

ANEXO DE LA MEMORIA TITULADA:

Diseño e implantación de un sistema SCADA para una planta de producción y envasado de líquidos

realitzado por
Manel Redondo Sol
y dirigido por
Romualdo Moreno Ortiz
Bellaterra, 16 de juny de 2008

ÍNDICE ANEXO

A1. PROCEDIMIENTO DE CALIBRACIÓN DE LOS SENSORES ULTRASÓNICOS	3
A2. CÓDIGO FUENTE DEL PROGRAMA DEL PLC.....	4
A3. CÓDIGO FUENTE DE LOS SCRIPTS ASOCIADOS A LA APLICACIÓN DE SUPERVISIÓN	36

A1. PROCEDIMIENTO DE CALIBRACIÓN DE LOS SENSORES ULTRASÓNICOS

En este apartado se muestra la metodología utilizada para calibrar los sensores ultrasónicos de la planta. Puesto que hay dos tipos se explicarán dos metodologías diferentes.

Para el sensor UB400-12GM-I-V1 la metodología es la siguiente:

- Conectar Ub+ y Ub- a una corriente continua permanentemente, en este caso 24V (Mediante la fuente de alimentación).
- Situar el objeto a medir (líquido) en la posición más alejada (0 cm altura).
- Puentear la señal Teach-In (cable blanco) con Ub- (para facilitar los movimientos se ha optado por utilizar un cable de banana).
- Esperar más de un segundo y retirar la banana del pin negativo de la fuente.
- Llenar tanque hasta la posición más cercana al sensor (27cm altura).
- Puentear la señal Teach-In con Ub+.
- Esperar más de un segundo y apagar fuente.

Para el sensor UC2000-30GM-IUR2-V15 la metodología es la siguiente:

- Apagar la tensión de alimentación.
- Retirar el selector.
- Restaurar la tensión de alimentación. Hasta este punto se ha indicado al sensor que se va a realizar un proceso Teach-In de algún parámetro configurable. Éstos son:
 - Mediante el botón E2/E3 se selecciona el tipo de rampa.
 - A1 corresponde al límite superior.
 - A2 corresponde al límite inferior.
- Se coloca el objeto en la parte más cercana al sensor.
- Mediante el selector se introduce A1 durante unos segundos. Así aprende el límite superior. Mediante el LED indicador se observa si se ha detectado o no el objeto (color verde sí).
- Se coloca el objeto en la parte más alejada del sensor.
- Mediante el selector A2 introducido durante unos segundos se aprende el límite inferior. El LED verde nos indicará que se ha detectado el objeto.
- Enchufar el selector con la entrada E2/E3. Depende si se conecta una, dos o tres veces se obtiene la siguiente funcionalidad:
 1. Rampa ascendente, LED A2 parpadea.
 2. Rampa descendente, LED A1 parpadea (seleccionada en nuestro caso).
 3. Recta punto cero, LEDs A1 y A2 parpadean.
- Enchufando el conector en la posición T se almacenan los cambios y se termina el proceso Teach-In.

A2. CÓDIGO FUENTE DEL PROGRAMA DEL PLC

A continuación se muestra el código fuente del programa de control del PLC realizado en este proyecto. El orden en que están expuestos es:

- Bloques de datos
- Funciones
- Bloques de organización

Para mayor entendimiento, las direcciones reales están etiquetadas según la siguiente tabla:

ETIQUETA	DIRECCIÓN	TIPO DE DATO	SIGNIFICADO
¿altura mínima?	FC 3	FC 3	función que nos devuelve TRUE si tenemos altura mínima en tanque
ALARM_S	SFC 18	SFC 18	Generate Permanently Acknowledged Block-Related Messages
ALARM_SC	SFC 19	SFC 19	Query the Acknowledgment Status of the last ALARM_SQ Entering State Message
ALARM_SQ	SFC 17	SFC 17	Generate Block-Related Messages with Acknowledgment
Altura Buffer L1	PEW 292	WORD	I[4-20]
Altura Buffer L2	PEW 294	WORD	I[4-20]
Altura Tanque Emb L1	PEW 280	WORD	I[4-20]
Altura Tanque Emb L2	PEW 282	WORD	I[4-20]
Altura Tanque Mez L1	PEW 288	WORD	I[4-20]
Altura Tanque Mez L2	PEW 290	WORD	I[4-20]
B7	A 12.2	BOOL	
B8	A 12.3	BOOL	
B9	A 12.4	BOOL	
Batch Embotellado L1	FC 8	FC 8	Implementa los timers asociados a cada tarea
Batch Embotellado L2	FC 7	FC 7	
Botella 1 lista	E 8.0	BOOL	Embotellado L1
Botella 2 lista	E 8.1	BOOL	Embotellado L2
calculo caudal	FC 4	FC 4	Función que calcula el caudal a partir de una altura conocida
caud_ref	FC 12	FC 12	
Caudalimetro B4	PEW 286	WORD	V[0-5]
Caudalimetro B3	PEW 284	WORD	V[0-5]
Cod Control B3	PAW 256	WORD	Código control variador B3 profibus
Cod Control B4	PAW 260	WORD	Código control variador B4
CONT_C	FB 41	FB 41	Continuous Control
Conversion Tiempos Timer	FC 19	FC 19	
conversion V-Q	FC 14	FC 14	
conversion word a real	FC 2	FC 2	función utilizada para convertir los datos leídos de la periferia a REAL
Cycle Execution	OB 1	OB 1	

datos de referencia	DB 2	DB 2	impuestos por nosotros y alterables en cualquier momento
datos paro	DB 5	DB 5	
datos periferia	DB 1	DB 1	
datos reales calculados	DB 3	DB 3	
datos salida PIDs	DB 4	DB 4	datos de salida de los PID's
DIS_AIRT	SFC 41	SFC 41	Delay the Higher Priority Interrupts and Asynchronous Errors
Embotellado L1	FC 1	FC 1	
Embotellado L2	FC 6	FC 6	
EN_AIRT	SFC 42	SFC 42	Enable Higher Priority Interrupts and Asynchronous Errors
EV1	A 16.0	BOOL	Embotellado L1
EV2	A 16.1	BOOL	Embotellado L2
EV3	A 16.2	BOOL	BUFFER 1 L1
EV4	A 16.3	BOOL	BUFFER 1 L2
EV5	A 16.4	BOOL	BUFFER 2 L1
EV6	A 16.5	BOOL	BUFFER 2 L2
EV7	A 16.6	BOOL	Evaporación L1
EV8	A 16.7	BOOL	Evaporación L2
funcion kv-altura	FC 9	FC 9	
G7_STD_3	FC 72	FC 72	
inicializacion parametro	FC 25	FC 25	
L1 Embotellado	E 3.0	BOOL	Control embotellado L1
L1 Evaporación	E 3.4	BOOL	Control Evaporación L1
L1 Mezcla	E 3.2	BOOL	Control Mezcla L1
L1_evaporación	FC 17	FC 17	función implementa timer necesario para llenar el evaporador L1
L1_mezcla	FC 10	FC 10	implementación de dicha etapa
L1_PID	FC 11	FC 11	función donde se ha implementado el PID que controla el nivel de liquido en t_mezcla
L2 Embotellado	E 3.1	BOOL	Control embotellado L2
L2 Evaporación	E 3.5	BOOL	Control Evaporación L2
L2 Mezcla	E 3.3	BOOL	Control Mezcla L2
L2_evaporación	FC 18	FC 18	
L2_mezcla	FC 15	FC 15	etapa de mezcla para la línea 2
L2_PID	FC 16	FC 16	PID tanque mezcla línea 2
M/A	M 20.0	BOOL	En función de esta señal sabemos si se trabaja en Auto o Manual
mA_perif	FC 13	FC 13	
Manual	FC 20	FC 20	
Manual DB	DB 7	DB 7	
Motor Noria L1	A 12.0	BOOL	
Motor Noria L2	A 12.1	BOOL	
Paro Total	E 3.6	BOOL	Control Paro Planta
PID_block	OB 35	OB 35	Bloque cíclico para la ejecución continua de los PIDs
PID1	M 15.0	BOOL	Auto/Manual PID L1

PID2	M 15.1	BOOL	Auto/Manual PID L2
Produccion	DB 6	DB 6	DB enfocado a la pantalla producción del SCADA
RDSYSST	SFC 51	SFC 51	Read a System Status List or Partial List
Referencia Trovis L1	PAW 301	WORD	Ref Trovis L1
Referencia Trovis L2	PAW 303	WORD	Ref Trovis L2
Referencia Variador B1	PAW 297	WORD	Referencia B1
Referencia Variador B2	PAW 299	WORD	Referencia B2
Referencia Variador B3	PAW 258	WORD	Referencia B3 profibus
Referencia Variador B4	PAW 262	WORD	Referencia B4 profibus
SFM_DB	DB 49	FB 49	Bloque de diagnóstico para Notificar errores de sistema
SFM_FB	FB 49	FB 49	Bloque de diagnóstico para Notificar errores de sistema
STP	SFC 46	SFC 46	Change the CPU to STOP
tiempo de llenado	FC 5	FC 5	función que nos calcula el tiempo de llenado correspondiente al caudal calculado
TIME_TCK	SFC 64	SFC 64	Read the System Time
VAT_1	VAT 1		simulación etapa de embotellado
VAT_2	VAT 2		simulación etapa de mezcla
VAT_3	VAT 3		Visualización altura tanques Planta
WR_DPARM	SFC 56	SFC 56	Write Default Parameters
WR_USMSG	SFC 52	SFC 52	Write a User-Defined Diagnostic Event to the Diagnostic Buffer

Tabla 1: Etiquetas de las direcciones reales y su significado

BLOQUES DE DATOS

DATA_BLOCK "datos periferia"
 TITLE =
 VERSION : 0.1

```

STRUCT
altura_1_perif : WORD ; //altura en word periferia
altura_2_perif : WORD ; //altura en word periferia
bomba_mez_1 : REAL ; //salida en real
bomaba_mez_2 : REAL ; //salida real
altura_m1 : WORD ;
altura_m2 : WORD ;
caudalimetro_L1 : WORD ;
caudalimetro_L2 : WORD ;
ref_trovis_B5 : WORD ;
ref_trovis_B6 : WORD ;
altura_b1 : WORD ; //buffer L1
altura_b2 : WORD ; //buffer L2
END_STRUCT ;
BEGIN
altura_1_perif := W#16#0;
altura_2_perif := W#16#0;
bomba_mez_1 := 0.000000e+000;
bomaba_mez_2 := 0.000000e+000;
altura_m1 := W#16#0;
altura_m2 := W#16#0;
caudalimetro_L1 := W#16#0;
caudalimetro_L2 := W#16#0;

```

```

ref_trovis_B5 := W#16#0;
ref_trovis_B6 := W#16#0;
altura_b1 := W#16#0;
altura_b2 := W#16#0;
END_DATA_BLOCK

```

```

DATA_BLOCK "datos de referencia"
TITLE =
VERSION : 0.1

```

```

STRUCT
altura_minima_emb1 : REAL ;
altura_maxima_emb1 : REAL ;
alt_min_sensor : REAL ; //Altura minima de la calibración del sensor
alt_max_sensor : REAL ; //Altura minima de la calibración del sensor
kv_1 : REAL ;
kv_2 : REAL ;
volumen_bot : REAL ; //volumen de la botella a llenar
altura_ref_PID1 : REAL ; //set point para el PID del tanque de mezcla 1 y 2
altura_ref_PID2 : REAL ;
altura_minima_emb2 : REAL ;
B3_ref_manual : REAL ;
B4_ref_manual : REAL ;
END_STRUCT ;
BEGIN
altura_minima_emb1 := 0.000000e+000;
altura_maxima_emb1 := 0.000000e+000;
alt_min_sensor := 0.000000e+000;
alt_max_sensor := 0.000000e+000;
kv_1 := 0.000000e+000;
kv_2 := 0.000000e+000;
volumen_bot := 0.000000e+000;
altura_ref_PID1 := 0.000000e+000;
altura_ref_PID2 := 0.000000e+000;
altura_minima_emb2 := 0.000000e+000;
B3_ref_manual := 0.000000e+000;
B4_ref_manual := 0.000000e+000;
END_DATA_BLOCK

```

```

DATA_BLOCK "datos reales calculados"
TITLE =
VERSION : 0.1

```

```

STRUCT
altura_1_real : REAL ;
altura_2_real : REAL ;
caudal_1 : REAL ;
caudal_2 : REAL ;
TL_1 : REAL ;
TL_2 : REAL ;
altura_M1_real : REAL ;
altura_M2_real : REAL ; //ç
voltage_B3_real_M1 : REAL ;
voltage_B4_real_M2 : REAL ;
caudal_real_L1 : REAL ;
caudal_real_L2 : REAL ;
ref_trovis_M1 : REAL ;
ref_trovis_M2 : REAL ;
altura_b1_real : REAL ;
altura_b2_real : REAL ;
END_STRUCT ;
BEGIN
altura_1_real := 0.000000e+000;
altura_2_real := 0.000000e+000;
caudal_1 := 0.000000e+000;
caudal_2 := 0.000000e+000;
TL_1 := 0.000000e+000;
TL_2 := 0.000000e+000;
altura_M1_real := 0.000000e+000;
altura_M2_real := 0.000000e+000;

```

```

voltage_B3_real_M1 := 0.000000e+000;
voltage_B4_real_M2 := 0.000000e+000;
caudal_real_L1 := 0.000000e+000;
caudal_real_L2 := 0.000000e+000;
ref_trovis_M1 := 0.000000e+000;
ref_trovis_M2 := 0.000000e+000;
altura_b1_real := 0.000000e+000;
altura_b2_real := 0.000000e+000;
END_DATA_BLOCK

```

```

DATA_BLOCK "datos salida PIDs"
TITLE =DATOS DE SALIDA DE LOS PID's
VERSION : 0.0

```

```

STRUCT      // LINEA 1 EMBOTELLADO
LMN_L1_MEZ : REAL ;
LMN_PER_L1 : WORD ;
LMN_P_L1 : REAL ;
LMN_I_L1 : REAL ;
PV_L1 : REAL ;
ER_L1 : REAL ;
LMN_L2_MEZ : REAL ;
LMN_PER_L2 : WORD ;
LMN_P_L2 : REAL ;
LMN_I_L2 : REAL ;
PV_L2 : REAL ;
ER_L2 : REAL ;
END_STRUCT ;
BEGIN
LMN_L1_MEZ := 0.000000e+000;
LMN_PER_L1 := W#16#0;
LMN_P_L1 := 0.000000e+000;
LMN_I_L1 := 0.000000e+000;
PV_L1 := 0.000000e+000;
ER_L1 := 0.000000e+000;
LMN_L2_MEZ := 0.000000e+000;
LMN_PER_L2 := W#16#0;
LMN_P_L2 := 0.000000e+000;
LMN_I_L2 := 0.000000e+000;
PV_L2 := 0.000000e+000;
ER_L2 := 0.000000e+000;
END_DATA_BLOCK

```

```

DATA_BLOCK "datos paro"
TITLE =
VERSION : 0.1

```

```

STRUCT
bomba1 : WORD ;
bomba2 : WORD ;
bomba3 : WORD ;
bomba4 : WORD ;
ref_trovis_B5 : REAL ;
ref_trovis_B6 : REAL ;
END_STRUCT ;
BEGIN
bomba1 := W#16#0;
bomba2 := W#16#0;
bomba3 := W#16#0;
bomba4 := W#16#0;
ref_trovis_B5 := 0.000000e+000;
ref_trovis_B6 := 0.000000e+000;
END_DATA_BLOCK

```

```

DATA_BLOCK "Produccion"
TITLE =
VERSION : 0.1

```

```

STRUCT

```

```

Caudal_Salida_M1 : REAL ;
Caudal_Salida_M2 : REAL ;
Ratio_M1 : REAL ;
Ratio_M2 : REAL ;
Vol_Botella1 : REAL ;
Vol_Botella2 : REAL ;
T_Llenado_EV1 : REAL ;
T_Llenado_EV2 : REAL ;
T_Evap_Proceso_1 : REAL ;
T_Evap_Proceso_2 : REAL ;
T_Espera1 : REAL ;
T_Espera2 : REAL ;
TLEV1 : REAL ;
TLEV2 : REAL ;
TPEV1 : REAL ;
TPEV2 : REAL ;
TEEV1 : REAL ;
TEEV2 : REAL ;
T_Descarga_1 : REAL ;
TDEV1 : REAL ;
T_Descarga_2 : REAL ;
TDEV2 : REAL ;
tlev_1 : REAL ;
tpev_1 : REAL ;
teev_1 : REAL ;
tdev_1 : REAL ;
tlev_2 : REAL ;
tpev_2 : REAL ;
teev_2 : REAL ;
tdev_2 : REAL ;
END_STRUCT ;
BEGIN
Caudal_Salida_M1 := 0.000000e+000;
Caudal_Salida_M2 := 0.000000e+000;
Ratio_M1 := 0.000000e+000;
Ratio_M2 := 0.000000e+000;
Vol_Botella1 := 0.000000e+000;
Vol_Botella2 := 0.000000e+000;
T_Llenado_EV1 := 0.000000e+000;
T_Llenado_EV2 := 0.000000e+000;
T_Evap_Proceso_1 := 0.000000e+000;
T_Evap_Proceso_2 := 0.000000e+000;
T_Espera1 := 0.000000e+000;
T_Espera2 := 0.000000e+000;
TLEV1 := 0.000000e+000;
TLEV2 := 0.000000e+000;
TPEV1 := 0.000000e+000;
TPEV2 := 0.000000e+000;
TEEV1 := 0.000000e+000;
TEEV2 := 0.000000e+000;
T_Descarga_1 := 0.000000e+000;
TDEV1 := 0.000000e+000;
T_Descarga_2 := 0.000000e+000;
TDEV2 := 0.000000e+000;
tlev_1 := 0.000000e+000;
tpev_1 := 0.000000e+000;
teev_1 := 0.000000e+000;
tdev_1 := 0.000000e+000;
tlev_2 := 0.000000e+000;
tpev_2 := 0.000000e+000;
teev_2 := 0.000000e+000;
tdev_2 := 0.000000e+000;
END_DATA_BLOCK

DATA_BLOCK "Manual DB"
TITLE =
VERSION : 0.1

STRUCT
Ref_B3_Manual : WORD ;
Ref_B4_Manual : WORD ;

```

```

END_STRUCT ;
BEGIN
Ref_B3_Manual := W#16#0;
Ref_B4_Manual := W#16#0;
END_DATA_BLOCK

```

```

DATA_BLOCK DB 21
TITLE =
AUTHOR : SIMATIC
FAMILY : ICONT
VERSION : 0.0

```

```

"CONT_C"
BEGIN
COM_RST := FALSE;
MAN_ON := TRUE;
PVPER_ON := FALSE;
P_SEL := TRUE;
I_SEL := TRUE;
INT_HOLD := FALSE;
I_ITL_ON := FALSE;
D_SEL := FALSE;
CYCLE := T#1S;
SP_INT := 0.000000e+000;
PV_IN := 0.000000e+000;
PV_PER := W#16#0;
MAN := 0.000000e+000;
GAIN := 2.000000e+000;
TI := T#20S;
TD := T#10S;
TM_LAG := T#2S;
DEADB_W := 0.000000e+000;
LMN_HLM := 1.000000e+002;
LMN_LLM := 0.000000e+000;
PV_FAC := 1.000000e+000;
PV_OFF := 0.000000e+000;
LMN_FAC := 1.000000e+000;
LMN_OFF := 0.000000e+000;
I_ITLVAL := 0.000000e+000;
DISV := 0.000000e+000;
LMN := 0.000000e+000;
LMN_PER := W#16#0;
QLMN_HLM := FALSE;
QLMN_LLM := FALSE;
LMN_P := 0.000000e+000;
LMN_I := 0.000000e+000;
LMN_D := 0.000000e+000;
PV := 0.000000e+000;
ER := 0.000000e+000;
sInvAlt := 0.000000e+000;
sIanteilAlt := 0.000000e+000;
sRestInt := 0.000000e+000;
sRestDif := 0.000000e+000;
sRueck := 0.000000e+000;
sLmn := 0.000000e+000;
sbArwHLMOn := FALSE;
sbArwLLMOn := FALSE;
sbILlmOn := TRUE;
END_DATA_BLOCK

```

```

DATA_BLOCK DB 22
TITLE =
AUTHOR : SIMATIC
FAMILY : ICONT
VERSION : 0.0

```

```

"CONT_C"
BEGIN
COM_RST := FALSE;
MAN_ON := TRUE;
PVPER_ON := FALSE;

```

```

P_SEL := TRUE;
I_SEL := TRUE;
INT_HOLD := FALSE;
I_ITL_ON := FALSE;
D_SEL := TRUE;
CYCLE := T#1S;
SP_INT := 0.000000e+000;
PV_IN := 0.000000e+000;
PV_PER := W#16#0;
MAN := 0.000000e+000;
GAIN := 4.000000e+001;
TI := T#2S;
TD := T#1S;
TM_LAG := T#2S;
DEADB_W := 0.000000e+000;
LMN_HLM := 1.000000e+002;
LMN_LLM := 0.000000e+000;
PV_FAC := 1.000000e+000;
PV_OFF := 0.000000e+000;
LMN_FAC := 1.000000e+000;
LMN_OFF := 0.000000e+000;
I_ITLVAL := 0.000000e+000;
DISV := 0.000000e+000;
LMN := 0.000000e+000;
LMN_PER := W#16#0;
QLMN_HLM := FALSE;
QLMN_LLM := FALSE;
LMN_P := 0.000000e+000;
LMN_I := 0.000000e+000;
LMN_D := 0.000000e+000;
PV := 0.000000e+000;
ER := 0.000000e+000;
sInvAlt := 0.000000e+000;
sIanteilAlt := 0.000000e+000;
sRestInt := 0.000000e+000;
sRestDif := 0.000000e+000;
sRueck := 0.000000e+000;
sLmn := 0.000000e+000;
sbArwHLmOn := FALSE;
sbArwLLmOn := FALSE;
sbILimOn := TRUE;
END_DATA_BLOCK

```

FUNCIONES

```

FUNCTION "conversion word a real" : VOID
TITLE =
//funcion utilizada para convertir los datos leidos de la periferia en WORD a
//datos REAL listos para ser manejados.
VERSION : 0.1

```

```

VAR_INPUT
CANAL : WORD ;    //valor leído periferia
LIM_SUP : REAL ;  //limite superior
LIM_INF : REAL ;  //limite inferior
END_VAR
VAR_OUTPUT
SALIDA : REAL ;    //valor de la periferia en real
END_VAR
VAR_TEMP
SALIDA_PROV : REAL ;
TINT : INT ;
TDOUBLEINT : DINT ;
END_VAR
BEGIN
NETWORK
TITLE =

L      #CANAL;

```

```

T      #TINT;
ITD    ;
T      #TDOUBLEINT;
DTR    ;
T      #SALIDA_PROV;
L      #LIM_SUP;
L      #LIM_INF;
-R     ;
L      2.764800e+004;
/R     ;
L      #SALIDA_PROV;
*R     ;
L      #LIM_INF;
+R     ;
T      #SALIDA;
END_FUNCTION

```

```

FUNCTION "caud_ref" : VOID
TITLE =CONVERSION CAUDAL BOMBAS 3 Y 4 EN REFERENCIA BOMBAS 5 Y 6
VERSION : 0.0

```

```

VAR_INPUT
CAUDAL : REAL ;
REL_CAUDAL : REAL ;
DEPEN : REAL ;
INDEPEN : REAL ;
END_VAR
VAR_OUTPUT
REF : REAL ;
END_VAR
VAR_TEMP
REF_BTM : REAL ;
END_VAR
BEGIN
NETWORK
TITLE =Q a I
//ECUACION: REFERENCIA=CAUDAL * CONSIGNA (R)
//ECUACION DE SALIDA: REF(Ma)= DEPEN*REFERENCIA - INDEPEN
L      #CAUDAL;
L      #REL_CAUDAL;
*R     ;
T      #REF_BTM;
L      #DEPEN;
*R     ;
L      #INDEPEN;
+R     ;
T      #REF;

```

```

END_FUNCTION

```

```

FUNCTION "ma_perif" : VOID
TITLE = DATO DE CORRIENTE EN DATO PARA LA PERIFERIA
VERSION : 0.0

```

```

VAR_INPUT
REF : REAL ;
END_VAR
VAR_OUTPUT
REF_TROVIS : WORD ;
END_VAR
VAR_TEMP
DATO : DWORD ;
END_VAR
BEGIN
NETWORK
TITLE =

```

```

L      #REF;
L      2.000000e+001;

```

```

-R      ;
SPP     no;

L      #REF;
L      4.000000e+000;
-R      ;
SPM     noo;

L      #REF;
SPA     si;

no:     L      2.000000e+001;
SPA     si;
noo:    L      0.000000e+000;

si:     L      4.000000e+000;

-R      ;
L      2.764800e+004;
*R      ;
L      1.600000e+001;
/R      ;
RND     ;
T      #REF_TROVIS;
T      "datos_periferia".ref_trovis_B5;

```

```
END_FUNCTION
```

```

FUNCTION "conversion V-Q" : VOID
TITLE =
VERSION : 0.1

```

```

VAR_INPUT
VOLTAJE : REAL ;
DEPEN   : REAL ;
INDEPEN : REAL ;
END_VAR
VAR_OUTPUT
REF     : REAL ;
END_VAR
BEGIN
NETWORK
TITLE =

```

```

L      0.000000e+000;
L      #VOLTAJE;
-R      ;
SPZ     S1;
//salto condicional
L      #VOLTAJE;
L      #DEPEN;
*R      ;
L      #INDEPEN;
+R      ;
SPA     fi;
S1:     L      0.000000e+000;
fi:     T      #REF;

```

```
END_FUNCTION
```

```

FUNCTION "¿altura minima?" : VOID
TITLE =
//Funcion que nos hace la comparacion de la altura actual que tenemos en el
//tanque con la altura minima necesaria para embotellar.
VERSION : 0.1

```

```

VAR_INPUT
altura_real : REAL ;
altura_compara : REAL ;
END_VAR

```

```

VAR_OUTPUT
SINO : BOOL ;
END_VAR
BEGIN
NETWORK
TITLE =

L      #altura_compara;
L      #altura_real;
-R      ;
SPPZ   v3;
SPA    v4;
NETWORK
TITLE =

v3:    UN      M      8.0;
R      #SINO;
SPA    v5;
NETWORK
TITLE =

v4:    UN      M      8.0;
S      #SINO;

NETWORK
TITLE =

v5:    BE      ;

END_FUNCTION

FUNCTION "calculo caudal" : VOID
TITLE =
//Funcion que nos calcula el caudal que sale por una electrovalvula, con kv
//conocido, en funcion de la altura del tanque.
VERSION : 0.1

VAR_INPUT
altura_real : REAL ;
kv : REAL ;
END_VAR
VAR_OUTPUT
caudal : REAL ;
END_VAR
BEGIN
NETWORK
TITLE =

L      9.800000e-002; //cargamos gravedad
L      #altura_real;
*R      ;
NETWORK
TITLE =

SQRT   ;
NETWORK
TITLE =

L      #kv;
*R      ;
NETWORK
TITLE =

L      1.000000e+002;
*R      ;
NETWORK
TITLE =

L      6.000000e+000;

```

```

/R      ;
NETWORK
TITLE =

T      #caudal;

END_FUNCTION

FUNCTION "tiempo de llenado" : VOID
TITLE =
//Funcion que nos calcula el Tiempo de Llenado correspondiente a la apertura de
//la e-valvula.
VERSION : 0.1

VAR_INPUT
caudal : REAL ;
VolBotella : REAL ;
END_VAR
VAR_OUTPUT
TL : REAL ;
END_VAR
BEGIN
NETWORK
TITLE =
//pasamos el tiempo de minutos a segundos
L      #VolBotella; //cargamos el volumen de una botella
L      #caudal; //cargamos caudal y dividimos
/R      ;

L      6.000000e+001; //transformamos a segundos el tiempo de llenado
*R      ;

L      1.000000e+002; //centesimas de segundo
*R      ;
RND-   ; //redondear hacia abajo
DTB    ; //pasar a BCD
T      #TL; //guardamos el tiempo
END_FUNCTION

FUNCTION "Batch Embotellado L2" : VOID
TITLE =
VERSION : 0.1

VAR_TEMP
bobina_T11 : BOOL ;
bobina_t12 : BOOL ;
bobina_t13 : BOOL ;
END_VAR
BEGIN
NETWORK
TITLE =

U      M      10.1;
S      M      2.7;

NETWORK
TITLE =

U      M      2.7;
U      M      2.0;
U      "Botella 2 lista";
R      "Motor Noria L2";
=      M      4.0;

NETWORK
TITLE =

U      M      4.0;
FR     T      11;

```

```

L      DB3.DBW    22; //cargamos el tiempo de llenado
SE     T         11;
U      T         11;
=      #bobina_T11;

```

```

NETWORK
TITLE =

```

```

UN     #bobina_T11;
U      M         4.0;
=      "EV2";
NETWORK
TITLE =

```

```

U      #bobina_T11;
S      M         2.1;
R      T         11;
R      M         2.0;
NETWORK
TITLE =

```

```

U      M         2.1;
R      "EV2";
=      M         4.1;

```

```

NETWORK
TITLE =

```

```

U      M         4.1;
FR     T         12;
L      S5T#500MS;
SE     T         12;
U      T         12;
=      #bobina_t12;

```

```

NETWORK
TITLE =

```

```

U      #bobina_t12;
R      T         12;
R      M         2.1;
S      M         2.2;
NETWORK
TITLE =

```

```

U      M         2.2;
=      M         4.2;

```

```

NETWORK
TITLE =

```

```

U      M         4.2;
FR     T         13;
L      S5T#50MS;
SE     T         13;
U      T         13;
=      #bobina_t13;

```

```

NETWORK
TITLE =

```

```

UN     #bobina_t13;

```

```

U      M      4.2;
=      "Motor Noria L2";
R      "Botella 2 lista";
NETWORK
TITLE =

```

```

U      #bobina_t13;
S      M      2.0;
R      T      13;
R      M      2.2;
R      M      2.7;
END_FUNCTION

```

```

FUNCTION "Batch Embotellado L1" : VOID
TITLE =
VERSION : 0.1

```

```

VAR_TEMP
bobina_t1 : BOOL ;
bobina_t2 : BOOL ;
bobina_t3 : BOOL ;
END_VAR
BEGIN
NETWORK
TITLE =

```

```

U      M      10.0;
S      M      1.7;

```

```

NETWORK
TITLE =

```

```

U      M      1.7;
U      M      1.0;
U      "Botella 1 lista";
R      "Motor Noria L1";
=      M      5.0;

```

```

NETWORK
TITLE =

```

```

U      M      5.0;
FR     T      1;
L      DB3.DBW 18; //cargamos el tiempo de llenado
SE     T      1;
U      T      1;
=      #bobina_t1;

```

```

NETWORK
TITLE =

```

```

UN     #bobina_t1;
U      M      5.0;
=      "EV1";
NETWORK
TITLE =

```

```

U      #bobina_t1;
S      M      1.1;
R      T      1;
R      M      1.0;
NETWORK
TITLE =

```

```

U      M      1.1;

```

```

R      "EV1";
=      M      5.1;

NETWORK
TITLE =

U      M      5.1;
FR     T      2;
L      S5T#500MS; //cargamos el tiempo de llenado
SE     T      2;
U      T      2;
=      #bobina_t2;

```

```

NETWORK
TITLE =

```

```

U      #bobina_t2;
R      T      2;
R      M      1.1;
S      M      1.2;
NETWORK
TITLE =

```

```

U      M      1.2;
=      M      5.2;

```

```

NETWORK
TITLE =

```

```

U      M      5.2;
FR     T      3;
L      S5T#250MS;
SE     T      3;
U      T      3;
=      #bobina_t3;

```

```

NETWORK
TITLE =

```

```

UN     #bobina_t3;
U      M      5.2;
=      "Motor Noria L1";
R      "Botella 1 lista";
NETWORK
TITLE =

```

```

U      #bobina_t3;
S      M      1.0;
R      T      3;
R      M      1.2;
R      M      1.7;
END_FUNCTION

```

```

FUNCTION "funcion kv-altura" : VOID
TITLE =
VERSION : 0.1

```

```

VAR_INPUT
g3 : REAL ;
g2 : REAL ;
g1 : REAL ;
ind : REAL ;
altura : REAL ;
END_VAR

```

```

VAR_OUTPUT
kv : REAL ;
END_VAR
VAR_TEMP
temp3 : REAL ;
temp2 : REAL ;
END_VAR
BEGIN
NETWORK
TITLE =

L      #altura;
L      #altura;
*R      ;
*R      ;
NETWORK
TITLE =

L      #g3;
*R      ;
T      #temp3;
NETWORK
TITLE =

L      #altura;
L      #altura;
*R      ;
L      #g2;
*R      ;
T      #temp2;
NETWORK
TITLE =

L      #altura;
L      #g1;
*R      ;
NETWORK
TITLE =

L      #ind;
+R      ;
NETWORK
TITLE =

L      #temp2;
+R      ;
NETWORK
TITLE =

L      #temp3;
+R      ;
T      #kv;
END_FUNCTION

FUNCTION "L1_mezcla" : VOID
TITLE =
VERSION : 0.1

BEGIN
NETWORK
TITLE =
//Se lee dato de la periferia (caudalimetro linea 1)
L      "Caudalimtero B3";
T      "datos periferia".caudalimetro_L1;
NETWORK
TITLE =
//Se interpretará la señal del caudalímetro como una señal de corriente, para ser
//posteriormente aplicada la relación caudal-corriente y, así, obtener el caudal.
CALL "conversion word a real" (
CANAL                                     := "datos periferia".caudalimetro_L1,
LIM_SUP                                  := 5.000000e+000,
LIM_INF                                  := 0.000000e+000,

```

```

SALIDA                                     := "datos reales calculados".voltage_B3_real_M1);

NETWORK
TITLE =
//Se tiene que aplicar la fórmula encontrada caudal-corriente para obtener el
//caudal asociado a la señal real de corriente obtenida.
CALL "conversion V-Q" (
VOLTAJE                                     := "datos reales calculados".voltage_B3_real_M1,
DEPEN                                       := 1.128800e+000,
INDEPEN                                    := 6.822000e-001,
REF                                         := "datos reales calculados".caudal_real_L1);

//se aprovecha la función fc12 para aplicar la formula de primer grado de la bomba
//Q=1.1288V+0.6822
NETWORK
TITLE =
//Convertimos señal de caudal a señal de corriente para referencia de trovis. Se
//aplica la fórmula de conversión caudal-corriente asociada a la bomba 5
//controlada por trovis.

CALL "caud_ref" (
CAUDAL                                     := "datos reales calculados".caudal_real_L1,
REL_CAUDAL                               := "Produccion".Ratio_M1,
DEPEN                                     := 2.020600e+000,
INDEPEN                                  := 3.199900e+000,
REF                                       := "datos reales calculados".ref_trovis_M1);
//I=2.0206Q+3.1999

NETWORK
TITLE =
//Conversion corriente a word y envio de la referencia

CALL "mA_perif" (
REF                                       := "datos reales calculados".ref_trovis_M1,
REF_TROVIS                               := "Referencia Trovis L1");

END_FUNCTION

FUNCTION "L2_mezcla" : VOID
TITLE =
VERSION : 0.1

BEGIN
NETWORK
TITLE =
//Se lee dato de la periferia (caudalimetro linea 1)
L      "Caudalimetro B4";
T      "datos periferia".caudalimetro_L2;
NETWORK
TITLE =
//Se interpretará la señal del caudalímetro como una señal de corriente, para ser
//posteriormente aplicada la relación caudal-corriente y, así, obtener el caudal.
CALL "conversion word a real" (
CANAL                                     := "datos periferia".caudalimetro_L2,
LIM_SUP                                  := 5.000000e+000,
LIM_INF                                  := 0.000000e+000,
SALIDA                                   := "datos reales calculados".voltage_B4_real_M2);

NETWORK
TITLE =
//Se tiene que aplicar la fórmula encontrada caudal-corriente para obtener el
//caudal asociado a la señal real de corriente obtenida.
CALL "conversion V-Q" (
VOLTAJE                                   := "datos reales calculados".voltage_B4_real_M2,
DEPEN                                     := 1.199300e+000,
INDEPEN                                  := 7.379000e-001,
REF                                       := "datos reales calculados".caudal_real_L2);

//se aprovecha la función fc12 para aplicar la formula de primer grado de la bomba
//Q=1.1993V+0.7379
NETWORK
TITLE =

```

```
//Convertimos señal de caudal a señal de corriente para referencia de trovis. Se
//aplica la fórmula de conversión caudal-corriente asociada a la bomba 5
//controlada por trovis.
```

```
CALL "caud_ref" (
CAUDAL           := "datos reales calculados".caudal_real_L2,
REL_CAUDAL       := "Produccion".Ratio_M2,
DEPEN            := 2.052700e+000,
INDEPEN          := 3.832300e+000,
REF              := "datos reales calculados".ref_trovis_M2);
//I=2.0527Q+3.8323
NETWORK
TITLE =
//Conversion corriente a word y envio de la referencia
CALL "mA_perif" (
REF              := "datos reales calculados".ref_trovis_M2,
REF_TROVIS       := "Referencia Trovis L2");
//      CALL "mA_perif"
//      REF      :=1.000000e+001
//      REF_TROVIS:=PAW303
END_FUNCTION
```

```
FUNCTION "Paro Mezcla L1" : VOID
TITLE =
VERSION : 0.1
```

```
BEGIN
NETWORK
TITLE =
```

```
CALL "mA_perif" (
REF              := 0.000000e+000,
REF_TROVIS       := "Referencia Trovis L1");
END_FUNCTION
```

```
FUNCTION "Paro Mezcla L2" : VOID
TITLE =
VERSION : 0.1
```

```
BEGIN
NETWORK
TITLE =
```

```
CALL "mA_perif" (
REF              := 0.000000e+000,
REF_TROVIS       := "Referencia Trovis L2");
```

```
END_FUNCTION
```

```
FUNCTION "Embotellado L1" : VOID
TITLE =
VERSION : 0.1
```

```
BEGIN
NETWORK
TITLE =
```

```
CALL "¿altura minima?" (
altura_real      := "datos reales calculados".altura_1_real,
altura_compara   := "datos de referencia".altura_minima_emb1,
SINO             := M      10.0);
```

```
NETWORK
TITLE =
```

```
CALL "funcion kv-altura" (
g3               := -9.905000e-001,
g2               := 2.531200e+000,
g1               := -2.212000e+000,
ind              := 1.828500e+000,
altura           := "datos reales calculados".altura_1_real,
```

```

kv                                     := "datos de referencia".kv_1);

NETWORK
TITLE =

CALL "calculo caudal" (
altura_real                           := "datos reales calculados".altura_1_real,
kv                                    := "datos de referencia".kv_1,
caudal                               := "datos reales calculados".caudal_1);
NETWORK
TITLE =

CALL "tiempo de llenado" (
caudal                               := "datos reales calculados".caudal_1,
VolBotella                           := "Produccion".Vol_Botella1,
TL                                    := "datos reales calculados".TL_1);

NETWORK
TITLE =

CALL "Batch Embotellado L1" ;

END_FUNCTION

FUNCTION "Embotellado L2" : VOID
TITLE =
VERSION : 0.1

BEGIN
NETWORK
TITLE =

CALL "funcion kv-altura" (
g3                                   := -3.812000e-001,
g2                                   := 1.162600e+000,
g1                                   := -1.269000e+000,
ind                                  := 1.695200e+000,
altura                              := "datos reales calculados".altura_2_real,
kv                                   := "datos de referencia".kv_2);

NETWORK
TITLE =

CALL "?altura minima?" (
altura_real                         := "datos reales calculados".altura_2_real,
altura_compara                     := "datos de referencia".altura_minima_emb2,
SINO                                := M      10.1);

NETWORK
TITLE =

CALL "calculo caudal" (
altura_real                         := "datos reales calculados".altura_2_real,
kv                                   := "datos de referencia".kv_2,
caudal                             := "datos reales calculados".caudal_2);

NETWORK
TITLE =

CALL "tiempo de llenado" (
caudal                             := "datos reales calculados".caudal_2,
VolBotella                         := "Produccion".Vol_Botella2,
TL                                  := "datos reales calculados".TL_2);

NETWORK
TITLE =

CALL "Batch Embotellado L2" ;

END_FUNCTION

```

```

FUNCTION "L1_PID" : VOID
TITLE =
VERSION : 0.1

BEGIN
NETWORK
TITLE =
//Introducimos los valores correspondientes al PID. Se marca un set point de 50cm
//y los datos de entrada se introducen directamente en formato WORD de periferia.
//El valor final que nos interesa será LMN_PER, ya preparado para enviarlo a la
//periferia.

CALL "CONT_C" , DB      22 (
MAN_ON              := "PID1",
PVPER_ON            := FALSE,
CYCLE                := T#100MS,
SP_INT              := DB22.DBD      6, //se quiere fijar consigna de altura a
20cm
PV_IN                := "datos reales calculados".altura_M1_real,
PV_PER              := "datos periferia".altura_m1,
MAN                  := "datos de referencia".B3_ref_manual,
LMN                  := "datos salida PIDs".LMN_L1_MEZ,
LMN_PER              := "datos salida PIDs".LMN_PER_L1,
LMN_P                := "datos salida PIDs".LMN_P_L1,
LMN_I                := "datos salida PIDs".LMN_I_L1,
PV                   := "datos salida PIDs".PV_L1,
ER                   := "datos salida PIDs".ER_L1);
NETWORK
TITLE =

L      "datos salida PIDs".LMN_PER_L1;
// T      "Referencia Variador B3"
T      "Manual DB".Ref_B3_Manual;
END_FUNCTION

FUNCTION "L2_PID" : VOID
TITLE =
VERSION : 0.1

BEGIN
NETWORK
TITLE =
//Introducimos los valores correspondientes al PID. Se marca un set point de 50cm
//y los datos de entrada se introducen directamente en formato WORD de periferia.
//El valor final que nos interesa será LMN_PER, ya preparado para enviarlo a la
//periferia.

CALL "CONT_C" , DB      23 (
MAN_ON              := "PID2",
PVPER_ON            := FALSE,
CYCLE                := T#100MS,
SP_INT              := DB23.DBD      6,
PV_IN                := "datos reales calculados".altura_M2_real,
PV_PER              := "datos periferia".altura_m2,
MAN                  := "datos de referencia".B4_ref_manual,
LMN                  := "datos salida PIDs".LMN_L2_MEZ,
LMN_PER              := "datos salida PIDs".LMN_PER_L2,
LMN_P                := "datos salida PIDs".LMN_P_L2,
LMN_I                := "datos salida PIDs".LMN_I_L2,
PV                   := "datos salida PIDs".PV_L2,
ER                   := "datos salida PIDs".ER_L2);
NETWORK
TITLE =

L      "datos salida PIDs".LMN_PER_L2;
T      "Manual DB".Ref_B4_Manual;
END_FUNCTION

FUNCTION "L1_evaporación" : VOID
TITLE =

```

```
VERSION : 0.1
```

```
VAR_TEMP
bobina_T4 : BOOL ;
bobina_T5 : BOOL ;
bobina_T6 : BOOL ;
bobina_T7 : BOOL ;
bobina_T24 : BOOL ;
bobina_T25 : BOOL ;
bobina_T26 : BOOL ;
bobina_T27 : BOOL ;
END_VAR
BEGIN
NETWORK
TITLE =Evaporación L1
```

```
U      A      4.0;
R      "EV7";
=      M      6.0;
```

```
NETWORK
TITLE =
```

```
U      M      6.0;
FR     T      4;
L      DB6.DBW 50; //cargamos el tiempo de llenado
SE     T      4;
U      T      4;
=      #bobina_T4;
```

```
NETWORK
TITLE =
```

```
U      #bobina_T4;
U      M      6.0;
FR     T      24;
L      DB6.DBW 90; //cargamos el tiempo de llenado   ms
SE     T      24;
U      T      24;
=      #bobina_T24;
```

```
NETWORK
TITLE =
```

```
UN     #bobina_T4;
U      M      6.0;
=      "B7";
```

```
NETWORK
TITLE =
```

```
UN     #bobina_T24;
U      M      6.0;
=      "B7";
```

```
NETWORK
TITLE =
```

```
U      #bobina_T4;
U      #bobina_T24;
S      A      4.1;
R      T      4;
R      T      24;
R      A      4.0;
```

```
NETWORK
TITLE =
```

```

U      A      4.1;
R      "B7";
=      M      6.1;

```

```

NETWORK
TITLE =

```

```

U      M      6.1;
FR     T      5;
L      DB6.DBW 58; //proceso
SE     T      5;
U      T      5;
=      #bobina_T5;

```

```

NETWORK
TITLE =

```

```

U      #bobina_T5;
U      M      6.1;
FR     T      25;
L      DB6.DBW 94; //proceso ms
SE     T      25;
U      T      25;
=      #bobina_T25;

```

```

NETWORK
TITLE =

```

```

U      #bobina_T5;
U      #bobina_T25;
S      A      4.2;
R      T      5;
R      T      25;
R      A      4.1;

```

```

NETWORK
TITLE =

```

```

U      A      4.2;
=      M      6.2;

```

```

NETWORK
TITLE =

```

```

U      M      6.2;
FR     T      6;
L      DB6.DBW 66; //espera
SE     T      6;
U      T      6;
=      #bobina_T6;

```

```

NETWORK
TITLE =

```

```

U      #bobina_T6;
U      M      6.2;
FR     T      26;
L      DB6.DBW 98; //espera ms
SE     T      26;
U      T      26;
=      #bobina_T26;

```

```

NETWORK
TITLE =

```

```

U      #bobina_T6;

```

```

U      #bobina_T26;
S      A      4.3;
R      T      6;
R      T      26;
R      A      4.2;
NETWORK
TITLE =

U      A      4.3;
=      M      6.3;

NETWORK
TITLE =

U      M      6.3;
FR     T      7;
L      DB6.DBW 78; //vaciado
SE     T      7;
U      T      7;
=      #bobina_T7;

NETWORK
TITLE =

U      #bobina_T7;
U      M      6.3;
FR     T      27;
L      DB6.DBW 102; //vaciado
SE     T      27;
U      T      27;
=      #bobina_T27;

NETWORK
TITLE =Evaporación L1

UN     #bobina_T7;
U      M      6.3;
=      "EV7";
NETWORK
TITLE =Evaporación L1

UN     #bobina_T27;
U      M      6.3;
=      "EV7";
NETWORK
TITLE =

U      #bobina_T7;
U      #bobina_T27;
S      A      4.0;
R      T      7;
R      T      27;
R      A      4.3;
END_FUNCTION

FUNCTION "L2_evaporación" : VOID
TITLE =
VERSION : 0.1

VAR_TEMP
bobina_T14 : BOOL ;
bobina_T15 : BOOL ;
bobina_T16 : BOOL ;
bobina_T17 : BOOL ;
bobina_T34 : BOOL ;
bobina_T35 : BOOL ;
bobina_T36 : BOOL ;

```

```

bobina_T37 : BOOL ;
END_VAR
BEGIN
NETWORK
TITLE =Evaporación L2

```

```

U      A      5.0;
R      "EV8";
=      M      7.0;

```

```

NETWORK
TITLE =

```

```

U      M      7.0;
FR     T      14;
L      DB6.DBW 54;
SE     T      14;
U      T      14;
=      #bobina_T14;

```

```

NETWORK
TITLE =

```

```

U      #bobina_T14;
U      M      7.0;
FR     T      34;
L      DB6.DBW 106;
SE     T      34;
U      T      34;
=      #bobina_T34;

```

```

NETWORK
TITLE =

```

```

UN     #bobina_T14;
U      M      7.0;
=      "B8";

```

```

NETWORK
TITLE =

```

```

UN     #bobina_T34;
U      M      7.0;
=      "B8";

```

```

NETWORK
TITLE =

```

```

U      #bobina_T14;
U      #bobina_T34;
S      A      5.1;
R      T      14;
R      T      34;
R      A      5.0;

```

```

NETWORK
TITLE =

```

```

U      A      5.1;
R      "B8";
=      M      7.1;

```

```

NETWORK
TITLE =

```

```

U      M      7.1;
FR     T      15;
L      DB6.DBW 62;
SE     T      15;

```

```

U      T      15;
=      #bobina_T15;

```

```

NETWORK
TITLE =

```

```

U      #bobina_T15;
U      M      7.1;
FR     T      35;
L      DB6.DBW 110;
SE     T      35;
U      T      35;
=      #bobina_T35;

```

```

NETWORK
TITLE =

```

```

U      #bobina_T15;
U      #bobina_T35;
S      A      5.2;
R      T      15;
R      T      35;
R      A      5.1;

```

```

NETWORK
TITLE =

```

```

U      A      5.2;
=      M      7.2;

```

```

NETWORK
TITLE =

```

```

U      M      7.2;
FR     T      16;
L      DB6.DBW 70;
SE     T      16;
U      T      16;
=      #bobina_T16;

```

```

NETWORK
TITLE =

```

```

U      #bobina_T16;
U      M      7.2;
FR     T      36;
L      DB6.DBW 114;
SE     T      36;
U      T      36;
=      #bobina_T36;

```

```

NETWORK
TITLE =

```

```

U      #bobina_T16;
U      #bobina_T36;
S      A      5.3;
R      T      16;
R      T      36;
R      A      5.2;

```

```

NETWORK
TITLE =

```

```

U      A      5.3;
=      M      7.3;

```

```
NETWORK
TITLE =
```

```
U      M      7.3;
FR     T      17;
L      DB6.DBW  86;
SE     T      17;
U      T      17;
=      #bobina_T17;
```

```
NETWORK
TITLE =
```

```
U      #bobina_T17;
U      M      7.3;
FR     T      37;
L      DB6.DBW 118;
SE     T      37;
U      T      37;
=      #bobina_T37;
```

```
NETWORK
TITLE =Evaporación L1
```

```
UN     #bobina_T17;
U      M      7.3;
=      "EV8";
```

```
NETWORK
TITLE =Evaporación L2
```

```
UN     #bobina_T37;
U      M      7.3;
=      "EV8";
```

```
NETWORK
TITLE =
```

```
U      #bobina_T17;
U      #bobina_T37;
S      A      5.0;
R      T      17;
R      T      37;
R      A      5.3;
END_FUNCTION
```

```
FUNCTION "Conversion Tiempos Timer" : VOID
TITLE =
VERSION : 0.1
```

```
VAR_INPUT
NREAL : REAL ;
END_VAR
VAR_OUTPUT
Segundos : REAL ;
MSegundos : REAL ;
END_VAR
VAR_TEMP
pentera : DWORD ;
END_VAR
BEGIN
NETWORK
TITLE =

L      #NREAL;
RND-   ;
T      #pentera;
```

```

DTB      ;
L        DW#16#FFF;
UD       ;
L        DW#16#2000;
OD       ;
T        #Segundos;
NETWORK
TITLE =

L        #pentera;
DTR      ;
L        #NREAL;
TAK      ;
-R       ;
L        1.000000e+002;
*R       ;
RND-     ;
DTB      ;
T        #MSegundos;

END_FUNCTION

FUNCTION "Manual" : VOID
TITLE =
VERSION : 0.1

BEGIN
NETWORK
TITLE =

U        M        20.1;
=        "B7";

NETWORK
TITLE =

U        M        20.3;
=        "B8";

NETWORK
TITLE =Evaporación L1

U        M        20.2;
=        "EV7";

NETWORK
TITLE =Evaporación L1

U        M        20.4;
=        "EV8";

NETWORK
TITLE =

CALL "L1_mezcla" ;

NETWORK
TITLE =

CALL "L2_mezcla" ;

NETWORK
TITLE =

U        "L1 Embotellado";
U        "M/A";
CC       "Embotellado L1";
NETWORK
TITLE =

U        "L2 Embotellado";

```

```

U      "M/A";
CC     "Embotellado L2";
END_FUNCTION

FUNCTION "inicializacion parametro" : VOID
TITLE =
VERSION : 0.1

BEGIN
NETWORK
TITLE =

//inicialización valores de referencia
L      1.000000e+001;
T      "datos de referencia".altura_minima_emb1;
L      1.100000e+002;
T      "datos de referencia".altura_maxima_emb1;
L      2.000000e+000;
T      "datos de referencia".alt_min_sensor;
L      1.020000e+002;
T      "datos de referencia".alt_max_sensor;
L      1.340000e+000;
T      "datos de referencia".kv_1;
L      1.600000e-001;
T      "datos de referencia".volumen_bot;
L      2.500000e-001;
T      "datos de referencia".altura_ref_PID1;

END_FUNCTION

FUNCTION "Paro General" : VOID
TITLE =
VERSION : 0.1

BEGIN
NETWORK
TITLE =
//Reseteamos todas las etapas
R      "L1 Embotellado";
R      "L2 Embotellado";
R      "L1 Mezcla";
R      "L2 Mezcla";
R      "L1 Evaporación";
R      "L2 Evaporación";
R      E      2.0;
R      E      2.1;
CALL "Paro Mezcla L1" ;
CALL "Paro Mezcla L2" ;
END_FUNCTION

```

BLOQUES DE ORGANIZACIÓN

```

ORGANIZATION_BLOCK "Cycle Execution"
TITLE = "Main Program Sweep (Cycle)"
VERSION : 0.1

VAR_TEMP
OB1_EV_CLASS : BYTE ; //Bits 0-3 = 1 (Coming event), Bits 4-7 = 1 (Event class 1)
OB1_SCAN_1 : BYTE ; //1 (Cold restart scan 1 of OB 1), 3 (Scan 2-n of OB 1)
OB1_PRIORITY : BYTE ; //1 (Priority of 1 is lowest)
OB1_OB_NUMBR : BYTE ; //1 (Organization block 1, OB1)
OB1_RESERVED_1 : BYTE ; //Reserved for system
OB1_RESERVED_2 : BYTE ; //Reserved for system
OB1_PREV_CYCLE : INT ; //Cycle time of previous OB1 scan (milliseconds)
OB1_MIN_CYCLE : INT ; //Minimum cycle time of OB1 (milliseconds)
OB1_MAX_CYCLE : INT ; //Maximum cycle time of OB1 (milliseconds)
OB1_DATE_TIME : DATE_AND_TIME ; //Date and time OB1 started
END_VAR
BEGIN
NETWORK

```

```

TITLE =
//
//¿Manual o Automático?
U      "M/A";
CC      "Manual";
NETWORK
TITLE =
//Referente a etapa de embotellado linea 1

U      "L1 Embotellado";
UN      "M/A";
CC      "Embotellado L1";

NETWORK
TITLE =
//embotellado solo simulación
UN      M      1.1;
UN      M      1.2;
=      M      1.0;
NETWORK
TITLE =
//embotellado linea 2
U      "L2 Embotellado";
UN      "M/A";
CC      "Embotellado L2";
NETWORK
TITLE =

NETWORK
TITLE =
//mezcla linea 1
U      "L1 Mezcla";
UN      "M/A";
CC      "L1_mezcla";
NETWORK
TITLE =

UN      "L1 Mezcla";
UN      "M/A";
CC      "Paro Mezcla L1";
NETWORK
TITLE =
//mezcla linea 2
U      "L2 Mezcla";
UN      "M/A";
CC      "L2_mezcla";
NETWORK
TITLE =

UN      "L2 Mezcla";
UN      "M/A";
CC      "Paro Mezcla L2";
NETWORK
TITLE =

U      "L1 Evaporación";
UN      A      4.1;
UN      A      4.2;
UN      A      4.3;
=      A      4.0;
NETWORK
TITLE =
//Evaporación 1
U      "L1 Evaporación";
UN      "M/A";
CC      "L1_evaporación";

NETWORK
TITLE =

U      "L2 Evaporación";
UN      A      5.1;

```

```

UN      A      5.2;
UN      A      5.3;
=       A      5.0;
NETWORK
TITLE =
//Evaporación 2
U       "L2 Evaporación";
UN      "M/A";
CC      "L2_evaporación";

NETWORK
TITLE =

U       "Paro Total";
CC      "Paro General";

END_ORGANIZATION_BLOCK

ORGANIZATION_BLOCK "PID_block"
TITLE = "Cyclic Interrupt"
VERSION : 0.1

VAR TEMP
OB35_EV_CLASS : BYTE ;      //Bits 0-3 = 1 (Coming event), Bits 4-7 = 1 (Event class
1)
OB35_STRT_INF : BYTE ;      //16#36 (OB 35 has started)
OB35_PRIORITY : BYTE ;      //Priority of OB Execution
OB35_OB_NUMBR : BYTE ;      //35 (Organization block 35, OB35)
OB35_RESERVED_1 : BYTE ;    //Reserved for system
OB35_RESERVED_2 : BYTE ;    //Reserved for system
OB35_PHASE_OFFSET : WORD ;   //Phase offset (msec)
OB35_RESERVED_3 : INT ;      //Reserved for system
OB35_EXC_FREQ : INT ;        //Frequency of execution (msec)
OB35_DATE_TIME : DATE_AND_TIME ; //Date and time OB35 started
END_VAR
BEGIN
NETWORK
TITLE =

U       "L1 Mezcla";
UN      "M/A";
CC      "L1_PID";

NETWORK
TITLE =

U       "L2 Mezcla";
UN      "M/A";
CC      "L2_PID";

NETWORK
TITLE =
//Tanque embotellado L1
L       "Altura Tanque Emb L1";
T       "datos_periferia".altura_1_perif;
NETWORK
TITLE =

CALL "conversion word a real" (
CANAL           := "datos_periferia".altura_1_perif,
LIM_SUP         := 1.070000e+000,
LIM_INF         := 0.000000e+000,
SALIDA          := "datos_reales_calculados".altura_1_real);

NETWORK
TITLE =
//Tanque embotellado L2
L       "Altura Tanque Emb L2";
T       "datos_periferia".altura_2_perif;
NETWORK

```

```

TITLE =

CALL "conversion word a real" (
CANAL                := "datos periferia".altura_2_perif,
LIM_SUP              := 1.070000e+000,
LIM_INF              := 0.000000e+000,
SALIDA               := "datos reales calculados".altura_2_real);

NETWORK
TITLE =
//Tanque mezcla L1
L      "Altura Tanque Mez L1";
T      "datos periferia".altura_m1;
NETWORK
TITLE =

CALL "conversion word a real" (
CANAL                := "datos periferia".altura_m1,
LIM_SUP              := 2.700000e+001,
LIM_INF              := 0.000000e+000,
SALIDA               := "datos reales calculados".altura_M1_real);

NETWORK
TITLE =
//Tanque Mezcla L2
L      "Altura Tanque Mez L2";
T      "datos periferia".altura_m2;
NETWORK
TITLE =

CALL "conversion word a real" (
CANAL                := "datos periferia".altura_m2,
LIM_SUP              := 2.700000e+001,
LIM_INF              := 1.000000e+000,
SALIDA               := "datos reales calculados".altura_M2_real);

NETWORK
TITLE =
//Tanque Buffer L1
L      "Altura Buffer L1";
T      "datos periferia".altura_b1;
NETWORK
TITLE =

CALL "conversion word a real" (
CANAL                := "datos periferia".altura_b1,
LIM_SUP              := 2.600000e+001,
LIM_INF              := 0.000000e+000,
SALIDA               := "datos reales calculados".altura_b1_real);

NETWORK
TITLE =
//Tanque Buffer L2
L      "Altura Buffer L2";
T      "datos periferia".altura_b2;
NETWORK
TITLE =

CALL "conversion word a real" (
CANAL                := "datos periferia".altura_b2,
LIM_SUP              := 2.600000e+001,
LIM_INF              := 0.000000e+000,
SALIDA               := "datos reales calculados".altura_b2_real);

NETWORK
TITLE =

CALL "Conversion Tiempos Timer" (
NREAL                := "Produccion".T_Llenado_EV1,
Segundos              := "Produccion".TLEV1,
MSegundos             := "Produccion".tlev_1);

```

```
NETWORK
TITLE =
```

```
CALL "Conversion Tiempos Timer" (
NREAL          := "Produccion".T_Llenado_EV2,
Segundos       := "Produccion".TLEV2,
MSegundos      := "Produccion".tleV2);
```

```
NETWORK
TITLE =
```

```
CALL "Conversion Tiempos Timer" (
NREAL          := "Produccion".T_Evap_Proceso_1,
Segundos       := "Produccion".TPEV1,
MSegundos      := "Produccion".tpeV1);
```

```
NETWORK
TITLE =
```

```
CALL "Conversion Tiempos Timer" (
NREAL          := "Produccion".T_Evap_Proceso_2,
Segundos       := "Produccion".TPEV2,
MSegundos      := "Produccion".tpeV2);
```

```
NETWORK
TITLE =
```

```
CALL "Conversion Tiempos Timer" (
NREAL          := "Produccion".T_Espera1,
Segundos       := "Produccion".TEEV1,
MSegundos      := "Produccion".teev1);
```

```
NETWORK
TITLE =
```

```
CALL "Conversion Tiempos Timer" (
NREAL          := "Produccion".T_Espera2,
Segundos       := "Produccion".TEEV2,
MSegundos      := "Produccion".teev2);
```

```
NETWORK
TITLE =
```

```
CALL "Conversion Tiempos Timer" (
NREAL          := "Produccion".T_Descarga_1,
Segundos       := "Produccion".TDEV1,
MSegundos      := "Produccion".tdev1);
```

```
NETWORK
TITLE =
```

```
CALL "Conversion Tiempos Timer" (
NREAL          := "Produccion".T_Descarga_2,
Segundos       := "Produccion".TDEV2,
MSegundos      := "Produccion".tdev2);
```

```
NETWORK
TITLE =
```

```
//Se procede a enviar la salida del PID a la periferia mediante profibus. Como se
//ha configurado el hardware para trabajar con datos no consistentes (PPO type 3
//word consistent), se ha mapeado una dirección de memoria para enviar los datos.
//En PAW256 se introducirá el código de control, y en PAW 258 la referencia
//encontrada mediante PID
L      W#16#47C;
T      "Cod Control B3";
```

```

NETWORK
TITLE =

L      "Manual DB".Ref_B3_Manual;
T      "Referencia Variador B3";
NETWORK
TITLE =

L      "datos paro".bomba1;
T      "Referencia Variador B1";
NETWORK
TITLE =
//Se procede a enviar la salida del PID a la periferia mediante profibus. Como se
//ha configurado el hardware para trabajar con datos no consistentes (PPO type 3
//word consistent), se ha mapeado una dirección de memoria para enviar los datos.
//En PAW256 se introducirá el código de control, y en PAW 258 la referencia
//encontrada mediante PID
L      W#16#47C;
T      "Cod Control B4";
NETWORK
TITLE =

L      "Manual DB".Ref_B4_Manual;
T      "Referencia Variador B4";
NETWORK
TITLE =

L      "datos paro".bomba2;
T      "Referencia Variador B2";
NETWORK
TITLE =

U      E      0.1;
=      "B9";
END_ORGANIZATION_BLOCK

```

A3. CÓDIGO FUENTE DE LOS SCRIPTS ASOCIADOS A LA APLICACIÓN DE SUPERVISIÓN

Seguidamente se expone el código fuente asociado a los scripts programados en la aplicación de supervisión realizada en InTouch. Distinguiremos dos tipos de Scripts en función de cuándo se ejecutan:

- **Scripts de Aplicación:** Son aquellos que se ejecutan en función del evento que se produzca en la aplicación. Existen tres tipos:
 - **On Startup:** se ejecutan al iniciar la aplicación. Se utilizarán, por ejemplo, para mostrar las ventanas que se visualizarán por defecto al iniciar.
 - **While running:** se ejecutan continuamente durante el transcurso de la aplicación. Se utilizarán para hacer conversiones y realizar comprobaciones en tiempo de ejecución.
 - **On Shutdown:** se ejecutan al finalizar la aplicación y se utilizarán para poner los contadores de las botellas producidas a 0.
- **Scripts de ejecución en los cambios de datos (Data Change):** Son aquellos que se ejecutan una vez cuando cambia el valor de la variable de que dependen.

APPLICATION SCRIPT

On Startup

```
Show "Barra_nav";
ActiveWindow = "General";

IOReinitialize( );
IOStartUninitConversations( );

rT_Descarga_EV1_PLC1 = 20;
rRatio1_PLC1 = 1;
rRatio2_PLC1 = 1;
rT__Evap_Proceso1_PLC1 = 5;
Vol_botella1 = 150;
Vol_Evap1 = 1000;

rT_Descarga_EV2_PLC1 = 20;
rT__Evap_Proceso2_PLC1 = 5;
Vol_botella2 = 150;
Vol_Evap2 = 1000;

alt_embot__L1 = 10;
alt_embot__L2 = 10;

X = 0;
Y = 0;
```

While Running

```
rT_Espera_EV1_PLC1 = ((60 * (Vol_Evap1/1000)) / (rCau_Salida_M1_PLC1 / (1 + rRatio1_PLC1))) -
(rT__Llenado_EV1_PLC1 + rT__Evap_Proceso1_PLC1 + rT_Descarga_EV1_PLC1);
rT_Espera_EV2_PLC1 = ((60 * (Vol_Evap2/1000)) / (rCau_Salida_M2_PLC1 / (1 + rRatio2_PLC1))) -
(rT__Llenado_EV2_PLC1 + rT__Evap_Proceso2_PLC1 + rT_Descarga_EV2_PLC1);

IF rT_Espera_EV1_PLC1 < 0 THEN
    rT_Espera_EV1_PLC1 = 0.1;
ENDIF;

IF rT_Espera_EV2_PLC1 < 0 THEN
    rT_Espera_EV2_PLC1 = 0.1;
ENDIF;

Vol_botella1= rVol__Bot1_PLC1 * 1000;
Vol_botella2= rVol__Bot2_PLC1 * 1000;

Vol_Evap1 = (rT__Llenado_EV1_PLC1 * 928) / 6;
Vol_Evap2 = (rT__Llenado_EV2_PLC1 * 984) / 6;

IF F_pump1 == 1 OR F_pump2 == 1 OR F_pump3 == 1 OR F_pump4 == 1 THEN
    Fallo = 1;
ELSE
    Fallo = 0;
ENDIF;

IF Fallo == 1 THEN
    Show "Fallo rellenar";
    F_pump1 = 0;
```

```

F_pump2 = 0;
F_pump3 = 0;
F_pump4 = 0;
ENDIF;

IF Pump1==1 AND NOT dPA_PM_PL1 THEN
  IF rCau_Salida_M1_PL1<=0 OR rRatio1_PL1 <1 OR rVol__Bot1_PL1 <= 0 OR Vol_Evap1<=0 OR
  rT__Evap_Proceso1_PL1<=0 OR rT_Descarga_EV1_PL1<=0 THEN
    F_pump1 =1;
    Pump1=0;
  ELSE F_pump1 = 0;
  ENDIF;
  IF Celda_Llenado1_ON == 1 AND dPA_PM_PL1 == 0 THEN
    iBomba1_PL1 = iBomba1;
  ENDIF;
ENDIF;

IF Pump2==1 AND NOT dPA_PM_PL1 THEN
  IF rCau_Salida_M2_PL1<=0 OR rRatio2_PL1 <1 OR rVol__Bot2_PL1 <= 0 OR Vol_Evap1<=0 OR
  rT__Evap_Proceso1_PL1<=0 OR rT_Descarga_EV1_PL1<=0 THEN
    F_pump2 =1;
    Pump2=0;
  ELSE F_pump2 = 0;
  ENDIF;
  IF Celda_Llenado2_ON == 1 AND dPA_PM_PL1 == 0 THEN
    iBomba2_PL1 = iBomba2;
  ENDIF;
ENDIF;

IF Pump3==1 AND NOT dPA_PM_PL1 THEN
  IF rCau_Salida_M2_PL1<=0 OR rRatio2_PL1 <1 OR rVol__Bot2_PL1 <= 0 OR Vol_Evap2<=0 OR
  rT__Evap_Proceso2_PL1<=0 OR rT_Descarga_EV2_PL1<=0 THEN
    F_pump3 =1;
    Pump3=0;
  ELSE F_pump3 = 0;
  ENDIF;
  IF Celda_Llenado2_ON == 1 AND dPA_PM_PL1 == 0 THEN
    iBomba2_PL1 = iBomba2;
  ENDIF;
ENDIF;

IF Pump4==1 AND NOT dPA_PM_PL1 THEN
  IF rCau_Salida_M1_PL1<=0 OR rRatio1_PL1 <1 OR rVol__Bot1_PL1 <= 0 OR Vol_Evap2<=0 OR
  rT__Evap_Proceso2_PL1<=0 OR rT_Descarga_EV2_PL1<=0 THEN
    F_pump4 =1;
    Pump4=0;
  ELSE F_pump4 = 0;
  ENDIF;
  IF Celda_Llenado1_ON == 1 AND dPA_PM_PL1 == 0 THEN
    iBomba1_PL1 = iBomba1;
  ENDIF;
ENDIF;

rAlt__Minima_Emb1_PL1 = alt_embot__L1 / 100;
rAlt__Minima_Emb2_PL1 = alt_embot__L2 / 100;
alt_embot__L1 = rAlt__Minima_Emb1_PL1 * 100;
alt_embot__L2 = rAlt__Minima_Emb2_PL1 * 100;

```

```
IF dPID1_PLC1 THEN
  rB3_accion_real_PLC2 = rB3_Ref_Man_PLC2;
ENDIF;
```

```
IF dPID2_PLC1 THEN
  rB4_accion_real_PLC2 = rB4_Ref_Man_PLC2;
ENDIF;
```

```
IF dPA_PM_PLC1 == 0 THEN
```

```
  IF Pump1 == 1 AND F_pump1 == 0 THEN
```

```
    dL1_Embotellado_PLC1=1;
    dL1_Evaporación_PLC1=1;
    dL1_Mezcla_PLC1=1;
    Valve_3 = 1;
    Celda_Llenado1_ON = 1;
    Celda_Evap1_ON = 1;
    iBomba1_PLC1 = iBomba1;
```

```
  ENDIF;
```

```
  IF Pump2 == 1 THEN
```

```
    dEmbotellado_L2_PLC1=1;
    dL1_Evaporación_PLC1=1;
    dMezcla_L2_PLC1=1;
    Valve5 = 1;
    Celda_Llenado2_ON = 1;
    Celda_Evap1_ON = 1;
    iBomba2_PLC1 = iBomba2;
```

```
  ENDIF;
```

```
  IF Pump3 == 1 THEN
```

```
    dEmbotellado_L2_PLC1=1;
    dEvaporación_L2_PLC1=1;
    dMezcla_L2_PLC1=1;
    Valve_6 = 1;
    Celda_Llenado2_ON =1;
    Celda_Evap2_ON=1;
    iBomba2_PLC1 = iBomba2;
```

```
  ENDIF;
```

```
  IF Pump4 == 1 THEN
```

```
    dL1_Embotellado_PLC1=1;
    dEvaporación_L2_PLC1=1;
    dL1_Mezcla_PLC1=1;
    Valve_4 = 1;
    Celda_Llenado1_ON =1;
    Celda_Evap2_ON=1;
    iBomba1_PLC1 = iBomba1;
```

```
  ENDIF;
```

```
ENDIF;
```

On shutdown

```
X = 0;
```

```
Y = 0;
```

DATA CHANGE**Celda_evap1_ON**

```
IF Valve3 == 1 AND Celda_Evap1_ON == 1 AND Celda_Llenado1_ON THEN
```

```
    Pump1 = 1;
```

```
ELSE
```

```
    Pump1 = 0;
```

```
ENDIF;
```

```
IF Valve_5 == 1 AND Celda_Evap1_ON == 1 AND Celda_Llenado2_ON == 1 THEN
```

```
    Pump2 = 1;
```

```
ELSE
```

```
    Pump2 = 0;
```

```
ENDIF;
```

```
IF dPA_PM_PL1 == 1 AND Celda_Llenado1_ON == 1 THEN
```

```
    IF ( NOT Celda_Evap1_ON AND NOT Celda_Evap2_ON ) THEN
```

```
        dL1_Embotellado_PL1 = 1;
```

```
    ELSE
```

```
        dL1_Embotellado_PL1 = 0;
```

```
        dNoria_1_PL2 = 0;
```

```
        Valve_1 = 0;
```

```
    ENDIF;
```

```
ELSE
```

```
    dL1_Embotellado_PL1 = 0;
```

```
    dNoria_1_PL2 = 0;
```

```
    Valve_1 = 0;
```

```
ENDIF;
```

```
IF dPA_PM_PL1 == 1 AND Celda_Llenado2_ON == 1 THEN
```

```
    IF ( NOT Celda_Evap1_ON AND NOT Celda_Evap2_ON ) THEN
```

```
        dEmbotellado_L2_PL1 = 1;
```

```
    ELSE
```

```
        dEmbotellado_L2_PL1 = 0;
```

```
        dNoria_2_PL2 = 0;
```

```
        Valve_2 = 0;
```

```
    ENDIF;
```

```
ELSE
```

```
    dEmbotellado_L2_PL1 = 0;
```

```
    dNoria_2_PL2 = 0;
```

```
    Valve_2 = 0;
```

```
ENDIF;
```

Celda_Evap2_ON

```
IF Valve6 == 1 AND Celda_Evap2_ON == 1 AND Celda_Llenado2_ON THEN
```

```
    Pump3 = 1;
```

```
ELSE
```

```
    Pump3 = 0;
```

```
ENDIF;
```

```
IF Valve4 == 1 AND Celda_Evap2_ON == 1 AND Celda_Llenado1_ON == 1 THEN
```

```
    Pump4 = 1;
```

```
ELSE
```

```
    Pump4 = 0;
```

```
ENDIF;
```

```

IF dPA_PM_PL1 == 1 AND Celda_Llenado1_ON == 1 THEN
  IF ( NOT Celda_Evap1_ON AND NOT Celda_Evap2_ON ) THEN
    dL1_Embotellado_PL1 = 1;
  ELSE
    dL1_Embotellado_PL1 = 0;
    dNoria_1_PL2 = 0;
    Valve_1 = 0;
  ENDIF;
ELSE
  dL1_Embotellado_PL1 = 0;
  dNoria_1_PL2 = 0;
  Valve_1 = 0;
ENDIF;

```

```

IF dPA_PM_PL1 == 1 AND Celda_Llenado2_ON == 1 THEN
  IF ( NOT Celda_Evap1_ON AND NOT Celda_Evap2_ON ) THEN
    dEmbotellado_L2_PL1 = 1;
  ELSE
    dEmbotellado_L2_PL1 = 0;
    dNoria_2_PL2 = 0;
    Valve_2 = 0;
  ENDIF;
ELSE
  dEmbotellado_L2_PL1 = 0;
  dNoria_2_PL2 = 0;
  Valve_2 = 0;
ENDIF;

```

Celda_Llenado1_ON

```

IF Valve3 == 1 AND Celda_Evap1_ON == 1 AND Celda_Llenado1_ON THEN
  Pump1 = 1;
ELSE
  Pump1 = 0;
ENDIF;

```

```

IF Valve4 == 1 AND Celda_Evap2_ON == 1 AND Celda_Llenado1_ON THEN
  Pump4 = 1;
ELSE
  Pump4 = 0;
ENDIF;

```

```

IF dPA_PM_PL1 == 1 AND Celda_Llenado1_ON == 1 THEN
  IF ( NOT Celda_Evap1_ON AND NOT Celda_Evap2_ON ) THEN
    dL1_Embotellado_PL1 = 1;
  ELSE
    dL1_Embotellado_PL1 = 0;
    dNoria_1_PL2 = 0;
    Valve_1 = 0;
  ENDIF;
ELSE
  dL1_Embotellado_PL1 = 0;
  dNoria_1_PL2 = 0;
  Valve_1 = 0;
ENDIF;

```

Celda_Llenado2_ON

```

IF Valve_5 == 1 AND Celda_Evap1_ON == 1 AND Celda_Llenado2_ON == 1 THEN
    Pump2 = 1;
ELSE
    Pump2 = 0;
ENDIF;

```

```

IF Valve6 == 1 AND Celda_Evap2_ON == 1 AND Celda_Llenado2_ON == 1 THEN
    Pump3 = 1;
ELSE
    Pump3 = 0;
ENDIF;

```

```

IF dPA_PM_PL1 == 1 AND Celda_Llenado2_ON == 1 THEN
    IF ( NOT Celda_Evap1_ON AND NOT Celda_Evap2_ON ) THEN
        dEmbotellado_L2_PL1 = 1;
    ELSE
        dEmbotellado_L2_PL1 = 0;
        dNoria_2_PL2 = 0;
        Valve_2 = 0;
    ENDIF;
ELSE
    dEmbotellado_L2_PL1 = 0;
    dNoria_2_PL2 = 0;
    Valve_2 = 0;
ENDIF;

```

dBot1_lista_PL2

```

IF NOT dBot1_lista_PL2 AND dL1_Embotellado_PL1 THEN
    X = X + 1;
ENDIF;

```

dBot2_lista_PL2

```

IF NOT dBot2_lista_PL2 AND dEmbotellado_L2_PL1 THEN
    Y = Y + 1;
ENDIF;

```

dPA_PM_PL1

```

IF dPA_PM_PL1 == 1 THEN
    iB3_ref_swich_PL2 = 0;
    iBomba1_PL1 = 0;
    iB4_ref_swich_PL2 = 0;
    iBomba2_PL1 = 0;
    dMezcla_L2_PL1 = 0;
    dEvaporación_L2_PL1 = 0;
    dL1_Evaporación_PL1 = 0;
    dL1_Mezcla_PL1 = 0;
ENDIF;

```

```

IF dPA_PM_PL1 == 0 THEN
    Valve_1 = 0;
    dNoria_1_PL2 = 0;
    Valve_2 = 0;

```

```

    dNoria_2_PLC2 = 0;
ENDIF;

IF dPA_PM_PLC1 == 1 AND Celda_Llenado1_ON == 1 THEN
    IF ( NOT Celda_Evap1_ON AND NOT Celda_Evap2_ON ) THEN
        dL1_Embotellado_PLC1 = 1;
    ELSE
        dL1_Embotellado_PLC1 = 0;
        dNoria_1_PLC2 = 0;
        Valve_1 = 0;
    ENDIF;
ELSE
    dL1_Embotellado_PLC1 = 0;
    dNoria_1_PLC2 = 0;
    Valve_1 = 0;
ENDIF;

IF dPA_PM_PLC1 == 1 AND Celda_Llenado2_ON == 1 THEN
    IF ( NOT Celda_Evap1_ON AND NOT Celda_Evap2_ON ) THEN
        dEmbotellado_L2_PLC1 = 1;
    ELSE
        dEmbotellado_L2_PLC1 = 0;
        dNoria_2_PLC2 = 0;
        Valve_2 = 0;
    ENDIF;
ELSE
    dEmbotellado_L2_PLC1 = 0;
    dNoria_2_PLC2 = 0;
    Valve_2 = 0;
ENDIF;

```

iBomba1

```

rCau_Salida_M1_PLC1 = (iBomba1* 0.0653) - 0.0496;

```

iBomba2

```

rCau_Salida_M2_PLC1 = (iBomba2 * 0.0641) - 0.0214;

```

Pump1

```

IF Pump1 == 1 AND F_pump1 == 0 AND dPA_PM_PLC1 == 0 THEN
    dL1_Embotellado_PLC1=1;
    dL1_Evaporación_PLC1=1;
    dL1_Mezcla_PLC1=1;
    Valve_3 = 1;
    Celda_Llenado1_ON = 1;
    Celda_Evap1_ON = 1;
    IF dPA_PM_PLC1 == 1 THEN
        iBomba1_PLC1 = 0;
    ELSE
        iBomba1_PLC1 = iBomba1;
    ENDIF;
ENDIF;

IF Pump1 == 0 AND dPA_PM_PLC1 == 0 THEN
    Valve_3 = 0;

```

```

IF Pump2 == 1 AND Pump4 == 0 THEN
    dL1_Embotellado_PLC1= 0;
    dL1_Mezcla_PLC1=0;
    Celda_Llenado1_ON = 0;
    iBomba1_PLC1 = 0;
    iB3_ref_swich_PLC2 = 0;
ENDIF;
IF Pump4 == 1 AND Pump2 == 0 THEN
    Celda_Evap1_ON = 0;
    dL1_Evaporación_PLC1 = 0;
ENDIF;
IF Pump2==0 AND Pump4==0 THEN
    dL1_Embotellado_PLC1= 0;
    dL1_Mezcla_PLC1=0;
    Celda_Llenado1_ON = 0;
    Celda_Evap1_ON = 0;
    dL1_Evaporación_PLC1 = 0;
    iBomba1_PLC1 = 0;
    iB3_ref_swich_PLC2 = 0;
ENDIF;
ENDIF;

```

Pump2

```

IF Pump2 == 1 AND F_pump2 == 0 AND NOT dPA_PM_PLC1 THEN
    dEmbotellado_L2_PLC1=1;
    dL1_Evaporación_PLC1=1;
    dMezcla_L2_PLC1=1;
    Valve5 = 1;
    Celda_Llenado2_ON = 1;
    Celda_Evap1_ON = 1;
    IF dPA_PM_PLC1 == 1 THEN
        iBomba2_PLC1 = 0;
    ELSE
        iBomba2_PLC1 = iBomba1;
    ENDIF;
ENDIF;

IF Pump2 == 0 AND NOT dPA_PM_PLC1 THEN
    Valve5 = 0;
    IF Pump1 == 1 AND Pump3 == 0 THEN
        dEmbotellado_L2_PLC1= 0;
        dMezcla_L2_PLC1=0;
        Celda_Llenado2_ON = 0;
        iBomba2_PLC1 = 0;
        iB4_ref_swich_PLC2 = 0;
    ENDIF;
    IF Pump3 == 1 AND Pump1 == 0 THEN
        Celda_Evap1_ON = 0;
        dL1_Evaporación_PLC1 = 0;
    ENDIF;
    IF Pump1==0 AND Pump3==0 THEN
        dEmbotellado_L2_PLC1= 0;
        dMezcla_L2_PLC1=0;
        Celda_Llenado2_ON = 0;
        Celda_Evap1_ON = 0;
        dL1_Evaporación_PLC1 = 0;
    ENDIF;

```

```

        iBomba2_PLC1 = 0;
        iB4_ref_swich_PLC2 = 0;
    ENDIF;
ENDIF;

```

Pump3

```

IF Pump3 == 1 AND F_pump3 == 0 AND NOT dPA_PM_PLC1 THEN
    dEmbotellado_L2_PLC1=1;
    dEvaporación_L2_PLC1=1;
    dMezcla_L2_PLC1=1;
    Valve_6 = 1;
    Celda_Llenado2_ON =1;
    Celda_Evap2_ON=1;
    IF dPA_PM_PLC1 == 1 THEN
        iBomba2_PLC1 = 0;
    ELSE
        iBomba2_PLC1 = iBomba1;
    ENDIF;
ENDIF;

```

```

IF Pump3 == 0 AND NOT dPA_PM_PLC1 THEN
    Valve_6 = 0;
    IF Pump2 == 1 AND Pump4 == 0 THEN
        dEvaporación_L2_PLC1=0;
        Celda_Evap2_ON=0;
    ENDIF;
    IF Pump4 == 1 AND Pump2 == 0 THEN
        Celda_Llenado2_ON =0;
        dEmbotellado_L2_PLC1=0;
        dMezcla_L2_PLC1=0;
        iBomba2_PLC1 = 0;
        iB4_ref_swich_PLC2 = 0;
    ENDIF;
    IF Pump2==0 AND Pump4==0 THEN
        Celda_Llenado2_ON =0;
        dEmbotellado_L2_PLC1=0;
        dMezcla_L2_PLC1=0;
        dEvaporación_L2_PLC1=0;
        Celda_Evap2_ON=0;
        iBomba2_PLC1 = 0;
        iB4_ref_swich_PLC2 = 0;
    ENDIF;
ENDIF;

```

Pump4

```

IF Pump4 == 1 AND F_pump4 == 0 AND NOT dPA_PM_PLC1 THEN
    dL1_Embotellado_PLC1=1;
    dEvaporación_L2_PLC1=1;
    dL1_Mezcla_PLC1=1;
    Valve_4 = 1;
    Celda_Llenado1_ON =1;
    Celda_Evap2_ON=1;
    IF dPA_PM_PLC1 == 1 THEN
        iBomba1_PLC1 = 0;
    ELSE

```

```

    iBomba1_PLC1 = iBomba1;
ENDIF;
ENDIF;

```

```

IF Pump4 == 0 AND NOT dPA_PM_PLC1 THEN
    Valve_4 = 0;
    IF Pump1 == 1 AND Pump3 == 0 THEN
        dEvaporación_L2_PLC1=0;
        Celda_Evap2_ON=0;
    ENDIF;
    IF Pump3 == 1 AND Pump1 == 0 THEN
        Celda_Llenado1_ON =0;
        dL1_Embotellado_PLC1=0;
        dL1_Mezcla_PLC1=0;
        iBomba1_PLC1 = 0;
        iB3_ref_swich_PLC2 = 0;
    ENDIF;
    IF Pump3==0 AND Pump1==0 THEN
        Celda_Llenado1_ON =0;
        dL1_Embotellado_PLC1=0;
        dL1_Mezcla_PLC1=0;
        dEvaporación_L2_PLC1=0;
        Celda_Evap2_ON=0;
        iBomba1_PLC1 = 0;
        iB3_ref_swich_PLC2 = 0;
    ENDIF;
ENDIF;

```

rCau_Salida_M1_PLC1

```

iBomba1=(rCau_Salida_M1_PLC1 + 0.0496) / 0.0653;

```

rCau_Salida_M2_PLC1

```

iBomba2=(rCau_Salida_M2_PLC1 + 0.0214) / 0.0641;

```

Valve3

```

IF Valve3 == 1 AND Celda_Evap1_ON == 1 AND Celda_Llenado1_ON THEN
    Pump1 = 1;
ELSE
    Pump1 = 0;
ENDIF;

```

```

IF Valve3 == 1 THEN
    Valve_3 = 1;
ELSE
    Valve_3 = 0;
ENDIF;

```

Valve4

```

IF Valve4 == 1 AND Celda_Evap2_ON == 1 AND Celda_Llenado1_ON == 1 THEN
    Pump4 = 1;
ELSE
    Pump4 = 0;

```

ENDIF;

IF Valve4 == 1 THEN

Valve_4 = 1;

ELSE

Valve_4 = 0;

ENDIF;

Valve6

IF Valve6 == 1 AND Celda_Evap2_ON == 1 AND Celda_Llenado2_ON == 1 THEN

Pump3 = 1;

ELSE

Pump3 = 0;

ENDIF;

IF Valve6 == 1 THEN

Valve_6 = 1;

ELSE

Valve_6 = 0;

ENDIF;

Valve_5

IF Valve_5 == 1 AND Celda_Evap1_ON == 1 AND Celda_Llenado2_ON == 1 THEN

Pump2 = 1;

ELSE

Pump2 = 0;

ENDIF;

IF Valve_5 == 1 THEN

Valve5 = 1;

ELSE

Valve5 = 0;

ENDIF;

Vol_botella1

rVol__Bot1_PLC1=Vol_botella1/1000;

Vol_botella2

rVol__Bot2_PLC1=Vol_botella2/1000;

Vol_Evap1

rT__Llenado_EV1_PLC1 = (Vol_Evap1 * 6) / 928;

Vol_Evap2

rT__Llenado_EV2_PLC1 = (Vol_Evap2 * 6) / 984;