

Capítulo 6: **Descripción interna del Sistema Integral de Paletizado**

En este capítulo se describirá la estructura interna del Sistema Integral de Paletizado. Se detallarán las tablas que componen la base de datos Oracle que sustenta todo el sistema, la estructura de los procesos que corren en el servidor, cómo establecer un PLC concentrador...

Se analizará la estructura de cada uno de los módulos software de sistema y se mostrarán las tablas que formarán parte de dichos módulos (así como cada uno de sus campos).

Las características descritas en este capítulo consiguieron definirse tras numerosas conversaciones con el desarrollador del software y tras la redacción de varios documentos de especificaciones.

6.1 GENERALIDADES DEL FUNCIONAMIENTO INTERNO DEL SIP

6.1.1 Arquitectura del sistema

Tal y como se explicó en el capítulo cuatro, la arquitectura hardware del Sistema Integral de Paletizado, aplicada a la línea de paletizado genérica, es la siguiente:



Figura 6.1: Arquitectura global del Sistema Integral de Paletizado

Los ordenadores y autómatas se comunican a través de una red ethernet de fábrica, y los elementos a comunicar son:

- Servidor. En este equipo residen la base de datos, los servidores OPC y los procesos de comunicación con los elementos de las líneas.
- Ordenadores Cliente. Actualmente sólo existe uno y está situado en la zona de muelles. Tiene instalado el cliente Oracle (para el acceso a la base de datos) y la aplicación de gestión.
- Ordenadores de gestión de etiquetadoras. Estos ordenadores disponen de conexión ethernet. El SIP se comunicará con ellos a través de ficheros.
- Ordenadores SCADA de las líneas. Estos ordenadores disponen de conexión ethernet.

- **Autómatas.** Se ha definido un autómata concentrador en cada línea (con carácter general, será el autómata de enfardado). Los autómatas concentradores disponen de tarjetas de comunicación ethernet. De este modo el servidor OPC, instalado en el servidor principal, tiene acceso a los autómatas a través de la ethernet de fábrica.

6.1.1.1 Base de datos

La información se almacenará en una base de datos Oracle 10G, que reside en el servidor principal.

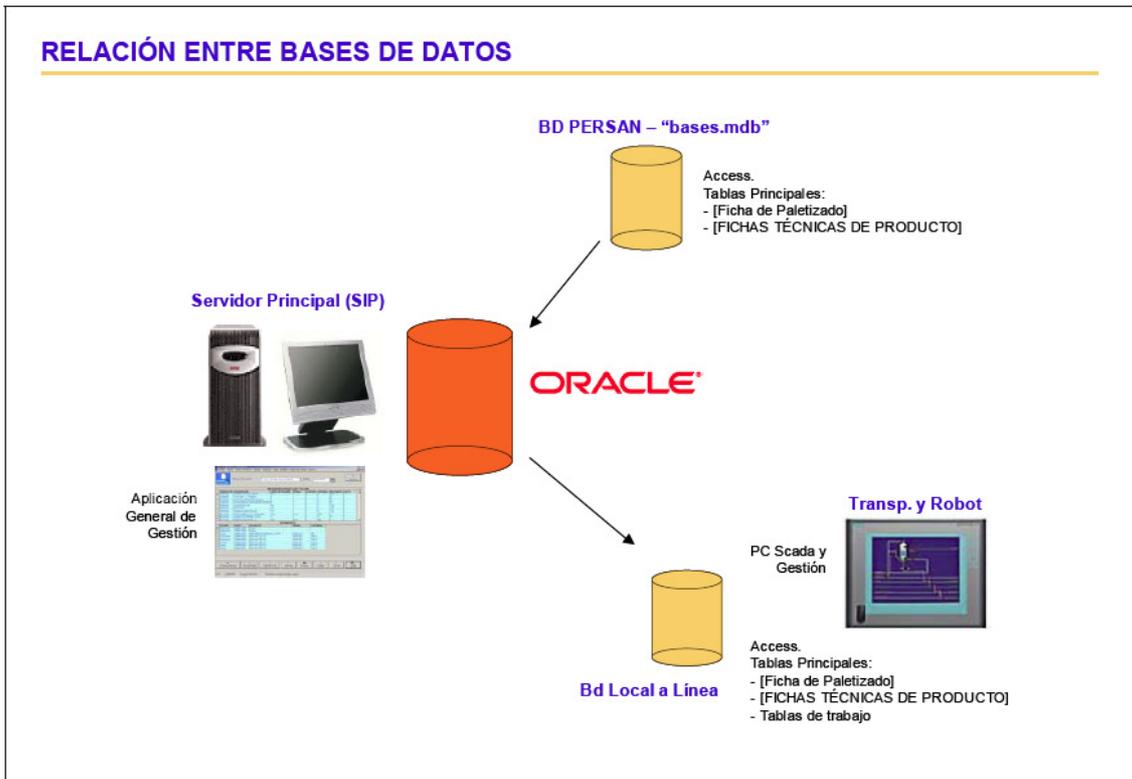


Figura 6.2: Relación entre bases de datos I+D, Oracle y locales a línea

6.1.1.2 Programación de los autómatas

Para realizar las modificaciones en los programas de los autómatas, se ha utilizado el paquete software:

- Step 7, versiones 5.2, para tecnología Siemens.

6.1.1.3 Procesos de comunicación con autómatas

Los procesos de comunicación con autómatas se han desarrollado en Visual C++ y son clientes OPC.

Los procesos están estructurados en forma de máquina de estados que se ejecuta periódicamente.

6.1.1.4 Usuarios de Windows

En los ordenadores a los que accede el sistema (servidor, ordenadores de etiquetadoras, paneles PC) se ha creado un usuario de Windows determinado. Este usuario es necesario para el acceso a determinadas carpetas (intercambio de ficheros en etiquetadoras, copias de seguridad, acceso a bases de datos Access, etc.).

6.1.1.5 Direcciones IP

Para los ordenadores y paneles táctiles, las direcciones IP son las siguientes:

| IP | LÍNEA | DESCRIPCIÓN |
|----------------|--------------|---|
| 192.168.25.1 | GENERICICO | SERVIDOR |
| | | |
| 192.168.25.101 | FANUC 1/2 | PC etiquetadora |
| 192.168.25.105 | MALETAS 2 | PC etiquetadora |
| 192.168.25.109 | EMMETTI A/B | PC etiquetadora |
| 192.168.25.115 | SAMOVI / M 1 | PC etiquetadora |
| | | |
| 192.168.25.123 | FANUC 1 | PC Siemens Fanuc 1 |
| 192.168.25.124 | FANUC 2 | PC Siemens Fanuc 2 |
| 192.168.25.126 | MALETAS 2 | PC Siemens Scalextric |
| 192.168.25.127 | EMMETTI A | Panel PC Siemens |
| 192.168.25.128 | EMMETTI B | Panel PC Siemens |
| 192.168.25.135 | MALETAS 2 | PC OCME |
| 192.168.25.164 | MALETAS 1 | PC OCME |
| | | |
| 192.168.25.210 | GENERICICO | PC ordenador de trabajo, situado en el muelle |

Tabla 6.1: Direcciones IP de ordenadores y paneles táctiles

Para los autómatas, las direcciones IP son las siguientes:

| IP | LÍNEA | DESCRIPCIÓN |
|---------------|-----------|---|
| 192.168.25.12 | SAMOVI | Autómata Omron |
| 192.168.25.13 | SAMOVI | Autómata concentrador Siemens (enfardadora) |
| 192.168.25.14 | MALETAS 1 | Autómata concentrador Siemens (enfardadora) |
| 192.168.25.15 | MALETAS 2 | Autómata concentrador Siemens (enfardadora) |
| 192.168.25.17 | FANUC 1 | Autómata concentrador Siemens (enfardadora) |
| 192.168.25.16 | FANUC 2 | Autómata concentrador Siemens (enfardadora) |
| 192.168.25.18 | EMMETTI A | Autómata concentrador Siemens (enfardadora) |
| 192.168.25.19 | EMMETTI B | Autómata concentrador Siemens (enfardadora) |

Tabla 6.2: Direcciones IP de los autómatas

6.1.2 Procesos del sistema

En el sistema existen una serie de procesos que realizan diversas funciones y cuyo estado se visualiza en la pantalla de gestión de procesos de la aplicación de supervisión.

En la siguiente tabla se muestra de forma general todos estos procesos, indicando algunos datos significativos:

| NOMBRE | COMUNICACION | FUNCIONALIDAD |
|-----------|--------------------------|---|
| COM_SAMO | PLC Omron – L3 SAMOVI | Proceso que obtiene los datos de diagnóstico y de los FP de SAMOVI |
| COM_PLC_3 | PLC Pieri – L3 SAMOVI | Proceso que obtiene los datos de diagnóstico de la Pieri de SAMOVI |
| COM_PLC_4 | PLC Pieri – L4 MALETAS 1 | Proceso que obtiene los datos de diagnóstico de la Pieri y del Robot de MALETAS 1 |
| COM_PLC_5 | PLC Pieri – L5 MALETAS 2 | Proceso que obtiene los datos de diagnóstico de la Pieri, del Robot y del scalextric de MALETAS 2 |
| COM_PLC_6 | PLC Pieri – L6 FANUC 1 | Proceso que obtiene los datos de diagnóstico de la Pieri, del Robot y del transporte de FANUC 1 |
| COM_PLC_7 | PLC Pieri – L7 FANUC 2 | Proceso que obtiene los datos de diagnóstico de la Pieri, del Robot y del transporte de FANUC 2 |

| | | |
|-----------|---------------------------------|---|
| COM_PLC_8 | PLC Pieri – L8 EMMETTI A | Proceso que obtiene los datos de diagnosis de la Pieri, del Robot y del transporte de EMMETTI A |
| COM_PLC_9 | PLC Pieri – L9 EMMETTI B | Proceso que obtiene los datos de diagnosis de la Pieri, del Robot y del transporte de EMMETTI B |
| | | |
| COM_ENF_3 | PLC Pieri – L3 SAMOVI | Proceso que lee y escribe los parámetros de programas de enfardado de SAMOVI |
| COM_ENF_4 | PLC Pieri – L4 MALETAS 1 | Proceso que lee y escribe los parámetros de programas de enfardado de MALETAS1 |
| COM_ENF_5 | PLC Pieri – L5 MALETAS 2 | Proceso que lee y escribe los parámetros de programas de enfardado de MALETAS2 |
| COM_ENF_6 | PLC Pieri – L6 FANUC 1 | Proceso que lee y escribe los parámetros de programas de enfardado de FANUC 1 |
| COM_ENF_7 | PLC Pieri – L7 FANUC 2 | Proceso que lee y escribe los parámetros de programas de enfardado de FANUC 2 |
| COM_ENF_8 | PLC Pieri – L8 EMMETTI A | Proceso que lee y escribe los parámetros de programas de enfardado de EMMETTI A |
| COM_ENF_9 | PLC Pieri – L9 EMMETTI B | Proceso que lee y escribe los parámetros de programas de enfardado de EMMETTI B |
| | | |
| COM_ETIQ | Con los PC's de etiquetadoras | Intercambia ficheros de texto con las aplicaciones de MACSA que hay en cada uno de los PC's de etiquetado |
| | | |
| RESPALDO | Origen y destinos configurables | Proceso que se encarga de realizar copias de seguridad de ficheros de un origen a un destino |

Tabla 6.3: Procesos que componen el sistema

6.1.3 Módulos del sistema

Como ya se explicó en el capítulo cinco, desde el punto de vista funcional el sistema se estructura en los siguientes módulos:

- Módulo de gestión de parámetros de referencias: gestión de las referencias de productos y los parámetros de paletizado.
- Módulo de estado y diagnosis: desde donde se mostrará en un sinóptico de línea el estado de cada una de las unidades lógicas que componen la línea.
- Módulo de enfardado: gestión de los parámetros de los programas de enfardado.

- Módulo de etiquetados: visualización de los datos de producción y configuración de etiquetadoras.
- Módulo de copias de seguridad: realización de copias de seguridad de ficheros o directorios configurados.

Sin embargo, desde el punto de vista interno del sistema, podemos establecer dos módulos más:

- Módulo servidor: herramientas generales de configuración y mantenimiento del sistema (tablas, usuarios, etc.).
- Módulo de comunicaciones con autómatas: lectura y escritura de datos en los autómatas a través de OPC.

6.1.4 Estructura genérica de los procesos desarrollados en C++

Los procesos desarrollados en C++ son ejecutables con extensión “.exe”. Corren en el servidor principal y necesitan conexión a la base de datos. Adicionalmente, los procesos que comunican con los autómatas son clientes OPC, y tienen acceso a los servidores OPC de Siemens y Omron que residen en el servidor principal.

6.1.4.1 Estructura de directorios

La estructura de directorios de los procesos desarrollados en C++ es la siguiente:

- bin: este directorio contiene los ejecutables, con extensión “.exe”.
- cfg: este directorio contiene los ficheros de inicialización “.ini”. Los ficheros de inicialización contienen los datos necesarios para la conexión a la base de datos y parámetros varios para el funcionamiento de proceso.
- include: este directorio contiene los archivos de cabecera con extensión “.h”.
- lib: este directorio contiene las librerías que contienen diversas funcionalidades necesarias para el correcto funcionamiento de los procesos.
- src: este directorio contiene los archivos fuente, con extensión “.cpp”
- traza: este directorio almacena los ficheros de traza. Los ficheros de traza son archivos de texto con extensión “.trz”, que es un registro con las acciones que va ejecutando el proceso.

6.1.4.2 Estructura interna: Máquina de estados

Los procesos están desarrollados en Visual C++ y tienen una estructura de máquina de estados que se ejecuta periódicamente.

En general, un proceso dispone de un estado inicial, un estado de conexión a base de datos, un estado de conexión a OPC, un estado de reposo y uno o varios estados de trabajo. La función de cada estado se detalla a continuación.

6.1.4.2.1 Estado inicial

Es el estado que se establece en el arranque de la aplicación.

En este estado el proceso trata el fichero de inicio, de donde extrae los parámetros de la aplicación.

6.1.4.2.2 Estado de conexión a base de datos

A partir de los parámetros del fichero de inicio, se establece la comunicación con la base de datos por medio del método “conecta”.

Una vez iniciada la conexión con la base de datos, el proceso accede a la tabla PROCESOS y establece la localización del fichero de traza y el nivel de detalle con el que se registrarán los eventos ocurridos durante la ejecución del mismo.

6.1.4.2.3 Estado de conexión a OPC

En este estado, el sistema establece comunicación con el PLC. Para ello accede a la tabla OPC_SERVERS, de donde obtiene la información sobre la conexión.

La función “ConexionPLC” se encarga de definir los grupos y los ítems OPC, y establece la conexión.

Si el proceso en C++ no necesita conexión a autómatas, no existirá este estado.

6.1.4.2.4 Estado de reposo

En este estado el proceso pasa a inactividad durante un tiempo establecido. Durante este tiempo permanecerá detenido y no se ejecutará ninguna operación.

En esta etapa, también se gestiona el funcionamiento del proceso (arranque, parada y vigilancia). La función “Bd_GetMuerteProceso” obtiene la información de la tabla PROCESOS.

6.1.4.2.5 Estado de trabajo

En este estado, el proceso realiza diferentes acciones dependiendo de las condiciones y funcionalidades particulares de cada proceso.

6.2 MÓDULO SERVIDOR

Desde este módulo se podrán gestionar herramientas generales de configuración y mantenimiento del sistema (tablas, históricos, usuarios,...). La funcionalidad de este módulo es implementada mediante la aplicación general de gestión y por un proceso interno, llamado Imon, que monitoriza los procesos y permite su arranque y parada.

6.2.1 Proceso “IMON”

El proceso interno Imon es el encargado de monitorizar el estado de los procesos y de actuar sobre ellos, tal y como se explica en el apartado 5.3.2.2

6.2.1.1 Arrancar el servidor

El proceso Imon debe estar corriendo en el servidor. Este proceso es fundamental para el correcto funcionamiento del sistema.

Debido a que es habitual acceder al servidor en modo “Escritorio remoto”, no se puede incluir el arranque del proceso “imon.exe” en el menú de Inicio de Windows.

Por tanto, al arrancar el servidor se debe arrancar el proceso Imon manualmente.

6.2.1.2 Funciones

Imon tiene como misión lanzar los diferentes procesos que componen el sistema global y vigilarlos para evitar paradas no deseadas.

Vigila que no se haya producido la muerte de ninguno de los procesos que ha lanzado. Si algún proceso ha muerto de forma no solicitada, Imon lo relanzará. Se puede configurar si el proceso es o no relanzable y existe un número máximo de relanzamientos por hora.

Tal y como se explica en el apartado 5.3.2.1, existe un formulario que se utiliza para mostrar el estado de los procesos que intervienen en el proyecto. Para cada proceso muestra el estado (parado o arrancado), y permite actuar sobre él (arrancándolo o parándolo).

Imon se basa en las tablas PROCESOS y MSG_PROC de la base de datos, descritas a continuación.

6.2.1.3 Tabla PROCESOS (PC)

Descripción: Tabla con los procesos que conforman el sistema

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|----------------|---------------|----------------|--|
| PC_ORDENADOR | VARCHAR2(50) | Clave Primaria | Nombre del ordenador |
| PC_NOMBRE | VARCHAR2(24) | | Nombre del proceso |
| PC_PID | NUMBER(4) | | PID del proceso |
| PC_EJEC | VARCHAR2(100) | | Ejecutable |
| PC_ARG | VARCHAR2(100) | | Argumentos de ejecución |
| PC_EST | NUMBER(3) | | Comprobación de estado |
| PC_EST_ACT | NUMBER(3) | | Estado actual |
| PC_REL | NUMBER(1) | IN (0,1) | Relanzable |
| PC_NUM_REL | NUMBER(2) | | Número actual de relanzamientos |
| PC_MAX_REL | NUMBER(2) | | Número máximo de relanzamientos por hora |
| PC_ULT_REL | DATE | | Ultimo relanzamiento |
| PC_WATCHDOG | NUMBER(1) | IN (0,1) | Vigilancia por Watchdog |
| PC_MUERTE_ORD | NUMBER(1) | IN (0,1) | Solicitud de muerte ordenada |
| PC_ESP_MUERTE | NUMBER(1) | IN (0,1) | Vigilancia por Watchdog |
| PC_RUTA_TRAZA | VARCHAR2(100) | | Path de la traza |
| PC_NIVEL_TRAZA | NUMBER(2) | | Nivel de traza |

Tabla 6.4: Tabla PROCESOS

6.2.1.4 Tabla MSG_PROC

Descripción: Tabla con los mensajes a Imon para el arranque y parada de procesos

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|-----------------|--------------|------------|-----------------------------------|
| MP_PC_ORDENADOR | VARCHAR2(50) | | Nombre del ordenador |
| MP_PC_NOMBRE | VARCHAR2(24) | | Nombre del proceso |
| MP_ACCION | NUMBER(1) | IN (0,1) | Acción (0 parar, 1 arrancar) |
| MP_EST | NUMBER(1) | IN (0,1) | Estado (0 sin tratar, 1 tratado) |
| MP_FCH | DATE | | Fecha de inserción de la petición |

Tabla 6.5: Tabla MSG_PROC

6.2.2 Copias de seguridad de la base de datos

Se realizan copias de seguridad diarias de la base de datos (esta acción es independiente del módulo de copias de seguridad del SIP). Las copias de seguridad tienen las siguientes características:

- Se realizan todos los días, a la una hora determinada. Esta acción se realiza mediante una tarea programada de Windows, que ejecuta el archivo "EXPORTAR.BAT".
- Se exporta el contenido del usuario de la base de datos PERSGCP_DBA.
- La copia se almacena en el fichero "PERSGCP_DBA.DMP" en un directorio determinado del servidor.

6.2.3 Eliminación de registros antiguos

Para evitar el crecimiento de la base de datos con datos antiguos, se ha implementado una funcionalidad de eliminación de registros. Esta funcionalidad consiste en la eliminación de registros anteriores a un número de días especificado para las tablas que se definan.

Se basa en los siguientes elementos:

- Un procedimiento almacenado llamado BORRAR_DATOS, que se ejecuta todos los días a través de un job de Oracle.
- Una tabla llamada ACTUALIZAR_TABLA, que almacena la información de las tablas cuyos registros antiguos se desea eliminar.

6.2.3.1 Tabla ACTUALIZAR_TABLA (AT)

Descripción: Configuración para eliminación de registros antiguos.

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|-----------------|--------------|----------------|--|
| AT_NOMBRE_TABLA | VARCHAR2(30) | Clave primaria | Nombre de la tabla |
| AT_NUM_DIAS | NUMBER(4) | | Número de días. |
| AT_CAMPO_FECHA | VARCHAR2(20) | | Nombre del campo de fecha de la tabla utilizado como referencia para la eliminación de registros |

Tabla 6.6: Tabla ACTUALIZAR_TABLA

6.3 MÓDULO DE COMUNICACIONES CON AUTÓMATAS

El sistema incluye un módulo de comunicaciones con los autómatas. Esta comunicación utiliza el estándar OPC.

6.3.1 Componentes del módulo

El módulo de comunicaciones con autómatas se compone de procesos desarrollados en C++, que comunican vía OPC con los autómatas. Son los siguientes:

- COM_PLC: Procesos de captura de datos de diagnosis de las líneas (Siemens).
- COM_ENF: Programas de comunicaciones con enfardadoras (Siemens).
- COM_SAMO: Proceso de comunicaciones con línea Samovi (Omron)

6.3.2 Procesos de comunicaciones

6.3.2.1 Generalidades

Los procesos de comunicaciones serán clientes OPC que necesitan un servidor OPC que les proporcione los datos.

Un servidor OPC es un software adicional que proporciona el fabricante de la tecnología, y que se instala en el servidor del sistema.

Existirán dos servidores OPC, uno para cada tecnología:

- Servidor OPC de Siemens, llamado "OPC.SimaticNET".
- Servidor OPC de Omron, llamado "OMRON.OpenDataServer.1".

Ambos servidores OPC corren en el servidor que sustenta al SIP.

Las comunicaciones con autómatas necesarias son las siguientes:

- Obtención del estado y diagnóstico de las líneas de paletizado.
- Lectura y volcado de programas de enfardado.

6.3.2.2 Dimensionamiento del número de procesos

Existirán varios procesos de comunicaciones con autómatas. El dimensionamiento del número de procesos tiene en cuenta las siguientes observaciones:

- Un proceso puede conectar a un solo servidor OPC. Por tanto, un mismo proceso no podrá comunicar simultáneamente con Omron y Siemens.
- Los datos de estado y diagnóstico se capturarán periódicamente. Otros datos, por ejemplo los de enfardadoras, se capturarán sólo bajo ciertas consideraciones.
- Es preferible que un proceso se conecte a un sólo autómatas, por las siguientes razones:
 - Un proceso se debe conectar correctamente a todos los autómatas que tenga asignado. Esta conexión se realiza en el estado inicial (ver apartado 6.1.4.2)
 - Si uno de los autómatas se encuentra en mantenimiento o si se caen comunicaciones con un autómatas, el proceso no conectaría correctamente con el resto.

Por tanto, el sistema dispone un proceso de comunicación por cada enfardadora y un proceso de captura de datos de diagnóstico por cada línea.

6.3.3 Modelo de datos

En este apartado se definen las tablas de la base de datos que almacenan la información relacionada con el módulo de comunicaciones con autómatas.

6.3.3.1 Tabla OPC_SERVERS (OS)

Descripción: Servidores OPC

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|-----------------|---------------|----------------------------|-------------------------------|
| OS_COD | NUMBER(3) | Clave primaria NOT NULL | Código de servidor |
| OS_DESC | VARCHAR2(24) | Clave primaria | Descripción |
| OS_SERVER | VARCHAR2(100) | | Nombre del servidor |
| OS_PC_ORDENADOR | VARCHAR2(24) | | PC donde reside |
| OS_PC_NOMBRE | VARCHAR2(24) | | Nombre del proceso que accede |

Tabla 6.7: Tabla OPC_SERVERS

6.3.3.2 Tabla OPC_GROUPS (OG)

Descripción: Declaración de grupos OPC.

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|-----------|--------------|------------------------------------|-------------------------|
| OG_OS_COD | NUMBER(3) | Clave primaria Ref: OPC_SERVERS | Código de servidor OPC |
| OG_COD | NUMBER(3) | Clave primaria | Código del grupo |
| OG_NOM | VARCHAR2(24) | | Nombre del grupo |
| OG_ACT | NUMBER(1) | | Activo: 1; No Activo: 0 |

Tabla 6.8: Tabla OPC_GROUPS

6.3.3.3 Tabla OPC_ITEMS (OI)

Descripción: Declaración de ítems OPC.

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|--------------|---------------|--------------------------------------|---|
| OI_OG_OS_COD | NUMBER(3) | Clave primaria Ref: OPC_GROUPS | Código de servidor OPC |
| OI_OG_COD | NUMBER(3) | Clave primaria Ref: OPC_GROUPS | Código del grupo OPC |
| OI_COD | NUMBER(3) | Clave primaria | Código del ítem |
| OI_DIR | VARCHAR2(100) | | Dirección OPC del tag |
| OI_DESC | VARCHAR2(100) | | Descripción del ítem, útil para aplicaciones de supervisión |
| OI_OT_COD | NUMBER(2) | Ref: OPC_TIPOS | Tipo de ítem OPC |
| OI_VAL_LEC | VARCHAR2(50) | | Valor leído |
| OI_FEC_LEC | DATE | | Fecha de la última lectura |
| OI_VAL_ESC | VARCHAR2(50) | | Valor a escribir |
| OI_EST_ESC | NUMBER(1) | | Se escribirá el ítem (0) o no (1) |

Tabla 6.9: Tabla OPC_ITEMS

6.3.3.4 Tabla OPC_TIPOS (OT)

Descripción: Tipos de datos de los ítems OPC declarados.

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|--------------|--------------|-------------------|--------------------|
| OT_COD | NUMBER(2) | Clave primaria | Código de tipo |
| OT_DESC | VARCHAS2(24) | | Descripción |

Tabla 6.10: Tabla OPC_TIPOS

6.4 **MODIFICACIONES EN LOS AUTÓMATAS**

Para la obtención de los datos de diagnóstico (estado y alarma) de las líneas de paletizado, fue necesario realizar modificaciones en los programas de los autómatas y en su configuración hardware.

6.4.1 **Modificaciones en la configuración hardware**

En todos los PLC's de las enfardadoras Pieri (7 PLC concentradores) se instaló una tarjeta de comunicaciones ethernet mod. CP-343-1 Lean.

En la siguiente tabla se muestran las direcciones IP de cada una de estas tarjetas de comunicación ethernet:

| LINEA | PLC | IP |
|--------------|-----------------------|---------------|
| SAMOVI | Omron - Transporte | 192.168.25.12 |
| | Siemens – Enfardadora | 192.168.25.13 |
| MALETAS 1 | Siemens – Enfardadora | 192.168.25.14 |
| MALETAS 2 | Siemens – Enfardadora | 192.168.25.15 |
| FANUC L1 | Siemens – Enfardadora | 192.168.25.17 |
| FANUC L2 | Siemens – Enfardadora | 192.168.25.16 |
| EMMETTI A | Siemens – Enfardadora | 192.168.25.18 |
| EMMETTI B | Siemens – Enfardadora | 192.168.25.19 |

Tabla 6.11: Direcciones IP de los elementos concentradores del sistema

6.4.1.1 Red MPI de PLC's

En todas las líneas, excepto en Samovi, el PLC concentrador está conectado vía MPI con el resto de elementos de la línea:

- Otros PLC de Siemens.
- Pantallas de supervisión, de la enfardadora y SCADA.

Por tanto, se tuvo cuidado a la hora de configurar las tarjetas ethernet, ya que hay que asignarles una dirección MPI que no entre en conflicto con el resto de elementos. Las direcciones MPI de cada uno de estos elementos se muestran en el capítulo cuatro.

6.4.2 Intercambio de información entre autómatas

Para que el PLC concentrador de cada una de las líneas de paletizado lea del resto de PLC's las posiciones de memoria utilizadas para diagnóstico, se han definido en la red MPI de comunicaciones GLOBAL DATA. Consiste en definir dentro de una red de comunicaciones una serie de posiciones de memoria de un PLC y declararlas como públicas para todos los elementos (PLC's) conectados a esa misma red.

En las siguientes imágenes se muestran los pasos para definir los global data de la línea de Maletas 2 usando la herramienta Simatic Manager:

1. Seleccionar la red MPI común a todos los PLC's



Figura 6.3: Primer paso para la definición de global data

2. En el menú Herramientas seleccionar "Definir datos globales". Estos corresponderán a la red MPI

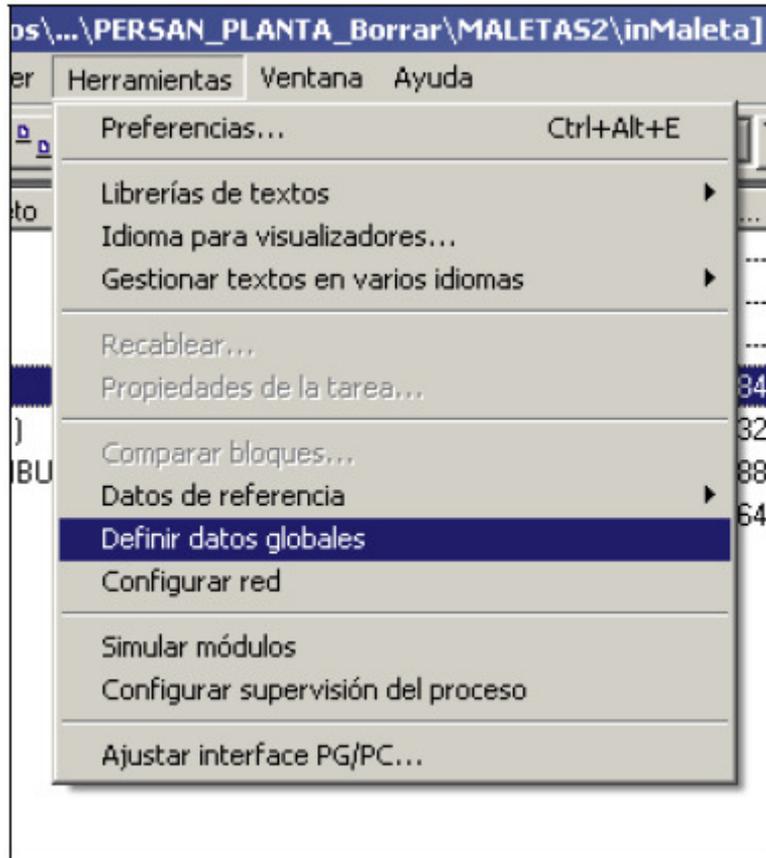


Figura 6.4: Segundo paso para la definición de global data

3. Escribir la tabla de datos globales a compartir entre PLC's definiendo el PLC emisor y el PLC receptor:

| | Identificador GD | inPieri\ CPU 313C | inRobot\ CPU 315-2 DP | inTransp\ CPU 315E-2 DP |
|----|------------------|----------------------|--------------------------|----------------------------|
| 1 | GD 1.1.1 | DB125.DBD78 | >DB9.DBD20 | |
| 2 | GD 1.1.2 | DB125.DBW116 | >DB9.DBW4 | |
| 3 | GD 1.1.3 | DB125.DBW122 | >DB20.DBW80 | |
| 4 | GD 1.1.4 | DB125.DBW124 | >DB20.DBW82 | |
| 5 | GD 1.1.5 | DB125.DBX148.0 | >A2.0 | |
| 6 | GD 1.1.6 | DB125.DBX148.1 | >A2.2 | |
| 7 | GD 2.1.1 | DB125.DBX150.0 | >A1.0 | |
| 8 | GD 2.1.2 | DB125.DBX150.1 | >A1.5 | |
| 9 | GD 2.1.3 | DB125.DBX150.2 | >A2.1 | |
| 10 | GD 2.1.4 | DB125.DBX150.5 | >M28.0 | |
| 11 | GD 2.1.5 | DB125.DBX150.6 | >M29.0 | |
| 12 | GD 3.1.1 | DB125.DBW208 | | >DB50.DBW30 |
| 13 | GD 3.1.2 | DB125.DBX224.0 | | >DB11.DEX0.1 |
| 14 | GD 3.1.3 | DB125.DBX224.1 | | >DB12.DEX0.1 |
| 15 | GD 3.1.4 | DB125.DBX224.2 | | >DB13.DEX0.1 |
| 16 | GD 3.1.5 | DB125.DBX224.3 | | >DB14.DEX0.1 |
| 17 | GD 3.1.6 | DB125.DBX224.4 | | >DB15.DEX0.1 |
| 18 | GD 3.1.7 | DB125.DBX224.5 | | >DB16.DEX0.1 |
| 19 | GD 4.1.1 | DB125.DBX224.6 | | >DB17.DEX0.1 |
| 20 | GD 4.1.2 | DB125.DBX226.0 | | >DB100.DEX8.2 |
| 21 | GD 4.1.3 | DB125.DBX226.1 | | >DB100.DEX8.3 |
| 22 | GD 4.1.4 | DB125.DBX226.2 | | >DB100.DEX8.4 |
| 23 | GD 4.1.5 | DB125.DBX226.3 | | >DB100.DEX8.5 |
| 24 | GD 4.1.6 | DB125.DBX226.4 | | >DB100.DEX8.6 |
| 25 | GD 4.1.7 | DB125.DBX226.5 | | >DB100.DEX8.7 |
| 26 | GD 4.1.8 | DB125.DBX226.6 | | >DB100.DEX9.0 |
| 27 | GD | | | |

Figura 6.5: Tercer paso para la definición de global data

4. Pulsar guardar y compilar. Si no aparece ningún mensaje de error se puede hacer una carga (download) sobre los PLC's que intervienen.

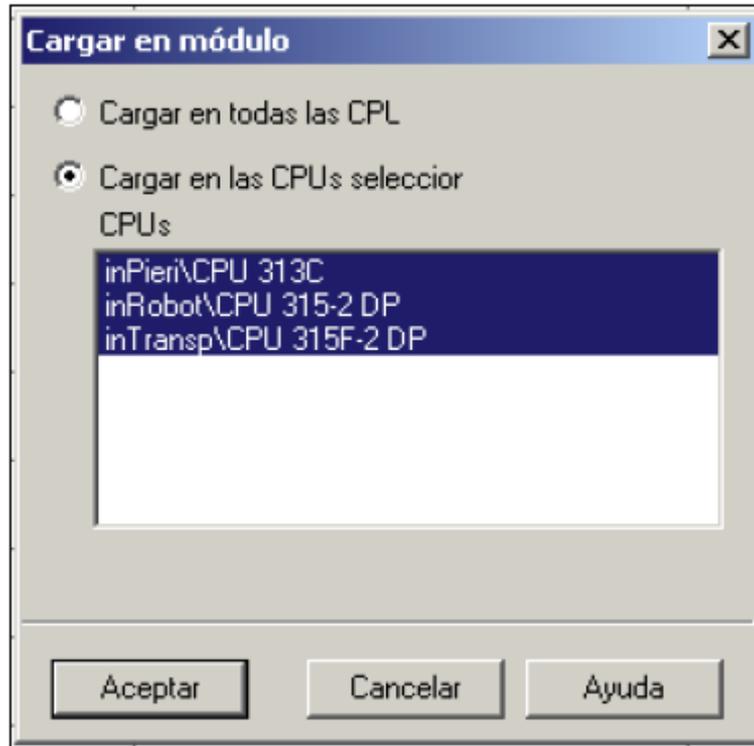


Figura 6.6: Cuarto paso para la definición de global data

6.4.3 Obtención de parámetros de diagnosis

Los PLC's de las enfundadoras se han utilizado como concentradores de parámetros. Para ello se ha instalado una tarjeta de comunicaciones ethernet CP343-1 en cada uno de ellos y se ha creado la función FC125 y el bloque de datos DB125.

En la siguiente figura se muestra un detalle del proyecto de la línea Maletas 2:

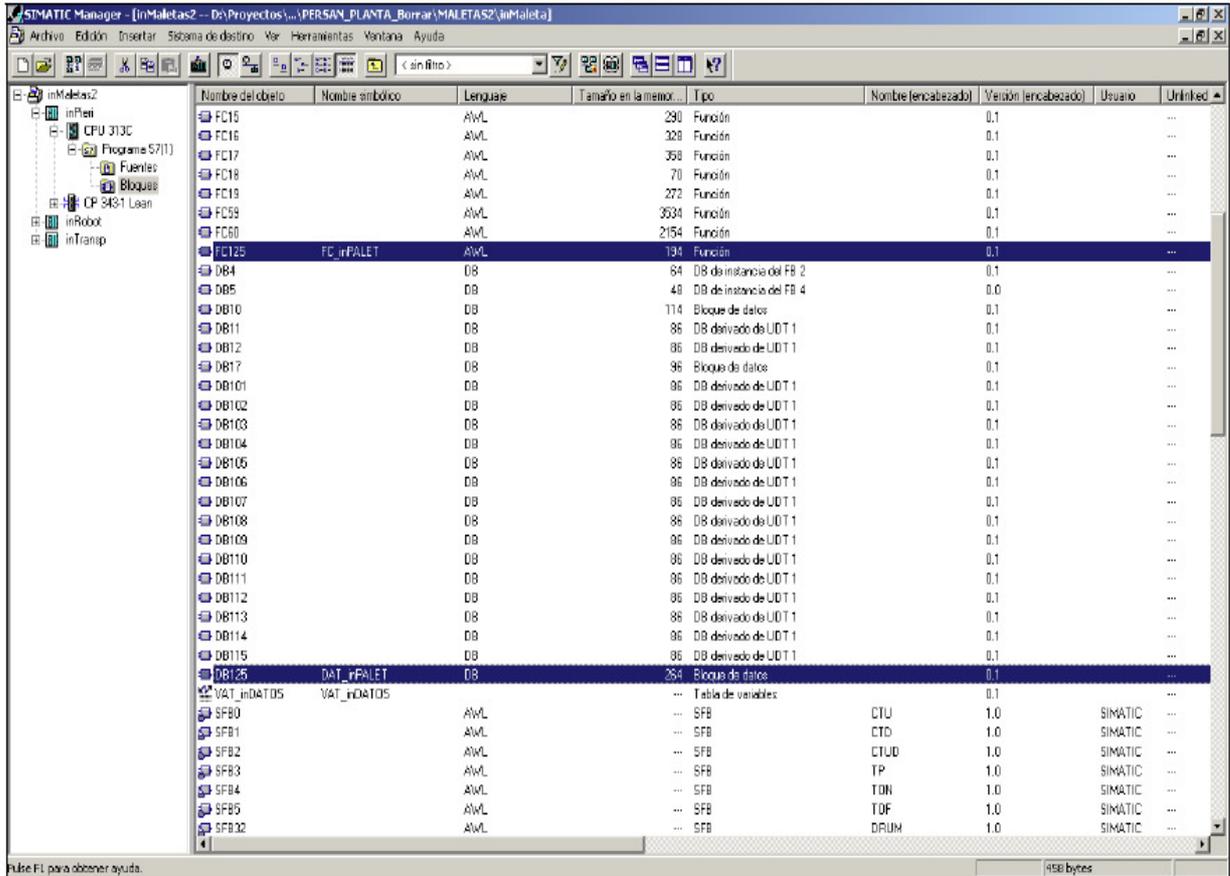


Figura 6.7: Captura del proyecto de Simatic Manager para la línea Maletas 2

En la rutina FC125 se realiza el cableado de todos los parámetros de diagnóstico de la enfardadora a las posiciones de lectura del proceso en VC++. De igual modo, en esta rutina se realiza el cableado de los bits generales de estado de todos los elementos: MARCHA y ALARMA.

En el DB125 se cablean todos los bits y parámetros de diagnóstico utilizados de cada elemento. Este fichero de datos está compuesto por 3 estructuras de datos:

- inPALET1: Datos de la enfardadora.
- inPALET2: Datos del robot paletizador.
- inPALET3: Datos del transporte.

Como ya se ha explicado, los parámetros de los PLC's, que no son el de la enfardadora, se comunican al PLC de la enfardadora (concentrador) mediante datos globales.

6.4.4 Parámetros de programa de enfardado

Para la lectura y modificación de los parámetros de los distintos programas de enfardado de cada una de las líneas, se ha utilizado la estructura de DB's y de FC's existentes en el código de la máquina.

En líneas generales, una enfardadora cuenta con los siguientes elementos comunes:

- DB's donde se guardan los parámetros de los 15 programas de enfardado.
- Rutina donde se gestiona la escritura / lectura de los DB's de datos.
- DB de trabajo donde se guardan los parámetros del programa a leer o escribir.
- DB de parámetros del programa de enfardado que se está utilizando actualmente.
- Órdenes de lectura / escritura.

En la instalación existen 2 tipos de programas de enfardadora:

- Tipo A: Samovi, Maletas 1, Emmetti A y B.
- Tipo B: Maletas 2, Fanuc 1 y 2.

En la siguiente tabla se muestran estos elementos para cada una de las enfardadoras.

| LINEA | FC TRABAJO | DB's DATOS | ORDEN LECTURA | ORDEN ESCRITURA |
|-----------|------------|-------------|--|---|
| SAMОВI | FC71 | DB21..DB35 | TIPO A: Lee el programa si las variables "Program_to_change" (DB10.DBW24) y "Aux_program" (DB10.DBW26) son diferentes o con el FP de M111.7 | TIPO A: Si se activa la marcha M54.0 (FP-flanco positivo) se dará la orden de guardar datos, mediante la M122.7 |
| MALETAS 1 | FC71 | DB21..DB35 | TIPO A | TIPO A |
| MALETAS 2 | FC59 | DB101.DB115 | TIPO B: Paso DB17.DBW24 de 16 a otro valor. Utiliza flanco negativo de un bit (M111.7) que se activa cuando es igual a 16 | TIPO B: Poner el bit DB17.DBX0.7 a 1 de forma momentánea, después se debe pasar a 0. Este bit es activado con el botón de guardar de la pantalla |
| FANUC L1 | FC59 | DB101.DB115 | TIPO B | TIPO B |

| | | | | |
|-----------|------|-------------|--------|--------|
| FANUC L2 | FC59 | DB101.DB115 | TIPO B | TIPO B |
| EMMETTI A | FC71 | DB21..DB35 | TIPO A | TIPO A |
| EMMETTI A | FC71 | DB21..DB35 | TIPO A | TIPO A |

Tabla 6.12: FC's y DB's de importancia para los dos tipos de enfundadora

6.4.5 PLC omron control transporte Samovi

En la línea de Samovi existen dos PLC's con los que comunica la aplicación de gestión:

- PLC Siemens concentrador de la enfundadora Pieri.
- PLC Omron que controla el transporte hacia los robots paletizadores.

En el PLC Omron residen los parámetros locales de todas las referencias que se pueden paletizar en dichas líneas. Por tanto, la aplicación accede vía ethernet a dicho PLC para leer/escribir los parámetros locales de dicho PLC. Al PLC de Omron se accede vía ethernet configurando el servidor de OPC de Omron llamado CX Server OPC. Dicho servidor de OPC reside en el servidor de la aplicación.

Se ha creado un proyecto "Omron_Samovi_v01.cdm", donde se han declarado los ítems que posteriormente van a ser leídos por el proceso de VC++.

6.4.5.1 Ítems configurados

En el proyecto de lectura de datos vía OPC se han creado dos tipos de ítems:

- Ítems de parámetros de diagnosis.
- Ítems de parámetros locales de referencias.

6.4.5.1.1 Parámetros de diagnosis

En la siguiente imagen se muestran las palabras y bits de diagnosis leídas de cada una de las líneas Samovi:

| Nombre | Tipo | Dirección |
|-------------------------|-------|--------------------|
| ^ ALM_R | Punto | Samovi/1100.3/1/B |
| ^ ALM_T | Punto | Samovi/1100.1/1/B |
| ▣ L1_DUN | Punto | Samovi/D0/30/RAW |
| ^ L1_MARCHA | Punto | Samovi/303.0/1/B |
| ▬ L1_ORIGEN | Punto | Samovi/D15/1/USH |
| ^ L1_PAROEMERG | Punto | Samovi/50.2/1/B |
| ▬ L1_ROBOTESTADO | Punto | Samovi/D620/1/USH |
| ▣ L2_DUN | Punto | Samovi/D30/30/RAW |
| ^ L2_MARCHA | Punto | Samovi/303.1/1/B |
| ▬ L2_ORIGEN | Punto | Samovi/D45/1/USH |
| ^ L2_PAROEMERG | Punto | Samovi/50.3/1/B |
| ▬ L2_ROBOTESTADO | Punto | Samovi/D621/1/USH |
| ▣ L3_DUN | Punto | Samovi/D60/30/RAW |
| ^ L3_MARCHA | Punto | Samovi/303.2/1/B |
| ▬ L3_ORIGEN | Punto | Samovi/D75/1/USH |
| ^ L3_PAROEMERG | Punto | Samovi/50.4/1/B |
| ▬ L3_ROBOTESTADO | Punto | Samovi/D622/1/USH |
| ▣ L4_DUN | Punto | Samovi/D90/30/RAW |
| ^ L4_MARCHA | Punto | Samovi/303.3/1/B |
| ▬ L4_ORIGEN | Punto | Samovi/D105/1/USH |
| ^ L4_PAROEMERG | Punto | Samovi/50.5/1/B |
| ▬ L4_ROBOTESTADO | Punto | Samovi/D623/1/USH |
| ▣ L5_DUN | Punto | Samovi/D120/30/RAW |
| ^ L5_MARCHA | Punto | Samovi/303.4/1/B |
| ▬ L5_ORIGEN | Punto | Samovi/D135/1/USH |
| ^ L5_PAROEMERG | Punto | Samovi/50.6/1/B |
| ▬ L5_ROBOTESTADO | Punto | Samovi/D624/1/USH |
| ▣ L6_DUN | Punto | Samovi/D150/30/RAW |
| ^ L6_MARCHA | Punto | Samovi/303.5/1/B |
| ▬ L6_ORIGEN | Punto | Samovi/D165/1/USH |
| ^ L6_PAROEMERG | Punto | Samovi/50.7/1/B |
| ▬ L6_ROBOTESTADO | Punto | Samovi/D625/1/USH |
| ^ MARCHA_R | Punto | Samovi/1100.2/1/B |
| ^ MARCHA_T | Punto | Samovi/1100.0/1/B |
| ^ PAROEMERGENCIAGENERAL | Punto | Samovi/50.1/1/B |

Figura 6.8: Palabras y bits de diagnosis leídas de las líneas Samovi

6.4.5.1.2 Parámetros de referencias

En la siguiente imagen se muestran las palabras de datos leídas del buzón de escritura de parámetros:

| Nombre | Tipo | Dirección |
|-----------------|-------|---------------------|
| Confirmacion | Punto | Samovi/4000.3/1/B |
| IndiceProd | Punto | Samovi/D4000/1/USD |
| Peticion | Punto | Samovi/4000.2/1/B |
| ProdDUN | Punto | Samovi/D4128/14/RAW |
| ProdDescripcion | Punto | Samovi/D4110/30/RAW |
| ProdFormato | Punto | Samovi/D4127/1/USD |
| ProdOrigen | Punto | Samovi/D4125/1/USH |
| ProdPisos | Punto | Samovi/D4126/1/USD |

Figura 6.9: Palabras de datos leídas del buzón de escritura de parámetros en Samovi

En la siguiente imagen se muestran las palabras de datos leídas del buzón de lectura de parámetros:

| Nombre | Tipo | Dirección |
|-----------------|-------|---------------------|
| Confirmacion | Punto | Samovi/4000.1/1/B |
| IndiceProd | Punto | Samovi/D4000/1/USD |
| Peticion | Punto | Samovi/4000.0/1/B |
| ProdDUN | Punto | Samovi/D4028/14/RAW |
| ProdDescripcion | Punto | Samovi/D4010/30/RAW |
| ProdFormato | Punto | Samovi/D4027/1/USD |
| ProdOrigen | Punto | Samovi/D4025/1/USH |
| ProdPisos | Punto | Samovi/D4026/1/USD |

Figura 6.10: Palabras de datos leídas del buzón de lectura de parámetros en Samovi

Si se desea leer / escribir alguna posición de memoria del PLC vía OPC se deberá crear primero en este proyecto, para que sea accesible desde cualquier cliente de OPC.

6.5 **MÓDULO DE GESTIÓN FP Y FTP**

El módulo de gestión FP y FTP permite visualizar e importar las Fichas Técnicas de Producto y las Fichas de Paletizado. Estos valores están en la base de datos Access “bases.mdb”, que proporciona el departamento de I+D.

6.5.1 Componentes del módulo

El módulo de gestión FP y FTP tiene los siguientes componentes:

- Pantallas de la aplicación de supervisión, para la visualización y la modificación de los valores.
- Importación de base de datos I+D.
- Gestión de parámetros locales.

6.5.2 Modelo de datos

6.5.2.1 Tabla FTP (FTP)

Descripción: Tabla de Fichas Técnicas de Producto

La estructura de esta tabla será similar a la correspondiente tabla en la base de datos Access proporcionada por I+D.

6.5.2.2 Tabla FP (FP)

Descripción: Tabla de Fichas de Paletizado

La estructura de esta tabla será similar a la correspondiente tabla en la base de datos Access proporcionada por I+D.

6.5.2.3 Tabla PARAM_RUTAS (PR)

Descripción: Tabla de Rutas de bases de datos Access de Parámetros Locales.

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|-----------|-----------|----------------|--|
| PR_LP_COD | NUMBER(3) | Clave Primaria | Línea de paletizado a la que pertenece la ruta |

| | | | (0=TODAS) |
|-----------------|---------------|----------------|---|
| PR_PG_COD | NUMBER(3) | Clave Primaria | Código de la ruta |
| PR_PC | VARCHAR2(20) | | Ordenador donde reside el archivo |
| PR_RUTA | VARCHAR2(200) | | Ruta del archivo |
| PR_PERIODO | NUMBER(6) | | Periodicidad (minutos), si fuese necesario un chequeo periódico |
| PR_ULTIMA_FECHA | DATE | | Fecha de la última actualización del fichero |
| PR_FLAG_FORZADO | NUMBER(1) | | Indica si se ha activado el forzado (1) o no (0) desde la aplicación de supervisión |
| PR_HABILITACION | NUMBER(1) | | Indica si la copia de seguridad está habilitada (1) o no (0) |

Tabla 6.13: Tabla PARAM_RUTAS

6.5.2.4 Tabla PARAM_LOCALES (PL)

Descripción: Tabla que almacena los parámetros locales de todas las líneas de paletizado.

Para cada línea y grupo hay una serie de parámetros locales. El concepto grupo es de utilidad por la existencia de varias bases de datos locales en una línea (por ejemplo, maletas 2 tiene las bases de datos Access de Albatros y las de ProTools).

Cada parámetro tendrá:

- Código secuencial.
- Nombre, para su visualización en las pantallas de la aplicación de supervisión.
- El tipo de dato (bit, string, entero, flotante).
- En el caso de Emmetti, existirán los campos de dirección, bit de inicio y longitud de los bits, debido a las condiciones particulares de la herramienta ProTools que controla dicha línea de paletizado.

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|--------------|-----------|--|----------------------------------|
| PL_PG_LP_COD | NUMBER(3) | Clave Primaria Ref: PARAM_GRUPOS | Código de la línea de paletizado |
| PL_PG_COD | NUMBER(3) | Clave Primaria Ref: | Código del grupo para la línea |

| | | PARAM_GRUPOS | |
|--------------|--------------|------------------|---|
| PL_COD | NUMBER(4) | Clave Primaria | Código del parámetro dentro del grupo |
| PL_NOMBRE | VARCHAR2(50) | | Nombre del parámetro |
| EN_LIM_SUP | NUMBER(7,1) | | Límite superior del parámetro, si es numérico |
| EN_LIM_INF | NUMBER(7,1) | | Límite inferior del parámetro, si es numérico |
| EN_TD_COD | NUMBER(3) | Ref: TIPOS_DATOS | Código del tipo de dato (bit, string, entero, flotante) |
| EN_DIRECCION | VARCHAR2(50) | | Dirección de la palabra entero largo (sólo Emmetti) |
| EN_BIT_INI | NUMBER(3) | | Bit de inicio dentro de la palabra (sólo Emmetti) |
| EN_LONG_BITS | NUMBER(3) | | Longitud en bits (sólo Emmetti) |

Tabla 6.14: Tabla PARAM_LOCALES

6.5.2.5 Tabla PARAM_VALORES (PV)

Descripción: Tabla que almacena los valores de los parámetros locales de todas las líneas de paletizado.

Esta tabla asocia valores a los parámetros declarados en la tabla PARAM_LOCALES. Un parámetro no tiene un valor único, sino que va adquiriendo diferentes valores según la referencia a la que esté asociada.

Existe un campo llamado COD_INTERNO, que permite asociar los valores de los parámetros a una determinada referencia. Esa asociación se hará de forma diferente dependiendo de la línea (cada base de datos local es diferente), aunque el modelo de datos es común a todas las líneas.

El valor de cada parámetro se almacena en tipo "string" para hacer común el modelo de datos a todas las líneas, y se convertirá al tipo de datos correspondiente (el tipo de datos está almacenado en la tabla PARAM_LOCALES).

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|-----------------|-----------|---|----------------------------------|
| PV_PL_PG_LP_COD | NUMBER(3) | Clave Primaria Ref: PARAM_LOCALES | Código de la línea de paletizado |

| | | | |
|----------------|--------------|---|--|
| PV_PL_PG_COD | NUMBER(3) | Clave Primaria Ref: PARAM_LOCALES | Código del grupo para la línea |
| PV_PL_COD | NUMBER(3) | Clave Primaria Ref: PARAM_LOCALES | Código del parámetro dentro del grupo |
| PV_COD_INTERNO | VARCHAR2(20) | Clave Primaria | Código del parámetro dentro del grupo |
| PV_VALOR | VARCHAR2(40) | | Valor del parámetro (tipo string, que luego se convertirá al tipo deseado) |

Tabla 6.15: Tabla PARAM_VALORES

6.5.2.6 Tabla PARAM_GRUPOS (PG)

Descripción: Tabla que almacena los grupos de parámetros.

Por ejemplo, en Maletas existirán dos grupos: Maletas Albatros y Maletas Scalextric.

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|-----------|--------------|-----------------------------------|----------------------------------|
| PG_LP_COD | NUMBER(3) | Clave Primaria Ref: LINEAS_PAL | Código de la línea de paletizado |
| PG_COD | NUMBER(3) | Clave Primaria | Código del grupo para cada línea |
| PG_NOMBRE | VARCHAR2(30) | | Nombre del grupo |

Tabla 6.16: Tabla PARAM_GRUPOS

6.5.2.7 Tabla PARAM_EQUIVALENCIAS (PE)

Descripción: Tabla que almacena las equivalencias.

Por ejemplo, en Samovi es necesario tener relacionados el código de la línea de envasado con un índice en el autómata.

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|---------------|-----------|-----------------------------------|----------------------------------|
| PE_LP_COD | NUMBER(3) | Clave Primaria Ref: LINEAS_PAL | Código de la línea de paletizado |
| PE_TIPO_EQUIV | NUMBER(3) | Clave Primaria | Tipo de la equivalencia |

| | | | |
|-------------|--------------|------------------|--------------------|
| PE_COD_ORIG | VARCHAR2(30) | Clave Primaria | Código origen |
| PE_COD_DEST | VARCHAR2(30) | | Código equivalente |
| PE_TD_COD | VARCHAR2(30) | Ref: TIPOS_DATOS | Tipo de dato |

Tabla 6.17: Tabla PARAM_EQUIVALENCIAS

6.5.2.8 Tabla PARAM_EMMETTI (PI)

Descripción: Tabla que almacena los datos de intercambio con los paneles Emmetti, para el tratamiento y la generación de los ficheros de intercambio “.csv”.

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|-------------------|--------------|----------------|---|
| PI_PL_PG_LP_COD | NUMBER(3) | Clave Primaria | Código de la línea de paletizado |
| PI_PL_PG_COD | NUMBER(3) | Clave Primaria | Código del grupo para la línea |
| PI_PL_COD | NUMBER(4) | Clave Primaria | Código del parámetro dentro del grupo |
| PI_COD_INTERNO | NUMBER(4) | Clave Primaria | Código secuencial, que ayuda a la generación del “.csv” |
| PI_COD_REFERENCIA | VARCHAR2(15) | | Nombre de la ficha de paletizado |
| PI_VALOR | VARCHAR2(20) | | Valor (almacenado en forma de entero largo) |
| PI_DIRECCION | VARCHAR2(20) | | Dirección de la palabra entero largo |

Tabla 6.18: Tabla PARAM_EMMETTI

6.5.3 Importación de datos I+D

Este módulo incluye la funcionalidad para importar a la base de datos Oracle los datos de la base de datos Access proporcionada por I+D.

La importación de la base de datos Access “bases.mdb” sobre Oracle está desarrollada en Visual Basic, y es llamada cuando en la aplicación se pulsa en el botón “Importar I+D”.

El botón “Importar I+D” existe en la aplicación de supervisión (ver apartado 5.4).

La base de datos Access “bases.mdb” debe ser copiada desde su ubicación original en I+D a un directorio (configurable) del servidor.

El sistema copia automáticamente el “bases.mdb” desde el directorio del servidor a los ordenadores de línea que lo necesitan, a la siguiente ruta por defecto:

C:\Mantenimiento\Bases.mdb

Todas las rutas son configurables en la tabla PARAM_RUTAS.

6.5.3.1 Proceso de importación

La secuencia de pasos es la siguiente:

1. Se consulta en la tabla PARAM_RUTAS las localizaciones de la base de datos Access “bases.mdb”.
2. Se realiza la consulta de las fichas técnicas de producto de “bases.mdb”.
3. Se compara la información de Access con la de la tabla FTP de Oracle, y se actualiza ésta con los nuevos valores. Los casos posibles son:
 - a. Si la referencia existe en “bases.mdb” pero no existe en Oracle, se inserta un nuevo registro.
 - b. Si la referencia existe en “bases.mdb” y en Oracle, se modifican los valores de Oracle con los valores de “bases.mdb”.
4. Se realiza la consulta de las fichas de paletizado de “bases.mdb”.
5. Se compara la información de Access con la de la tabla FP de Oracle, y se actualiza ésta con los nuevos valores, de forma similar a como se hace en las fichas técnicas de producto.
6. Se actualiza la tabla ETIQ_PRODUCTOS de Oracle, que une la información que proporciona “bases.mdb” con la información que proporcionan las etiquetadoras.
7. Se copia el archivo “bases.mdb” a localizaciones remotas.

En los PC de las líneas de paletizado Maletas y Fanuc existen bases de datos Access vinculadas a la base de datos “bases.mdb”.
El sistema copia automáticamente “bases.mdb” a las localizaciones que tienen configuradas si contestamos afirmativamente a la pregunta que nos realiza la aplicación.
El “bases.mdb” se copia automáticamente a los directorios configurados por defecto en la tabla PARAM_RUTAS. Los ordenadores de línea que necesitan el “bases.mdb” son los dos paneles de Fanuc y el de Maletas 2 Scalextric.
En Samovi las referencias residen en el autómatas, y en las Emmetti residen en los paneles de Siemens. En estas líneas no hay referencia a la base de datos Access proporcionada por I+D.

Observaciones

- Para garantizar que la instalación de paletizado siga funcionando ante una caída de la red ethernet, las líneas no dependen del acceso a Oracle.
- En caso de caída de la red, la copia del “bases.mdb” se deberá realizar manualmente.

- El operador debe actualizar la base de datos Access local, pulsando al botón “actualizar BD” existente en los ordenadores de línea. Este proceso es similar a las acciones que realizaban anteriormente a la implantación del nuevo sistema.

6.5.3.2 Funciones

- ImportarFP

Es llamada desde el botón “Importar I+D” y se encarga de la funcionalidad principal de la importación.

- AbrirConexionAccess

Recibe una ruta de un archivo “.mdb” y asocia una conexión a la base de datos Access.

Es utilizada para abrir el “bases.mdb” y realizar la consulta de las fichas técnicas de producto y las fichas de paletizado.

- AbrirIP

Utiliza el comando “net use” del sistema operativo para abrir conexiones en los equipos diferentes a los que corre la aplicación.

- ActualizarDesdeRecordset2

Versión mejorada de la función ActualizarDesdeRecordset, que recibe un recordset y actualiza los valores de la tabla en función de dicho recordset, considerando los casos siguientes:

- Si en el recordset existe un registro que no existe en la tabla, se inserta un nuevo registro.
- Si en el recordset existe un registro que ya existe en la tabla, se actualiza el registro existente.

Esta función es utilizada pasándole como parámetros el recordset con la consulta en Access, para actualizar la tabla Oracle correspondiente.

- CopiarFichero

Esta función copia ficheros desde una localización a otra.

Es utilizada para copiar el “bases.mdb” a los ordenadores de la línea que sea necesario.

6.5.3.3 Proceso PL/SQL

A la hora de hacer el Import de la base de datos “bases.mdb” sobre la base de datos de Oracle, se llama a un proceso PL/SQL, llamado “FTP_EXCEPCIONES” y que realiza las siguientes operaciones:

6.5.3.3.1 Cantidad nominal

Consulta las tablas FP y FTP de la base de datos Oracle y en función del valor de los campos FTP_GAMA y FP_TIPO_PALET, rellena el campo FP_NOMINAL de la tabla FP y el EP_NOMINAL de la tabla ETIQ_PRODUCTOS.

El valor de la cantidad nominal depende de:

- Si GAMA = 11 y TIPO_PALET = Media Paleta el valor será = 1.
- En otro caso, el valor será = FP_UNIDAD_BASE * FP_NUMERO_ALTURAS.

6.5.3.3.2 Programa enfardado

Consulta las tablas FP y FTP de la base de datos Oracle y en función del valor de los campos FTP_GAMA y FP_TIPO_PALET, rellena el campo FTP_PROGENFARD de la tabla FP.

El valor del programa de enfardado depende de:

- Si TIPO_PALET = Media Paleta el valor será = 1.
- En cualquier otro caso, el valor será = 6.

6.5.4 Gestión de parámetros Samovi

Las referencias de las líneas de Samovi residen en un único autómata.

En el autómata, la línea de envasado se almacena como un número. Para las equivalencias necesarias, se utilizarán las tablas PARAM_EQUIVALENCIAS y LINEAS_ENV.

Tal y como ya se ha descrito, existe un proceso de comunicaciones desarrollado en Visual C++, llamado COM_SAMO, con estructura de máquina de estados que se ejecuta periódicamente.

El proceso de gestión de parámetros locales en Samovi necesita conexión a la base de datos y conexión al autómata de Samovi (tecnología Omron) vía OPC.

6.5.4.1 Objetivo

Este proceso de comunicaciones tiene como objetivo acceder vía OPC al autómata de Samovi, y realizar un protocolo de comunicaciones que permita:

- Capturar los valores de las referencias de Samovi.
- Volcar modificaciones a las referencias.

6.5.4.2 Funcionamiento

Las solicitudes de lectura o escritura de una FP o FTP se realizan desde la aplicación en Visual Basic, a través de la pantalla de gestión de parámetros locales.

Al realizar estas solicitudes, se insertan registros en la tabla llamada COLA_MENSAJES. Estas solicitudes pueden ser petición de lectura o petición de escritura de una referencia.

El proceso de comunicaciones COM_SAMO, que tiene conexión al autómeta vía OPC procesa esas solicitudes. Para ello, chequea periódicamente si hay registros en COLA_MENSAJES con estado 'I' (iniciado).

Al comenzar el procesamiento, se actualiza el estado a 'P' (procesando).

Al finalizar el procesamiento correctamente, se escribe el estado a 'T' (terminado). Si ha habido algún error, el estado se escribe a 'E' (error).

6.5.4.2.1 Funciones

En este apartado se describen las funciones relevantes del proceso de comunicación con Samovi.

- ConexionPLC

En esta función se realizan los pasos necesarios para realizar la conexión con el PLC mediante la OPC Custom Interface v2.0.

- Plc_Leeltems

Devuelve datos leídos vía OPC.

- ConsultarCM

El proceso de comunicación con Samovi, desarrollado en C++, recibe peticiones de la aplicación en Visual Basic.

Las peticiones se gestionan a través de una cola de mensajes, existiendo para ello la tabla COLA_MENSAJES en la base de datos.

Un mensaje tiene:

- Tipo de mensaje:
 - Capturar referencia desde el autómeta de Samovi.
 - Volcar referencia al autómeta de Samovi.
 - Estado:
 - I: iniciado.
 - P: procesando.
 - T: terminado correctamente.
 - E: terminado con error.
- LeerReferenciaSamovi

Indica al autómatas qué referencia se desea obtener.

Cuando el autómatas responde actualizando la palabra correspondiente, el proceso de comunicaciones lee los items de la referencia y los almacena en la base de datos.

- VolcarReferenciaSamovi

Esta función indica al autómatas qué referencia se desea modificar.

El proceso de comunicaciones escribe los nuevos valores de la referencia en las posiciones intermedias. Posteriormente, escribe el índice de la referencia.

Finalmente, el proceso escribe el flag de escritura, que indica al autómatas que escriba los nuevos valores desde las posiciones intermedias a las posiciones de la referencia indicada por el índice.

Adicionalmente a este protocolo de comunicaciones para la lectura y escritura de referencias en el autómatas de Samovi, el proceso COM_SAMO también incluye la función de lectura de items de diagnosis.

6.5.5 Gestión de parámetros maletas

Las líneas de paletizado de Maletas están monitorizadas por aplicaciones Albatros, que almacenan los parámetros de referencias en bases de datos locales Access.

En el caso de Maletas 2, existe además una aplicación ProTools de Siemens, que también toma los datos de otra base de datos Access local al panel en el que reside.

6.5.5.1 Funciones

La funcionalidad de gestión de parámetros locales se realiza desde Visual Basic e implica acceso a bases de datos Access y Oracle. Estas funcionalidades permiten capturar referencias, modificarlas, duplicarlas y borrarlas.

Las funciones relevantes son:

- ObtenerTablasAccess

Abre una base de datos Access y devuelve una colección con los nombres de sus tablas.

- CapturarPLtblCodigos, CapturarPLAlbatros

Abre la base de datos Access que corresponde, extrae valores desde la tabla de Access (Albatros) o del registro (tblCodigos) y los inserta en la tabla PARAM_VALORES de Oracle.

- VolcarPLtblCodigos, VolcarPLAlbatros

Al volcar una referencia nueva, se crea la tabla (Albatros) o un registro (tblCodigos) correspondiente a partir de los valores almacenados en PARAM_VALORES.

- DuplicarPLtblCodigos, DuplicarPLAlbatros

Generan una referencia nueva en Oracle, a partir de una referencia existente, insertando en PARAM_VALORES.

- BorrarPLAlbatros, BorrarPLAlbatros

Borra los parámetros locales asociados a una referencia, tanto en Oracle como en Access.

6.5.5.2 Gestión de datos en Ocme-Albatros

Las zonas controladas por Ocme (“Albatros”) funcionan contra una base de datos Access local, existiendo una tabla por cada referencia.

Se utilizarán las tablas PARAM_LOCALES y PARAM_VALORES, para la creación, modificación y borrado de las tablas de Access.

En la obtención de datos locales, se consultará la tabla, teniendo en cuenta que cada fila de la tabla Access es un parámetro, y se actualiza la tabla PARAM_VALORES con los valores obtenidos.

En la escritura de parámetros locales, se consultará la tabla PARAM_VALORES y se actualiza la correspondiente tabla local. En caso de que no exista, se crea una nueva tabla.

6.5.5.3 Gestión de datos en ProTools de Siemens

La aplicación ProTools funciona contra una base de datos Access local, existiendo una tabla “tblCodigos” que almacena los parámetros locales asociados a las fichas de paletizado.

Se utilizan las tablas PARAM_LOCALES y PARAM_VALORES, para la modificación de los valores almacenados en la tabla “tblCodigos”.

En la obtención de datos locales, se consultará la tabla, teniendo en cuenta que cada columna es un parámetro, y se actualizará la tabla PARAM_VALORES con los valores obtenidos.

En la escritura de parámetros locales, se consultará la tabla PARAM_VALORES y se actualizará el registro correspondiente a la tabla “tblCodigos”.

6.5.6 Gestión de parámetros Fanuc

Las líneas de paletizado Fanuc están monitorizadas por aplicaciones ProTools de Siemens, que toman los datos de una base de datos Access local al panel en el que reside. También existe una tabla “tblCodigos”, y el tratamiento es similar al descrito en el apartado de Maletas para el caso de ProTools.

6.5.7 Gestión parámetros Emmetti

Las líneas de paletizado Emmetti tendrán un panel de operador de Siemens, que será accesible por ethernet. Las aplicaciones de los paneles están desarrolladas con ProTools, y permiten crear, modificar y eliminar los datos de paletizado de las referencias.

En estas líneas de paletizado, no existen bases de datos Access locales. En su lugar, el intercambio de datos se realizará mediante la funcionalidad Import/Export a archivos “.csv” de ProTools.

6.5.7.1 Modificaciones en los paneles Siemens

La exportación de los datos de los formatos se realiza a un fichero “recetaA.csv”, cuya ubicación está en el directorio compartido D:\EmmettiA; por tanto, se ha configurado la pantalla para que esté dentro del grupo de trabajo de “Paletizado”.

Los pasos que se han seguido son:

1. Configuración en el servidor:
 - a) Se verifica el usuario de acceso.
 - b) Se comparte el directorio donde se va a ubicar el fichero.
2. Configuración en pantalla
 - a) Configurar la dirección IP en Panel de Control \ Network.
 - b) Configurar el usuario con el que se va a conectar al servidor.
 - c) Hacer ping hacia el servidor y viceversa.
 - d) Abrir el Explorer y ejecutar “\\server-operaciones\emmettiA”.

6.5.7.1.1 Variables para la exportación / importación

Para poder leer y escribir parámetros locales de los formatos de paletización de las líneas de Emmetti, se ha recurrido a las funciones de Exportar_datos e Importar_datos con las que cuenta la pantalla.

Realmente, los datos de los formatos residen en un fichero de datos, donde están configurados como si se tratasen de registros de una receta. Estos datos no son accesibles y se almacenan en la tarjeta CF de la pantalla.

Por tanto, se ha programado la Exportación e Importación de dichos datos. Estas operaciones se pueden lanzar de forma manual desde el propio panel táctil, o bien de forma automática desde la aplicación Visual Basic del SGCP.

6.5.7.1.2 Variables para la exportación / importación

Para cada una de las operaciones se han creado las variables:

- inExport_Orden: DB126.DBX0.1, bit para hacer la exportación de datos.
- inExport_Estado: DB126.DBW2, palabra donde se escribe el estado de la operación.
- inImport_Orden: DB126.DBX1.1, bit para hacer la importación de datos.
- inImport_Estado: DB126.DBW4, palabra donde se escribe el estado de la operación.

Estas variables son leídas del PLC de la Pieri, correspondiente a la línea. En la rutina FC125 de dicho PLC está programada la lógica de lanzamiento de orden de exportación o importación.

6.5.7.1.3 Función de exportación del panel Siemens

Cada vez que la variable inExport_Orden cambie de estado (de 0 a 1 ó de 1 a 0), se lanzará la función: "Exportar_registros_de_datos".

Los parámetros de dicha función son:

- Número / nombre de receta:1
- Número / nombre de registro de datos: 0
- Nombre de archivo: "\\svr-operaciones\emmettia\recetaA.csv"
- Sobrescribir sin consulta.
- Emitir avisos de estado: On
- Valor de retorno de estado: inExport_Estado.

Cuando el bit inExport_Orden cambia de valor se ejecutará la función de exportación, momento en el cual se abrirá el fichero de registros de receta (formatos), y se exportarán todos los datos al fichero destino.

En todo momento se indicará en pantalla si la operación se ha realizado correctamente, o si ha habido un error.

6.5.7.1.4 Función de importación del panel Siemens

Cada vez que la variable inImport_Orden cambie de estado, de 0 a 1 ó de 1 a 0, se lanzará la función: "Importar_registros_de_datos".

Los parámetros de dicha función son:

- Nombre de archivo: "\\svr-operaciones\emmettia\recetaA.csv"
- Número / nombre de registro de datos: 0

- Sobrescribir sin consulta.
- Emitir avisos de estado: On
- Valor de retorno de estado: inImport_Estado.

Cuando el bit inImport_Orden cambia de valor se ejecutará la función de importación, momento en el cual se abrirá el fichero de registros de receta (formatos), y se importarán todos los datos.

6.5.7.2 Funciones

La funcionalidad de gestión de parámetros locales se realiza desde Visual Basic e implica tratamiento de ficheros .csv. Estas funcionalidades permiten capturar referencias, modificarlas, duplicarlas y borrarlas.

Las funciones relevantes son:

- CapturarPLEmmetti

Convierte los datos almacenados en la tabla intermedia PARAM_EMMETTI (que tiene los valores enteros largos del ".csv") y los inserta en la tabla PARAM_VALORES de Oracle.

- VolcarPLEmmetti

Una vez esté completa la tabla PARAM_EMMETTI, se generará el fichero "csv" y se insertará un mensaje en COLA_MENSAJES' para que el programa llamado COM_ENF escriba el flag al PLC.

- DuplicarPLEmmetti

Generan una referencia nueva en Oracle, a partir de una referencia existente, insertando en PARAM_EMMETTI.

- BorrarPLEmmetti

Borra los parámetros locales asociados a una referencia, tanto en PARAM_VALORES como en PARAM_EMMETTI.

- ConvertirCsvAParamEmmetti

Inserta en la tabla PARAM_EMMETTI los valores extraídos del fichero "csv".

- ConvertirValoresAParamEmmetti

Inserta en la tabla PARAM_EMMETTI a partir de la tabla PARAM_VALORES.

- GenerarFicheroEmmetti

Genera fichero ".csv" a partir de la tabla PARAM_EMMETTI.

- ConvertirBitsANumero

Convierte una cadena de bits a un número, con la posibilidad de indicarle bit de inicio y bit de fin.

- ObtenerBitsParciales

Devuelve un nuevo string de bits sobre un string de bits existente, sobrescribiendo los bits desde el bit de inicio, y durante un cierto número de bits.

Para clarificar la secuencia de pasos en el proceso de transferencia de datos con los paneles de Siemens, se detallan a continuación los procesos de captura y volcado de parámetros locales en Emmetti.

6.5.7.2.1 Secuencia de captura de datos

La secuencia de pasos en la captura de datos desde la aplicación ProTools a la base de datos Oracle es la siguiente:

1. Se escribe un bit en el autómata, para generar un trigger de exportación en ProTools.
2. Una vez finalizada la exportación, se genera un fichero de texto, en la ruta por defecto de la herramienta ProTools.
3. Se abre el fichero para su tratamiento:
 - a) Cada columna del fichero es una referencia de paletizado.
 - b) En el fichero “.csv”, cada celda es un entero largo.
 - c) Se realiza la descomposición de los enteros largos en los tipos correspondientes (bits, enteros, cadenas, etc).
4. Se almacenan los datos tratados en la base de datos.

6.5.7.2.2 Secuencia de volcado de datos

La secuencia de pasos en el volcado de datos desde la base de datos Oracle a la aplicación ProTools es la siguiente:

1. Se genera un fichero de texto, en la ruta por defecto de la herramienta ProTools, a partir de los valores de la tabla PARAM_EMMETTI:
 - a) Se consultarán los datos en la base de datos Oracle.
 - b) Se realizará la composición de estos datos en enteros largos que acepta la aplicación ProTools.
 - c) Cada columna del fichero “.csv” será una referencia de paletizado.
2. El fichero generado tiene las siguientes características:
 - a) Cada columna del fichero es una referencia de paletizado.
 - b) En el fichero “.csv”, cada celda es un entero largo.

3. Se habilitará un bit en el autómata, para generar un trigger de importación en ProTools.
4. El trigger activará la función de importación de datos de ProTools.

6.6 **MÓDULO DE ESTADO Y DIAGNOSIS**

El módulo de estado y diagnosis mostrará el estado de los dispositivos relevantes que componen cada línea.

6.6.1 Componentes del módulo

El módulo tendrá los siguientes componentes:

- Procesos desarrollados en C++, que comunican vía OPC con los autómatas.
- Pantallas de la aplicación de supervisión, que consistirán en sinópticos.

6.6.2 Modelo de datos

La tabla OPC_ITEMS, descrita en el apartado 6.3.3.3, es utilizada por el módulo de estado y diagnosis para la obtención de los datos.

Adicionalmente, para la visualización de los datos en la aplicación de supervisión son necesarias las siguientes tablas:

6.6.2.1 Tabla DIAGNOSIS_GRUPOS (DG)

Descripción: Asociación de un grupo de ítems OPC de diagnosis a cada línea. Cada línea tendrá asociado un solo grupo OPC de diagnosis y esta información será utilizada por la aplicación de supervisión.

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|--------------|-----------|-----------------------------------|----------------------------------|
| DG_LP_COD | NUMBER(3) | Clave primaria Ref: LINEAS_PAL | Código de la línea de Paletizado |
| DG_OG_OS_COD | NUMBER(3) | Clave primaria Ref: OPC_GROUPS | Código de servidor OPC |
| DG_OG_COD | NUMBER(3) | Clave primaria Ref: OPC_GROUPS | Código del grupo OPC |

Tabla 6.19: Tabla DIAGNOSIS_GRUPOS

6.6.3 Procesos

El módulo de estado y diagnosis obtiene los datos de los autómatas a través del módulo de comunicaciones con autómatas. Por tanto, estos procesos están

estructurados de la forma genérica ya explicada: los procesos de comunicaciones están desarrollados en Visual C++ y tienen una estructura de máquina de estados que se ejecutan periódicamente.

6.6.3.1 Dimensionamiento del número de procesos

Teniendo en cuenta las consideraciones para el dimensionamiento del número de procesos comentadas en el apartado 6.3.2.2, se ha creado un proceso de comunicaciones por cada autómatas concentrador.

Los procesos asociados al módulo de estado y diagnóstico son: COM_PLC_3... COM_PLC_9, uno por cada línea de paletizado. Estos procesos comunican con los autómatas concentradores (tecnología Siemens):

| NOMBRE | COMUNICACIÓN | FUNCIONALIDAD |
|-----------|-----------------------------|---|
| COM_PLC_3 | PLC Pieri – L3 SAMOVI | Proceso que obtiene los datos de diagnóstico de la Pieri de SAMOVI |
| COM_PLC_4 | PLC Pieri – L4 MALETAS 1 | Proceso que obtiene los datos de diagnóstico de la Pieri y del Robot de MALETAS 1 |
| COM_PLC_5 | PLC Pieri – L5 MALETAS 2 | Proceso que obtiene los datos de diagnóstico de la Pieri, del Robot y del scalextric de MALETAS 2 |
| COM_PLC_6 | PLC Pieri – L6 FANUC 1 | Proceso que obtiene los datos de diagnóstico de la Pieri, del Robot y del transporte de FANUC 1 |
| COM_PLC_7 | PLC Pieri – L7 FANUC 2 | Proceso que obtiene los datos de diagnóstico de la Pieri, del Robot y del transporte de FANUC 2 |
| COM_PLC_8 | PLC Pieri – L8 EMMETTI A | Proceso que obtiene los datos de diagnóstico de la Pieri, del Robot y del transporte de EMMETTI A |
| COM_PLC_9 | PLC Pieri – L9 EMMETTI B | Proceso que obtiene los datos de diagnóstico de la Pieri, del Robot y del transporte de EMMETTI B |

Tabla 6.20: Procesos asociados al módulo de estado y diagnóstico

Como ya se ha explicado, los autómatas concentradores de cada línea son los autómatas de enfardado (Pieri).

6.6.3.2 Estructura interna

Los procesos de comunicaciones se han desarrollado en Visual C++ y tienen una estructura de máquina de estados que se ejecuta periódicamente (ver apartado 6.1.4.2).

Los procesos de estado y diagnosis necesitan conexión a la base de datos y conexión a los autómatas vía OPC.

6.6.3.3 Objetivo

Acceder vía OPC a los diferentes autómatas y actualizar los valores leídos en la base de datos.

6.6.3.4 Funciones

En este apartado se describen las funciones relevantes de los procesos de diagnosis.

- ConexionPLC

En esta función se realizan los pasos necesarios para realizar la conexión con el PLC mediante la OPC Custom Interface v2.0.

- Plc_Leeltems

Devuelve datos leídos vía OPC.

- LeerValores

Se encarga de la llamada a la función de lectura vía OPC y la actualización en la tabla OPC_ITEMS de los valores leídos. Las pantallas de supervisión obtendrán los datos directamente de la tabla OPC_ITEMS.

6.7 **MÓDULO DE ENFARDADO**

El módulo de enfardado permitirá visualizar y configurar los diferentes programas de enfardado de cada enfardadora.

6.7.1 Componentes del módulo

El módulo tiene los siguientes componentes:

- Procesos desarrollados en C++, que comunican vía OPC con los autómatas.
- Pantallas de la aplicación de supervisión, para la visualización y configuración de los programas de enfardado.

6.7.2 Modelo de datos

La tabla OPC_ITEMS (ver apartado 6.3.3.3), tiene un campo donde se almacena el valor leído de cada ítem. El módulo de enfardado tiene definidos los ítems.

Adicionalmente, son necesarias otras tablas:

6.7.2.1 Tabla COLA_MENSAJES (CM)

Descripción: Tabla de mensajes entre aplicaciones C y Vbasic.

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|------------------|---------------|----------------|---|
| CM_FECHA_PET | DATE | Clave Primaria | Fecha solicitud de la petición |
| CM_COD_PET | NUMBER(4) | Clave Primaria | Código de la petición |
| CM_COD_ELEM | NUMBER(4) | Clave Primaria | Código del elemento |
| CM_ESTADO | VARCHAR2(1) | | I (iniciado), P (procesando), T (terminado correcto), E (terminado con error) |
| CM_MENSAJE | VARCHAR2(200) | | Mensaje |
| CM_FECHA_PROCESA | DATE | | Fecha de procesamiento de la petición |

Tabla 6.21: Tabla COLA_MENSAJES

6.7.2.2 Tabla ENFARDADORAS (EN)

Descripción: Tabla que almacena las enfardadoras existentes en paletizado. Varias líneas pueden compartir una misma enfardadora, por ejemplo en Samovi.

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|-----------|--------------|----------------|---|
| EN_COD | NUMBER(3) | Clave Primaria | Código de la enfardadora |
| EN_NOMBRE | VARCHAR2(30) | | Nombre de la enfardadora |
| EN_LI_COD | NUMBER(3) | | Línea a la que pertenece la enfardadora |

Tabla 6.22: Tabla ENFARDADORAS

6.7.2.3 Tabla ENF_PARAMETROS (EP)

Descripción: Tabla que almacena los parámetros de cada programa de enfardado.

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|------------|--------------|--|---|
| EP_EN_COD | NUMBER(3) | Clave Primaria Ref: ENFARDADORAS | Código de la enfardadora |
| EP_COD | NUMBER(3) | Clave Primaria | Código del parámetro |
| EP_NOMBRE | VARCHAR2(50) | | Nombre del parámetro |
| EN_LIM_SUP | NUMBER(7,1) | | Límite superior del parámetro, si es numérico |
| EN_LIM_INF | NUMBER(7,1) | | Límite inferior del parámetro, si es numérico |
| EN_TD_COD | NUMBER(3) | Ref: TIPOS_DATOS | Código del tipo de dato (bit, string, entero, flotante) |

Tabla 6.23: Tabla ENF_PARAMETROS

6.7.2.4 Tabla ENF_VALORES (EV)

Descripción: Tabla que almacena los valores de cada programa de enfardado.

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|--------------|-----------|---------------------------------------|--------------------------|
| EV_EP_EN_COD | NUMBER(3) | Clave Primaria Ref: ENF_PARAMETROS | Código de la enfardadora |

| | | | |
|-------------|--------------|---------------------------------------|----------------------|
| EV_EP_COD | NUMBER(3) | Clave Primaria Ref: ENF_PARAMETROS | Código del parámetro |
| EV_PROGRAMA | NUMBER(3) | Clave Primaria | Código del programa |
| EP_VALOR | VARCHAR2(50) | | Valor del parámetro. |

Tabla 6.24: Tabla ENF_VALORES

6.7.3 Procesos

El módulo de enfiado obtiene los datos de los autómatas a través del módulo de comunicaciones con autómatas. Por tanto, estos procesos tienen una estructura de máquina de estados que se ejecuta periódicamente.

6.7.3.1 Dimensionamiento del número de procesos

Teniendo en cuenta las consideraciones para el dimensionamiento del número de procesos, se ha creado un proceso de comunicaciones por cada autómata de enfiado.

Los procesos asociados al módulo de enfiado son COM_ENF_3... COM_ENF_9, uno por cada línea. Estos procesos comunican con los autómatas de enfiado (tecnología Siemens):

| NOMBRE | COMUNICACIÓN | FUNCIONALIDAD |
|-----------|--------------------------|---|
| COM_ENF_3 | PLC Pieri – L3 SAMOVI | Proceso que lee y escribe los parámetros de programas de enfiado de SAMOVI |
| COM_ENF_4 | PLC Pieri – L4 MALETAS 1 | Proceso que lee y escribe los parámetros de programas de enfiado de MALETAS1 |
| COM_ENF_5 | PLC Pieri – L5 MALETAS 2 | Proceso que lee y escribe los parámetros de programas de enfiado de MALETAS2 |
| COM_ENF_6 | PLC Pieri – L6 FANUC 1 | Proceso que lee y escribe los parámetros de programas de enfiado de FANUC 1 |
| COM_ENF_7 | PLC Pieri – L7 FANUC 2 | Proceso que lee y escribe los parámetros de programas de enfiado de FANUC 2 |
| COM_ENF_8 | PLC Pieri – L8 EMMETTI A | Proceso que lee y escribe los parámetros de programas de enfiado de EMMETTI A |
| COM_ENF_9 | PLC Pieri – L9 EMMETTI B | Proceso que lee y escribe los parámetros de programas de enfiado de EMMETTI B |

Tabla 6.25: Procesos asociados al módulo de enfiado

6.7.3.2 Estructura interna

Los procesos de comunicaciones se han desarrollado en Visual C++ y tienen una estructura de máquina de estados que se ejecuta periódicamente (ver apartado 6.1.4.2).

Los procesos de enfardado necesitan conexión a la base de datos y conexión a los autómatas vía OPC.

6.7.3.3 Objetivo

Este tipo de procesos de comunicaciones tiene como objetivo acceder vía OPC a los diferentes autómatas de enfardado, y realizar un protocolo de comunicaciones que permita:

- Capturar los programas de enfardado.
- Volcar modificaciones a los programas de enfardado.

6.7.3.4 Funcionamiento

Las solicitudes de lectura o escritura de un programa de enfardado se realizan desde la aplicación en Visual Basic.

Al realizar estas solicitudes, se insertan registros en la tabla llamada COLA_MENSAJES. Estas solicitudes pueden ser petición de lectura o petición de escritura de un programa de enfardado.

Los procesos en C++ que tienen conexión a los autómatas vía OPC procesan esas solicitudes. Para ello, chequean periódicamente si hay registros en la COLA_MENSAJES con estado 'I' (iniciado).

Al comenzar el procesamiento, pondrán el estado a 'P' (procesando).

Al finalizar el procesamiento correctamente, escribirán el estado a 'T' (terminado). Si ha habido algún error, el estado se escribe a 'E' (error).

6.7.3.4.1 Funciones

En este apartado se describen las funciones relevantes de los procesos de comunicación con enfardadoras.

- ConexionPLC

En esta función se realizan los pasos necesarios para realizar la conexión con el PLC mediante la OPC Custom Interface v2.0.

- Plc_Leeltems

Devuelve datos leídos vía OPC.

- ConsultarCM

Los procesos de comunicaciones con las enfardadoras, desarrollados en C++, reciben peticiones de la aplicación en Visual Basic.

Las peticiones se gestionan a través de una cola de mensajes, existiendo para ello la tabla COLA_MENSAJES en la base de datos.

Cuando el usuario de la aplicación de supervisión solicita una lectura de un programa de enfardado, se inserta un registro en la cola de mensajes.

Un mensaje tiene:

- Tipo de mensaje:
 - Capturar programa de enfardado.
 - Volcar programa de enfardado.
 - Importar fichero "csv".
 - Exportar fichero "csv".
- Estado:
 - I: iniciado.
 - P: procesando.
 - T: terminado correctamente.
 - E: terminado con error.

- LeerProgramaEnfardado

Indica al autómatas qué programa de enfardado desea obtener.

Cuando el autómatas responde actualizando la palabra correspondiente, el proceso de comunicaciones lee los ítems del programa de enfardado y los almacena en la base de datos.

- VolcarProgramaEnfardado

Esta función indica al autómatas qué programa de enfardado desea modificar.

Cuando el autómatas responde actualizando la palabra correspondiente, el proceso de comunicaciones escribe los nuevos valores del programa de enfardado en las posiciones intermedias.

Finalmente, el proceso escribe el flag de escritura, que indica al autómatas que escriba los nuevos valores desde las posiciones intermedias en las posiciones del programa correspondiente.

6.7.3.4.2 Observaciones

En los programas de los autómatas de enfardado, existen unas posiciones de memoria sobre las que trabaja la pantalla táctil situada en cada cuadro eléctrico.

Se han aprovechado dichas posiciones de memoria para realizar la escritura y la lectura de los programas de enfardado, de forma que:

- No ha sido necesario modificar los programas de los autómatas.
- Se minimizan las comunicaciones, al no comunicar directamente con todos los ítems de todos los programas de enfardado.

La estructura de datos y las funciones implicadas en los programas de los autómatas de Siemens están explicadas en el apartado 6.4, modificaciones en los autómatas.

6.8 **MÓDULO DE ETIQUETADO**

El módulo de etiquetado implementa la comunicación con las etiquetadoras a través de ficheros. Permitirá visualizar la información capturada de las etiquetadoras y enviar los productos a todos los ordenadores de etiquetadoras.

6.8.1 Componentes del módulo

El módulo tiene los siguientes componentes:

- Proceso “COM_ETIQ” desarrollado en C++, que comunica a través de ficheros con las etiquetadoras.
- Pantallas de la aplicación de supervisión, para la visualización de la información capturada de las etiquetadoras.

6.8.2 Modelo de datos

Son necesarias las siguientes tablas:

6.8.2.1 Tabla ETIQUETADORAS (EQ)

Descripción: Tabla que almacena las etiquetadoras existentes en paletizado. Un mismo ordenador puede tener conectadas varias etiquetadoras.

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|-------------------|--------------|----------------|--|
| EQ_COD | NUMBER(3) | Clave Primaria | Código de la etiquetadora |
| EQ_NOMBRE | VARCHAR2(30) | | Nombre de la etiquetadora |
| EQ_PUESTO | VARCHAR2(30) | | Nombre del ordenador al que está asociada |
| EQ_NODO | NUMBER(3) | | Nodo de la etiquetadora |
| EQ_LP_COD | NUMBER(3) | | Código de la línea de paletizado a que pertenece |
| EQ_NODO_ORDENADOR | NUMBER(3) | | Código del nodo del ordenador que controla las etiquetadoras |
| EQ_EST | VARCHAR2(50) | | Estado (mensaje de texto) capturado de las etiquetadoras |

Tabla 6.26: Tabla ETIQUETADORAS

6.8.2.2 Tabla ETIQ_PARAMETROS (EM)

Descripción: Tabla que almacena las rutas de los ficheros de intercambio con las etiquetadoras.

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|-----------------|---------------|---|---|
| EM_EQ_COD | NUMBER(3) | Clave Primaria Ref: ETIQUETADORAS | Código de la etiquetadora |
| EM_RUTA | VARCHAR2(200) | | Ruta del fichero. |
| EM_RUTA_PROC | VARCHAR2(200) | | Ruta de procesamiento de los ficheros, para evitar diferentes sistemas accediendo al mismo fichero a la vez |
| EM_EF_COD | NUMBER(3) | Ref: ETIQ_FICHEROS Clave Primaria | Código del tipo de fichero (productos, configuración, producción, diagnóstico, etc) |
| EM_PERIODO | NUMBER(6) | | Periodo (en minutos) |
| EM_ULTIMA_FECHA | DATE | | Fecha del último tratamiento del fichero |
| EM_FLAG_FORZADO | NUMBER(1) | | Indica si se ha activado el forzado (1) o no (0) desde la aplicación de supervisión |
| EM_HABILITACION | NUMBER(1) | | Indica si el tratamiento del fichero está habilitado (1) o no (0) |
| EM_BORRAR | NUMBER(1) | | Indica si tras el tratamiento del fichero se debe borrar o se deja una copia de seguridad en disco |

Tabla 6.27: Tabla ETIQ_PARAMETROS

6.8.2.3 Tabla ETIQ_FICHEROS (EF)

Descripción: Tabla que almacena los tipos de ficheros en la comunicación con las etiquetadoras (productos, configuración, producción, diagnóstico).

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|-----------|--------------|----------------|----------------------------|
| EF_COD | NUMBER(3) | Clave Primaria | Código del tipo de fichero |
| EF_NOMBRE | VARCHAR2(50) | | Nombre del tipo de fichero |

Tabla 6.28: Tabla ETIQ_FICHEROS

6.8.2.4 Tabla ETIQ_PRODUCION (ER)

Descripción: Tabla que almacena los datos de producción enviados por las etiquetadoras.

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|-----------------|--------------|---|-----------------------------------|
| ER_EQ_COD | NUMBER(3) | Clave Primaria Ref: ETIQUETADORAS | Código de la etiquetadora |
| ER_PALET | NUMBER(7) | Clave Primaria | Código del palet |
| ER_CODIGO_SAP | VARCHAR2(9) | | Código SAP del producto |
| ER_DUN14 | VARCHAR2(14) | | Código DUN de la etiqueta |
| ER_DESCRIPCION1 | VARCHAR2(23) | | Descripción de la etiqueta |
| ER_DESCRIPCION2 | VARCHAR2(23) | | Descripción de la etiqueta |
| ER_CANTIDAD | NUMBER(6) | | Cantidad |
| ER_LOTE | VARCHAR2(8) | | Lote |
| ER_SSCC | VARCHAR2(18) | | Código SSCC |
| ER_FECHA | DATE | | Fecha |
| ER_NOMINAL | VARCHAR2 (2) | | SI/NO ha impreso cantidad nominal |
| ER_APLICADORA | VARCHAR2 (2) | | SI/NO ha impreso por etiquetadora |
| ER_MARCA | NUMBER(1) | | Un dígito (0/9) |

Tabla 6.29: Tabla ETIQ_PRODUCION

6.8.2.5 Tabla ETIQ_DIAGNOSTICO (ED)

Descripción: Tabla que almacena los diagnósticos de cada etiquetadora.

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|-----------|-----------|---|---------------------------|
| ED_EQ_COD | NUMBER(3) | Clave Primaria Ref: ETIQUETADORAS | Código de la etiquetadora |

| | | | |
|----------------|--------------|----------------|--|
| ED_FECHA | DATE | Clave Primaria | Fecha |
| ED_ESTADO | VARCHAR2(2) | | Estado de la etiquetadora 'OK' o 'KO' |
| ER_DESCRIPCION | VARCHAR2(30) | | Sólo en caso de que la etiquetadora esté en error |

Tabla 6.30: Tabla ETIQ_DIAGNOSTICO

6.8.2.6 Tabla ETIQ_CONFIGURACION (EC)

Descripción: Tabla que almacena la configuración de cada etiquetadora.

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|----------------------|--------------|------------|---|
| EC_FECHA | DATE | | Fecha |
| EC_ID | VARCHAR2(20) | | Código de etiquetadora |
| EC_PUERTO_METEC_1 | VARCHAR2(20) | | Configuración Puerto Metec 1 |
| EC_PUERTO_METEC_2 | VARCHAR2(20) | | Configuración Puerto Metec 2 |
| EC_PUERTO_ZEBRA | VARCHAR2(20) | | Configuración Puerto Zebra |
| EC_PUERTO_SAI | VARCHAR2(20) | | Configuración Puerto Sai |
| EC_PUERTO_SCANNER1 | VARCHAR2(20) | | Configuración Puerto Scanner1 |
| EC_PUERTO_SCANNER2 | VARCHAR2(20) | | Configuración Puerto Scanner2 |
| EC_HORA_JULIANA | VARCHAR2(20) | | Hora Juliana |
| EC_MINUTO_JULIANA | VARCHAR2(20) | | Minuto Juliano |
| EC_CONSECUTIVO | VARCHAR2(20) | | Contador consecutivo de la impresora |
| EC_JULIANAACTIVA | VARCHAR2(20) | | |
| EC_ASIGNACIÓN | VARCHAR2(20) | | |
| EC_CONSECUTIVO | VARCHAR2(20) | | Contador consecutivo de la impresora |
| EC_NODO_ETIQUETADORA | NUMBER(3) | | Nodo de la etiquetadora |
| EC_PATH_PRODUCTOS | VARCHAR2(20) | | Path para ficheros |
| EC_TIME_PRODUCTOS | VARCHAR2(20) | | Periodo para ficheros |
| EC_PATH_PRODUCCION | VARCHAR2(20) | | Path para ficheros |
| EC_TIME_PRODUCCION | VARCHAR2(20) | | Periodo para ficheros |
| EC_PATH_DIAGNOSTICO | VARCHAR2(20) | | Path para ficheros |

| | | | |
|-----------------------|--------------|--|-----------------------|
| EC_TIME_DIAGNOSTICO | VARCHAR2(20) | | Periodo para ficheros |
| EC_PATH_CONFIGURACION | VARCHAR2(20) | | Path para ficheros |
| EC_TIME_CONFIGURACION | VARCHAR2(20) | | Periodo para ficheros |
| EC_PATHALLPRODUCCION | VARCHAR2(20) | | |
| EC_PATHCODESAP | VARCHAR2(20) | | |
| EC_PATHMARCAPROD | VARCHAR2(20) | | |
| EC_CONTRJETAS | VARCHAR2(20) | | |
| EC_ADRESSMPI1 | VARCHAR2(20) | | |
| EC_SEGMENTMPI1 | VARCHAR2(20) | | |
| EC_RACKMPI1 | VARCHAR2(20) | | |
| EC_SLOTMPI1 | VARCHAR2(20) | | |
| EC_MEMORIADBMPI1 | VARCHAR2(20) | | |
| EC_ADRESSMPI2 | VARCHAR2(20) | | |
| EC_SEGMENTMPI2 | VARCHAR2(20) | | |
| EC_RACKMPI2 | VARCHAR2(20) | | |
| EC_SLOTMPI2 | VARCHAR2(20) | | |
| EC_MEMORIADBMPI2 | VARCHAR2(20) | | |

Tabla 6.31: Tabla ETIQ_CONFIGURACION

6.8.3 Procesos

El módulo de etiquetado comunica con las etiquetadoras a través de ficheros:

- Envía ficheros de texto con la información de los productos a todos los PC de etiquetadoras (archivo productos1.dat).
- Recoge de los PC de etiquetadoras los ficheros de:
 - Configuración: el sistema captura los valores de configuración de la etiquetadora.
 - Diagnóstico: el sistema captura si la etiquetadora está funcionando correctamente o no.
 - Producción: El sistema captura información de todos los palet etiquetados.
 - Marca de producción: actualiza el valor de la marca de producción en los datos de producción.
 - Código SAP modificado: actualiza el valor del campo “cantidad mínima” en la tabla ETIQ_PRODUCTOS cuando este valor se modifica en un PC de etiquetado.

6.8.3.1 Rutas

Las rutas de intercambio de ficheros son configurables. Existe una pantalla en la que se permite configurar las rutas de los ficheros y el periodo.

Para cada ordenador, las rutas configuradas por defecto son:

- C:\inplanta\FicherosActuales
En esta ruta el sistema de etiquetado escribe sus ficheros y aquí recibe el fichero "productos1.dat".
- C:\inplanta\FicherosProcesamiento
El sistema mueve a esta ruta los ficheros que va a tratar.

Hay cuatro ordenadores de etiquetado, y cada uno de ellos se encarga del etiquetado de dos líneas de paletizado.

6.8.3.2 Ficheros de datos recibidos por el módulo de etiquetado

6.8.3.2.1 Fichero de producción

1. Mueve el fichero origen (ruta configurable) de producción que contienen los ordenadores de control de las etiquetadoras a una ruta de procesados. El objetivo de esta acción es minimizar la posibilidad de que ambos sistemas intenten acceder al fichero a la vez y se produzcan errores.
2. Se abre el fichero y se extraerán valores.
3. Se almacena la información en la tabla ETIQ_PRODUCION.
4. Una vez finalizado el procesamiento, se borra el fichero o se conserva renombrado a la fecha de procesamiento, según esté configurado.

6.8.3.2.2 Fichero de marca de producción

El procesamiento del fichero es similar al descrito para el fichero de producción.

1. Se actualiza el campo que almacena la marca de producción en la tabla ETIQ_PRODUCION, para el código SAP y el SSCC indicados en el fichero.
2. Una vez finalizado el procesamiento, se borra el fichero o se conserva renombrado a la fecha de procesamiento, según esté configurado.

6.8.3.2.3 Fichero de diagnóstico

El procesamiento del fichero es similar al descrito para el fichero de producción.

1. Se almacena la información en la tabla ETIQ_DIAGNOSTICO.

2. Una vez finalizado el procesamiento, se borra el fichero o se conserva renombrado a la fecha de procesamiento, según esté configurado.

6.8.3.2.4 Fichero de configuración

El procesamiento del fichero es similar al descrito para el fichero de producción.

1. Se almacena la información en la tabla ETIQ_CONFIGURACION.
2. Una vez finalizado el procesamiento, se borra el fichero o se conserva renombrado a la fecha de procesamiento, según esté configurado.

6.8.3.2.5 Fichero de código SAP modificado

El procesamiento del fichero es similar al descrito para el fichero de producción.

1. Busca el código SAP en la tabla ETIQ_PRODUCTOS y almacena el campo que captura del fichero leído (cantidad mínima).
2. Una vez finalizado el procesamiento, se borra el fichero o se conserva renombrado a la fecha de procesamiento, según esté configurado.

6.8.3.3 Estructura interna: Máquina de Estados

El proceso de tratamiento de ficheros es único para todas las etiquetadoras, corre en el servidor svr-operaciones y se llama COM_ETIQ. Está desarrollado en Visual C++ y de forma similar al resto de procesos tendrá una estructura de máquina de estados que se ejecuta periódicamente.

6.8.3.3.1 Estado de trabajo

El tratamiento de cada fichero se puede realizar:

- Periódicamente, en función de la fecha del último tratamiento y el periodo configurado.
- Por solicitud manual, desde la aplicación de supervisión.

Desde la tabla ETIQ_PARAMETROS se consulta la siguiente información:

- Si corresponde tratar el fichero por solicitud manual o por tratamiento periódico.
- Si el tratamiento para un determinado tipo de fichero está habilitado o no.
- Ruta de los ficheros.

Para los ficheros habilitados, se ejecutará el tratamiento si se cumple alguna de las siguientes condiciones:

- La fecha actual supera a la última fecha más el periodo (en minutos).
- El flag de forzado está activo. Este flag se puede activar desde el correspondiente botón en la aplicación de supervisión.

6.8.3.4 Funciones

- ConsultarTratamientos

Consulta la tabla ETIQ_PARAMETROS para ver si hay un flag de forzado o corresponde tratar un fichero por periodo.

- RealizarTratamiento

Busca el fichero deseado en la ruta configurada. Si lo encuentra, llama a la función de tratamiento de ficheros.

- TratarFichero

Analiza el fichero recibido como parámetro (asociado a la etiquetadora correspondiente), descompone la línea del fichero leída en campos (separados por “;”), y se llama a las funciones según el tipo de registro (producción, diagnosis, etc.).

- GenerarFichero

Se genera un fichero (ruta configurable) en los ordenadores de control de etiquetadoras que estén configurados, con los datos de todos los productos. Si ya existe un fichero, se sobrescribirá todo el contenido del existente

- ActualizarMarcaProductos

Actualiza en la tabla ETIQ_PRODUCION el valor de la marca de producción, obtenido del fichero.

- ActualizarCodigoSAP

Actualiza en la tabla ETIQ_PRODUCTOS el valor del campo “cantidad mínima”, obtenido del fichero.

- InsertarConfiguracion

Inserta en la tabla ETIQ_CONFIGURACION los valores de configuración de las etiquetadoras, asociado a la fecha de tratamiento para permitir un histórico.

- InsertarDiagnostico

Inserta en la tabla ETIQ_DIAGNOSTICO los valores de diagnóstico de las etiquetadoras, asociado a la fecha de tratamiento para permitir un histórico.

- InsertarProduccion

Inserta en la tabla ETIQ_PRODUCION los valores de producción leídos del fichero.

6.9 **MÓDULO DE COPIAS DE SEGURIDAD**

El módulo de copias de seguridad permite realizar copias en el servidor de los ficheros o directorios configurados.

Algunos ficheros de interés para realizar copias de seguridad son:

- Ficheros de configuración de los robots y paletizadores.
- Bases de datos locales a las líneas.

6.9.1 Componentes del módulo

El módulo dispone de los siguientes componentes:

- Proceso que realiza las copias de seguridad en función del periodo que tenga configurada cada copia y la fecha actual. Está desarrollado en Visual Basic, y usa las funciones de copia de ficheros y directorios desarrolladas en la aplicación de supervisión.
- Pantalla de la aplicación de supervisión para configurar las copias de seguridad.

6.9.2 Modelo de datos

Este apartado define las tablas de la base de datos que almacenan la información relacionada con este módulo.

6.9.2.1 Tabla COPIAS_SEGURIDAD (CS)

Descripción: Tabla que almacena las rutas de las que se desea realizar copias de seguridad.

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|-----------------|---------------|----------------|---|
| CS_PC_ORIG | VARCHAR2(20) | Clave Primaria | Ordenador origen |
| CS_RUTA_ORIG | VARCHAR2(200) | Clave Primaria | Ruta origen |
| CS_PC_DEST | VARCHAR2(20) | Clave Primaria | Ordenador destino |
| CS_RUTA_DEST | VARCHAR2(200) | Clave Primaria | Ruta destino |
| CS_TIPO | VARCHAR2(1) | | Fichero ('F') o Directorio ('D') |
| CS_PERIODICO | NUMBER(6) | | Periodicidad, en horas |
| CS_ULTIMA_FECHA | DATE | | Fecha de la última copia |
| CS_FLAG_FORZADO | NUMBER(1) | | Indica si se ha activado el forzado (1) o no (0) desde la |

| | | | |
|-----------------|---------------|--|--|
| | | | aplicación de supervisión |
| CS_HABILITACION | NUMBER(1) | | Indica si la copia de seguridad está habilitada (1) o no (0) |
| CS_NOTAS | VARCHAR2(200) | | Comentarios que puede introducir el usuario para clarificar el contenido del que se hace la copia de seguridad |

Tabla 6.32: Tabla COPIAS_SEGURIDAD

6.9.2.2 Tabla MSG_ERROR (ME)

Descripción: Tabla que almacena errores que interesa mostrar en una pantalla al usuario de la aplicación.

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|---------------|---------------|----------------|---------------------------------------|
| ME_FECHA | DATE | Clave Primaria | Fecha del error |
| ME_ORDENADOR | VARCHAR2(40) | | Ordenador |
| ME_APLICACION | VARCHAR2(40) | | Aplicación que genera el error |
| ME_MENSAJE | VARCHAR2(200) | | Mensaje que se le mostrará al usuario |

Tabla 6.33: Tabla MSG_ERROR

6.9.3 Procesos

El correcto funcionamiento del módulo se basa en que el proceso de copias de seguridad se ejecute correctamente.

6.9.3.1 Generalidades

El proceso de copias de seguridad tiene las siguientes características:

- Está desarrollado en Visual Basic.
- Corre en el servidor, realizando las copias de seguridad cuando corresponde.
- El ejecutable es “respaldo.exe” y su arranque es controlado por Imon.

El proceso “respaldo” consulta la base de datos para conocer si se ha de realizar una copia de seguridad. El proceso realizará copias de seguridad de las rutas que estén habilitadas en la tabla COPIAS_SEGURIDAD y cuando se cumpla alguna de las siguientes condiciones:

- Según la fecha actual, la periodicidad y la fecha de la última copia de seguridad corresponda hacer una nueva copia de seguridad.
- El flag de forzado esté activado (se puede activar desde la aplicación de supervisión).

6.9.3.2 Funciones

El proceso tiene las siguientes funciones relevantes:

- ConsultarCopiasSeguridad

Consulta en la base de datos si es necesario realizar una copia de seguridad.

- RealizarCopiasSeguridad

Realiza una copia de seguridad, llamando a las correspondientes funciones según se trate de ficheros o directorios, y actualiza la fecha en la base de datos.

- CopiarFichero

Copia un fichero desde la ruta origen a la destino.

Si se elige hacer copia de seguridad, se renombra el archivo existente añadiendo la fecha actual antes de la extensión.

- CopiarDirectorio

Similar a la función anterior, copiando directorios.

- InsertarError

Inserta en la tabla MSG_ERROR los errores que se producen en la copia de ficheros o directorios, para su consulta a través de una pantalla.

6.10 MODELO DE DATOS

En este capítulo se detalla la estructura de las tablas de uso general.

6.10.1 Tablas generales

6.10.1.1 Tabla LINEAS_PAL (LP)

Descripción: Se rellena con la información de las líneas de paletizado existentes.

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|-----------|--------------|------------------|----------------------------------|
| LP_COD | NUMBER(3) | Clave primaria | Código de la Línea de Paletizado |
| LP_NOMBRE | VARCHAR2(30) | | Nombre de Línea de Paletizado |
| LP_TL_COD | NUMBER(3) | Ref: TIPOS_LINEA | Tipo de línea de paletizado |

Tabla 6.34: Tabla LINEAS_PAL

6.10.1.2 Tabla TIPOS_LINEAS (TL)

Descripción: Almacena los tipos de línea de paletizado (Samovi, Fanuc, Maletas y Emmetti).

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|-----------|--------------|----------------|--------------------------|
| TL_COD | NUMBER(3) | Clave primaria | Código del tipo de línea |
| TL_NOMBRE | VARCHAR2(30) | | Nombre del tipo de línea |

Tabla 6.35: Tabla TIPOS_LINEAS

6.10.1.3 Tabla LINEAS_ENV (LE)

Descripción: Almacena las líneas de envasado existentes.

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|--------|-----------|----------------|--------------------------------|
| LE_COD | NUMBER(3) | Clave primaria | Código de la Línea de Envasado |

| | | | |
|--------------|--------------|--|--|
| LE_NOMBRE | VARCHAR2(30) | | Nombre de Línea de Envasado |
| LE_COD_SAMOV | NUMBER(3) | | Equivalencia entre Línea de Envasado y código del autómeta en Samovi |

Tabla 6.36: Tabla LINEAS_ENV

6.10.1.4 Tabla TIPOS_DATOS (TD)

Descripción: Tabla que almacena los tipos de datos de los parámetros (bit, entero, real, string).

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|-----------|--------------|----------------|-------------------------|
| TD_COD | NUMBER(3) | Clave Primaria | Código del tipo de dato |
| TD_NOMBRE | VARCHAR2(30) | | Nombre del tipo de dato |

Tabla 6.37: Tabla TIPOS_DATOS

6.10.2 Tablas de usuarios

6.10.2.1 Tabla USUARIOS (US)

Descripción: Usuarios del sistema con su correspondiente perfil

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|---------------|--------------|----------------|-------------------------|
| US_COD | VARCHAR2(10) | Clave Primaria | Código del usuario |
| US_PASSWORD | VARCHAR2(6) | NOT NULL | Contraseña |
| US_NIV_ACCESO | NUMBER(3) | NOT NULL | Nivel de acceso |
| US_DESC | VARCHAR2(80) | | Descripción del usuario |
| US_PF_COD | NUMBER(3) | Ref: PERFILES | Perfil del usuario |

Tabla 6.38: Tabla USUARIOS

6.10.2.2 Tabla PERFILES(PF)

Descripción: Perfiles existentes en el sistema

| CAMPO | TIPO | CONSTRAINT | DESCRIPCION |
|--------|------------|----------------|-------------------|
| PF_COD | VARCHAR(2) | Clave Primaria | Código del perfil |

| | | | |
|-------------|---------------|----------|---|
| PF_DESC | VARCHAR2(50) | | Descripción del perfil |
| PF_VALOR | VARCHAR2(100) | NOT NULL | Cadena con permisos de menús 1: Habilitado 2: Deshabilitado |
| PF_BOTONERA | VARCHAR2(100) | | Permisos de botonera |

Tabla 6.39: Tabla PERFILES