

## CONVERSIÓN DEL CONTROL DE HUMEDAD Y TEMPERATURA DE UN PLC SLC 500 A S7-1200

Daniel Hernández González, Víctor Manuel Campos Escutia, José Landeros Guzmán  
Universidad Tecnológica del Valle de Toluca  
Carretera del Departamento del D.F. km 7.5, Ejido de Santa María Atarasquillo  
C.P. 52050, Lerma, México  
dany\_white@hotmail.com, torvic\_mce@hotmail.com, ingjoselanderos@hotmail.com

### RESUMEN

En el presente trabajo se plantea la manera de realizar una conversión de la programación de un PLC SLC-500 marca Allen-Bradley a un PLC S7-1200 marca Siemens la cual describe un proceso de control y monitoreo de una cámara de incubación donde se desarrollan vacunas para el sector avícola, en él interviene un conjunto de variables como son temperatura y humedad relativa las cuales no son un clase de datos booleanos sino que adoptan un carácter de tipo de valor analógico. Se verán datos que permiten movimientos de Bits en la memoria de los PLC's requeridos para la manipulación de una entrada analógica, la diferencia del programa original y nuevas funciones a utilizar y los resultados obtenidos de la nueva programación.

### 1. INTRODUCCIÓN

Actualmente en la mayoría de las industrias hay un creciente desarrollo de actualizaciones, en cuanto a su equipo de control y monitoreo de procesos industriales, por lo cual las modificaciones de los mismos, son un paso para el mejoramiento de sus procesos, por ello se realizan grandes cambios de la programación en sus dispositivos de control lógico programable o cambios totales de los mismos.

En la industria farmacéutica, en especial la que desarrolla vacunas para aves de corral, los cambios han sido eminentes en los procesos y en la adquisición de equipo moderno y por ser un proceso muy delicado y de fácil contaminación requiere un control del proceso estricto. Actualmente se tiene antecedentes de paros imprevistos en la producción provocando principalmente en los dispositivos de control de las incubadoras, lo anterior genera desechos de las vacunas se tiene dependencia de control de tres cámaras de incubación por medio de un control lógico programable (PLC) central.

Por lo que se plantea la estructura que rige al Software RSLogix500 de Allen-Bradley, la cual está construida por el árbol de organización, donde se describe las diferentes carpetas de utilidades, desglosando las tareas que proporcionan información de la programación y prioridades para un conjunto de uno o más programas. Y la estructura en el software TIA-POR para el S7-1200 donde se realiza la conversión de la programación original, en el cual aparecerán nuevas carpetas, una llamada HMI donde se tienen utilidades de imágenes, administrador de imágenes, y variables HMI [1], herramientas que permitirán programar un Touch Panel y en la carpeta de PLC se encuentra el bloque de programa donde se ubica el OB1 (Main) donde se realiza la programación en escalera, de acuerdo con la norma ISO 9241-400 [2].

Se analizan los datos que permiten movimientos de Bits en la memoria de los PLC's requeridos para la manipulación de una entrada analógica, y el cómo evitar un desborde de información, causando la saturación de la memoria, mandando al PLC a una falla mayor, además ver la diferencia del programa original y nuevas funciones a utilizar, en base al diagrama de escalera (Ladder Diagram-LD) [3].

### 2. DESARROLLO

El funcionamiento de la cámara de incubación/refrigeración, es por medio de un evaporador, resistencia de calentamiento y una unidad de refrigeración (evaporador, compresor y condensador), con ayuda de estos elementos genera frío o calor en el interior de la cámara, según los parámetros programados, la cámara ésta controlada por un PLC el cual proporciona el punto de ajuste requerido, también cuenta con un sistema de humidificación a través de un aspersor de rocío en el interior de la cámara.

## 2.1 Analogías entre SLC-500 y S7-1200

Para llevar a cabo la conversión del control de humedad y temperatura de un PLC a otro es necesario identificar los tipos de datos a utilizar dentro de la programación, así como los direccionamientos comunes entre los dos PLC a utilizar.

Describiendo las diferentes funciones e instrucciones utilizadas para el desarrollo de la programación tanto en razón de memoria como de datos y algoritmos utilizados. En la tabla 1 se describen los tipos de datos utilizados en la programación de plc's, conocerlos, es una de las ventajas al crear una conversión, ya que cada tipo de dato referente a la cantidad de bits se rigen de la misma manera [4].

Tabla 1 Datos a utilizar en la programación de plc's

Denominación	Bits	Ejemplo	Descripción
<b>BOOL</b>	1	FALSE o TRUE	Variable binaria o lógica (boolean)
<b>INT</b>	16	-32768 ... 32767	Número entero con signo
<b>DINT</b>	32	$-2^{31} \dots +2^{31}-1$	Número entero doble con signo
<b>REAL</b>	32	0.4560	Número real
<b>BYTE</b>	8	0 ... 255	Conjunto de 8 bits
<b>WORD</b>	16	0 ... 65535	Conjunto de 16 bits
<b>DWORD</b>	32	0 ... $2^{32}-1$	Conjunto de 32 bits (double Word)
<b>TIME</b>	32	T#5d4h2m38s3.5ms	Duración
<b>DATE</b>	16	D#2002-01-01	Fecha
<b>TIME_OF_DAY</b>	32	TOD#15:35:08.36	Hora del día
<b>S5TIME</b>	16	S5T#2h2m38s	Duración
<b>DATE_AND_TIME</b>	64	DT#2002-01-01-15:35:08.36	Fecha y hora
<b>CHAR</b>	8	"A"	Carácter
<b>STRING</b>		"AUTOMATA"	Cadena de caracteres

Debido a la diferencia de marcas de PLC's, cada fabricante, tiene su propio software y muchas veces de cada modelo sus propias instrucciones. En la tabla 2 se observan los símbolos e identificadores del PLC SLC-500 y su analogía para el PLC S7-1200 mostrados en la tabla 3.

Tabla 2 Direccionamiento de datos comunes para PLC SLC-500.

PLC SLC-500		
Símbolo	Identificador	Dirección
	I (entrada)	%I0.0 ...
	Q (salida)	%Q0.0 ...
	I, Q, M ... (set)	%I0.0 ... %Q0.0 ... %M0.0 ...
	I, Q, M ... (reset)	%I0.0 ... %Q0.0 ... %M0.0 ...
	DB (bloque de datos)	%DB1 ...
	DB (bloque de datos)	%DB1 ...
	==, <, >, <=, >=, <, > (comparadores)	INT, DINT, REAL

Tabla 3 Direccionamiento de datos comunes para PLC S7-1200.

PLC S7-1200		
Símbolo	Identificador	Dirección
	I (entrada)	I:1/0 ...
	O (salida)	O:1/0 ...
	I, O, M ... (latch)	I:0/0 ... O:0/0 ... M:0/0 ...
	I, O, M ... (unlatch)	I:0/0 ... O:0/0 ... M:0/0 ...
	T (time)	T4:0 ...
	C (count)	C5:0 ...
	A=B, A<B, A>B, A<=, >B (comparadores)	INT, DINT, REAL, DREAL, TIME

Como se aprecia en las tablas anteriores, las funciones de los controladores lógicos programables son compatibles en cuanto su funcionamiento, por ejemplo en los temporizadores se dan condiciones de funcionamiento similares, por ello si tenemos un cambio de 0 a 1 del parámetro de entrada, inicia los temporizadores TON y TONR ó si hay un cambio de 1 a 0 del parámetro de entrada, inicia el temporizador TOF, además de que la cantidad de bits utilizada será equivalente.

La diferencia de programación se denota en condiciones de *palabras direccionables* o *bits direccionables*. Además se debe tomar en cuenta la diferencia de direccionamiento que se le debe de asignar a cada uno de estos.

Ahora debido que el controlador S7-1200 basa su programación en creación de bloque, cabe mencionar que para los valores PT (tiempo predeterminado) y ET (tiempo transcurrido) se almacenan en la memoria como enteros dobles con signo que representan milisegundos. Los datos TIME utilizan el identificador T# y pueden introducirse como unidad de tiempo simple "T#200ms" o como unidades de tiempo compuestas "T#2s\_200ms" [5], como lo muestra la tabla 4.

Tabla 4 Parámetros de tiempo

PLC S7-1200		
Parámetro	Tipo de dato	Descripción
IN	Bool	Habilitar entrada del temporizador
R	Bool	Poner a cero el tiempo transcurrido de TONR
PT (tiempo predeterminado)	Bool	Entrada que indica el tiempo predeterminado
Q	Bool	Salida del temporizador
ET(tiempo transcurrido)	Time	Salida que indica el tiempo transcurrido
Bloque de datos temporizador	DB	Indica que temporizador debe inicializarse con la instrucción RT.

## 2.2 Escalamientos de señales analógicas

Las instrucciones de escalamiento son utilizadas para el tratamiento de las señales analógicas y van direccionadas directamente hacia un logaritmo de la programación (PID).

Para la instrucción SCP del PLC SLC 500, la escala numérica se define como la variable de proceso (PV) y la variable de control (CV). Para

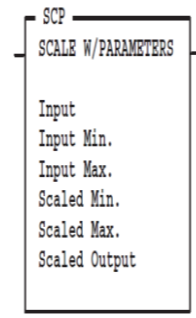
usar unidades de ingeniería, tal como lo requiere el proceso que es la medición de grados y humedad, primero se definen los parámetros que requiere la función de escalamiento y el direccionamiento adecuado hacia el PID, como lo define la tabla 5, se deben escalar sus rangos de E/S analógicas dentro de la escala numérica PV, CV. Para realizarlo se utiliza la instrucción de escala (SCP) que tiene capacidad para valores de entero y punto (coma) flotante.

La fórmula para para convertir los datos de entrada analógicos en unidades de ingeniería:

$$y = mx + b \quad (1)$$

Donde y = salida escalada, m = pendiente (escala máx. – escala mín.) / (entrada máx. – entrada mín.), x = valor de entrada, b = offset (intersección y) = escala mín. – (entrada mín. × inclinación). La entrada mínima y máxima, así como la escala mínima y máxima se usan para determinar los valores de inclinación y offset. El valor de entrada puede salir de los límites de entrada especificados sin requerir la puesta en orden.

Tabla 5 Función SCP

PLC SLC-500		
Símbolo	Parámetro	Ejemplo
	El valor de entrada	Puede ser una dirección de palabra o una dirección de elementos de datos de punto (coma) flotante.
	Los valores mínimos y máximos de entrada	Determinan el rango de datos que aparece en el parámetro de valor de entrada.
	Los valores mínimos y máximos escalonados	Determina el rango de datos que aparece en el parámetro de salida con escala.
	El valor de salida escalonado	Puede ser una dirección de palabra o una dirección de elementos de punto (coma) flotante

Para las instrucciones de escalamiento de los PLC S7-1200 se requiere de dos funciones SCALE\_X y NORM\_X que se describen en la tabla 6. La instrucción SCALE\_X escala el parámetro VALUE real normalizado (donde 0,0 ≤ VALUE ≤ 1,0) al tipo de datos y rango de valores especificados por los parámetros MIN y MAX: OUT = VALUE (MAX - MIN) + MIN. Para

SCALE\_X, los parámetros MIN, MAX y OUT deben tener un mismo tipo de dato.

La instrucción NORM\_X normaliza el parámetro VALUE dentro del rango de valores especificado por los parámetros MIN y MAX:  $OUT = (VALUE - MIN) / (MAX - MIN)$ , donde  $(0,0 \leq OUT \leq 1,0)$ . Para NORM\_X, los parámetros MIN, VALUE y MAX deben tener un mismo tipo de datos.

Tabla 6 Función SCALE\_X y NORM\_X

Símbolo	Parámetro	Tipo de dato	Descripción
	MIN	Sint, Int, DInt, USInt, UInt, UDInt, Real	Entrada que indica el valor mínimo de rango
	VALUE	SCALE_X: Real NORM_X: Sint, Int, DInt, USInt, UInt, UDInt, Real	Valor de entrada que se debe escalar o normalizar
	MAX	Sint, Int, DInt, USInt, UInt, UDInt, Real	Entrada que indica el valor máximo de rango
	OUT	SCALE_X: Sint, Int, DInt, USInt, UInt, UDInt, Real NORM_X: Real	Valor de salida escalonado o normalizado

### 2.3 Configuración del PID

En general los sistemas de control se dividen en lazo abierto y en lazo cerrado, el sistema en lazo abierto se aplica directamente un valor en la entrada y se obtiene un valor de salida al proceso, el cual no tiene efecto sobre la acción de control, en el sistema en lazo cerrado o retroalimentado, la señal de salida tiene efecto sobre la acción de control, ya que es retroalimentado y comparado con una entrada de referencia, con lo que permite rectificar las modificaciones que se producen en todas las entradas del proceso y así controlar el valor de la salida, como se observa en la figura 1, este sistema es el que normalmente se aplica en el

control industrial [6].

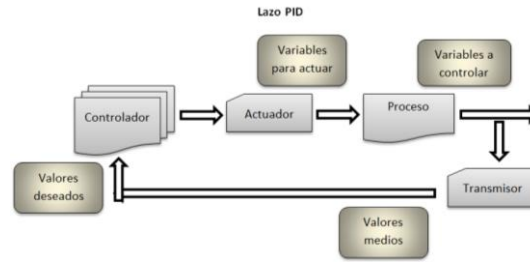


Figura 1 PID lazo cerrado

Para el controlador SLC 500 la ecuación PID controla el proceso enviando una señal de salida al actuador. Cuanto más grande sea el error entre el punto de ajuste y la entrada de variable de proceso, tanto más grande es la señal de salida y viceversa. Un valor adicional (alimentación hacia adelante) se puede añadir a la salida de control como offset. El resultado del cálculo PID (variable de control) impulsará la variable de proceso que controla hacia el punto de ajuste.

Para la programación, la instrucción PID se coloca en un renglón sin lógica condicional, normalmente. La salida permanece en su último valor cuando el renglón es falso. El término integral también se borra cuando el renglón es falso y durante la programación, se introducen las direcciones del bloque de control, variable de proceso y variable de control después de colocar la instrucción PID en un renglón definido como se observa en la tabla 7.

Tabla 7 PID SLC 500


Operando	Dato	Descripción
PID	PID	Estructura PID
Variable de proceso	SINT, INT, DINT, REAL	Valor que desea controlar
Variable retenido	SINT, INT, DINT, REAL	Salida de una estación manual o automática de hardware que evita la salida del controlador (opcional, introducir 0, si se desea este parámetro)
Variable de control	SINT, INT, DINT, REAL	Valor que va a l control del dispositivo final
Lazo maestro	PID	(opcional, si se realiza el control en cascada y este PID es un lazo de esclavo)

Para determinar la configuración de programación del PID\_Compact del PLC S7-1200 la instrucción define cada uno de los parámetros cómo se identifica en la tabla 8.

Tabla 8 PID\_Compact

Instrucción PID\_Compact S7-1200

Símbolo	Parámetro	Dato	Descripción
	Setpoint	Real	Consigna del regulador PID en modo automático. Valor predeterminado: 0.0
	Input	Real	Valor de proceso. Valor predeterminado: 0.0. También debe ponerse sPid_Cmpt.b_Input_PER_On= FALSE.
	Input_PER	Word	Valor de proceso analógico (opcional). Valor predeterminado: W#16#0. También debe ponerse sPid_Cmpt.b_Input_PER_On= TRUE.
	Output	Real	Valor de salida. Valor predeterminado: 0.0
	Output_PER	Word	Valor de salida analógico. Valor predeterminado: W#16#0.
	Output_PWM	Bool	Valor de salida para la modulación de ancho de pulso. Valor predeterminado: FALSE
	State	Int	Estado operativo actual del regulador PID. Valor predeterminado: 0 Use sRet.i_Mode para cambiar el estado. State=0: Inactivo State=1: Optimización inicial. State=2: Optimización final manual State=3: Modo automático State=4: Modo manual
	Error	Dword	Mensaje de error. Valor predeterminado DW#16#0000 (sin errores)



El PID\_Compact ofrece un regulador PID con función de optimización automática para los modos automático y manual, este regulador PID utiliza la fórmula 2 para calcular el valor de salida de la instrucción.

(2)

$$y = K_p \left[ (b \cdot w - x) + \frac{1}{T_i \cdot s} (w - x) + \frac{T_d \cdot s}{a \cdot T_d \cdot s + 1} (c \cdot w - x) \right]$$

Donde y = valor de salida, x = valor de proceso, w = consigna, s = operador laplaciano, k<sub>p</sub> = ganancia proporcional, T<sub>i</sub> = tiempo de acción integral, T<sub>d</sub> = tiempo de acción derivativa, a = coeficiente de retardo derivativo, b = ponderación de acción proporcional, c = ponderación de acción derivativa.

### 3. RESULTADOS

En base a lo obtenido por el escalamiento definido por el programa original se define como se observa en la tabla 9, una vez obtenido se procede a diseñar el algoritmo.

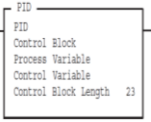
Tabla 9 Parámetros y direcciones para escalamientos obtenidos

Símbolo	Parámetro	Dirección	Descripción
	MIN	0	Entrada mínimo del valor de rango
	VALUE	%IW64	Valor de entrada que se debe escalar o normalizar señal de sensor
	MAX	17150	Entrada máxima del valor de rango
	OUT	%MD44	Valor de salida escalado normalizado a scale

En la configuración de PLC SLC 500 se sintoniza de forma de tiempo continuo ya que el CPU que posee el PLC es un controlador discreto, además un aspecto importante a tomar en la programación del bloque, es la relación que existe entre el período de actualización y la estabilidad del sistema.

De forma general, la programación original estructura la memoria como la variable del control que es el archivo que almacena los datos requeridos para operar la instrucción y la longitud de archivo se fija a 23 palabras, la variable de proceso es la dirección del elemento que almacena el valor de entrada del proceso y la variable de control es la dirección del elemento que almacena la salida del PID de forma que se tiene una configuración como se observa en la tabla 10.

Tabla 10 Configuración PID, dirección y tipo de dato

Símbolo	Dirección	Tipo de dato	Descripción
	N7:2	INT	Bloque de parámetro de función PID
	N7:0	INT	Variable controlada
	N7:1	INT	Salida de controlador

Con la misma configuración base se ajusta el PID\_compact del PLC S7-1200, que de acuerdo a los parámetros requeridos por el sistema se debe regular el PID para controlar un rango de temperatura de 36° C a 37.5°C, y conservar una humedad relativa de 40-60% y mantener la consigna lo más constante posible.

Una vez obtenidos los parámetros requeridos se configura el algoritmo, realizando la estructuración de un bloque de organización OB en el cual el regulador PID queda designado como el FB del programa de simulación, mediante el FB PID\_Compact se integra el objeto tecnológico en el programa de usuario, ya obtenidos estos parámetros el objeto tecnológico PID pone a disposición un regulador PID con autoajuste de optimización donde intervienen dos acciones, la configuración que contiene los valores y parámetros para utilizar el objeto tecnológico y la puesta en marcha desde la ventana para optimizar los parámetros del regulador quedando como se muestra la figura 2.

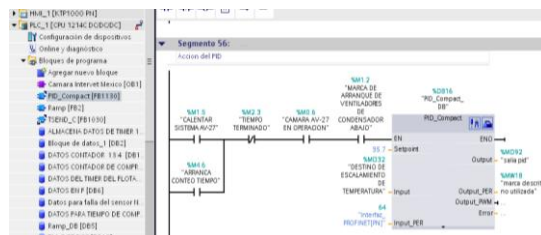


Figura 2 Resultado de la programación PID del S7-1200

#### 4. CONCLUSIONES

Los algoritmos PID y funciones de escalamientos de señales, fueron aplicadas en la programación, de manera satisfactoria, en el controlador lógico programable S7-1200 Siemens, gracias a la

utilizando los módulos de entradas/salidas analógicas, fue posible que el PLC obtuviera información del proceso (monitorear de las variables controladas) y enviará la señal de control a los actuadores (salida del controlador).

Es posible utilizar varios bloques de función tipo PID en un mismo programa, por lo que un PLC es capaz de controlar tantas variables como canales de entrada/salida que posean sus módulos.

#### 5. BIBLIOGRAFÍA

- [1] Nielsen, Jakob. *Usability Engineering*. Ed. Academic Press.1993. ISBN 0-12-518406-9
- [2] Manzini Riccardo, Alberto Regattieri, Hoang Pham, Emilio Ferrari. *Maintenance for industrial systems*. Ed. Springer, 2010, ISBN 978-1-84882-575-8
- [3] Karl-Heinz John, Michael Tiegelkamp. IEC 61131-3 *Programming industrial automation systems*. 2<sup>nd</sup> Edition. Ed. Springer, 2010, ISBN 978-3-642-12014-5.
- [4] Enrique Mandado Pérez, Jorge Marcos Acevedo, Celso Fernández Silva, José I. Armesto Quiroga. *Autómatas programables y sistemas de automatización*. Segunda Edición. Ed. Marcombo. 2009. ISBN 13-978-84267-1575-3
- [5] C.T. Jones. *STEP 7 in Steps*. United States of America. 2006. ISBN 1-889101-03-6
- [6] García Higuera, Andrés. *El control automático en la industria*. Ediciones de la Universidad de la Castilla. La mancha. 2005. ISBN 84-8427-405-5.