



**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE GUAYAQUIL**

FACULTAD DE INGENIERÍAS

TESIS

**PREVIA A LA OBTENCIÓN DEL TÍTULO DE:
INGENIERO ELECTRÓNICO**

**DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE
SUPERVISIÓN Y CONTROL DIDÁCTICO PARA EL
LABORATORIO DE FABRICACIÓN FLEXIBLE -MPS DE LA
UPS-G**

AUTORES:

**VICTOR VIRGILIO ORTEGA CELY
ULISES AQUILES NÚÑEZ ANTÓN**

DIRECTOR:

ING. GARY AMPUÑO AVILÉS

GUAYAQUIL, 2013

DECLARATORIA DE RESPONSABILIDAD

Los diseños elaborados y las conclusiones del presente documento, son de exclusiva responsabilidad de los autores.

Guayaquil, Julio 03 del 2013

VICTOR VIRGILIO ORTEGA CELY
ULISES AQUILES NÚÑEZ ANTÓN

DEDICATORIA

A Dios, por brindarme salud, vida y por darme esta oportunidad de salir adelante en mi vida.

Dedicado a mis padres que son un pilar importante en mi vida, con su ayuda económica, anímica y su amor pude lograr metas propuestas.

Especialmente a mi madre que con sus consejos, aciertos y su ejemplo de una madre luchadora y emprendedora me ha enseñado a no rendirme y ser una mejor persona.

A mi familia y amigos, que han estado conmigo en toda esta etapa de mi vida y que me han enseñado a ser una persona de bien y un buen profesional.

(f) _____
VICTOR VIRGILIO ORTEGA CELY

DEDICATORIA

Este proyecto es dedicado a mi familia. A mi madre, Gloria, que fue mi eje principal. Mi padre, Polo, que me encaminó en mi carrera universitaria. Para finalizar a mis hermanos, Polo y Madelayne, que en algún momento me brindaron su apoyo incondicional y oportuno.

(f) _____
ULISES AQUILES NÚÑEZ ANTÓN

AGRADECIMIENTO

Gracias a Dios por darme una familia que siempre se ha preocupado por mí.

A los profesores por impartirnos sus conocimientos y experiencias.

Al ingeniero Luis Neira que me encaminó en la vida industrial y me dio la oportunidad de conocer el mundo industrial.

A mi tutor de tesis por estar siempre alerta, preocupado para poder dar el último paso y poder cumplir la meta que es la Ingeniería.

A mis compañeros de trabajo especialmente a los ingenieros: Ronald Suriaga y Erwin Wolf que con sus conocimientos y experiencias en el área de automatización supieron guiarme y darme la pauta para la realización del tema propuesto.

A mis compañeros y amigos de la Universidad, especialmente a mi compañero de tesis, que siempre hemos compartido conocimiento y experiencias vividas.

A la universidad y cada una de las personas que contribuyeron con la conclusión de este trabajo.

(f)_____

VICTOR VIRGILIO ORTEGA CELY

AGRADECIMIENTO

Mis agradecimientos son, primero a Dios por permitirme llegar hasta este punto de mi vida. Después un inmenso “Gracias” a mis padres, Polo y Gloria, que fueron el pilar fundamental durante mis estudios. Por último estoy profundamente agradecido a mi compañero de tesis Victor Ortega, que con su empeño, dedicación y responsabilidad logramos alcanzar el objetivo.

(f) _____
ULISES AQUILES NÚÑEZ ANTÓN

ÍNDICE

CAPÍTULO I

1	EL PROBLEMA.....	1
1.1	Planteamiento del problema	1
1.2	Delimitación del problema	1
1.3	Objetivos.....	2
1.3.1	Objetivo general	2
1.3.2	Objetivos específicos	2
1.4	Justificación	3
1.5	Hipótesis	3
1.6	Variables e indicadores.....	4
1.7	Metodología.....	4
1.7.1	Método Teórico y Sistémico.....	4
1.8	Población y Muestra	5
1.9	Descripción de la propuesta.....	5
1.9.1	Beneficiarios	5
1.9.2	Impacto	6

CAPÍTULO II

2	MARCO TEÓRICO.....	7
2.1	Antecedentes.....	7
2.2	Fabricación flexible	8
2.2.1	FMC (Células de fabricación flexible)	8
2.2.2	FMS (Sistemas de fabricación flexible)	9
2.3	MPS - 500 FMS	9
2.3.1	Sistema transportador	12
2.3.2	Distribución	13
2.3.3	Verificación	14
2.3.4	Manipulación I.....	15

2.3.5	Proceso	16
2.3.6	Visión	18
2.3.7	Robot y ensamblaje	19
2.3.8	Almacén.....	20
2.3.9	Manipulación II	22
2.3.10	Clasificación	23
2.4	Automatización industrial	24
2.4.1	Grados de la automatización	25
2.4.1.1	Nivel I.....	26
2.4.1.2	Nivel II.....	26
2.4.1.3	Nivel III	26
2.4.1.4	Nivel IV	26
2.4.1.5	Nivel V	27
2.5	Redes industriales	28
2.5.1	Arquitectura de redes.....	28
2.5.1.1	Topología en bus	28
2.5.1.1.1	Ventajas de la topología en bus	28
2.5.1.1.2	Desventajas de la topología de bus.....	29
2.5.1.2	Topología en estrella	29
2.5.1.2.1	Ventajas de la topología en estrella	30
2.5.1.2.2	Desventajas de la topología en estrella.....	30
2.5.1.3	Topología en árbol.....	31
2.5.1.3.1	Ventajas de la topología en árbol	32
2.5.1.3.2	Desventajas de la topología en árbol	32
2.5.1.4	Topología en anillo.....	33
2.5.1.4.1	Ventajas de la topología en anillo.....	33
2.5.1.4.2	Desventajas de la topología en anillo	34
2.6	Tipos de redes.....	35
2.6.1	Red de factoría.....	35
2.6.2	Red de planta	35
2.6.3	Red de célula	35
2.6.4	Bus de campo	35
2.6.4.1	Ventajas de un bus de campo	36

2.6.4.2	Desventajas de un bus de campo	36
2.6.4.3	Clasificación de los buses de campo	37
2.6.4.3.1	Buses de alta velocidad y funcionalidad baja.....	37
2.6.4.3.2	Buses de alta velocidad y funcionalidad media.....	37
2.6.4.3.3	Buses de alto rendimiento	38
2.6.4.3.4	Buses para áreas de seguridad intrínseca.....	39
2.7	Red ASI	39
2.7.1	Propiedades principales	39
2.7.2	Módulos AS-Interface	41
2.7.2.1	Módulos activos.....	41
2.7.2.2	Módulos pasivos	42
2.7.2.3	Módulo de acoplamiento	42
2.7.2.4	Módulo de usuario	43
2.7.3	Repetidor y extensor.....	43
2.7.3.1	Repetidor	43
2.7.3.2	Extensor.....	44
2.7.4	Terminal de direccionamiento	45
2.8	Profibus.....	46
2.8.1	Elementos del bus	47
2.8.1.1	Nodos activos	47
2.8.1.2	Nodos pasivos.....	47
2.8.2	Repetidor	47
2.8.3	Conexiones físicas	47
2.8.3.1	RS-485	47
2.8.3.2	MBP.....	48
2.8.3.3	RS-485 IS	48
2.8.3.4	MBP IS	48
2.8.3.5	Fibra óptica.....	48
2.8.4	Comunicación.....	48
2.8.4.1	Aplicación monomaestro.....	49
2.8.4.2	Aplicación multimaestro.....	49
2.8.4.3	Características de la comunicación.....	49
2.8.5	Terminadores de bus.....	50

2.8.6	Tipos de redes Profibus	50
2.8.6.1	Profibus DP	50
2.8.6.2	Profibus PA	51
2.8.6.3	Profibus FMS.....	51
2.9	SCADA.....	52
2.9.1	Funciones primordiales	52
2.9.2	Beneficios	53
2.9.3	Objetivos.....	53
2.10	Controlador lógico programable (PLC).....	54
2.11	Instrumentación industrial	55
2.11.1	Presión	56
2.11.2	Sensor capacitivo.....	57
2.11.3	Sensor inductivo	57
2.11.4	Silenciador	58
2.11.5	Micro interruptor	58
2.11.6	Cilindro neumático	59
2.11.7	Terminal de I/O	59
2.11.8	Copa de succión (ventosa).....	60
2.11.9	Válvula electro neumática	61
2.11.10	Sensor opto-electrónico	61
2.11.11	Filtro con regulador para aire comprimido.....	62
2.11.11.1	Filtro	62
2.11.11.2	Regulador	62
2.11.12	Sensor magnético.....	63
2.11.13	Encoder.....	64
2.11.13.1	Encoder incremental	65
2.11.13.2	Encoder absoluto	65
2.12	Variador de frecuencia (VFD).....	65

CAPÍTULO III

3	SOFTWARE WinCC SCADA Y ENLACE DE VARIABLES.....	67
3.1	Funcionamiento del proceso.....	67

3.2	Descripción de la partición funcional.....	68
3.3	Preparar la configuración para la aplicación en el software WinCC	
SCADA	69
3.4	Crear un proyecto	69
3.5	Especificar las propiedades del proyecto.....	71
3.5.1	Ciclos de actualización	71
3.5.2	Teclas de acceso rápido	72
3.5.3	Interfaz de usuario y diseño.....	72
3.6	Propiedades de la lista de equipos	73
3.6.1	Llamar a la lista de equipos	73
3.6.2	General.....	73
3.6.3	Arranque	73
3.6.4	Parámetros	73
3.6.5	<i>Runtime</i> de gráficos	74
3.7	Módulos de función en WinCC.....	75
3.7.1	Equipo.....	75
3.7.2	Administración de variables	76
3.7.3	Diseñador de gráficos	76
3.7.4	Registro de alarmas	76
3.7.5	Registro de variables	77
3.7.6	Diseñador de informes.....	77
3.7.7	Global scripts.....	77
3.7.8	Librería de texto.....	77
3.7.9	Administrador de usuario	77
3.7.10	Referencia cruzada	78
3.8	Estructura de la comunicación.....	79
3.8.1	Administrador de datos.....	79
3.8.2	Driver de comunicación.....	79
3.8.3	Driver hardware.....	80
3.9	Configurar variables de proceso para SIMATIC S7	80
3.9.1	Variables de proceso.....	80
3.9.2	Nuevo grupo	81
3.9.3	Nueva variable (de proceso).....	81

3.9.4	Variables internas	81
3.10	Términos de operación y monitoreo	82
3.10.1	Pantallas gráficas	82
3.10.2	Elementos de las pantallas gráficas	83
3.10.3	Valores digitales	83
3.10.4	Configuración del editor de gráficos	84
3.10.5	Activar/ desactivar una paleta.....	84
3.10.6	Tamaño de la pantalla.....	84
3.11	Paleta de objetos gráficos	85
3.11.1	Configurar objetos	85
3.11.2	Objetos estándar	85
3.11.3	Objetos inteligentes	86
3.11.3.1	Ventana de aplicación.....	86
3.11.3.2	Ventana de imagen	87
3.11.3.3	Control	87
3.11.3.4	Objeto OLE	87
3.11.3.5	Campo E/S	87
3.11.3.6	Barra	88
3.11.3.7	Objeto gráfico	88
3.11.3.8	Vista de estado.....	88
3.11.3.9	Lista de texto	88
3.11.4	Objetos de windows	89
3.11.4.1	Botón	89
3.11.4.2	Casilla de verificación	90
3.11.4.3	Radio-box	90
3.11.4.4	Botón redondo	90
3.11.4.5	Objeto deslizante	90
3.12	Objetos gráficos.....	91
3.12.1	Alinear objetos.....	91
3.12.2	Seleccionar objetos	92
3.12.3	Cambiar atributos o propiedades	92
3.12.4	Borrar objeto.....	92
3.12.5	Cambiar la visualización del objeto.....	93

3.12.6	Mover objeto	93
3.12.7	Rotación.....	93
3.12.8	Simetría.....	93
3.13	Librería de símbolos	94
3.14	Paleta de niveles	97
3.14.1	Renombrar niveles	97
3.14.2	Cambiar la asignación de niveles	97
3.15	Paleta de estilos	98
3.16	Barra de herramientas de variables.....	99
3.17	Definir una imagen inicial	100
3.18	Disparador	102
3.19	Propiedades del objeto.....	103
3.19.1	Cuadro de diálogo dinámico.....	103
3.19.2	Acciones en propiedades del objeto	104
3.19.3	Conexión a variable	104
3.19.4	Acciones en eventos del objeto	104
3.19.5	Conexión directa.....	105
3.20	Administrador de usuario	106
3.20.1	Configuración	106
3.20.2	Abrir el administrador de usuario.....	107
3.20.3	Crear un usuario	108
3.21	Listado de variables y dirección	110
3.21.1	Enlace controlador estación distribución.....	110
3.21.2	Enlace controlador estación verificación.....	111
3.21.3	Enlace controlador estación proceso	112
3.21.4	Enlace controlador estación manipulación I.....	113
3.21.5	Enlace controlador estación clasificación.....	114
3.21.6	Enlace controlador estación manipulación II	115
3.21.7	Enlace controlador estación almacenamiento.....	116
3.21.8	Enlace controlador estación transportador	119
3.22	Variables internas	124
3.23	Tipos de datos.....	125
3.24	Animaciones y acciones	125

3.24.1	Diálogo dinámico	126
3.24.2	Acción en C	127
3.24.3	Acción en VBS	128
3.24.4	Funciones con números	129
3.24.5	Operadores relacionales.....	130
3.24.6	Operadores lógicos	131
3.24.7	Variable	131
3.24.8	Conexión directa.....	132
3.25	Cómo y porqué el uso del software WinCC SCADA	133

CAPÍTULO IV

4	COMPARACIÓN ENTRE PANTALLAS.....	137
4.1	Pantalla de inicio	137
4.2	Sistema transportador	139
4.3	Comisionar	141
4.4	Red profibus	142
4.5	Estación de distribución	144
4.6	Estación de verificación.....	147
4.7	Estación de manipulación I.....	150
4.8	Estación de procesamiento	153
4.9	Estación de visión.....	155
4.10	Estación de robot y ensamblaje	157
4.11	Almacén.....	159
4.12	Estación de manipulación II	161
4.13	Estación de clasificación	164
	CRONOGRAMA DE ACTIVIDADES.....	168
	PRESUPUESTO	170
	CONCLUSIONES	171
	RECOMENDACIONES	172
	REFERENCIAS BIBLIOGRÁFICAS.....	173

ÍNDICE DE TABLAS

Tabla 1:	Tipos de datos	125
Tabla 2:	Operadores lógicos para escribir en el cuadro de diálogo dinámico	126
Tabla 3:	Operadores aritméticos	130
Tabla 4:	Operadores relacionales.....	130
Tabla 5:	Tablas lógicas	131
Tabla 6:	Cronograma de actividades	168
Tabla 7:	Presupuesto	170

ÍNDICE DE FIGURAS

Figura 1:	MPS - 500 FMS UPS sede Guayaquil.....	12
Figura 2:	Sistema transportador	13
Figura 3:	Estación de distribución	14
Figura 4:	Estación de verificación.....	15
Figura 5:	Estación de manipulación I.....	16
Figura 6:	Estación de proceso	18
Figura 7:	Estación de visión.....	19
Figura 8:	Estación de robot y ensamblaje	20
Figura 9:	Estación de almacén	21
Figura 10:	Estación de manipulación II	22
Figura 11:	Estación de clasificación	24
Figura 12:	Niveles jerárquicos de la automatización	27
Figura 13:	Topología en bus	29
Figura 14:	Topología en estrella	31
Figura 15:	Topología en árbol.....	33
Figura 16:	Topología en anillo.....	34
Figura 17:	Módulo activo de Siemens	42
Figura 18:	Módulo de acoplamiento y módulo de usuario	43
Figura 19:	Repetidor y prolongador	44
Figura 20:	Extensor de Siemens.....	45

Figura 21: Cable profibus con conectores.....	46
Figura 22: Conector profibus con resistencia terminadora	50
Figura 23: Esquema básico de un sistema SCADA.....	54
Figura 24: Controlador lógico programable.....	55
Figura 25: Indicador de presión	56
Figura 26: Sensor capacitivo.....	57
Figura 27: Sensor inductivo	57
Figura 28: Silenciador	58
Figura 29: Micro interruptor	59
Figura 30: Cilindro neumático	59
Figura 31: Terminal I/O	60
Figura 32: Presostato y copa de succión	60
Figura 33: Válvula electro neumática	61
Figura 34: Sensor opto-electrónico	61
Figura 35: Filtro con regulador para aire comprimido.....	63
Figura 36: Sensor magnético sobre un pistón	64
Figura 37: Motor con encoder.....	64
Figura 38: Variador de frecuencia.....	66
Figura 39: Descripción funcional.....	69
Figura 40: Crear un proyecto	70
Figura 41: Crear proyecto nuevo.....	71
Figura 42: Propiedades del proyecto.....	72
Figura 43: Propiedades del equipo.....	75
Figura 44: Módulos de función en WinCC.....	78
Figura 45: Agregar un driver	80
Figura 46: Propiedades de variable	82
Figura 47: Ejemplo de pantallas gráficas	84
Figura 48: Editor de gráficos.....	85
Figura 49: Objetos estándar	86
Figura 50: Objetos <i>Smart</i>	89
Figura 51: Objetos de windows	91
Figura 52: Barra de alineación	91
Figura 53: Propiedades del objeto.....	92

Figura 54: Barra de herramientas estándar.....	94
Figura 55: Símbolo para mostrar librería.....	94
Figura 56: Librerías.....	95
Figura 57: Imagen con atributo en original.....	95
Figura 58: Imagen con atributo en sombreado.....	96
Figura 59: Imagen con atributo en sólido	96
Figura 60: Imagen con atributo en contorno	96
Figura 61: Barra de herramientas niveles.....	97
Figura 62: Configuración de niveles	98
Figura 63: Paleta de estilos	99
Figura 64: Barra de herramientas de variables.....	100
Figura 65: Definir imagen inicial opción 1	101
Figura 66: Definir imagen inicial opción 2.....	101
Figura 67: Salir de <i>runtime</i>	102
Figura 68: Configurar el <i>trigger</i> predeterminado.....	103
Figura 69: Dinamizar atributos	104
Figura 70: Dinamizar ejecuciones.....	105
Figura 71: Teclas de acceso directo	107
Figura 72: Administrador de usuario	108
Figura 73: Crear un usuario y asignar autorizaciones.....	109
Figura 74: Ejemplo de cómo seleccionar el tipo de permiso	109
Figura 75: Dialogo dinámico	127
Figura 76: Acción en C	128
Figura 77: Animación por medio de una variable.....	132
Figura 78: Animación por medio de una conexión directa.....	132
Figura 79: Ejecución de un evento por medio de una conexión directa	133
Figura 80: Evento realizado dando clic con el ratón y ejecutándose con una conexión directa	135
Figura 81: Un evento realizado dando clic con el ratón y ejecutándose bajo una acción en C.....	136
Figura 82: Pantalla principal (actual).....	138
Figura 83: Pantalla principal (nueva).....	138
Figura 84: Pantalla del sistema transportador (actual).....	139

Figura 85: Pantalla del sistema transportador demo (nuevo).....	140
Figura 86: Pantalla del sistema transportador (nuevo).....	140
Figura 87: Pantalla de comisionar (actual).....	141
Figura 88: Pantalla de comisionar (nueva).....	142
Figura 89: Pantalla de la red profibus (actual).....	143
Figura 90: Pantalla de la red profibus (nueva).....	143
Figura 91: Pantalla de la estación de distribución (actual)	145
Figura 92: Pantalla de la estación de distribución demo (nueva).....	146
Figura 93: Pantalla de la estación de distribución(nueva)	146
Figura 94: Pantalla de la estación de verificación (actual)	148
Figura 95: Pantalla de la estación de verificación demo (nueva).....	149
Figura 96: Pantalla de la estación de verificación (nueva)	149
Figura 97: Pantalla de la estación de manipulación I (actual)	151
Figura 98: Pantalla de la estación de manipulación I demo (nueva).....	152
Figura 99: Pantalla de la estación de manipulación I (nueva)	152
Figura 100: Pantalla de la estación de procesamiento (actual)	154
Figura 101: Pantalla de la estación de procesamiento demo (nueva)	154
Figura 102: Pantalla de la estación de procesamiento (nueva)	155
Figura 103: Pantalla de la estación de visión (actual).....	156
Figura 104: Pantalla de la estación de visión (nueva).....	157
Figura 105: Pantalla de la estación de robot y ensamblaje (actual).....	158
Figura 106: Pantalla de la estación de robot y ensamblaje (nueva).....	158
Figura 107: Pantalla de la estación de almacén (actual)	160
Figura 108: Pantalla de la estación de almacén demo (nueva)	160
Figura 109: Pantalla de la estación de almacén (nueva)	161
Figura 110: Pantalla de la estación de manipulación II (actual)	163
Figura 111: Pantalla de la estación de manipulación II demo (nueva)	163
Figura 112: Pantalla de la estación de manipulación II (nueva)	164
Figura 113: Pantalla de estación de clasificación (actual)	166
Figura 114: Pantalla de estación de clasificación demo (nueva)	166
Figura 115: Pantalla de estación de clasificación (nueva)	167
Figura 116: Pantalla de inicio	179
Figura 117: Conexión directa para navegar a la ventana de "mensajes y alarmas".	180

Figura 118: Pantalla de la red profibus	181
Figura 119: Pantalla de la ventana de comisionado	182
Figura 120: Mensaje a mostrar en la pantalla	184
Figura 121: La pantalla del sistema transportador demo	185
Figura 122: Pantalla de la ventana de distribución demo	187
Figura 123: Conexión directa para navegar a la ventana de "verificación"	188
Figura 124: Pantalla de la ventana de verificación demo	189
Figura 125: Conexión directa para navegar a la ventana de "distribución"	190
Figura 126: Pantalla de la ventana de manipulación I demo	191
Figura 127: Conexión directa para navegar a la ventana de "proceso"	191
Figura 128: Pantalla de la ventana de proceso demo	192
Figura 129: Conexión directa por medio de la configuración de un botón.....	193
Figura 130: Pantalla de la ventana de visión.....	194
Figura 131: Conexión directa para navegar a la ventana de "inicio"	194
Figura 132: Pantalla de la ventana de robot y ensamblaje.....	195
Figura 133: Pantalla de la ventana de almacén ASRS 20 demo	196
Figura 134: Pantalla de la ventana de manipulación II demo	198
Figura 135: Pantalla de la ventana de clasificación demo	199
Figura 136: Pantalla de la ventana de mensajes y alarmas	200
Figura 137: Pantalla de la ventana de curvas	200
Figura 138: Pantalla del sistema transportador	201
Figura 139: Pantalla de la ventana de distribución	202
Figura 140: Animación por medio de cuadro de diálogo dinámico.....	203
Figura 141: Pantalla de la ventana de verificación	204
Figura 142: Pantalla de la ventana de manipulación I	205
Figura 143: Pantalla de la ventana de proceso	206
Figura 144: Pantalla de la ventana de almacén ASRS 20	207
Figura 145: Pantalla de la ventana de manipulación II.....	208
Figura 146: Pantalla de la ventana de clasificación	209

ANEXOS

MANUAL DE USUARIO	178
A1 Pantalla de inicio	179
A2 Pantalla red profibus	181
A3 Pantalla de comisionado	182
A4 Sistema transportador demo	184
A5 Distribución demo	186
A6 Verificación demo	188
A7 Manipulación I demo	190
A8 Proceso demo	192
A9 Visión	193
A10 Robot y ensamblaje	195
A11 Almacén ASRS20 demo	195
A12 Manipulación II demo	197
A13 Clasificación demo	198
A14 Pantalla de mensajes y alarmas	199
A15 Curvas	200
A16 Banda transportadora	201
A17 Distribución	202
A18 Verificación	203
A19 Manipulación I	204
A20 Proceso	205
A21 Almacén ASRS20	206
A22 Manipulación II	208
A23 Clasificación	209

RESUMEN

AÑO	ALUMNOS	DIRECTOR DE TESIS	TEMA TESIS
2013	VICTOR VIRGILIO ORTEGA CELY ULISES AQUILES NÚÑEZ ANTÓN	ING. GARY AMPUÑO AVILÉS	DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE SUPERVISIÓN Y CONTROL DIDÁCTICO PARA EL LABORATORIO DE FABRICACIÓN FLEXIBLE -MPS DE LA UPS-G

La aplicación realizada tiene como objetivo modificar el sistema de supervisión y control del sistema MPS - 500 del Laboratorio de Fabricación Flexible, el SCADA a utilizar es WinCC y su fabricante es Siemens, se trabaja con el mismo sistema implementado por motivo que la Universidad Politécnica Salesiana sede Guayaquil cuenta con las licencias para su correcto funcionamiento. El tema nació debido a la necesidad de los alumnos y del docente de conseguir un sistema SCADA donde se pueda obtener o visualizar la mayor parte de las señales de entrada y de salida de los autómatas programables, con esto se logra supervisar, controlar todos los equipos, mejorando las animaciones de las pantallas y con una adquisición de datos en un tiempo real , donde el estándar de adquisición de datos es de 2 segundos. Este proyecto ayudará a los estudiantes que se encuentran cursando los ciclos superiores y en la cátedra como Robótica, Automatización Industrial e Informática Industrial. La aplicación cuenta con múltiples pantallas animadas donde observamos el proceso, advertencias, alarmas, curvas y da la facilidad de interactuar entre el usuario y la máquina. El trabajo realizado tiene diferentes maneras de efectuar la supervisión de un proceso y ejecutar con distintas acciones un evento, utilizando la mayoría de las herramientas que brinda el software.

Palabras claves

Diseño, implementación, aplicación, supervisión, control, animación, WinCC SCADA, interacción entre usuario y máquina.

ABSTRACT

YEAR	STUDENTS	THESIS DIRECTOR	THESIS TOPIC
2013	VICTOR VIRGILIO ORTEGA CELY ULISES AQUILES NÚÑEZ ANTÓN	ENG. GARY AMPUÑO AVILÉS	DESIGN AND IMPLEMENTATION OF A DIDACTIC SYSTEM MONITORING AND CONTROL FOR FLEXIBLE MANUFACTURING LABORATORY - MPS OF UPS- G

The application is to amend on the monitoring system and control system MPS - 500 of Flexible Manufacturing Laboratory, the WinCC SCADA is used and its manufacturer is Siemens, working with the same system implemented on the basis that the Guayaquil-based Salesian Polytechnic University has licenses for proper operation. The theme was born due to the need of students and teachers to get a SCADA system where you can get or view most of the input signals and output PLC's, this is accomplished with monitoring, controlling all teams improving displays animations and data acquisition in real time, where the standard data acquisition is 2 seconds. This project will help students who are pursuing higher education levels and the cathedra as Robotics, Industrial Automation and Industrial Informatics. The application features multiple animated screens where we observe the process, warnings, alarms, bends and gives the facility to interact with the user and the machine. The work has different ways to make monitoring a process with various actions and execute an event, using most of the tools provided by the software.

Keywords

Design, implementation, application, monitoring, control, animation, WinCC SCADA, interaction between user and machine.

CAPÍTULO I

1 EL PROBLEMA

1.1 Planteamiento del problema

La Universidad Politécnica Salesiana sede Guayaquil cuenta con un Laboratorio de Fabricación Flexible y el sistema que maneja es el MPS - 500; que es un sistema compatible, modular y versátil.

Este sistema cuenta con una aplicación realizada en WinCC SCADA para supervisión y control pero las ventanas no son muy dinámicas ni animadas; utiliza pocas herramientas con las que cuenta el software WinCC, el sistema de supervisión, control es básico y poco dinámico; no interactúa mucho con el usuario. Este laboratorio lo utilizan con mayor frecuencia los alumnos que se encuentran cursando el sexto semestre hasta el décimo semestre de las carreras de Ingeniería Eléctrica y Electrónica.

Es un problema para los estudiantes su manejo remotamente porque no se visualiza el proceso con alguna animación y el docente no encuentra muchas herramientas utilizadas que el software posee, esto complica la exposición sobre el uso del software y sus diferentes herramientas y utilidades que brinda el programa.

1.2 Delimitación del problema

El tema se enfoca en modificar y diseñar una aplicación en WinCC SCADA para poder realizar la interfaz humano – máquina, que supervise y controle el proceso de fabricación flexible de todos los módulos que tienen comunicación con los PLC's.

Se va a diseñar diferentes pantallas para que el docente de la Universidad Politécnica Salesiana sede Guayaquil pueda usar como instrumento de cátedra y el estudiante posea una herramienta avanzada de entrenamiento y que permita obtener una visión más amplia de lo que se puede encontrar en el área industrial.

Se va a realizar el levantamiento de información neumático y eléctrico de todas las señales digitales de cada uno de los módulos (en total 10 módulos constituido con 8 PLC's Siemens S7 300, un esclavo Festo y una cámara que solo ve la silueta de la pieza y la compara con una imagen grabada en su memoria).

Todo esto se va efectuar en los años 2012 y 2013.

1.3 Objetivos

1.3.1 Objetivo general

- Mejorar el sistema de supervisión y control del sistema MPS - 500 del Laboratorio de Fabricación Flexible de la Universidad Politécnica Salesiana.

1.3.2 Objetivos específicos

- Analizar los sistemas que componen el Laboratorio de Fabricación Flexible: Neumáticos, eléctricos, electro-neumáticos, automatización, redes industriales.
- Implementar herramientas de control, supervisión, adquisición de datos y gestión del Laboratorio de Fabricación Flexible.
- Modificar y diseñar un sistema SCADA con la herramienta WinCC que permita controlar y supervisar los diferentes componentes del laboratorio.
- Realizar guías de laboratorio para el docente y el alumno sobre el uso de la aplicación.

1.4 Justificación

El tema nació debido a la necesidad de los alumnos y de los docentes de la Universidad Politécnica Salesiana sede Guayaquil de conseguir un sistema SCADA donde se pueda obtener o visualizar todas las señales de entrada y de salida de los autómatas programables para poder monitorear, supervisar y controlar todo el sistema de fabricación flexible MPS 500.

Con esto se logra ayudar a las cátedras que se encuentran a partir del sexto semestre como Robótica, Automatización Industrial e Informática Industrial por todos los componentes que tiene el equipo como sensores, autómatas programables, la parte de comunicación (profibus), ASI y el SCADA en donde nos vamos a enfocar en crear un programa más dinámico y didáctico donde el estudiante o el docente pueda interactuar con el equipo y visualizar el proceso.

El sistema SCADA nos brinda la posibilidad de observar el comportamiento del proceso y nos da una herramienta para tomar decisiones mediante curvas y sistemas de mensajes y alarmas.

Mediante estos resultados y las herramientas que presenta la aplicación se logra mejoras en el proceso y proyecciones a futuro.

1.5 Hipótesis

El laboratorio adquirido por la Universidad Politécnica Salesiana no cuenta con un buen SCADA para el monitoreo del sistema, se pretende ayudar con los conocimientos obtenidos para realizar una mejor aplicación de supervisión y control de los equipos del sistema MPS – 500.

El aprendizaje del software y las variables del sistema MPS – 500, puede mejorarse con un nueva aplicación que sea más dinámico y amigable para el usuario.

Para realizar una aplicación ideal y mejorar la visualización remota del proceso se podrían incrementar las pantallas y variables necesarias.

Los cambios realizados en la aplicación, sería una ayuda al docente ya que facilitaría la enseñanza hacia los alumnos en sus clases.

1.6 Variables e Indicadores

Variable SCADA: Los indicadores son supervisión del proceso, adquisición de datos, control de equipos y software de programación.

Variable Programa: Los indicadores son control automatizado, diseños gráficos y visualización del proceso.

Variable Remoto: Los indicadores son control del proceso, facilidad de trabajo y detección de fallas.

1.7 Metodología

1.7.1 Método Teórico y Sistemático

Para la ejecución del proyecto lo primero que se realizó es el levantamiento de la información como datos del proceso, funcionamiento del sistema MPS – 500, fotos para realizar la programación, etc.

Se utilizó el software WinCC SCADA para realizar la programación, poder monitorear y controlar desde un punto remoto los equipo y elementos finales del sistema MPS – 500. Este programa se lo realizó con gráficos, animaciones amigables y dinámicas para el usuario, en este caso son los estudiantes de la Universidad Politécnica Salesiana.

Se consultó con archivos de Siemens ya que se utilizó su software de programación WinCC SCADA, también se buscó información en internet, textos de ayuda y se obtuvo ayuda por parte de ingenieros que trabajan en el área de automatización de procesos.

Se utilizó la mayoría de los beneficios que ofrece el software WinCC SCADA, como métodos de programación visual basic, diálogo dinámico, diseñadores de gráficos, se usó el protocolo profibus para la comunicación entre PLC's, etc.

1.8 Población y Muestra

Población: se puede definir a los alumnos que se encuentran en la carrera de Ingeniería Eléctrica y Electrónica, ya que se dieron varias charlas del proyecto realizado. Se explicó algo general sobre el funcionamiento del sistema MPS – 500, el programa realizado a partir de los datos obtenidos en hardware y software, funcionamiento y uso del programa desde la PC que se encuentra en el Laboratorio de Fabricación Flexible de la Universidad Politécnica Salesiana sede Guayaquil.

Muestra: se pueden definir a los alumnos que se encuentran en las materias de Robótica, Automatización e Informática Industrial.

1.9 Descripción de la propuesta

1.9.1 Beneficiarios

El mayor beneficiario será la Universidad Politécnica Salesiana sede Guayaquil, los docentes universitarios y alumnos de las materias de Robótica, Automatización e Informática Industrial.

1.9.2 Impacto

Los docentes tendrán un material de ayuda para poder ofrecer una mejor enseñanza en las clases.

Los alumnos contarán con una herramienta especializada para un correcto aprendizaje y estar mejor preparados profesionalmente para el futuro laboral.

CAPÍTULO II

2 MARCO TEÓRICO

2.1 Antecedentes

En el área industrial se diseñan e implementan sistemas de supervisión y control para manejar un proceso o varios procesos, esto implica muchas piezas mecánicas, sensores, elementos finales, elementos eléctricos, elementos neumáticos, elementos hidráulicos, equipos electrónicos, etc. Donde estos necesitan ser monitoreados y controlados para poder obtener una mayor eficiencia en producción y mantenimiento por las diferentes herramientas y respuestas que obtenemos de las estadísticas del rendimiento del proceso.

Con el sistema de supervisión y control se logra obtener todo el control e información de varios procesos que pueden ser visualizados en varias pantallas de forma remota y poder tomar decisiones acerca de lo que se está observando para mejorar el proceso.

En un mundo muy competitivo y por los avances tecnológicos los estudiantes y los profesionales tienen que esforzarse, actualizarse y autoeducarse cada día más, por eso las instituciones educativas para lograr una educación eficiente necesitan invertir en equipos que van ayudar al estudiantes a obtener mayores conocimientos sobre lo que se encuentra en el área industrial.

Por eso los diferentes fabricantes de equipos tiene una línea didáctica para los que estén interesados en el aprendizaje de dichos equipos e indiferente con la marca que se está practicando, los principios y las metodologías son muy similares.

2.2 Fabricación flexible

Los sistemas de fabricación flexible están formados por un grupo de máquinas y equipos auxiliares unidos mediante un sistema de control y transporte, que permiten fabricar piezas de forma automática.

La fabricación flexible ayuda a la reducción de costos de producción e inventario, hay un incremento en el mejoramiento de la calidad del producto.

Estos tipos de tecnologías o de sistemas pueden dividirse en 2 segmentos:

FMS (Sistemas de fabricación flexible)

FMC (Células de fabricación flexible)

2.2.1 FMC (Células de fabricación flexible)

Un FMC es un conjunto de componentes eléctricos y mecánicos que ejecutan un proceso en particular o es un paso para ejecutar un proceso de fabricación que tiene más elementos eléctricos y mecánicos.

Una célula o módulo puede ser separada debido al requerimiento que se tenga o a la utilidad que le va a dar el usuario para el proceso de fabricación solicitada.

La célula o el módulo no está limitada a un solo tipo de proceso, se puede acomodar a distintos procesos donde utilicen las mismas características físicas.

La unión de equipos en una célula o módulo nos brinda la oportunidad de realizar una tarea básica como manipulación de piezas, taladrado, inspección de piezas o almacenamiento.

2.2.2 FMS (Sistemas de fabricación flexible)

Es un conjunto de máquinas o equipos de fabricación dedicada a una sola tarea, dando flexibilidad debido a las diferentes combinaciones que se pueden realizar con las estaciones o los módulos de operaciones simples.

La consecuencia de poder realizar estas combinaciones es la capacidad de fabricar piezas o ensamblado usando el mismo conjunto de máquinas o equipos.

El sistema completo está controlado y supervisado por medio de software a través de computadoras o cualquier dispositivo de control y supervisión el cual nos da la oportunidad de fabricar colectivamente diferentes productos desde el comienzo hasta el final de la línea de proceso.

2.3 MPS - 500 FMS

El MPS - 500 es una máquina de entrenamiento de Festo donde se va a poder manejar, comprender y analizar el funcionamiento de cada una de las piezas, dispositivos o equipos que comprende el sistema y está involucrado en este proceso la Ingeniería Mecánica, Eléctrica, Neumática, Automatización y Redes Industriales.

El proceso del MPS - 500 es el siguiente:

Arrancar la banda con los pallets transportadores, una vez que el transportador se está ejecutando correctamente la primera estación en ejecutarse es la de distribución, aquí se encuentran almacenadas las piezas para ser transportadas, el siguiente paso es la estación de verificación donde comprueba la altura de la pieza y verifica si es negra la pieza, una vez realizada esta acción el equipo automáticamente toma la decisión de pasar a la siguiente estación o rechaza la pieza que no cumple con los requisitos.

El sistema de transporte por medio de los pallets transportadores lleva la pieza a la siguiente estación, esta estación manipula la pieza para la siguiente estación que se llama estación de procesamiento.

La estación de procesamiento o proceso prueba y perfora las piezas, una vez realizada dicha acción regresa la pieza a la estación de manipulación y la lleva al transportador.

La estación de manipulación es un puente entre la estación del transportador y la estación de procesamiento.

Una vez que la pieza se encuentra en la banda transportadora se dirige a la estación de visión, esta estación toma una foto y compara esa imagen con una imagen guardada en su memoria, si la pieza es correcta o incorrecta le manda comandos al transportador para que le avise a la estación de robot y ensamblaje que la pieza es aceptada o rechazada.

Cuando la estación da esa información continua desplazándose el transportador hasta la estación de robot y ensamble, en esta estación se va a ejecutar 2 acciones:

Si la pieza es rechazada por la cámara el robot va a depositar las piezas en los depósitos de rechazo de piezas, si la pieza es aceptada el robot comienza el ensamblaje del cilindro esperando que tenga todos los componentes que se requieren, una vez ensamblado regresa la pieza ensamblada a los pallets transportadores.

Una vez ensamblada las piezas de trabajo nos dirigimos a la estación de almacenamiento, esta estación comienza a ejecutar su programa cuando la estación de clasificación no se encuentra funcionando, la estación de almacenamiento es un auxiliar para la estación de clasificación porque el final del proceso es identificar y clasificar las piezas de trabajo.

La estación de almacenamiento guarda y retira piezas de la estación dependiendo de la orden que le da el transportador que a su vez esa orden se la da la estación de clasificación.

El transportador es un nexo entre la estación de clasificación y la estación de almacenamiento.

La siguiente estación es la de manipulación, esta estación traslada las piezas de los pallets transportadores hacia la estación de clasificación.

Cuando no se encuentra en funcionamiento la estación de clasificación y se encuentran piezas de trabajo después de la estación de almacenamiento, la estación de manipulación recoge las piezas del pallet transportador y las almacena en las resbaladeras su estación, porque no puede regresar la pieza al proceso, cuando el pallet transportador es detectado con pieza de trabajo en la estación de verificación va dar una alarma de error de que la pieza a regresado al proceso, cuando sucede eso hay que retirar la pieza de trabajo y acusar la falla en la estación de la banda transportadora.

La comunicación con el sistema de supervisión y control la realiza por medio de una red profibus.



Figura 1: MPS - 500 FMS UPS sede Guayaquil

Fuente: Laboratorio de Fabricación Flexible UPS Guayaquil, año 2013

2.3.1 Sistema transportador

La estación de transporte es un nexo de comunicación entre las estaciones, el escucha a todas las estaciones que están conectadas y les da la orden para que ejecute sus respectivas acciones.

El sistema transportador contiene pallets transportadores codificados, es decir, se puede identificar cual es el pallet y en donde se encuentra.

Nos va a dar la oportunidad de saber si el pallet tiene una pieza en las estaciones de proceso por medio de sensores ópticos reflectivos.

El sistema regula el tránsito de los pallets, es decir, si el pallet sale de alguna estación se espera a que llegue a la siguiente estación para dar paso a que se desplacen los pallets que vienen atrás, es decir se mueven uno por uno a la estación siguiente.

El módulo utiliza una red llamada ASI para interactuar con las entradas y las salidas de los dispositivos que se encuentran en la banda y se comunica con los otros módulos.



Figura 2: Sistema transportador

Fuente: Laboratorio de Fabricación Flexible UPS Guayaquil, año 2013

2.3.2 Distribución

Esta estación tiene como función distribuir piezas de trabajo del compartimiento de almacenamiento. El depósito de almacenamiento puede contener hasta 8 piezas. El nivel de llenado del depósito se controla por medio de un sensor a través de un haz de luz. Un cilindro de doble efecto empuja las piezas de trabajo individualmente para que sean distribuidas.

El módulo cambiador agarra las piezas de trabajo utilizando una copa de succión. Un interruptor de vacío comprueba si una pieza de trabajo ha sido recogida. El brazo de la unidad de transferencia, que es accionado por un actuador rotativo, transmite la pieza de trabajo del punto de transferencia a la siguiente estación.



Figura 3: Estación de distribución

Fuente: Laboratorio de Fabricación Flexible UPS Guayaquil, año 2013

2.3.3 Verificación

La estación determina las características físicas de la pieza de trabajo insertada. El módulo de detección identifica si la pieza es no negra por medio de un sensor de proximidad óptico y un sensor capacitivo detecta cada pieza sin considerar el color.

Un sensor retro - reflectivo controla si el módulo de elevación se puede desplazar o no evitando que tropiece con objetos que se encuentren en su recorrido.

El sensor analógico del módulo de medición determina la altura de la pieza. La señal de salida es digitalizada por medio de un comparador con un valor umbral ajustable. Un cilindro lineal guía las piezas correctas a la siguiente estación y las piezas rechazadas van a la resbaladera de la parte inferior para ser contabilizadas.



Figura 4: Estación de verificación

Fuente: Laboratorio de Fabricación Flexible UPS Guayaquil, año 2013

2.3.4 Manipulación I

El dispositivo de manipulación recupera las piezas con la ayuda de una pinza neumática, que está equipada con un sensor óptico, el sensor identifica si se encuentra una pieza en la pinza, todas las piezas son transportadas hasta la estación

siguiente, vamos a encontrar sensores magnéticos en los extremos y en medio para indicar la posición y luego de que la estación siguiente ejecute su operación la pieza es manipulada de regreso hasta el pallet transportador que se encuentra en la banda.

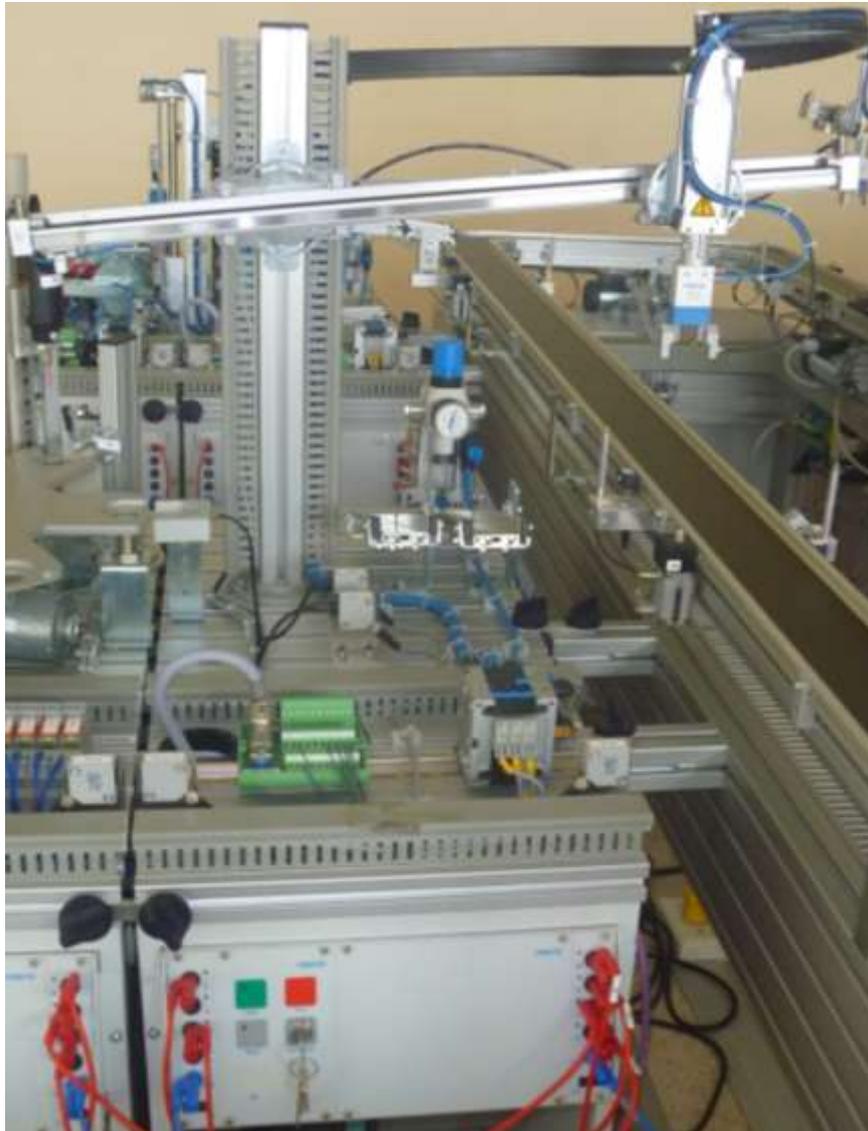


Figura 5: Estación de manipulación I

Fuente: Laboratorio de Fabricación Flexible UPS Guayaquil, año 2013

2.3.5 Proceso

Esta estación depende de un tablero rotativo donde se colocan las piezas de trabajo que son probadas y procesadas. El proceso comienza con un sensor capacitivo que detecta si hay pieza para comenzar el ciclo de trabajo, el tablero rotativo es

impulsado por un motor de corriente continua. La posición del tablero rotativo es dada por un sensor inductivo.

Se encuentra un sensor capacitivo que indica que hay una pieza disponible en el tablero rotativo esto hace que baje un actuador para comprobar el hueco y un sensor inductivo indica que el actuador realiza el recorrido completo.

En el siguiente paso encontramos otro sensor capacitivo que indica que hay una pieza disponible en el tablero rotativo esto hace que se active un actuador para sujetar la pieza de trabajo, después se enciende el taladro y desciende para hacer la perforación.



Figura 6: Estación de proceso

Fuente: Laboratorio de Fabricación Flexible UPS Guayaquil, año 2013

2.3.6 Visión

La cámara toma la foto y comienza a procesar la imagen, primero la pasa a escala de grises, luego la transforma en una señal binaria para al final filtrar la imagen, el programa lo que hace es comparar la imagen con una imagen guardada en su memoria y le manda los datos de pieza aceptada o rechazada al sistema transportador.



Figura 7: Estación de visión

Fuente: Laboratorio de Fabricación Flexible UPS Guayaquil, año 2013

2.3.7 Robot y ensamblaje

Esta estación comienza a trabajar cuando el sistema transportador le envía datos que recibe de la estación de Visión, si la pieza fue rechazada el robot la va alojar en el contenedor de piezas rechazadas. Hay un límite de piezas que se pueden rechazar, una vez que se excede de este límite el robot detiene su funcionamiento.

Si la pieza es aceptada entonces el robot ejecuta su programa de ensamblaje:

Agarra la pieza que se encuentra en el pallet transportador, para ingresarla en el torno, una vez ejecutado el torno, el robot vuelva a agarrar la pieza para verificar si la pieza es negra o no negra y seleccionar el pistón adecuado, ya insertado el pistón, el robot se traslada a coger resortes y tapas.

El robot traslada la pieza ensamblada hacia el pallet transportador.



Figura 8: Estación de robot y ensamblaje

Fuente: Laboratorio de Fabricación Flexible UPS Guayaquil, año 2013

2.3.8 Almacén

Este módulo entra en operación cuando no está funcionando el módulo de clasificación o está lleno o donde no haya donde descargar las piezas para que no regresen al proceso.

Una vez en operación la pinza va agarrar la pieza que se encuentra en el transportador dando la señal el sensor de barrera de luz (cuando en el receptor hay luz quiere decir que no hay ningún obstáculo "pieza" y cuando se interrumpe la luz quiere decir que se encuentra algún obstáculo) esa señal el transportador se la hace conocer al almacén para que comience a trabajar o no haga ninguna operación.

Una vez que el almacén sabe que hay pieza comienza a ejecutar su programa y a ingresar piezas en el almacenamiento sabiendo que las celdas de almacenamiento se encuentran vacías o llenas, dependiendo de que si algunas de las celdas están ocupadas comienza desde la primera posición que se encuentra vacía.

Una vez restablecido el módulo de clasificación el almacén comienza a sacar piezas a espera de que haya un transportador vacío, y el orden de salida de las piezas es el siguiente:

El primero que entra es el primero que sale.



Figura 9: Estación de almacén

Fuente: Laboratorio de Fabricación Flexible UPS Guayaquil, año 2013

2.3.9 Manipulación II

El dispositivo de manipulación recupera las piezas con la ayuda de una pinza neumática, que está equipado con un sensor óptico, el sensor indica si se encuentra la pieza en la pinza, todas las piezas son transportadas hasta la estación siguiente y nos va indicar la posición por medio de sensores magnéticos en los extremos y en medio, luego de que la pieza se encuentre en la estación siguiente y dicha estación ejecuta su programa la pinza es transportada de regreso sin pieza hacia la banda, cuando se encuentra lleno la estación siguiente deposita las piezas en las resbaladeras de su módulo.

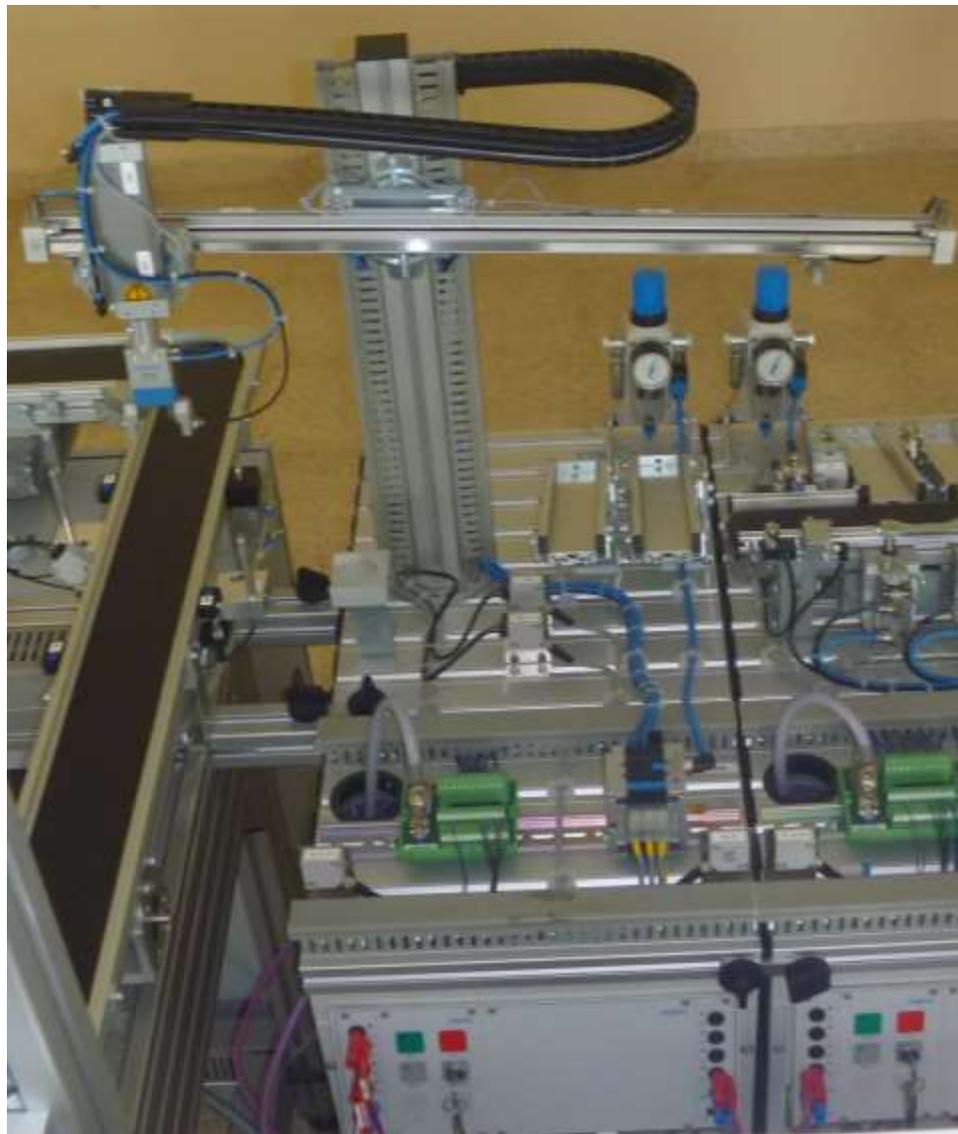


Figura 10: Estación de manipulación II

Fuente: Laboratorio de Fabricación Flexible UPS Guayaquil, año 2013

2.3.10 Clasificación

Esta estación ordena las piezas de trabajo en 3 resbaladeras. Un sensor óptico detecta las piezas insertadas en el comienzo de la banda transportadora. Las características de las piezas son sus colores (rojas, negras y plateadas) esto es detectado por un sensor óptico y un sensor inductivo, luego de saber el respectivo color de las piezas se activa unos cilindros transportando las piezas a sus respectivas resbaladeras.

Se encuentra un sensor retro - reflectivo en el inicio de las resbaladeras, si la pieza interrumpe el haz de luz por corto tiempo este nos indica que la pieza ya ingreso en las resbaladeras y está a la espera de más piezas para seguir llenando sus respectivas resbaladeras.

Si la pieza interrumpe el haz de luz por un tiempo más prolongado este nos indica que la o las resbaladeras se encuentran llenas y que necesitan ser desocupadas.



Figura 11: Estación de clasificación

Fuente: Laboratorio de Fabricación Flexible UPS Guayaquil, año 2013

2.4 Automatización industrial

La automatización industrial incluye a los sistemas computarizados que reemplaza las operaciones manuales, es decir las tareas realizadas por personas ahora son ejecutadas por máquinas.

Esto ayuda a incrementar la productividad del trabajo, reducir los gastos en materiales, aumentar la calidad del producto y mejorar las condiciones del trabajador en su ambiente laboral y social.

La automatización aporta para mecanizar el proceso, es decir, se suministra a los trabajadores herramientas para ayudar en el trabajo físico y mental, esto favorece a que el hombre se desempeñe en las funciones de supervisión y monitoreo.

En la época actual la automatización es imprescindible para la industria, esto ayuda a mantener o mejorar la competitividad, los estándares de calidad y niveles de producción que demanda el mercado interno y externo.

2.4.1 Grados de la automatización

Los grados de la automatización de los procesos industriales se encuentra constituido por una organización de jerarquías establecida en niveles. Se determina como un procedimiento de fabricación ayudado por computadoras, este método introduce al proceso y a la producción eficiente conceptos como gestión empresarial, programación y planificación de las tareas, etc., donde debe cumplirse el principio de:

Planificar de arriba hacia abajo de la organización jerárquica.

Implementar de abajo hacia arriba de la organización jerárquica.

La base de la organización comprende el área de control que son autómatas inteligentes, robots, sensores, actuadores, equipos y máquinas, todo esto se unen en un sistema control jerárquico y distribuido que posibilita la toma de decisiones de la parte gerencial.

Estos componentes lógicos y de control proporciona una interacción hombre – máquina, es decir el sistema de control interactúa con el usuario.

En la pirámide de niveles jerárquicos, encontramos 5 niveles que son los siguientes:

2.4.1.1 Nivel I

Este es el nivel de entradas y salidas, los instrumentos nos da la información de la condición en que encuentra el estado de las variables. El hombre realiza la función de ver que todos los dispositivos se encuentren en buen estado y que estén funcionando correctamente, es decir, el hombre es un normalizador en el control del proceso. En este nivel encontramos dispositivos de campo, instrumentos, maquinarias, motores, etc. Con esto se ejecuta las acciones para la productividad de la industria.

2.4.1.2 Nivel II

Este es el nivel de control, aquí se encuentran los dispositivos lógicos de control, tales como autómatas programables (PLC), tarjetas de control, ordenadores industriales, etc. Este nivel se encarga de controlar los dispositivos del nivel anterior y de mandar dicha información al nivel posterior.

2.4.1.3 Nivel III

Este es el nivel de supervisión y control. Este nivel utiliza ordenadores y dispositivos computacionales con software o aplicaciones que ayudan a supervisar, controlar, adquirir datos. Este nivel entrega órdenes al nivel anterior para interactuar con el usuario y recibe el estado de las variables para poder supervisar el proceso.

2.4.1.4 Nivel IV

Es el nivel de planificación, este nivel desempeña un punto crítico en la producción, se ejecutan acciones como la programación de producción, el trámite de compras de repuestos y de materia prima, paradas por mantenimiento preventivo, aseguramiento

de calidad y control de inventarios y análisis de costos. Este nivel recibe todos los eventos sucedidos en el proceso (información, tablas, curvas, etc.) del nivel anterior para ser analizadas.

Este nivel tiene la función de proveer información segura al nivel siguiente, para realizar gestiones con prudencia de ingeniería de productos y procesos, ofertas de ventas en el mercado y lleva información referente al cumplimiento de cronogramas, costos.

2.4.1.5 Nivel V

Es el nivel corporativo o de gestión empresarial. Es el encargado de ejecutar las administraciones de marketing, comerciales, talento humano, de sistemas, planificación administrativa, financiera estratégica, ingeniería de proceso, etc.

Este nivel provee información al nivel anterior sobre la situación y demanda del mercado (pedidos y proyección de almacenamiento), información de ingeniería de proceso, etc. Para poder cumplir con las metas planteadas este nivel recibe información del nivel anterior sobre cumplimiento de cronogramas, costos, etc.



Figura 12: Niveles jerárquicos de la automatización

Fuente: Los Autores, año 2013

2.5 Redes industriales

Una red es un conjunto de equipos conectados entre sí, que pueden enviar y recibir datos, existen diferentes tipos y arquitecturas de redes, en las cuales cada una tiene sus ventajas y desventajas sobre la otra dependiendo de la aplicación requerida y del medio utilizado.

2.5.1 Arquitectura de redes

La topología de una red se denomina por la forma de conexión que tienen sus nodos físicamente.

Las más comunes y las más usadas topologías son las siguientes:

Topología en bus.

Topología en estrella.

Topología en árbol.

Topología en anillo.

2.5.1.1 Topología en bus

Este tipo de topología ocupa el mismo canal para comunicar dispositivos o nodos entre sí, es decir, tiene un solo canal de comunicación este canal recibe el nombre de *backbone*, bus o troncal, este bus necesita una resistencia terminadora en cada extremo de la red.

2.5.1.1.1 Ventajas de la topología en bus

- Se facilita la conexión de nuevos nodos a la red.
- Para realizar el cableado se necesita menos cable que la topología en estrella.

2.5.1.1.2 Desventajas de la topología de bus

- Se necesita resistencia terminadora.
- Si existe una ruptura en el cable se va a perder toda la comunicación.
- Si hubiera una pérdida de la señal de comunicación es complicada descubrir el origen del problema.



Figura 13: Topología en bus

Fuente: Los Autores, año 2013

2.5.1.2 Topología en estrella

La topología en estrella se caracteriza por concentrar los datos, es decir los datos se transmiten por medio de un enlace punto a punto de dispositivo o nodo a concentrador, el concentrador ejecuta todas las funciones de la red.

2.5.1.2.1 Ventajas de la topología en estrella

- Se puede desconectar dispositivos de la red sin producir problemas a la misma.
- Por la forma física que tiene la topología es mucho más fácil localizar fallas.
- La instalación resulta ser más sencilla.

2.5.1.2.2 Desventajas de la topología en estrella

- Para realizar el cableado se necesita más cable que la topología en bus.
- Un problema en el concentrador causa la pérdida de comunicación de la red.
- Se requiere concentradores o *hubs*.

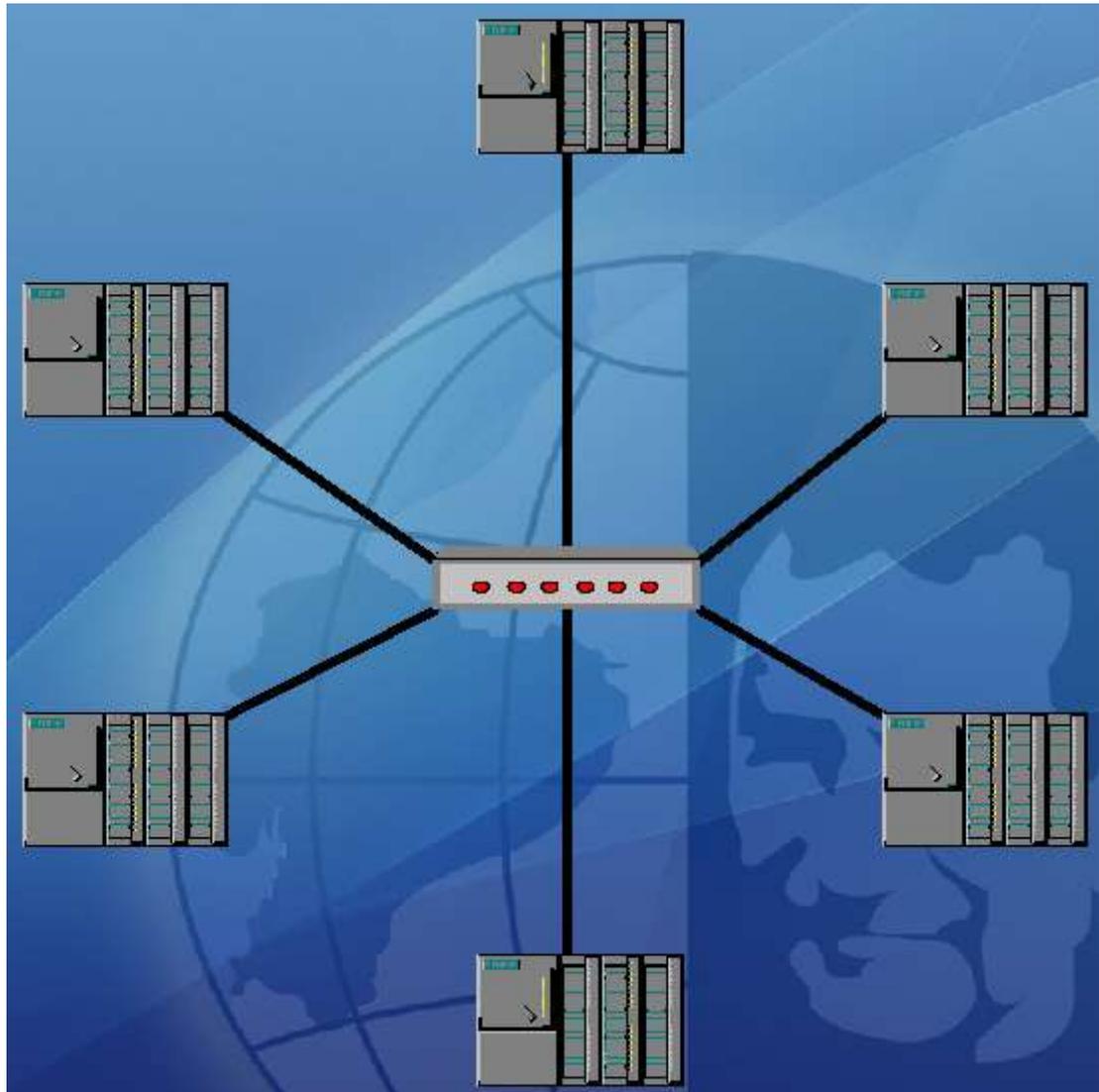


Figura 14: Topología en estrella

Fuente: Los Autores, año 2013

2.5.1.3 Topología en árbol

La topología en árbol es un conjunto de propiedades de la topología en estrella y de la topología en bus.

En este tipo de topología no interesa donde vaya dirigido los datos porque van hacer receptado por todas las estaciones, esto forma un inconveniente y para poder solucionarlo es indispensable proporcionar a la red un dispositivo o herramienta que permita reconocer al receptor de los datos.

2.5.1.3.1 Ventajas de la topología en árbol

- Segmentos individuales se lo cablea punto a punto.
- Los problemas se los puede resolver fácilmente.
- Facilidad para ampliar la red.

2.5.1.3.2 Desventajas de la topología en árbol

- Por el tipo de cable empleado la longitud de cada sección viene establecida.
- Se requiere demasiado cable.
- Su configuración tiene un mayor grado de dificultad.
- Cuando 2 o más estaciones transfieren datos al mismo tiempo puede producirse interferencia entre las señales debido que entre las estaciones tiene un medio de transmisión compartido.
- Si se pierde comunicación en la sección primordial toda la red se cae.

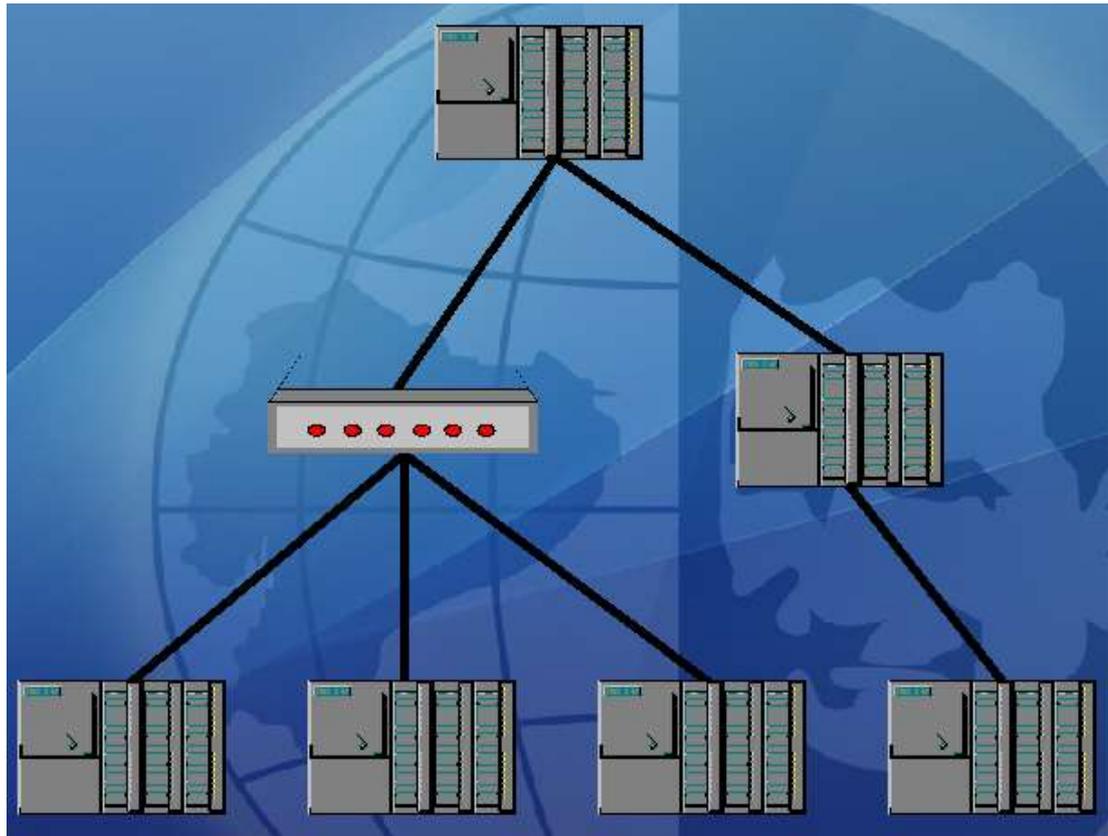


Figura 15: Topología en árbol
Fuente: Los Autores, año 2013

2.5.1.4 Topología en anillo

Tiene una topología en forma circular. Para pasar la señal a la siguiente estación cada estación tiene un receptor y un transmisor, también cada estación tiene una única conexión de entrada y otra de salida, esto hace que toda la información de la red pasa a través de cada nodo hasta que es tomado por el nodo apropiado. La configuración doble anillo permite que los datos se envíen y se recepen en ambas direcciones.

2.5.1.4.1 Ventajas de la topología en anillo

- La red provee un acceso equitativo a todos los dispositivos conectados a la red.

- La red puede ser usada por muchos usuarios y el rendimiento no va ser afectado.
- Sólida es la arquitectura.

2.5.1.4.2 Desventajas de la topología en anillo

- Hay que tener cuidado con la longitud del canal.
- Cuando la red aumenta de estaciones el canal usualmente perderá cualidades.
- Se pierde la comunicación si falla el canal o una estación, es un circuito unidireccional.

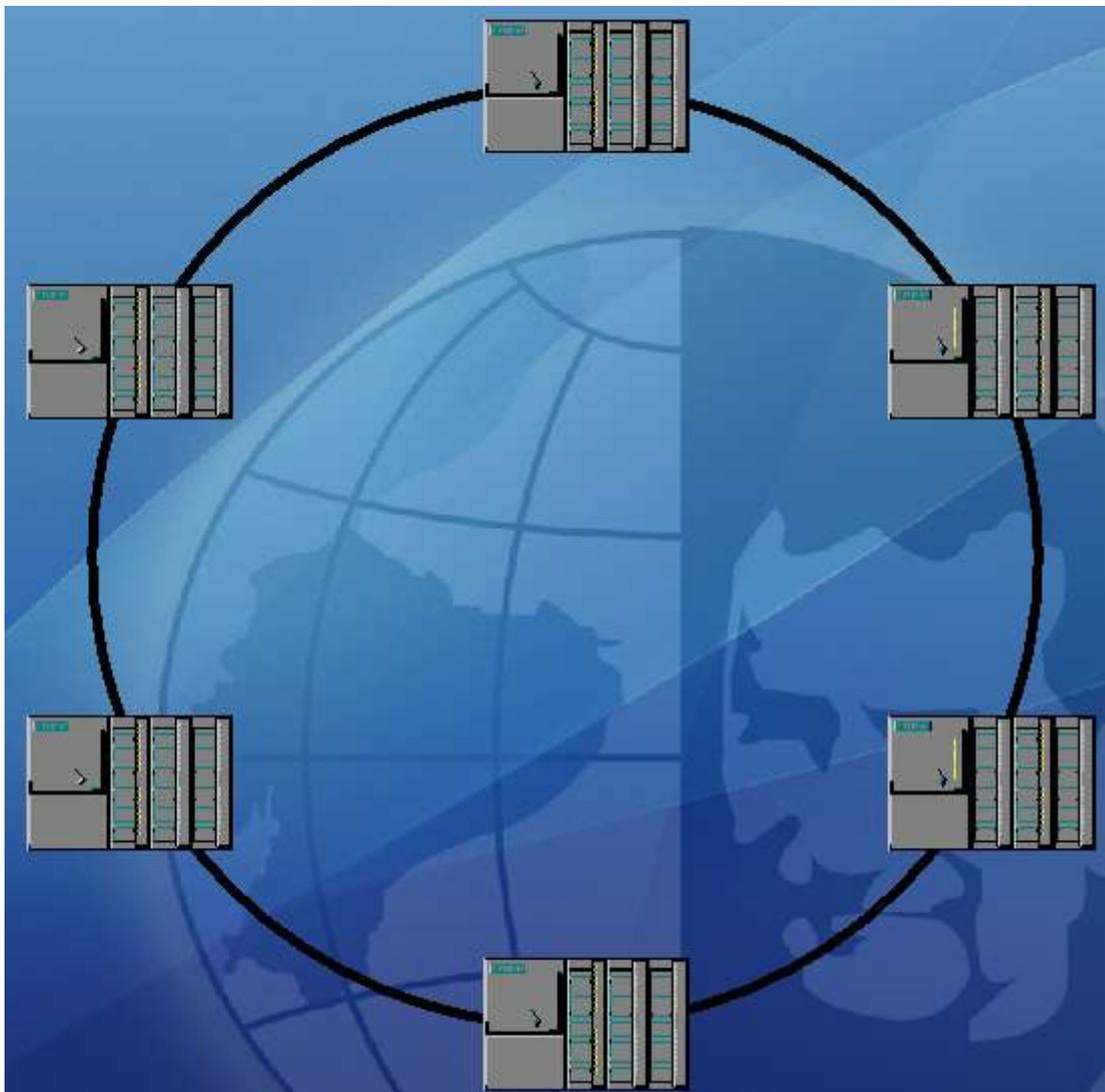


Figura 16: Topología en anillo

Fuente: Los Autores, año 2013

2.6 Tipos de redes

En redes industriales encontramos la red de factoría, red de planta, red de célula y bus de campo.

2.6.1 Red de factoría

La cantidad de información intercambiada es bastante elevada y el tiempo de respuesta de la red no es crítico, este tipo de red es usada en oficinas o despachos, gestión comercial y marketing, planificación estratégica, financiera y administrativa.

2.6.2 Red de planta

La red de planta debe manejar mensajes de diferentes tamaños, el área de cobertura debe ser extensa, y el ancho de banda tiene que ser amplio, este tipo de red es utilizado en el área de ingeniería, planificación, control de producción de planta y análisis de coste.

2.6.3 Red de célula

La red de célula debe gestionar mensajes, tener capacidad de administrar el tráfico y la probabilidad de emitir mensajes prioritarios, la red nos da la posibilidad de interconectar dispositivos de fabricación, esta red inicia en el nivel de control y termina en el nivel de supervisión.

2.6.4 Bus de campo

Un bus de campo es un medio de transferencia de información (datos) que ayuda excesivamente en el montaje y operación de equipos o maquinarias industriales empleados en el proceso de producción.

El propósito del bus de campo es reemplazar las conexiones punto a punto entre los dispositivos de campo y el elemento de control.

Específicamente son redes digitales, multipunto, bidireccionales, instaladas en un bus serie que unen equipos de campo como actuadores, transductores, sensores, módulos de E/S, variadores de velocidad, PLC, PC, etc.

Cada equipo de campo ingresa información a los sistemas de control, lo que los convierte a los equipos en dispositivos inteligentes, cada uno de estos elementos es capaz de realizar funciones de diagnósticos, control y de comunicación a través del bus.

2.6.4.1 Ventajas de un bus de campo

- Probabilidad para poder conectar equipos de diferentes fabricantes.
- Posibilidad de crecimiento de la red.
- Se reduce masivamente el uso de cable.
- Longitud de operación superior al cableado tradicional, se puede usar repetidores y amplificadores de señal.
- La puesta en servicio se facilita.

2.6.4.2 Desventajas de un bus de campo

- La inversión inicial es muy alta.
- Necesidad de personal capacitado y que tenga conocimientos superiores.
- Inversión de instrumentos, de equipos de diagnósticos y de configuración.

2.6.4.3 Clasificación de los buses de campo

Se clasifican por su campo de utilidad y por los diferentes beneficios que brindan, los buses son los siguientes:

- Buses de alta velocidad y funcionalidad baja.
- Buses de alta velocidad y funcionalidad media.
- Buses de alto rendimiento.
- Buses para áreas de seguridad intrínseca.

2.6.4.3.1 Buses de alta velocidad y funcionalidad baja

Este tipo de bus es proyectado para incorporar dispositivos sencillos como relés, actuadores, finales de carrera, etc. Nos da la posibilidad de que los dispositivos funcionen en aplicaciones de tiempos reales y reunidos en un espacio reducido, normalmente en una máquina. Especifican con frecuencia las capas físicas y de enlace del modelo OSI, se nombran algunos ejemplos.

RED CAN: Únicamente fue diseñado para ser aplicado en el área automotriz.

RED SDS: El bus es basado en CAN para añadir sensores y actuadores.

RED ASI: Bus serial diseñado para integrar actuadores y sensores, este bus es fabricado por Siemens.

2.6.4.3.2 Buses de alta velocidad y funcionalidad media

En este tipo de buses se utilizan bloques de datos de tamaño medio y se apoya en el diseño de una capa de enlace para llevar los datos. Se puede programar, configurar o calibrar el dispositivo, es decir estos bloques de datos posibilita que tenga un alto grado de funcionalidad el dispositivo. De forma eficiente y a costos reducidos estos buses son idóneos para controlar dispositivos complejos de campo. Utiliza la capa de aplicación del modelo OSI, lo que se quiere destacar es la disponibilidad del uso de

programas basados en PC`s para modificar, controlar y tener acceso a los diferentes dispositivos que forma el sistema, se nombran algunas redes.

DeviceNet: Fabricado por Allen – Bradley.

LONWorks: Fabricada por Echelon.

BitBus: Fabricado por INTEL.

DIN MessBus: Esta red funciona bajo el estándar de la comunicación RS – 232.

InterBus – S: Bus desarrollado en Alemania.

2.6.4.3.3 Buses de alto rendimiento

Los buses de alto rendimiento son aptos para aguantar comunicaciones a nivel de la factoría. Aunque son buses de alto rendimiento, ciertos buses presentan problemas debidos a excesos necesarios para conseguir las propiedades funcionales y de seguridad que se les pide. La capa de usuario recibe una gran cantidad de servicios que ofrece la capa de aplicación. Entre sus particularidades presentan:

Tienen que tener redundancia las redes multi-maestro.

Se usa el esquema pregunta-respuesta a la comunicación maestro-esclavo.

Se utiliza un límite máximo de tiempo para recuperar datos desde el esclavo.

Se emplea *unicast*, *multicast* y *broadcast* como capacidad de direccionamiento.

Se manejan eventos para pedir servicios a los esclavos.

Se puede cargar, descargar y dar inicio a un programa de forma remota.

Se nombran algunas redes:

Profibus

FIP

FieldbusFoundation

2.6.4.3.4 Buses para áreas de seguridad intrínseca

Cumplen condiciones particulares de seguridad intrínseca en áreas con ambientes explosivos que implica cambios en la capa física. La seguridad intrínseca es un tipo de protección para que el dispositivo no pueda radiar niveles de energía como para producir la ignición de materiales inflamables, es decir que no cause ignición cuando se produzca chispa o temperaturas que sobrepasen los niveles tolerables. Los buses más conocidos son:

HART

Profibus PA

FIP

2.7 Red ASI

AS-Interface (ASI) – *Aktuator-Sensor-Interface* – es un bus de campo sencillo y eficaz. Desempeña 2 funciones que son las siguientes:

Se lo utiliza como bus abierto y preparado para la unión en cualquier plataforma, proporciona la emisión de estados digitales y señales analógicas vinculadas con la maquinaria y el proceso. El bus ASI constituye una interfaz global entre actuadores sencillos y sensores binarios, y es una parte fundamental para poder realizar algún tipo de control.

2.7.1 Propiedades principales

ASI está localizada en la parte inferior de los niveles jerárquicos de la automatización industrial, teniendo un maestro en el nivel de campo que conecta los actuadores y los sensores. Los maestros pueden ser un autómatas inteligente, un PLC o una PC, y también se puede asociar con otros tipos de redes como Profibus o DeviceNet por medio de un puente o interfaz.

Las Propiedades Principales de AS-Interface son:

- Idóneo para la conexión entre maestro y esclavos, es decir se interconectan sensores y actuadores binarios.
- Se transmiten datos y energía por medio del cable ASI. Cableado barato y fácil.
- Se puede usar cualquier cable bifilar de 2 x 1.5 mm² no apantallado ni trenzado.
- El cable ideal para ASI, es un cable amarillo, plano físicamente, autocicatrizante y está marcado mecánicamente para evitar que se polarice incorrectamente.
- Se puede utilizar distintas topologías tales como bus, estrella, árbol y anillo, que ayuda al cableado para la instalación.
- Utiliza un protocolo de comunicación muy sencillo con los esclavos, es un sistema de un solo maestro.
- El ciclo del bus es rápido. Tiempo máximo de ciclo es de 5 ms con direccionamiento estándar y 10 ms con direccionamiento extendido, es decir uso de repetidores o extensores.
- Por medio de módulos activos se admite la conexión de actuadores y sensores no ASI.
- El número máximo es de 31 esclavos, es decir los dispositivos que se pueden conectar es de 124 sensores y 124 actuadores binarios con direccionamiento estándar.
- El número máximo es de 62 esclavos, es decir los dispositivos que se pueden conectar es de 248 sensores y 248 actuadores binarios con direccionamiento extendido.
- Longitud máxima de cableado es de 100 m anexando todos los tramos, o hasta 300 m con repetidores.

La revisión 2.1 de la red ASI del modelo estándar posibilita la conexión de sensores y actuadores analógicos.

Comunicación de estados por modulación de corriente que asegura un alto grado de seguridad. El maestro de la red está obligado a localizar errores en la comunicación y supervisar el adecuado funcionamiento de los esclavos.

La fuente de tensión para su correcto funcionamiento es de 30 VDC. La alimentación auxiliar para una tensión de 24 VDC para esclavos ASI utiliza un cable de color negro. La alimentación auxiliar para una tensión de 220 VAC para esclavos ASI utiliza un cable de color rojo.

Para ambientes rigurosos se encuentran dispositivos con grado de protección IP-65/67. Temperaturas de proceso entre -25°C y $+85^{\circ}\text{C}$.

2.7.2 Módulos AS-Interface

Para la red AS-Interface se encuentran 2 tipos de módulos conectables:

Módulos Activos

Módulos Pasivos

2.7.2.1 Módulos activos

Son los módulos que integran un chip ASI, por lo que adquiere una dirección en la red (se designa la dirección con un equipo direccionador para ASI o por el mismo maestro). Al tener una dirección, tendrán asignados 4 bits de entrada y 3 ó 4 bits de salida dependiendo del direccionamiento y de los módulos. Estos módulos se aplican para acoplar actuadores y sensores que no sean ASI, en este caso son sensores y actuadores binarios normales.



Figura 17: Módulo activo de Siemens

Fuente: Laboratorio de Fabricación Flexible UPS Guayaquil, año 2013

2.7.2.2 Módulos pasivos

Electrónica integrada no tienen estos módulos, entonces, sólo suministran medios para cambiar el tipo de cable, nos ayuda para hacer ramificaciones en la red con la topología tipo árbol o como un método de conexión de sensores y actuadores ASI con chip integrado. Estos módulos no adquieren dirección de red, ya que serán los dispositivos con electrónica ASI integrada los que contengan la dirección. En 2 partes se distribuye cada módulo.

2.7.2.3 Módulo de acoplamiento

Nos brinda una interfaz electromecánica con el cable Amarillo ASI. La parte superior tiene cuchillas para penetrar el cable ASI.

2.7.2.4 Módulo de usuario

Estos módulos tienen una función en particular son diseñados para el uso que está designado. Hay módulos de usuario que es revestimiento de cable para la ejecución de ramificaciones, y encontramos otros módulos que incorporan un chip ASI para la conexión de actuadores y sensores binarios. Estos módulos tienen leds de diagnóstico.

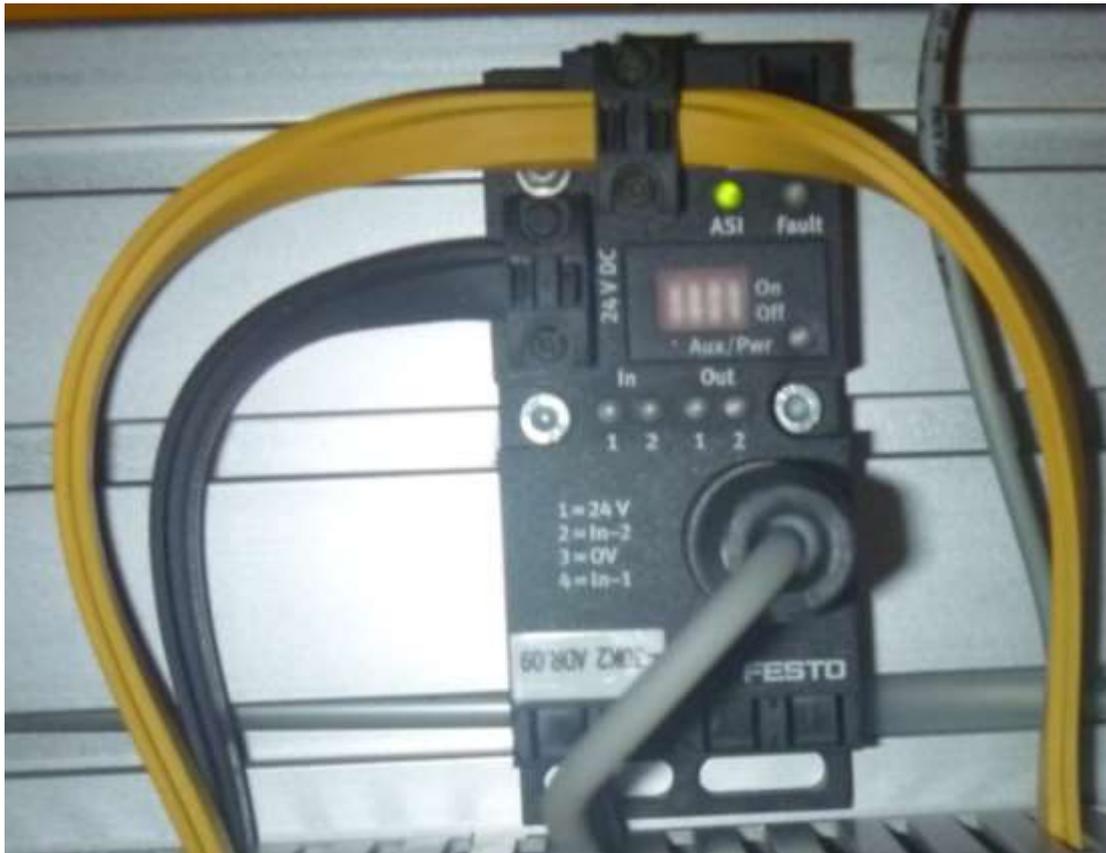


Figura 18: Módulo de acoplamiento y módulo de usuario

Fuente: Laboratorio de Fabricación Flexible UPS Guayaquil, año 2013

2.7.3 Repetidor y extensor

2.7.3.1 Repetidor

Este elemento se comporta como un amplificador de señal y necesita de una fuente de alimentación en cada extremo. Su principal función es extender la longitud del

cableado, que cubran distancias mayores a 100 m, se pueden colocar esclavos en cada lado de la red y dicha red puede alcanzar una longitud máxima de 300m.



Figura 19: Repetidor y prolongador

Fuente: Siemens AG 2009, año 2013

2.7.3.2 Extensor

Es un elemento pasivo que posee como función duplicar la longitud máxima que tiene el cableado de un sensor o actuador en un segmento de la red AS-Interface, entonces, tiene la disponibilidad de extender un tramo de red de 100 a 200 metros. Este tipo de dispositivo tiene utilidad cuando un maestro está lejos de sus esclavos.



Figura 20: Extensor de Siemens

Fuente: Siemens AG 2009, año 2013

2.7.4 Terminal de direccionamiento

Los esclavos por defecto traen almacenada la dirección '0'. Todos los esclavos en una Red ASI requiere de una dirección propia, (si hubiera 1 o más esclavos con la dirección repetida se producirán errores en la red), la función del terminal de direccionamiento es de conceder a cada esclavo una única dirección. El terminal de direccionamiento reconoce al esclavo y le concede una dirección comprendida entre la 01 y la 31.

Las propiedades que tienen los terminales de direccionamiento son los siguientes:

- Ejecutar test de funcionamiento a los esclavos.
- Realizar diagnósticos a sensores analógicos y digitales.
- Ejecutar un escaneo completo a la red y detectar toda la configuración del sistema.
- Se pueden almacenar información en su memoria.
- Efectuar la operación de puente con la PC.

2.8 Profibus

Profibus es un tipo de red de comunicación para bus de campo, donde su interfaz posibilita aumentar aplicaciones en proceso, fabricación y automatización. Este tipo de red es usado para transmisión de datos en tiempo real y a alta velocidad, también para actividades de comunicación extensa y compleja. Profibus procede de las palabras **PRO**cess **FI**eld **BUS**.



Figura 21: Cable profibus con conectores

Fuente: Laboratorio de Fabricación Flexible UPS Guayaquil, año 2013

2.8.1 Elementos del bus

Se encuentran los siguientes elementos en el bus:

Nodos Activos

Nodos Pasivos

2.8.1.1 Nodos activos

Son elementos que se ejecutan como maestros y toman el control del bus.

2.8.1.2 Nodos pasivos

Son elementos que se ejecutan como esclavos y no pueden controlar el bus. Se comunican con los nodos activos (maestro) por medio de la herramienta pregunta-respuesta, pero no puede interactuar directamente con otros esclavos.

2.8.2 Repetidor

Es un amplificador de señal bidireccional.

2.8.3 Conexiones físicas

Profibus tiene 5 formas de diferentes de comunicación que son las siguientes:

2.8.3.1 RS-485

Emplea un cable de 2 hilos trenzado y apantallado, este protocolo admite velocidades entre 9,6 kbit/s y 12 Mbit/s, y se pueden conectar 32 estaciones, o más si se usan repetidores.

2.8.3.2 MBP. Manchester Coding y Bus Powered

Se maneja en una sola velocidad fija de 31,25 kbit/s y es transmisión sincrónica, se pueden conectar como máximo 126 dispositivos de campo, dividido en 10 a 32 dispositivos por segmento.

2.8.3.3 RS-485 IS

Tiene las mismas características técnicas del protocolo RS-485, la diferencia que es empleada en áreas potencialmente explosivas, es la versión para zonas intrínsecamente seguras.

2.8.3.4 MBP IS

Tiene las mismas características técnicas del protocolo MBP, la diferencia que es empleada en áreas potencialmente explosivas, es la versión para zonas intrínsecamente seguras.

2.8.3.5 Fibra óptica

Se pueden utilizar fibra de plástico, fibra de vidrio en versión multimodo y monomodo. Los beneficios de utilizar fibra son los siguientes:

- Es inmune al ruido.
- Se pueden cubrir grandes distancias.
- Soporta configuraciones en anillo, doble anillo y estrella.

2.8.4 Comunicación

En la parte de comunicación con profibus se pueden realizar diferentes aplicaciones.

2.8.4.1 Aplicación monomaestro

Esta aplicación proporciona que los ciclos de lectura sean más cortos porque sólo se encuentra un maestro activo en el bus, los demás dispositivos son esclavos.

2.8.4.2 Aplicación multimaestro

Esta aplicación acepta más de un maestro, es decir, puede haber varios maestros trabajando independientemente o varios maestros manejando sus propios esclavos.

2.8.4.3 Características de la comunicación

La velocidad de transmisión depende de la longitud de la red y del número de estaciones

Las velocidades de transmisión que se puede configurar son:

9.6, 19.2, 93.75, 187.5, 500, 1500, 3000, 6000 y 12 000 kbit/s.

El número máximo de estaciones de estaciones que se permite configurar son 127 (32 sin utilizar repetidores).

Longitud máxima es de:

93,75 KBaudios a 1200 metros

187.5 KBaudios a 600 metros

500 KBaudios a 200 metros

Soporta conexiones de tipo bidireccionales, *multicast* que es el envío de los datos en una red a diversos destinos simultáneamenteo *broadcast* que es el envío de datos en una red a todos los dispositivos que se encuentren en dicha red.

2.8.5 Terminadores de bus

La red profibus necesita para saber que ya no hay más dispositivos de un terminador en cada extremo del bus.



Figura 22: Conector profibus con resistencia terminadora

Fuente: Los Autores, año 2013

2.8.6 Tipos de redes Profibus

Encontramos 3 tipos de redes profibus que son las siguientes:

Profibus DP

Profibus PA

Profibus FMS

2.8.6.1 Profibus DP

La periferia descentralizada es perfecta para aplicaciones de alta velocidad, su crecimiento fue mejorado para que se comuniquen los sistemas de automatización, control, los dispositivos distribuidos de entradas y salidas. Realiza intercambio de

datos cíclicos por lo que es bastante utilizado en controles con tiempo crítico y traspaso de cantidades reducidas de datos.

Profibus DP tiene 3 versiones

DP-V0 (1993), esta versión suministra las funcionalidades esenciales; incorporando transferencia cíclica de datos, soporte de interrupciones y evaluación de estaciones, módulos y canales.

DP-V1 (1997), adiciona comunicación de datos que no se repite periódicamente (acíclica), guiada a traspaso de operación, visualización y parámetros.

DP-V2 (2002), nos da la oportunidad de comunicación entre esclavos.

2.8.6.2 Profibus PA

Profibus PA (Automatización de Procesos), es dedicado a sustituir el modo de transmisión de la señal de 4 a 20 mA en sensores y actuadores. Permitir comunicación de datos y alimentación en el bus a través de 2 hilos. Proporcionar el mantenimiento, la conexión y la desconexión de instrumentos en el bus en zonas de peligro de explosión. Es una extensión de la periferia descentralizada con una técnica que es apta para operar en áreas no seguras y con peligro de explosión.

2.8.6.3 Profibus FMS (Fieldbus Message Specification)

Es un estándar de comunicación universal utilizado para resolver acciones complejas de comunicación entre PLC's (controladores lógicos programables) y DCS's (Sistema de Control Distribuido). Utiliza el protocolo de comunicación entre sistemas de automatización, además del intercambio de datos entre equipos inteligentes.

Transmisión de datos cíclicos o acíclicos en las comunicaciones maestro – esclavo. Realiza intercambios de datos como programas y bloques.

2.9 SCADA

Su nombre se debe a las iniciales *Supervisory Control And Data Acquisition* y en español quiere decir (Supervisión, Control y Adquisición de Datos).

Es un software donde se realiza una aplicación que posibilita supervisar y controlar procesos de cualquier índole utilizándolo mayormente en la industria.

Por medio de un sistema de control que se comunica en tiempo real con los dispositivos de campo se adquieren los datos necesarios para realizar el control automático y la supervisión del proceso que puede ser visualizada por diferentes usuarios (operadores, supervisores, producción, mantenimiento, calidad, gerencia, etc.), que es el ente que toma las decisiones para el buen funcionamiento del proceso.

Todas estas funciones las realiza por medio de una estación maestra, una unidad terminal maestra (MTU) o una Unidad Terminal Remota (RTU).

2.9.1 Funciones primordiales

Las funciones primordiales son las siguientes:

- Adquirir datos para reunir, procesar y almacenar la información obtenida.
- Supervisar para visualizar desde una pantalla el comportamiento de las variables de control.
- Control para cambiar el avance del proceso, ejecutándose sobre los básicos reguladores autónomos (mensajes, alarmas, menús, consignas, etc). También se puede cambiar las salidas conectadas directamente en el SCADA.
- Transmisión de datos con equipos de campo y otros ordenadores.

- Los datos se presentan en forma gráfica.
- Utilización de los datos obtenidos para la gestión de calidad, gestión de producción, gestión financiera, gestión administrativa y control estadístico.

2.9.2 Beneficios

Posibilidad de crear paneles de alarma, que requiere la presencia del operador para reconocer una emergencia, parada o situación de alarma, también se pueden registrar las incidencias en curva y generar históricos.

Posibilidad de programación numérica, que posibilita realizar cálculos aritméticos de alta resolución sobre la CPU del ordenador donde se encuentra el SCADA, y no sobre el controlador. Con esto, se pueden desarrollar aplicaciones basadas en ordenadores con captura de información, exhibiciones en pantallas, análisis de señales, envío de respuestas a discos e impresoras, etc.

2.9.3 Objetivos

Deben ser sistemas de diseños abiertos, capaces de agrandarse o ajustarse según las necesidades cambiantes que requiere el proceso.

Deben de comunicarse con facilidad y sin problemas con el operador, con los equipos de planta y con los equipos terminales (redes locales y de gestión).

Debe ser software sencillo de instalar, sin exageradas exigencias de hardware, y sencillo de utilizar, con interfaces amistosos con el operador.

Ejemplos de software SCADA y sus fabricantes:

Aimax, es el software SCADA de Desin Instruments S.A.

CUBE, es el software SCADA de Orsi España S.A.

IFIX, es el software SCADA de Intellution.

Lookout, es el software SCADA de National Instruments.

Monitor Pro, es el software SCADA de Schneider Electric.

SCADA InTouch, es el software SCADA de Logitek.

SYSMAC SCS, es el software SCADA de Omron.

ScattGraph 5000, es el software SCADA de ABB.

WinCC, es el software SCADA de Siemens.

Factorytalk, es el software SCADA de Rockwell.

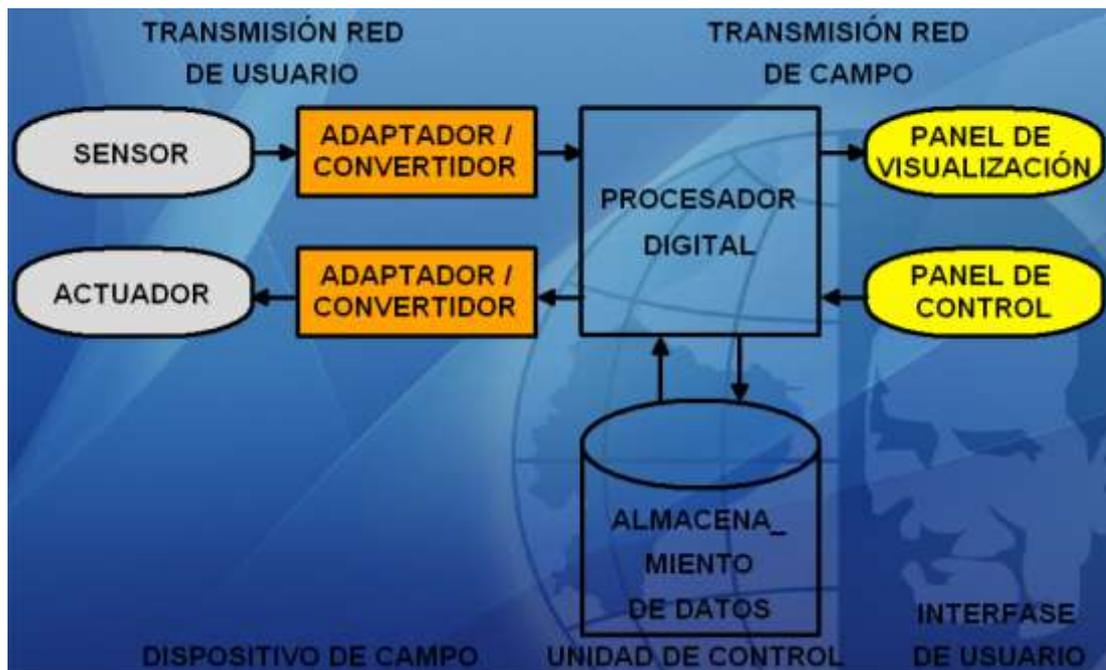


Figura 23: Esquema básico de un sistema SCADA

Fuente: Los Autores, año 2013

2.10 Controlador lógico programable (PLC)

Se puede definir al PLC (*Programmable Logic Controller*) como una computadora muy útil para la automatización industrial. Por medio de programas que son almacenados en la memoria del dispositivo electrónico se puede realizar el control de procesos electromecánicos como maquinas industriales, sistemas de fabricación flexibles, robot, etc.

El PLC está construido para recibir gran cantidad de señales de entrada y salida, ya sea digital o análoga. Las señales digitales o discretas son las que varían en dos estados, los cuales son 1 (ON) ó 0 (OFF). Las señales análogas son las que envían lecturas variables en un rango determinado de voltaje y/o corriente. El rango de aplicación del PLC es demasiado amplio, como control de procesos, visualización de estados de una instalación, control de maquinarias, etc. Los diferentes fabricantes de PLC, ofrecen software para realizar la programación del mismo. Este software tiene varios lenguajes de programación, no es difícil de aprender pero se necesita experiencia para realizar un buen trabajo.



Figura 24: Controlador lógico programable

Fuente: Laboratorio de Fabricación Flexible UPS Guayaquil, año 2013

2.11 Instrumentación industrial

Se puede definir como instrumentación industrial a los dispositivos, equipos o elementos que sirven para realizar mediciones, transmisión de datos, visualizar variables de procesos y también pueden actuar como elemento final.

Para el proceso, se pueden citar las siguientes variables físicas con sus tipos de sensores:

- Presión.
- Posicionamiento.
- Elemento final.

2.11.1 Presión

Se la conoce con el símbolo **P**, sirve para determinar la fuerza **F** que se aplica sobre un objeto o superficie **A**. "En el Sistema Internacional la presión se mide en una unidad derivada que se denomina pascal (Pa) que es equivalente a una fuerza total de un newton actuando uniformemente en un metro cuadrado. En el Sistema Inglés la presión se mide en libra por pulgada cuadrada (*pound per square inch o psi*) que es equivalente a una fuerza total de una libra actuando en una pulgada cuadrada."¹

$$P = F / A$$

En resumen presión es una magnitud física escalar que mide la fuerza en dirección perpendicular en una unidad de superficie.



Figura 25: Indicador de presión

Fuente: Laboratorio de Fabricación Flexible UPS Guayaquil, año 2013

¹ Presión, 19/ 06/2013, <http://es.wikipedia.org/wiki/Presi%C3%B3n>

2.11.2 Sensor capacitivo

Se lo puede definir como un sensor eléctrico que actúa por la variación de capacidad, espacio en un entorno definido por dos electrodos. Un electrodo puede ser el dispositivo y el otro la carcasa, pieza o estructura sobre la cual se desea realizar la medición donde un punto debe estar aterrizado.



Figura 26: Sensor capacitivo

Fuente: Laboratorio de Fabricación Flexible UPS Guayaquil, año 2013

2.11.3 Sensor inductivo

Estos dispositivos varían su magnitud cuando detectan un material metálico ferroso. Están constituidos por una bobina por la cual circula una corriente que al aproximarle una pieza de metal crea un campo magnético y genera dos corrientes opuestas, estas corrientes con ayuda de dispositivos electrónicos se transforma en señal digital. Se usa mucho en industrias con aplicaciones de posición, ausencia o presencia de material, conteo, etc.



Figura 27: Sensor inductivo

Fuente: Laboratorio de Fabricación Flexible UPS Guayaquil, año 2013

2.11.4 Silenciador

Es un dispositivo usado para disminuir el ruido generado por alguna maquina o equipo. Generalmente son de forma cilíndrica, su tamaño varia en diámetro y longitud según el lugar donde será instalado.



Figura 28: Silenciador

Fuente: Laboratorio de Fabricación Flexible UPS Guayaquil, año 2013

2.11.5 Micro interruptor

Dispositivo eléctrico - mecánico usado como elemento de posición, presencia, etc. El cual al activarse genera cambios en un circuito. Su trabajo es por medio de contactos NO (normalmente abierto) – NC (normalmente cerrado) que al accionarse cambian de estado.



Figura 29: Micro interruptor

Fuente: Laboratorio de Fabricación Flexible UPS Guayaquil, año 2013

2.11.6 Cilindro neumático

Elemento que transforma la energía de aire comprimido en trabajo mecánico. Se los puede encontrar como cilindro de simple efecto y doble efecto.



Figura 30: Cilindro neumático

Fuente: Laboratorio de Fabricación Flexible UPS Guayaquil, año 2013

2.11.7 Terminal de I/O

Usados para la conexión y visualización de las entradas y salidas del proceso. Fabricadas para 8 I/O, provistos con leds para visualizar el estado y conectores.

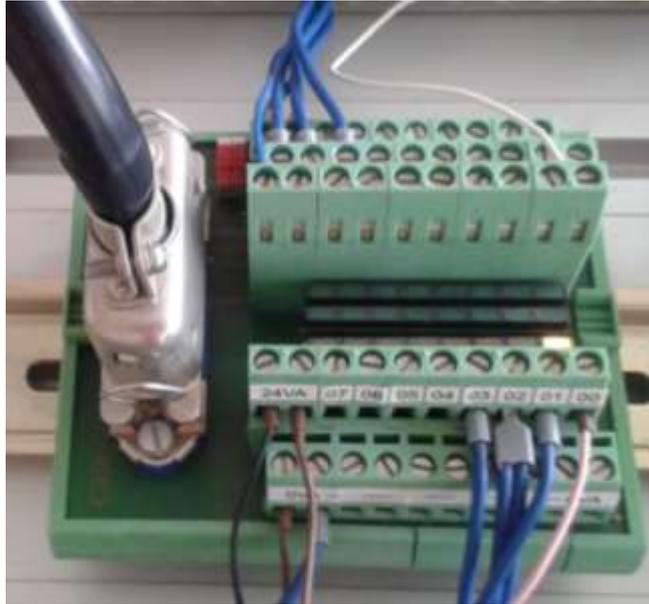


Figura 31: Terminal I/O

Fuente: Laboratorio de Fabricación Flexible UPS Guayaquil, año 2013

2.11.8 Copa de succión (ventosa)

La ventosa utiliza la presión negativa del fluido ya sea aire o agua, se adhieren a superficies planas y sin poros ya que crean un vacío momentáneo en el proceso, esta señal es transmitida por medio de un sensor de vacío.



Figura 32: Presostato y copa de succión

Fuente: Laboratorio de Fabricación Flexible UPS Guayaquil, año 2013

2.11.9 Válvula electro neumática

Este tipo de válvula funciona al aplicar corriente a un conductor en forma de bobina. Al aplicarle dicha corriente crea un campo magnético que atrae un embolo ubicado en la mitad, cuando se desenergiza vuelve a su posición normal. Son usadas en procesos neumáticos e hidráulicos.



Figura 33: Válvula electro neumática

Fuente: Laboratorio de Fabricación Flexible UPS Guayaquil, año 2013

2.11.10 Sensor opto-electrónico

Los sensores opto-electrónicos dependen o actúan al detectar la presencia de ciertos tipos de luces como infrarroja ultravioleta o visible.



Figura 34: Sensor opto-electrónico

Fuente: Laboratorio de Fabricación Flexible UPS Guayaquil, año 2013

2.11.11 Filtro con regulador para aire comprimido

2.11.11.1 Filtro

La función primordial del filtro es extraer las impurezas (partículas sólidas) y el agua condensada.

El aire de red o de algún compresor ingresa al filtro dirigiéndose hacia la parte inferior, hallando un deflector en forma de turbina que modifica la forma de la corriente del aire haciéndola rotar. Esta rotación retira por centrifugado las partículas más pesadas: gotas de agua, aceite, cascarillas de óxido, etc.

Despojado de las impurezas con mayor espesor, el aire avanza hacia la salida cruzando necesariamente por un filtro sinterizado o de fibras sintéticas, capaz de retener las partículas sólidas no precipitadas en el filtro ciclónico. En el mercado se encuentran varios rangos de filtración expresada en micro.

2.11.11.2 Regulador

La función del regulador es mantener una presión constante en la línea, independizar del suministro de aire a los diferentes equipos instalados y evitar un exceso de consumo y daños en los equipos por utilizar presiones mayores que las que requiere.



Figura 35: Filtro con regulador para aire comprimido

Fuente: Laboratorio de Fabricación Flexible UPS Guayaquil, año 2013

2.11.12 Sensor magnético

Es un interruptor de proximidad que consta de un sistema de contactos cuyo accionamiento es ocasionado por un campo magnético externo.

Los sensores magnéticos pueden detectar la posición del émbolo magnético de los actuadores y transmite una señal eléctrica o neumática a los controladores.



Figura 36: Sensor magnético sobre un pistón

Fuente: Laboratorio de Fabricación Flexible UPS Guayaquil, año 2013

2.11.13 Encoder

Encoder también llamado codificador o decodificador, su principal función es de convertir el movimiento mecánico en pulsos digitales o señales analógicas y se forma primordialmente de un disco conectado a un eje giratorio. Estos son utilizados en motores para saber la posición y la velocidad.



Figura 37: Motor con encoder

Fuente: Laboratorio de Fabricación Flexible UPS Guayaquil, año 2013

Se encuentran 2 tipos de encoders que son:

Encoder incremental

Encoder absoluto

2.11.13.1 Encoder incremental

Es un encoder que define el ángulo de posición porque suministra un número específico de pulsos, este tipo de encoder proporciona una posición estratégica desde donde siempre comenzará la cuenta.

2.11.13.2 Encoder absoluto

Un encoder absoluto produce mensajes digitales lo cual representa la posición actual del encoder, así como su dirección de movimiento y velocidad. Se los usa normalmente en motores DC.

2.12 Variador de frecuencia (VFD)

El variador de frecuencia es un dispositivo que sirve para controlar la velocidad de rotación de un motor de corriente alterna, esto lo realiza controlando la frecuencia de la alimentación proporcionada al motor. En este proceso también varía el voltaje paralelamente a la frecuencia.

Un variador de frecuencia o VFD (*Variable Frequency Drive*), normalmente está constituido por tres módulos. El primero es el circuito rectificador ya que la potencia de entrada en corriente alterna (C.A.) deber ser convertido en corriente continua (C.C.), esto se logra mediante un puente de diodos trifásico o rectificador controlado.

El segundo es un circuito de filtrado, capacitivo o condensadores de filtrado donde se mantiene el voltaje rectificado y por ultimo tenemos el circuito inversor que es donde nuevamente se realiza la conversión de energía pero a la inversa, es decir de corriente directa (DC) a corriente alterna (AC) la cual alimenta el motor.

Los convertidores de frecuencia pueden ser utilizados para que el motor funcione en ambos sentidos de giro, así mismo da la opción de frenado. Se debe tener en cuenta que la frecuencia de salida puede ser superior a la frecuencia de alimentación.

El convertidor de frecuencia se lo puede usar como arrancador o freno eléctrico para la puesta en marcha y realizar la parada por medio de configuración de rampa, permite la opción de adaptar la protección térmica del motor, sobrecarga y cortocircuito.



Figura 38: Variador de frecuencia

Fuente: Laboratorio de Fabricación Flexible UPS Guayaquil, año 2013

CAPÍTULO III

3 SOFTWARE WinCC SCADA Y ENLACE DE VARIABLES

3.1 Funcionamiento del proceso

El Sistema de Fabricación Flexible cuenta con 10 estaciones, su respectivo PLC, programa de control, componentes eléctricos, electrónicos, neumáticos y etc.

El proceso comienza con el sistema de banda transportadora que es el que nos permite transportar las piezas de un módulo a otro y poder ubicar las piezas para su debido desplazamiento entre los módulos.

Luego en la estación de distribución las piezas se comienzan a desplazar de una estación a otra para ser verificadas por el color y la altura, si la altura es apropiada avanza a la siguiente estación y si no cumple es rechazada.

En la estación de manipulación la pieza es guiada a la estación siguiente, en esta estación se posiciona y se verificada el agujero para ser devuelta al sistema de banda transportadora para ser trasladada al siguiente módulo.

En el módulo de visión se compara la pieza con una imagen grabada y observa el contraste y la silueta de la pieza, si la pieza es aceptada por el sistema de visión la estación de robot comienza a ensamblarla y si no es aceptada es rechazada por la estación de robot.

En la estación de almacenamiento se guardan las piezas cuando no está en funcionamiento la estación de clasificación y una vez en funcionamiento dicha estación las piezas se comienzan a desalojarse del almacén hacia el sistema de banda transportadora.

La siguiente estación es la de manipulación que nos ayuda a transportar las piezas hacia la estación de clasificación, si dicha estación no se encuentra funcionando y la pieza se encuentra en la banda transportadora después de la estación de almacenamiento dicha pieza es alojada en una resbaladera por la estación de manipulación, si la estación de clasificación se encuentra operativa entonces ordena las piezas dependiendo del color que presente: negra, roja y plateada (metálica).

3.2 Descripción de la partición funcional

Lo primero que se debe realizar es el levantamiento de la información, entender el proceso y el funcionamiento de todos los elementos que compone el proceso.

Luego hay que designar las variables de proceso que se requieren para su respectivo control y supervisión en el software WinCC, una vez obtenida las variables de proceso se crea un proyecto, se designa las propiedades del proyecto y los ciclos de actualización.

Una vez realizado los pasos anteriores se procede a efectuar el enlace de las variables por cada estación de control.

En el diseñador de gráficos, crear las pantallas necesaria para la supervisión y control del proceso. En las respectivas pantallas realizar las animaciones correspondientes con las respectivas variables de proceso y asignar alarmas, mensajes, curvas, administradores de usuario, etc

Se ejecuta 2 tipos de pantallas, las pantallas que trabaja con el programa DEMO que utiliza en su mayor parte un script realizado en el lenguaje VBS.

Y las otras pantallas que utiliza en su mayor parte diálogos dinámicos y conexiones directas para trabajar con el DEMO o algún otro programa que se realice en el PLC sin realizar ningún cambio en la parte mecánica del módulo.

Se realiza una comparación entre las pantallas actuales y las pantallas nuevas para poder visualizar los cambios dados.



Figura 39: Descripción funcional

Fuente: Los Autores, año 2013

3.3 Preparar la configuración para la aplicación en el software WinCC SCADA

Conectar al proceso, definir los controladores lógicos programables (PLC's) para el intercambio de datos, especificar el nivel físico para la transmisión de datos, declarar el número de datos de entrada y de salida para la comunicación con el PLC y luego definir las áreas de datos para los ciclos de envío individuales.

3.4 Crear un proyecto

Se inicia WinCC desde windows, y se pide crear un proyecto, WinCC crea independientemente una carpeta de proyecto para cada proyecto creado, entonces se

crea la base de datos MS SQL Server. Cuando la configuración se inicia de nuevo, se llama al último proyecto editado o ejecutado.

Si quiere crear un nuevo proyecto, entonces seleccione lo siguiente en el WinCC Explorer:

En la barra de menú hacer clic en el icono "Archivo", luego en el menú desplegable hacer clic en el icono "Nuevo" y aparece el cuadro de diálogo del Explorer inmediatamente seleccionar el tipo de estación y hacer clic en el icono "Aceptar".

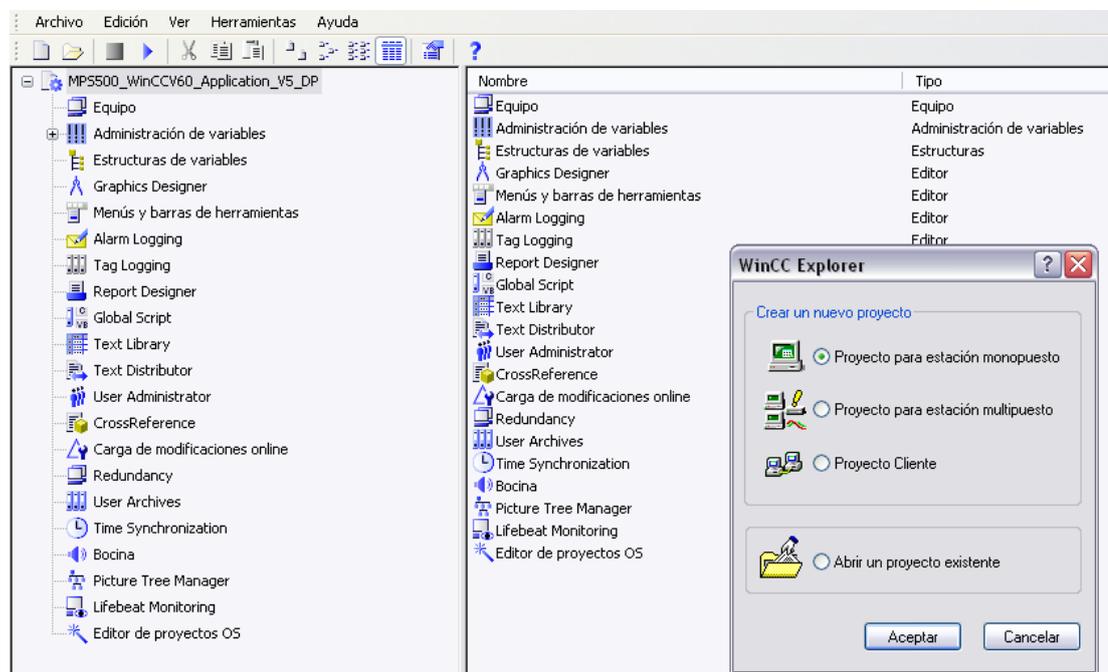


Figura 40: Crear un proyecto

Fuente: Software WinCC, , año 2013

En el cuadro de diálogo "Crear proyecto nuevo", introducir el nombre del proyecto y Hacer clic en el botón "Crear".

Se puede usar letras minúsculas y mayúsculas en un nombre de proyecto. Los caracteres no permitidos son los siguientes:

. , ; : ! ? " ' + = / \ @ * [] { } < > Espacio

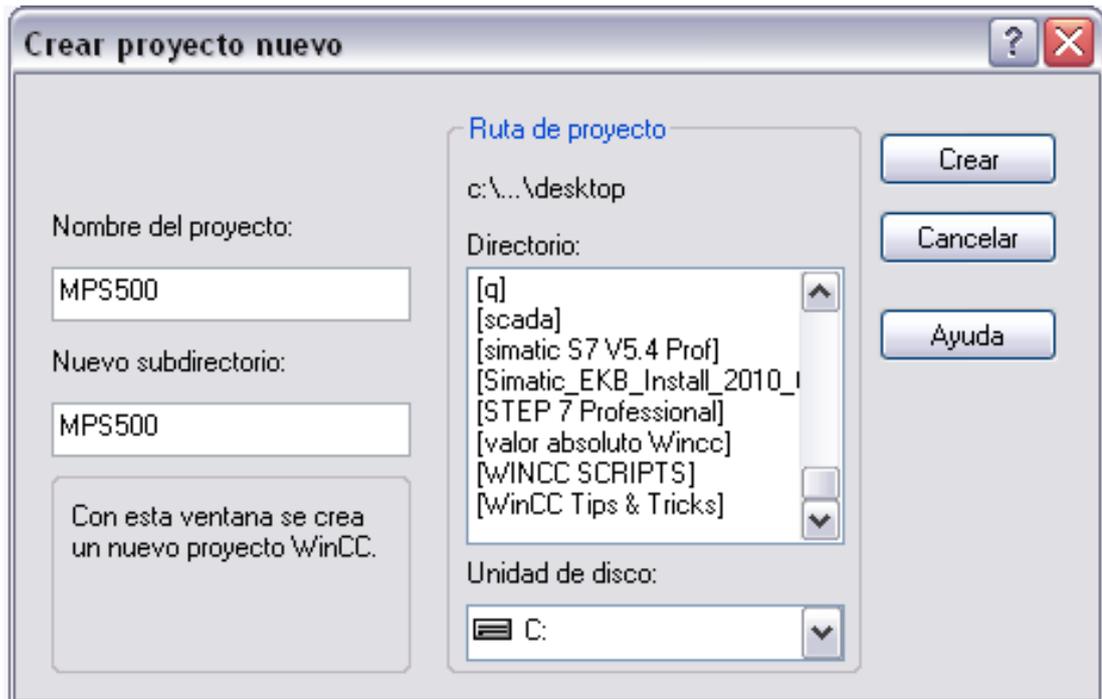


Figura 41: Crear proyecto nuevo
Fuente: Software WinCC, año 2013

3.5 Especificar las propiedades del proyecto

Por medio del menú emergente para el nombre de su proyecto introducir el creador, la versión, la última persona que modifica el proyecto y comentarios en la pestaña "General" que aparece en el cuadro de diálogo de las propiedades del proyecto.

3.5.1 Ciclos de actualización

Por medio de la pestaña "Ciclos de actualización", se puede observar los ciclos de actualización del proyecto. De estos ciclos de actualización se logra cambiar algunos ajustes (actualización: acíclica, 250 ms a 1 hr).

3.5.2 Teclas de acceso rápido

Las teclas de acceso rápido son combinaciones de teclas que emiten ciertas acciones en WinCC, como copiar pantalla por impresora, iniciar la sesión (*Logon*), y finalizar la sesión (*Logoff*).

3.5.3 Interfaz de usuario y diseño

Permite editar la paleta de colores y cambiar el tipo de diseño que desee visualizar con WinCC.

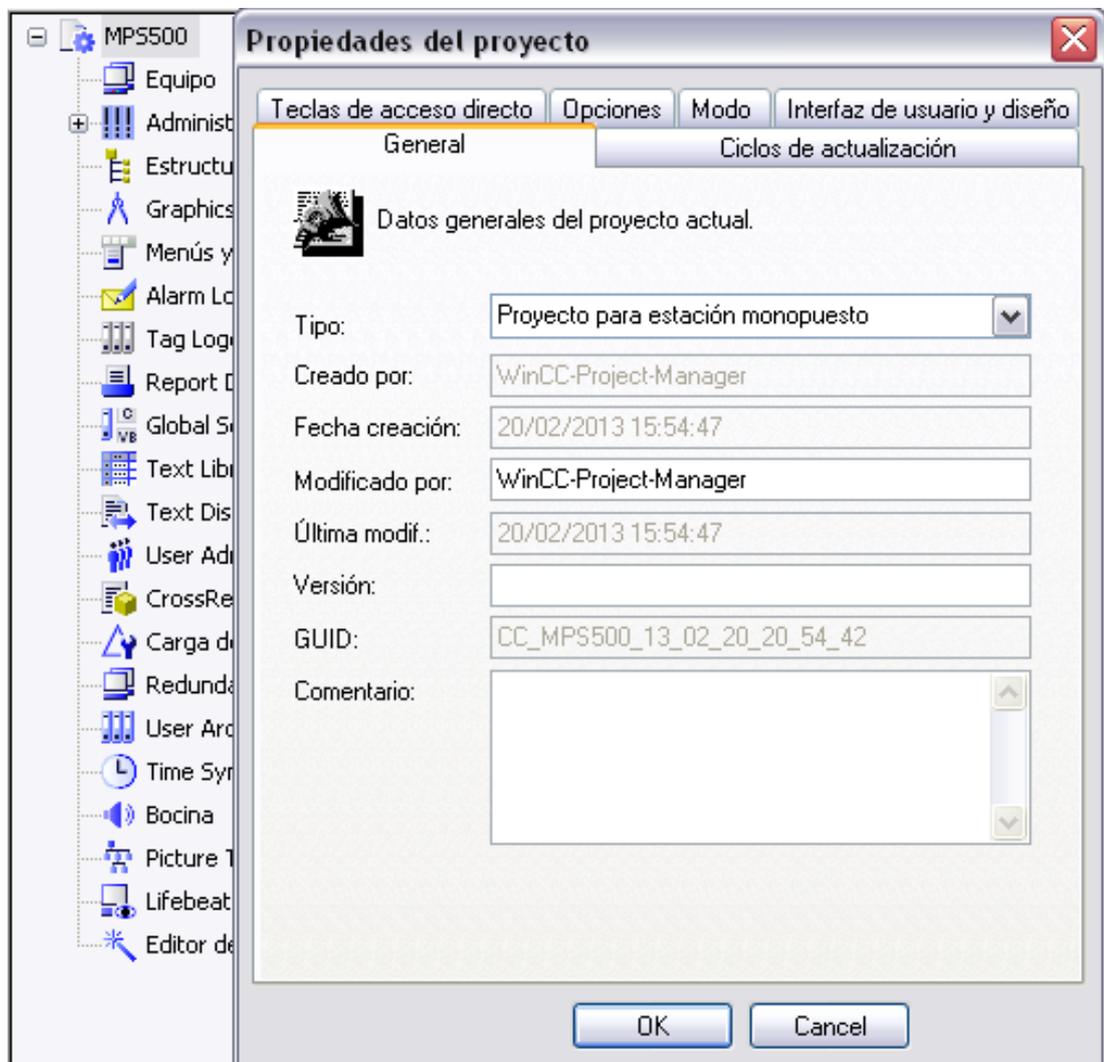


Figura 42: Propiedades del proyecto

Fuente: Software WinCC, año 2013

3.6 Propiedades de la lista de equipos

3.6.1 Llamar a la lista de equipos

La lista de equipos contiene todos los equipos asignados al proyecto actual y se selecciona siguiendo los siguientes pasos:

- Seleccionar el icono "Equipo" en la ventana de navegación, luego usar el botón derecho del ratón para llamar al menú rápido y elegir "Propiedades".
- Para cambiar el nombre del equipo, seleccionar "Propiedades" de nuevo e introducir ahora un nuevo nombre de equipo.

3.6.2 General

El nombre del equipo y el tipo de equipo se especifican en la pestaña "General".

3.6.3 Arranque

Se puede usar la pestaña "Arranque" para especificar que programa estándar de WinCC sea cargado cuando la aplicación es activada (se arranca *runtime*). Las aplicaciones de WinCC seleccionables son: *graphic runtime*, *report runtime*, *tag logging runtime*, *alarm logging runtime*, *global scripts runtime*, *text library runtime*. Puede arrancar otras aplicaciones (de Windows) durante el arranque usando el botón "Agregar".

3.6.4 Parámetros

Usar la pestaña "Parámetros" para establecer la configuración del idioma en *runtime* y bloquear combinaciones de teclas de Windows. La sección ajuste de la hora del PLC le permite visualizar la hora en *runtime* para los datos de WinCC en la zona horaria local (PLC está ajustado todo el año al horario de invierno local) o en la zona

de la hora universal coordinada (UTC) *Universal Time Coordinated* (el PLC está ajustado a la hora universal coordinada).

3.6.5 Runtime de gráficos

Permite hacer ajustes individuales del proyecto en la pestaña *runtime* de gráficos con la herramienta de ayuda online.

El nombre del equipo no puede contener un espacio en blanco, un guión o un subrayado. El primer carácter debe ser en mayúscula. Se recomienda que sólo use los caracteres de la a - z, A - Z, 0 - 9.

Los caracteres no permitidos son los siguientes:

. , ; : ! ? " ' ^ ` ~ - + = / \ | @ * # \$ % & § ° () [] { } < >

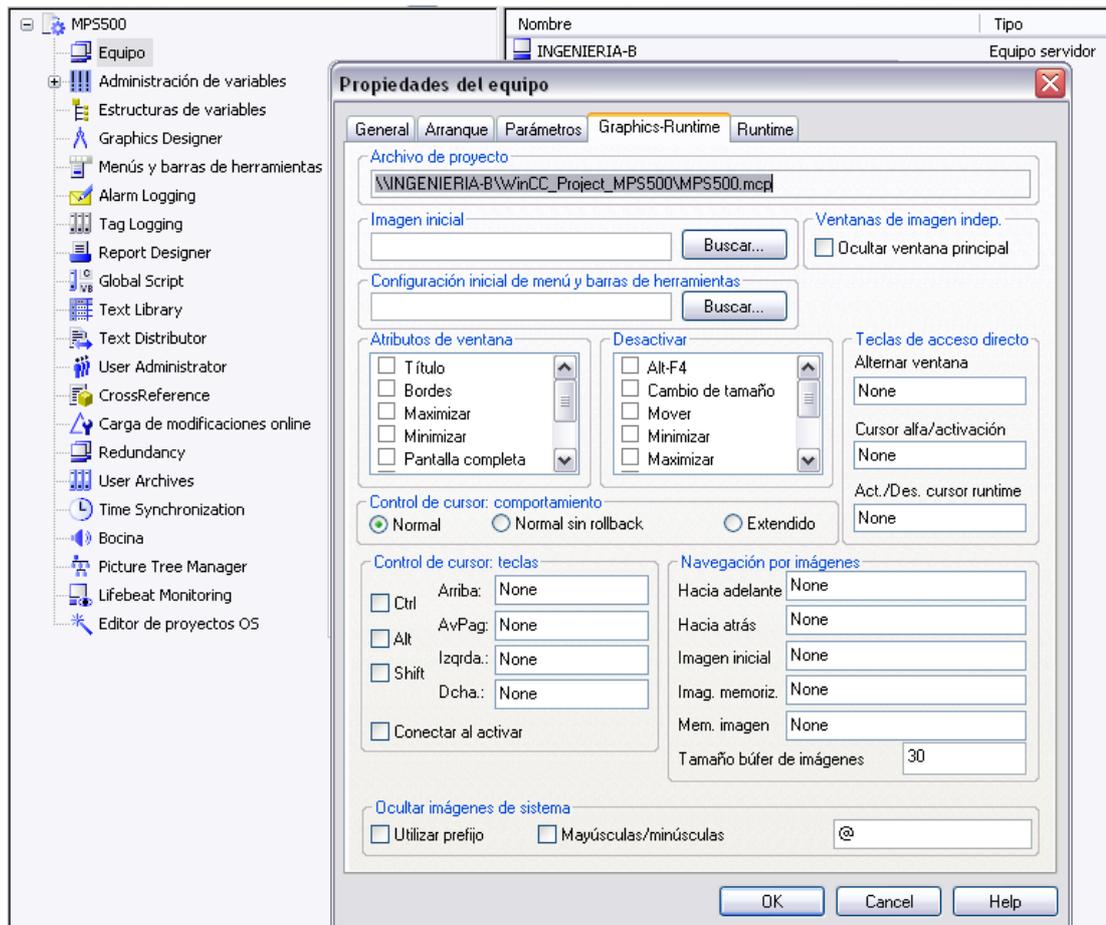


Figura 43: Propiedades del equipo

Fuente: Software WinCC, año 2013

3.7 Módulos de función en WinCC

Después de crear un proyecto, se visualiza la ventana de navegación. En esta ventana encontramos lo siguiente:

3.7.1 Equipo

Usa el nodo de la lista de equipos para definir las propiedades del equipo de una estación de trabajo.

3.7.2 Administración de variables

La administración de variables nos ayuda a gestionar los drivers para el acoplamiento, las conexiones lógicas, las variables de proceso, variables internas y además de los grupos de variables.

3.7.3 Diseñador de gráficos (*Graphics designer*)

Es una herramienta que nos ayuda a visualizar los gráficos y nos permite realizar las conexiones al proceso; sus principales propiedades son las siguientes:

- Sencillo y amigable interfase de usuario con paletas de herramientas y paletas gráficas.
- Configuración eficiente con una librería integrada de objetos y de símbolos.
- Características dinámicas y configurables de realización de los objetos para las pantallas, con ANSI C, VBS y el apoyo de un asistente dinámico. Integración de funciones adicionales mediante una potente configuración de acciones. Integración de sus propias funciones y objetos gráficos que desarrolla el usuario.
- Con Visual Basic para aplicación, puede llevar a cabo ampliaciones para el *graphics designer* y automatizar procedimientos según el Modelo de Objeto COM. VBA representa una solución abierta para acceder a cada aplicación según el Modelo de Objeto COM.

3.7.4 Registro de alarmas (*Alarm logging*)

Es un sistema que nos sirve para transferir avisos desde los procesos para su preparación, visualización, acuse y archivado.

3.7.5 Registro de variables (*Tag logging*)

Archiva y adquiere valores de medida como:

Procesamiento de valores medidos y su almacenamiento a largo plazo (puede usarse para curvas de históricos).

3.7.6 Diseñador de informes (*Report designer*)

Es un sistema integrado de informes con el cual puede registrar datos de usuario, valores de proceso actuales, valores de proceso archivados, mensajes actuales, mensajes archivados, su propia documentación del sistema entre otras cosas.

3.7.7 *Global scripts*

Expresión general para funciones en C, funciones VBS y acciones.

El global script hace el proyecto dinámico (anima el proyecto) para los requisitos específicos del usuario. Este editor hace posible crear funciones en C, en VBS y acciones que pueden usarse para todo un proyecto o incluso para más de un proyecto, dependiendo del tipo (módulo de proyecto, módulo estándar o acción).

3.7.8 Librería de texto (*Text library*)

Textos de usuario para el sistema de avisos, multilingüe, etc.

3.7.9 Administrador de usuario (*User administrator*)

Aporta para realizar protecciones con contraseña, administrar los derechos de acceso para grupos y usuarios.

3.7.10 Referencia cruzada (*Crossreference*)

La lista de referencias cruzadas sirve para indicar la localización en la aplicación de varios tipos de objetos, como por ejemplo variables, pantallas y funciones. También contiene el salto al punto de aplicación y funciones de recableado.

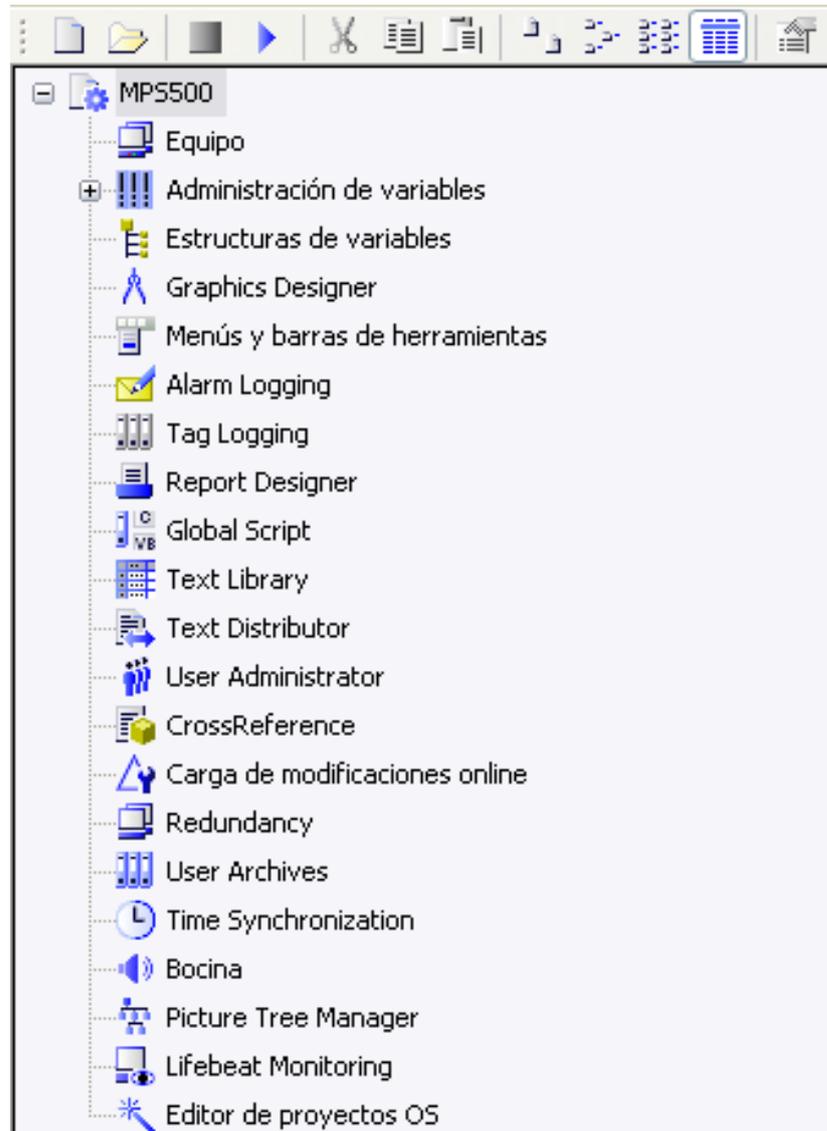


Figura 44: Módulos de función en WinCC

Fuente: Software WinCC, año 2013

3.8 Estructura de la comunicación

El administrador de datos de WinCC gestiona las variables de WinCC durante la ejecución de la *runtime*. Las distintas aplicaciones de WinCC encuentran sus requisitos de variables en el administrador de datos.

Entonces el administrador de datos tiene la misión de recoger los valores de la variable demandada del proceso usando el driver de comunicación integrado dentro de cada proyecto de WinCC. El driver de comunicación hace de interfase entre WinCC y el proceso usando una de sus unidades de canal. La conexión hardware con el proceso se hace usualmente utilizando un procesador de comunicaciones. El driver de comunicación de WinCC envía entonces mensajes de petición al PLC usando este procesador de comunicaciones. El controlador lógico programable devuelve al WinCC los valores de proceso solicitados en los correspondientes mensajes de respuesta.

3.8.1 Administrador de datos

Se encarga de la gestión completa de las variables de WinCC mientras se encuentra ejecutándose en *runtime*; el usuario no puede visualizar este funcionamiento. El administrador de datos trabaja con los datos creados en el proyecto de WinCC y almacenados en la base de datos del proyecto. Todas las aplicaciones de WinCC tienen que solicitar los datos al administrador de datos en forma de variables que WinCC pueda leer. Estas aplicaciones son el *graphics runtime*, el *alarm logging runtime* y el *tag logging runtime*.

3.8.2 Driver de comunicación

Se utilizan varios drivers de comunicación de forma que WinCC se pueda comunicar con los diferentes tipos de controladores lógicos programables disponibles. Los drivers de comunicación de WinCC enlazan el administrador de datos con el PLC.

El driver de comunicación es una DLL en C++ que comunica con el administrador de datos a través de la interfase denominada canal API. Las variables de WinCC son provistas de valores de proceso a través de este driver de comunicación.

3.8.3 Driver hardware

La conexión hardware con el proceso se hace usualmente usando un procesador de comunicaciones. Cada unidad de canal forma la interfase con un driver hardware subordinado y por lo tanto con un módulo de comunicación en el equipo.



Figura 45: Agregar un driver

Fuente: Software WinCC, año 2013

3.9 Configurar variables de proceso para SIMATIC S7

3.9.1 Variables de proceso

Establecen la conexión entre el proceso y la estación HMI. Se configuran después de que se establezca una conexión al controlador lógico programable (PLC).

3.9.2 Nuevo grupo

WinCC le permite combinar variables en grupos de trabajo. Esto resulta en una estructura de variables que es fácil de leer.

3.9.3 Nueva variable (de proceso)

Creamos el nombre de la variable, el tipo de dato y la dirección a la que va hacer asignada.

3.9.4 Variables internas

WinCC hace posible definir variables que no afectan a la conexión con el proceso en todo un proyecto. También se pueden configurar aquí las variables individualmente o combinadas en grupos, solo que éstas no van asociadas al programa que se realiza en el controlador lógico programable porque no da la opción de conectarse al proceso.

Cuando se le asigna un nombre a una variable tiene las siguientes restricciones:

No se puede usar . : ? , ' \ * \$ % Espacio en blanco

El carácter @ debería reservarse para las variables del sistema de WinCC; sin embargo, es posible usar este carácter. Es irrelevante el uso de letras mayúsculas y minúsculas. Se usa el punto como separador en estructuras de variables. Los nombres que comienzan con "\$" no son visibles en la Administración de variables.

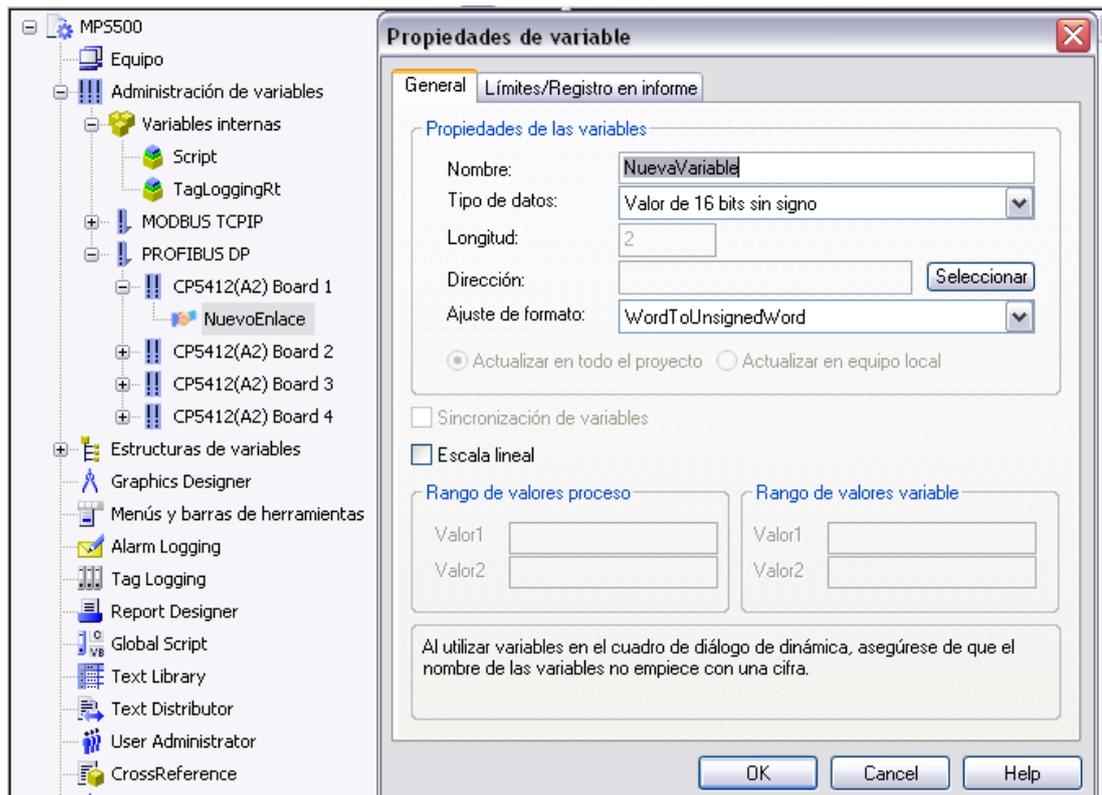


Figura 46: Propiedades de variable

Fuente: Software WinCC, año 2013

3.10 Términos de operación y monitoreo

3.10.1 Pantallas gráficas

Las pantallas gráficas son imágenes que se crean libremente y que generalmente representan al sistema o a componentes del sistema. Las pantallas hacen posible operar sistemas automatizados. Las pantallas gráficas se muestran en la ventana de gráficos, consisten en componentes estáticos, dinámicos o ambos. Los elementos de las figuras estáticas y dinámicas pueden ser operados. Combinaciones específicas de teclas pertenecen a cada pantalla gráfica desde la cual también pueden activarse operaciones.

Para crear pantallas gráficas se tienen las siguientes notas:

- Se pueden llamar y visualizar distintas pantallas gráficas para configuración al mismo tiempo.
- El portapapeles o la facilidad de arrastrar y soltar hace posible copiar componentes de la imagen desde otras pantallas gráficas.
- El tamaño de una pantalla gráfica puede adaptarse a la resolución del monitor o la mejor que requiera el cliente.

3.10.2 Elementos de las pantallas gráficas

Todos los componentes de las figuras que no cambian ni de forma ni de color corresponden a la componente estática de la pantalla. Todos los elementos de la pantalla que muestran valores de proceso o estados del proceso corresponden a la componente dinámica de la pantalla. Entre los componentes dinámicos de la pantalla se encuentran los valores digitales, atributos, textos de salida, barras gráficas, y cambios en la visualización.

La configuración de acciones permite a todos los objetos (estáticos o dinámicos) experimentar cambios de color, cambios del relleno, geometría, parpadeo, estilo, y nivel.

3.10.3 Valores digitales

Los valores digitales muestran valores del proceso en forma de números o son usados como campos de entrada o salida para cambiar valores de consigna directamente en la imagen. La visualización del valor puede cambiarse con atributos y formatos, dependiendo del valor que debe ser mostrado.

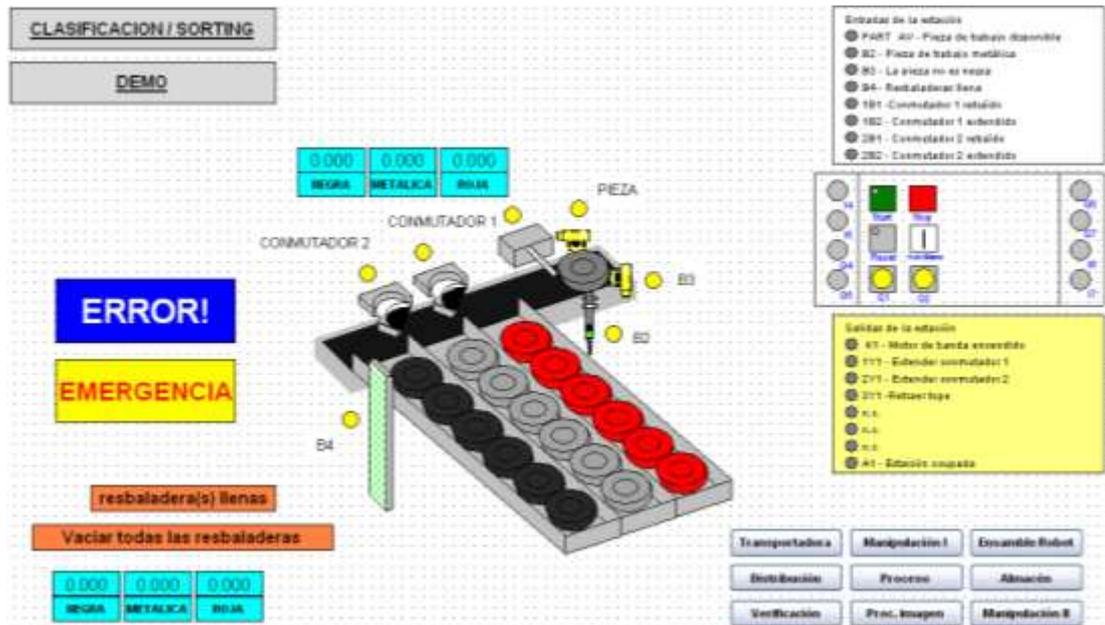


Figura 47: Ejemplo de pantallas gráficas

Fuente: Los Autores, año 2013

3.10.4 Configuración del editor de gráficos

El editor de gráficos se usa para llevar a cabo la creación de las pantallas (imágenes) necesarias (componentes estáticos y dinámicos) de acuerdo a la definición de la tarea o al proceso que se requiere representar.

3.10.5 Activar/ desactivar una paleta

En la barra de menú, hacer clic en "Ver" y en el menú desplegable, hacer clic en "Barra de herramientas".

3.10.6 Tamaño de la pantalla

Se logra determinar el tamaño de la pantalla y ajustar el tamaño dando doble clic en la pantalla, aparece la pestaña "Propiedades" en la ventana "Propiedades del objeto"

seleccione "Geometría." y encuentra la opción "Alto de imagen" o "Ancho de imagen" , en esta opción se puede introducir un nuevo valor.

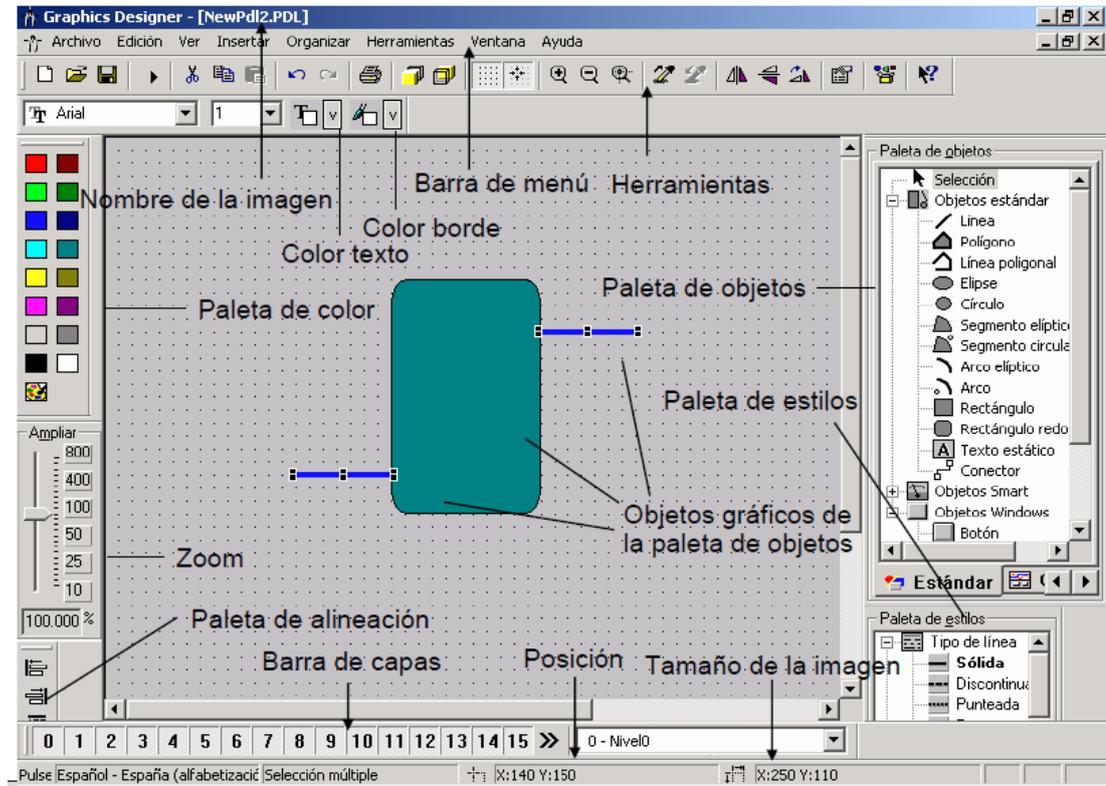


Figura 48: Editor de gráficos

Fuente: Sitrain – Curso de WinCC, año 2003

3.11 Paleta de objetos gráficos

3.11.1 Configurar objetos

Usar el botón izquierdo del ratón para seleccionar un objeto en la paleta de objetos. Hacer clic con el ratón para colocar el objeto en la pantalla gráfica.

3.11.2 Objetos estándar

Los objetos estándar están compuestos por objetos gráficos como línea, figuras geométricas, textos estáticos, etc. El conector es un objeto de tipo línea cuyos extremos pueden conectarse con los puntos de conexión de otros objetos.

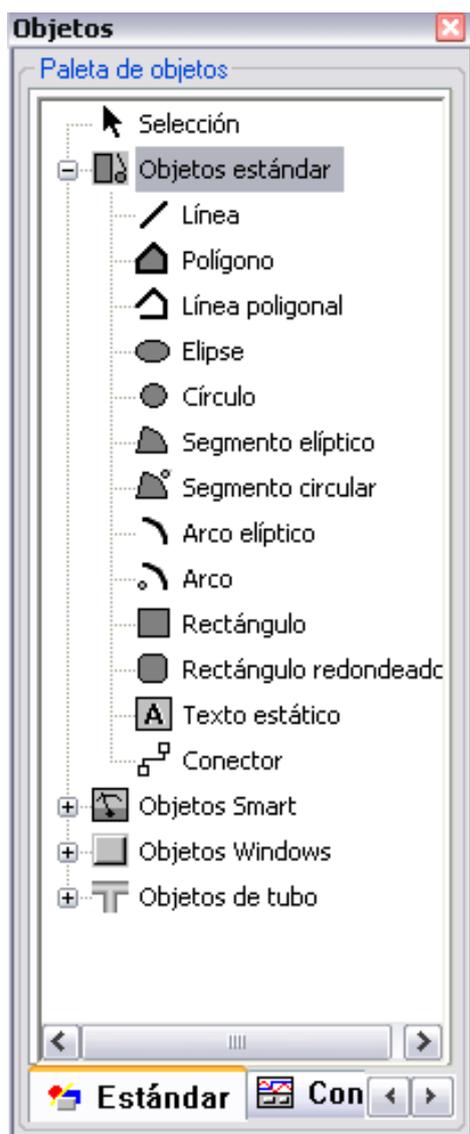


Figura 49: Objetos estándar

Fuente: Software WinCC, año 2013

3.11.3 Objetos inteligentes (Objetos *Smart*)

3.11.3.1 Ventana de aplicación

Este objeto es suministrado por el sistema de diagnóstico, el sistema de informes, o incluso aplicaciones externas. Las propiedades externas (posición, tamaño y atributos externos) se configuran en el *graphics designer* y se transfieren a la aplicación

externa en *runtime*. La aplicación externa abre la ventana de aplicación y se encarga de la operación y la visualización.

3.11.3.2 Ventana de imagen

Este objeto se configura en su tamaño, posición, visualización, escala y otras propiedades que pueden hacerse dinámicos. Una propiedad importante es la referencia a otra pantalla con el atributo "Nombre de imagen" que debe mostrarse en la ventana de imagen.

3.11.3.3 Control

Se usa para implementar elementos de Windows (como un objeto deslizante, controles de WinCC, de Windows o un reloj analógico, etc.). Un control tiene atributos que se visualizan en la ventana "Propiedades del objeto" en la pestaña "Propiedades" y que también pueden editarse en esta ventana.

3.11.3.4 Objeto OLE

Puede integrarse en una pantalla de gráficos. Se pueden utilizar programas instalados en Windows. En las propiedades del objeto se puede editar este objeto con la aplicación OLE correspondiente. Otros tipos de objetos como objetos de audio o vídeo se ponen en marcha cuando hace doble clic en ellos. Su aplicación fuente no se abre.

3.11.3.5 Campo E/S

Se puede usar como campo de entrada, salida, o como campo combinado de E/S. Son posibles los formatos de datos binario, decimal, cadena de caracteres o hexadecimal.

3.11.3.6 Barra

Muestra valores de proceso de forma analógica y en un área se muestra el valor actual. Haciendo doble clic derecho se puede ingresar a las propiedades del objeto.

3.11.3.7 Objeto gráfico

Se usan para integrar objetos gráficos externos (.EMF,.WMF,.DIB .BMP, GIF,.JPEG,.ICO) en una imagen en el *graphics designer*. Normalmente se busca los gráficos en la carpeta GraCS.

3.11.3.8 Vista de estado

La vista de estado ofrece la posibilidad de mostrar casi cualquier número de diferentes estados de un objeto. Los estados se realizan mediante variables que se corresponden con cada uno de los mismos. Los estados se muestran a través de las imágenes asignadas. En *graphics designer* se definen el tamaño y las propiedades que adopta una vista de estado en *runtime*.

3.11.3.9 Lista de texto

La lista de texto brinda la posibilidad de asignar un texto a determinados valores. La lista de texto se puede definir como lista de entrada, lista de salida o lista de entrada/salida combinada. Están disponibles los siguientes formatos de datos:
Decimal, binario o bit.

En *graphics designer* se definen el tamaño y las propiedades que adopta una lista de texto en *runtime*.

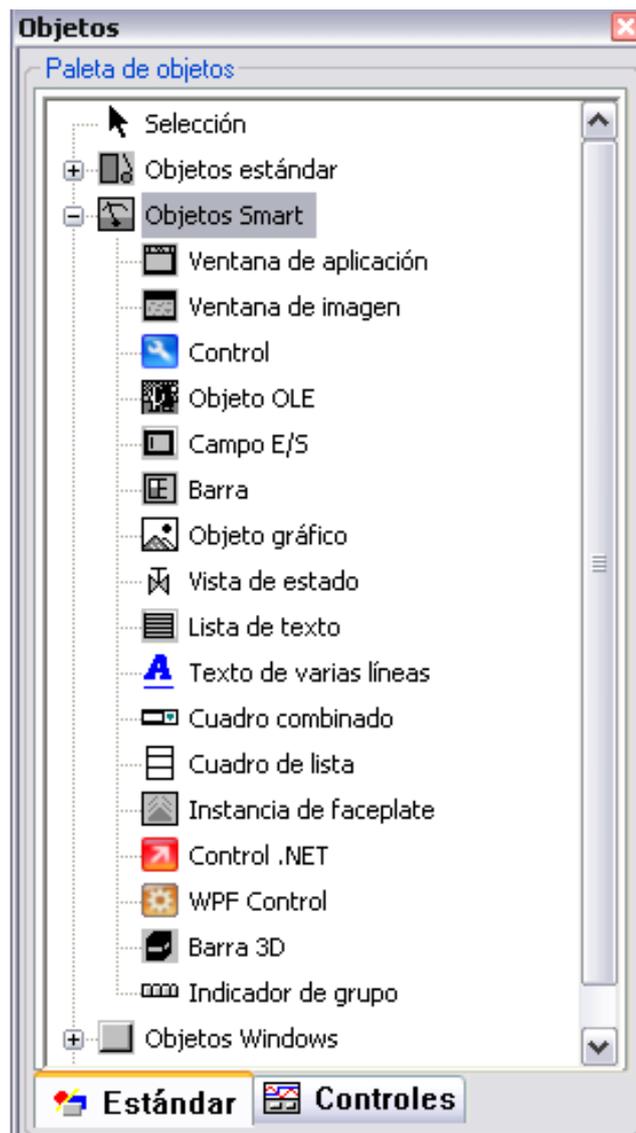


Figura 50: Objetos *Smart*

Fuente: Software WinCC, año 2013

3.11.4 Objetos de windows

3.11.4.1 Botón

Se usa para manipular eventos del proceso.

3.11.4.2 Casilla de verificación

Permite al usuario seleccionar múltiples operaciones y activar las correspondientes casillas de verificación. Casillas de verificación se pueden activar como estándar para que el usuario modifique sólo en caso necesario el valor pre ajustado.

3.11.4.3 Radio-box

Permite al usuario escoger una de las opciones ofrecidas y ejecutar el campo de opciones correspondiente. Los campos de opciones se puede activar como estándar para que el usuario modifique sólo en caso necesario el valor pre ajustado.

3.11.4.4 Botón redondo

Se usa para manipular eventos del proceso.

3.11.4.5 Objeto deslizante

Se usa como regulador lineal para el ajuste del valor de proceso analógico.



Figura 51: Objetos de windows
 Fuente: Software WinCC, año 2013

3.12 Objetos gráficos

3.12.1 Alinear objetos

Esta función puede realizarse mediante la paleta de alineación, tiene que seleccionar primero el objeto de referencia y después el objeto a alinear.

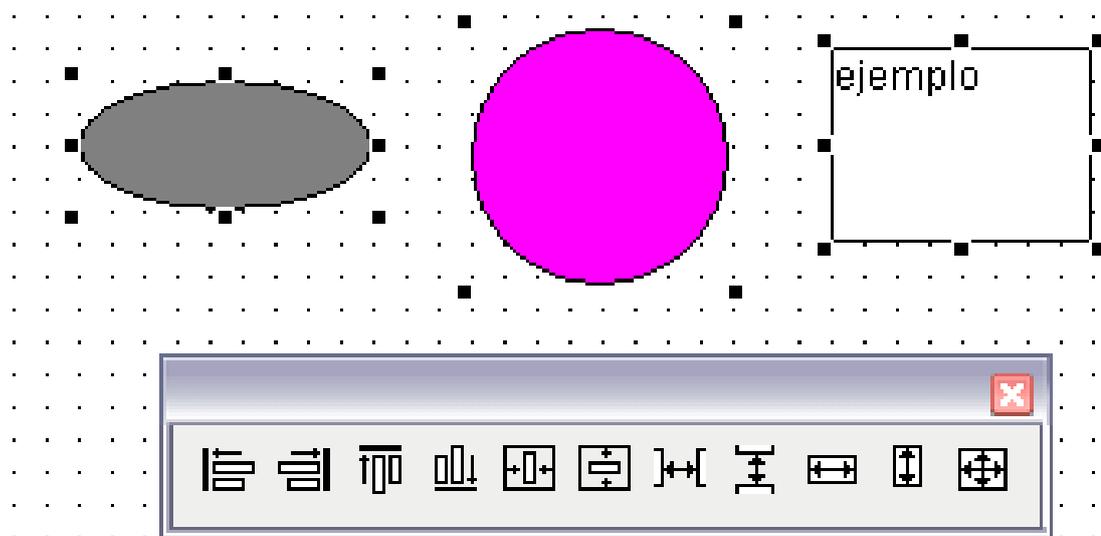


Figura 52: Barra de alineación
 Fuente: Los Autores, año 2013

3.12.2 Seleccionar objetos

1. Use el botón izquierdo del ratón para seleccionar un objeto.
2. Cuando mantiene presionado el botón izquierdo del ratón y lo arrastra a través de varios objetos, todos los objetos dentro del marco rectangular se seleccionan.
3. Cuando mantiene presionada la tecla Shift y selecciona objetos individuales con el botón izquierdo del ratón, los objetos se pueden “recoger”. Esta función se usa para borrar, alinear, duplicar, mover y editar objetos.

3.12.3 Cambiar atributos o propiedades

Para cambiar los atributos seleccionar un objeto y usar el botón izquierdo del ratón luego va aparecer unas opciones y seleccionamos la opción "Propiedades", va emerger la ventana "Propiedades del objeto" y en ese instante se puede ajusta los atributos o las propiedades.

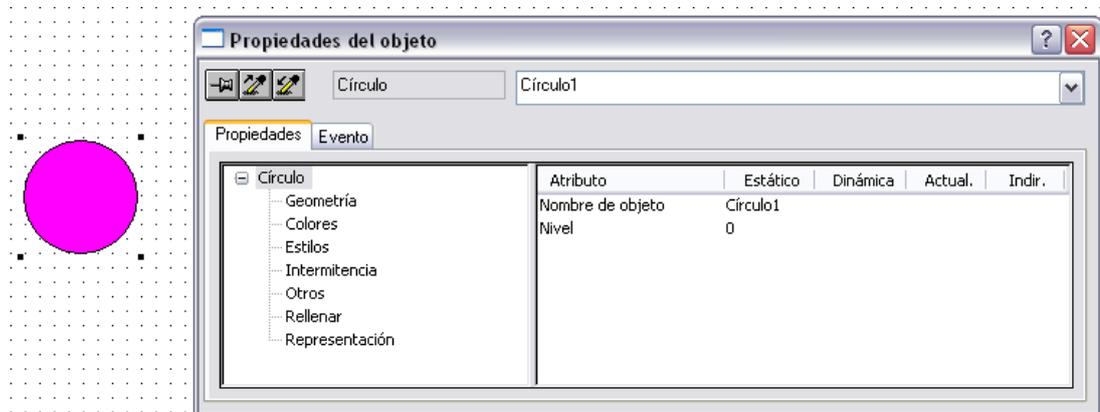


Figura 53: Propiedades del objeto

Fuente: Software WinCC, año 2013

3.12.4 Borrar objeto

Seleccionar el objeto y elimine el objeto del *graphics designer* haciendo.

1. En la barra de menú, clic en el menú "Edición".
2. En el menú desplegable, clic en "Borrar".

Es más rápido si usa la tecla Supr.

3.12.5 Cambiar la visualización del objeto

Dependiendo del objeto a seleccionar, se puede cambiar el tamaño, alto, ancho, y ángulo. Una forma rápida es seleccionar un objeto, mantener presionado el botón izquierdo del ratón, y desplazar los nodos de configuración en la dirección que desee.

3.12.6 Mover objeto

El movimiento tiene lugar con la facilidad de arrastrar y soltar. Seleccionar un objeto y mover el objeto mientras mantiene presionado el botón izquierdo del ratón.

3.12.7 Rotación

Rotar objetos en círculo:

1. En la barra de menú, haga clic en "Organizar"
2. En el menú desplegable, haga clic en "Rotar". O use el botón de la barra de herramientas.

3.12.8 Simetría

Realiza simetría de objetos en el eje horizontal o vertical.

1. En la barra de menú, haga clic en "Organizar".

2. En el menú desplegable, haga clic en "Voltear". O use el botón de la barra de herramientas.



Figura 54: Barra de herramientas estándar

Fuente: Software WinCC, año 2013

3.13 Librería de símbolos

La librería de símbolos se la encuentra haciendo clic en el menú "Ver" y escoger la opción "librería ", o en la barra de herramientas se lo representa con el símbolo.

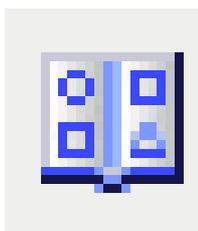


Figura 55: Símbolo para mostrar librería

Fuente: Software WinCC, año 2013

Luego va encontrar 2 librerías: la librería global que se encuentra todas las graficas como elementos de campo, operación, pantallas, Siemens HMI Symbol Library 1.4.1 y Símbolos.

La librería de proyecto, es donde se guarda las gráficas realizadas por el usuario.

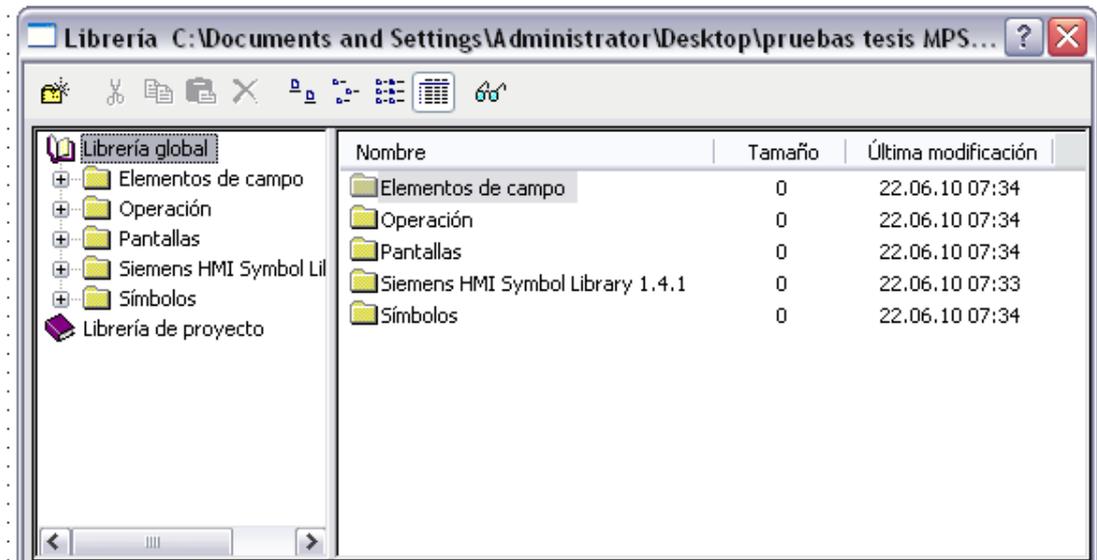


Figura 56: Librerías

Fuente: Software WinCC, año 2013

Se puede cambiar el aspecto del símbolo:

Original - 0

El color de la superficie es dada por el fabricante.



Figura 57: Imagen con atributo en original

Fuente: Software WinCC, año 2013

Sombreado - 1

El color de la superficie se relaciona a los ajustes del atributo color de fondo y color primer plano.



Figura 58: Imagen con atributo en sombreado

Fuente: Software WinCC, año 2013

Sólido - 2

El color de la superficie se relaciona a los ajustes del atributo color primer plano.



Figura 59: Imagen con atributo en sólido

Fuente: Software WinCC, año 2013

Contorno - 3

El color de la superficie se relaciona a los ajustes de atributo Estilo Fondo donde encuentra Transparente - 0 y Opaco - 1 donde prevalece es el color de fondo.



Figura 60: Imagen con atributo en contorno

Fuente: Software WinCC, año 2013

3.14 Paleta de niveles

Sirve para facilitar la edición de objetos individuales en el *graphics designer* esto permite hacer el trabajo en niveles. De esta forma se pueden dividir los contenidos de una imagen de proceso en hasta 32 niveles. Estos niveles se pueden mostrar y ocultar individualmente. En el ajuste por defecto todos los niveles están visibles e inclusive se pueden renombrar los niveles.

3.14.1 Renombrar niveles

Para renombrar los niveles tiene que seguir los siguientes pasos:

1. En la barra de menú, hacer clic en el menú Herramientas.
2. En el menú desplegable, hacer clic en Preferencias.
3. En la pestaña "Niveles visibles" realizar los cambios.

Hacer doble clic en un nivel y podrá introducir un nuevo nombre.

3.14.2 Cambiar la asignación de niveles

La asignación de un objeto a un nivel se puede cambiar en la ventana "Propiedades del objeto".

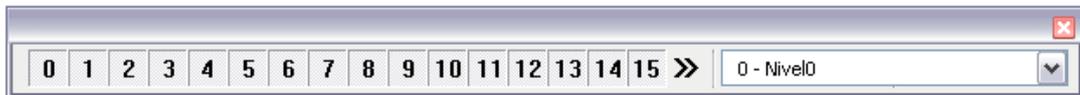


Figura 61: Barra de herramientas niveles

Fuente: Software WinCC, año 2013

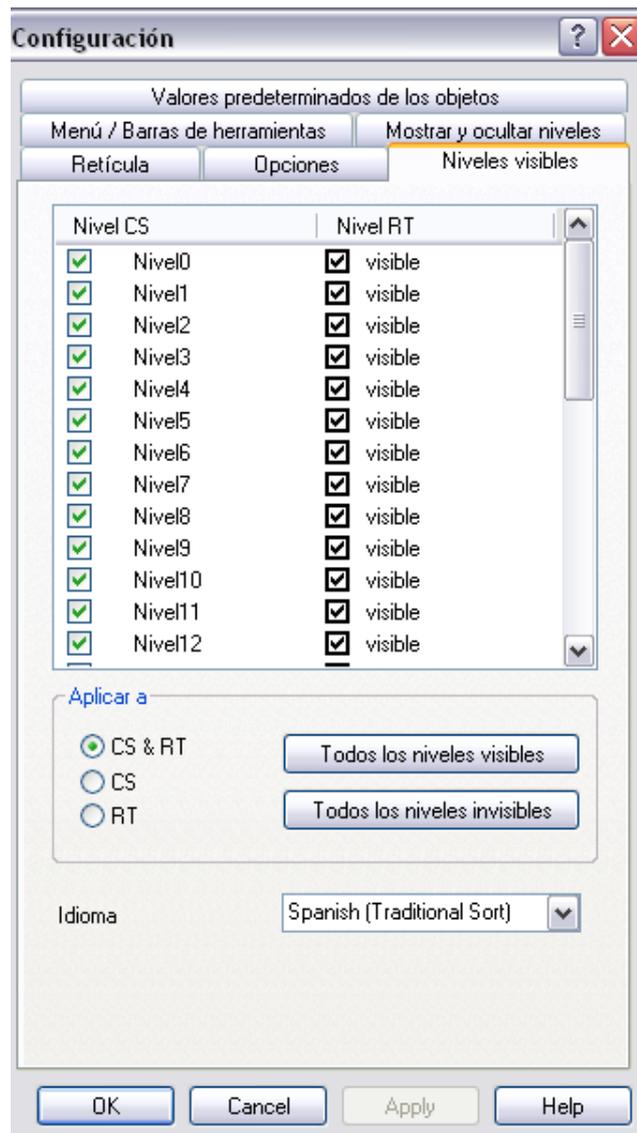


Figura 62: Configuración de niveles

Fuente: Software WinCC, año 2013

3.15 Paleta de estilos

Nos ayuda a seleccionar el tipo de línea, el grosor de la línea, la finalización de la línea, y el tipo de relleno.



Figura 63: Paleta de estilos

Fuente: Software WinCC, año 2013

3.16 Barra de herramientas de variables

Podemos visualizar las variables que han sido creada para el proceso.

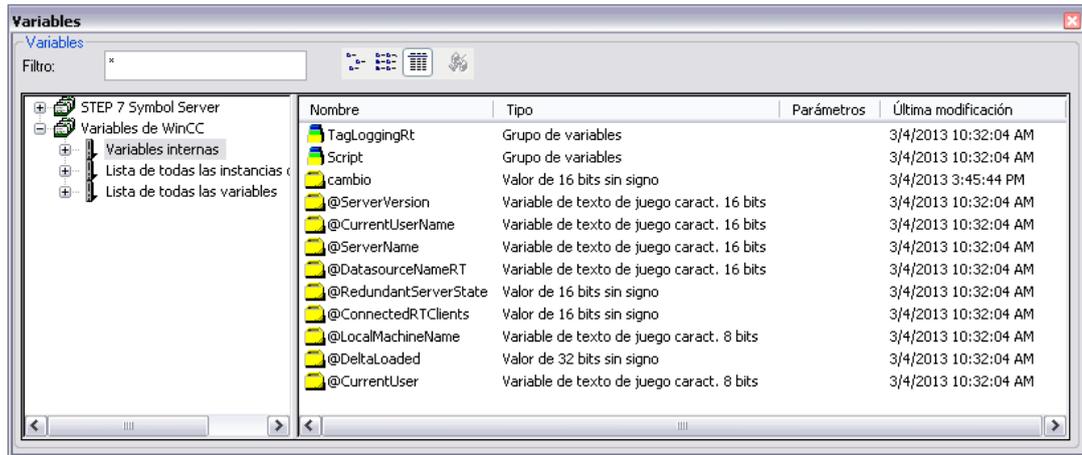


Figura 64: Barra de herramientas de variables

Fuente: Software WinCC, año 2013

3.17 Definir una imagen inicial

Para activar el modo *runtime*, se debe definir una imagen inicial para cada proyecto.

La imagen inicial es la base para todas las pantallas desde las cuales se pueden configurar estructuras ramificadas para la operación. Cada proyecto debe proporcionar una posibilidad operativa de terminar el modo *runtime*, de forma que puedan llevarse a cabo cambios en la configuración, especialmente si están deshabilitadas las operaciones de Windows.

Activar la *Runtime*

1. Llamar al WinCC Explorer y en la barra de menú, hacer clic en el menú "Archivo".
2. En el menú desplegable, hacer clic en "Activar".

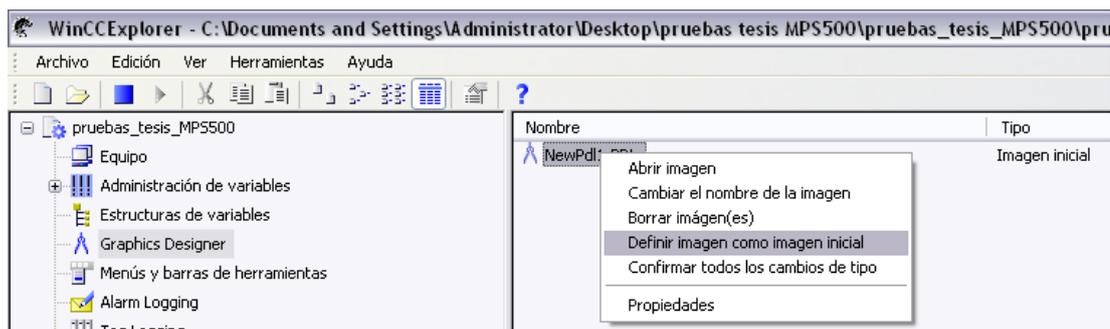


Figura 65: Definir imagen inicial opción 1

Fuente: Software WinCC, año 2013

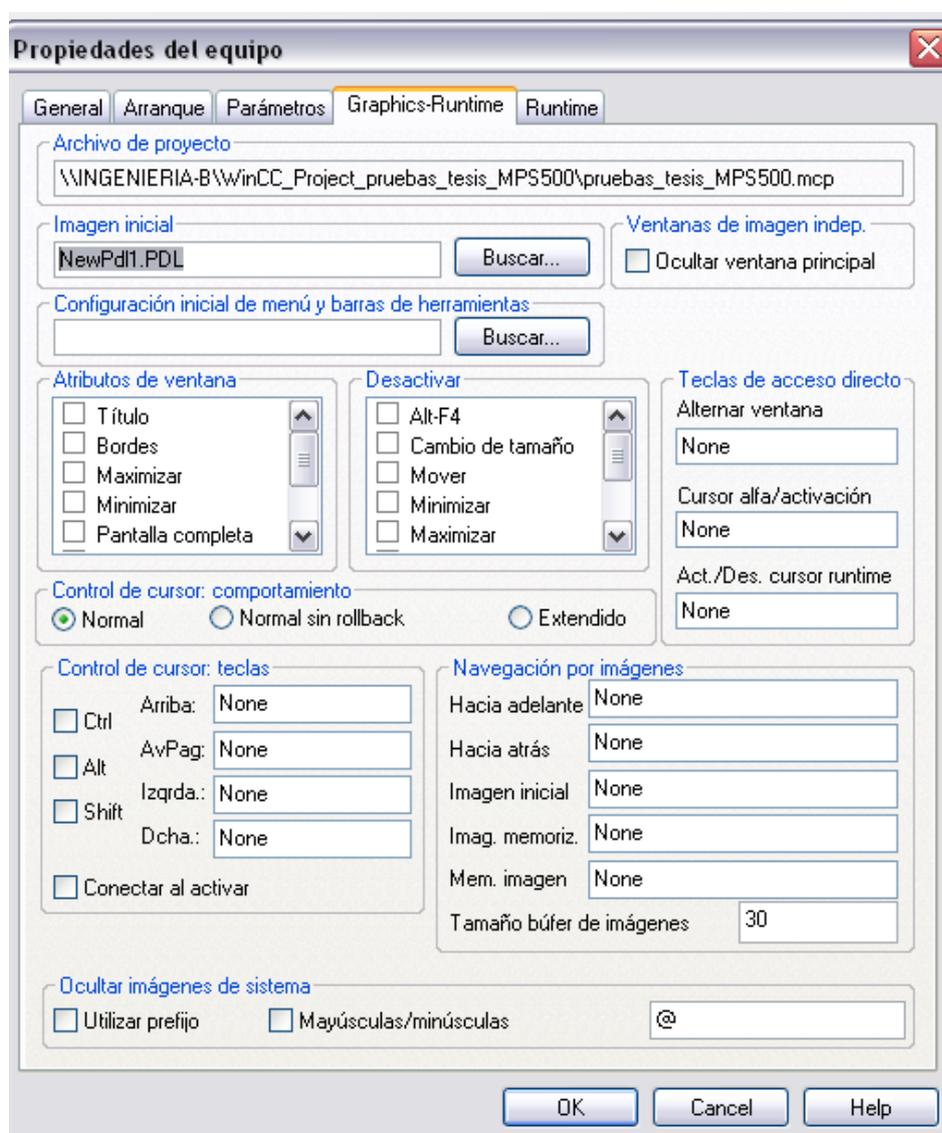


Figura 66: Definir imagen inicial opción 2

Fuente: Software WinCC, año 2013

Esta es una de las formas de salir de *runtime*, usando una acción en C.

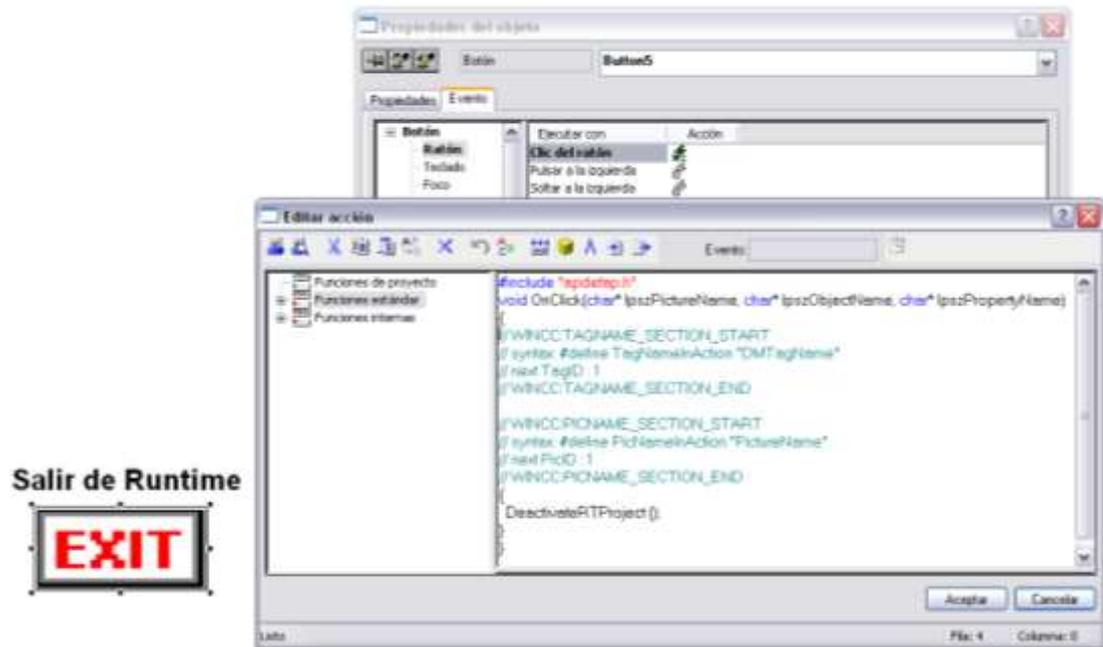


Figura 67: Salir de *runtime*

Fuente: Software WinCC, año 2013

3.18 Disparador (Trigger)

Para conseguir una selección de imagen y comunicación de altas prestaciones hay que realizar el ajuste de un *trigger* para los objetos de una imagen. Lo importante es conseguir una uniformidad para el *trigger* en cada imagen.

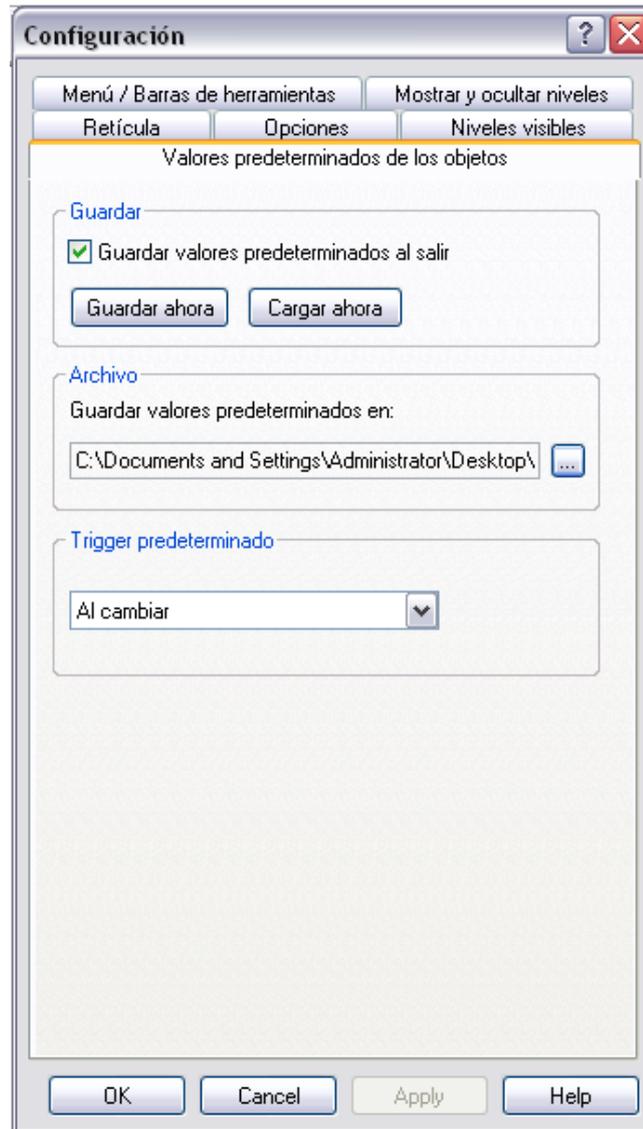


Figura 68: Configurar el *trigger* predeterminado

Fuente: Software WinCC, año 2013

3.19 Propiedades del objeto

Se puede hacer dinámico los atributos y se encuentra las siguientes opciones:

3.19.1 Cuadro de diálogo dinámico

Nos ayuda a dinamizar de forma rápida realizando cambios en el tributo por medio de confirmaciones o habilitando rango de valores.

3.19.2 Acciones en propiedades del objeto

Se puede utilizar acciones en sintaxis C y VBS en los objetos. Muchas propiedades tales como el color, tamaño del objeto, posición, tipo de letra, visibilidad, autorización, tamaño de la letra, etc. Pueden crearse dinámicamente mediante variables de proceso o variables internas. El disparo (*trigger*) se ejecuta por medio un cambio en la variable o a través de un temporizador.

3.19.3 Conexión a variable

Si los *tags* o variables deben visualizarse individualmente, está disponible la conexión directa a la variable para tags internos o de proceso. Estas variables de proceso también pueden direccionarse indirectamente.

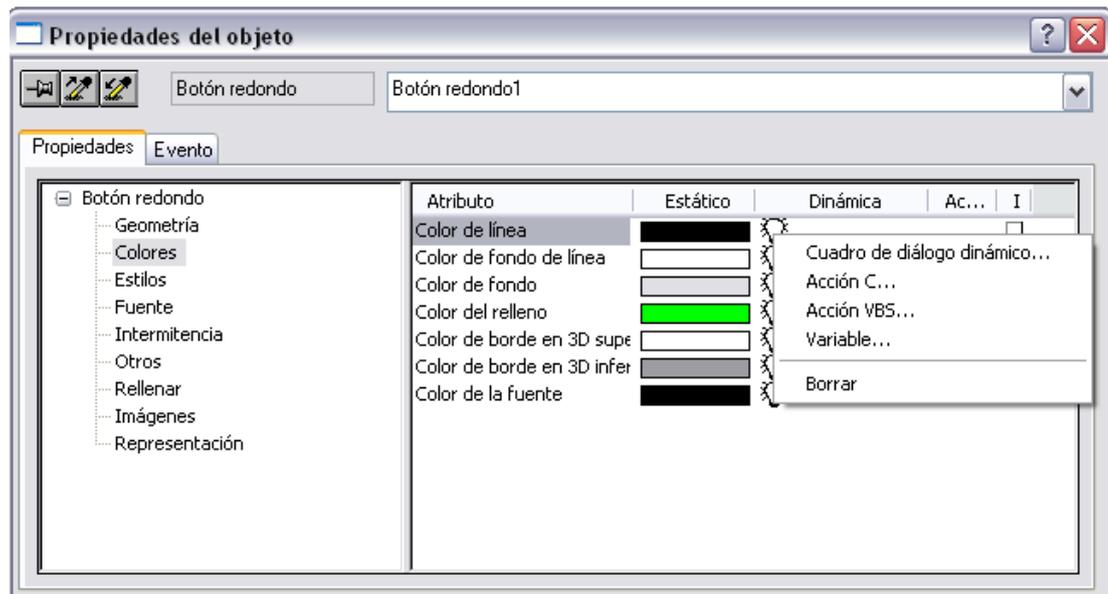


Figura 69: Dinamizar atributos

Fuente: Software WinCC, año 2013

3.19.4 Acciones en eventos del objeto

Se puede utilizar acciones en sintaxis C y VBS para acciones frecuentes que tienen que ser modificadas cuando se origine un evento. Los resultados pueden visualizarse

en objetos e imágenes de WinCC. El disparo (*trigger*) tiene lugar cuando se cambian propiedades del objeto, tales como cambio del color, posición, tamaño del objeto, etc. El disparo se da lugar en eventos en el objeto, tales como un clic de ratón, etc.

3.19.5 Conexión directa (*Direct connection*)

La Conexión directa es un tipo especial de acción. Usando Conexión directa se puede cargar muy rápidamente en variables de magnitudes de proceso fijas. Además, aquí pueden conectarse propiedades individuales del objeto directamente a variables como, un *tag* de tipo cadena de caracteres (*string*) a la propiedad texto de un texto estático. Usando conexión directa, conseguirá un alto nivel de ejecución en *runtime*.

Si se configuró una conexión directa, el símbolo del rayo se muestra en color azul.

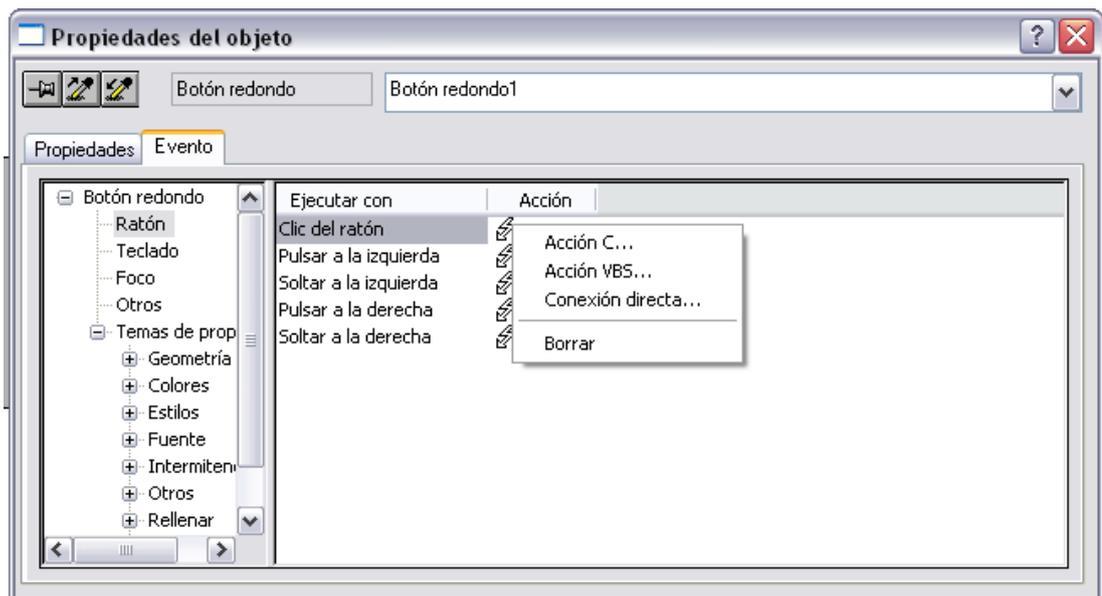


Figura 70: Dinamizar ejecuciones

Fuente: Software WinCC, año 2013

3.20 Administrador de usuario (*User administrator*)

Se configura el administrador de usuario para la asignación y administración de derechos de acceso de forma que no haya acceso no autorizado. Esto se realiza en cada operación con el proceso, operación con el sistema WinCC, operación con archivos, etc. Todas estas operaciones pueden ser bloqueadas contra acceso no autorizado.

El administrador de usuario nos permite crear grupos, usuarios, agregar funciones y autorizar las funciones a los diferentes usuarios o grupos para poder realizar cambios.

3.20.1 Configuración

Para configurar las combinaciones de teclas primero damos clic derecho en el nombre del proyecto y seleccionar Propiedades. En las propiedades del proyecto seleccionar la pestaña de teclas de acceso directo y escoger las acciones que se desea aplicar con la combinación de tecla que se desee.

Para concretar la combinación de teclas “*logon*” se usa para (registrarse como usuario) y “*logoff*” (abandonar la sesión del usuario). Para asignar el nombre para el usuario (*login*) y la contraseña. se recurre al editor del administrador de usuario.

No se puede usar Ctrl+P.

Desactivar la *runtime*, antes de comenzar con la configuración.

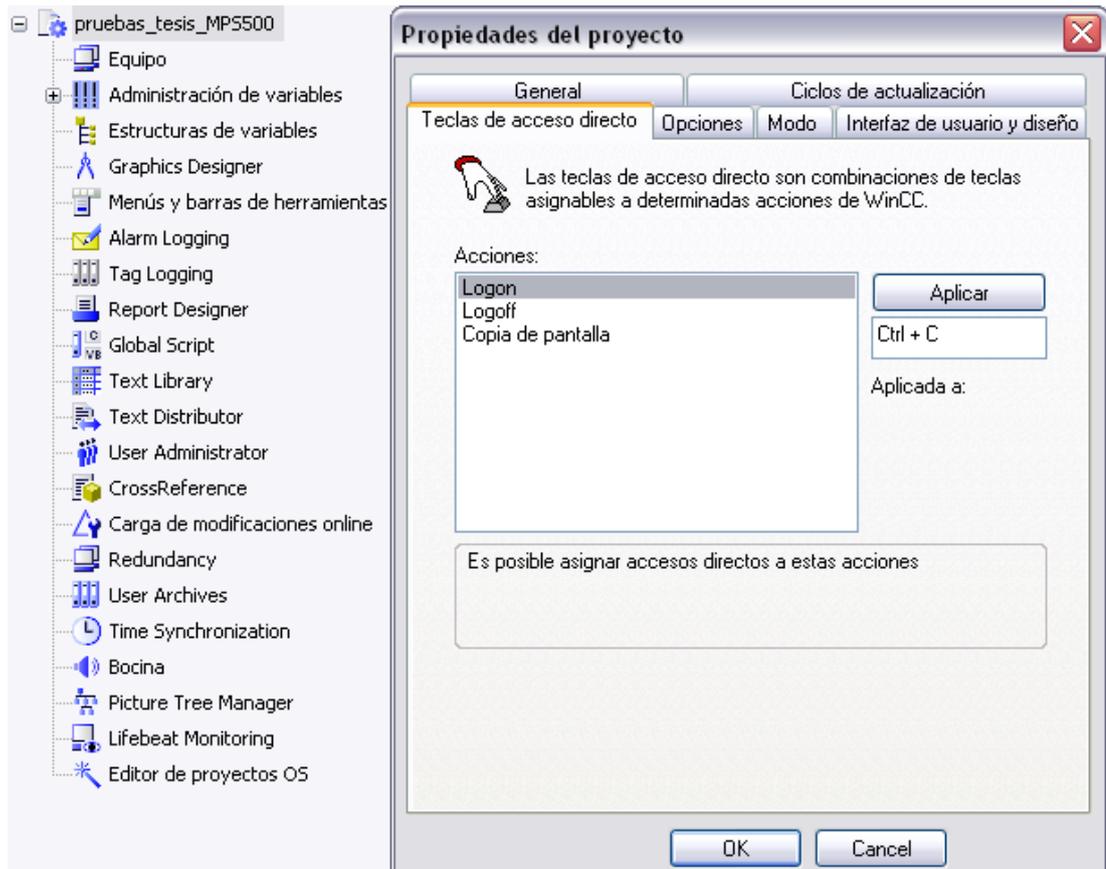


Figura 71: Teclas de acceso directo

Fuente: Software WinCC, año 2013

3.20.2 Abrir el administrador de usuario

1. Hacer clic derecho del ratón en el editor "*User Administrator*", escoger el ítem "Abrir" en el menú rápido y se arranca el "*User Administrator*".
2. Se puede asignar las autorizaciones de usuario y las autorizaciones de grupo para las funciones que se requiere autorizar.

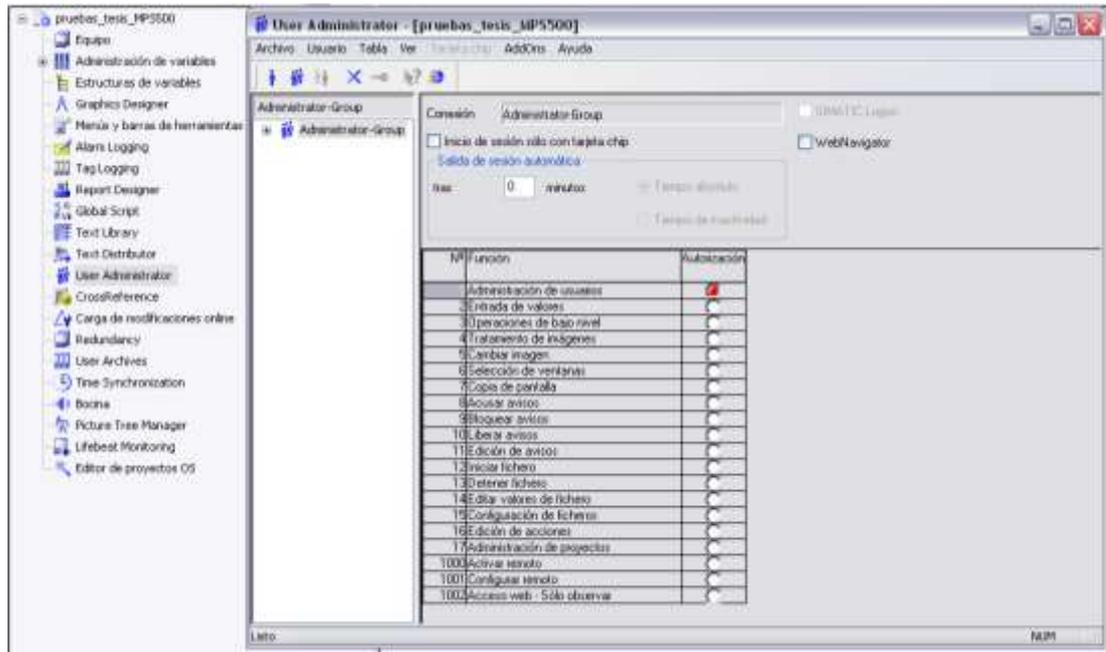


Figura 72: Administrador de usuario

Fuente: Software WinCC, año 2013

3.20.3 Crear un usuario

En el menú desplegable, haga clic en "Crear usuario". El nombre de usuario debe ser de una longitud de al menos cuatro caracteres y la contraseña de al menos seis caracteres de longitud.

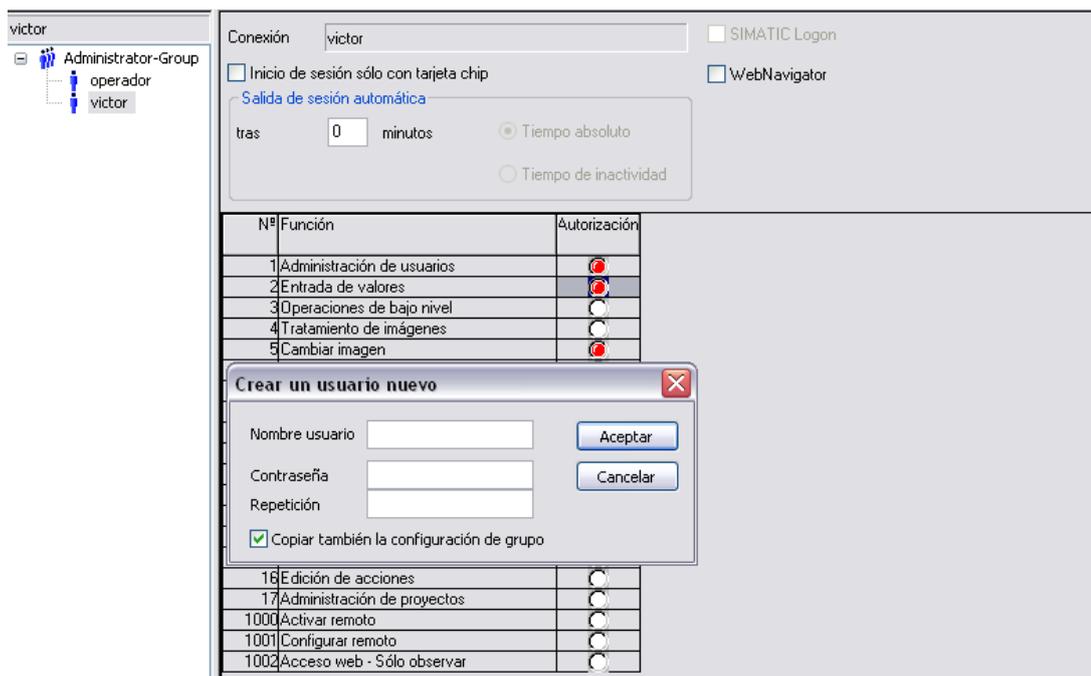


Figura 73: Crear un usuario y asignar autorizaciones

Fuente: Software WinCC, año 2013

En el *graphics designer* escoger los objetos que se requiere que tengan protección con contraseña y seleccionar el nivel de permiso.

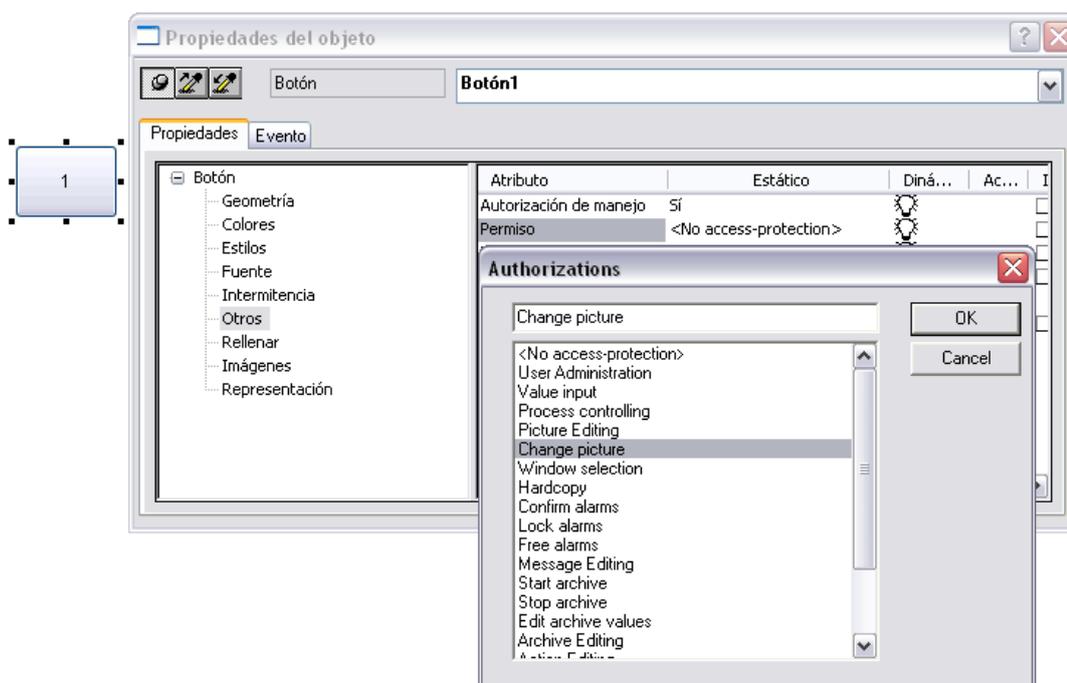


Figura 74: Ejemplo de cómo seleccionar el tipo de permiso

Fuente: Software WinCC, año 2013

3.21 Listado de variables y dirección

La comunicación del sistema SCADA WinCC con los PLC's lo realiza por medio del driver profibus, las variables de cada PLC se encuentran dentro de un enlace.

Estos son los siguientes enlaces con sus respectivas variables:

3.21.1 Enlace controlador estación distribución

Nombre del enlace es **1VE_AS_B** que contiene la dirección profibus número **2** y se comunica con el controlador de la estación de distribución, aquí se encuentran las principales variables que son las siguientes:

@1VE_AS_B@ConnectionStatees una variable con el tipo de datos de 32 bits sin signo que se encuentra en la dirección DD0 y recibe la información sobre el estado de la comunicación.

VE_Control_In es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW126 y recibe los estados de entrada del tablero de control.

VE_Control_Out es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW128 y recibe los estados de salida del tablero de control.

VE_Station_In es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW122 y recibe los estados de entrada de los sensores que se encuentran conectados en el controlador.

VE_Station_Out es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW124 y recibe los estados de salida de los dispositivos se encuentran conectados en el controlador.

VE_RC_Bits es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW120; entrega las órdenes de marcha paro y *reset* al controlador.

3.21.2 Enlace controlador estación verificación

Nombre del enlace es **2PR_AS_B** que contiene la dirección profibus número **4** y se comunica con el controlador de la estación de verificación, aquí se encuentran las principales variables que son las siguientes:

@2PR_AS_B@ConnectionState es una variable con el tipo de datos de 32 bits sin signo que se encuentra en la dirección DD0 y recibe la información sobre el estado de la comunicación.

PR_Control_In es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW126 y recibe los estados de entrada del tablero de control.

PR_Control_Out es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW128 y recibe los estados de salida del tablero de control.

PR_Station_In es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW122 y recibe los estados de entrada de los sensores que se encuentran conectados en el controlador.

PR_Station_Out es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW124 y recibe los estados de salida de los dispositivos se encuentran conectados en el controlador.

PR_RC_Bits es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW120; entrega las órdenes de marcha paro y *reset* al controlador.

3.21.3 Enlace controlador estación proceso

Nombre del enlace es **3BE_AS_B** que contiene la dirección profibus número **6** y se comunica con el controlador de la estación de proceso, aquí se encuentran las principales variables que son las siguientes:

@3BE_AS_B@ConnectionState es una variable con el tipo de datos de 32 bits sin signo que se encuentra en la dirección DD0 y recibe la información sobre el estado de la comunicación.

BE_Control_In es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW126 y recibe los estados de entrada del tablero de control.

BE_Control_Out es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW128 y recibe los estados de salida del tablero de control.

BE_Station_In es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW122 y recibe los estados de entrada de los sensores que se encuentran conectados en el controlador.

BE_Station_Out es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW124 y recibe los estados de salida de los dispositivos se encuentran conectados en el controlador.

BE_RC_Bits es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW120; entrega las órdenes de marcha paro y *reset* al controlador.

3.21.4 Enlace controlador estación manipulación I

Nombre del enlace es **4PA_AS_B** que contiene la dirección profibus número **8** y se comunica con el controlador de la estación de manipulación I, aquí se encuentran las principales variables que son las siguientes:

@4PA_AS_B@ConnectionState es una variable con el tipo de datos de 32 bits sin signo que se encuentra en la dirección DD0 y recibe la información sobre el estado de la comunicación.

PA4_Control_In es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW126 y recibe los estados de entrada del tablero de control.

PA4_Control_Out es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW128 y recibe los estados de salida del tablero de control.

PA4_Station_In es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW122 y recibe los estados de entrada de los sensores que se encuentran conectados en el controlador.

PA4_Station_Out es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW124 y recibe los estados de salida de los dispositivos se encuentran conectados en el controlador.

PA4_RC_Bits es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW120; entrega las órdenes de marcha paro y *reset* al controlador.

3.21.5 Enlace controlador estación clasificación

Nombre del enlace es **9SO_AS_B** que contiene la dirección profibus número **18** y se comunica con el controlador de la estación de clasificación, aquí se encuentran las principales variables que son las siguientes:

@9SO_AS_B@ConnectionState es una variable con el tipo de datos de 32 bits sin signo que se encuentra en la dirección DD0 y recibe la información sobre el estado de la comunicación.

SO_Control_In es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW126 y recibe los estados de entrada del tablero de control.

SO_Control_Out es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW128 y recibe los estados de salida del tablero de control.

SO_Station_In es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW122 y recibe los estados de entrada de los sensores que se encuentran conectados en el controlador.

SO_Station_Out es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW124 y recibe los estados de salida de los dispositivos se encuentran conectados en el controlador.

SO_RC_Bits es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW120; entrega las órdenes de marcha paro y *reset* al controlador.

3.21.6 Enlace controlador estación manipulación II

Nombre del enlace es **8PA_AS_B** que contiene la dirección profibus número **16** y se comunica con el controlador de la estación de manipulación II, aquí se encuentran las principales variables que son las siguientes:

@8PA_AS_B@ConnectionState es una variable con el tipo de datos de 32 bits sin signo que se encuentra en la dirección DD0 y recibe la información sobre el estado de la comunicación.

PA8_Control_In es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW126 y recibe los estados de entrada del tablero de control.

PA8_Control_Out es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW128 y recibe los estados de salida del tablero de control.

PA8_Station_In es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW122 y recibe los estados de entrada de los sensores que se encuentran conectados en el controlador.

PA8_Station_Out es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW124 y recibe los estados de salida de los dispositivos se encuentran conectados en el controlador.

PA8_RC_Bits es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW120; entrega las órdenes de marcha paro y *reset* al controlador.

3.21.7 Enlace controlador estación almacenamiento

Nombre del enlace es **7HRL20_KFA** que contiene la dirección profibus número **14** y se comunica con el controlador de la estación de almacenamiento, aquí se encuentran las principales variables que son las siguientes:

@7HRL20_KFA@ConnectionState es una variable con el tipo de datos de 32 bits sin signo que se encuentra en la dirección DD0 y recibe la información sobre el estado de la comunicación.

HRL20_Control_In es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW126 y recibe los estados de entrada del tablero de control.

HRL20_Control_Out es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW128 y recibe los estados de salida del tablero de control.

HRL20_Station_In es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW122 y recibe los estados de entrada de los sensores que se encuentran conectados en el controlador.

HRL20_Station_Out es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW124 y recibe los estados de salida de los dispositivos se encuentran conectados en el controlador.

HRL20_RC_Bits es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW120; entrega las órdenes de marcha paro y *reset* al controlador.

StorageCell_1 es una variable con el tipo de datos de 16 bits sin signo que se encuentra en el área de datos DB31 y direccionamiento DBW0; entrega la información sobre el estado del almacenamiento número 1.

StorageCell_2 es una variable con el tipo de datos de 16 bits sin signo que se encuentra en el área de datos DB31 y direccionamiento DBW2; entrega la información sobre el estado del almacenamiento número 2.

StorageCell_3 es una variable con el tipo de datos de 16 bits sin signo que se encuentra en el área de datos DB31 y direccionamiento DBW4; entrega la información sobre el estado del almacenamiento número 3.

StorageCell_4 es una variable con el tipo de datos de 16 bits sin signo que se encuentra en el área de datos DB31 y direccionamiento DBW6; entrega la información sobre el estado del almacenamiento número 4.

StorageCell_5 es una variable con el tipo de datos de 16 bits sin signo que se encuentra en el área de datos DB31 y direccionamiento DBW8; entrega la información sobre el estado del almacenamiento número 5.

StorageCell_6 es una variable con el tipo de datos de 16 bits sin signo que se encuentra en el área de datos DB31 y direccionamiento DBW10; entrega la información sobre el estado del almacenamiento número 6.

StorageCell_7 es una variable con el tipo de datos de 16 bits sin signo que se encuentra en el área de datos DB31 y direccionamiento DBW12; entrega la información sobre el estado del almacenamiento número 7.

StorageCell_8 es una variable con el tipo de datos de 16 bits sin signo que se encuentra en el área de datos DB31 y direccionamiento DBW14; entrega la información sobre el estado del almacenamiento número 8.

StorageCell_9 es una variable con el tipo de datos de 16 bits sin signo que se encuentra en el área de datos DB31 y direccionamiento DBW16; entrega la información sobre el estado del almacenamiento número 9.

StorageCell_10 es una variable con el tipo de datos de 16 bits sin signo que se encuentra en el área de datos DB31 y direccionamiento DBW18; entrega la información sobre el estado del almacenamiento número 10.

StorageCell_11 es una variable con el tipo de datos de 16 bits sin signo que se encuentra en el área de datos DB31 y direccionamiento DBW20; entrega la información sobre el estado del almacenamiento número 11.

StorageCell_12 es una variable con el tipo de datos de 16 bits sin signo que se encuentra en el área de datos DB31 y direccionamiento DBW22; entrega la información sobre el estado del almacenamiento número 12.

StorageCell_13 es una variable con el tipo de datos de 16 bits sin signo que se encuentra en el área de datos DB31 y direccionamiento DBW24; entrega la información sobre el estado del almacenamiento número 13.

StorageCell_14 es una variable con el tipo de datos de 16 bits sin signo que se encuentra en el área de datos DB31 y direccionamiento DBW26; entrega la información sobre el estado del almacenamiento número 14.

StorageCell_15 es una variable con el tipo de datos de 16 bits sin signo que se encuentra en el área de datos DB31 y direccionamiento DBW28; entrega la información sobre el estado del almacenamiento número 15.

StorageCell_16 es una variable con el tipo de datos de 16 bits sin signo que se encuentra en el área de datos DB31 y direccionamiento DBW30; entrega la información sobre el estado del almacenamiento número 16.

StorageCell_17 es una variable con el tipo de datos de 16 bits sin signo que se encuentra en el área de datos DB31 y direccionamiento DBW32; entrega la información sobre el estado del almacenamiento número 17.

StorageCell_18 es una variable con el tipo de datos de 16 bits sin signo que se encuentra en el área de datos DB31 y direccionamiento DBW34; entrega la información sobre el estado del almacenamiento número 18.

StorageCell_19 es una variable con el tipo de datos de 16 bits sin signo que se encuentra en el área de datos DB31 y direccionamiento DBW36; entrega la información sobre el estado del almacenamiento número 19.

StorageCell_20 es una variable con el tipo de datos de 16 bits sin signo que se encuentra en el área de datos DB31 y direccionamiento DBW38; entrega la información sobre el estado del almacenamiento número 20.

3.21.8 Enlace controlador estación transportador

Nombre del enlace es **ConveyorF6_V02** que contiene la dirección profibus número **20** y se comunica con el controlador de la estación de transporte, aquí se encuentran las principales variables que son las siguientes:

@ConveyorF6_V02@ConnectionState es una variable con el tipo de datos de 32 bits sin signo que se encuentra en la dirección DD0 y recibe la información sobre el estado de la comunicación.

HRL20_Control_In es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW126 y recibe los estados de entrada del tablero de control.

HRL20_Control_Out es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW128 y recibe los estados de salida del tablero de control.

HRL20_RC_Bits es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW120; entrega las órdenes de marcha y paro al controlador.

ConvF6_SpecFunc2 es una variable con el tipo de datos de 16 bits sin signo que se encuentra en la dirección MW136; recibe la confirmación de encendido y apagado del transportador.

ConvF6_IX6_In es una variable con el tipo de datos de 32 bits sin signo que se encuentra en el área de datos DB80 y direccionamiento DD32; recibe los estados de entrada del sistema transportador en el área de la estación de manipulación II.

ConvF6_IX5_In es una variable con el tipo de datos de 32 bits sin signo que se encuentra en el área de datos DB80 y direccionamiento DD26; recibe los estados de entrada del sistema transportador en el área de la estación de almacenamiento.

ConvF6_IX4_In es una variable con el tipo de datos de 32 bits sin signo que se encuentra en el área de datos DB80 y direccionamiento DD20; recibe los estados de entrada del sistema transportador en el área de la estación de robot y ensamblaje.

ConvF6_IX3_In es una variable con el tipo de datos de 32 bits sin signo que se encuentra en el área de datos DB80 y direccionamiento DD14; recibe los estados de entrada del sistema transportador en el área de la estación de visión.

ConvF6_IX2_In es una variable con el tipo de datos de 32 bits sin signo que se encuentra en el área de datos DB80 y direccionamiento DD8; recibe los estados de entrada del sistema transportador en el área de la estación de manipulación I.

ConvF6_IX1_In es una variable con el tipo de datos de 32 bits sin signo que se encuentra en el área de datos DB80 y direccionamiento DD2; recibe los estados de entrada del sistema transportador en el área de la estación de verificación.

camara_entrada es una variable con el tipo de datos de 8 bits sin signo que se encuentra en el área de datos DB80 y direccionamiento DBB18; recibe los estados de entrada de la estación de visión.

camara_salida es una variable con el tipo de datos de 8 bits sin signo que se encuentra en el área de datos DB80 y direccionamiento DBB80; recibe los estados de salida de la estación de visión.

Entradas_Robot es una variable con el tipo de datos de 8 bits sin signo que se encuentra en el área de datos DB80 y direccionamiento DBB24; recibe los estados de entrada de la estación de robot.

Salidas_Robot es una variable con el tipo de datos de 8 bits sin signo que se encuentra en el área de datos DB80 y direccionamiento DBB86; recibe los estados de salida de la estación de robot.

Dentro de la conexión encontramos grupos de variables que son los siguientes:

DriverValue

Pallet_Ident

torno

En el grupo **DriverValue** se encuentran las siguientes variables:

DriverValue_IX1 es una variable con el tipo de datos de 8 bits sin signo que se encuentra en el área de datos DB1 y direccionamiento DBB108; recibe los estados de habilitación de la estación número 1.

DriverValue_IX2 es una variable con el tipo de datos de 8 bits sin signo que se encuentra en el área de datos DB1 y direccionamiento DBB109; recibe los estados de habilitación de la estación número 2.

DriverValue_IX3 es una variable con el tipo de datos de 8 bits sin signo que se encuentra en el área de datos DB1 y direccionamiento DBB110; recibe los estados de habilitación de la estación número 3.

DriverValue_IX4 es una variable con el tipo de datos de 8 bits sin signo que se encuentra en el área de datos DB1 y direccionamiento DBB111; recibe los estados de habilitación de la estación número 4.

DriverValue_IX5 es una variable con el tipo de datos de 8 bits sin signo que se encuentra en el área de datos DB1 y direccionamiento DBB112; recibe los estados de habilitación de la estación número 5.

DriverValue_IX6 es una variable con el tipo de datos de 8 bits sin signo que se encuentra en el área de datos DB1 y direccionamiento DBB113; recibe los estados de habilitación de la estación número 6.

En el grupo **Pallet_Ident** se encuentran las siguientes variables:

Pallet_at_Station_1 es una variable con el tipo de datos de 8 bits sin signo que se encuentra en el área de datos DB60 y direccionamiento DBB2; recibe la identificación del pallet que se encuentra en la estación número 1.

Pallet_at_Station_2 es una variable con el tipo de datos de 8 bits sin signo que se encuentra en el área de datos DB60 y direccionamiento DBB32; recibe la identificación del pallet que se encuentra en la estación número 2.

Pallet_at_Station_3 es una variable con el tipo de datos de 8 bits sin signo que se encuentra en el área de datos DB60 y direccionamiento DBB62; recibe la identificación del pallet que se encuentra en la estación número 3.

Pallet_at_Station_4 es una variable con el tipo de datos de 8 bits sin signo que se encuentra en el área de datos DB60 y direccionamiento DBB92; recibe la identificación del pallet que se encuentra en la estación número 4.

Pallet_at_Station_5 es una variable con el tipo de datos de 8 bits sin signo que se encuentra en el área de datos DB60 y direccionamiento DBB122; recibe la identificación del pallet que se encuentra en la estación número 5.

Pallet_at_Station_6 es una variable con el tipo de datos de 8 bits sin signo que se encuentra en el área de datos DB60 y direccionamiento DBB152; recibe la identificación del pallet que se encuentra en la estación número 6.

En el grupo **torno** se encuentran las siguientes variables:

puerta_torno es una variable binaria que se encuentra en la dirección E4.5; recibe el estado de la puerta del torno.

carros_funcionando_torno es una variable binaria que se encuentra en la dirección E4.6; recibe el estado si está en movimiento el carro del torno.

plato_de_tres_mordazas_torno es una variable binaria que se encuentra en la dirección E4.7; recibe el estado si el plato se encuentra abierto o cerrado.

robot_listo es una variable binaria que se encuentra en la dirección E3.4; recibe el estado si el robot está listo para trabajar con el torno.

comenzar_trabajo_torno es una variable binaria que se encuentra en la dirección E3.5; recibe el estado si el torno va empezar a trabajar.

torno_listo es una variable binaria que se encuentra en la dirección E3.6; recibe el estado si el torno está listo para trabajar con el robot.

3.22 Variables internas

Para poder realizar las animaciones es necesario agregar algunas variables internas con diferentes tipos de datos.

Nota: Es recomendable que la parte de control lo realice el controlador y no la aplicación de visualización por motivo de tiempos de ejecución.

3.23 Tipos de datos

Tipo de datos	Longitud del dato	Rango del valor
Variable binaria	Bit	0 , 1
Valor de 8 bits con signo	Byte	-128 , 127
Valor de 8 bits sin signo	Byte	0 , 255
Valor de 16 bits con signo	Word	-32768 , 32767
Valor de 16 bits sin signo	Word	0 , 65535
Valor de 32 bits con signo	DoubleWord	-2147483648 , 2147483647
Valor de 32 bits sin signo	DoubleWord	0 , 4294967295
Número en coma flotante 32 bits IEEE 754	DoubleWord	+3.402823e+38
Número en coma flotante 64 bits IEEE 754	8 Bytes	+1.79769313486231e+308
Variable de texto, juego de caracteres 8 bits	0 hasta 255 Bytes	
Variable de texto, juego de caracteres 16 bits	0 hasta 255 Word	

Tabla 1: Tipos de datos

Fuente: Ayuda de Software WinCC, año 2013

3.24 Animaciones y acciones

En el sistema SCADA WinCC se encuentran 4 formas de realizar animaciones, las cuales son las siguientes: diálogo dinámico, acción en C, acción en VBS y variable. También tenemos 3 formas de realizar una acción en las cuáles son acción en C, acción en VBS y conexión directa.

3.24.1 Diálogo dinámico

Se puede elegir el tipo de datos a utilizarse, escoger el disparo de la variable expresión o fórmula.

Para poder escribir una expresión tenemos los siguientes operadores lógicos.

Operador	Descripción	Representación
Y (AND)	Comparación de bits	&
O (OR)	Comparación de bits	
NOT	Inversión lógica	!

Tabla 2: Operadores lógicos para escribir en el cuadro de diálogo dinámico

Fuente: <http://www.monografias.com/trabajos2/guiavb/guiavb.shtml>, año 2000

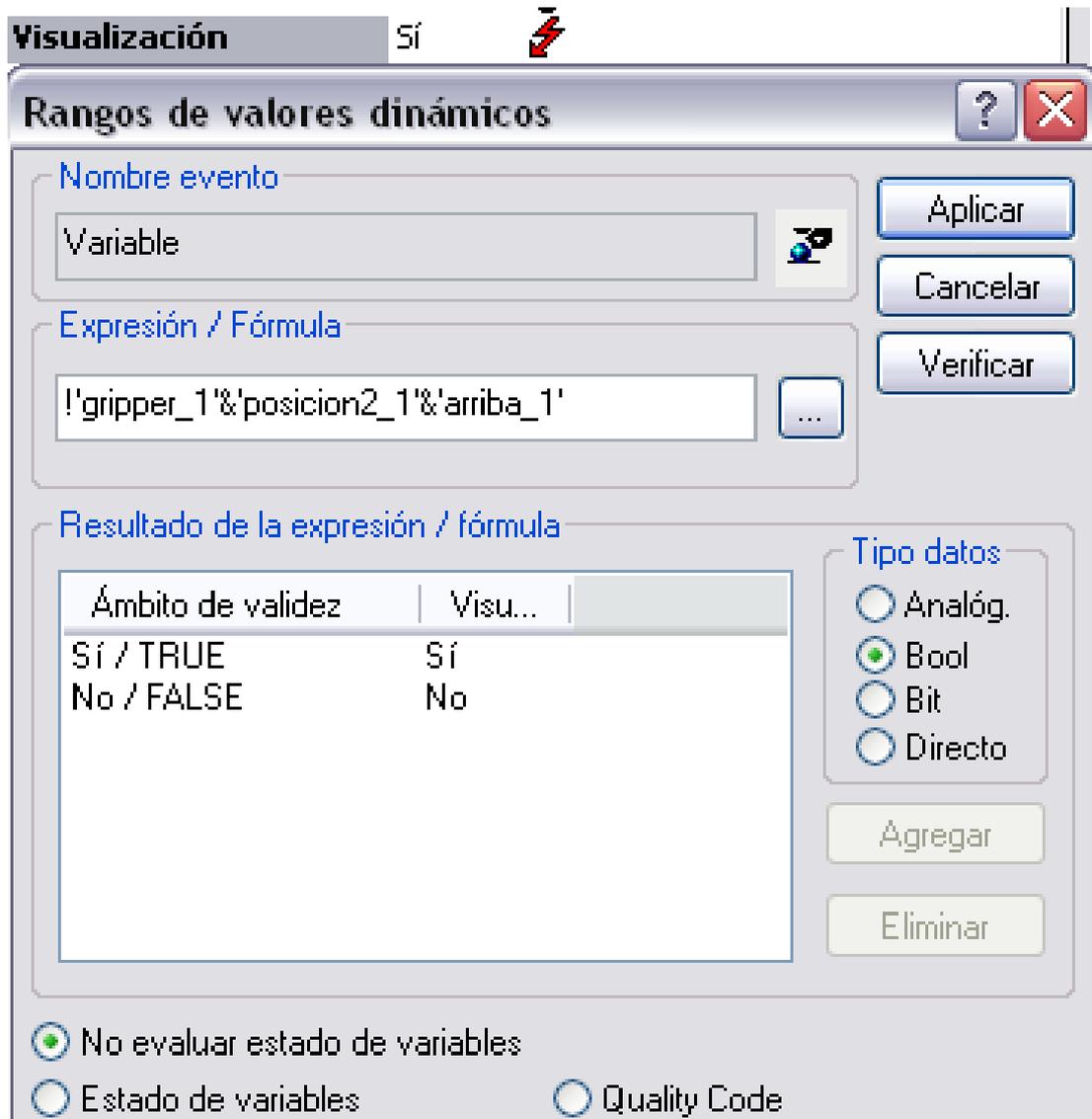


Figura 75: Dialogo dinámico

Fuente: Los Autores, año 2013

3.24.2 Acción en C

Usa el lenguaje de programación en C y tenemos un ejemplo de programación de una acción en C.

```

#include "apdefap.h"
void OnClick(char*  lpszPictureName,  char*  lpszObjectName,  char*
lpszPropertyName)
{
// WINCC:TAGNAME_SECTION_START
// syntax: #define TagNameInAction "DMTagName"
// next TagID : 1
// WINCC:TAGNAME_SECTION_END

// WINCC:PICNAME_SECTION_START
// syntax: #define PicNameInAction "PictureName"
// next PicID : 1
// WINCC:PICNAME_SECTION_END
{
DeactivateRTProject ();
}
}
}

```

Figura 76: Acción en C

Fuente: Software WinCC, año 2013

3.24.3 Acción en VBS

Usa el lenguaje de programación en Visual Basic y tenemos las declaraciones más usadas para poder realizar una acción.

Dim es para declarar un variable como privada.

Set y **HMIRuntime.Tags("")** crea la variable en objeto y se la puede visualizar en las referencias cruzadas.

Dim variable

Set variable=**HMIRuntime.Tags**("variable")

Read sirve para leer una variable

variable.**Read**

Write sirve para escribir una variable

variable.**Write**

Value indica que la variable es un número

color.**Value**

If Then... Else End If

Es una sentencia condicional que si se cumple una condición, realiza una instrucción determinada y si no cumple la condición realiza otra instrucción.

If (cumple condición) **Then**

Ejecuta acción

Else

Ejecuta otra acción

EndIf

3.24.4 Funciones con números

Para las operaciones básicas se utiliza los siguientes operadores aritméticos:

Operador	Operación
+	Suma
-	Resta
*	Multiplicación
/	División
\	División sin decimales
Mod	Resto de una división
^	Eleva a un exponente

Tabla 3: Operadores aritméticos

Fuente: <http://www.monografias.com/trabajos2/guiavb/guiavb.shtml>, año 2000

Ejemplo

`variable.Value = variable.Value+1`

3.24.5 Operadores relacionales

Comprueba 2 o más variables o números para mostrar sus relaciones, diferencias o semejanzas. Esta relación solo aceptan 2 resultados: verdadero o falso.

Operador	Relación
=	Igual que
<>	No igual que
<	Mayor que
<=	Mayor o igual que
>	Menor que
=>	Igual o menor que

Tabla 4: Operadores relacionales

Fuente: <http://www.monografias.com/trabajos2/guiavb/guiavb.shtml>, año 2000

3.24.6 Operadores lógicos

Encontramos los siguientes operadores lógicos:

AND			OR			XOR		
<u>A</u>	<u>B</u>	<u>X</u>	<u>A</u>	<u>B</u>	<u>X</u>	<u>A</u>	<u>B</u>	<u>X</u>
0	0	0	0	0	0	0	0	0
0	1	0	0	1	1	0	1	1
1	0	0	1	0	1	1	0	1
1	1	1	1	1	1	1	1	0

EQV			IMP			NOT	
<u>A</u>	<u>B</u>	<u>X</u>	<u>A</u>	<u>B</u>	<u>X</u>	<u>A</u>	<u>X</u>
0	0	1	0	0	1	0	1
0	1	0	0	1	1	1	0
1	0	0	1	0	0		
1	1	1	1	1	1		

Tabla 5: Tablas lógicas

Fuente: <http://www.monografias.com/trabajos2/guiavb/guiavb.shtml>, año 2000

Abs nos entrega el valor absoluto de un número variable. `variable.Value=Abs(variable1.Value)`

3.24.7 Variable

Toma el valor actualizado para realizar la animación que está implicada



Figura 77: Animación por medio de una variable

Fuente: Laboratorio de Fabricación Flexible UPS Guayaquil, año 2013

3.2.4.8 Conexión directa

Podemos ejecutar una acción realizando una conexión directa.

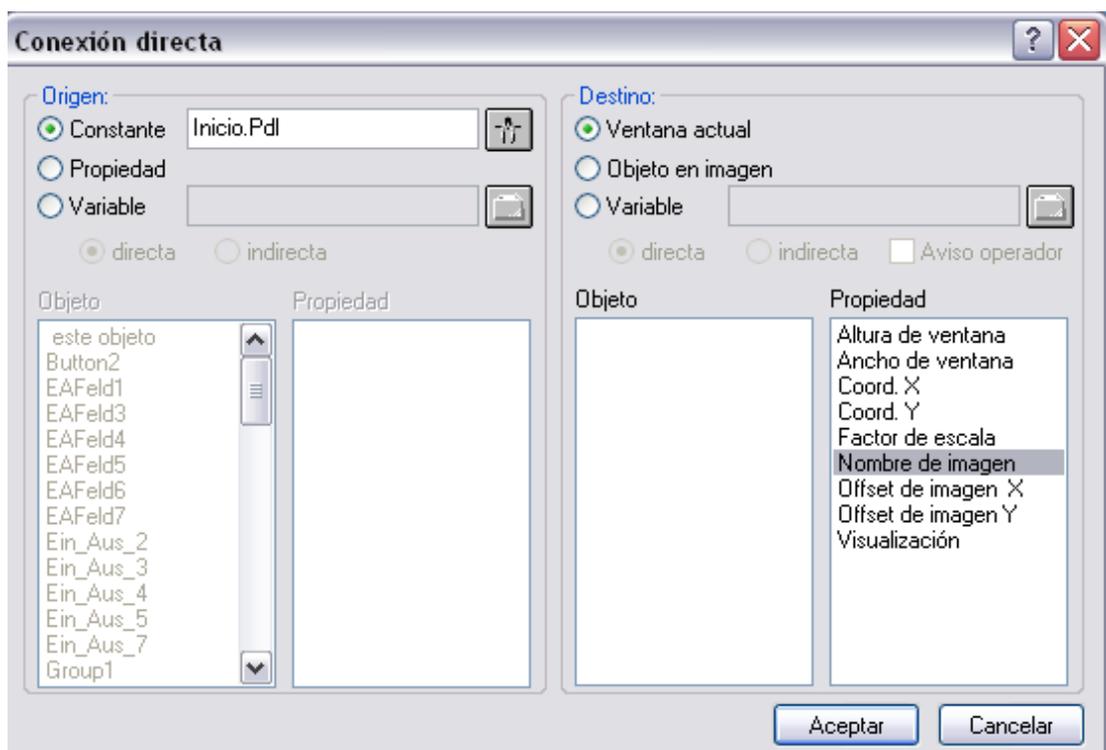


Figura 78: Animación por medio de una conexión directa

Fuente: Software WinCC, año 2013

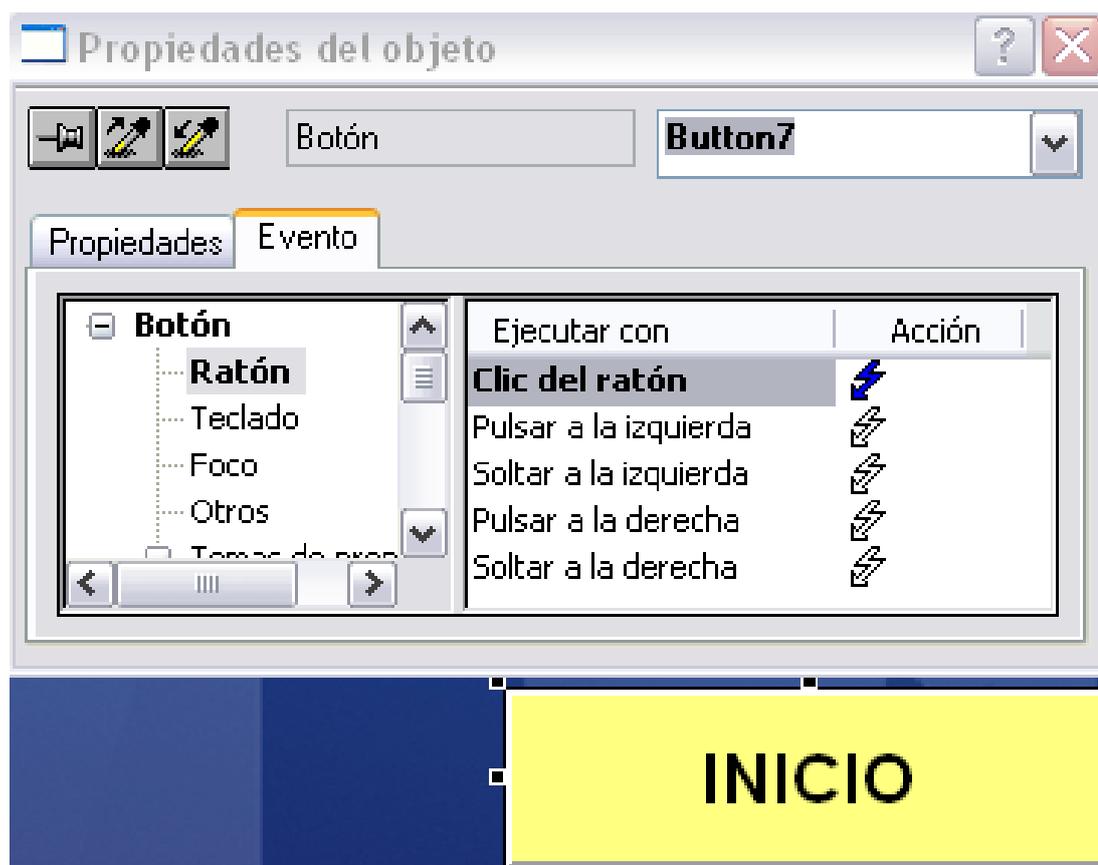


Figura 79: Ejecución de un evento por medio de una conexión directa

Fuente: Los Autores, año 2013

3.25 Cómo y porqué el uso del software WinCC SCADA

Se usa el software WinCC SCADA porque la Universidad Politécnica Salesiana sede Guayaquil cuenta con el software y las licencias requeridas para su funcionamiento.

Se utiliza como base la aplicación que provee el sistema MPS - 500 y el programa demo para su control, porque se encuentran las variables y la licencia depende de la cantidad de variables externas que se emplean pues la mayoría de las variables son datos de palabra o doble palabra.

En la aplicación desarrollada específicamente en la pantalla de Inicio se encuentran varios botones, el uno de ellos llamado "Aplicación Demo", este botón nos lleva a la pantalla de la banda transportadora demo y desde ahí se puede navegar a las

diferentes estaciones donde la mayor parte de la supervisión y control están realizados con una acción en VBS porque es un lenguaje de alto nivel. Para esta acción se utiliza un disparo de 1s de forma acíclica, es decir, cada segundo se va ejecutar la acción planteada porque las ejecuciones de las estaciones son bastantes rápidas y para poder leer todas las instrucciones y ejecutarlas se escoge el tiempo del disparador de 1 s y no el disparo estándar que es de 2 s, si se escoge un tiempo menor hay inconvenientes con el tiempo de escaneo y de ejecución por la cantidad de variables que se maneja.

En el desarrollo de la acción se utiliza algunos métodos para acondicionar las variables por ejemplo.

```
banda.value=PA8_Station_Out.value And 256  
banda.value=banda.value / 256  
banda.Write
```

PA8_Station_Out es una variable de 16 bits donde solo se necesita el estado del bit 8 por eso se aplica una and con el valor 256 en decimal (100000000 en binario) porque se consigue hacer cero los demás bits, este resultado es guardado en otra variable y esta variable se la divide para 256 o se la desplaza 8 espacios, porque de esta manera se consigue el estado de ese bit en el bit menos significativo.

Se usan variables internas porque nos ayuda a guardar datos temporales o cambiantes en el ordenador, nos permite ejecutar una acción o una animación y no necesita ser comunicado con los PLC's.

El otro botón llamado "Aplicación" nos lleva a la pantalla de banda transportadora y desde ahí se puede navegar a las diferentes estaciones donde la mayor parte de la supervisión y control están realizados por medio del cuadro de diálogo dinámico.

En diferentes pantallas encontramos animaciones para los atributos realizadas con cuadro de diálogo dinámico, acción en C, acción VBS y conexión por variable. Para ejecutar un evento encontramos acción en C, acción VBS y conexión directa. El software nos brinda esta variedad de herramientas y se las está poniendo en práctica porque se pretende utilizar la mayor parte de las herramientas del software.

Por ejemplo se puede navegar a diferentes pantallas por medio de una conexión directa, se puede ejecutar una acción en VBS o C.

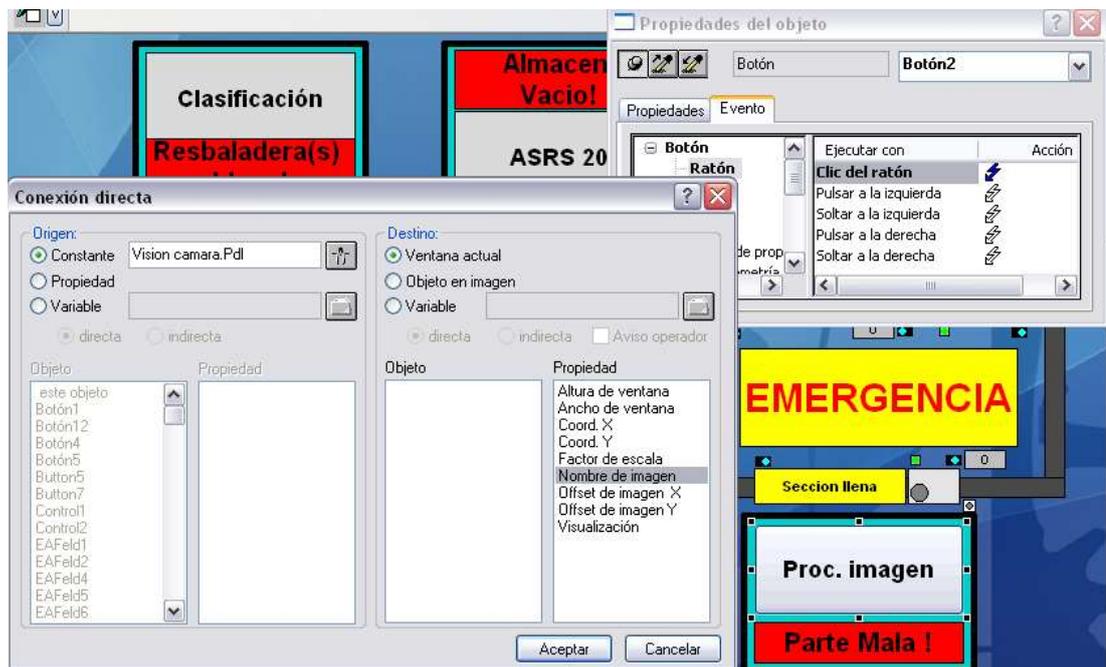


Figura 80: Evento realizado dando clic con el ratón y ejecutándose con una conexión directa

Fuente: Los Autores, año 2013

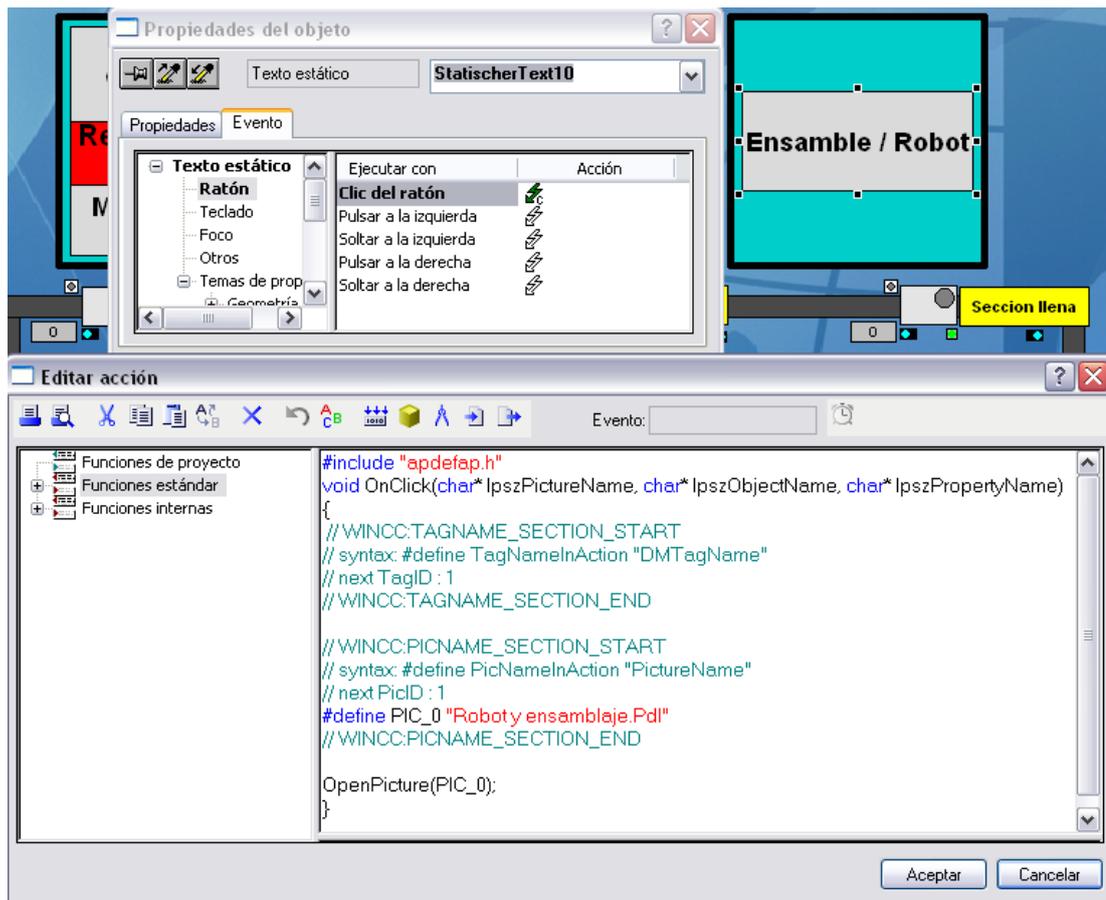


Figura 81: Un evento realizado dando clic con el ratón y ejecutándose bajo una acción en C

Fuente: Los Autores, año 2013

CAPÍTULO IV

4 COMPARACIÓN ENTRE PANTALLAS

4.1 Pantalla de inicio

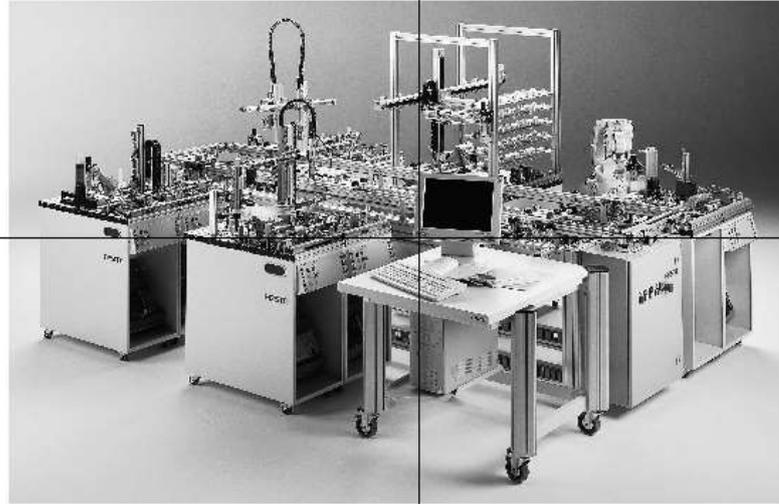
La diferencia entre la pantalla de la Figura 82, con respecto a la Figura 83 son las siguientes:

En la Figura 83, se encuentran botones para poder navegar a diferentes pantallas.

En la Figura 82, se manejan varios idiomas, mientras que en la Figura 83 solo se interactúa con texto en español.

En la Figura 83, se encuentra el logo de la universidad, la fecha y la hora.

MPS 500 - Process visualization



Exit Runtime



Change Language



Start



© Festo AG 2006

Figura 82: Pantalla principal (actual)

Fuente: MPS500_WinCCV60_Application_V5_DP, año 2006



Figura 83: Pantalla principal (nueva)

Fuente: Los Autores, año 2013



Figura 85: Pantalla del sistema transportador demo (nuevo)

Fuente: Los Autores, año 2013

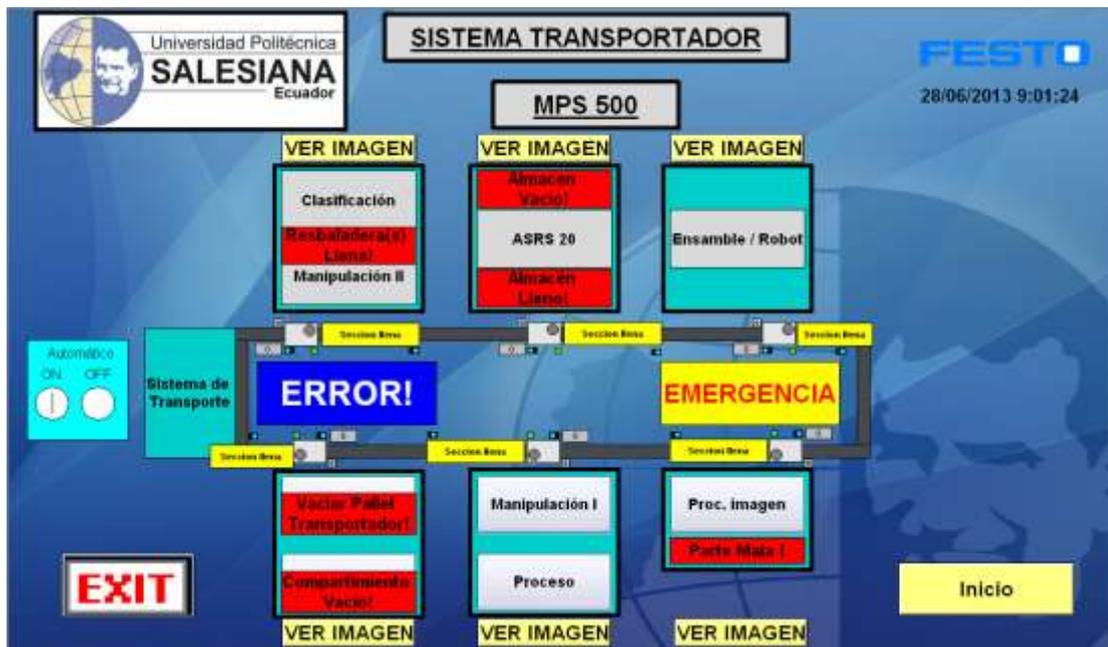


Figura 86: Pantalla del sistema transportador (nuevo)

Fuente: Los Autores, año 2013

4.3 Comisionar

El texto que se presenta en la Figura 87 es en inglés y en la Figura 88 el texto que se presenta es en español.

En la Figura 87 el orden de las estaciones no es el correcto, mientras que en la Figura 88 está ordenado el MPS 500 y se consigue regresar mediante un botón hacia la pantalla de inicio.

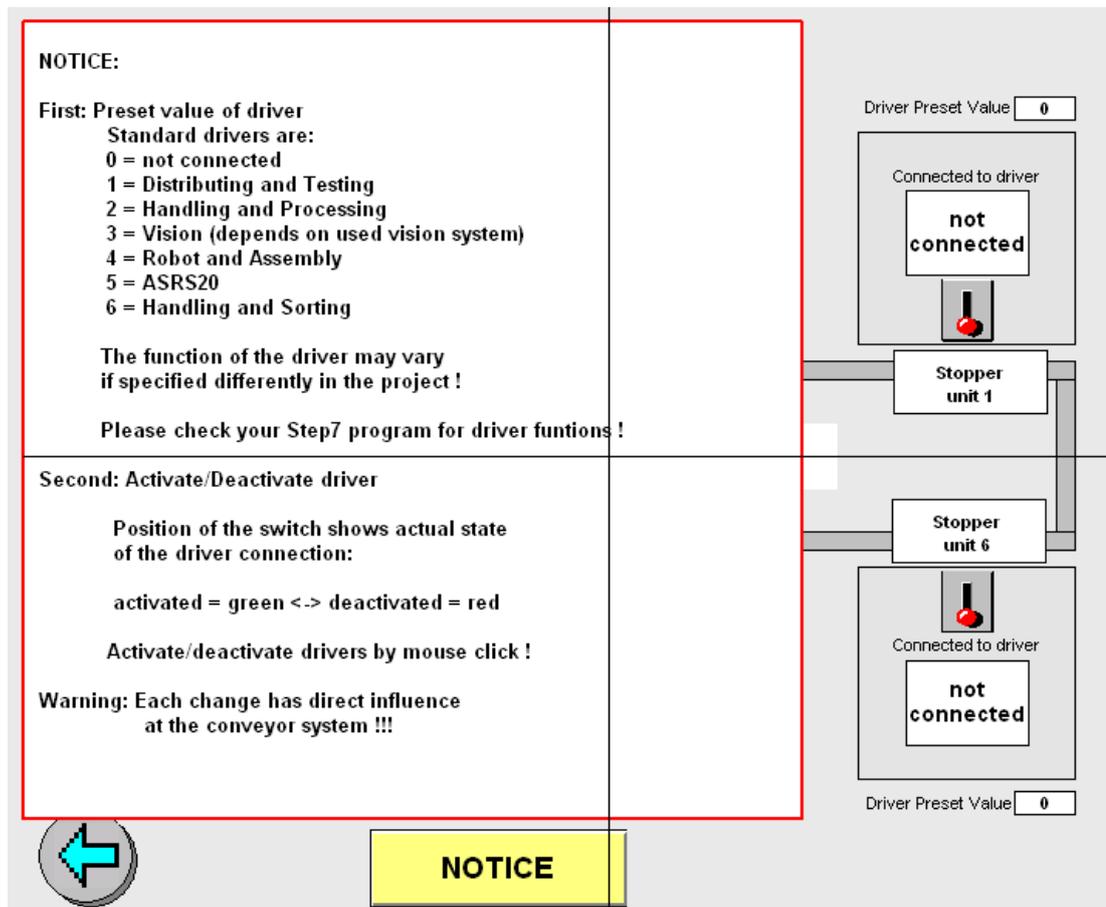


Figura 87: Pantalla de comisionar (actual)

Fuente: MPS500_WinCCV60_Application_V5_DP, año 2003

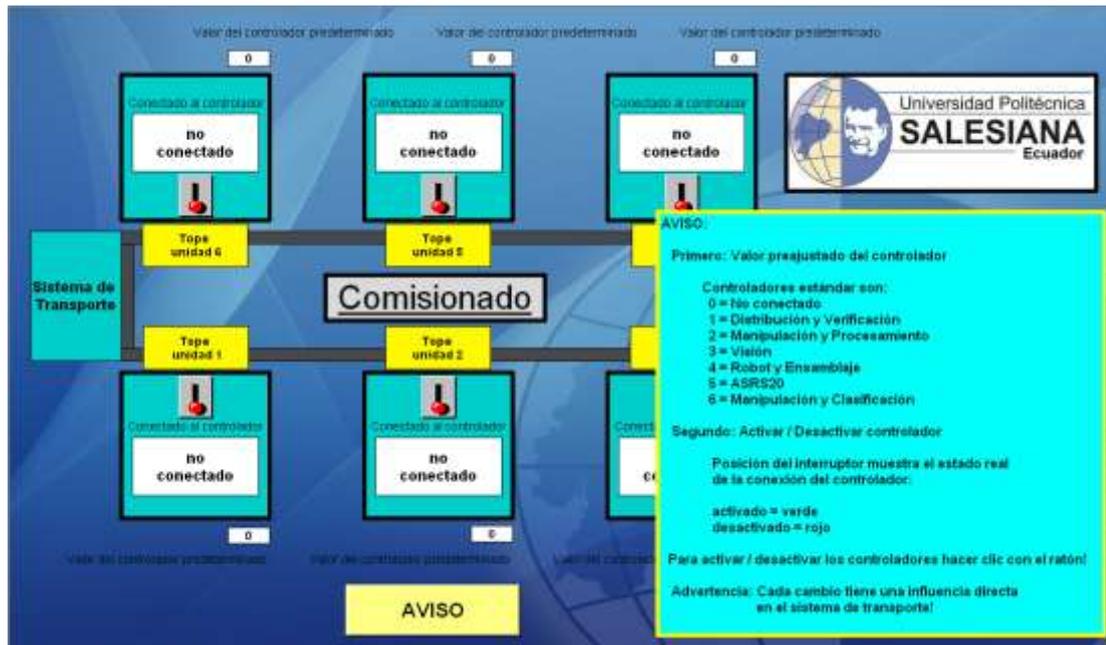


Figura 88: Pantalla de comisionar (nueva)

Fuente: Los Autores, año 2013

4.4 Red profibus

El texto que se presenta en la Figura 89 es en inglés y en la Figura 90 el texto que se presenta es en español.

En la Figura 89 se encuentra con dirección profibus la cámara, pero esto no existe. La cámara se comunica con el PLC del sistema transportador por medio de señales cableadas de entradas y salidas.

Profibus network for visualisation with WinCC

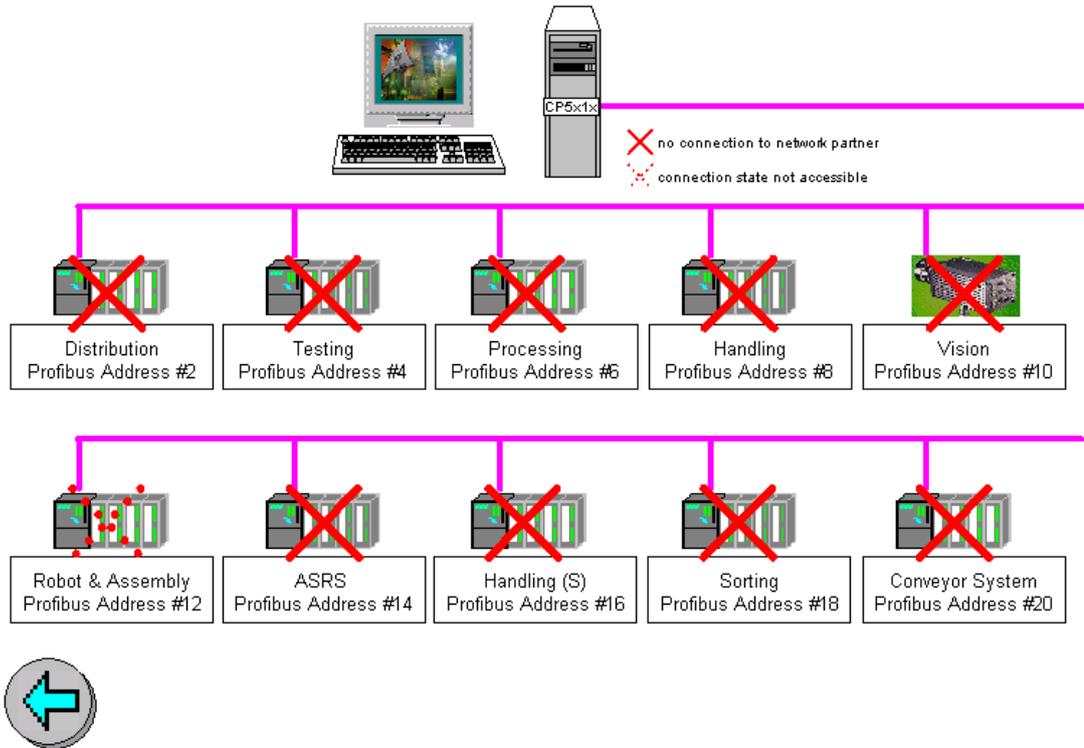


Figura 89: Pantalla de la red profibus (actual)

Fuente: MPS500_WinCCV60_Application_V5_DP, año 2003

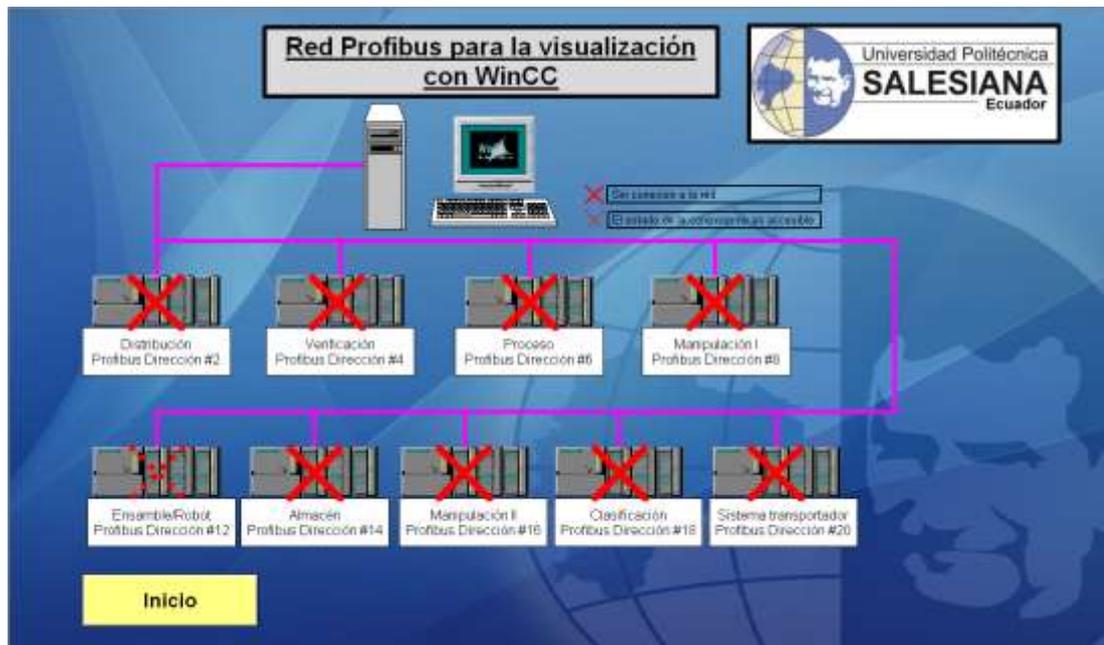


Figura 90 Pantalla de la red profibus (nueva)

Fuente: Los Autores, año 2013

4.5 Estación de distribución

De la pantalla actual de la estación de distribución Figura 91, se puede describir lo siguiente:

- Se encuentra una imagen de la estación y si se visualiza animación en una vista de planta.
- Para simular el proceso se tienen círculos que cambian de color cuando alguna variable cambia de estado.
- Tablero de control para encender, apagar o reiniciar el proceso.
- Todos los textos se encuentran en el idioma inglés.
- Se observa botones para indicar que el proceso se encuentra en emergencia o falla.

De la pantalla nueva de la estación de distribución Figura 92, se puede resumir los siguientes cambios:

- Se visualizan las diferentes posiciones que se puede encontrar el brazo, se puede introducir la cantidad de fichas que se colocan en el compartimiento para poder observar una animación más real a lo que está pasando en ese momento de ejecutar el programa.
- Veremos como el contador va ir mermando y como van desapareciendo las piezas del compartimiento.
- Los textos se podrán observar en idioma español.
- Se colocan botones para ingresar a cualquier estación (pantalla) desde este punto, por medio de conexión directa y de Scripts realizados en C.
- Se muestra la imagen de un tablero donde se podrá encender, apagar o reiniciar la estación desde pantalla.
- La animación es realiza con una acción en VBS.

La pantalla de verificación Figura 93, se realizaron las animaciones con cuadro de diálogo dinámico, a continuación se las describe:

- Se visualizan indicadores como flechas para saber en qué posición y hacia donde se está moviendo el brazo.
- Los textos se podrán observar en idioma español.
- Se colocan botones para poder navegar a cualquier estación (pantalla) desde este punto, por medio de conexión directa y de Scripts realizados en C.
- Se muestra la imagen de un tablero donde se podrá encender, apagar o reiniciar la estación desde pantalla.

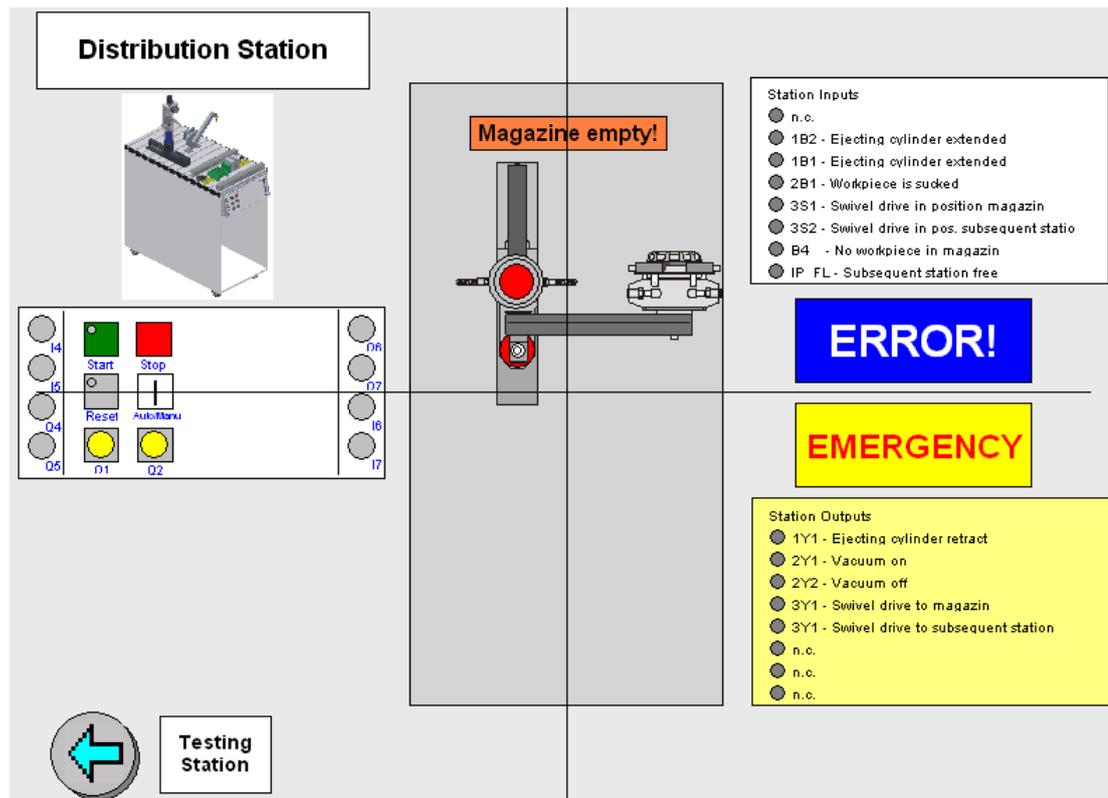


Figura 91: Pantalla de la estación de distribución (actual)

Fuente: MPS500_WinCCV60_Application_V5_DP, año 2003

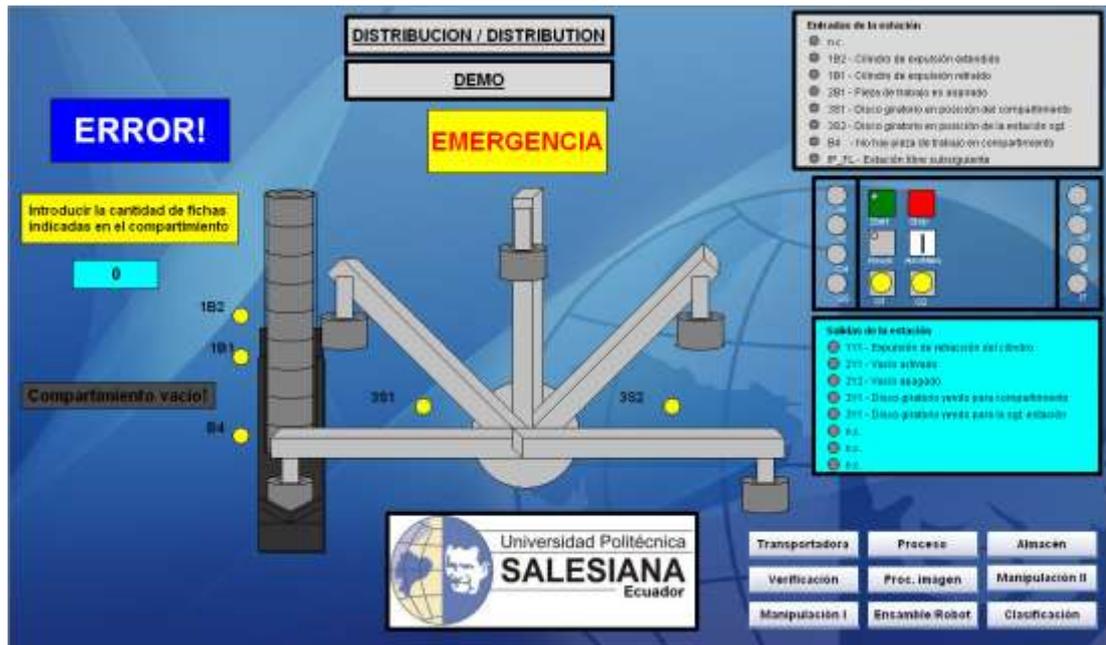


Figura 92: Pantalla de la estación de distribución demo (nueva)

Fuente: Los Autores, año 2013

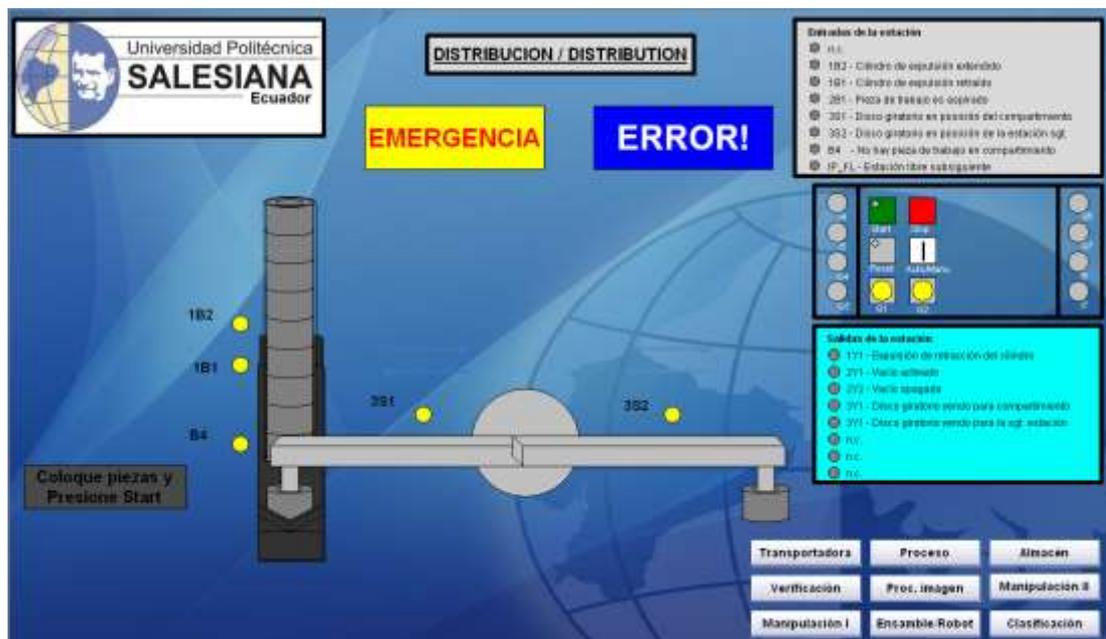


Figura 93: Pantalla de la estación de distribución (nueva)

Fuente: Los Autores, año 2013

4.6 Estación de verificación

De la pantalla actual de la estación de verificación Figura 94, se puede describir lo siguiente:

- Se encuentra una imagen de la estación y no se visualiza animación alguna cuando está en funcionamiento.
- Para simular el proceso se tienen círculos que cambian de color cuando alguna variable cambia de estado.
- Tablero de control para encender, apagar o reiniciar el proceso.
- Todos los textos se encuentran en el idioma inglés.
- Se observa botones para indicar que el proceso se encuentra en emergencia o falla.

De la pantalla nueva de la estación de verificación Figura 95, se puede resumir los siguientes cambios:

- Se observa como sube y baja la pieza con el cilindro elevador, también se visualiza el movimiento del cilindro de expulsión.
- Cuando se enciende la cama de aire se encuentra la pieza en la cama y hace el respectivo movimiento hacia el transportador.
- Cuando no pasa la pieza la verificación de la altura se ve en la visualización que la pieza baja con el cilindro elevador y el cilindro de expulsión obliga a la pieza a salir a la resbaladera de piezas rechazadas donde va ir incrementando un contador que cuando llegue a 5 manda un mensaje de "resbaladera llena".
- Los textos se podrán observar en idioma español.
- Se colocan botones para ingresar a cualquier estación (pantalla) desde este punto, por medio de conexión directa y de Scripts realizados en C.
- Se muestra la imagen de un tablero donde se podrá encender, apagar o reiniciar la estación desde pantalla.
- La animación es realiza con una acción en VBS.

La pantalla de verificación Figura 96, se realizaron las animaciones con cuadro de diálogo dinámico, a continuación se las describe:

- Se visualizan indicadores como flechas para saber en qué posición y hacia donde se está moviendo el brazo.
- Los textos se podrán observar en idioma español.
- Se colocan botones para poder navegar a cualquier estación (pantalla) desde este punto, por medio de conexión directa y de Scripts realizados en C.
- Se muestra la imagen de un tablero donde se podrá encender, apagar o reiniciar la estación desde pantalla.

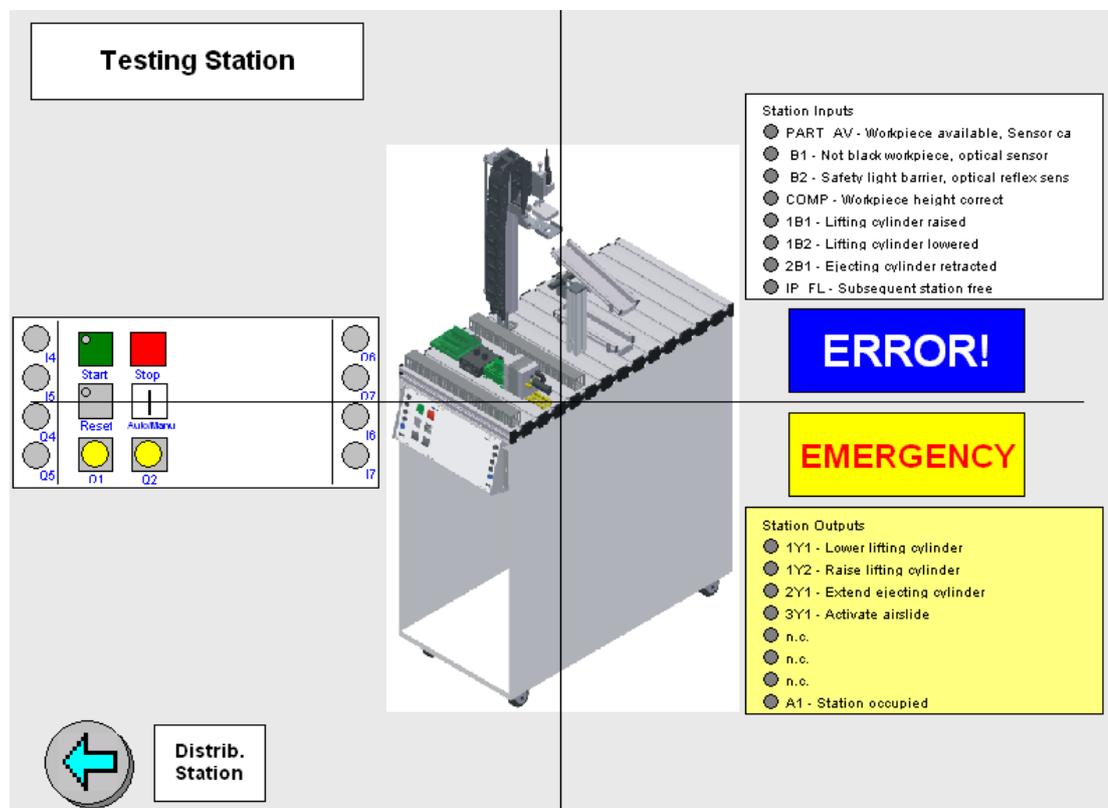


Figura 94: Pantalla de la estación de verificación (actual)

Fuente: MPS500_WinCCV60_Application_V5_DP, año 2003

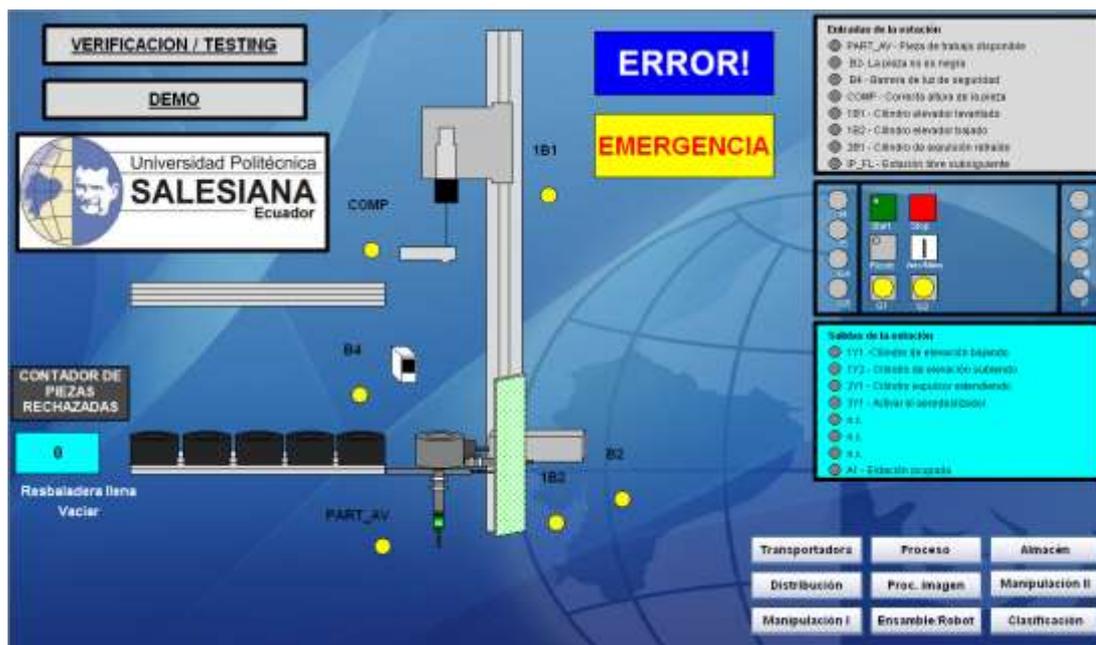


Figura 95: Pantalla de la estación de verificación demo (nueva)

Fuente: Los Autores, año 2013

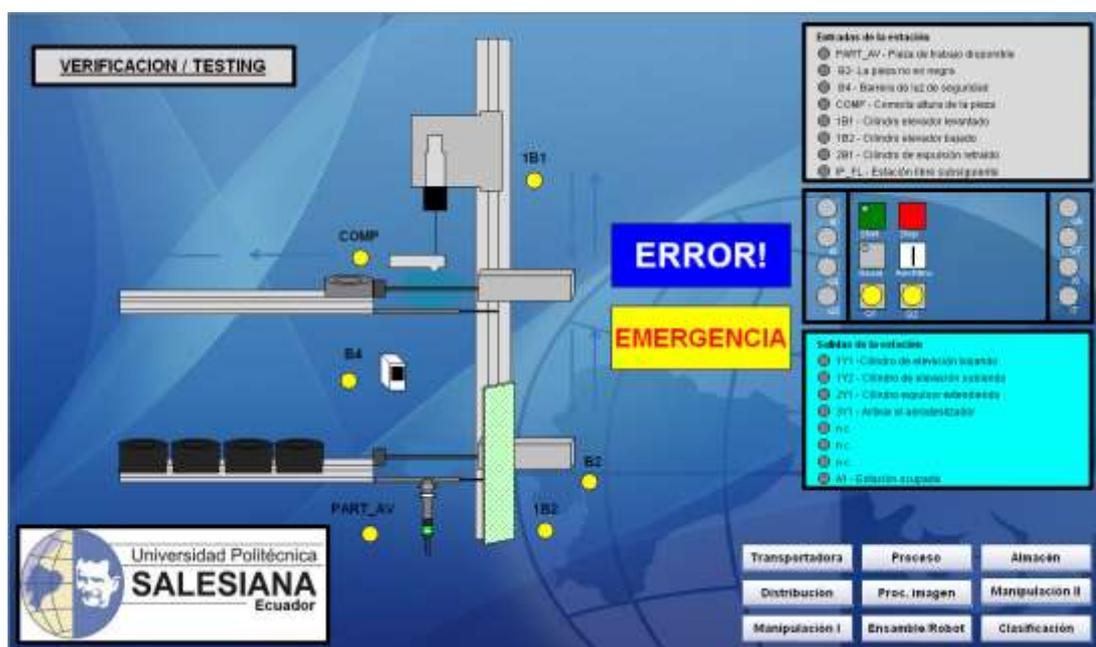


Figura 96: Pantalla de la estación de verificación (nueva)

Fuente: Los Autores, año 2013

4.7 Estación de manipulación I

De la pantalla actual de la estación de manipulación I Figura 97, se puede describir lo siguiente:

- Se encuentra una imagen de la estación y no se visualiza animación alguna cuando está en funcionamiento.
- Para simular el proceso se tienen círculos que cambian de color cuando alguna variable cambia de estado.
- Tablero de control para encender, apagar o reiniciar el proceso.
- Todos los textos se encuentran en el idioma inglés.
- Se observa botones para indicar que el proceso se encuentra en emergencia o falla.

De la pantalla nueva de la estación de manipulación I Figura 98, se puede resumir los siguientes cambios:

- Se coloca una imagen compuesta de soportes, brazo para la pinza de agarre de fichas.
- El brazo realiza movimientos en cuatro direcciones, estos son abajo, arriba, derecha e izquierda.
- Se puede visualizar la apertura y cierre de la pinza para simular el agarre de la ficha. Igualmente se observa aparecer y desaparecer la ficha según el proceso de esta estación.
- Los textos se podrán observar en idioma español.
- Se colocan botones para ingresar a cualquier estación (pantalla) desde este punto, por medio de conexión directa y de Scripts realizados en C.
- Se muestra la imagen de un tablero donde se podrá encender, apagar o reiniciar la estación desde pantalla.
- La animación es realiza con una acción en VBS.

La pantalla de manipulación I Figura 99, se realizaron las animaciones con cuadro de diálogo dinámico, a continuación se las describe:

- Se coloca una imagen compuesta de soportes, brazo para la pinza de agarre de fichas.
- El brazo realiza movimientos en 2 direcciones, estos son derecha e izquierda, tiene flechas indicando el movimiento que está realizando en 4 direcciones estas son derecha, izquierda, arriba y abajo.
- Los textos se podrán observar en idioma español.
- Se colocan botones para poder navegar a cualquier estación (pantalla) desde este punto, por medio de conexión directa y de Scripts realizados en C.
- Se muestra la imagen de un tablero donde se podrá encender, apagar o reiniciar la estación desde pantalla.

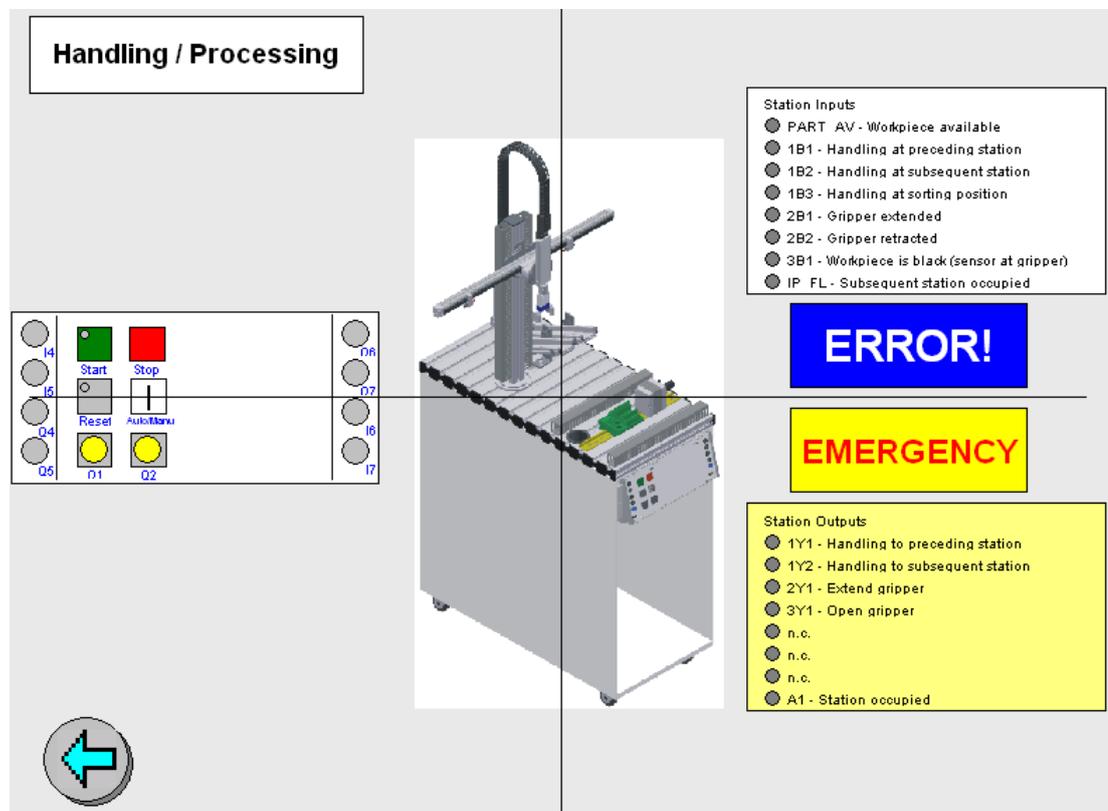


Figura 97: Pantalla de la estación de manipulación I (actual)

Fuente: MPS500_WinCCV60_Application_V5_DP, año 2003

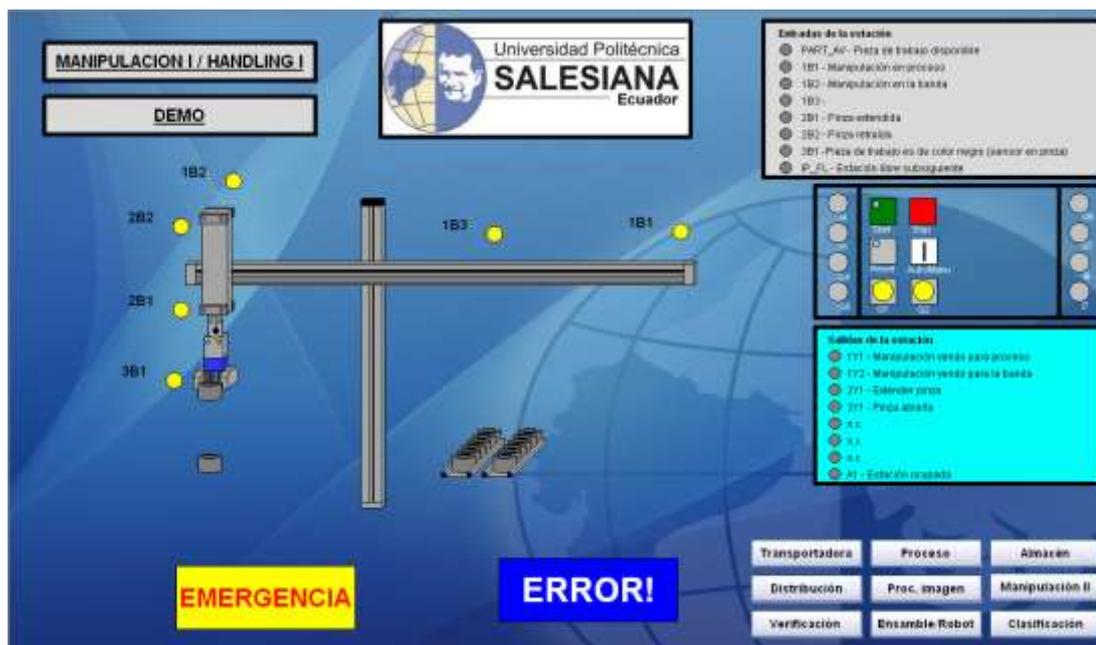


Figura 98: Pantalla de la estación de manipulación I demo (nueva)

Fuente: Los Autores, año 2013

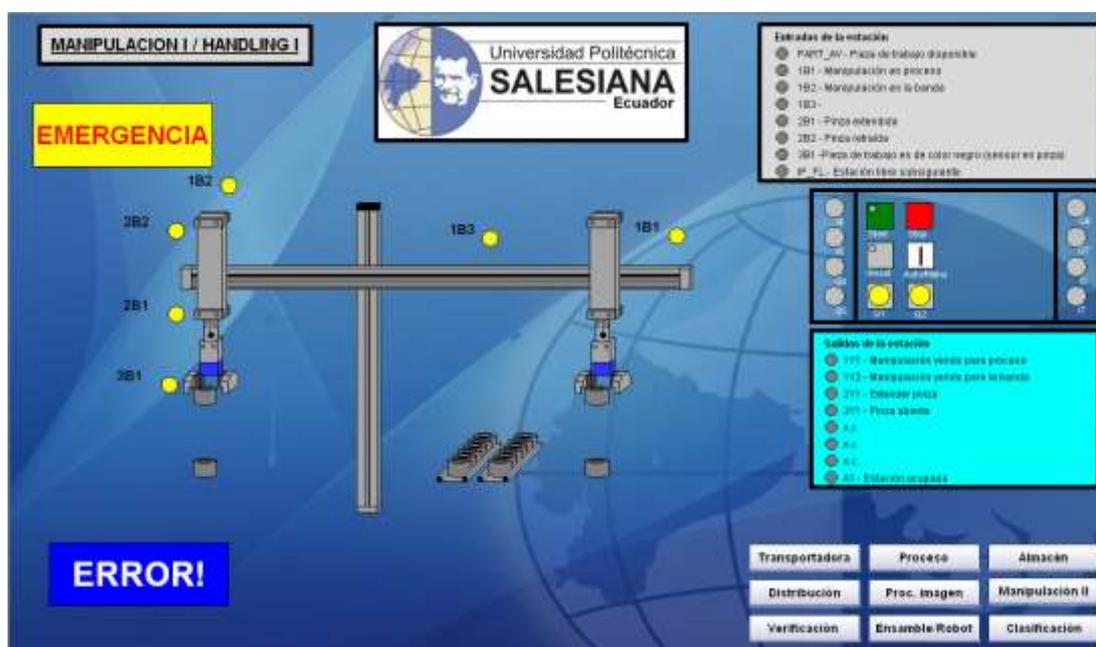


Figura 99: Pantalla de la estación de manipulación I (nueva)

Fuente: Los Autores, año 2013

4.8 Estación de procesamiento

En la Figura 100 se puede observar lo siguiente:

- Se visualiza un gráfico sin animación.
- Los textos en la pantalla se aprecian únicamente en el idioma inglés.
- La visualización de estados de las entradas y salidas en por un círculo que cambia su color al activarse.

En la Figura 101 se realizaron algunas mejoras, a continuación se las describe:

- Se observa una pantalla didáctica, con varias animaciones en sus gráficas.
- Se adicionaron varias gráficas para representar las maquinas en esta estación.
- Se colocan botones para ingresar en las diferentes estaciones.
- Se realiza la programación en script VBS de tal forma que la ficha cambie de lugar para simular que el disco gira, tal como sucede físicamente.
- Se colocan los indicadores circulares de estado (on/off), cerca de las maquinarias para visualizar cuando están en funcionamiento.

En la Figura 102 se realizaron las animaciones con cuadro de diálogo dinámico, a continuación se las describe:

- Se observa una pantalla didáctica, con varias animaciones en sus gráficas.
- Se adicionaron varias gráficas para representar las maquinas en esta estación.
- Se colocan botones para ingresar en las diferentes estaciones.
- Se colocan los indicadores circulares de estado (on/off), cerca de las maquinarias para visualizar cuando están en funcionamiento.

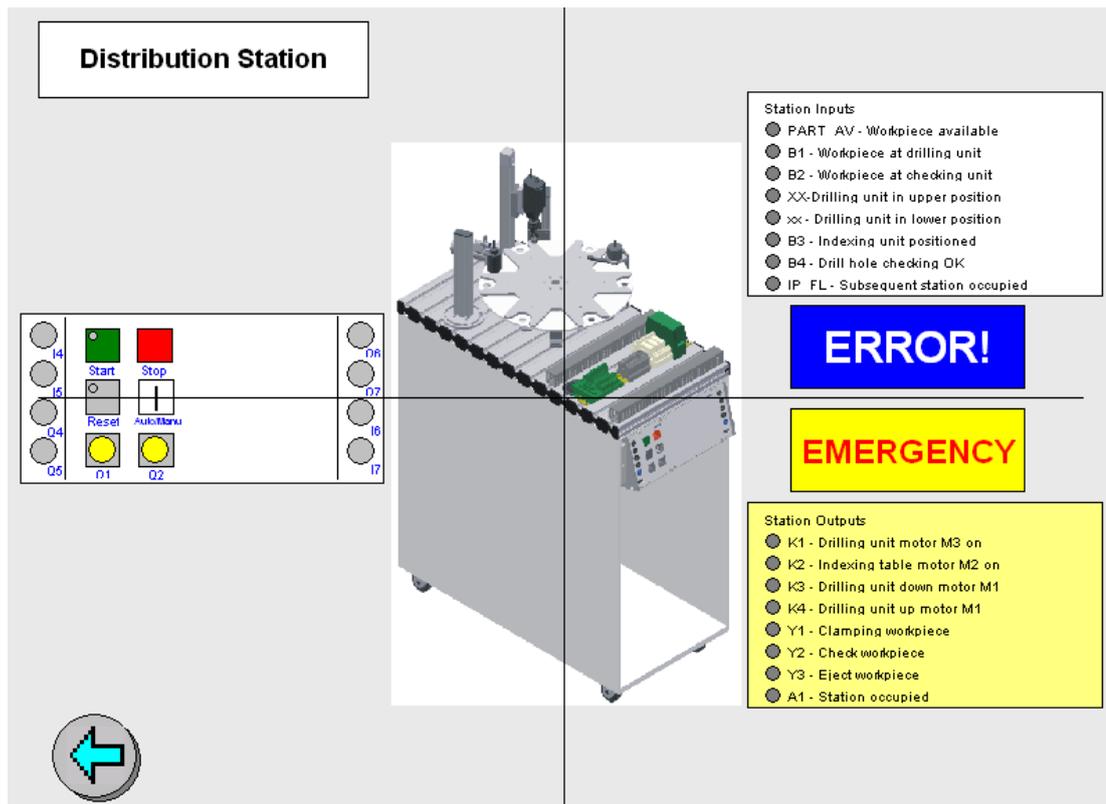


Figura 100: Pantalla de la estación de procesamiento (actual)

Fuente: MPS500_WinCCV60_Application_V5_DP, año 2003



Figura 101: Pantalla de la estación de procesamiento demo (nueva)

Fuente: Los Autores, año 2013

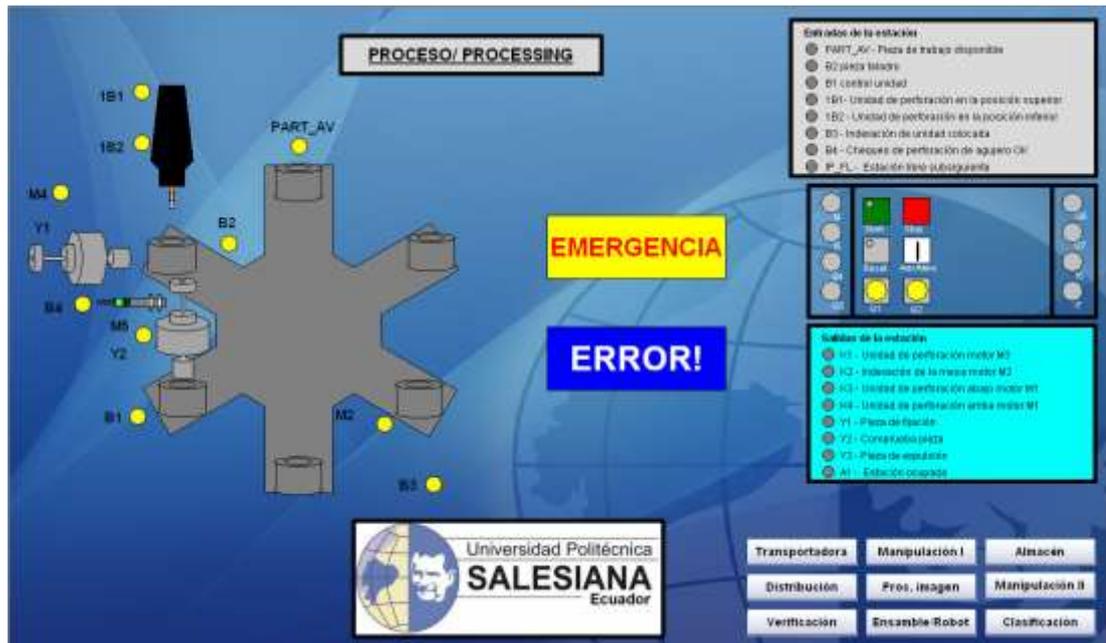


Figura 102: Pantalla de la estación de procesamiento (nueva)

Fuente: Los Autores, año 2013

4.9 Estación de visión

En la pantalla actual Figura 103, se observa lo siguiente:

- Se visualiza una pantalla en blanco, no se puede apreciar alguna animación cuando la ficha ingresa en esta estación.
- Cuando la cámara termina de verificar la ficha se observa sólo si no es aceptada mediante el texto de “*badpart*”.
- Los textos en la pantalla se aprecian únicamente en el idioma inglés.

En la pantalla nueva Figura 104, se visualizan los siguientes cambios:

- No se observa alguna gráfica mientras no lleguen fichas a esta estación, cuando ingresa una ficha se visualiza en la pantalla.
- Después que la cámara haya hecho el reconocimiento de la ficha, si es aceptada aumenta en 1 un contador de pieza aceptada y sobre la gráfica de la ficha aparece un “✓” de color verde. Si es rechazada aumenta en 1 un contador de pieza rechazada y sobre la ficha aparece una “X” de color rojo.
- El texto que se presenta es en idioma español.

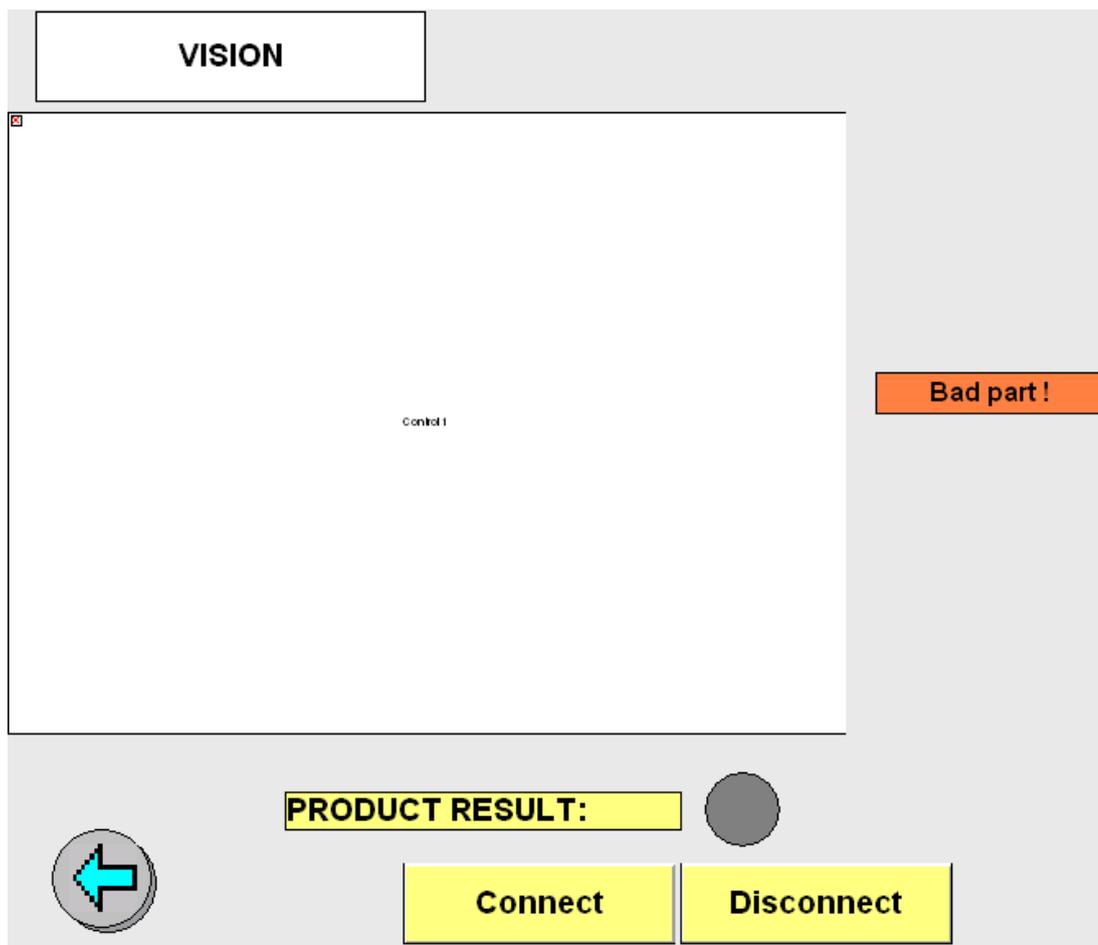


Figura 103: Pantalla de la estación de visión (actual)

Fuente: MPS500_WinCCV60_Application_V5_DP, año 2006

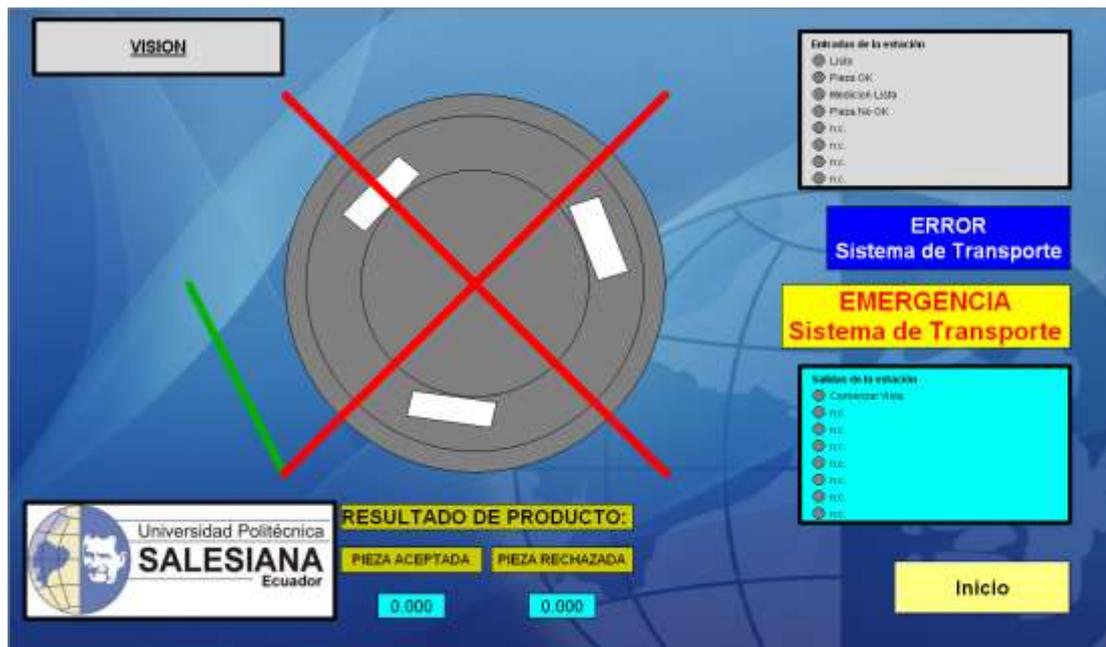


Figura 104: Pantalla de la estación de visión (nueva)

Fuente: Los Autores, año 2013

4.10 Estación de robot y ensamblaje

La pantalla actual de esta estación Figura 105, muestra lo siguiente:

- Imagen de la estación la cual no tiene animación.
- Textos en inglés.
- Dos botones que indican cuando comienza el ensamblaje y cuando una pieza ha sido rechazada, pero no funciona correctamente.

La pantalla nueva de esta estación Figura 106, muestra los siguientes cambios:

- Se cambia la imagen de la estación de robot y ensamblaje.
- Los textos fueron modificados al idioma español.
- Se agrega una nueva imagen para la simulación del torno.
- Las animaciones se las realiza mediante flechas y textos para indicar que la estación está trabajando o en reposo.
- Se le coloca un botón para regresar a la pantalla principal (INICIO).

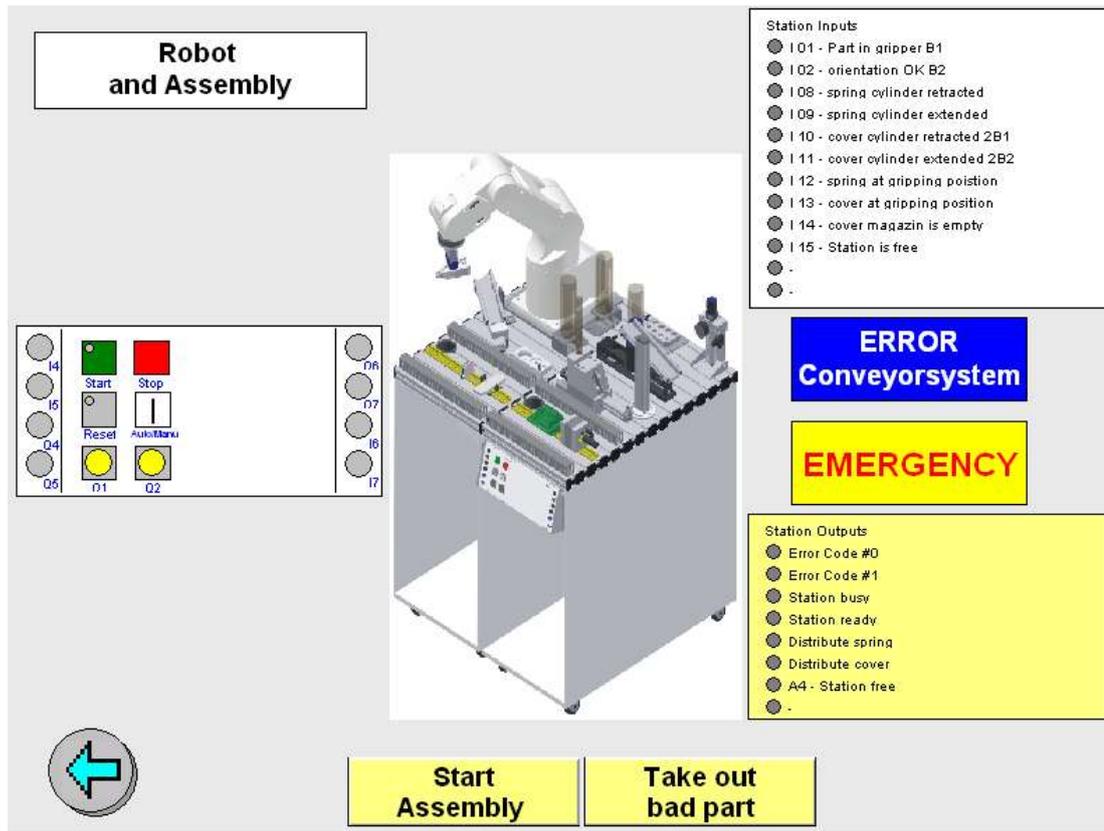


Figura 105: Pantalla de la estación de robot y ensamblaje (actual)

Fuente: MPS500_WinCCV60_Application_V5_DP, año 2003

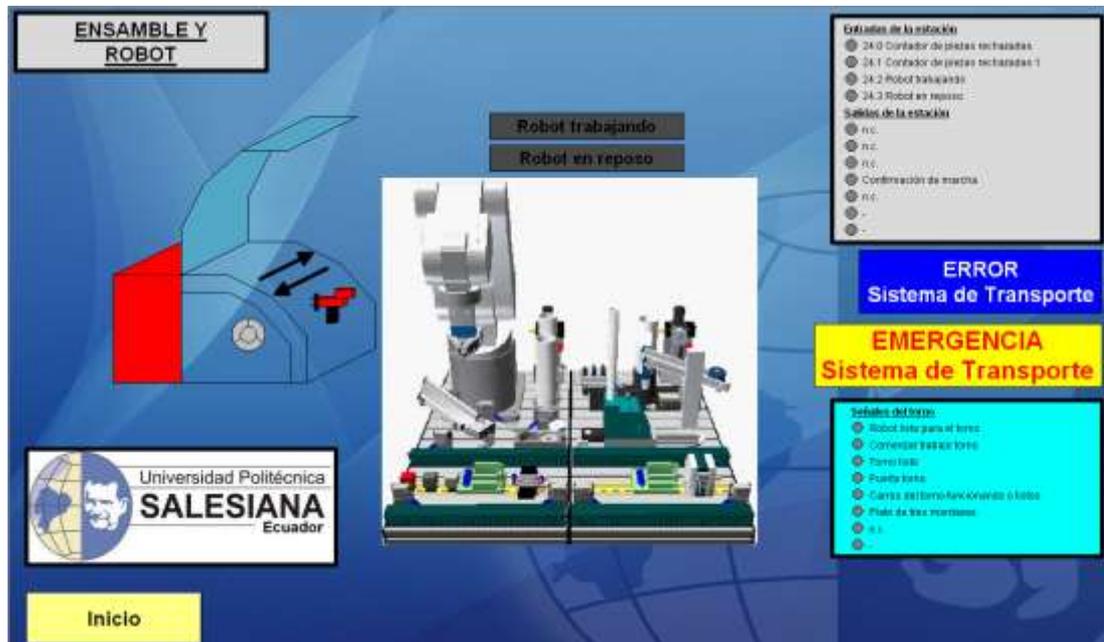


Figura 106: Pantalla de la estación de robot y ensamblaje (nueva)

Fuente: Los Autores, año 2013

4.11 Almacén

La pantalla actual del almacén Figura 107, contiene lo siguiente:

- Imagen de la estación del almacén sin animación.
- Textos de los estados y variables en el idioma inglés.
- Textos que indican cuando el almacén está lleno y vacío.
- Textos que indican cuando el almacén está en emergencia y falla.

La pantalla nueva del almacén Figura 108, se le hicieron varios cambios y se programa el control en VBS, las novedades de esta pantalla es presentada a continuación:

- Se realizan varias gráficas y se le coloca animación adaptándolo a lo que se puede visualizar realmente en el proceso.
- Se observa el movimiento del brazo que transporta las fichas para almacenarlas y retirarlas.
- Se agregan varios botones para navegar entre las diferentes pantallas.
- Los textos se los cambia al idioma español.
- Se agregan textos para indicar que el almacén se encuentra lleno o vacío.
- Se coloca un botón para reiniciar el proceso de almacenaje.

La pantalla del almacén Figura 109, se realizaron las animaciones con cuadro de diálogo dinámico, a continuación se las describe:

- Se agregan varios botones para navegar entre las diferentes pantallas.
- Los textos se los cambia al idioma español.
- Se agregan textos para indicar que el almacén se encuentra lleno o vacío.
- Se coloca un botón para reiniciar el proceso de almacenaje.
- Se observa una pantalla didáctica, con varias animaciones en sus gráficas.

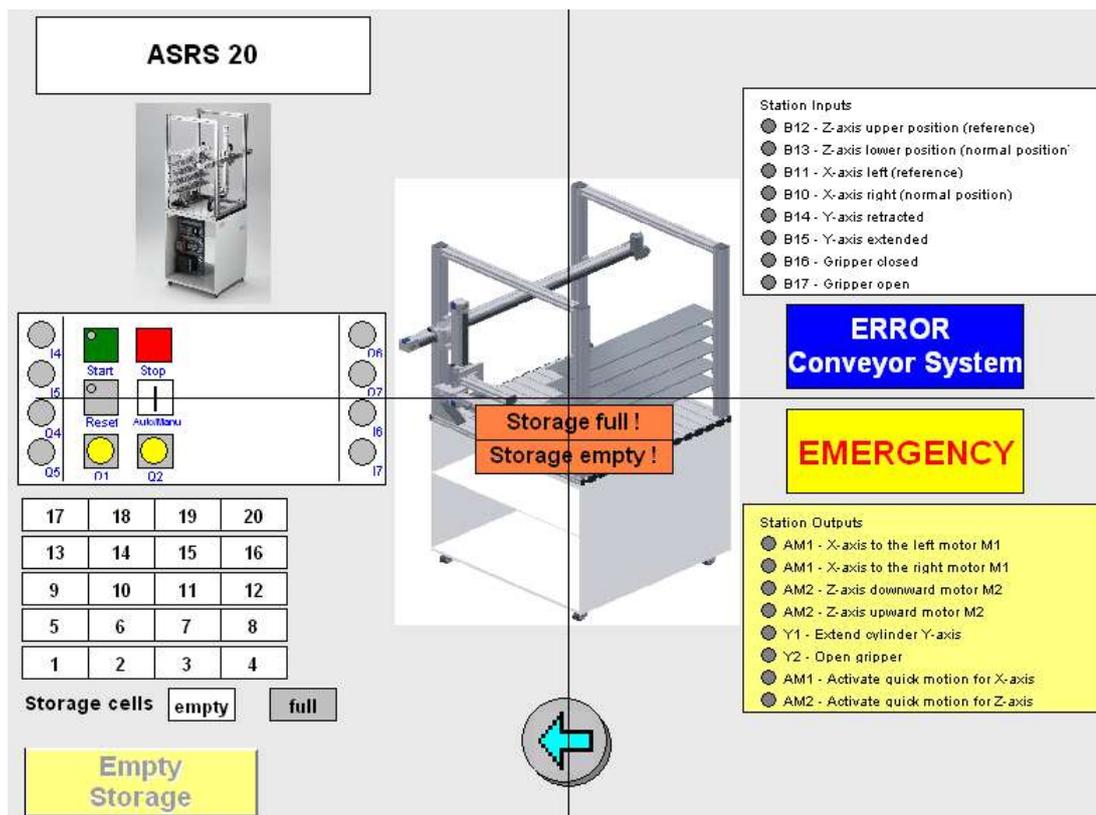


Figura 107: Pantalla de la estación de almacén (actual)

Fuente: MPS500_WinCCV60_Application_V5_DP, año 2003

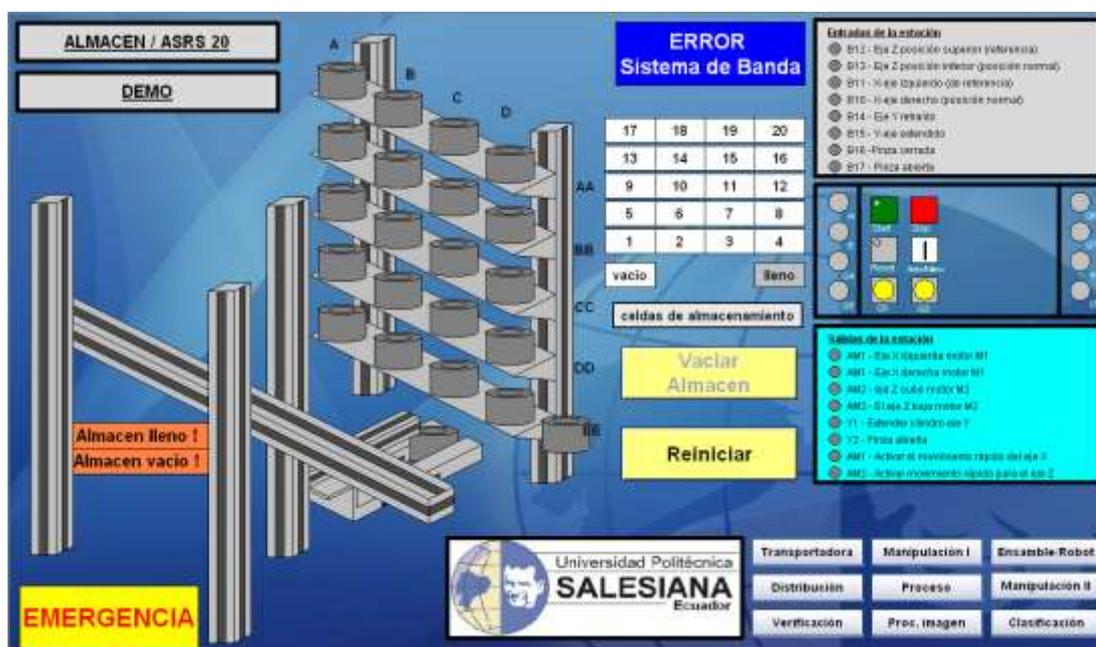


Figura 108: Pantalla de la estación de almacén demo (nueva)

Fuente: Los Autores, año 2013

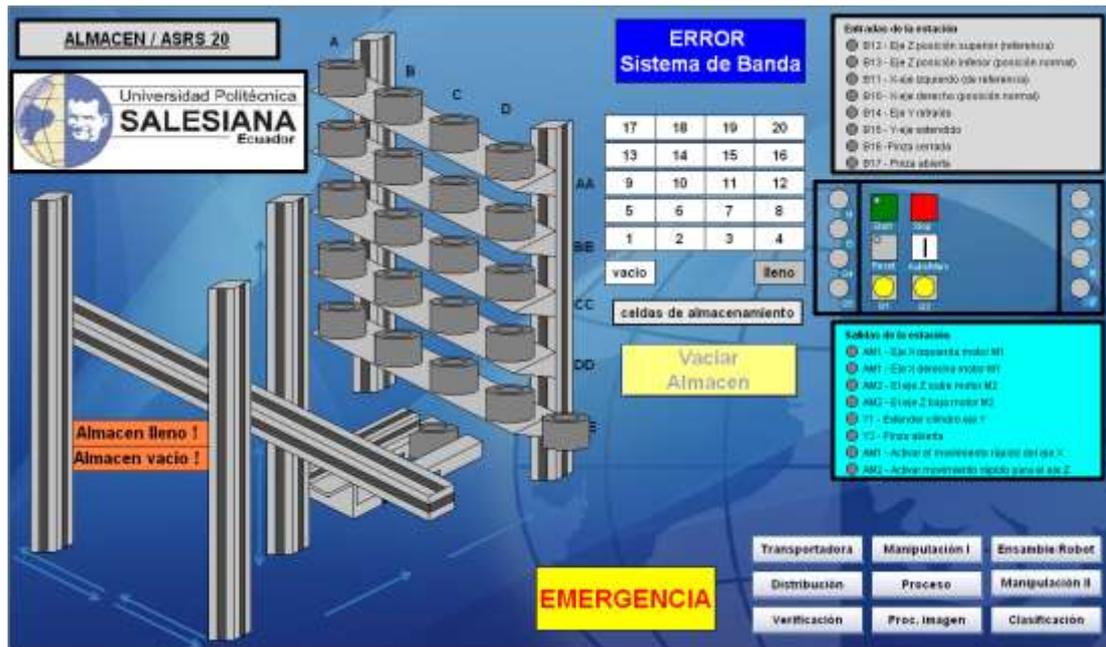


Figura 109: Pantalla de la estación de almacén (nueva)

Fuente: Los Autores, año 2013

4.12 Estación de manipulación II

De la pantalla actual de la estación de manipulación II Figura 110, se puede describir lo siguiente:

- Se encuentra una imagen de la estación y no se visualiza animación alguna cuando está en funcionamiento.
- Para simular el proceso se tienen círculos que cambian de color cuando alguna variable cambia de estado.
- Tablero de control para encender, apagar o reiniciar el proceso.
- Todos los textos se encuentran en el idioma inglés.
- Se observa botones para indicar que el proceso se encuentra en emergencia o falla.

De la pantalla nueva de la estación de manipulación II Figura 111, se puede resumir los siguientes cambios:

- Se coloca una imagen compuesta de soportes, brazo para la pinza de agarre de fichas.

- El brazo realiza movimientos en cuatro direcciones, estos son abajo, arriba, derecha e izquierda.
- Se puede visualizar la apertura y cierre de la pinza para simular el agarre de la ficha. Igualmente se observa aparecer y desaparecer la ficha según el proceso de esta estación.
- Los textos se podrán observar en idioma español.
- Se colocan botones para ingresar a cualquier estación (pantalla) desde este punto.
- Cuenta con un almacén de fichas rechazadas, contiene su propio contador de fichas.
- Se muestra la imagen de un tablero donde se podrá encender, apagar o reiniciar la estación desde pantalla.
- La animación es realiza con una acción en VBS.

La pantalla de manipulación II Figura 112, se realizaron las animaciones con cuadro de diálogo dinámico, a continuación se las describe:

- Se coloca una imagen compuesta de soportes, brazo para la pinza de agarre de fichas.
- El brazo realiza movimientos en 2 direcciones, estos son derecha e izquierda, tiene flechas indicando el movimiento que está realizando en 4 direcciones estas son derecha, izquierda, arriba y abajo.
- Los textos se puede observar en idioma español.
- Se colocan botones para poder navegar a cualquier estación (pantalla) desde este punto.
- Se muestra la imagen de un tablero donde se podrá encender, apagar o reiniciar la estación desde pantalla.

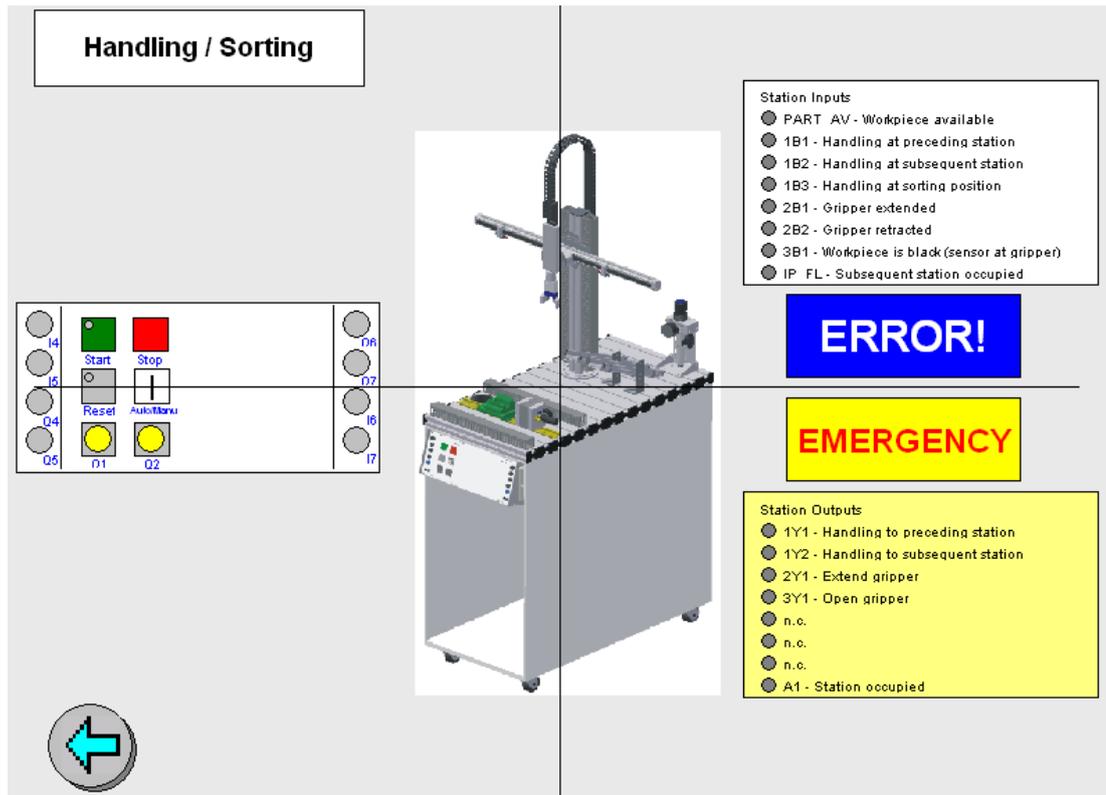


Figura 110: Pantalla de la estación de manipulación II (actual)
 Fuente: MPS500_WinCCV60_Application_V5_DP, año 2003

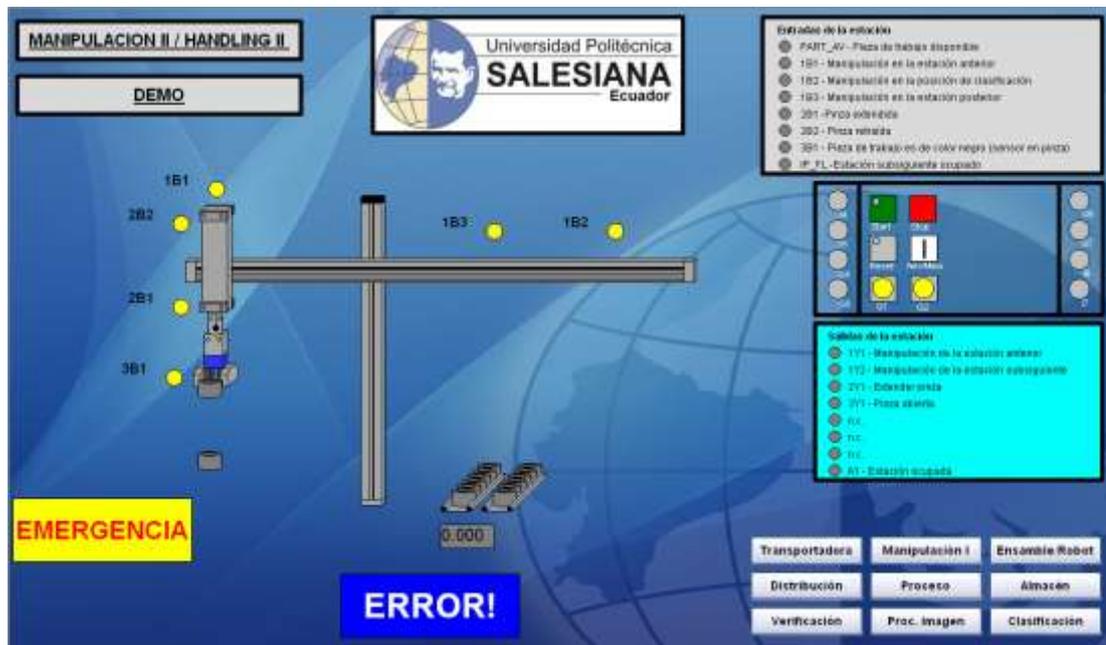


Figura 111: Pantalla de la estación de manipulación II demo (nueva)
 Fuente: Los Autores, año 2013

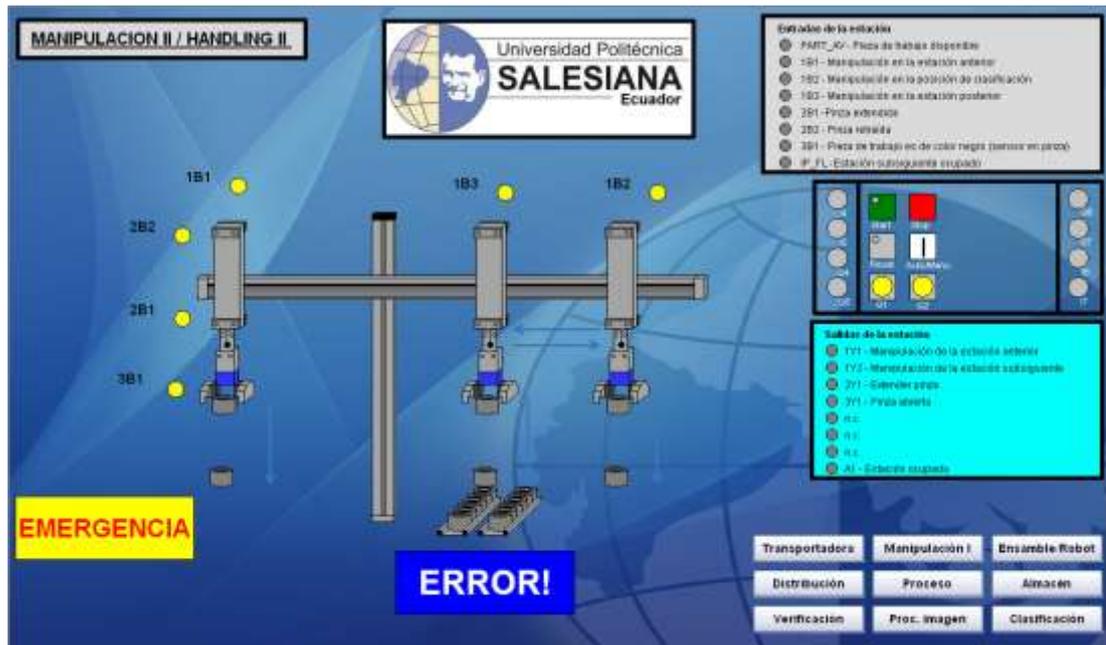


Figura 112: Pantalla de la estación de manipulación II (nueva)

Fuente: Los Autores, año 2013

4.13 Estación de clasificación

En la gráfica actual de la estación de clasificación Figura 113, se observa lo siguiente:

- Se observa una imagen de la estación sin animación.
- Se pueden visualizar los textos en inglés.
- Para simular que está en funcionamiento dicha estación, se encuentran círculos que cambian de color cuando varían los estados de entradas/salidas.
- Se encuentra la imagen de un tablero con botones para el encendido, apagado y reinicio de la estación.

En la gráfica nueva de la estación de clasificación Figura 114, se pueden extraer los siguientes cambios:

- Se coloca una imagen que contiene espacios para la clasificación de fichas rojas, negras y plateadas. Se incrementan las gráficas de sensores ópticos y paletas para la guiar cada ficha.

- Se colocan contadores de ficha para cada color y dependiendo de la cantidad de fichas van apareciendo en las resbaladeras.
- Se introduce texto para indicar que las resbaladeras (espacios de almacenamiento), se encuentran llenas.
- Se colocan los textos y mensajes en el idioma español.
- Se pueden observar los botones para ingresar a las diferentes pantallas desde la estación actual.
- Encontramos contadores totalizadores para las piezas clasificadas.
- Para poder reiniciar el contador de las resbaladeras se tiene que ingresar usuario y clave.

La pantalla de la estación de distribución Figura 115, se realizaron las animaciones con cuadro de diálogo dinámico, a continuación se las describe:

- Cuando arranca la banda de la estación se visualizan flechas indicando la dirección de arranque.
- Los textos se podrán observar en idioma español.
- Se colocan botones para poder navegar a cualquier estación (pantalla) desde este punto.
- Se muestra la imagen de un tablero donde se podrá encender, apagar o reiniciar la estación desde pantalla.

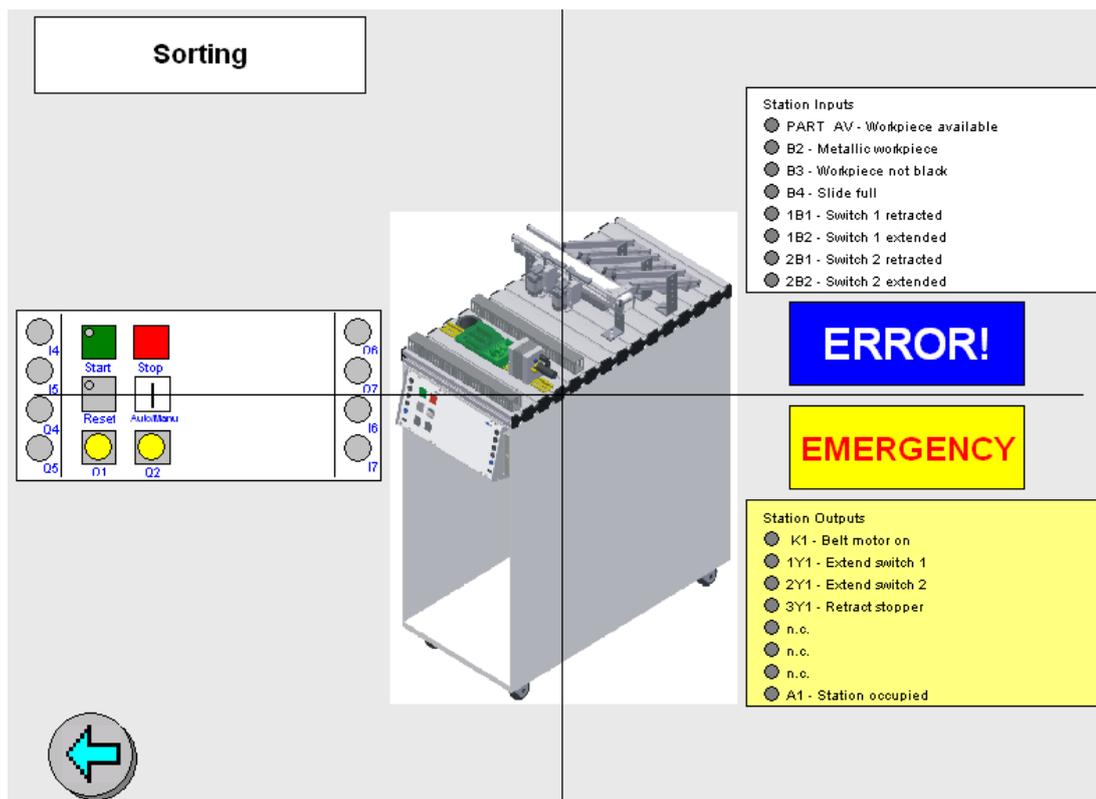


Figura 113: Pantalla de estación de clasificación (actual)

Fuente: MPS500_WinCCV60_Application_V5_DP, año 2003

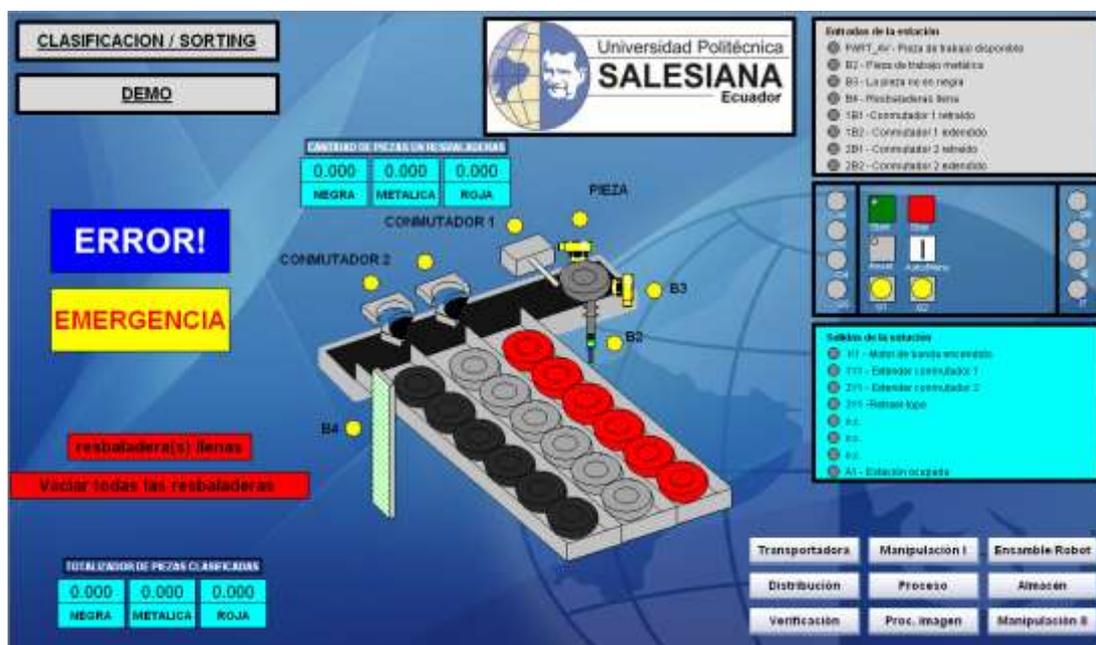


Figura 114: Pantalla de estación de clasificación demo (nueva)

Fuente: Los Autores, año 2013

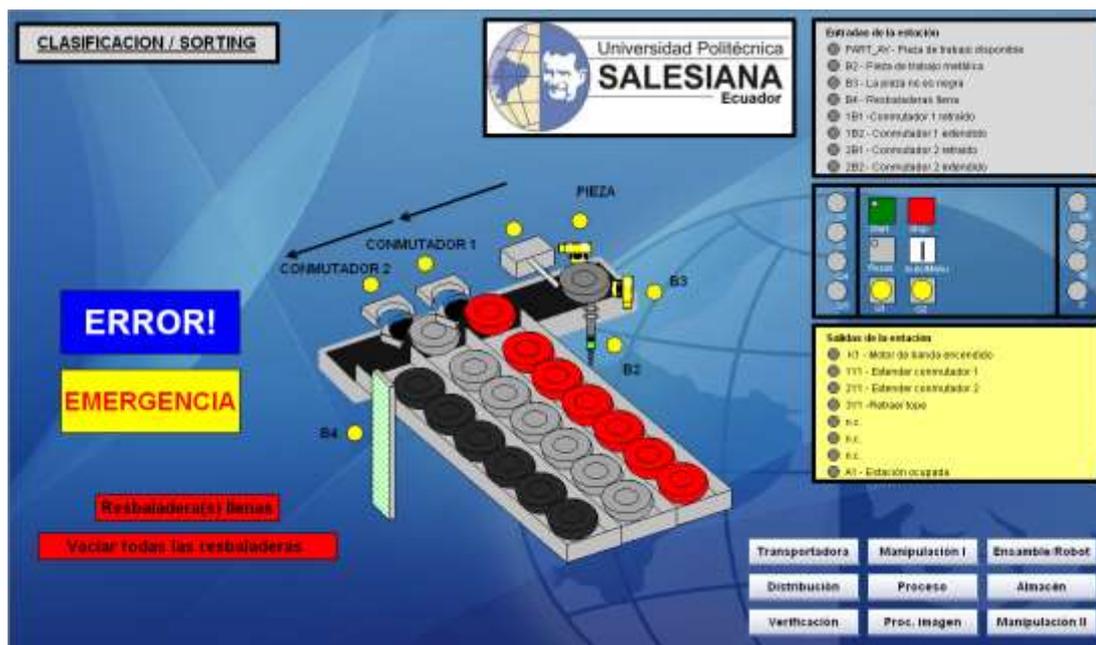


Figura 115: Pantalla de estación de clasificación (nueva)

Fuente: Los Autores, año 2013

CRONOGRAMA DE ACTIVIDADES

ACTIVIDADES PLANIFICADAS		TIEMPOS DE EJECUCIÓN
	Revisión de la red profibus.	JULIO 2012
	Avance teórico sobre redes.	
Módulo 1 Distribución	Análisis de los sistemas neumáticos, eléctricos y electro-neumáticos.	AGOSTO 2012
	Análisis, clasificación de las variables a leer y tratar.	
	Diseño de pantallas y lectura de variables	
	Avance teórico sobre el sistema a implementar.	
Módulo 2 Verificación	Análisis de los sistemas neumáticos, eléctricos y electro-neumáticos.	SEPTIEMBRE 2012
	Análisis, clasificación de las variables a leer y tratar.	
	Diseño de pantallas y lectura de variables.	
	Avance teórico sobre el sistema a implementar.	
Módulo 3 Proceso	Análisis de los sistemas neumáticos, eléctricos y electro-neumáticos.	OCTUBRE 2012
	Análisis, clasificación de las variables a leer y tratar.	
	Diseño de pantallas y lectura de variables.	
	Avance teórico sobre el sistema a implementar.	
Módulo 4 Manipulación I	Análisis de los sistemas neumáticos, eléctricos y electro-neumáticos.	NOVIEMBRE 2012
	Análisis, clasificación de las variables a leer y tratar.	
	Diseño de pantallas y lectura de variables.	
	Avance teórico sobre el sistema a implementar.	
Módulo 5 Visión	Análisis de los sistemas neumáticos, eléctricos y electro-neumáticos.	DICIEMBRE 2012
	Análisis, clasificación de las variables a leer y tratar.	
	Diseño de pantallas y lectura de variables.	
	Avance teórico sobre el sistema a implementar.	
Módulo 6 Robot y Ensamblaje	Análisis de los sistemas neumáticos, eléctricos y electro-neumáticos.	ENERO 2013
	Análisis, clasificación de las variables a leer y tratar.	
	Diseño de pantallas y lectura de variables.	
	Avance teórico sobre el sistema a implementar.	

Módulo 7 ASRS (Almacén)	Análisis de los sistemas neumáticos, eléctricos y electro-neumáticos.	FEBRERO 2013
	Análisis, clasificación de las variables a leer y tratar.	
	Diseño de pantallas y lectura de variables.	
	Avance teórico sobre el sistema a implementar.	
Módulo 8 Manipulación II	Análisis de los sistemas neumáticos, eléctricos y electro-neumáticos.	MARZO 2013
	Análisis, clasificación de las variables a leer y tratar.	
	Diseño de pantallas y lectura de variables.	
	Avance teórico sobre el sistema a implementar.	
Módulo 9 Clasificación	Análisis de los sistemas neumáticos, eléctricos y electro-neumáticos.	ABRIL 2013
	Análisis, clasificación de las variables a leer y tratar.	
	Diseño de pantallas y lectura de variables.	
	Avance teórico sobre el sistema a implementar.	
Módulo 10 Sistema de Transporte	Análisis de los sistemas neumáticos, eléctricos, electro-neumáticos.	MAYO 2013
	Análisis, clasificación de las variables a leer y tratar.	
	Diseño de pantallas y lectura de variables.	
	Avance teórico sobre el sistema a implementar.	
	Prueba de funcionamiento en red.	JUNIO 2013
	Elaboración de guía de usuario.	JULIO 2013

Tabla 6: Cronograma de actividades

Fuente: Los Autores, año 2012

PRESUPUESTO

DESCRIPCIÓN	UNIDAD	CANTIDAD	VALOR UNIT.	VALOR T.
Horas de configuración, dibujo, programación y pruebas	h	480	\$ 5,00	\$ 2.400,00
Movilización	Glb	1	\$ 120,00	\$ 120,00
Alimentación	Glb	1	\$ 240,00	\$ 240,00
Consumo Eléctrico (horas de programación y dibujo)	Glb	1	\$ 20,00	\$ 20,00
TOTAL				\$ 2.780,00

Tabla 7: Presupuesto

Fuente: Los Autores, año 2012

CONCLUSIONES

- Este proyecto ayudará a los estudiantes que se encuentran cursando los ciclos superiores, y en la cátedra como Robótica, Automatización Industrial e Informática Industrial por motivo de que se trata de utilizar varias herramientas que proporciona el software y el estudiante puede revisar la aplicación para que visualice como se lo ha realizado.
- La aplicación realizada tiene diferentes maneras de efectuar la supervisión de un proceso y ejecutar con distintas acciones un evento, utilizando la mayoría de las herramientas que presta el software.
- Son pocas las señales adicionales que se agregaron al antiguo sistema por motivo de que el software WinCC trabaja con licencias que dependen de la cantidad de variables externas.
- El tiempo de ejecución de la acción realizada es de 1 segundo por motivo de que si se le agrega un tiempo menor habrá inconvenientes de tiempos de ejecución y si se le coloca el tiempo estándar (2 segundos) algunas señales no van hacer visualizadas por motivo de que hay señales que cambian de estado rápidamente.
- Cualquier sistema SCADA tiene que dar facilidad de interacción entre usuario y máquina, por motivo las pantallas tiene animaciones que son aplicadas en cualquier área industrial y son de fácil manejo para el beneficiario.

RECOMENDACIONES

- Al momento de realizar un control con la aplicación SCADA evitar realizar lazos de control, pues la aplicación tiene que dar la posibilidad de establecerse y corregirse por sí mismo después de una mala ejecución en la siguiente actualización de datos (*trigger*).
- En el área industrial no es recomendable que el SCADA ejecute demasiados controles para su visualización por motivo de tiempos de ejecución, para eso utiliza un controlador para que efectúe ese trabajo en tiempos muy rápidos. Para métodos didácticos no hay inconvenientes en realizar estas ejecuciones.
- Leer el manual de usuario antes de encender el equipo y de interactuar con la aplicación.

REFERENCIAS BIBLIOGRÁFICAS

A. Rosado, Redes de comunicación industriales, 22/04/2003,
http://www.uv.es/rosado/courses/sid/Capitulo3_rev0.pdf

AlmaguelRaúl, Automatización, 18/05/2012,
<http://www.ecured.cu/index.php/Automatizaci%C3%B3n>.

Antonio Barragan, Bus de sensores y actuadores AS-Interface,
<http://www.uhu.es/antonio.barragan/book/export/html/125>.

Ariel Paz e Silva, Topología de red, 5/02/2008,
<http://www.monografias.com/trabajos53/topologias-red/topologias-red2.shtml>.

Bus de campo, http://es.wikipedia.org/wiki/Bus_de_campo.

Camozzi, Sensores magnéticos de proximidad, 25/05/2009,
<http://www.esperia.es/modulos/usuariosFtp/conexion/archi434A.pdf>.

Carlos de Castro Lozano y Cristóbal Romero, Introducción a SCADA, 22/03/2001,
<http://www.uco.es/grupos/eatco/automatica/ihtm/descargar/scada.pdf>.

David Goma, Manual de aplicación de encoders 15/06/2006,
<http://www.westmexico.com.mx/pfd/dynapar/catalogos/4.Manual%20de%20Aplicacion%20de%20Encoders.pdf>.

ECKART, Wolfgang, e Intercon-Asia, Mechatronics Teachware FMS50 Conveyor MPS 2000, Instructors Edition, Festo Didactic GmbH & Co., D-73770 Denkendorf, 2004.

EBEL, Frank y PANY, Markus, D:MP - TD - SV - DE/EN, Festo Didactic GmbH & Co.KG, D-73770 Denkendorf, 2006.

EBEL, Frank y PANY, Markus, D:MP - TD - SP - DE/EN, Festo Didactic GmbH & Co.KG, D-73770 Denkendorf, 2006.

EBEL, Frank y PANY, Markus, D:MP - TD - SHPA - DE/EN, Festo Didactic GmbH & Co.KG, D-73770 Denkendorf, Germany, 2010.

EBEL, Frank y PANY, Markus, D:MP - TB - SB - DE/EN, Festo Didactic GmbH & Co.KG, D-73770 Denkendorf, 2006.

EBEL, Frank y PANY, Markus, D:MP - TD - SM - DE/EN, Festo Didactic GmbH & Co.KG, D-73770 Denkendorf, 2006.

EBEL, Frank y PANY, Markus, D:MP - TD - SSO - DE/EN, Festo Didactic GmbH & Co.KG, D-73770 Denkendorf, 2006.

ECKART, Wolfgang, y ADIRO Automatisierungstechnik GmbH, Mechatronics Teachware ASRS MPS 2000, Instructors Edition, Festo Didactic GmbH & Co., D-73770 Denkendorf, 2002.

Edwuis Romero, Redes de comunicaciones industriales,
<http://neutron.ing.ucv.ve/revista-e/No4/RCI.html>.

Faberflorez, Sensores de posición,
<http://www.slideshare.net/faberflorez/presentacion-sensores-11897714>.

FERNÁNDEZ, Amador, *Sensores magnéticos e inductivos*, Monografía Universidad
Autómata del Estado de Hidalgo

2005,<http://www.uaeh.edu.mx/docencia/Tesis/icbi/licenciatura/documentos/Sensores%20magneticos.pdf>.

Henry Mendiburu Díaz, Sistema SCADA, 30/09/2005,
<http://www.galeon.com/hamd/pdf/scada.pdf>.

Henry Mendoza, Redes y comunicaciones, 2/07/2008,
<http://www.monografias.com/trabajos58/redes-comunicaciones/redes-comunicaciones.shtml>.

http://www.profibus.com/fileadmin/media/wbt/en/wbt1/3_2_Uebertragungstechnologie.swf.

Introducción sobre redes,
<http://www.profesores.frc.utn.edu.ar/industrial/sistemasinteligentes/UT5/INTRODUCCI%C3%93N%20GENERAL.htm>.

LBA, ¿Qué es un encoder?, <http://www.lbaindustrial.com.mx/que-es-un-encoder/>.

Manufactura Flexible ,
http://www.virtual.unal.edu.co/cursos/ingenieria/mecatronica/docs_curso/Anexos/TUTORIALcnc/DOCUMENTOS/TEORIA/MANUFACTURA%20FLEXIBLE.pdf.

Matias Guerrero, Topología de redes, <http://html.rincondelvago.com/topologia-de-redes.html>.

Mendoza Dario, Definición de Sistemas Flexibles de Manufactura, 10/09/2010,
<http://es.scribd.com/doc/39010609/Definicion-de-Sistemas-Flexibles-de-Manufactura>.

Micro, Unidades FRL, 3/05/2001,
<http://www.microautomacion.com/catalogo/Tratamientodelaire.pdf>.

ORTIZ, Roberto, Simatic HMI, SWinCC, Siemens S.A, Alemania, 2003.

Presión, 19/ 06/2013, <http://es.wikipedia.org/wiki/Presi%C3%B3n>.

Profibus, 23 /05/2013, <http://es.wikipedia.org/wiki/Profibus>.

Que es profibus, <http://www.smar.com/espanol/profibus.asp>.

Real, Sensores, 23/02/2010, <http://cmapublic2.ihmc.us/rid=1H2B63T5G-1SLKJ1L-J52/Sensores%20fundamentos,%20tipos%20y%20caracter%C3%ADsticas.pdf>.

Red en anillo, http://es.wikipedia.org/wiki/Red_en_anillo.

Red en árbol, http://es.wikipedia.org/wiki/Topolog%C3%ADa_en_%C3%A1rbol.

Red Industrial, http://es.wikipedia.org/wiki/Red_industrial.

Redes industriales, 3/16/2012,
<http://www.smar.com/espanol/articulostecnicos/article.asp?id=142>.

SCADA, 24/02/2013, <http://es.wikipedia.org/wiki/SCADA>.

SCHOBER, MPS500_Manual_Vision.doc, Festo Didactic GmbH & Co.KG., D-73770 Denkendorf, 2010.

Siemens AG, AS-Interface, 2009, <http://www.automation.siemens.com/cd-static/material/info/e20001-a550-p305-v4-7800.pdf>.

sirgo@SUPRA, *Redes*, Universidad de Oviedo, 17/03/2004, <http://www.isa.uniovi.es/docencia/redes/Apuntes/tema9.pdf>.

SpiraxSarco, Filtro/Regulador, 31/08/2004, <http://www.spiraxsarco.com/mx/pdfs/TL/p054-01.pdf>.

Usuario, Redes de comunicación industriales, 23/10/2006, <http://linux0.unsl.edu.ar/~rvilla/c3m10/tema13.pdf>.

Verónica, Sistema SCADA, 31/08/2008, <http://dspace.epn.edu.ec/bitstream/15000/8546/4/T10905CAP3.pdf>.

Visual Basic: guía del estudiante, 8/02/2000, <http://www.monografias.com/trabajos2/guiavb/guiavb.shtml>.

Yaritzza, Ventajas y desventajas de tipos de topología, 23/02/2012, <http://construiryadministrarred35yaritzza.blogspot.com/2012/02/ventajas-y-desventajas-de-tipos-de.html>.

ANEXOS

MANUAL DE USUARIO

Para que el sistema funcione correctamente se necesita que todos los equipos se encuentren encendidos, se requiere suministro de aire comprimido, es decir encender el compresor y abrir la válvula de paso, verificar en todas los filtros reguladores una presión máxima de 6 bares de aire comprimido.

Colocar los pallets transportadores en la banda transportadora, arrancar el módulo transportador y reiniciar todos los módulos a encender.

Después de reiniciar los módulos y de interactuar con el sistema de supervisión, se procede a dar marcha al sistema teniendo en cuenta que se debe de encender un módulo de salida de piezas con un módulo de llegada de piezas, por motivo de que si la pieza no es retirada en la parte final de la banda, regresará al proceso dando un error en el sistema.

Al momento de encender el módulo del robot y ensamblaje colocar el programa P.0001 que es el programa Demo, color la llave en el modo Auto (Op), encender los servos y aplastar el botón *start*.

Para encender el torno primero colocar el interruptor en ON, hacer correr el programa EMCO WinNC, en modo JOG llevar al torno al punto de referencia y después regresar el torno a una posición alejada del plato de 3 mordazas por ejemplo posición $Z = + 150$. Colocar el torno en modo automático y verificar que el programa que se ejecute sea el 00020.

En el sistema de supervisión y control encontramos las siguientes ventanas para poder interactuar con el sistema.

A1 Pantalla de inicio

En esta pantalla podemos observar la fecha, hora y logramos navegar entre las diferentes pantalla haciendo clic en los siguientes botones.



Figura 116: Pantalla de inicio

Fuente: Los Autores, año 2013

EXIT está compuesto por una imagen, encima de la imagen tiene un botón y por medio de este botón podemos salir de la aplicación haciendo clic encima de la palabra EXIT, dando clic comienza a ejecutarse una acción realizada en C.

La acción a ejecutarse es la siguiente:

```
#include "apdefap.h"
void OnClick(char* lpszPictureName, char* lpszObjectName, char*
lpszPropertyName)
{
// WINCC:TAGNAME_SECTION_START
// syntax: #define TagNameInAction "DMTagName"
// next TagID : 1
// WINCC:TAGNAME_SECTION_END
```

```

// WINCC:PICNAME_SECTION_START
// syntax: #define PicNameInAction "PictureName"
// next PicID : 1
// WINCC:PICNAME_SECTION_END
{
DeactivateRTProject ();
}
}

```

Mensajes y Alarmas está compuesto por un botón y haciendo clic podemos navegar a la pantalla por medio de una conexión directa.

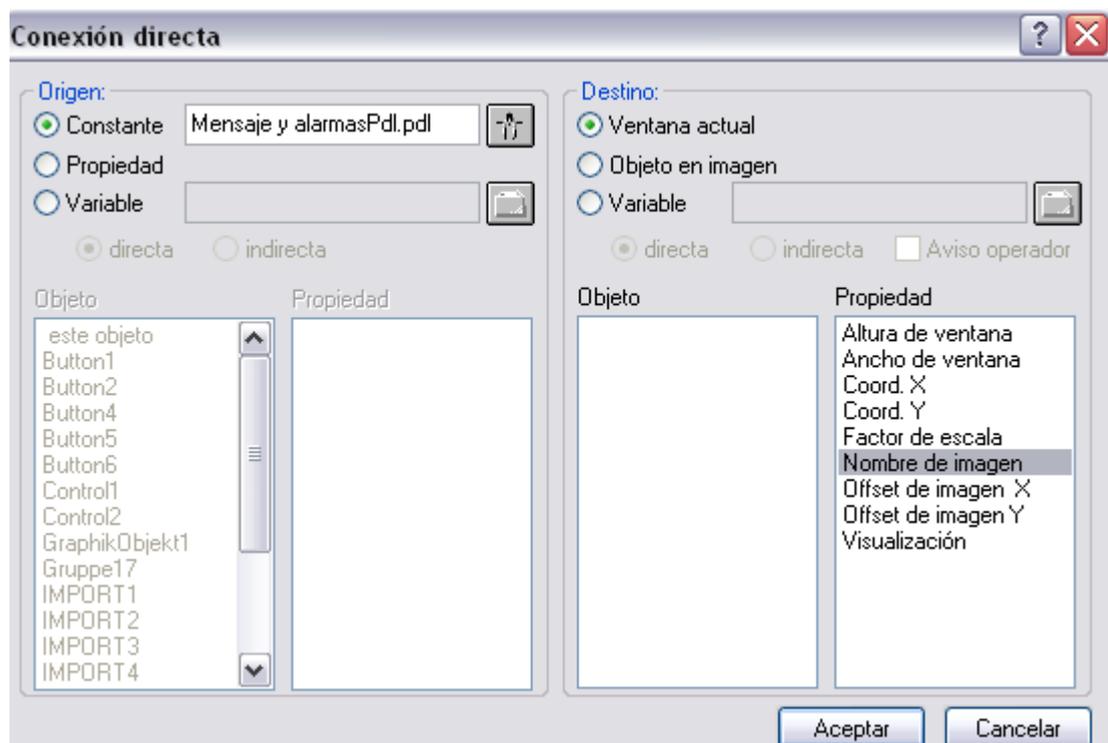


Figura 117: Conexión directa para navegar a la ventana de "mensajes y alarmas"

Fuente: Los Autores, año 2013

Red Profibus está compuesta por un botón y haciendo clic podemos navegar a la pantalla por medio de una conexión directa.

Comisionado está compuesto por un botón y haciendo clic podemos navegar a la pantalla por medio de una conexión directa.

Aplicación Demo está compuesta por un botón y haciendo clic podemos navegar a la pantalla por medio de una conexión directa.

Aplicación está compuesta por un botón y haciendo clic podemos navegar a la pantalla por medio de una conexión directa.

A2 Pantalla red profibus

En esta pantalla podemos observar si tenemos conexión o no conexión con los PLC's que conforma el MPS-500, si aparece una "X" significa que ese PLC no se está comunicando con la aplicación.

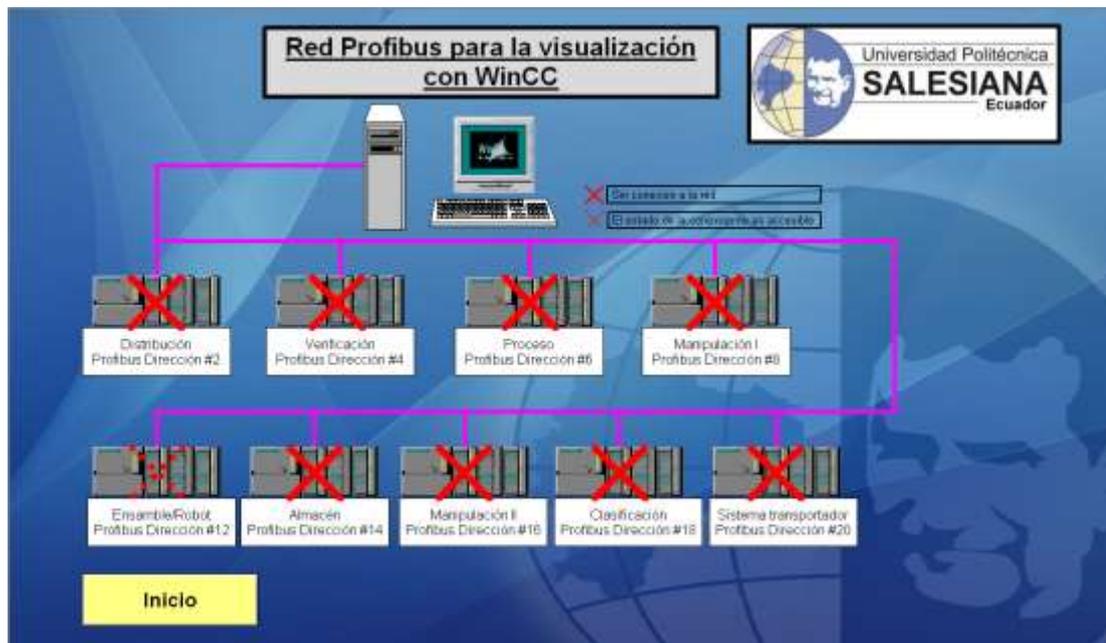


Figura 118: Pantalla de la red profibus

Fuente: Los Autores, año 2013

Inicio está compuesto por un botón y haciendo clic podemos navegar a la pantalla por medio de una conexión directa.

A3 Pantalla de comisionado

En esta pantalla podemos activar o desactivar las estaciones que se requieren, no se apagan ni se colocan las estaciones en posición *stop* o *reset* sino que el tope de cada unidad va dejar seguir el transportador sin dar opción a que se ejecute el funcionamiento de la estación del tope desactivado.

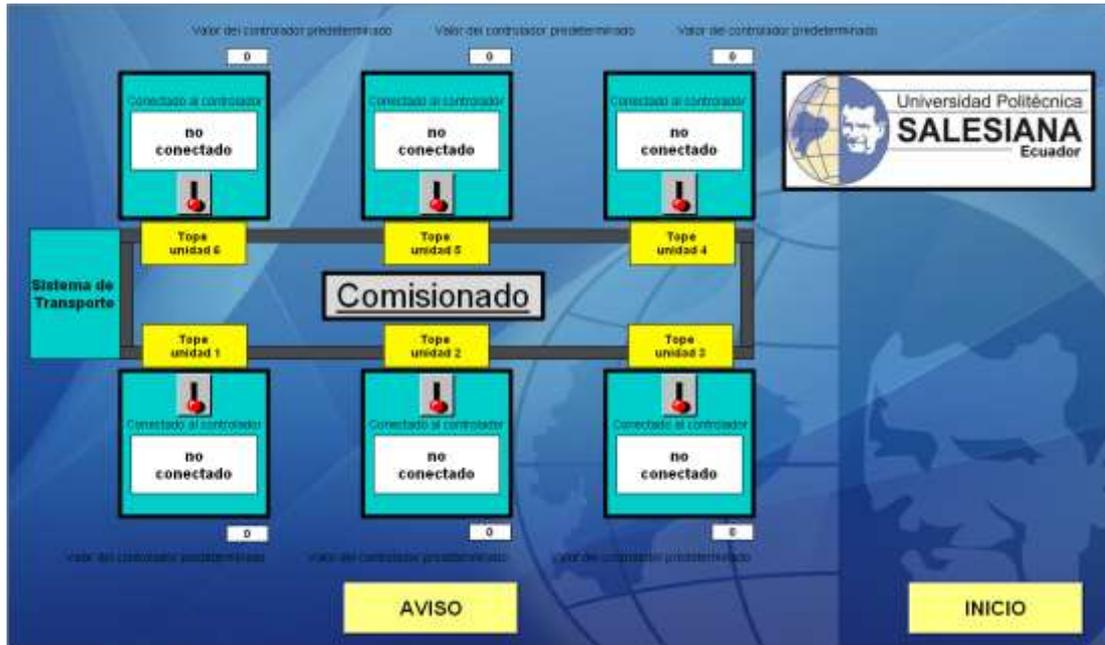


Figura 119: Pantalla de la ventana de comisionado

Fuente: Los Autores, año 2013

Aviso está compuesto por un botón, teniendo pulsado el botón izquierdo del mouse y con una acción en C podemos visualizar un mensaje sobre cómo está configurado el comisionado.

La acción a ejecutarse es la siguiente:

```
#include "apdefap.h"
```

```
void OnLButtonDown(char* lpszPictureName, char* lpszObjectName, char* lpszPropertyName, UINT nFlags, int x, int y)
```

```
{
```

```
// WINCC:TAGNAME_SECTION_START
```

```
// syntax: #define TagNameInAction "DMTagName"
```

```

// next TagID : 1
// WINCC:TAGNAME_SECTION_END
// WINCC:PICNAME_SECTION_START
// syntax: #define PicNameInAction "PictureName"
// next PicID : 1
// WINCC:PICNAME_SECTION_END
SetTagBit("notice1",1);    //Rückgabe-Typ :BOOL
}

```

Soltando el botón izquierdo del mouse y con una acción en C dejamos de visualizar el mensaje.

La acción a ejecutarse es la siguiente:

```

#include "apdefap.h"
void OnLButtonUp(char* lpszPictureName, char* lpszObjectName, char*
lpszPropertyName, UINT nFlags, int x, int y)
{
// WINCC:TAGNAME_SECTION_START
// syntax: #define TagNameInAction "DMTagName"
// next TagID : 1
// WINCC:TAGNAME_SECTION_END
// WINCC:PICNAME_SECTION_START
// syntax: #define PicNameInAction "PictureName"
// next PicID : 1
// WINCC:PICNAME_SECTION_END
SetTagBit("notice1",0);    //Rückgabe-Typ :BOOL
}

```

El mensaje que aparece al momento de dar clic en Aviso.

AVISO:

Primero: Valor preajustado del controlador

Controladores estándar son:

0 = No conectado

1 = Distribución y Verificación

2 = Manipulación y Procesamiento

3 = Visión

4 = Robot y Ensamblaje

5 = ASRS20

6 = Manipulación y Clasificación

Segundo: Activar / Desactivar controlador

**Posición del interruptor muestra el estado real
de la conexión del controlador:**

activado = verde

desactivado = rojo

Para activar / desactivar los controladores hacer clic con el ratón!

**Advertencia: Cada cambio tiene una influencia directa
en el sistema de transporte!**

Figura 120: Mensaje a mostrar en la pantalla

Fuente: Los Autores, año 2013

A4 Sistema transportador demo

Esta pantalla nos dirige a la aplicación realizada para trabajar con el programa Demo del PLC.

Esta pantalla se caracteriza por lo siguiente:

- Poder navegar a las diferentes pantallas y visualizar los estados del proceso de cada estación. Para navegar tenemos conexión directa y scripts realizados en C.
- Poder dar marcha o paro a la banda transportadora y visualizar su estado.

- Visualizar si los transportadores en cada estación se encuentra con pieza y se identifica el transportador que se encuentra en la estación con el número que tiene cada transportador.
- Navegar a la pantalla Inicio por medio de una conexión directa.
- Salir de *runtime* por medio de un script en C.
- Lograr ver la imagen de la estación pulsando el botón izquierdo del ratón y se desaparece la imagen soltando el botón izquierdo del ratón.



Figura 121: La pantalla del sistema transportador demo

Fuente: Los Autores, año 2013

El script para abrir la ventana de clasificación es el siguiente:

```
#include "apdefap.h"
void OnClick(char* lpszPictureName, char* lpszObjectName, char*
lpszPropertyName)
{
// WINCC:TAGNAME_SECTION_START
// syntax: #define TagNameInAction "DMTagName"
// next TagID : 1
// WINCC:TAGNAME_SECTION_END
```

```
// WINCC:PICNAME_SECTION_START
// syntax: #define PicNameInAction "PictureName"
// next PicID : 1
#define PIC_0 "clasificacion.Pdl"
// WINCC:PICNAME_SECTION_END
OpenPicture(PIC_0);
}
```

A5 Distribución demo

En esta pantalla se debe introducir la cantidad de fichas indicadas en el compartimiento, se va a visualizar que el contador va ir disminuyendo su cantidad y el compartimiento va a ir vaciándose poco a poco hasta llegar a cero. Luego muestra un mensaje de compartimiento vacío, además se visualiza el movimiento del brazo que sale con pieza de la estación distribución a la siguiente estación y regresa sin pieza a la estación distribución.

Se logra reiniciar, parar o arrancar el módulo desde WinCC, se va poder visualizar el estado de las entradas, las salidas, el mensaje de emergencia que se presenta al momento de pulsar la parada de emergencia y un icono indicando que hubo un error en el proceso.

Se consigue navegar a las diferentes pantallas del proceso por medio de conexiones directas y scripts realizados en C.

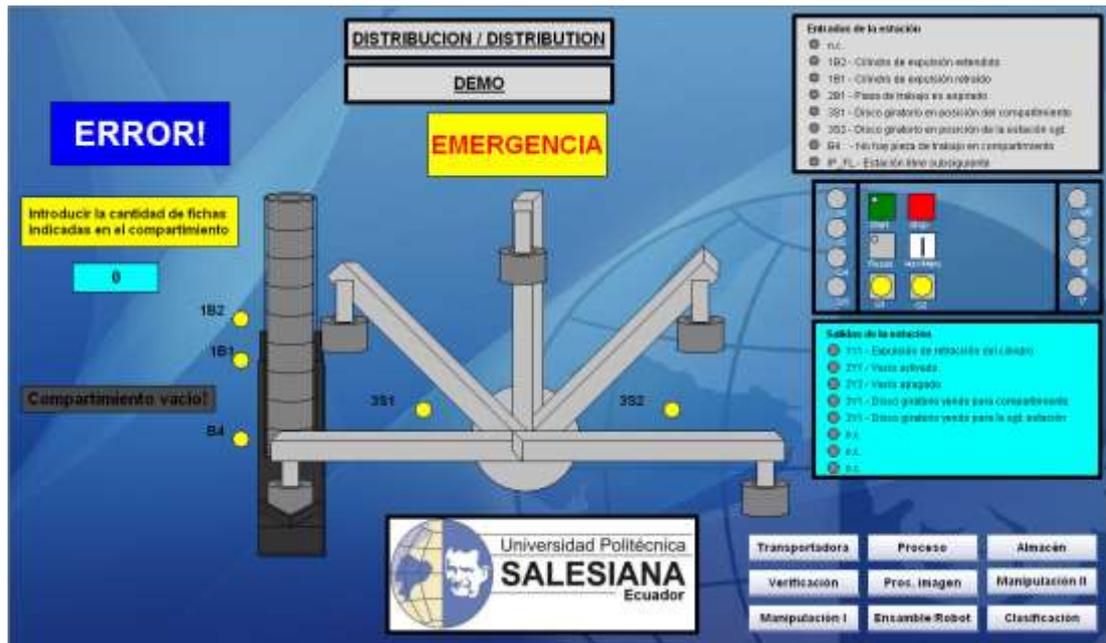


Figura 122: Pantalla de la ventana de distribución demo

Fuente: Los Autores, año 2013

El script para abrir la ventana de manipulación II es el siguiente:

```
#include "apdefap.h"
void  OnClick(char*  lpszPictureName,  char*  lpszObjectName,  char*
lpszPropertyName)
{
// WINCC:TAGNAME_SECTION_START
// syntax: #define TagNameInAction "DMTagName"
// next TagID : 1
// WINCC:TAGNAME_SECTION_END
// WINCC:PICNAME_SECTION_START
// syntax: #define PicNameInAction "PictureName"
// next PicID : 1
#define PIC_0 "manipulacionII.Pdl"
// WINCC:PICNAME_SECTION_END
OpenPicture(PIC_0);
}
```

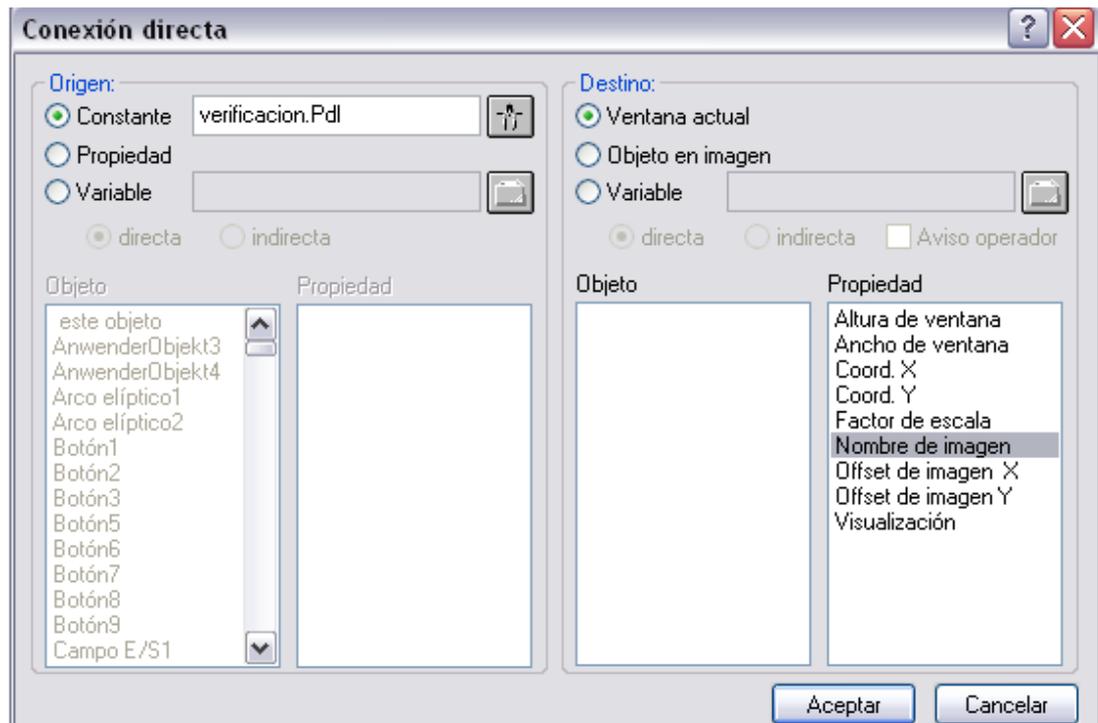


Figura 123: Conexión directa para navegar a la ventana de "verificación"

Fuente: Los Autores, año 2013

A6 Verificación demo

Esta pantalla nos da la oportunidad de visualizar la cantidad de piezas rechazadas. Lo realiza por medio de un contador que cuando llega a cinco da un mensaje que dice lo siguiente "resbaladera llena", en ese momento hay que vaciar la resbaladera y colocar el contador en cero.

Se logra reiniciar, parar o arrancar el módulo desde WinCC, se va poder visualizar el estado de las entradas, las salidas, el mensaje de emergencia que se presenta al momento de pulsar la parada de emergencia y un icono indicando que hubo un error en el proceso.

Se consigue navegar a las diferentes pantallas del proceso por medio de conexiones directas y scripts realizados en C.

Para modificar el valor del contador de piezas rechazadas se debe introducir el usuario y la clave. Primero realizar la combinación de teclas Ctrl + s.

Usuario: upsgye

Clave: upsgye

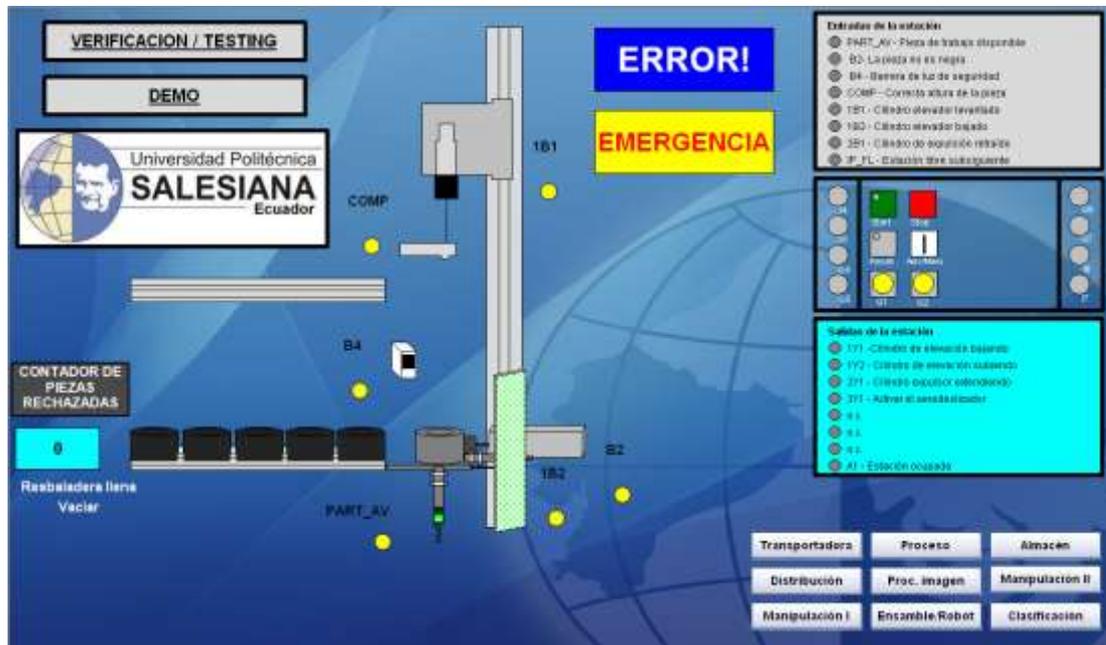


Figura 124: Pantalla de la ventana de verificación demo

Fuente: Los Autores, año 2013

El script para abrir la ventana de almacén ASRS20 es el siguiente:

```
#include "apdefap.h"
void  OnClick(char*  lpszPictureName,  char*  lpszObjectName,  char*
lpszPropertyName)
{
// WINCC:TAGNAME_SECTION_START
// syntax: #define TagNameInAction "DMTagname"
// next TagID : 1
// WINCC:TAGNAME_SECTION_END
// WINCC:PICNAME_SECTION_START
// syntax: #define PicNameInAction "PictureName"
// next PicID : 1
#define PIC_0 "almacenASRS20.Pdl"
// WINCC:PICNAME_SECTION_END
```

```
OpenPicture(PIC_0);  
}
```

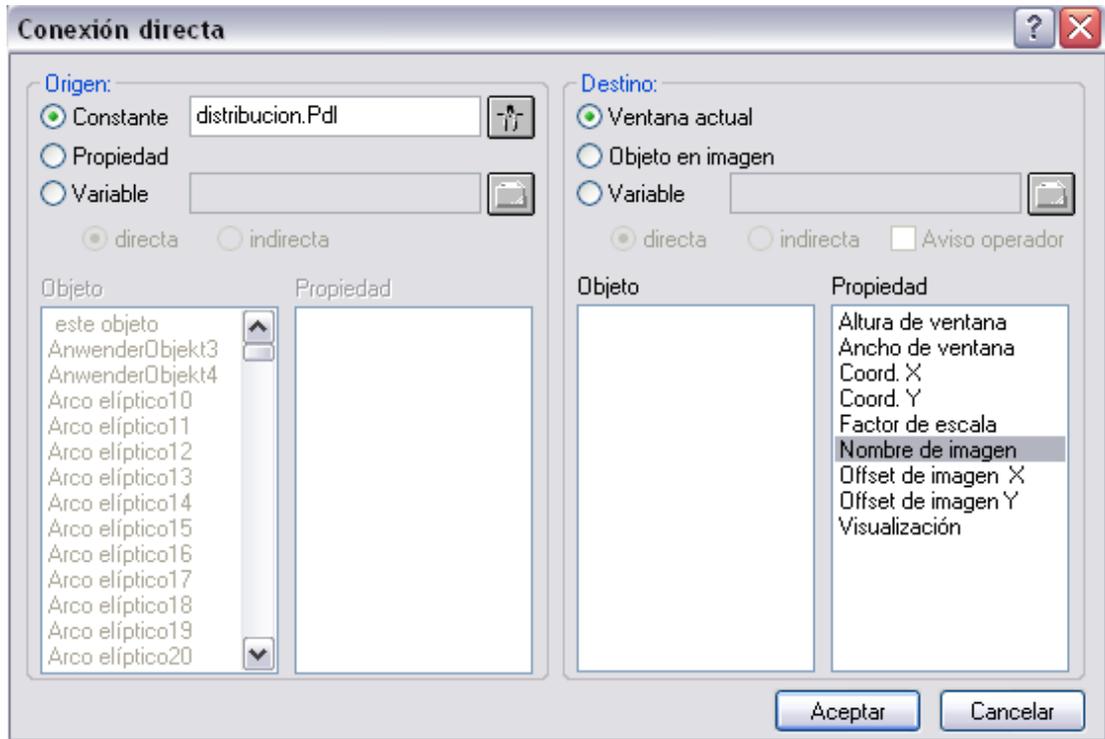


Figura 125: Conexión directa para navegar a la ventana de "distribución"

Fuente: Los Autores, año 2013

A7 Manipulación I demo

Se puede visualizar como sube, baja la pinza con o sin pieza y como se transporta de un lado a otro la pinza con o sin pieza.

Se logra reiniciar, parar o arrancar el módulo desde WinCC, se va poder visualizar el estado de las entradas, las salidas, el mensaje de emergencia que se presenta al momento de pulsar la parada de emergencia y un icono indicando que hubo un error en el proceso.

Se consigue navegar a las diferentes pantallas del proceso por medio de conexiones directas y scripts realizados en C.

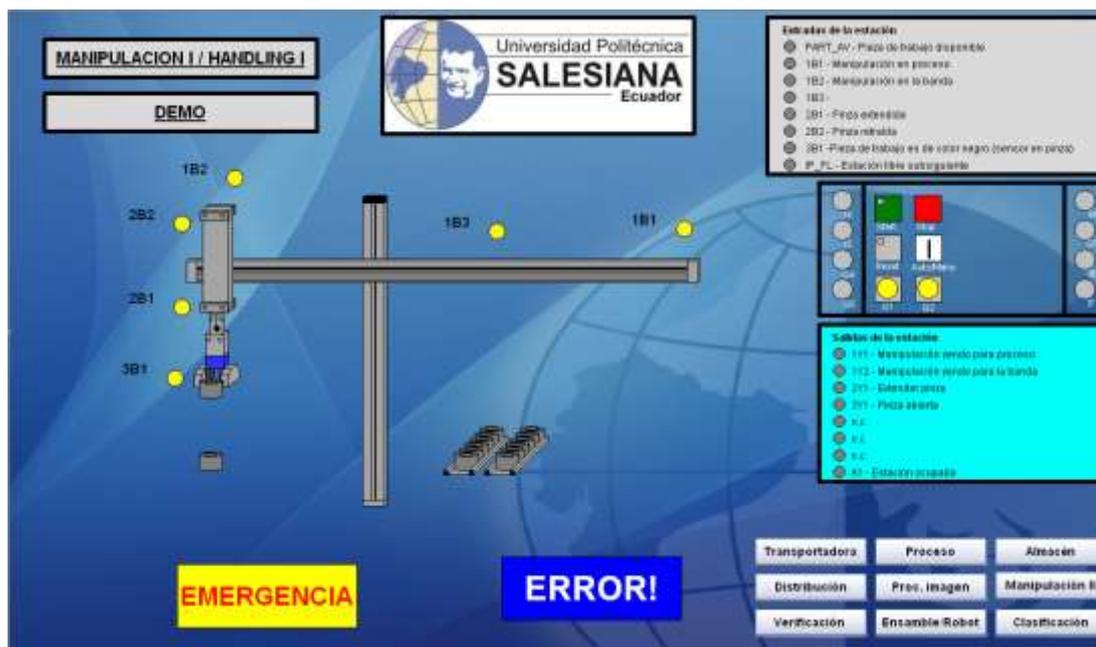


Figura 126: Pantalla de la ventana de manipulación I demo

Fuente: Los Autores, año 2013

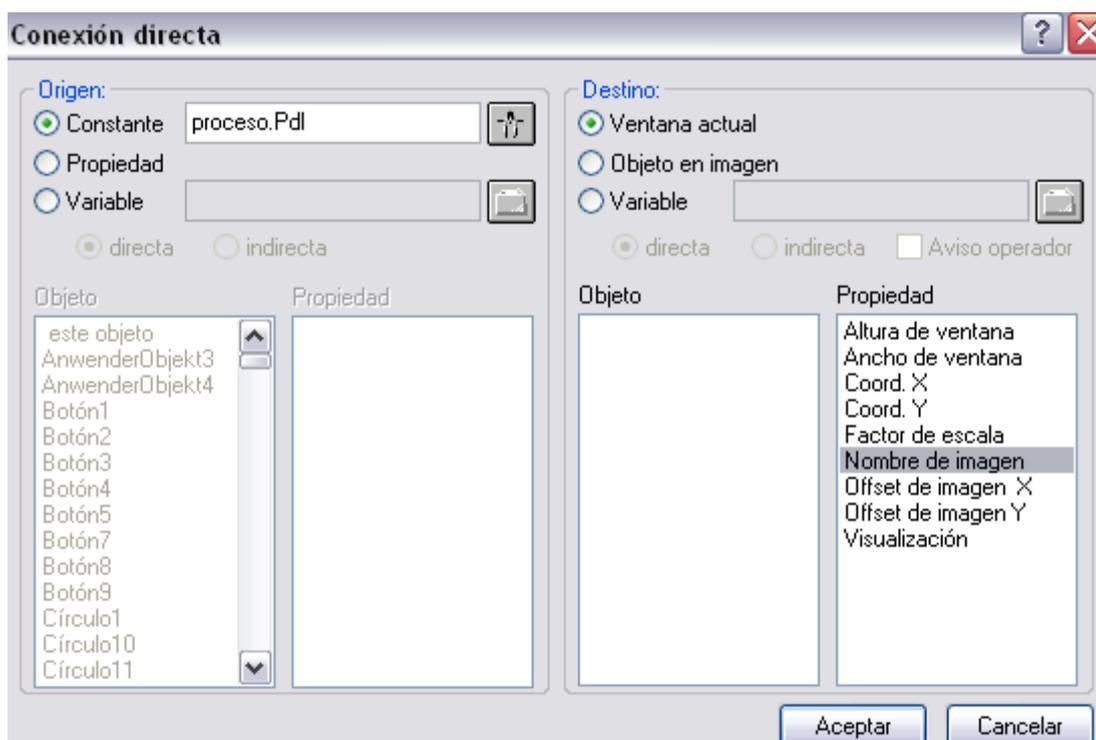


Figura 127: Conexión directa para navegar a la ventana de "proceso"

Fuente: Los Autores, año 2013

A8 Proceso demo

Se puede visualizar cuando la pieza está en la mesa, como se transporta hasta llegar a los puntos donde se comprueba el agujero, se perfora visualizando que los equipos bajen y suban.

Se logra reiniciar, parar o arrancar el módulo desde WinCC, se va poder visualizar el estado de las entradas, las salidas, el mensaje de emergencia que se presenta al momento de pulsar la parada de emergencia y un icono indicando que hubo un error en el proceso.

Se consigue navegar a las diferentes pantallas del proceso por medio de conexiones directas y scripts realizados en C.

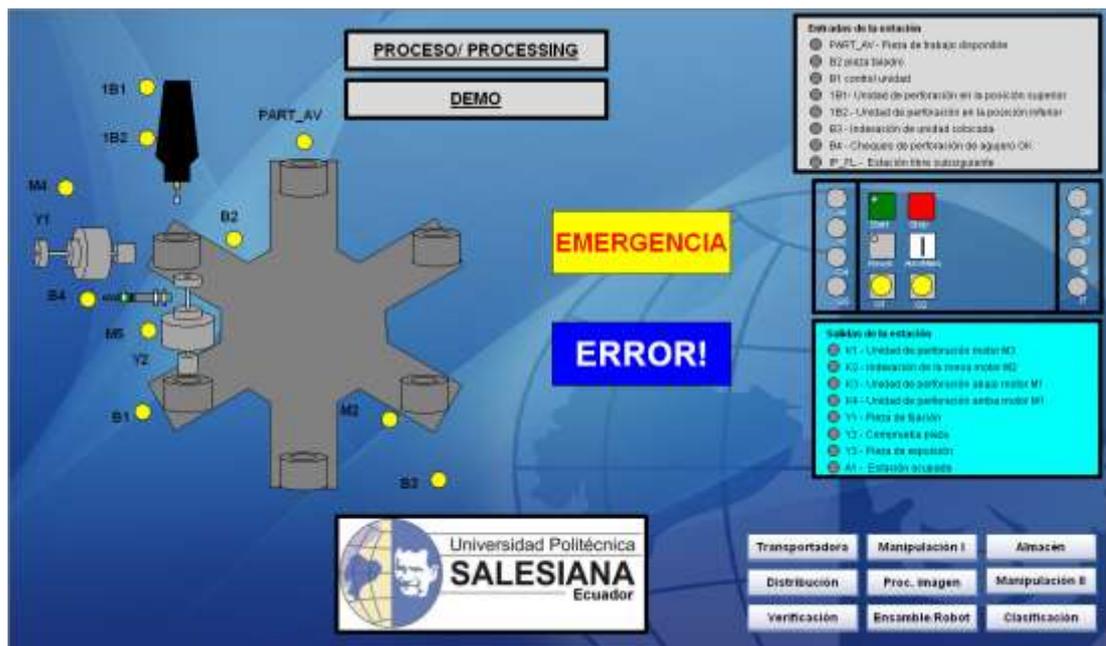


Figura 128: Pantalla de la ventana de proceso demo

Fuente: Los Autores, año 2013



Figura 129: Conexión directa por medio de la configuración de un botón

Fuente: Los Autores, año 2013

A9 Visión

Se puede visualizar si la pieza es aceptada o rechazada, si es rechazada va a mostrar una "X" en la imagen, si es aceptada va a mostrar un "✓". Encontramos un contador para contabilizar la cantidad de piezas rechazadas y aceptadas.

El transportador tiene una lámina reflectiva que sirve para indicar que no hay pieza y la cámara percibe una imagen en blanco. Se visualiza el estado de las entradas, las salidas y conseguir navegar a la pantalla de inicio por medio de conexión directa.

Para modificar el valor de las piezas aceptadas y rechazadas se debe introducir el usuario y la clave. Primero realizar la combinación de teclas Ctrl + s

Usuario: upsgye

Clave: upsgye

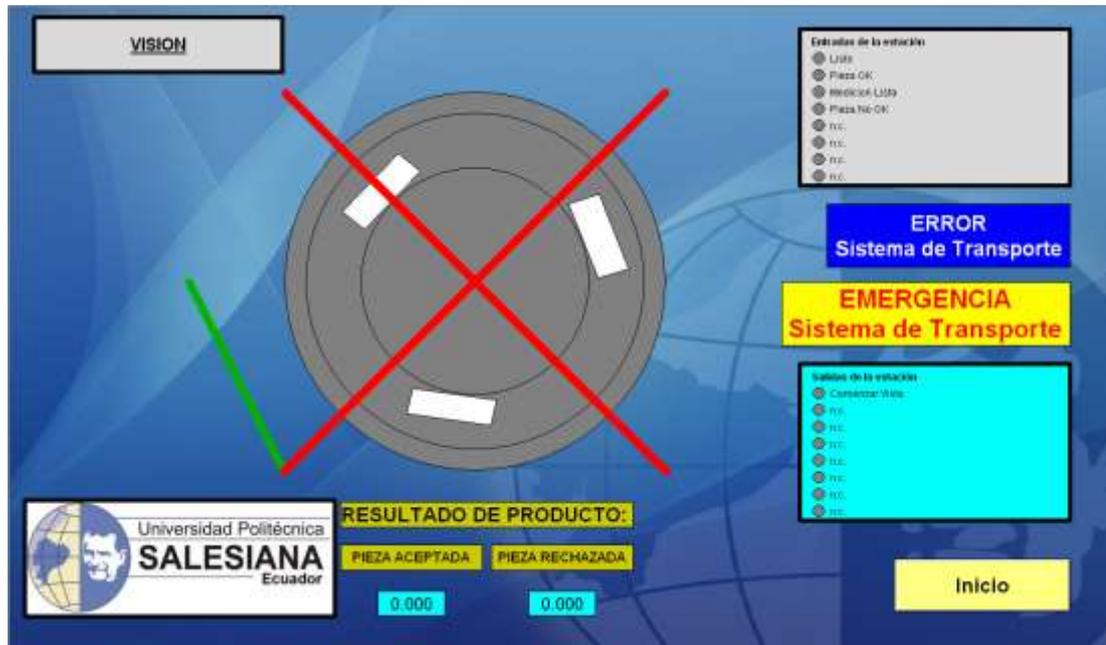


Figura 130: Pantalla de la ventana de visión

Fuente: Los Autores, año 2013

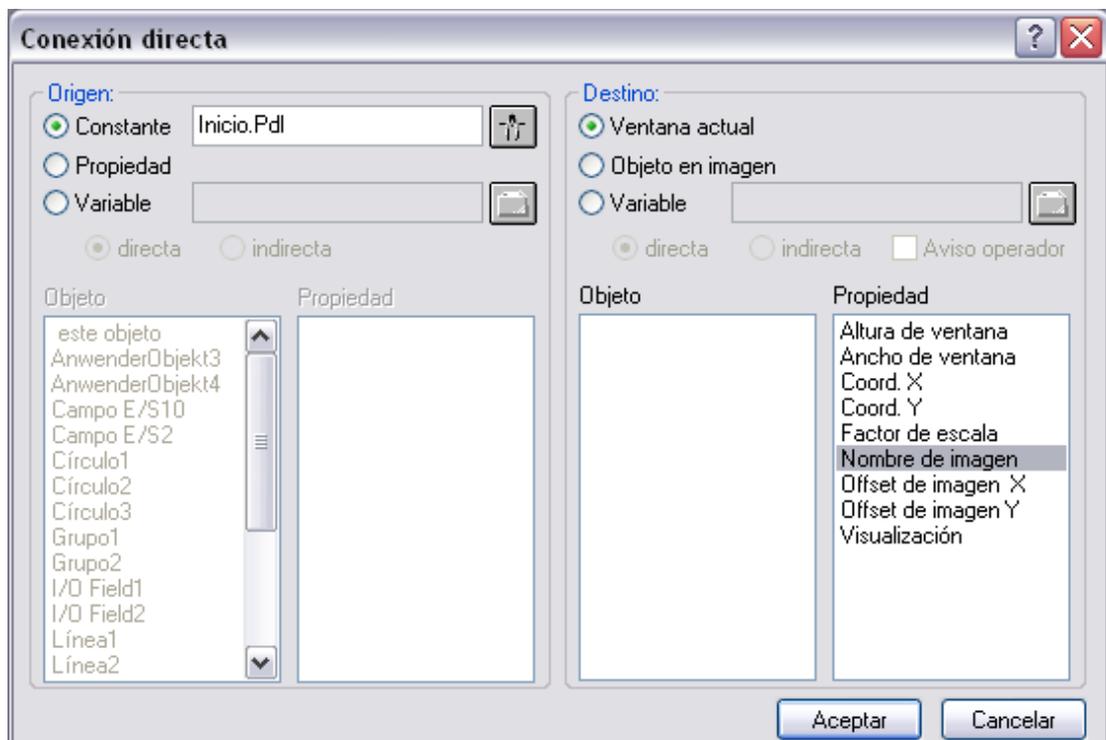


Figura 131: Conexión directa para navegar a la ventana de "inicio"

Fuente: Los Autores, año 2013

A10 Robot y ensamblaje

En esta pantalla se va a poder visualizar si el robot se encuentra trabajando o está en reposo, también se observa como el torno cierra y abre la protección contra virutas.

El plato de 3 mordazas se abre, se cierra para agarrar la pieza y el movimiento del carro se representa con unas flechas.

Se percibe el estado de las entradas, las salidas y conseguir navegar a la pantalla de inicio por medio de conexión directa.

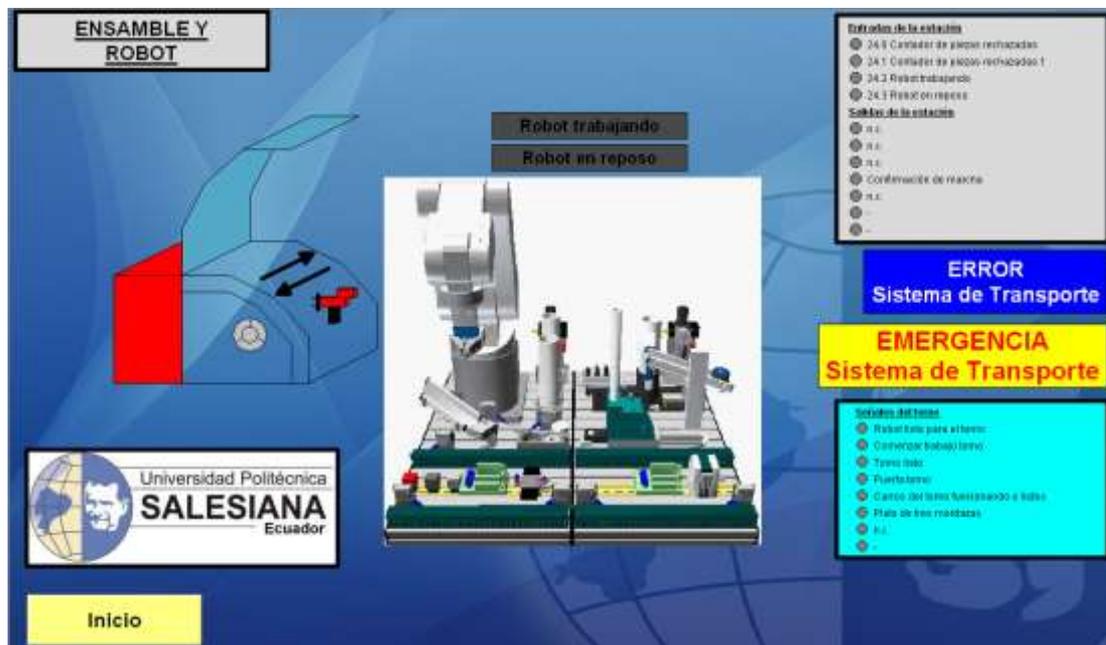


Figura 132: Pantalla de la ventana de robot y ensamblaje

Fuente: Los Autores, año 2013

A11 Almacén ASRS20 demo

En esta ventana vamos a poder visualizar el movimiento de la pinza yendo a dejar o desalojar fichas en las celdas de almacenamiento, también se observa el movimiento de las piezas con la pinza, como aparecen o desaparecen las fichas de las celdas de almacenamiento, todo esta acción realizada con Scripts en C.

Tenemos mensajes de almacén vacío o lleno, unas casillas que indican si las piezas entran o salen del almacén. Cuando se llena totalmente el almacén aparece un botón para poder vaciar el almacén por programa y tendrán que retirar las piezas de las celdas de almacenamiento. Se encuentra otro botón con acceso con clave para poder habilitar el botón, poder reiniciar el almacenamiento y el script que ejecuta el programa.

Se logra reiniciar, parar o arrancar el módulo desde WinCC, se va poder visualizar el estado de las entradas, las salidas, el mensaje de emergencia que se presenta al momento de pulsar la parada de emergencia y un icono indicando que hubo un error en el proceso.

Se consigue navegar a las diferentes pantallas del proceso por medio de conexiones directas y scripts realizados en C.

Nota: Realizar la operación de reinicio solo cuando sea necesario, cuando haya un apagón en el módulo de almacenamiento o el PLC se reinicie.

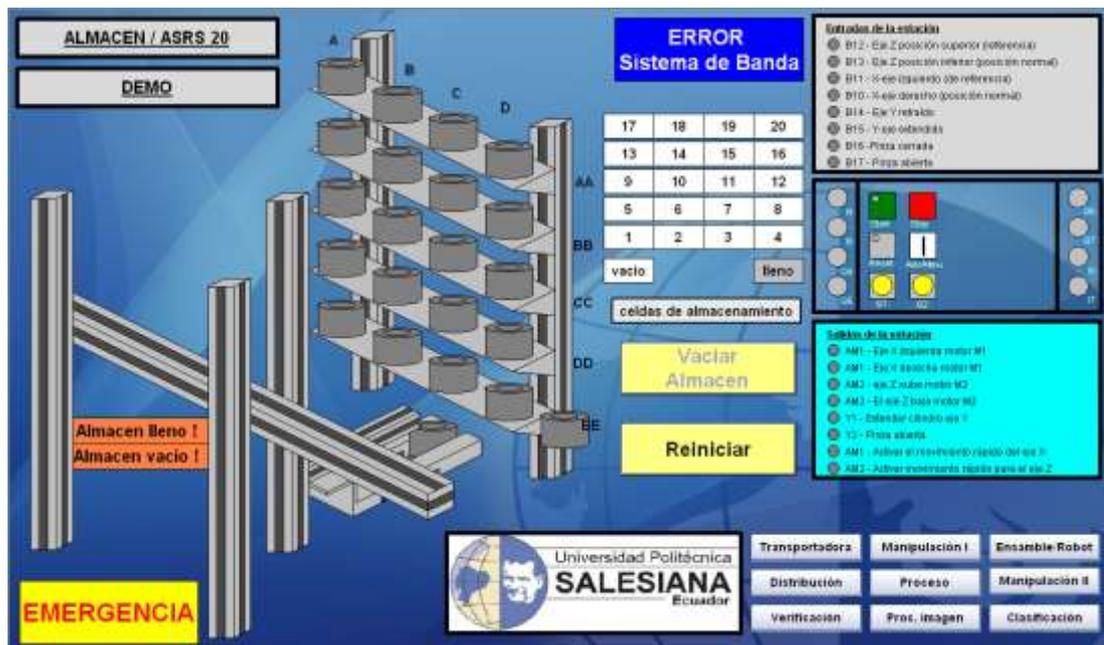


Figura 133: Pantalla de la ventana de almacén ASRS 20 demo

Fuente: Los Autores, año 2013

A12 Manipulación II demo

Se puede visualizar como sube o baja la pinza con pieza o sin pieza, y como se transporta de un lado a otro la pinza con o sin pieza. Cuando está lleno el módulo de clasificación, el módulo de manipulación envía las piezas a las resbaladeras y comienza a incrementar un contador.

Se logra reiniciar, parar o arrancar el módulo desde WinCC, se va poder visualizar el estado de las entradas, las salidas, el mensaje de emergencia que se presenta al momento de pulsar la parada de emergencia y un icono indicando que hubo un error en el proceso.

Se consigue navegar a las diferentes pantallas del proceso por medio de conexiones directas y scripts realizados en C.

Para modificar el valor de las piezas ubicadas en las resbaladeras se debe introducir el usuario y la clave. Primero realizar la combinación de teclas Ctrl + s.

Usuario: upsgye

Clave: upsgye

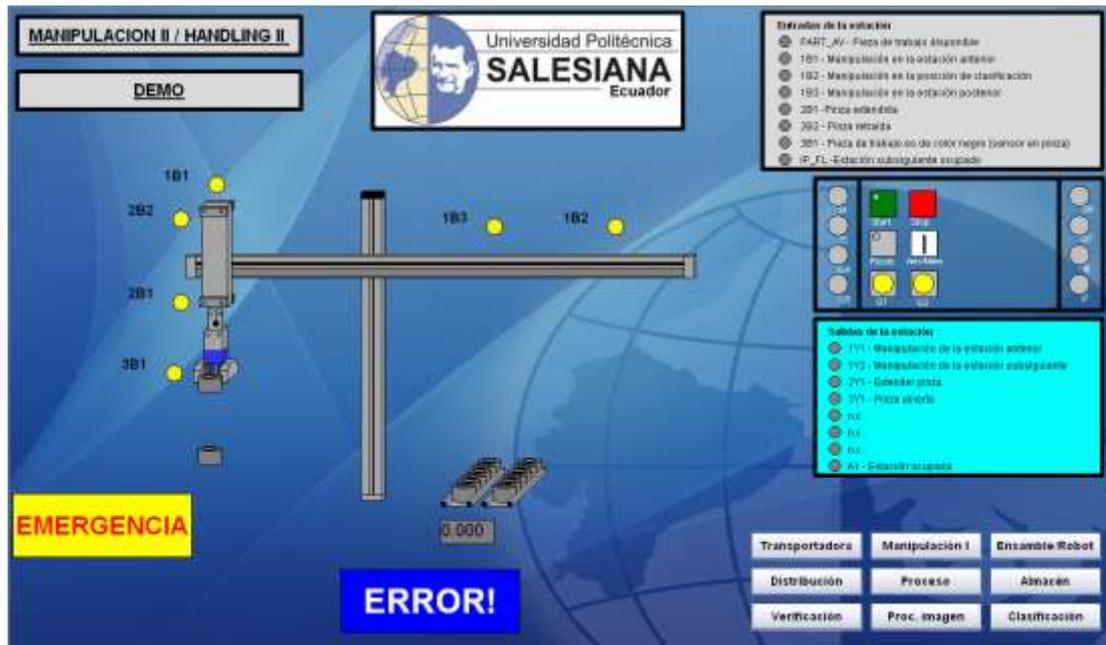


Figura 134: Pantalla de la ventana de manipulación II demo

Fuente: Los Autores, año 2013

A13 Clasificación demo

En esta estación podemos visualizar cuando llega la pieza, el sistema trata de identificar el color de la pieza. Una vez reconocido el color se recoge el pestillo que no deja avanzar la pieza y dependiendo del color se activan los conmutadores.

También se observa el avance de la pieza hasta la resbaladera que le corresponde, activando el contador de piezas que se encuentra en las resbaladeras y el contador de piezas procesadas.

Se visualiza mensajes de "resbaladeras llenas" y que se deben de "vaciar todas las resbaladeras". El contador de piezas para las resbaladeras tiene acceso restringido, para poder realizar alguna modificación se debe introducir el usuario y la clave, una vez realizada la modificación debe salir realizando *logout*. Primero realizar la combinación de teclas Ctrl + s.

Usuario: upsgye

Clave: upsgye

Se logra reiniciar, parar o arrancar el módulo desde WinCC, se va poder visualizar el estado de las entradas, las salidas, el mensaje de emergencia que se presenta al momento de pulsar la parada de emergencia y un icono indicando que hubo un error en el proceso.

Se consigue navegar a las diferentes pantallas del proceso por medio de conexiones directas y scripts realizados en C.

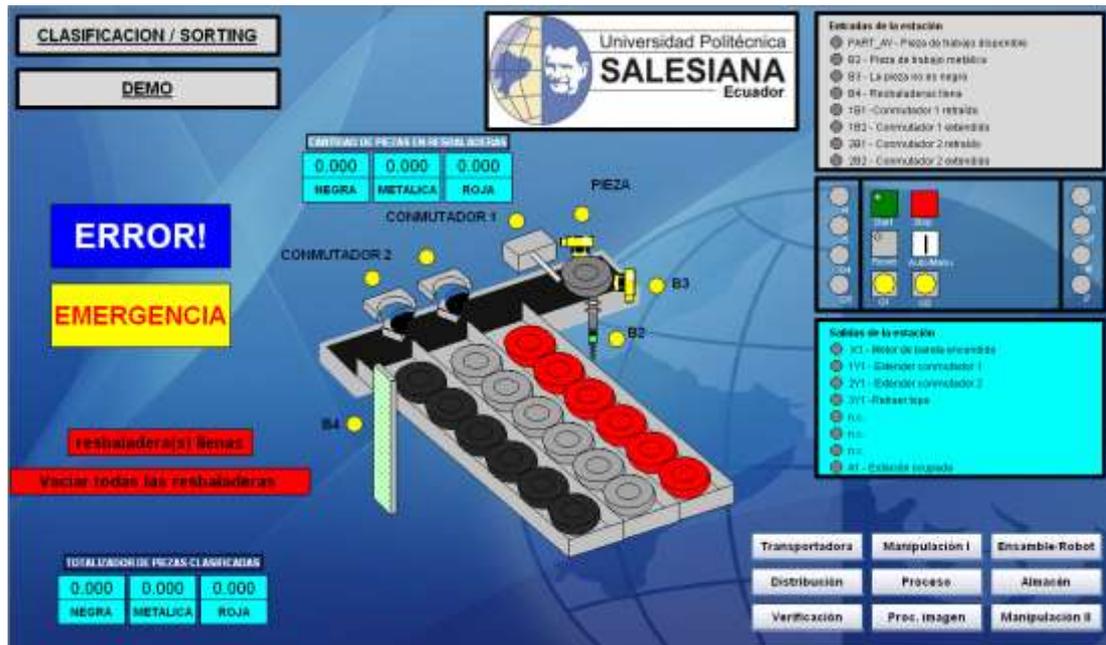


Figura 135: Pantalla de la ventana de clasificación demo

Fuente: Los Autores, año 2013

A14 Pantalla de mensajes y alarmas

En esta pantalla se logra visualizar los mensajes y alarmas seleccionados, cuando un valor deseado cambie de estado manda un aviso para que el operador tome una decisión o envía una alarma y el controlador toma la decisión de ejecutar una acción.

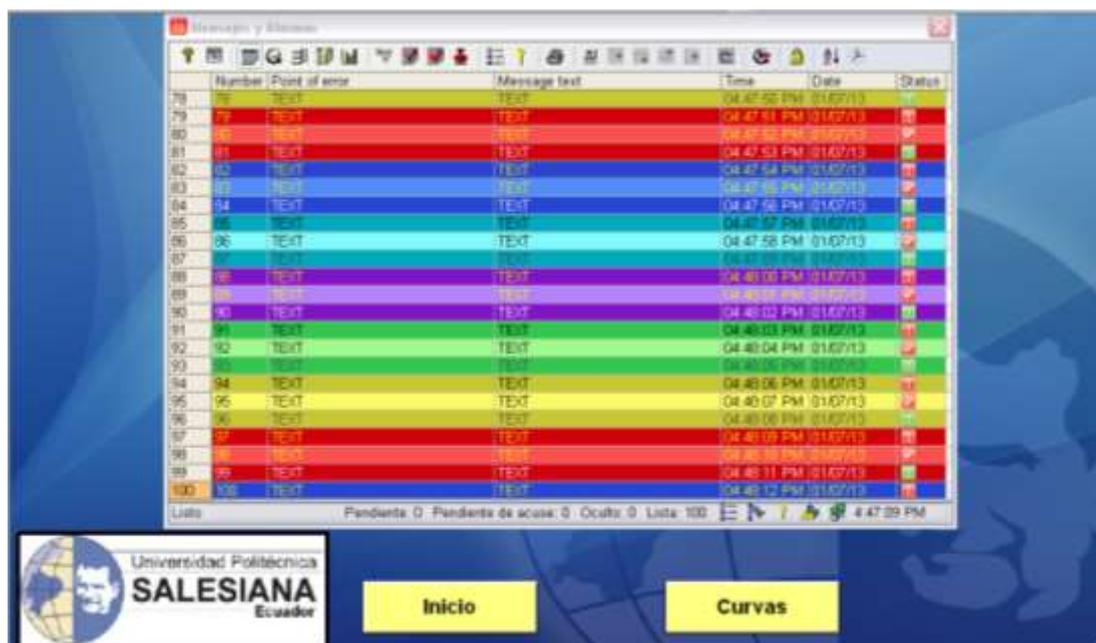


Figura 136: Pantalla de la ventana de mensajes y alarmas

Fuente: Los Autores, año 2013

A15 Curvas

En esta pantalla se logra visualizar las curvas de las piezas clasificadas. Por medio de 2 botones podemos ir a las pantallas de “Inicio” y de “Mensajes y Alarmas”.

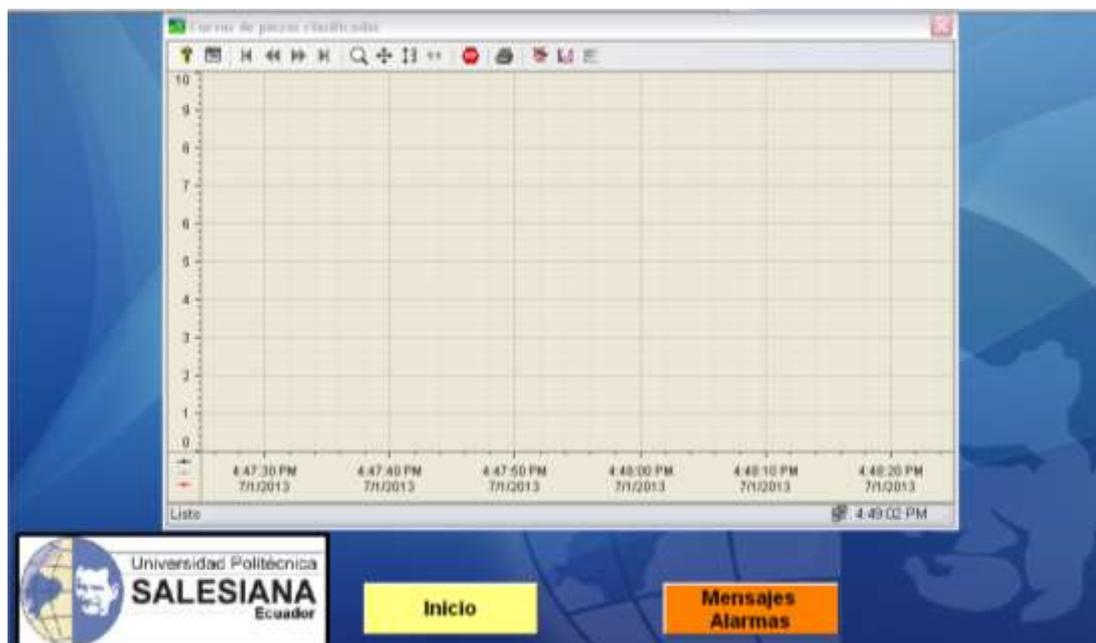


Figura 137: Pantalla de la ventana de curvas

Fuente: Los Autores, año 2013

A16 Banda transportadora

Esta pantalla nos dirige a la aplicación realizada para trabajar con cualquier programa que se ejecute en el PLC (si realiza cambios mecánicos en el sistema, tendrán que cambiar las ubicaciones de las piezas en las pantallas).

Esta pantalla se caracteriza por lo siguiente:

- Poder navegar a las diferentes pantallas y visualizar los estados del proceso de cada estación. Para navegar tenemos conexión directa y scripts realizados en C.
- Poder dar marcha o paro a la banda transportadora y visualizar su estado.
- Visualizar si los transportadores en cada estación se encuentra con pieza y se identifica el transportador que se encuentra en la estación con el número que tiene cada transportador.
- Navegar a la pantalla Inicio por medio de una conexión directa.
- Salir de *runtime* por medio de un script en C.
- Lograr ver la imagen de la estación pulsando el botón izquierdo del ratón y se desaparece la imagen soltando el botón izquierdo del ratón.

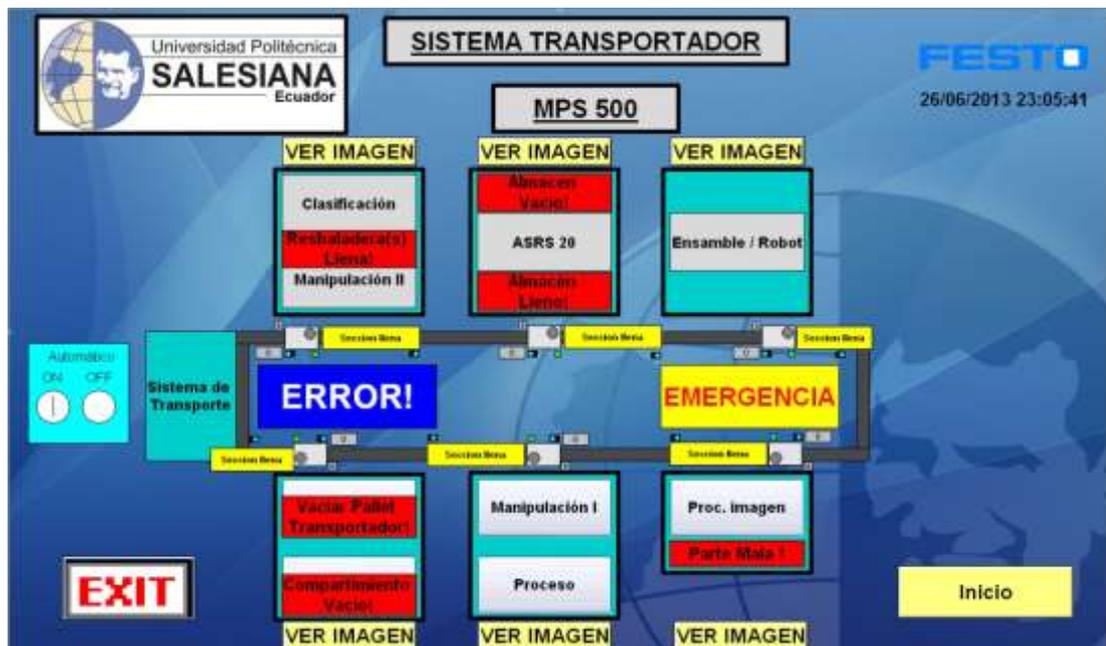


Figura 138: Pantalla del sistema transportador

Fuente: Los Autores, año 2013

A17 Distribución

En esta pantalla las animaciones se realizan por medio de diálogos dinámicos, se encuentran indicadores como flechas apuntando en diferentes direcciones, es decir se visualizan dependiendo de la dirección que se mueve el brazo, se presentan mensajes dando un aviso al operador "Coloque piezas y Presione Start".

Las animaciones aparecen en la pantalla con ayuda de los sensores de la estación, los atributos que se utilizan son visualización, intermitencia y colores.

Se logra reiniciar, parar o arrancar el módulo desde WinCC, se va poder visualizar el estado de las entradas, las salidas, el mensaje de emergencia que se presenta al momento de pulsar la parada de emergencia y un icono indicando que hubo un error en el proceso.

Se consigue navegar a las diferentes pantallas del proceso por medio de conexiones directas y scripts realizados en C.

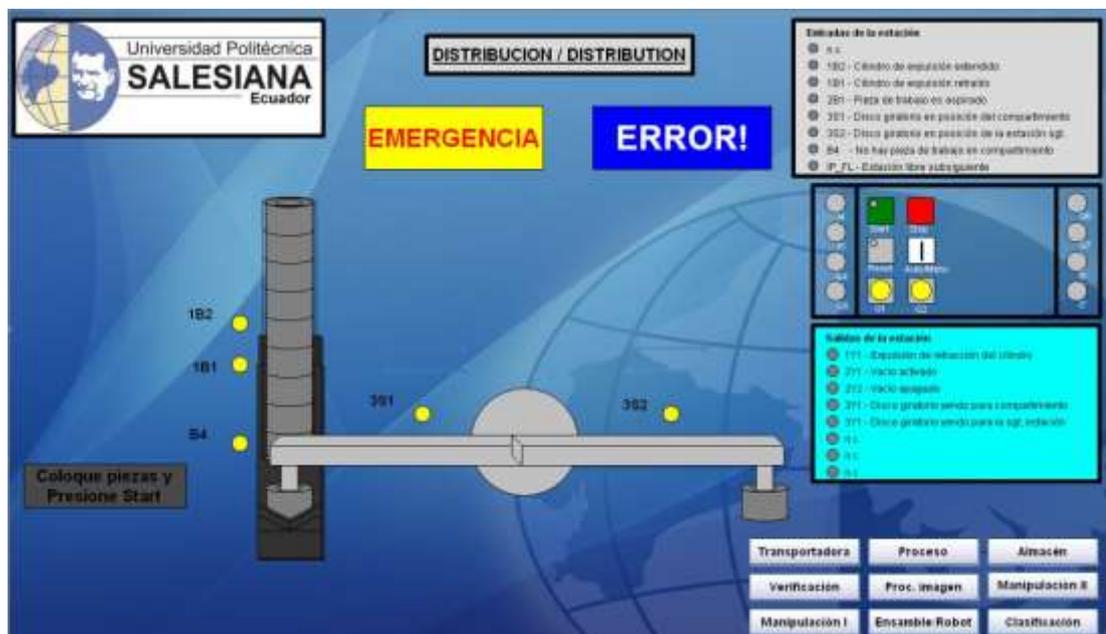


Figura 139: Pantalla de la ventana de distribución

Fuente: Los Autores, año 2013

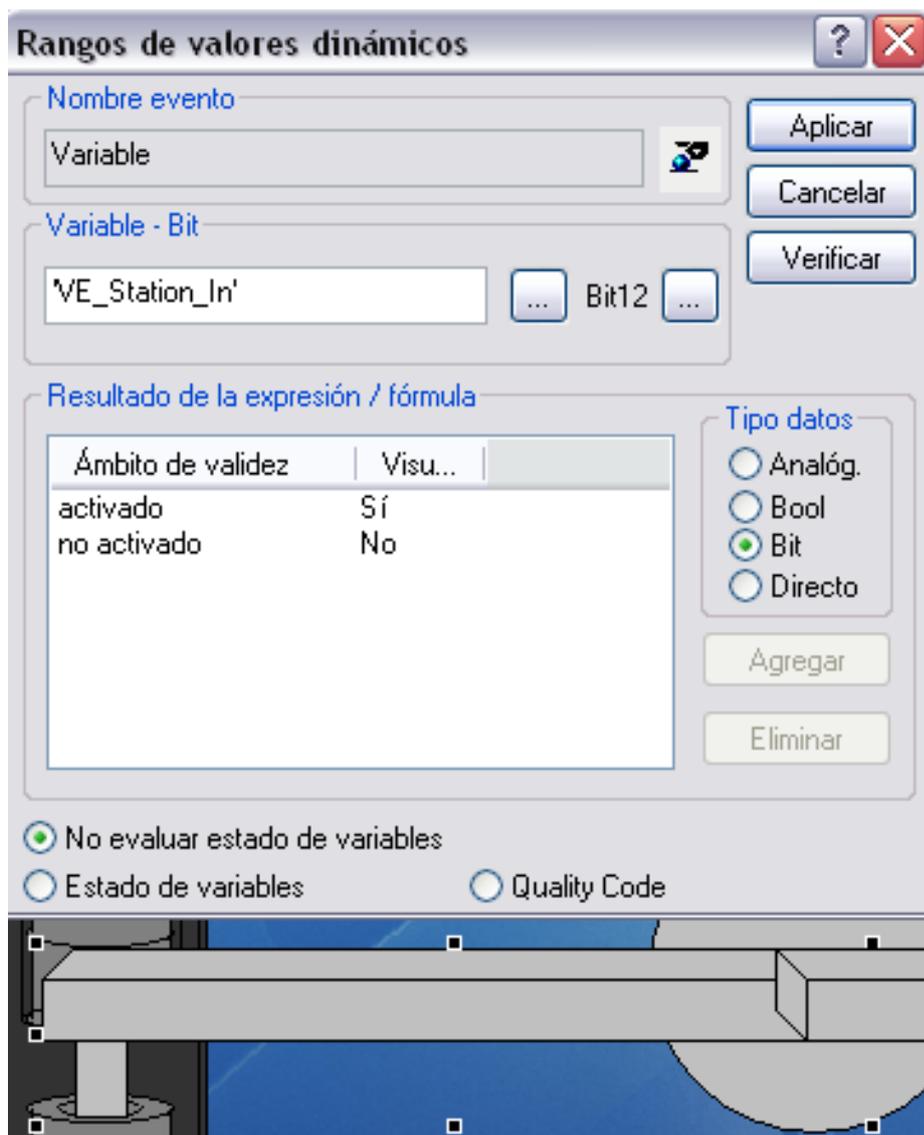


Figura 140: Animación por medio de cuadro de diálogo dinámico

Fuente: Los Autores, año 2013

A18 Verificación

En esta pantalla las animaciones se realizan por medio de diálogos dinámicos, se encuentran indicadores como flechas apuntando en diferentes direcciones, es decir se visualizan dependiendo de la dirección que se mueve el émbolo y el pistón.

Las animaciones aparecen en la pantalla con ayuda de los sensores de la estación, los atributos que se utilizan son visualización, intermitencia y colores.

Se logra reiniciar, parar o arrancar el módulo desde WinCC, se va poder visualizar el estado de las entradas, las salidas, el mensaje de emergencia que se presenta al momento de pulsar la parada de emergencia y un icono indicando que hubo un error en el proceso.

Se consigue navegar a las diferentes pantallas del proceso por medio de conexiones directas y scripts realizados en C.

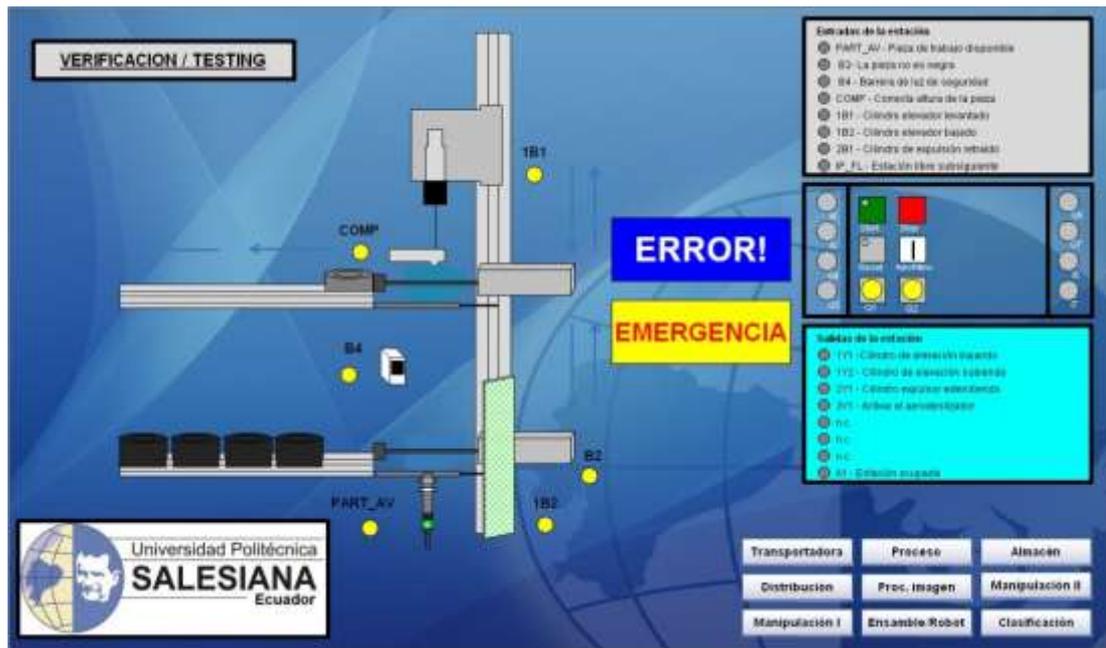


Figura 141: Pantalla de la ventana de verificación

Fuente: Los Autores, año 2013

A19 Manipulación I

En esta pantalla las animaciones se realizan por medio de diálogos dinámicos, se encuentran indicadores como flechas apuntando en diferentes direcciones, es decir se visualizan dependiendo de la dirección que se mueve la pinza (derecha, izquierda y abajo).

Las animaciones aparecen en la pantalla con ayuda de los sensores de la estación, los atributos que se utilizan son visualización, intermitencia y colores.

Se logra reiniciar, parar o arrancar el módulo desde WinCC, se va poder visualizar el estado de las entradas, las salidas, el mensaje de emergencia que se presenta al momento de pulsar la parada de emergencia y un icono indicando que hubo un error en el proceso.

Se consigue navegar a las diferentes pantallas del proceso por medio de conexiones directas y scripts realizados en C.

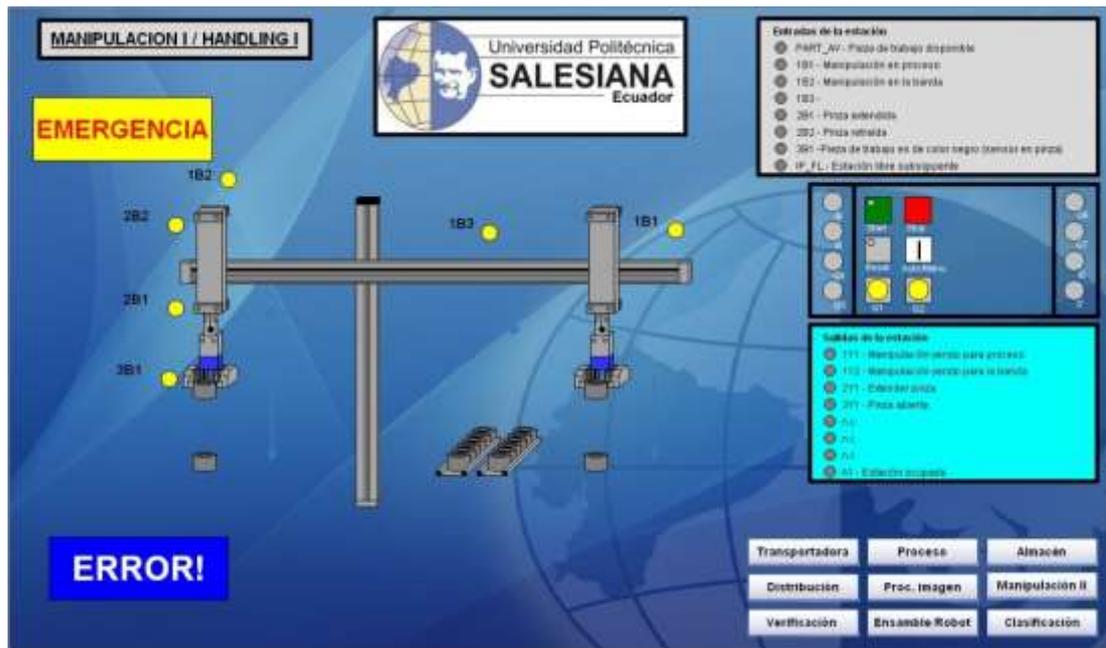


Figura 142: Pantalla de la ventana de manipulación I

Fuente: Los Autores, año 2013

A20 Proceso

En esta pantalla las animaciones se realizan por medio de diálogos dinámicos.

Cuando el sensor B3 envía el estado lógico cero las piezas aparecen en la mesa, indicando de que la mesa se encuentra en movimiento.

Las animaciones aparecen en la pantalla con ayuda de los sensores de la estación, los atributos que se utilizan son visualización, intermitencia y colores.

Se logra reiniciar, parar o arrancar el módulo desde WinCC, se va poder visualizar el estado de las entradas, las salidas, el mensaje de emergencia que se presenta al momento de pulsar la parada de emergencia y un icono indicando que hubo un error en el proceso.

Se consigue navegar a las diferentes pantallas del proceso por medio de conexiones directas y scripts realizados en C.

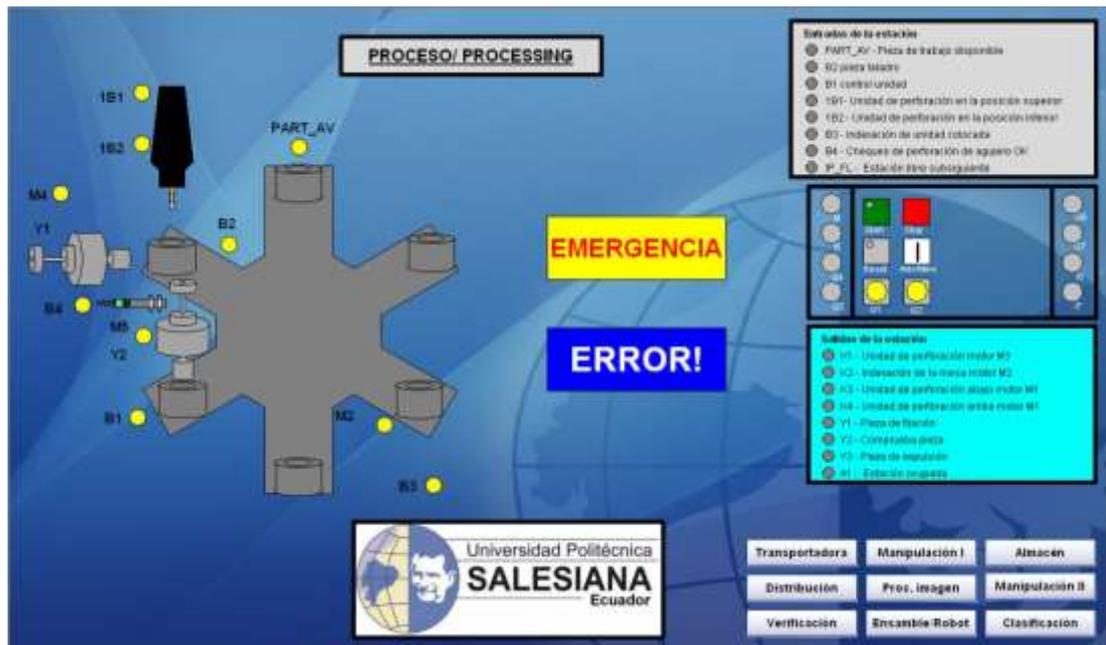


Figura 143: Pantalla de la ventana de proceso

Fuente: Los Autores, año 2013

A21 Almacén ASRS20

En esta pantalla las animaciones se realizan por medio de diálogos dinámicos, se encuentran indicadores como flechas apuntando en diferentes direcciones, es decir se visualizan dependiendo de la dirección que se mueve el brazo, se presentan mensajes dando un aviso al operador "Almacén lleno" y "Almacén vacío".

Las animaciones aparecen en la pantalla con ayuda de los sensores de la estación, los atributos que se utilizan son visualización, intermitencia y colores.

Cuando las piezas se dirigen hacia el almacén se va visualizando las piezas en el almacenamiento, y cuando se retiran las piezas del almacén las piezas no se visualizan.

Si la pinza se encuentra cerrada se visualiza la pieza en la pinza.

Se logra reiniciar, parar o arrancar el módulo desde WinCC, se va poder visualizar el estado de las entradas, las salidas, el mensaje de emergencia que se presenta al momento de pulsar la parada de emergencia y un icono indicando que hubo un error en el proceso.

Se consigue navegar a las diferentes pantallas del proceso por medio de conexiones directas y scripts realizados en C.

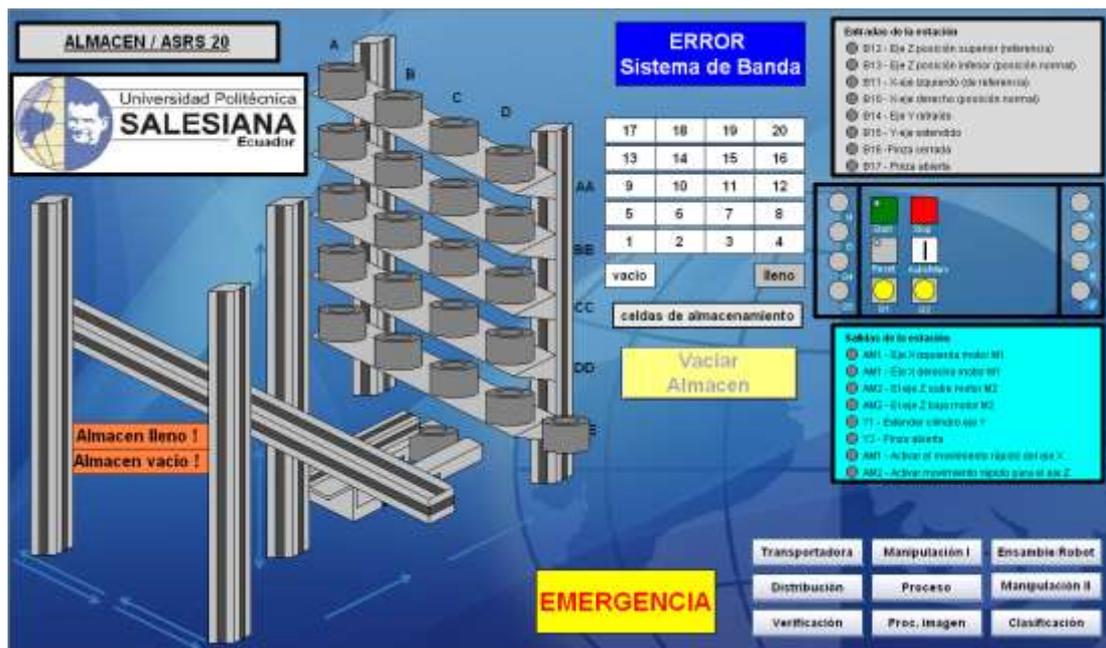


Figura 144: Pantalla de la ventana de almacén ASRS 20

Fuente: Los Autores, año 2013

A22 Manipulación II

En esta pantalla las animaciones se realizan por medio de diálogos dinámicos, se encuentran indicadores como flechas apuntando en diferentes direcciones, es decir se visualizan dependiendo de la dirección que se mueve la pinza (derecha, izquierda y abajo).

Las animaciones aparecen en la pantalla con ayuda de los sensores de la estación, los atributos que se utilizan son visualización, intermitencia y colores.

Se logra reiniciar, parar o arrancar el módulo desde WinCC, se va poder visualizar el estado de las entradas, las salidas, el mensaje de emergencia que se presenta al momento de pulsar la parada de emergencia y un icono indicando que hubo un error en el proceso.

Se consigue navegar a las diferentes pantallas del proceso por medio de conexiones directas y scripts realizados en C.

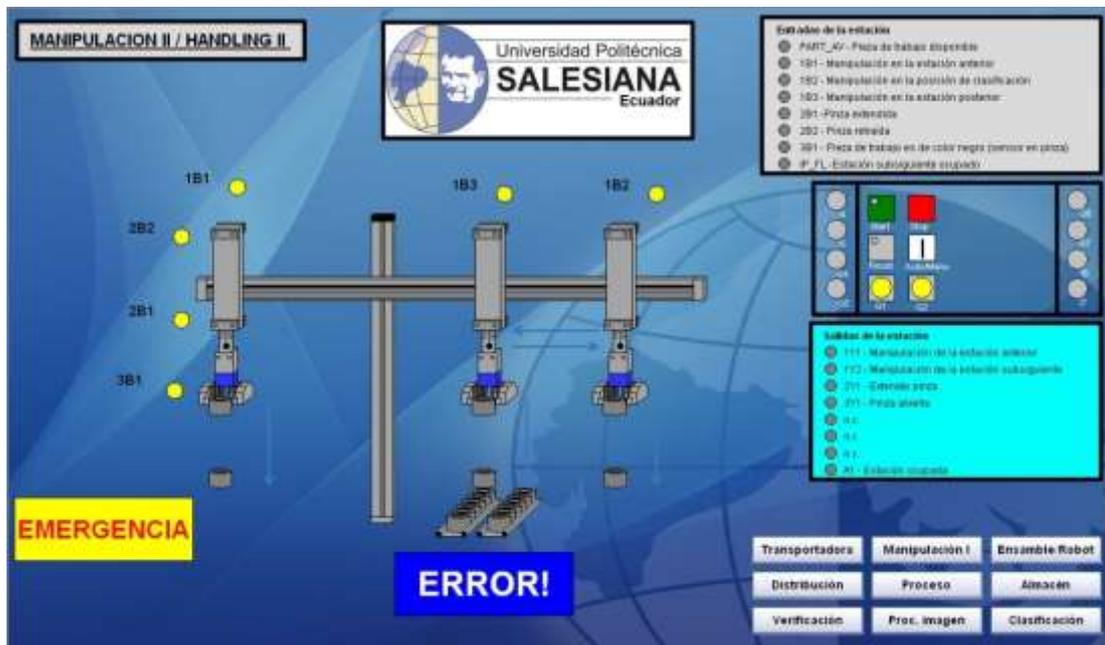


Figura 145: Pantalla de la ventana de manipulación II

Fuente: Los Autores, año 2013

A23 Clasificación

En esta pantalla las animaciones se realizan por medio de diálogos dinámicos, se encuentran indicadores como flechas apuntando en una sola dirección, se presentan mensajes dando un aviso al operador "Resbaladera(s) llenas".

Las animaciones aparecen en la pantalla con ayuda de los sensores de la estación, los atributos que se utilizan son visualización, intermitencia y colores.

Se logra reiniciar, parar o arrancar el módulo desde WinCC, se va poder visualizar el estado de las entradas, las salidas, el mensaje de emergencia que se presenta al momento de pulsar la parada de emergencia y un icono indicando que hubo un error en el proceso.

Se consigue navegar a las diferentes pantallas del proceso por medio de conexiones directas y scripts realizados en C.

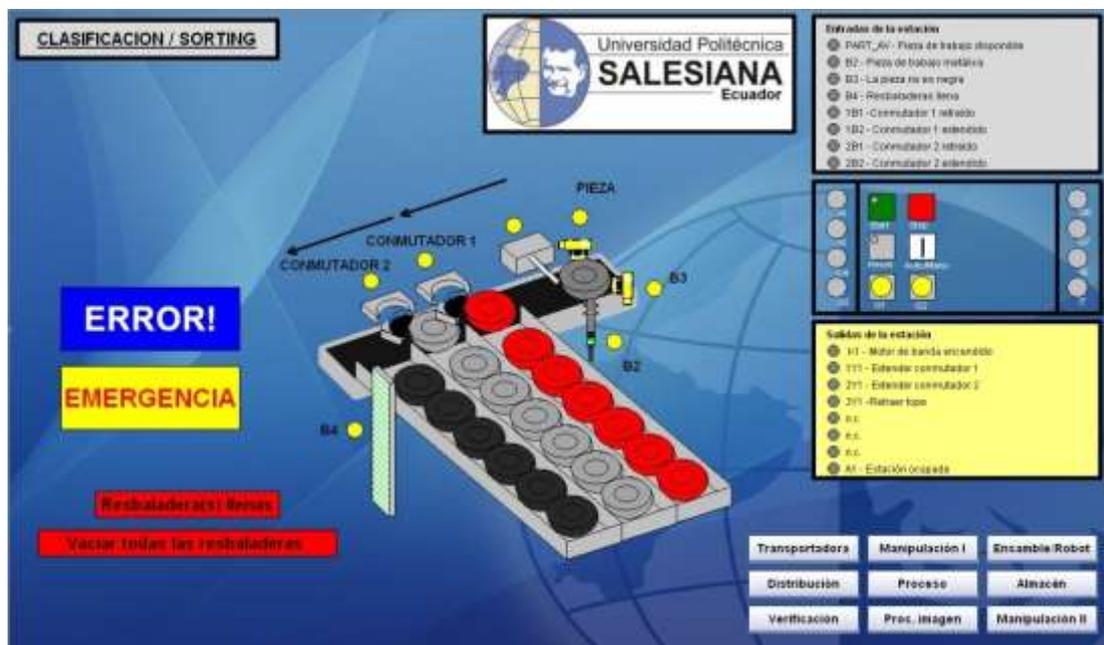


Figura 146: Pantalla de la ventana de clasificación

Fuente: Los Autores, año 2013