


El FB 41 descrito en este capítulo sólo está validado para la CPU S7/C7 que disponen del nivel de procesamiento de alarmas cíclicas:

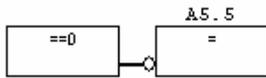
Programa en FUP



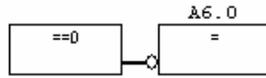




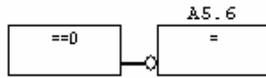
Falla del tiempo de ciclo (OB 80)



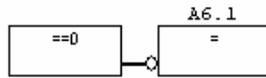
Error de programación (OB 121)



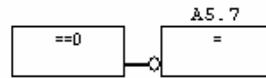
Falla en la CPU (OB84)



Error de acceso a módulo (OB 122)



Falla en la comunicación (OB 87)



## Programa en KOP



Bloque de regulación FB 41 para tanque 2

```

U(
O   M   100.0
O   E   0.6
O   A   5.2
)
R   M   0.0
U   M   0.0
=   L   20.0
UN  L   20.0
=   A   4.6
UN  L   20.0
L   SST#3S
SI  T   1
NOP 0
NOP 0
NOP 0
U   T   1
=   A   4.7
U   E   0.7
=   L   20.1
BLD 103
U   M   10.0
=   L   20.2
BLD 103
UN  ==0
=   L   20.4
BLD 103
UN  ==0
=   L   20.5
BLD 103
UN  ==0
=   L   20.7
BLD 103
UN  ==0
=   L   21.0
BLD 103
U   L   20.0
SPENB _001
CALL "CONT_C" , DB50
COM_RST :=L20.1
MAN_ON  :=L20.2
PVPER_ON:=
P_SEL   :=L20.4
I_SEL   :=L20.5
INT_HOLD:=
I_ITL_ON:=L20.7
D_SEL   :=L21.0
CYCLE   :=T#40S
SP_INT  :=DB1.DBD4
PV_IN   :=DB1.DBD0
PV_PER  :=
MAN     :=
GAIN    :=4.500000e+000
TI      :=T#40S
TD      :=T#40S
TM_LAC  :=T#20S
DEADB_W :=0.000000e+000
LMN_HLM :=1.000000e+002
LMN_LLM :=0.000000e+000
PV_FAC  :=1.000000e+000
PV_OFF  :=0.000000e+000
LMN_FAC :=
LMN_OFF :=
I_ITLVAL:=0.000000e+000
DISV    :=
LMN     :=DB1.DBD8
LMN_PER :=
QLMN_HLM:=
QLMN_LLM:=
LMN_P   :=
LMN_I   :=
LMN_D   :=
PV      :=
ER      :=
_001: NOP 0

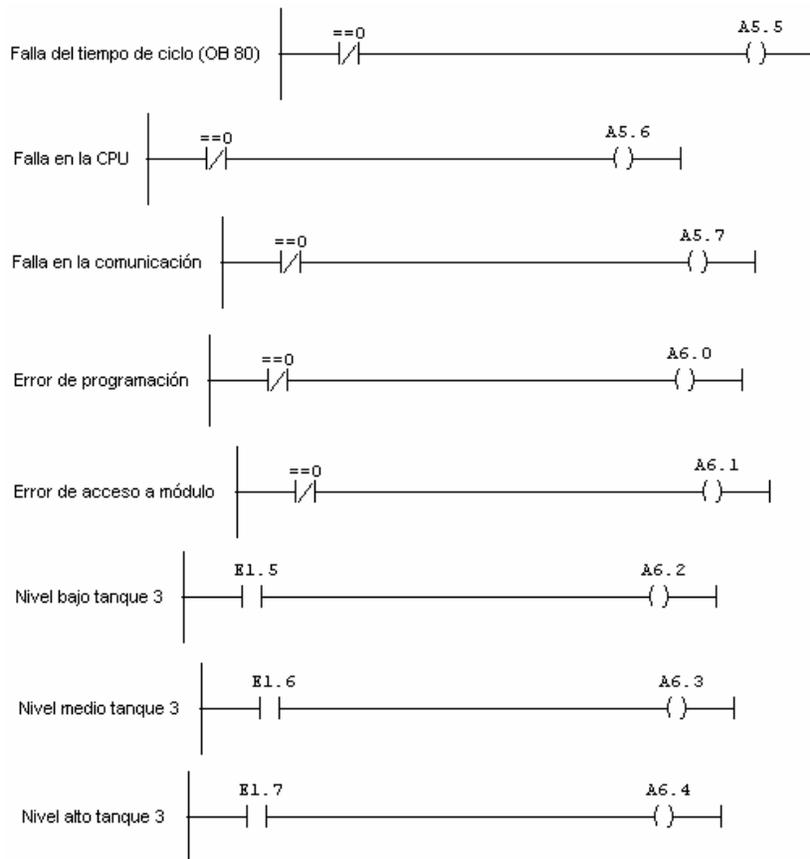
```

Bloque de regulación FB 41 para tanque 3

```

U(
O   M   100.3
O   A   5.2
)
S   M   0.1
U(
O   M   100.2
O   E   1.3
O   A   6.4
)
R   M   0.1
U   M   0.1
=   L   20.0
UN  L   20.0
=   A   5.3
UN  L   20.0
L   SST#3S
SI  T   2
NOP 0
NOP 0
NOP 0
U   T   2
=   A   5.4
U   E   1.4
=   L   20.1
BLD 103
U   ==0
=   L   20.2
BLD 103
UN  ==0
=   L   20.4
BLD 103
U   ==0
=   L   20.5
BLD 103
U   ==0
=   L   20.7
BLD 103
UN  ==0
=   L   21.0
BLD 103
U   L   20.0
SPENB _002
CALL "CONT_C" , DB35
COM_RST :=L20.1
MAN_ON  :=L20.2
PVPER_ON:=
P_SEL   :=L20.4
I_SEL   :=L20.5
INT_HOLD:=
I_ITL_ON:=L20.7
D_SEL   :=L21.0
CYCLE   :=T#40S
SP_INT  :=DB2.DBD4
PV_IN   :=DB2.DBD0
PV_PER  :=
MAN     :=
GAIN    :=4.500000e+000
TI      :=T#40S
TD      :=T#40S
TM_LAC  :=T#20S
DEADB_W :=0.000000e+000
LMN_HLM :=1.000000e+002
LMN_LLM :=0.000000e+000
PV_FAC  :=1.000000e+000
PV_OFF  :=0.000000e+000
LMN_FAC :=
LMN_OFF :=
I_ITLVAL:=0.000000e+000
DISV    :=
LMN     :=DB2.DBD8
LMN_PER :=
QLMN_HLM:=
QLMN_LLM:=
LMN_P   :=
LMN_I   :=
LMN_D   :=
PV      :=
ER      :=
_002: NOP 0

```



## Programa en AWL

Conducto 1    U    E    0.0  
               =    A    4.0

Conducto 2    U    E    0.1  
               =    A    4.1

Conducto 3    U    E    0.2  
               =    A    4.2

Nivel bajo tanque 1    U    E    0.3  
                           =    A    4.3

Nivel medio tanque 1    U    E    0.4  
                           =    A    4.4

Nivel alto tanque 1    U    E    0.5  
                           =    A    4.5

Nivel bajo tanque 2    U    E    1.0  
                           =    A    5.0

Nivel medio tanque 2    U    E    1.1  
                           =    A    5.1

Nivel alto tanque 2    U    E    1.2  
                           =    A    5.2

Bloque de regulación FB 41 para tanque 2

```

U(
O      M      100.1
O      A      4.5
)
S      M      0.0
U(
O      M      100.0
O      E      0.6
O      A      5.2
)
R      M      0.0
U      M      0.0
=      L      20.0
UN     L      20.0
=      A      4.6
UN     L      20.0
L      SST#3S
SI     T      1
NOP    O
NOP    O
NOP    O
U      T      1
=      A      4.7
U      E      0.7
=      L      20.1
BLD   103
U      M      10.0
=      L      20.2
BLD   103
UN     ==0
=      L      20.4
BLD   103
UN     ==0
=      L      20.5
BLD   103
UN     ==0
=      L      20.7
BLD   103
UN     ==0
=      L      21.0
BLD   103
U      L      20.0
SPENB _001
CALL  "CONT_C" , DB50
COM_RST :=L20.1
MAN_ON  :=L20.2
PVPER_ON:=
P_SEL   :=L20.4
I_SEL   :=L20.5
INT_HOLD:=
I_ITL_ON:=L20.7
D_SEL   :=L21.0
CYCLE   :=T#40S
SP_INT  :=DB1.DBD4
PV_IN   :=DB1.DBD0
PV_PER  :=
MAN     :=
GAIN    :=4.500000e+000
TI      :=T#40S
TD      :=T#40S
TM_LAG  :=T#20S
DEADB_W :=0.000000e+000
LMN_HLM :=1.000000e+002
LMN_LLM :=0.000000e+000
PV_FAC  :=1.000000e+000
PV_OFF  :=0.000000e+000
LMN_FAC :=
LMN_OFF :=
I_ITLVAL:=0.000000e+000
DISV    :=
LMN     :=DB1.DBD8
LMN_PER :=
QLMN_HLM:=
QLMN_LLM:=
LMN_P   :=
LMN_I   :=
LMN_D   :=
PV      :=
ER      :=
_001: NOP 0

```

Bloque de regulación FB 41 para tanque 3

```

U(
O      M      100.3
O      A      5.2
)
S      M      0.1
U(
O      M      100.2
O      E      1.3
O      A      6.4
)
R      M      0.1
U      M      0.1
=      L      20.0
UN     L      20.0
=      A      5.3
UN     L      20.0
L      SST#3S
SI     T      2
NOP    O
NOP    O
NOP    O
U      T      2
=      A      5.4
U      E      1.4
=      L      20.1
BLD   103
U      ==0
=      L      20.2
BLD   103
UN     ==0
=      L      20.4
BLD   103
U      ==0
=      L      20.5
BLD   103
U      ==0
=      L      20.7
BLD   103
UN     ==0
=      L      21.0
BLD   103
U      L      20.0
SPENB _002
CALL  "CONT_C" , DB35
COM_RST :=L20.1
MAN_ON  :=L20.2
PVPER_ON:=
P_SEL   :=L20.4
I_SEL   :=L20.5
INT_HOLD:=
I_ITL_ON:=L20.7
D_SEL   :=L21.0
CYCLE   :=T#40S
SP_INT  :=DB2.DBD4
PV_IN   :=DB2.DBD0
PV_PER  :=
MAN     :=
GAIN    :=4.500000e+000
TI      :=T#40S
TD      :=T#40S
TM_LAG  :=T#20S
DEADB_W :=0.000000e+000
LMN_HLM :=1.000000e+002
LMN_LLM :=0.000000e+000
PV_FAC  :=1.000000e+000
PV_OFF  :=0.000000e+000
LMN_FAC :=
LMN_OFF :=
I_ITLVAL:=0.000000e+000
DISV    :=
LMN     :=DB2.DBD8
LMN_PER :=
QLMN_HLM:=
QLMN_LLM:=
LMN_P   :=
LMN_I   :=
LMN_D   :=
PV      :=
ER      :=
_002: NOP 0

```

Falla del tiempo de ciclo (OB 80)	UN =	==0 A	5.5
Falla en la CPU	UN =	==0 A	5.6
Falla en la comunicación	UN =	==0 A	5.7
Error de programación	UN =	==0 A	6.0
Error de acceso a módulo	UN =	==0 A	6.1
Nivel bajo tanque 3	U =	E A	1.5 6.2
Nivel medio tanque 3	U =	E A	1.6 6.3
Nivel alto tanque 3	U =	E A	1.7 6.4

## Parámetros de entrada de módulo FB 41

<u>Parámetro</u>	<u>Tipo de datos</u>	<u>Valores posibles</u>	<u>Por defecto</u>	<u>Descripción</u>
COM_RST	BOOL		FALSE	COMPLETE RESTART / Rearranque completo El bloque tiene una rutina de inicialización que se procesa cuando está activada la entrada "COM_RST".
MAN_ON	BOOL		TRUE	MANUAL VALUE ON / Conectar a modo manual Si está activada la entrada "Conectar a modo manual", está interrumpido el lazo de regulación. Como valor manipulado se fuerza un valor manual.
PVPER_ON	BOOL		FALSE	PROCESS VARIABLE PERIPHERY ON / Conectar valor real de periferia Si debe leerse el valor real de la periferia, debe interconectarse la entrada PV_PER con la periferia y activarse la entrada "Conectar valor real de periferia".
P_SEL	BOOL		TRUE	PROPORTIONAL ACTION ON / Conectar acción P En el algoritmo PID pueden conectarse y desconectarse individualmente las acciones PID. La acción P está conectada si está activada la entrada "Conectar acción P".
I_SEL	BOOL		TRUE	INTEGRAL ACTION ON / Conectar acción I En el algoritmo PID pueden conectarse y desconectarse individualmente las acciones PID. La acción I está conectada si está activada la entrada "Conectar acción I".
INT_HOLD	BOOL		FALSE	INTEGRAL ACTION HOLD / Congelar acción I La salida del integrador puede congelarse. Para ello se ha de activar la entrada "Congelar acción I".
I_ITL_ON	BOOL		FALSE	INITIALIZATION OF THE INTEGRAL ACTION / Inicializar acción I La salida del integrador puede inicializarse a la entrada I_ITLVAL. Para ello se ha de activar la entrada "Inicializar acción I".
D_SEL	BOOL		FALSE	DERIVATIVE ACTION ON / Conectar acción D En el algoritmo PID pueden conectarse y desconectarse individualmente las acciones PID. La acción D está conectada si está activada la entrada "Conectar acción D".
CYCLE	TIME	>= 1ms	T#1s	SAMPLE TIME / Tiempo de muestreo El tiempo entre las llamadas del bloque debe ser constante. La entrada "Tiempo de muestreo" indica el tiempo entre las llamadas del bloque.
SP_INT	REAL	-100.0...100.0 (%) o bien magnitud física 1)	0.0	INTERNAL SETPOINT / Consigna interna La entrada "Consigna interna" sirve para ajustar un valor de consigna.
PV_IN	REAL	-100.0...100.0 (%) o bien magnitud física 1)	0.0	PROCESS VARIABLE IN / Entrada de valor real En la entrada "Entrada de valor real" puede parametrizarse un valor de puesta en servicio, o interconectarse un valor real externo en formato en coma flotante.
PV_PER	WORD		W#16#00	PROCESS VARIABLE PERIPHERY / Valor real de periferia El valor real en formato de periferia se interconecta con el regulador en la entrada "Valor real de periferia".
MAN	REAL	-100.0...100.0 (%) o bien magnitud física 2)	0.0	MANUAL VALUE / Valor manual La entrada "Valor manual" sirve para establecer un valor manual mediante función de manejo/visualización (interface hombre máquina)..
GAIN	REAL		2.0	PROPORTIONAL GAIN / Ganancia proporcional La entrada "Ganancia proporcional" indica la ganancia del regulador.

TI	TIME	>= CYCLE	T#20s	RESET TIME / Tiempo de acción integral La entrada "Tiempo de acción integral" determina el comportamiento temporal del integrador.
TD	TIME	>= CYCLE	T#10s	DERIVATIVE TIME / Tiempo de diferenciación (acción derivativa) La entrada "Tiempo de diferenciación" determina el comportamiento temporal del diferenciador.
TM_LAG	TIME	>= CYCLE/2	T#2s	TIME LAG OF THE DERIVATIVE ACTION / Tiempo de retardo de la acción D El algoritmo de la acción D contiene un retardo que puede parametrizarse en la entrada "Tiempo de retardo de la acción D".
DEADB_W	REAL	>= 0.0 (%) o bien magnitud física 1)	0.0	DEAD BAND WIDTH / Ancho de zona muerta La diferencia de regulación se conduce por una zona muerta. La entrada "Ancho de zona muerta" determina el tamaño de la zona muerta.
LMN_HLM	REAL	LMN_LLM ...100.0 (%) o bien magnitud física 2)	100.0	MANIPULATED VALUE HIGH LIMIT / Límite superior del valor manipulado El valor manipulado tiene siempre un límite superior y uno inferior. La entrada "Límite superior del valor manipulado" indica la limitación superior.
LMN_LLM	REAL	-100.0... LMN_HLM (%) o bien magnitud física 2)	0.0	MANIPULATED VALUE LOW LIMIT / Límite inferior del valor manipulado El valor manipulado tiene siempre un límite superior y uno inferior. La entrada "Límite inferior del valor manipulado" indica la limitación inferior.
PV_FAC	REAL		1.0	PROCESS VARIABLE FACTOR / Factor de valor real La entrada "Factor de valor real" se multiplica por el valor real. La entrada sirve para la adaptación del margen de valor real.
PV_OFF	REAL		0.0	PROCESS VARIABLE OFFSET / Offset del valor real La entrada "Offset del valor real" se suma con el valor real. La entrada sirve para la adaptación del margen de valor real.
LMN_FAC	REAL		1.0	MANIPULATED VALUE FACTOR / Factor del valor manipulado La entrada "Factor del valor manipulado" se multiplica por el valor manipulado. La entrada sirve para la adaptación del margen de valor manipulado.
LMN_OFF	REAL		0.0	MANIPULATED VALUE OFFSET / Offset del valor manipulado La entrada "Offset del valor manipulado" se suma al valor manipulado. La entrada sirve para la adaptación del margen de valor manipulado.
I_ITLVAL	REAL	-100.0...100.0 (%) o bien magnitud física 2)	0.0	INITIALIZATION VALUE OF THE INTEGRAL ACTION / Valor de inicialización de la acción I La salida del integrador puede ponerse en la salida I_ITL_ON. En la entrada "Valor de inicialización de la acción I" está el valor de inicialización.
DISV	REAL	-100.0...100.0 (%) o bien magnitud física 2)	0.0	DISTURBANCE VARIABLE / Magnitud perturbadora Para control anticipativo de la magnitud perturbadora, ésta se conecta en la entrada "Magnitud perturbadora".

1) Parámetros en la rama de valor de consigna, rama de valor real, con las mismas unidades

2) Parámetros en la rama de valor manipulado, con las mismas unidades

## Parámetros de salida de módulo FB 41

<u>Parámetro</u>	<u>Tipo de datos</u>	<u>Valores posibles</u>	<u>Por defecto</u>	<u>Descripción</u>
LMN	REAL		0.0	MANIPULATED VALUE / Valor manipulado En la salida "Valor manipulado" se saca en formato en coma flotante el valor manipulado que actúa efectivamente.
LMN_PER	WORD		W#16#0000	MANIPULATED VALUE PERIPHERY / Valor manipulado periferia El valor manipulado en formato de periferia se interconecta con el regulador en la salida "Valor manipulado periferia".
QLMN_HLM	BOOL		FALSE	HIGH LIMIT OF MANIPULATED VALUE REACHED / Alcanzado el límite superior del valor manipulado El valor manipulado tiene siempre un límite superior y un límite inferior. La salida "Alcanzada limitación superior del valor manipulado" indica la superación de la limitación superior.
QLMN_LLM	BOOL		FALSE	LOW LIMIT OF MANIPULATED VALUE REACHED / Alcanzado el límite inferior del valor manipulado El valor manipulado tiene siempre un límite superior y un límite inferior. La salida "Alcanzado el límite inferior del valor manipulado" indica la superación de la limitación inferior.
LMN_P	REAL		0.0	PROPORTIONALITY COMPONENT / Acción P La salida "Acción P" contiene la componente proporcional de la magnitud manipulada.
LMN_I	REAL		0.0	INTEGRAL COMPONENT / Acción I La salida "Acción I" contiene la componente integral de la magnitud manipulada.
LMN_D	REAL		0.0	DERIVATIVE COMPONENT / Acción D La salida "Acción D" contiene la componente diferencial de la magnitud manipulada.
PV	REAL		0.0	PROCESS VARIABLE / Valor real Por la salida "Valor real" se emite el valor real que actúa efectivamente.
ER	REAL		0.0	ERROR SIGNAL / Error de regulación Por la salida "Error de regulación" se emite la diferencia o error de regulación que actúa efectivamente.

## **4.5 CONCLUSIONES PARCIALES.**

### **➤ De la arquitectura del modelo de control propuesto.**

De lo anteriormente expuesto en este capítulo podemos resumir que el modelo de automatización propuesto (arquitectura de control basado en CitectScada- PLC) y el cual es de amplia utilización en los diferentes procesos de control de las industrias modernas, permite cierta flexibilidad en la toma y procesamiento de la información, además posibilita la realización de las tareas de control en tiempo real o en análisis posteriores (a fin de analizar los posibles errores), posee gran capacidad de almacenamiento, rápido acceso a la información y toma de decisión, también provee de toda la información que se genera en el proceso productivo, los que se muestran en pantalla al usuario para poder ser analizados y corregir las desviaciones.

## Capítulo V: Valoración técnica económica.

### 5.1 GENERALIDADES.

Uno de los puntos más preocupantes hoy en día a la hora de seleccionar la instrumentación de control de una planta o proceso es su encarecido costo. Esto conlleva a seleccionar los Instrumentos de control con mayor fiabilidad y reducidos costes tanto del equipamiento como de su mantenimiento periódico. Una vez ajustados precio y calidad se debe garantizar el mayor tiempo de explotación del mismo para obtener las mayores ganancias posibles, esto se debe a que realizando un cambio de tecnología anual los costos de producción se elevarían a raíz de este.

## 5.2 COSTOS DE LA INSTALACIÓN.

Cuando se quiere automatizar un proceso industrial de gran magnitud se debe conocer los costos de producción así como el de los autómatas y sus transductores para realizar posteriormente la valoración técnica económica.

<b>Costo de la instalación.</b>			
Equipos y personal	Cantida d	Precio en Franco Canadiense	Precio en CUC
PC	1	1200	1300
CPU del PLC	1	4885	4900
Tarjeta profibus	1	2000	2030
Tarjeta DI	1	263	279
Tarjeta DO	3	401	420
Tarjeta AI	1	810	830
Tarjeta AO	1	710	730
Rail	1	41	50
FM 355C		1344	1380
Conector de 20(PLC)		32	38
Conector de 40(PLC)		51	59
Conector de Bus		69	75
PS 24		213	240
Cable de señal analógico. (PLC)	1(m)		0.23
Cable de señal para tarjeta profibus.	1(m)		2.9305
Cable eléct. # 16	1(m)	-	0.39
Interruptor automático.	4	-	53.04
Sensor de nivel	3	-	900
Licencia de programa STEP 7 V5.4		-	-
Licencia de programa CITECT (llave).	1		2000
Ing. de montaje automático.	2		71
Obrero operador	1	-	50
Obrero supervisor	1	-	52
Tanques	3	-	-
Botones pulsadores			6.65
Electro válvulas			
Motores de inducción 11KW	2	-	4410.08

Variadores de velocidad 15HP	2	-	5374.78
UPS para PLC	1		100
Total	-	12019	25352.10

Tabla 10. Precio de la instrumentación.

Es necesario cuando se quiere implementar el control automático de proceso que todos los componentes del sistema estén calculados lo mejor posible para que a la hora de realizar el montaje no existan complicaciones por parte del equipo o sea que esté fuera de rango o mal calibrado puesto que si se incurre en uno de estos las pérdidas podrían ser alentadoras.

### **5.3 IMPACTO SOCIAL.**

Este proyecto más que un gran impacto económico tiene como objetivo proyectarse en el aprendizaje de los futuros ingenieros así como ampliar sus conocimientos en el mundo del control automático como aplicación de mayor peso en la industria en desarrollo, además de comprender cuán necesario es mejorar tiempos, efectividad y la sustitución de la mano de obra en la regulación de un proceso a escala mayor.

## **CONCLUSIONES**

- 1- En el departamento de los especialistas en automática de la Empresa “Comandante Pedro Sotto Alba MOA NÍQUEL.SA” se realizaron las pruebas físicas verificándose que los datos mostrados en la tesis (programa CITECT, programa STEP 7, Bloque declarado para la regulación) son verídicos y compilan en sus respectivos programas.
- 2- Se realizó una prueba verificándose por qué el regulador continuo es más eficiente que los demás contenidos en el programa Step 7.
- 3- Se pudo llegar a la conclusión de que el proceso que esta siendo supervisado y monitoreado en tiempo real facilita la corrección del error por parte del hombre y garantiza un mejor estudio del comportamiento del mismo así como las tendencias históricas.

## **RECOMENDACIONES**

- 1- Que el presente trabajo constituya un material didáctico de consulta donde los estudiantes de la carrera de Ingeniería Eléctrica del ISMM y profesores interesados en el tema puedan desarrollar y ampliar sus conocimientos acerca del mundo del control automático.
- 2- Este proyecto da paso a posibles investigaciones científicas por parte del personal universitario.
- 3- Se recomienda no variar el código de programa puesto que este se diseñó en (Ladder logic) lógica de escalera.

## Bibliografía:

- [1] Ogata, Katsuhiko: *Ingeniería de control moderna*, Tomo 1. Editorial Pueblo y Educación. Ciudad de la Habana
- [2] Kuo, Benjamín C: *Automatic Control Systems*, Seventh Edition, Prentice Hall, 1995.
- [3] Aguado Behar, Alberto: *Temas de Identificación y Control Adaptable*, Instituto de Cibernética, Matemática y Física, 2000.
- [4] Smith, Carlos A; Corripio, Armando B: *Principles and Practice of Automatic Process Control*, 1997
- [5] Costa, Angel: *Principios Fundamentales de Accionamiento Eléctrico*. Editorial Pueblo y Educación, 1989.

## Anexos

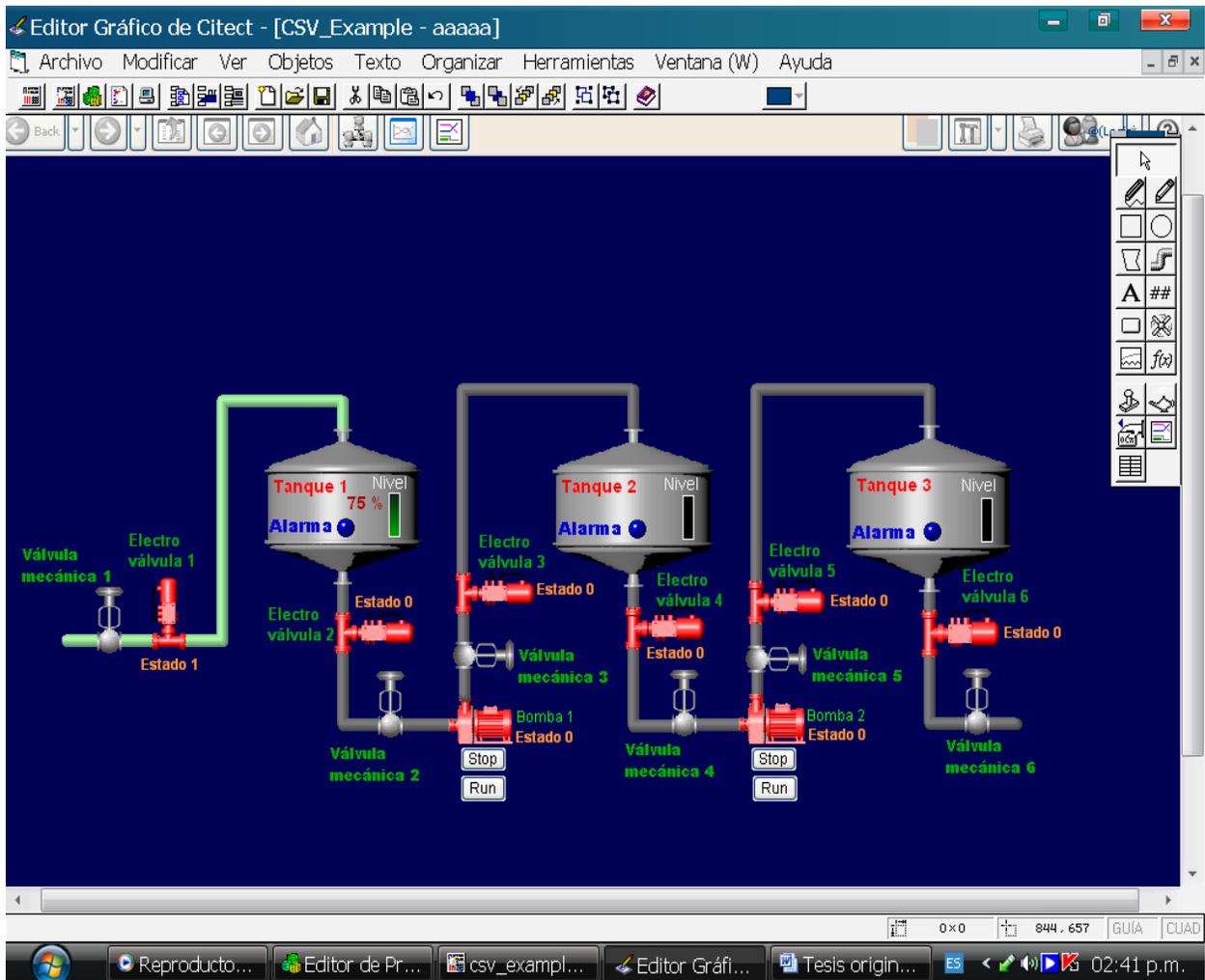


Foto 1. Simulación del proceso en tiempo real.

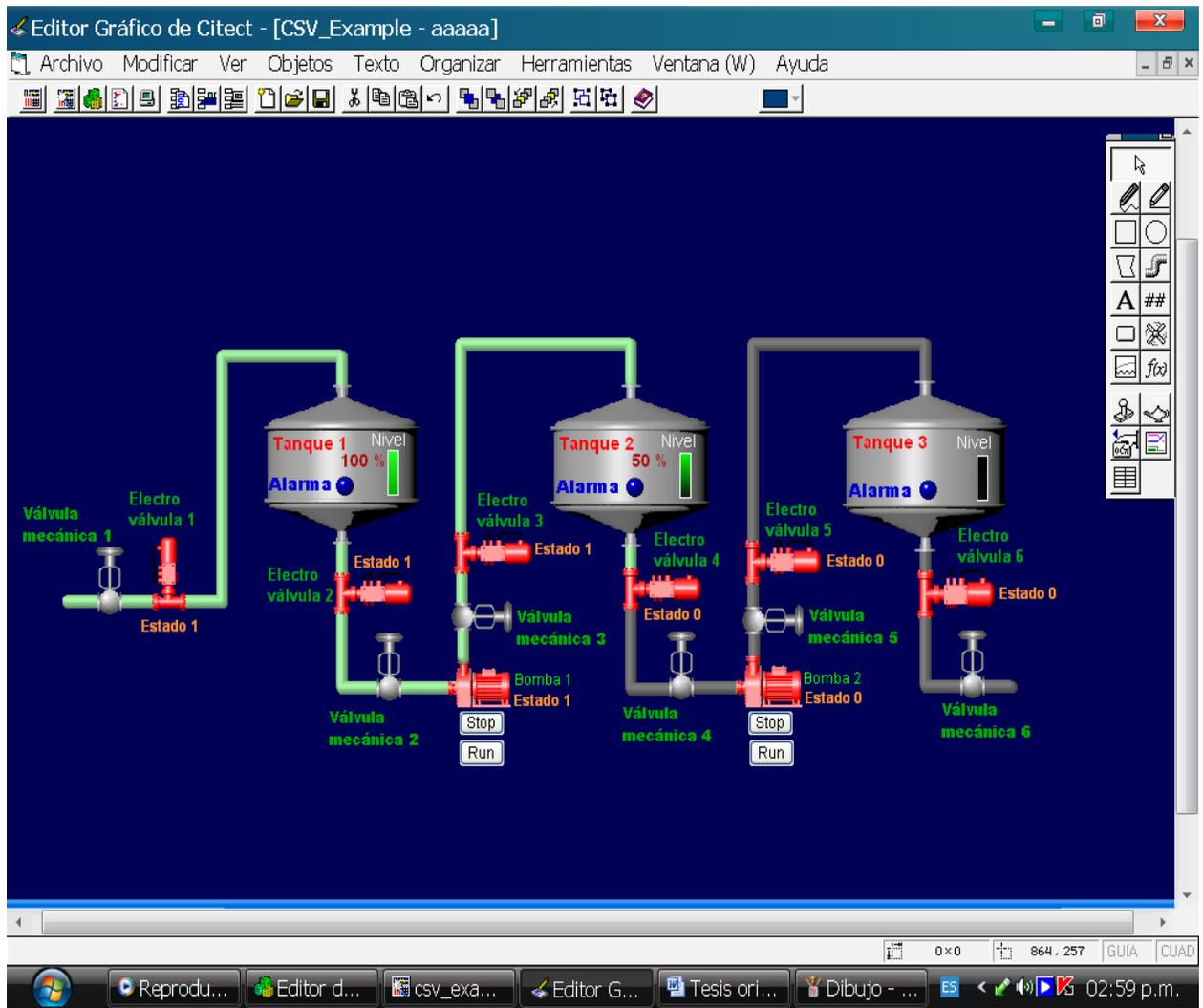


Foto 2. Simulación del proceso en tiempo real.

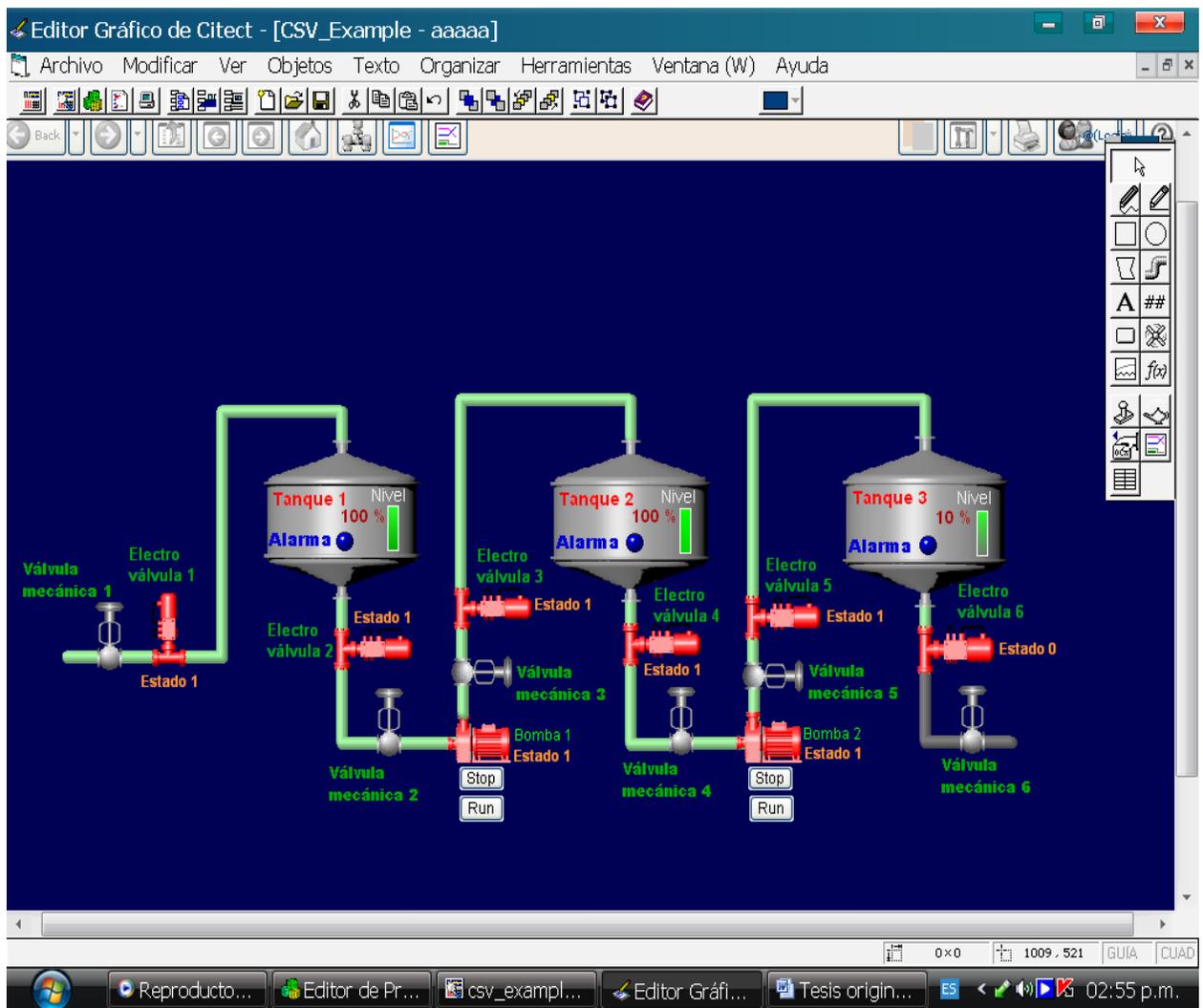


Foto 3. Simulación del proceso en tiempo real.



Foto 4. Autómatas de Siemens.