

**UNIVERSIDAD POLITÉCNICA DE CARTAGENA**  
**Escuela Técnica Superior de Ingeniería Industrial**



**Calibración, Control y Diseño SCADA de un  
robot paralelo neumático con el autómeta  
S7-300.**

**Titulación:** I.T.I Electrónica.  
**Intensificación:** Automática.  
**Alumno/a:** Javier Giménez García.  
**Director:** Miguel Almonacid Kroeger.  
**Coodirector:** Julio José Ibarrola Lacalle.

Cartagena 19 de Junio de 2008



# Índice



---

---

Capítulo 1 Objetivos .....	8
1.1 Introducción .....	9
1.2 Objetivos de este proyecto .....	10
1.3 Antecedentes históricos.....	10
1.3.1 Robots de estructura paralela .....	10
1.3.2 Historia de los autómetas programables .....	11
Capítulo 2 Descripción de los componentes de la planta.....	13
2.1 Introducción .....	14
2.2 Hardware.....	15
2.2.1 Sistema de automatización S7-300 .....	15
2.2.2 Cilindros normalizados de doble efecto.....	16
2.2.3 El convertidor de valor medido.....	17
2.2.4 Válvulas de control proporcional.....	21
2.2.5 Válvula reguladora de presión .....	23
2.2.6 Pantalla táctil TP177B Color PN/DP.....	24
2.2.7 Otros elementos utilizados .....	25
2.3 Software .....	25
Capítulo 3 Calibración del robot.....	30
3.1 Introducción .....	31
3.2 Calibración de un cilindro neumático .....	32
3.2.1 Tabla de símbolos del Programa “Control de 1 cilindro” .....	33
3.2.2 GRAFCET nivel 1: Calibrado manual 1 cilindro .....	35
3.2.3 GRAFCET nivel 2: Calibrado manual 1 cilindro .....	36
3.2.4 GRAFCET nivel 2: Calibrado automático 1 cilindro .....	37
3.2.5 Ecuaciones de activación desactivación de 1 cilindro .....	38
3.3 Calibración de dos cilindros neumáticos .....	39
3.3.1 Tabla de símbolos Programa “2cilindros” .....	39
3.3.2 GRAFCET nivel 1: Calibrado manual 2 cilindros.....	41
3.3.3 GRAFCET nivel 2: Calibrado manual 2 cilindros.....	42
3.3.4 Grafcet de nivel 1 Calibrado automático: .....	43
3.3.5 Grafcet nivel 2: Calibrado automático .....	44
3.3.6 Ecuaciones activación desactivación “2 Cilindros” .....	45
3.4 Cinemática directa.....	46
3.4.1 Tabla de símbolos del Programa “Cinemática_Directa” .....	46
3.4.2 GRAFCET nivel 2: Calibrado manual cinemática directa.....	49
3.4.3 Ecuaciones activación desactivación “Cinemática Directa” .....	50
3.5 Cinemática Inversa.....	51
3.5.1 Tabla de símbolos del Programa “Cinemática Inversa” .....	51
3.5.2 GRAFCET nivel 2: Calibrado manual cinemática inversa.....	53
3.5.3 Ecuaciones de activación desactivación “Cinemática Inversa” .....	54
3.6 Funciones utilizadas.....	55
3.6.1 Desescalar valores" (UNSCALE): FC106.....	55
3.6.2 Escalar valores (SCALE): FC105.....	56
3.6.3 S_VIMP: Parametrizar y arrancar temporizador como impulso prolongado .....	59

---

---



---

---

3.6.4 S_Ever_z: Parametrizar y arrancar temporizador .....	61
3.6.5 MOVE Asignar un valor.....	62
Capítulo 4 Programación del control de posición.....	64
4.1 Introducción.....	65
4.2 Sintonía mediante Ziegler-Nichols (1 cilindro).....	65
4.2.1 Primer método (mediante lazo abierto).....	65
4.2.2 Segundo método (mediante lazo cerrado).....	66
4.3 Sintonía mediante HARRIOT.....	68
4.4 Sintonía mediante Ensayo-Error (1 cilindro).....	71
4.4.1 Control proporcional (P).....	71
4.4.2 Control proporcional derivativo (PD).....	72
4.4.3 Control proporcional integral derivativo (PID).....	72
4.5 Sintonía mediante Ensayo-Error (2 cilindros).....	72
4.5.1 Control proporcional (P).....	72
4.5.2 Control proporcional derivativo (PD).....	72
4.5.3 Control Proporcional, Integral, Derivativo (PID).....	72
4.6 Funciones utilizadas.....	72
4.6.1 SFB 41/FB 41 "CONT_C".....	72
4.7 Cinemática.....	72
4.7.1 Cinemática Directa.....	72
4.7.2 Cinemática Inversa.....	72
Capítulo 5 Diseño SCADA.....	72
5.1 Comunicación.....	72
5.2 Abrir aplicación diseñada en el sistema SCADA.....	72
5.3 Ejecutar aplicación diseñada en el sistema SCADA.....	72
5.4 Control SCADA de 1 cilindro neumático.....	72
5.4.1 Descripción de la pantalla del "Calibrado automático".....	72
5.4.2 Descripción de la pantalla de "Calibrado Manual".....	72
5.4.3 Descripción de la pantalla "Control PID".....	72
5.4.4 Parametrizador PID.....	72
5.4.5 Pulsador "Stop".....	72
5.5 Control SCADA de 2 cilindros neumáticos.....	72
5.5.1 Descripción de la pantalla "Calibrado automático".....	72
5.5.2 Descripción de la pantalla "Calibrado Manual".....	72
5.5.3 Descripción de la pantalla "Control PID".....	72
5.6 Control SCADA Cinemática Directa.....	72
5.6.1 Descripción de la pantalla "Sintonizar PID1".....	72
5.6.2 Descripción de la pantalla "Sintonizar PID2".....	72
5.6.3 Descripción de la pantalla "Cinemática Directa".....	72
5.7 Control SCADA "Cinemática Inversa".....	72
5.7.1 Descripción de la pantalla "Cinemática Inversa".....	72
5.8 Transferencia a pantalla táctil TP177B color PN/DP.....	72
5.8.1 Cambiar conexión y configuración MPI→PPI.....	72
5.8.2 Transferir programa SCADA a pantalla táctil.....	72
5.8.3 Comunicación del PLC con la pantalla táctil.....	72

---

---



Capítulo 6 Conclusiones y desarrollos futuros .....	72
6.1 Conclusiones.....	72
6.2 Desarrollos futuros.....	72
6.3 Ventajas e inconvenientes entre la utilización de la tarjeta de adquisición de datos y el autómata S7-300 de Siemens. ....	72
Bibliografía .....	72
Anexos .....	721



# Indice de figuras



## Capítulo 2

Figura 2. 1: a) Válvula proporcional, Convertidor DADE; b) Cilindros de doble efecto; c) Plc, d) Fuente de alimentación ; e) Pantalla táctil .....	15
Figura 2. 2 S7-300 CPU 314C-2DP.....	15
Figura 2. 3 Cilindro normalizado de doble efecto. ....	16
Figura 2. 4 Cable S1 (Cilindro→ convertidor). ....	17
Figura 2. 5 Conexión PLC, Convertidor, Cilindro.....	17
Figura 2. 6 Convertidor del valor medido.....	17
Figura 2. 7 Cable S2 (Convertidor→PLC). ....	18
Figura 2. 8 Resumen. ....	20
Figura 2. 9 Diagnósis. ....	21
Figura 2. 10 Válvulas MPYE-5-1/8-LF-010-B.....	21
Figura 2. 11 Conexión de las válvulas proporcionales. ....	22
Figura 2. 12 Eliminación de fallos.....	22
Figura 2. 13 Válvula reguladora de presión.....	23
Figura 2. 14 Vistas frontal y lateral de la pantalla. ....	24
Figura 2. 15 Vista inferior.....	24
Figura 2. 16 Vista posterior.....	25
Figura 2. 17 Conector.    Figura 2. 18 Silenciador    Figura 2. 19 Conector en T.....	25

## Capítulo 3

Figura 3. 1 Un cilindro neumático. ....	31
Figura 3. 2 Dos cilindros unidos.....	31
Figura 3. 3 Bloque Unscale (FC 106). ....	55
Figura 3. 4 Bloque Scale (FC 105). ....	56
Figura 3. 5 S_VIMP.....	59
Figura 3. 6 Diagrama de temporización.....	60
Figura 3. 7 S_Everz.....	61
Figura 3. 8 Diagrama de temporización 1.....	61
Figura 3. 9 Move.....	62

## Capítulo 4

Figura 4. 1 Sintonía mediante Ziegler-Nichols.....	65
Figura 4. 2 Tipo de controlador. 1 .....	66
Figura 4. 3 Prueba 1.....	66
Figura 4. 4 Tipo de controlador 2.            Figura 4. 5 Oscilaciones sostenidas. ....	67
Figura 4. 6 La salida siempre es acotada para cualquier entrada acotada. No podemos utilizar el ...	67
Figura 4. 7 Planta controlada mediante acción proporcional.....	68
Figura 4. 8 Sintonía mediante HARRIOT (Oscilaciones sostenidas).1.....	68
Figura 4. 9 Sintonía mediante el método de HARRIOT (Oscilaciones sostenidas).2.....	68
Figura 4. 10: a) $K_p=4.3$ ; b) $K_p=4,3$ , $T_i=0,55s$ ; c) $K_p=3$ ; $T_i=100$ ; d) $K_p=4$ , $T_i=100$ ;	70



---

---

Figura 4. 11 Un cilindro.....	71
Figura 4. 12 a) $K_p=1$ ; b) $K_p=1$ ; c) $K_p=1$ ; d) $K_p=2$ ; e) $K_p=2$ ; f) $K_p=2$ ; g) $K_p=3$ ; h) $K_p=3$ ;.....	72
Figura 4. 13 a) $K_p=2$ , $T_d=10s$ ; b) $K_p=2$ , $T_d=10s$ ; c) $K_p=3$ , $T_d=10s$ ; d) $K_p=3$ , $T_d=10s$ ;.....	72
Figura 4. 14 a) $K_p=3$ , $T_d=10$ , $T_i=40$ ; b) $K_p=2$ , $T_d=10$ , $T_i=40$ ;.....	72
Figura 4. 15 Dos cilindros unidos.....	72
Figura 4. 16 a) Cilindro n°1 $K_p=1$ ; b) Cilindro n°2 $K_p=1$ ; c) Cilindro n°1 $K_p=2$ .....	72
Figura 4. 17 a) Cilindro n°1 $K_p=1$ , $T_d=2s$ ; b) Cilindro n°2 $K_p=1$ , $T_d=2s$ ; .....	72
Figura 4. 18 a) Cilindro n°1 $K_p=1$ , $T_d=4s$ , $T_i=50$ ; b) Cilindro n°2 $K_p=1$ , $T_d=4s$ , $T_i=100$ ; .....	72
Figura 4. 19 SFB 41 “CONT_C”.....	72

## Capítulo 5

Figura 5. 1 Abrir proyecto existente.1.....	72
Figura 5. 2 Pantalla principal (1 cilindro).....	72
Figura 5. 3 Pantalla de calibrado automático: a) Pantalla 1 ; b) Pantalla 2.....	72
Figura 5. 4 Pantallas de calibrado manual: a) Pantalla 1 ; b) Pantalla 2 ; c) Pantalla 3;.....	72
Figura 5. 5 Pantalla de control PID. (1 cilindro).....	72
Figura 5. 6 Abrir parametrizador.....	72
Figura 5. 7 Abrir DB100.....	72
Figura 5. 8 Ventana para Introducir parámetros PID.....	72
Figura 5. 9 Pantalla principal. (2 cilindros).....	72
Figura 5. 10 Calibrado automático de 2 cilindros): a) Pantalla1 ; b)Pantalla 2.....	72
Figura 5. 11 Calibrado manual: a) Pantalla 1 ; b) Pantalla 2 ; c) Pantalla3 ; d) Pantalla 4.....	72
Figura 5. 12 Control PID. (2 cilindros).....	72
Figura 5. 13 Pantalla principal. (Cinemática directa).....	72
Figura 5. 14 Sintonizar PID1. (Cinemática directa).....	72
Figura 5. 15 Sintonizar PID2. (Cinemática directa).....	72
Figura 5. 16 Cinemática directa. (Cinemática directa).....	72
Figura 5. 17 Pantalla principal. (Cinemática inversa).....	72
Figura 5. 18 Pantalla “Cinemática inversa”.....	72
Figura 5. 19 Cambiar configuración_1.....	72
Figura 5. 20 Cambiar configuración.....	72
Figura 5. 21 Cambiar configuración_4.....	72
Figura 5. 22 Cambiar panel de operador: a) Menú “Proyecto” ; b) Cambiar panel de operador.....	72
Figura 5. 23 Transferir.....	72





# Capítulo 1

## *Objetivos*



## 1.1 Introducción.

En este proyecto se ha realizado la puesta en marcha, calibración y el control de un robot de estructura paralela de actuación neumática mediante el autómeta S7-300 de SIEMENS.

Antes de comenzar con el desarrollo conviene señalar que este proyecto fin de carrera se desarrolla como continuación de otros dos proyectos previos desarrollados sobre el mismo robot. El primero titulado "Modelado, simulación y diseño de un robot paralelo de 2GDL con actuación neumática" desarrollado por D. Miguel Torregrosa Puig estudiante de la Universidad Politécnica de Cartagena y cuyo contenido es el siguiente:

Modelado:

- Simulación gráfica del robot.
- Planteamiento y resolución de la cinemática directa.
- Planteamiento y resolución de la cinemática inversa.
- Representación de los resultados obtenidos en el cálculo del espacio de trabajo.

Simulación:

- Estudiar las características proporcionadas por el fabricante.
- Representar las posibles posiciones del robot mediante MATLAB.
- Ensamblar, plantear las restricciones geométricas y físicas entre piezas y animación gráfica con restricciones de movimiento en el programa Catia.

Diseño:

- Diseñar las piezas necesarias para el funcionamiento del robot.
- Diseñar la estructura que incorpora el robot.
- Diseñar la instalación neumática y de control.

En segundo lugar y teniendo como base el proyecto anteriormente mencionado se presentó "Montaje, programación y puesta en marcha de un robot neumático de estructura paralela" realizado por: *Alejandro Rosillo Meseguer* y cuyo contenido se expone a continuación:

- Conexión eléctrico entre la tarjeta de adquisición de datos, los circuitos de adaptación de tensión las válvulas electroneumáticas y los cilindros.
- Realización del montaje físico de todas las partes que componen el robot.
- Programación de la tarjeta de adquisición.
- Programación de los algoritmos de control y de un interface amigable mediante un GUI realizado en MATLAB.
- Calibración y puesta en marcha de todo el robot paralelo.
- Averiguar el grado de efectividad que puede llegar a tener el control todo-nada en el control de procesos que varían rápidamente y que requieran cierta precisión, en lugar del control analógico mediante PID.



- Control de posición de un cilindro neumático.
- Control de posición del robot.
- Aplicación de distintas técnicas experimentales de sintonía de controladores PID.

Ambos proyectos pueden ser encontrados en la biblioteca de la UPCT.

## **1.2 Objetivos de este proyecto.**

Los objetivos seguidos se han realizado tomando como base los proyectos anteriores, realizando cambios de hardware como la sustitución de la tarjeta de adquisición de datos por el PLC S7-300 de Siemens, eliminación de los circuitos de potencia así como también se han realizado cambios de software, sistemas SCADA, STEP 7, etc.

Objetivos:

- Realizar el Conexionado eléctrico entre autómatas (s7-300) válvulas electroneumáticas y cilindros.
- Calibrar un cilindro neumático.
- Realizar el control de posición de un cilindro neumático.
- Realizar el Control SCADA de un cilindro neumático mediante el sistema SCADA (Panel de operador PC).
- Realizar la transferencia a pantalla táctil TP177B Color PN/DP.
- Calibrar dos cilindros neumáticos simultáneamente.
- Controlar la posición del Robot (MIMO: Múltiple Input Múltiple Output, Entradas múltiples Salidas múltiples).
- Realizar cinemática directa.
- Realizar cinemática inversa.
- Explicar ventajas e inconvenientes entre la utilización de la tarjeta de adquisición de datos y el autómatas S7-300 de Siemens.

## **1.3 Antecedentes históricos.**

Una buena definición de robot fue la aportada por “Robot Institute of América”, que lo define como:

*“Mecanismo programable multifuncional diseñado para mover materiales, piezas, herramientas o dispositivos especiales, mediante movimientos variados, programados para la ejecución de distintas tareas”.*

### **1.3.1 Robots de estructura paralela.**

*Un robot paralelo es aquel cuya estructura mecánica está formada por un mecanismo de cadena cerrada en el que el efector final se une a la base por al menos dos cadenas cinemáticas independientes.*



Los Robot paralelos pueden ser utilizados en aplicaciones donde tienen ventaja sobre los manipuladores seriales, estas son:

- Tienen la capacidad de manipular cargas muy superiores a su propio peso.
- Ofrecen alta rigidez y bajo peso.
- Presentan elevadas velocidades de operación.
- También presentan otras características, que pueden representar una desventaja según se apliquen:
- Su cinemática es más complicada.
- El espacio de trabajo suele ser pequeño y no es sencillo su cálculo.
- Debe resolverse para cada topología el problema de las configuraciones singulares.

La combinación del número de cadenas cinemáticas, el tipo de estas, sus restricciones en el movimiento de las articulaciones y otras características, hacen que existan innumerables configuraciones estructurales para los mecanismos paralelos.

### **1.3.2 Historia de los autómetas programables.**

Los PLC's se introdujeron por primera vez en la industria en la década de 1960. La razón principal fue la necesidad de eliminar el alto coste que se producía al reemplazar el complejo sistema de control basado en relés y contactores.

Bedford Associates propuso algo denominado Controlador Digital Modular (MODICON, Modular Digital Controlar) a un gran fabricante de coches.

Otras compañías propusieron esquemas basados en ordenador, uno de los cuales estaba basado en el PDP-8. El MODICON 084 resultó ser el primer PLC del mundo en ser producido comercialmente.

El problema de los relés era que cuando los requerimientos de producción cambiaban también lo hacía el sistema de control. Esto comenzó a resultar bastante caro cuando los cambios fueron frecuentes. Dado que los relés son dispositivos mecánicos y poseen una vida limitada se requería un mantenimiento planificado. Por otra parte, a veces se debían realizar conexiones entre cientos o miles de relés, lo que implicaba un enorme esfuerzo de diseño y mantenimiento.

Los nuevos controladores debían ser fácilmente programables por ingenieros de planta o personal de mantenimiento. El tiempo de vida debía ser largo y los cambios en el programa tenían que realizarse de forma sencilla. Finalmente se imponía que trabajaran sin problemas en entornos industriales adversos. La solución fue el empleo de una técnica de programación familiar y reemplazar los relés mecánicos por relés de estado sólido.

A mediados de los 70 las tecnologías dominantes de los PLC eran máquinas de estado secuenciales y CPU's basadas en desplazamiento de bit. Los AMD 2901 y 2903 fueron muy populares. Los microprocesadores convencionales proveyeron la potencia necesaria para resolver de forma rápida y completa la lógica de los pequeños PLC's. Por cada modelo de microprocesador había un modelo de PLC basado en el mismo. No obstante, el 2903 fue de los más utilizados.



La capacidad de comunicación comenzó a aparecer en 1973 aproximadamente. El primer sistema fue el bus Modicon (Modbus). El PLC podía ahora dialogar con otros PLC y ubicarse alejado de las máquinas que controlaba. También se introdujeron en el mundo analógico al poder enviar y recibir señales de tensión. Desafortunadamente, la

falta de un estándar acompañado de un continuo cambio tecnológico ha hecho que la comunicación del PLC sea un cúmulo inconexo de sistemas físicos y protocolos incompatibles entre sí.

En los años 80 se produjo un intento de estandarización de las comunicaciones con el protocolo MAP (Manufacturing Automation Protocol) de General Motor's. También se redujeron las dimensiones del PLC y se pasó a programar con lenguajes simbólicos a través de ordenadores personales en vez de los clásicos terminales de programación.

Los años 90 mostraron una gradual reducción en el número de nuevos protocolos de comunicación, y en la modernización de las capas físicas de los protocolos más populares que sobrevivieron a los 80. El estándar IEC 1131-3 intenta unificar el sistema de programación de todos los PLC en un único estándar internacional.

Ahora disponemos de PLC's que pueden ser programados en diagramas de bloques, lista de instrucciones, C y texto estructurado al mismo tiempo.

Los PC están comenzando a reemplazar al PLC en algunas aplicaciones, incluso la compañía que introdujo el Modicon 084 ha cambiado al control basado en PC.

### **Campos de aplicación**

Un autómatas programable suele emplearse en procesos industriales que tengan una o varias de las siguientes necesidades:

- Espacio reducido.
- Utilización en ambientes exigentes o agresivos.
- Procesos de producción periódicamente cambiantes.
- Procesos secuenciales.
- Maquinaria de procesos variables.
- Instalaciones de procesos complejos y amplios.
- Chequeo de programación centralizada de las partes del proceso.

Aplicaciones generales:

- Maniobra de máquinas.
- Maniobra de instalaciones.
- Señalización y control.

Esto se refiere a los autómatas programables industriales, dejando de lado los pequeños autómatas para uso más personal (que se pueden emplear, incluso, para automatizar procesos en el hogar, como la puerta de una cochera o las luces de la casa).

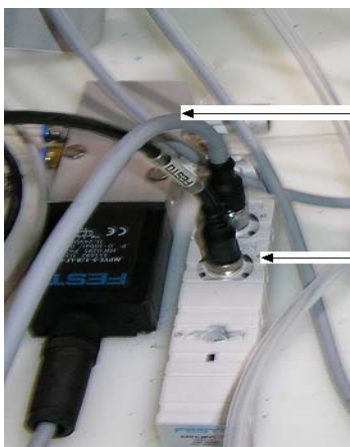


# Capítulo 2

## *Descripción de los componentes de la planta.*

## 2.1 Introducción

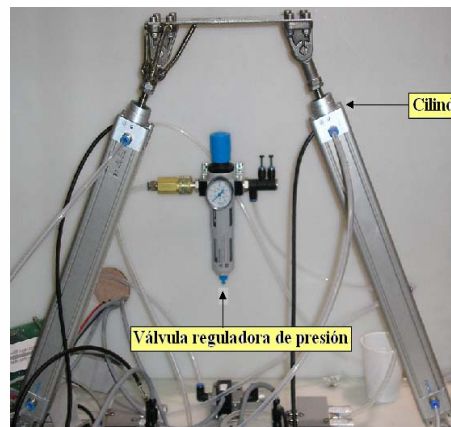
A continuación se presentan unas fotografías describiendo las diferentes partes del robot de estructura paralela de actuación neumática. En la fotografía a) se muestra una válvula de control proporcional 5/3 vías y un convertidor DADE el cual nos devuelve la salida en posición del cilindro neumático, en la Fig. b) se presenta el robot completo con los cilindros neumáticos unidos mediante una plataforma que lo clasifica dentro de los robots paralelos planos, en la Fig. c) nos aparece el autómatas S7-300 de SIEMENS el cual definiremos más adelante, en la figura d) aparece la fuente de alimentación, la cual utilizamos para alimentar los 2 convertidores DADE así como las 2 Válvulas proporcionales ya que nuestro autómatas no nos saca la intensidad suficiente. En la Fig. e) se muestra la pantalla táctil utilizada para realizar el control SCADA TP177-B color PN-DP.



Válvula Proporcional

Convertidor DADE

a)



Cilindro de doble efecto

Válvula reguladora de presión

b)



S7 300, CPU 314C-2DP

c)



d)

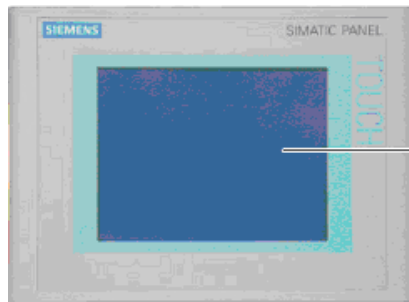


Figura 2. 1: a) Válvula proporcional, Convertidor DADE; b) Cilindros de doble efecto; c) Plc, d) Fuente de alimentación; e) Pantalla táctil.

## 2.2 Hardware.

### 2.2.1 Sistema de automatización S7-300.

En un principio se utilizó el autómatas S7-200 (CPU224) de SIEMENS, para el control PID de un cilindro neumático, este autómatas posee una salida analógica la cual nos es insuficiente para el control del robot, por lo tanto decidimos utilizar el autómatas S7-300 (CPU 314C-2DP) de SIEMENS disponible en el laboratorio de automatización de la UPCT, a la vez pasamos de utilizar un autómatas de gama baja S7-200, (con 14 entradas digitales, 2 entradas analógicas, 10 salidas digitales y 1 salida analógica) a un autómatas de gama media S7-300, (el cual posee 21 entradas digitales, 4 analógicas, 14 salidas digitales y 2 analógicas) .

En la figura 2.2 podemos observar una imagen detallada del autómatas S7-300 de SIEMENS y a continuación, en la figura 2.3 podemos observar el cable MPI, utilizado para realizar la conexión del PC con nuestro autómatas.

Más información sobre este dispositivo en Anexo 1.

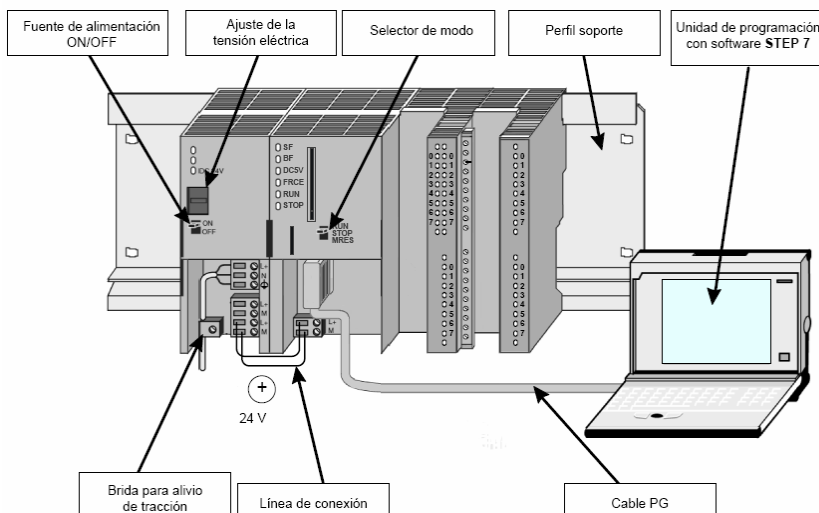


Figura 2. 2 S7-300 CPU 314C-2DP.



## 2.2.2 Cilindros normalizados de doble efecto.

El cilindro elegido es el modelo *DNCI-32-300-P-A* y se hará uso de dos de ellos que serán los encargados del movimiento del robot sobre el plano de trabajo, este sofisticado cilindro está pensado para transportar masas y lleva integrado un sensor de medición de desplazamiento tipo DNCI (normalmente el sensor es externo), la fijación se ha realizado con 4 tornillos en los agujeros roscados de la culata a la base del armario, donde está colocado el robot, la unión al eje entre los dos cilindros, se realiza mediante una rótula para uno de los cilindros y roscada directamente a la pieza en el caso del otro cilindro.

Para el control de estos cilindros usaremos una válvula para cada uno, de tipo *MPYE-5-1/8-LF-010-B*.

El sensor del cilindro genera una onda senoidal y otra cosenoidal y mediante la diferencia de fase entre estas dos señales se consigue conocer la posición del cilindro, luego será necesario el uso de convertidores *DADE-MVC-010* que aumentarán y traducirán esta señal a una señal lineal de 0 a 10 V que simplifica enormemente el uso de la señal adquirida ya que no será necesario hacer la conversión por “software” a valores lineales lo que permite además ahorrar recursos de computación, y permite también disminuir el número de conexiones al autómatas, de cuatro entradas diferenciales a sólo dos por cilindro.

En la figura 2.4 se muestra un cilindro de doble efecto en la que se detallan todos sus componentes y en la figura 2.5 se detalla la: asignación de pines, designación... del cable S1, nº 3 en la figura 2.4

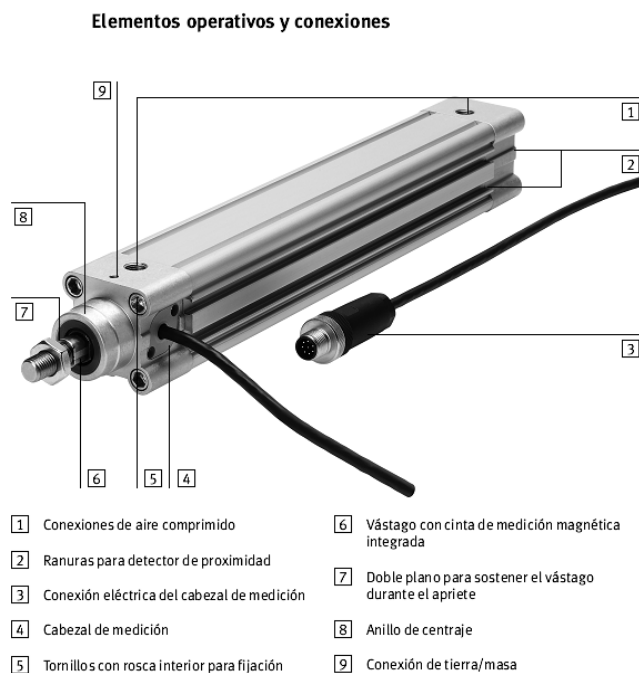


Figura 2. 3 Cilindro normalizado de doble efecto.



Pin	Color de cable	Designación	Asignación de pines (vista de la clavija)
1	negro	+Ub	
2	marrón	0 V	
3	rojo	Señal seno +	
4	naranja	Señal seno -	
5	verde	Señal coseno -	
6	amarillo	Señal coseno +	
7	Apantallamiento/ blindaje	Apantallamiento/ tierra	
8	-	n.c.	

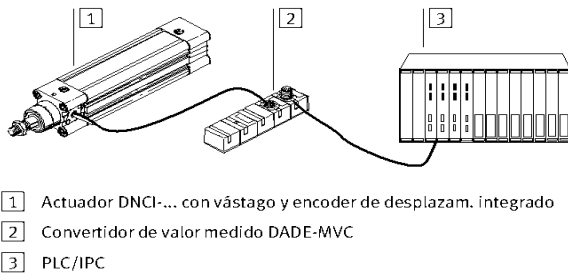
Figura 2. 4 Cable S1 (Cilindro→ convertidor).

### 2.2.3 El convertidor de valor medido.

Realiza la conversión de pulsos digitales del sistema de medida DNCI en una señal analógica de 0 a 10 V. Realiza la función de adaptador de señal entre el sensor del cilindro y el PLC.

En la figura 2.5 se muestra una fotografía de cómo se debe de conectar el cilindro al convertidor y este al PLC.

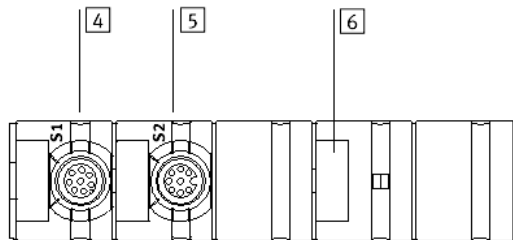
En la figura 2.6 nos aparecen las conexiones del convertidor detalladas.



- 1 Actuador DNCI-... con vástago y encoder de desplazam. integrado
- 2 Convertidor de valor medido DADE-MVC
- 3 PLC/IPC

Figura 2. 5 Conexión PLC, Convertidor, Cilindro

#### Conexiones y elementos operativos



- 4 S1: Conexión para PLC (con LED verde/rojo)
- 5 S2: Conexión para el encoder de desplazamiento del DNCI-... (con LED amarillo/rojo)
- 6 Etiquetas de identificación ISB-8x20 (accesorios)

Figura 2. 6 Convertidor del valor medido.

Son necesarios dos de estos convertidores *DADE-MVC-010* para controlar los dos cilindros *DNCI-32-300-P-A*.



### Conexión de los convertidores DADE-MVC-010.

La conexión entre los convertidores de tensión y el sensor de los cilindros *DNCI* es directa mediante el cable *S2*, mientras que entre los convertidores y el autómatas la conexión se realiza mediante el cable *S1*, que está compuesto internamente por 8 cablecillos de colores que son los que conectaremos en nuestro autómatas

Pin	Color	Asignación	Conexión S1
1	blanco	+24 V alimentación	
2	marrón	Señal medida (analógica)	
3	verde	Salida de referencia	
4	amarillo	0 V de la señal medida	
5	gris	Entrada de referencia	
6	rosa	Entrada de calibración	
7	azul	Salida Ready (preparado)	
8	rojo	0 V alimentación y entradas/salidas	

Figura 2. 7 Cable S2 (Convertidor→PLC).

-El cable de color **blanco** va conectado a 24V o, lo que es lo mismo, al terminal positivo de la fuente.

-El cable de color **marrón** es el valor medido y varía entre 0 y 10V. Los cables de los dos convertidores van conectados a los terminales positivos de las entradas analógicas de nuestro autómatas (PEW ...) respectivamente.

-El cable **verde** se usa para comprobar que hemos introducido la señal de referencia (24V en el cable gris),

-El cable **amarillo** es el 0V de la señal medida, conectado al terminal negativo de la entrada analógica de nuestro autómatas (PEW\_\_)

-El cable **gris** sirve para establecer el punto del recorrido del cilindro en el que deseamos que se considere el punto para el cual el convertidor dará una salida de 0V, por encima de este punto la salida aumentará hasta alcanzar el valor en el que se estableció el punto de calibración. Esta señal de referencia se pierde tras una desconexión eléctrica por lo que habrá que referenciar los cilindros cada vez que se conecta la fuente de alimentación.

-El cable **rosa** es similar al gris excepto que éste establece el punto de calibración que se caracteriza por ser el lugar en el que el valor de salida del convertidor alcanza su valor máximo de 10V. Además, es un punto permanente (sólo es necesario establecerlo una vez).

-El cable **azul** es de comprobación, se encuentra a nivel alto tras haber introducido la señal de calibración (en el cable rosa) .



-El cable **rojo** va unido al terminal negativo de la fuente de alimentación.

### **Primera puesta a punto**

Durante la primera puesta a punto o cuando la memoria ha sido restablecida, el *DADE-MVC* debe incluir lo siguiente:

- El punto de referencia (referencing).
- Longitud de la carrera de trabajo (calibration).

Pasos a seguir:

1. Conecte la tensión de alimentación.
2. Mueva el *DNCI* al punto de referencia (punto cero de la carrera de trabajo).
3. Active la entrada de referencia (PIN 5 en el S1) por lo menos durante 0,5s.
4. Desactive la señal de entrada de referencia. Tras aparecer una señal (retardo 0,1 s) en la salida de referencia (PIN 3 en el S1), el punto de referencia es guardado (el *DADE* se halla entonces “referenciado”).
5. Mueva a la posición final de la carrera de trabajo y asegúrese de que el *DNCI* se haya detenido.
6. Active la entrada de calibración (PIN 6 en el S1) por lo menos durante 0,5s.
7. Desactive la señal de entrada de calibración. Tras aparecer una señal (retardo 0,1 s) en la salida Ready (PIN 7 en el S1), el punto de calibración es guardado remanentemente (el *DADE* se halla entonces “calibrado”).

### **Puesta a punto tras un encendido/apagado.**

Si se desconecta la alimentación, el punto de referencia se perderá.

-Realice de nuevo los pasos 1...4.

-Si la referencia es correcta, aparecerá una señal 1 directamente en la salida Ready con el paso 4.



## Resumen

	LED		INPUT		OUTPUT	
	S1	S2	Ref	Cal	Ref	Rdy
<b>Primera puesta a punto</b>						
24 V aplicados	verde parpadeante	rojo	0	0	0	0
Inicio referencia	verde	rojo	1	0	0	0
Fin referencia	verde	amarillo	0	0	1	0
Inicio calibración	verde	apagado	0	1	1	0
Fin calibración	verde	apagado	0	0	1	1
<b>Puesta a punto del sistema calibrado</b>						
24 V aplicados	verde parpadeante	apagado	0	0	0	0
Inicio referencia	verde	apagado	1	0	0	0
Fin referencia	verde	apagado	0	0	1	1

Figura 2. 8 Resumen.

### Restablecimiento al estado de suministro.

1. Active la entrada de referencia (PIN 5 en el S1) y la entrada de calibración (PIN6 en el S1) simultáneamente durante al menos 2 segundos, los valores para el punto de referencia y la longitud de la carrera guardados en el *DADE* se borrarán.
2. Desactive las dos entradas de nuevo.

### Salida analógica

En el caso de un fallo o si el sistema no ha sido calibrado, la salida analógica está en 0V.

Caso especial fallo de la carrera:

- Si el punto de calibración ha sido sobrepasado: 10V
- Si el punto de referencia ha sido sobrepasado: 0V

### Diagnosis

- Si hay un fallo: salida de referencia y salida Ready = 0
- Si hay un fallo de la carrera: sólo la salida Ready = 0



LED S1	LED S2	Fallo / Solución	Reposición
Verde	rojo parpadea 1x	Fallo del sensor: Verifique el sensor y el cable	- Desconectar, conectar
Verde	rojo parpadea 2x	Fallo de referencia: El DNCI se ha movido durante Ref = 1. El DNCI debe estar parado. Realice de nuevo la referencia.	- Desconectar, conectar - o entrada de referencia = 1
Verde	rojo parpadea x3	Fallo de calibración: El DNCI se ha movido durante Cal = 1. El DNCI debe estar parado. Realice de nuevo la calibración	- Desconectar, conectar - o entrada de calibración = 1
Verde	rojo parpadea x4	Fallo de incremento: El número de incrementos entre el punto de referencia y el final de la carrera demasiado pequeños (el DNCI no se ha movido). Realice de nuevo la calibración.	- Desconectar, conectar - o entrada de calibración = 1
Rojo	apagado	Subtensión: Verifique la tensión.	- Desconectar, conectar
Verde	rojo	Fallo carrera: Valor de la señal fuera del margen previsto. Verifique los topes o restablezca.	- Desconectar, conectar - o llevar el DNCI al margen permitido - o restablecer (ver cap. 6.1).
Rojo	rojo	Cortocircuito en la salida: eliminar.	- Desconectar, conectar

Figura 2. 9 Diagnósis.

## 2.2.4 Válvulas de control proporcional

La señal analógica de entrada a la válvula permite una regulación progresiva del caudal volumétrico (mediante función progresiva de estrangulamiento y permite también cambiar la dirección del caudal volumétrico. La válvula sirve al mismo tiempo para regular la velocidad como para posicionar el cilindro. Se trata de una 5/3 vías.

Se requieren dos unidades de válvulas *MPYE-5-1/8-LF-010-B* una para controlar cada uno de los dos cilindros normalizados de doble efecto.



Figura 2. 10 Válvulas MPYE-5-1/8-LF-010-B.



### Conexión de las válvulas proporcionales.

-El cable **marrón** va al terminal positivo de la fuente de alimentación (24V), mientras que el **blanco** se conecta al terminal negativo (0V) en ambas válvulas.

-El cable **verde** de una válvula va unido a la salida analógica de nuestro autómeta PAW752, al terminal positivo, mientras que el de la otra válvula va unido a la salida PAW754 (también al terminal positivo) el autómeta les proporcionará una señal entre 0 y 10V dependiendo de las necesidades de control.

-El cable **amarillo** de una válvula se conecta a la salida PAW752 al terminal negativo y el de la otra a la salida PAW754 (al terminal negativo).

A continuación se muestra una tabla en la que se indican las conexiones de la válvula proporcional así como la asignación de sus pines.

Pin nº	Color *)	Ocupación: Variante de tensión	Ocupación: Variante de corriente
Pin 1	Marrón	$U_n$ (+17...30 V c.c.)	tensión de alimentación
Pin 2	Blanco	0 V (-)	tensión de alimentación
Pin 3	Verde	$U_c$ (0...+10 V) tensión nominal	$I_c$ (4...20 mA) corriente nominal
Pin 4	Amarillo	0 V (-) tensión nominal	0 mA (-) corriente nominal

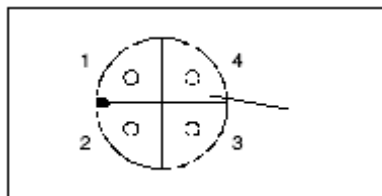


Figura 2. 11 Conexión de las válvulas proporcionales.

### Eliminación de fallos.

Problema	Posible causa	Solución
La corredera no se mueve	Falta tensión de alimentación	Controlar la conexión de 24 V de tensión de alimentación
	Falta valor nominal de referencia	Controlar la unidad de mando, controlar la conexión
	Corredera atascada	Enviar la unidad MPYE a Festo
La corredera vibra	Tensión de alimentación demasiado baja	Procurar la alimentación de 17...30 V c.c.
	Falta apantallamiento del cable o conexión deficiente	Colocar apantallamiento en el extremo del cable alejado de la válvula
	Bucle de conexión a tierra	Aislar la unidad MPYE
Cilindro demasiado lento	Estrangulamiento del paso por error de conexionado (racores basculantes)	Recurrir a conexiones de alternativa

Figura 2. 12 Eliminación de fallos.

## 2.2.5 Válvula reguladora de presión.

Utilizaremos la válvula reguladora LFR-D-MINI, que regula la presión y compensa las fluctuaciones a la entrada; además efectúa el filtrado de las partículas de suciedad y el condensado de agua por lo que se trata de una unidad muy compacta y completa.



Figura 2. 13 Válvula reguladora de presión.

### Especificaciones técnicas

Presión de entrada p1 max.	16 bar (sin purga automática de condensados)
p1 min	12 bar (con purga automática de condensados)
Margen de presión de trabajo máx. p2	min. p1 1,5 bar; 1 bar (sólo LR-...)
Margen de temperaturas:	de 0,5 a 7 bar (con LR-/LFR-...-D7-...) de 0,5 a 12 bar (con LR-.../LFR-...-D-...)
Posición de montaje	-10° C ... +60° C (Almacenaje, fluido, ambiente) +1,5° C ... +60° C (Temp. del fluido con LFM-...-...) +1,5° C ... +30° C (Temp. del fluido con LFX-...-...)
Conexiones del manómetro	Vertical (± 5°); indiferente (sólo LR-...)
Finura del filtro	G1/8 (en LR-/LFR-...-MINI-...) G1/4 (en LR-/LFR-...-MIDI-/MAXI-...)
Fluido	40 µm (con LF-/LFR-...-D-...); 5 µm (LF-/LFR-...-D-5M-...); 1 µm (con LFMB-...-D-...); 0,01 µm (LFMA-/LFMBA-...-D-...)
	Aire comprimido: - filtrado a finura de ≤ 40 µm con LR-... - no lubricado, filtrado a finura de ≤ 5 µm con LFMB-.../≤ 1 µm con LFMA-.../ ≤ 0,01 µm con LFX-...
Materiales: Cuerpo: GD-Zn; brida de conexión: Al, GD-Zn; protección: Al; vaso: PC (macrolone); piezas internas: POM, PA; pomo de ajuste de la presión: PA ; medio del filtro: PE (40 µm, 5 µm), microfibras (1 µm y 0,01 µm); carbón activo (LFX-...-AC); juntas: NBR	





## 2.2.6 Pantalla táctil TP177B Color PN/DP.

Para realizar el control SCADA se ha utilizado la pantalla táctil a color que se muestra en la figuras 2.15, 2.16, 2.17 la cual poseemos en el laboratorio de automática de la UPCT.

La configuración del SCADA es la misma en la pantalla táctil que en el PC, con la única diferencia de que el tamaño de los componentes del sistema SCADA hay que reducirlos debido a que la pantalla táctil tiene la pantalla más pequeña que la del PC.

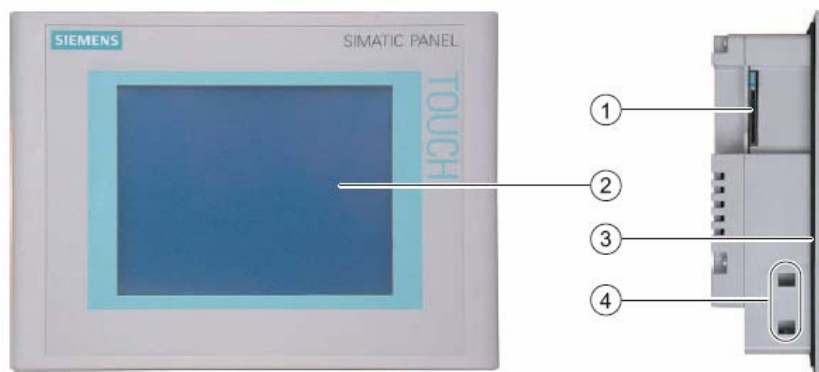


Figura 2. 14 Vistas frontal y lateral de la pantalla.

- ① Ranura para una tarjeta multimedia
- ② Display/Pantalla táctil
- ③ Junta de montaje
- ④ Escotadura para tensores



Figura 2. 15 Vista inferior

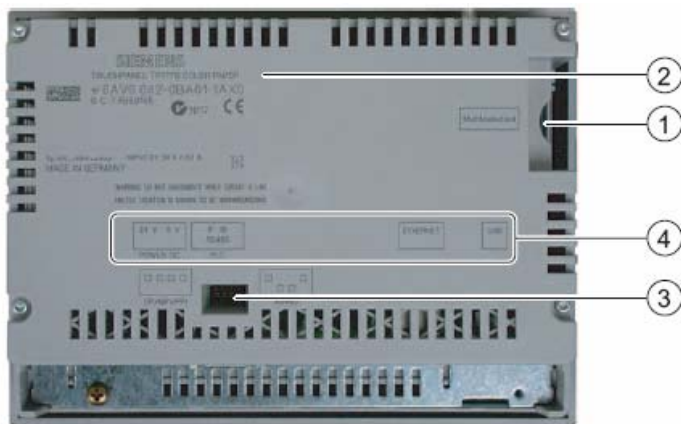


Figura 2. 16 Vista posterior

- ① Ranura para una tarjeta multimedia
- ② Placa de características
- ③ Interruptor DIL
- ④ Nombre del puerto

## 2.2.7 Otros elementos utilizados.

Tubos flexibles, distribuidores múltiples, racores, silenciadores y tapones para las válvulas, son otros elementos importantes empleados para el conexionado neumático del robot.



Figura 2. 17 Conector.



Figura 2. 18 Silenciador



Figura 2. 19 Conector en T

## 2.3 Software.

Para la realización de este proyecto se han utilizado las siguientes herramientas informáticas:

### -Step7: Versión 5.4

El código necesario se ha realizado en STEP 7 (archivos .s7) así como su compilación.

### -PCL-SIM V5.2

Simulador muy potente del autómata s7-300.

### -Parametrizador PID

Herramienta de Step7 necesaria para introducir los parámetros a los bloques PID que se encuentran en nuestro programa.

Ruta: Inicio/todos los programas/Simatic/Step7/Parametrizar regulación PID.



### **-Sistema SCADA.**

La industria posee cada vez procesos productivos más automatizados, complejos y en los que coexiste una gran diversidad de elementos: autómatas, ordenadores, accionamientos neumáticos, robots, etc. Además los sistemas de producción fuertemente centralizados y poco flexibles que se utilizaban hace varias décadas ya no son admisibles.

Esto dió lugar hace unos años a la aparición de los sistemas de producción flexibles que proporcionan respuestas rápidas al mercado fuertemente cambiante en el que están inmersas las empresas. A medida que aumenta la flexibilidad de los sistemas de producción, aumenta la complejidad del sistema de control necesario para gobernarle.

Esto ha dado lugar a la aparición de los sistemas de control "inteligentes" basados en conceptos de: descentralización, autonomía, monitorización, cooperación y colaboración.

Todo este aumento de complejidad lleva a que el personal de mantenimiento cada vez se encuentra con más dificultades a la hora de intervenir, necesitando un personal cada vez más preparado técnicamente y sobre todo en un proceso de formación continuo, debido a los cambios frecuentes que sufren los procesos productivos. Además, debido a la modificación constante del proceso productivo, en periodos muy cortos, se hace necesario un sistema que garantice la fiabilidad, idoneidad y normalización de esas modificaciones, de acuerdo a unas normas acordadas.

Con esta línea de investigación se pretende desarrollar e implantar herramientas dirigidas a los responsables de mantenimiento de las empresas, de tal forma que se pueda intervenir rápidamente en caso de avería y con una mínima formación del personal.

Potenciar el campo del control y supervisión de sistemas complejos basados en Redes y Buses de Campo. Interesándose de forma especial por aquellas configuraciones que "mezclan" distintas tecnologías y que requieren que el diálogo entre todos los participantes sea lo más transparente posible para explotar al máximo las posibilidades de todo el conjunto, tanto desde el punto de vista de Fabricación como desde el punto de vista de Supervisión, ayuda al Mantenimiento y Gestión.

Tenemos en la industria actual sistemas productivos muy complejos controlados por autómatas. Estos procesos están continuamente siendo modificados, y por consiguiente los programas de los autómatas. Esto da lugar a que en situaciones de avería el personal de mantenimiento sea capaz de identificar correctamente la situación del puesto de producción y trasladársela correctamente al autómeta, siendo una tarea compleja y que ocasiona paradas de importancia.

### **SCADA, Una primera idea**

SCADA es un acrónimo por Supervisory Control And Data Acquisition (control y adquisición de datos de supervisión). Los sistemas SCADA utilizan el computador y las 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, y la presentan a un operador en una forma amigable.



Los sistemas SCADA mejoran la eficacia del proceso de monitoreo y control proporcionando la información oportuna para poder tomar decisiones operacionales apropiadas.

Los primeros SCADA eran simplemente sistemas de telemetría que proporcionaban reportes periódicos de las condiciones de campo vigilando las señales que representaban medidas y/o condiciones de estado en ubicaciones de campo remotas. Estos sistemas ofrecían capacidades muy simples de monitoreo y control, sin proveer funciones de aplicación alguna. La visión del operador en el proceso estaba basada en los contadores y las lámparas detrás de paneles llenos de indicadores. Mientras la tecnología se desarrollaba, los ordenadores asumieron el papel de manejar la recolección de datos, disponiendo comandos de control, y una nueva función - presentación de la información sobre una pantalla de CRT. Los ordenadores agregaron la capacidad de programar el sistema para realizar funciones de control más complejas.

Los primeros sistemas automatizados SCADA fueron altamente modificados con programas de aplicación específicos para atender a requisitos de algún proyecto particular. Como ingenieros de varias industrias asistieron al diseño de estos sistemas, su percepción de SCADA adquirió las características de su propia industria. Proveedores de sistemas de software SCADA, deseando reutilizar su trabajo previo sobre los nuevos proyectos, perpetuaron esta imagen de industria-específicos por su propia visión de los ambientes de control con los cuales tenían experiencia. Solamente cuando nuevos proyectos requirieron funciones y aplicaciones adicionales, hizo que los desarrolladores de sistemas SCADA tuvieran la oportunidad de desarrollar experiencia en otras industrias.

Hoy, los proveedores de SCADA están diseñando sistemas que son pensados para resolver las necesidades de muchas industrias con módulos de software industria específicos disponibles para proporcionar las capacidades requeridas comúnmente. No es inusual encontrar software SCADA comercialmente disponible adaptado para procesamiento de papel y celulosa, industrias de aceite y gas, hidroeléctricas, gerenciamiento y provisión de agua, control de fluidos, etc. Puesto que los proveedores de SCADA aún tienen tendencia en favor de algunas industrias sobre otras, los compradores de estos sistemas a menudo dependen del proveedor para una comprensiva solución a su requisito, y generalmente procurar seleccionar un vendedor que pueda ofrecer una completa solución con un producto estándar que esté apuntado hacia las necesidades específicas del usuario final.

Si selecciona a un vendedor con experiencia limitada en la industria del comprador, el comprador debe estar preparado para asistir al esfuerzo de ingeniería necesario para desarrollar el conocimiento adicional de la industria requerido por el vendedor para poner con éxito el sistema en ejecución.

La mayoría de los sistemas SCADA que son instalados hoy se están convirtiendo en una parte integral de la estructura de gerenciamiento de la información corporativa. Estos sistemas ya no son vistos por la gerencia simplemente como herramientas operacionales, sino como un recurso importante de información. En este papel continúan sirviendo como centro de responsabilidad operacional, pero también proporcionan datos a los sistemas y usuarios fuera del ambiente del centro de control que dependen de la información oportuna en la cual basan sus decisiones económicas cotidianas. La mayoría de los vendedores principales de SCADA han reconocido esta



tendencia, y están desarrollando rápidamente métodos eficientes para hacer disponibles los datos, mientras protegen la seguridad y funcionamiento del sistema SCADA. La arquitectura de los sistemas de hoy integra a menudo muchos ambientes de control diferentes, tales como tuberías de gas y aceite, en un solo centro de control.

Para alcanzar un nivel aceptable de tolerancia de fallas con estos sistemas, es común tener ordenadores SCADA redundantes operando en paralelo en el centro primario del control, y un sistema de reserva del mismo situado en un área geográficamente distante.

Esta arquitectura proporciona la transferencia automática de la responsabilidad del control de cualquier ordenador que pueda llegar a ser inasequible por cualquier razón, a una computadora de reserva en línea, sin interrupción significativa de las operaciones.

### **Prestaciones de un SCADA**

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 autómeta, bajo ciertas condiciones.

-Posibilidad de programación numérica, que permite realizar cálculos aritméticos de elevada resolución sobre la CPU del ordenador.

Con ellas, se pueden desarrollar aplicaciones para ordenadores (tipo PC, por ejemplo), con captura de datos, análisis de señales, presentaciones en pantalla, envío de resultados a disco e impresora, etc.

Además, todas estas acciones se llevan a cabo mediante un paquete de funciones que incluye zonas de programación en un lenguaje de uso general (como C, Pascal, o Basic), lo cual confiere una potencia muy elevada y una gran versatilidad. 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.

### **Requisitos de un SCADA.**

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.

### **Módulos de un SCADA.**

Los módulos o bloques software que permiten las actividades de adquisición, supervisión y control son los siguientes:

#### **Configuración:**

Permite al usuario definir el entorno de trabajo de su SCADA, adaptándolo a la aplicación particular que se desea desarrollar.

#### **Interfaz gráfico del operador:**

Proporciona al operador las funciones de control y supervisión de la planta. El proceso se representa mediante sinópticos gráficos almacenados en el ordenador de proceso y generados desde el editor incorporado en el SCADA o importados desde otra aplicación durante la configuración del paquete.

#### **Módulo de proceso:**

Ejecuta las acciones de mando pre-programadas a partir de los valores actuales de variables leídas.

#### **Gestión y archivo de datos:**

Se encarga del almacenamiento y procesado ordenado de los datos, de forma que otra aplicación o dispositivo pueda tener acceso a ellos.

#### **Comunicaciones:**

Se encarga de la transferencia de información entre la planta y la arquitectura hardware que soporta el SCADA, y entre ésta y el resto de elementos informáticos de gestión.



# Capítulo 3

## *Calibración del robot*



### 3.1 Introducción

En este capítulo se explica con todo detalle el calibrado manual y automático de los cilindros.

En la calibración manual se procede a grabar la referencia (0v) y el punto final de la carrera de trabajo (10v) de forma manual, con la toma de aire comprimido desconectada, siguiendo unos pasos que nos va marcando el sistema SCADA.

En la calibración automática se realizan todos estos pasos con la toma de aire conectada, pulsando un botón que nos muestra el sistema SCADA.

En primer lugar se ha realizado con un cilindro neumático para ir adaptándonos al sistema, como podemos observar en la figura 3.1. el cilindro se encuentra sujeto a la estructura del robot mediante un sargento para evitar movimientos de éste de una forma descontrolada.

En segundo lugar procedemos al ensamblaje del robot y realizamos la calibración de ambos cilindros unidos como podemos observar en la figura 3.2.



Figura 3. 1 Un cilindro neumático.

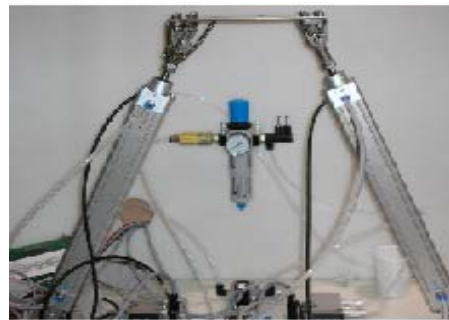


Figura 3. 2 Dos cilindros unidos.





### 3.2 Calibración de un cilindro neumático.

Si decidimos realizar el calibrado manual de un cilindro neumático tenemos que pulsar el botón correspondiente a éste que nos aparece en la pantalla principal de nuestro sistema SCADA. Al pulsar este botón nos aparece un mensaje en la pantalla del PC, “Soltar toma de aire comprimido y pulsar”. Una vez suelta la toma de aire y hemos pulsado el botón que nos indica el sistema SCADA, el programa realizará un reset en el convertidor DADE. El PC cambia el mensaje” recoge vástago y pulsa”. Al pulsar el PLC manda una señal (24v) durante 0,5 segundos al convertidor DADE el cual graba la referencia, nuestro cilindro se hallará entonces referenciado.

A continuación el PC cambia de pantalla y nos muestra otro mensaje “extiende el vástago hasta la posición final de la carrera de trabajo y pulsa”. Al pulsar el PLC manda una señal (24v) durante 0,5 segundos al convertidor DADE, nuestro cilindro se hallará entonces calibrado, nos aparece una pantalla en el PC la cual nos indica que el calibrado se ha realizado de una forma correcta.

En caso de que se produzca algún error durante el calibrado nos aparece un mensaje en la pantalla del PC “Error de calibrado” entonces volvemos a la pantalla principal desconectamos la fuente de alimentación durante unos segundos y realizamos de nuevo el calibrado manual.

Estos pasos se pueden observar con más detalle en el graficet de nivel 1 y nivel 2 del calibrado manual de un cilindro, puntos 3.2.2 y 3.2.3.

Si por el contrario decimos realizar el calibrado automático de un cilindro neumático tenemos que pulsar el botón correspondiente a éste que nos aparece en la pantalla principal de nuestro sistema SCADA, al pulsarlo aparecerá un mensaje en nuestro PC “Para comenzar calibrado manual conecta toma de aire y pulsa START”. Una vez conectada la toma de aire y pulsado START nos pueden ocurrir varias opciones:

#### 1. Es la primera vez que se va a calibrar el convertidor, o se ha realizado un RESET en este.

El PLC graba la referencia, mandando una señal (24v) al convertidor DADE durante 0,5 segundos por su entrada de referencia, el cilindro se halla entonces referenciado, se extiende el vástago de forma automática, graba el punto final de la carrera de trabajo introduciendo una señal (24v) durante 0,5 segundos a la entrada Ready del convertidor DADE, a continuación se recoge el vástago de forma automática, el cilindro se halla calibrado, nos aparece entonces una pantalla en el PC que nos indica que el cilindro ha sido calibrado de forma correcta.

#### 2. Encendido/apagado de la fuente de alimentación.

Si se desconecta la alimentación, el punto de referencia se perderá pero solamente el punto de referencia, por lo tanto el PLC manda una señal (24v) durante 0,5 segundos a la entrada de referencia del convertidor DADE, nuestro cilindro se hallará entonces calibrado, a continuación nos aparece una nueva pantalla en el PC que nos avisa de que nuestro cilindro ha sido calibrado de forma correcta.

En caso de que se produzca algún error durante el calibrado nos aparece un mensaje en la pantalla del PC “Error de calibrado” entonces volveremos a la pantalla principal de nuestro sistema SCADA y volveremos a realizar el calibrado automático.



Estos pasos se pueden ver con más detalle en “graficet de nivel 2 calibrado automático de un cilindro” apartado 3.1.3.

En este punto he realizado solamente el graficet de nivel 2 del calibrado automático ya que para los programas “Cinemática directa”, “Cinemática inversa” y “Control de 2 cilindros” el graficet de nivel 1 es el mismo. Éste lo podemos ver en el apartado 3.3.2.

### 3.2.1 Tabla de símbolos del Programa “Control de 1 cilindro”.

En esta tabla de símbolos se muestran las marcas que se han utilizado para definir etapas, pulsadores, variables analógicas y digitales así como los nombres que le he asignado en el programa realizado en STEP 7 para el control de un cilindro neumático.

#### Variables internas(marcas).

Cycle Execution	OB	1	OB	1
Calibrado_Automático	M	44.0	BOOL	
Calibrado_Manual	M	44.3	BOOL	
Reset_Manual	M	44.4	BOOL	
Conectar_Aire	M	44.7	BOOL	
E0	M	45.0	BOOL	
E1	M	45.1	BOOL	
E2	M	45.2	BOOL	
E3	M	45.3	BOOL	
E4	M	45.4	BOOL	
E5	M	45.5	BOOL	
E6	M	45.6	BOOL	
E7	M	45.7	BOOL	
E8	M	46.0	BOOL	
E9	M	46.1	BOOL	
E10	M	46.2	BOOL	
E11	M	46.3	BOOL	
E12	M	46.4	BOOL	
E13	M	46.5	BOOL	
E14	M	46.6	BOOL	
E15	M	46.7	BOOL	
E16	M	47.0	BOOL	
M44.1	M	44.1	BOOL	
M44.2	M	44.2	BOOL	
Graba_0V_Manual	M	44.5	BOOL	
Graba_10V_Manual	M	44.6	BOOL	
pid	M	48.0	BOOL	

#### Variables analógicas en formato real.

SP_Real	MD	20	REAL	
PV	MD	100	REAL	



**Funciones utilizadas:**

CONT_C	SFB 41	SFB 41	Continuous	Control	(Integrated
	Function,	CPU 314 IFM)			
SCALE	FC 105	FC 105	Scaling Values		
UNSCALE	FC 106	FC 106	Unscaling Values		

**Entradas Digitales utilizadas:**

Entrada de Referencia	E 0.0	BOOL
Entrada de Calibración	E 0.1	BOOL

**Salidas digitales utilizadas:**

Salida de Referencia	A 0.0	BOOL
Salida de Calibración	A 0.1	BOOL

**Entradas analógicas utilizadas:**

Convertidor(C)	PEW 754	BOOL
----------------	---------	------

**Salidas analógicas utilizadas:**

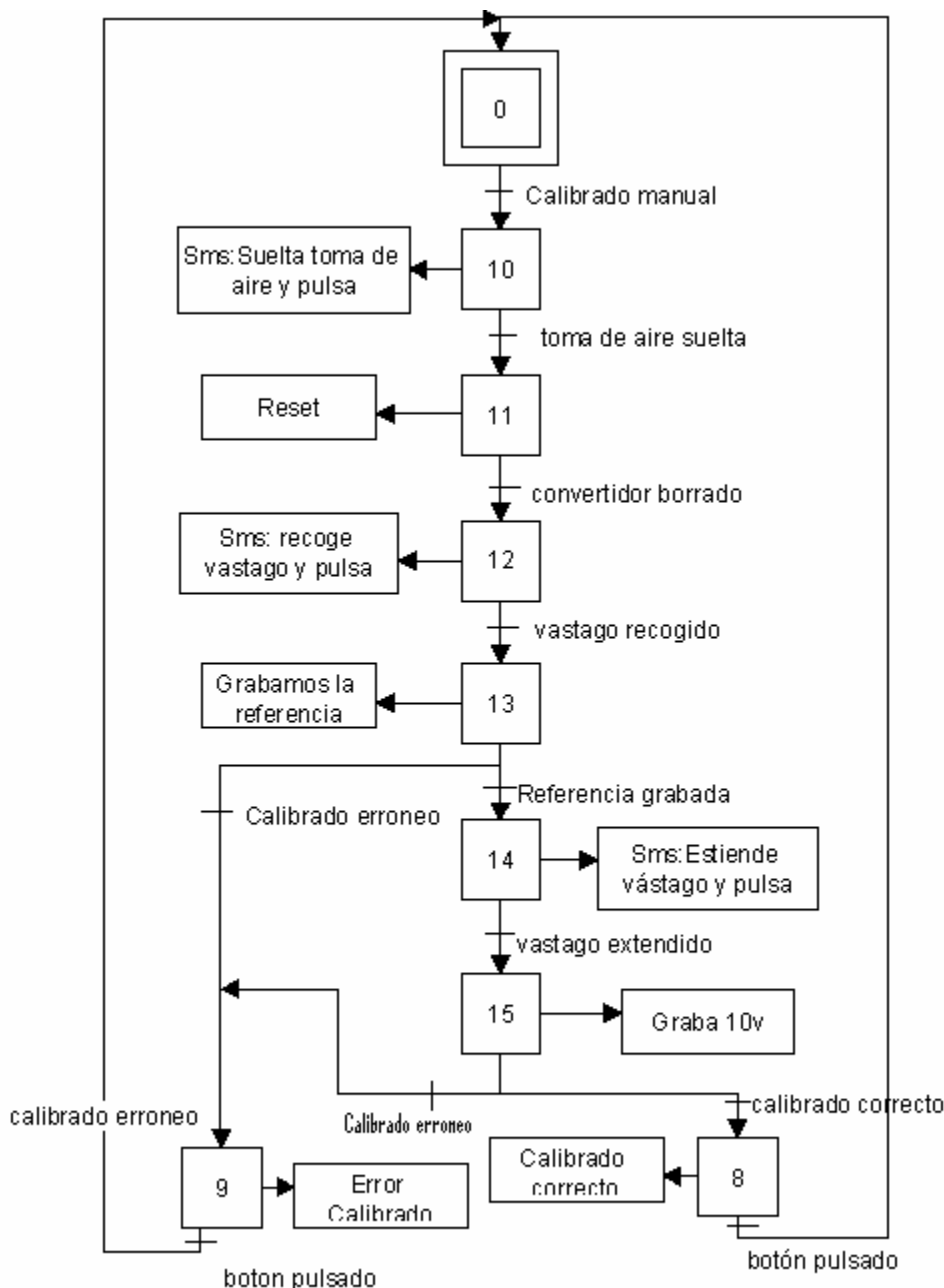
Válvula proporcional(SVP)	PAW 752	BOOL
---------------------------	---------	------



### 3.2.2 GRAFCET nivel 1: Calibrado manual 1 cilindro.

A lo largo de este capítulo observaremos como los graficet de calibrado manual son muy parecidos, ya que todos poseen las mismas etapas.

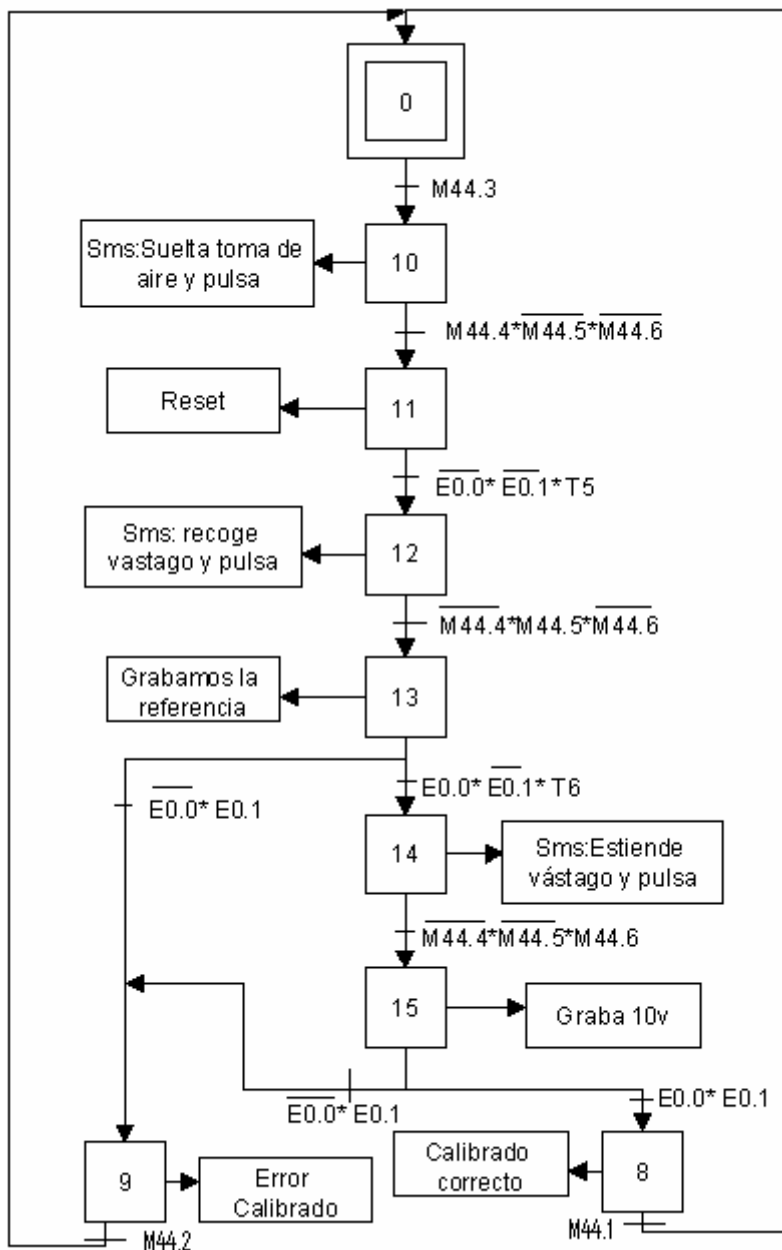
La principal diferencia entre todos estos graficet son las variables utilizadas, en algunos programas se han tenido que cambiar y usar otro direccionamiento debido a la solapación de estas por lo tanto se han tenido que representar todos los graficet de calibrado manual en los que hemos cambiado las variables.





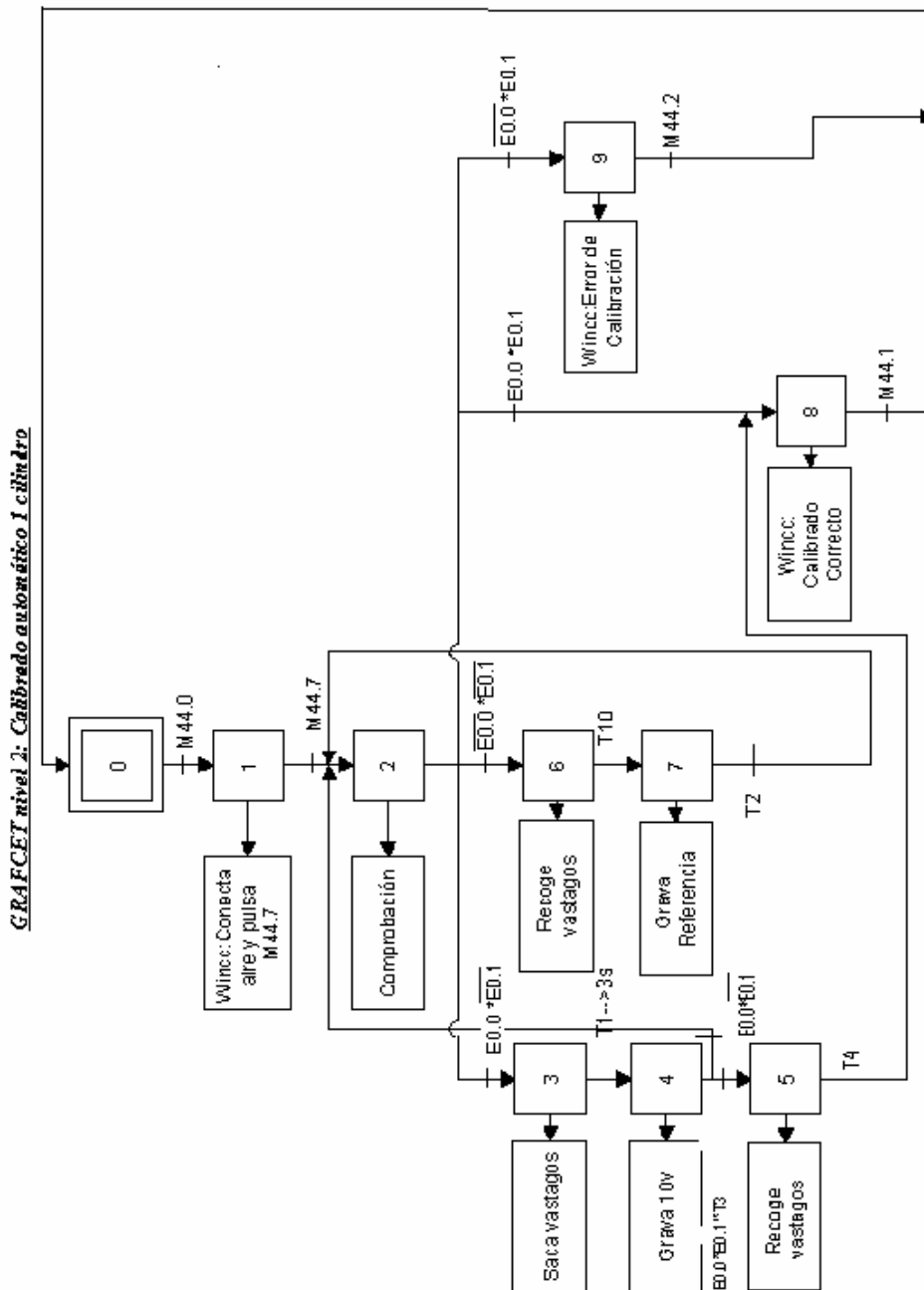
### 3.2.3 GRAFCET nivel 2: Calibrado manual 1 cilindro.

A continuación se muestra el grafcet de nivel 2 del calibrado manual de un cilindro, en éste se pueden observar las variables utilizadas para que nuestro programa cambie de etapa (transiciones), así como las funciones que se realizan en cada una de las etapas.



### 3.2.4 GRAFCET nivel 2: Calibrado automático 1 cilindro.

Como se puede observar en el graficet de calibrado automático de un cilindro que se muestra a continuación éste se complica un poco más que los de calibrado manual ya que al comienzo de este calibrado no se realiza un RESET como se hace en el calibrado manual lo que implica que el convertidor DADE se pueda encontrar con varias configuraciones (mirar apartado 2.2.3).





### 3.2.5 Ecuaciones de activación desactivación de 1 cilindro.

A partir de los diagramas de grafcet de nivel 2 representados anteriormente, se ha procedido a la obtención de las ecuaciones de paso, las que utilizaremos posteriormente para realizar el programa informático “Control de 1 cilindro” en STEP 7.

Como podemos observar en estas ecuaciones disponemos de una etapa 16 la cual no hemos comentado antes. En esta etapa es en la que se realiza todo lo relacionado con el control PID, y no se ha hablado antes de ella ni se ha hecho el grafcet ya que sale directamente desde la etapa 0 a la etapa 16 si pulsamos M48.0. y solo tenemos este camino para acceder a ella.

$$E0 = \overline{E1} * \overline{E2} * \overline{E3} * \overline{E4} * \overline{E5} * \overline{E6} * \overline{E7} * \overline{E6} * \overline{E7} * \overline{E6} * \overline{E7} * \overline{E8} + \overline{E9} + \overline{E10} + \overline{E11} * \overline{E12} * \overline{E13} * \overline{E14} * \overline{E15} * \overline{E16} + E8 * M44.1 + E9 * M44.2$$

$$E1 = E0 * M44.0 + \overline{E1} * \overline{E2}$$

$$E2 = E1 * M44.7 + E4 * E0.0 * \overline{E0.1} + E7 * T1 + E2 * \overline{E3} + E2 * \overline{E6} + E2 * \overline{E8} + E2 * \overline{E9}$$

$$E3 = E2 * E0.0 * \overline{E0.1} + E3 * \overline{E4}$$

$$E4 = E3 * T1 + E4 * \overline{E5} + E4 * \overline{E2}$$

$$E5 = E4 * E0.0 * E0.1 * T3 + E5 * \overline{E8}$$

$$E6 = E2 * \overline{E0.0} * \overline{E0.1} + E6 * \overline{E7}$$

$$E7 = E6 * T10 + E7 * \overline{E2}$$

$$E8 = E2 * E0.0 * E0.1 + E5 * T4 + E15 * E0.0 * E0.1 + E8 * \overline{E0}$$

$$E9 = E2 * \overline{E0.0} * E0.1 + E13 * \overline{E0.0} * E0.1 + E15 * \overline{E0.0} * E0.1 + E9 * \overline{E0}$$

$$E10 = E0 * M44.3 + E10 * \overline{E11}$$

$$E11 = E10 * M44.4 * \overline{M44.5} * \overline{M44.6} + E11 * \overline{E12}$$

$$E12 = E11 * \overline{E0.0} * \overline{E0.1} * T5 + E12 * \overline{E13}$$

$$E13 = E12 * \overline{M44.4} * M44.5 * \overline{M44.6} * + E13 * \overline{E14}$$

$$E14 = E13 * E0.0 * \overline{E0.1} * T6 + E14 * \overline{E15}$$

$$E15 = E14 * \overline{M44.4} * \overline{M44.5} * M44.6 + E15 * \overline{E0}$$

$$E16 = E0 * M48.0 * \overline{M44.0} * \overline{M44.3} + E16 * \overline{E0}$$



### 3.3 Calibración de dos cilindros neumáticos.

Si decidimos realizar el calibrado manual de dos cilindros neumáticos unidos como se muestra en la figura 3.2 se procede de la misma manera que se ha explicado para el calibrado manual de un cilindro neumático, apartado 3.2, se ha utilizado el mismo número de etapas con la única diferencia de que a la hora de extender y recoger los vástagos, realizar un reset, grabar la referencia y el punto final de la carrera de trabajo se hace en los dos cilindros simultáneamente.

Esto se puede observar en el graficet de nivel 1 y graficet de nivel 2 del calibrado manual de 2 cilindros, apartados 3.3.2 y 3.2.2 de este proyecto.

Lo mismo nos ocurre con el calibrado automático con la diferencia de que a la hora de extender y recoger los vástagos lo hace de forma simultánea al igual que al grabar la referencia y el punto final de la carrera de trabajo. Esto lo podemos observar en el punto 3.2.3 y 3.2.4. graficet de nivel 1 y nivel 2 de 2 cilindros.

#### 3.3.1 Tabla de símbolos Programa “2cilindros”.

Como podemos observar esta tabla de símbolos posee las mismas variables booleanas que el programa de un cilindro, esto es porque se ha utilizado el mismo número de etapas y los mismos pulsadores, sin embargo las variables analógicas han aumentado, esto es debido a que hemos introducido otro controlador PID en el programa.

##### Variables internas.

Cycle Execution	OB	1	OB	1
E0	M	45.0	BOOL	
E1	M	45.1	BOOL	
E2	M	45.2	BOOL	
E3	M	45.3	BOOL	
E4	M	45.4	BOOL	
E5	M	45.5	BOOL	
E6	M	45.6	BOOL	
E7	M	45.7	BOOL	
E8	M	46.0	BOOL	
E9	M	46.1	BOOL	
E10	M	46.2	BOOL	
E11	M	46.3	BOOL	
E12	M	46.4	BOOL	
E13	M	46.5	BOOL	
E14	M	46.6	BOOL	
E15	M	46.7	BOOL	
E16	M	47.0	BOOL	
Calibrado_Automático	M	44.0	BOOL	
Calibrado_Manual	M	44.3	BOOL	
Conectar_Aire	M	44.7	BOOL	
Graba_0V_Manual	M	200.0	BOOL	





---

---

Graba_10V_Manual	M	44.6	BOOL
M44.1	M	44.1	BOOL
M44.2	M	44.2	BOOL
pid	M	48.0	BOOL
Reset_Manual	M	210.0	BOOL

**Variables analógicas en formato real.**

PV_1	MD	100	REAL
Set-Point_1	MD	20	REAL
LMN	MD	105	REAL
Set-Point_2	MD	40	REAL
PV_2	MD	103	REAL
LMN_2	MD	110	REAL

**Variables analógicas en formato palabra.**

LMN PALABRA	MW	10	WORD
LMN_2 PALABRA	MW	21	WORD
MW51	MW	51	WORD
MW52	MW	52	WORD

**Funciones utilizadas.**

SCALE	FC	105	FC	105	Scaling Values
UNSCALE	FC	106	FC	106	Unscaling Values
CONT_C	SFB	41	SFB	41	Continuous Control (Integrated Function, CPU 314 IFM)

**Entradas Digitales utilizadas:**

Entrada de Referencia 1 (ER1)	E	0.0	BOOL
Entrada de Calibración 1 (EC1)	E	0.1	BOOL
Entrada de Referencia 2 (ER2)	E	0.2	BOOL
Entrada de Calibración 2 (EC1)	E	0.3	BOOL

**Salidas digitales utilizadas:**

Salida de Referencia 1(SR1)	A	0.0	BOOL
Salida de Calibración 1(SC1)	A	0.1	BOOL
Salida de Referencia 2(SR2)	A	0.2	BOOL
Salida de Calibración 2(SC2)	A	0.3	BOOL

**Entradas analógicas utilizadas:**

Convertidor 1(C1)	PEW	754	BOOL
Convertidor 2(C2)	PEW	756	BOOL

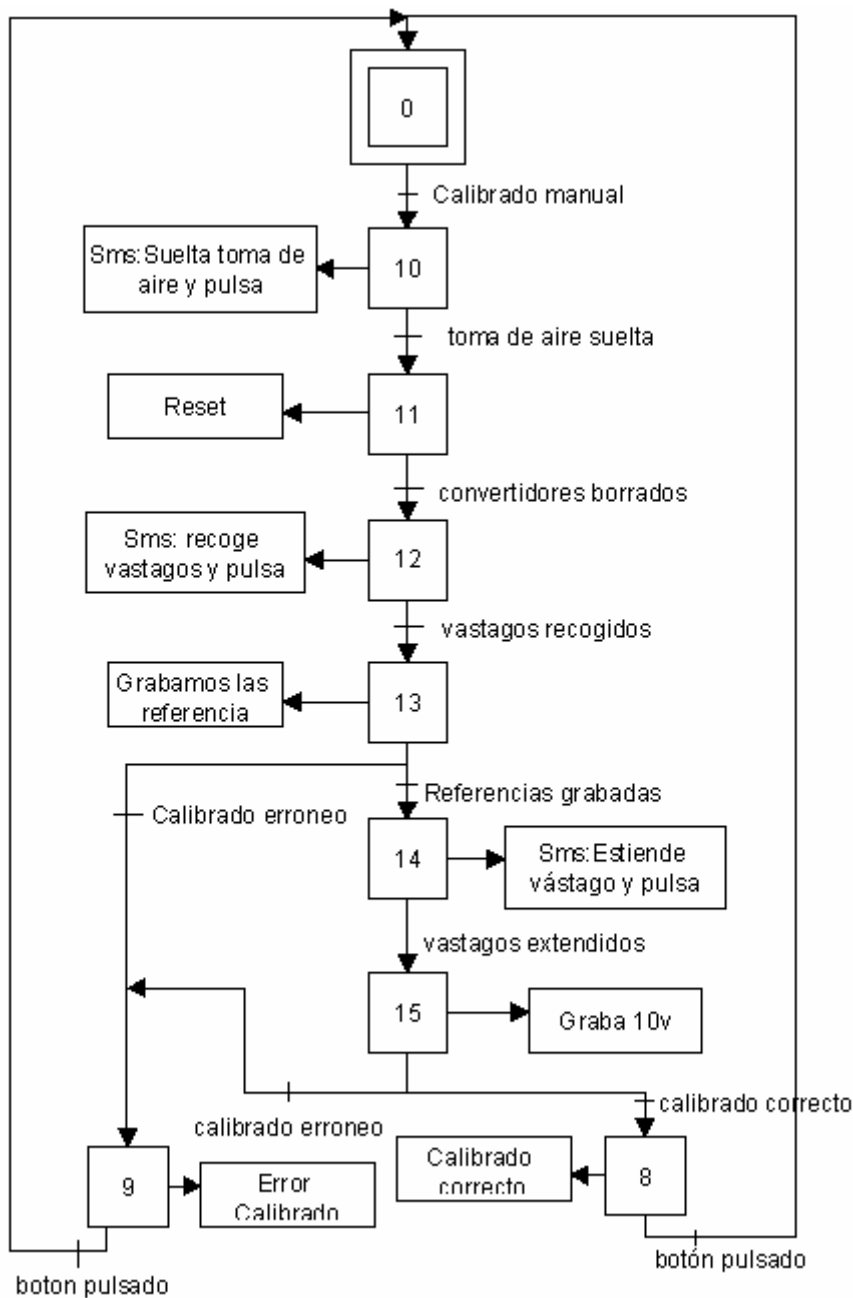
**Salidas analógicas utilizadas:**

Válvula proporcional(SVP1)	PAW	752	BOOL
Válvula proporcional(SVP2)	PAW	754	BOOL



### 3.3.2 GRAFCET nivel 1: Calibrado manual 2 cilindros

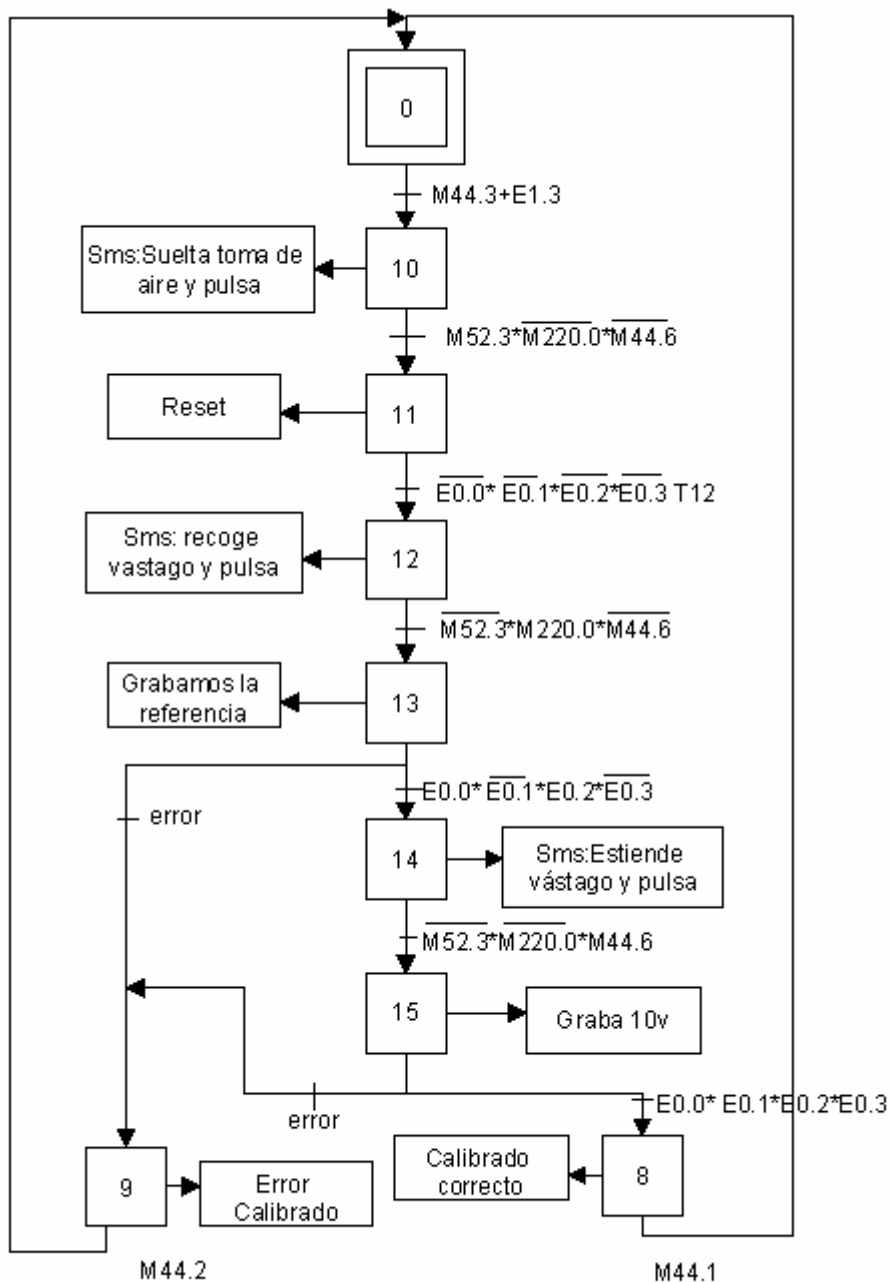
El graficet que se muestra a continuación es igual que el representado en el punto 3.2.2 con la única diferencia, como he comentado anteriormente de que las acciones que realiza en todas las etapas las realiza en los dos cilindros y en los dos convertidores simultáneamente.





### 3.3.3 GRAFCET nivel 2: Calibrado manual 2 cilindros.

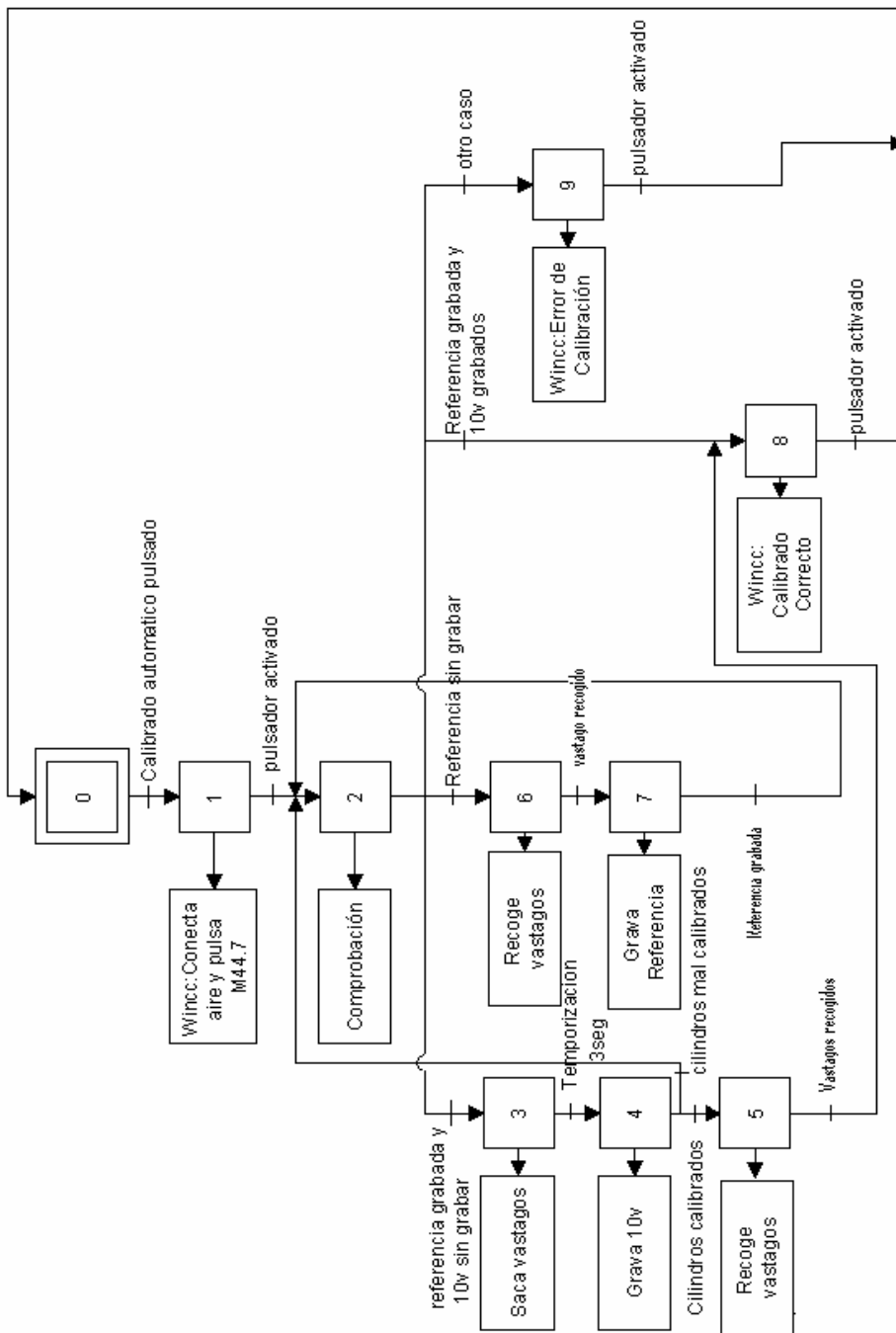
Como podemos observar en este diagrama el direccionamiento de las variables que se utilizan para las transiciones ha cambiado



### 3.3.4 Graficet de nivel 1 Calibrado automático: "2cilindros" "Cinemática directa" "Cinemática inversa".

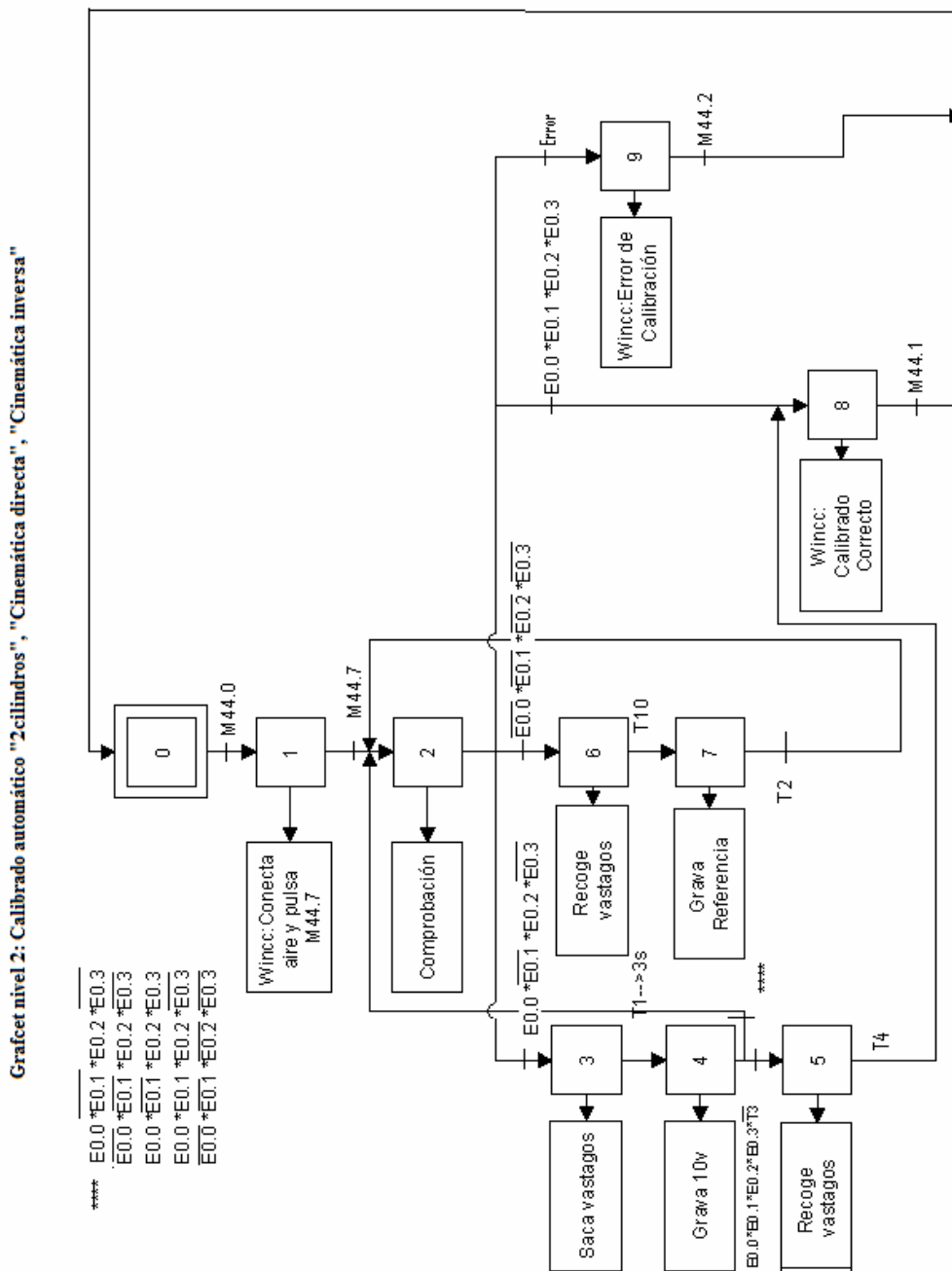
En este apartado se ha representado el graficet de nivel 1 de los tres programas que utilizan los dos cilindros siendo igual para los tres ya que como se ha comentado anteriormente para este calibrado no se ha modificado ninguna variable.

Graficet de nivel 1: "2cilindros", "Cinemática directa", "Cinemática inversa"



### 3.3.5 Grafcet nivel 2: Calibrado automático “2 cilindros” “Cinemática directa” “Cinemática inversa”.

En este apartado se ha representado el grafcet del calibrado automático de los programas que utilizan dos cilindros, si observamos el punto 3.2.4 vemos que los grafcet son muy parecidos, mismo número de etapas, las variables digitales son las mismas, etc. Lo que cambia son las acciones que se realizan en cada etapa ya que cuando se utilizan dos cilindros este tiene que realizar el calibrado de los dos simultáneamente.





### 3.3.6 Ecuaciones activación desactivación “2 Cilindros”.

Estas ecuaciones han sido obtenidas a partir de los digramas representados en los puntos 3.3.3 y 3.3.5 incluyendo además la etapa 16 en la que se realizan las funciones de control PID.

$$E0 = \overline{E1} * \overline{E2} * \overline{E3} * \overline{E4} * \overline{E5} * \overline{E6} * \overline{E7} * \overline{E6} * \overline{E7} * \overline{E6} * \overline{E7} * \overline{E8} + \overline{E9} + \\ + \overline{E10} + \overline{E11} * \overline{E12} * \overline{E13} * \overline{E14} * \overline{E15} * \overline{E16} + E8 * M44.1 + E9 * M44.2$$

$$E1 = E0 * M44.0 + E1 * \overline{E2}$$

$$E2 = E1 * M44.7 + E4 * E0.0 * \overline{E0.1} * E0.2 * \overline{E0.3} + E7 * T2 + E2 * \overline{E3} \\ + E2 * \overline{E6} + E2 * \overline{E8} + E2 * \overline{E9}$$

$$E3 = E2 * E0.0 * \overline{E0.1} * E0.2 * \overline{E0.3} + E3 * \overline{E4}$$

$$E4 = E3 * T1 + E4 * \overline{E5} + E4 * \overline{E2}$$

$$E5 = E4 * E0.0 * E0.1 * E0.2 * E0.3 * T3 + E5 * \overline{E8}$$

$$E6 = E2 * \overline{E0.0} * \overline{E0.1} * \overline{E0.2} * \overline{E0.3} + E6 * \overline{E7}$$

$$E7 = E6 * T10 + E7 * \overline{E2}$$

$$E8 = E2 * E0.0 * E0.1 * E0.2 * E0.3 + E5 * T4 + E15 * E0.0 * E0.1 * E0.2 \\ * E0.3 + E8 * \overline{E0}$$

$$E9 = E2 * \overline{E0.0} * E0.1 * \overline{E0.2} * E0.3 + E15 * \overline{E0.0} * E0.1 * \overline{E0.2} * E0.3 \\ + E9 * \overline{E0}$$

$$E10 = E0 * M44.3 + E10 * \overline{E11}$$

$$E11 = E10 * M52.3 * \overline{M220.0} * \overline{M44.6} + E11 * \overline{E12}$$

$$E12 = E11 * \overline{E0.0} * \overline{E0.1} * \overline{E0.2} * \overline{E0.3} * T12 + E12 * \overline{E13}$$

$$E13 = E12 * M52.3 * \overline{M220.0} * \overline{M44.6} + E13 * \overline{E14}$$

$$E14 = E13 * E0.0 * \overline{E0.1} * E0.2 * \overline{E0.3} + E14 * \overline{E15} + E14 * \overline{E8}$$

$$E15 = E14 * \overline{M210.0} * \overline{M220.0} * \overline{M44.6} + E15 * \overline{E8} + E15 * \overline{E9}$$

$$E16 = E0 * M48.0 * \overline{M44.0} * \overline{M44.3} + E16 * \overline{E0}$$



### 3.4 Cinemática directa.

Este graficet es muy parecido al calibrado manual de 2 cilindros la única diferencia es que a la hora de realizar la cinemática directa se han usado muchas más variables analógicas que en los programas anteriores, nos hemos encontrado con la solapación de variables, con lo que nos hemos visto obligados a modificar algunas variables del Calibrado manual de los programas “Cinemática directa” y “Cinemática inversa”.

#### 3.4.1 Tabla de símbolos del Programa “Cinemática\_Directa”

En esta tabla, las variables booleanas siguen siendo las mismas que en las anteriores tablas mientras que las variables analógicas siguen aumentando ya que para hacer las operaciones aritméticas de cálculo de la Cinemática se han utilizado variables reales.

##### Variables internas.

Cycle Execution	OB	1	OB	1
E0	M	45.0	BOOL	
E1	M	45.1	BOOL	
E2	M	45.2	BOOL	
E3	M	45.3	BOOL	
E4	M	45.4	BOOL	
E5	M	45.5	BOOL	
E6	M	45.6	BOOL	
E7	M	45.7	BOOL	
E8	M	46.0	BOOL	
E9	M	46.1	BOOL	
E10	M	46.2	BOOL	
E11	M	46.3	BOOL	
E12	M	46.4	BOOL	
E13	M	46.5	BOOL	
E14	M	46.6	BOOL	
E15	M	46.7	BOOL	
E16	M	47.0	BOOL	
Graba_0V_Manual	M	210.0	BOOL	
Graba_10V_Manual	M	200.0	BOOL	
Calibrado_Automático	M	44.0	BOOL	
Calibrado_Manual	M	44.3	BOOL	
Conectar_Aire	M	44.7	BOOL	
M44.1	M	44.1	BOOL	
M44.2	M	44.2	BOOL	
pid	M	48.0	BOOL	
Reset_Manual	M	205.0	BOOL	

##### Variables analógicas en formato palabra.

MW10	MW	10	WORD
------	----	----	------



MW51	MW	51	WORD
LMN PALABRA	MW	100	WORD
LMN_2 PALABRA	MW	104	WORD

**Variables analógicas en formato real.**

PV_1	MD	50	REAL
Set-Point_1	MD	20	REAL
LMN	MD	105	REAL
Set-Point_2	MD	40	REAL
PV_2	MD	103	REAL
LMN_2	MD	102	REAL
a	MD	0	REAL
b	MD	2	REAL
UPa	MD	30	REAL
Upb	MD	32	REAL
R	MD	36	REAL
S	MD	38	REAL
N= Xd	MD	184	REAL
r	MD	82	REAL
Yd	MD	86	REAL
d	MD	116	REAL
Alfa	MD	120	REAL
Beta	MD	134	REAL
Tita2	MD	136	REAL
Xc	MD	140	REAL
Yc	MD	144	REAL
fi	MD	148	REAL
Xp	MD	164	REAL
Yp	MD	170	REAL

**Funciones utilizadas.**

SCALE	FC	105	FC	105	Scaling Values
UNSCALE	FC	106	FC	106	Unscaling Values
CONT_C	SFB	41	SFB	41	Continuous Control (Integrated Function, CPU 314 IFM)

**Entradas Digitales utilizadas:**

Entrada de Referencia 1 (ER1)	E	0.0	BOOL
Entrada de Calibración 1 (EC1)	E	0.1	BOOL
Entrada de Referencia 2 (ER2)	E	0.2	BOOL
Entrada de Calibración 2 (EC2)	E	0.3	BOOL

**Salidas digitales utilizadas:**

Salida de Referencia 1 (SR1)	A	0.0	BOOL
Salida de Calibración 1 (SC1)	A	0.1	BOOL
Salida de Referencia 2 (SR2)	A	0.2	BOOL





**Calibración, Control y Diseño SCADA de un robot paralelo neumático con el autómeta S7-300.**

---

---

Salida de Calibración 2(SC2)      A    0.3      BOOL

**Entradas analógicas utilizadas:**

Convertidor 1(C1)                    PEW 754      BOOL

Convertidor 2(C2)                    PEW 756      BOOL

**Salidas analógicas utilizadas:**

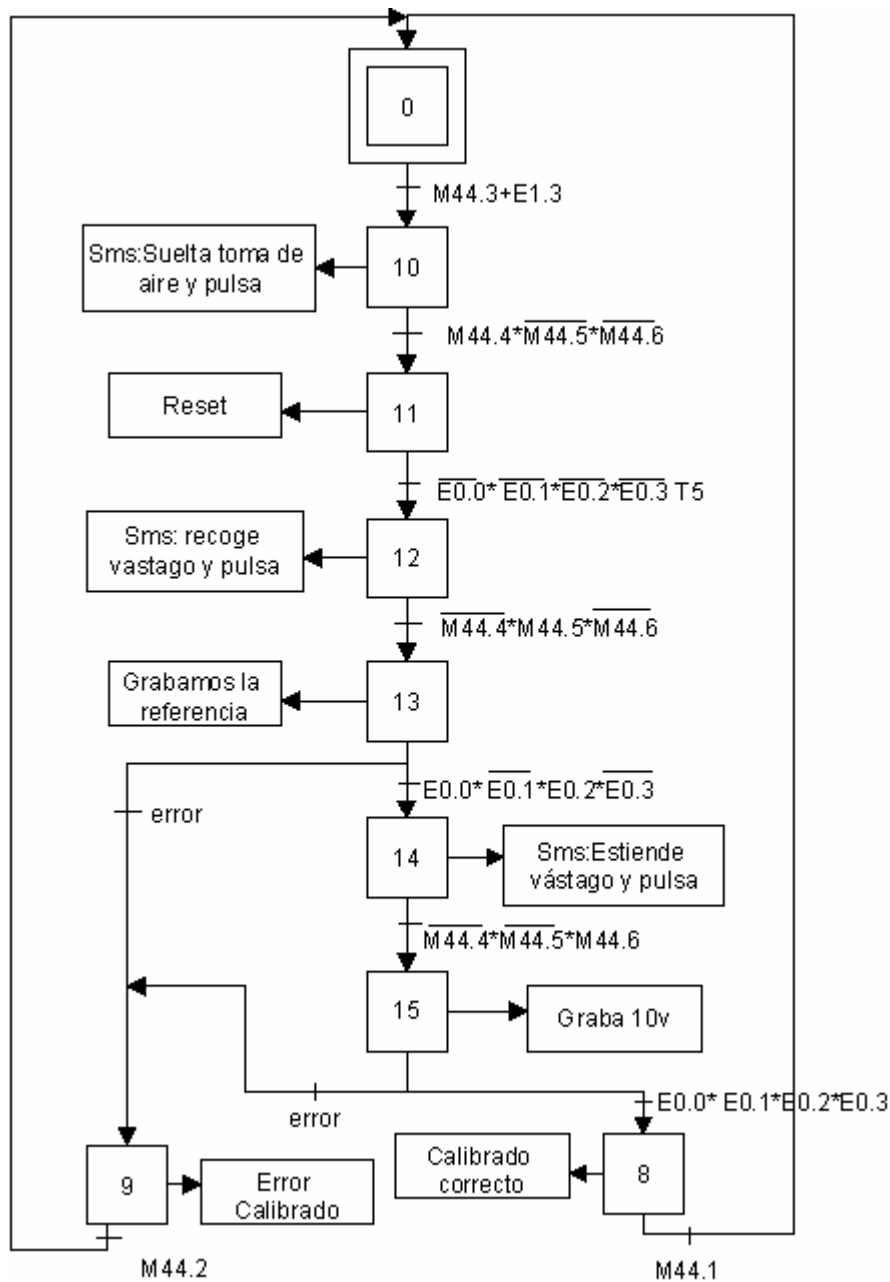
Válvula proporcional(SVP1)        PAW 752      BOOL

Válvula proporcional(SVP2)        PAW 754      BOOL



### 3.4.2 GRAFCET nivel 2: Calibrado manual cinemática directa

Observar en este apartado como la única diferencia con el apartado 3.3.3 son las variables booleanas que se utilizan para algunas de las transiciones.





### 3.4.3 Ecuaciones activación desactivación “Cinemática Directa”.

Las ecuaciones que se muestran a continuación han sido obtenidas a partir de los diagramas de grafset representados en los puntos 3.4.2 y 3.3.5.

$$E0 = \overline{E1} * \overline{E2} * \overline{E3} * \overline{E4} * \overline{E5} * \overline{E6} * \overline{E7} * \overline{E6} * \overline{E7} * \overline{E6} * \overline{E7} * \overline{E8} + \overline{E9} + \\ + \overline{E10} + \overline{E11} * \overline{E12} * \overline{E13} * \overline{E14} * \overline{E15} * \overline{E16} + E8 * M44.1 + E9 * M44.2$$

$$E1 = E0 * M44.0 + E1 * \overline{E2}$$

$$E2 = E1 * M44.7 + E4 * E0.0 * \overline{E0.1} * E0.2 * \overline{E0.3} + E7 * T2 + E2 * \overline{E3} \\ + E2 * \overline{E6} + E2 * \overline{E8} + E2 * \overline{E9}$$

$$E3 = E2 * E0.0 * \overline{E0.1} * E0.2 * \overline{E0.3} + E3 * \overline{E4}$$

$$E4 = E3 * T1 + E4 * \overline{E5} + E4 * \overline{E2}$$

$$E5 = E4 * E0.0 * E0.1 * E0.2 * E0.3 * \overline{T3} + E5 * \overline{E8}$$

$$E6 = E2 * \overline{E0.0} * \overline{E0.1} * \overline{E0.2} * \overline{E0.3} + E6 * \overline{E7}$$

$$E7 = E6 * T10 + E7 * \overline{E2}$$

$$E8 = E2 * E0.0 * E0.1 * E0.2 * E0.3 + E5 * T4 + E15 * E0.0 * E0.1 * E0.2 \\ * E0.3 + E8 * \overline{E0}$$

$$E9 = E2 * \overline{E0.0} * E0.1 * \overline{E0.2} * E0.3 + E13 * \overline{E0.0} * E0.1 * \overline{E0.2} * E0.3 \\ + E15 * \overline{E0.0} * E0.1 * \overline{E0.2} * E0.3 + E9 * \overline{E0}$$

$$E10 = E0 * M44.3 + E10 * \overline{E11}$$

$$E11 = E10 * M44.4 * \overline{M44.5} * \overline{M44.6} + E11 * \overline{E12}$$

$$E12 = E11 * \overline{E0.0} * \overline{E0.1} * \overline{E0.2} * \overline{E0.3} * T5 + E12 * \overline{E13}$$

$$E13 = E12 * \overline{M44.4} * M44.5 * \overline{M44.6} * + E13 * \overline{E14} + E13 * \overline{E9}$$

$$E14 = E13 * E0.0 * \overline{E0.1} * E0.2 * \overline{E0.3} + E14 * \overline{E15}$$

$$E15 = E14 * \overline{M44.4} * \overline{M44.5} * M44.6 + E15 * \overline{E8} + E15 * \overline{E9}$$

$$E16 = E0 * M48.0 * \overline{M44.0} * \overline{M44.3} + E16 * \overline{E0}$$



### 3.5 Cinemática Inversa.

En este graficet al igual que en la cinemática directa también se han tenido que modificar variables, por lo demás el graficet sigue los mismos pasos que los anteriores.

#### 3.5.1 Tabla de símbolos del Programa “Cinemática Inversa”.

Al igual que ocurre en el programa de la cinemática directa en la cinemática inversa también se han utilizado un número considerable de variables reales para realizar todas las operaciones aritméticas que supone su desarrollo.

##### Variables internas.

Cycle Execution	OB	1	OB	1
E0	M	45.0	BOOL	
E1	M	45.1	BOOL	
E2	M	45.2	BOOL	
E3	M	45.3	BOOL	
E4	M	45.5	BOOL	
E6	M	45.6	BOOL	
E7	M	45.7	BOOL	
E8	M	46.0	BOOL	
E9	M	46.1	BOOL	
E10	M	46.2	BOOL	
E11	M	46.3	BOOL	
E12	M	46.4	BOOL	
E13	M	46.5	BOOL	
E14	M	46.6	BOOL	
E15	M	46.7	BOOL	
E16	M	47.0	BOOL	
Calibrado_Automático	M	200.0	BOOL	
Calibrado_Manual	M	120.1	BOOL	
Conectar_Aire	M	44.7	BOOL	
Graba_0V_Manual	M	220.0	BOOL	
Graba_10V_Manual	M	44.6	BOOL	
M44.1	M	44.1	BOOL	
M44.2	M	44.2	BOOL	
pid	M	48.0	BOOL	
Reset_Manual	M	52.3	BOOL	

##### Variables analógicas en formato real.

MD20	MD	20	REAL
MD22	MD	22	REAL
PV_1	MD	100	REAL
Set-Point_1	MD	184	REAL
LMN_1	MD	105	REAL



## Calibración, Control y Diseño SCADA de un robot paralelo neumático con el autómatas S7-300.

---

---

Set-Point_2	MD	190	REAL
PV_2	MD	103	REAL
LMN_2	MD	110	REAL
Alfa	MD	56	REAL
Yp	MD	180	REAL
Xp	MD	196	REAL
d	MD	64	REAL
Beta	MD	82	REAL
Tita_2	MD	84	REAL
fi	MD	88	REAL
Xc	MD	94	REAL
Yc	MD	120	REAL
m	MD	128	REAL
d1(cm)	MD	130	REAL
d1(voltios)	MD	148	REAL
Xd	MD	132	REAL
Yd	MD	134	REAL
n	MD	144	REAL
d2(cm)	MD	146	REAL
d2(voltios)	MD	150	REAL

### Variables analógicas en formato palabra.

LMN_1 PALABRA	MW	10	WORD
LMN_2 PALABRA	MW	21	WORD

### Funciones utilizadas.

SCALE	FC	105	FC	105	Scaling Values	
UNSCALE	FC	106	FC	106	Unscaling Values	
CONT_C	SFB	41	SFB	41	Continuous Control	(Integrated Function, CPU 314 IFM)

### Entradas Digitales utilizadas:

Entrada de Referencia 1 (ER1)	E	0.0	BOOL
Entrada de Calibración 1 (EC1)	E	0.1	BOOL
Entrada de Referencia 2 (ER2)	E	0.2	BOOL
Entrada de Calibración 2 (EC1)	E	0.3	BOOL

### Salidas digitales utilizadas:

Salida de Referencia 1(SR1)	A	0.0	BOOL
Salida de Calibración 1(SC1)	A	0.1	BOOL
Salida de Referencia 2(SR2)	A	0.2	BOOL
Salida de Calibración 2(SC2)	A	0.3	BOOL

### Entradas analógicas utilizadas:

Convertidor 1(C1)	PEW	754	BOOL
-------------------	-----	-----	------



Convertidor 2(C2) PEW 756 BOOL

**Salidas analógicas utilizadas:**

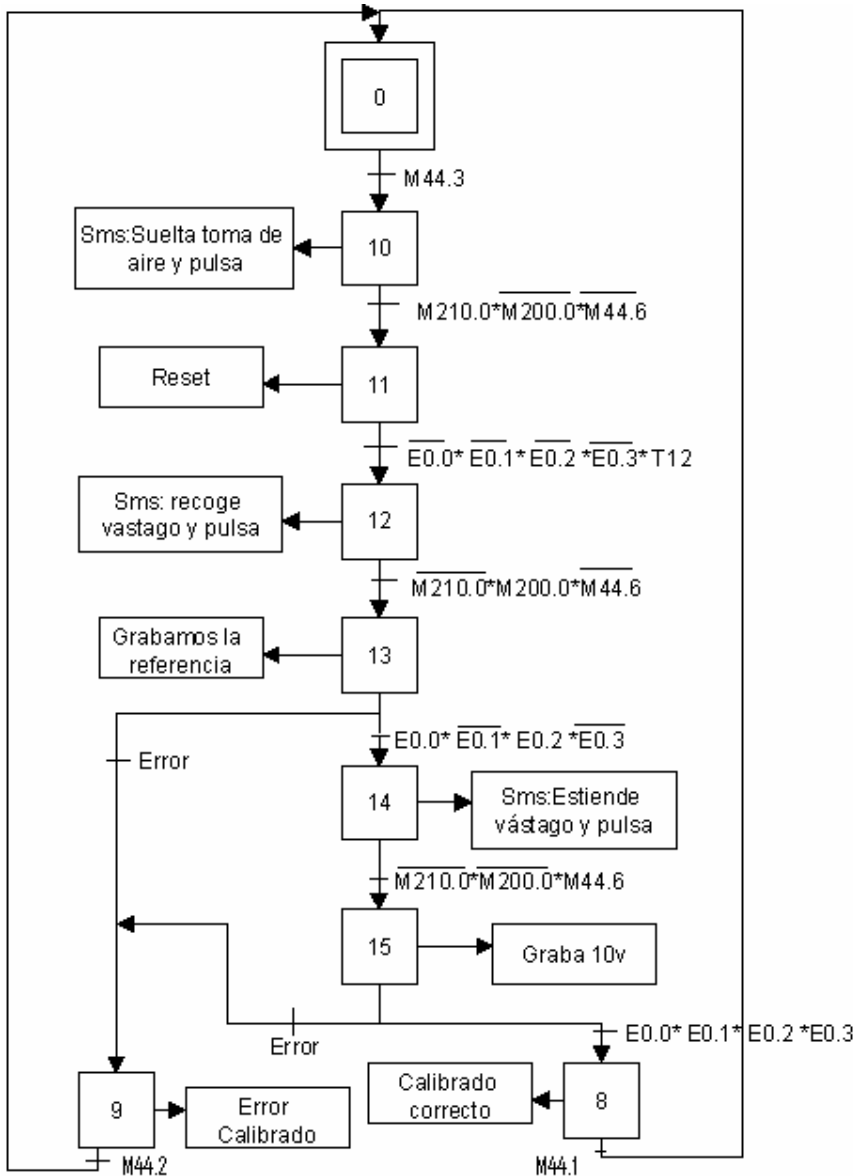
Válvula proporcional(SVP1) PAW 752 BOOL

Válvula proporcional(SVP2) PAW 754 BOOL

Al igual que las tablas de variables anteriores las variables Booleanas son las mismas. Las variables analógicas aumentan debido a las operaciones aritméticas que hemos realizado para el cálculo de la cinemática inversa.

### 3.5.2 GRAFCET nivel 2: Calibrado manual cinemática inversa.

A continuación se muestra el grafcet de nivel 2 del programa informático de la cinemática inversa.





### 3.5.3 Ecuaciones de activación desactivación “Cinemática Inversa”.

Las ecuaciones representadas a continuación han sido obtenidas de los graficet representados en los apartados 3.5.2 y 3.3.5.

$$\begin{aligned} E0 &= \overline{E1} * \overline{E2} * \overline{E3} * \overline{E4} * \overline{E5} * \overline{E6} * \overline{E7} * \overline{E8} + \overline{E9} \\ &+ \overline{E10} + \overline{E11} * \overline{E12} * \overline{E13} * \overline{E14} * \overline{E15} * \overline{E16} + E8 * M44.1 + E9 * M44.2 \\ E1 &= E0 * M200.0 + E1 * \overline{E2} \\ E2 &= E1 * M44.7 + E4 * E0.0 * \overline{E0.1} * E0.2 * \overline{E0.3} * \overline{T3} + E7 * \overline{T2} + E2 * \overline{E3} \\ &+ E2 * \overline{E6} + E2 * \overline{E8} + E2 * \overline{E9} \\ E3 &= E2 * E0.0 * \overline{E0.1} * E0.2 * \overline{E0.3} + E3 * \overline{E4} \\ E4 &= E3 * T1 + E4 * \overline{E5} + E4 * \overline{E2} \\ E5 &= E4 * E0.0 * E0.1 * E0.2 * E0.3 * T3 + E5 * \overline{E8} \\ E6 &= E2 * \overline{E0.0} * \overline{E0.1} * \overline{E0.2} * \overline{E0.3} + E6 * \overline{E7} \\ E7 &= E6 * T10 + E7 * \overline{E2} \\ E8 &= E2 * E0.0 * E0.1 * E0.2 * E0.3 + E5 * T4 + E15 * E0.0 * E0.1 * E0.2 \\ &* E0.3 + E8 * \overline{E0} \\ E9 &= E2 * \overline{E0.0} * E0.1 * \overline{E0.2} * E0.3 + E15 * \overline{E0.0} * E0.1 * \overline{E0.2} * E0.3 + \\ &E14 * \overline{E0.0} * \overline{E0.1} * \overline{E0.2} * \overline{E0.3} + E9 * \overline{E0} \\ E10 &= E0 * M120.0 + E10 * \overline{E11} \\ E11 &= E10 * M52.3 * \overline{M220.0} + E11 * \overline{E12} \\ E12 &= E11 * \overline{E0.0} * \overline{E0.1} * \overline{E0.2} * \overline{E0.3} * T12 + E12 * \overline{E13} \\ E13 &= E12 * \overline{M52.3} * \overline{M220.0} * \overline{M44.6} + E13 * \overline{E14} \\ E14 &= E13 * E0.0 * \overline{E0.1} * E0.2 * \overline{E0.3} * T13 + E14 * \overline{E15} + E14 * \overline{E8} \\ &+ E14 * \overline{E13} \\ E15 &= E14 * \overline{M52.3} * \overline{M220.0} * M44.6 + E15 * \overline{E8} + E15 * \overline{E9} \\ E16 &= E0 * M210.0 + E16 * \overline{E0} \end{aligned}$$



### 3.6 Funciones utilizadas.

Para realizar los programas informáticos de calibración en STEP 7 se han utilizado una serie de funciones las cuales se muestran a continuación:

#### 3.6.1 Desescalar valores" (UNSCALE): FC106.

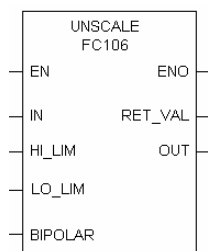


Figura 3. 3 Bloque Unscale (FC 106).

#### Descripción:

La función "Desescalar valores" (UNSCALE) toma en la entrada IN un valor real que está ajustado a escala en un rango comprendido entre un límite inferior y un límite superior (LO\_LIM y HI\_LIM), y lo convierte en un valor entero. El resultado se escribe en la salida OUT. La función UNSCALE aplica la fórmula siguiente:

$$OUT = [ ((IN-LO\_LIM)/(HI\_LIM-LO\_LIM)) * (K2-K1) ] + K1$$

Las constantes K1 y K2 se aplican de forma diferente, dependiendo de si el valor de entrada es BIPOLAR o UNIPOLAR.

**BIPOLAR:** Se supone que el valor entero de salida debe estar entre -27648 y 27648, por lo tanto:

$$K1 = -27648.0 \text{ y } K2 = +27648.0$$

**UNIPOLAR:** Se supone que el valor entero de salida debe estar entre 0 y 27648, por lo tanto:

$$K1 = 0.0 \text{ y } K2 = +27648.0$$

Si el valor real de entrada queda fuera del rango comprendido entre los límites inferior y superior de su tipo (BIPOLAR o UNIPOLAR), la salida (OUT) se fija al valor límite más cercano (a LO\_LIM o a HI\_LIM), y se devuelve un error.

#### Parámetros de la función UNSCALE (FC106).

**EN** Entrada:

BOOL E, A, M, D, L La entrada de habilitación con estado de señal 1 activa el cuadro.

**ENO** Salida:





**BOOL E, A, M, D, L** La salida de habilitación tiene el estado de señal 1 si la función se ejecuta sin errores.

**IN** Entrada:

**REAL E, A, M, D, L, P**, o constante Valor de entrada a desescalar convirtiéndolo en un valor entero.

**HI\_LIM** Entrada:

**REAL E, A, M, D, L, P**, o constante Límite superior del rango escalar.

**LO\_LIM** Entrada:

**REAL E, A, M, D, L, P**, o constante Límite inferior del rango escalar.

**BIPOLAR** Entrada:

**BOOL E, A, M, D, L** El estado de señal 1 indica que el valor de entrada es bipolar; con el estado de señal 0 indica que es unipolar.

**OUT** Salida:

**INT E, A, M, D, L, P** Resultado del desescalado.

**RET\_VAL** Salida:

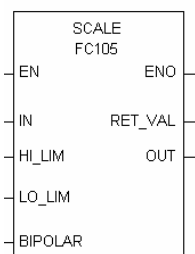
**WORD E, A, M, D, L, P** Da el valor W#16#0000 cuando la función se ejecuta sin errores; si los valores son distintos de W#16#0000, véase la información sobre errores.

Información sobre errores.

Si el valor real de entrada queda fuera del rango comprendido entre los límites inferior y superior de su tipo (**BIPOLAR** o **UNIPOLAR**), la salida (**OUT**) se une al valor límite más cercano (a **LO\_LIM** o a **HI\_LIM**), y se devuelve un error. El estado de señal de la salida de habilitación (**ENO**) se pone a 0, y el valor de respuesta (**RET\_VAL**) toma el valor W#16#0008.

### 3.6.2 Escalar valores (**SCALE**): **FC105**.

En la figura 3.4 se muestra la representación del bloque FC105 en kop (diagrama de contactos)



**Figura 3. 4 Bloque Scale (FC 105).**



## Descripción.

La función "Escalar valores" (SCALE), toma un valor entero en la entrada IN y lo convierte en un valor real, convirtiéndolo a escala en un rango comprendido entre un límite inferior y un límite superior (LO\_LIM y HI\_LIM). El resultado se escribe en la salida OUT. La función SCALE aplica la fórmula siguiente:

$$\text{OUT} = [ ((\text{FLOAT}(\text{IN}) - \text{K1}) / (\text{K2} - \text{K1})) * (\text{HI\_LIM} - \text{LO\_LIM}) ] + \text{LO\_LIM}$$

Las constantes K1 y K2 se aplican de forma diferente, dependiendo de si el valor de entrada es BIPOLAR o UNIPOLAR.

**BIPOLAR:** Se supone que el valor entero de entrada debe estar entre -27648 y 27648, por lo tanto, K1 = -27648.0 y K2 = +27648.0

**UNIPOLAR:** Se supone que el valor entero de entrada debe estar entre 0 y 27648, por lo tanto, K1 = 0.0 y K2 = +27648.0

Si el valor entero de entrada es **mayor que K2**, la salida (OUT) se une a HI\_LIM y se indica un error.

Si el valor entero de entrada es **menor que K1**, la salida se une a LO\_LIM y se indica un error.

Se puede efectuar la conversión escalar inversa programando los límites de tal forma que el límite inferior sea mayor que el límite superior (LO\_LIM > HI\_LIM). En la conversión escalar inversa el valor de la salida disminuye cuando aumenta el valor de la entrada.

## Parámetros de la función SCALE (FC105).

### EN Entrada:

BOOL E, A, M, D, L La entrada de habilitación con estado de señal 1 activa el cuadro.

### ENO Salida:

BOOL E, A, M, D, L La salida de habilitación tiene el estado de señal 1 si la función se ejecuta sin errores.

### IN Entrada:

INT E, A, M, D, L, P, o constante Valor de entrada a convertir a escala en valor REAL.

### HI\_LIM Entrada:

REAL E, A, M, D, L, P, o constante Límite superior del rango escalar.

### LO\_LIM Entrada:

REAL E, A, M, D, L, P, o constante Límite inferior del rango escalar.



**BIPOLAR** Entrada:

BOOL E, A, M, D, L El estado de señal 1 indica que el valor de entrada es bipolar; con el estado de señal 0 indica que es unipolar.

**OUT** Salida:

REAL E, A, M, D, L, P Resultado de la conversión a escala.

**RET\_VAL** Salida:

WORD E, A, M, D, L, P Da el valor W#16#0000 cuando la función se ejecuta sin errores; si los valores son distintos de W#16#0000, véase la información sobre errores.

**Información sobre errores.**

Si el valor entero de entrada es mayor que K2, la salida (OUT) se une a HI\_LIM y se indica un error. Si el valor entero de entrada es menor que K1, la salida se une a LO\_LIM y se indica un error. El estado de señal de la salida de habilitación (ENO) se pone a 1 y el valor de respuesta (RET\_VAL) toma el valor W#16#0008.



### 3.6.3 S\_VIMP: Parametrizar y arrancar temporizador como impulso prolongado.

Descripción de la operación:

**Símbolo**



<u>Parámetro Inglés</u>	<u>Parámetro Alemán</u>	<u>Tipo de datos</u>	<u>Area de memoria</u>	<u>Descripción</u>
N.º de T	N.º de T	TIMER	T	Número de identificación del temporizador, el área varía según la CPU que se utilice
S	S	BOOL	E, A, M, L, D	Entrada de arranque
TV	TW	S5TIME	E, A, M, L, D	Valor de temporización predeterminado
R	R	BOOL	E, A, M, L, D	Entrada de desactivación
BI	DUAL	WORD	E, A, M, L, D	Valor de temporización actual, codificado en binario
BCD	DEZ	WORD	E, A, M, L, D	Tiempo restante, formato BCD
Q	Q	BOOL	E, A, M, L, D	Estado del temporizador

**Figura 3.5 S\_VIMP.**

S\_VIMP (Parametrizar y arrancar temporizador como impulso prolongado): arranca el temporizador indicado cuando hay un flanco creciente en la entrada de arranque S. Para arrancar un temporizador tiene que producirse necesariamente un cambio de señal. El temporizador continúa en marcha durante el tiempo predeterminado -indicado en la entrada TV/TW-, aunque el estado de señal en la entrada S se ponga a "0" antes de haber transcurrido el intervalo de tiempo. El estado de señal en la salida Q es "1" mientras el temporizador esté en marcha. El temporizador vuelve a arrancar con el valor de temporización predeterminado si el estado de señal en la entrada S cambia de "0" a "1" mientras está en marcha el temporizador.

El temporizador se pone a 0 si la entrada de desactivación R del temporizador se pone a "1" mientras el temporizador está funcionando. El valor de temporización actual y la base de tiempo se ponen a 0.

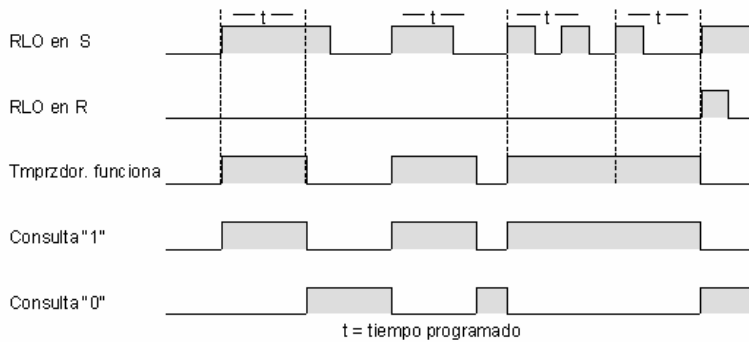
El valor de temporización actual queda depositado en las salidas BI/DUAL y BCD/DEZ. El valor de temporización en la salida BI/DUAL está en código binario, el valor en la salida BCD/DEZ está en formato decimal codificado en binario. El valor de temporización actual equivale al valor inicial



de TV/TW menos el valor de temporización que ha transcurrido desde el arranque del temporizador.

Consulte también Área de memoria y componentes de un temporizador.

Diagrama de temporización.



**Figura 3. 6 Diagrama de temporización**



### 3.6.4 S\_Everz: Parametrizar y arrancar temporizador como retardo a la conexión.

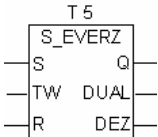


Figura 3. 7 S\_Everz..

#### Descripción de la operación:

S\_EVERZ (Parametrizar y arrancar temporizador como retardo a la conexión) arranca el temporizador indicado cuando hay un flanco creciente en la entrada de arranque S. Para arrancar un temporizador tiene que producirse necesariamente un cambio de señal. El temporizador continúa en marcha con el valor de temporización indicado en la entrada TV/TW mientras el estado de señal en la entrada S sea positivo. El estado de señal en la salida Q será "1" si el tiempo ha transcurrido sin que se produjeran errores y si el estado de señal en la entrada S es "1". Si el estado de señal en la entrada S cambia de "1" a "0" mientras está en marcha el temporizador, éste se para. En este caso, el estado de señal en la salida Q será "0".

El temporizador se pone a 0 si la entrada de desactivación R del temporizador se pone a "1" mientras funciona el temporizador. El valor de temporización y la base de tiempo se ponen a 0. Entonces el estado de señal en la salida Q es "0". El temporizador también se pone a 0 si en la entrada de desactivación R el valor es "1", mientras el temporizador no está en marcha y el RLO en la entrada S es "1".

El valor de temporización actual queda depositado en las salidas BI/DUAL y BCD/DEZ. El valor de temporización en la salida BI/DUAL está en código binario, el valor en la salida BCD/DEZ está en formato decimal codificado en binario. El valor de temporización actual equivale al valor inicial de TV/TW menos el valor de temporización que ha transcurrido desde el arranque del temporizador.

#### Diagrama de temporización.

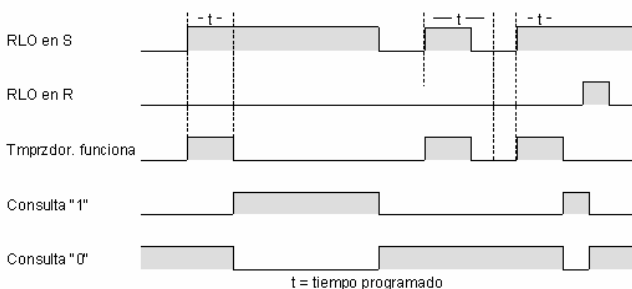


Figura 3. 8 Diagrama de temporización 1.

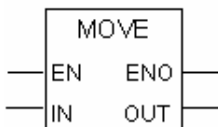


**Palabra de estado**

	<u>RB</u>	<u>A1</u>	<u>A0</u>	<u>OV</u>	<u>OS</u>	<u>OR</u>	<u>STA</u>	<u>RLO</u>	<u>/ER</u>
se escribe:	-	-	-	-	-	x	x	x	1

### 3.6.5 MOVE Asignar un valor.

**Simbolo**



<u>Parámetro</u>	<u>Tipo de datos</u>	<u>Area de memoria</u>	<u>Descripción</u>
EN	BOOL	E, A, M, L, D	Entrada de habilitación
ENO	BOOL	E, A, M, L, D	Salida de habilitación
IN	Todos los <u>tipos de datos simples</u> con una longitud de 8, 16 o 32 bits	E, A, M, L, D o constante	Valor de fuente
OUT	Todos los tipos de datos simples con una longitud de 8, 16 o 32 bits	E, A, M, L, D	Dirección de destino

**Figura 3. 9 Move.**

**Descripción de la operación.**

MOVE (Asignar un valor) es activada por la entrada de habilitación EN. El valor indicado por la entrada IN se copia en la dirección que la salida OUT. La salida de habilitación ENO tiene el mismo estado de señal que la entrada de habilitación EN. La operación MOVE sólo puede copiar los objetos de datos que tengan las longitudes de BYTE, WORD o de DWORD. Los tipos de datos de usuario tales como los arrays o las estructuras han de copiarse con SFC 20 „BLKMOV“.

Palabra de estado:

RB	A1	A0	OV	OS	OR	STA	RLO	/ER
se escribe:	1	-	-	-	-	0	1	1



Dependencia del MCR (Master Control Relay).

La dependencia del MCR solamente se activa si el cuadro MOVE se coloca dentro de un área de MCR activa. En área MCR los datos direccionados se copian tal como se ha descrito arriba, siempre que el MCR esté conectado y haya corriente en la entrada de habilitación, el bit direccionado se pone a "1" (se activa) ó a "0" (se desactiva), tal como se ha descrito más arriba. Si el MCR está desconectado y se ejecuta una operación MOVE, en la dirección indicada por OUT se escribirá siempre el valor "0", independientemente de cuál sea el estado actual de IN.

**Nota:**

Al transferir un valor a un tipo de datos de longitud diferente los bytes más significativos se truncan o se rellenan con ceros si es preciso:

<b><u>Palabra doble</u></b>	<b><u>1111 1111</u></b>	<b><u>0000 1111</u></b>	<b><u>1111 0000</u></b>	<b><u>0101 0101</u></b>
<b><u>Transferencia</u></b>	<b><u>Resultado</u></b>			
a una palabra doble:	1111 1111	0000 1111	1111 0000	0101 0101
a un byte:				0101 0101
a una palabra:			1111 0000	0101 0101
<b><u>Byte</u></b>				<b><u>1111 0000</u></b>
<b><u>Transferencia</u></b>	<b><u>Resultado</u></b>			
a un byte:				1111 0000
a una palabra:			0000 0000	1111 0000
a una palabra doble:	0000 0000	0000 0000	0000 0000	1111 0000





# Capítulo 4

## *Programación del control de posición.*



## 4.1 Introducción.

Una aproximación experimental al control de un proceso mediante PID supone la sintonía por métodos experimentales que a través de un experimento nos proporcionan unos parámetros que indican la dinámica del proceso. Estos parámetros serán los valores de entrada a unas tablas con los valores del PID para una sintonía adecuada.

Todas las gráficas representadas en este capítulo el eje **X** representa **Tiempo (segundos)** y el eje **Y**, **Voltios**.

## 4.2 Sintonía mediante Ziegler-Nichols (1 cilindro).

Ziegler y Nichols propusieron unas reglas para determinar los valores de la ganancia proporcional  $K_p$ , del tiempo integral  $T_i$  y del tiempo derivativo  $T_d$ , con base en las características de respuesta transitoria de una planta específica.

Existen dos métodos denominados reglas de sintonización de Ziegler-Nichols. En ambos se pretende obtener un 25% de sobrepaso máximo en la respuesta escalón.

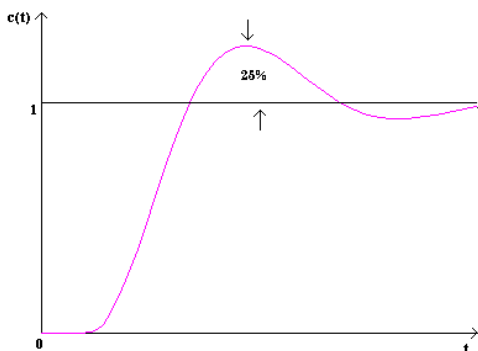


Figura 4. 1 Sintonía mediante Ziegler-Nichols.

### 4.2.1 Primer método (mediante lazo abierto).

En el primer método, la respuesta de la planta a una entrada escalón unitario se obtiene de manera experimental. Si la planta no contiene integradores ni polos dominantes complejos conjugados, la curva de respuesta escalón unitario puede tener forma de S, como se observa en la ilustración 4.1. Si la respuesta no exhibe una curva con forma de S, este método no es pertinente. Tales curvas de respuesta escalón se generan experimentalmente o a partir de una simulación dinámica de la planta.

La curva con forma de S se caracteriza por dos parámetros: el **tiempo de retardo  $L$**  y la **constante de tiempo  $T$** . El tiempo de retardo y la constante de tiempo se determinan dibujando una recta tangente en el punto de inflexión de la curva con forma de S y determinando las intersecciones de esta tangente con el eje del tiempo y la línea  $c(t)=K$ , como se aprecia en la ilustración 4.2. En este caso, la función de transferencia  $C(s)/U(s)$  se aproxima mediante un sistema de primer orden con un retardo de transporte del modo siguiente:

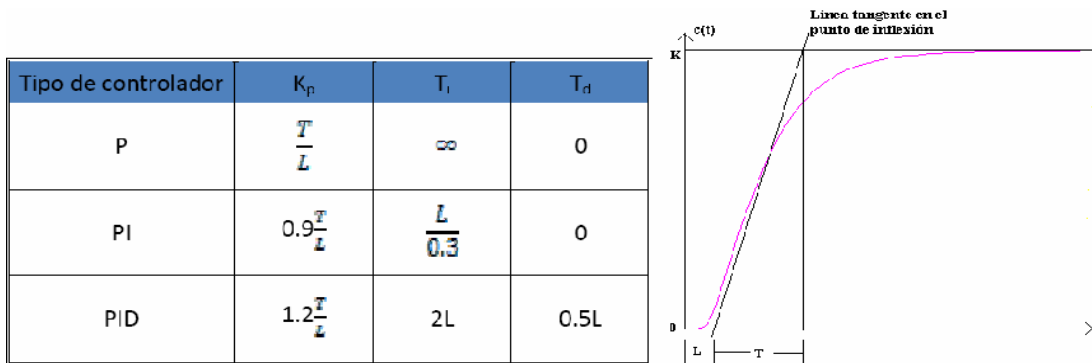


Figura 4. 2 Tipo de controlador. 1

Las gráficas que se muestran a continuación no son muy claras, por lo tanto a la hora de trazar la recta tangente no se puede realizar con precisión, y son muy difíciles de explicar por lo tanto no vamos a realizar este experimento.

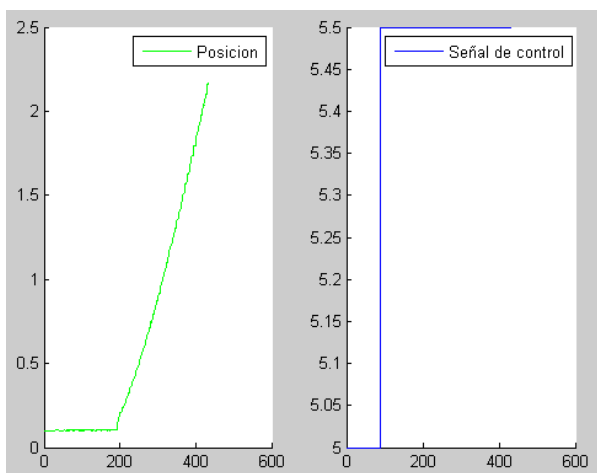
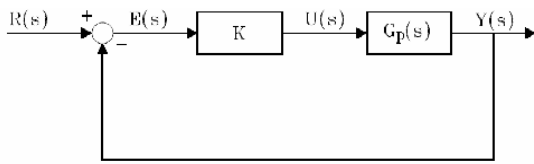


Figura 4. 3 Prueba 1.

### 4.2.2 Segundo método (mediante lazo cerrado).

En el segundo método, primero establecemos  $T_i = \infty$  y  $T_d = 0$ . Usando sólo la acción de control proporcional, se incrementa  $K_p$  de 0 a un valor crítico  $K_{cr}$  en donde la salida exhiba primero oscilaciones sostenidas. Por tanto, la ganancia crítica  $K_{cr}$  y el periodo  $P_{cr}$  correspondiente se determinan experimentalmente. Si la salida no presenta oscilaciones sostenidas para cualquier valor que pueda tomar  $K_p$ , no se aplica este método. Ziegler-Nichols sugirieron que se establecieran los valores de los parámetros  $K_p$ ,  $T_i$  y  $T_d$  de acuerdo con la fórmula que aparece en la figura 4.4.



Tipo de controlador	$K_p$	$T_i$	$T_d$
P	$0.5K_{cr}$	$\infty$	0
PI	$0.45K_{cr}$	$\frac{1}{1.2}P_{cr}$	0
PID	$0.6K_{cr}$	$0.5P_{cr}$	$0.125P_{cr}$

Figura 4. 4 Tipo de controlador 2.

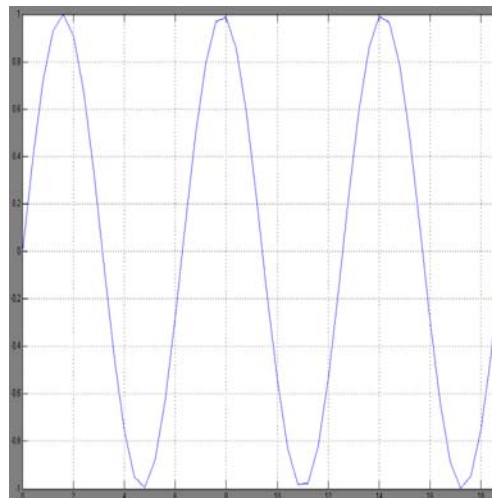


Figura 4. 5 Oscilaciones sostenidas.

Realizamos el experimento en lazo cerrado, aumentado el valor de  $K_p$  y vemos que a partir de un valor de ganancia  $K_p = 5$  se consigue obtener oscilaciones sostenidas; pero si seguimos aumentando la ganancia las oscilaciones se mantienen en vez de llegar a inestabilizar el sistema.

Esto ocurre cuando las ramas del lugar de las raíces no cortan al eje  $jw$ , en estos casos este método no se puede utilizar; también puede ocurrir que no se alcancen oscilaciones sostenidas para ningún valor que pueda tomar  $K_p$ , debido a la saturación del actuador, en este caso tampoco se puede utilizar el segundo método de Ziegler-Nichols.

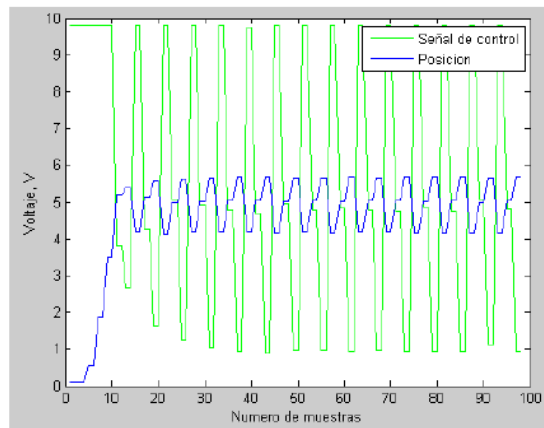
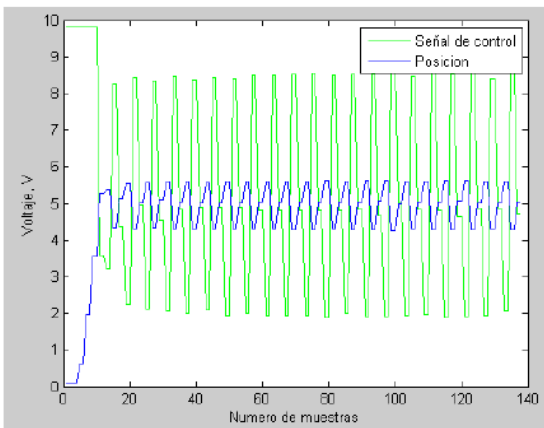


Figura 4. 6 La salida siempre es acotada para cualquier entrada acotada. No podemos utilizar el segundo método de ZIEGLER-NICHOLS.

### Comentarios.

Dado que en este caso los métodos de sintonía de Ziegler Nichols no son adecuados hemos optado por buscar otros métodos de sintonía como son:

### 4.3 Sintonía mediante HARRIOT.

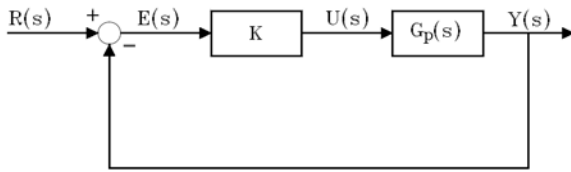


Figura 4. 7 Planta controlada mediante acción proporcional.

Este método de sintonía es similar al segundo método de Ziegler-Nichols, aunque sin necesidad de llevar el sistema a la inestabilidad, por lo que se puede aplicar en plantas con funciones de transferencia que presente un término integrador, a diferencia de los métodos de Ziegler-Nichols.

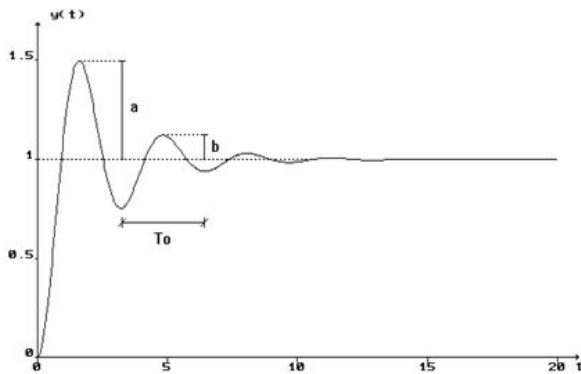


Figura 4. 8 Sintonía mediante HARRIOT (Oscilaciones sostenidas).1.

La relación entre el primer y segundo sobrepaso debe ser igual a 0.25, esto es:

$$\frac{b}{a} = 0,25$$

Las fórmulas de sintonización son en este caso:

Tipo de controlador	$K_p$	$T_i$	$T_d$
P	$K_0$	-	-
PI	$K_0$	$T_0$	-
PID	$K_0$	$\frac{T_0}{1.5}$	$\frac{T_0}{6}$

Figura 4. 9 Sintonía mediante el método de HARRIOT (Oscilaciones sostenidas).2.



## Calibración, Control y Diseño SCADA de un robot paralelo neumático con el autómata S7-300.

Posteriormente se necesitará una reducción del valor de  $K_p$ , siendo el adecuado aquél para el que la relación sea aproximadamente 0.25 manteniendo los valores de las variables integral y derivativa, que nos ha dado la tabla, inalterados.

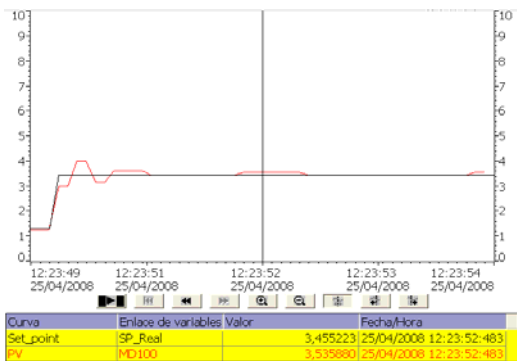
En el robot el valor proporcional con el que se ha cumplido la relación de 0.25 ha sido  $K_p = 4,3$  este será el valor adecuado para un controlador P.

Siendo:

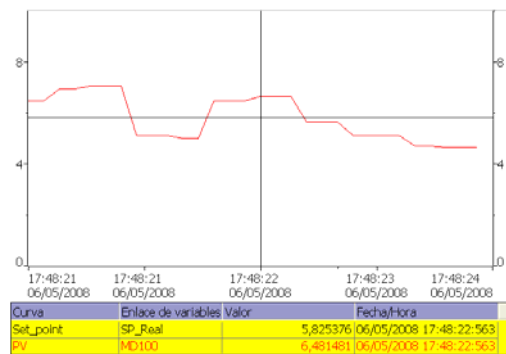
$$K_0 = 4,3$$

$$T_0 = 0.55$$

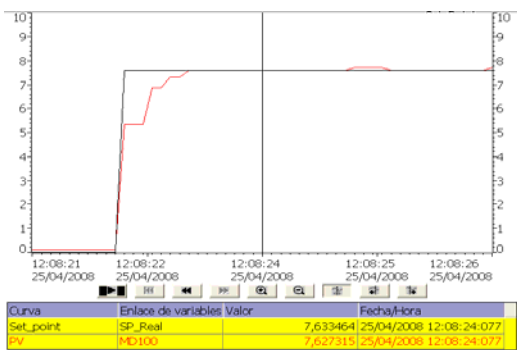
Se han realizado varios experimentos de control con sintonías próximas a las aportadas por este método con resultados que se describen a continuación:



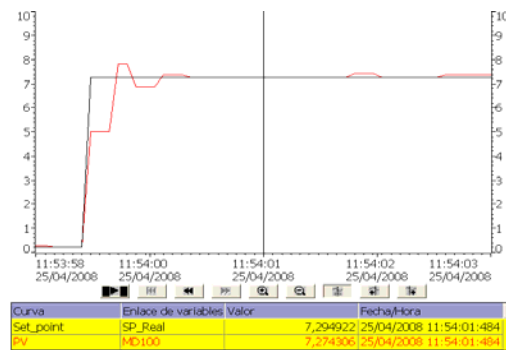
a)



b)



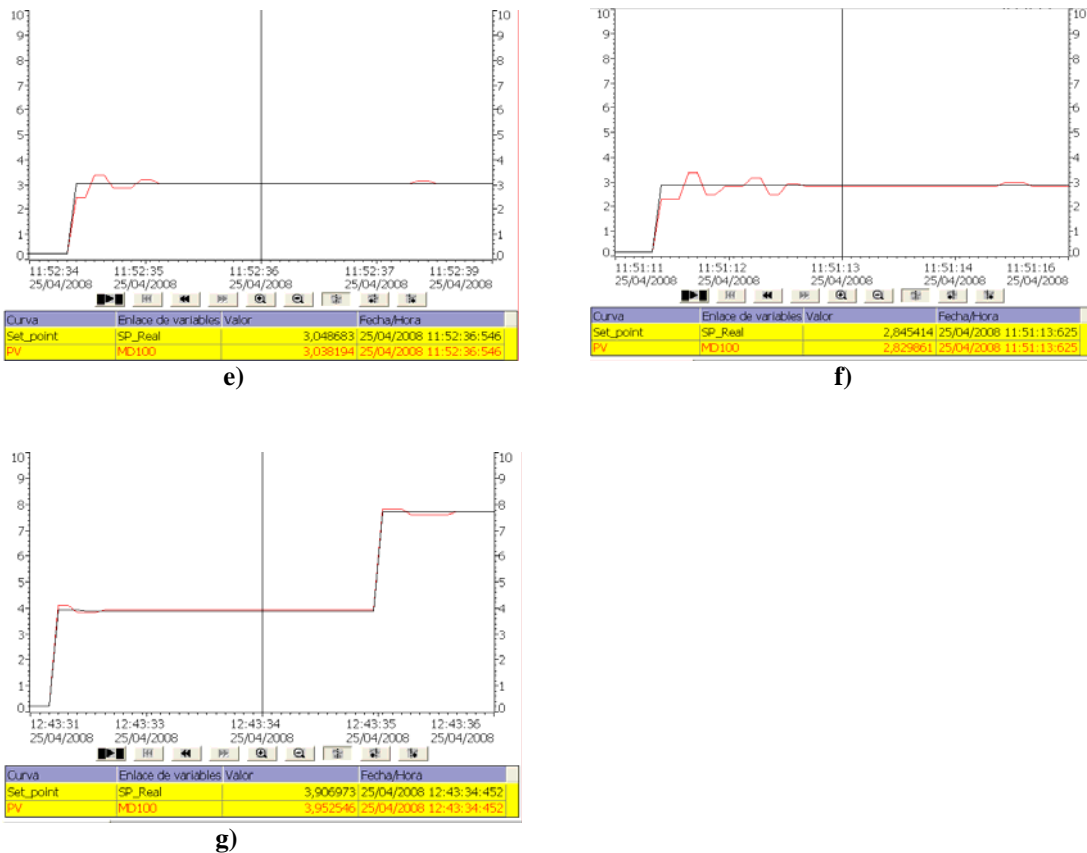
c)



d)



## Calibración, Control y Diseño SCADA de un robot paralelo neumático con el autómat S7-300.



**Figura 4. 10:** a)  $K_p= 4.3$ ; b)  $K_p=4,3$  ,  $T_i=0,55s$ ; c)  $K_p=3$ ;  $T_i=100$ ; d)  $K_p=4$  ,  $T_i=100$ ; e)  $K_p=4.1$  ,  $T_i=100$  f)  $K_p=4.3$  ,  $T_i=100$ ; g)  $K_p=3$  ,  $T_i=550$ .

Con los resultados que hemos obtenido al realizar el método de HARRIOT el sistema posee muchas oscilaciones, la respuesta no es la esperada, por lo tanto no nos va a ser útil para la sintonización de nuestro controlador.

Al aumentar el término integral la respuesta nos va mejorando.

En algunos casos los datos que dan estos métodos no son útiles para la sintonización, debido a esto vamos a optar por sintonizar mediante técnicas de ensayo-error que son un poco más laboriosas pero llegamos a conseguir una respuesta bastante buena.



## 4.4 Sintonía mediante Ensayo-Error (1 cilindro).

En este apartado vamos a ir modificando los parámetros del controlador hasta conseguir que la variable del proceso siga lo mas rápido y preciso posible a la consigna, lo vamos a realizar con un cilindro neumático como se muestra en la figura 4.11. y los distintos experimentos realizados los podemos observar en la figura 4.12.

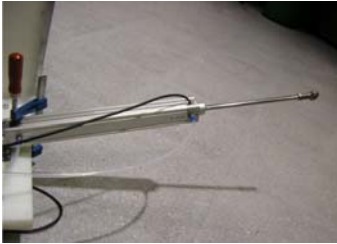


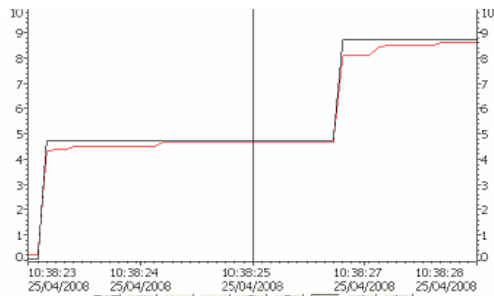
Figura 4. 11 Un cilindro.

### 4.4.1 Control proporcional (P).

Con un valor de  $K_p=1$ , obtenemos una respuesta muy buena, pero en nuestro caso vamos a escoger el valor de  $K_p= 3$  como un valor rápido y preciso.

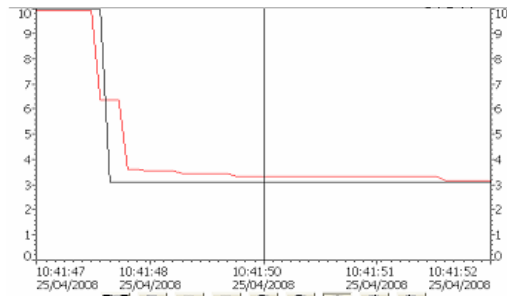
$K_p=4$  también es bueno para el control de un cilindro, aunque el cilindro realiza unas oscilaciones que podrían ser un inconveniente si se requiere un control preciso.

Con  $K_p= 5$  el sistema se descontrola. Por lo tanto seguiremos con una  $K_p= 4$  (como máximo) y procederemos a modificar los parámetros integral y derivativo.



Curva	Enlace de variables	Valor	Fecha/Hora
Set_point	SP_Reel	4,720052	25/04/2008 10:38:25:906
PV	IND100	4,664352	25/04/2008 10:38:25:906

a)



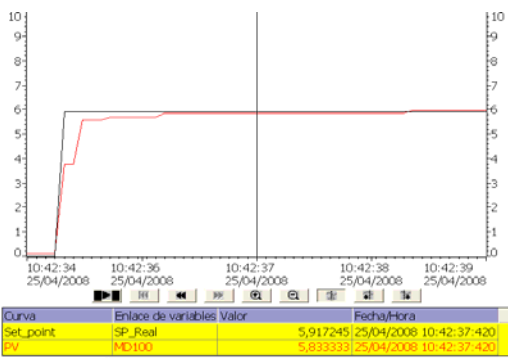
Curva	Enlace de variables	Valor	Fecha/Hora
Set_point	SP_Reel	3,071470	25/04/2008 10:41:50:202
PV	IND100	3,110165	25/04/2008 10:41:50:202

b)

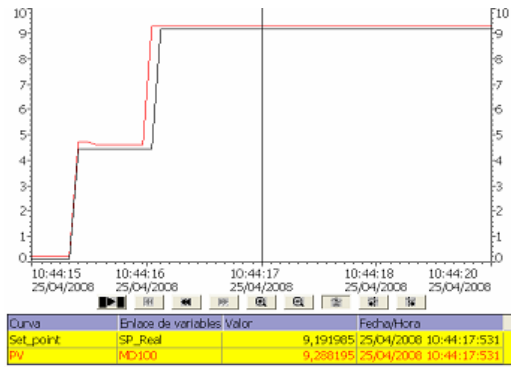




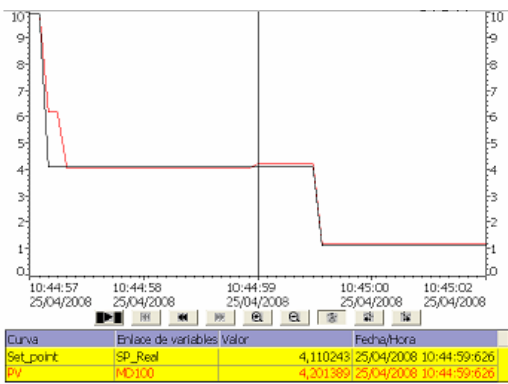
# Calibración, Control y Diseño SCADA de un robot paralelo neumático con el autómat S7-300.



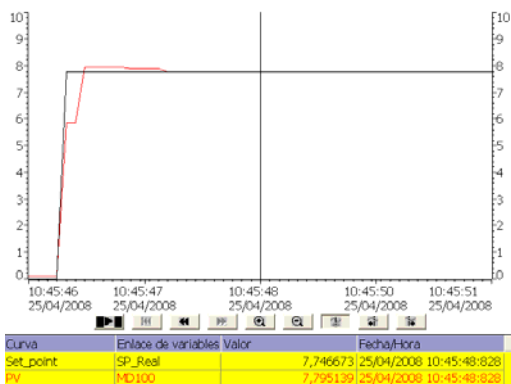
c)



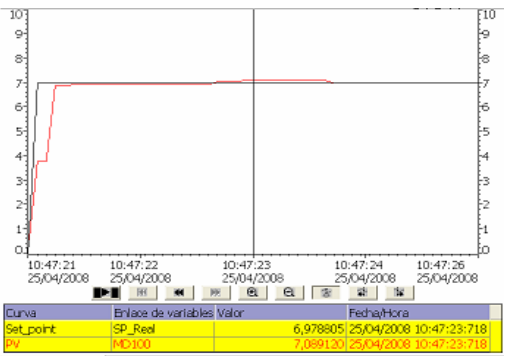
d)



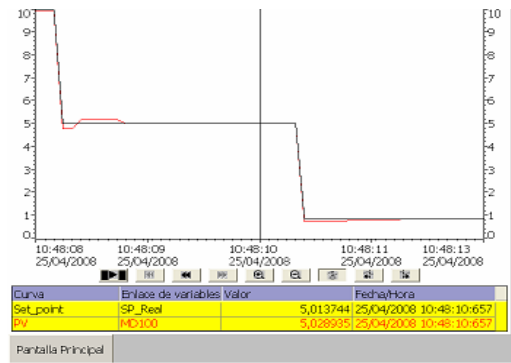
e)



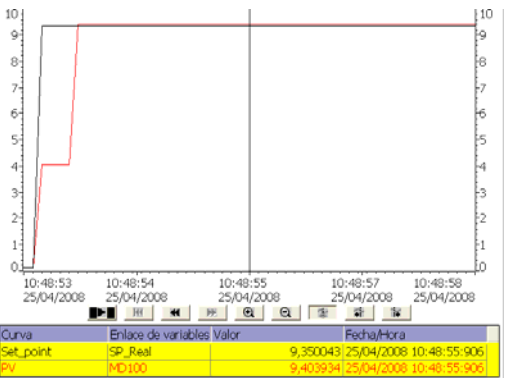
f)



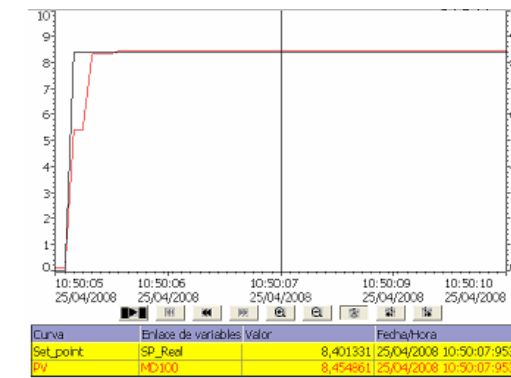
g)



h)



i)



j)

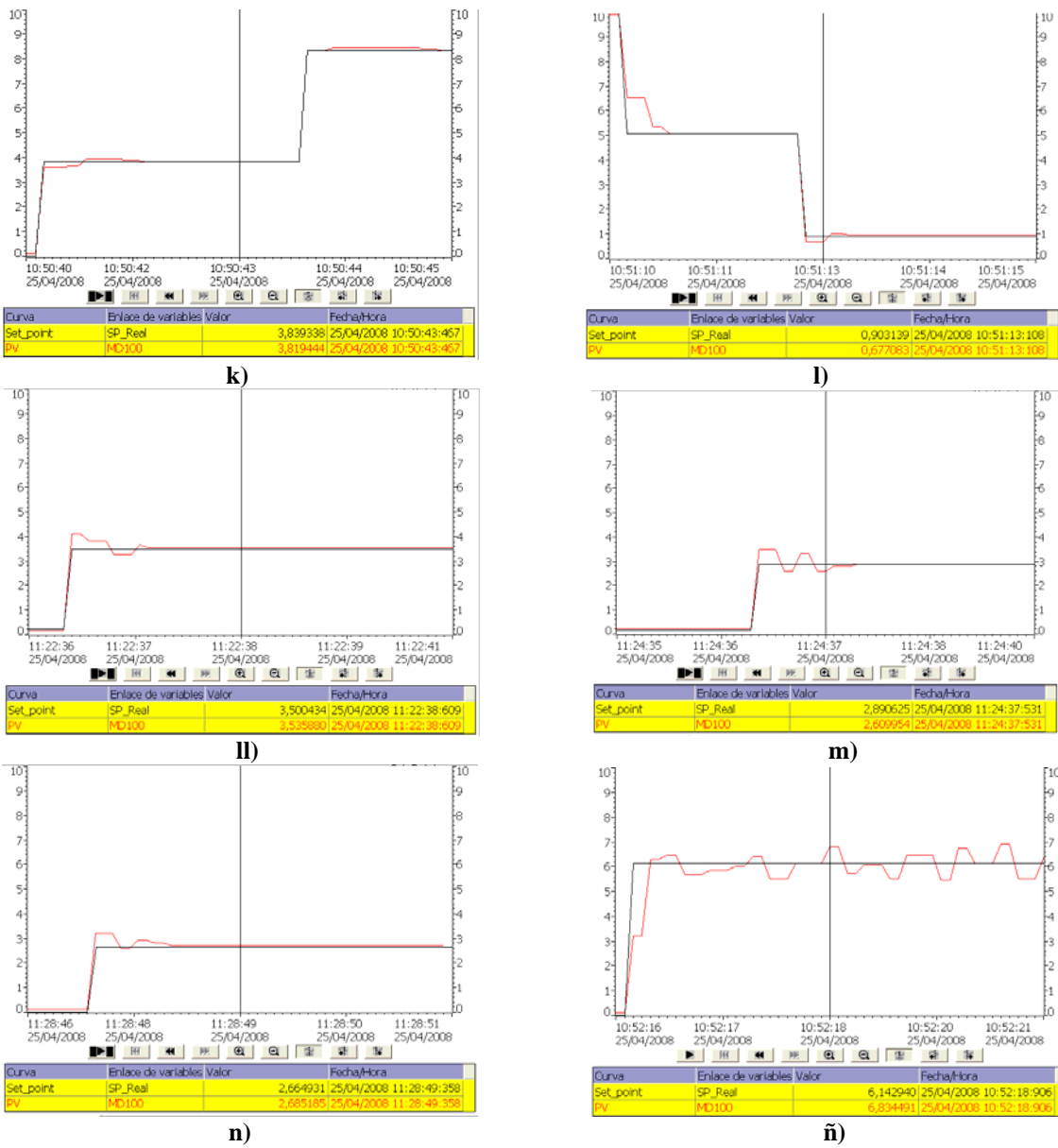


Figura 4. 12 : a)  $K_p=1$  ; b)  $K_p=1$  ; c)  $K_p=1$  ; d)  $K_p=2$  ; e)  $K_p=2$  ; f)  $K_p=2$  ; g)  $K_p=3$  ; h)  $K_p=3$  ; i)  $K_p=3$  ; j)  $K_p=4$  ; k)  $K_p=4$  ; l)  $K_p=4$  ; ll)  $K_p=4,4$  ; m)  $K_p=4,6$  ; n)  $K_p=4,7$  ; ñ)  $K_p=5$ .

#### 4.4.2 Control proporcional derivativo (PD)

$K_p=3$   $T_d=10$  son unos parámetros muy buenos para el control de un cilindro, obtenemos una respuesta casi perfecta figura 4.1.3 c), d), e).

Como se puede comprobar, con los valores introducidos en la figura 4.13 f) el cilindro se descontrola, por lo tanto vamos a proceder a introducir el *Tiempo Integral*.



## Calibración, Control y Diseño SCADA de un robot paralelo neumático con el autómeta S7-300.

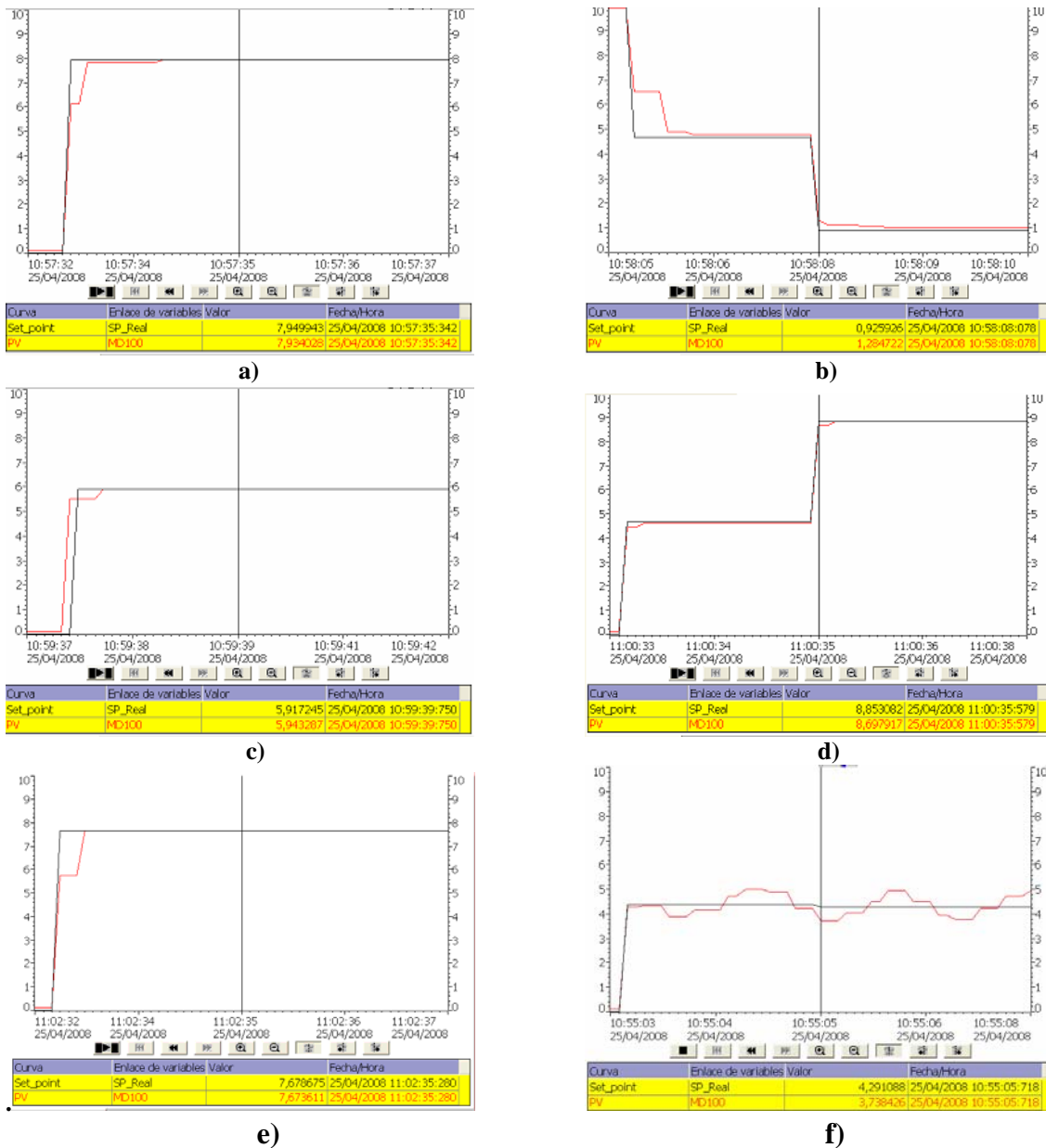


Figura 4. 13: a)  $K_p=2$ ,  $T_d=10s$  ; b)  $K_p=2$ ,  $T_d=10s$  ; c)  $K_p=3$ ,  $T_d=10s$  ; d)  $K_p=3$ ,  $T_d=10s$  ; e)  $K_p=3$ ,  $T_d=10s$  ; f)  $K_p=4$ ,  $T_d=10s$ .

Si observamos las figuras 4.13 a), b) veremos que la variable de proceso sigue muy bien al set-point.



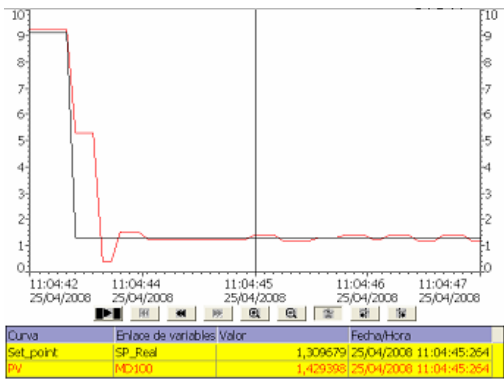
### 4.4.3 Control proporcional integral derivativo (PID).

Vemos como la variable de proceso oscila en torno a la consigna, de una forma muy brusca, figuras 4,14 (a, (b, (c, (d esto se debe, al término integral y a la zona muerta del actuador.

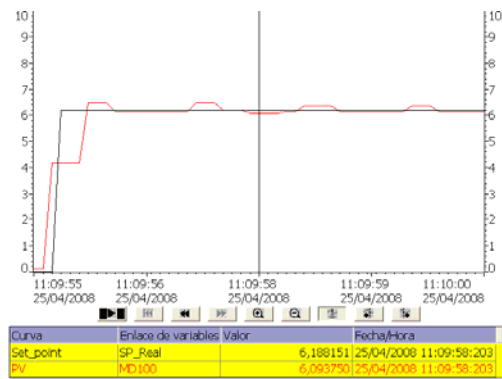
Hemos ido reduciendo el valor de la constante proporcional ( $K_p$ ) para ver si conseguimos reducir los picos y que las oscilaciones sean un poco más pequeñas.

En la gráfica de la figura 4.14 f) ya podemos notar la diferencia con respecto a las anteriores, la oscilación de la variable de proceso en torno a la consigna se ha reducido y la respuesta a mejorado notoriamente, pero aún no nos valen estos valores de sintonía, por lo tanto se ha seguido aumentando el término integral como se muestra en la figura 4.14 g). Se ha dejado de aumentar el término integral, ya que con estos valores tan grandes no tiene casi efecto en la respuesta.

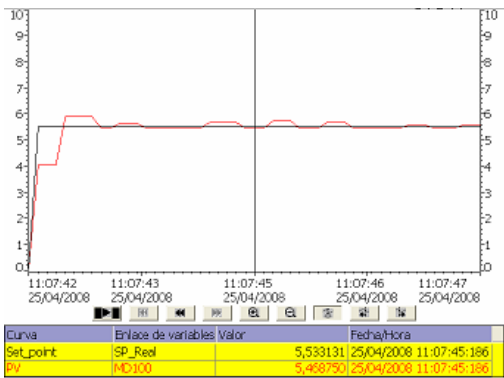
Como podemos observar (figura 8.27), el sistema responde de una forma mucho más eficaz cuando posee un PD (Proporcional derivativo), también responde de una forma muy eficaz solo con término proporcional ya que el cilindro es un integrador nos elimina casi por completo el error estacionario (figura 8.19).



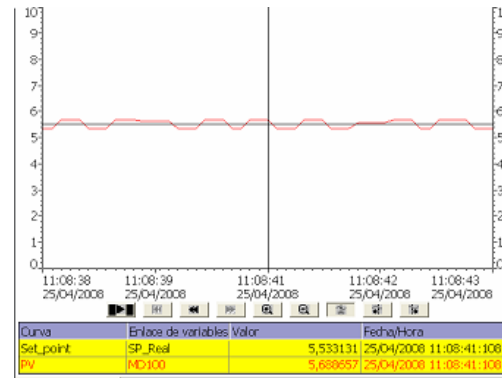
a)



b)



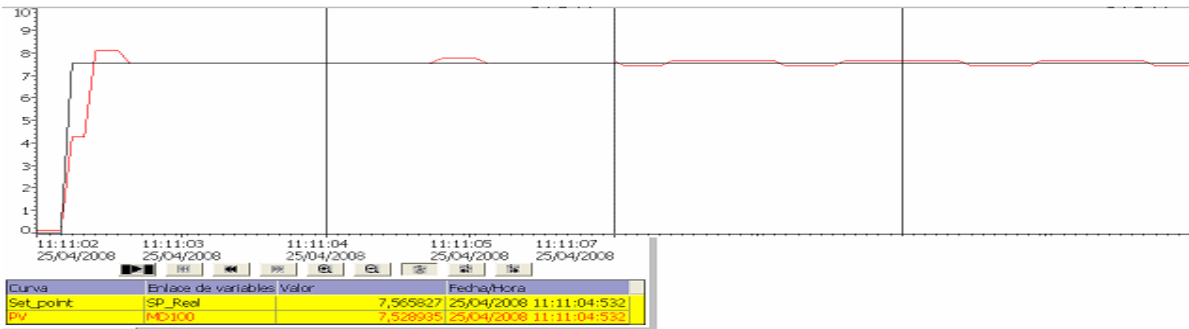
c)



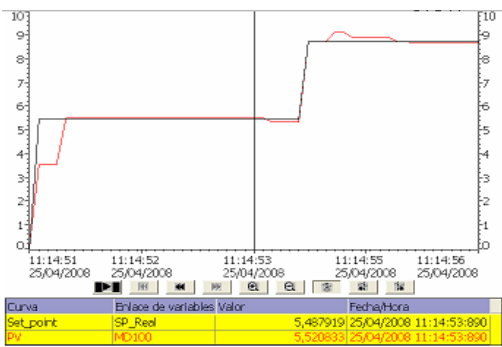
d)



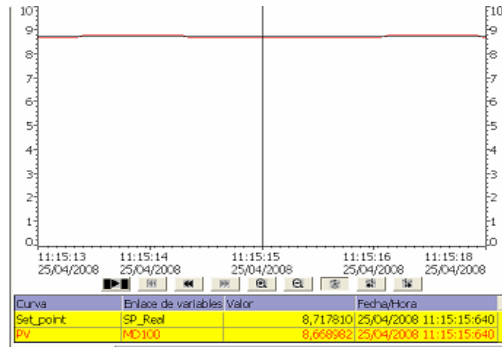
Calibración, Control y Diseño SCADA de un robot paralelo neumático con el autómeta S7-300.



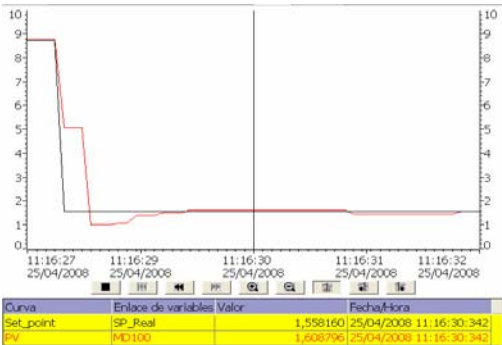
e)



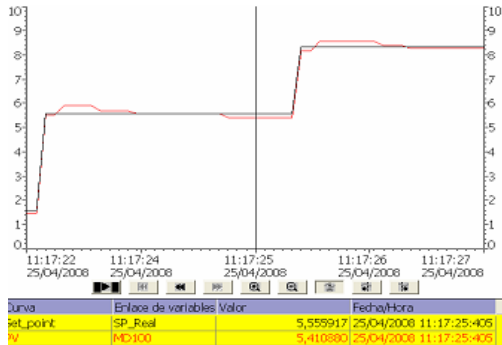
f)



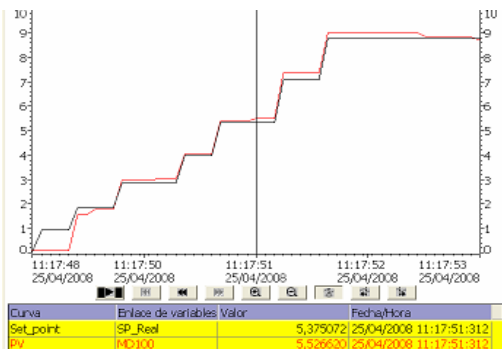
g)



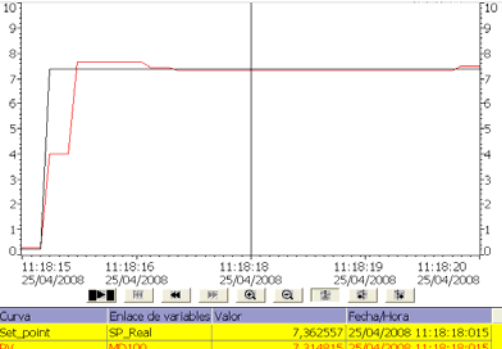
h)



i)



j)



k)

Figura 4. 14:a)  $K_p=3, T_d=10, T_i=40$ ; b)  $K_p=2, T_d=10, T_i=40$ ;  
c)  $K_p=2, T_d=10, T_i=40$ ; d)  $K_p=2, T_d=10, T_i=40$ ;



- e)  $K_p=2$ ,  $T_d=10$ ,  $T_i=60$ ; f)  $K_p=2$ ,  $T_d=10$ ,  $T_i=100$ ;  
g)  $K_p=2$ ,  $T_d=10$ ,  $T_i=100$ ; h)  $K_p=2$ ,  $T_d=10$ ,  $T_i=100$ ;  
i)  $K_p=2$ ,  $T_d=10$ ,  $T_i=150$ ; j)  $K_p=2$ ,  $T_d=10$ ,  $T_i=150$ ;  
k)  $K_p=2$ ,  $T_d=10$ ,  $T_i=150$ .

## 4.5 Sintonía mediante Ensayo-Error (2 cilindros).

En este apartado a diferencia del anterior se afronta el control de los dos cilindros que componen el robot de forma conjunta como se muestra en la figura 4.15. Esto supone una dificultad añadida debido al acoplamiento de lazos que aparece.

Los experimentos que se realizan, se hacen modificando la consigna de los dos cilindros por separado (primero uno y después el otro). Cada cilindro posee su bloque PID independiente por lo tanto, al estar los dos cilindros conectados a la toma de aire, cuando movemos un cilindro el otro permanece sin moverse, en el último lugar que lo hayamos situado.



Figura 4. 15 Dos cilindros unidos.

-Si miramos el robot como aparece en la figura 4.15 (de frente):

Cilindro n°1 derecha.

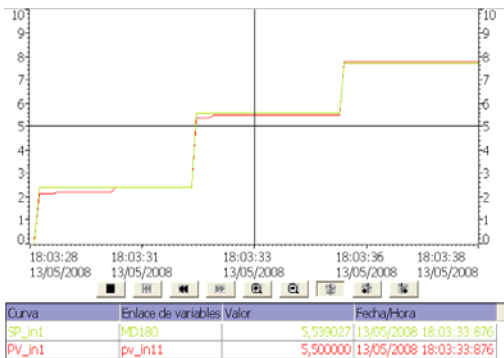
Cilindro n°2 izquierda.

### 4.5.1 Control proporcional (P).

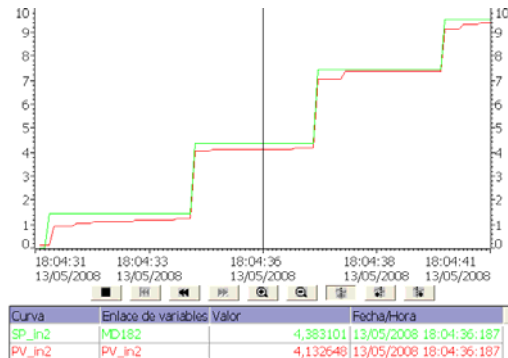
Con  $K_p=1$  obtenemos una respuesta bastante buena.

Como podemos observar en las gráficas 4.16 a y b la respuesta del cilindro n°1 es un poco más estable que la del n°2 para el mismo valor de  $K_p$ , por lo tanto comenzaremos haciendo las pruebas siempre con el cilindro 1 ya que cuando se descontrola los picos serán más suaves evitando así que se dañen los cilindros.

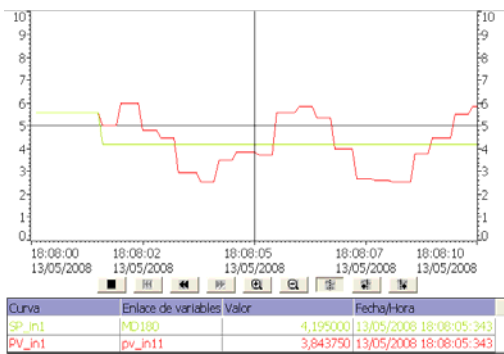
Hemos seguido aumentando el valor del término proporcional para mejorar la respuesta, pero el cilindro se descontrola para un valor cercano a 2 por lo tanto no vamos a hacer la prueba con el otro cilindro, nos vamos a quedar con una  $K_p=1$  como un buen valor de sintonía. A continuación vamos a proceder a introducir el término derivativo.



a)



b)



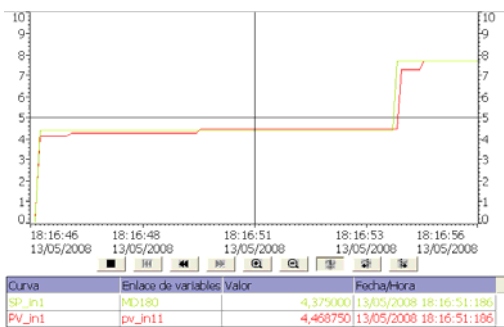
c)

Figura 4. 16: a) Cilindro n°1  $K_p = 1$  ; b) Cilindro n°2  $K_p=1$  c) Cilindro n°1  $K_p=2$ .

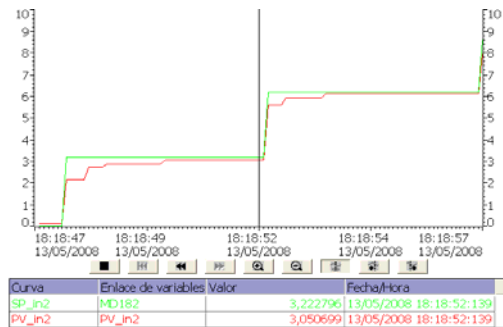
## 4.5.2 Control proporcional derivativo (PD).

Vemos que las gráficas de las figuras 4,17 g) y 4,17 h) es la mejor respuesta que se puede conseguir para un controlador PD antes de llegar al descontrol de los cilindros.

Con  $K_p=1$  y  $T_d=5$  ya empieza a tener unas pequeñas oscilaciones que podrían resultar peligrosas, por lo tanto vamos a llegar solo hasta  $T_d=4$ .



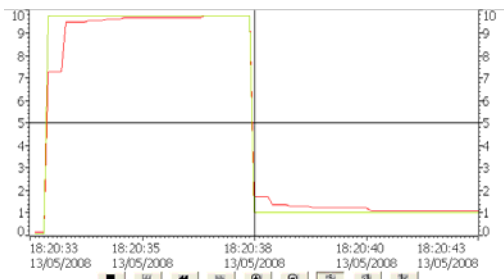
a)



b)

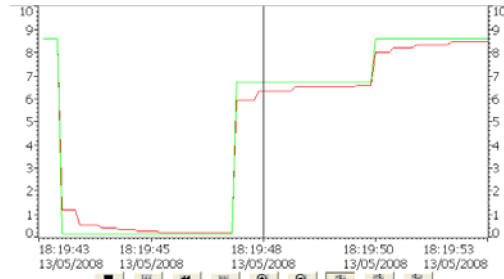


**Calibración, Control y Diseño SCADA de un robot paralelo neumático con el autómeta S7-300.**



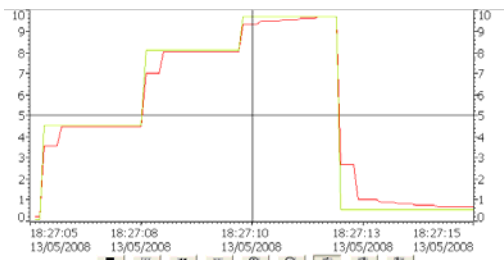
Curva	Enlace de variables	Valor	Fecha/Hora
SP_in1	MD180	1,040971	13/05/2008 18:20:38:421
PV_in1	pv_in11	1,718750	13/05/2008 18:20:38:421

**b)**



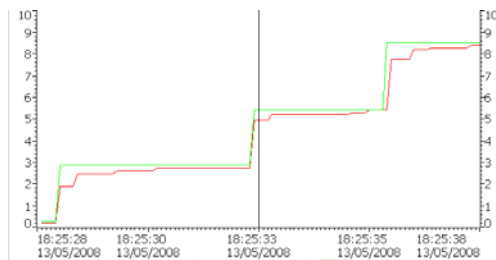
Curva	Enlace de variables	Valor	Fecha/Hora
SP_in2	MD182	6,726866	13/05/2008 18:19:48:124
PV_in2	PV_in2	6,351398	13/05/2008 18:19:48:124

**d)**



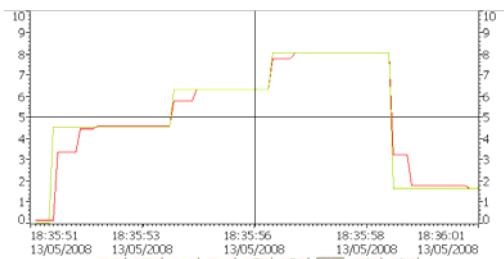
Curva	Enlace de variables	Valor	Fecha/Hora
SP_in1	MD180	9,702622	13/05/2008 18:27:10:859
PV_in1	pv_in11	9,375000	13/05/2008 18:27:10:859

**e)**



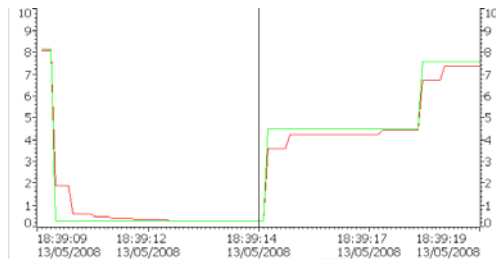
Curva	Enlace de variables	Valor	Fecha/hora
SP_in2	MD182	5,414354	13/05/2008 18:25:33:124
PV_in2	PV_in2	4,945148	13/05/2008 18:25:33:124

**f)**



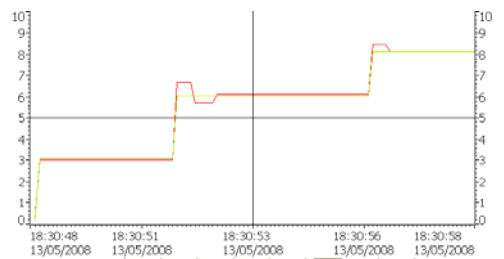
Curva	Enlace de variables	Valor	Fecha/Hora
SP_in1	MD180	6,312500	13/05/2008 18:35:56:249
PV_in1	pv_in11	6,343750	13/05/2008 18:35:56:249

**g)**



Curva	Enlace de variables	Valor	Fecha/hora
SP_in2	MD182	0,235604	13/05/2008 18:39:14:657
PV_in2	PV_in2	0,260244	13/05/2008 18:39:14:657

**h)**



Curva	Enlace de variables	Valor	Fecha/Hora
SP_in1	MD180	6,031250	13/05/2008 18:30:53:968
PV_in1	pv_in11	6,125000	13/05/2008 18:30:53:968

**i)**

**Figura 4. 17:** a) Cilindro n°1  $K_p=1$ ,  $T_d=2s$  ; b) Cilindro n°2  $K_p=1$ ,  $T_d=2s$  ; i) Cilindro n°1  $K_p=1$ ,  $T_d=5s$  ;  
 c) Cilindro n°1  $K_p=1$ ,  $T_d=2s$  ; d) Cilindro n°2  $K_p=1$ ,  $T_d=2s$  ;  
 e) Cilindro n°1  $K_p=1$ ,  $T_d=3s$  ; f) Cilindro n° 2  $K_p=1$ ,  $T_d=3s$  ;  
 g) Cilindro n°1  $K_p=1$ ,  $T_d=4s$  ; h) Cilindro n°2  $K_p=1$ ,  $T_d=4s$  ;





### 4.5.3 Control Proporcional, Integral, Derivativo (PID).

Con  $T_i=50$ , figura 4.18 a) el cilindro presenta picos muy elevados, por lo tanto se ha optado por aumentar el valor del tiempo integral.

Conforme aumentamos  $T_i$  obtenemos una respuesta considerablemente mejor, esto quiere decir que conforme aumentamos  $T_i$  esta va teniendo menos efecto sobre la respuesta por lo tanto nos vamos a quedar con un controlador proporcional derivativo, como mejor respuesta obtenida.

También se obtiene una buena respuesta sólo con término proporcional con  $K_p=1$

Las oscilaciones que se producen de la variable del proceso alrededor de la consigna son debidas al tiempo integral y a la zona muerta del actuador.

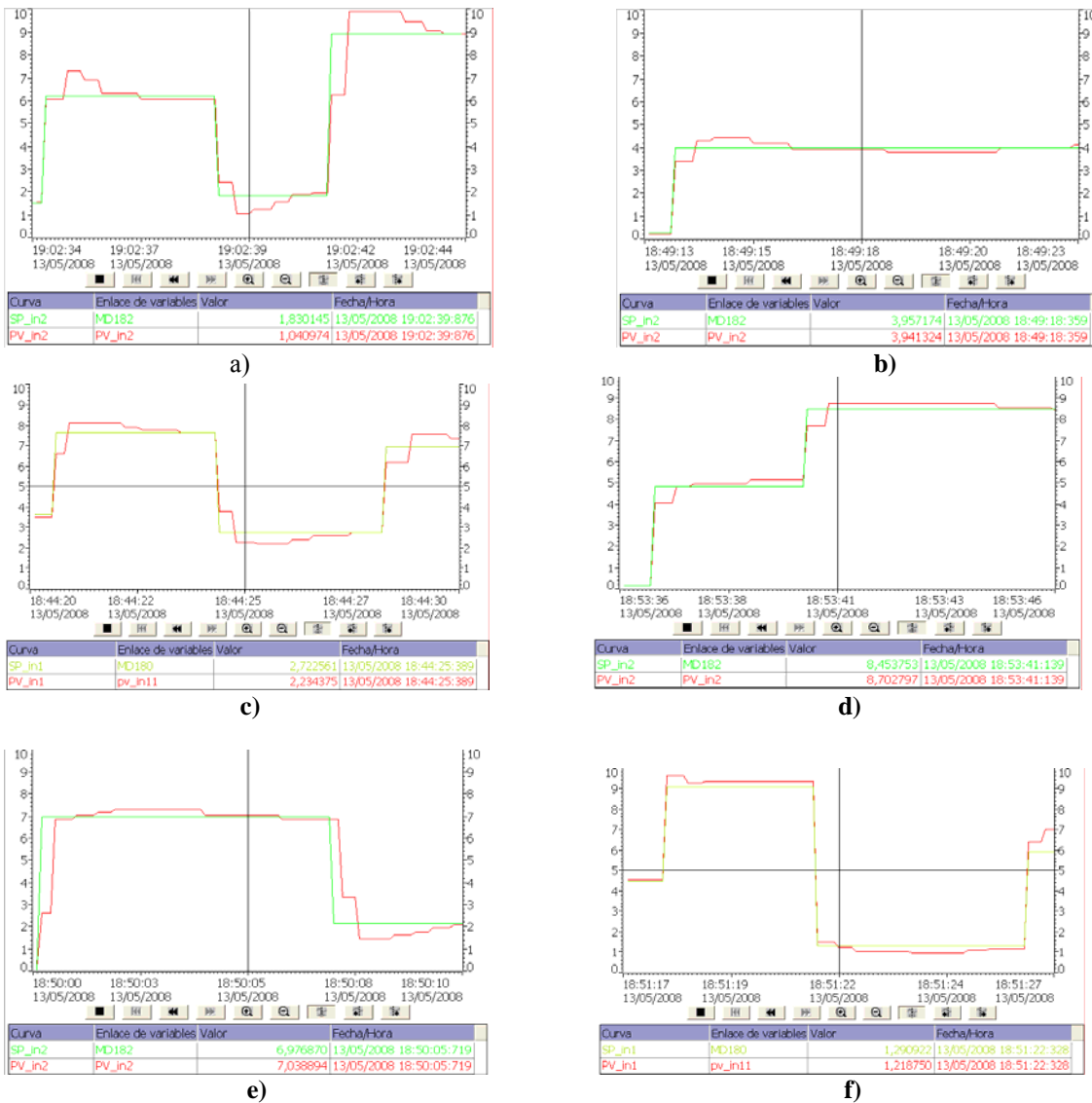


Figura 4. 18: a) Cilindro n°1  $K_p=1, T_d=4s, T_i=50$ ; b) Cilindro n°2  $K_p=1, T_d=4s, T_i=100$ ;



**Calibración, Control y Diseño SCADA de un robot paralelo neumático con el autómatas S7-300.**

---

---

- c) Cilindro nº1  $K_p=1$ ,  $T_d=4s$ ,  $T_i=100$ ;      d) Cilindro nº2  $K_p=1$ ,  $T_d=4s$ ,  $T_i=150$ ;  
e) Cilindro nº1  $K_p=1$ ,  $T_d=4s$ ,  $T_i=150$ ;      f) Cilindro nº1  $K_p=1$ ,  $T_d=4s$ ,  $T_i=150$ .



## 4.6 Funciones utilizadas.

Las funciones utilizadas para realizar el control PID en STEP7 son las que se muestran a continuación:

### 4.6.1 SFB 41/FB 41 "CONT\_C".

El SFB/FB "CONT\_C" (continuous controller) sirve para la regulación de procesos industriales con magnitudes de entrada y salida continuas utilizando sistemas de automatización SIMATIC S7. Mediante la parametrización es posible conectar o desconectar las funciones parciales del regulador PID, adaptándolo así al proceso regulado.

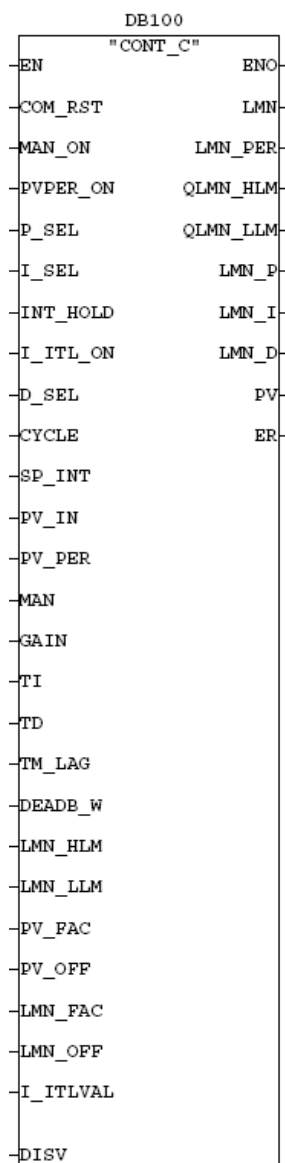


Figura 4. 19 SFB 41 "CONT\_C".



### **Aplicación.**

El regulador puede aplicarse como regulador PID de consigna fija, individualmente o también en regulaciones de varios lazos como regulador en cascada, regulador de mezcla o regulador de relación. El modo de trabajar se basa en el algoritmo de regulación PID del regulador muestreado con señal de salida analógica, complementada dado el caso por una etapa de formación de impulsos para la creación de señales de salida con modulación de ancho de impulsos para regulaciones de dos o tres puntos con actuadores proporcionales.

**Nota:** El cálculo de los valores en los bloques de regulación sólo será correcto si el bloque se llama en intervalos regulares. Por esa razón debería llamar los bloques de regulación en un OB de alarma despertador OB (OB 30 a OB 38). El intervalo debe especificarse en el parámetro CYCLE.

### **Descripción.**

Junto a las funciones en las ramas de valores de consigna y real, el SFB realiza un regulador PID completo con salida continua de magnitud manipulada y posibilidad de influenciar manualmente el valor manipulado.

A continuación, se describen las funciones parciales:

#### **Rama de valor de consigna.**

El valor de consigna se introduce en la entrada SP\_INT en formato en coma flotante.

#### **Rama de valor real.**

El valor real puede ser leído en formato de periferia y en formato en coma flotante. La función CRP\_IN transforma el valor de periferia PV\_PER en un formato en coma flotante de -100 ... +100 %, según la siguiente fórmula:

La función PV\_NORM normaliza la salida de CRP\_IN según la siguiente regla:

PV\_FAC está preasignado con 1 y PV\_OFF, con 0.

Formación del error de regulación

La diferencia entre el valor de consigna y el valor real es la diferencia o error de regulación. Para suprimir la pequeña oscilación permanente debida a la cuantificación de la magnitud manipulada (p. ej. en una modulación de ancho de impulsos con PULSEGEN), el error de regulación se conduce por una zona muerta (DEADBAND). Con DEADB\_W = 0 está desconectada la zona muerta.

### **Algoritmo PID**

El algoritmo PID trabaja en el algoritmo de posición. Las acciones proporcional, integral (INT) y derivativa (DIF) están conectadas en paralelo y pueden conectarse y desconectarse individualmente. De esta forma pueden parametrizarse reguladores P, PI, PD y PID. Pero también son posibles reguladores I puros.

### **Procesamiento de valores manuales.**

Es posible conmutar entre modo manual y modo automático. En el modo manual, la magnitud manipulada sigue a un valor ajustado manualmente. El integrador (INT) se pone internamente a



LMN - LMN\_P - DISV y el diferenciador (DIF) se pone a 0 y se compensa internamente. La conmutación al modo automático se efectúa así sin choques.

#### **Procesamiento de valores manipulados.**

El valor manipulado se limita con la función LMNLIMIT a valores prefijables. La superación de los límites por parte de la magnitud de entrada se señala mediante bits al efecto.

La función LMN\_NORM normaliza la salida de LMNLIMIT según la siguiente regla:

LMN\_FAC está preasignado con 1 y LMN\_OFF, con 0.

El valor manipulado está también a disposición en formato de periferia.

#### **Aplicación de magnitud perturbadora (control anticipativo).**

En la entrada DISV puede aplicarse aditivamente una magnitud perturbadora.

#### **Inicialización**

El SFB/FB "CONT\_C" dispone de una rutina de inicialización que se tramita cuando el parámetro de entrada COM\_RST = TRUE.

Al inicializar, el integrador se ajusta internamente al valor de inicialización I\_ITVAL. Si se llama en un nivel de alarma cíclica, continúa trabajando a partir de este valor.

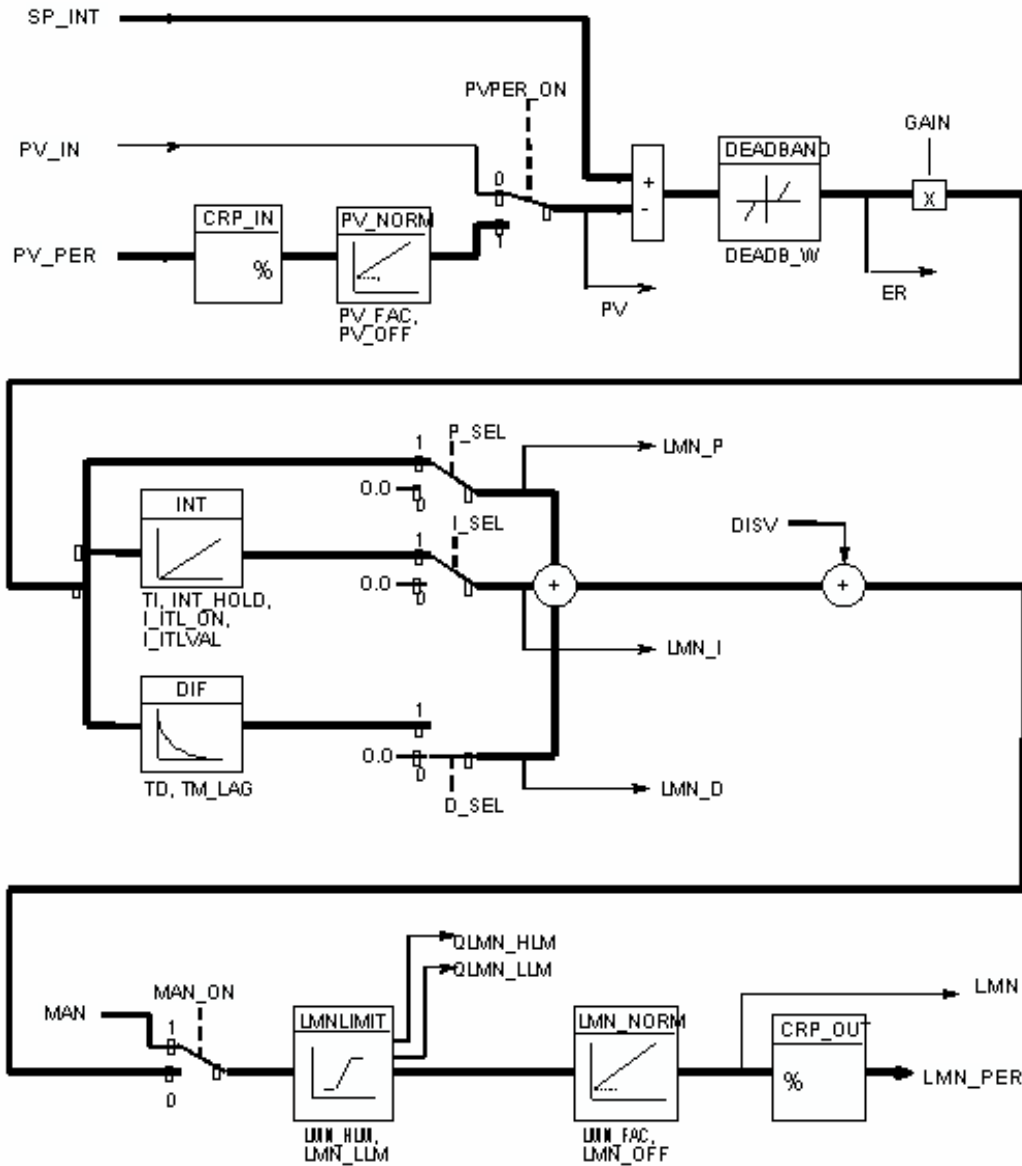
Todas las demás salidas se ponen a sus valores por defecto.

#### **Informaciones de error**

No se emplea la palabra de información de error RET\_VAL.



### Esquema de bloques



### Parámetros de entrada

La tabla siguiente contiene los parámetros de entrada del SFB 41/FB 41 "CONT\_C".

Parámetro	Tipo de datos	Valores posibles	Por defecto	Descripción
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



## Calibración, Control y Diseño SCADA de un robot paralelo neumático con el autómatas S7-300.

PVPER_ON	BOOL		FALSE	<p>Si está activada la entrada "Conectar a modo manual", está interrumpido el lazo de regulación. Como valor manipulado se fuerza un valor manual.</p> <p>PROCESS VARIABLE PERIPHERY ON / Conectar valor real de periferia</p>
P_SEL	BOOL		TRUE	<p>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> <p>PROPORTIONAL ACTION ON / Conectar acción P</p> <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".</p>
I_SEL	BOOL		TRUE	<p>INTEGRAL ACTION ON / Conectar acción I</p> <p>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".</p>
INT_HOLD	BOOL		FALSE	<p>INTEGRAL ACTION HOLD / Congelar acción I</p> <p>La salida del integrador puede congelarse. Para ello se ha de activar la entrada "Congelar acción I".</p>
I_ITL_ON	BOOL		FALSE	<p>INITIALIZATION OF THE INTEGRAL ACTION / Inicializar acción I</p> <p>La salida del integrador puede inicializarse a la entrada I_ITLVAL. Para ello se ha de activar la entrada "Inicializar acción I".</p>
D_SEL	BOOL		FALSE	<p>DERIVATIVE ACTION ON / Conectar acción D</p> <p>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".</p>
CYCLE	TIME	>= 1ms	T#1s	<p>SAMPLE TIME / Tiempo de muestreo</p> <p>El tiempo entre las llamadas del bloque debe ser constante. La entrada "Tiempo de muestreo" indica el tiempo entre las llamadas del bloque.</p>
SP_INT	REAL	-100.0...100.0	0.0	<p>INTERNAL SETPOINT / Consigna interna</p> <p>La entrada "Consigna interna" sirve para ajustar un valor de consigna.</p>
PV_IN	REAL	-100.0...100.0	0.0	<p>PROCESS VARIABLE IN / Entrada de valor real</p> <p>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.</p>
PV_PER	WORD		W#16#0000	<p>PROCESS VARIABLE PERIPHERY / Valor real de periferia</p> <p>El valor real en formato de periferia se interconecta con el regulador en la entrada "Valor real de periferia".</p>
MAN	REAL	-100.0...100.0	0.0	<p>MANUAL VALUE / Valor manual</p> <p>La entrada "Valor manual" sirve para establecer un valor manual mediante función de manejo/visualización (interface hombre máquina)..</p>
GAIN	REAL		2.0	<p>PROPORTIONAL GAIN / Ganancia proporcional</p> <p>La entrada "Ganancia proporcional" indica la ganancia del regulador.</p>
TI	TIME	>= CYCLE	T#20s	<p>RESET TIME / Tiempo de acción integral</p> <p>La entrada "Tiempo de acción integral" determina el comportamiento temporal del integrador.</p>
TD	TIME	>= CYCLE	T#10s	<p>DERIVATIVE TIME / Tiempo de diferenciación (acción</p>



## Calibración, Control y Diseño SCADA de un robot paralelo neumático con el autómeta S7-300.

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

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

La tabla siguiente contiene los parámetros de salida del SFB 41 "CONT\_C".

<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.

A la herramienta de parametrización se accede a través de Inicio > Simatic > STEP 7 > Parametrizar Regulación PID. El manual electrónico se encuentra en Inicio > Simatic > Manuales S7 > Regulación PID.



## 4.7 Cinemática.

### 4.7.1 Cinemática Directa.

La cinemática directa consiste en determinar la posición del extremo final conocidos los valores de las coordenadas articulares  $d_1$  y  $d_2$  y las características geométricas del robot.

La posición del actuador final vendrá determinada por las coordenadas absolutas  $(X_p, Y_p)$  tomando como origen de referencia uno de los apoyos fijos.

Las **constantes** que vamos a utilizar tanto para la cinemática directa como para la inversa son las siguientes:

$a = 60$ ; “distancia entre apoyos”.

$b = 20$ ; “longitud del eslabón unión”.

Factor (cm) = 0,5; “factor numérico para situar el efector final sobre el eslabón unión.

Sit =  $b \cdot \text{factor}$ ; “Situación del efector final sobre el eslabón unión (cm)”.

$Tita_3 = 2 \cdot \pi / 3$ ;

$L = 50$ ; “longitud del cilindro (cm)”.

$G = 9.81$ ; “Aceleración de la gravedad ( $m/s^2$ )”.

#### Coordenadas de las articulaciones fijas del robot.

##### Punto a:

$$X_a = 0;$$

$$Y_a = 0;$$

##### Punto b:

$$X_b = a;$$

$$Y_b = 0;$$

Para facilitar la comprensión del código se describe gráficamente cada una de las coordenadas asignadas a las partes del robot en el proyecto Modelado, simulación y diseño de un robot paralelo de 2-GDL con actuación neumática, realizado por Miguel Torregrosa Puig, proyecto que se puede encontrar en la biblioteca de la Universidad Politécnica de Cartagena.

Para el cálculo de la cinemática directa tomamos como entradas las coordenadas articulares  $d_1$  y  $d_2$  y el sistema SCADA nos devolverá la **posición del efector final**  $(X_p, Y_p)$ .

---



$$U_{pa} = \sqrt{((l+d1)^2 + (b^2) - (2*(l+d1)*b*\cos(\text{tita3})))};$$

$$U_{pb} = (l+d2);$$

$$R = (X_a)^2 + (Y_a)^2 - U_{pa}^2;$$

$$S = (X_b)^2 + (Y_b)^2 - U_{pb}^2;$$

$$M = (Y_a - Y_b) / (X_b - X_a);$$

$$N = (1/2) * ((S - R) / (X_b - X_a));$$

$$P = M^2 + 1;$$

$$q = 2 * ((M * N) - (M * X_a) - Y_a);$$

$$r = (N^2) - (2 * N * X_a) + R;$$

$$z = (q^2 - 4 * p * r);$$

A continuación se establece la condición que deben de cumplir las ecuaciones cuadráticas para trabajar con soluciones en el dominio del campo real.

$$Z = \sqrt{b^2 - 4*a*c} > 0$$

### Posibles soluciones de la ecuación cuadrática.

Si  $Z > 0$ .

$$Y_{d1} = (-q + \sqrt{q^2 - 4 * p * r}) / (2 * p);$$

$$Y_{d2} = (-q - \sqrt{q^2 - 4 * p * r}) / (2 * p);$$

$$X_{d1} = (M * Y_{d1}) + N;$$

$$X_{d2} = (M * Y_{d2}) + N; \text{ Solución no válida.}$$

### Cálculo del ángulo tita2.

$$d = \sqrt{(X_{d1}^2) + (Y_{d1}^2)};$$

$$\text{alfa} = \text{Atan}(Y_{d1} / X_{d1});$$

$$\text{beta} = \text{Acos}(((b^2 - (l + d1)^2 - d^2) / (-2 * d * (l + d1))));$$



$\text{tita2} = \text{alfa} + \text{beta};$

$X_c = (1 + d1) * \text{Cos}(\text{tita2});$

$Y_c = (1 + d1) * \text{Sen}(\text{tita2});$

De este código se descarta una de las soluciones obtenidas. Esto es debido a que una de las soluciones nos dará siempre en el eje negativo de la coordenada Y. Si traducimos este hecho a la realidad, se deduce que esta solución no aportará una solución física posible, de ahí que sea descartada. Por tanto solo se trabajará con  $(X_d, Y_d)$ .

Por último se muestra el resto de las ecuaciones con las que se calcula la posición del efector final y se representan gráficamente los eslabones del robot y el punto de actuación.

Si  $Y_c > 0$ .

$f_i = \text{tita2} + \text{tita3} - \text{pi};$

$X_p = X_c + \text{sit} * \text{Cos}(f_i);$

$Y_p = Y_c + \text{sit} * \text{Sin}(f_i);$

La implementación de este código se encuentra en el programa informático llamado “*cinemática directa*”.



## 4.7.2 Cinemática Inversa.

En el problema inverso de la cinemática se encuentra la solución a la determinación de las variables articulares  $d_1$  y  $d_2$  conocida la posición del actuador.

Se puede plantear también desde el punto de vista siguiente: Si se quiere que el efector final se sitúe en una posición del plano determinada, ¿Qué longitudes  $d_1$  y  $d_2$  deberán extenderse los cilindros para alcanzar esa posición?

La solución a este problema viene al igual que en la cinemática directa determinada mediante rozamientos geométricos. Esta vez se parte de los valores  $(X_p, Y_p)$  conocidos, para llegar a los valores de  $d_1$  y  $d_2$ .

Ecuaciones para la resolución de la cinemática inversa:

$$\alpha = \arctan(Y_p / X_p);$$

$$d = \sqrt{(Y_p)^2 + (X_p)^2};$$

$$\beta = \arcsin((s \cdot \sin(\theta_3)) / d)$$

$$\theta_2 = \alpha + \beta;$$

$$\theta_1 = \theta_2 + \theta_3 - \pi;$$

### Cálculo del pto C para calcular $d_1$ :

$$X_c = X_p - s \cdot \cos(\theta_1);$$

$$Y_c = Y_p - s \cdot \sin(\theta_1);$$

$$m = \sqrt{(Y_c)^2 + (X_c)^2};$$

$$d_1 = m - l;$$

### Cálculo del Pto D para calcular $d_2$ .

$$X_d = X_p + s \cdot \cos(\theta_1);$$

$$Y_d = Y_p + s \cdot \sin(\theta_1);$$

$$n = \sqrt{(Y_d)^2 + (a - X_d)^2};$$

$$d_2 = n - l;$$



## Calibración, Control y Diseño SCADA de un robot paralelo neumático con el autómeta S7-300.

---

En primer lugar se calculan los ángulos necesarios para determinar la posición de las articulaciones: puntos C y D, conocidos estos puntos podemos obtener el valor total de la extensión del cilindro (longitud del cilindro + longitud extendida del vástago. Si le restamos a dicho valor la longitud propia de los cilindros l se obtienen las coordenadas articulares buscadas.

La implementación de este código se encuentra en el programa informático llamado “*cinemática inversa*”.

Para cualquier duda sobre este tema consultar proyecto Modelado, simulación y diseño de un robot paralelo de 2-GDL con actuación neumática. Realizado por: D. Miguel Torregrosa Puig.



# Capítulo 5

## *Diseño SCADA*



## 5.1 Comunicación.

La comunicación del sistema SCADA (con panel de operador el PC) y el PLC se hace mediante el cable MPI el mismo que utilizamos para realizar la comunicación de STEP7 y el PCL no hay que realizar ningún cambio de configuración.

## 5.2 Abrir aplicación diseñada en el sistema SCADA.

Al abrir el sistema SCADA nos aparece la ventana de la figura 5.1, pulsamos “abrir proyecto existente” y nos aparecerá un menú donde tenemos que buscar nuestro proyecto “.hmi”.

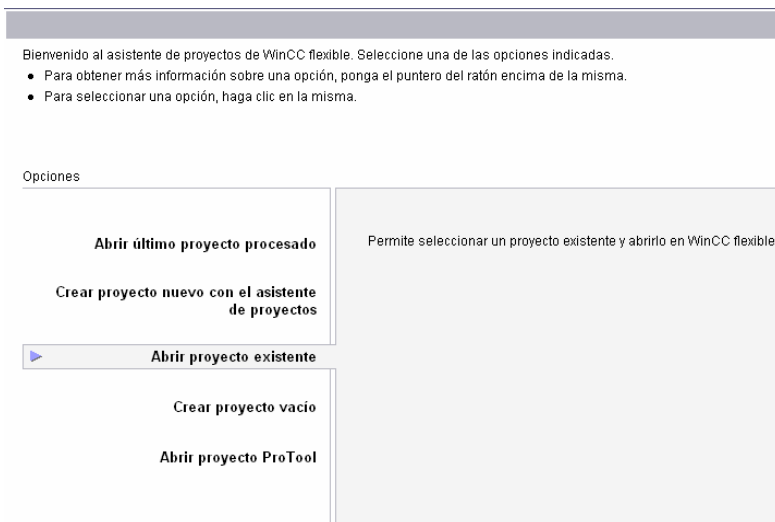
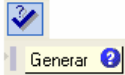


Figura 5. 1 Abrir proyecto existente.1.

## 5.3 Ejecutar aplicación diseñada en el sistema SCADA.

Una vez creado nuestro programa en el sistema SCADA hay que seguir los siguientes pasos:

1. Compilar el programa para comprobar que no tenemos errores, esto se hace pulsando el

siguiente botón  que nos encontramos en la barra de herramientas del sistema SCADA.

Si compilamos y no tenemos errores nuestro programa ya puede ser ejecutado. De todas formas si pulsamos iniciar runtime directamente y nuestro programa tiene errores, este no se ejecuta.



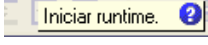


## Calibración, Control y Diseño SCADA de un robot paralelo neumático con el autómeta S7-300.

---

2. Para ejecutar nuestro programa pulsaremos el siguiente botón de la barra de herramientas del



sistema SCADA  y ya podremos controlar nuestro robot desde el sistema SCADA con el PC como panel de operador.



## 5.4 Control SCADA de 1 cilindro neumático.

-En la figura 5.2 se muestra la pantalla principal para el control de un cilindro neumático. Como podemos observar disponemos de 4 opciones las cuales podemos pulsar según la acción que deseemos realizar:

1. Calibrado automático.
2. Calibrado manual.
3. Control PID
4. Stop.



Figura 5. 2 Pantalla principal (1 cilindro).

### 5.4.1 Descripción de la pantalla del “Calibrado automático”.

Si pulsamos el botón “Calibrado automático” saltamos a la pantalla de la figura 5.3 a) la cual nos muestra el siguiente mensaje.

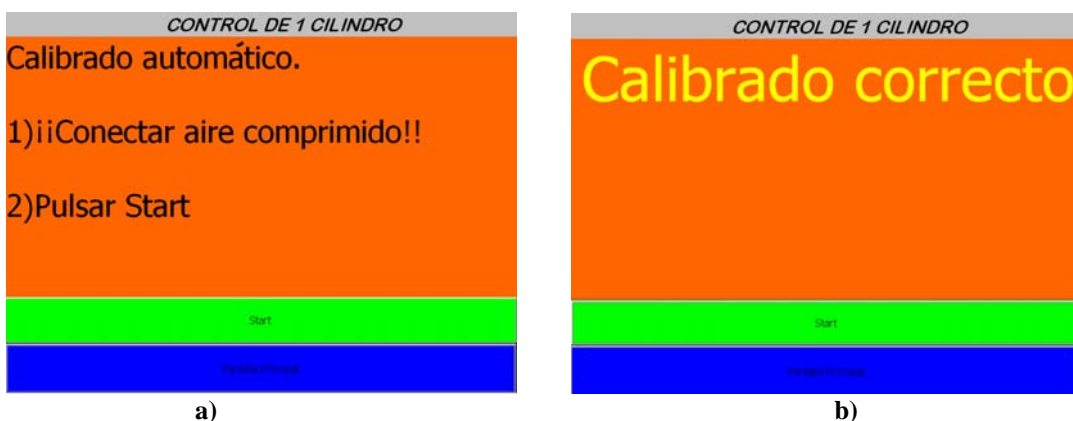


Figura 5. 3 Pantalla de calibrado automático: a) Pantalla 1 ; b) Pantalla 2.

Si el convertidor DADE se encuentra sin calibrar se procederá de forma automática a su calibración cuando pulsemos el botón START.



Cuando se haya calibrado desaparecerá el mensaje de la figura 5.3 a) y aparecerá el mensaje de la figura 5.3 b), si el calibrado se ha realizado de forma correcta.

Si se pulsa el botón START estando calibrado el convertidor DADE, nos aparecerá directamente la pantalla de la figura 5.3 b).

### 5.4.2 Descripción de la pantalla de “Calibrado Manual”.

Si pulsamos “Calibrado manual” saltamos a la pantalla de la figura 5.4 a).

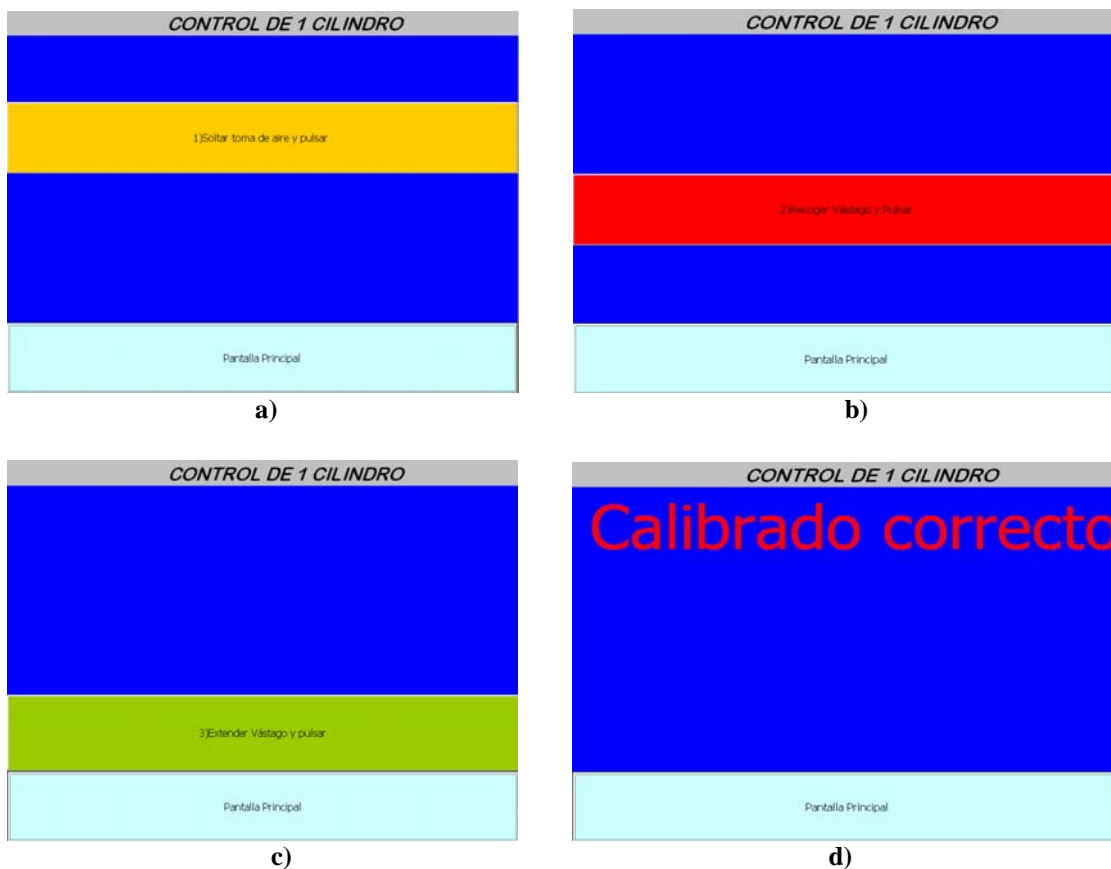


Figura 5. 4 Pantallas de calibrado manual: a) Pantalla 1 ; b) Pantalla 2 ; c) Pantalla 3; d) Pantalla 4

Cuando soltamos la toma de aire comprimido y pulsamos el botón amarillo de la figura 5.4 a) nos aparece la figura 5.4 b)

Una vez que recogemos el vástago y lo dejamos fijo (sin que se produzca ningún movimiento), se graba la referencia.

Extendemos vástago y dejamos fijo, una vez que pulsamos el botón verde procedemos al grabado del punto final de la carrera de trabajo (10v) en el convertidor DADE.



Si el calibrado manual se ha realizado de una forma correcta nos aparece la ventana de la figura 5.4 b).

### 5.4.3 Descripción de la pantalla “Control PID”.

-El potenciómetro que se encuentra a la derecha de la pantalla de la figura 5.5 es el set-point, marcado en negro en esta misma gráfica.

-La variable del proceso (PV) se encuentra marcada en rojo en la gráfica y se trata de la salida en posición del cilindro.

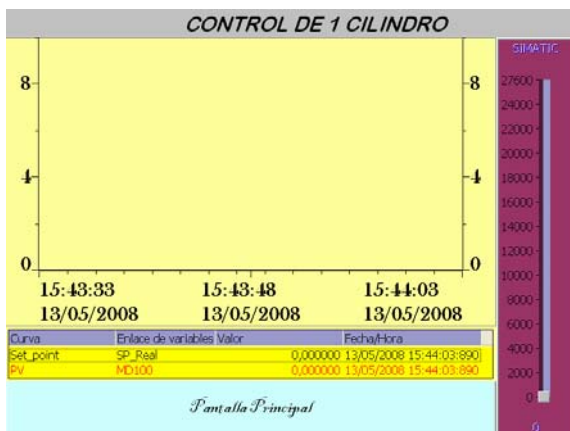


Figura 5. 5 Pantalla de control PID. (1 cilindro)

## 5.4.4 Parametrizador PID

-Si queremos ajustar los valores del controlador debemos de seguir los siguientes pasos:

1. Pinchamos en Inicio/todos los programas/simatic /Step7/ parametrizador PID como muestra la figura 5.6. Nos aparece la ventana de la figura 5.7.

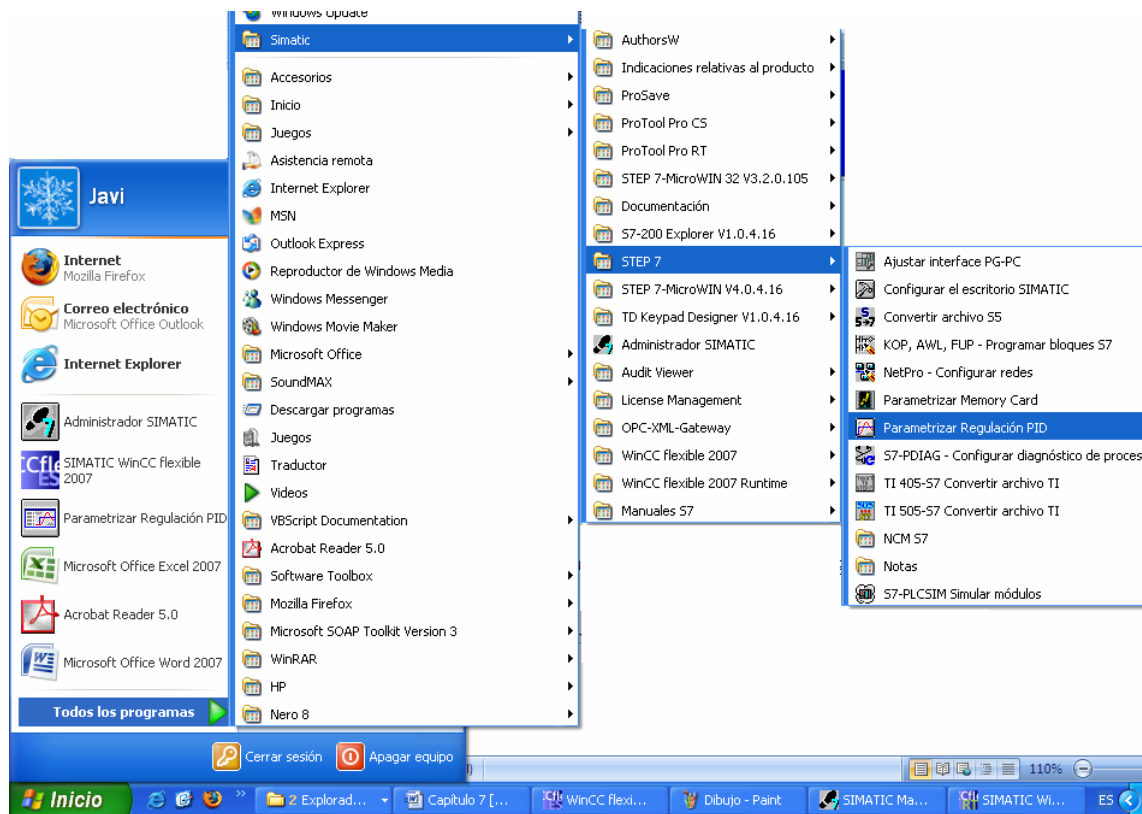


Figura 5. 6 Abrir parametrizador.

2. Pinchamos en Archivo/abrir, donde en Ruta, tenemos que buscar nuestro proyecto.
3. El siguiente paso es desplegar el árbol de nuestro proyecto como se muestra en la figura 5.7 y dentro de la carpeta “bloques” nos aparecerá “DB100”el cual tenemos que abrir (se trata de un DB de instancia asociado a la función SFB41 creado anteriormente y que contiene los datos del controlador PID).

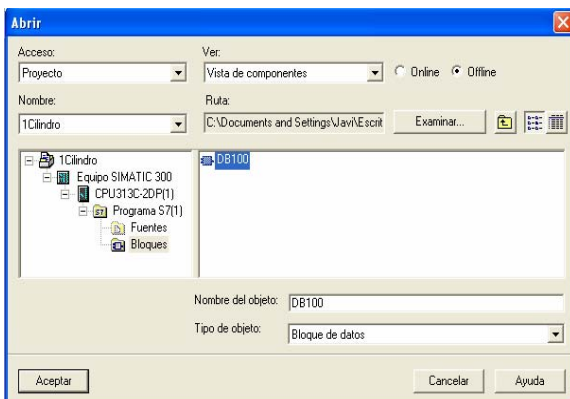


Figura 5. 7 Abrir DB100

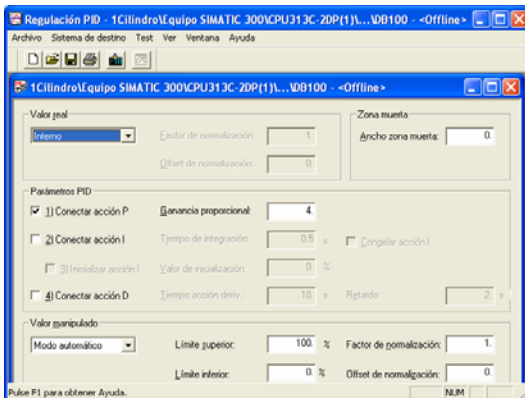



Figura 5. 8 Ventana para Introducir parámetros PID.

Una vez que nos encontramos en la pantalla de la figura 5.8 ya podemos modificar los parámetros del controlador, pero tenemos que tener en cuenta que estamos offline es decir, para que los cambios que realicemos tengan efecto en el controlador, una vez introducidos los parámetros en esta pantalla deberemos pulsar el botón  de la barra de herramientas que nos aparece arriba a la izquierda, entonces los datos que hayamos introducido se grabarán en el autómata.

Estos pasos se tienen que seguir en todos los programas informáticos para modificar los parámetros PID del controlador.

### 5.4.5 Pulsador “Stop”

Este botón finaliza “Runtime”, en todos los programas informáticos realiza la misma función.



## 5.5 Control SCADA de 2 cilindros neumáticos.

-En la figura 5.9 se muestra la pantalla principal para el control de 2 cilindros neumáticos. Como podemos observar disponemos de 4 opciones las cuales podemos utilizar según la acción que deseemos realizar:

1. Calibrado automático.
2. Calibrado manual.
3. Control PID
4. Stop.



Figura 5. 9 Pantalla principal. (2 cilindros).

### 5.5.1 Descripción de la pantalla “Calibrado automático”

Si pulsamos “Calibrado automático” saltamos a la pantalla de la figura 5.10 a).

Si el convertidor DADE se encuentra sin calibrar se procederá de forma automática a su calibración cuando pulsamos el botón START.

Cuando se haya calibrado desaparecerá el mensaje de la figura 5.10 a) y aparecerá el mensaje de la figura 5.10 b), si el calibrado se ha realizado de una forma correcta.

Si se pulsa el botón START estando grabado el convertidor DADE, nos aparecerá directamente la pantalla de la figura 5.10 b).



Figura 5. 10 Calibrado automático de 2 cilindros): a) Pantalla 1 ; b)Pantalla 2.



### 5.5.2 Descripción de la pantalla “Calibrado Manual” .

- Si pulsamos “Calibrado manual”, saltamos a la pantalla de la figura 5.11 a).
- Cuando soltamos la toma de aire comprimido y pulsamos el botón amarillo nos aparece la figura 5.11 b).
- Una vez que recogemos el vástago, lo dejamos fijo y pulsamos el botón rojo nos aparece la pantalla de la figura 5.11 c)
- Extendemos vástagos y dejamos fijos, una vez que se pulsa el botón verde procedemos al grabado del punto final de la carrera de trabajo (10v) y si el calibrado manual se ha realizado de una forma correcta nos aparece la ventana de la figura 5.25 c)

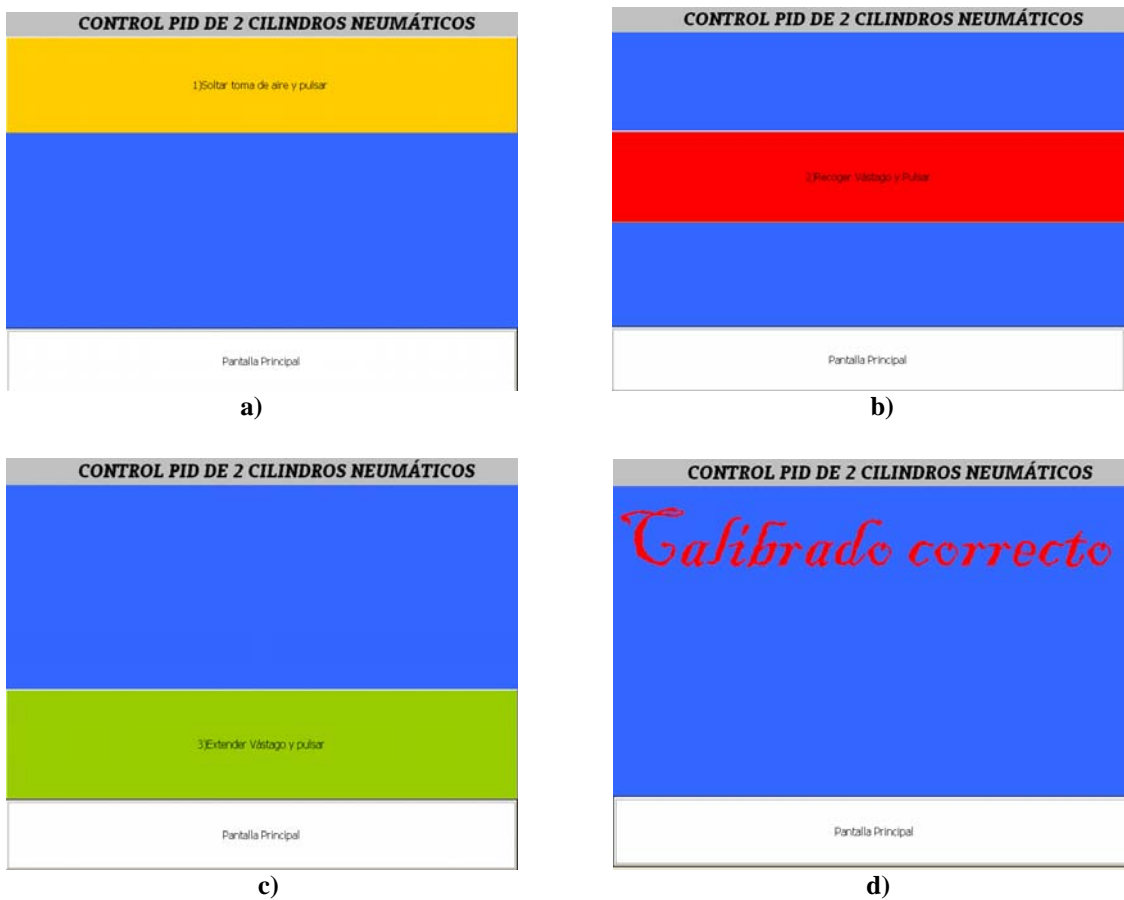


Figura 5. 11 Calibrado manual: a) Pantalla 1 ; b) Pantalla 2 ; c) Pantalla3 ; d) Pantalla 4.





### 5.5.3 Descripción de la pantalla “Control PID”.

El potenciómetro que se encuentra a la derecha de la pantalla de la figura 5.12 (color verde) es el set-point, del cilindro de la derecha (si miramos el robot de frente), marcado en negro en la gráfica de la figura 5.12.

La variable del proceso (PV) se encuentra marcada en rojo en la gráfica y se trata de la salida en posición del cilindro neumático.

El potenciómetro que se encuentra a la izquierda (color azul) es el set-point del cilindro de la izquierda.

En este programa se han puesto los dos SET-POINT y las dos PV en una misma gráfica. Si se quiere un control más exhaustivo de los cilindros en el programa “cinemática directa” se encuentran las gráficas por separado para realizar una sintonización más fina.

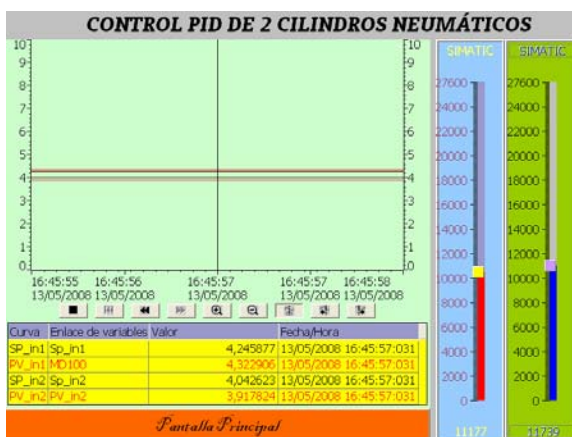


Figura 5. 12 Control PID. (2 cilindros).



## 5.6 Control SCADA Cinemática Directa.

En este apartado y en el apartado 5.8 no se han explicado las pantallas “Calibrado manual” y “Calibrado automático” ya que son iguales que en el diseño SCADA de: “Control PID de 2 cilindros neumáticos”, si se desea observar alguna de estas pantallas mirar apartados 5.5.1 y 5.5.2.

Como podemos observar en la figura 5.13 pantalla principal de este programa tenemos más opciones que en los anteriores. Por lo tanto si queremos realizar pruebas relacionadas con la sintonización PID deberemos de cargar este programa en el PLC.

1. Calibrado automático.(Explicado anteriormente)
2. Calibrado manual.(Explicado anteriormente)
3. Sintonizar PID1
4. Sintonizar PID2
5. Cinemática directa
6. Stop.



Figura 5. 13 Pantalla principal. (Cinemática directa)



### 5.6.1 Descripción de la pantalla “Sintonizar PID1”.

Como se ha comentado anteriormente en la figura 5.14 se encuentra el set-point y la variable del proceso del cilindro número 1 en una gráfica (separada de la gráfica del cilindro numero 2) .

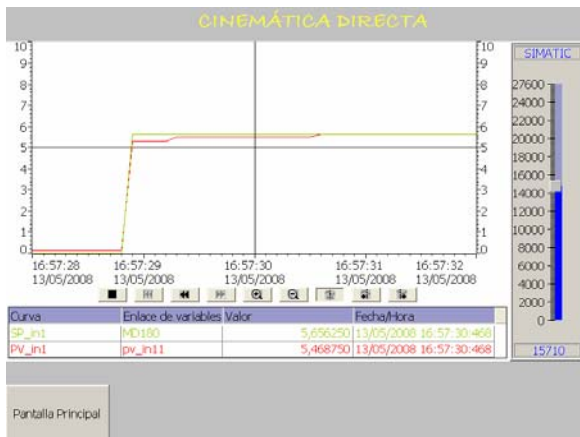


Figura 5. 14 Sintonizar PID1. (Cinemática directa).

### 5.6.2 Descripción de la pantalla “Sintonizar PID2”.

Si pulsamos el botón sintonizar PID2 nos encontramos con la figura 5.15 la cual nos muestra el SET-POINT y la variable del proceso (posición del cilindro) del cilindro numero 2.

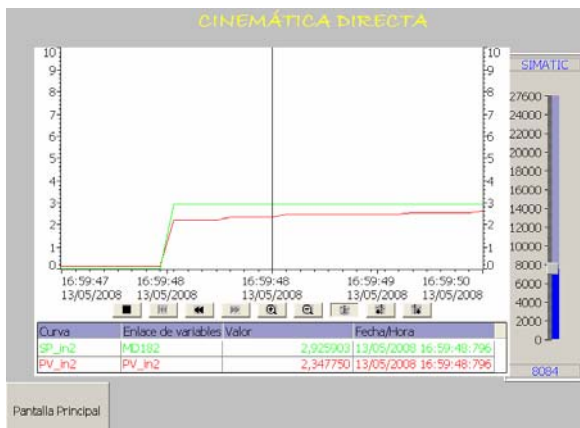


Figura 5. 15 Sintonizar PID2. (Cinemática directa).



### 5.6.3 Descripción de la pantalla “Cinemática Directa”.

En la grafica de la figura 5.16 como podemos observar aparecen todos los términos necesarios para la realización de la cinemática directa donde Xp e Yp son las coordenadas del efector final.

Para más información sobre este tema mirar apartado 4.7 de este proyecto, “Modelado y si no resulta suficiente, simulación y diseño de un robot paralelo de 2 grados de libertad con actuación neumática”, realizado por: D. Miguel Torregrosa Puig.

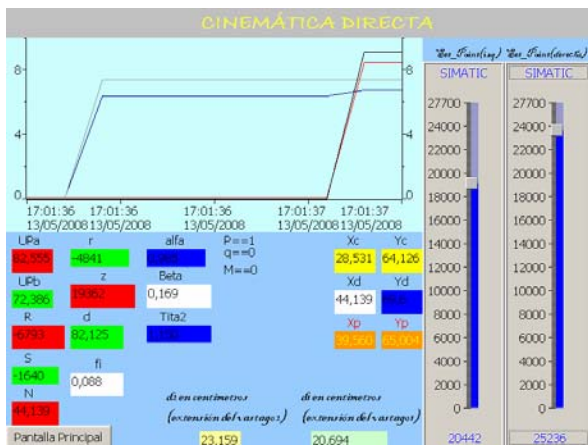


Figura 5. 16 Cinemática directa. (Cinemática directa).



## 5.7 Control SCADA “Cinemática Inversa”.

Como podemos observar en la figura 5.17 pantalla principal de este programa tenemos 3 opciones.

1. Calibrado automático.(Explicado anteriormente)
2. Calibrado manual.(Explicado anteriormente)
3. Cinemática inversa.



Figura 5.17 Pantalla principal. (Cinemática inversa).

### 5.7.1 Descripción de la pantalla “Cinemática Inversa”.

En la grafica de la figura 5.18 aparecen todos los términos necesarios para la realización de la cinemática inversa donde  $X_p$  e  $Y_p$  son las coordenadas del efector final.

Para más información sobre este tema mirar apartado 4.7 de este proyecto y en caso de no ser suficiente mirar “Modelado, simulación y diseño de un robot paralelo de 2 grados de libertad con actuación neumática”, realizado por: D. Miguel Torregrosa Puig.

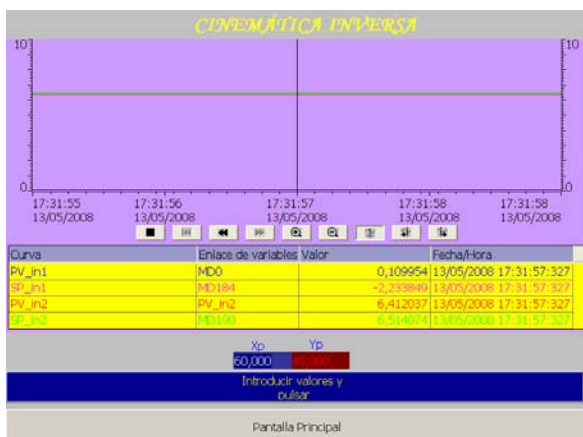


Figura 5.18 Pantalla “Cinemática inversa”.



En la pantalla de la figura 5.18 cuando introducimos los valores de la coordenada  $X_p$  o  $Y_p$  en la casilla correspondiente hay que pulsar intro para que el valor se grave. Una vez grabados hay que pulsar el botón azul para que los valores sean leídos por el autómatas y los cilindros actúen.

A la hora de introducir las coordenadas tenemos que tener en cuenta que el origen de coordenadas se encuentra en la base del cilindro de la izquierda (si miramos el robot de frente), por lo tanto los valores que deberemos de introducir para el correcto funcionamiento de la cinemática inversa son los siguientes:

$$1 \leq X_p \leq 60$$
$$45 \leq Y_p \leq 80$$

## 5.8 Transferencia a pantalla táctil TP177B color PN/DP

Al estar utilizando el PC como panel de operador tenemos que realizar una serie de modificaciones tanto en el panel de operador como en la configuración. Las modificaciones son las siguientes:

### 5.8.1 Cambiar conexión y configuración MPI→PPI

Para realizar la transferencia del programa del sistema SCADA a la pantalla táctil necesitamos el cable PPI. Por lo tanto tenemos que cambiar la configuración del panel de control que teníamos ya que para transferir el programa de STEP7 al PLC utilizamos el cable MPI, por lo tanto primero realizaremos la transferencia del programa de STEP7 al autómatas y luego cambiaremos la configuración y el cable para transferir el programa del sistema a la pantalla táctil y lo haremos de la siguiente manera:

-En panel de control hacemos doble click en Setting the PG/PC interface como se muestra en la figura 5.19.

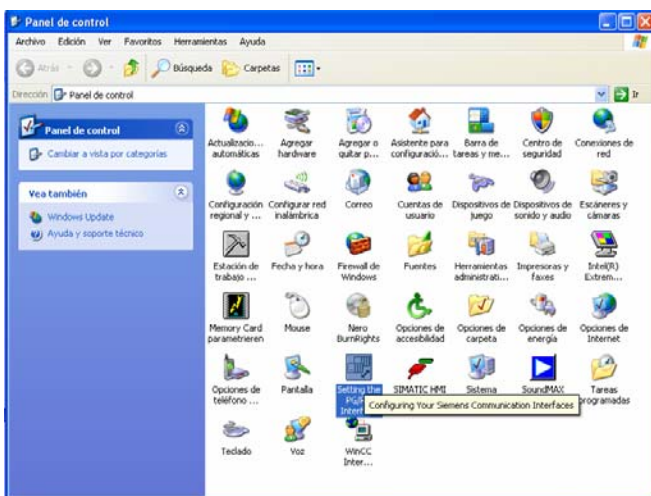


Figura 5.19 Cambiar configuración\_1.



Nos aparece la ventana de la figura 7 en la cual tenemos que cambiar la parametrización utilizada. En estos momentos tendremos PC adapter (MPI) ya que primero hemos realizado la transferencia del programa de STEP7 al PLC. Una vez realizada la transferencia al PLC cambiaremos esta configuración para usar el cable PPI y transferir el programa de WINCC a la pantalla táctil. Como se muestra en la figura 5.20.

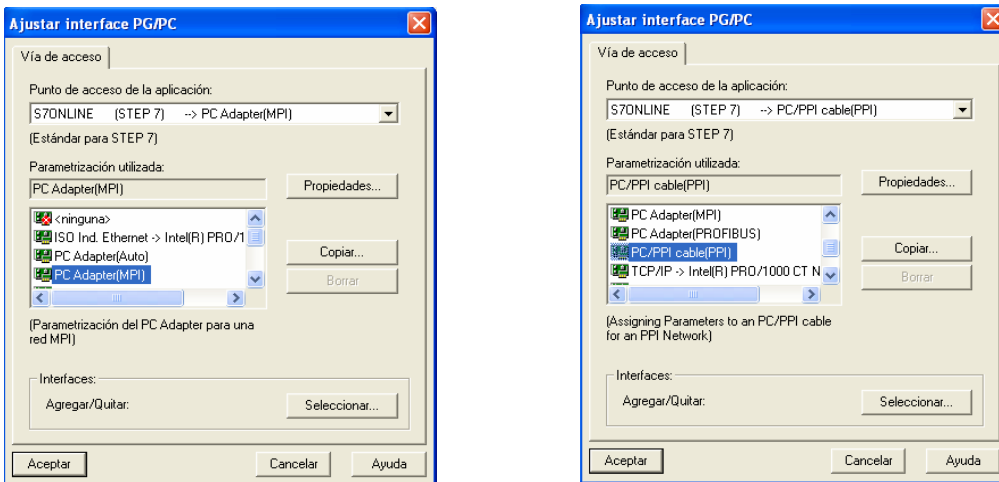


Figura 5. 20 Cambiar configuración

Cuando cambiemos la conexión a cable PPI hacemos doble click sobre este y nos aparece la ventana de propiedades de la figura 5.21 la cual deberemos de configurar como aparece en la siguiente figura.

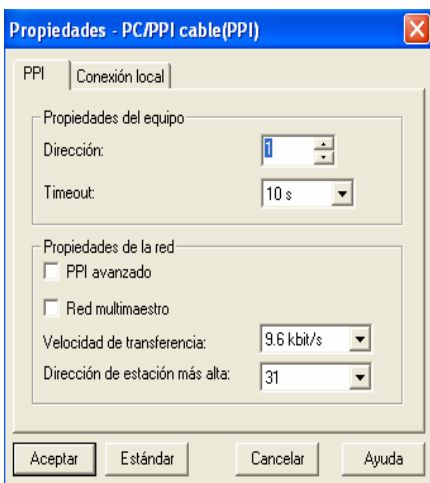
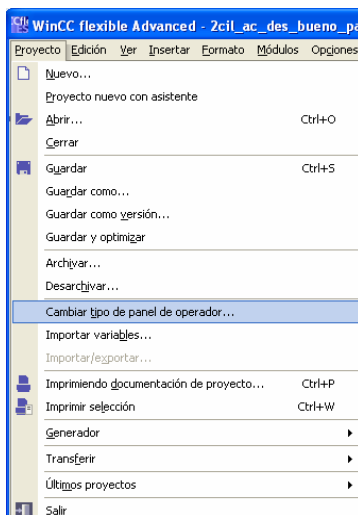


Figura 5. 21 Cambiar configuración\_4.

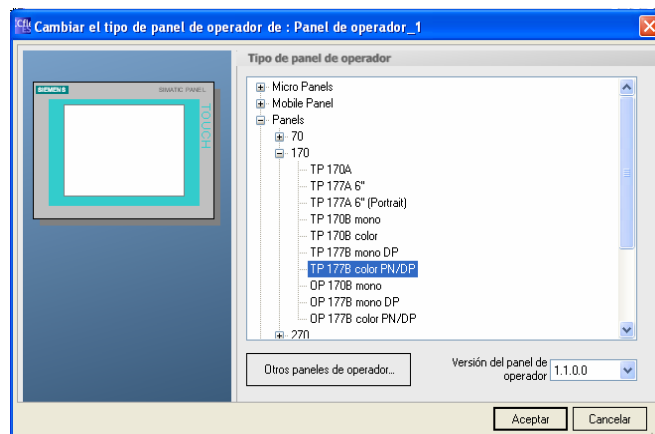
## 5.8.2 Transferir programa SCADA a pantalla táctil.

Una vez realizados estos pasos tendremos que entrar en el sistema SCADA y cambiar la configuración del panel de operador ya que anteriormente hemos estado trabajando con el PC como panel de operador. Para hacer esto vamos a proceder de la siguiente manera:

1. Abrimos el sistema SCADA
2. Una vez dentro del sistema SCADA pinchamos en “Proyecto”, en el menú desplegable que nos aparece pinchamos en “Cambiar tipo de panel de operador”, como se muestra en la figura 5.22 a), nos aparece la ventana de la figura 5.22 b) en la cual tendremos que pinchar en “Panels/170” y en el menú desplegable que nos aparece tendremos que elegir la pantalla que vamos a utilizar, en nuestro caso “TP177B color PN/DP” como nos muestra la figura 5.22 b). Al realizar esta acción, nos aparecerá el panel de operador de esta pantalla el cual es un poco más pequeño que el del PC, por lo tanto hay que ajustar los componentes a este nuevo tamaño.



a)



b)

Figura 5. 22 Cambiar panel de operador: a) Menú “Proyecto” ; b) Cambiar panel de operador

3. Conectamos la pantalla táctil a la fuente de alimentación y a través del cable PPI lo conectamos a nuestro ordenador.
4. Una vez en funcionamiento la pantalla táctil, en el menú que nos aparece en esta pulsamos TRANSFER. En este momento la pantalla ya se encuentra en disposición de recibir los datos de nuestro ordenador.





5. Cuando hayamos realizados estos pasos ya podemos transferir el programa desde el PC a la pantalla táctil y se hace como muestra la figura 5.23.

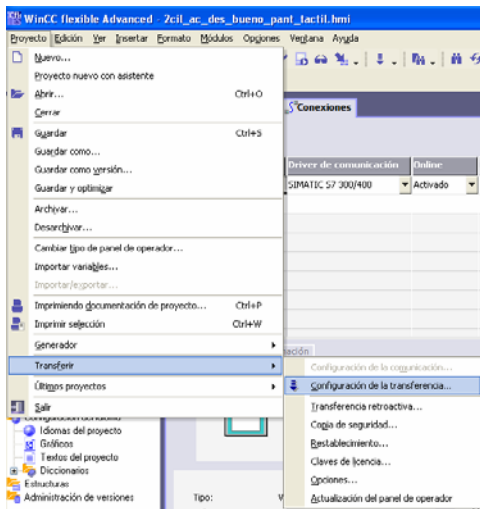


Figura 5. 23 Transferir.

### 5.8.3 Comunicación del PLC con la pantalla táctil.

Para comunicar nuestro autómatas con la pantalla táctil, deberemos de realizar los siguientes pasos:

1. Desconectar el cable PPI de la pantalla.
2. Conectar el cable profibus (cable lila) desde el PLC a la pantalla táctil.
3. Dentro del menú que nos aparece en la pantalla pulsaremos “START”.

Una vez realizados estos pasos ya podemos manejar nuestro robot desde la pantalla táctil.



# Capítulo 6

## Conclusiones y desarrollos futuros.



## 6.1 Conclusiones.

En el presente proyecto se ha realizado la, calibración, programación, control SCADA, Control PID de un cilindro neumático, control PID de un robot de estructura paralela de dos grados de libertad, cinemática directa y cinemática inversa. Para ello se han llevado a cabo varias tareas:

1. Se ha efectuado de forma satisfactoria el conexionado eléctrico de los diferentes componentes que forman el robot.
2. Se ha sustituido la tarjeta de adquisición de datos por el autómatas S7-300 de SIEMENS,
3. Se ha reforzado la unión de los cilindros
4. Se ha mejorado la calibración del robot, ya que en el anterior proyecto la salida que marcaba la referencia y la salida ready del convertidor DADE así como las entradas no estaban conectadas a la tarjeta de adquisición de datos, por lo tanto no se podía saber en que estado se encontraba el convertidor.
5. Se ha aprendido a utilizar STEP7 versión 5.4 para la programación del autómatas S7-300 (CPU 314C-2DP).
6. Se ha utilizado un simulador muy potente PLC-SIM V 5.4 el cual simula de una forma muy eficaz el funcionamiento de nuestro autómatas. Este simulador ha sido muy útil ya que no hemos tenido que tener conectado el PLC al PC para poder realizar nuestros programas y compilarlos de una forma eficaz.
7. También se ha aprendido a utilizar el sistema SCADA para la realización del control SCADA mediante el PC y su posterior transferencia y control mediante la pantalla táctil TP 177B Color PN/DP.
8. Se ha procedido al calibrado y la puesta en marcha de un cilindro neumático.
9. Se ha procedido al calibrado y la puesta en marcha del robot.
10. Se han realizado los cálculos necesarios para realizar la cinemática directa así como la cinemática inversa (en step7).
11. Se han implementado y ensayado un conjunto de sistemas de control de posición de cilindros tanto por separado como con varios cilindros acoplados. Los resultados obtenidos con controladores P, PD, PI y PID en el control de los cilindros por separado han sido satisfactorios para un primer acercamiento al control de los mismos. Sin embargo, al tratar de controlar dos cilindros acoplados, los resultados no han resultado tan satisfactorios, aunque sí aceptables.



## **6.2 Desarrollos futuros.**

Los trabajos futuros deberán encargarse fundamentalmente de mejorar el control conseguido y adecuar con más detalle el hardware de la planta para conseguir un proceso adecuado para distintos problemas de control.

Con estos objetivos, algunas de las tareas aconsejables son:

-Realizar diversos experimentos sobre la cinemática directa y la cinemática inversa para conseguir más precisión. Intentar calcular el espacio de trabajo mejorando el programa de la cinemática inversa, ya que en este proyecto se pueden introducir coordenadas que están fuera del espacio de trabajo.

-Realizar programas en STEP 7 para que el robot realice movimientos predeterminados(rayas, círculos...)

-Fijar el habitáculo en el que se encuentra el robot. Aumentará aún más su precisión y seguridad.

-Cambiar articulaciones rotacionales de la base de los cilindros, por otro componente que tenga menos holgura, si queremos obtener aún más precisión.

-Realizar Script (Programación en C) en STEP 7 para mejorar la programación y obtener mejores resultados.

Investigar a fondo para ver si se puede archivar una variable en menos tiempo del que hemos conseguido en este proyecto, bien cambiando el software utilizado, realizando script o utilizando punteros en STEP 7.

## **6.3 Ventajas e inconvenientes entre la utilización de la tarjeta de adquisición de datos y el autómatas S7-300 de Siemens.**

-La tarjeta funciona a 5v, por lo tanto hay que realizar circuitos de potencia para todos los actuadores que vayamos a conectar a esta, mientras que el PLC funciona a 24v al igual que todos los componentes que componen el robot (no necesitamos ningún circuito adaptador de potencia).

-Una vez programado el autómatas no dependemos de un PC.

-El uso de PLC está más próximo a la industria.

- Mayor robustez del PLC.

- Mayor coste del dispositivo PLC.



## Calibración, Control y Diseño SCADA de un robot paralelo neumático con el autómeta S7-300.

---

---

-La tarjeta de adquisición de datos toma más muestras por segundo que el PLC unas 30 muestras por segundo, mientras que el PLC solo es capaz de tomar 10 muestras.



# Bibliografía



- [1] Step 7. [www.dte.us.es/tec\\_ind/electron/ai/practicas/2005-06/hardwareSTEP7.pdf](http://www.dte.us.es/tec_ind/electron/ai/practicas/2005-06/hardwareSTEP7.pdf)
- [2] FESTO. [http://a1989.g.akamai.net/f/1989/7101/1d/www3.festo.com/\\_C1256D56002E7B89.nsf/html/PSI\\_136\\_2\\_es.pdf/\\$FILE/PSI\\_136\\_2\\_es.pdf](http://a1989.g.akamai.net/f/1989/7101/1d/www3.festo.com/_C1256D56002E7B89.nsf/html/PSI_136_2_es.pdf/$FILE/PSI_136_2_es.pdf)
- [3] FESTO. [http://www.festo.com/INetDomino/coorp\\_sites/es/d828feaec0e5a2eec12572f80032a6a1.htm](http://www.festo.com/INetDomino/coorp_sites/es/d828feaec0e5a2eec12572f80032a6a1.htm)
- [4] FESTO. <http://reviews.lunettehitch.com/F/Festo-dnc-32-125-pva-S10-max-12-bar-pneumatic-cylinder/>
- [5] SIEMENS. <http://support.automation.siemens.com>
- [6] SIEMENS. <http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&objId=2201072&objAction=csOpen&nodeid=10806868&lang=es&siteid=cseus&aprim=0&extranet=standard&viewreg=WW>
- [7] “Modelado, simulación y diseño de un robot paralelo de 2-GDL con actuación neumática”. Realizado por D. Miguel Torregrosa Puig como proyecto final de carrera en la Universidad Politécnica de Cartagena. 2006. El contenido de este proyecto se puede encontrar en la biblioteca de la UPCT.
- [8] Montaje, programación y puesta en marcha de un robot neumático de estructura paralela. Realizado por *Alejandro Rosillo Meseguer* estudiante de la Universidad Politécnica de Cartagena en el año 2007. El contenido de este proyecto se puede encontrar en la biblioteca de la UPCT.
- [9] Automatismos y Cuadros eléctricos. Equipos e Instalaciones Electrotécnicas. Fermín Moreno, Joseba Zubiaurre. CEYSA Editorial Técnica. ISBN: 84-86108 -33-0.
- [10] Automatismos Eléctricos, Neumáticos e Hidráulicos. Florencio J., Cembranos Mistral. Editorial Thomson-Paraninfo. ISBN: 8497323203.
- [11] Autómatas Programables. Teoría y Práctica. N. García, M. Almonacid, R.J. Saltarén, R. Puerto. Universidad Miguel Hernández, 2000.
- [12] Autómatas Programables. Entorno y Aplicaciones. E. Mandado, J. Marcos, C. Fernández, J.I. Armesto, S. Pérez. Ed. Thomson Paraninfo, 2004.
- [13] Autómatas Programables. Josep Balcells, Jose Luis Romeral. Ed. Marcombo, 1997.
- [14] Automatización: problemas resueltos con autómatas programables, J. P. Romera, J. A. Lorite, S. Montoro. Paraninfo, 1994.



[15] Problemas de diseño de automatismos: electrónico -eléctricos y electróniconeumáticos, F. Ojeda Cherta. Paraninfo, 1996.

[16] <http://www.sapiensman.com>

[17] <http://www.control-systems.net>

[18] Universidad de Oviedo. <http://www.isa.uniovi.es/~vsuarez/ii2/index.htm>

[19] Universidad de Oviedo. <http://www.isa.uniovi.es/~vsuarez/Download/WinCCflexible2007-GettingStarted-Opciones.pdf>





# Anexos



## **A.2 Puesta en marcha y desconexión del robot.**

El proceso de **puesta en marcha** del robot debe realizarse en el siguiente orden:

- 1º Conecte el PC y el autómatas.
- 2º Cargue el programa de STEP7 en el autómatas (archivo.s7).
- 3º Conecte la fuente de alimentación.
- 4º Conectar el racor de liberación rápida. Abra el aire a presión a 2 bares de presión. El regulador se deja fijo.
- 5º Ponga el autómatas en modo Run.
- 6º Cargue el programa en el sistema SCADA (.hmi)
- 7º Proceda al calibrado de los cilindros.

**Nota:** Probablemente las válvulas proporcionales MPYE vibren hasta que comiencen a utilizarse.

El proceso de **desconexión** deberá realizarse en el siguiente orden:

- 1º Finalice el programa.
- 2º Desconecte el aire a presión mediante el racor de liberación rápida.
- 3º Desconecte la fuente de alimentación de 24V.
- 4º Ahora puede apagar el ordenador.

El orden de conexión y desconexión es importante para el correcto funcionamiento del robot y por motivos de seguridad. De no seguir este orden el robot podría efectuar movimientos no deseados.

---



## A3.1 Problema\_ 1

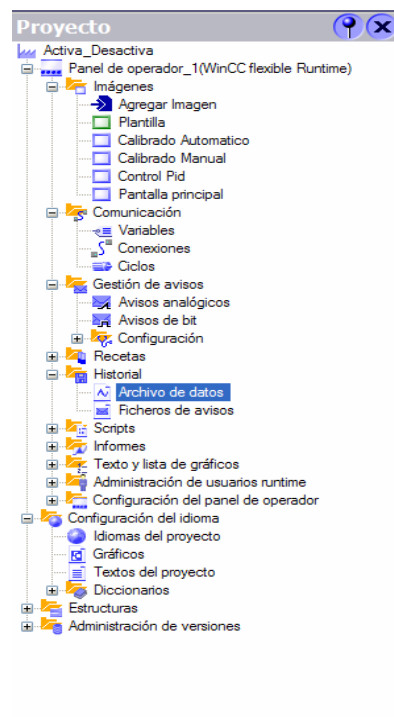
Elección del autómata. En primer lugar se pensó en utilizar CPU 224(S7-200) para el control de un cilindro neumático pero a la hora de conectar el robot (los 2cilindros) se observó que solo posee una salida analógica la cual es insuficiente para el control, por eso se optó por usar la CPU 314C-2DP, S7-300 (disponible en el laboratorio de automática) la cual posee dos salidas analógicas.

## A3.2 Problema\_ 2:

Archivar datos. Para guardar los datos en un archivo se ha utilizado el sistema SCADA el cual nos muestra los datos en una gráfica cada 100 ms, pero a la hora de archivar la variable hemos tenido problemas ya que no hemos podido conseguir ese tiempo, a continuación se detallan los pasos realizados para archivar la variable, así como los problemas con los que nos hemos encontrado.

### A3.2.1 Creación del fichero de datos

En primer lugar pinchamos en la pestaña “archivo de datos” como se muestra en la



Al pulsar do veces sobre la pestaña nos aparece la ventana siguiente.



Nombre	Nº de registros por archivo	Ubicación	Ruta	Modo del origen de datos	Nombre del origen de datos	Método de fichero	Número de registros
pv	500	Archivo	C:\...	Sistema		Fichero cíclico	10
set-point	500	Archivo	C:\D\...	Sistema		Fichero cíclico	10

Nombre	Método de fichero	Número de registros	Nivel de relleno	Activar fichero al inicio de runtime	Respuesta al inicio de runtime	Comentario
pv	Fichero cíclico	10	90	Activado	Continuar fichero	
set-point	Fichero cíclico	10	90	Activado	Continuar fichero	

### Configuración del fichero

En esta ventana se va a definir el fichero donde vamos a guardar las variables, para su posterior procesado.

### A.3.2.2 Configuración de la variable

Una vez creado y configurado el fichero, se configurarán las variables que queramos guardar.

- "fichero" introduciremos el fichero (creado anteriormente) en el que se va a guardar la variable.

-"Modo de adquisición de archivado", tenemos 3 opciones:

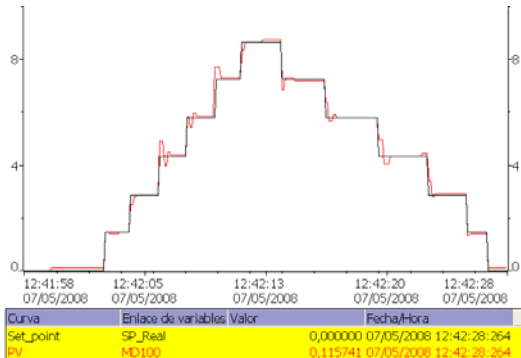
- 1 "Cíclico continuo ": Los valores de variables se archivan de acuerdo con el ciclo de archivo configurado.
- 2 "Al modificar": Los valores de variables se archivan en cuanto el panel de operador detecta una modificación del valor.
- 3 "Sobre demanda": Los valores de variables se archivan al ejecutar la función de sistema "ArchivarVariable".

-Si los valores de variables deben archivarse cíclicamente, seleccione un " ciclo de archivo" o defina un ciclo propio utilizando para ello la lista de objetos en la cual el mínimo ciclo de archivo es de 1 segundo (no se puede poner menos de 1sg) esto representa un problema para nosotros ya que necesitamos que nos guarde la variable cada 100 o 200 ms, como máximo. Por lo tanto esta opción no nos va a ser útil.

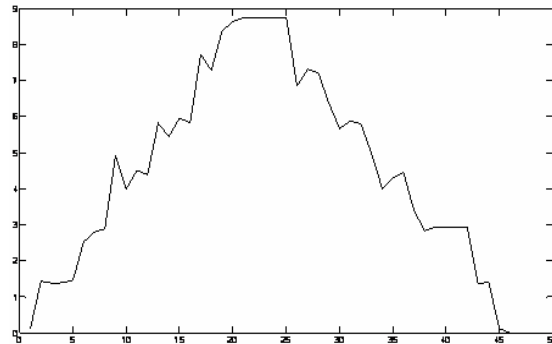
-Si los valores de variables deben archivarse cuando se modifique su valor, el ciclo de archivo se reduce al tiempo aproximado que necesitamos pero nos encontramos con otro problema, cuando la variable del proceso (posición del cilindro) es constante no nos archiva los datos de ese valor ya que archiva cuando modifica su valor.

La demostración de esto se encuentra en las dos gráficas siguientes.

-La gráfica de la izquierda capturada desde el sistema SCADA toma datos cada 100ms (posición del cilindro en rojo) y la segunda(derecha) representada mediante MATLAB desde el **archivo** obtenido de esta gráfica del sistema SCADA(set-point negro).



Gráfica Wincc.



Gráfica Matlab.

-Si los valores de variables deben archivar sobre demanda se hará al ejecutar la función de sistema "Archivar Variable".

Nota: El archivo de Excel del cual hemos sacado la gráfica mediante MATLAB se encuentra en la carpeta del proyecto llamada Archivo Excel.

### A 3.3 Problema\_3: Rotura de la válvula Proporcional.

En un principio esta válvula presentaba todos sus conectores doblados, se procedió a su reparación pero a la hora de sacar la funda en la que iban introducidos nos encontramos con un problema aún más grave que fue la rotura de las pistas, como se muestra en la siguiente figura.



Para que este problema no vuelva a suceder, cuando se vaya a desplazar el habitáculo que contiene el robot, se debe de hacer con todas las válvulas desconectadas.

La solución de este problema fue la compra de otra válvula proporcional igual que la anterior.

En un futuro se debería de cambiar el habitáculo del robot y situarlo en un lugar fijo donde no se pudiera mover y donde la conexión de las válvulas estuviera protegida.

### A3.4 Problema\_4:

Rotura de la pieza de Unión. En el anterior proyecto se utilizó la pieza de unión inicial para que el cilindro formara un ángulo de  $120^\circ$  con la pieza de unión de los 2 cilindros. Este soporte era muy débil ya, la tuerca que llevaba soldada en la punta tenía la rosca comida y estaba soldada en uno de sus lados, era la que realizaba toda la fuerza por que la rótula del cilindro no entraba correctamente.



Pieza de unión inicial.

La solución que se le ha dado a este problema, consiste en la supresión de la tuerca y en la adaptación de cuatro soportes soldados alrededor de la rótula a la que se le ha puesto un brazaete de hierro para poder soldarlos. Dicho cabezal se ha dejado fijo (con el mismo ángulo que poseía) esto hace que la unión sea mucho más resistente y eficaz, ya que es casi imposible que llegue a romperse y produzca daños materiales o personales, la modificación realizada se puede observar en las figuras siguientes. Pieza de unión final\_1, y Pieza de unión final\_2.



Pieza de unión final\_1.



Pieza de unión final\_2.

De esta forma el robot ha funcionado y funcionará correctamente y de una forma segura durante su manejo.

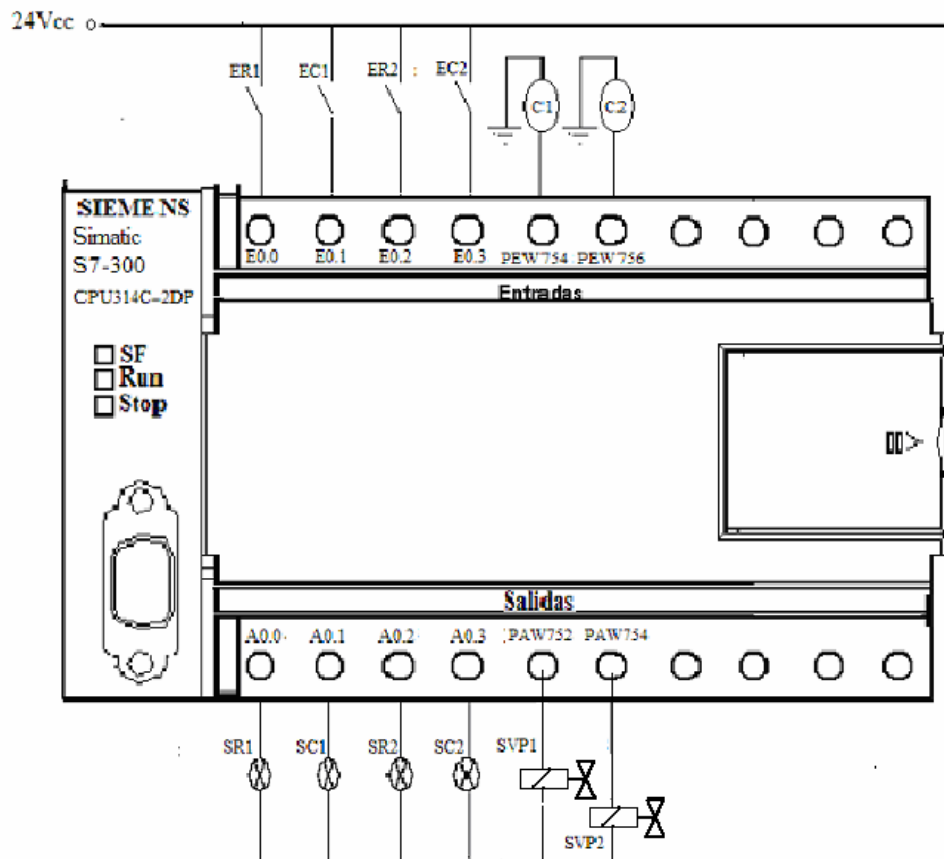


## Calibración, Control y Diseño SCADA de un robot paralelo neumático con el autómatas S7-300 de SIEMENS.

Anexo A9. Esquema de conexiones.

A continuación se muestra el esquema de conexiones al autómatas, en el que se han utilizado 4 entradas digitales, 4 salidas digitales, 2 entradas analógicas y 2 salidas analógicas para realizar la conexión de nuestro robot.

Si queremos realizar pruebas solamente con un cilindro, no hace falta desconectar nada, ya que se utiliza la misma conexión.



- [1] Step 7. [www.dte.us.es/tec\\_ind/electron/ai/practicas/2005-06/hardwareSTEP7.pdf](http://www.dte.us.es/tec_ind/electron/ai/practicas/2005-06/hardwareSTEP7.pdf)
- [2] FESTO. [http://a1989.g.akamai.net/f/1989/7101/1d/www3.festo.com/\\_C1256D56002E7B89.nsf/html/PSI\\_136\\_2\\_es.pdf/\\$FILE/PSI\\_136\\_2\\_es.pdf](http://a1989.g.akamai.net/f/1989/7101/1d/www3.festo.com/_C1256D56002E7B89.nsf/html/PSI_136_2_es.pdf/$FILE/PSI_136_2_es.pdf)
- [3] FESTO. [http://www.festo.com/INetDomino/coorp\\_sites/es/d828feaec0e5a2eec12572f80032a6a1.htm](http://www.festo.com/INetDomino/coorp_sites/es/d828feaec0e5a2eec12572f80032a6a1.htm)
- [4] FESTO. <http://reviews.lunettehitch.com/F/Festo-dnc-32-125-pva-S10-max-12-bar-pneumatic-cylinder/>
- [5] SIEMENS. <http://support.automation.siemens.com>
- [6] SIEMENS. <http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&objId=2201072&objAction=csOpen&nodeid0=10806868&lang=es&siteid=cseus&aprim=0&extranet=standard&viewreg=WW>
- [7] “Modelado, simulación y diseño de un robot paralelo de 2-GDL con actuación neumática”. Realizado por D. Miguel Torregrosa Puig como proyecto final de carrera en la Universidad Politécnica de Cartagena. 2006. El contenido de este proyecto se puede encontrar en la biblioteca de la UPCT.
- [8] Montaje, programación y puesta en marcha de un robot neumático de estructura paralela. Realizado por *Alejandro Rosillo Meseguer* estudiante de la Universidad Politécnica de Cartagena en el año 2007. El contenido de este proyecto se puede encontrar en la biblioteca de la UPCT.
- [9] Automatismos y Cuadros eléctricos. Equipos e Instalaciones Electrotécnicas. Fermín Moreno, Joseba Zubiaurre. CEYSA Editorial Técnica. ISBN: 84-86108 -33-0.
- [10] Automatismos Eléctricos, Neumáticos e Hidráulicos. Florencio J., Cembranos Mistral. Editorial Thomson-Paraninfo. ISBN: 8497323203.
- [11] Automatas Programables. Teoría y Práctica. N. García, M. Almonacid, R.J. Saltarén, R. Puerto. Universidad Miguel Hernández, 2000.
- [12] Automatas Programables. Entorno y Aplicaciones. E. Mandado, J. Marcos, C. Fernández, J.I. Armesto, S. Pérez. Ed. Thomson Paraninfo, 2004.
- [13] Automatas Programables. Josep Balcells, Jose Luis Romeral. Ed. Marcombo, 1997.
- [14] Automatización: problemas resueltos con autómatas programables, J. P. Romera, J. A. Lorite, S. Montoro. Paraninfo, 1994.
- [15] Problemas de diseño de automatismos: electrónico -eléctricos y electróniconeumáticos, F. Ojeda Cherta. Paraninfo, 1996.
- [16] <http://www.sapiensman.com>



[17] <http://www.control-systems.net>

[18] Universidad de Oviedo. <http://www.isa.uniovi.es/~vsuarez/ii2/index.htm>

[19] Universidad de Oviedo. <http://www.isa.uniovi.es/~vsuarez/Download/WinCCflexible2007-GettingStarted-Opciones.pdf>