

Trabajo Fin de Grado
Ingeniería Electrónica, Robótica y Mecatrónica

Automatización mediante PLC de Máquina de
Termoconformado de Piezas de Aviones

Autor: María José Vázquez Camacho

Tutor: Francisco Rodriguez Rubio

Dpto. Ingeniería de Sistemas y Automática
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla

Sevilla, 2013



Trabajo Fin de Grado
Ingeniería Electrónica Robótica y Mecatrónica

Automatización mediante PLC de Máquina de Termoconformado de Piezas de Aviones

Autor:

María José Vázquez Camacho

Tutor:

Francisco Rodríguez Rubio

Catedrático de Universidad

Dpto. de Ingeniería de Sistemas y Automática

Escuela Técnica Superior de Ingeniería

Universidad de Sevilla

Sevilla, 2019

Proyecto Fin de Carrera: Automatización mediante PLC de Máquina de Termoconformado de Piezas de Aviones

Autor: María José Vázquez Camacho

Tutor: Francisco Rodríguez Rubio

El tribunal nombrado para juzgar el Proyecto arriba indicado, compuesto por los siguientes miembros:

Presidente:

Vocales:

Secretario:

Acuerdan otorgarle la calificación de:

Sevilla, 2018

El Secretario del Tribunal

A toda mi familia

A mi novio

*A esos amigos que me ha dejado
esta etapa tan especial*

Agradecimientos

Mi Trabajo Fin de Grado va dedicado a toda mi familia, ya que sin dudarlo ni un segundo me han apoyado en todo momento, sin dejar que me rinda ni una sola vez.

En especial va dedicado a mi padre, Javier, quien ha vivido esta etapa como si fuese suya, a mi madre, Juana María, quien me llamaba cada noche antes de un examen para mandarme el beso de la suerte, a mi abuela, Josefa, quien esperaba con ansias cada nota, mucho más que yo. Sin ellos nada hubiese sido posible.

También va dedicado a esos compañeros de clases que se han convertido en amigos, y han estado ahí cuando los he necesitado.

Y, por último, quiero agradecerles a mis compañeros de trabajo, en especial a Alejandro Muñoz, la confianza que tienen en mí y todo lo que me enseñan para poder seguir creciendo y formándome profesionalmente.

María José Vázquez Camacho

Sevilla, 2018

Resumen

La automatización y la robótica cada vez están más presentes en nuestras vidas, indudablemente en el sector de la industria.

La automatización industrial es el uso de diferentes tecnologías que monitorean y controlan maquinaria, robots, aparatos o procesos que realizan tareas repetitivas, haciendo que funcionen automáticamente, reduciendo la actuación de los trabajadores al mínimo.

Las primeras máquinas que reducían el esfuerzo del ser humano fueron las poleas o las palancas, posteriormente se consiguió sustituir la energía animal o humana por energías renovables (viento, agua...). Tiempo después aparecieron las primeras máquinas automáticas tales como los mecanismos de relojería, telares, caja de música... y fueron evolucionando hasta llegar a las máquinas más modernas que van desde un ascensor hasta una línea de transportadores con robots que monta coches automáticamente.

Este documento tratará del diseño del PLC de una máquina de conformado en caliente de láminas de planos de fibra de carbono y fibra de vidrio mediante la aplicación de calor y vacío. Dicha máquina está formada por dos mesas donde se colocan piezas de aviones cubiertas por láminas y rodeadas de termopares, los cuales se encargarán de regular el ciclo de termoconformado, y por una campana que tiene una membrana, dentro de esta campana es donde se realiza el ciclo automático y donde realizando un programa de temperaturas y vacío se realiza la compactación de dichas láminas.

Abstract

Automation and robotic are increasingly present in our lives, undoubtedly in the industry sector.

Industrial automation is the use of different technologies that monitor and control machinery, robots, devices or processes that perform repetitive tasks, making them work automatically, reducing the performance of workers to a minimum.

The first machines that reduced the effort of the human being were pulleys or levers, later it was possible to substitute animal or human energy for renewable energies (wind, water ...). Some time later the first automatic machines appeared, such as watchmaking mechanisms, looms, music boxes ... and they evolved to reach the most modern machines that go from an elevator to a line of transporters with robots that assemble cars automatically.

This document will deal with the design of the PLC of a hot forming machine for sheets of carbon fiber and glass fiber planes by applying heat and vacuum. Said machine is formed by two tables where aircraft pieces are placed covered by sheets and surrounded by thermocouples, which will be in charge of regulating the thermoforming cycle, and by a bell that has a membrane, inside this bell is where the automatic cycle and where a temperature and vacuum program is carried out the compaction of said sheets.

Índice

Agradecimientos	19
Resumen	21
Abstract	23
Índice	24
Índice de Tablas	27
Índice de Figuras	28
1 Intruducción	34
1.1 Descripción general del ciclo	34
2 Descripción de la máquina	35
2.1 Sinóptico general representativo	35
2.2 Características principales de la máquina	36
2.3 Modos de funcionamiento	37
2.3.1 Condiciones de seguridad	37
2.3.2 Manual	38
2.3.3 Automático	39
2.4 Programas	41
2.5 Resultado del ciclo	42
3 Diseño del automatismo	43
3.1 Programa utilizado	43
3.2 Harware	44
3.3 Tabla de simbolos	44
3.3.1 Entradas analógicas	44
3.3.2 Entradas digitales	47
3.3.3 Salidas analógicas	50
3.3.4 Salidas digitales	50
4 Módulos del programa	53
4.1 Bloques de organización	53
4.1.1 Ob1	53
4.1.2 Ob35	55
4.1.3 Ob70	55

4.1.4 Ob72	55
4.1.5 Ob80	55
4.1.6 Ob81	56
4.1.7 Ob82	56
4.1.8 Ob83	56
4.1.9 Ob84	56
4.1.10 Ob85	56
4.1.11 Ob86	56
4.1.12 Ob88	57
4.1.13 Ob121	57
4.1.14 Ob122	57
4.2 Funciones y bloques de funciones del Ob1	57
4.2.1 Fc1: Inicializaciones	57
4.2.3 Fc9: Tratamiento analógicas	61
4.2.4 Fc20: Lectura de información del scada	69
4.2.5 Fc601: Calculo de termopares virtuales	69
4.2.6 Fc701: Selección de variables de proceso	93
4.2.7 Fc801: Decodificación de eventos	96
4.2.8 Fc4: Gestión de mesas	97
4.2.9 Fc5: Gestión de campana	105
4.2.10 Fc6: Gestión de válvulas de vacío	114
4.2.11 Fc11: Gestión del parpadeo de las barreras	116
4.2.12 Fc7: Gestión de refrigeración	116
4.2.13 Fc2: Gestión de alarmas	120
4.2.14 Fb10: Máquina de estados para ciclo automático	144
4.2.15 Fc10: Escritura de información en el SCADA	161
4.2.16 Fc8: Escritura de salidas	172
4.3 Funciones y bloques de funciones del ob35	183
4.3.1 Fb2: Gestión programadores	183
4.3.2 Fb4: Reloj ciclo auto	197
4.4 Bloques de datos	199
4.4.1 Bloques de datos globales	199
4.4.1.1 Db2: Alarmas	199
4.4.1.2 Db3: Analógicas	201
4.4.1.3 Db15: Termopar seleccionado	205
4.4.1.4 Db17: Eventos y comandos	206
4.4.1.5 Db60: Programa temperatura puntos	208

4.4.1.6 Db61: Programa temperatura eventos	208
4.4.1.7 Db70: Db60 programa vacío puntos	208
4.4.1.8 Db71: Programa vacío eventos	208
4.4.2 Bloque de datos de instancia	209
4.4.2.1 Db120: Db de instancia de la fb1	209
4.4.2.2 Db200: Db de instancia de la fb2	213
4.4.2.3 Db210: Db de instancia de la fb10	213
4.4.2.4 Db300: Db de instancia de la fb4	214
4.5 Tipo de datos de usuario (udt)	214
4.5.1 Udt1: udt_termopares	214
4.5.2 Udt2: udt_sensores_infrarojos	214
4.5.3 Udt3: udt_sensor_vacio	215
4.5.4 Udt4: udt_sensor_campana	215
4.5.5 Udt5: eventos_prog_temp	215
4.5.6 Udt6: eventos_prog_vacio	216
4.5.7 Udt7: puntos_prog	216
4.5.8 Udt8: puntos_prog_vacio	216
5 Conclusiones y futuros trabajos	217
6 Bibliografía	218

ÍNDICE DE TABLAS

Tabla 1: Entradas Analógicas	47
Tabla 2: Entradas Digitales	49
Tabla 3: Salidas Analógicas	50
Tabla 4: Salidas Digitales	52

ÍNDICE DE FIGURAS

Ilustración 1: Sinóptico de máquina de termoconformado	35
Ilustración 2: Termopares Mesa 1	35
Ilustración 3: Termopares Campana	36
Ilustración 4: Diagrama modo manual	38
Ilustración 5: Tipo de ciclo y termopares seleccionados	39
Ilustración 6: Programa de Temperatura	41
Ilustración 7: Perfil de Temperatura	41
Ilustración 8: Software STEP 7	43
Ilustración 9: Hardware	44
Ilustración 10: OB1 Segmentos 1, 2, 3 y 4	53
Ilustración 11:OB1 Segmentos 5, 6 y 7	54
Ilustración 12:OB1 Segmentos 8, 9, 10, 11 y 12	54
Ilustración 13:OB1 Segmentos 13 y 14	55
Ilustración 14: OB35 segmentos 1 y 2	55
Ilustración 15: FC1 Segmentos 1, 2, 3 y 4	58
Ilustración 16: FC1 Segmentos 5 y 6	58
Ilustración 17: Segmentos 7 y 8	59
Ilustración 18: FC1 Segmento 9_1	59
Ilustración 19: FC1 Segmento 9_2	60
Ilustración 20: FC1 Segmento 9_3	60
Ilustración 21: FC9 en AWL_1	61
Ilustración 22: FC9 en AWL_2	62
Ilustración 23:FC9 en AWL_3	63
Ilustración 24:FC9 en AWL_4	64
Ilustración 25: FC9 en AWL_5	65
Ilustración 26: FC1 en AWL_6	66
Ilustración 27: FC1 en AWL_7	67
Ilustración 28: FC1 en AWL_8	68
Ilustración 29: FC20 Segmentos 1, 2 y 3	69
Ilustración 30: FC20 Segmentos 4 y 5	69
Ilustración 31: FC4 Segmento 1_1	98
Ilustración 32: FC4 Segmento 1_2	99
Ilustración 33: FC4 Segmentos 2 y 3	99

Ilustración 34: FC4 Segmentos 4 y 5	100
Ilustración 35: FC4 Segmentos 6, 7 y 8	100
Ilustración 36: FC4 Segmentos 9, 10 y 11	101
Ilustración 37: FC4 Segmentos 12, 13 y 14	101
Ilustración 38: FC4 Segmento 16	102
Ilustración 39: FC4 Segmento 17	102
Ilustración 40: FC4 Segmento 18	103
Ilustración 41: FC4 Segmento 19	103
Ilustración 42: FC4 Segmentos 20 y 21	104
Ilustración 43: FC4 Segmentos 22, 23 y 24	104
Ilustración 44: FC4 Segmentos 25, 26 y 27	105
Ilustración 45: FC5 Segmentos 1 y 2	105
Ilustración 46: FC5 Segmentos 3 y 4	106
Ilustración 47: FC5 Segmento 5	107
Ilustración 48: FC5 Segmento 6	107
Ilustración 49: FC5 Segmentos 7, 8 y 9	108
Ilustración 50: FC5 Segmentos 10, 11 y 12	108
Ilustración 51: FC5 Segmentos 13, 14 y 15	109
Ilustración 52: FC5 Segmentos 16, 17 y 18	109
Ilustración 53: FC5 Segmento 19	110
Ilustración 54: FC5 Segmentos 20 y 21	110
Ilustración 55: FC5 Segmentos 22 y 23	111
Ilustración 56: FC5 Segmentos 24 y 25	111
Ilustración 57: FC5 Segmento 26	112
Ilustración 58: FC5 Segmento 27_1	112
Ilustración 59: FC5 Segmento 27_2	112
Ilustración 60: FC5 Segmentos 27_3	113
Ilustración 61: FC5 Segmentos 28, 29 y 30	113
Ilustración 62: FC6 Segmento 1	114
Ilustración 63: FC6 Segmentos 2 y 3	114
Ilustración 64: FC6 Segmentos 4 y 5	115
Ilustración 65: FC6 Segmento 6	115
Ilustración 66: FC11 Segmentos 1 y 2	116
Ilustración 67: FC7 Segmentos 1 y 2	116
Ilustración 68: FC7 Segmento 3	117
Ilustración 69: FC7 Segmentos 4 y 5	117
Ilustración 70: FC7 Segmento 6	118
Ilustración 71: FC7 Segmento 7_1	118
Ilustración 72: FC7 Segmento 7_2	118

Ilustración 73: FC7 Segmentos 7_3	119
Ilustración 74: FC7 Segmento 7_4	119
Ilustración 75: FC7 Segmento 7_5	120
Ilustración 76: FC7 Segmento 7_6	120
Ilustración 77: FC2 Segmento 1	120
Ilustración 78: FC2 Segmento 2_1	121
Ilustración 79: FC2 Segmento 2_2	121
Ilustración 80: FC2 Segmento 3_1	122
Ilustración 81: FC2 Segmento 3_2	122
Ilustración 82: FC2 Segmentos 4, 5, 6 y 7	123
Ilustración 83: FC2 Segmento 8	123
Ilustración 84: FC2 Segmento 9	124
Ilustración 85: FC2 Segmento 10	124
Ilustración 86: FC2 Segmento 11	125
Ilustración 87: FC2 Segmento 12	125
Ilustración 88: FC2 Segmentos 13 y 14	126
Ilustración 89: FC2 Segmento 15	127
Ilustración 90: FC2 Segmento 16_1	128
Ilustración 91: FC2 Segmento 16_2	128
Ilustración 92: FC2 Segmentos 17 y 18	129
Ilustración 93: FC2 Segmento 19 y 20	129
Ilustración 94: FC2 Segmentos 21, 22 y 23	130
Ilustración 95: FC2 Segmentos 24 y 25	131
Ilustración 96: FC2 Segmentos 26 y 27	132
Ilustración 97: FC2 Segmentos 28 y 29	133
Ilustración 98: FC2 Segmento 30	133
Ilustración 99: FC2 Segmento 31	134
Ilustración 100: FC2 Segmentos 32	135
Ilustración 101: FC2 Segmentos 33 y 34	136
Ilustración 102: FC2 Segmentos 35 y 36	137
Ilustración 103: FC2 Segmentos 38 y 39	137
Ilustración 104: FC2 Segmento 40_1	138
Ilustración 105: FC2 Segmento 40_2	138
Ilustración 106: FC2 Segmento 40_3	139
Ilustración 107: FC2 Segmento 40_4	139
Ilustración 108: FC2 Segmento 40_5	140
Ilustración 109: FC2 Segmento 41_1	140
Ilustración 110: FC2 Segmento 41_2	141

Ilustración 111: FC2 Segmento 41_3	141
Ilustración 112: FC2 Segmento 41_4	142
Ilustración 113: FC2 Segmento 41_5	142
Ilustración 114: FC2 Segmento 41_6	143
Ilustración 115: FC2 Segmento 41_7	143
Ilustración 116: FB10 Segmentos 1, 2 y 3	145
Ilustración 117: FB10 Segmentos 4, 5 y 6	146
Ilustración 118: FB10 Segmentos 7, 8 y 9	146
Ilustración 119: FB10 Segmentos 10, 11, 12, 13 y 14	147
Ilustración 120: FB10 Segmentos 15,16 y 17	148
Ilustración 121: FB10 Segmentos 18, 19 y 20	149
Ilustración 122: FB10 Segmentos 21, 22 y 23	150
Ilustración 123: FB10 Segmentos 24, 25 y 26	151
Ilustración 124: FB10 Segmento 27_1	152
Ilustración 125: FB10 Segmento 27_2	152
Ilustración 126: FB10 Segmento 28_1	153
Ilustración 127: FB10 Segmento 28_2	153
Ilustración 128: FB10 Segmentos 29, 30 y 31	154
Ilustración 129: FB10 Segmentos 32 y 33	155
Ilustración 130: FB10 Segmentos 34 y 35	156
Ilustración 131: FB10 Segmentos 36, 37 y 38	157
Ilustración 132: FB10 Segmento 39_1	158
Ilustración 133: FB10 Segmento 39_2	158
Ilustración 134: FB10 Segmentos 40 y 41	159
Ilustración 135: FB10 Segmentos 42 y 43	160
Ilustración 136: FB10 Segmentos 44 y 45	161
Ilustración 137: FC10 Segmentos 1, 2 y 3	162
Ilustración 138: FC10 Segmentos 4, 5, 6 y 7	163
Ilustración 139: FC10 Segmentos 8, 9, 10 y 11	164
Ilustración 140: FC10 Segmentos 12, 13 y 14	165
Ilustración 141: FC10 Segmentos 15, 16 y 17	166
Ilustración 142: FC10 Segmentos 18, 19, 20 y 21	167
Ilustración 143: FC10 Segmentos 22, 23 y 24	168
Ilustración 144: FC10 Segmentos 25, 26 y 27	169
Ilustración 145: FC10 Segmentos 28, 29 y 30	170
Ilustración 146: FC10 Segmentos 31, 32 y 33	171
Ilustración 147: FC10 Segmentos 34 y 35	171
Ilustración 148: FC8 Segmentos 1 y 2	173
Ilustración 149: FC8 Segmento 3	173

Ilustración 150: FC8 Segmentos 4 y 5	174
Ilustración 151 FC8 Segmentos 9, 10 y 11	175
Ilustración 152 FC8 Segmento 13	175
Ilustración 153 FC8 Segmento 14_1	176
Ilustración 154 FC8 Segmento 14_2	177
Ilustración 155 FC8 Segmentos 15 y 16	177
Ilustración 156 FC8 Segmentos 17, 18 y 19	178
Ilustración 157 FC8 Segmentos 20, 21, 22 y 23	179
Ilustración 158 FC8 Segmentos 24,25, 26 y 27	180
Ilustración 159 FC8 Segmentos 28, 29, 30 y 31	181
Ilustración 160 FC8 Segmentos 32 y 33	181
Ilustración 161 FC8 Segmento 34_1	182
Ilustración 162 FC8 Segmentos 34, 35, 36 y 37	183
Ilustración 163: FB2 Segmentos 1, 2 y 3	184
Ilustración 164: FB2 Segmentos 4, 5 y 6	185
Ilustración 165: FB2 Segmentos 7, 8 y 9	186
Ilustración 166: FB2 Segmentos 10, 11 y 12	186
Ilustración 167: FB2 Segmentos 13,14 y 15	187
Ilustración 168: FB2 Segmento 16	187
Ilustración 169: FB2 Segmento 17	188
Ilustración 170: FB2 Segmento 18_1	188
Ilustración 171: FB2 Segmento 18_2	189
Ilustración 172: FB2 Segmentos 19, 20 y 21	189
Ilustración 173: FB2 Segmento 22_1	190
Ilustración 174: FB2 Segmento 22_2	191
Ilustración 175: FB2 Segmento 22_3	192
Ilustración 176: FB2 Segmento 22_4	193
Ilustración 177: FB2 Segmento 22_5	194
Ilustración 178: FB2 Segmento 22_6	195
Ilustración 179: FB2 Segmentos 23 y 24	195
Ilustración 180: FB2 Segmentos 25, 26 y 27	196
Ilustración 181: FB2 Segmentos 28 y 29	196
Ilustración 182: FB4 Segmentos 1, 2 y 3	197
Ilustración 183: FB4 Segmentos 4 y 5	198
Ilustración 184: FB4 Segmento 6	198
Ilustración 185: FB4 Segmento 7	199
Ilustración 186: DB2_1	200
Ilustración 187: DB2_2	200

Ilustración 188: DB3_1	201
Ilustración 189: DB3_2	202
Ilustración 190:DB3_3	203
Ilustración 191: DB3_4	204
Ilustración 192: DB3_5	205
Ilustración 193: DB15	205
Ilustración 194: DB17_1	206
Ilustración 195: DB17_2	207
Ilustración 196: DB17_3	207
Ilustración 197: DB60	208
Ilustración 198: DB61	208
Ilustración 199: DB70	208
Ilustración 200: DB71	209
Ilustración 201: DB120_1	209
Ilustración 202: DB120_2	210
Ilustración 203: DB120_3	211
Ilustración 204: DB120_4	212
Ilustración 205: DB120_5	213
Ilustración 206: DB200	213
Ilustración 207: DB210	214
Ilustración 208: DB300	214
Ilustración 209: UDT1	214
Ilustración 210: UDT2	215
Ilustración 211: UDT3	215
Ilustración 212: UDT4	215
Ilustración 213: UDT5	215
Ilustración 214: UDT6	216
Ilustración 215: UDT7	216
Ilustración 216: UDT8	216

1 INTRODUCCIÓN

Este documento pretende mostrar parte de un trabajo realizado en una fábrica de piezas de aviones, concretamente se centrará en el diseño e implementación de un sistema automático con el cual se consigue hacer el termoconformado de láminas de planos de fibra de carbono y fibra de vidrio mediante la aplicación de calor y vacío. Para ello se ha usado un programador lógico programable de Siemens, concretamente una CPU 412-5 PN/DP y se ha realizado en Step7.

1.1 DESCRIPCIÓN GENERAL DEL CICLO

La máquina de termoconformado de piezas está formada por una campana y dos mesas, su funcionamiento es el siguiente:

Primero el operario se encarga de poner las piezas y todo lo necesario encima de una de las mesas, a continuación, a través del SCADA le indica al PLC que mesa quiere usar y que tipo de receta quiere usar.

Una receta se compone de dos programas, uno de temperatura y otro de vacío, los cuales serán los que sigan los controladores PID del PLC.

Una vez cumplidas todas las condiciones de seguridad y el operario ha pulsado inicio de ciclo, el ciclo comienza:

En primer lugar, la mesa seleccionada entra debajo de la campana, una vez que la mesa está dentro (debajo de la campana) la campana comienza a bajar, y cuando llega a la posición de precalentamiento empieza a calentarse. Transcurrido el tiempo de precalentamiento la campana baja completamente y una vez ahí comienzan a actuar los reguladores PID para seguir el programa de temperatura seleccionado, hasta que le llega la señal de comienzo de programa de vacío. Una vez comenzado el programa de vacío se hará el vacío dentro de la campana hasta que llegue la señal de inicio de refrigeración, cuando esta señal llega se apaga el programa de vacío y se activa la refrigeradora, la cual está controlada por un controlador Johnson. Una vez la temperatura ha bajado de la consigna de refrigeración el programa de vacío acaba y se activa la válvula de rotura de vacío. El ciclo terminará cuando la presión de la máquina ha subido de -5mmHg entonces la campana sube, y una vez arriba la mesa sale.

A continuación, se describirán la máquina y la programación realizada.

2 DESCRIPCIÓN DE LA MÁQUINA

2.1 SINÓPTICO GENERAL REPRESENTATIVO

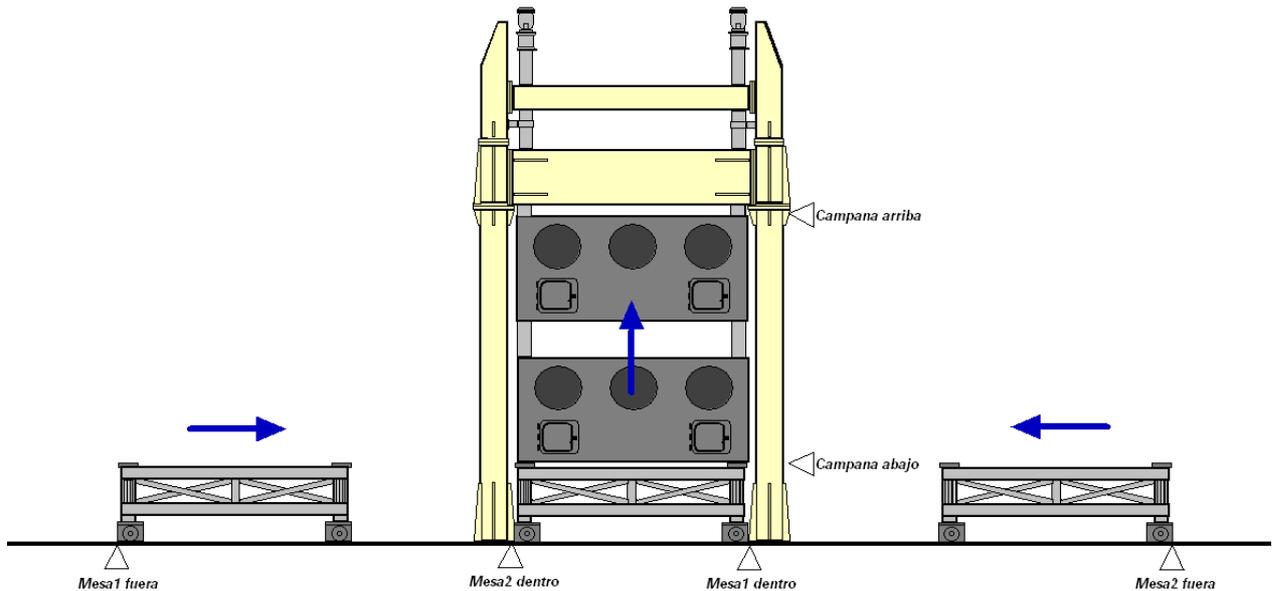


Ilustración 1: Sinóptico de máquina de termoconformado

El sistema realiza termoconformado de piezas de fibra de vidrio o fibra de carbono, mediante la aplicación de vacío y calor.

Para ello se sirve de dos mesas de vacío con desplazamiento motorizado, cada una de ellas dividida en 4 zonas de trabajo. Cada zona de trabajo de la mesa desplazable consta de 8 termopares, haciendo un total de 32 termopares por mesa.

<p>Zona 3 Activa</p> <p><input checked="" type="checkbox"/> TC17M1 <input checked="" type="checkbox"/> TC18M1</p> <p><input checked="" type="checkbox"/> TC19M1 <input checked="" type="checkbox"/> TC20M1</p> <p><input checked="" type="checkbox"/> TC21M1 <input checked="" type="checkbox"/> TC22M1</p> <p><input checked="" type="checkbox"/> TC23M1 <input checked="" type="checkbox"/> TC24M1</p> <p><input type="checkbox"/> Zona3MAX <input type="checkbox"/> Zona3AVG <input type="checkbox"/> Zona3MIN</p>	<p>Zona 1 Activa</p> <p><input checked="" type="checkbox"/> TC1M1 <input checked="" type="checkbox"/> TC2M1</p> <p><input checked="" type="checkbox"/> TC3M1 <input checked="" type="checkbox"/> TC4M1</p> <p><input checked="" type="checkbox"/> TC5M1 <input checked="" type="checkbox"/> TC6M1</p> <p><input checked="" type="checkbox"/> TC7M1 <input checked="" type="checkbox"/> TC8M1</p> <p><input type="checkbox"/> Zona1MAX <input type="checkbox"/> Zona1AVG <input type="checkbox"/> Zona1MIN</p>
<p>Zona 4 Activa</p> <p><input checked="" type="checkbox"/> TC25M1 <input checked="" type="checkbox"/> TC26M1</p> <p><input checked="" type="checkbox"/> TC27M1 <input checked="" type="checkbox"/> TC28M1</p> <p><input checked="" type="checkbox"/> TC29M1 <input checked="" type="checkbox"/> TC30M1</p> <p><input checked="" type="checkbox"/> TC31M1 <input checked="" type="checkbox"/> TC32M1</p> <p><input checked="" type="checkbox"/> Zona4MAX <input type="checkbox"/> Zona4AVG <input type="checkbox"/> Zona4MIN</p>	<p>Zona 2 Activa</p> <p><input checked="" type="checkbox"/> TC9M1 <input checked="" type="checkbox"/> TC10M1</p> <p><input checked="" type="checkbox"/> TC11M1 <input checked="" type="checkbox"/> TC12M1</p> <p><input checked="" type="checkbox"/> TC13M1 <input checked="" type="checkbox"/> TC14M1</p> <p><input checked="" type="checkbox"/> TC15M1 <input checked="" type="checkbox"/> TC16M1</p> <p><input checked="" type="checkbox"/> Zona2MAX <input type="checkbox"/> Zona2AVG <input type="checkbox"/> Zona2MIN</p>

Ilustración 2: Termopares Mesa 1

Estas mesas de vacío estarán fuera de la zona de campana, cada una a un lado, para la preparación de las piezas a tratar, o dentro de la campana para realizar un ciclo de trabajo manual o automático. Por tanto, estas dos mesas de vacío tienen dos posiciones válidas, dentro o fuera de la campana, realizando su transición de una a otra con ayuda motorizada, bajo órdenes del pupitre del operador y siempre que se den las condiciones operacionales y de seguridad adecuadas.

Una vez preparada la pieza, es introducida la mesa en posición de trabajo, esto es, debajo de la campana. La campana posee un accionamiento motorizado vertical accionado por el operario desde el pupitre, así como tres posiciones de trabajo: retraído o posición de descanso, posición intermedia de precalentamiento y posición extendida para permitir la operación de vacío. También posee una membrana para permitir la operación de vacío, un sistema de refrigeración independiente, 4 resistencias controladas por tiristores y un total de 8 termopares divididos en 4 zonas (su valor está representado con un fondo azul), coincidentes con las zonas delimitadas por las resistencias y las zonas de trabajo de la mesa de vacío. También consta con un total de 8 termopares de infrarrojos adicionales a la membrana, dos por cada zona de trabajo (su valor está representado con un fondo naranja).

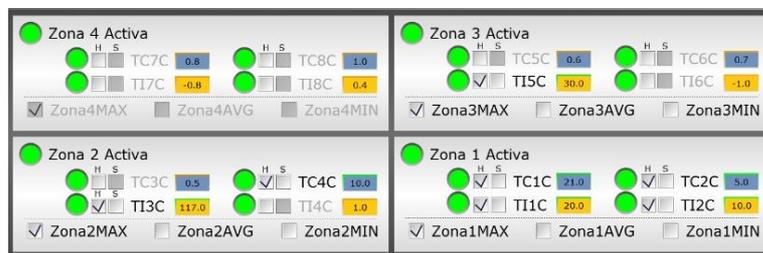


Ilustración 3: Termopares Campana

Cuando la mesa ya está en posición de trabajo y la campana en posición extendida comenzaría la operación de termoconformado. Esta operación podrá llevarse a cabo de modo Manual o de modo Automático donde el PLC se encarga de todo el ciclo.

2.2 CARACTERÍSTICAS PRINCIPALES DE LA MÁQUINA

Anchura útil de conformado.....	3.500 mm
Máxima anchura de mesa.....	3.830 mm
Longitud útil de Conformado.....	15.500 mm
Altura de la superficie de trabajo.....	900 mm
Área de calentamiento, 4 zonas de.....	7.750 x 1.750 mm
Potencia de calefacción.....	224 kW
Máximo vacío.....	600 mm Hg
Recorrido máx. regulable de la Campana.....	1.200 mm
Potencia frigorífica enfriadora.....	210 kW
Caudal climatizador.....	30.000 m3/h
Presión máx. regulable equipo neumático.....	7 bar
Velocidad Campana Descenso.....	30 mm/s
Velocidad Campana Retorno.....	30 mm/s
Mesa de Trabajo Desplazable	
Desplazamiento rápido.....	180 mm/s

Desplazamiento lento.....45 mm/s

2.3 MODOS DE FUNCIONAMIENTO

2.3.1 CONDICIONES DE SEGURIDAD

Las mesas no se pueden mover si:

- Hay alguien en el recinto, el recinto está bordeado por barreras de seguridad fotoeléctricas, estas barreras se activan cuando alguien las cruza, para su rearme hay que pulsar un botón que hay en el pupitre.
- La campana no está en su posición home o cero (arriba).
- Están sobrecargadas.
- Hay alguna alarma.
- No está seleccionada desde el pupitre (hay un selector en el pupitre que indica que mesa se va a mover, nunca se pueden mover las dos a la vez).
- La mesa que no está seleccionada no está fuera de la campana.

La campana no se puede mover si:

- Hay alguien en el recinto, el recinto está bordeado por barreras de seguridad fotoeléctricas, estas barreras se activan cuando alguien las cruza, para su rearme hay que pulsar un botón que hay en el pupitre.
- No hay una mesa dentro.
- Tiene los cilindros o los variadores sobrecargados.
- Hay alguna alarma.
- En caso de querer moverla hacia arriba hay menos de un -5mmHg de vacío en la campana o la temperatura en el interior de la campana es superior a 40°C.

Las resistencias de la campana no se activarán si:

- No hay una mesa dentro.
- La campana no está abajo.
- Hay alguna alarma.

La bomba de vacío y la válvula de rotura de vacío no se activará si:

- No hay una mesa dentro.
- La campana no está abajo.
- Hay alguna alarma.

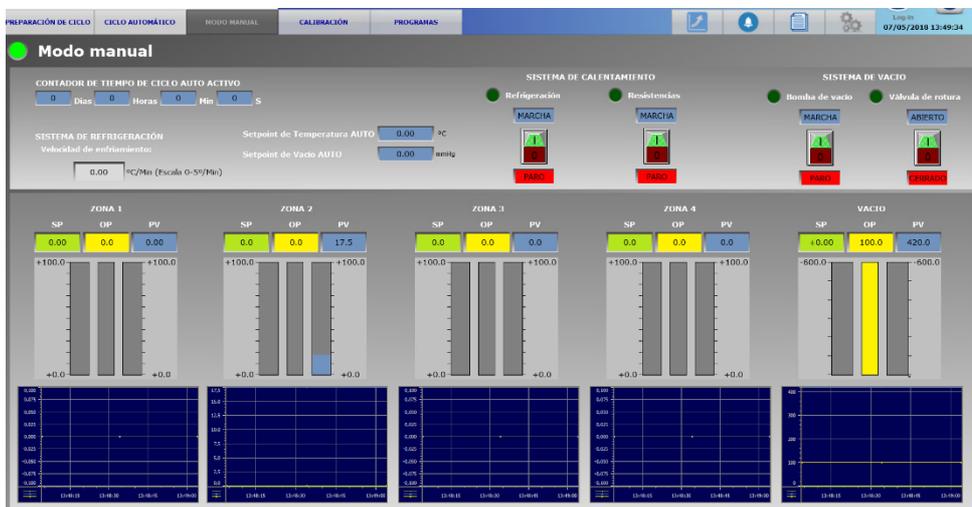
2.3.2 MODO MANUAL

La máquina consta de una pantalla que facilita la operación manual de los distintos elementos y dispositivos de la máquina, para operar con ellos es necesario que el selector del pupitre se encuentre en modo manual.

Independientemente a esta pantalla, un operario podrá controlar manualmente los elementos disponibles en el pupitre de control ya que son necesarios para preparar la máquina para un ciclo automático o para recuperar un estado de error.

En concreto siempre estarán disponibles los controles de seleccionar mesa, sacar mesa, meter mesa, subir campana, bajar campana, modo manual o automático, permiso de movimiento, resets y setas de emergencia.

Para los demás controles es obligatorio encontrarse en esta pantalla. Desde aquí el operario podrá controlar las resistencias de la campana, el marcha/paro de la refrigeración, la consigna de refrigeración, la bomba de vacío y la válvula de rotura de vacío.



En esta pantalla se puede observar un contador de ciclo automático acumulado, el cual sólo se pondrá a cero reseteándolo.

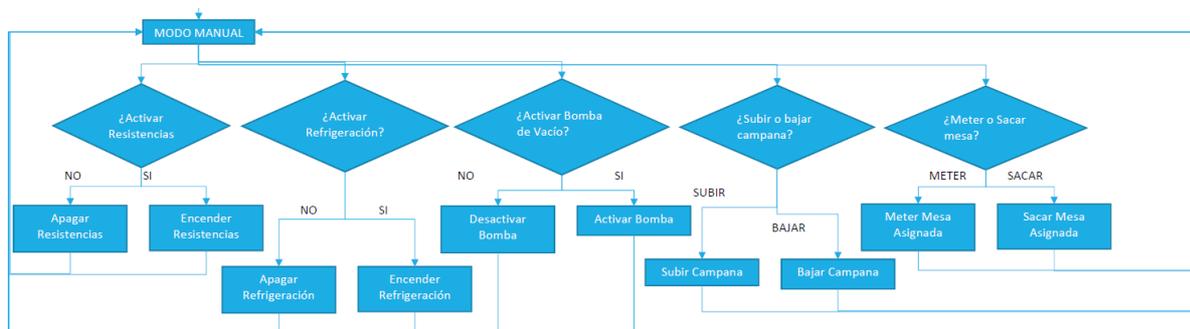


Ilustración 4: Diagrama modo manual

2.3.3 MODO AUTOMÁTICO

En modo automático la máquina realiza ciclos automáticos, sólo puede realizarlos de uno en uno, siendo necesario la participación del operario para preparar el ciclo. Para su preparación lo primero que debe de hacer es seleccionar el tipo de ciclo que se va a ejecutar.

La máquina permite dos formas de funcionamiento de ciclo, de tipo Membrana o de tipo Pieza:

- Tipo Membrana: Es un ciclo donde el control de las resistencias de se realiza con cuatro PID's los cuales tienen como variables de procesos o de control cuatro termopares pertenecientes a la campana.
- Tipo Pieza: Es un ciclo donde el control de las resistencias de se realiza con cuatro PID's los cuales tienen como variables de procesos o de control cuatro termopares pertenecientes a la mesa con la cual se va a realizar el ciclo automático.

El resto de los termopares se usarán para registrar, monitorizar y validar el proceso.

Una vez seleccionado el tipo de ciclo que queremos (Membrana o Pieza) también habría que indicarle a la máquina que modo de funcionamiento se quiere realizar puesto que hay dos modos:

- Un modo de trabajo de '4 Zonas', en el cual los cuatro PID's mencionados anteriormente se regulan con distintas variables de procesos, es decir, se elige un termopar de cada zona y ese termopar es el que servirá de referencia al PID que controla el tiristor de esa zona. Por lo que la Zona 1 estará controlada por un termopar de la Zona 1, la Zona 2 por un termopar de la Zona 2, la Zona 3 con un termopar de la Zona 3 y la Zona 4 por un termopar de la Zona 4.
- Un modo de trabajo de '1 Zona', este modo simplificará el control de las resistencias de calentamiento a un único termopar de toda la mesa, en caso de tipo pieza, o de toda la campana, en caso de tipo membrana, en lugar de uno por cada una de las zonas habituales. Por lo que los cuatro PID's están regulados por la misma variable de proceso.

A la hora de seleccionar el termopar de control, se podrá seleccionar directamente uno de los existentes o un termopar virtual, estos termopares virtuales se corresponden al máximo, media y mínimo de todos los termopares habilitado en cada zona, existen termopares virtuales en cada zona de la campana, en cada zona de la mesa, en el total de la campana, en el total de la mesa y en el total del conjunto campana y mesa. Los termopares virtuales de los tres últimos casos sólo podrán ser seleccionados para el modo de trabajo de 1 Zona.

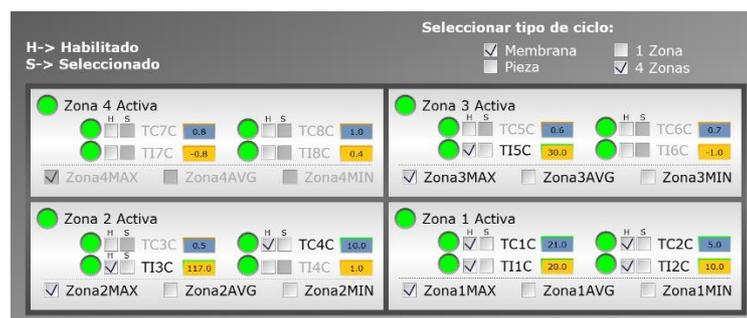


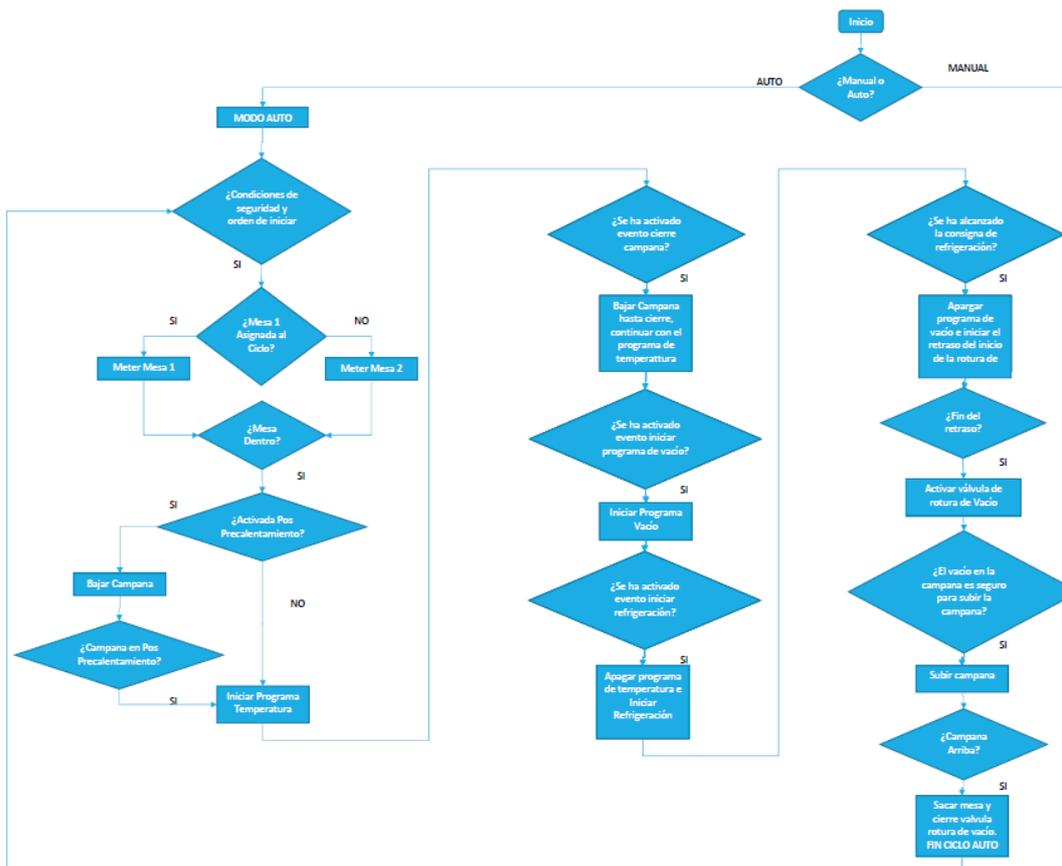
Ilustración 5: Tipo de ciclo y termopares seleccionados

En la ilustración 5 podemos ver un ejemplo del tipo de ciclo que podría ejecutarse, en este caso es un ciclo de tipo Membrana (se controlará con los termopares de la campana), trabajará a 4 Zonas (los tiristores se regularán de forma individual) y los termopares de control seleccionados son los máximos de cada zona. También se pueden observar los termopares habilitados en cada zona de la campana. Por ejemplo, en la zona 3 sólo se encuentra habilitado uno (TI5C) por lo que su valor coincidirá con el máximo de esa zona.

Una vez seleccionado el tipo de ciclo y los termopares de control, se debe de poner la máquina en condiciones seguras para arrancar el ciclo, estas condiciones son las siguientes:

- Mesas fuera.
- Campana arriba.
- Barreras fotoeléctricas rearmadas.
- Ninguna alarma activa.
- Selector en modo Auto.

Una vez se cumplan estas condiciones, al pulsar el botón de inicio se iniciará el ciclo automático, el cual sigue una máquina de estados lineal, la cual se muestra a continuación, es importante indicar que cualquier alarma grave (aquellas que puedan afectar al ciclo o a las personas que están cerca de la máquina) o un cambio de automático a manual, abortará el ciclo y será necesario volver a las condiciones nombradas anteriormente de forma manual para poder ejecutar un nuevo ciclo automático.



El funcionamiento del ciclo automático es el siguiente:

- En primer lugar, la mesa seleccionada entra debajo de la campana, una vez que la mesa está dentro (debajo de la campana) la campana comienza a bajar, y cuando llega a la posición de precalentamiento empieza a calentarse.
- Transcurrido el tiempo de precalentamiento la campana baja completamente y una vez ahí comienzan a actuar los reguladores PID para seguir el programa de temperatura seleccionado, el cual sigue un perfil de temperatura, hasta que le llega la señal de comienzo de programa de vacío.
- Una vez comenzado el programa de vacío se hará el vacío dentro de la campana hasta que llegue la

señal de inicio de refrigeración, cuando esta señal llega se apaga el programa de vacío y se activa la refrigeradora, la cual está controlada por un controlador Johnson.

- Una vez la temperatura ha bajado de la consigna de refrigeración el programa de vacío acaba y se activa la válvula de rotura de vacío. El ciclo terminará cuando la presión de la máquina ha subido de -5mmHg entonces la campana sube, y una vez arriba la mesa sale.

2.4 PROGRAMAS

Durante el ciclo se realizan dos programas para controlarlo, uno de temperatura y otro de vacío, ambos tienen la misma estructura, se componen de varios segmentos que se irán realizando de uno en uno. Cada segmento está formado por su amplitud (indica hasta que temperatura tiene que llegar), duración (en cuanto tiempo tiene que llegar a dicha temperatura) y por el evento que se realiza en él (abrir campana o iniciar programa de vacío en el caso del programa de temperatura, e iniciar refrigeración o fin de programa en el caso del programa de vacío).

Dichos segmentos forman un perfil de temperatura o de vacío, el cual es seguido por los controladores. En las siguientes ilustraciones podemos observar la configuración de un programa de temperatura y su perfil:

Segmentos:				
Nr.	Amplitud (°C)	Duración (min)	Cerrar Campana	Iniciar Programa Vacío
1	18	1	True	False
2	18	27	False	False
3	61	6	False	False
4	61	15	False	True
5	61	0	False	False

Ilustración 6: Programa de Temperatura

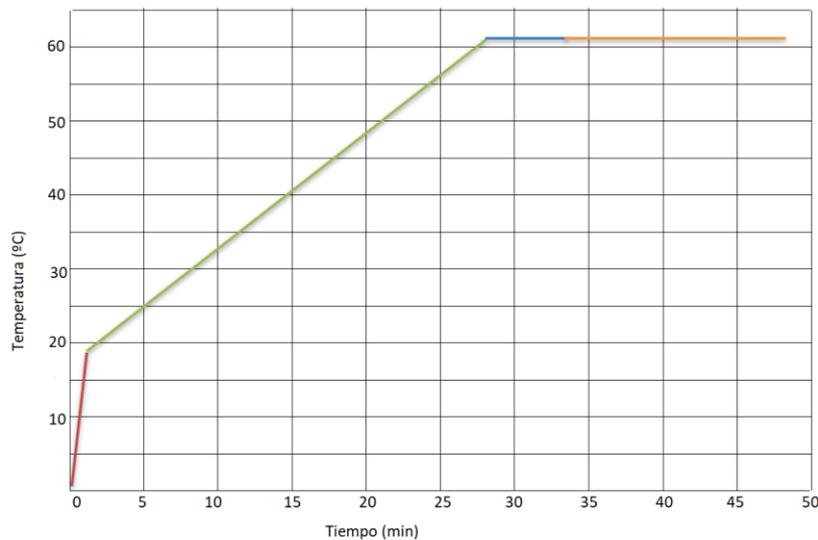


Ilustración 7: Perfil de Temperatura

El sistema soporta la carga, edición y creación de programas que controlan y caracterizan los ciclos de temperatura y vacío, así como otra serie de parámetros necesarios tales y como el control de la velocidad de enfriamiento, el control de la válvula de rotura de vacío y la posición de precalentamiento de la campana.

Soporta dos modos de marcha manual y automático, ambos seleccionados desde el pupitre. Además, permite la ejecución paso a paso de los segmentos que componen los programas cargados. El modo automático ejecutará el ciclo completo según los programas cargados y la forma de funcionamiento seleccionada, siempre y cuando

se cumplan las condiciones operacionales y de seguridad para su inicio. Por otro lado, el modo manual, permite controlar cada uno de los elementos de forma arbitraria, así como ejecutar paso a paso los programas cargados y manejar los controladores de temperatura y vacío de forma independiente.

2.5 RESULTADOS DE UN CICLO

Una vez acabado el ciclo se generan una serie de informes en los cuales se ven los siguientes datos:

Tipo de Ciclo: Pieza
4Zonas
Mesa 2

Programa de Temperatura Seguido

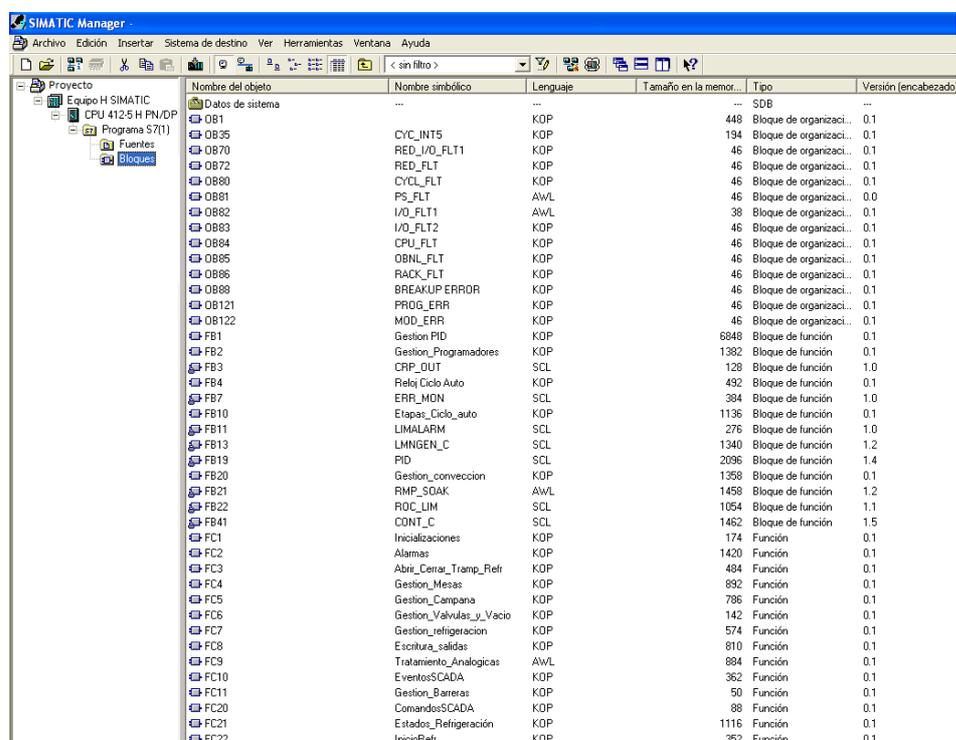
Programa de Vacío Seguido

Gráficas:
Termopares de Mesa
Termopares de campana
Vacío

3 DISEÑO DEL AUTOMATISMO

3.1 SOFTWARE DE PROGRAMACIÓN

El programa se ha realizado en STEP1 7 o S7, el cual es un software de programación de programadores lógicos programables de Siemens. STEP 7 utiliza la norma de programación DIN EN 61131-3, en el se puede programar en tres lenguajes distintos: FBS o FUP (diagrama de funciones), en la realización de este proyecto no se ha utilizado este lenguaje. AWL o STL (lista de instrucciones) este lenguaje se ha utilizado para la lectura y el escalado de las entradas analógicas, recorriendo su DB con un puntero. Y KOP, LD o LAD (diagrama de contacto), la mayoría de este trabajo se ha realizado en lenguaje de contacto. También dispone de un lenguaje de texto estructurado SCL, con el cual se han programado 3 funciones.



Nombre del objeto	Nombre simbólico	Lenguaje	Tamaño en la memor...	Tipo	Versión (encabezado)
Datos de sistema	---	---	---	SDB	---
OB1	---	KOP	448	Bloque de organizaci...	0.1
OB35	CYC_INT5	KOP	194	Bloque de organizaci...	0.1
OB70	RED_I/O_FLT1	KOP	45	Bloque de organizaci...	0.1
OB72	RED_FLT	KOP	45	Bloque de organizaci...	0.1
OB80	CYCL_FLT	KOP	45	Bloque de organizaci...	0.1
OB81	PS_FLT	AWL	45	Bloque de organizaci...	0.0
OB82	I/O_FLT1	AWL	38	Bloque de organizaci...	0.1
OB83	I/O_FLT2	KOP	45	Bloque de organizaci...	0.1
OB84	CPU_FLT	KOP	45	Bloque de organizaci...	0.1
OB85	DBNL_FLT	KOP	45	Bloque de organizaci...	0.1
OB86	RACK_FLT	KOP	45	Bloque de organizaci...	0.1
OB88	BREAKUP_ERROR	KOP	45	Bloque de organizaci...	0.1
OB121	PROG_ERR	KOP	45	Bloque de organizaci...	0.1
OB122	MOD_ERR	KOP	45	Bloque de organizaci...	0.1
FB1	Gestion PID	KOP	6848	Bloque de función	0.1
FB2	Gestion_Programadores	KOP	1382	Bloque de función	0.1
FB3	CRP_OUT	SCL	128	Bloque de función	1.0
FB4	Reloj_Ciclo_Auto	KOP	492	Bloque de función	0.1
FB7	ERR_MON	SCL	384	Bloque de función	1.0
FB10	Etapas_Ciclo_autom	KOP	1136	Bloque de función	0.1
FB11	LIMLARM	SCL	276	Bloque de función	1.0
FB13	LMNGEN_C	SCL	1340	Bloque de función	1.2
FB19	PID	SCL	2096	Bloque de función	1.4
FB20	Gestion_conveccion	KOP	1358	Bloque de función	0.1
FB21	RMP_SDAK	AWL	1458	Bloque de función	1.2
FB22	ROC_LIM	SCL	1054	Bloque de función	1.1
FB41	CONT_C	SCL	1452	Bloque de función	1.5
FC1	Inicializaciones	KOP	174	Función	0.1
FC2	Alarmas	KOP	1420	Función	0.1
FC3	Abriu_Cerrar_Tramp_Refr	KOP	484	Función	0.1
FC4	Gestion_Mesas	KOP	632	Función	0.1
FC5	Gestion_Campana	KOP	786	Función	0.1
FC6	Gestion_Valvulas_y_Vacio	KOP	142	Función	0.1
FC7	Gestion_refrigeracion	KOP	574	Función	0.1
FC8	Escritura_salidas	KOP	810	Función	0.1
FC9	Tratamiento_Analogicas	AWL	894	Función	0.1
FC10	EventosSCADA	KOP	362	Función	0.1
FC11	Gestion_Barreras	KOP	50	Función	0.1
FC20	ComandosSCADA	KOP	88	Función	0.1
FC21	Estados_Refrigeracion	KOP	1116	Función	0.1
FC22	InicioRefr	KOP	352	Función	0.1

Ilustración 8: Software STEP 7

3.2 HARWARE

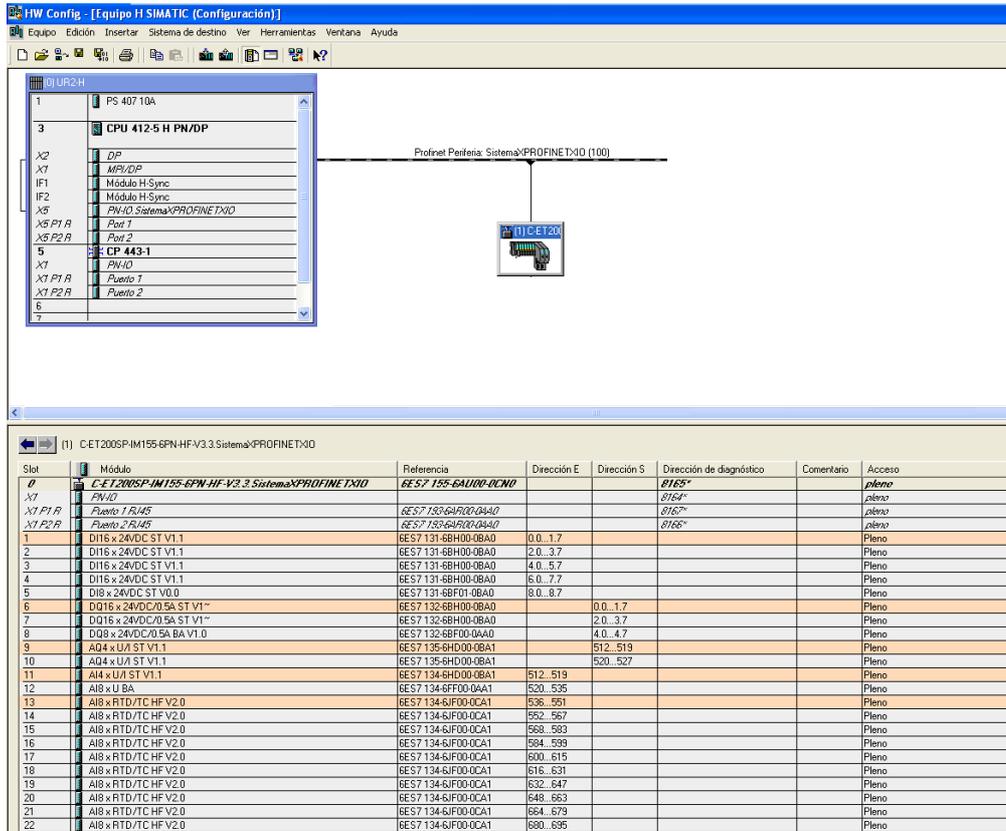


Ilustración 9: Hardware

Para la realización de este proyecto se ha usado una CPU 412-5H PN/DP y una ET200SP, también se han utilizado los siguientes módulos de expansión:

- 4 Módulos de entradas digitales 6ES7 131-6BH00-0BA0 con 16 DI cada uno.
- 1 Módulo de entradas digitales 6ES7 131-6BF01-0BA0 con 8 DI.
- 2 Módulos de salidas digitales 6ES7 132-6BH00-0BA0 con 16 DQ cada uno.
- 1 Módulo de salidas digitales 6ES7 132-6BF00-0BA0 con 8 DI.
- 2 Módulos de salidas analógicas 6ES7 135-6HD00-0BA1 con 4 AQ cada uno.
- 1 Módulo de entradas analógicas 6ES7 134-6HD00-0AA1 con 4 AQ.
- 10 Módulo de entradas analógicas 6ES7 134-6JF00-0CA1 con 8 AQ cada uno.

3.3 TABLA DE SIMBOLOS

3.3.1 ENTRADAS ANALÓGICAS

Transmisor_vacio	PEW 512	WORD
Posición campana	PEW 514	WORD

RESERVA	PEW 516	WORD
RESERVA	PEW 518	WORD
TI1C	PEW 520	WORD
TI2C	PEW 522	WORD
TI3C	PEW 524	WORD
TI4C	PEW 526	WORD
TI5C	PEW 528	WORD
TI6C	PEW 530	WORD
TI7C	PEW 532	WORD
TI8C	PEW 534	WORD
TC1M1	PEW 536	WORD
TC2M1	PEW 538	WORD
TC3M1	PEW 540	WORD
TC4M1	PEW 542	WORD
TC5M1	PEW 544	WORD
TC6M1	PEW 546	WORD
TC7M1	PEW 548	WORD
TC8M1	PEW 550	WORD
TC9M1	PEW 552	WORD
TC10M1	PEW 554	WORD
TC11M1	PEW 556	WORD
TC12M1	PEW 558	WORD
TC13M1	PEW 560	WORD
TC14M1	PEW 562	WORD
TC15M1	PEW 564	WORD
TC16M1	PEW 566	WORD
TC17M1	PEW 568	WORD
TC18M1	PEW 570	WORD
TC19M1	PEW 572	WORD
TC20M1	PEW 574	WORD
TC21M1	PEW 576	WORD

TC22M1	PEW 578	WORD
TC23M1	PEW 580	WORD
TC24M1	PEW 582	WORD
TC25M1	PEW 584	WORD
TC26M1	PEW 586	WORD
TC27M1	PEW 588	WORD
TC28M1	PEW 590	WORD
TC29M1	PEW 592	WORD
TC30M1	PEW 594	WORD
TC31M1	PEW 596	WORD
TC32M1	PEW 598	WORD
TC1M2	PEW 600	WORD
TC2M2	PEW 602	WORD
TC3M2	PEW 604	WORD
TC4M2	PEW 606	WORD
TC5M2	PEW 608	WORD
TC6M2	PEW 610	WORD
TC7M2	PEW 612	WORD
TC8M2	PEW 614	WORD
TC9M2	PEW 616	WORD
TC10M2	PEW 618	WORD
TC11M2	PEW 620	WORD
TC12M2	PEW 622	WORD
TC13M2	PEW 624	WORD
TC14M2	PEW 626	WORD
TC15M2	PEW 628	WORD
TC16M2	PEW 630	WORD
TC17M2	PEW 632	WORD
TC18M2	PEW 634	WORD
TC19M2	PEW 636	WORD

TC20M2	PEW 638	WORD
TC21M2	PEW 640	WORD
TC22M2	PEW 642	WORD
TC23M2	PEW 644	WORD
TC24M2	PEW 646	WORD
TC25M2	PEW 648	WORD
TC26M2	PEW 650	WORD
TC27M2	PEW 652	WORD
TC28M2	PEW 654	WORD
TC29M2	PEW 656	WORD
TC30M2	PEW 658	WORD
TC31M2	PEW 660	WORD
TC32M2	PEW 662	WORD
TC1C	PEW 664	WORD
TC2C	PEW 666	WORD
TC3C	PEW 668	WORD
TC4C	PEW 670	WORD
TC5C	PEW 672	WORD
TC6C	PEW 674	WORD
TC7C	PEW 676	WORD
TC8C	PEW 678	WORD
RESERVA	PEW 680	WORD
RESERVA	PEW 682	WORD
RESERVA	PEW 684	WORD
RESERVA	PEW 686	WORD
RESERVA	PEW 688	WORD
RESERVA	PEW 690	WORD
RESERVA	PEW 692	WORD

Tabla 1: Entradas Analógicas

3.3.2 ENTRADAS DIGITALES

Mesa1_Fuera1	E0.0	BOOL
Mesa1_Fuera2	E0.1	BOOL
Mesa1_Dentro1	E0.2	BOOL
Mesa1_Dentro2	E0.3	BOOL
Mesa1_Frenado	E0.4	BOOL
RESERVA	E0.5	BOOL
RESERVA	E0.6	BOOL
RESERVA	E0.7	BOOL
Sobrec_Bomba_Va	E1.0	BOOL
Sobrec_Variador_Cam	E1.1	BOOL
Sobrec_Cilindros_Cam	E1.2	BOOL
Sobrec_Motores_Mesa1	E1.3	BOOL
Fallo_Tiristor_Zona1	E1.4	BOOL
Fallo_Tiristor_Zona2	E1.5	BOOL
Fallo_Tiristor_Zona3	E1.6	BOOL
Fallo_Tiristor_Zona4	E1.7	BOOL
Alarma_Refrigeracion	E2.0	BOOL
RESERVA	E2.1	BOOL
RESERVA	E2.2	BOOL
Modulo_de_Emergencias	E2.3	BOOL
Barrera_Fotoe_izquierda	E2.4	BOOL
Barrera_Fotoe_derecha	E2.5	BOOL
Selec_Manual_Aut	E2.6	BOOL
Selec_Permission_Mov_Mes	E2.7	BOOL
Puls_Sacar_Mes	E3.0	BOOL
Puls_Meter_Mes	E3.1	BOOL
Puls_Subir_Cam	E3.2	BOOL
Puls_Bajar_Cam	E3.3	BOOL
Puls_Rearme_Rese	E3.4	BOOL
Sobrec_Motores_Mesa2	E3.5	BOOL

Selec_Mesa1_Mesa2	E3.6	BOOL
Selec_Ilumina_Mesa	E3.7	BOOL
Det_Seg_Camp_1_Arriba	E4.0	BOOL
Det_Seg_Camp_1_Abajo	E4.1	BOOL
Det_Seg_Camp_2_Arriba	E4.2	BOOL
Det_Seg_Camp_2_Abajo	E4.3	BOOL
Det_Seg_Camp_3_Arriba	E4.4	BOOL
Det_Seg_Camp_3_Abajo	E4.5	BOOL
Det_Seg_Camp_4_Arriba	E4.6	BOOL
Det_Seg_Camp_4_Abajo	E4.7	BOOL
Detec_Campana_1_Arriba	E5.0	BOOL
Detec_Campana_1_Abajo	E5.1	BOOL
Detec_Campana_2_Arriba	E5.2	BOOL
Detec_Campana_2_Abajo	E5.3	BOOL
Detec_Campana_3_Arriba	E5.4	BOOL
Detec_Campana_3_Abajo	E5.5	BOOL
Detec_Campana_4_Arriba	E5.6	BOOL
Detec_Campana_4_Abajo	E5.7	BOOL
Mesa2_Fuera1	E6.0	BOOL
Mesa2_Fuera2	E6.1	BOOL
Mesa2_Dentro1	E6.2	BOOL
Mesa2_Dentro2	E6.3	BOOL
Mesa2_Frenado	E6.4	BOOL
RESERVA	E6.5	BOOL
Interferencia_Portico_1	E6.6	BOOL
Interferencia_Portico_2	E6.7	BOOL

Tabla 2: Entradas Digitales

3.3.3 SALIDAS ANALÓGICAS

Salida_Tiristor_Zona_1	PAW 512	WORD
Salida_Tiristor_Zona_2	PAW 514	WORD
Salida_Tiristor_Zona_3	PAW 516	WORD
Salida_Tiristor_Zona_4	PAW 518	WORD
Valvula_Regula_vacio	PAW 520	WORD
RESERVA	PAW 522	WORD
RESERVA	PAW 524	WORD
RESERVA	PAW 526	WORD

Tabla 3: Salidas Analógicas

3.3.4 SALIDAS DIGITALES

Mesa1_Avance_lento	A0.0	BOOL
Mesa1_Avance_rapido	A0.1	BOOL
Mesa1_Retroceso_lento	A0.2	BOOL
Mesa1_Retroceso_rapido	A0.3	BOOL
Mesa1_Iluminacion	A0.4	BOOL
Rele_Marcha_Tristores	A0.5	BOOL
Subir_Campana	A0.6	BOOL
Bajar_Campana	A0.7	BOOL
Iniciar_Refrigeracion	A1.0	BOOL
Marcha_Motor_Vacio	A1.1	BOOL
Contador_Ciclo_Auto	A1.2	BOOL
Mesa2_Iluminacion	A1.3	BOOL
Sirena_de_campo	A1.4	BOOL
Lamp_Barr_Fotoe_Delan	A1.5	BOOL
Lamp_Barr_Fotoe_Delan	A1.6	BOOL
Valv_Rotura_de_Vacio	A1.7	BOOL
Motor_Campana_1	A2.0	BOOL
Motor_Campana_2	A2.1	BOOL
Motor_Campana_3	A2.2	BOOL
Motor_Campana_4	A2.3	BOOL
Valv_Vac_Cerrar_Mesa	A2.4	BOOL
Valv_Vac_Abrir_Mesa	A2.5	BOOL
Mesa2_Avance_lento	A2.6	BOOL
Mesa2_Avance_rapido	A2.7	BOOL
Mesa2_Retroceso_lento	A3.0	BOOL
Mesa2_Retroceso_rapido	A3.1	BOOL
RESERVA	A3.2	BOOL
Baliza_Sobretemp_Naranja	A3.3	BOOL
Baliza_Sobretemp_Roja	A3.4	BOOL
Mesa1 Fuera	A3.5	BOOL
Mesa2 Fuera	A3.6	BOOL

RESERVA	A3.7	BOOL
RESERVA	A4.0	BOOL
RESERVA	A4.1	BOOL
RESERVA	A4.2	BOOL

Tabla 4: Salidas Digitales

4 MÓDULOS DEL PROGRAMA

4.1 BLOQUES DE ORGANIZACIÓN

Estos bloques son módulos específicos que no se llaman desde ningún bloque, sino que es el autómata el que gestiona las llamadas a éstos en función de sus características. En este programa se han usado los siguientes:

4.1.1 OBI

El OBI es el bloque de organización principal, en él se hacen las llamadas a los FC y FB del programa y se puede escribir código, en este caso no hay nada de código en este OB, sólo las llamadas de las funciones, es muy importante el orden en el que se las llama porque en ese orden es como se van a ejecutar:

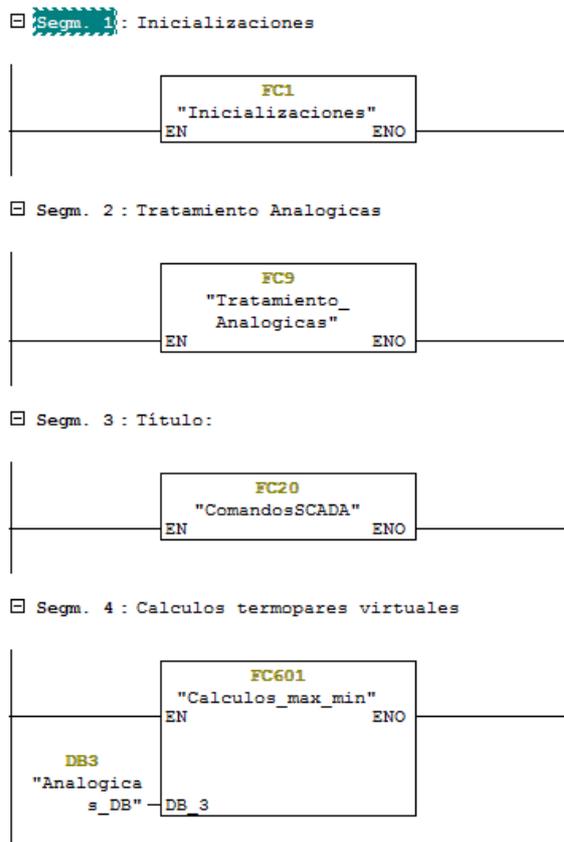
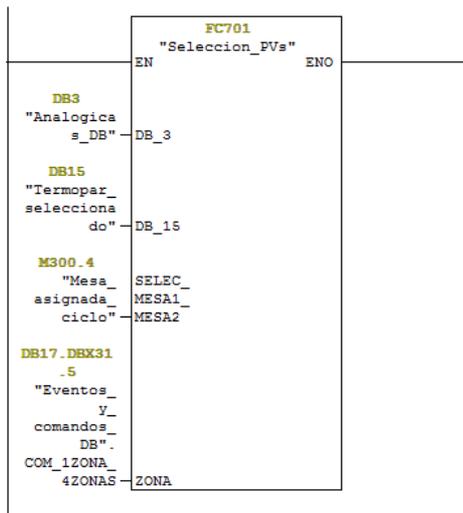
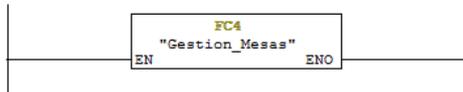


Ilustración 10: OBI Segmentos 1, 2, 3 y 4

Segm. 5 : Seleccion de Variables de proceso



Segm. 6 : Gestion de mesas



Segm. 7 : Gestion de campana

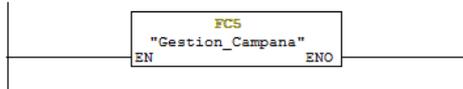
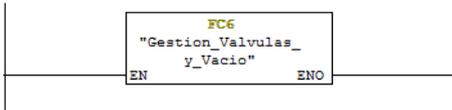
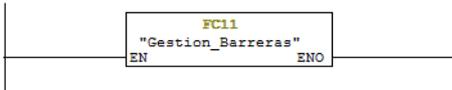


Ilustración 11:OB1 Segmentos 5, 6 y 7

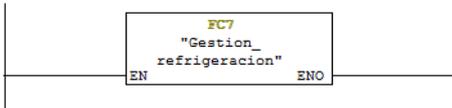
Segm. 8 : Gestion de valvulas y vacio



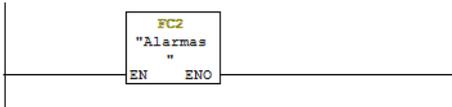
Segm. 9 : Gestion de barreras



Segm. 10 : Gestion de refrigeracion



Segm. 11 : Gestion de alarmas



Segm. 12 : Maquina de estados para ciclo automatico



Ilustración 12:OB1 Segmentos 8, 9, 10, 11 y 12

Segm. 13 : Escritura informacion SCADA



Segm. 14 : Escritura de salidas

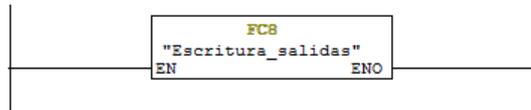


Ilustración 13:OB1 Segmentos 13 y 14

4.1.2 OB35

Es un bloque de organización cíclico, también puede llamar a los FC y FB o ejecutar código, pero tienen la particularidad de que se ejecutan cada cierto tiempo fijo, en este caso se ejecuta cada 100ms. Se ha usado para llamar a los bloques de funciones FB2, que gestiona los programadores, y FB4, que controla el tiempo que se está ejecutando ciclos automáticos, en esta función es necesario que se llame desde un OB cíclico porque necesitamos saber cada cuanto tiempo se ejecuta y que este tiempo sea siempre el mismo.

Segm. 1 : Gestion de programadores Ramp - Soak y PID



Segm. 2 : Gestion del reloj del ciclo auto

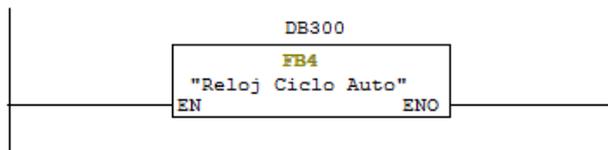


Ilustración 14: OB35 segmentos 1 y 2

4.1.3 OB70

Para errores de redundancias en la periferia. El sistema operativo de la CPU llama al OB70 cuando en un sistema PROFINET IO / sistema maestro DP se produce una pérdida o un retorno de la redundancia en un dispositivo IO / Esclavo DP como mínimo.

4.1.4 OB72

Para errores de redundancias en las CPU. El sistema operativo de la CPU llama al OB72 cuando se produce uno de los siguientes eventos:

- Pérdida de redundancia de las CPU.
- Conmutación del maestro de reserva.
- Error de sincronización.
- Error en un módulo de sincronización.
- Interrupción de la operación "Igualar datos".
- Error de comparación.

4.1.5 OB80

Error de tiempo. El sistema operativo de la CPU llamará al OB80 cuando en la ejecución de un OB surja uno de los siguientes errores: Se ha excedido el tiempo de ciclo, error de acuse en el procesamiento de un OB, adelanto de la hora (salto horario) para el arranque de un OB, regreso a RUN después de CiR.

4.1.6 OB81

Fallo de alimentación. El sistema operativo de la CPU llama al 81 cuando se presenta un evento activado por un fallo en la alimentación normal o de respaldo en tampón.

4.1.7 OB82

Alarma de diagnóstico. Cuando un módulo apto para el diagnóstico, en el que se ha habilitado la alarma de diagnóstico, detecta que ha cambiado su estado de diagnóstico, envía una solicitud de alarma de diagnóstico a la CPU:

- Hay una avería o es necesario realizar el mantenimiento de un componente, o bien ambas cosas (evento entrante).
- Ya no hay avería y ya no es necesario realizar el mantenimiento de un componente (evento saliente).

4.1.8 OB83

Presencia de módulos (Extraer/Insertar). El sistema operativo de la CPU llama al OB83 en los casos siguientes:

- Después de extraer o insertar un módulo configurado.
- Después de haber realizado cambios en los parámetros de un módulo en STEP 7 en el marco de una instalación modificada durante el funcionamiento (proceso CiR) y de haber cargado este módulo en la CPU en modo RUN.

4.1.9 OB84

Avería de CPU. El sistema operativo de la CPU llama al OB84 en los siguientes casos:

- Tras la detección y eliminación de errores de memoria.
- En la configuración de CPUs H hasta la versión V4.5 inclusive: en caso de potencia reducida del acoplamiento de redundancia entre ambas CPUs.
- En WinAC RTX: error en el sistema operativo del PC.

4.1.10 OB85

Error de ejecución del programa. El sistema operativo de la CPU llama al OB85 cuando surge uno de los siguientes eventos:

- Evento de arranque para un OB no cargado (excepto OB80, OB81, OB82B, OB83, OB86).
- Error al acceder el sistema operativo a un bloque.
- Error de acceso de periferia en la actualización de la imagen del proceso correspondiente al sistema (caso de la llamada al OB85 no hay asido suprimida por configuración).

4.1.11 OB86

Fallo del bastidor. El sistema operativo de la CPU llama al OB86 cuando:

- Se detecta el fallo de un aparato de ampliación centralizado.
- Se detecta el fallo de un sistema maestro DP.
- Se detecta el fallo de un equipo de la periferia descentralizada.
- Se ha desactivado un equipo de la periferia descentralizada con la SFC 12 "D_ACT_DP" mediante MODE=4.

- Se ha activado un equipo de la periferia descentralizada con la SFC 12 “D_ACT_DP” mediante MODE=3.
- Se ha detectado el fallo de un sistema PROFINET IO o un equipo PROFINET IO o una parte de los submódulos de un I- device PROFINET.

4.1.12 OB88

Procesamiento interrumpido. El sistema operativo de la CPU llama al OB88 cuando se interrumpe el procesamiento de un bloque del programa.

4.1.13 OB121

Error de programación. El sistema operativo de la CPU llama al OB121 cuando aparece un evento activado por un error durante la ejecución del programa.

4.1.14 OB122

Error de acceso a la periferia. El sistema operativo de la CPU llama al OB122 cuando aparece un error al acceder a datos de un módulo.

4.2 FUNCIONES Y BLOQUES DE FUNCIÓN DEL OB1

FUNCIONES (FC): Las funciones son bloques que pueden ser llamados desde un OB, un FC o un FB y se usan para estructurar el código de tal manera que no esté metido todo en el OB1, sino que se divide el programa en funciones para permitir la reutilización de éstas y para aportar claridad al código. Tienen la particularidad de poder definir variables temporales dentro del bloque, pero no se pueden usar desde fuera y su valor se pierde cada vez que se ejecuta un nuevo ciclo.

BLOQUES DE FUNCIÓN (FB): Los bloques de función son parecidos a los anteriores con la peculiaridad de que tienen variables internas llamadas estáticas, estas variables a diferencia de las temporales guardan su valor de un ciclo a otro.

A continuación, se van a describir los módulos FC y FB que forman el programa, se hará en el orden en que son llamados en el OB1 para así poder entender mejor el funcionamiento de la máquina:

4.2.1 FC1: INICIALIZACIONES

Esta función inicializa valores que necesitaremos más adelante, en los dos primeros segmentos se han creado dos marcas, una que es siempre verdadero y otra que es siempre falso, estas marcas han servido para puentear un segmento o para no ejecutarlo en un momento dado, como por ejemplo en las simulaciones en la oficina. En los dos siguientes segmentos se captura el pulso inicial, es decir, cuando arranca el programa del STEP 7.

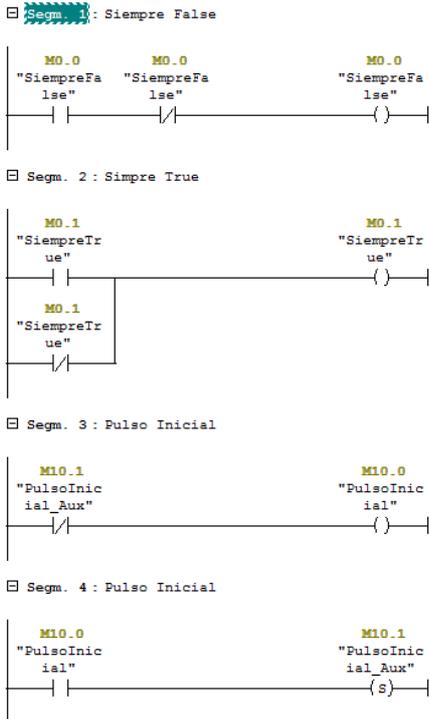
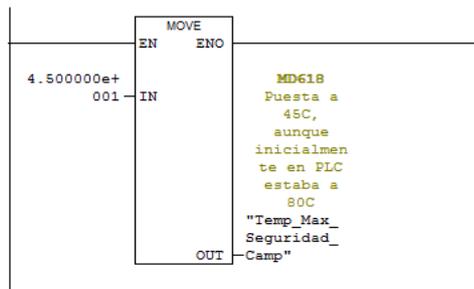


Ilustración 15: FC1 Segmentos 1, 2, 3 y 4

A continuación, le indica al PLC los datos de la campana, como la temperatura máxima a la que se puede subir garantizando la seguridad, su posición actual y la histéresis que tiene la posición de la campana de precalentamiento, esta posición es a la cual empieza la campana a calentarse antes de bajar del todo. También se le indica el valor de presión mínimo que puede haber para asegurar que ya no existe el vacío.

Segm. 5 : Temperatura de seguridad para elevación de campana tras vacío



Segm. 6 : Posicion actual Campana

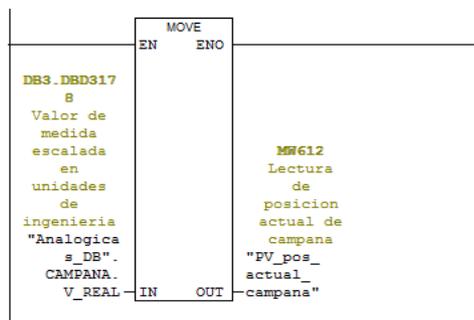
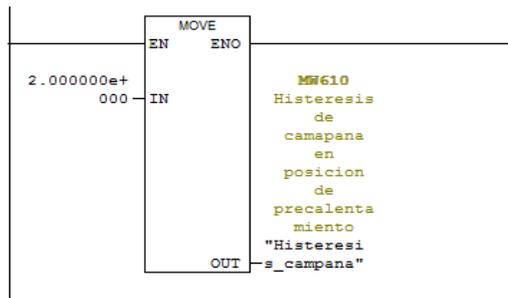


Ilustración 16: FC1 Segmentos 5 y 6

Segm. 7 : Histeris Camapana



Segm. 8 : Consigna sin vacio

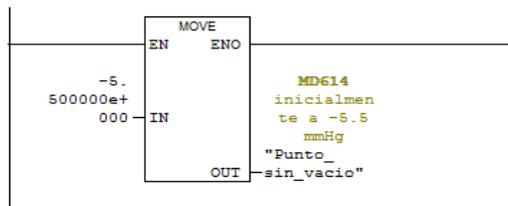


Ilustración 17: Segmentos 7 y 8

Por último, se garantiza una parada segura cuando estamos en modo automático y se le hace un reset, cuando pasamos a modo manual, a todas las señales que pueden haber sido activadas durante ese modo, siendo esta la forma más adecuada de abortar un ciclo automático sin necesidad de rearmar nada.

Segm. 9 : RESET CICLO AUTO

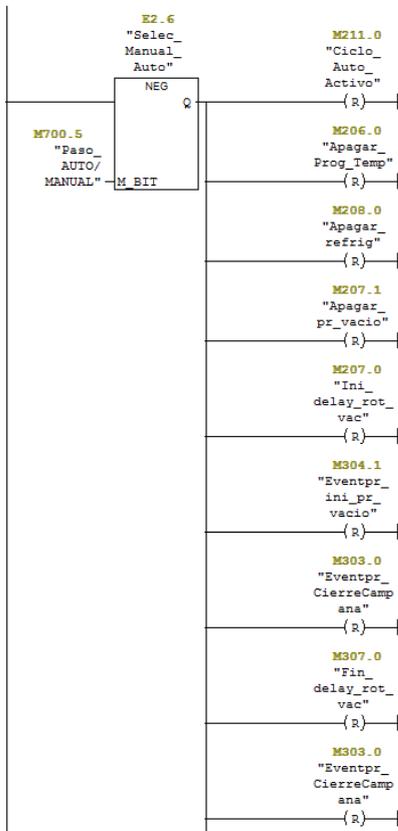


Ilustración 18: FC1 Segmento 9_1

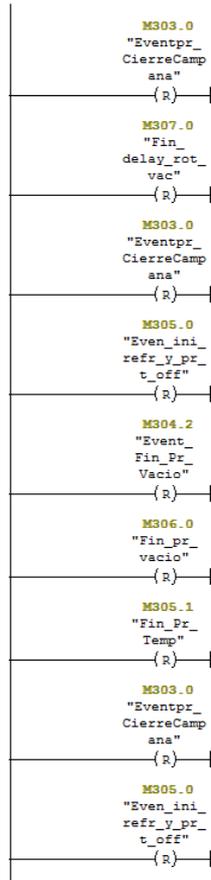


Ilustración 19: FC1 Segmento 9_2

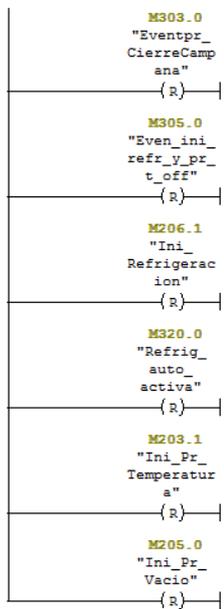


Ilustración 20: FC1 Segmento 9_3

4.2.3 FC9: TRATAMIENTO ANALÓGICAS

Esta función se ha realizado en AWL, en ella se recorren los valores de las entradas analógicas y teniendo en cuenta los rangos máximos y mínimos, la ganancia y el offset de cada tipo de sensor se escalan y normalizan dando como resultado su valor real, para así poder trabajar con esos valores a lo largo del programa de una forma más cómoda.

```
FC9 : Tratamiento_Analogicas
ESCALADO Y CALIBRACION DE ENTRADAS ANALOGICAS
Segm. 1: TERMOPARES
Tomamos una precision de 2 decimales en punto fijo, por lo que dividimos entre
100 para pasar a real.
//LECTURA PARA LOS 72 TERMOPARES

// PUNTERO A LA DB
AUF "Analogicas_DB"          DB3
LAR1 P#0.0
// PUNTERO DE ENTRADAS ANALOGICAS
LAR2 P#536.0

//BUCLE DE 72 TERMOPARES
L 72
FOR1: T #INDICE              #INDICE
//LECTURA DE TERMOPAR
L PEW [AR2,P#0.0]
T #TEMP_LECTURA            #TEMP_LECTURA
//T "Analogicas_DB".TC1M1.V_ANALOGICO
T DEW [AR1,P#8.0]

//ESCALADO, A REAL Y DIVISION POR 100
ITD
DTR
L 1.000000e+002
/R
// T #TEMP_VALOR

// AJUSTE OFFSET Y GANANCIA
//"Analogicas_DB".TC1M1.GANANCIA
L DEB [AR1,P#18.0]
*R
//"Analogicas_DB".TC1M1.OFFSET
L DEB [AR1,P#14.0]
+R
//"Analogicas_DB".TC1M1.V_REAL
T DEB [AR1,P#10.0]

// FALLO LECTURA SEÑAL ANALOGICA FUERA DE RANGO
// TERMOPAR DESCONECTADO O ROTURA DE HILO
```

Ilustración 21: FC9 en AWL_1

```

U(
L   #TEMP_LECTURA          #TEMP_LECTURA
L   32511
>I
)
O
U(
L   #TEMP_LECTURA          #TEMP_LECTURA
L   -32512
<I
)
//=   "Analogicas_DB".TC1M1.LECTURA_INCORRECTA
=     DBX [AR1,P#22.1]

//INCREMENTO DE PUNTERO A SIGUIENTE REGISTRO DE DB
+AR1 P#24.0
//INCREMENTO DE PUNTERO A SIGUIENTE TERMOPAR
+AR2 P#2.0

L   #INDICE                  #INDICE
LOOP FOR1

```

☐ **Segm. 2 : Sensores Infrarrojos**

El rango de los sensores infrarrojos montados inicialmente corresponde a 0 - 150 C.

```

// PUNTERO A LA DB

LAR1 P#2952.0
// PUNTERO DE ENTRADAS ANALOGICAS
LAR2 P#520.0

//BUCLE DE 8 Transmisores infrarrojo
L   8
FOR2: T   #INDICE          #INDICE

//LECTURA DE SENSOR
L   PEW [AR2,P#0.0]
T   #TEMP_LECTURA        #TEMP_LECTURA
//T   "Analogicas_DB".TC1M1.V_ANALOGICO
T   DBW [AR1,P#8.0]

//Cargamos limites
L   DED [AR1,P#0.0]
T   #Lim_alto             #Lim_alto

L   DED [AR1,P#4.0]
T   #Lim_bajo             #Lim_bajo
//Hacemos el escalado
CALL "SCALE"              FC105          -- Scaling Values
IN   :=#TEMP_LECTURA     #TEMP_LECTURA

```

Ilustración 22: FC9 en AWL_2

```

HI_LIM :=#Lim_alto           #Lim_alto
LO_LIM :=#Lim_bajo           #Lim_bajo
BIPOLAR:=FALSE
RET_VAL:=#Retval             #Retval
OUT     :=#Salida             #Salida

// AJUSTE OFFSET Y GANANCIA

L     #Salida                 #Salida

// AJUSTE OFFSET Y GANANCIA
//"Analogicas_DB".TC1M1.GANANCIA
L     DBD [AR1,P#18.0]
*R
//"Analogicas_DB".TC1M1.OFFSET
L     DBD [AR1,P#14.0]
+R

//"Analogicas_DB".TC1M1.V_REAL
T     DBD [AR1,P#10.0]

// FALLO LECTURA SEÑAL ANALOGICA

U(
L     #TEMP_LECTURA         #TEMP_LECTURA
L     27648
>I
)
O
U(
L     #TEMP_LECTURA         #TEMP_LECTURA
L     0
<I
)
//= "Analogicas_DB".TC1M1.LECTURA_INCORRECTA
=     DEX [AR1,P#22.1]
//INCREMENTO DE PUNTERO A SIGUIENTE REGISTRO DE DB
+AR1 P#24.0
//INCREMENTE DE PUNTERO A SIGUIENTE TERMOPAR
+AR2 P#2.0

L     #INDICE                 #INDICE
LOOP FOR2

☐ Segm. 3 : Sensor de Depresion Y Posición Campana
// PUNTERO A LA DB
LAR1 P#3144.0
// PUNTERO DE ENTRADAS ANALOGICAS
LAR2 P#512.0

```

Ilustración 23:FC9 en AWL_3

```

//SENSOR VACIO
//LECTURA DE SENSOR
  L   PEW [AR2,P#0.0]
  T   #TEMP_LECTURA          #TEMP_LECTURA
//T   "Analogicas_DB".TC1M1.V_ANALOGICO
  T   DEW [AR1,P#8.0]

//Cargamos limites
  L   DED [AR1,P#0.0]
  T   #Lim_alto                #Lim_alto

  L   DED [AR1,P#4.0]
  T   #Lim_bajo                #Lim_bajo

//Hacemos el escalado
CALL  "SCALE"                  FC105          -- Scaling Values
  IN   :=#TEMP_LECTURA       #TEMP_LECTURA
  HI_LIM :=#Lim_alto          #Lim_alto
  LO_LIM :=#Lim_bajo          #Lim_bajo
  BIPOLAR:=FALSE
  RET_VAL:=#Retval            #Retval
  OUT   :=#Salida              #Salida

// AJUSTE OFFSET Y GANANCIA

  L   #Salida                  #Salida

// AJUSTE OFFSET Y GANANCIA
//"Analogicas_DB".TC1M1.GANANCIA
  L   DED [AR1,P#18.0]
  *R
//"Analogicas_DB".TC1M1.OFFSET
  L   DED [AR1,P#14.0]
  +R

//"Analogicas_DB".TC1M1.V_REAL
  T   DED [AR1,P#10.0]

// FALLO LECTURA SEÑAL ANALOGICA

  U(
  L   #TEMP_LECTURA          #TEMP_LECTURA
  L   27648
  >I
  )
  O
  U(
  L   #TEMP_LECTURA          #TEMP_LECTURA
  L   0
  <I
  )

```

Ilustración 24:FC9 en AWL_4

```

//=      "Analogicas_DB".TC1M1.LECTURA_INCORRECTA
      =      DBX [AR1,P#22.1]

//SENSOR CAMPANA
//INCREMENTO DE PUNTERO A SIGUIENTE REGISTRO DE DB
      +AR1 P#24.0
//INCREMENTO DE PUNTERO A SIGUIENTE SENSOR
      +AR2 P#2.0
//SENSOR VACIO
//LECTURA DE SENSOR
      L      PEW [AR2,P#0.0]
      T      #TEMP_LECTURA      #TEMP_LECTURA
//T      "Analogicas_DB".TC1M1.V_ANALOGICO
      T      DBW [AR1,P#8.0]

//Cargamos limites
      L      DEB [AR1,P#0.0]
      T      #Lim_alto      #Lim_alto

      L      DEB [AR1,P#4.0]
      T      #Lim_bajo      #Lim_bajo

//Hacemos el escalado
      CALL "SCALE"      FC105      -- Scaling Values
      IN      :=#TEMP_LECTURA      #TEMP_LECTURA
      HI_LIM :=#Lim_alto      #Lim_alto
      LO_LIM :=#Lim_bajo      #Lim_bajo
      BIPOLAR:=FALSE
      RET_VAL:=#Retval      #Retval
      OUT      :=#Salida      #Salida

// AJUSTE OFFSET Y GANANCIA

      L      #Salida      #Salida

// AJUSTE OFFSET Y GANANCIA
//"Analogicas_DB".TC1M1.GANANCIA
      L      DEB [AR1,P#18.0]
      +R
//"Analogicas_DB".TC1M1.OFFSET
      L      DEB [AR1,P#14.0]
      +R

//"Analogicas_DB".TC1M1.V_REAL
      T      DEB [AR1,P#10.0]

// FALLO LECTURA SEÑAL ANALOGICA

      U(
      L      #TEMP_LECTURA      #TEMP_LECTURA
      L      27648
      >I

```

Ilustración 25: FC9 en AWL_5

```

    )
    O
  U(
  L   #TEMP_LECTURA          #TEMP_LECTURA
  L   0
  <I
  )
//=  "Analogicas_DB".TC1M1.LECTURA_INCORRECTA
    =  DEX [AR1,P#22.1]

□ Segm. 4: Sensores Temperatura refrigeradora

// PUNTERO A LA DB

    LAR1 P#3192.0
// PUNTERO DE ENTRADAS ANALOGICAS
    LAR2 P#696.0

//BUCLE DE 4 Sensores de Temperatura Refrigeradora
  L   4
FOR4: T   #INDICE              #INDICE

//LECTURA DE SENSOR
  L   DEW [AR2,P#0.0]
  T   #TEMP_LECTURA          #TEMP_LECTURA
//T   "Analogicas_DB".TC1M1.V_ANALOGICO
  T   DEW [AR1,P#8.0]

//Cargamos limites
  L   DED [AR1,P#0.0]
  T   #Lim_alto              #Lim_alto

  L   DED [AR1,P#4.0]
  T   #Lim_bajo              #Lim_bajo
//Hacemos el escalado
  CALL "SCALE"                FC105          -- Scaling Values
  IN   :=#TEMP_LECTURA      #TEMP_LECTURA
  HI_LIM :=#Lim_alto         #Lim_alto
  LO_LIM :=#Lim_bajo         #Lim_bajo
  BIPOLAR:=FALSE
  RET_VAL:=#Retval          #Retval
  OUT   :=#Salida            #Salida

// AJUSTE OFFSET Y GANANCIA

  L   #Salida                  #Salida

// AJUSTE OFFSET Y GANANCIA
//"Analogicas_DB".TC1M1.GANANCIA
  L   DED [AR1,P#18.0]

```

Ilustración 26: FC1 en AWL_6

```

//"Analogicas_DE".TC1M1.V_REAL
T    DED [AR1,P#10.0]

// FALLO LECTURA SEÑAL ANALOGICA

    U(
    L    #TEMP_LECTURA          #TEMP_LECTURA
    L    27648
    >I
    )
    O
    U(
    L    #TEMP_LECTURA          #TEMP_LECTURA
    L    0
    <I
    )
//=    "Analogicas_DE".TC1M1.LECTURA_INCORRECTA
      =    DEX [AR1,P#22.1]
//INCREMENTO DE PUNTERO A SIGUIENTE REGISTRO DE DB
+AR1  P#24.0
//INCREMENTO DE PUNTERO A SIGUIENTE TERMOPAR
+AR2  P#2.0

    L    #INDICE                #INDICE
    LOOP FOR4


```

Segm. 5 : Sensor Feedback Valvula

```

// PUNTERO A LA DB
LAR1  P#3288.0
// PUNTERO DE ENTRADAS ANALOGICAS
LAR2  P#704.0

//SENSOR VACIO
//LECTURA DE SENSOR
    L    PEW [AR2,P#0.0]
    T    #TEMP_LECTURA          #TEMP_LECTURA
//T    "Analogicas_DE".TC1M1.V_ANALOGICO
    T    DEW [AR1,P#8.0]

//Cargamos limites
    L    DED [AR1,P#0.0]
    T    #Lim_alto              #Lim_alto

    L    DED [AR1,P#4.0]
    T    #Lim_bajo              #Lim_bajo

//Hacemos el escalado
CALL  "SCALE"                  FC105          -- Scaling Values
    IN    :=#TEMP_LECTURA      #TEMP_LECTURA
    HI_LIM :=#Lim_alto          #Lim_alto

```

Ilustración 27: FC1 en AWL_7

```

        LO_LIM :=#Lim_bajo           #Lim_bajo
        BIPOLAR:=FALSE
        RET_VAL:=#Retval             #Retval
        OUT     :=#Salida            #Salida

// AJUSTE OFFSET Y GANANCIA

        L     #Salida                #Salida

// AJUSTE OFFSET Y GANANCIA
//"Analogicas_DB".TC1M1.GANANCIA
        L     DED [A1,P#18.0]
        +R
//"Analogicas_DB".TC1M1.OFFSET
        L     DED [A1,P#14.0]
        +R

//"Analogicas_DB".TC1M1.V_REAL
        T     DED [A1,P#10.0]

// FALLO LECTURA SEÑAL ANALOGICA

        U(
        L     #TEMP_LECTURA         #TEMP_LECTURA
        L     27648
        >I
        )
        O
        U(
        L     #TEMP_LECTURA         #TEMP_LECTURA
        L     0
        <I
        )
//= "Analogicas_DB".TC1M1.LECTURA_INCORRECTA
=     DBX [A1,P#22.1]

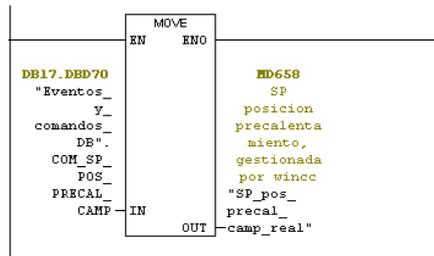
```

Ilustración 28: FC1 en AWL_8

4.2.4 FC20: LECTURA DE INFORMACIÓN DEL SCADA

En esta función se vuelcan las órdenes del SCADA a marcas internas del PLC

Segm. 1: SP posicion precalentamiento campana



Segm. 2: Marcha rotura de vacío



Segm. 3: Marcha bomba de vacío



Ilustración 29: FC20 Segmentos 1, 2 y 3

Segm. 4: Marcha refrigeración



Segm. 5: Anulación posición de precalentamiento



Ilustración 30: FC20 Segmentos 4 y 5

4.2.5 FC601: CALCULO DE TERMOPARES VIRTUALES

Esta función se ha realizado en SCL, en ella se recorre la DB donde se encuentran todos los datos de los termopares, sólo se tienen en cuenta aquellos que se han habilitado, a partir de ellos se calculan los termopares máximos, mínimos y la media de cada zona de ambas mesas y de la campana, los máximos, mínimo y media del total de cada mesa y de la campana y por último los máximos, mínimo y media del total de mesa1 y campana por un lado y mesa 2 y campana por otro. A estos se les ha denominado termopares virtuales, y sirven para poder controlar el programa de temperatura con alguno de ellos en vez de con un termopar concreto. Para la realización de esta función se ha usado el siguiente código:

```
FUNCTION FC601 : VOID
```

```
// Creación de variables temporales
```

```
VAR_TEMP
```

```
ADDRESS_1: INT;
```

```
ADDRESS_3: INT;
```

```
CONT_M1_Z1: INT;
```

```
CONT_M1_Z2: INT;
```

```
CONT_M1_Z3: INT;
```

```
CONT_M1_Z4: INT;
```

```
CONT_M1: INT;
```

```
SUMA_VALORES_M1Z1: REAL;
```

```
SUMA_VALORES_M1Z2: REAL;
```

```
SUMA_VALORES_M1Z3: REAL;
```

```
SUMA_VALORES_M1Z4: REAL;
```

```
SUMA_VALORES_M1: REAL;
```

```
CONT_M2_Z1: INT;
```

```
CONT_M2_Z2: INT;
```

```
CONT_M2_Z3: INT;
```

```
CONT_M2_Z4: INT;
```

```
CONT_M2: INT;
```

```
SUMA_VALORES_M2Z1: REAL;
```

```
SUMA_VALORES_M2Z2: REAL;
```

```
SUMA_VALORES_M2Z3: REAL;
```

```
SUMA_VALORES_M2Z4: REAL;
```

```
SUMA_VALORES_M2: REAL;
```

```
CONT_CAMP_Z1: INT;
```

```
CONT_CAMP_Z2: INT;
```

```
CONT_CAMP_Z3: INT;
```

```
CONT_CAMP_Z4: INT;
```

```
CONT_CAMP: INT;
```

```
CONT_CAMP_C: INT;
```

```
CONT_CAMP_I: INT;
```

```
SUMA_VALORES_CAMPZ1: REAL;
```

SUMA_VALORES_CAMPZ2: REAL;
SUMA_VALORES_CAMPZ3: REAL;
SUMA_VALORES_CAMPZ4: REAL;
SUMA_VALORES_CAMP: REAL;
SUMA_VALORES_CAMP_C: REAL;
SUMA_VALORES_CAMP_I: REAL;
SUMA_VALORES_CAMPYM1: REAL;
SUMA_VALORES_CAMPYM2: REAL;
CONT_CAMPYM1: REAL;
CONT_CAMPYM2: REAL;

MAX_M1_Z1: REAL;
MAX_M1_Z2: REAL;
MAX_M1_Z3: REAL;
MAX_M1_Z4: REAL;
MAX_M1: REAL;
MAX_M2_Z1: REAL;
MAX_M2_Z2: REAL;
MAX_M2_Z3: REAL;
MAX_M2_Z4: REAL;
MAX_M2: REAL;
MAX_CAMP_Z1: REAL;
MAX_CAMP_Z2: REAL;
MAX_CAMP_Z3: REAL;
MAX_CAMP_Z4: REAL;
MAX_CAMP: REAL;

MIN_M1_Z1: REAL;
MIN_M1_Z2: REAL;
MIN_M1_Z3: REAL;
MIN_M1_Z4: REAL;
MIN_M1: REAL;
MIN_M2_Z1: REAL;
MIN_M2_Z2: REAL;
MIN_M2_Z3: REAL;
MIN_M2_Z4: REAL;
MIN_M2: REAL;
MIN_CAMP_Z1: REAL;
MIN_CAMP_Z2: REAL;
MIN_CAMP_Z3: REAL;
MIN_CAMP_Z4: REAL;
MIN_CAMP: REAL;

MED_M1_Z1: REAL;
MED_M1_Z2: REAL;
MED_M1_Z3: REAL;
MED_M1_Z4: REAL;

MED_M1: REAL;
MED_M2_Z1: REAL;
MED_M2_Z2: REAL;
MED_M2_Z3: REAL;
MED_M2_Z4: REAL;
MED_M2: REAL;
MED_CAMP_Z1: REAL;
MED_CAMP_Z2: REAL;
MED_CAMP_Z3: REAL;
MED_CAMP_Z4: REAL;
MED_CAMP: REAL;
MED_CAMP_C: REAL;
MED_CAMP_I: REAL;

MAX_M1CAMP: REAL;
MAX_M2CAMP: REAL;
MIN_M1CAMP: REAL;
MIN_M2CAMP: REAL;
MED_M1CAMP: REAL;
MED_M2CAMP: REAL;
Q_DB1: INT;
Q_DB3: INT;

Q_MAXIMO : REAL;
Q_MINIMO : REAL;
Q_MEDIA : REAL;
HABILITAR_TER_VIRTUAL: BOOL;

HABILITAR_M1_Z1: BOOL;
HABILITAR_M1_Z2: BOOL;
HABILITAR_M1_Z3: BOOL;
HABILITAR_M1_Z4: BOOL;
HABILITAR_M2_Z1: BOOL;
HABILITAR_M2_Z2: BOOL;
HABILITAR_M2_Z3: BOOL;
HABILITAR_M2_Z4: BOOL;
HABILITAR_CAMP_Z1: BOOL;
HABILITAR_CAMP_Z2: BOOL;
HABILITAR_CAMP_Z3: BOOL;
HABILITAR_CAMP_Z4: BOOL;
HABILITAR_M1: BOOL;
HABILITAR_M2: BOOL;
HABILITAR_CAMP: BOOL;
HABILITAR_M1CAMP: BOOL;

HABILITAR_M2CAMP: **BOOL**;

NING_HABILITADO_CAMPZ1: **BOOL**;

NING_HABILITADO_CAMPZ2: **BOOL**;

NING_HABILITADO_CAMPZ3: **BOOL**;

NING_HABILITADO_CAMPZ4: **BOOL**;

VALOR_REAL: **REAL**;

HABILITADO: **BOOL**;

LECTURA_INCORRECTA: **BOOL**;

END_VAR

// Creación de variables de entrada

VAR_INPUT

DB_3: **BLOCK_DB**;

END_VAR

//Inicialización de variables

MIN_M1_Z1:=1000;

MIN_M1_Z2:=1000;

MIN_M1_Z3:=1000;

MIN_M1_Z4:=1000;

MIN_M1:=1000;

MIN_M2_Z1:=1000;

MIN_M2_Z2:=1000;

MIN_M2_Z3:=1000;

MIN_M2_Z4:=1000;

MIN_M2:=1000;

MIN_CAMP_Z1:=1000;

MIN_CAMP_Z2:=1000;

MIN_CAMP_Z3:=1000;

MIN_CAMP_Z4:=1000;

MIN_CAMP:=1000;

MIN_M1CAMP:= 1000;

MIN_M2CAMP:= 1000;

CONT_M1_Z1:=0;

CONT_M1_Z2:=0;

CONT_M1_Z3:=0;

CONT_M1_Z4:=0;

CONT_M2_Z1:=0;
CONT_M2_Z2:=0;
CONT_M2_Z3:=0;
CONT_M2_Z4:=0;
CONT_CAMP_Z1:=0;
CONT_CAMP_Z2:=0;
CONT_CAMP_Z3:=0;
CONT_CAMP_Z4:=0;
CONT_CAMP:=0;
CONT_CAMP_C:=0;
CONT_CAMP_I:=0;
CONT_CAMPYM1:= 0;
CONT_CAMPYM2:= 0;

SUMA_VALORES_M1Z1:= 0;
SUMA_VALORES_M1Z2:= 0;
SUMA_VALORES_M1Z3:= 0;
SUMA_VALORES_M1Z4:= 0;
SUMA_VALORES_M1:= 0;

SUMA_VALORES_M2Z1:= 0;
SUMA_VALORES_M2Z2:= 0;
SUMA_VALORES_M2Z3:= 0;
SUMA_VALORES_M2Z4:= 0;
SUMA_VALORES_M2:= 0;

SUMA_VALORES_CAMPZ1:= 0;
SUMA_VALORES_CAMPZ2:= 0;
SUMA_VALORES_CAMPZ3:= 0;
SUMA_VALORES_CAMPZ4:= 0;
SUMA_VALORES_CAMP:= 0;
SUMA_VALORES_CAMP_C:= 0;
SUMA_VALORES_CAMP_I:= 0;

SUMA_VALORES_CAMPYM1:= 0;
SUMA_VALORES_CAMPYM2:= 0;

MAX_M1_Z1:= 0;
MAX_M1_Z2:= 0;
MAX_M1_Z3:= 0;
MAX_M1_Z4:= 0;
MAX_M1:= 0;
MAX_M2_Z1:= 0;
MAX_M2_Z2:= 0;
MAX_M2_Z3:= 0;
MAX_M2_Z4:= 0;

MAX_M2:= 0;
MAX_CAMP_Z1:= 0;
MAX_CAMP_Z2:= 0;
MAX_CAMP_Z3:= 0;
MAX_CAMP_Z4:= 0;
MAX_CAMP:= 0;
MED_M1_Z1:= 0;
MED_M1_Z2:= 0;
MED_M1_Z3:= 0;
MED_M1_Z4:= 0;
MED_M1:= 0;
MED_M2_Z1:= 0;
MED_M2_Z2:= 0;
MED_M2_Z3:= 0;
MED_M2_Z4:= 0;
MED_M2:= 0;
MED_CAMP_Z1:= 0;
MED_CAMP_Z2:= 0;
MED_CAMP_Z3:= 0;
MED_CAMP_Z4:= 0;
MED_CAMP:= 0;
MED_CAMP_C:= 0;
MED_CAMP_I:= 0;
MAX_M1CAMP:= 0;
MAX_M2CAMP:= 0;
MED_M1CAMP:= 0;
MED_M2CAMP:= 0;

Q_MAXIMO := 0;
Q_MINIMO := 0;
Q_MEDIA := 0;
HABILITAR_TER_VIRTUAL:= FALSE;

HABILITAR_M1_Z1:= FALSE;
HABILITAR_M1_Z2:= FALSE;
HABILITAR_M1_Z3:= FALSE;
HABILITAR_M1_Z4:= FALSE;
HABILITAR_M2_Z1:= FALSE;
HABILITAR_M2_Z2:= FALSE;
HABILITAR_M2_Z3:= FALSE;
HABILITAR_M2_Z4:= FALSE;
HABILITAR_CAMP_Z1:= FALSE;
HABILITAR_CAMP_Z2:= FALSE;
HABILITAR_CAMP_Z3:= FALSE;
HABILITAR_CAMP_Z4:= FALSE;

```
HABILITAR_M1:= FALSE;
HABILITAR_M2:= FALSE;
HABILITAR_CAMP:= FALSE;
HABILITAR_M1CAMP:= FALSE;
HABILITAR_M2CAMP:= FALSE;
```

```
NING_HABILITADO_CAMPZ1:= FALSE;
NING_HABILITADO_CAMPZ2:= FALSE;
NING_HABILITADO_CAMPZ3:= FALSE;
NING_HABILITADO_CAMPZ4:= FALSE;
```

```
VALOR_REAL:= 0.0;
HABILITADO:= FALSE;
LECTURA_INCORRECTA:= FALSE;
```

//Recorremos las direcciones del bloque de datos, cada termopar tiene asignados 24 palabras.

```
FOR ADDRESS_3:= 0 TO 1704 BY 24 DO
```

//Se asigna el valor real del termopar a la variable VALOR_REAL, en la variable HABILITADO sabremos si se ha habilitado para el ciclo o no, y LECTURA INCORRECTA si el termopar está roto.

```
VALOR_REAL:=DWORD_TO_REAL(WORD_TO_BLOCK_DB(INT_TO_WORD(3)).DD[ADDRESS_3 + 10]);
HABILITADO:=DB_3.DX[ADDRESS_3+ 22,0];
LECTURA_INCORRECTA:=DB_3.DX[ADDRESS_3+ 22,1];
```

//Para el calculo de los termopares virtuales sólo se tendrán en cuenta los termopares habilitados. Según la dirección en la que se encuentre en la debe sabremos a que mesa o campana pertenece y a qué zona.

```
IF(HABILITADO=1)THEN
```

```
IF(ADDRESS_3<768)THEN //MESA1
```

```
IF (ADDRESS_3<190)THEN //MESA1 ZONA1
```

```
CONT_M1_Z1:=(CONT_M1_Z1+1); //Aumentamos el contador y sumamos los habilitados de la zona
SUMA_VALORES_M1Z1:=(SUMA_VALORES_M1Z1+VALOR_REAL);
CONT_M1:=(CONT_M1+1); //Aumentamos el contador y sumamos los habilitados de su mesa
SUMA_VALORES_M1:=(SUMA_VALORES_M1+VALOR_REAL);
```

//Si el valor del termopar es mayor que el máximo lo asignamos como máximo

```
IF(VALOR_REAL>MAX_M1_Z1)THEN
MAX_M1_Z1:=VALOR_REAL;
END_IF;
```

//Si el valor del termopar es menor que el mínimo lo asignamos como mínimo

IF(VALOR_REAL<MIN_M1_Z1)**THEN**

MIN_M1_Z1:=VALOR_REAL;

END_IF;

ELSIF (ADDRESS_3<384)**THEN** //MESA1 ZONA2

CONT_M1_Z2:=(CONT_M1_Z2+1);

SUMA_VALORES_M1Z2:=(SUMA_VALORES_M1Z2+VALOR_REAL);

CONT_M1:=(CONT_M1+1);

SUMA_VALORES_M1:=(SUMA_VALORES_M1+VALOR_REAL);

IF(VALOR_REAL>MAX_M1_Z2)**THEN**

MAX_M1_Z2:=VALOR_REAL;

END_IF;

IF(VALOR_REAL<MIN_M1_Z2)**THEN**

MIN_M1_Z2:=VALOR_REAL;

END_IF;

ELSIF (ADDRESS_3<576)**THEN** //MESA1 ZONA3

CONT_M1_Z3:=(CONT_M1_Z3+1);

SUMA_VALORES_M1Z3:=(SUMA_VALORES_M1Z3+VALOR_REAL);

CONT_M1:=(CONT_M1+1);

SUMA_VALORES_M1:=(SUMA_VALORES_M1+VALOR_REAL);

IF(VALOR_REAL>MAX_M1_Z3)**THEN**

MAX_M1_Z3:=VALOR_REAL;

END_IF;

IF(VALOR_REAL<MIN_M1_Z3)**THEN**

MIN_M1_Z3:=VALOR_REAL;

END_IF;

ELSE //MESA1 ZONA4

CONT_M1_Z4:=(CONT_M1_Z4+1);

SUMA_VALORES_M1Z4:=(SUMA_VALORES_M1Z4+VALOR_REAL);

CONT_M1:=(CONT_M1+1);

SUMA_VALORES_M1:=(SUMA_VALORES_M1+VALOR_REAL);

IF(VALOR_REAL>MAX_M1_Z4)**THEN**

MAX_M1_Z4:=VALOR_REAL;

```

END_IF;

IF (VALOR_REAL < MIN_M1_Z4) THEN
    MIN_M1_Z4 := VALOR_REAL;
END_IF;

END_IF;

//Repetimos el cálculo para el total de la mesa

IF (VALOR_REAL > MAX_M1) THEN
    MAX_M1 := VALOR_REAL;
END_IF;

IF (VALOR_REAL < MIN_M1) THEN
    MIN_M1 := VALOR_REAL;
END_IF;

ELSIF (ADDRESS_3 < 1536) THEN //MESA2

    IF (ADDRESS_3 < 960) THEN //MESA2 ZONA1
        CONT_M2_Z1 := (CONT_M2_Z1 + 1);
        SUMA_VALORES_M2Z1 := (SUMA_VALORES_M2Z1 + VALOR_REAL);
        CONT_M2 := (CONT_M2 + 1);
        SUMA_VALORES_M2 := (SUMA_VALORES_M2 + VALOR_REAL);

        IF (VALOR_REAL > MAX_M2_Z1) THEN
            MAX_M2_Z1 := VALOR_REAL;
        END_IF;

        IF (VALOR_REAL < MIN_M2_Z1) THEN
            MIN_M2_Z1 := VALOR_REAL;
        END_IF;

    ELSIF (ADDRESS_3 < 1152) THEN //MESA2 ZONA2

        CONT_M2_Z2 := (CONT_M2_Z2 + 1);
        SUMA_VALORES_M2Z2 := (SUMA_VALORES_M2Z2 + VALOR_REAL);
        CONT_M2 := (CONT_M2 + 1);
        SUMA_VALORES_M2 := (SUMA_VALORES_M2 + VALOR_REAL);

        IF (VALOR_REAL > MAX_M2_Z2) THEN
            MAX_M2_Z2 := VALOR_REAL;
        END_IF;

```

IF(VALOR_REAL<MIN_M2_Z2)**THEN**

MIN_M2_Z2:=VALOR_REAL;

END_IF;

ELSIF (ADDRESS_3<1344)**THEN** //MESA2 ZONA3

CONT_M2_Z3:=(CONT_M2_Z3+1);

SUMA_VALORES_M2Z3:=(SUMA_VALORES_M2Z3+VALOR_REAL);

CONT_M2:=(CONT_M2+1);

SUMA_VALORES_M2:=(SUMA_VALORES_M2+VALOR_REAL);

IF(VALOR_REAL>MAX_M2_Z3)**THEN**

MAX_M2_Z3:=VALOR_REAL;

END_IF;

IF(VALOR_REAL<MIN_M2_Z3)**THEN**

MIN_M2_Z3:=VALOR_REAL;

END_IF;

ELSE //MESA2 ZONA4

CONT_M2_Z4:=(CONT_M2_Z4+1);

SUMA_VALORES_M2Z4:=(SUMA_VALORES_M2Z4+VALOR_REAL);

CONT_M2:=(CONT_M2+1);

SUMA_VALORES_M2:=(SUMA_VALORES_M2+VALOR_REAL);

IF(VALOR_REAL>MAX_M2_Z4)**THEN**

MAX_M2_Z4:=VALOR_REAL;

END_IF;

IF(VALOR_REAL<MIN_M2_Z4)**THEN**

MIN_M2_Z4:=VALOR_REAL;

END_IF;

END_IF;

IF(VALOR_REAL>MAX_M2)**THEN**

MAX_M2:=VALOR_REAL;

END_IF;

IF(VALOR_REAL<MIN_M2)**THEN**

MIN_M2:=VALOR_REAL;

END_IF;

ELSE

IF (ADDRESS_3<1584)**THEN** //CAMPANA ZONA1

CONT_CAMP_Z1:=(CONT_CAMP_Z1+1);
SUMA_VALORES_CAMPZ1:=(SUMA_VALORES_CAMPZ1+VALOR_REAL);
CONT_CAMP:=(CONT_CAMP+1);
SUMA_VALORES_CAMP:=(SUMA_VALORES_CAMP+VALOR_REAL);
CONT_CAMP_C:=(CONT_CAMP_C+1);
SUMA_VALORES_CAMP_C:=(SUMA_VALORES_CAMP_C+VALOR_REAL);

IF(VALOR_REAL>MAX_CAMP_Z1)**THEN**

MAX_CAMP_Z1:=VALOR_REAL;

END_IF;

IF(VALOR_REAL<MIN_CAMP_Z1)**THEN**

MIN_CAMP_Z1:=VALOR_REAL;

END_IF;

ELSIF (ADDRESS_3<1632)**THEN** //CAMPANA ZONA2

CONT_CAMP_Z2:=(CONT_CAMP_Z2+1);
SUMA_VALORES_CAMPZ2:=(SUMA_VALORES_CAMPZ2+VALOR_REAL);
CONT_CAMP:=(CONT_CAMP+1);
SUMA_VALORES_CAMP:=(SUMA_VALORES_CAMP+VALOR_REAL);
CONT_CAMP_C:=(CONT_CAMP_C+1);
SUMA_VALORES_CAMP_C:=(SUMA_VALORES_CAMP_C+VALOR_REAL);

IF(VALOR_REAL>MAX_CAMP_Z2)**THEN**

MAX_CAMP_Z2:=VALOR_REAL;

END_IF;

IF(VALOR_REAL<MIN_CAMP_Z2)**THEN**

MIN_CAMP_Z2:=VALOR_REAL;

END_IF;

ELSIF (ADDRESS_3<1680)**THEN** //CAMPANA ZONA3

CONT_CAMP_Z3:=(CONT_CAMP_Z3+1);
SUMA_VALORES_CAMPZ3:=(SUMA_VALORES_CAMPZ3+VALOR_REAL);
CONT_CAMP:=(CONT_CAMP+1);
SUMA_VALORES_CAMP:=(SUMA_VALORES_CAMP+VALOR_REAL);
CONT_CAMP_C:=(CONT_CAMP_C+1);
SUMA_VALORES_CAMP_C:=(SUMA_VALORES_CAMP_C+VALOR_REAL);

IF(VALOR_REAL>MAX_CAMP_Z3)**THEN**

MAX_CAMP_Z3:=VALOR_REAL;

END_IF;

```
IF (VALOR_REAL < MIN_CAMP_Z3) THEN
```

```
    MIN_CAMP_Z3 := VALOR_REAL;
```

```
END_IF;
```

```
ELSE //CAMPANA ZONA4
```

```
    CONT_CAMP_Z4 := (CONT_CAMP_Z4 + 1);
```

```
    SUMA_VALORES_CAMPZ4 := (SUMA_VALORES_CAMPZ4 + VALOR_REAL);
```

```
    CONT_CAMP := (CONT_CAMP + 1);
```

```
    SUMA_VALORES_CAMP := (SUMA_VALORES_CAMP + VALOR_REAL);
```

```
    CONT_CAMP_C := (CONT_CAMP_C + 1);
```

```
    SUMA_VALORES_CAMP_C := (SUMA_VALORES_CAMP_C + VALOR_REAL);
```

```
IF (VALOR_REAL > MAX_CAMP_Z4) THEN
```

```
    MAX_CAMP_Z4 := VALOR_REAL;
```

```
END_IF;
```

```
IF (VALOR_REAL < MIN_CAMP_Z4) THEN
```

```
    MIN_CAMP_Z4 := VALOR_REAL;
```

```
END_IF;
```

```
END_IF;
```

```
IF (VALOR_REAL > MAX_CAMP) THEN
```

```
    MAX_CAMP := VALOR_REAL;
```

```
END_IF;
```

```
IF (VALOR_REAL < MIN_CAMP) THEN
```

```
    MIN_CAMP := VALOR_REAL;
```

```
END_IF;
```

```
END_IF;
```

```
END_IF;
```

```
END_FOR;
```

```
////INICIO CODIGO TERMOPARES INFRARROJOS
```

```
//Estos termopares están en separados de los otros en la DB, por eso se hace en un FOR distinto
```

```
FOR ADDRESS_3 := 2952 TO 3120 BY 24 DO
```

```
VALOR_REAL:=DWORD_TO_REAL(WORD_TO_BLOCK_DB(INT_TO_WORD(3)),DD[ADDRESS_3 + 10]);
HABILITADO:=DB_3.DX[ADDRESS_3 + 22,0];
LECTURA_INCORRECTA:=DB_3.DX[ADDRESS_3 + 22,1];
```

```
IF(HABILITADO=1)THEN
```

```
IF (ADDRESS_3<3000)THEN //CAMPANA ZONA1
```

```
CONT_CAMP_Z1:=(CONT_CAMP_Z1+1);
SUMA_VALORES_CAMPZ1:=(SUMA_VALORES_CAMPZ1+VALOR_REAL);
CONT_CAMP:=(CONT_CAMP+1);
SUMA_VALORES_CAMP:=(SUMA_VALORES_CAMP+VALOR_REAL);
CONT_CAMP_I:=(CONT_CAMP_I+1);
SUMA_VALORES_CAMP_I:=(SUMA_VALORES_CAMP_I+VALOR_REAL);
```

```
IF(VALOR_REAL>MAX_CAMP_Z1)THEN
```

```
MAX_CAMP_Z1:=VALOR_REAL;
```

```
END_IF;
```

```
IF(VALOR_REAL<MIN_CAMP_Z1)THEN
```

```
MIN_CAMP_Z1:=VALOR_REAL;
```

```
END_IF;
```

```
ELSIF (ADDRESS_3<3048)THEN //CAMPANA ZONA2
```

```
CONT_CAMP_Z2:=(CONT_CAMP_Z2+1);
SUMA_VALORES_CAMPZ2:=(SUMA_VALORES_CAMPZ2+VALOR_REAL);
CONT_CAMP:=(CONT_CAMP+1);
SUMA_VALORES_CAMP:=(SUMA_VALORES_CAMP+VALOR_REAL);
CONT_CAMP_I:=(CONT_CAMP_I+1);
SUMA_VALORES_CAMP_I:=(SUMA_VALORES_CAMP_I+VALOR_REAL);
```

```
IF(VALOR_REAL>MAX_CAMP_Z2)THEN
```

```
MAX_CAMP_Z2:=VALOR_REAL;
```

```
END_IF;
```

```
IF(VALOR_REAL<MIN_CAMP_Z2)THEN
```

```
MIN_CAMP_Z2:=VALOR_REAL;
```

```
END_IF;
```

```
ELSIF (ADDRESS_3<3096)THEN //CAMPANA ZONA3
```

```
CONT_CAMP_Z3:=(CONT_CAMP_Z3+1);
SUMA_VALORES_CAMPZ3:=(SUMA_VALORES_CAMPZ3+VALOR_REAL);
CONT_CAMP:=(CONT_CAMP+1);
SUMA_VALORES_CAMP:=(SUMA_VALORES_CAMP+VALOR_REAL);
CONT_CAMP_I:=(CONT_CAMP_I+1);
```

```
SUMA_VALORES_CAMP_I:=(SUMA_VALORES_CAMP_I+VALOR_REAL);
```

```
IF(VALOR_REAL>MAX_CAMP_Z3)THEN  
    MAX_CAMP_Z3:=VALOR_REAL;  
END_IF;
```

```
IF(VALOR_REAL<MIN_CAMP_Z3)THEN  
    MIN_CAMP_Z3:=VALOR_REAL;  
END_IF;
```

```
ELSE //CAMPANA ZONA4
```

```
CONT_CAMP_Z4:=(CONT_CAMP_Z4+1);  
SUMA_VALORES_CAMPZ4:=(SUMA_VALORES_CAMPZ4+VALOR_REAL);  
CONT_CAMP:=(CONT_CAMP+1);  
SUMA_VALORES_CAMP:=(SUMA_VALORES_CAMP+VALOR_REAL);  
CONT_CAMP_I:=(CONT_CAMP_I+1);  
SUMA_VALORES_CAMP_I:=(SUMA_VALORES_CAMP_I+VALOR_REAL);
```

```
IF(VALOR_REAL>MAX_CAMP_Z4)THEN  
    MAX_CAMP_Z4:=VALOR_REAL;  
END_IF;
```

```
IF(VALOR_REAL<MIN_CAMP_Z4)THEN  
    MIN_CAMP_Z4:=VALOR_REAL;  
END_IF;
```

```
END_IF;
```

```
IF(VALOR_REAL>MAX_CAMP)THEN  
    MAX_CAMP:=VALOR_REAL;  
END_IF;
```

```
IF(VALOR_REAL<MIN_CAMP)THEN  
    MIN_CAMP:=VALOR_REAL;  
END_IF;
```

```
END_IF;
```

```
END_FOR;
```

```
/// FIN CODIGO TERMOPARES INFRARROJOS
```

//Si no hay ningún termopar habilitado en alguna zona, se pone su media a cero y se deshabilita esa zona para que posteriormente no se pueda usar ese termopar virtual para el control. En cambio, si había alguno se calcula su media y se habilita la zona.

```
IF (CONT_M1_Z1=0) THEN
```

MED_M1_Z1:=0;
HABILITAR_M1_Z1:=0;

ELSE

MED_M1_Z1:=(SUMA_VALORES_M1Z1/CONT_M1_Z1);
HABILITAR_M1_Z1:=1;

END_IF;

IF (CONT_M1_Z2=0) THEN

MED_M1_Z2:=0;
HABILITAR_M1_Z2:=0;

ELSE

MED_M1_Z2:=(SUMA_VALORES_M1Z2/CONT_M1_Z2);
HABILITAR_M1_Z2:=1;

END_IF;

IF (CONT_M1_Z3=0) THEN

MED_M1_Z3:=0;
HABILITAR_M1_Z3:=0;

ELSE

MED_M1_Z3:=(SUMA_VALORES_M1Z3/CONT_M1_Z3);
HABILITAR_M1_Z3:=1;

END_IF;

IF (CONT_M1_Z4=0) THEN

MED_M1_Z4:=0;
HABILITAR_M1_Z4:=0;

ELSE

MED_M1_Z4:=(SUMA_VALORES_M1Z4/CONT_M1_Z4);
HABILITAR_M1_Z4:=1;

END_IF;

IF (CONT_M1=0) THEN

MED_M1:=0;
HABILITAR_M1:=0;

ELSE

MED_M1:=(SUMA_VALORES_M1/CONT_M1);
HABILITAR_M1:=1;

END_IF;

IF (CONT_M2_Z1=0) THEN

MED_M2_Z1:=0;
HABILITAR_M2_Z1:=0;

ELSE

MED_M2_Z1:=(SUMA_VALORES_M2Z1/CONT_M2_Z1);
HABILITAR_M2_Z1:=1;

END_IF;

IF (CONT_M2_Z2=0) THEN

MED_M2_Z2:=0;
HABILITAR_M2_Z2:=0;

ELSE

MED_M2_Z2:=(SUMA_VALORES_M2Z2/CONT_M2_Z2);
HABILITAR_M2_Z2:=1;

END_IF;

IF (CONT_M2_Z3=0) THEN

MED_M2_Z3:=0;
HABILITAR_M2_Z3:=0;

ELSE

MED_M2_Z3:=(SUMA_VALORES_M2Z3/CONT_M2_Z3);
HABILITAR_M2_Z3:=1;

END_IF;

IF (CONT_M2_Z4=0) THEN

MED_M2_Z4:=0;
HABILITAR_M2_Z4:=0;

ELSE

MED_M2_Z4:=(SUMA_VALORES_M2Z4/CONT_M2_Z4);
HABILITAR_M2_Z4:=1;

END_IF;

IF (CONT_M2=0) THEN

MED_M2:=0;
HABILITAR_M2:=0;

ELSE

MED_M2:=(SUMA_VALORES_M2/CONT_M2);

HABILITAR_M2:=1;

END_IF;

IF (CONT_CAMP_Z1=0) THEN

MED_CAMP_Z1:=0;

HABILITAR_CAMP_Z1:=0;

NING_HABILITADO_CAMPZ1:=1;

ELSE

MED_CAMP_Z1:=(SUMA_VALORES_CAMPZ1 / CONT_CAMP_Z1);

HABILITAR_CAMP_Z1:=1;

NING_HABILITADO_CAMPZ1:=0;

END_IF;

IF (CONT_CAMP_Z2=0) THEN

MED_CAMP_Z2:=0;

HABILITAR_CAMP_Z2:=0;

NING_HABILITADO_CAMPZ2:=1;

ELSE

MED_CAMP_Z2:=(SUMA_VALORES_CAMPZ2 / CONT_CAMP_Z2);

HABILITAR_CAMP_Z2:=1;

NING_HABILITADO_CAMPZ2:=0;

END_IF;

IF (CONT_CAMP_Z3=0) THEN

MED_CAMP_Z3:=0;

HABILITAR_CAMP_Z3:=0;

NING_HABILITADO_CAMPZ3:=1;

ELSE

MED_CAMP_Z3:=(SUMA_VALORES_CAMPZ3 / CONT_CAMP_Z3);

HABILITAR_CAMP_Z3:=1;

NING_HABILITADO_CAMPZ3:=0;

END_IF;

IF (CONT_CAMP_Z4=0) THEN

 MED_CAMP_Z4:=0;

HABILITAR_CAMP_Z4:=0;

NING_HABILITADO_CAMPZ4:=1;

ELSE

MED_CAMP_Z4:=(SUMA_VALORES_CAMPZ4 / CONT_CAMP_Z4);

HABILITAR_CAMP_Z4:=1;

NING_HABILITADO_CAMPZ4:=0;

END_IF;

IF (CONT_CAMP=0) THEN

MED_CAMP:=0;

HABILITAR_CAMP:=0;

ELSE

MED_CAMP:=(SUMA_VALORES_CAMP / CONT_CAMP);

HABILITAR_CAMP:=1;

END_IF;

IF (CONT_CAMP_C=0) THEN

MED_CAMP_C:=0;

ELSE

MED_CAMP_C:=(SUMA_VALORES_CAMP_C / CONT_CAMP_C);

END_IF;

IF (CONT_CAMP_I=0) THEN

MED_CAMP_I:=0;

ELSE

MED_CAMP_I:=(SUMA_VALORES_CAMP_I / CONT_CAMP_I);

END_IF;

IF((CONT_M1=0) & (CONT_CAMP=0))THEN

MED_M1CAMP:=0;

ELSE

SUMA_VALORES_CAMPYM1:=SUMA_VALORES_M1+SUMA_VALORES_CAMP;

CONT_CAMPYM1:=CONT_M1+CONT_CAMP;

MED_M1CAMP:=SUMA_VALORES_CAMPYM1/CONT_CAMPYM1;

END_IF;

IF((CONT_M2=0) & (CONT_CAMP=0))THEN

MED_M2CAMP:=0;

ELSE

MED_M2CAMP:=((SUMA_VALORES_M2+SUMA_VALORES_CAMP) / (CONT_M2+CONT_CAMP));

END_IF;

IF (MAX_M1>MAX_CAMP) THEN

MAX_M1CAMP:=MAX_M1;

ELSE

MAX_M1CAMP:=MAX_CAMP;

END_IF;

IF (MAX_M2>MAX_CAMP) **THEN**

MAX_M2CAMP:=MAX_M2;

ELSE

MAX_M2CAMP:=MAX_CAMP;

END_IF;

IF (MIN_M1>MIN_CAMP) **THEN**

MIN_M1CAMP:=MIN_CAMP;

ELSE

MIN_M1CAMP:=MIN_M1;

END_IF;

IF (MIN_M2>MIN_CAMP) **THEN**

MIN_M2CAMP:=MIN_CAMP;

ELSE

MIN_M2CAMP:=MIN_M2;

END_IF;

//Se recorren las direcciones donde se van a almacenar los datos de los termopares virtuales. Se recorren de 72 en 72 porque cada grupo (Mesa1_Zona1, Mesa1_Z2, Mesa1_Z3...) contiene 3 termopares virtuales, máximo, mínimo, y media. Y se le asigna su valor correspondiente a cada uno de ellos

FOR ADDRESS_1:= 1728 **TO** 2928 **BY** 72 **DO**

CASE ADDRESS_1 **OF**

1728 :

Q_MAXIMO:=MAX_M1_Z1;

Q_MEDIA:=MED_M1_Z1;

HABILITAR_TER_VIRTUAL:=HABILITAR_M1_Z1;

IF (MIN_M1_Z1<>1000) **THEN**

Q_MINIMO:=MIN_M1_Z1;

ELSE

Q_MINIMO:=0;

END_IF;

1800 :

Q_MAXIMO:=MAX_M1_Z2;

Q_MEDIA:=MED_M1_Z2;

HABILITAR_TER_VIRTUAL:=HABILITAR_M1_Z2;

IF (MIN_M1_Z2<>1000) **THEN**

Q_MINIMO:=MIN_M1_Z2;

ELSE

```
Q_MINIMO:=0;  
END_IF;
```

1872:

```
Q_MAXIMO:=MAX_M1_Z3;  
Q_MEDIA:=MED_M1_Z3;  
HABILITAR_TER_VIRTUAL:=HABILITAR_M1_Z3;
```

```
IF (MIN_M1_Z3<>1000) THEN
```

```
Q_MINIMO:=MIN_M1_Z3;
```

```
ELSE
```

```
Q_MINIMO:=0;
```

```
END_IF;
```

1944:

```
Q_MAXIMO:=MAX_M1_Z4;  
Q_MEDIA:=MED_M1_Z4;  
HABILITAR_TER_VIRTUAL:=HABILITAR_M1_Z4;
```

```
IF (MIN_M1_Z4<>1000) THEN
```

```
Q_MINIMO:=MIN_M1_Z4;
```

```
ELSE
```

```
Q_MINIMO:=0;
```

```
END_IF;
```

2016:

```
Q_MAXIMO:=MAX_M1;  
MED_M1:=(MED_M1_Z1+MED_M1_Z2+MED_M1_Z3+MED_M1_Z4)/4;  
Q_MEDIA:=MED_M1;  
HABILITAR_TER_VIRTUAL:=HABILITAR_M1;
```

```
IF (MIN_M1<>1000) THEN
```

```
Q_MINIMO:=MIN_M1;
```

```
ELSE
```

```
Q_MINIMO:=0;
```

```
END_IF;
```

2088:

```
Q_MAXIMO:=MAX_M2_Z1;  
Q_MEDIA:=MED_M2_Z1;  
HABILITAR_TER_VIRTUAL:=HABILITAR_M2_Z1;
```

```
IF (MIN_M2_Z1<>1000) THEN
```

```
Q_MINIMO:=MIN_M2_Z1;
```

```
ELSE
```

```
Q_MINIMO:=0;
```

```
END_IF;
```

2160:

```
Q_MAXIMO:=MAX_M2_Z2;  
Q_MEDIA:=MED_M2_Z2;  
HABILITAR_TER_VIRTUAL:=HABILITAR_M2_Z2;
```

```
IF (MIN_M2_Z2<>1000) THEN
```

```
    Q_MINIMO:=MIN_M2_Z2;
```

```
ELSE
```

```
    Q_MINIMO:=0;
```

```
END_IF;
```

2232:

```
Q_MAXIMO:=MAX_M2_Z3;  
Q_MEDIA:=MED_M2_Z3;  
HABILITAR_TER_VIRTUAL:=HABILITAR_M2_Z3;
```

```
IF (MIN_M2_Z3<>1000) THEN
```

```
    Q_MINIMO:=MIN_M2_Z3;
```

```
ELSE
```

```
    Q_MINIMO:=0;
```

```
END_IF;
```

2304:

```
Q_MAXIMO:=MAX_M2_Z4;  
Q_MEDIA:=MED_M2_Z4;  
HABILITAR_TER_VIRTUAL:=HABILITAR_M2_Z4;
```

```
IF (MIN_M2_Z4<>1000) THEN
```

```
    Q_MINIMO:=MIN_M2_Z4;
```

```
ELSE
```

```
    Q_MINIMO:=0;
```

```
END_IF;
```

2376:

```
Q_MAXIMO:=MAX_M2;  
MED_M2:=(MED_M2_Z1+MED_M2_Z2+MED_M2_Z3+MED_M2_Z4)/4;  
Q_MEDIA:=MED_M2;  
HABILITAR_TER_VIRTUAL:=HABILITAR_M2;
```

```
IF (MIN_M2<>1000) THEN
```

```
    Q_MINIMO:=MIN_M2;
```

```
ELSE
```

```
    Q_MINIMO:=0;
```

```
END_IF;
```

2448:

```
Q_MAXIMO:=MAX_CAMP_Z1;  
Q_MEDIA:=MED_CAMP_Z1;  
HABILITAR_TER_VIRTUAL:=HABILITAR_CAMP_Z1;
```

```
IF (MIN_CAMP_Z1<>1000) THEN
```

```
    Q_MINIMO:=MIN_CAMP_Z1;
```

```
ELSE
```

```
    Q_MINIMO:=0;
```

```
END_IF;
```

2520:

```
Q_MAXIMO:=MAX_CAMP_Z2;  
Q_MEDIA:=MED_CAMP_Z2;  
HABILITAR_TER_VIRTUAL:=HABILITAR_CAMP_Z2;
```

```
IF (MIN_CAMP_Z2<>1000) THEN
```

```
    Q_MINIMO:=MIN_CAMP_Z2;
```

```
ELSE
```

```
    Q_MINIMO:=0;
```

```
END_IF;
```

2592:

```
Q_MAXIMO:=MAX_CAMP_Z3;  
Q_MEDIA:=MED_CAMP_Z3;  
HABILITAR_TER_VIRTUAL:=HABILITAR_CAMP_Z3;
```

```
IF (MIN_CAMP_Z3<>1000) THEN
```

```
    Q_MINIMO:=MIN_CAMP_Z3;
```

```
ELSE
```

```
    Q_MINIMO:=0;
```

```
END_IF;
```

2664:

```
Q_MAXIMO:=MAX_CAMP_Z4;  
Q_MEDIA:=MED_CAMP_Z4;  
HABILITAR_TER_VIRTUAL:=HABILITAR_CAMP_Z4;
```

```
IF (MIN_CAMP_Z4<>1000) THEN
```

```
    Q_MINIMO:=MIN_CAMP_Z4;
```

```
ELSE
```

```
    Q_MINIMO:=0;
```

```
END_IF;
```

2736:

```
Q_MAXIMO:=MAX_CAMP;  
MED_CAMP:=(MED_CAMP_Z1+MED_CAMP_Z2+MED_CAMP_Z3+MED_CAMP_Z4)/4;
```

```
Q_MEDIA:=MED_CAMP;  
HABILITAR_TER_VIRTUAL:=HABILITAR_CAMP;
```

```
IF (MIN_CAMP<>1000) THEN
```

```
    Q_MINIMO:=MIN_CAMP;
```

```
ELSE
```

```
    Q_MINIMO:=0;
```

```
END_IF;
```

```
2808:
```

```
Q_MAXIMO:=MAX_M1CAMP;
```

```
MED_M1CAMP:=(MED_M1+MED_CAMP)/2;
```

```
Q_MEDIA:=MED_M1CAMP;
```

```
HABILITAR_TER_VIRTUAL:=HABILITAR_M1CAMP;
```

```
IF (MIN_M1CAMP<>1000) THEN
```

```
    Q_MINIMO:=MIN_M1CAMP;
```

```
ELSE
```

```
    Q_MINIMO:=0;
```

```
END_IF;
```

```
2880:
```

```
Q_MAXIMO:=MAX_M2CAMP;
```

```
MED_M2CAMP:=(MED_M2+MED_CAMP)/2;
```

```
Q_MEDIA:=MED_M2CAMP;
```

```
HABILITAR_TER_VIRTUAL:=HABILITAR_M1CAMP;
```

```
IF (MIN_M2CAMP<>1000) THEN
```

```
    Q_MINIMO:=MIN_M2CAMP;
```

```
ELSE
```

```
    Q_MINIMO:=0;
```

```
END_IF;
```

```
END_CASE;
```

```
//Se escriben sus valores en la DB
```

```
WORD_TO_BLOCK_DB(INT_TO_WORD(3)).DD[ADDRESS_1 + 10]:= REAL_TO_DWORD(Q_MAXIMO);
```

```
WORD_TO_BLOCK_DB(3).DX[ADDRESS_1 + 22]:= HABILITAR_TER_VIRTUAL;
```

```
WORD_TO_BLOCK_DB(INT_TO_WORD(3)).DD[ADDRESS_1 + 34]:= REAL_TO_DWORD(Q_MINIMO);
```

```
WORD_TO_BLOCK_DB(3).DX[ADDRESS_1 + 46]:= HABILITAR_TER_VIRTUAL;
```

```
WORD_TO_BLOCK_DB(INT_TO_WORD(3)).DD[ADDRESS_1 + 58]:= REAL_TO_DWORD(Q_MEDIA);
```

```
WORD_TO_BLOCK_DB(3).DX[ADDRESS_1 + 70]:= HABILITAR_TER_VIRTUAL;
```

```
END_FOR;
```

```
//Si después de recorrer todos los termopares no había ninguno habilitado en alguna de las zonas de la campana se escribe en la DB
```

```

WORD_TO_BLOCK_DB(2).DX[4,5]:= NING_HABILITADO_CAMPZ1;
WORD_TO_BLOCK_DB(2).DX[4,6]:= NING_HABILITADO_CAMPZ2;
WORD_TO_BLOCK_DB(2).DX[4,7]:= NING_HABILITADO_CAMPZ3;
WORD_TO_BLOCK_DB(2).DX[5,0]:= NING_HABILITADO_CAMPZ4;
WORD_TO_BLOCK_DB(INT_TO_WORD(3)).DD[3322]:= REAL_TO_DWORD(MED_CAMP_C);
WORD_TO_BLOCK_DB(INT_TO_WORD(3)).DD[3346]:= REAL_TO_DWORD(MED_CAMP_I);

```

```
END_FUNCTION
```

4.2.6 FC701: SELECCIÓN DE VARIABLES DE PROCESO

Esta función también se ha realizado en SCL, en ella se recorre toda la DB donde se encuentran todos los datos de los termopares, y teniendo en cuenta los termopares habilitados se busca, en caso de estar en modo 1 zona, el primer termopar seleccionado para control, y en caso de estar en modo 4 zonas se busca el primero por cada una de ellas. A nivel del PLC no habría ningún problema de que se seleccionasen más de uno, pero el orden de recorrido de los termopares siempre es el mismo y cogería el primero que se encuentre, es a nivel de SCADA donde sólo se permite que esté seleccionado uno o cuatro según el modo de funcionamiento. Serán estos termopares los que se usen para controlar el programa de temperatura. Para la realización de esta función se ha usado el siguiente código:

```
FUNCTION FC701 : VOID
```

```
//Creación de variables temporales
```

```
VAR_TEMP
```

```
ADDRESS_3: INT;
```

```
ADDRESS_1: INT;
```

```
CONT_T: INT;
```

```
PV_Z1: REAL;
```

```
PV_Z2: REAL;
```

```
PV_Z3: REAL;
```

```
PV_Z4: REAL;
```

```
VALOR_REAL: REAL;
```

```
HABILITADO: BOOL;
```

```
SELECCIONADO_CONTROL: BOOL;
```

```
ERROR: BOOL;
```

```
Z1_HABILITADA: BOOL;
```

```
Z2_HABILITADA: BOOL;
```

```
Z3_HABILITADA: BOOL;
```

```
Z4_HABILITADA: BOOL;
```

```
END_VAR
```

```
//Creación de variables de entrada
```

```
VAR_INPUT
```

```
DB_3: BLOCK_DB; // Número de DB
```

```
DB_15: BLOCK_DB;
```

```
SELEC_MESA1_MESA2: BOOL; //0->MESA1;1->MESA2
```

```
ZONA: BOOL; //0->1ZONA 1->4ZONAS
```

```
END_VAR
```

```
//Inicialización de las variables de procesos y del contador
```

```
PV_Z1:=-1000;
```

```
PV_Z2:=-1000;
```

```
PV_Z3:=-1000;
```

```
PV_Z4:=-1000;
```

```
CONT_T:=-1000;
```

```
//Se recorre la DB de termopares de abajo hacia arriba, dándole prioridad a los virtuales, después a los de campana y por ultimo a los de las mesas
```

```
FOR ADDRESS_3:= 3120 TO 0 BY -24 DO
```

```
VALOR_REAL:=DWORD_TO_REAL(WORD_TO_BLOCK_DB(INT_TO_WORD(3)).DD[ADDRESS_3 + 10]);
```

```
HABILITADO:=DB_3.DX[(ADDRESS_3+ 22),0];
```

```
SELECCIONADO_CONTROL:=DB_3.DX[(ADDRESS_3+ 22),2];
```

```
//Si el termopar está habilitado, se ha seleccionado para el control y hay alguna zona sin asignarle variable de proceso entonces sigue recorriendo el bucle
```

```
IF ((HABILITADO=1)&(SELECCIONADO_CONTROL=1)&((PV_Z1=-1000)OR(PV_Z2=-1000)  
OR(PV_Z3=-1000)OR(PV_Z4=-1000)))THEN
```

```
IF (ZONA=0)THEN //1 ZONA
```

```
//En el caso una zona sólo se busca una variable de proceso y todas las zonas se controlan con la misma.
```

```
IF ((SELEC_MESA1_MESA2=0)&((ADDRESS_3<768) OR ((ADDRESS_3>1512)&(ADDRESS_3<2088)) OR  
((ADDRESS_3>2424)&(ADDRESS_3<2808)) OR (ADDRESS_3>2928)))THEN //MESA 1
```

```
PV_Z1:=VALOR_REAL;
```

```
PV_Z2:=VALOR_REAL;
```

```
PV_Z3:=VALOR_REAL;
```

```
PV_Z4:=VALOR_REAL;
```

```
CONT_T:=CONT_T+1;
```

```
ELSIF((SELEC_MESA1_MESA2=1)&((ADDRESS_3>744)&(ADDRESS_3<1728)) OR((ADDRESS_3>2064)  
&(ADDRESS_3<2784)) OR (ADDRESS_3>2928)))THEN //MESA 2
```

```
PV_Z1:=VALOR_REAL;
```

```
PV_Z2:=VALOR_REAL;
```

```
PV_Z3:=VALOR_REAL;
```

```
PV_Z4:=VALOR_REAL;
```

```
CONT_T:=CONT_T+1;
```

```
END_IF;
```

```
ELSE //4 ZONAS
```

```
//En el caso cuatro zonas se buscan cuatro variables de proceso, una por cada zona
```

```
//TERMOPAR PERTENECIENTE A LA ZONA 1 CON MESA 1
```

```
IF ((SELEC_MESA1_MESA2=0)&(PV_Z1=-1000)&((ADDRESS_3<192)OR((ADDRESS_3>1512)
```

```
&(ADDRESS_3<1584))OR((ADDRESS_3>1704)&(ADDRESS_3<1800))OR((ADDRESS_3>2424)
&(ADDRESS_3<2520))OR((ADDRESS_3>2928)&(ADDRESS_3<3000))))THEN
```

```
CONT_T:=CONT_T+1;
```

```
PV_Z1:=VALOR_REAL;
```

```
//TERMOPAR PERTENECIENTE A LA ZONA 1 CON MESA 2
```

```
ELSIF ((SELEC_MESA1_MESA2=1)&(PV_Z1=-1000)&(((ADDRESS_3>744) &(ADDRESS_3<960))
OR((ADDRESS_3>1512)&(ADDRESS_3<1584))OR((ADDRESS_3>2064)&(ADDRESS_3<2160))OR((ADDRESS_3>2424)
&(ADDRESS_3<2520)) OR((ADDRESS_3>2928)&(ADDRESS_3<3000))))THEN
```

```
CONT_T:=CONT_T+1;
```

```
PV_Z1:=VALOR_REAL;
```

```
//TERMOPAR PERTENECIENTE A LA ZONA 2 CON MESA 1
```

```
ELSIF ((SELEC_MESA1_MESA2=0)&(PV_Z2=-1000)&(((ADDRESS_3>168)&(ADDRESS_3<384))
OR((ADDRESS_3>1560)&(ADDRESS_3<1632))OR((ADDRESS_3>1776)&(ADDRESS_3<1872)) OR((ADDRESS_3>2496)
&(ADDRESS_3<2592))OR((ADDRESS_3>2976)&(ADDRESS_3<3048))))THEN
```

```
CONT_T:=CONT_T+1;
```

```
PV_Z2:=VALOR_REAL;
```

```
//TERMOPAR PERTENECIENTE A LA ZONA 2 CON MESA 2
```

```
ELSIF ((SELEC_MESA1_MESA2=1)&(PV_Z2=-1000)&(((ADDRESS_3>936)&(ADDRESS_3<1152))
OR((ADDRESS_3>1560)&(ADDRESS_3<1632))OR((ADDRESS_3>2136)&(ADDRESS_3<2232))OR((ADDRESS_3>2496)
&(ADDRESS_3<2592)) OR((ADDRESS_3>2976)&(ADDRESS_3<3048))))THEN
```

```
CONT_T:=CONT_T+1;
```

```
PV_Z2:=VALOR_REAL;
```

```
//TERMOPAR PERTENECIENTE A LA ZONA 3 CON MESA 1
```

```
ELSIF ((SELEC_MESA1_MESA2=0)&(PV_Z3=-1000)&(((ADDRESS_3>360)&(ADDRESS_3<576))
OR((ADDRESS_3>1608)&(ADDRESS_3<1680))OR((ADDRESS_3>1848)&(ADDRESS_3<1944))OR((ADDRESS_3>2568)
&(ADDRESS_3<2664)) OR((ADDRESS_3>3024)&(ADDRESS_3<3096))))THEN
```

```
CONT_T:=CONT_T+1;
```

```
PV_Z3:=VALOR_REAL;
```

```
//TERMOPAR PERTENECIENTE A LA ZONA 3 CON MESA 2
```

```
ELSIF ((SELEC_MESA1_MESA2=1)&(PV_Z3=-1000)&(((ADDRESS_3>1128)&(ADDRESS_3<1344))
OR((ADDRESS_3>1608)&(ADDRESS_3<1680))OR((ADDRESS_3>2208)&(ADDRESS_3<2304)) OR((ADDRESS_3>2568)
&(ADDRESS_3<2664)) OR((ADDRESS_3>3024)&(ADDRESS_3<3096))))THEN
```

```
CONT_T:=CONT_T+1;
```

```
PV_Z3:=VALOR_REAL;
```

```
//TERMOPAR PERTENECIENTE A LA ZONA 4 CON MESA 1
```

```
ELSIF ((SELEC_MESA1_MESA2=0)&(PV_Z4=-1000)&(((ADDRESS_3>552)&(ADDRESS_3<768))
OR((ADDRESS_3>1656)&(ADDRESS_3<1704))OR((ADDRESS_3>1920)&(ADDRESS_3<2016)) OR((ADDRESS_3>2640)
&(ADDRESS_3<2736))OR((ADDRESS_3>3072)&(ADDRESS_3<3140)))) THEN
```

```
CONT_T:=CONT_T+1;
```

```
PV_Z4:=VALOR_REAL;
```

```
//TERMOPAR PERTENECIENTE A LA ZONA 4 CON MESA 2
```

```
ELSIF ((SELEC_MESA1_MESA2=1)&(PV_Z4=-1000)&(((ADDRESS_3>1320)&(ADDRESS_3<1536))
OR((ADDRESS_3>1656)&(ADDRESS_3<1704))OR((ADDRESS_3>2280)&(ADDRESS_3<2376)) OR((ADDRESS_3>2640)
&(ADDRESS_3<2736))OR((ADDRESS_3>3072)&(ADDRESS_3<3140)))) THEN
```

```
CONT_T:=CONT_T+1;
```

```
PV_Z4:=VALOR_REAL;
```

```

        END_IF;
    END_IF;
END_IF;

IF (CONT_T=0) THEN //No hay ningún termopar habilitado para el control
    ERROR:=1;
END_IF;
IF (PV_Z1<>-1000) THEN //Se ha encontrado un termopar en la zona 1
    Z1_HABILITADA:=1;
END_IF;
IF (PV_Z2<>-1000) THEN //Se ha encontrado un termopar en la zona 2
    Z2_HABILITADA:=1;
END_IF;
IF (PV_Z3<>-1000) THEN //Se ha encontrado un termopar en la zona 3
    Z3_HABILITADA:=1;
END_IF;
IF (PV_Z4<>-1000) THEN //Se ha encontrado un termopar en la zona 4
    Z4_HABILITADA:=1;
END_IF;

//Volcamos los valores de las variables de procesos a la DB, y si están o no las zonas habilitadas.
WORD_TO_BLOCK_DB(INT_TO_WORD (15)).DD[0]:= REAL_TO_DWORD(PV_Z1);
WORD_TO_BLOCK_DB(INT_TO_WORD (15)).DD[4]:= REAL_TO_DWORD(PV_Z2);
WORD_TO_BLOCK_DB(INT_TO_WORD (15)).DD[8]:= REAL_TO_DWORD(PV_Z3);
WORD_TO_BLOCK_DB(INT_TO_WORD (15)).DD[12]:= REAL_TO_DWORD(PV_Z4);

WORD_TO_BLOCK_DB(15).DX[16,0]:=Z1_HABILITADA;
WORD_TO_BLOCK_DB(15).DX[16,1]:=Z2_HABILITADA;
WORD_TO_BLOCK_DB(15).DX[16,2]:=Z3_HABILITADA;
WORD_TO_BLOCK_DB(15).DX[16,3]:=Z4_HABILITADA;
WORD_TO_BLOCK_DB(15).DX[16,4]:=ERROR;

END_FUNCTION

```

4.2.7 FC801: DECODIFICACIÓN DE EVENTOS

Esta función se encarga de decodificar los Bytes de eventos que se encuentran en la DB61, para poder saber en cada momento que eventos se realizan en el escalón del programa en el que nos encontramos.

```
FUNCTION FC801: VOID
```

```
VAR_TEMP
```

```
END_VAR
```

```
//Creación de variables de entradas
```

VAR_INPUT

```
DB_61: BLOCK_DB;  
INDICE: INT;  
HABILITAR: BOOL;
```

END_VAR

//Creación de variables de salidas

VAR_OUTPUT

```
BIT0: BOOL;  
BIT1: BOOL;  
BIT2: BOOL;  
BIT3: BOOL;  
BIT4: BOOL;  
BIT5: BOOL;  
BIT6: BOOL;  
BIT7: BOOL;
```

END_VAR

//La DB que se recorre está formada por un programa, el cual tiene distintos eventos, el código decodifica cada escalón para saber en que evento está según el índice que esté activo.

```
IF (HABILITAR=1 & INDICE>0) THEN
```

```
BIT0:=DB_61.DX[(INDICE-1)*2,0];  
BIT1:=DB_61.DX[(INDICE-1)*2,1];  
BIT2:=DB_61.DX[(INDICE-1)*2,2];  
BIT3:=DB_61.DX[(INDICE-1)*2,3];  
BIT4:=DB_61.DX[(INDICE-1)*2,4];  
BIT5:=DB_61.DX[(INDICE-1)*2,5];  
BIT6:=DB_61.DX[(INDICE-1)*2,6];  
BIT7:=DB_61.DX[(INDICE-1)*2,7];
```

ELSE

```
BIT0:=FALSE;  
BIT1:=FALSE;  
BIT2:=FALSE;  
BIT3:=FALSE;  
BIT4:=FALSE;  
BIT5:=FALSE;  
BIT6:=FALSE;  
BIT7:=FALSE;
```

```
END_IF;
```

```
END_FUNCTION
```

4.2.8 FC4: GESTIÓN DE MESAS

En esta función se gestiona todo lo relacionado con las dos mesas, si están dentro o fuera de la campana, si están sobrecargadas o no, si es seguro moverlas o no, etc.

En primer lugar, cuando se inicia un ciclo automático se asigna una mesa al ciclo según la que esté seleccionada en el instante de pulsar el botón, una vez asignada, aunque se cambie el selector de mesa, la mesa que realizará el ciclo será la que se seleccionó en el instante inicial.

Segm. 1: Asignación de mesa de un ciclo

```
Mesa_asignada_ciclo=0 -> Mesa 1, Mesa_asignada_ciclo=1 -> Mesa 2.
```

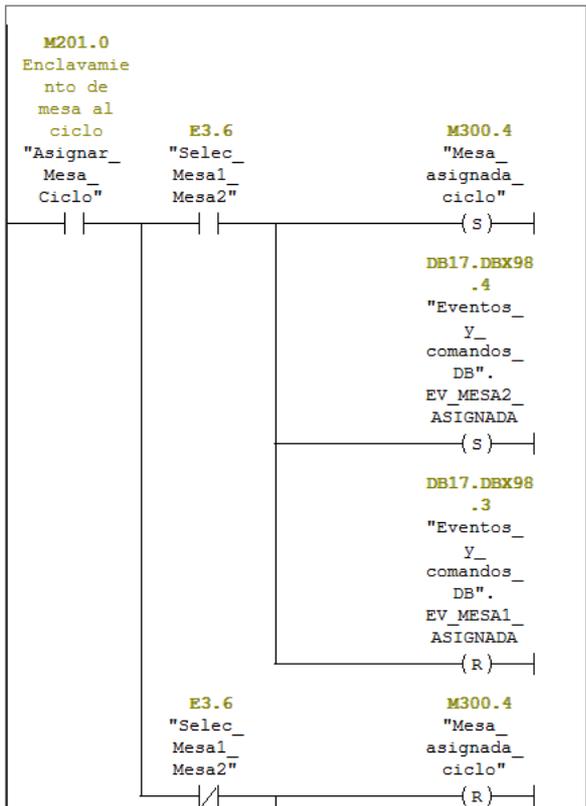


Ilustración 31: FC4 Segmento 1_1

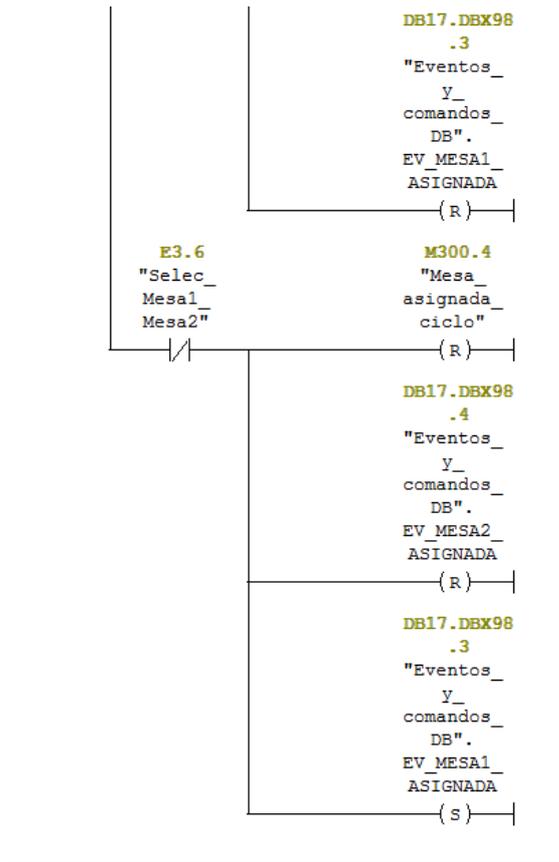
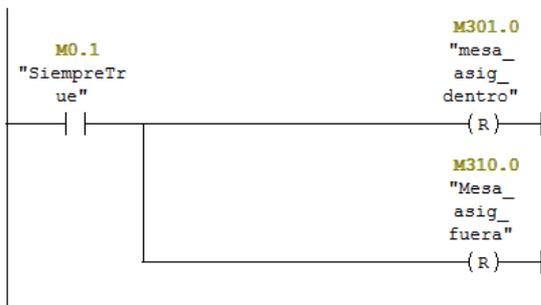


Ilustración 32: FC4 Segmento 1_2

Segm. 2 : Título:



Segm. 3 : Mesa 1 Dentro

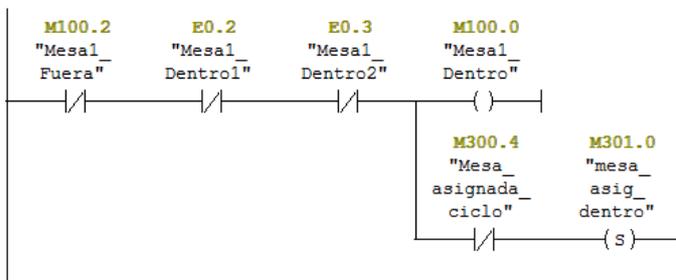
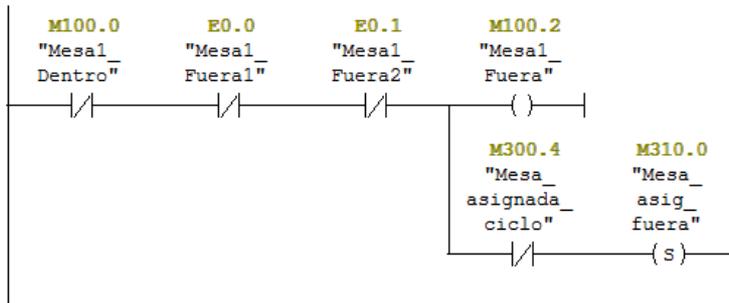


Ilustración 33: FC4 Segmentos 2 y 3

Segm. 4 : Mesa 1 Fuera



Segm. 5 : Mesa 2 Dentro

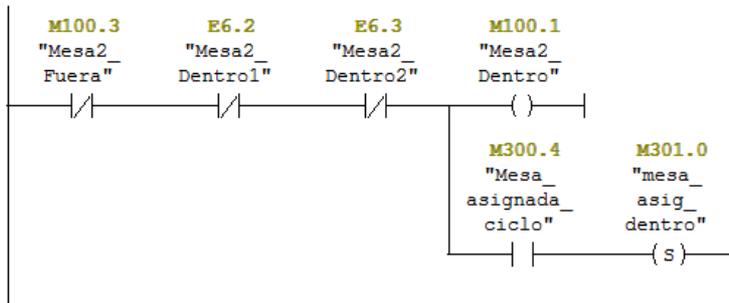
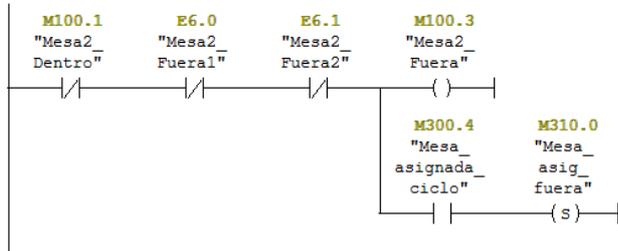


Ilustración 34: FC4 Segmentos 4 y 5

Segm. 6 : Mesa 2 Fuera



Segm. 7 : warning sobrecarga mesa1

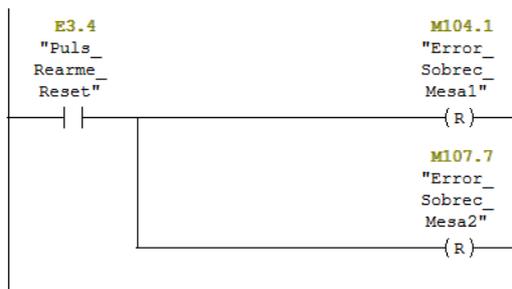


Segm. 8 : warning sobrecarga mesa2

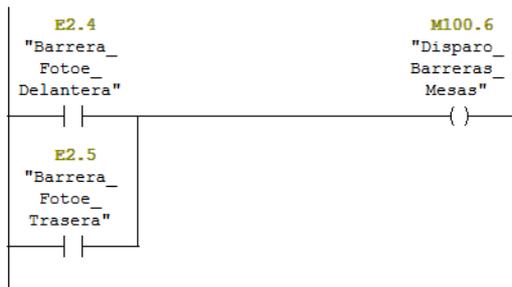


Ilustración 35: FC4 Segmentos 6, 7 y 8

Segm. 9 : Título:



Segm. 10 : Disparo de barreras fotoelectricas



Segm. 11 : Seguridad meter mesal

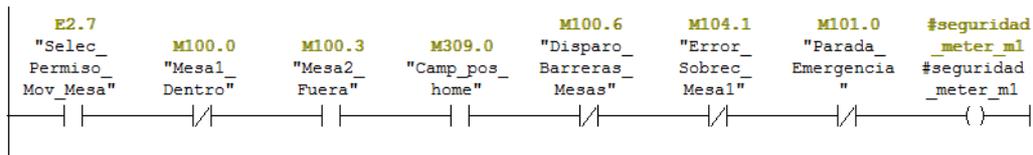
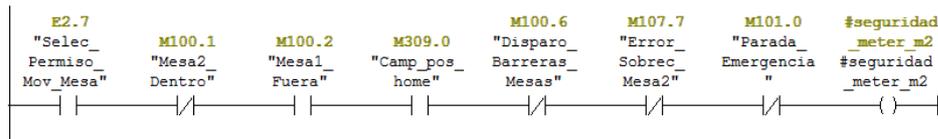
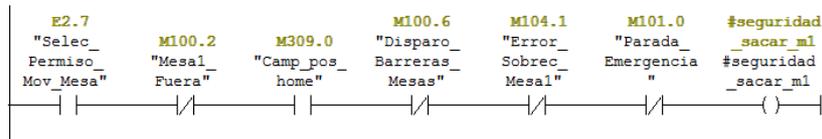


Ilustración 36: FC4 Segmentos 9, 10 y 11

Segm. 12 : Seguridad meter mesa2



Segm. 13 : Seguridad sacar mesal



Segm. 14 : Seguridad sacar mesa2

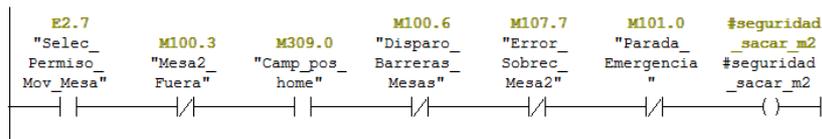


Ilustración 37: FC4 Segmentos 12, 13 y 14

Segm. 16: Control Meter Mesa Manual

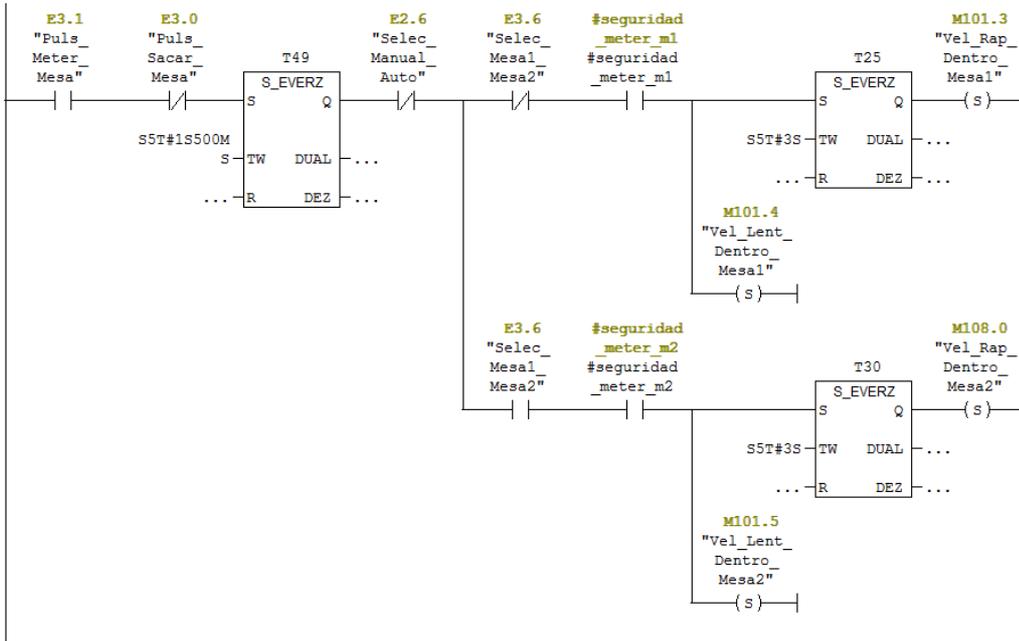


Ilustración 38: FC4 Segmento 16

Segm. 17: Control Sacar Mesa Manual

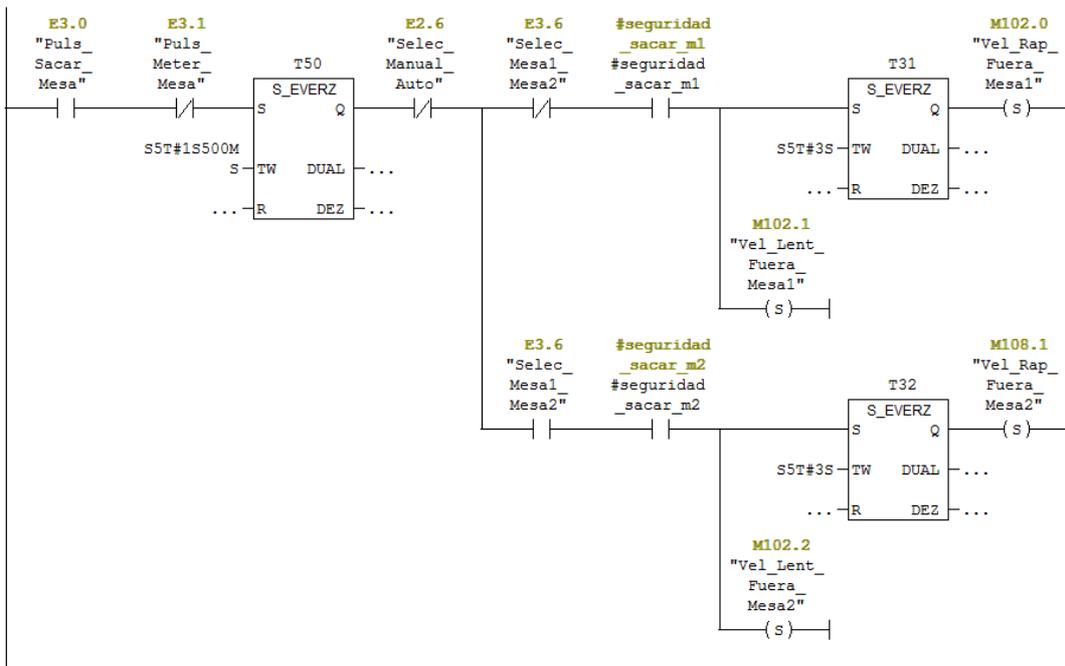


Ilustración 39: FC4 Segmento 17

Segm. 18 : Control Meter Mesa Automático

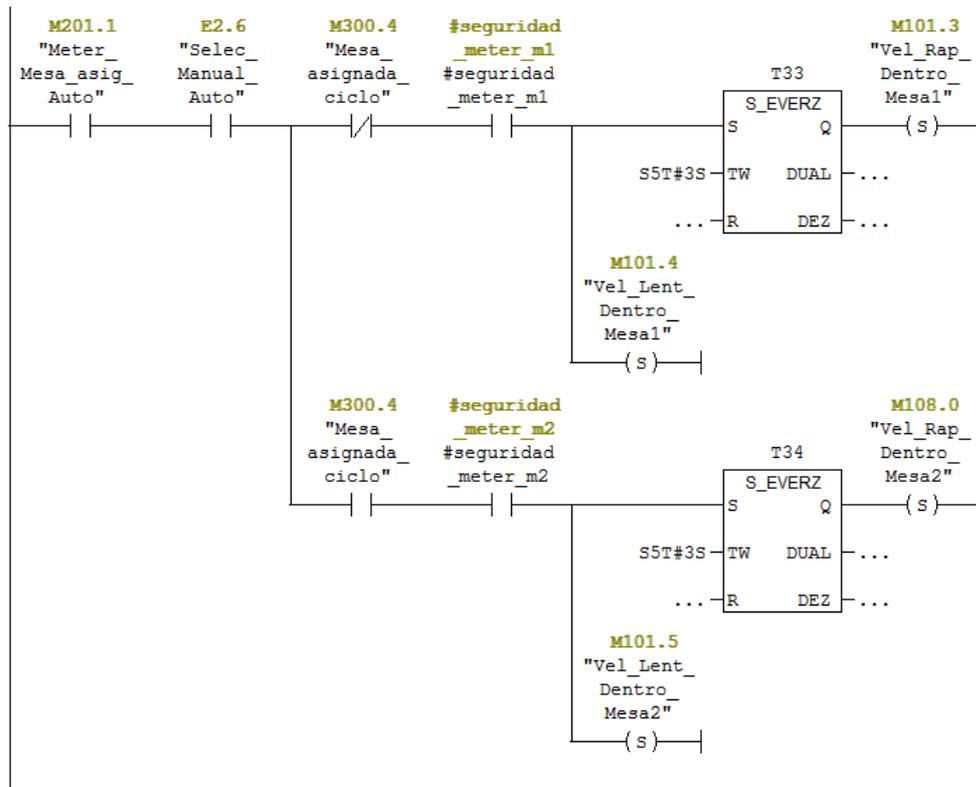


Ilustración 40: FC4 Segmento 18

Segm. 19 : Control Sacar Mesa Automático

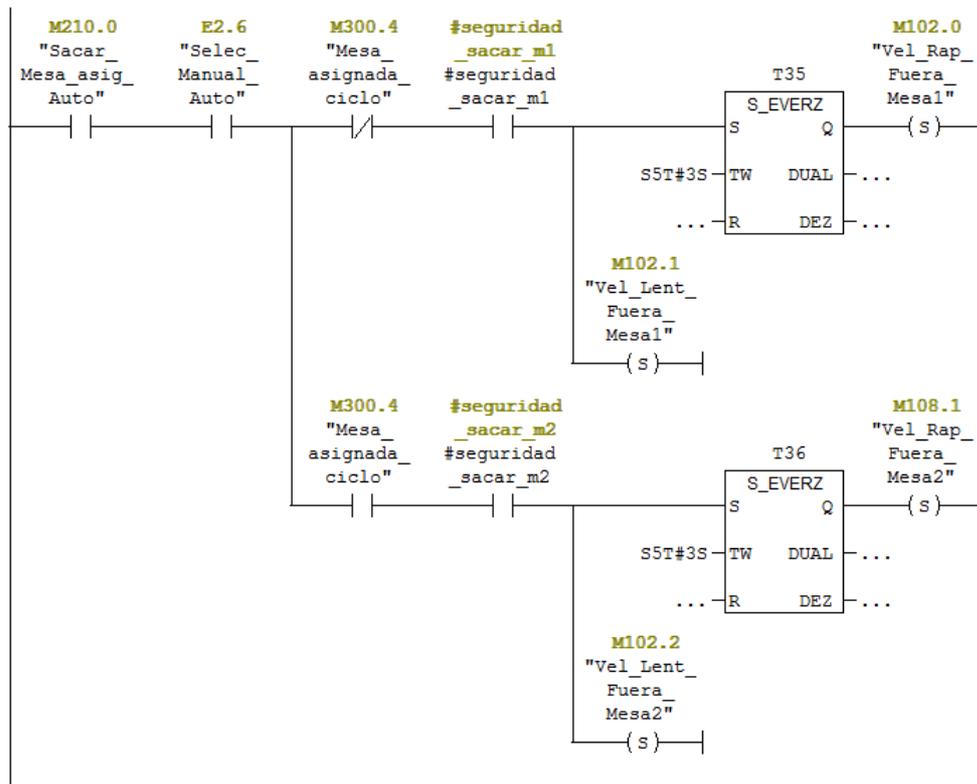
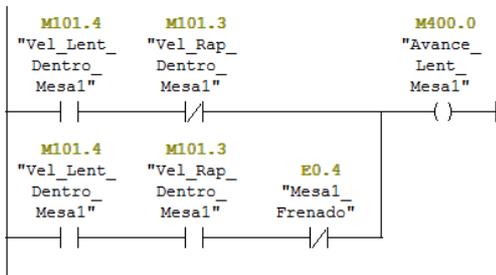


Ilustración 41: FC4 Segmento 19

☐ Segm. 20 : Marcha lenta y frenado entrar mesa 1



☐ Segm. 21 : Marcha lenta y frenado entrar mesa 2

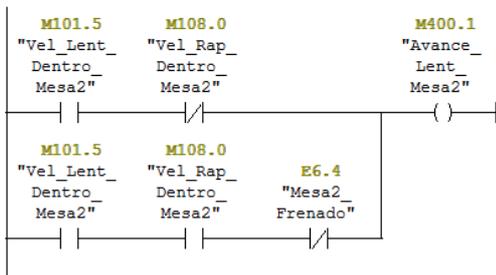
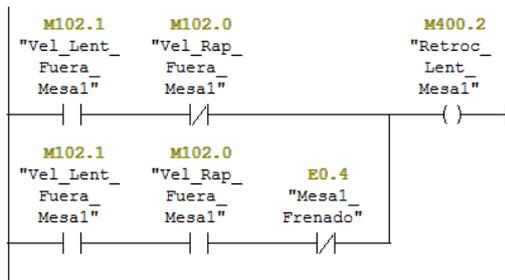
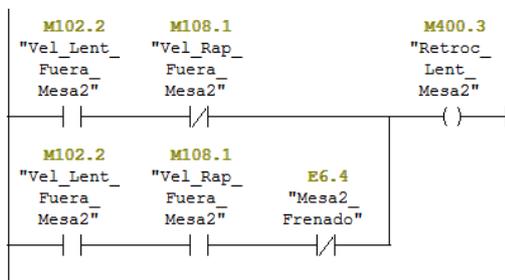


Ilustración 42: FC4 Segmentos 20 y 21

☐ Segm. 22 : Marcha lenta y frenado sacar mesa 1



☐ Segm. 23 : Marcha lenta y frenado sacar mesa 2



☐ Segm. 24 : Marcha rapida entrar mesa 1

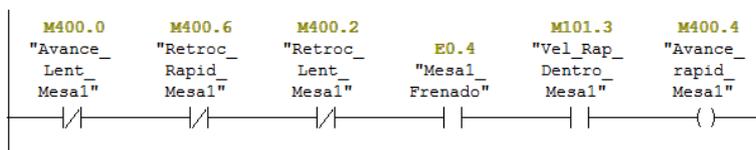
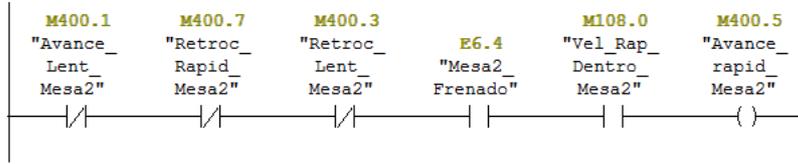
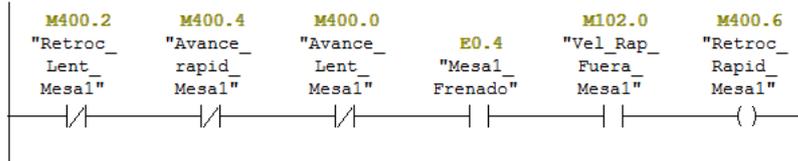


Ilustración 43: FC4 Segmentos 22, 23 y 24

☐ Segm. 25: Marcha rapida entrar mesa 2



☐ Segm. 26: Marcha rapida sacar mesa 1



☐ Segm. 27: Marcha rapida sacar mesa 2

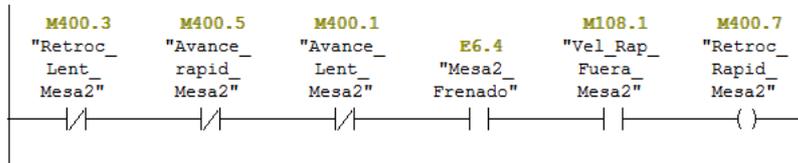


Ilustración 44: FC4 Segmentos 25, 26 y 27

4.2.9 FC5: GESTIÓN DE CAMPANA

En esta FC se gestiona todo lo relacionado con la campana, en qué posición se encuentra, si da algún fallo, si es seguro o no moverla...

FC5 : Título:

```

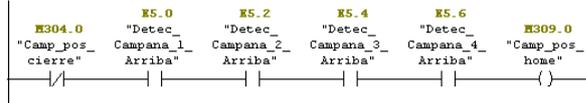
CONTROL CAMPANA
MOTOR 1 - DELANTERO DERECHO
MOTOR 2 - DELANTERO IZQUIERDO
MOTOR 3 - TRASERO DERECHO
MOTOR 4 - TRASERO IZQUIERDO

Si al llegar a arriba o abajo, pasa mas de un segundo desde que se activa el
primer sensor al ultimo, tenemos de que dar un fallo.

Los finales de carrera de seguridad de los cilindros tambien darán fallo,
siendo activos en nivel bajo.
El movimiento se comanda con 4 contactores, uno por motor, que se regiran por
su correspondiente sensor de campana arriba o abajo, aparte de dos contactores
más que rigen el sentido de marcha (arriba o abajo).

Falta añadir que para subir campana esta la valvula de rotura de vacio abierta,
y la valvula de vacio que toque segun la mesa que este dentro
Para la posicion e campana hay que tomar menor que 13 como 0 y mayor que 80
como
100 segun el plc honeywell
    
```

☐ Segm. 1: Campana Arriba



☐ Segm. 2: Campana Abajo

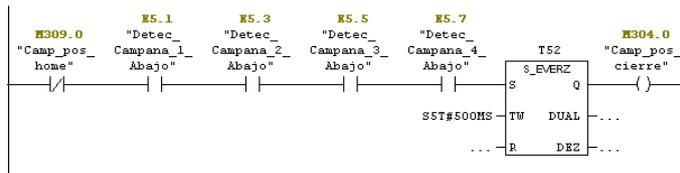
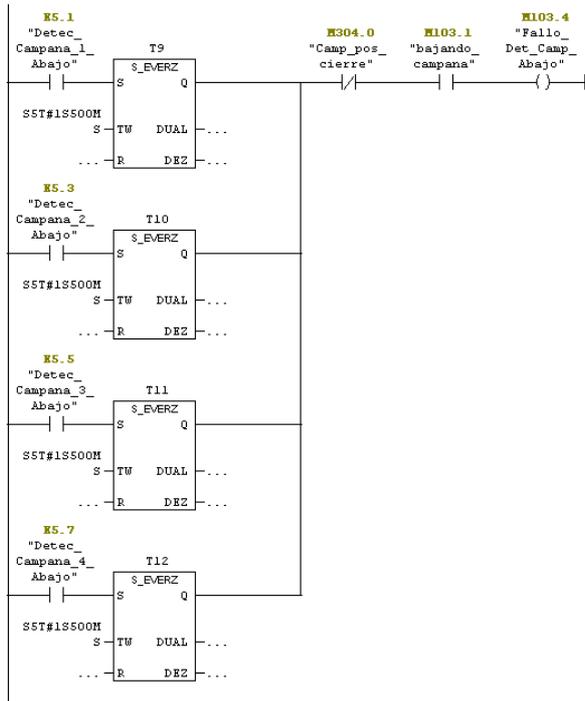


Ilustración 45: FC5 Segmentos 1 y 2

Segm. 3: Seguridad sensores cierre campana



Segm. 4: Seguridad sensores home campana

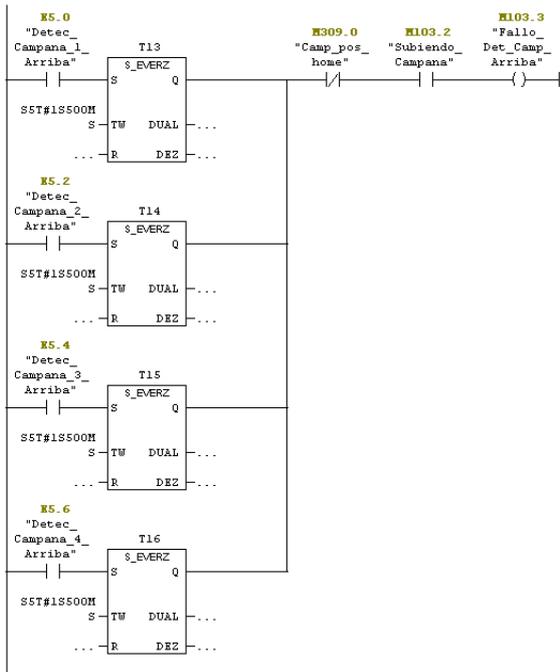


Ilustración 46: FC5 Segmentos 3 y 4

Segm. 5 : Seguridad finales de carrera home campana

Son activas en Baja, en el PLC original de Honeywell hay un error aqui.

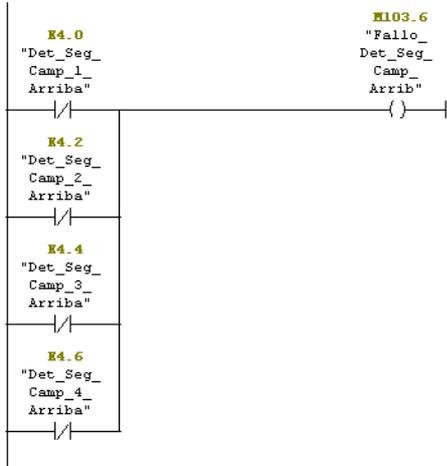


Ilustración 47: FC5 Segmento 5

Segm. 6 : Seguridad finales de carrera cierre campana

Son activas en Baja, en el PLC original de Honeywell hay un activor aqui.

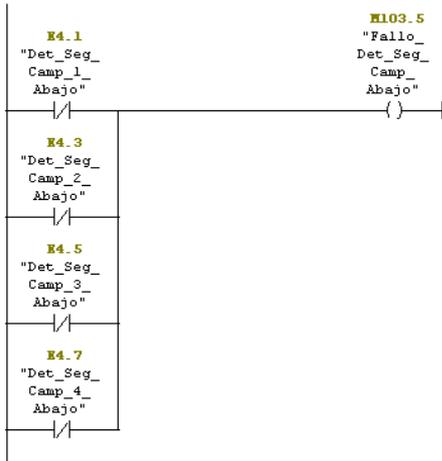
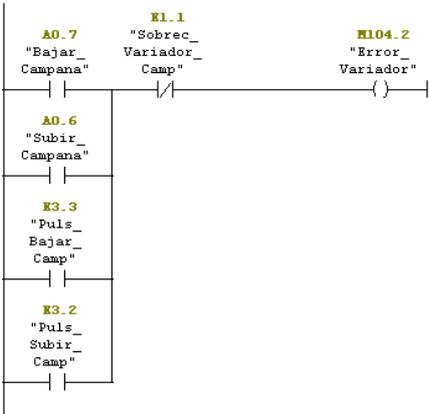


Ilustración 48: FC5 Segmento 6

Segm. 7 : Error variador

Estas alarmas y errores deberían ser gestionadas en otro sitio



Segm. 8 : Titulo:

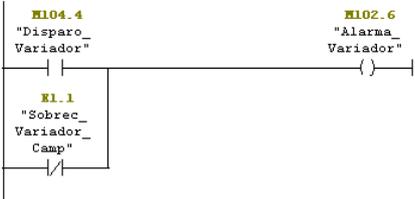


Segm. 9 : Titulo:

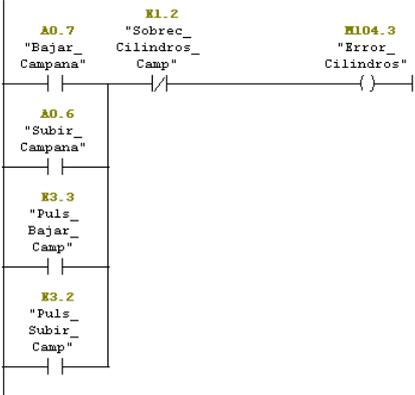


Ilustración 49: FC5 Segmentos 7, 8 y 9

Segm. 10 : Alarma Variador



Segm. 11 : Titulo:



Segm. 12 : Titulo:

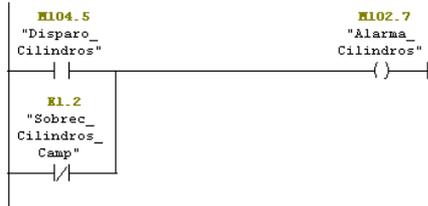


Ilustración 50: FC5 Segmentos 10, 11 y 12

Segm. 13 : Título:



Segm. 14 : Alarma Cilindros



Segm. 15 : Disparo_barreras

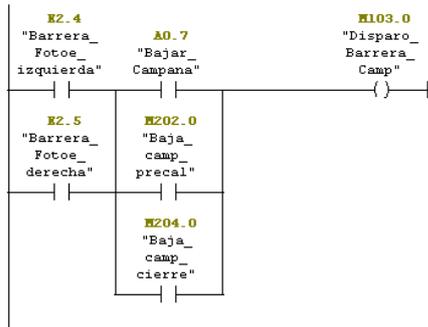
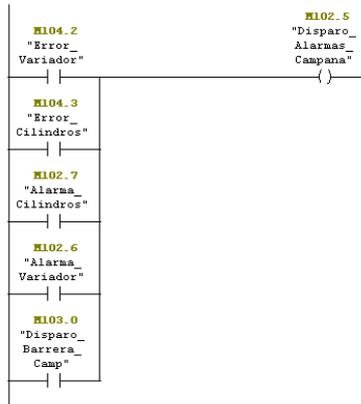


Ilustración 51: FC5 Segmentos 13, 14 y 15

Segm. 16 : Alarma Campana



Segm. 17 : Título:



Segm. 18 : Control movimiento hacia Abajo Manual

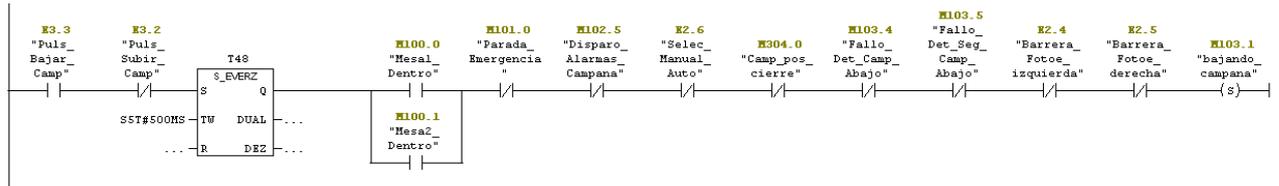


Ilustración 52: FC5 Segmentos 16, 17 y 18

Segm. 19: calculo de rango en posicion de campana

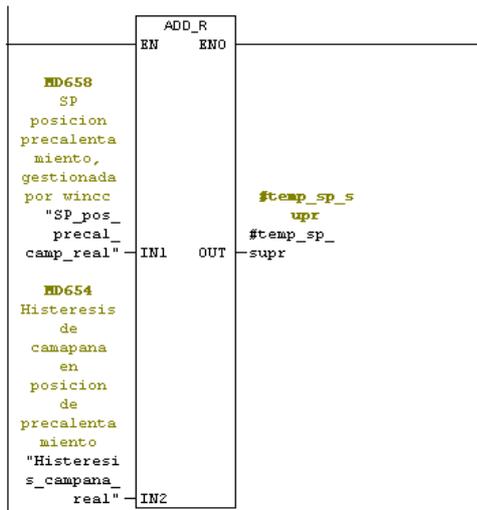
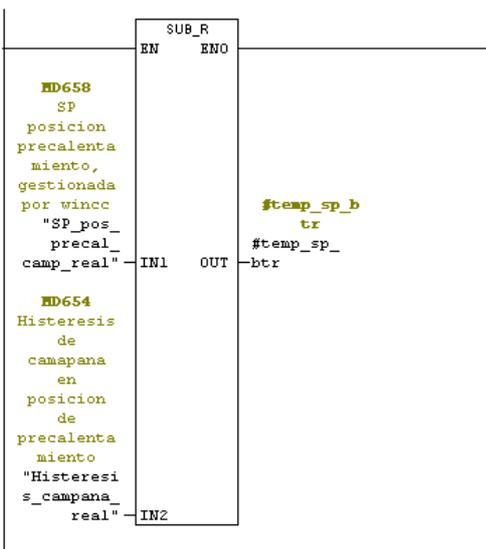


Ilustración 53: FC5 Segmento 19

Segm. 20: calculo de rango en posicion de campana



Segm. 21: calculo de rango en posicion de campana

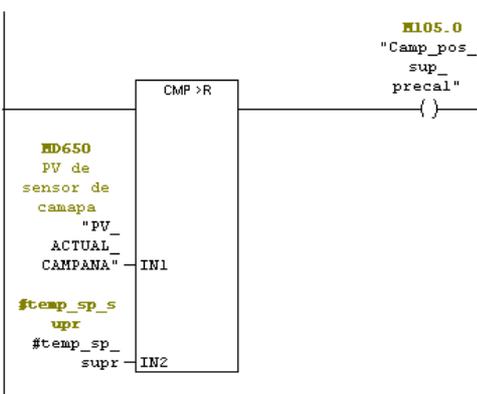
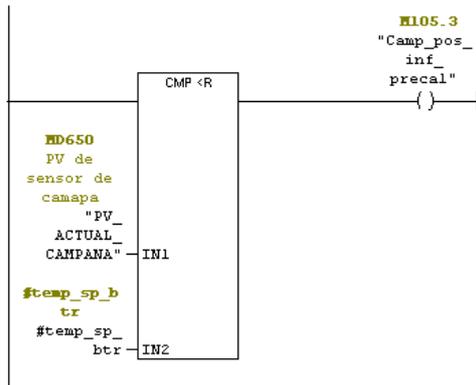


Ilustración 54: FC5 Segmentos 20 y 21

Segm. 22 : calculo de rango en posicion de campana



Segm. 23 : Campana en posicion de precalentamiento

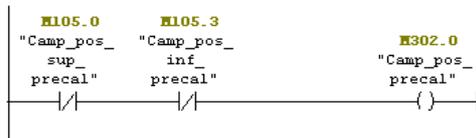
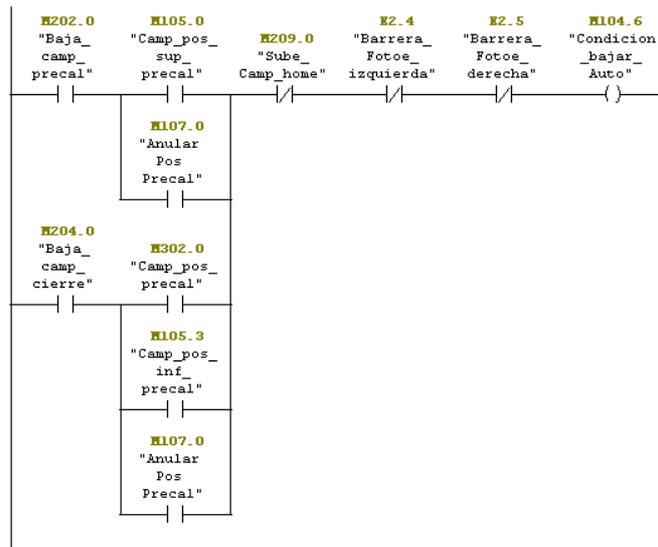


Ilustración 55: FC5 Segmentos 22 y 23

Segm. 24 : Ordenes de bajar campana en automatico



Segm. 25 : Control movimiento hacia Abajo Automático

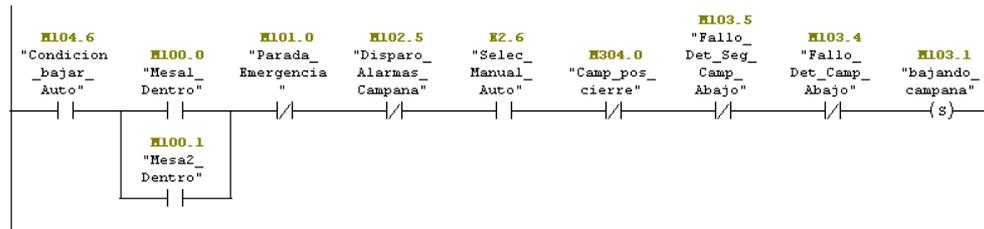


Ilustración 56: FC5 Segmentos 24 y 25

Segm. 26 : Condicion de membrana sin vacio

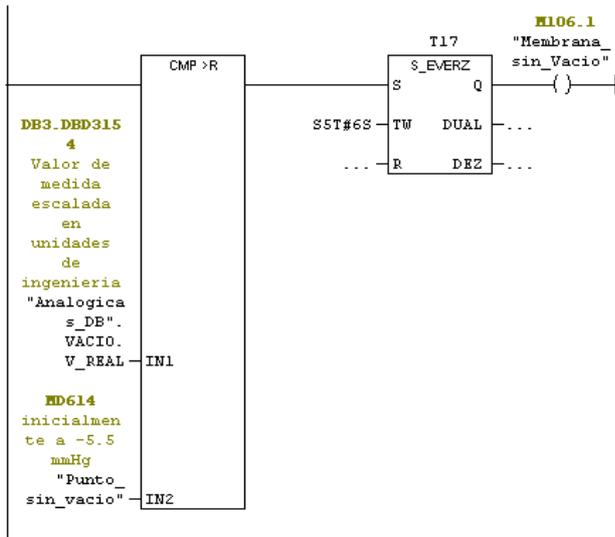


Ilustración 57: FC5 Segmento 26

Segm. 27 : Seguridad temperatura subir campana

Se comprueba la temperatura maxima de todos los termopares habilitados de membrana y de la mesa actualmente bajo la campana. No debe exceder 45 grados

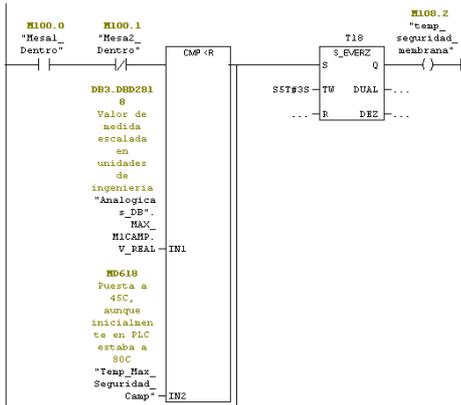


Ilustración 58: FC5 Segmento 27_1

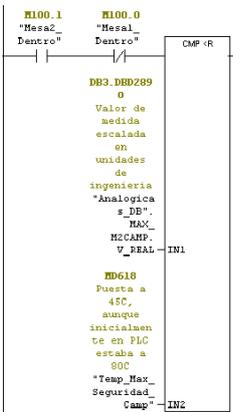


Ilustración 59: FC5 Segmento 27_2

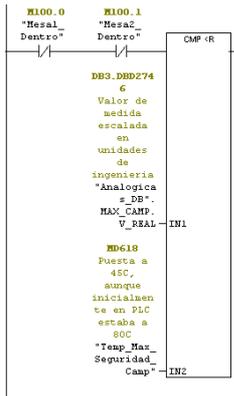
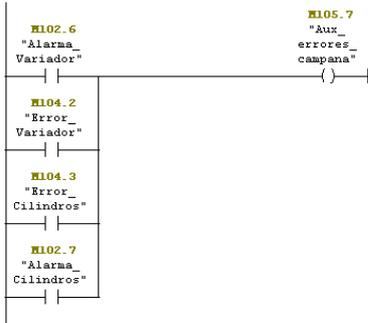


Ilustración 60: FC5 Segmentos 27_3

Segm. 28: seguridad subir campana



Segm. 29: Condiciones de Seguridad



Segm. 30: Titulo:



Ilustración 61: FC5 Segmentos 28, 29 y 30

4.2.10 FC6: GESTIÓN DE VÁLVULAS DE VACÍO

FC6 : Gestion de valvulas y vacio

Gestion de las valvulas de vacio y rotura de vacio asi como de la bomba de vacio.
Activamos la valvula de vacio correspondiente siempre que este una mesa dentro.
Son dos valvulas, una tipo NA correspondiente a la mesa 1, y otra tipo NC, correspondiente a la mesa2.

Segm. 1: Control vacio para la mesa 1

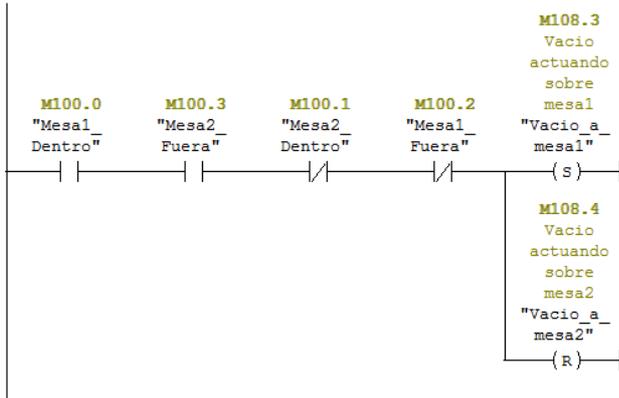
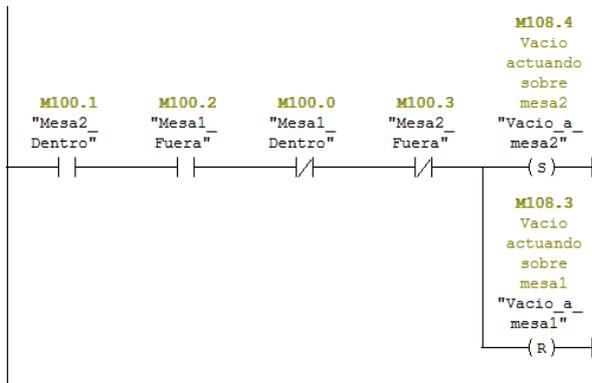


Ilustración 62: FC6 Segmento 1

Segm. 2: Control vacio para la mesa 2



Segm. 3: Abrir rotura de vacio

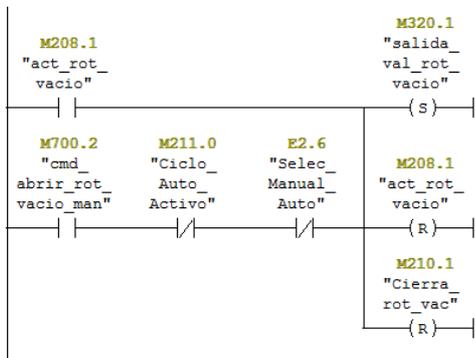
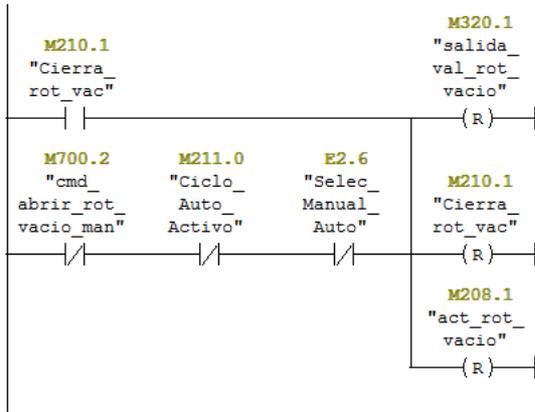


Ilustración 63: FC6 Segmentos 2 y 3

Segm. 4: Cerrar Rotura de vacio



Segm. 5: Activar Bomba de vacio

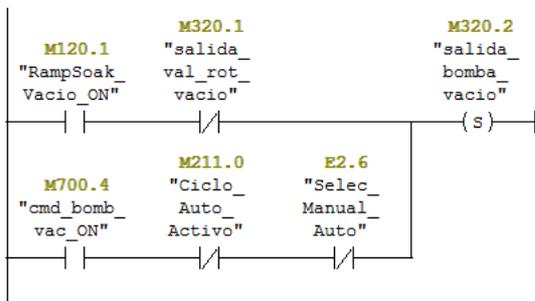


Ilustración 64: FC6 Segmentos 4 y 5

Segm. 6: Desactivar Bomba de vacio

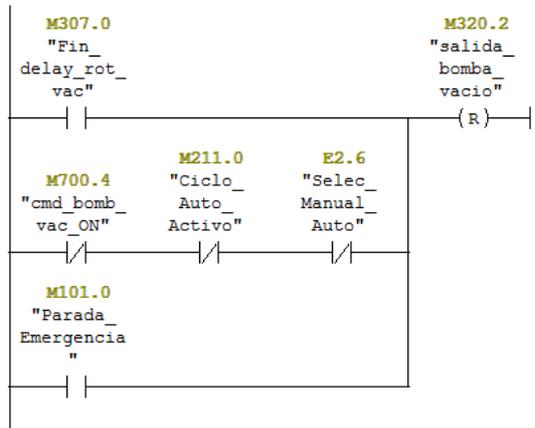


Ilustración 65: FC6 Segmento 6

4.2.11 FC11: GESTIÓN DEL PARPADEO DE LAS BARRERAS

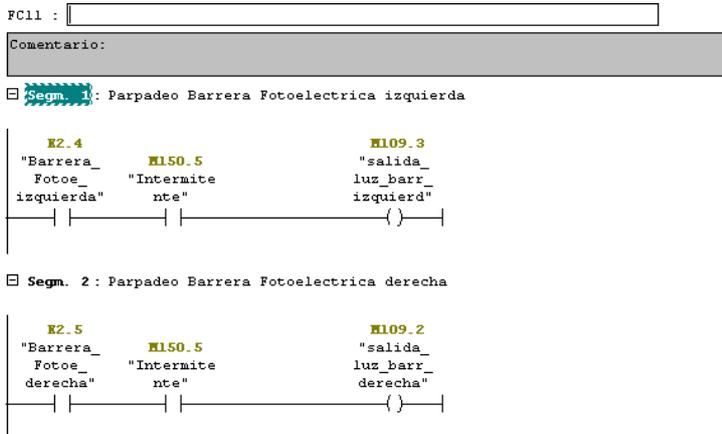
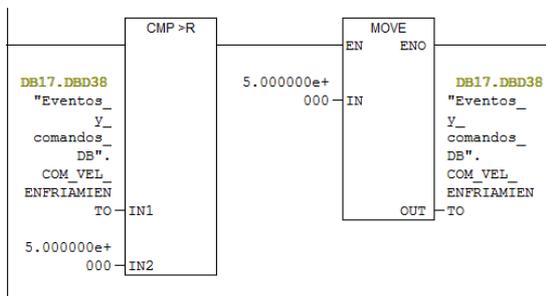


Ilustración 66: FC11 Segmentos 1 y 2

4.2.2.12 FC7: GESTIÓN DE REFRIGERACIÓN

La refrigeración se puede hacer a varias velocidades, pero la máxima permitida es 5°C/min, por lo que, si en el SCADA dan orden de ir más rápido, en el PLC se cambia ese valor por el máximo, no permitiendo que esa consigna sea superior. Lo mismo ocurre con el mínimo que es 0°C/min.

Segm. 1: Adecuación de valor de consigna de ref a maximos permitidos



Segm. 2: Adecuación de valor de consigna de ref a minimos permitidos

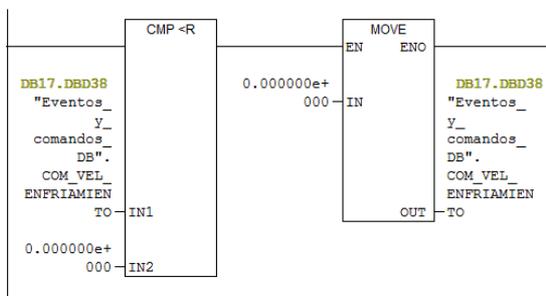


Ilustración 67: FC7 Segmentos 1 y 2

Segm. 3: Escritura Consigna de refrigeración

Consigna de 0-5 C°, desescalamos y escribimos una marca para la salida

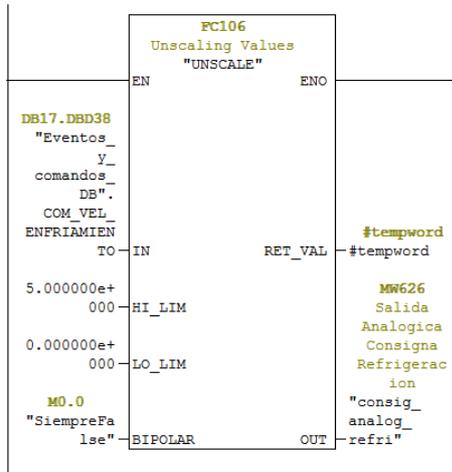
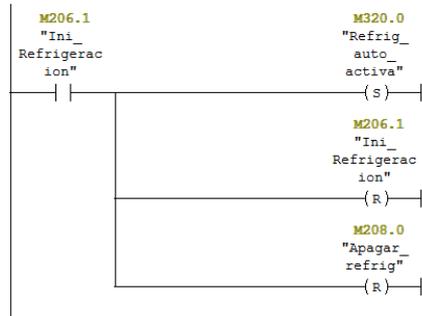


Ilustración 68: FC7 Segmento 3

Cuando se recibe la orden de iniciar refrigeración en el programa de ciclo auto, se activa la refrigeración automática, la cual también se para cuándo lo comunica el programa.

Segm. 4: Encendido Refrigeracion auto

Repasar esto conceptualmente y añadir guardas para modo manual



Segm. 5: Apagado Refrigeracion auto

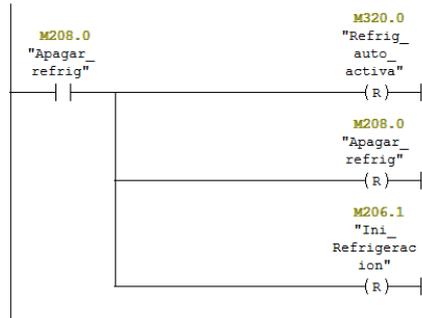


Ilustración 69: FC7 Segmentos 4 y 5

La refrigeración funciona cuando en manual se le da la orden de activar refrigeración o bien cuando es el ciclo automático quien da la orden.

Segm. 6: Activacion de UTA de refrigeracion

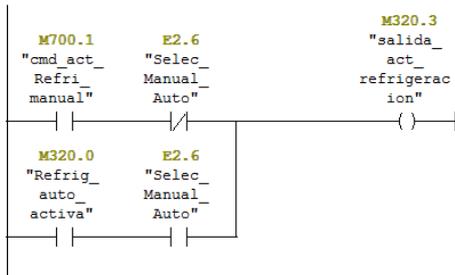


Ilustración 70: FC7 Segmento 6

Para saber si se ha llegado a la consigna de refrigeración es necesario comparar las temperaturas del interior de la campana con la consigna que se nos indica por el SCADA. Esto se puede hacer de distintas formas, se puede comparar la temperatura máxima, la mínima o la media, inicialmente esto se podía variar, actualmente se ha quitado esta opción y desde el PLC se ha forzado que siempre se compare la máxima por posibles riesgos al levantar la campana. Para ello primero se mira con que mesa se está trabajando y segundo si la temperatura máxima de esa mesa y de la campana es inferior a la consigna de enfriamiento, cuando esto se cumpla se parará la refrigeración. Esto sólo se usará en el modo automático.

Segm. 7: Calculo de la consigna de refrigeracion

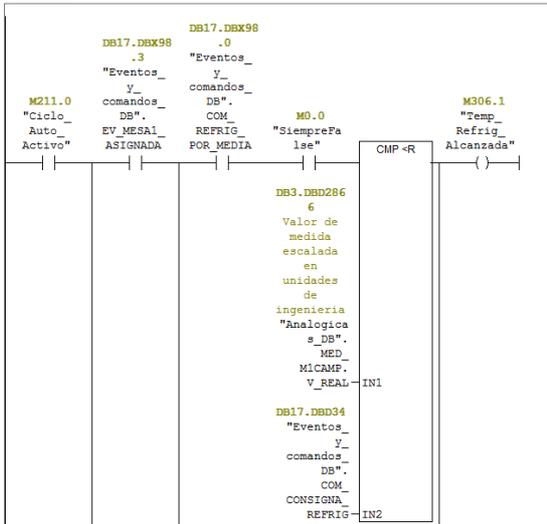


Ilustración 71: FC7 Segmento 7_1

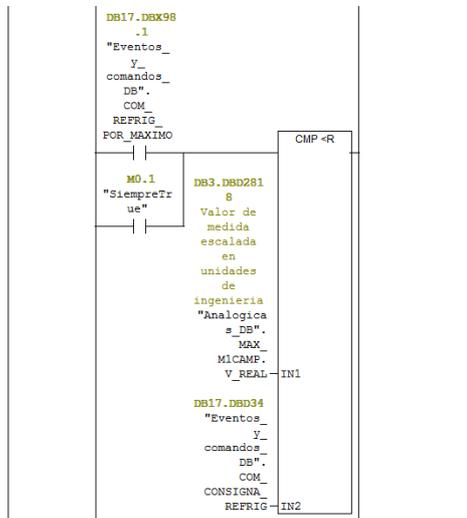


Ilustración 72: FC7 Segmento 7_2

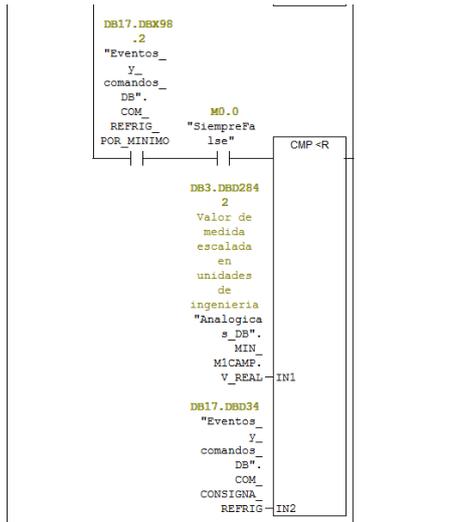


Ilustración 73: FC7 Segmentos 7_3

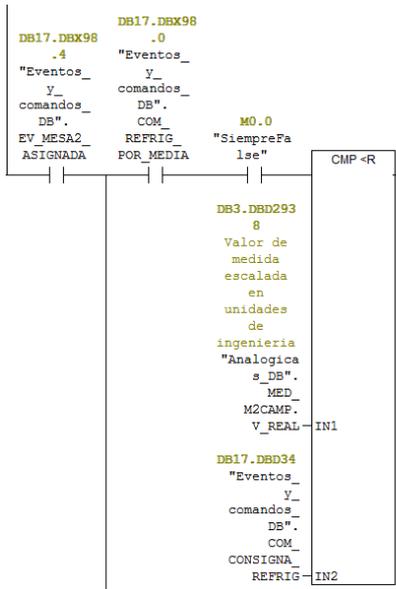


Ilustración 74: FC7 Segmento 7_4

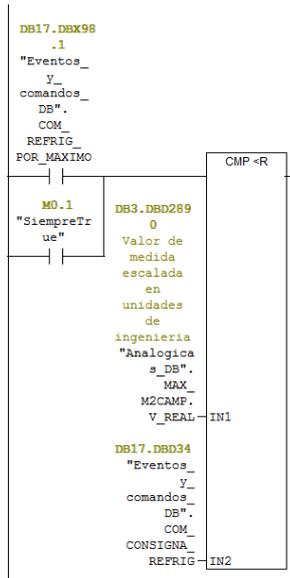


Ilustración 75: FC7 Segmento 7_5

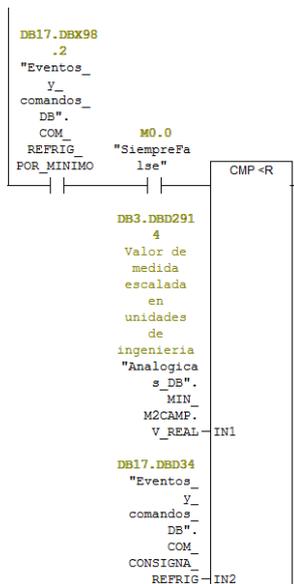


Ilustración 76: FC7 Segmento 7_6

4.2.13 FC2: GESTIÓN DE ALARMAS

En esta función se activan los diferentes avisos y alarmas que hay que mostrar por el SCADA para garantizar la seguridad de la máquina y de las personas que trabajan con ella. Si la alarma es grave parará el ciclo. Y con ellas no se podrán mover las mesas o la campana.

Secm.1: Parada de Emergencia



Ilustración 77: FC2 Segmento 1

Segm. 2: Reset Emergencia Comandos

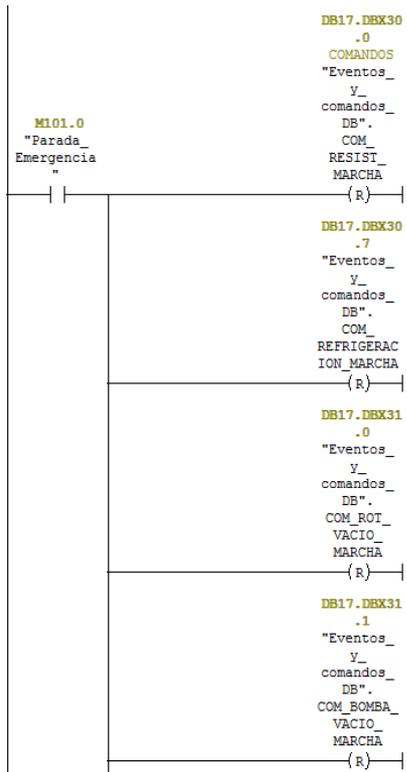


Ilustración 78: FC2 Segmento 2_1

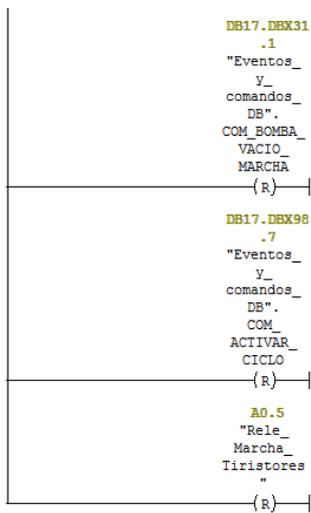


Ilustración 79: FC2 Segmento 2_2

Segm. 3 : Reset Ciclo Auto Etapas

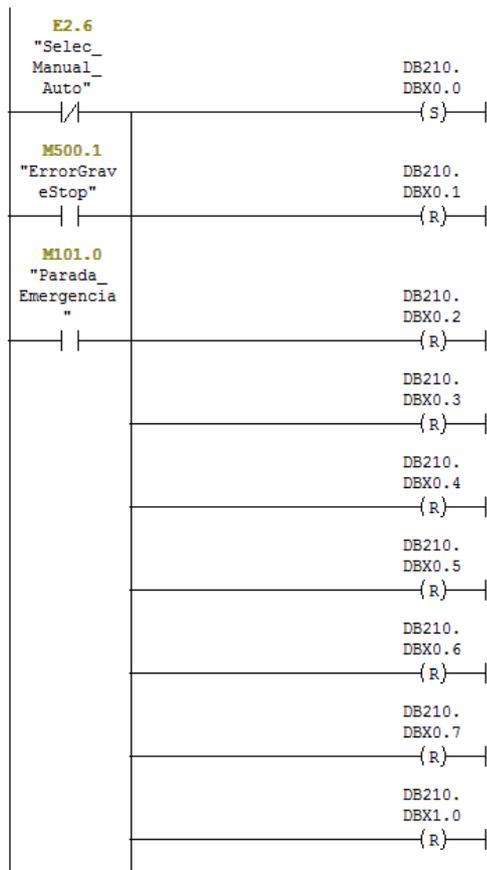


Ilustración 80: FC2 Segmento 3_1

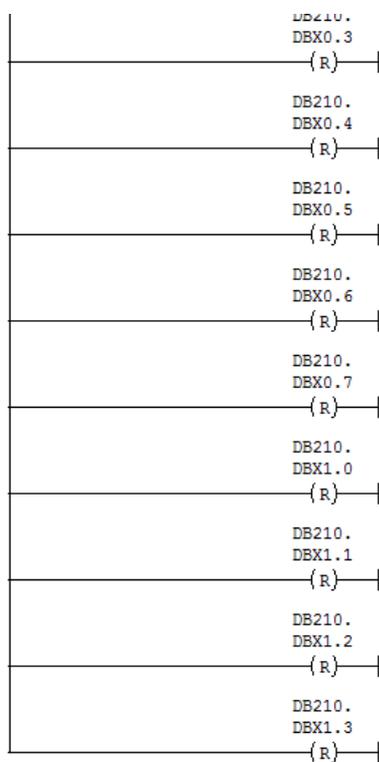


Ilustración 81: FC2 Segmento 3_2

☐ Segm. 4 : SETAS DE EMERGENCIA



☐ Segm. 5 : FALLO TIRISTORES 1



☐ Segm. 6 : FALLO TIRISTORES 2



☐ Segm. 7 : FALLO TIRISTORES 3



Ilustración 82: FC2 Segmentos 4, 5, 6 y 7

☐ Segm. 8 : FALLO TIRISTORES 4



Ilustración 83: FC2 Segmento 8

Segm. 9 : Sobrecarga Mesa1

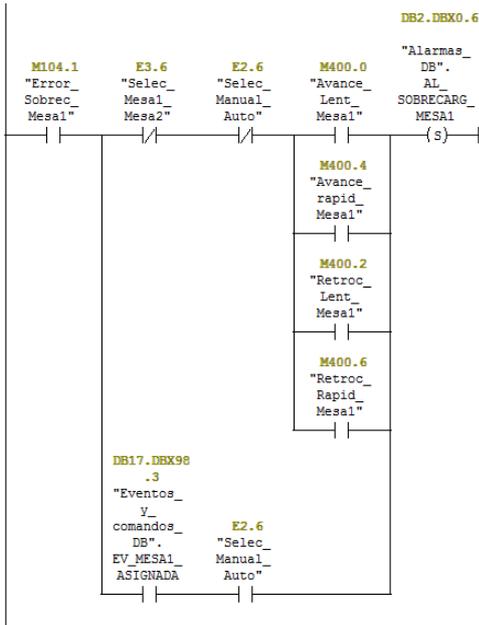


Ilustración 84: FC2 Segmento 9

Segm. 10 : Sobrecarga mesa2

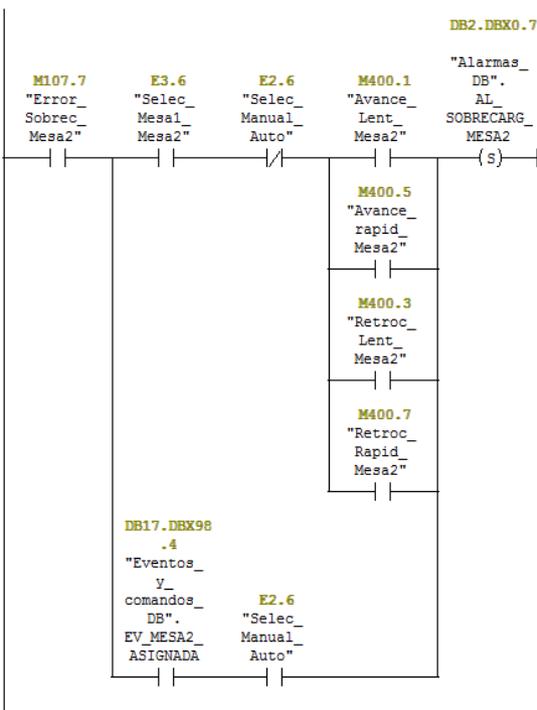


Ilustración 85: FC2 Segmento 10

Segm. 11: Alarma Sobrecarga cilindros campana

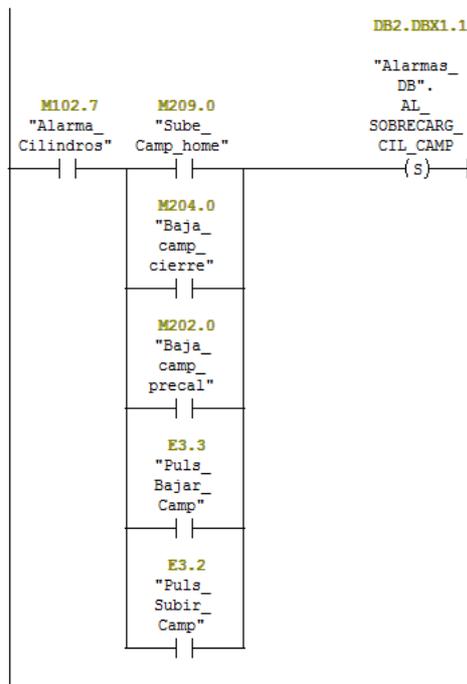


Ilustración 86: FC2 Segmento 11

Segm. 12: Alarma Sobrecarga Variador Camapana

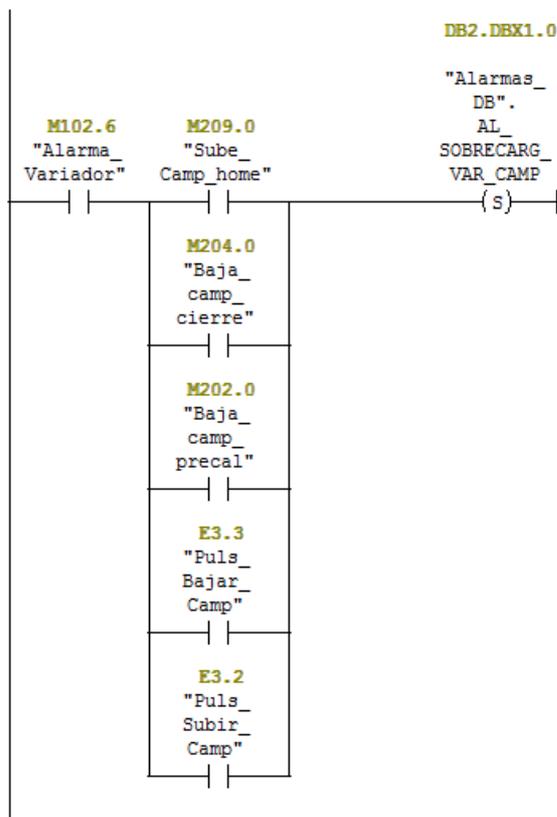


Ilustración 87: FC2 Segmento 12

Segm. 15 : Campana moviendose y la mesa no está dentro

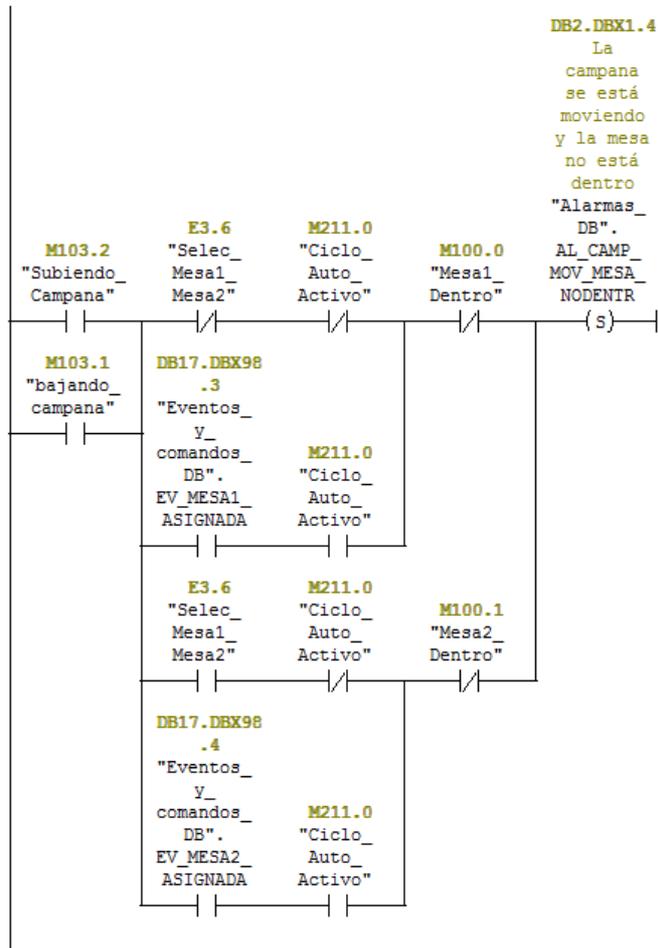


Ilustración 89: FC2 Segmento 15

Segm. 16: Tª superior a los 110°C de todos los termopares

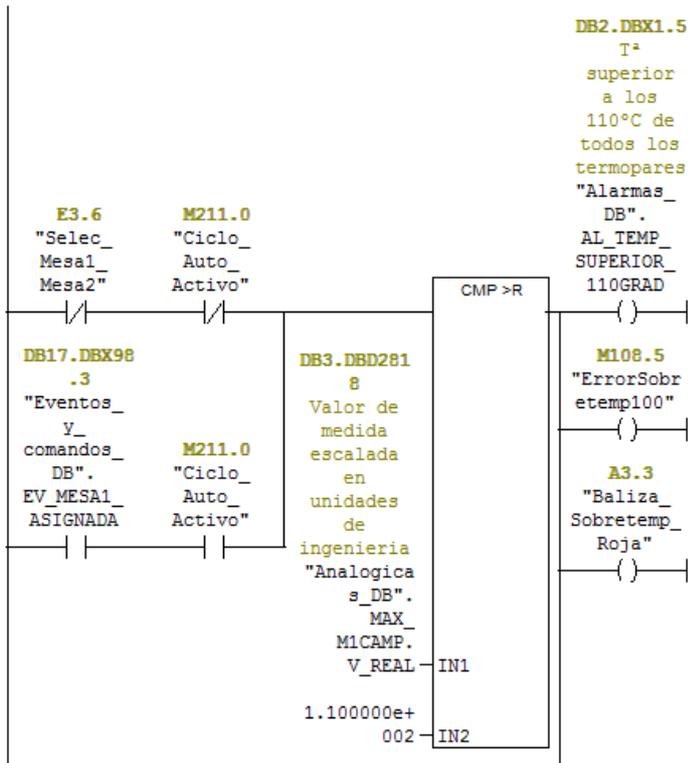


Ilustración 90: FC2 Segmento 16_1

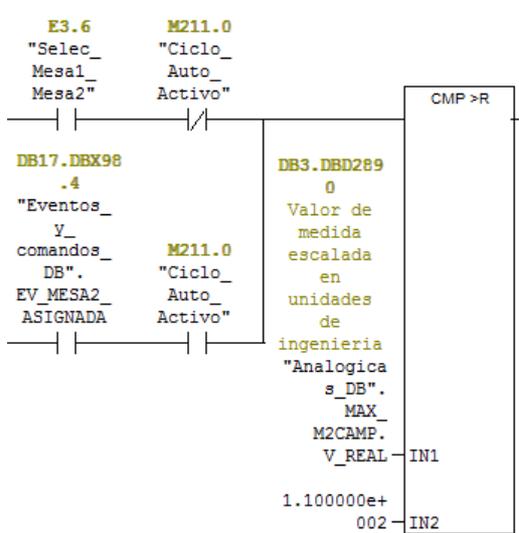
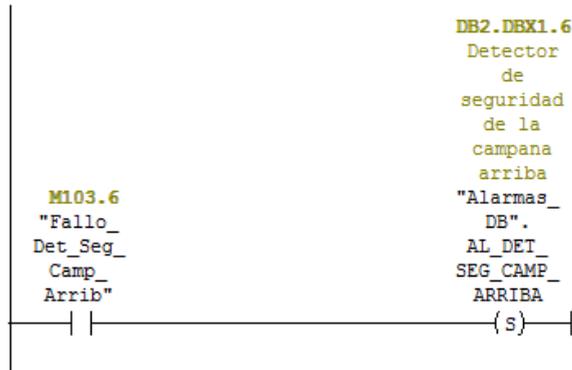


Ilustración 91: FC2 Segmento 16_2

☐ Segm. 17 : Alarma detector de seguridad campana arriba



☐ Segm. 18 : Alarma detector de seguridad campana abajo

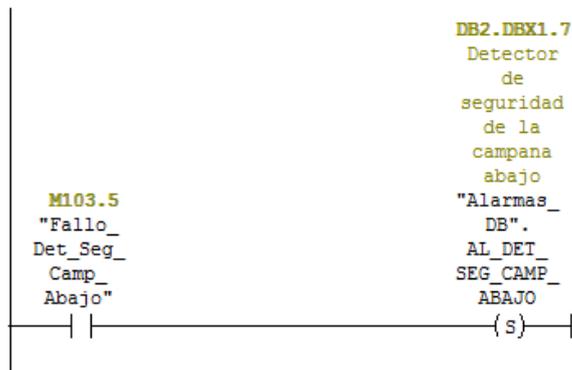
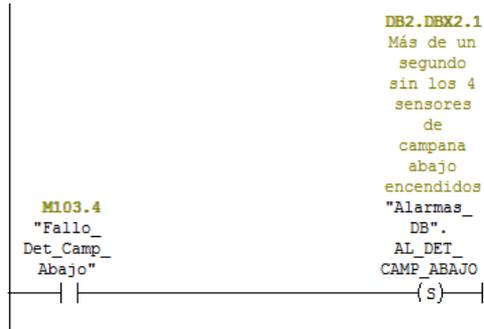


Ilustración 92: FC2 Segmentos 17 y 18

☐ Segm. 19 : Detecta la campana abajo y la campana no está abajo



☐ Segm. 20 : Detecta la campana arriba y la campana no está arriba

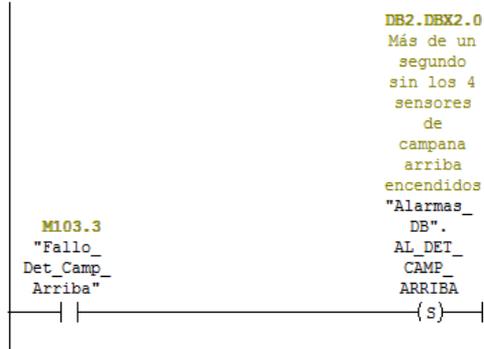
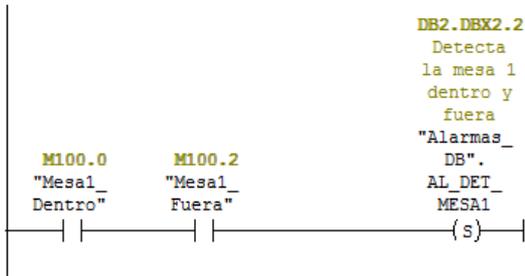
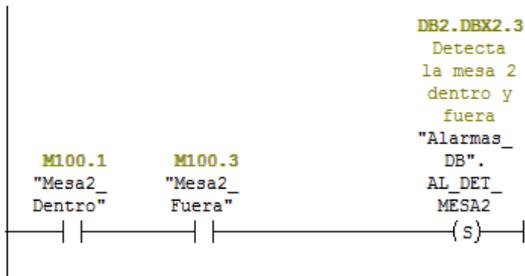


Ilustración 93: FC2 Segmento 19 y 20

☐ Segm. 21 : Alarma mesa 1 dentro y fuera



☐ Segm. 22 : Alarma mesa 2 dentro y fuera



☐ Segm. 23 : Alarma campana arriba y campana abajo

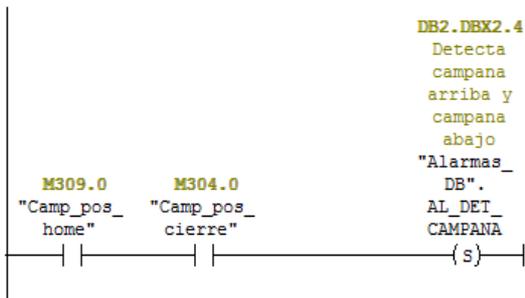
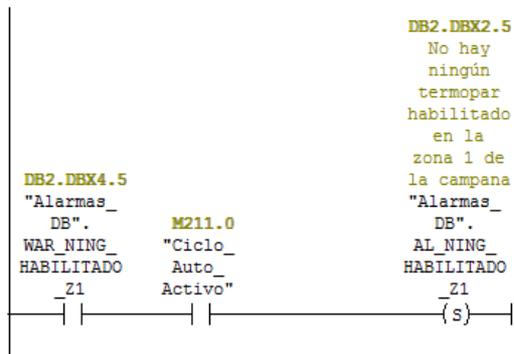


Ilustración 94: FC2 Segmentos 21, 22 y 23

☐ Segm. 24 : Alarma ningún termopar habilitado en la zona 1



☐ Segm. 25 : Alarma ningún termopar habilitado en la zona 2

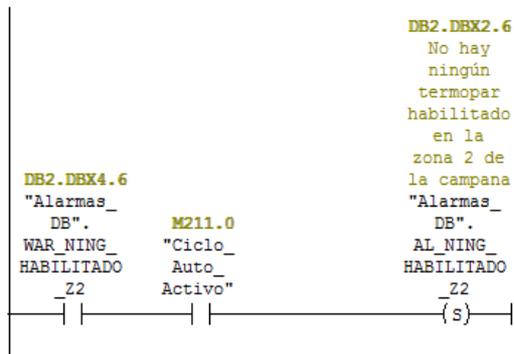
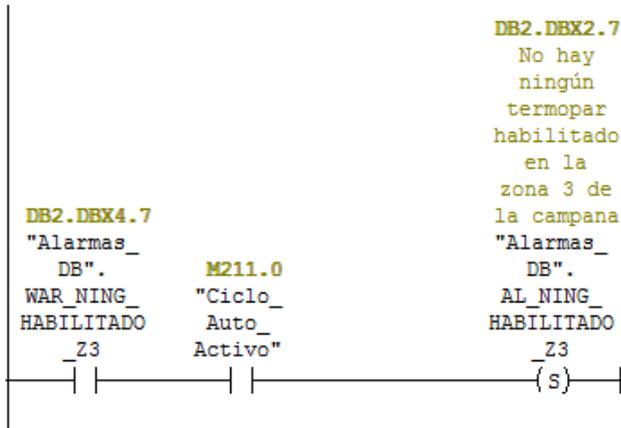


Ilustración 95: FC2 Segmentos 24 y 25

☐ Segm. 26 : Alarma ningún termopar habilitado en la zona 3



☐ Segm. 27 : Alarma ningún termopar habilitado en la zona 4

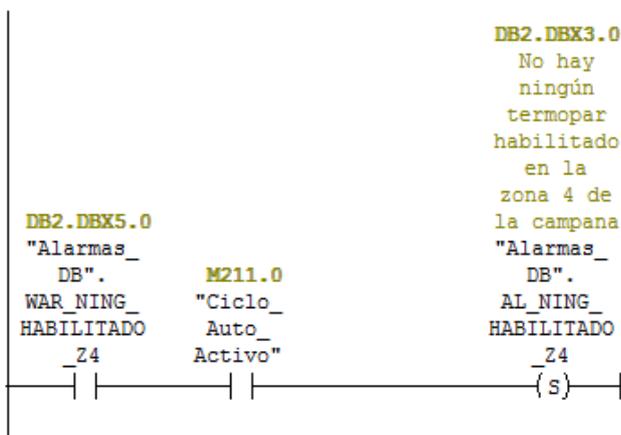


Ilustración 96: FC2 Segmentos 26 y 27

☐ Segm. 28 : advertencia barrera delantera activada



☐ Segm. 29 : Barrera trasera activada



Ilustración 97: FC2 Segmentos 28 y 29

☐ Segm. 30 : Tª superior a los 100°C en la zona 1 de la campana

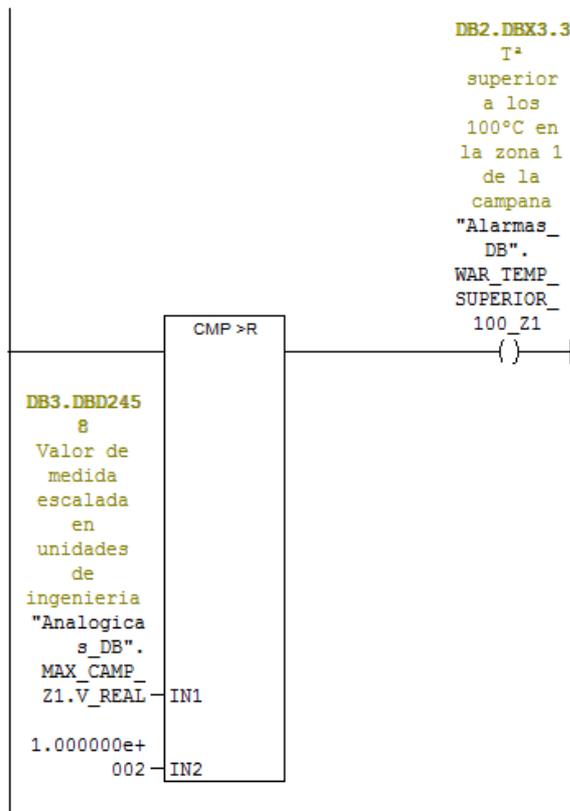


Ilustración 98: FC2 Segmento 30

Segm. 31: Tª superior a los 100°C en la zona 2 de la campana

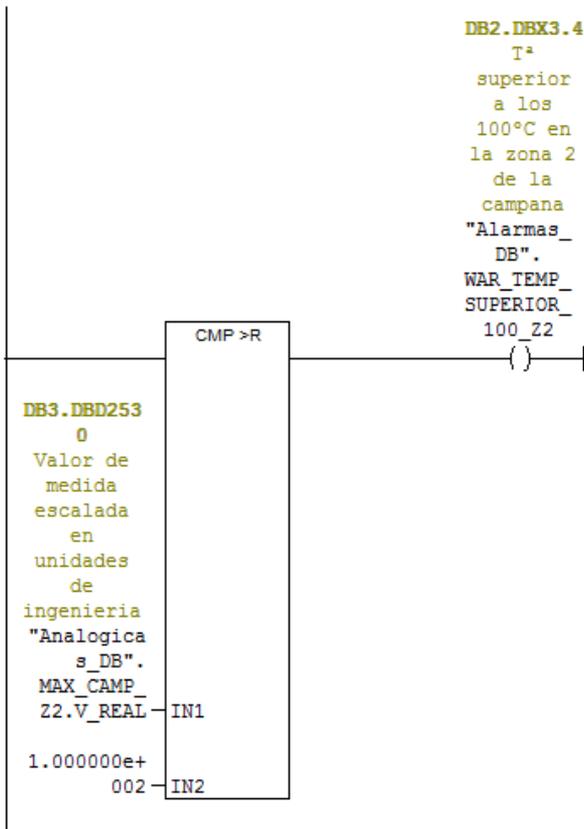


Ilustración 99: FC2 Segmento 31

Segm. 32 : Tª superior a los 100°C en la zona 3 de la campana

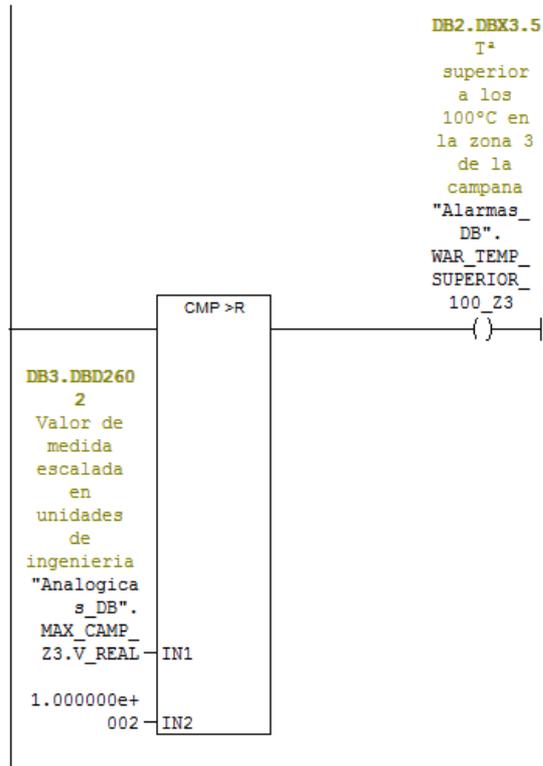
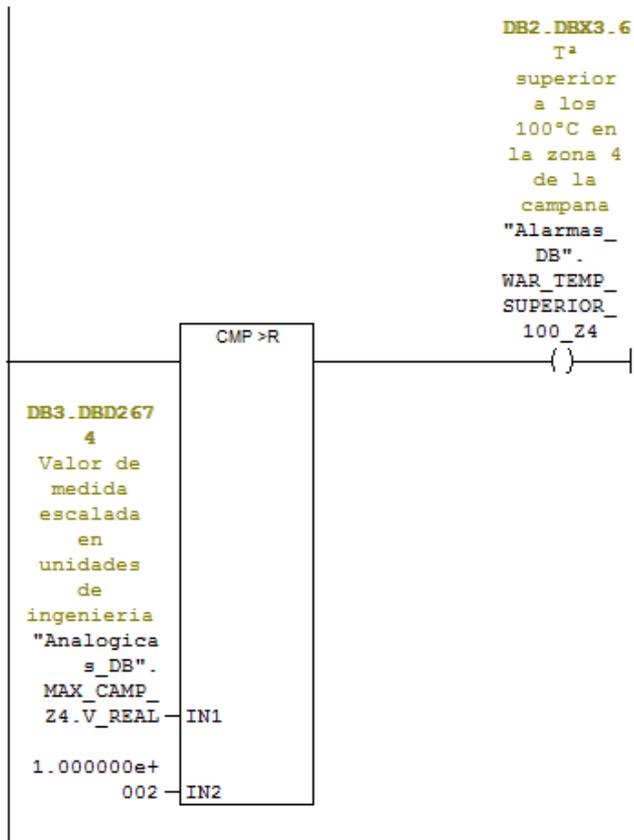


Ilustración 100: FC2 Segmentos 32

Segm. 33 : Tª superior a los 100°C en la zona 4 de la campana



Segm. 34 : Advertencia sobrecarga mesa1



Ilustración 101: FC2 Segmentos 33 y 34

Segm. 35 : Advertencia sobrecarga mesa2



Segm. 36 : Advertencia Refrigeración ok

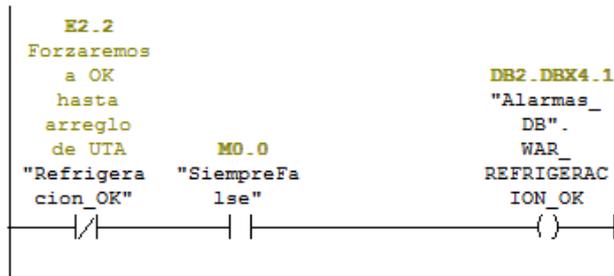


Ilustración 102: FC2 Segmentos 35 y 36

Segm. 38 : Advertencia de Sobrecarga cilindros campana



Segm. 39 : Advertencia sobrecarga variador campana



Ilustración 103: FC2 Segmentos 38 y 39

Segm. 40: Alarmas criticas que paran todo

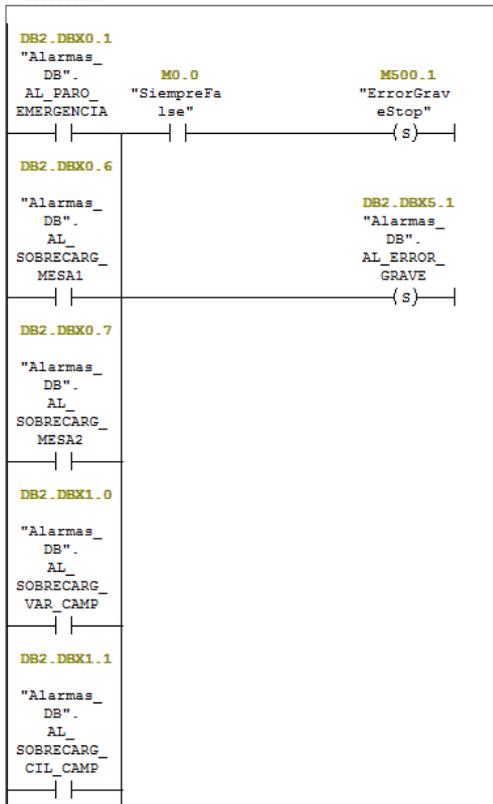


Ilustración 104: FC2 Segmento 40_1

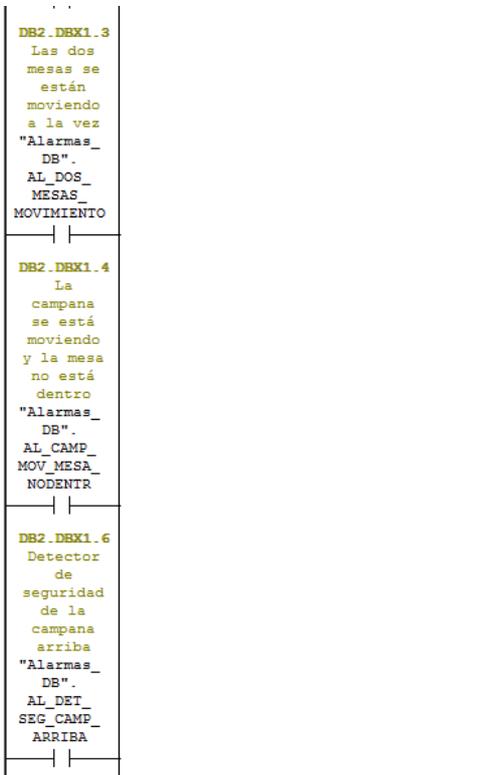


Ilustración 105: FC2 Segmento 40_2

DB2.DBX1.7 Detector de seguridad de la campana abajo "Alarmas_ DB". AL_DET_ SEG_CAMP_ ABAJO
DB2.DBX2.0 Más de un segundo sin los 4 sensores de campana arriba encendidos "Alarmas_ DB". AL_DET_ CAMP_ ARRIBA
DB2.DBX2.1 Más de un segundo sin los 4 sensores de campana abajo encendidos "Alarmas_ DB". AL_DET_ CAMP_ABAJO

Ilustración 106: FC2 Segmento 40_3

DB2.DBX2.2 Detecta la mesa 1 dentro y fuera "Alarmas_ DB". AL_DET_ MESA1
DB2.DBX2.3 Detecta la mesa 2 dentro y fuera "Alarmas_ DB". AL_DET_ MESA2
DB2.DBX2.4 Detecta campana arriba y campana abajo "Alarmas_ DB". AL_DET_ CAMPANA
DB2.DBX2.5 No hay ningún termopar habilitado en la zona 1 de la campana "Alarmas_ DB". AL_NING_ HABILITADO _Z1

Ilustración 107: FC2 Segmento 40_4

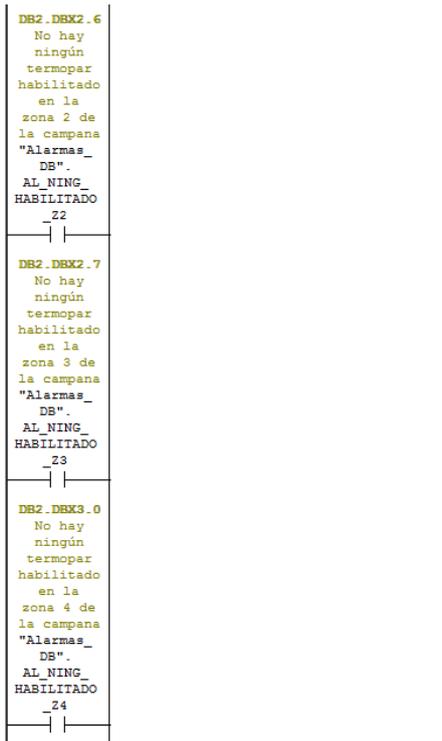


Ilustración 108: FC2 Segmento 40_5

Segm. 41: Reset de las alarmas y de las advertencias

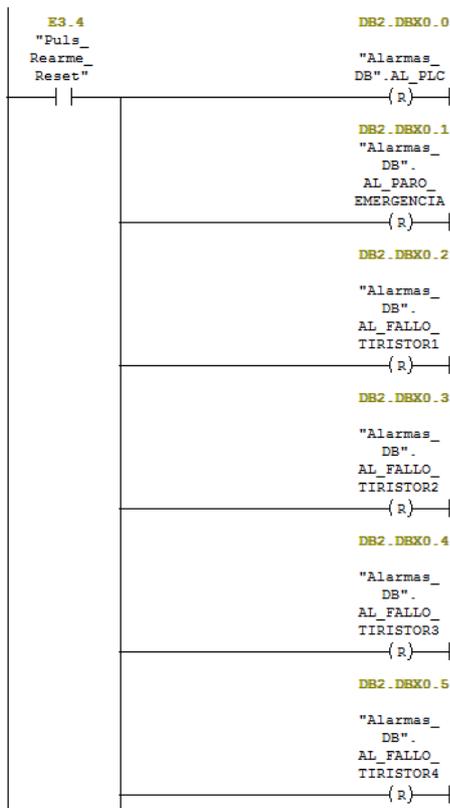


Ilustración 109: FC2 Segmento 41_1

<p>DB2.DBX0.6</p> <p>"Alarmas_ DB". AL_ SOBRECARG_ MESA1</p> <p>{R}</p>
<p>DB2.DBX0.7</p> <p>"Alarmas_ DB". AL_ SOBRECARG_ MESA2</p> <p>{R}</p>
<p>DB2.DBX1.0</p> <p>"Alarmas_ DB". AL_ SOBRECARG_ VAR_CAMP</p> <p>{R}</p>
<p>DB2.DBX1.1</p> <p>"Alarmas_ DB". AL_ SOBRECARG_ CIL_CAMP</p> <p>{R}</p>
<p>DB2.DBX1.3</p> <p>Las dos mesas se están moviendo a la vez</p> <p>"Alarmas_ DB". AL_DOS_ MESAS_ MOVIMIENTO</p> <p>{R}</p>

Ilustración 110: FC2 Segmento 41_2

<p>DB2.DBX1.4</p> <p>La campana se está moviendo y la mesa no está dentro</p> <p>"Alarmas_ DB". AL_CAMP_ MOV_MESA_ NODENTR</p> <p>{R}</p>
<p>DB2.DBX1.6</p> <p>Detector de seguridad de la campana arriba</p> <p>"Alarmas_ DB". AL_DET_ SEG_CAMP_ ARRIBA</p> <p>{R}</p>
<p>DB2.DBX1.7</p> <p>Detector de seguridad de la campana abajo</p> <p>"Alarmas_ DB". AL_DET_ SEG_CAMP_ ABAJO</p> <p>{R}</p>

Ilustración 111: FC2 Segmento 41_3

<p>DB2.DBX2.0 Más de un segundo sin los 4 sensores de campana arriba encendidos "Alarmas_ DB". AL_DET_ CAMP_ ARRIBA {R}</p>
<p>DB2.DBX2.1 Más de un segundo sin los 4 sensores de campana abajo encendidos "Alarmas_ DB". AL_DET_ CAMP_ABAJO {R}</p>
<p>DB2.DBX2.2 Detecta la mesa 1 dentro y fuera "Alarmas_ DB". AL_DET_ MESA1 {R}</p>

Ilustración 112: FC2 Segmento 41_4

<p>DB2.DBX2.3 Detecta la mesa 2 dentro y fuera "Alarmas_ DB". AL_DET_ MESA2 {R}</p>
<p>DB2.DBX2.4 Detecta campana arriba y campana abajo "Alarmas_ DB". AL_DET_ CAMPANA {R}</p>
<p>DB2.DBX2.5 No hay ningún termopar habilitado en la zona 1 de la campana "Alarmas_ DB". AL_NING_ HABILITADO _Z1 {R}</p>

Ilustración 113: FC2 Segmento 41_5

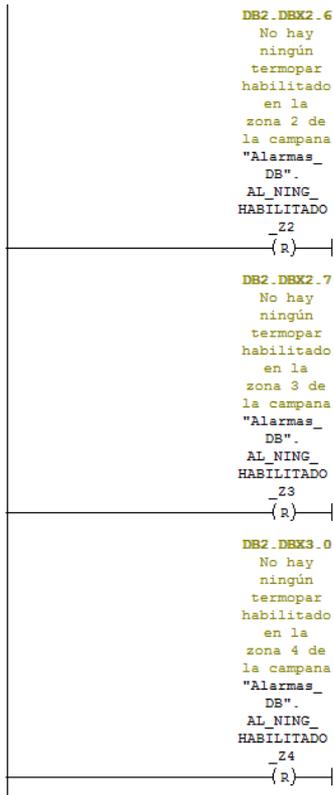


Ilustración 114: FC2 Segmento 41_6

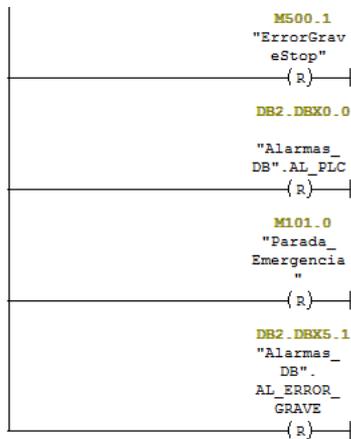
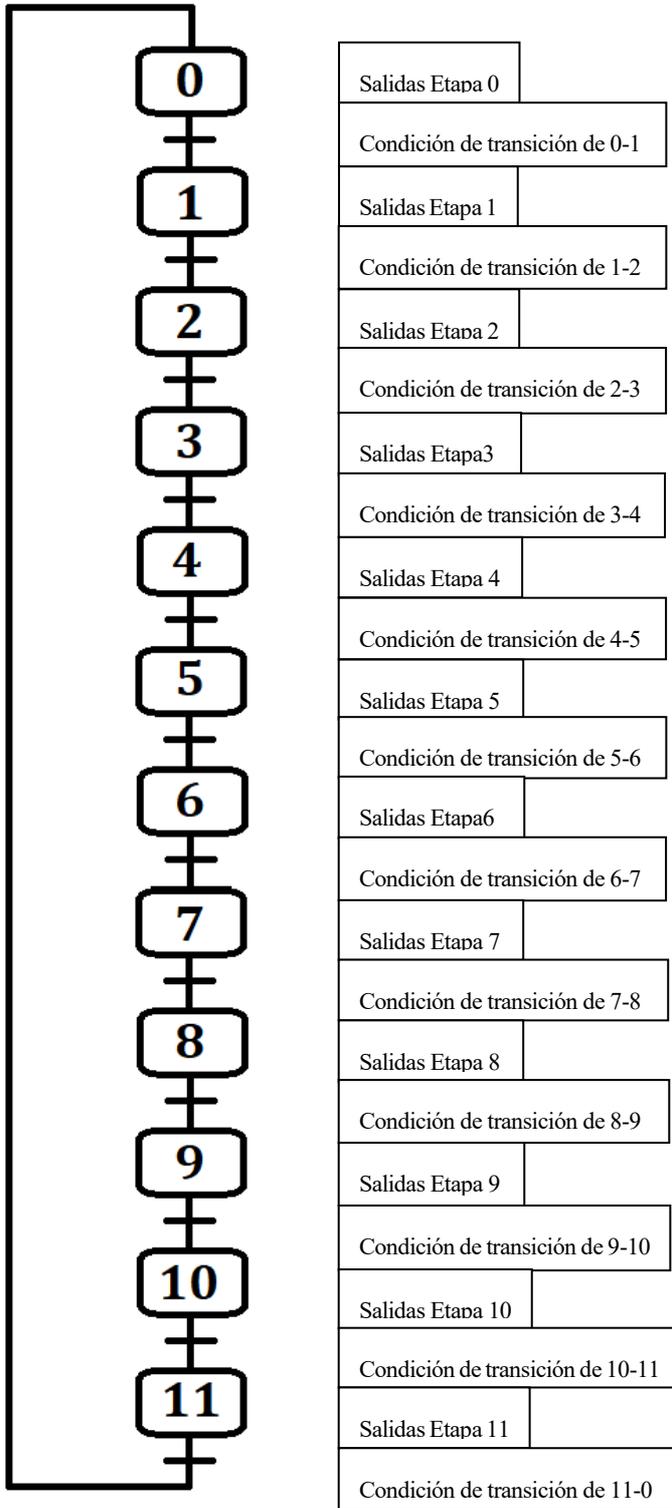


Ilustración 115: FC2 Segmento 41_7

4.2.14 FB10: MÁQUINA DE ESTADOS PARA CICLO AUTOMÁTICO

Para la programación del funcionamiento de un ciclo automático se ha realizado una máquina de estado, la cual sigue la forma indicada en el siguiente diagrama, también podemos observar en la imagen los segmentos donde se han programado cada parte del diagrama.



A continuación, se describirá detalladamente el funcionamiento de un ciclo automático:

Como podemos ver en la descripción del OB1 cuando llamamos a esta función en el segmento 12, es necesario que no haya una parada de emergencia y que el selector del pupitre esté en modo automático para que se active la función, lo primero que hace es asignar la mesa que esté seleccionada en el pupitre en ese instante al ciclo, seguidamente se comprueba que sea seguro arrancar un ciclo automático, para ello se tienen que cumplir una serie de condiciones, que las mesas estén fuera, que la campana esté arriba, que no estén cortadas las barreras fotoeléctricas, que no haya ningún error grave y que el sistema esté en modo automático, este segmento tiene un timar de 1s por seguridad. Una vez hecho esto lo siguiente que hace la función es ver si se cumplen las condiciones de transición de una etapa a otra, y activa las bobinas correspondientes a cada etapa, podemos observar esto en las siguientes imágenes:

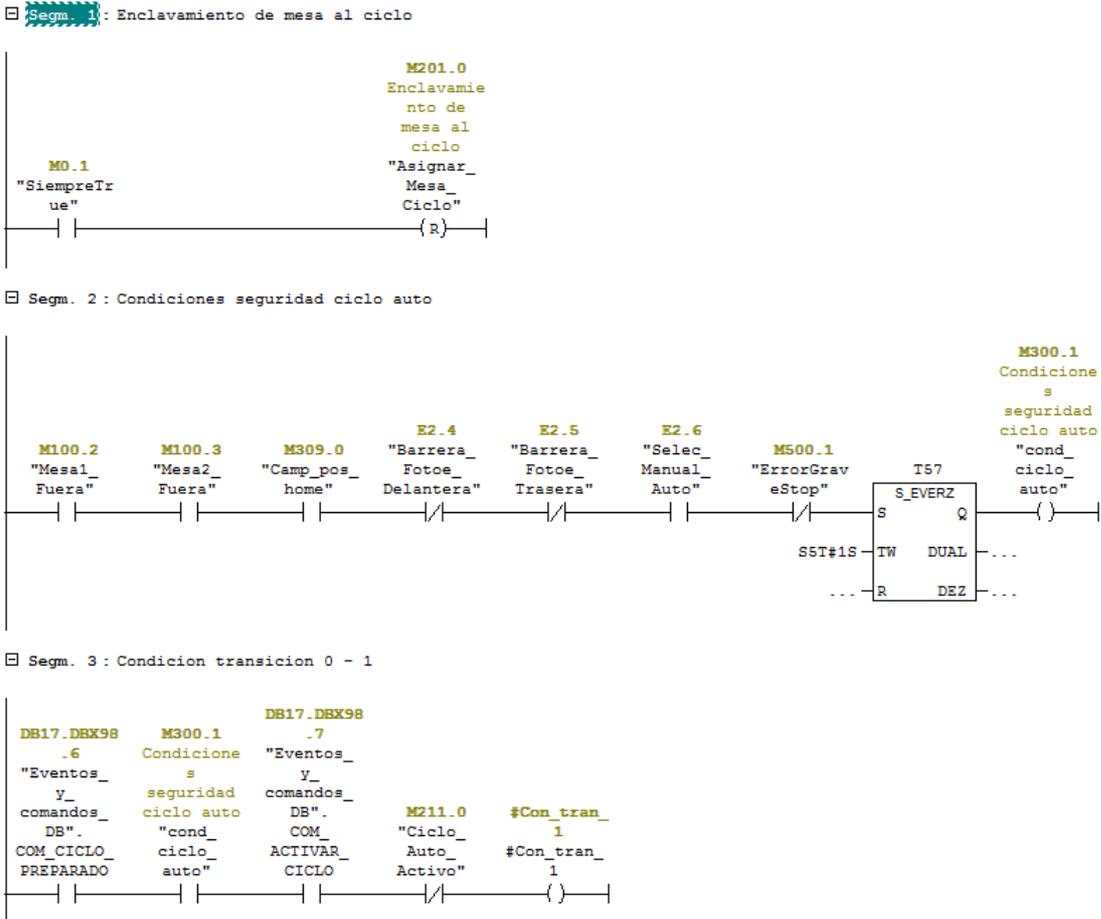


Ilustración 116: FB10 Segmentos 1, 2 y 3

Segm. 4 : Condicion transicion 1 - 2



Segm. 5 : Condicion transicion 2 - 3

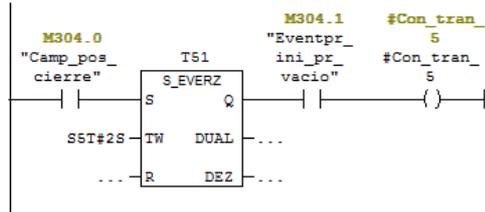


Segm. 6 : Condición de transición de 3 - 4

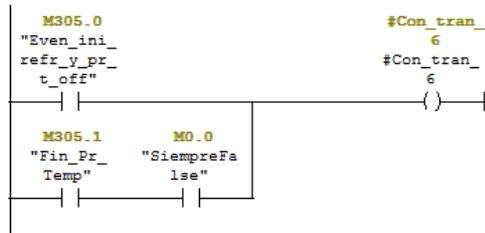


Ilustración 117: FB10 Segmentos 4, 5 y 6

Segm. 7 : Condición de transición de 4 - 5



Segm. 8 : Condición de transición de 5 - 6



Segm. 9 : Condición de transición de 6 - 7

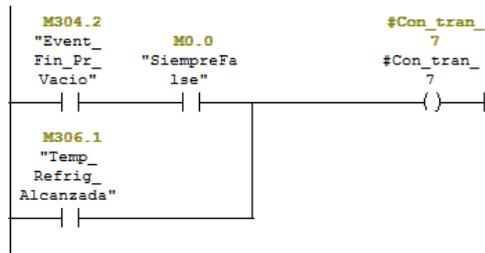


Ilustración 118: FB10 Segmentos 7, 8 y 9

☐ Segm. 10 : Condición de transición de 7 - 8



☐ Segm. 11 : Condición de transición de 8 - 9



☐ Segm. 12 : Condición de transición de 9 - 10



☐ Segm. 13 : Condición de transición de 10 - 11



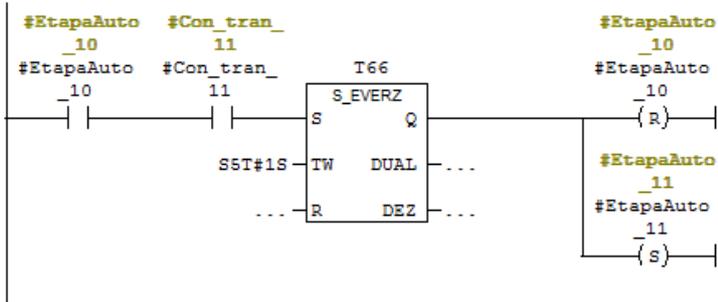
☐ Segm. 14 : Condición de transición de 11 - 0



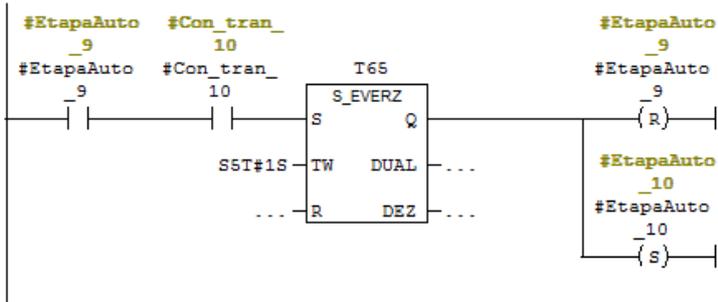
Ilustración 119: FB10 Segmentos 10, 11, 12, 13 y 14

Una vez que la función es consciente de que condiciones de transición se cumplen y cuáles no se activa la etapa en la que se encuentra el ciclo, por ejemplo, la primera activación que ocurriría en el ciclo automático sería la activación de la etapa 1 puesto que el ciclo arranca siempre en la etapa cero, para que esta etapa se active tiene que encontrarse el ciclo en la etapa cero y tiene que activarse la condición de transición de la etapa uno, entonces el ciclo pasaría a no estar en la etapa 0 y estar en la etapa 1, ocurre lo mismo con todas las activaciones. Como se puede comprobar en las siguientes imágenes, el orden de las activaciones está invertido, esto ocurre para asegurarnos que entre la activación de una etapa y de la siguiente ocurre al menos un ciclo del OB1 y se han actualizado todas las señales en las diferentes funciones, para que así si se cumple varias condiciones a la vez la máquina de estado no salte por ejemplo de la etapa 2 a la 4, para evitar esto también se le ha puesto un temporizador a las activaciones, para asegurarnos de que espera al menos un segundo entre una y otra.

☐ Segm. 15 : Activación etapa 11



☐ Segm. 16 : Activación etapa 10



☐ Segm. 17 : Activación etapa 9

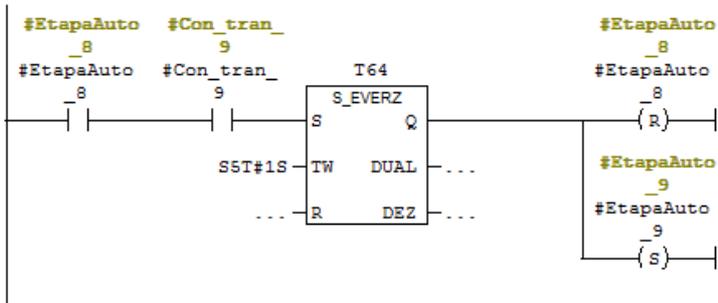
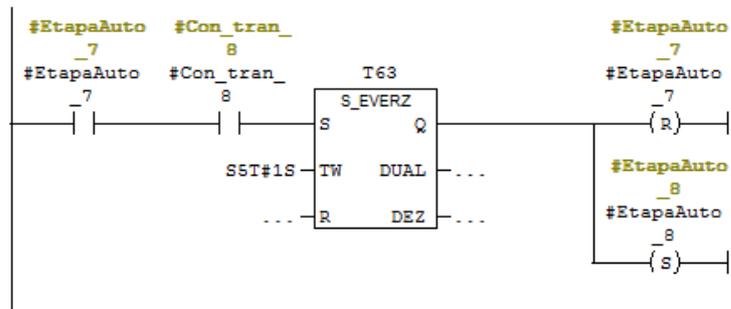
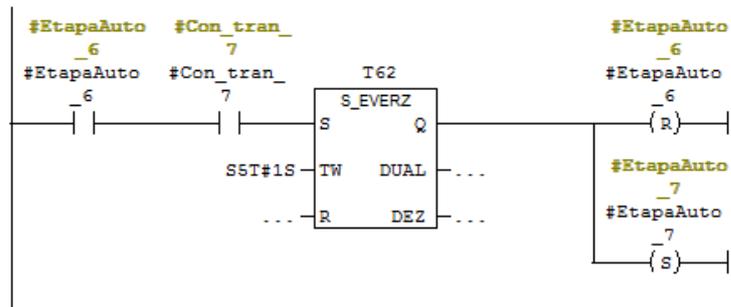


Ilustración 120: FB10 Segmentos 15,16 y 17

Segm. 18 : Activación etapa 8



Segm. 19 : Activación etapa 7



Segm. 20 : Activación etapa 6

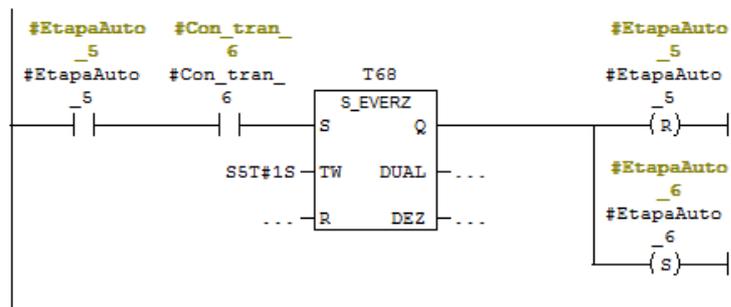
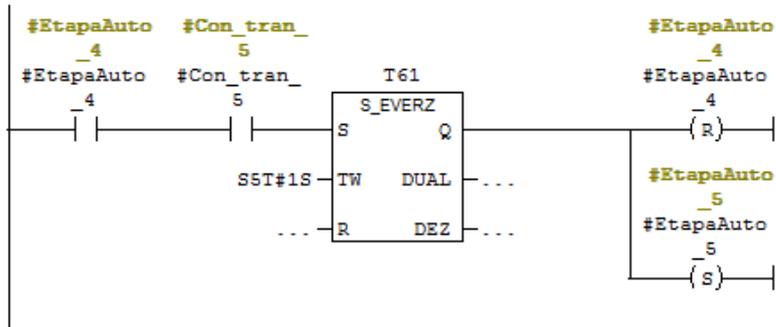
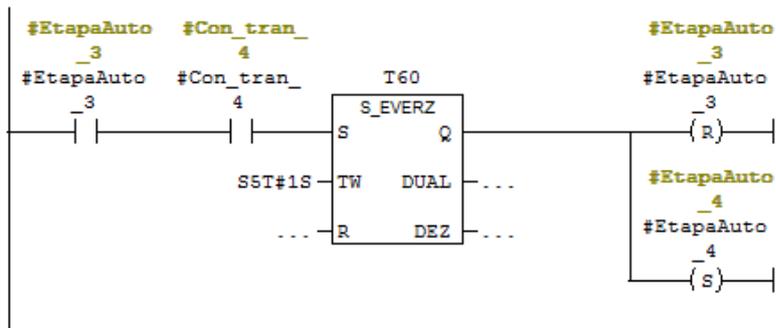


Ilustración 121: FB10 Segmentos 18, 19 y 20

☐ Segm. 21 : Activación etapa 5



☐ Segm. 22 : Activación etapa 4



☐ Segm. 23 : Activación etapa 3

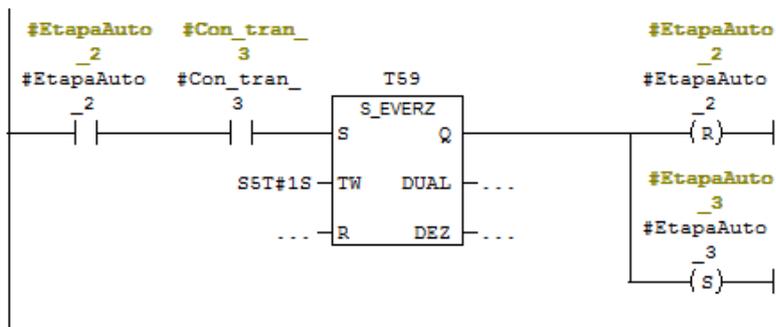
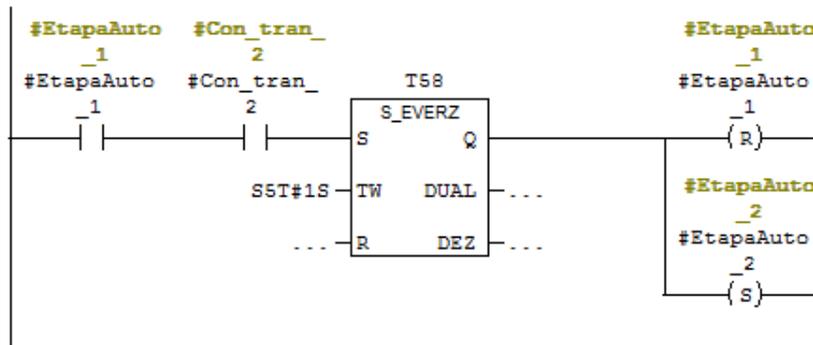
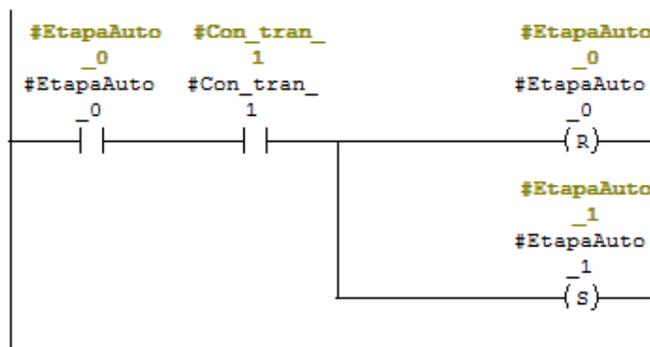


Ilustración 122: FB10 Segmentos 21, 22 y 23

Segm. 24 : Activación etapa 2



Segm. 25 : Activación etapa 1



Segm. 26 : Activación etapa 0

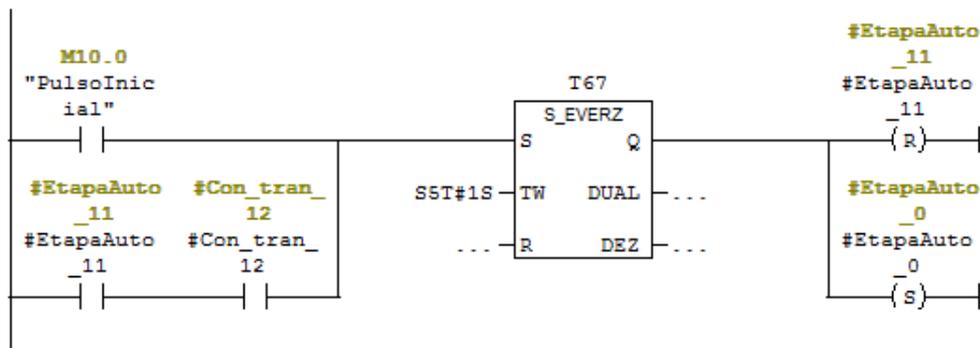


Ilustración 123: FB10 Segmentos 24, 25 y 26

A partir del segmento 27 se activan las salidas de cada etapa, para el mejor entendimiento de la máquina de estado se explicará todo junto a continuación:

Etapa 0: Cuando se activa pone a cero todas las señales que se activan durante la ejecución del ciclo y lo deja reseteado para poder comenzar uno nuevo, esta etapa se activa con el pulso inicial del programa, cuando hay un paso a modo manual, cuando acaba un ciclo automático (estando en la etapa 11 se cumple la condición de transición a la etapa 12) o cuando hay un error grave.

Segm. 27: Salidas Etapa 0

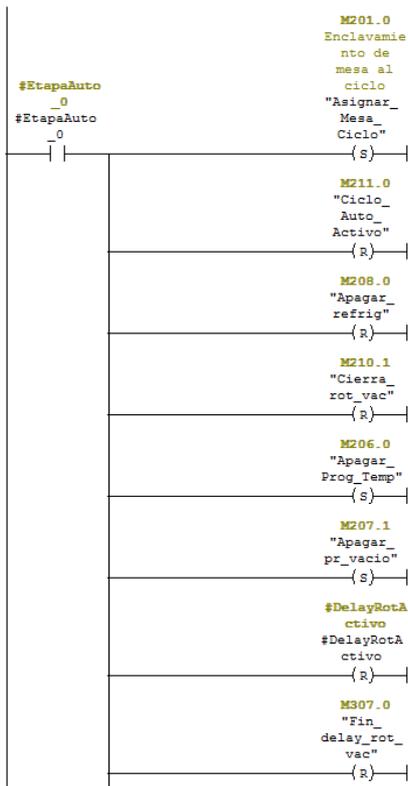


Ilustración 124: FB10 Segmento 27_1

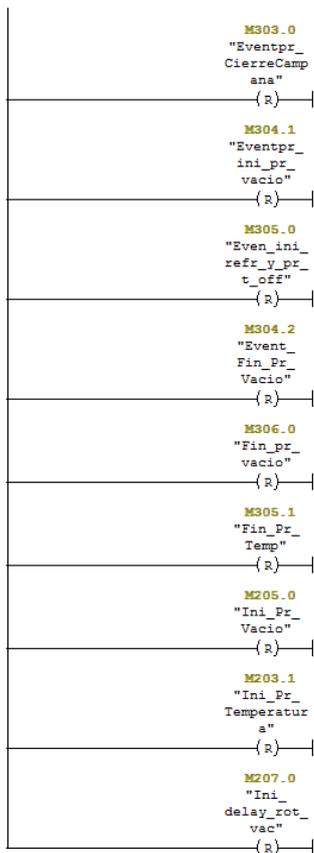


Ilustración 125: FB10 Segmento 27_2

Etapa 1: Esta etapa se activa cuando estando en la etapa 0 se cumplen las condiciones de seguridad de la etapa 1, provoca un reset en algunas señales, inicialmente sólo se reseteaban aquí, pero se añadieron a la etapa 0 por seguridad, ya que es la única que se activa cuando se aborta el ciclo, además de éstas activa la señal de meter la mesa asignada, y la señal indica que el ciclo auto está activo.

Segm. 28 : Salidas Etapa 1

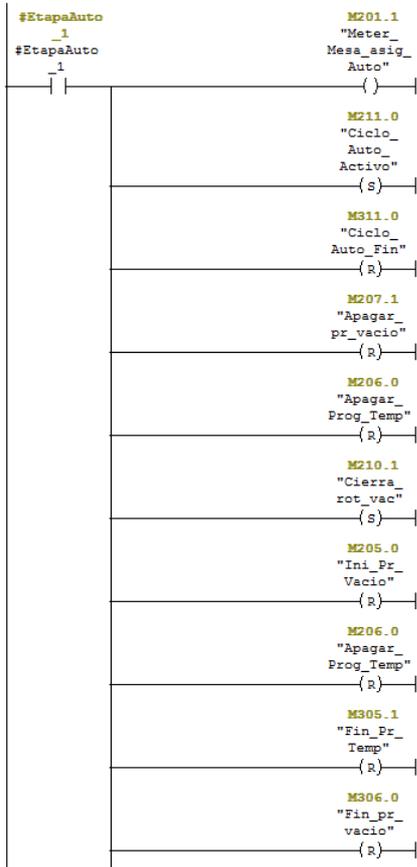


Ilustración 126: FB10 Segmento 28_1

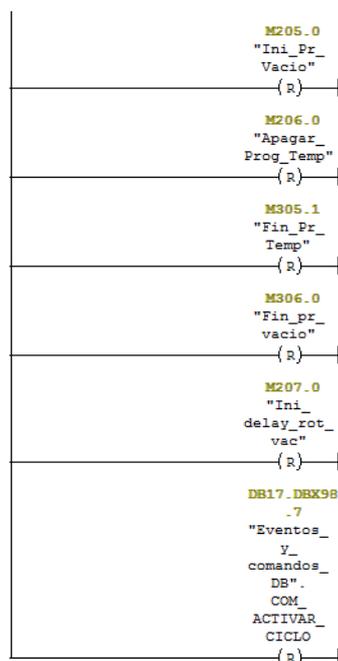


Ilustración 127: FB10 Segmento 28_2

Etapa 2: Se activa cuando estando en la etapa 1 se cumplen las condiciones de seguridad de la etapa 2, esta etapa da la señal para que la campana baje a calentamiento.

Etapa 3: Se activa cuando estando en la etapa 2 se cumplen las condiciones de seguridad de la etapa 3, esta etapa da la señal para que se inicie el programa de temperatura.

Etapa 4: Se activa cuando estando en la etapa 3 se cumplen las condiciones de seguridad de la etapa 4, esta etapa baja la campana hasta el final, y resetea el evento que ordena cerrar la campana y la orden de iniciar el programa de temperatura.

Segm. 29 : Salidas Etapa 2



Segm. 30 : Salidas Etapa 3



Segm. 31 : Salidas Etapa 4

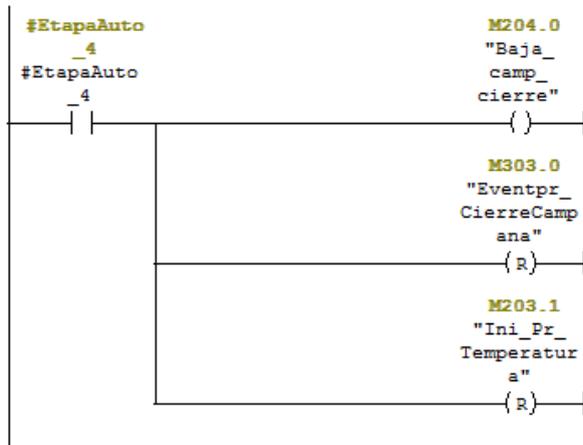
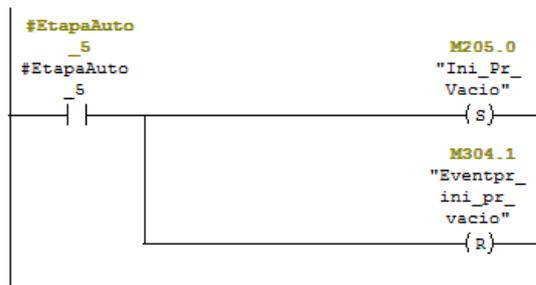


Ilustración 128: FB10 Segmentos 29, 30 y 31

Etapa 5: Se activa cuando estando en la etapa 4 se cumplen las condiciones de seguridad de la etapa 5, esta etapa da la señal para que se inicie el programa de vacío y resetea el evento que ordena iniciar el programa de vacío.

Etapa 6: Se activa cuando estando en la etapa 5 se cumplen las condiciones de seguridad de la etapa 6, esta etapa da la señal para que se apague el programa temperatura y que se inicie la refrigeración y resetea el evento que ordena iniciar la refrigeración y el que ordena iniciar el programa de vacío.

Segm. 32 : Salidas Etapa 5



Segm. 33 : Salidas Etapa 6

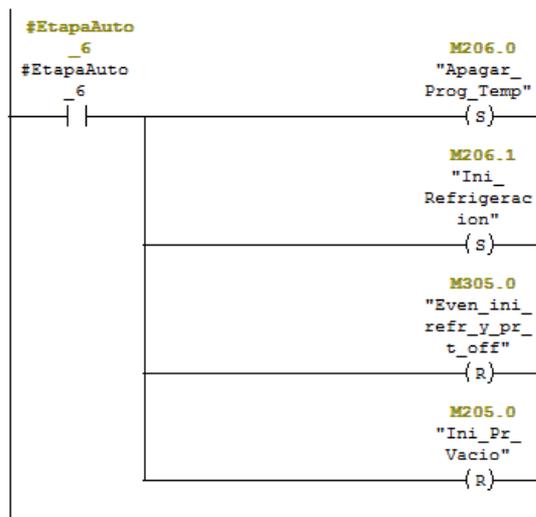
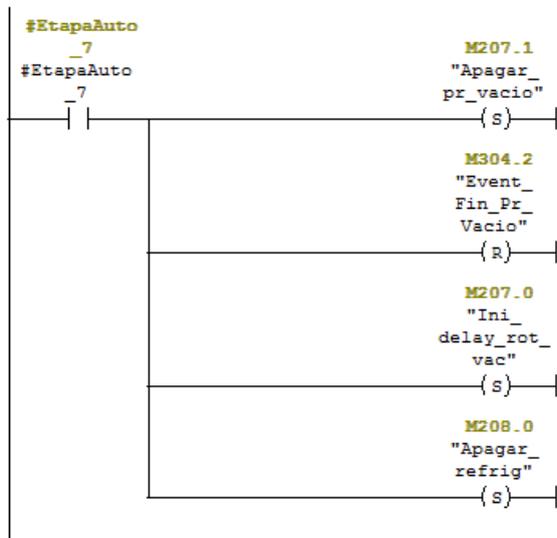


Ilustración 129: FB10 Segmentos 32 y 33

Etapa 7: Se activa cuando estando en la etapa 6 se cumplen las condiciones de seguridad de la etapa 7, esta etapa da la señal para que se apague el programa de vacío e inicia la refrigeración y el delay de rotura de vacío, que es una espera antes de abrir la válvula de rotura de vacío, y resetea el evento que ordena finalizar el programa de vacío.

Etapa 8: Se activa cuando estando en la etapa 7 se cumplen las condiciones de seguridad de la etapa 8, esta etapa da la señal para que se active la válvula de rotura de vacío, resetea la señal que avisa de que el delay de rotura ha terminado.

Segm. 34 : Salidas Etapa 7



Segm. 35 : Salidas Etapa 8

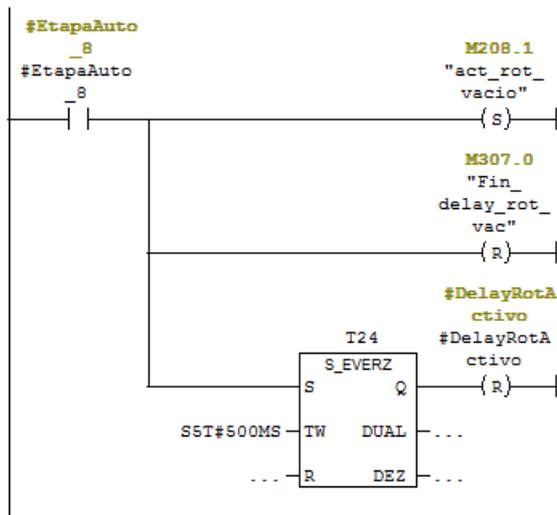


Ilustración 130: FB10 Segmentos 34 y 35

Etapa 9: Se activa cuando estando en la etapa 8 se cumplen las condiciones de seguridad de la etapa 9, esta etapa da la señal para que la campana suba hasta arriba.

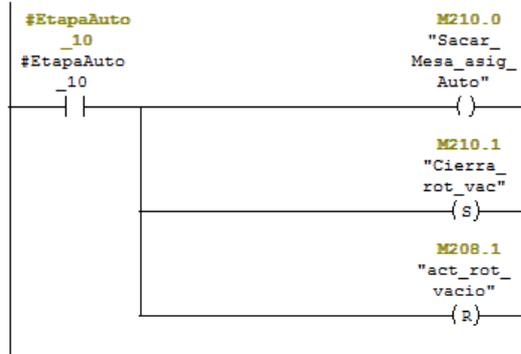
Etapa 10: Se activa cuando estando en la etapa 9 se cumplen las condiciones de seguridad de la etapa 10, esta etapa da la señal para sacar la mesa que está dentro, cierra la válvula de rotura de vacío y resetea la señal que activa la rotura de.

Etapa 11: Se activa cuando estando en la etapa 10 se cumplen las condiciones de seguridad de la etapa 11, esta etapa da la señal de que el ciclo auto ha llegado a su fin.

Segm. 36 : Salidas Etapa 9



Segm. 37 : Salidas Etapa 10



Segm. 38 : Salidas Etapa 11



Ilustración 131: FB10 Segmentos 36, 37 y 38

En el segmento 39, que es el que viene a continuación, se puede observar lo que ocurre cuando se aborta un ciclo, esto puede ocurrir cuando hay un error grave o cuando se pasa el selector del pupitre a manual, cuando esto ocurre se resetean todas las etapas y se activa la etapa 0

Segm. 39 : Reset Errores

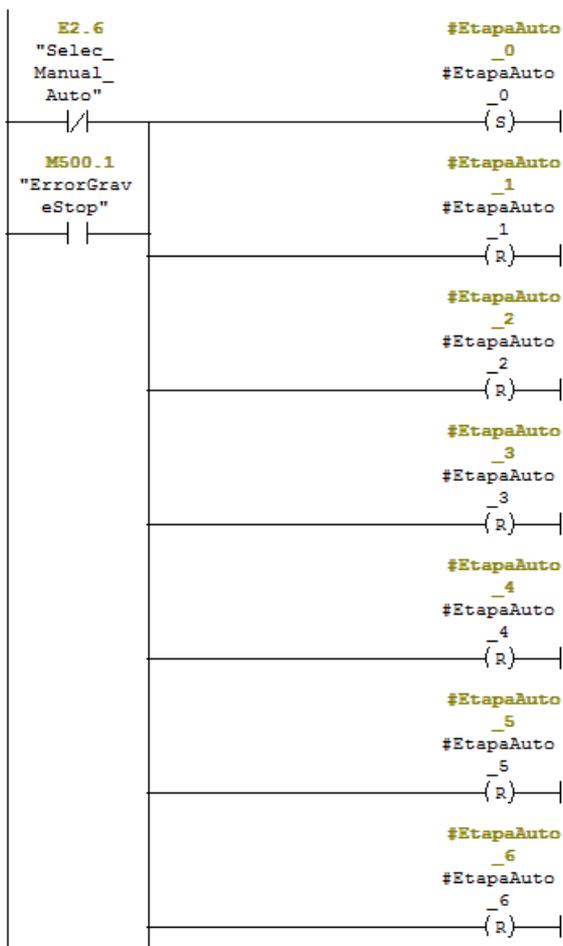


Ilustración 132: FB10 Segmento 39_1

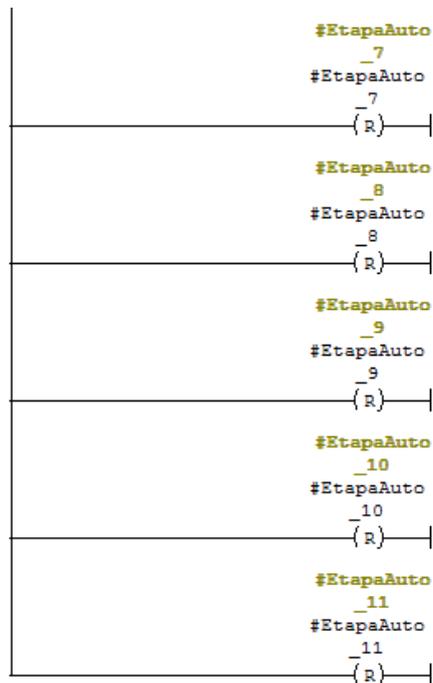
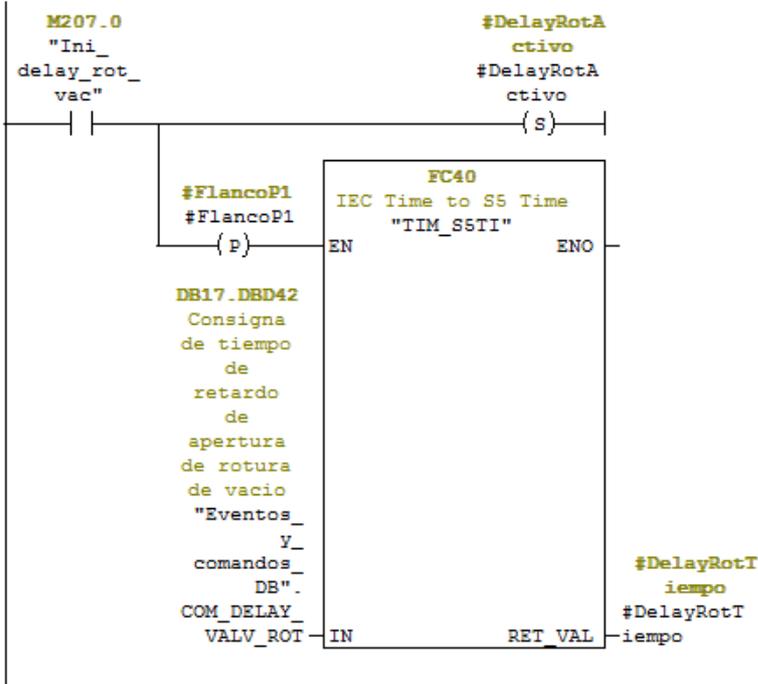


Ilustración 133: FB10 Segmento 39_2

En los últimos segmentos de esta función se controla el delay de rotura, es un retardo que tiene la válvula de rotura, este retardo está controlado desde el SCADA y el operario puede cambiar la consigna según le convenga, en los siguientes segmentos capturamos el flanco positivo de la orden de iniciar el delay y convertimos la consigna en S5TIME, iniciamos un temporizador hacia atrás y mantenemos el delay de rotura hasta que este temporizador llega a cero y mientras esto ocurre le devolvemos una señal al SCADA del tiempo restante que queda.

Segm. 40 : Delay de rotura de vacio



Segm. 41 : Titulo:

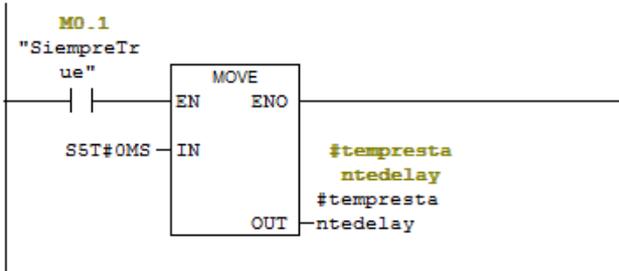
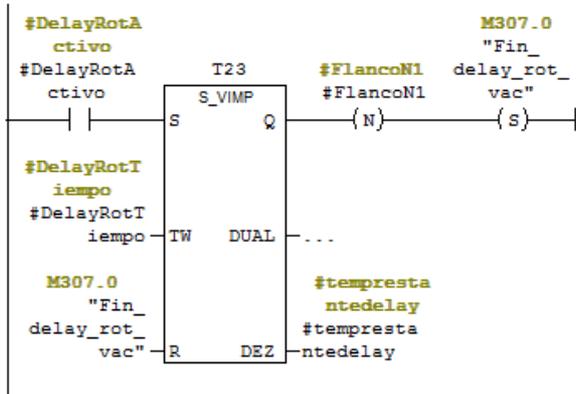


Ilustración 134: FB10 Segmentos 40 y 41

Segm. 42 : Temporizacion de delay rotura



Segm. 43 : Conversicon a TIME

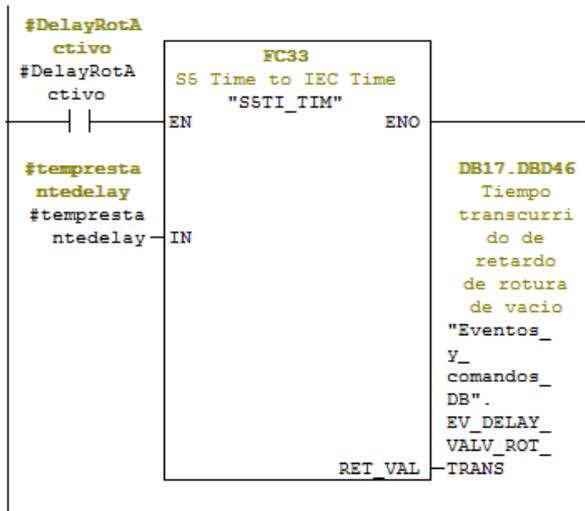
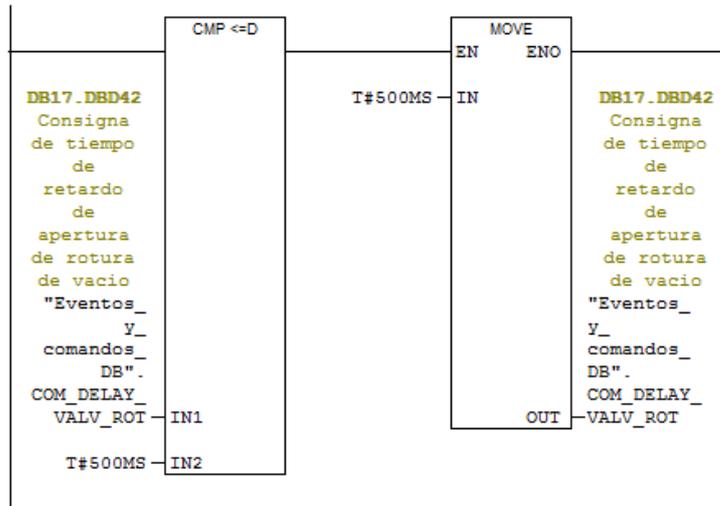


Ilustración 135: FB10 Segmentos 42 y 43

Segm. 44 : Calculo delay rotura



Segm. 45 : Calculo delay rotura

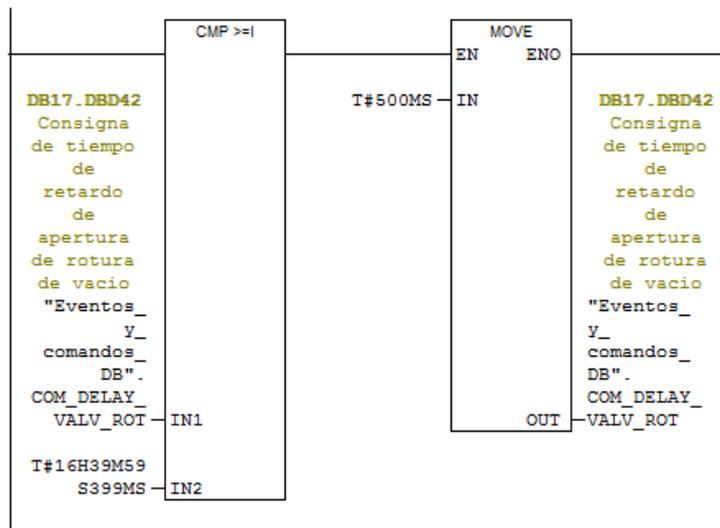


Ilustración 136: FB10 Segmentos 44 y 45

4.2.15 FC10: ESCRITURA DE INFORMACIÓN EN EL SCADA

En la siguiente función se vuelcan todos los datos necesarios para la buena comprensión de lo que ocurre en la máquina, todas esas señales son las que se mostrarán por la pantalla del SCADA para que el operario conozca todos los estados en cada instante.

FC10 : Título:

Comentario:

☐ **Segm. 1 :** EV Campana Arriba



☐ **Segm. 2 :** EV Campana Abajo



☐ **Segm. 3 :** EV Campana en posición de Precalentamiento



Ilustración 137: FC10 Segmentos 1, 2 y 3

Segm. 4 : EV Mesa 1 Fuera



Segm. 5 : EV Mesa 2 Fuera



Segm. 6 : EV Mesa 1 Dentro

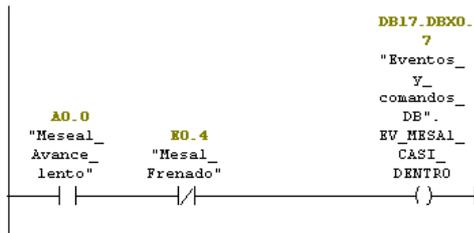


Segm. 7 : EV Mesa 2 Dentro

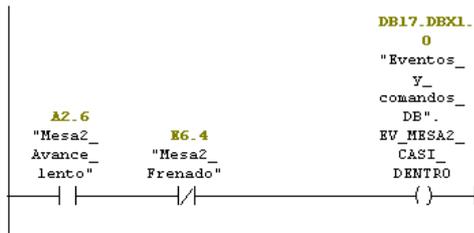


Ilustración 138: FC10 Segmentos 4, 5, 6 y 7

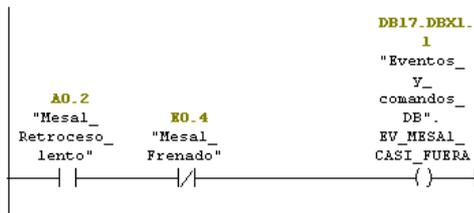
☐ Segm. 8 : EV mesa 1 Casi dentro



☐ Segm. 9 : EV Mesa 2 Casi dentro



☐ Segm. 10 : EV Mesa 1 Casi Fuera



☐ Segm. 11 : EV Mesa 2 Casi Fuera

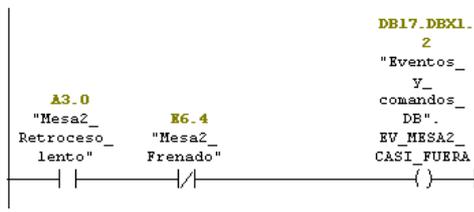


Ilustración 139: FC10 Segmentos 8, 9, 10 y 11

☐ Segm. 12 : EV Mesa 1 Avance Lento



☐ Segm. 13 : EV Mesa 1 Avance Rápido



☐ Segm. 14 : EV Mesa 2 Avance Lento



Ilustración 140: FC10 Segmentos 12, 13 y 14

☐ **Segm. 15** : EV Mesa 2 Avance Rápido



☐ **Segm. 16** : EV Mesa 1 Retroceso Lento

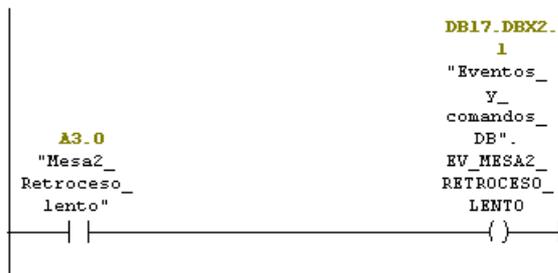


☐ **Segm. 17** : EV Mesa 1 Retroceso Rápido



Ilustración 141: FC10 Segmentos 15, 16 y 17

☐ **Segm. 18** : EV Mesa 2 Retroceso Lento



☐ **Segm. 19** : EV Mesa 2 Retroceso Rápido



☐ **Segm. 20** : EV Vacío actuando sobre mesa 1



☐ **Segm. 21** : EV Vacío actuando sobre mesa 2



Ilustración 142: FC10 Segmentos 18, 19, 20 y 21

☐ Segm. 22 : EV Abriendo valvula de rotura



☐ Segm. 23 : EV Cerrando valvula de rotura



☐ Segm. 24 : EV Ciclo automático encendido

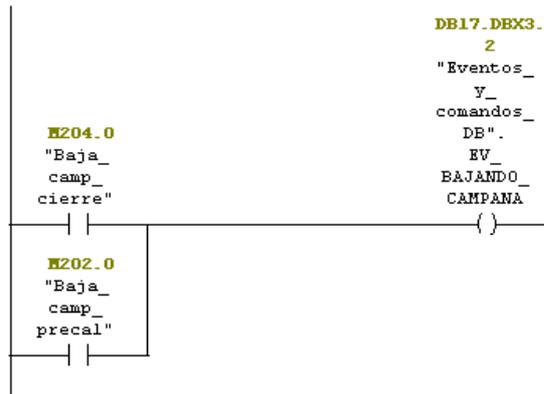


Ilustración 143: FC10 Segmentos 22, 23 y 24

☐ **Segm. 25** : EV Subiendo Campana



☐ **Segm. 26** : EV Bajando Campana



☐ **Segm. 27** : EV Refrigeración en marcha



Ilustración 144: FC10 Segmentos 25, 26 y 27

☐ **Segm. 28** : EV Programa Automatico encendido



☐ **Segm. 29** : EV Resistencias ON



☐ **Segm. 30** : Evento de fin de ciclo



Ilustración 145: FC10 Segmentos 28, 29 y 30

Segm. 31: Evento de Refrigeracion UTA preparada



Segm. 32: Evento bomba de vacio ON



Segm. 33: EV OP FREECOOLING

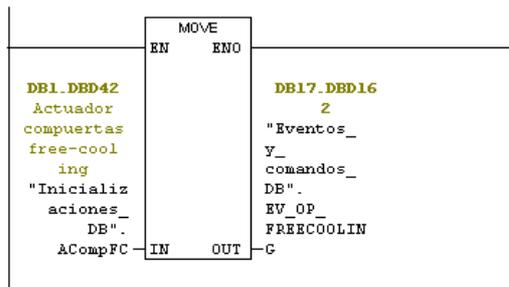
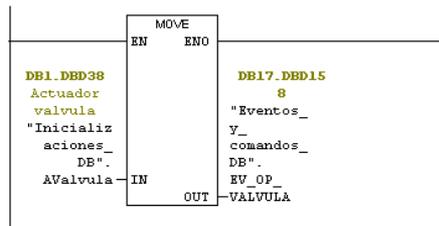


Ilustración 146: FC10 Segmentos 31, 32 y 33

Segm. 34: EV OP VALVULA



Segm. 35: EV SP REFRIGERACION

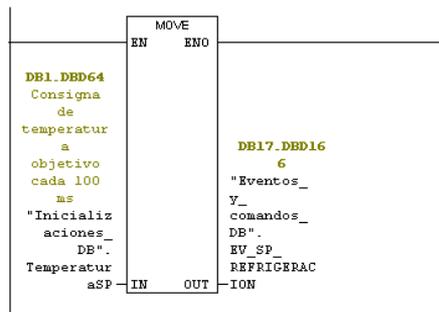
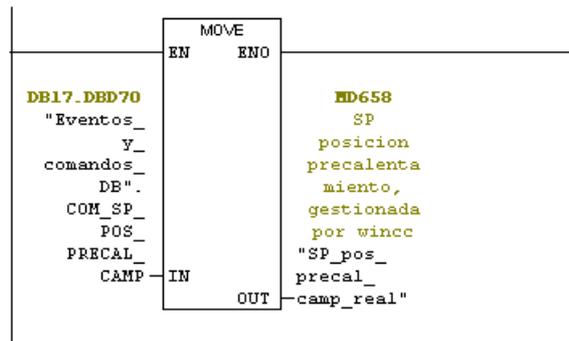


Ilustración 147: FC10 Segmentos 34 y 35

☐ **Segm. 1:** SP posicion precalentamiento campana



☐ **Segm. 2:** Marcha rotura de vacio



☐ **Segm. 3:** Marcha bomba de vacio



4.2.16 FC8: ESCRITURA DE SALIDAS

En esta función se activas las salidas del PLC, volcando las marcas del programa en las bobinas de salida que les corresponden.

FC8 : Título:

Escritura de Salidas

Segm. 1 : Indicacion de mesal fuera para porticos



Segm. 2 : Indicacion de mesa2 fuera para porticos



Ilustración 148: FC8 Segmentos 1 y 2

Segm. 3 : Activacion de sirena al mover mesa

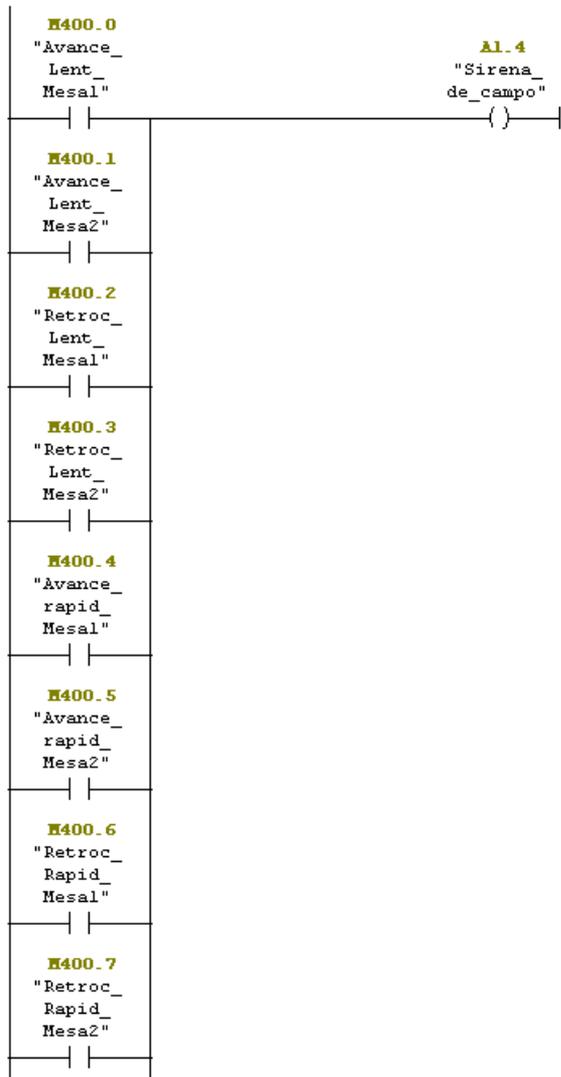
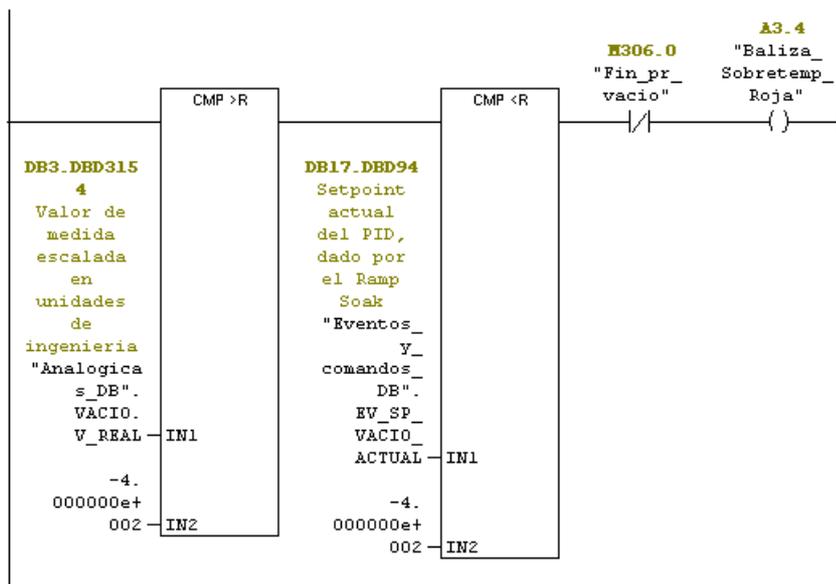


Ilustración 149: FC8 Segmento 3

☐ **Segm. 4** : Activar Baliza de vacio

Esto se enciende en el PLC de Honeywell cuando el setpoint es menor de -400 y el vacio pasa a ser menor de -400 (se enciende el tiempo desde -400 a SP_vacio escalon actual



☐ **Segm. 5** : Mesa 1 avance lento

Activación de contactores de las mesas



☐ **Segm. 6** : Mesa 2 avance lento



Ilustración 150: FC8 Segmentos 4 y 5

☐ **Segm. 9** : Mesa 1 avance rapido



☐ **Segm. 10** : Mesa 2 avance rapido



☐ **Segm. 11** : Mesa 1 retroceso rapido



☐ **Segm. 12** : Mesa 2 retroceso rápido



Ilustración 151 FC8 Segmentos 9, 10 y 11

☐ **Segm. 13** : Resets

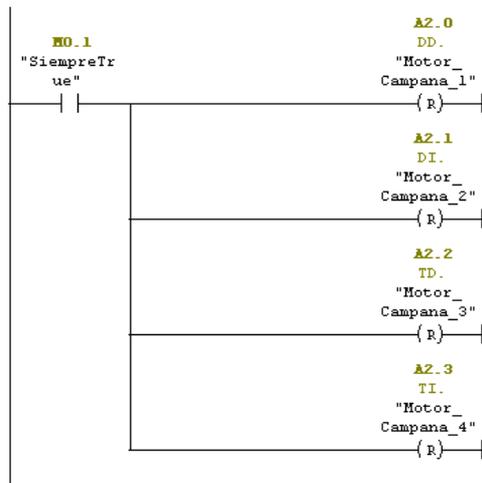


Ilustración 152 FC8 Segmento 13

Segm. 14 : Maniobra activación de motores subir y bajar campana

Es importante comprobar en máquina la coincidencia detector - motor en la campana

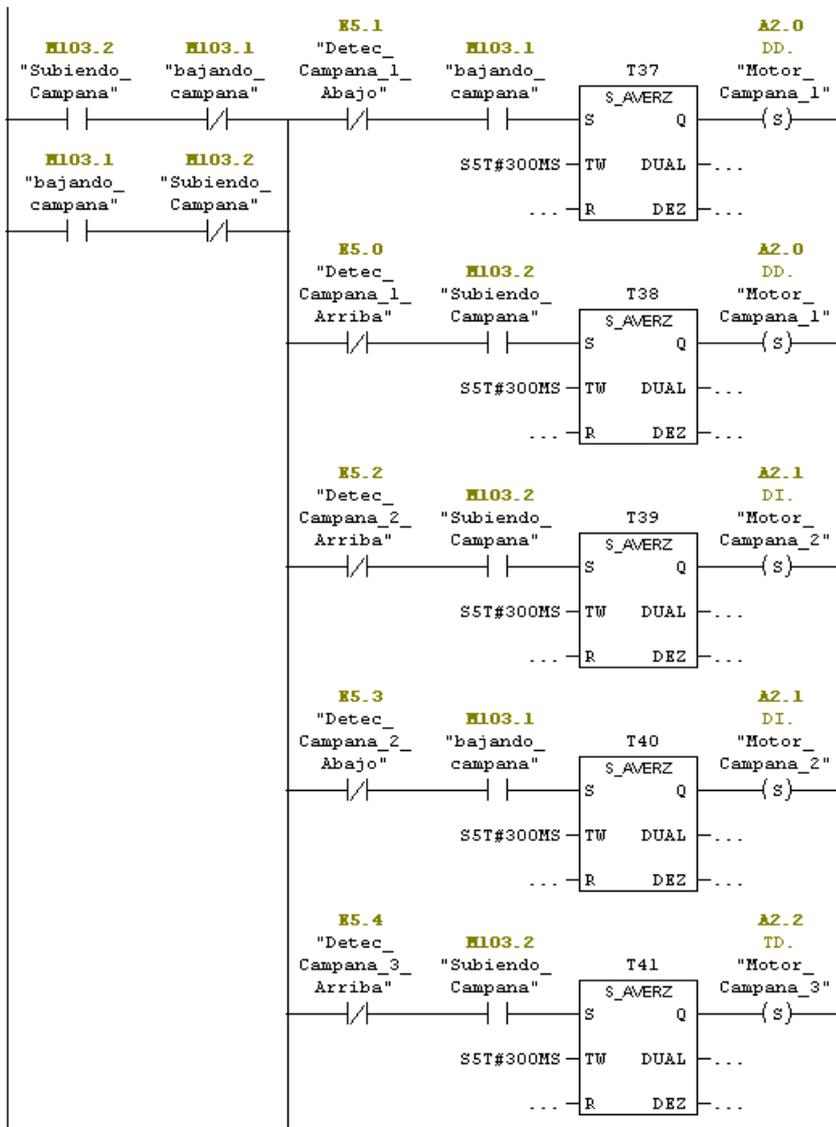


Ilustración 153 FC8 Segmento 14_1

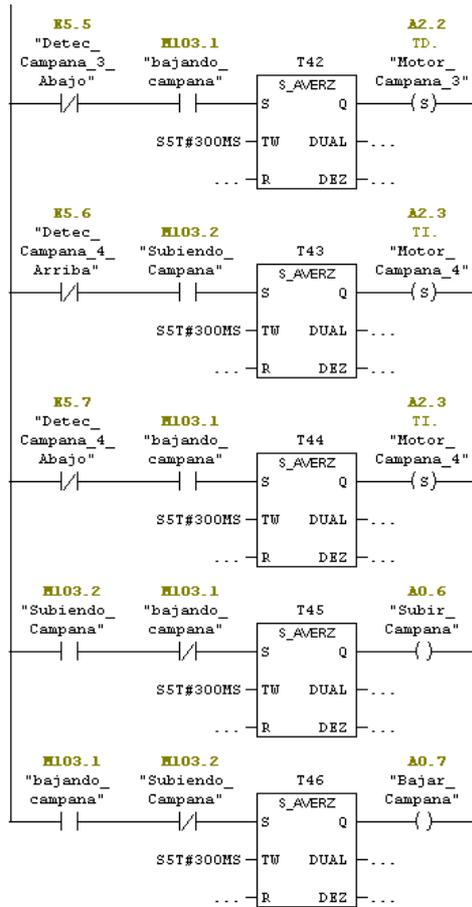
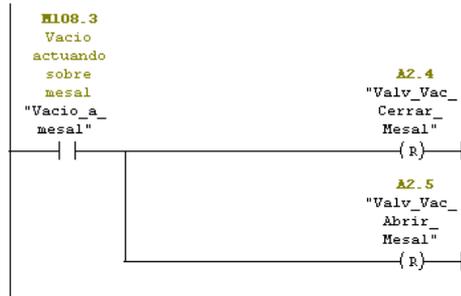


Ilustración 154 FC8 Segmento 14_2

Segm. 15: Actuador valvulas vacio para la mesa 1

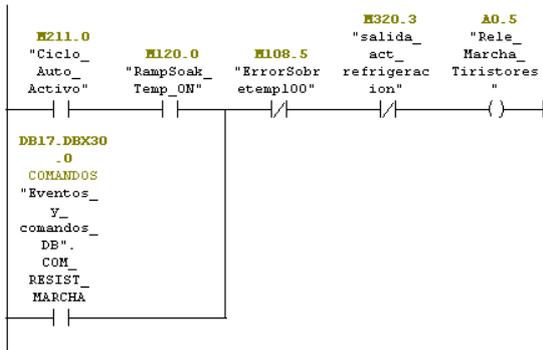


Segm. 16: Actuador valvulas vacio para la mesa 2

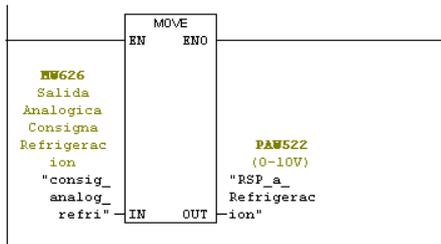


Ilustración 155 FC8 Segmentos 15 y 16

Segn. 17 : Marcha Réle de Tiristores



Segn. 18 : Escritura de salida de consigna refrigeracion



Segn. 19 : Activación UTA Refrigeración

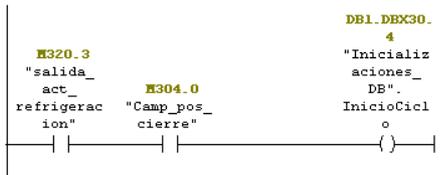


Ilustración 156 FC8 Segmentos 17, 18 y 19

☐ **Segm. 20** : Valvula de rotura de vacio



☐ **Segm. 21** : Salida activacion bomba de vacio



☐ **Segm. 22** : Salida de ciclo auto on para nodemcu



☐ **Segm. 23** : Escritura de consigna de vacio

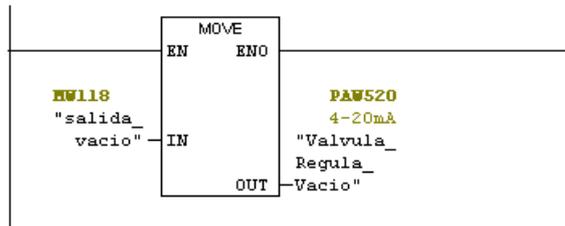
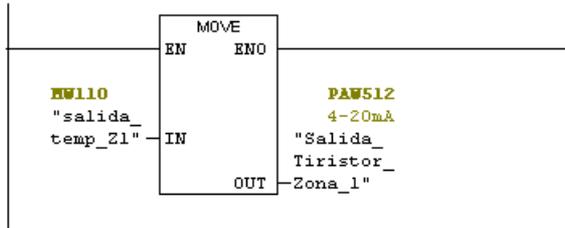
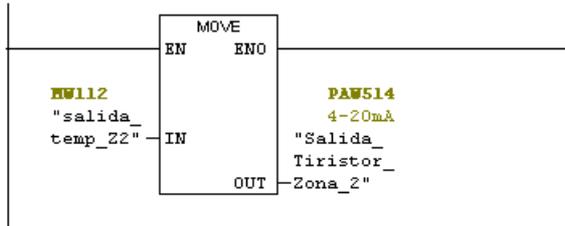


Ilustración 157 FC8 Segmentos 20, 21, 22 y 23

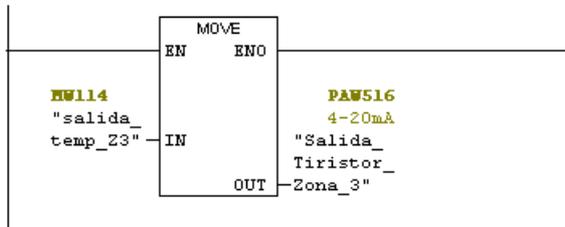
☐ **Segm. 24** : Escritura de consignas de temperatura



☐ **Segm. 25** : Escritura de consignas de temperatura



☐ **Segm. 26** : Escritura de consignas de temperatura



☐ **Segm. 27** : Escritura de consignas de temperatura

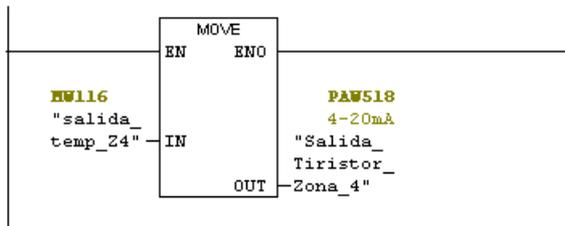


Ilustración 158 FC8 Segmentos 24,25, 26 y 27

☐ Segm. 28 : Lampara Barrera Delantera



☐ Segm. 29 : Lampara Barrera Trasera



☐ Segm. 30 : Iluminacion mesa 1

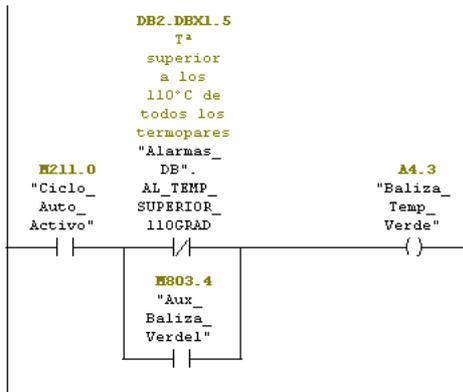


☐ Segm. 31 : Iluminacion mesa 2



Ilustración 159 FC8 Segmentos 28, 29, 30 y 31

☐ Segm. 32 : Baliza Verde



☐ Segm. 33 : Parpadeo Baliza Verde y Baliza Roja

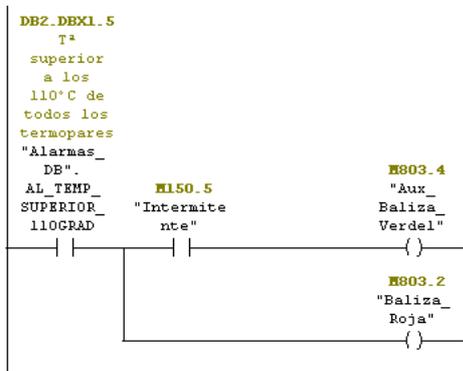


Ilustración 160 FC8 Segmentos 32 y 33

Segm. 34 : Baliza Naranja

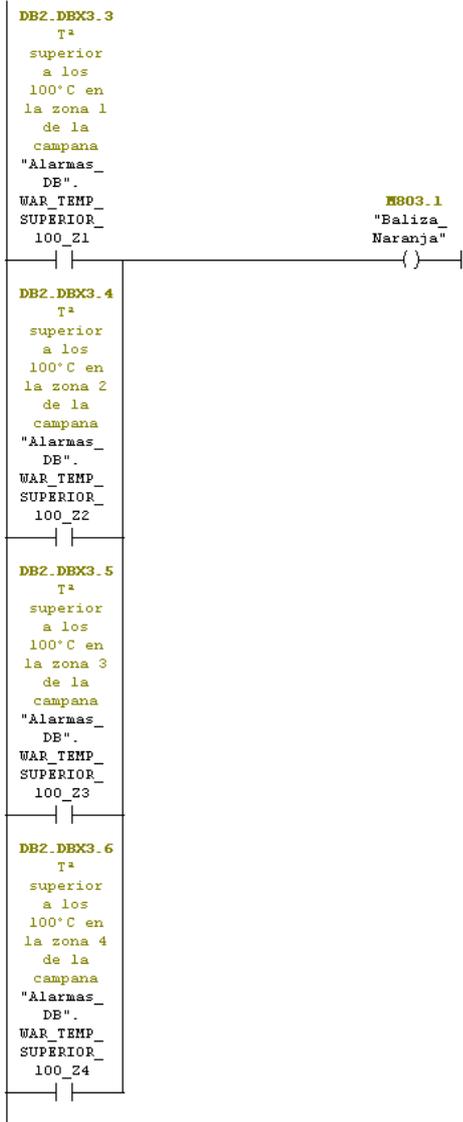
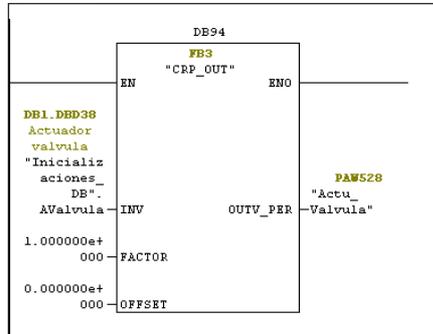


Ilustración 161 FC8 Segmento 34_1

Segm. 35 : Baliza Roja



Segm. 36 : Actuador valvula



Segm. 37 : Actuador compuertas freecooling

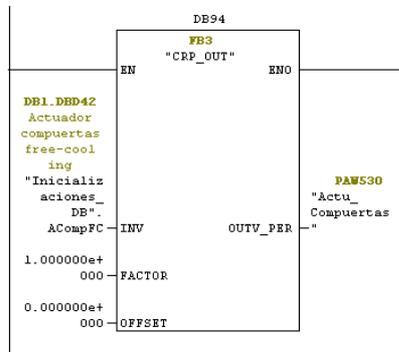


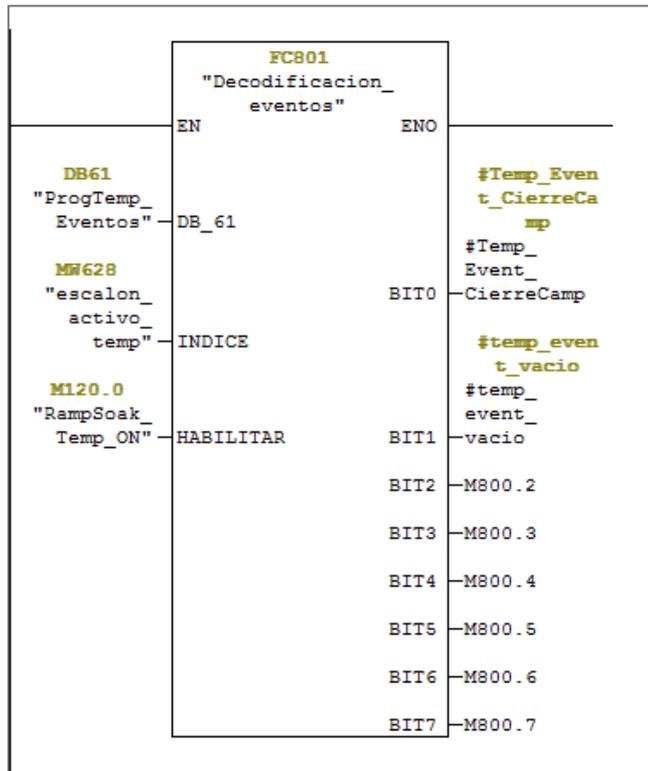
Ilustración 162 FC8 Segmentos 34, 35, 36 y 37

4.3 FUNCIONES Y BLOQUES DE FUNCIÓN DEL OB35

4.3.1 FB2: GESTIÓN PROGRAMADORES

En este bloque de función se gestiona todo lo que tiene que ver con los programas a seguir. En primer lugar, usamos la función FC801, la cual se explicará posteriormente, para decodificar los eventos, en el segmento uno podemos ver como indicándole la DB que tiene que mirar, el escalón activo y que el programa de temperatura está activo, devuelve el valor de los eventos en cada ciclo de escán, de momento sólo son necesarios 2 eventos, pero el programa está preparado para tener hasta 8. En los dos siguientes segmentos se puede observar cómo cuando se activa el bit 0 o el bit 1 activamos las marcas correspondientes para avisar al programa de que ya ha llegado ese evento y se tiene que realizar la acción correspondiente.

Segm. 1: Eventos programa de temperatura



Segm. 2: set evento iniciar programa vacio

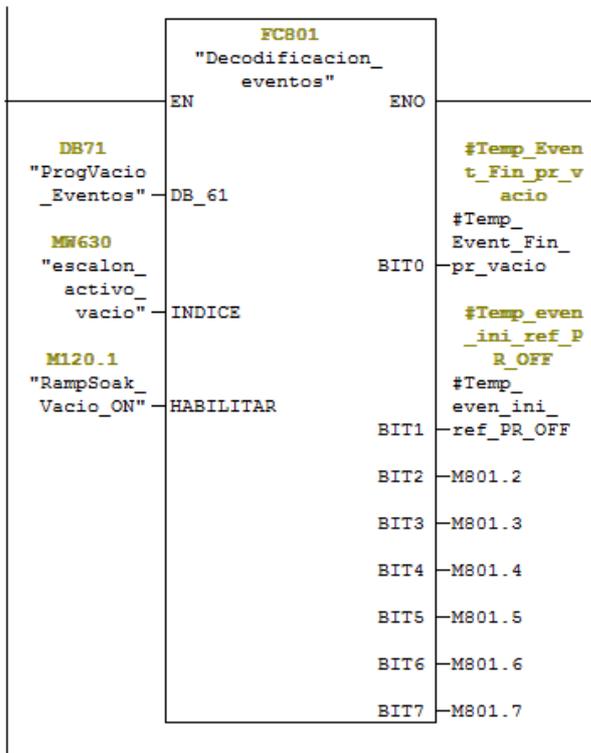


Segm. 3: set evento cerrar campana



Ilustración 163: FB2 Segmentos 1, 2 y 3

A continuación, se realiza la misma operación, pero con el programa de vacío.



Segm. 5 : set evento fin vacio



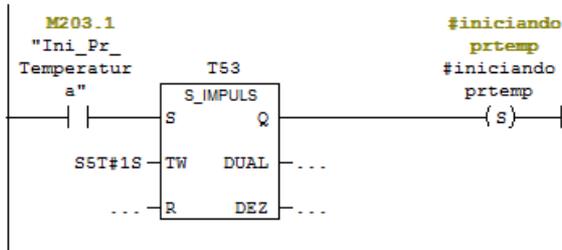
Segm. 6 : set evento refrigeracion y temperatura off



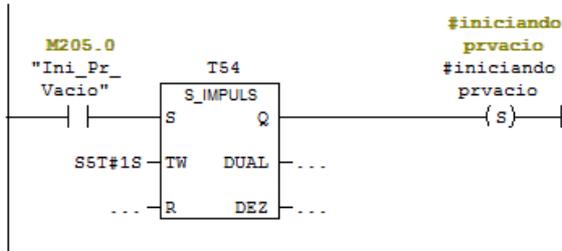
Ilustración 164: FB2 Segmentos 4, 5 y 6

En los segmentos 7 y 8 guardamos los órdenes de iniciar programa de temperatura o de vacío en dos variables estáticas. Como se puede observar en las siguientes imágenes estas variables activan sus programas correspondientes, RampSoak_Temp_ON o RampSoak_Vacio_ON, metiéndole un también un reset a estos, Reset_Temp_ON o Reset_Vacio_ON, para que empiecen desde el principio.

Segm. 7 : Orden de iniciar programador de temperatura



Segm. 8 : Orden de iniciar programador de vacio

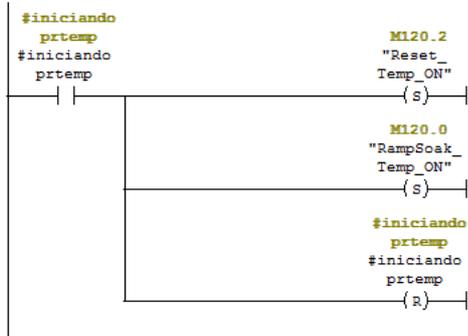


Segm. 9 : Titulo:



Ilustración 165: FB2 Segmentos 7, 8 y 9

Segm. 10 : Pulso de reset y activacion de programa de temperatura



Segm. 11 : informacion pr temp on a scada

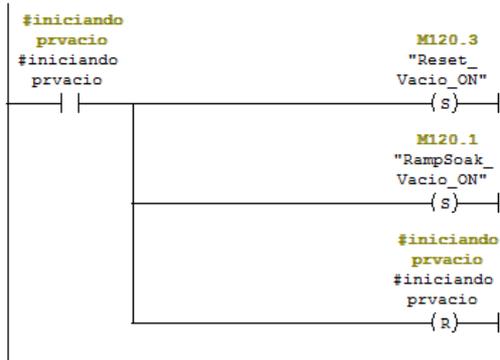


Segm. 12 : Titulo:



Ilustración 166: FB2 Segmentos 10, 11 y 12

Segm. 13 : Pulso de reset y activacion de programa de vacio



Segm. 14 : informacion pr temp on a scada



Segm. 15 : Apagado de programador de temperatura

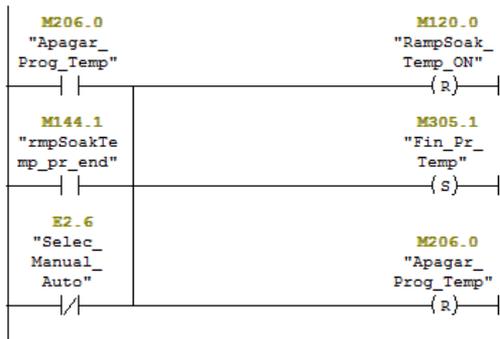


Ilustración 167: FB2 Segmentos 13,14 y 15

Tanto en el segmento 15 como el segmento 16 podemos observar cómo apagar los programas, estos se apagan cuando han terminado, cuando reciben la orden de apagar programa, o cuando se pasa de manual a automático.

Segm. 16 : Apagado de programador de temperatura

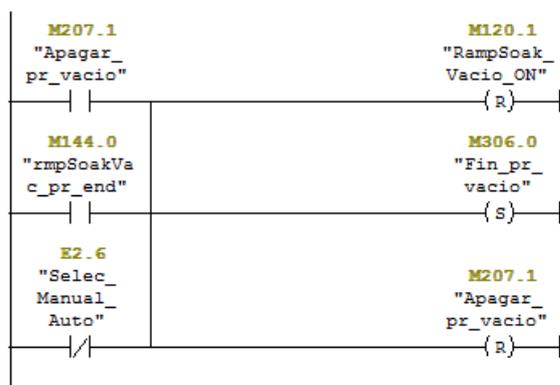


Ilustración 168: FB2 Segmento 16

Segm. 17: Inicializacion PVS

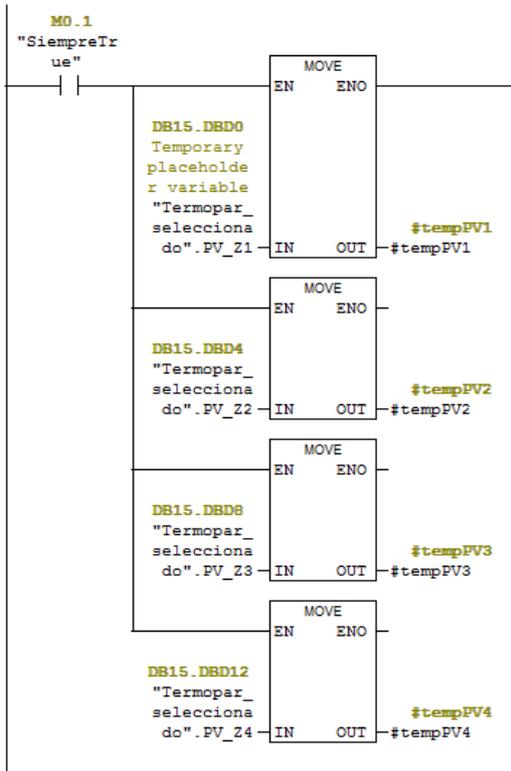


Ilustración 169: FB2 Segmento 17

Segm. 18: Cambio de PVs precalentamiento, max campana

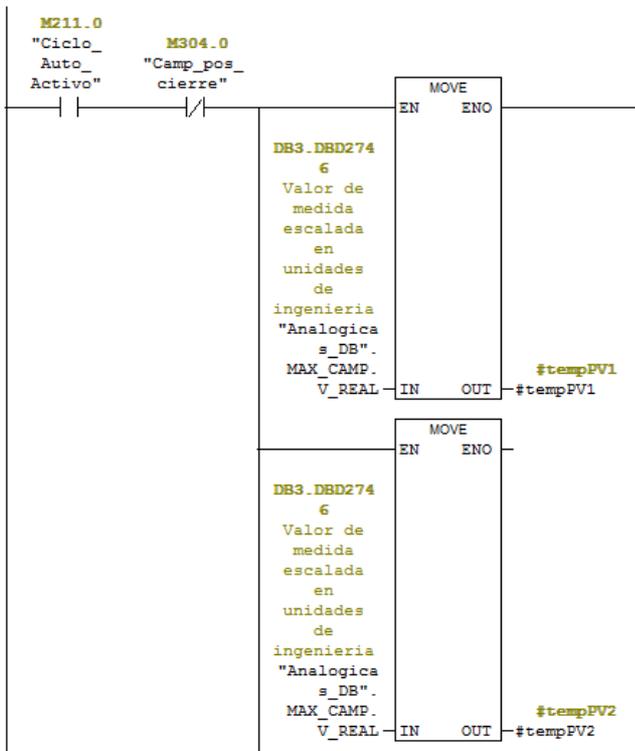


Ilustración 170: FB2 Segmento18_1

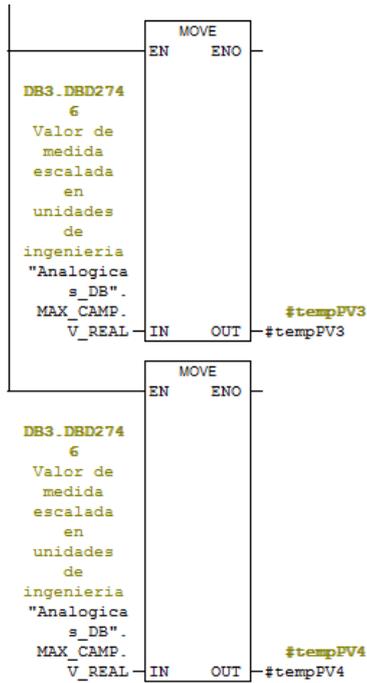
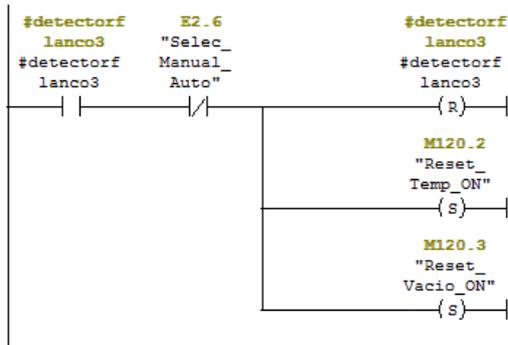


Ilustración 171: FB2 Segmento 18_2

Segm. 19 : Detector de flanco paso a manual



Segm. 20 : reset a los programadores por pasar a manual



Segm. 21 : Título:



Ilustración 172: FB2 Segmentos 19, 20 y 21

		FB1 "Gestion PID"		
		EN		ENO
#tempPV1				MW110
#tempPV1	PV1		Zona1_OUT	"salida_ temp_Z1"
#tempPV2				MW112
#tempPV2	PV2		Zona2_OUT	"salida_ temp_Z2"
#tempPV3				MW114
#tempPV3	PV3		Zona3_OUT	"salida_ temp_Z3"
#tempPV4				MW116
#tempPV4	PV4		Zona4_OUT	"salida_ temp_Z4"
DB3.DBD315				MW118
4			Vacio_OUT	"salida_ vacio"
Valor de medida escalada en unidades de ingenieria "Analogicas_DB". VACIO.			Prog_ Temp_On	...
V_REAL	PVvacio		Prog_ Vacio_On	...
DB15.DBX16				MW628
.0			Escalon_ Act_Temp	"escalon_ activo_ temp"
"Termopar seleccionado". Z1_				MW630
HABILITADA	Hab_Z1		Escalon_ Act_Vac	"escalon_ activo_ vacio"
DB15.DBX16				DB17.DBD6
.1				"Eventos_ Y_ comandos_ DB". EV_PRTEM_ TIEMPO_ TOTAL
"Termopar seleccionado". Z2_			T_Total_ PrTemp	
HABILITADA	Hab_Z2			
DB15.DBX16				DB17.DBD18
.2				"Eventos_ Y_ comandos_ DB". EV_PRVAC_ TIEMPO_ TOTAL
"Termopar seleccionado". Z3_				
HABILITADA	Hab_Z3			
DB15.DBX16				
.3			T_Total_ PrTemp	
...				

Ilustración 173: FB2 Segmento 22_1

DB15.DBX16 _3 "Termopar_ selecciona do". Z4_ HABILITADA	Hab_24	T_Total_ PrVacio	EV_PRVAC_ TIEMPO_ TOTAL
M120.0 "RampSoak_ Temp_ON"	Activar_ ProgTemp	T_ RestTotal _PrTemp	DB17.DBD10 "Eventos_ Y_ comandos_ DB". EV_PRTEM_ TIEMP_ RESTTOTAL
M120.1 "RampSoak_ Vacio_ON"	Activar_ ProgVac		DB17.DBD22 "Eventos_ Y_ comandos_ DB". EV_PRVAC_ TIEMP_ RESTTOTAL
M120.2 "Reset_ Temp_ON"	Reset_ ProgTemp	T_ RestTotal _PrVacio	DB17.DBD14 "Eventos_ Y_ comandos_ DB". EV_PRTE_ TIEMP_ RESTESCAL
M120.3 "Reset_ Vacio_ON"	Reset_ ProgVac		DB17.DBD26 "Eventos_ Y_ comandos_ DB". EV_PRVA_ TIEMP_ RESTESCAL
DB2.DBX4.2 T² superior a los 110° en el conjunto total "Alarmas_ DB". WAR_TEMP_ SUP_110_ TODO	Alarma90	T_ RestEscal on_PrTemp	
DB2.DBX3.3 T²		T_ RestEscal on_ PrVacio	

Ilustración 174: FB2 Segmento 22_2

DB2.DBX3.3 T² superior a los 100°C en la zona 1 de la campana "Alarmas_ DB". WAR_TEMP_ SUPERIOR_ 100_Z1	Alarma85_ z1	PrVacio RmpSoakTe mp_END RmpSoakVa cio_END	RESTESCAL M144.1 "rmpSoakTe mp_pr_end" M144.0 "rmpSoakVa c_pr_end" DB17.DBD78 Setpoint actual del PID, dado por el Ramp Soak "Eventos_ Y_ comandos_ DB". EV_ SPZONA1_ TEMP_ ACTUAL
DB2.DBX3.4 T² superior a los 100°C en la zona 2 de la campana "Alarmas_ DB". WAR_TEMP_ SUPERIOR_ 100_Z2	Alarma85_ z2	SP_ZONA1	DB17.DBD82 Setpoint actual del PID, dado por el Ramp Soak "Eventos_ Y_ comandos_ DB". EV_ SPZONA2_ TEMP_ ACTUAL
DB2.DBX3.5 T² superior a los 100°C en la zona 3 de la campana "Alarmas_ DB". WAR_TEMP_ SUPERIOR_ 100_Z3	Alarma85_ z3	SP_ZONA2	DB17.DBD86 Setpoint actual del PID, dado por el Ramp Soak "Eventos_ Y_ comandos_ DB". EV_ SPZONA3_ TEMP_ ACTUAL
DB2.DBX3.6 T² superior a los 100°C en la zona 4 de la campana "Alarmas_ DB". WAR_TEMP_ SUPERIOR_ 100_Z4	Alarma85_ z4		

Ilustración 175: FB2 Segmento 22_3

DB2.DBX3.6			DB17.DBD86
T ²			Setpoint
superior			actual
a los			del PID,
100°C en			dado por
la zona 4			el Ramp
de la			Soak
campana			"Eventos_
"Alarmas_			Y_
DB".			comandos_
WAR_TEMP_			DB".
SUPERIOR_	Alarma85_		EV_
100_24	z4		SPZONA3_
			TEMP_
	TiristorM	SP_ZONA3	ACTUAL
...	anual		
	Tiristor1		DB17.DBD90
...	_MAN		Setpoint
	Tiristor2		actual
...	_MAN		del PID,
	Tiristor3		dado por
...	_MAN		el Ramp
	Tiristor4		Soak
...	_MAN		"Eventos_
	Act_		Y_
	Vacio_	SP_ZONA4	comandos_
...	Manual		DB".
	ConsigVac		EV_
...	io_MAN		SPZONA4_
			TEMP_
			ACTUAL
DB17.DBX31			DB17.DBD94
.2			Setpoint
"Eventos_			actual
Y_			del PID,
comandos_			dado por
DB".			el Ramp
COM_			Soak
AVANCE_	Avance_		"Eventos_
PROG_TEMPE	Esc_temp		Y_
		SP_VACIO	comandos_
			DB".
			EV_SP_
			VACIO_
			ACTUAL
DB17.DBX98			DB17.DBD10
.5			4
"Eventos_			"Eventos_
Y_			Y_
comandos_			comandos_
DB".			DB".
COM_			EV_OPZ1
AVANCE_	Avance_	OPZ1	
PROG_VACIO	Esc_vacio		
			DB17.DBD10

Ilustración 176: FB2 Segmento 22_4

		SP_VACIO	ACTUAL
DB17.DBX98 -5 "Eventos_ Y_ comandos_ DB". COM_ AVANCE_ PROG_VACIO	Avance_ Esc_vacio	OPZ1	DB17.DBD10 4 "Eventos_ Y_ comandos_ DB". EV_OPZ1
DB17.DBX30 -0 COMANDOS "Eventos_ Y_ comandos_ DB". COM_ RESIST_ MARCHA	Lazo_ temp_ manual	OPZ2	DB17.DBD10 8 "Eventos_ Y_ comandos_ DB". EV_OPZ2
DB17.DBX31 -1 "Eventos_ Y_ comandos_ DB". COM_BOMBA_ VACIO_ MARCHA	Lazo_ vacio_ manual	OPZ3	DB17.DBD11 2 "Eventos_ Y_ comandos_ DB". EV_OPZ3
DB17.DBD50 "Eventos_ Y_ comandos_ DB". COM_SP_Z1_ MANUAL	SPZ1_ temp_Man	OPZ4	DB17.DBD11 6 "Eventos_ Y_ comandos_ DB". EV_OPZ4
DB17.DBD54 "Eventos_ Y_ comandos_ DB". COM_SP_Z2_ MANUAL	SPZ2_ temp_Man	OPVAC	DB17.DBD12 0 "Eventos_ Y_ comandos_ DB". EV_OPVACIO
DB17.DBD58 "Eventos_ Y_ comandos_ DB". COM_SP_Z3_ MANUAL	SPZ3_ temp_Man		

Ilustración 177: FB2 Segmento 22_5

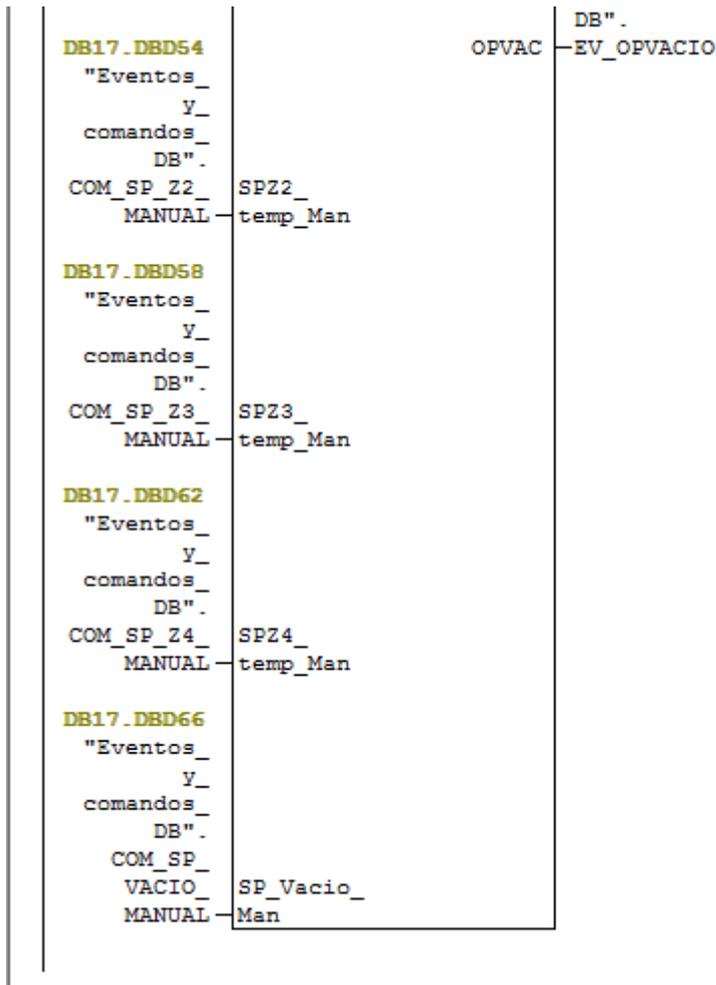
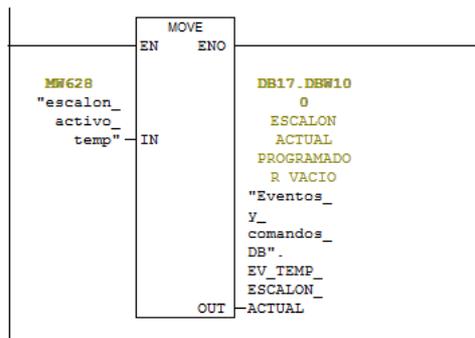


Ilustración 178: FB2 Segmento 22_6

Segm. 23: Informamos al SCADA del escalon actual temperatura



Segm. 24: Informamos al SCADA del escalon actual vacio

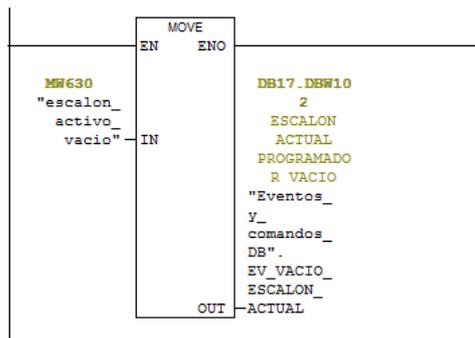
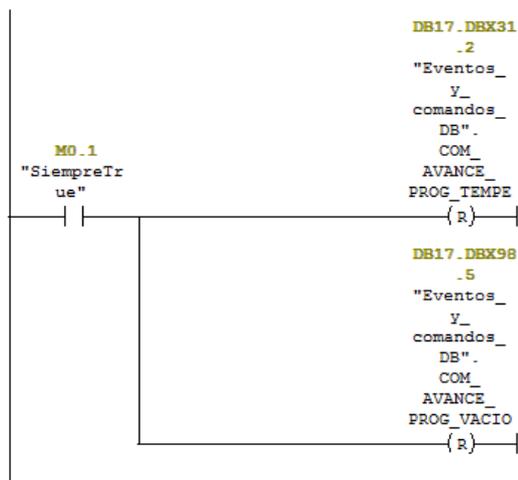


Ilustración 179: FB2 Segmentos 23 y 24

☐ Segm. 25 : reconocimiento señal de siguiente escalon



☐ Segm. 26 : Detector de flanco de subida en Ini Temp



☐ Segm. 27 : Titulo:



Ilustración 180: FB2 Segmentos 25, 26 y 27

☐ Segm. 28 : Detector de flanco de subida en Ini Vacio



☐ Segm. 29 : Titulo:



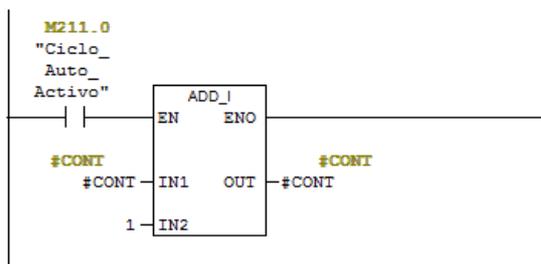
Ilustración 181: FB2 Segmentos 28 y 29

4.3.2 FB4: RELOJ CICLO AUTO

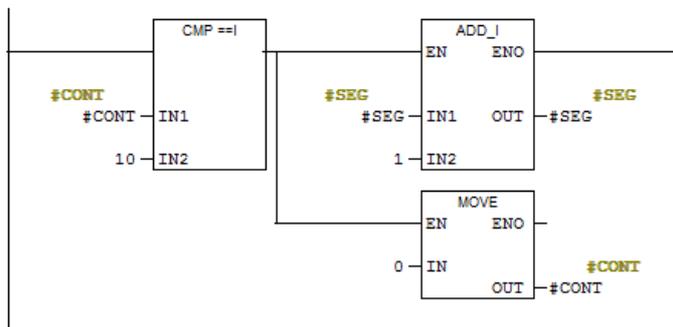
Este bloque de función es un contador del tiempo en el que la máquina se encuentra ejecutando un ciclo en modo automático, se encuentra dentro del OB35, por lo que se sabe que se ejecuta cada 100ms. Lo primero que se hace es saber si está el ciclo automático activo o no, si es que sí aumentamos en 1 el contador, lo que equivale a 100ms, cuando este contador llega a 10 se aumenta en 1 el contador de segundos y se pone a cero éste. Posteriormente, cuando el contador de segundos llega a 60, se aumenta en 1 el contador de minutos y se pone a cero el de segundos. Y se hace la misma operación para pasar a horas y a días.

Por último, en el caso de que llegue la orden de resetear el contador desde el SCADA, se ponen todos los contadores a cero.

Segm. 1: CONTADOR DE MILLISEGUNDOS



Segm. 2: CONTADOR DE SEGUNDOS



Segm. 3: CONTADOR DE MINUTOS

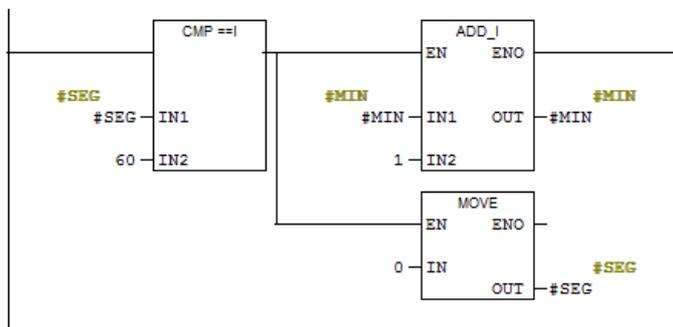
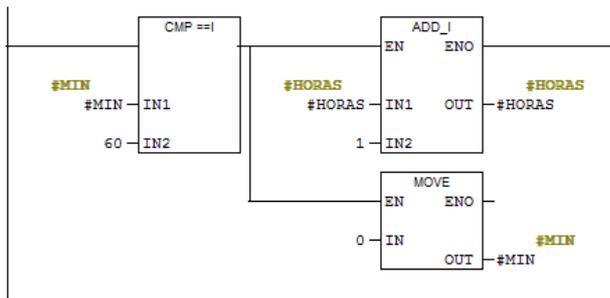


Ilustración 182: FB4 Segmentos 1, 2 y 3

Segm. 4 : CONTADOR DE HORAS



Segm. 5 : CONTADOR DE DIAS

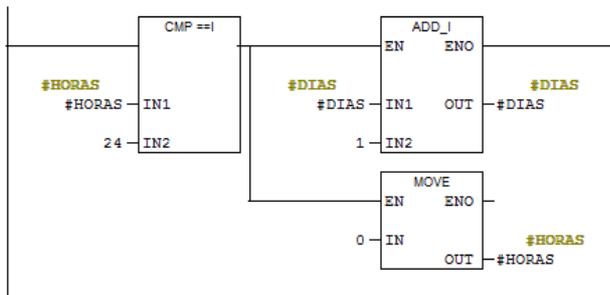


Ilustración 183: FB4 Segmentos 4 y 5

Segm. 6 : informacion a scada

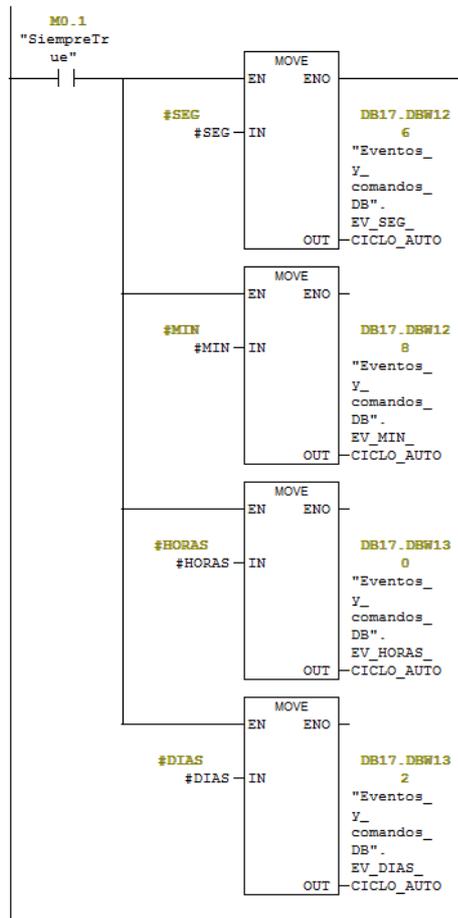


Ilustración 184: FB4 Segmento 6

Segm. 7 : RESET DEL RELOJ

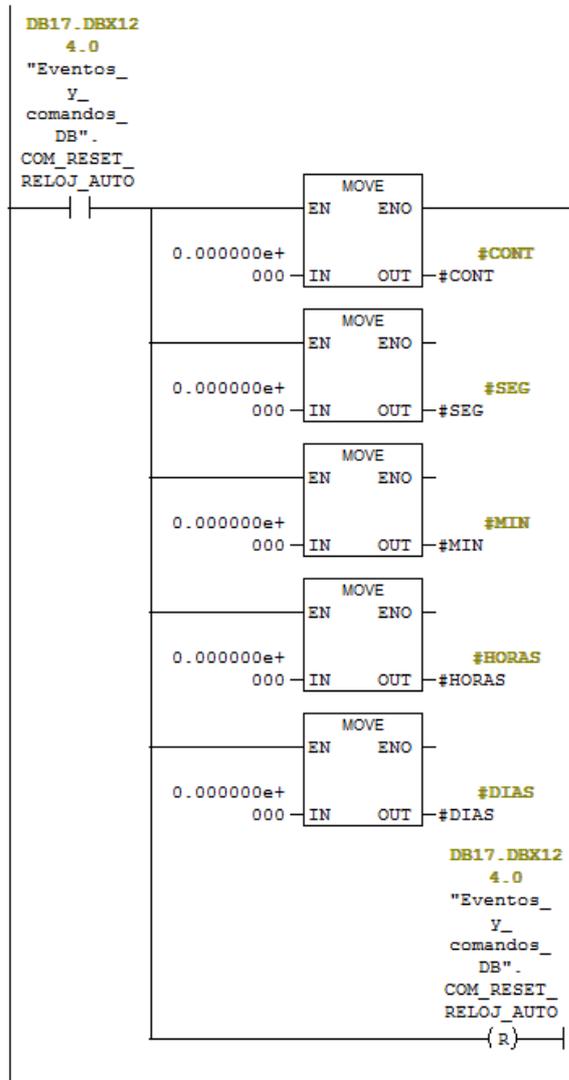


Ilustración 185: FB4 Segmento 7

4.4 BLOQUES DE DATOS

Los bloques de datos son áreas de memoria que, a diferencia de las marcas, están definidas por el usuario.

Las variables contenidas en estos bloques son remanentes por defecto al apagado del PLC, por lo que no se pierde su valor.

Los módulos de datos pueden ser del de tipo global o de instancia. Los de instancia son necesarios para la ejecución de los FB y toman la estructura de la declaración de las variables del propio FB. Los DB globales son módulos declarados por el usuario y pueden contener variables de todo tipo definiendo estructuras complejas.

En ambos casos el acceso a la información es global ya que incluso se puede acceder a la información de un DB de instancia desde otro bloque.

4.4.1 BLOQUES DE DATOS GLOBALES

4.4.1.1 DB2: ALARMAS

En la DB de alarmas, encontramos todas las memorias referentes a los avisos o alarmas que pueden ocurrir durante el funcionamiento de la máquina, éstas se escriben en la FC11: Gestión Alarmas. Esta DB es consultada por el SCADA para mostrar por la pantalla un listado de las alarmas que se encuentran activas y así poder solucionar el problema.

Dirección	Nombre	Tipo	Valor inicial	Comentario
0.0		STRUCT		
+0.0	AL_PLC	BOOL	FALSE	
+0.1	AL_PARO_EMERGENCIA	BOOL	FALSE	
+0.2	AL_FALLO_TIRISTOR1	BOOL	FALSE	
+0.3	AL_FALLO_TIRISTOR2	BOOL	FALSE	
+0.4	AL_FALLO_TIRISTOR3	BOOL	FALSE	
+0.5	AL_FALLO_TIRISTOR4	BOOL	FALSE	
+0.6	AL_SOBRECARG_MESA1	BOOL	FALSE	
+0.7	AL_SOBRECARG_MESA2	BOOL	FALSE	
+1.0	AL_SOBRECARG_VAR_CAMP	BOOL	FALSE	
+1.1	AL_SOBRECARG_CIL_CAMP	BOOL	FALSE	
+1.2	AL_TER_CONTROL_NO_SELEC	BOOL	FALSE	No hay seleccionado ningún termopar de control
+1.3	AL_DOS_MESAS_MOVIMIENTO	BOOL	FALSE	Las dos mesas se están moviendo a la vez
+1.4	AL_CAMP_HOV_MESA_MODENTR	BOOL	FALSE	La campana se está moviendo y la mesa no está dentro
+1.5	AL_TEMP_SUPERIOR_110GRAD	BOOL	FALSE	Tª superior a los 110°C de todos los termopares
+1.6	AL_DET_SEG_CAMP_ARRIBA	BOOL	FALSE	Detector de seguridad de la campana arriba
+1.7	AL_DET_SEG_CAMP_ABAJO	BOOL	FALSE	Detector de seguridad de la campana abajo
+2.0	AL_DET_CAMP_ARRIBA	BOOL	FALSE	Más de un segundo sin los 4 sensores de campana arriba encendidos
+2.1	AL_DET_CAMP_ABAJO	BOOL	FALSE	Más de un segundo sin los 4 sensores de campana abajo encendidos
+2.2	AL_DET_MESA1	BOOL	FALSE	Detecta la mesa 1 dentro y fuera
+2.3	AL_DET_MESA2	BOOL	FALSE	Detecta la mesa 2 dentro y fuera
+2.4	AL_DET_CAMPANA	BOOL	FALSE	Detecta campana arriba y campana abajo
+2.5	AL_NING_HABILITADO_Z1	BOOL	FALSE	No hay ningún termopar habilitado en la zona 1 de la campana
+2.6	AL_NING_HABILITADO_Z2	BOOL	FALSE	No hay ningún termopar habilitado en la zona 2 de la campana
+2.7	AL_NING_HABILITADO_Z3	BOOL	FALSE	No hay ningún termopar habilitado en la zona 3 de la campana
+3.0	AL_NING_HABILITADO_Z4	BOOL	FALSE	No hay ningún termopar habilitado en la zona 4 de la campana
+3.1	WAR_BARRERA_DELANTERA	BOOL	FALSE	Barrera delantera activada
+3.2	WAR_BARRERA TRASERA	BOOL	FALSE	Barrera trasera activada
+3.3	WAR_TEMP_SUPERIOR_100_Z1	BOOL	FALSE	Tª superior a los 100°C en la zona 1 de la campana
+3.4	WAR_TEMP_SUPERIOR_100_Z2	BOOL	FALSE	Tª superior a los 100°C en la zona 2 de la campana
+3.5	WAR_TEMP_SUPERIOR_100_Z3	BOOL	FALSE	Tª superior a los 100°C en la zona 3 de la campana
+3.6	WAR_TEMP_SUPERIOR_100_Z4	BOOL	FALSE	Tª superior a los 100°C en la zona 4 de la campana

Ilustración 186: DB2_1

+3.7	WAR_SOBRECARGA_MESA1	BOOL	FALSE	
+4.0	WAR_SOBRECARGA_MESA2	BOOL	FALSE	
+4.1	WAR_REFRIGERACION_OK	BOOL	TRUE	
+4.2	WAR_TEMP_SUP_110_TODO	BOOL	FALSE	Tª superior a los 110° en el conjunto total
+4.3	WAR_SOBREC_CILIN_CAMP	BOOL	FALSE	
+4.4	WAR_SOBREC_VARIA_CAMP	BOOL	FALSE	
+4.5	WAR_NING_HABILITADO_Z1	BOOL	FALSE	
+4.6	WAR_NING_HABILITADO_Z2	BOOL	FALSE	
+4.7	WAR_NING_HABILITADO_Z3	BOOL	FALSE	
+5.0	WAR_NING_HABILITADO_Z4	BOOL	FALSE	
+5.1	AL_ERROR_GRAVE	BOOL	FALSE	
+5.2	AL_REFRI_CRUZ_NO_ABIERTA	BOOL	FALSE	Se ha dado la orden de abrir la cruz de refrigeración pero no se ha abierto
+5.3	AL_REFRI_CRUZ_NO_CERRADA	BOOL	FALSE	Se ha dado la orden de cerrar la cruz de refrigeración pero no se ha cerrado
+5.4	AL_REFRI_DIREC_NO_ABIERT	BOOL	FALSE	Se ha dado la orden de abrir la directa de refrigeración pero no se ha abierto
+5.5	AL_REFRI_DIREC_NO_CERRAD	BOOL	FALSE	Se ha dado la orden de cerrar directa de refrigeración pero no se ha cerrado
+5.6	AL_TRAMPILLAS_CERRADAS	BOOL	FALSE	Están cerradas todas las trampillas del bypass
+5.7	AL_EN_EL_BYPASS	BOOL	FALSE	Hay alguna alarma en el bypass
+6.0	AL_NINGUNA_TRAMP_ABIERTA	BOOL	FALSE	No hay ninguna trampilla del bypass abierta
+6.1	AL_CIERRE_COMPUERTA1	BOOL	FALSE	
+6.2	AL_APERTURA_COMPUERTA1	BOOL	FALSE	
+6.3	WAR_FILTRO_UTA	BOOL	FALSE	
+6.4	WAR_FEEDBACK_VALVULA	BOOL	FALSE	
+6.5	WAR_REFRIGERADORA	BOOL	FALSE	
+6.6	WAR_REFRIGERADORA_10	BOOL	FALSE	
+6.7	AL_VENTILADOR_IMP_UTA	BOOL	FALSE	
+7.0	AL_VENTILADOR_RET_UTA	BOOL	FALSE	
+7.1	AL_APERTURA_COMPUERTA2	BOOL	FALSE	
+7.2	AL_CIERRE_COMPUERTA2	BOOL	FALSE	
+7.3	WAR_POS_CAMP_ARRIBA	BOOL	FALSE	
+7.4	WAR_POS_CAMP_ABAJO	BOOL	FALSE	
+7.5	AL_COMPUERTAS_HORNOS	BOOL	FALSE	
=8.0		END_STRUCT		

Ilustración 187: DB2_2

4.4.1.2 DB3: ANALÓGICAS

En esta DB se encuentran todas las señales analógicas, en ella se encuentra todo lo referente a ellas.

En las primeras 2951 direcciones están almacenados tanto los termopares de las mesas y la campana como sus termopares virtuales, estas variables son del tipo UDT_TERMOPARES.

Dirección	Nombre	Tipo	Valor inicial	Comentario
0.0		STRUCT		
+0.0	TC1M1	"UDT_TERMOPARES"		
+24.0	TC2M1	"UDT_TERMOPARES"		
+48.0	TC3M1	"UDT_TERMOPARES"		
+72.0	TC4M1	"UDT_TERMOPARES"		
+96.0	TC5M1	"UDT_TERMOPARES"		
+120.0	TC6M1	"UDT_TERMOPARES"		
+144.0	TC7M1	"UDT_TERMOPARES"		
+168.0	TC8M1	"UDT_TERMOPARES"		
+192.0	TC9M1	"UDT_TERMOPARES"		
+216.0	TC10M1	"UDT_TERMOPARES"		
+240.0	TC11M1	"UDT_TERMOPARES"		
+264.0	TC12M1	"UDT_TERMOPARES"		
+288.0	TC13M1	"UDT_TERMOPARES"		
+312.0	TC14M1	"UDT_TERMOPARES"		
+336.0	TC15M1	"UDT_TERMOPARES"		
+360.0	TC16M1	"UDT_TERMOPARES"		
+384.0	TC17M1	"UDT_TERMOPARES"		
+408.0	TC18M1	"UDT_TERMOPARES"		
+432.0	TC19M1	"UDT_TERMOPARES"		
+456.0	TC20M1	"UDT_TERMOPARES"		
+480.0	TC21M1	"UDT_TERMOPARES"		
+504.0	TC22M1	"UDT_TERMOPARES"		
+528.0	TC23M1	"UDT_TERMOPARES"		
+552.0	TC24M1	"UDT_TERMOPARES"		
+576.0	TC25M1	"UDT_TERMOPARES"		
+600.0	TC26M1	"UDT_TERMOPARES"		
+624.0	TC27M1	"UDT_TERMOPARES"		
+648.0	TC28M1	"UDT_TERMOPARES"		
+672.0	TC29M1	"UDT_TERMOPARES"		
+696.0	TC30M1	"UDT_TERMOPARES"		
+720.0	TC31M1	"UDT_TERMOPARES"		
+744.0	TC32M1	"UDT_TERMOPARES"		
+768.0	TC1M2	"UDT_TERMOPARES"		
+792.0	TC2M2	"UDT_TERMOPARES"		
+816.0	TC3M2	"UDT_TERMOPARES"		
+840.0	TC4M2	"UDT_TERMOPARES"		
+864.0	TC5M2	"UDT_TERMOPARES"		
+888.0	TC6M2	"UDT_TERMOPARES"		
+912.0	TC7M2	"UDT_TERMOPARES"		
+936.0	TC8M2	"UDT_TERMOPARES"		
+960.0	TC9M2	"UDT_TERMOPARES"		
+984.0	TC10M2	"UDT_TERMOPARES"		
+1008.0	TC11M2	"UDT_TERMOPARES"		
+1032.0	TC12M2	"UDT_TERMOPARES"		
+1056.0	TC13M2	"UDT_TERMOPARES"		
+1080.0	TC14M2	"UDT_TERMOPARES"		

Ilustración 188: DB3_1

+336.0	TC15M1	"UDT_TERMOPARES"		
+360.0	TC16M1	"UDT_TERMOPARES"		
+384.0	TC17M1	"UDT_TERMOPARES"		
+408.0	TC18M1	"UDT_TERMOPARES"		
+432.0	TC19M1	"UDT_TERMOPARES"		
+456.0	TC20M1	"UDT_TERMOPARES"		
+480.0	TC21M1	"UDT_TERMOPARES"		
+504.0	TC22M1	"UDT_TERMOPARES"		
+528.0	TC23M1	"UDT_TERMOPARES"		
+552.0	TC24M1	"UDT_TERMOPARES"		
+576.0	TC25M1	"UDT_TERMOPARES"		
+600.0	TC26M1	"UDT_TERMOPARES"		
+624.0	TC27M1	"UDT_TERMOPARES"		
+648.0	TC28M1	"UDT_TERMOPARES"		
+672.0	TC29M1	"UDT_TERMOPARES"		
+696.0	TC30M1	"UDT_TERMOPARES"		
+720.0	TC31M1	"UDT_TERMOPARES"		
+744.0	TC32M1	"UDT_TERMOPARES"		
+768.0	TC1M2	"UDT_TERMOPARES"		
+792.0	TC2M2	"UDT_TERMOPARES"		
+816.0	TC3M2	"UDT_TERMOPARES"		
+840.0	TC4M2	"UDT_TERMOPARES"		
+864.0	TC5M2	"UDT_TERMOPARES"		
+888.0	TC6M2	"UDT_TERMOPARES"		
+912.0	TC7M2	"UDT_TERMOPARES"		
+936.0	TC8M2	"UDT_TERMOPARES"		
+960.0	TC9M2	"UDT_TERMOPARES"		
+984.0	TC10M2	"UDT_TERMOPARES"		
+1008.0	TC11M2	"UDT_TERMOPARES"		
+1032.0	TC12M2	"UDT_TERMOPARES"		
+1056.0	TC13M2	"UDT_TERMOPARES"		
+1080.0	TC14M2	"UDT_TERMOPARES"		
+1104.0	TC15M2	"UDT_TERMOPARES"		
+1128.0	TC16M2	"UDT_TERMOPARES"		
+1152.0	TC17M2	"UDT_TERMOPARES"		
+1176.0	TC18M2	"UDT_TERMOPARES"		
+1200.0	TC19M2	"UDT_TERMOPARES"		
+1224.0	TC20M2	"UDT_TERMOPARES"		
+1248.0	TC21M2	"UDT_TERMOPARES"		
+1272.0	TC22M2	"UDT_TERMOPARES"		
+1296.0	TC23M2	"UDT_TERMOPARES"		
+1320.0	TC24M2	"UDT_TERMOPARES"		
+1344.0	TC25M2	"UDT_TERMOPARES"		
+1368.0	TC26M2	"UDT_TERMOPARES"		
+1392.0	TC27M2	"UDT_TERMOPARES"		
+1416.0	TC28M2	"UDT_TERMOPARES"		
+1440.0	TC29M2	"UDT_TERMOPARES"		
+1464.0	TC30M2	"UDT_TERMOPARES"		

Ilustración 189: DB3_2

+696.0	TC30M1	"UDT_TERMOPARES"		
+720.0	TC31M1	"UDT_TERMOPARES"		
+744.0	TC32M1	"UDT_TERMOPARES"		
+768.0	TC1M2	"UDT_TERMOPARES"		
+792.0	TC2M2	"UDT_TERMOPARES"		
+816.0	TC3M2	"UDT_TERMOPARES"		
+840.0	TC4M2	"UDT_TERMOPARES"		
+864.0	TC5M2	"UDT_TERMOPARES"		
+888.0	TC6M2	"UDT_TERMOPARES"		
+912.0	TC7M2	"UDT_TERMOPARES"		
+936.0	TC8M2	"UDT_TERMOPARES"		
+960.0	TC9M2	"UDT_TERMOPARES"		
+984.0	TC10M2	"UDT_TERMOPARES"		
+1008.0	TC11M2	"UDT_TERMOPARES"		
+1032.0	TC12M2	"UDT_TERMOPARES"		
+1056.0	TC13M2	"UDT_TERMOPARES"		
+1080.0	TC14M2	"UDT_TERMOPARES"		
+1104.0	TC15M2	"UDT_TERMOPARES"		
+1128.0	TC16M2	"UDT_TERMOPARES"		
+1152.0	TC17M2	"UDT_TERMOPARES"		
+1176.0	TC18M2	"UDT_TERMOPARES"		
+1200.0	TC19M2	"UDT_TERMOPARES"		
+1224.0	TC20M2	"UDT_TERMOPARES"		
+1248.0	TC21M2	"UDT_TERMOPARES"		
+1272.0	TC22M2	"UDT_TERMOPARES"		
+1296.0	TC23M2	"UDT_TERMOPARES"		
+1320.0	TC24M2	"UDT_TERMOPARES"		
+1344.0	TC25M2	"UDT_TERMOPARES"		
+1368.0	TC26M2	"UDT_TERMOPARES"		
+1392.0	TC27M2	"UDT_TERMOPARES"		
+1416.0	TC28M2	"UDT_TERMOPARES"		
+1440.0	TC29M2	"UDT_TERMOPARES"		
+1464.0	TC30M2	"UDT_TERMOPARES"		
+1488.0	TC31M2	"UDT_TERMOPARES"		
+1512.0	TC32M2	"UDT_TERMOPARES"		
+1536.0	TC1C	"UDT_TERMOPARES"		
+1560.0	TC2C	"UDT_TERMOPARES"		
+1584.0	TC3C	"UDT_TERMOPARES"		
+1608.0	TC4C	"UDT_TERMOPARES"		
+1632.0	TC5C	"UDT_TERMOPARES"		
+1656.0	TC6C	"UDT_TERMOPARES"		
+1680.0	TC7C	"UDT_TERMOPARES"		
+1704.0	TC8C	"UDT_TERMOPARES"		

Ilustración 190:DB3_3

+1752.0	MIN_M1_Z1	"UDT_TERMOPARES"		
+1776.0	MED_M1_Z1	"UDT_TERMOPARES"		
+1800.0	MAX_M1_Z2	"UDT_TERMOPARES"		
+1824.0	MIN_M1_Z2	"UDT_TERMOPARES"		
+1848.0	MED_M1_Z2	"UDT_TERMOPARES"		
+1872.0	MAX_M1_Z3	"UDT_TERMOPARES"		
+1896.0	MIN_M1_Z3	"UDT_TERMOPARES"		
+1920.0	MED_M1_Z3	"UDT_TERMOPARES"		
+1944.0	MAX_M1_Z4	"UDT_TERMOPARES"		
+1968.0	MIN_M1_Z4	"UDT_TERMOPARES"		
+1992.0	MED_M1_Z4	"UDT_TERMOPARES"		
+2016.0	MAX_M1	"UDT_TERMOPARES"		
+2040.0	MIN_M1	"UDT_TERMOPARES"		
+2064.0	MED_M1	"UDT_TERMOPARES"		
+2088.0	MAX_M2_Z1	"UDT_TERMOPARES"		
+2112.0	MIN_M2_Z1	"UDT_TERMOPARES"		
+2136.0	MED_M2_Z1	"UDT_TERMOPARES"		
+2160.0	MAX_M2_Z2	"UDT_TERMOPARES"		
+2184.0	MIN_M2_Z2	"UDT_TERMOPARES"		
+2208.0	MED_M2_Z2	"UDT_TERMOPARES"		
+2232.0	MAX_M2_Z3	"UDT_TERMOPARES"		
+2256.0	MIN_M2_Z3	"UDT_TERMOPARES"		
+2280.0	MED_M2_Z3	"UDT_TERMOPARES"		
+2304.0	MAX_M2_Z4	"UDT_TERMOPARES"		
+2328.0	MIN_M2_Z4	"UDT_TERMOPARES"		
+2352.0	MED_M2_Z4	"UDT_TERMOPARES"		
+2376.0	MAX_M2	"UDT_TERMOPARES"		
+2400.0	MIN_M2	"UDT_TERMOPARES"		
+2424.0	MED_M2	"UDT_TERMOPARES"		
+2448.0	MAX_CAMP_Z1	"UDT_TERMOPARES"		
+2472.0	MIN_CAMP_Z1	"UDT_TERMOPARES"		
+2496.0	MED_CAMP_Z1	"UDT_TERMOPARES"		
+2520.0	MAX_CAMP_Z2	"UDT_TERMOPARES"		
+2544.0	MIN_CAMP_Z2	"UDT_TERMOPARES"		
+2568.0	MED_CAMP_Z2	"UDT_TERMOPARES"		
+2592.0	MAX_CAMP_Z3	"UDT_TERMOPARES"		
+2616.0	MIN_CAMP_Z3	"UDT_TERMOPARES"		
+2640.0	MED_CAMP_Z3	"UDT_TERMOPARES"		
+2664.0	MAX_CAMP_Z4	"UDT_TERMOPARES"		
+2688.0	MIN_CAMP_Z4	"UDT_TERMOPARES"		
+2712.0	MED_CAMP_Z4	"UDT_TERMOPARES"		
+2736.0	MAX_CAMP	"UDT_TERMOPARES"		
+2760.0	MIN_CAMP	"UDT_TERMOPARES"		
+2784.0	MED_CAMP	"UDT_TERMOPARES"		
+2808.0	MAX_M1CAMP	"UDT_TERMOPARES"		
+2832.0	MIN_M1CAMP	"UDT_TERMOPARES"		
+2856.0	MED_M1CAMP	"UDT_TERMOPARES"		
+2880.0	MAX_M2CAMP	"UDT_TERMOPARES"		
+2904.0	MIN_M2CAMP	"UDT_TERMOPARES"		
+2928.0	MED_M2CAMP	"UDT_TERMOPARES"		

Ilustración 191: DB3_4

Desde la dirección 2952 hasta la dirección 3143, la DB almacena los datos de los sensores de infrarrojos, estos son del tipo UDT_SERSOR_INFRARROJOS. Por último, encontramos los sensores de vacío y de posición de la campana siendo estos del tipo UDT_SENSOR_VACIO y UDT_SENSOR_CAMPANA respectivamente.

+2952.0	TI1C	"UDT_SENSOR_INFR.		
+2976.0	TI2C	"UDT_SENSOR_INFR.		
+3000.0	TI3C	"UDT_SENSOR_INFR.		
+3024.0	TI4C	"UDT_SENSOR_INFR.		
+3048.0	TI5C	"UDT_SENSOR_INFR.		
+3072.0	TI6C	"UDT_SENSOR_INFR.		
+3096.0	TI7C	"UDT_SENSOR_INFR.		
+3120.0	TI8C	"UDT_SENSOR_INFR.		
+3144.0	VACIO	"UDT_SENSOR_VACIO		
+3168.0	CAMPANA	"UDT_SENSOR_CAMP.		

Ilustración 192: DB3_5

4.4.1.3 DB15: TERMOPAR SELECCIONADO

Esta DB almacena el valor real de las variables de procesos que se usan para regular el PID de cada zona en las primeras 4 variables, las siguientes 4 indican si hay termopares de campana habilitados en una zona o no, y, por último, y por último la variable ERROR_TERMOPARES_CONTROL indica que no hay ningún termopar seleccionado para el control.

Esta DB la escribe la FC701 una vez que ha recorrido la DB3.

Dirección	Nombre	Tipo	Valor inicial	Comentario
0.0		STRUCT		
+0.0	PV_Z1	REAL	7.565000e+001	
+4.0	PV_Z2	REAL	1.025770e+002	
+8.0	PV_Z3	REAL	0.000000e+000	
+12.0	PV_Z4	REAL	0.000000e+000	
+16.0	Z1_HABILITADA	BOOL	FALSE	
+16.1	Z2_HABILITADA	BOOL	FALSE	
+16.2	Z3_HABILITADA	BOOL	FALSE	
+16.3	Z4_HABILITADA	BOOL	FALSE	
+16.4	ERROR_TERMOPARES_CONTROL	BOOL	FALSE	
=18.0		END_STRUCT		

Ilustración 193: DB15

4.4.1.4 DB17: EVENTOS Y COMANDOS

En esta DB se almacenan todas las variables que sirven para comunicarse con el SCADA, tanto los eventos que están surgiendo en el PLC para mantener informado al operador como los comandos que éste le da a la máquina para que ésta realice alguna operación.

Esta DB se escribe desde la FC10, los eventos, y los comandos desde el propio SCADA.

Dirección	Nombre	Tipo	Valor inicial	Comentario
0.0		STRUCT		
+0.0	EV_CAMPANA_ARRIBA	BOOL	FALSE	
+0.1	EV_CAMPANA_ABAJO	BOOL	FALSE	
+0.2	EV_CAMPANA_EN_PRECALENTA	BOOL	FALSE	
+0.3	EV_MESA1_FUERA	BOOL	FALSE	
+0.4	EV_MESA2_FUERA	BOOL	FALSE	
+0.5	EV_MESA1_DENTRO	BOOL	FALSE	
+0.6	EV_MESA2_DENTRO	BOOL	FALSE	
+0.7	EV_MESA1_CASI_DENTRO	BOOL	FALSE	
+1.0	EV_MESA2_CASI_DENTRO	BOOL	FALSE	
+1.1	EV_MESA1_CASI_FUERA	BOOL	FALSE	
+1.2	EV_MESA2_CASI_FUERA	BOOL	FALSE	
+1.3	EV_MESA1_AVANCE_LENTO	BOOL	FALSE	
+1.4	EV_MESA1_AVANCE_RAPIDO	BOOL	FALSE	
+1.5	EV_MESA2_AVANCE_LENTO	BOOL	FALSE	
+1.6	EV_MESA2_AVANCE_RAPIDO	BOOL	FALSE	
+1.7	EV_MESA1_RETROCESO_LENTO	BOOL	FALSE	
+2.0	EV_MESA1_RETROCESO_RAPIDO	BOOL	FALSE	
+2.1	EV_MESA2_RETROCESO_LENTO	BOOL	FALSE	
+2.2	EV_MESA2_RETROCESO_RAPIDO	BOOL	FALSE	
+2.3	EV_VACIO_M1	BOOL	FALSE	
+2.4	EV_VACIO_M2	BOOL	FALSE	
+2.5	EV_ABIERTO_VALV_ROT_VAC	BOOL	FALSE	
+2.6	EV_SELEC_PERMISO_MESA	BOOL	FALSE	
+2.7	EV_RESIST_ON	BOOL	FALSE	
+3.0	EV_CICLO_AUTO_ACTIVO	BOOL	FALSE	
+3.1	EV_SUBIENDO_CAMPANA	BOOL	FALSE	
+3.2	EV_BAJANDO_CAMPANA	BOOL	FALSE	
+3.3	EV_REFRI_EN_MARCHA	BOOL	FALSE	
+3.4	EV_BOMBA_VACIO_ON	BOOL	FALSE	
+3.5	EV_PROG_TEMP_ON	BOOL	FALSE	
+3.6	EV_PROG_VACIO_ON	BOOL	FALSE	
+3.7	EV_SELEC_MANUAL_AUTO	BOOL	FALSE	
+4.0	EV_FIN_CICLO_AUTO	BOOL	FALSE	
+6.0	EV_PRTEM_TIEMPO_TOTAL	TIME	T#0MS	
+10.0	EV_PRTEM_TIEMP_RESTTOTAL	TIME	T#0MS	
+14.0	EV_PRTE_TIEMP_RESTESCAL	TIME	T#0MS	
+18.0	EV_PRVAC_TIEMPO_TOTAL	TIME	T#0MS	
+22.0	EV_PRVAC_TIEMP_RESTTOTAL	TIME	T#0MS	
+26.0	EV_PRVA_TIEMP_RESTESCAL	TIME	T#0MS	
+30.0	COM_RESIST_MARCHA	BOOL	FALSE	COMANDOS

Ilustración 194: DB17_1

+30.1	COM_RESETEAR_BYPASS	BOOL	FALSE	
+30.2	COM_CICLO_AUTO_EN_MARCHA	BOOL	FALSE	
+30.3	EV_REFRIG_PREPARADA	BOOL	FALSE	
+30.4	COM_REF_FIJO	BOOL	FALSE	
+30.5	COM_REF_FIJO_INVERSA	BOOL	FALSE	
+30.6	COM_REF_FIJO_DIRECTA	BOOL	FALSE	
+30.7	COM_REFRIGERACION_MARCHA	BOOL	FALSE	
+31.0	COM_ROT_VACIO_MARCHA	BOOL	FALSE	
+31.1	COM_BOMBA_VACIO_MARCHA	BOOL	FALSE	
+31.2	COM_AVANCE_PROG_TEMPE	BOOL	FALSE	
+31.3	COM_REF_POR_TIEMPO	BOOL	FALSE	
+31.4	COM_REF_POR_TEMPERATURA	BOOL	FALSE	
+31.5	COM_LZONA_4ZONAS	BOOL	FALSE	
+31.6	EV_CRUZ_REF_ABIERTA	BOOL	FALSE	
+31.7	EV_CRUZ_REF_CERRADA	BOOL	FALSE	
+32.0	EV_DIRECTA_REF_ABIERTA	BOOL	FALSE	
+32.1	EV_DIRECTA_REF_CERRADA	BOOL	FALSE	
+32.2	EV_ABIENDO_CRUZ_REF	BOOL	FALSE	
+32.3	EV_CERRANDO_CRUZ_REF	BOOL	FALSE	
+32.4	EV_ABIENDO_DIRECTA_REF	BOOL	FALSE	
+32.5	EV_CERRANDO_DIRECTA_REF	BOOL	FALSE	
+32.6	EV_DIRECTA	BOOL	FALSE	
+32.7	EV_INVERSA	BOOL	FALSE	
+33.0	EV_BYPASS_INICIALIZADO	BOOL	FALSE	
+33.1	COM_RESERVA4	BOOL	FALSE	
+33.2	COM_RESERVA5	BOOL	FALSE	
+33.3	COM_RESERVA6	BOOL	FALSE	
+33.4	COM_RESERVA7	BOOL	FALSE	
+33.5	COM_RESERVA8	BOOL	FALSE	
+33.6	COM_RESERVA24	BOOL	FALSE	
+33.7	COM_RESERVA25	BOOL	FALSE	
+34.0	COM_CONSIGNA_REFRIG	REAL	0.000000e+000	
+38.0	COM_VEL_ENFRIAMIENTO	REAL	0.000000e+000	
+42.0	COM_DELAY_VALV_ROT	TIME	T#OMS	Consigna de tiempo de retardo de apertura de rotura de vacio
+46.0	EV_DELAY_VALV_ROT_TRANS	TIME	T#OMS	Tiempo transcurrido de retardo de rotura de vacio
+50.0	COM_SP_21_MANUAL	REAL	0.000000e+000	
+54.0	COM_SP_22_MANUAL	REAL	0.000000e+000	
+58.0	COM_SP_23_MANUAL	REAL	0.000000e+000	
+62.0	COM_SP_24_MANUAL	REAL	0.000000e+000	
+66.0	COM_SP_VACIO_MANUAL	REAL	0.000000e+000	
+70.0	COM_SP_POS_PRECAL_CAMP	REAL	0.000000e+000	
+74.0	HISTERESIS_CAMPANA	REAL	0.000000e+000	
+78.0	EV_SPZONA1_TEMP_ACTUAL	REAL	9.999000e+001	Setpoint actual del PID, dado por el Ramp Soak
+82.0	EV_SPZONA2_TEMP_ACTUAL	REAL	0.000000e+000	Setpoint actual del PID, dado por el Ramp Soak
+86.0	EV_SPZONA3_TEMP_ACTUAL	REAL	0.000000e+000	Setpoint actual del PID, dado por el Ramp Soak
+90.0	EV_SPZONA4_TEMP_ACTUAL	REAL	0.000000e+000	Setpoint actual del PID, dado por el Ramp Soak

Ilustración 195: DB17_2

+94.0	EV_SP_VACIO_ACTUAL	REAL	0.000000e+000	Setpoint actual del PID, dado por el Ramp Soak
+98.0	COM_REFRIG_POR_MEDIA	BOOL	FALSE	
+98.1	COM_REFRIG_POR_MAXIMO	BOOL	FALSE	
+98.2	COM_REFRIG_POR_MINIMO	BOOL	FALSE	
+98.3	EV_MESA1_ASIGNADA	BOOL	FALSE	
+98.4	EV_MESA2_ASIGNADA	BOOL	FALSE	
+98.5	COM_AVANCE_PROG_VACIO	BOOL	FALSE	
+98.6	COM_CICLO_PREPARADO	BOOL	FALSE	
+98.7	COM_ACTIVAR_CICLO	BOOL	FALSE	
+100.0	EV_TEMP_ESCALON_ACTUAL	INT	0	ESCALON ACTUAL PROGRAMADOR VACIO
+102.0	EV_VACIO_ESCALON_ACTUAL	INT	0	ESCALON ACTUAL PROGRAMADOR VACIO
+104.0	EV_OP21	REAL	0.000000e+000	
+108.0	EV_OP22	REAL	0.000000e+000	
+112.0	EV_OP23	REAL	0.000000e+000	
+116.0	EV_OP24	REAL	0.000000e+000	
+120.0	EV_OPVACIO	REAL	0.000000e+000	
+124.0	COM_RESET_RELOJ_AUTO	BOOL	FALSE	
+126.0	EV_SEG_CICLO_AUTO	INT	0	
+128.0	EV_MIN_CICLO_AUTO	INT	0	
+130.0	EV_HORAS_CICLO_AUTO	INT	0	
+132.0	EV_DIAS_CICLO_AUTO	INT	0	

Ilustración 196: DB17_3

4.4.1.5 DB60: PROGRAMA TEMPERATURA PUNTOS

Esta DB está formado por un entero que indica el número de puntos que forman el programa de temperatura seleccionado y por un array formado por variables de tipo PuntosProg. Esta DB es escrita por el SCADA, y recorrida por el PID para regular la temperatura de la máquina.

Dirección	Nombre	Tipo	Valor inicial	Comentario
0.0		STRUCT		
+0.0	NBR_PTS	INT	4	number of points -1
+2.0	PI	ARRAY[0..255]		
*8.0		"PuntosProg"		
=2050.0		END_STRUCT		

Ilustración 197: DB60

4.4.1.6 DB61: PROGRAMA TEMPERATURA EVENTOS

Esta DB está formado por un entero que indica el número de puntos que forman el programa de temperatura seleccionado y por un array formado por variables de tipo Eventos_Prog_Temp. Esta DB es escrita por el SCADA, y es recorrida por la FC801 para decodificar los eventos que ocurren a lo largo del programa de temperatura.

Dirección	Nombre	Tipo	Valor inicial	Comentario
0.0		STRUCT		
+0.0	Escalones	ARRAY[0..49]		
*2.0		"Eventos_Prog_Tem"		
=100.0		END_STRUCT		

Ilustración 198: DB61

4.4.1.7 DB70: DB60 PROGRAMA VACÍO PUNTOS

Esta DB está formado por un array formado por variables de tipo PuntosVacio. Esta DB es escrita por el SCADA, y recorrida por el PID para regular el vacío de la máquina.

Dirección	Nombre	Tipo	Valor inicial	Comentario
0.0		STRUCT		
+0.0	NBR_PTS	INT	4	number of points -1
+2.0	PI	ARRAY[0..255]		
*8.0		"PuntosVacio"		
=2050.0		END_STRUCT		

Ilustración 199: DB70

4.4.1.8 DB71: PROGRAMA VACÍO EVENTOS

Esta DB está formado por un array formado por variables de tipo Eventos_Prog_Vacio. Esta DB es escrita por el SCADA, y es recorrida por la FC801 para decodificar los eventos que ocurren a lo largo del programa de vacío.

Dirección	Nombre	Tipo	Valor inicial	Comentario
0.0		STRUCT		
+0.0	Escalonesvacío	ARRAY[0..49]		
*2.0		"Eventos_Prog_Va		
=100.0		END_STRUCT		

Ilustración 200: DB71

4.4.2 BLOQUE DE DATOS DE INSTANCIA

4.4.2.1 DB120 DB de instancia de la FBI

	Dirección	Declaración	Nombre	Tipo	Valor inicial	Valor actual
1	0.0	in	PV1	REAL	0.000000...	2.206988...
2	4.0	in	PV2	REAL	0.000000...	2.163440...
3	8.0	in	PV3	REAL	0.000000...	2.183710...
4	12.0	in	PV4	REAL	0.000000...	2.192561...
5	16.0	in	PWvacío	REAL	0.000000...	-2.14843...
6	20.0	in	Hab_Z1	BOOL	FALSE	TRUE
7	20.1	in	Hab_Z2	BOOL	FALSE	TRUE
8	20.2	in	Hab_Z3	BOOL	FALSE	TRUE
9	20.3	in	Hab_Z4	BOOL	FALSE	TRUE
10	20.4	in	Activar_ProgTemp	BOOL	FALSE	FALSE
11	20.5	in	Activar_ProgVac	BOOL	FALSE	FALSE
12	20.6	in	Reset_ProgTemp	BOOL	FALSE	FALSE
13	20.7	in	Reset_ProgVac	BOOL	FALSE	FALSE
14	21.0	in	Alarma90	BOOL	FALSE	FALSE
15	21.1	in	Alarma85_z1	BOOL	FALSE	FALSE
16	21.2	in	Alarma85_z2	BOOL	FALSE	FALSE
17	21.3	in	Alarma85_z3	BOOL	FALSE	FALSE
18	21.4	in	Alarma85_z4	BOOL	FALSE	FALSE
19	21.5	in	TiristorManual	BOOL	FALSE	FALSE
20	22.0	in	Tiristor1_MAN	REAL	0.000000...	0.000000...
21	26.0	in	Tiristor2_MAN	REAL	0.000000...	0.000000...
22	30.0	in	Tiristor3_MAN	REAL	0.000000...	0.000000...
23	34.0	in	Tiristor4_MAN	REAL	0.000000...	0.000000...
24	38.0	in	Act_Vacío_Manual	BOOL	FALSE	FALSE
25	40.0	in	ConsigVacío_MAN	REAL	0.000000...	0.000000...
26	44.0	in	Avance_Esc_temp	BOOL	FALSE	FALSE
27	44.1	in	Avance_Esc_vacío	BOOL	FALSE	FALSE
28	44.2	in	Lazo_temp_manual	BOOL	FALSE	FALSE
29	44.3	in	Lazo_vacío_manual	BOOL	FALSE	FALSE
30	46.0	in	SPZ1_temp_Man	REAL	0.000000...	8.000000...
31	50.0	in	SPZ2_temp_Man	REAL	0.000000...	8.000000...
32	54.0	in	SPZ3_temp_Man	REAL	0.000000...	8.000000...
33	58.0	in	SPZ4_temp_Man	REAL	0.000000...	8.000000...
34	62.0	in	SP_Vacío_Man	REAL	0.000000...	0.000000...
35	66.0	out	Zona1_OUT	WORD	W#16#0	W#16#0
36	68.0	out	Zona2_OUT	WORD	W#16#0	W#16#0
37	70.0	out	Zona3_OUT	WORD	W#16#0	W#16#0
38	72.0	out	Zona4_OUT	WORD	W#16#0	W#16#0
39	74.0	out	Vacío_OUT	WORD	W#16#0	W#16#0
40	76.0	out	Prog_Temp_On	BOOL	FALSE	FALSE
41	76.1	out	Prog_Vacío_On	BOOL	FALSE	FALSE
42	78.0	out	Escalon_Act_Temp	INT	0	3

Ilustración 201: DB120_1

42	78.0	out	Escalon_Act_Temp	INT	0	3
43	80.0	out	Escalon_Act_Vac	INT	0	6
44	82.0	out	T_Total_PrTemp	TIME	T#0MS	T#53M
45	86.0	out	T_Total_PrVacio	TIME	T#0MS	T#2H9M1...
46	90.0	out	T_RestTotal_PrTemp	TIME	T#0MS	T#23M56...
47	94.0	out	T_RestTotal_PrVacio	TIME	T#0MS	T#1H45M...
48	98.0	out	T_RestEscalon_PrTemp	TIME	T#0MS	T#23M56...
49	102.0	out	T_RestEscalon_PrVacio	TIME	T#0MS	T#1H45M...
50	106.0	out	RmpSoakTemp_END	BOOL	FALSE	FALSE
51	106.1	out	RmpSoakVacio_END	BOOL	FALSE	FALSE
52	108.0	out	SP_ZONA1	REAL	0.000000...	0.000000...
53	112.0	out	SP_ZONA2	REAL	0.000000...	0.000000...
54	116.0	out	SP_ZONA3	REAL	0.000000...	0.000000...
55	120.0	out	SP_ZONA4	REAL	0.000000...	0.000000...
56	124.0	out	SP_VACIO	REAL	0.000000...	0.000000...
57	128.0	out	OPZ1	REAL	0.000000...	0.000000...
58	132.0	out	OPZ2	REAL	0.000000...	0.000000...
59	136.0	out	OPZ3	REAL	0.000000...	0.000000...
60	140.0	out	OPZ4	REAL	0.000000...	0.000000...
61	144.0	out	OPVAC	REAL	0.000000...	0.000000...
62	148.0	stat	Cero	BOOL	FALSE	FALSE
63	148.1	stat	Uno	BOOL	TRUE	TRUE
64	150.0	stat	CeroReal	REAL	0.000000...	0.000000...
65	154.0	stat	UnoReal	REAL	1.000000...	1.000000...
66	158.0	stat	PID1_Gain	REAL	5.000000...	5.000000...
67	162.0	stat	PID2_Gain	REAL	5.000000...	5.000000...
68	166.0	stat	PID3_Gain	REAL	5.000000...	5.000000...
69	170.0	stat	PID4_Gain	REAL	5.000000...	5.000000...
70	174.0	stat	PIDvacio_Gain	REAL	-1.000000...	-3.000000...
71	178.0	stat	PID1_TI	TIME	T#2M44S...	T#2M
72	182.0	stat	PID2_TI	TIME	T#2M44S...	T#2M
73	186.0	stat	PID3_TI	TIME	T#2M44S...	T#2M
74	190.0	stat	PID4_TI	TIME	T#2M44S...	T#2M
75	194.0	stat	PIDvacio_TI	TIME	T#2S	T#1S
76	198.0	stat	PID1_Rate	TIME	T#41S22...	T#10S
77	202.0	stat	PID2_Rate	TIME	T#41S22...	T#10S
78	206.0	stat	PID3_Rate	TIME	T#41S22...	T#10S
79	210.0	stat	PID4_Rate	TIME	T#41S22...	T#10S
80	214.0	stat	PIDvacio_Rate	TIME	T#300MS	T#200MS

Ilustración 202: DB120_2

81	218.0	stat	PID1_OutHightLimit	REAL	9.000000...	9.000000...
82	222.0	stat	PID2_OutHightLimit	REAL	9.000000...	9.000000...
83	226.0	stat	PID3_OutHightLimit	REAL	9.000000...	9.000000...
84	230.0	stat	PID4_OutHightLimit	REAL	9.000000...	9.000000...
85	234.0	stat	PIDvacio_OutHightLimit	REAL	1.000000...	1.000000...
86	238.0	stat	PID1_OutLowLimit	REAL	0.000000...	0.000000...
87	242.0	stat	PID2_OutLowLimit	REAL	0.000000...	0.000000...
88	246.0	stat	PID3_OutLowLimit	REAL	0.000000...	0.000000...
89	250.0	stat	PID4_OutLowLimit	REAL	0.000000...	0.000000...
90	254.0	stat	PIDvacio_OutLowLimit	REAL	0.000000...	0.000000...
91	258.0	stat	SPdefecto_z1	REAL	2.500000...	0.000000...
92	262.0	stat	SPdefecto_z2	REAL	2.500000...	0.000000...
93	266.0	stat	SPdefecto_z3	REAL	2.500000...	0.000000...
94	270.0	stat	SPdefecto_z4	REAL	2.500000...	0.000000...
95	274.0	stat	SPdefectoVacio	REAL	0.000000...	0.000000...
96	278.0	stat	SPrateUp_z1	REAL	1.666667...	1.666667...
97	282.0	stat	SPrateUp_z2	REAL	1.666667...	1.666667...
98	286.0	stat	SPrateUp_z3	REAL	1.666667...	1.666667...
99	290.0	stat	SPrateUp_z4	REAL	1.666667...	1.666667...
100	294.0	stat	SPrateUp_Vacio	REAL	-6.500000...	-6.500000...
101	298.0	stat	SPrateDown_z1	REAL	1.666667...	1.666667...
102	302.0	stat	SPrateDown_z2	REAL	1.666667...	1.666667...
103	306.0	stat	SPrateDown_z3	REAL	1.666667...	1.666667...
104	310.0	stat	SPrateDown_z4	REAL	1.666667...	1.666667...
105	314.0	stat	SPrateDown_Vacio	REAL	0.000000...	0.000000...
106	318.0	stat	SP_H_Lim_Temp	REAL	9.000000...	9.000000...
107	322.0	stat	SP_L_Lim_Temp	REAL	0.000000...	0.000000...
108	326.0	stat	SP_H_Lim_Vacio	REAL	0.000000...	0.000000...
109	330.0	stat	SP_L_Lim_Vacio	REAL	-7.000000...	-7.000000...
110	334.0	stat	TimeToCont	TIME	T#0MS	T#0MS
111	338.0	stat	LMNGEN_OUT_vacio.LMN	REAL	0.000000...	0.000000...
112	342.0	stat	LMNGEN_OUT_vacio.LMN_HLM	REAL	0.000000...	1.000000...
113	346.0	stat	LMNGEN_OUT_vacio.LMN_LLM	REAL	0.000000...	0.000000...
114	350.0	stat	LMNGEN_OUT_vacio.R_MTR_TM	REAL	0.000000...	0.000000...
115	354.0	stat	LMNGEN_OUT_vacio.ARWHL_ON	BOOL	FALSE	FALSE
116	354.1	stat	LMNGEN_OUT_vacio.ARWLL_ON	BOOL	FALSE	TRUE
117	354.2	stat	LMNGEN_OUT_vacio.MAN_ON	BOOL	FALSE	FALSE
118	354.3	stat	LMNGEN_OUT_vacio.LMNGS_ON	BOOL	FALSE	FALSE
119	354.4	stat	LMNGEN_OUT_vacio.LMNR_ON	BOOL	FALSE	FALSE
120	356.0	stat	LMNGEN_IN_vacio.OutV	REAL	0.000000...	-7.87672...
121	360.0	stat	LMNGEN_IN_vacio.Hvar1	REAL	0.000000...	-7.87672...
122	364.0	stat	LMNGEN_IN_vacio.Hvar2	REAL	0.000000...	0.000000...

Ilustración 203: DB120_3

122	364.0	stat	LMNGEN_IN_vacio.Hvar2	REAL	0.000000...	0.000000...
123	368.0	stat	LMNGEN_IN_vacio.Hvar3	REAL	0.000000...	-6.44531...
124	372.0	stat	LMNGEN_IN_vacio.Hbit1	BOOL	FALSE	FALSE
125	372.1	stat	LMNGEN_IN_vacio.Hbit2	BOOL	FALSE	FALSE
126	372.2	stat	LMNGEN_IN_vacio.Hbit3	BOOL	FALSE	FALSE
127	374.0	stat	Struct13.real1	REAL	0.000000...	-1.09185...
128	378.0	stat	Struct13.real2	REAL	0.000000...	-1.09185...
129	382.0	stat	Struct13.real3	REAL	0.000000...	0.000000...
130	386.0	stat	Struct13.rel4	REAL	0.000000...	-1.09185...
131	390.0	stat	Struct13.bool1	BOOL	FALSE	FALSE
132	390.1	stat	Struct13.bool2	BOOL	FALSE	FALSE
133	390.2	stat	Struct13.bool3	BOOL	FALSE	FALSE
134	392.0	stat	Struct12.real1	REAL	0.000000...	-1.08172...
135	396.0	stat	Struct12.real2	REAL	0.000000...	-1.08172...
136	400.0	stat	Struct12.real3	REAL	0.000000...	0.000000...
137	404.0	stat	Struct12.rel4	REAL	0.000000...	-1.08172...
138	408.0	stat	Struct12.bool1	BOOL	FALSE	FALSE
139	408.1	stat	Struct12.bool2	BOOL	FALSE	FALSE
140	408.2	stat	Struct12.bool3	BOOL	FALSE	FALSE
141	410.0	stat	Struct11.real1	REAL	0.000000...	-1.10349...
142	414.0	stat	Struct11.real2	REAL	0.000000...	-1.10349...
143	418.0	stat	Struct11.real3	REAL	0.000000...	0.000000...
144	422.0	stat	Struct11.rel4	REAL	0.000000...	-1.10349...
145	426.0	stat	Struct11.bool1	BOOL	FALSE	FALSE
146	426.1	stat	Struct11.bool2	BOOL	FALSE	FALSE
147	426.2	stat	Struct11.bool3	BOOL	FALSE	FALSE
148	428.0	stat	Struct22.real1	REAL	0.000000...	0.000000...
149	432.0	stat	Struct22.real2	REAL	0.000000...	9.000000...
150	436.0	stat	Struct22.real3	REAL	0.000000...	0.000000...
151	440.0	stat	Struct22.real4	REAL	0.000000...	0.000000...
152	444.0	stat	Struct22.bool1	BOOL	FALSE	FALSE
153	444.1	stat	Struct22.bool2	BOOL	FALSE	TRUE
154	444.2	stat	Struct22.bool3	BOOL	FALSE	FALSE
155	444.3	stat	Struct22.bool4	BOOL	FALSE	FALSE
156	444.4	stat	Struct22.bool5	BOOL	FALSE	FALSE
157	446.0	stat	Struct14.real1	REAL	0.000000...	-1.09628...
158	450.0	stat	Struct14.real2	REAL	0.000000...	-1.09628...
159	454.0	stat	Struct14.real3	REAL	0.000000...	0.000000...
160	458.0	stat	Struct14.rel4	REAL	0.000000...	-1.09628...
161	462.0	stat	Struct14.bool1	BOOL	FALSE	FALSE
162	462.1	stat	Struct14.bool2	BOOL	FALSE	FALSE

Ilustración 204: DB120_4

163	462.2	stat	Struct14.bool3	BOOL	FALSE	FALSE
164	464.0	stat	Struct21.real1	REAL	0.000000...	0.000000...
165	468.0	stat	Struct21.real2	REAL	0.000000...	9.000000...
166	472.0	stat	Struct21.real3	REAL	0.000000...	0.000000...
167	476.0	stat	Struct21.real4	REAL	0.000000...	0.000000...
168	480.0	stat	Struct21.bool1	BOOL	FALSE	FALSE
169	480.1	stat	Struct21.bool2	BOOL	FALSE	TRUE
170	480.2	stat	Struct21.bool3	BOOL	FALSE	FALSE
171	480.3	stat	Struct21.bool4	BOOL	FALSE	FALSE
172	480.4	stat	Struct21.bool5	BOOL	FALSE	FALSE
173	482.0	stat	Struct23.real1	REAL	0.000000...	0.000000...
174	486.0	stat	Struct23.real2	REAL	0.000000...	9.000000...
175	490.0	stat	Struct23.real3	REAL	0.000000...	0.000000...
176	494.0	stat	Struct23.real4	REAL	0.000000...	0.000000...
177	498.0	stat	Struct23.bool1	BOOL	FALSE	FALSE
178	498.1	stat	Struct23.bool2	BOOL	FALSE	TRUE
179	498.2	stat	Struct23.bool3	BOOL	FALSE	FALSE
180	498.3	stat	Struct23.bool4	BOOL	FALSE	FALSE
181	498.4	stat	Struct23.bool5	BOOL	FALSE	FALSE
182	500.0	stat	Struct24.real1	REAL	0.000000...	0.000000...
183	504.0	stat	Struct24.real2	REAL	0.000000...	9.000000...
184	508.0	stat	Struct24.real3	REAL	0.000000...	0.000000...
185	512.0	stat	Struct24.real4	REAL	0.000000...	0.000000...
186	516.0	stat	Struct24.bool1	BOOL	FALSE	FALSE
187	516.1	stat	Struct24.bool2	BOOL	FALSE	TRUE
188	516.2	stat	Struct24.bool3	BOOL	FALSE	FALSE
189	516.3	stat	Struct24.bool4	BOOL	FALSE	FALSE
190	516.4	stat	Struct24.bool5	BOOL	FALSE	FALSE
191	518.0	stat	ProximoSliceVac	INT	0	6
192	520.0	stat	ProximoSliceTemp	INT	0	3
193	522.0	stat	flancopositivo	BOOL	FALSE	FALSE
194	522.1	stat	flancopositivo1	BOOL	FALSE	FALSE
195	522.2	stat	resetz1	BOOL	FALSE	FALSE
196	522.3	stat	resetz2	BOOL	FALSE	FALSE
197	522.4	stat	resetz3	BOOL	FALSE	FALSE
198	522.5	stat	resetz4	BOOL	FALSE	FALSE

Ilustración 205: DB120_5

4.4.2.2 DB200 DB de instancia de la FB2

	Dirección	Declaración	Nombre	Tipo	Valor inicia	Valor actua
1	0.0	stat	flancopositivo1	BOOL	FALSE	FALSE
2	0.1	stat	flancopositivo	BOOL	FALSE	FALSE
3	0.2	stat	flancopositivo2	BOOL	FALSE	FALSE
4	0.3	stat	flancopositivo3	BOOL	FALSE	FALSE
5	0.4	stat	flancopositivo4	BOOL	FALSE	FALSE
6	0.5	stat	flancopositivo5	BOOL	FALSE	FALSE
7	0.6	stat	flancopositivo6	BOOL	FALSE	FALSE
8	0.7	stat	flancopositivo7	BOOL	FALSE	FALSE
9	1.0	stat	flancopositivo8	BOOL	FALSE	FALSE
10	1.1	stat	iniciandoprtemp	BOOL	FALSE	FALSE
11	1.2	stat	iniciandoprvacio	BOOL	FALSE	FALSE
12	1.3	stat	flanconegativo	BOOL	FALSE	FALSE
13	1.4	stat	flanconegativo1	BOOL	FALSE	FALSE
14	1.5	stat	flancopositivo10	BOOL	FALSE	FALSE
15	1.6	stat	flancopositivo11	BOOL	FALSE	FALSE
16	1.7	stat	detectorflanco1	BOOL	FALSE	FALSE
17	2.0	stat	detectorflanco2	BOOL	FALSE	FALSE
18	2.1	stat	detectorflanco3	BOOL	FALSE	TRUE
19	2.2	stat	detectorflanco4	BOOL	FALSE	TRUE

Ilustración 206: DB200

4.4.2.3 DB210 DB de instancia de la FB10

	Dirección	Declaración	Nombre	Tipo	Valor inicia	Valor actua
1	0.0	stat	EtapaAuto_0	BOOL	FALSE	TRUE
2	0.1	stat	EtapaAuto_1	BOOL	FALSE	FALSE
3	0.2	stat	EtapaAuto_2	BOOL	FALSE	FALSE
4	0.3	stat	EtapaAuto_3	BOOL	FALSE	FALSE
5	0.4	stat	EtapaAuto_4	BOOL	FALSE	FALSE
6	0.5	stat	EtapaAuto_5	BOOL	FALSE	FALSE
7	0.6	stat	EtapaAuto_6	BOOL	FALSE	FALSE
8	0.7	stat	EtapaAuto_7	BOOL	FALSE	FALSE
9	1.0	stat	EtapaAuto_8	BOOL	FALSE	FALSE
10	1.1	stat	EtapaAuto_9	BOOL	FALSE	FALSE
11	1.2	stat	EtapaAuto_10	BOOL	FALSE	FALSE
12	1.3	stat	EtapaAuto_11	BOOL	FALSE	FALSE
13	1.4	stat	DelayRotActivo	BOOL	FALSE	FALSE
14	2.0	stat	DelayRotTiempo	S5TIME	S5T#0MS	S5T#500...
15	4.0	stat	FlancoP1	BOOL	FALSE	FALSE
16	4.1	stat	FlancoN1	BOOL	FALSE	FALSE

Ilustración 207: DB210

4.4.2.4 DB300 DB de instancia de la FB4

	Dirección	Declaración	Nombre	Tipo	Valor inicia	Valor actua
1	0.0	stat	CONT	INT	0	8
2	2.0	stat	SEG	INT	0	53
3	4.0	stat	MIN	INT	0	16
4	6.0	stat	HORAS	INT	0	3
5	8.0	stat	DIAS	INT	0	0
6	10.0	stat	T	BOOL	FALSE	FALSE

Ilustración 208: DB300

4.5 TIPO DE DATOS DE USUARIO (UDT)

Los tipos de datos de usuario son estructuras de datos creadas por un mismo y que, una vez definidas, pueden usarse en todo el programa como tipos de datos en la declaración de variables o como tipo de datos para variables en un bloque de datos (DB). En este programa se han creado las siguientes:

4.5.1 UDT1: UDT_TERMOPARES

Dirección	Nombre	Tipo	Valor inicial	Comentario
0.0		STRUCT		
+0.0	RANGO_SUP	REAL	0.000000e+000	Rango superior unidades de ingenieria
+4.0	RANGO_INF	REAL	0.000000e+000	Rango superior unidades de ingenieria
+8.0	V_ANALOGICO	INT	0	Lectura directa de E/S
+10.0	V_REAL	REAL	0.000000e+000	Valor de medida escalada en unidades de ingenieria
+14.0	OFFSET	REAL	0.000000e+000	Offset de calibracion
+18.0	GANANCIA	REAL	1.000000e+000	Ganancia de calibracion
+22.0	HABILITADO	BOOL	FALSE	Termopar habilitado para su uso
+22.1	LECTURA_INCORRECTA	BOOL	FALSE	Medicion de termopar fuera de rango, desconectado o rotura de hilo
+22.2	SELECCIONADO_CONTROL	BOOL	FALSE	Termopar seleccionado para control de PID
+22.3	RESERVA3	BOOL	FALSE	
=24.0		END_STRUCT		

Ilustración 209: UDT1

4.5.2 UDT2: UDT_SENSORES_INFRAROJOS

Dirección	Nombre	Tipo	Valor inicial	Comentario
0.0		STRUCT		
+0.0	RANGO_SUP	REAL	1.500000e+002	Rango superior unidades de ingenieria
+4.0	RANGO_INF	REAL	0.000000e+000	Rango superior unidades de ingenieria
+8.0	V_ANALOGICO	INT	0	Lectura directa de E/S
+10.0	V_REAL	REAL	0.000000e+000	Valor de medida escalada en unidades de ingenieria
+14.0	OFFSET	REAL	0.000000e+000	Offset de calibracion
+18.0	GANANCIA	REAL	1.000000e+000	Ganancia de calibracion
+22.0	HABILITADO	BOOL	FALSE	Termopar habilitado para su uso
+22.1	LECTURA_INCORRECTA	BOOL	FALSE	Medicion de termopar fuera de rango, desconectado o rotura de hilo
+22.2	SELECCIONADO_CONTROL	BOOL	FALSE	Termopar seleccionado para control de PID
+22.3	RESERVA3	BOOL	FALSE	
=24.0		END_STRUCT		

Ilustración 210: UDT2

4.5.3 UDT3: UDT_SENSOR_VACIO

Dirección	Nombre	Tipo	Valor inicial	Comentario
0.0		STRUCT		
+0.0	RANGO_SUP	REAL	-6.000000e+000	Rango superior unidades de ingenieria
+4.0	RANGO_INF	REAL	0.000000e+000	Rango superior unidades de ingenieria
+8.0	V_ANALOGICO	INT	0	Lectura directa de E/S
+10.0	V_REAL	REAL	0.000000e+000	Valor de medida escalada en unidades de ingenieria
+14.0	OFFSET	REAL	0.000000e+000	Offset de calibracion
+18.0	GANANCIA	REAL	1.000000e+000	Ganancia de calibracion
+22.0	HABILITADO	BOOL	FALSE	Termopar habilitado para su uso
+22.1	LECTURA_INCORRECTA	BOOL	FALSE	Medicion de termopar fuera de rango, desconectado o rotura de hilo
+22.2	SELECCIONADO_CONTROL	BOOL	FALSE	Termopar seleccionado para control de PID
+22.3	RESERVA3	BOOL	FALSE	
=24.0		END_STRUCT		

Ilustración 211: UDT3

4.5.4 UDT4: UDT_SENSOR_CAMPANA

Dirección	Nombre	Tipo	Valor inicial	Comentario
0.0		STRUCT		
+0.0	RANGO_SUP	REAL	0.000000e+000	Rango superior unidades de ingenieria
+4.0	RANGO_INF	REAL	1.000000e+002	Rango superior unidades de ingenieria
+8.0	V_ANALOGICO	INT	0	Lectura directa de E/S
+10.0	V_REAL	REAL	0.000000e+000	Valor de medida escalada en unidades de ingenieria
+14.0	OFFSET	REAL	0.000000e+000	Offset de calibracion
+18.0	GANANCIA	REAL	1.000000e+000	Ganancia de calibracion
+22.0	HABILITADO	BOOL	FALSE	Termopar habilitado para su uso
+22.1	LECTURA_INCORRECTA	BOOL	FALSE	Medicion de termopar fuera de rango, desconectado o rotura de hilo
+22.2	SELECCIONADO_CONTROL	BOOL	FALSE	Termopar seleccionado para control de PID
+22.3	RESERVA3	BOOL	FALSE	
=24.0		END_STRUCT		

Ilustración 212: UDT4

4.5.5 UDT5: EVENTOS_PROG_TEMP

Dirección	Nombre	Tipo	Valor inicial	Comentario
0.0		STRUCT		
+0.0	Cerrar_Campana	BOOL	FALSE	
+0.1	Comenzar_Prog_Vacio	BOOL	FALSE	
+0.2	Reserva1	BOOL	FALSE	
+0.3	Reserva2	BOOL	FALSE	
+0.4	Reserva3	BOOL	FALSE	
+0.5	Reserva4	BOOL	FALSE	
+0.6	Reserva5	BOOL	FALSE	
+0.7	Reserva6	BOOL	FALSE	
=2.0		END_STRUCT		

Ilustración 213: UDT5

4.5.6 UDT6: EVENTOS_PROG_VACIO

Dirección	Nombre	Tipo	Valor inicial	Comentario
0.0		STRUCT		
+0.0	Fin_Prog_Vacio	BOOL	FALSE	
+0.1	Ini_Refrig_Fin_Prog_Temp	BOOL	FALSE	
+0.2	Reserva1	BOOL	FALSE	
+0.3	Reserva2	BOOL	FALSE	
+0.4	Reserva3	BOOL	FALSE	
+0.5	Reserva4	BOOL	FALSE	
+0.6	Reserva5	BOOL	FALSE	
+0.7	Reserva6	BOOL	FALSE	
=2.0		END_STRUCT		

Ilustración 214: UDT6

4.5.7: UDT7 PUNTOS_PROG

Dirección	Nombre	Tipo	Valor inicial	Comentario
0.0		STRUCT		
+0.0	OUTV	REAL	0.000000e+000	
+4.0	THV	TIME	T#0MS	
=8.0		END_STRUCT		

Ilustración 215: UDT7

4.5.8 UDT8: PUNTOS_PROG_VACIO

Dirección	Nombre	Tipo	Valor inicial	Comentario
0.0		STRUCT		
+0.0	OUTV	REAL	0.000000e+000	
+4.0	THV	TIME	T#0MS	
=8.0		END_STRUCT		

Ilustración 216: UDT8

5 CONCLUSIONES Y FUTUROS TRABAJOS

Tras la realización de este trabajo de fin de grado he tenido la suerte de conocer el mundo de la industria, con él pude confirmar lo que ya pensaba, la Automatización Industrial es la parte que más me gusta de mi carrera. Gracias a este proyecto pude demostrar mi capacidad en mi puesto de trabajo, lo que me ha abierto puertas en el mundo laboral y ha hecho que pueda dedicarme a ello.

Creo que la Automatización y la Robótica son un mundo muy interesante y agradecido ya que te pone a prueba constantemente haciendo que tengas que actualizar tus conocimientos, como, por ejemplo, aprender a programar distintas marcas en distintos softwares de programación cada vez que te enfrentas a un trabajo nuevo. Por el contrario, también pienso que es un mundo muy duro, ya que las instalaciones en las fábricas son contrarreloj y te hacen trabajar bajo mucha presión, no todo el mundo puede trabajar bajo dichas condiciones durante mucho tiempo.

En resumen, es muy gratificante cuando sacas el proyecto adelante dejas una máquina de este tipo trabajando en una fábrica.

Como futuros trabajos y mejoras de esta máquina existen varios proyectos, a los programas de vacío y de temperatura se les puede añadir hasta 6 eventos más.

Se prevee crear un modo convección donde tanto el aire que se está calentando como el de refrigeración se recirculen en ambas direcciones y así homogeneizar la temperatura dentro de la campana.

También habrá una mejora futura, donde se cambiará el controlador Johnson de la enfriadora, y la UTA pasará a controlarse desde el PLC actual de la máquina.

6 BIBLIOGRAFÍA

https://cache.industry.siemens.com/dl/files/056/18652056/att_70833/v1/S7prv54_s.pdf

<https://programacionsiemens.com/ob-de-fallo/>