

## Resumen

El objetivo del presente documento es la creación de un cuadro de control, ubicado dentro de una planta de cogeneración. Dicho cuadro será el encargado de almacenar todos los eventos eléctricos sucedidos en los diferentes sistemas tanto de alta como de baja tensión procedentes de los diferentes sistemas de protecciones, relees y diversos sistemas de seguridad que dispone la planta.

La base de este documento se centrará en la elección del autómata programable, y la creación del consiguiente algoritmo, mediante el cual se registrarán y almacenarán los eventos sucedidos, y, por otro lado, la creación de una interficie que permita la interconexión del cuadro con el usuario encargado del tratamiento de los datos. Esta interficie tendrá como núcleo la programación de un algoritmo adecuado, para la recopilación de los datos almacenados dentro del autómata programable del cuadro así como para el tratado posterior de dichos sucesos.

Se trabajará también para la presentación de los datos recogidos por esta interficie entre el usuario y el autómata mediante una base de datos de fácil creación y un tratamiento totalmente flexible a las necesidades de los propios consultores (usuarios).





# Sumario

<b>RESUMEN</b>	<b>1</b>
<b>SUMARIO</b>	<b>3</b>
<b>1. INTRODUCCIÓN Y OBJETO DEL DOCUMENTO</b>	<b>5</b>
1.1. Introducción.....	5
1.2. Objeto del documento.....	7
1.3. Datos de la propiedad .....	7
1.4. Ubicación y emplazamiento.....	8
<b>2. DESCRIPCIÓN GENERAL DEL PROYECTO</b>	<b>9</b>
2.1. Breve descripción del problema.....	9
2.2. Características técnicas principales.....	10
2.2.1. Control y seguridad existentes.....	10
2.2.2. Control y seguridad proyectada .....	10
2.3. Descripción funcional del proyecto .....	11
2.4. Implantación de los equipos.....	12
<b>3. COMPONENTES DEL PROYECTO</b>	<b>13</b>
3.1. Módulo PLC .....	13
3.1.1. Controlador Lógico Programable (PLC).....	14
3.1.2. Tarjetas de entradas digitales .....	20
3.1.3. SAI.....	21
3.1.4. Bornes de conexión.....	21
<b>4. ELECCIÓN DE LOS COMPONENTES</b>	<b>22</b>
4.1. Autómata programable PLC .....	22
4.2. Tarjetas de entradas digitales .....	24
4.3. Sistema de alimentación ininterrumpido en CC 24Vdc 10A .....	24
4.4. Fuente de alimentación .....	26
4.5. Baterías .....	26
4.6. Bornes conexión .....	26
4.7. Cableado de interconexión.....	27
<b>5. TRATAMIENTO DE LAS SEÑALES</b>	<b>28</b>



<b>6. DISEÑO TÉCNICO DEL RCE</b>	<b>30</b>
6.1. Cuadro de mando	30
6.1.1. Dimensionamiento	30
6.1.2. Elección del cuadro	34
6.2. Sistema de pantallas	34
<b>7. ALGORITMO DE PROGRAMACIÓN: ADQUISICIÓN Y ALMACENAJE DE DATOS</b>	<b>35</b>
7.1. Creación de un proyecto mediante Twincat PLC Control	36
7.2. Detección del dispositivo BC9000 via ethernet	39
7.3. Planteamiento del algoritmo de programación	41
7.3.1. Algoritmo de obtención de datos	42
7.3.2. Algoritmo de lectura bit a bit de la variable de entrada	42
7.3.3. Algoritmo de detección de flancos de subida y bajada	42
7.3.4. Algoritmo de TRIGGERS	42
7.3.5. Algoritmo de almacenaje de datos	43
7.3.6. Diagrama de flujo	44
<b>8. MONTAJE FÍSICO Y SIMULACIÓN DEL PROTOTIPO</b>	<b>45</b>
8.1. Ciclo de scan	49
8.2. Tratamiento en memoria del PLC	52
8.3. Algoritmo adquisición y tratamiento de 200 entradas	53
<b>9. INTERFACE PC_PLC</b>	<b>55</b>
9.1. Entorno PC	55
9.2. Programación en Visual C+ 6.1	56
9.3. Tratamiento de datos	56
<b>10. PRESENTACIÓN DE LOS RESULTADOS</b>	<b>58</b>
10.1. Base de datos	58
<b>CONCLUSIONES</b>	<b>60</b>
<b>AGRADECIMIENTOS</b>	<b>61</b>
<b>BIBLIOGRAFÍA</b>	<b>62</b>
Bibliografía complementaria	62



# 1. Introducción y objeto del documento

## 1.1. Introducción

La industria SOCIEDAD COOPERATIVA ANDALUZA GANADERA DEL VALLE DE LOS PEDROCHES (en adelante **COVAP**), es una empresa ubicada en la localidad de Pozoblanco (Córdoba), dedicada a la explotación ganadera (ganado vacuno y porcino). Para realizar su actividad la cooperativa consta de diversos centros productivos, entre los que cabe destacar:

1. Industria láctea
2. Fábrica de piensos
3. Industria de elaborados cárnicos
4. Matadero

Dichas factorías presentan un consumo de vapor, frío y electricidad, siendo la industria láctea la principal consumidora de vapor y frío debido a los procesos que en ella se realizan, entre los cuales se destacan:

- Equipos consumidores de energía térmica, como los terminadores-pasteurizados y estelirizadores, y en menor medida el CIP de limpieza.
- Demanda de agua entre 0 y 1°C para el enfriamiento de la leche a su llegada a la factoría, así como para el funcionamiento de los terminadores-pasteurizados.

Actualmente COVAP explota una central de cogeneración que cubre las demandas de vapor y frío de la industria láctea, así como los consumos eléctricos de la propia industria láctea y del Matadero. La cogeneración actual está formada por una turbina de gas SOLAR CENTAUR 50 de 4.3 MWe (ISO) y una caldera de recuperación pirotubular con quemador de post-combustión. Pudiendo producir 10.2 t/h de vapor a 15 bar-a saturado (sin postcombustión) y 20 t/h a 15 bar-a también saturado (con uso de postcombustión).



La cogeneración instalada en el año 2001 también dispone de un generador de vapor convencional de la misma capacidad que el anterior, para cubrir la demanda en casos en los que la central de cogeneración esté parada. El vapor producido mediante la cogeneración se utiliza para el proceso de producción de COVAP y, mediante una máquina de absorción de capacidad 1500 KWf, se enfría el agua glicolada a partir del calor obtenido del vapor. Como la zona de ubicación de la central no dispone de gas natural canalizado, se instaló una planta de almacenamiento y regasificación de gas natural licuado (GNL) transportado hasta allí mediante camiones cuba.

En la actualidad COVAP desea abastecer, mediante la central de cogeneración, la demanda de vapor, frío y electricidad de las tres factorías que forman la cooperativa: Industria Láctea, matadero y piensos. Esto supone añadir a la demanda de la Industria Láctea y matadero, las propias de la fabrica de piensos (vapor y electricidad) y de las del matadero (vapor, frío y electricidad).

La demanda de vapor y frío que presenta la cooperativa se puede cubrir prácticamente en su totalidad mediante la planta de cogeneración existente. Esto supondría un elevado uso del quemador de postcombustión y/o de la caldera convencional, provocando la existencia de un potencia de cogeneración desatendido y obteniendo un rendimiento energético muy bajo. Por otra parte, la demanda eléctrica de las tres factorías no puede satisfacerse mediante la central actual, teniendo que aumentar, en consecuencia, la capacidad de producción eléctrica de la planta.

Después de un estudio detallado del proyecto de COVAP, se ha determinado que la mejor opción es ampliar la cogeneración actual con una segunda línea de características similares a la existente. Siendo la turbina escogida para tal efecto la SOLAR MERCURY 50 de 4.6 MWe (ISO).

COVAP ha decidido renovar su central de cogeneración acogándose a las condiciones establecidas en el Real Decreto 436/2004, en base a sustituir su grupo turbogenerador a gas por otro de más potencia, y adaptar el sistema generador de vapor y otros auxiliares a las nuevas condiciones derivadas del cambio.

Como consecuencia de dicha necesidad, COVAP pretende también controlar de forma mucho más detallada todos los sucesos y eventos, tanto termodinámicos como eléctricos, que puedan repercutir de forma negativa en el funcionamiento de las plantas de cogeneración.



Para solventar esta necesidad se procederá a la creación de un Registrador Cronológico de Eventos (RCE) que será capaz de almacenar la actuación de las protecciones del sistema eléctrico que conforman la planta de cogeneración existente así como la proyectada.

## 1.2. Objeto del documento

El presente documento tiene como objeto la descripción del Registrador Cronológico de Eventos (**RCE**) que se llevará a cabo en la planta de cogeneración que tiene UTE IDEA-COVAP en la factoría de COVAP en Pozoblanco (Córdoba).

El presente documento cubre los siguientes objetivos:

- ⇒ Conocimiento de las necesidades reales de un sistema registrador de eventos
- ⇒ Descripción de la configuración del RCE
- ⇒ Especificación del sistema que se pretende instalar, de su proceso y de sus prestaciones
- ⇒ Visualización de los datos registrados por el RCE mediante una pantalla de control visual.
- ⇒ Volcado de los datos almacenados sobre un PC para su posterior análisis mediante un mensaje asociado al tipo de suceso.

## 1.3. Datos de la propiedad

La propietaria de la central es una Unión Temporal de Empresas, formada por COVAP y IDEA (INSTITUTO PARA LA DIVERSIFICACIÓN Y AHORRO ENERGETICO), cuyo nombre es UTE IDEA-COVAP.

COVAP consta como la empresa titular y exportado de la central.



## 1.4. Ubicación y emplazamiento

La ubicación de la empresa renovada será en el espacio donde actualmente se ubican las instalaciones de la planta de cogeneración actual, en terrenos de la factoría de COVAP.

### **COVAP**

Crta. de la Canaleja s/n

Pozoblanco – (Córdoba)

El plano de situación geográfica (CVA00021-0) muestra la ubicación general de la planta de cogeneración y el plano de áreas afectadas (CAV00041-1) define los espacios disponibles para el emplazamiento específico de las nuevas instalaciones, así como las zonas que pueden quedar afectadas.





## 2. Descripción general del proyecto

### 2.1. Breve descripción del problema

Las fallas que se producen en un Sistema Eléctrico se despejan (aíslan) mediante la actuación de sistemas automáticos o protecciones, los cuales producen la salida de servicio de uno o más segmentos del Sistema Eléctrico comprometidos por la falla. La consecuencia directa de esta acción cuyo objetivo es proteger al equipamiento de las corrientes generadas por la falla, produce como efecto colateral, la reducción de la capacidad de transmisión del Sistema, es decir, una repercusión directa a los restantes miembros del Mercado Eléctrico, los Generadores y los Consumidores.

En general puede aceptarse que la reducción del tiempo de duración de las fallas beneficia a todos los agentes del Mercado Eléctrico. Es por eso que las empresas de transporte de energía, vuelcan recursos en adquirir tecnología, adiestramiento y estudios, a fin de lograr resultados concretos en este tema.

La minimización del tiempo post-falla está íntimamente relacionada con el conocimiento de las causas que originaron la falla y el estado post-falla del Sistema. La provisión de esta información es llevada a cabo por los especialistas en protecciones, quienes analizan las operaciones y actuaciones de los equipos automáticos y emiten un diagnóstico. En función de esta información, las personas encargadas de ejecutar las operaciones deciden los pasos a seguir tendientes a recuperar el Sistema.

La mayor parte del tiempo necesario para este proceso es consumido por tareas poco relevantes, tales como el tiempo requerido para convocar al especialista o el necesario para obtener los datos adquiridos por los Sistemas de Adquisición de Datos (SAD), que es la información que documenta todos los eventos eléctricos sucedidos.

Además de lo mencionado, puede suceder que el causante de la perturbación que origina una salida de servicio resulte ser el sistema automático encargado de proteger el sistema eléctrico. Este hecho, a primera vista poco probable, también tiene que ser dilucidado por el especialista en protecciones.

Como tarea complementaria al diagnóstico, los especialistas captan, mediante un monitor, la actuación de las protecciones con el objeto de detectar cualquier indicio de defecto que pueda ocasionar en el futuro problemas graves como por ejemplo: la salida incorrecta de



algún segmento del sistema, al actuación retardada de la protección (esto pone en riesgo al equipamiento de la estación), la actuación incorrecta de algún modulo de la protección, etc.

En consecuencia, el trabajo del proyecto consiste en desarrollar un Registrador Cronológico de Eventos que almacene en tiempo real la información adquirida procedentes de relees, protecciones, etc. y que, frente a un suceso característico de una falla, emita una descripción de dicha falla que asistirá a los especialistas y operadores a identificar rápidamente el origen del problema y efectuar las operaciones que correspondan.

## **2.2. Características técnicas principales**

### **2.2.1. Control y seguridad existentes**

Actualmente la planta está diseñada para un funcionamiento automatizado y sin requisitos de intervención de operadores en situaciones de normal funcionamiento.

Sí son necesarias intervenciones en operaciones de puesta en marcha, paros, tratamiento de incidencias, etc., y se explicitaran por cada suministrador.

El concepto de control se basa actualmente en un Sistema de Adquisición de Datos y operación (SAD) para captación de señales analógicas, digitales y de los PLC's para la supervisión general de la planta, seguimiento en tiempo real, registro de históricos, preparación de informes, etc.

### **2.2.2. Control y seguridad proyectada**

El Sistema de Adquisición de Datos actual (SAD) capta las señales analógicas, digitales y de los PLC's con un tiempo de scan aproximadamente entre 1 y 2 segundos. Esto significa que en ese periodo de tiempo (1-2seg) el SAD analiza sus entradas y almacena los datos que en ella se encuentren, pero con el tipo de instalaciones eléctricas que rigen los comportamientos de las protecciones tanto de red, como la de la planta de cogeneración así como de la propia factoría, este tiempo de proceso es insuficiente para un control y análisis detallado de los eventos que se suceden en el sistema eléctrico.



Con lo cual el sistema de control y seguridad proyectado de basará en:

- ⇒ Un equipo Registrador Cronológico de Eventos (RCE) con su propio sistema de control formado por un módulo cuyo elemento principal será el PLC.
- ⇒ Sistema de Adquisición de Datos y operación (SAD) para captación de señales analógicas, digitales y de los PLC's para la supervisión general de la planta, seguimiento en tiempo real, registro de históricos, preparación de informes, etc.[existente]
- ⇒ Incorporación de un operador para la supervisión general de la planta.

### **2.3. Descripción funcional del proyecto**

El sistema de control RCE estará formado por un módulo, como se ha comentado anteriormente:

➤ **Módulo PLC**

Este módulo será el encargado de realizar la lectura de los puertos de entrada del propio PLC en el cual se recibe la información procedente de los relees. Dicha información procede de los cambios de estado de los propios relees, los cuales emiten una señal de activación y otra señal de actuación (si dicha actuación sobre el interruptor se produce: apertura / cierre).

Estas señales son binarias (0/1) que son enviadas al PLC mediante una subestación formada por un puente de bornes. Estos bornes reciben y transmiten las señales de los relees a los puertos de entrada del PLC, que serán constituidos por unas tarjetas de entrada de señales digitales compatibles con el PLC. Las actuaciones de los relees serán almacenadas cronológicamente en tiempo real dentro de la memoria de PLC.



## 2.4. Implantación de los equipos

La ubicación del cuadro de control de nuestro RCE vendrá condicionada al medio en el que se exponga. Debemos tener presente que en un ambiente industrial siempre se debe tener presente este aspecto a la hora de ubicar los cuadros de mando. En nuestro caso, el cuadro se situará en la planta de baja tensión con lo que no estará expuesta a un ambiente demasiado agresivo y, por lo tanto, se buscará una robustez adecuada a dicha situación en cuanto a la elección del tipo de cuadro a utilizar.

Finalmente, y teniendo en cuenta las premisas descritas, el sistema RCE se instalará en una cabina de control en paralelo a los cuadros de las nuevas cabinas de control de la nueva turbina de vapor.

El esquema de principio de baja tensión (CVA01071-1) nos muestra la ubicación exacta del cuadro de mando en el que se instalará el Registrador Cronológico de Eventos (RCE).



## 3. Componentes del proyecto

### 3.1. Módulo PLC

El módulo estará formado por los siguientes elementos:

- Un PLC, núcleo del módulo.
- Tarjetas de entrada de señales digitales.
- Un sistema de alimentación ininterrumpida (SAI).
- Un interruptor magnetotérmico.
- Un puente de bornes de conexión .

La figura 3.1 muestra la distribución orientativa de los elementos dentro del módulo.

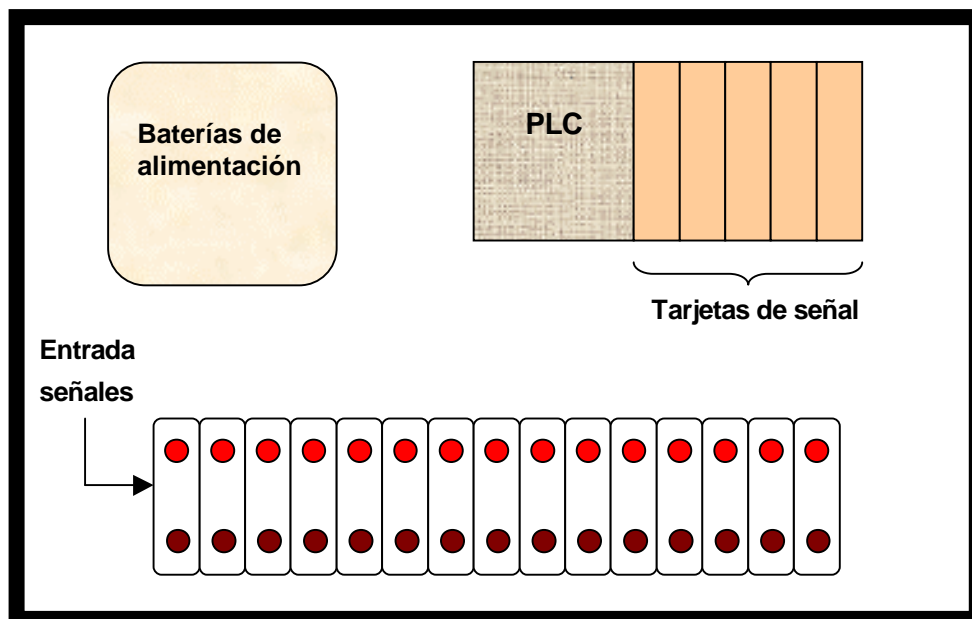


Figura 3.1.- Esquematación de los elementos del módulo PLC



### 3.1.1. Controlador Lógico Programable (PLC)

Nuestro PLC, elemento central del módulo y del conjunto RCE, tendrá la misión de captación y almacenamiento de los eventos sucedidos como consecuencia del cambio en los estados de los relees (apertura/cierre). Las especificaciones que regirán la elección del PLC más adecuado serán:

► Entradas:

Tipo : Binarias, mediante contacto libre de potencial NA/NC

Dichas entradas proceden de los relees, actuadores, seccionadores e interruptores propios de la central de carácter binario 0/1 con la siguiente lógica:

0 → Se considerará interruptor cerrado 

1 → Se considerará interruptor abierto 

Las señales que interesará procesar serán:

- Activación del relee.
- Apertura/cierre del interruptor con las protecciones adecuadas.

Número : Para un máximo de 200 entradas

Este número de entradas está sujeto a modificación ya que el diseño del PLC será del todo flexible para la incorporación o eliminación de entradas.

Resolución : 5ms

El tiempo de filtraje debe ser de unos 5ms, eso significa que en ese periodo de tiempo el PLC debe ser capaz de leer todas sus entradas con independencia de cualquier suceso externo o interno del módulo.



### Tiempo mínimo de permanencia de la señal : 2ms

Para que la señal sea totalmente registrada la duración de la misma debe ser de unos 2ms sino podría suceder que se perdieran eventos ya que estos no quedarían almacenados en la memoria del PLC.

### Puerto Ethernet de transmisión de datos

Escogeremos un PLC con una salida Ethernet debido a su fácil integración en la red de la fábrica y/o Internet además de la posibilidad que existan más de un sistema haciendo consulta, integración con **DCS/SCADA** (Digital Control System / Supervisory Control and Data Acquisition) y la posibilidad que cualquier usuario pueda con un computador descargar toda la información contenida en la memoria del PLC en cualquier momento que sea necesaria ya sea en el período de revisión del mismo como en posibles controles extraordinarios. (El tiempo de transmisión de datos no es un factor relevante en el proyecto).

### Puertos serie para posibles periféricos

Permite la conexión del módulo pantalla de visualización, la cual se situará junto al módulo PLC de tal forma que permita el control y evaluación de los últimos eventos sucedidos de forma totalmente visual facilitando así las actuaciones a tomar en caso de emergencia.



► Salidas:

Registro de cada uno de los sucesos con marca de tiempo y código de señal

Cada uno de los sucesos ocurridos en los relees queda registrado en la memoria del PLC. En dicho registro se almacenará el número de la entrada correspondiente a la perturbación, qué tipo de cambio se ha producido (paso de 0 a 1 ó de 1 a 0) así como el momento del evento con una marca de tiempo en milisegundos desde el momento de arranque del PLC (origen de tiempos).

Tabla de asignación de mensajes a las señales

Cada evento registrado deberá ser analizado posteriormente por un PC el cual asociará cada cambio de estado con un mensaje para el usuario. Este fenómeno podría darse también directamente sobre la pantalla pero por simplicidad del prototipo no se realizará por el momento.

Almacenamiento de, al menos, 10.000 sucesos

Nuestro sistema debe ser capaz de poder almacenar como mínimo unos 10.000 sucesos para garantizar el éxito del prototipo.

Posibilidad de descarga sobre ordenador en cualquier momento sin perturbar la captación de señales

Mediante el puerto Ethernet, nuestro sistema será capaz de descargar los datos almacenados en su memoria en cualquier momento y por cualquier usuario autorizado sin necesidad de parar el proceso continuo de captación de sucesos. De tal forma que si durante la descarga de información se producen fenómenos que deben ser registrados, el sistema no deberá tener problema en almacenarlos en las primeras posiciones de memoria liberadas en el momento de la descarga.





### Posibilidad de descarga sobre un visualizador externo para control visual

Mediante un puerto serie el sistema mostrará los datos almacenados en su memoria, mencionados anteriormente, directamente sobre un pantalla para un posible control visual (Ver figura 3.2)

<i>Input</i>	<i>changed</i>	<i>timestamp</i>
1	OFF-ON	13.000
54	ON-OFF	65.235
39	OFF-ON	10.005.658
⋮	⋮	⋮
⋮	⋮	⋮

Figura 3.2.- Representación visualización de señal

### Respaldo periódico en soporte permanente

Dichos registros deberán quedar gravados en la memoria de PLC, aún produciéndose un fallo de alimentación, todos aquellos sucesos hasta el momento del fallo.

### Posibilidad de interconexión con periféricos externos para posibles ampliaciones de funcionalidad y de prestaciones del PLC

Como se comenta en puntos anteriores la posibilidad de incorporar nuevos periféricos hace necesaria la incorporación de puertos de salida. Con tan solo la introducción de un puerto serie RS-232 seremos capaces de ofrecer nuevas prestaciones a nuestro sistema si en un futuro se producen ó son necesarias simplemente añadiendo un nuevo código a nuestro algoritmo de programación.



### 3.1.1.1. Lenguaje de programación

El lenguaje de programación que utilizaremos para la creación de las instrucciones a seguir para la finalidad del proyecto es el lenguaje **Structural Text (ST)** con las siguientes prestaciones:

- Lenguaje de alto nivel
- Sintaxis similar a Pascal o C
- Operadores, expresiones, asignaciones
- Llamadas a función
- Control del flujo de programa
- Funciones, Bloques Función

### 3.1.1.2. Posibles ampliaciones

De forma externa a nuestro PLC y para garantizar el buen funcionamiento del mismo se podría incorporar un sistema de **WatchDog**, de tal forma que éste generara cada cierto tiempo una señal en una de las entradas de PLC y, por tanto, esa señal debería quedar registrada en memoria como si un evento hubiera ocurrido. Así, comprobaremos al ver la aparición de esta señal en los periodos de tiempo estimados, si nuestro PLC está actuando correctamente o por lo contrario existe algún error o fallo en la captación de señales y, en consecuencia, un mal funcionamiento del sistema. Al ser este sistema totalmente externo, nos garantiza la fiabilidad de los resultados obtenidos por el PLC.

También de forma externa se podría incorporar un **Clock**. Este sistema nos permitiría no sólo almacenar el tiempo en que sucede el evento (timestamp) si no el tiempo real en que ocurre (día-hora). El PLC y el Clock estarían sincronizados y en el momento en que por algún motivo se parará en sistema PLC éste al realizar la lectura de sus puertos en el momento de la inicialización, leería también la hora y la sincronizaría.



Otra posible ampliación del proyecto sería la conexión del sistema RCE con el SAD/DCS/SCADA ya existente de forma que se pudiera relacionar la información contenida en el SAD con la del RCE y de esta forma obtener más detalle sobre los eventos sucedidos y registrados.

Tanto el WatchDog como el Clock y la relación RCE-SAD serán elementos que posiblemente por la necesidad de crear un modelo económico y simple se verán relegados a un segundo plano, quedando como posible ampliación al proyecto en caso de disponer de presupuesto, necesidad de diferenciar dicho producto o bien para garantizar con un 100% de fiabilidad el buen funcionamiento del sistema.

### 3.1.1.3. Necesidades de memoria (capacidad de la PLC)

Para poder determinar la capacidad de memoria del PLC debemos considerar que elementos serán guardados en dicha memoria:

8 bit de estado (0/1). Técnicamente con tan sólo un bit ya se almacenaría dicha información pero trabajaremos con bloques de 1 byte por facilidad de código.

Timestamp → por simplicidad del sistema tan sólo almacenaremos el tiempo en milisegundos referidos al inicio de conexión del PLC. 32 bits

Codificación de 200 eventos → 8 bits

Un byte de reserva por necesidades futuras.

Con lo cual las necesidades de memoria son:

56 bits x evento (debemos almacenar unos 10.000 eventos)

$56 * 10.000 = 560.000 \text{ bits} = 70.000 \text{ bytes} \approx \underline{\underline{70 \text{ kbytes}}}$

Hay que tener en cuenta, además de la capacidad mínima de memoria del PLC, la necesaria por los procesos internos propios del automatismo del PLC. Por ello, la elección del PLC se basará fundamentalmente en dos aspectos: por un lado, que la memoria interna del PLC sea lo suficientemente grande como para gestionar todos los eventos que se esperan y, por otro lado, el PLC debe ser capaz de gestionar sus propios programas y procesos sin que esto produzca una reducción en su tiempo de scan.



### 3.1.2. Tarjetas de entradas digitales

Tal y como hemos mencionado según las especificaciones de diseño, sería necesario el procesamiento de unas 200 señales digitales binarias como entradas del sistema. Para tal fin debemos incorporar a nuestro PLC una serie de tarjetas de entradas digitales en paralelo. Existen tarjetas con 2, 4, 8, 16, 32 y 64 entradas con lo que teniendo en cuenta estas 200 señales necesitaríamos:

$$200 : 2 = 100 \text{ tarjetas de dos entradas}$$

$$200 : 4 = 50 \text{ tarjetas de cuatro entradas}$$

$$200 : 8 = 25 \text{ tarjetas de ocho entradas}$$

$$200 : 16 = 13 \text{ tarjetas de dieciséis entradas}$$

$$200 : 32 = 7 \text{ tarjetas de treinta y dos entradas}$$

$$200 : 64 = 4 \text{ tarjetas de sesenta y cuatro entradas}$$

Estas tarjetas vendrán alimentadas por el propio conjunto de baterías que alimenta el PLC que en caso de corte en el suministro eléctrico ya sea por motivos tanto internos, como microcortes o cortes de mayor prolongación, o bien por motivos externos, como la extraviación de las conexión externas por causas meteorológicas o por necesidad, son las encargadas de mantener el funcionamiento del cuadro de control del RCE.



### 3.1.3. SAI

El SAI nos servirá para alimentar el RCE mediante la red, en su estado permanente, y mediante unas baterías, en caso que se produjera algún corte en el suministro normal de la red. La alimentación será de 24V de c.c. con una autonomía de las baterías de unas 4h. Estas baterías vendrán alimentadas desde la red y éstas a su vez darán servicio al conjunto PLC. En producirse funcionamiento en isla el sistema será capaz de funcionar independientemente de la red durante un periodo dicho periodo de 4h.

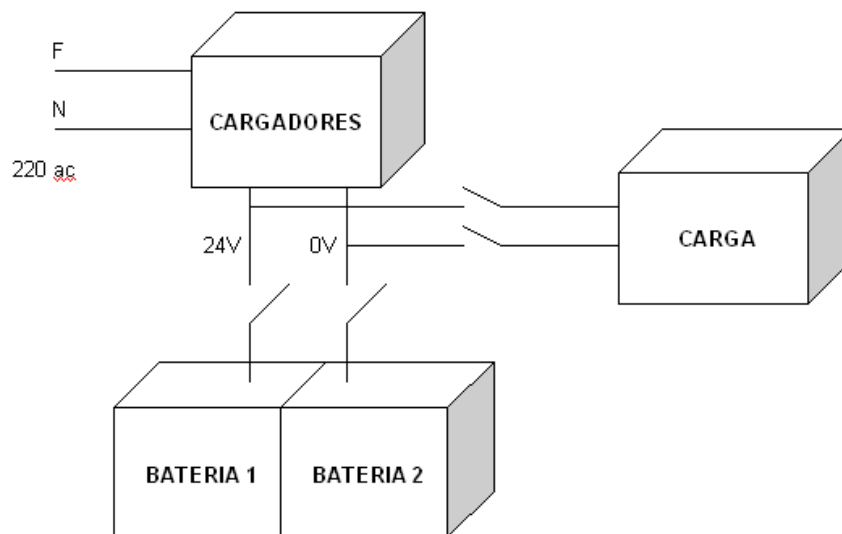


Figura 3.3. Diagrama de baterías

### 3.1.4. Bornes de conexión

Los relees de protección tanto de la planta de cogeneración como de la propia red generan dos señales digitales: una de activación del propio rele y otra de apertura o cierre del interruptor al que están asociados. Esas dos señales serán objeto de nuestro procesamiento con lo que nos va a interesar captar ambas señales y hacerlas llegar a nuestro RCE. Para ello instalaremos el cableado necesario para transportar dichas señales (el cableado se estudiará posteriormente) hasta el RCE. La conexión entre este cableado y los puertos de entrada del PLC se darán mediante una estación de bornes, la misión de la cual es simplemente de transmisión de la propia señal.

Incorporar esta estación será de gran comodidad en caso de producirse cualquier desperfecto en las tarjetas de entrada, fusión de alguno de los cables, mal contacto del cable con la propia entrada digital, u otro suceso, sin necesidad de desconectar todo el cable que une el rele origen y el módulo PLC.



## 4. Elección de los componentes

Tras la descripción de los componentes que configurarán el Registrador Cronológico de Eventos, existen en el mercado actual diversas opciones para llevar a cabo nuestro sistema. Por ejemplo:

- A. Instauración de un PC
- B. Un modelo de PLC

Tras haber analizado las necesidades del equipo, hemos descartado la opción de instalar un PC debido, entre otras cosas, por la necesidad de fiabilidad del sistema. Necesitamos un sistema, un autómata, con el mínimo de funciones a desarrollar de tal forma que se garantice la fiabilidad para lo que está programado.

### 4.1. Autómata programable PLC

El modelo escogido de PLC será el *Ethernet Bus Terminal Controller – Ethernet TCP/IP BC9000* de la casa BECKHOFF (ver Anexo I). Dicho modelo cumple con las especificaciones requeridas por el prototipo:

- Conexión Ethernet para volcado de datos (TCP/IP)

Este tipo de conexión permite que cualquier usuario autorizado para interconexión con el autómata, pueda acceder a los datos que en él se encuentran, simplemente conectando un PC o cualquier estación de trabajo en los puntos de conexión de este tipo.

- Capacidad suficiente para soportar la demanda de memoria (64/128 kbytes)

Tal y como hemos visto anteriormente, necesitamos mínimamente unos 70 KBytes para poder almacenar los eventos sucedidos en la instalación, además de tener en cuenta la necesidad de capacidad propia para los procesos internos del automatismo. Por ello, el BC9000 al tener una capacidad superior a la requerida es una buena opción para el buen funcionamiento del proceso de almacenamiento y transmisión de los datos.



- Capacidad de incorporar más de 200 entradas (512 inputs)

Una de las características del autómata BC9000 es la capacidad de integrar un total de 512 inputs. Esto significa que dicho PLC es capaz de incorporar tarjetas de entradas digitales hasta conseguir un total de 512 entradas. Dependiendo de cuantas entradas tenga cada tarjeta, variará el número de tarjetas a integrar. Aún así, la necesidad hasta ahora mencionada, no excede de 200 entradas, con la cual cosa, la capacidad del PLC es más que suficiente para llevar a cabo esta especificación.

- Tiempo de scan inferior a 5ms (aproximadamente 1,5ms variable según programación)

Los SAD hasta ahora utilizados en los sistemas de control de datos tienen un tiempo de scan de aproximadamente 1-2 segundos. Teniendo en cuenta que la velocidad de emisión de impulsos eléctricos es de milisegundos, estos sistemas de detección son, en comparación, muy lentos para la detección de todos y cada uno de los eventos eléctricos que se produzcan en la instalación. Por ello, una de las especificaciones más importantes de este registrador es su velocidad de lectura de los puertos de entrada. Teniendo en cuenta, esta premisa, el BC9000 asegura un ciclo de scan de unos 1,5 milisegundos aproximadamente que hará detectables todas las señales eléctricas enviadas por los relees y protecciones tanto del sistema de AT como el de BT.

- Número de estaciones de autobuses (tarjetas de entradas digitales) superior a las que se necesitan para implementar nuestro sistema.

El PLC es capaz de incorporar los suficientes dispositivos de tarjetas digitales como para abarcar la totalidad de nuestras necesidades. Se podrían incorporar hasta 64 tarjetas de 8 entradas digitales. Según nuestros cálculos, con 25 tarjetas de 8 entradas digitales sería suficiente y el PLC debe de tener capacidad suficiente para ello.



## 4.2. Tarjetas de entradas digitales

Las tarjetas de entradas digitales seleccionadas corresponden al modelo Terminal module KL1408 / 8-Input ( 25 módulos ) de la casa BECKHOFF (ver Anexo II) con las siguientes características:

- Compatibilidad con el BC900
- Número de entradas digitales : 8
- Alimentación necesaria de 24 Vdc

## 4.3. Sistema de alimentación ininterrumpido en CC 24Vdc 10A

Los fallos de red o fluctuaciones de la tensión de corriente alterna de las compañías eléctricas son habituales. En caso de un apagón o problema de red en muchos casos supone daños en nuestro sistema o pérdidas importantes de dinero al tener que paralizar nuestra actividad. El Buffer o SAI en corriente continua junto con una batería ofrece una protección fiable y económica para este tipo de problemas. En caso de corte de red o problemas en la misma instalación seguirá suministrando 24 Vdc a nuestra aplicación.

Al estar aisladas la entrada y la salida es fácil separar los ramales que necesitan seguir funcionando en caso de corte de red. La energía de la batería no se desperdiciará en alimentar los equipos que no necesitan seguir funcionando ante un corte de red.

Cuando la fuente de alimentación ofrece una tensión suficiente el SAI cargará la batería de 12V a la vez que ofrece 24V de salida gracias a un elevador de tensión Boost. En caso de un fallo de alimentación el equipo elevará la tensión de la batería a 24V y la energía se suministrará de manera regulada y controlada.

El controlador de batería interno del equipo tiene un cargador de batería, protección contra descarga profunda de batería y protección contra sobretensiones para conseguir la máxima duración de la batería.





El estado del sistema es mostrado mediante luces fáciles de entender y contactos de rele, que nos permiten accionar otro tipo de dispositivos de aviso como sirenas, grandes luces, etc. De esta forma en caso de cualquier incidencia en nuestro sistema podremos estar avisados y tomar las medidas oportunas sin necesidad de estar cerca del equipo.

Entre algunas de las protecciones incluidas en el equipo podemos destacar protección contra conexión incorrecta de la polaridad de la batería, protección contra conexión de una batería errónea (que no sea de 12V), protección contra una tensión de alimentación errónea y protección contra sobrecargas. (Anexo III)

La figura 4.1 nos muestra el esquema de la disposición de los elementos internos del SAI. Como se observa este sistema permite el funcionamiento en continua de la carga bien sea alimentada directamente por la red como, si por alguna situación se inutiliza dicha red, este sistema permite el continuo funcionamiento de la carga mediante la batería de 12V.

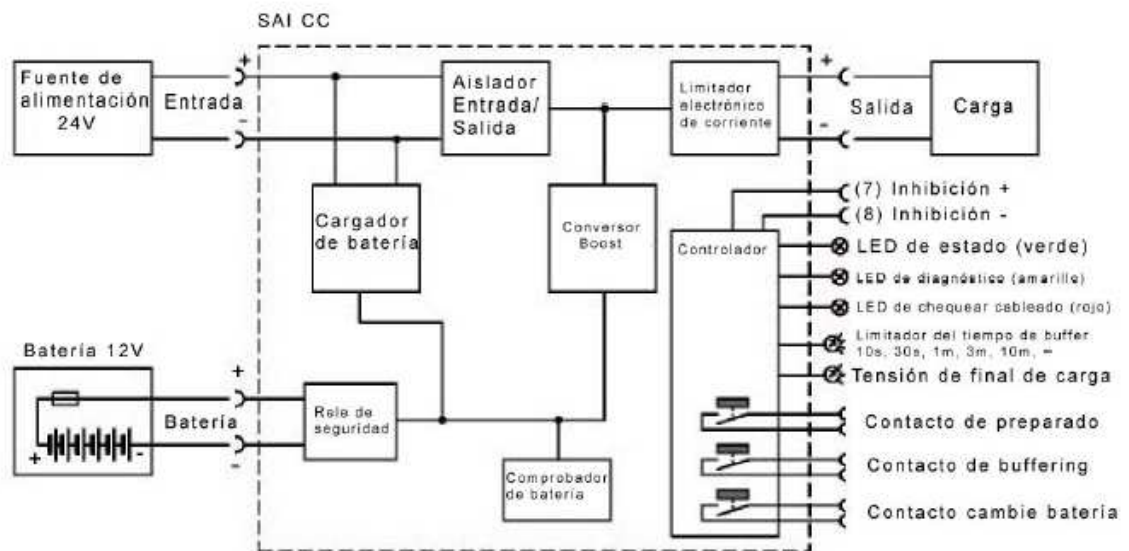


Figura 4.1. Sistema de Alimentación Ininterrumpida (SAI)



#### 4.4. Fuente de alimentación

Hemos optado por una fuente de alimentación DR-120 de la empresa Mean Well. Esta fuente de alimentación transforma la tensión alterna (AC) de entrada en continua a 24 V (CC). Consta también de una serie de protecciones contra cortocircuitos, sobrecargas, sobretensiones e incluso contra el sobrecalentamiento del sistema. Además es totalmente compatible con los carriles DIN, que son los carriles en los que se disponen todos los elementos que configuran el cuadro de control. (ver Anexo IV)

#### 4.5. Baterías

El conjunto formado por el SAI y la fuente de alimentación servirán para la carga de las baterías del sistema. Estas baterías proporcionaran la energía suficiente para en continuo funcionamiento de nuestros sistemas con un autonomía de unas 4 horas. Pasado este tiempo, la falla por la que se ha producido la puesta en marcha del sistema de baterías debe haberse solventado. En reconectar el sistema a su estado natural, las baterías con cargadas de nuevo para volver a ser usadas si se da el caso. (Ver Anexo V)

#### 4.6. Bornes conexión

Los bornes de conexión aseguran la correcta conexión entre los relees de los que se extraen las señales y las tarjetas de entradas digitales. Éstas son alimentadas a 24 Vdc mediante una fuente de alimentación. El modelo escogido para llevar acabo este conexionado son las WDK 2.5 D de la casa WEIDMÜLLER principalmente por las siguientes características (ver Anexo VI).

- Pequeño tamaño
- Tensión nominal 400 V
- Intensidad nominal 25 A
- Sección nominal 25mm<sup>2</sup>
- Sobretensión de choque nominal 6 kV



## 4.7. Cableado de interconexión

Tal y como hemos mencionado en el apartado anterior, los bornes reciben la alimentación de 24 Vcc que es enviada al releo al cual están conectados. Éste, devuelve una señal al bornero que es comunicado al PLC.

La interconexión entre el bornero, los releos o protecciones y el PLC se realiza mediante un cable coaxial de cobre de 1mm de diámetro. Este tipo de cable es el más utilizado en sistemas de transmisión de señales eléctricas dentro de las plantas de cogeneración y, en concreto, en las transmisiones a los cuadros de control como el SAD.

Principales características:

1. Conductor de cobre electrolítico, clase 5 (flexible)
2. Aislamiento de polietileno reticulado
3. Cubierta de poliolefina ignifugada, libre de halógenos y con baja emisión de humos en caso de incendio.



## 5. Tratamiento de las señales

Como se ha mencionado en el punto anterior, deberemos ser capaces de poder tratar cerca de unas 200 señales procedentes de relees, interruptores y seccionadores del sistema eléctrico de la instalación, con la posibilidad de hacer flexible este número, es decir, aumentarlo o disminuirlo sin que repercuta negativamente en el desarrollo del proyecto.

La elección de las señales a analizar se ve caracterizada por el tipo de relee del que proceden. Esto significa que no podremos obtener todas aquellas señales que deseemos sino que únicamente aquellas que por estructura de circuito y por prestaciones de los relees, nos permitan obtenerlas.

Los relees existentes y el tipo de protección que ofrecen en la instalación actual sin la ampliación prevista son los siguientes:

**SMOR 7000 B** : 50 / 51 / 51N / 59 / 27 / 81M / 81m / 59G

**TOV 5033 B** : 59G

**XG2** : 78

**MIFPI51** : 50 / 51 / 51N

**MIFNI05** : 64

Siendo, como nos muestra la tabla 5.1:

PROTECCIÓN	DESCRIPCIÓN
27	MINIMA TENSION
50	SOBRECARGA
51	SOBREINTENSIDAD
51N	MAXIMA INTENSIDAD HOMOPOLAR
59	MAXIMA TENSION
59G	MAXIMA TENSION HOMOPOLAR
64	PROTECCIÓN DE CUBA
78	VARIACIÓN DE FASE
81M	MAXIMA FRECUENCIA
81m	MINIMA FRECUENCIA

Tabla 5.1. Descripción de las protecciones



Todos estos relees (ver Anexo IX) permiten obtener la señal de actuación de las señales sobre el interruptor / seccionador al que están conectados. Pero no todos facilitan la señal de activación del relee. De echo, tan sólo el relee **TOV 5033 B** genera tanto la señal de actuación como la activación. De tal forma que por limitaciones eléctricas constructivas obtendremos señal de actuación de todos los relees y activación / actuación únicamente del relee **TOV 5033 B**.

Obtener la señal de activación de protección de máxima tensión homopolar (59G) conllevará la modificación del cuadro de control existente. Debemos añadir dos líneas de tensión (+24V / -24V) de tal forma que incorporando un nuevo relee podamos registrar dicha señal. Este esquema quedará modificado según los planos CVA00034-2 y CVA00035-1.

El Anexo VII proporciona toda la información completa y necesaria para contabilizar todas aquellas señales que deberemos tratar en nuestro proyecto. Como se puede observar, dicho cuadro incluye la ubicación de las cabinas, relee de procedencia, tipo de protección y sobre que interruptor / seccionador actúa. Esta información podrá ser utilizar posteriormente para el análisis detallado del tratamiento de las señales.

En el listado de señales de protección se han incluido también las nuevas señales que se desean obtener de la ampliación correspondiente a la central. El esquema unifilar correspondiente al plano CVA04021-5 nos muestra la situación de las protecciones descritas anteriormente tanto el esquema existente como la ampliación prevista y ya aprobada por todos los integrantes del proyecto.



## 6. Diseño técnico del RCE

Una vez definidos los componentes principales que configurarán el equipo del RCE, y las señales que queremos tratar, procedemos a la realización del diseño técnico. Éste conllevará el diseño del cuadro de mando, el sistema de pantallas y el sistema de programación. Tras este diseño podremos determinar que modelo y tipo de armario es el más adecuado para las especificaciones mencionadas así como la justificación de ello.

### 6.1. Cuadro de mando

El cuadro de mando tiene la misión de soportar físicamente todos aquellos componentes que conforman el RCE, es decir, incluirá el PLC, las tarjetas, el bornero de conexión, el conjunto cargador-baterías, la pantalla visualizadora y los canales de distribución del conexionado. Además de ello, el cuadro de mando deberá proteger estos componentes frente a cualquier evento externo como vibraciones provocadas por equipos colindantes, ruidos exteriores, impermeabilidad, resistencia a la temperatura, etc.

#### 6.1.1. Dimensionamiento

Para el dimensionamiento del cuadro de mando partiremos del diseño de la placa de montaje ya que será allí en donde se situarán todos los componentes ya citados.

Consideraciones a tener en cuenta para el diseño:

- a) Entre los bornes de conexión y el conjunto formado por el PLC y las tarjetas debe existir unos canales para el reparto del cableado.
- b) La separación entre borde de placa-canal y canal-bornes de conexión debe ser el suficiente como para que el caso de supervisión o reparación el operario pueda acceder a ellos sin dificultad. Esta distancia deberá estar comprendida entre 40-50mm.
- c) En el carril del PLC deberán ir dispuestos el magnetotérmico y el conjunto cargador-baterías.



Existen dos posibilidades de disposición de los componentes:

- I. Horizontal
- II. Vertical

Primeramente, teniendo en cuenta que salvo excepciones la disposición de este tipo de cuadros es de forma horizontal, se diseñó la placa de montaje de forma que todos los componentes se dispusieran en horizontal con un único bornero. Este sistema era inviable debido a que al disponer todos los bornes en horizontal y junto a la especificación de que debemos obtener unas 200 señales, al colocar los bornes tal y como hemos indicado el dimensionamiento de dicha placa representaba unas dimensiones de 1200mm x 500mm de altura. Obviamente, no resulta práctico una con esta distribución. Este diseño corresponde a la figura 6.1.

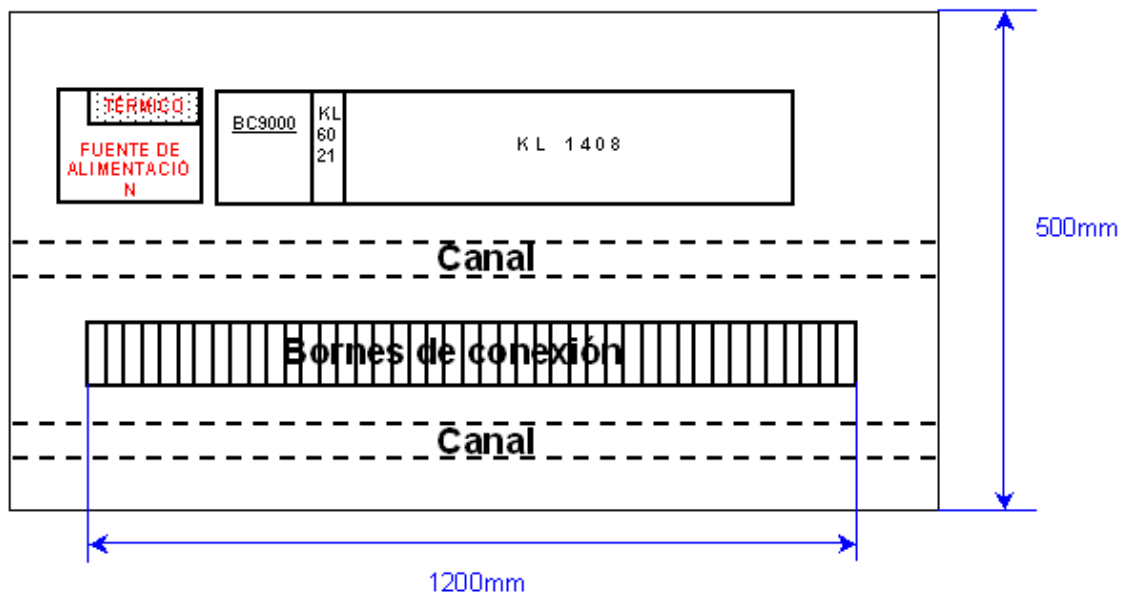


Figura 6.1. Distribución horizontal del cuadro de mando con un único bornes de conexión



Se optó entonces por redistribuir el bornero en dos secuencias equitativas de forma que se reducía sensiblemente el dimensionamiento de la placa de montaje tal y como se muestra en la figura 6.2.

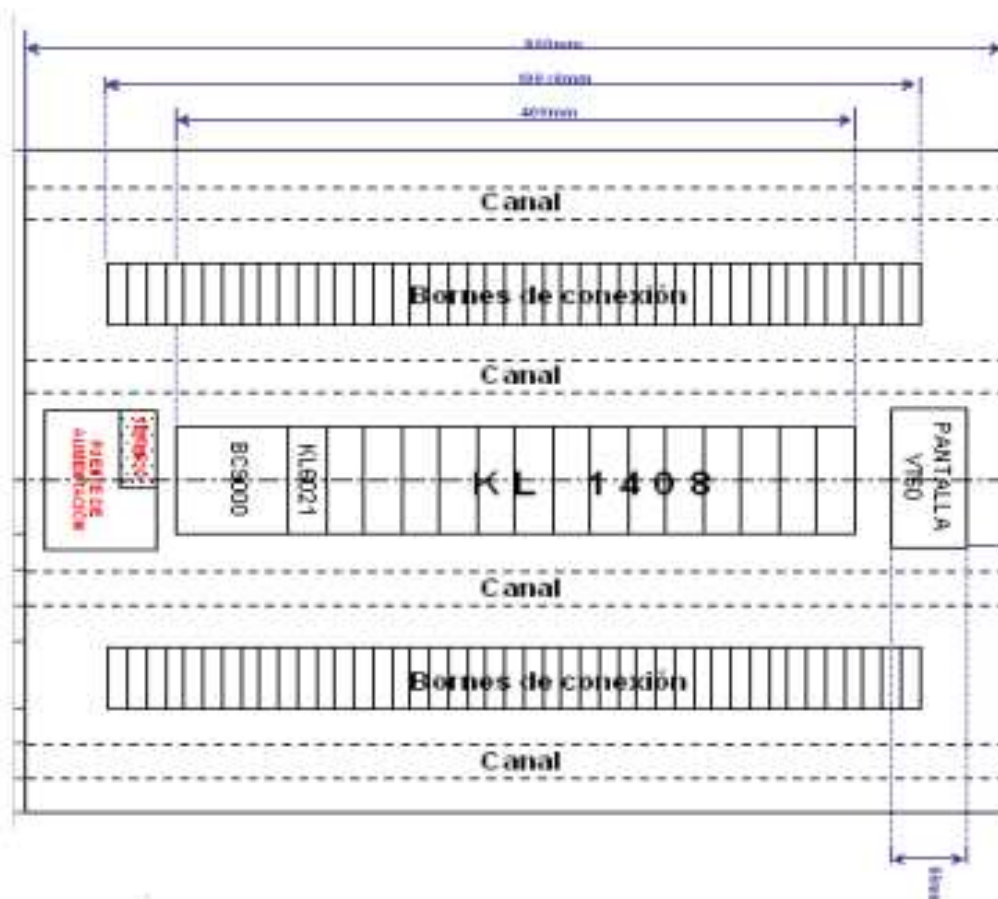


Figura 6.2. Disposición horizontal cuadro de control

Este último diseño era factible a nivel dimensional pero la dificultad reside en cómo hacer llegar los cables de señal a cada bornero. Para el bornero inferior no existe problema alguno pero para el bornero superior sí. Resulta muy complicado llevar los cables desde la entra del armario al bornero superior. Por ello, se propone en mismo diseño pero en disposición vertical de forma que la entrada del cableado el uniforme y se reparte de forma equitativa mediante los canales de distribución situados en la misma dirección que la entrada de los cables. Por tanto, la disposición preliminar del cuadro de montaje será la mostrada en la figura 6.3.





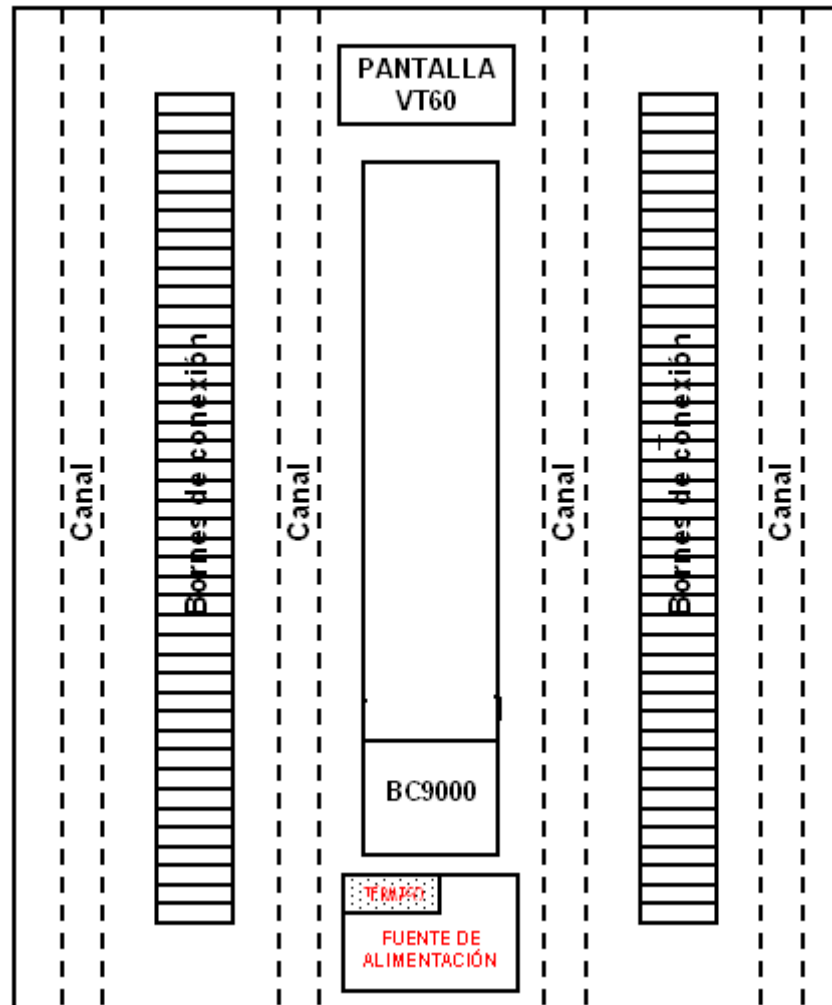


Figura 6.3 . Distribución horizontal del cuadro de mando con doble bornero de conexión

Este diseño permite la entrada del cableado por su zona inferior y se reparte de forma uniforme en los dos borneros existentes para su fácil conexionado.

Los canales facilitan la interconexión entre módulos, es decir, el cableado es destinado por el interior de los canales para evitar la aparición de estos por el interior del cuadro de control, de tal forma que los protege de agentes exteriores que puedan ocasionar cualquier desperfecto en el conexionado.



### 6.1.2. Elección del cuadro

Como se ha comentado anteriormente, la elección del cuadro de mando viene condicionado por el dimensionamiento de la placa de montaje. Tras esta evaluación, se ha determinado que según las medidas del cuadro (800 x 718 mm) se escoja un armario tabulado de 1056 x 852 x 350 mm de profundidad que corresponde al modelo **PLM-108** (placa perforada) de la empresa HIMEL (Ver anexo VIII).

Finalmente y para el mejor dimensionado y disposición de los elementos, la fuente de alimentación, el cargador y las baterías se dispondrán en la parte superior del cuadro.

## 6.2. Sistema de pantallas

El módulo de visualización es el encargado de mostrar los últimos eventos sucedidos en el sistema eléctrico. Dichos eventos son almacenados en la memoria del PLC. Mediante un terminal de comunicación (Ethernet) accederemos a dicha memoria para poder, así, obtener toda la información almacenada para su posterior análisis y tratamiento por parte del operario o supervisor de seguridad de la planta. Dicho operario (consultor del sistema) puede realizar una serie de acciones que a continuación se citan:

1. **Descarga de datos** directamente desde el PLC para estudio y tratamiento.
2. **Imprimir** directamente los eventos almacenados en la memoria de PLC.
3. **Liberar espacio de memoria** una vez descargados los archivos o bien por necesidad de incorporar nuevos sucesos.
4. **Exportar a excel** los datos almacenados por su facilidad de tratamiento y/o la realización de documentos con dichos datos.

Para que el consultor tenga la posibilidad e realizar todas estas acciones, debe existir la figura del configurador, el cual, deberá realizar las siguientes tareas:

1. **Asignar texto a cada entrada**, de forma que el consultor u operario reciba la mayor información posible a cerca de los eventos sucedidos.
2. **Imprimir la configuración** del sistema de manera que siempre quede reflejada dicha configuración para facilidad en caso de modificación o de fallo del sistema.



## 7. Algoritmo de programación: adquisición y almacenaje de datos

La programación del PLC la realizaremos en el lenguaje de programación **Standard Text (ST)** debido a la sencilla programación con comandos fácilmente entendibles y la rápida y precisa interpretación de los resultados obtenidos en las simulaciones experimentales. Tal y como se muestra en el anexo X existen diversas funciones estándar que nos ayudarán a la programación del PLC de forma totalmente estructurada mediante, si hiciera falta, bloques funcionales que ayudaran a facilitar la posterior corrección de dicho algoritmo en caso necesario.

Dicha programación la desarrollaremos sobre el programa **TWINCAT SYSTEM**, concretamente con el paquete **TWINCAT PLC CONTROL**. Este programa es un completo entorno desarrollado para funciones de PLC. Permite al programador una entrada sencilla mediante los eficaces recursos del lenguaje basado en **IEC 61131-3** que define las especificaciones de la sintaxis y semántica de los lenguajes de programación de PLC's, incluyendo el modelo de software y la estructura del lenguaje. El uso de los editores y de las funciones de Debugging tiene como modelo los entornos de desarrollo perfeccionados de lenguajes de programación superiores.



## 7.1. Creación de un proyecto mediante Twincat PLC Control

Una vez iniciado el programa desde el PC, se visualiza la pantalla inicial del TWINCAT PLC CONTROL, tras seleccionar un nuevo proyecto se activa la siguiente pantalla:

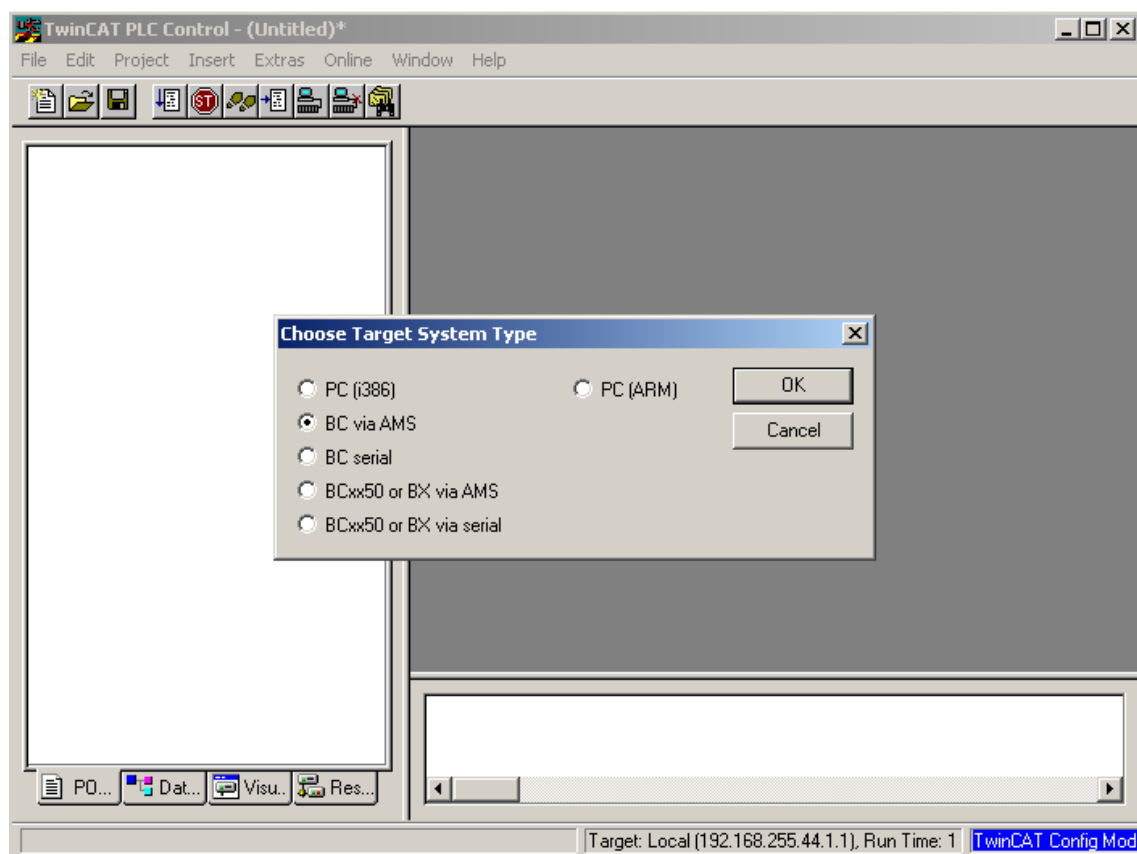
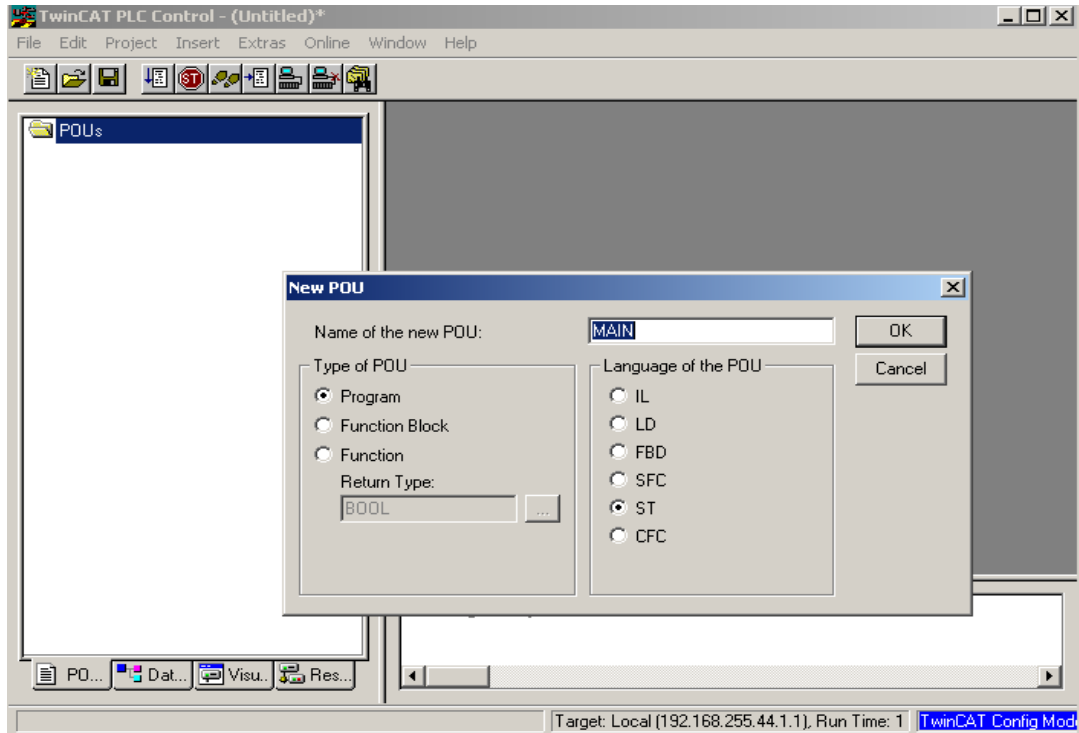


Figura 7.1. Elección del tipo de sistema

Debemos seleccionar la opción BC via AMS ya que es el tipo de sistema de tarjeta del que disponemos. Tras ello, procedemos a la elección del lenguaje en que se programará el PLC ya sea en ST, IL, FB u otros lenguajes, además del tipo programa, es decir, si se trata de un programa, un bloque funcional o una función:





Figura

7.2. Pantalla inicio de POU

Llegados a este punto debemos caracterizar el PLC usado en los experimentos. Mediante la opción *Project / Options / Control Settings* seleccionamos el PLC instalado en nuestro banco de pruebas que en nuestro caso se trata del BC9000:

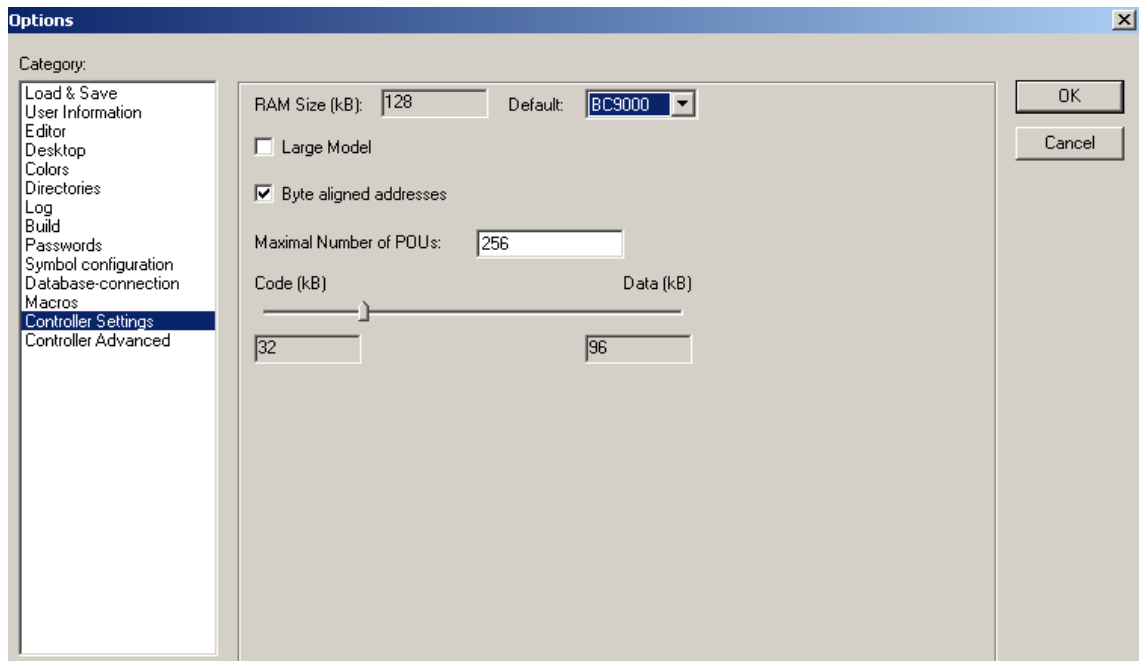


Figura 7.3. Elección del PLC



Una vez ya identificado el lenguaje a emplear, el PLC y el tipo de programa a realizar, se activa la pantalla principal del programa en la cual se escribirá el código del algoritmo del PLC en la que se muestran las variables, las estructuras necesarias y las funciones complementarias del algoritmo:

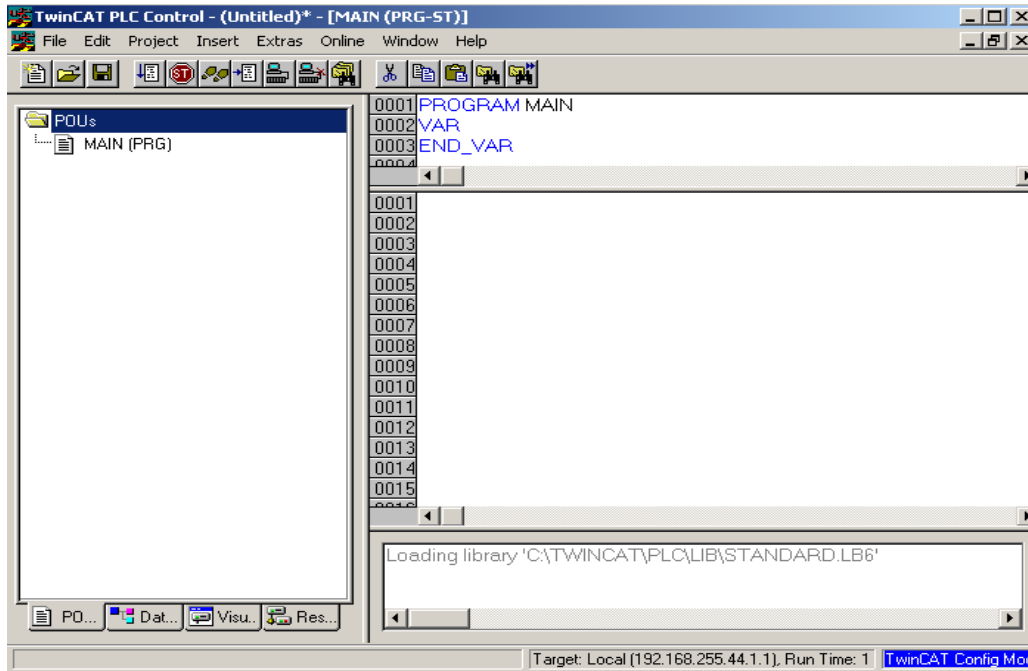


Figura 7.4. Pantalla principal de creación del algoritmo de programación

Previamente, definiremos el programa **MAIN** ya que es el programa principal de todo el algoritmo. Posteriormente, se irán definiendo los demás subprogramas de tal forma que, siguiendo los protocolos de escritura del lenguaje ST, tengamos un código algorítmico totalmente estructurado y de fácil comprensión.



A partir de este punto el programador ya puede empezar a escribir su código de programa haciendo uso de las funciones propias internas del TWINCAT PLC CONTROL que se encuentran descritas en Recursos:

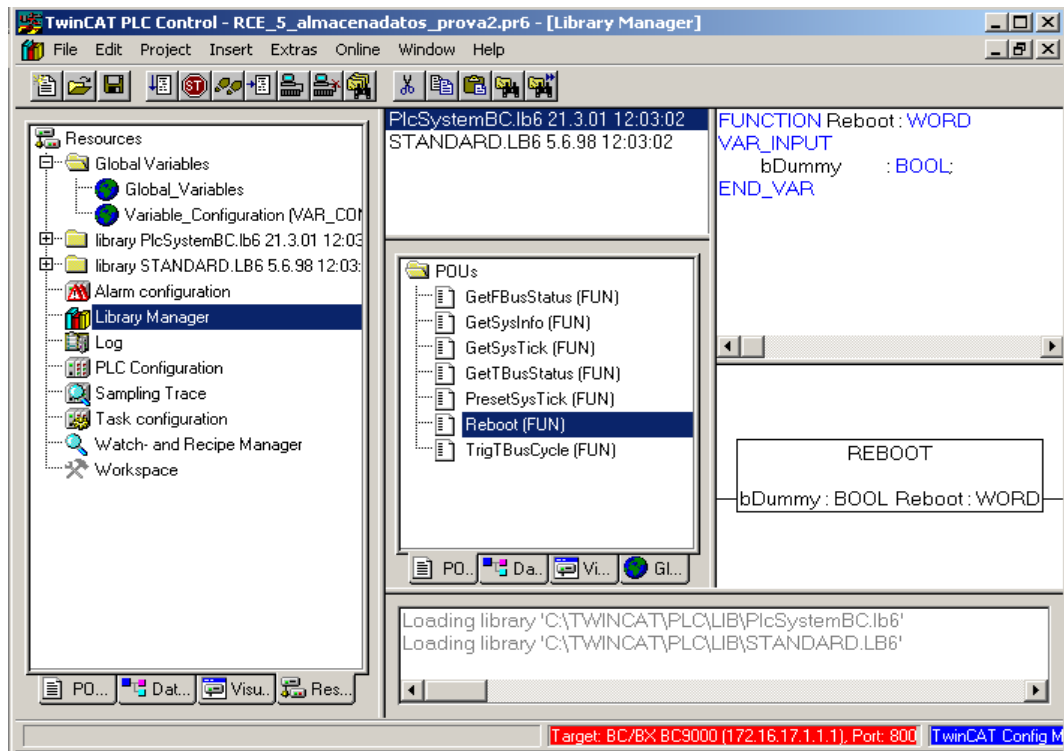


Figura 7.5. Pantalla de librerías internas del TWINCAT PLC CONTROL

## 7.2. Detección del dispositivo BC9000 via ethernet

Nuestro PLC BC9000 se conectara con el PC a través de la red Ethernet ya que de esta forma podremos acceder a ella desde cualquier punto con cualquier PC sin necesidad de conectarnos directamente equipo con equipo con el PLC.



Mediante el programa **TWINCAT SYSTEM MANAGER**, detectamos el equipo del BC9000 y se registran en el programa las variables de entrada, los equipos conectados, las tarjetas de entradas digitales, etc tal y como se muestra en la figura 7.6:

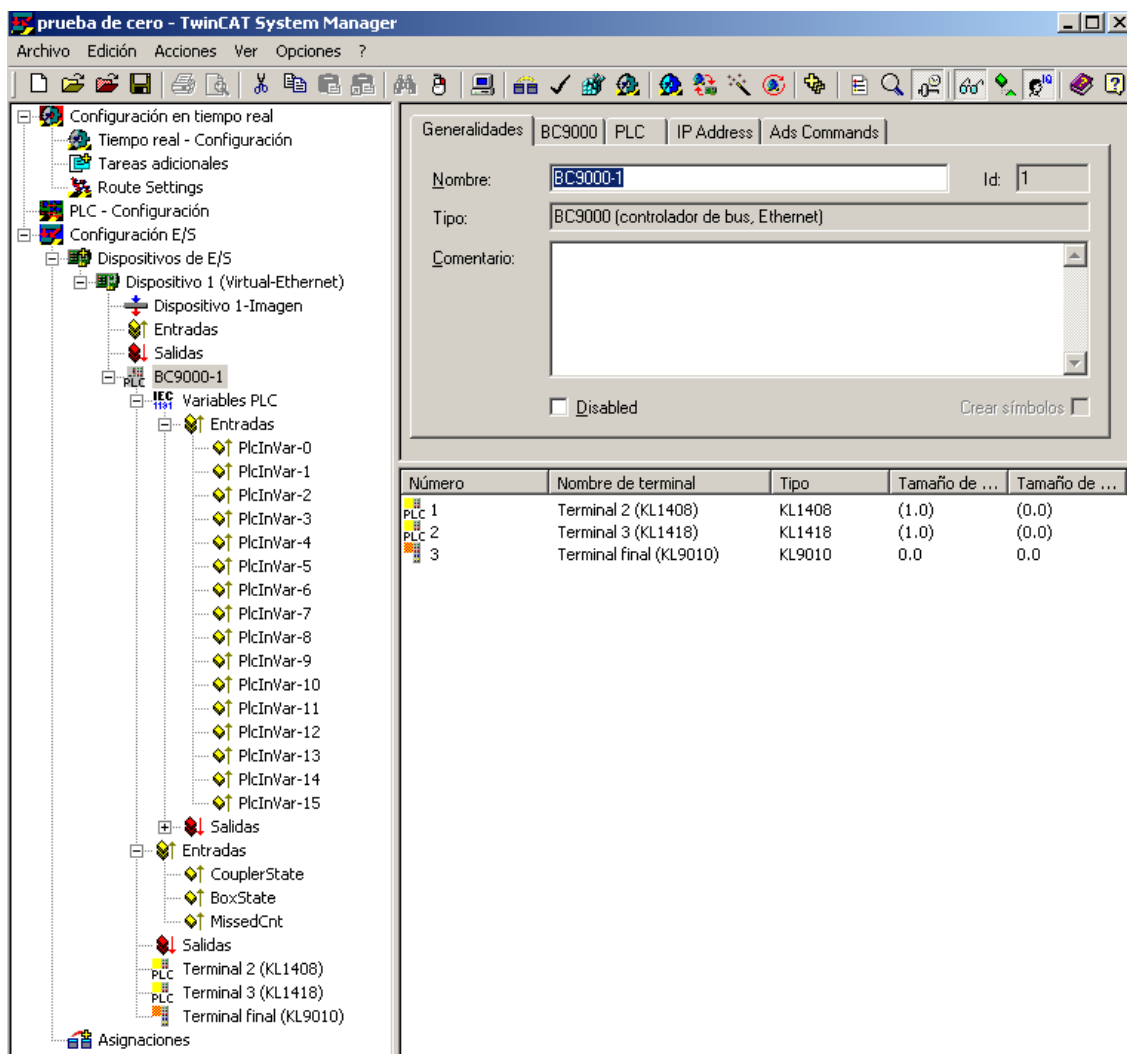


Figura 7.6. Descripción de los elementos detectados en la conexión Ethernet





### 7.3. Planteamiento del algoritmo de programación

Teniendo en cuenta las especificaciones del proyecto, es decir:

- Almacenaje de unos 10.000 eventos
- Número de entradas máxima : 200
- Tiempo de scan : 5ms aprox
- Almacenaje de datos y marca de tiempo

Entre otras especificaciones, realizaremos una simulación previa con los siguientes componentes:

- Un PLC BC9000
- Dos tarjetas de 8 entradas digitales cada una. Una de las tarjetas es la KL1408 y la otra KL1418 (16 entradas digitales).
- Un tarjeta de fin de carrera KL9010
- Un bornero con unos 20 bornes aproximadamente para realizar las interconexiones eléctricas

Todos estos componentes se encuentran ubicados en el cuadro de mando para llevar a cabo de forma práctica las conexiones y los experimentos.

La evolución del algoritmo de programación será la siguiente:

1. Algoritmo de obtención de datos.
2. Algoritmo de lectura bit a bit de la variable de entrada
3. Algoritmo de detección de flancos de subida y bajada
4. Algoritmo de TRIGGERS
5. Algoritmo de almacenaje de datos
6. Algoritmo de almacenaje en memoria PLC



### 7.3.1. Algoritmo de obtención de datos

El primer algoritmo que realizamos es el de obtención de datos. Una vez conectados todos los componentes, dicho algoritmo deberá simplemente hacer la lectura de los puertos de entrada digitales procedentes de las tarjetas y mostrarlas por pantalla.

### 7.3.2. Algoritmo de lectura bit a bit de la variable de entrada

Seguidamente, una vez certificamos que la lectura de los puertos de entrada es correcta, almacenamos esas entradas para tener esos valores accesibles y poder, con ello, consultar bit a bit los valores de cada entrada. El siguiente programa permite la lectura de cada uno de los bits y además, mediante una máscara, los compararemos para ver si ha sucedido un evento de cambio de estado de 0 a 1. Si así sucede incrementaremos un contador que será el que nos indique el número de cambios de 0 a 1 que se han producido.

### 7.3.3. Algoritmo de detección de flancos de subida y bajada

El propio programa TWINCAT permite obtener la información de cambio de flanco tanto de subida como de bajada mediante las funciones **F\_TRIG** y **R\_TRIG** (Ver anexo XI) con lo que no hará falta crear una máscara para cada caso.

El principal problema de este método de adquisición de los cambios de los estados reside en el echo que debemos consultar el estado y la comparación de los estados bit a bit, con lo que en nuestro caso real con unas 200 entradas resulta extremadamente larga la programación y la interpretación en caso de necesidad de cambio y/o error en la programación.

### 7.3.4. Algoritmo de TRIGGERS

Tras en razonamiento anterior crearemos un **ARRAY** de **R\_TRIG** y **F\_TRIG**, consistiendo estos en un vector de variables del mismo tipo, con lo que empaquetaremos los cambios obtenidos en los bits en este tipo de variables y mediante una estructura iterativa consultaremos el estado de cada bit y lo procesaremos.



### 7.3.5. Algoritmo de almacenaje de datos

Con estas aclaraciones la longitud de programa se simplifica notablemente. El siguiente paso consiste en el almacenaje y memorización de los eventos sucedidos. Esto resulta necesario ya que, los algoritmos hasta ahora descritos sobrescriben la información obtenida sobre las variables de salida que nos muestran la codificación de la tarjeta, la entrada, el estado y la marca de tiempo. Este método sólo almacena el último cambio producido en nuestras variables.

Si queremos obtener un histórico de los eventos sucedidos periódicamente nos debemos crear un buffer de memoria en la que ir almacenando de forma cronológica los sucesos registrados por el PLC incluyendo para cada uno de ellos la misma información: número de tarjeta y de entrada, el estado actual y la marca de tiempo (Timestamp).

El tamaño del buffer que crearemos estará limitado por el número de bit que se pueden registrar en la memoria del programa en el TWINCAT, con lo que previamente a este proceso, creamos una estructura de datos que llamaremos *Datos*. Estructura englobará toda la información que queremos de las señales de forma compacta

De forma que con esta descripción podremos llegar a almacenar 2000 eventos. El algoritmo descrito incluye esta estructura que facilitará posteriormente el direccionamiento a memoria accesible del PLC.

Finalmente, presentaremos los resultados obtenidos mediante simulación en TWINCAT ya que este programa finalmente se parecerá mucho al algoritmo que emplearemos para la resolución del proyecto.



### 7.3.6. Diagrama de flujo

El conjunto de algoritmos confeccionados en los apartados anteriores nos permiten reducir las cuestiones, planteadas en la realización del algoritmo final, en pequeños subprogramas que conformarán la totalidad del algoritmo final. Para visualizar la forma definitiva el transcurso de a evolución del programa a realizar, construimos un diagrama de bloques tal como el siguiente:

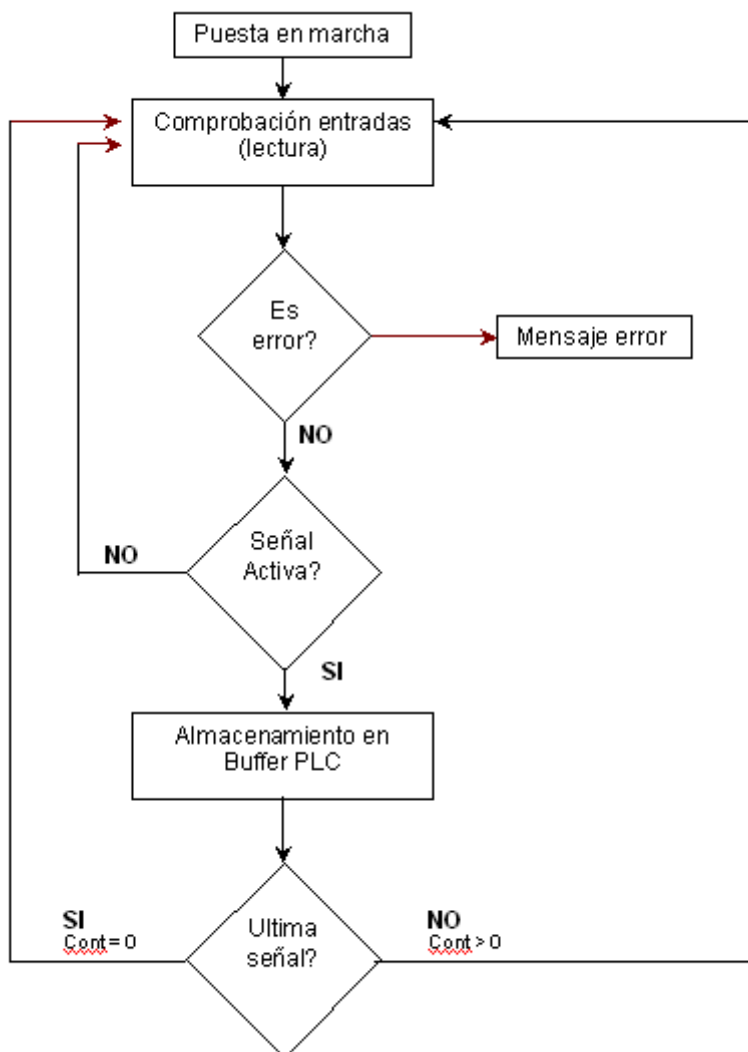


Figura 7.7. Diagrama de bloques funcionamiento PLC



## 8. Montaje físico y simulación del prototipo

Recordemos que la simulación realizada constará tan sólo de 16 entradas digitales. El experimento consistirá en la activación y desactivación de los contactos de forma manual con lo que previamente a la recepción de los eventos, seremos concedores de los resultados a esperar. A continuación se muestra el montaje físico realizado para el ensayo.

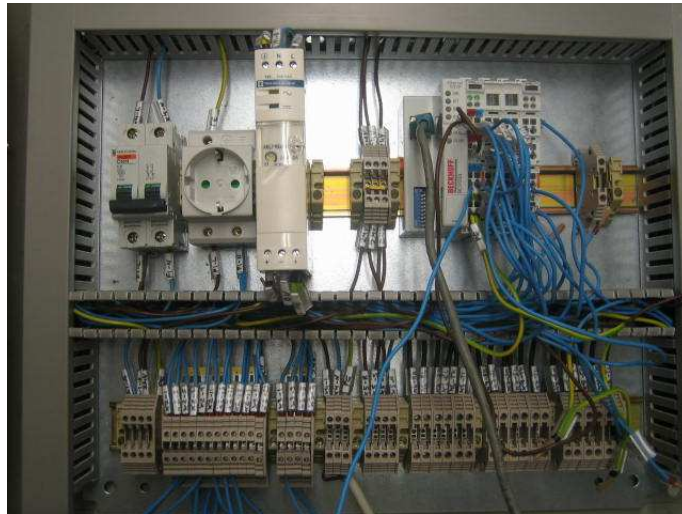


Figura 8.1. Cuadro mando del PLC



Figura 8.2. Fuente de alimentación junto con el magnetotérmico



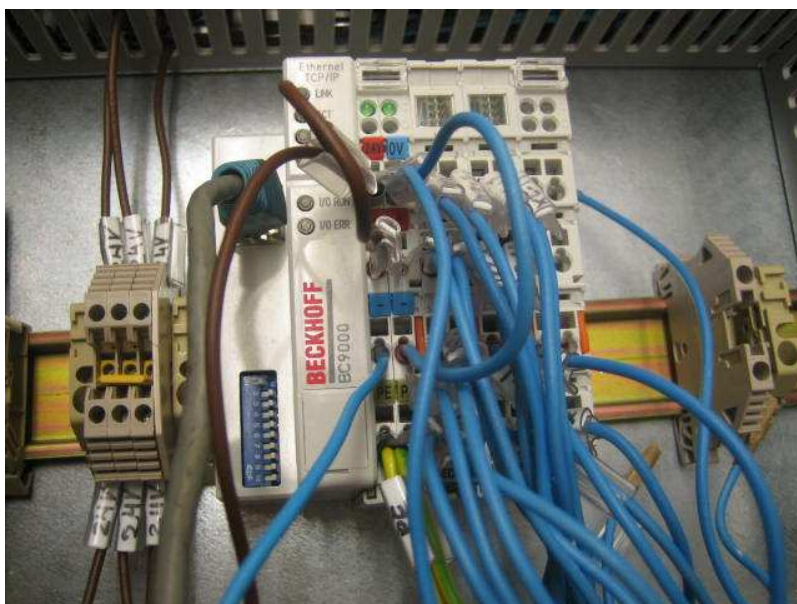


Figura 8.3. Automata BC9000, tarjetas de entradas digitales y conectado



Figura 8.4. Bornero de interconexión



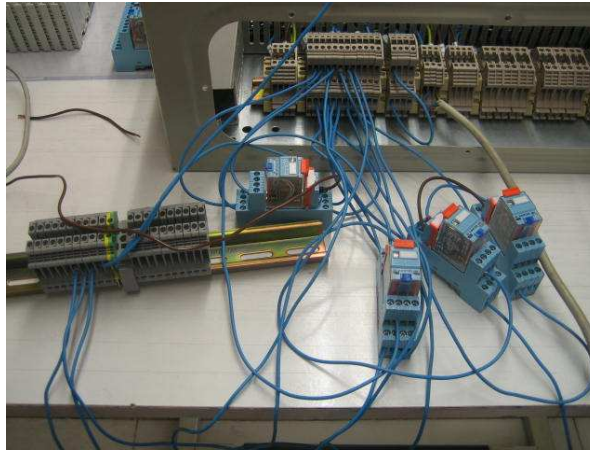


Figura 8.5. Interconexión de cuatro relees

Para la realización de la simulación, debemos cargar sobre el autómatas el programa descrito en el **POU**, siendo éste el programa principal. Para ello seleccionamos la opción de la barra de herramientas *ONLINE / Log in*. Esta acción conlleva la compilación del programa y la carga del algoritmo sobre el PLC. Hecho esto, mediante *ONLINE / Run* ponemos en marcha el programa y actuando manualmente sobre las entradas de nuestro prototipo obtenemos los siguientes resultados:

POUs	0008	BufferRCE[1]	..:Tarjeta_Codentrada = 4
Codificacio (PRG)	0009	..:Estado = <b>TRUE</b>	..:Timestamp = 12086
MAIN (PRG)	0010	..:num_pasos_por_cero = 0	
	0011	BufferRCE[2]	..:Tarjeta_Codentrada = 6
	0012	..:Estado = <b>TRUE</b>	..:Timestamp = 12087
	0013	..:num_pasos_por_cero = 0	
	0014	BufferRCE[3]	..:Tarjeta_Codentrada = 12
	0015	..:Estado = <b>TRUE</b>	..:Timestamp = 12087
	0016	..:num_pasos_por_cero = 0	
	0017	BufferRCE[4]	..:Tarjeta_Codentrada = 2
	0018	..:Estado = <b>FALSE</b>	..:Timestamp = 41916
	0019	..:num_pasos_por_cero = 0	
	0020	BufferRCE[5]	..:Tarjeta_Codentrada = 1
	0021	..:Estado = <b>TRUE</b>	..:Timestamp = 41926
	0022	..:num_pasos_por_cero = 0	
	0023	BufferRCE[6]	..:Tarjeta_Codentrada = 4
	0024	..:Estado = <b>FALSE</b>	..:Timestamp = 41926
	0025	..:num_pasos_por_cero = 0	
	0026	BufferRCE[7]	..:Tarjeta_Codentrada = 3
	0027	..:Estado = <b>TRUE</b>	..:Timestamp = 41936
	0028	..:num_pasos_por_cero = 0	
	0029	BufferRCE[8]	..:Tarjeta_Codentrada = 6
	0030	..:Estado = <b>FALSE</b>	..:Timestamp = 41936
	0031	..:num_pasos_por_cero = 0	
	0032		
	0033		
	0034		
	0035		
	0036		
	0037		
	0038		
	0039		
	0040		
	0041		
	0042		
	0043		
	0044		

Figura 8.6. Visualización de los ensayos



Esta simulación refleja los cambios de estado de las entradas que manualmente hemos excitado. El tiempo que aparece en pantalla está calculado en milisegundos desde el arranque del PLC. El buffer creado es una simulación de lo que posteriormente será la propia memoria del autómatas. Verificar el correcto funcionamiento del programa no implica solamente ver que se almacenan correctamente los cambios producidos en las entradas sino también realizar las siguientes comprobaciones:

- Comprobar que en finalizar el registro de 2000 eventos, los siguientes se sobrescriben de nuevo empezando por las primeras posiciones de memoria.
- Comprobar que el contador de pasos por cero funciona correctamente al llegar a timestampMAX.
- Calcular el ciclo de scan.

Efectivamente, los ensayos realizados han sido superados con éxito. Pero para ver que en la realidad el ciclo de scan es suficientemente rápido (ya que tendremos unas 200 entradas digitales aproximadamente) encadenaremos una serie de relees en paralelo con la finalidad que en activarse uno de ellos (el primero) se desencadene una serie de saltos (activaciones) del resto de relees en cascada obteniendo, de este modo, una secuencia ordenada de eventos que serán registrados en el correcto orden si la especificación de nuestro programa está bien construida. Podremos comparar así los tiempos de retardo de dichos relees para poder posteriormente analizar este comportamiento y valorar el número de señales que como máximo se pueden registrar por milisegundo de forma totalmente cronológica. La estructura constará de una secuencia en cascada de 10 relees interconectados como muestra la figura 8.7. Mediante excitación manual, provocaremos la sucesión de eventos a registrar en el buffer de memoria creado en el algoritmo. En el Anexo XII se observan de los esquemas eléctricos que describen la interconexión de los relees utilizados en la simulación.

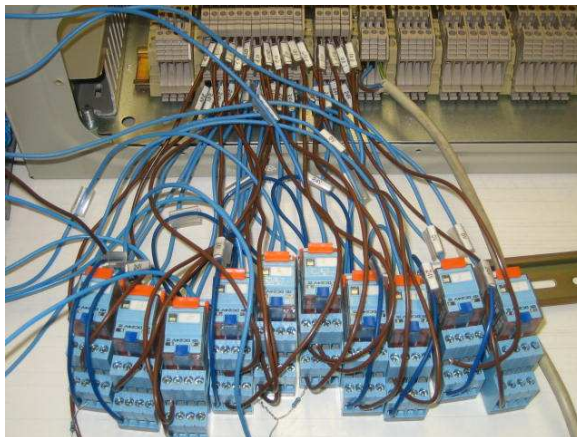


Figura 8.7. Interconexión de diez relees





Como se puede observar, existen relees que son activados al mismo tiempo de forma paralela como son por ejemplo los relees C, D y E, F, G. Con lo que los tiempos de detección de los eventos deben ser iguales. Compararemos los tiempos de registro de los relees cogiendo siempre como referencia el relee A, es decir, que si calculamos el retardo una señal obtenida en el relee H, éste deberá ser igual a 4 veces el valor del retardo de un relee. Aquellos relees que son excitados de forma paralela tendrán el mismo retardo como sucederá con los relees C, D y E, F, G.

Tras realizar el experimentos vemos que los tiempos de retardo de los relees son aceptables. Concretamente, el retardo en el momento de la activación (flanco de subida) es de unos 10 ms y el de desactivación (flanco de bajada) del orden de 2 ms.

En la simulación fue incluido un osciloscopio mediante el cual pudimos visualizar de forma más concreta los tiempos de activación y desactivación de los relees, realizando diversas pruebas para comprobar el buen funcionamiento de ellos. Con tal de tener certeza sobre los resultados obtenidos, realizamos diversas pruebas para ver que en todos los casos se cumplieran los tiempos esperados.

## 8.1. Ciclo de scan

Los tiempos de retardo de los relees así como el tiempo de ejecución del programa, que siempre es función de la cantidad de entradas, son los factores que determinan la duración del ciclo de scan, desde que se leen las entradas en un estado inicial hasta el envío, mediante un bus de comunicaciones, de la información a un procesador de datos.

Definir de forma concreta en ciclo de scan es fundamental para el correcto funcionamiento del autómatas y de todos los elementos ligados a él. Debemos acotar bien este ciclo para permitir la mayor velocidad de proceso del PLC así como posteriormente la transmisión de datos desde el PLC hacia cualquier estación de trabajo autorizada para ello.

Este tiempo de ciclo de scan se puede modificar de tal forma que se ajuste más a la velocidad crítica de ejecución. Esto conllevará la posibilidad de poder realizar dicho ciclo más veces por segundo. El software de **Beckhoff KS2000 v.4** permite ajustar este periodo tanto como sea posible sin que se genere un error en el PLC y, por lo tanto, realice el algoritmo de forma correcta, ya que se perderían eventos.



Dicho software detecta primeramente los dispositivos conectados en la red Ethernet para poder obtener la características del PLC y su entorno, tal y como se muestra en la figura 8.8.

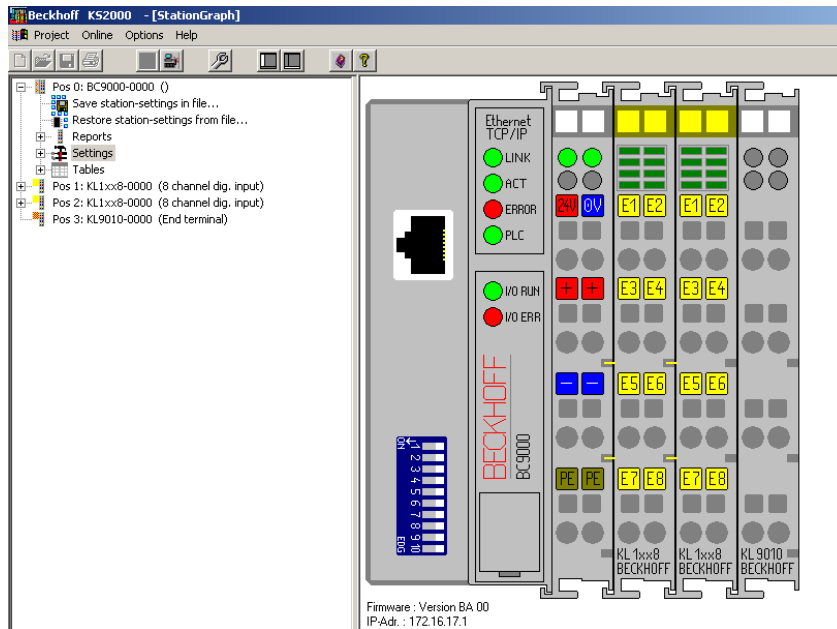


Figura 8.8. Entorno KS2000 – visualización del autómat BC9000

Activando la opción *Settings*, se habilitan una serie de aplicaciones de las cuales debemos seleccionar *Controller Settings*. En clicar sobre él observamos la siguiente ventana (figura 8.9):

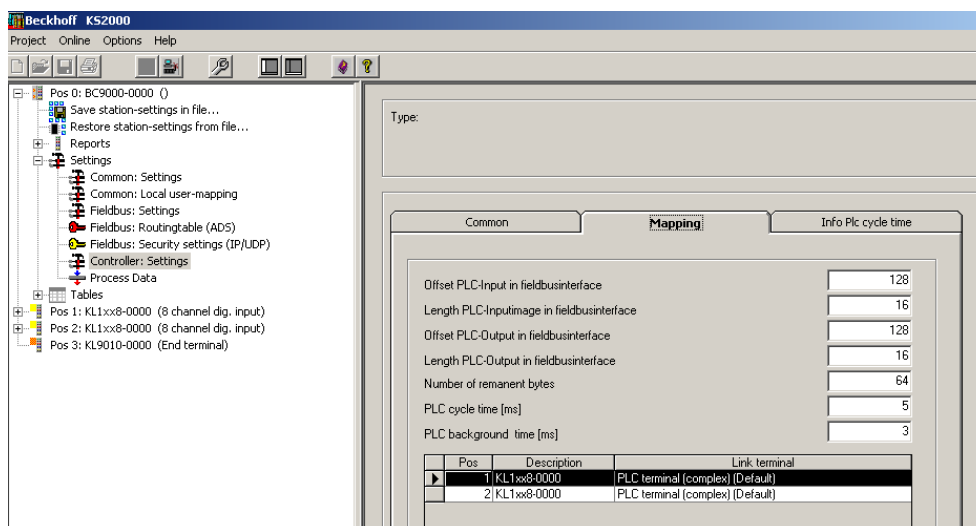


Figura 8.9. Ventana de descripción del ciclo de scan



Los valores por defecto del **PLC Cycle Time** y **PLC Background Time** son 10 y 5 ms, respectivamente. Tras los ensayos realizados, vemos que el ciclo máximo de scan que hemos tenido era de 6582  $\mu$ s. Eso nos indica que podríamos bajar ese tiempo de trabajo a unos 5 y 3 ms. Con estas especificaciones, los tiempos máximo, mínimo y la media de todos nos da unos valores como los mostrados en la figura 8.10:

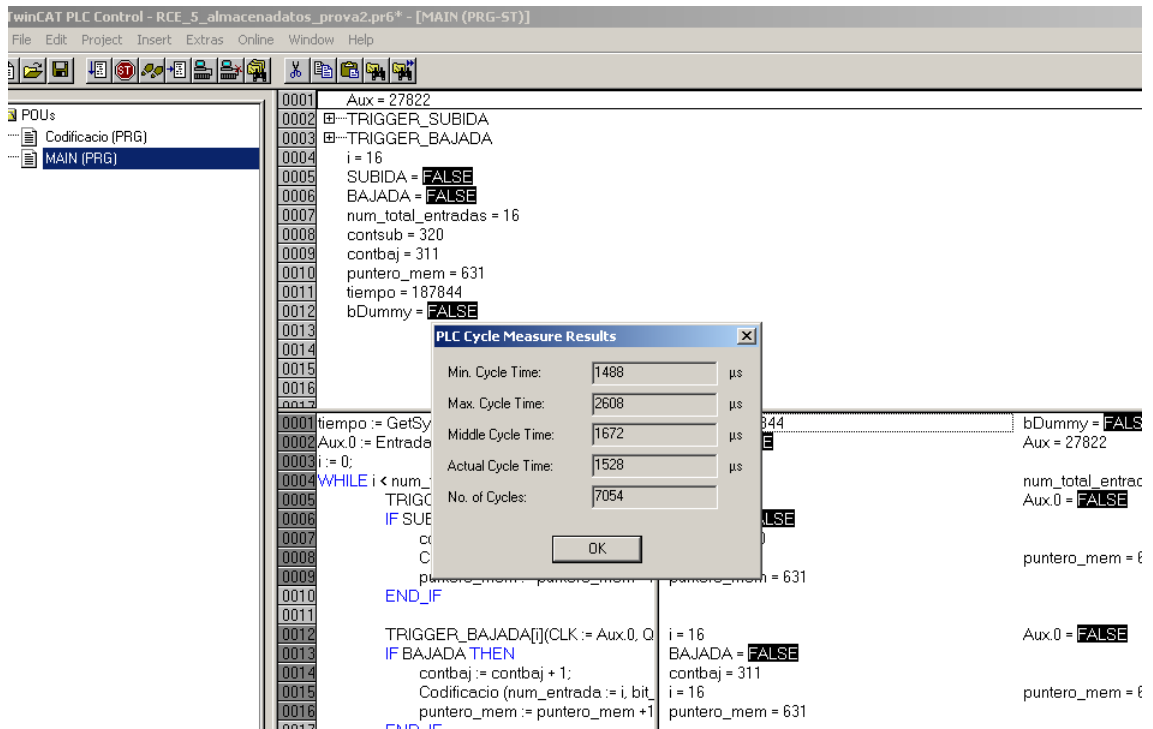


Figura 8.10. Resultados de las medidas de ciclo de scan del PLC

Pero la realidad de nuestro proyecto nos hace no fiarnos mucho de estos valores ya que esperamos alrededor de unas 200 entradas con lo que el trabajo de ciclo de scan se multiplica. Por tanto, intentaremos experimentar este hecho con cerca de unas 50 entradas con los 10 relees de los que disponemos.



## 8.2. Tratamiento en memoria del PLC

De tal forma que una vez almacenados los registros, podamos visualizar y descargar los eventos sucedidos en el sistema de AT de nuestra planta de cogeneración. Para ello crearemos un subprograma que copie la información procedente del buffer de almacenaje del RCE en la memoria interna del PLC.

Cada destacar el hecho que el autómatas BC9000 tiene una paginación de 4 Kbytes, esto significa que tan sólo podremos descargar la información en bloques de 4 en 4 Kbytes. Para ello, deberemos saber cual es la capacidad de este buffer para saber cuantos eventos podemos almacenar y, por lo tanto, descargar en la misma página de información:

4 Kbytes = 4000 bytes = 32000 bits de memoria en el PLC correspondientes a una página.

Cada evento, ocupa en memoria, 56 bits.

Con lo que, si podemos almacenar 2000 eventos según programa:

$2000 \text{ registros} \times 56 \text{ bits / registro} = 112.000 \text{ bits para almacenar los 2000 eventos.}$

Como solamente disponemos de 32000 bits de memoria, entonces:

$32000 \text{ bits} / (56 \text{ bits / registro}) = 571 \text{ registros por página.}$

Si queremos saber cuantas páginas nos ocuparan los 2000 eventos, calculamos:

$\text{Número de páginas} = \text{número total eventos} / \text{número de eventos por página}$

$\text{Número de páginas} = 2000 / 571 = 3,5 \approx \underline{4 \text{ páginas}}$

Para enviar un número constante de eventos mediante el bus de comunicaciones, para cubrir las cuatro páginas de las que disponemos, el número de sucesos enviados podrían ser de 500 cada uno, de este modo sabríamos que en cada emisión el número de datos transmitidos. Pero no debemos olvidarnos de la principal finalidad de la realización del algoritmo, la rapidez.



Transmitir un número elevado de datos ocupando en gran parte la totalidad de la capacidad del bus de comunicaciones. Implica la necesidad de un mayor tiempo de operación, con lo que el tiempo de ciclo de scan aumenta. Este hecho no es beneficioso para nuestro proyecto ya que buscaremos perder el menor tiempo posible en tareas, que posiblemente, podamos traspasar al programa del PC en el cual el tiempo de proceso ya no es tan crítico.

Diseñaremos, por lo tanto, un subprograma que enviará, mediante el bus de comunicaciones, tan sólo eventos de 10 en 10. De esta forma nos aseguraremos de no colapsar el bus de comunicaciones y, además, adaptaremos el programa al hecho que se hayan registrado 2000 eventos o más de tal forma que posteriormente serán ordenados cronológicamente. La principal ventaja de realizar este algoritmo según esta especificación, reside en el ahorro de tiempo de ejecución de programa, ya que el tiempo de ciclo de programa incluye el tiempo que se emplea para las comunicaciones y la principal finalidad de nuestro programa es que sea lo más rápido posible.

### **8.3. Algoritmo adquisición y tratamiento de 200 entradas**

El algoritmo tratado hasta el momento, contempla solamente la adquisición y procesamiento de 16 entradas digitales. Esto ha servido de prototipo para crear este primer programa que a continuación ampliaremos a la totalidad de las entradas digitales que en realidad tendremos en nuestra planta.

Las pruebas y comprobaciones han tenido un resultado correcto y un funcionamiento exactamente al esperado.

En el anexo IV se indican la totalidad de las entradas digitales de la planta de cogeneración que se obtendrán y deberán tratarse por nuestro algoritmo. En total 151 señales son las que se contabilizan procedentes de todos los relees, seccionadores, disyuntores, etc. que dispone el sistema de protecciones y control. Dado que el número de tarjetas digitales dispuestas en el cuadro de mando es de 25, eso significa que podremos llegar a tratar 200 entradas (tal y como se especificó en un principio), suficiente para abarcar las 151 señales de la planta. Cabe tener en cuenta, que el algoritmo de programación, existe un variable tipo WORD (16 bits) que usaremos para almacenar los estados de cada una de las entradas. Eso significa que necesitaremos 13 variables tipo WORD:  $200 / 16 = 12,5 \approx 13$  variables de tipo WORD → 208 entradas digitales



Como se observa, ha aumentado el número posible de señales digitales que se podrían tratar. Esto lo utilizaremos en un futuro, ya que al disponer de un mayor número de entradas, se podrá hacer frente a la posible ampliación de la central de cogeneración de COVAP y, por tanto, no hará falta añadir tarjetas digitales, a no ser que nos excediéramos en el número de señales, para el tratamiento de las señales que procederán de la instalación futura y, en consecuencia, tan sólo hará falta cablear aquellas señales nuevas que se quieran estudiar sin necesidad de retocar el cuadro de control RCE.

Con estas especificaciones, diseñaremos el programa del PLC para el tratamiento de todas las señales existentes en la planta existente y la parte ampliada de la instalación. El algoritmo final será exactamente igual al prototipo pero variando la adquisición de datos (en lugar una variable de entrada tipo WORD, tendremos 13 variables tipo WORD). Ver anexo XII.



## 9. Interface PC\_PLC

Los datos almacenados en la memoria del PLC quedan registrados en un buffer de datos que contiene los últimos 2000 eventos sucedidos hasta el momento. Como hemos comentado anteriormente, la transmisión de datos se realizará en bloques de 10 en 10 eventos para que dicha transmisión sea más rápida y no sature el buffer.

### 9.1. Entorno PC

Los datos almacenados en el autómata serán volcados una base de datos dentro del PC del usuario que solicite dicha información. Por lo tanto, será necesaria la interacción entre el PLC y el PC del consultor.

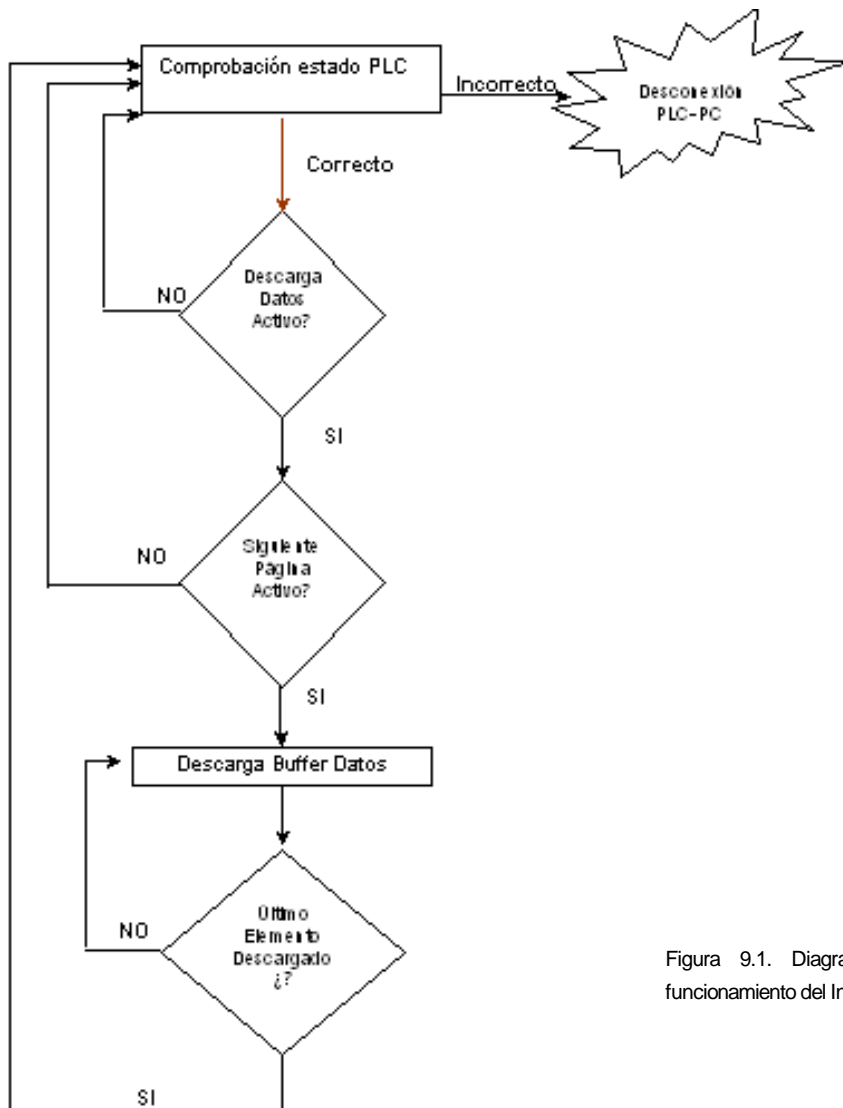


Figura 9.1. Diagrama bloques del funcionamiento del Interface PLC\_PC



## 9.2. Programación en Visual C+ 6.1

Este programador nos permitirá de forma fácilmente entendible, la creación de un algoritmo capaz interaccionar con el autómata y de esa forma, volcar los datos procedentes de la memoria del PLC, al PC del usuario.

El autómata elegido para la captación de datos contiene un puerto Ethernet de interconexión con la red, de tal forma que todo usuario con permiso de interconexión con este medio, tiene acceso a la información almacenado en el PLC. Este echo, permitirá al entorno de PC establecer una conexión con el PLC via Ethernet para, así, volcar los datos almacenados en su memoria.

El funcionamiento del programa consistirá en:

1. Reconocimiento del autómata utilizado.
2. Detección de la dirección IP del BC9000
3. Activación de las variables de descarga de datos
4. Volcado de datos en un vector dentro de la memoria del PC.
5. Tratamiento de PC.

## 9.3. Tratamiento de datos

Una vez hemos volcado todos los datos en el vector de almacenamiento del PC, presentaremos dichos eventos en una base de datos muy útil y fácil para su posterior tratamiento. Es decir, traspasaremos todos los sucesos a un fichero Excel para poder, de esta forma, tratarlos y procesarlos a voluntad del consultor.





Para ello, tan solo cabe tener en cuenta que en la transmisión de los datos, como el buffer tiene una capacidad máxima de 2000 eventos, se pueden diferenciar dos casos:

- A. No se llega a ocupar toda la capacidad del buffer
- B. Se superan los 2000 eventos y, por tanto, al ser un buffer circular, los últimos eventos se sobrescriben empezando de nuevo por la primera posición de memoria.

Solucionar este problema de forma global resulta relativamente sencillo. El programa de PC almacena la fecha del último evento copiado, de tal forma que tomará como referencia esa fecha para preguntarse, en primera instancia, si la fecha del siguiente evento que ha leído desde el PLC es superior a la actual. Si eso ocurre, dicho evento será considerado apto para su volcado en el fichero Excel, si no cumple con ese requisito, se procederá a la lectura del siguiente suceso.

Con este sistema, conseguimos la transmisión y el volcado de los eventos de forma cronológica y evitando posibles señales duplicadas. [Ver anexo XIII]



## 10. Presentación de los resultados

Los datos registrados por el autómata son descargados en la estación de trabajo, PC o dispositivo con conexión ethernet que permita la ejecución del programa en Visual C, de tal forma que se crea en la base de datos un fichero .xls (Excel) en donde se recogen los eventos sucedidos en el sistema eléctrico de la planta de cogeneración.

Para poder identificar la procedencia de las señales, el programa **Interface PLC\_PC** contiene un fichero de configuración en el que se recogen la ubicación exacta de cada una de las señales del cuadro de control.

Mediante este fichero de configuración el operario o consultar sabrá de forma inmediata de que relee se trata y el motivo exacto de su activación/desactivación.

### 10.1. Base de datos

La configuración del fichero de históricos que almacena los eventos descargados consta de una primera fase de filtraje de tal forma que le sea presentado al usuario de una forma sencilla y práctica para su posterior tratamiento y estudio. De tal forma, la distribución del archivo será la siguiente:

Evento numero	de entrada	Estado	Timestamp(ms )	pasos_cer o	reserv a	Fecha suceso	Hora del suceso	Milisegundo s suceso	Descripción
1	4	1	12206	0	0	30/10/2006	17:23:06	419	La señal detectada num 4 procede de la tarjeta 1, bit num 4 correspondiente al relee C.
2	5	1	12206	0	0	30/10/2006	17:23:06	419	La señal detectada num 5 procede de la tarjeta 1, bit num 5 correspondiente al relee D.
3	7	1	12206	0	0	30/10/2006	17:23:06	419	La señal detectada num 7 procede de la tarjeta 1, bit num 7 correspondiente al relee F.
4	8	1	12206	0	0	30/10/2006	17:23:06	419	La señal detectada num 8 procede de la tarjeta 2, bit num 0 correspondiente al relee E.
5	10	1	12206	0	0	30/10/2006	17:23:06	419	La señal detectada num 10 procede de la tarjeta 2, bit num 2 correspondiente al relee F.
6	13	1	12206	0	0	30/10/2006	17:23:06	419	La señal detectada num 13 procede de la tarjeta 2, bit num 5 correspondiente al relee H.

Evento numero	de entrada	Estado	Timestamp(ms )	pasos_cer o	reserv a	Fecha suceso	Hora del suceso	Milisegundo s suceso	Descripción
1	4	1	12206	0	0	30/10/2006	17:23:05	620	La señal detectada num 4 procede de la tarjeta 1, bit num 4 correspondiente al relee C.
2	5	1	12206	0	0	30/10/2006	17:23:06	620	La señal detectada num 5 procede de la tarjeta 1, bit num 5 correspondiente al relee D.
3	7	1	12206	0	0	30/10/2006	17:23:06	620	La señal detectada num 7 procede de la tarjeta 1, bit num 7 correspondiente al relee F.
4	8	1	12206	0	0	30/10/2006	17:23:06	620	La señal detectada num 8 procede de la tarjeta 2, bit num 0 correspondiente al relee E.
5	10	1	12206	0	0	30/10/2006	17:23:06	620	La señal detectada num 10 procede de la tarjeta 2, bit num 2 correspondiente al relee F.
6	13	1	12206	0	0	30/10/2006	17:23:06	620	La señal detectada num 13 procede de la tarjeta 2, bit num 5 correspondiente al relee H.
7	0	0	0	0	0	30/10/2006	17:22:54	414	La señal detectada num 0 procede de la tarjeta 1, bit num 0 correspondiente al relee ND.
8	0	0	0	0	0	30/10/2006	17:22:55	414	La señal detectada num 0 procede de la tarjeta 1, bit num 0 correspondiente al relee ND.
9	0	0	0	0	0	30/10/2006	17:22:55	414	La señal detectada num 0 procede de la tarjeta 1, bit num 0 correspondiente al relee ND.
10	0	0	0	0	0	30/10/2006	17:22:55	414	La señal detectada num 0 procede de la tarjeta 1, bit num 0 correspondiente al relee ND.

Evento numero	de entrada	Estado	Timestamp(ms )	pasos_cer o	reserv a	Fecha suceso	Hora del suceso	Milisegundo s suceso	Descripción
1	4	1	12206	0	0	30/10/2006	17:23:06	40	La señal detectada num 4 procede de la tarjeta 1, bit num 4 correspondiente al relee C.
2	5	1	12206	0	0	30/10/2006	17:23:06	40	La señal detectada num 5 procede de la tarjeta 1, bit num 5 correspondiente al relee D.
3	7	1	12206	0	0	30/10/2006	17:23:07	40	La señal detectada num 7 procede de la tarjeta 1, bit num 7 correspondiente al relee F.
4	8	1	12206	0	0	30/10/2006	17:23:07	40	La señal detectada num 8 procede de la tarjeta 2, bit num 0 correspondiente al relee E.
5	10	1	12206	0	0	30/10/2006	17:23:07	40	La señal detectada num 10 procede de la tarjeta 2, bit num 2 correspondiente al relee F.

Figura 10.1. Formato presentación de los datos.



Como se ve en la figura 10.1 los datos se vuelcan y se presentan en una primera instancia en bloques de 10 en 10 eventos. Posteriormente, se engloban todos los sucesos en una misma hoja de históricos en los que se recogen de forma cronológica todos eventos registrados en el buffer del PLC.

De esta forma cada usuario tratará los datos en base a las necesidades de su estudio. El fichero de históricos también incluye una evolución porcentual de cada una de las señales con la finalidad de saber cuales de los relees, interruptores y demás dispositivos suelen presentar mayor número de incidencias, así como la frecuencia de aparición y la situación en la que se produjo, es decir, si la activación / desactivación de la protección se debió a una situación de puesta en marcha, isla, fallo externo, etc.

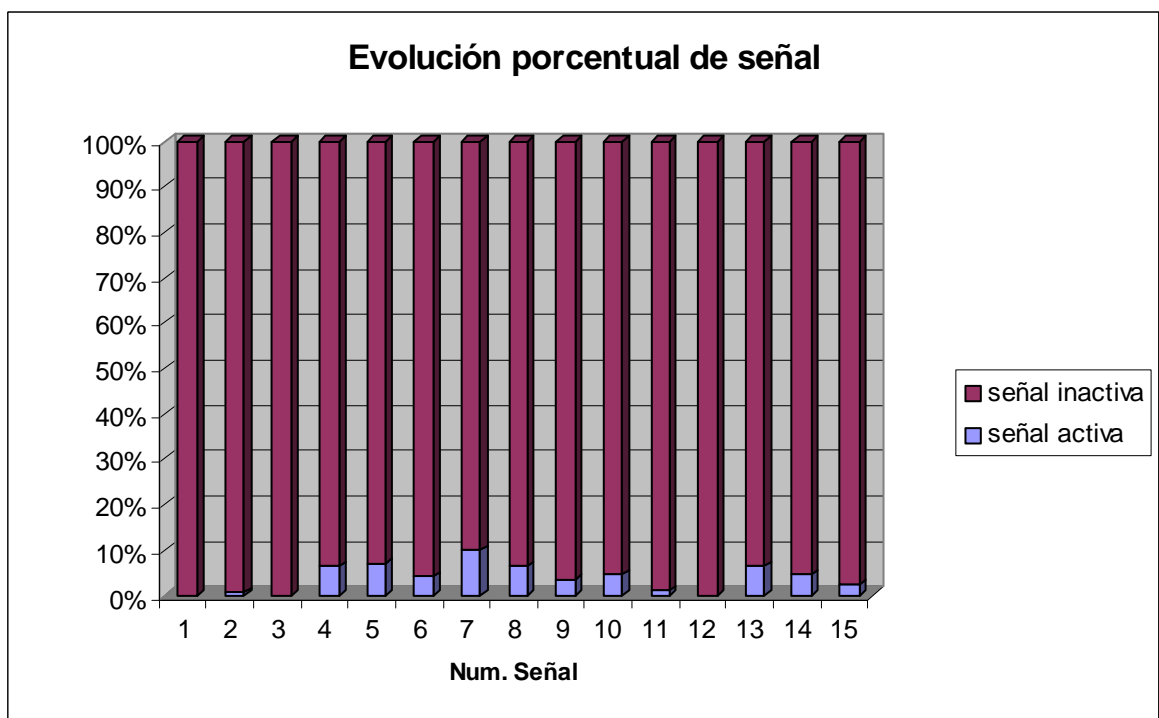


Figura 10.2. Resumen porcentual de la evolución histórica de las señales digitales

Estas conclusiones serán de fácil análisis gracias a los históricos del fichero, su porcentaje y su ubicación exacta.



## Conclusiones

Tras la construcción del cuadro de control y la posterior puesta en marcha, tanto del programa del PLC como el de PC, hemos podido corroborar el buen funcionamiento de ambos sistemas y el cumplimiento de los objetivos marcados en un inicio.

Se ha optado finalmente por no incorporar la pantalla visualizadora ya que ello conllevaría la complicación del propio cuadro y por el diseño y practicidad del proyecto hemos decidido prescindir de ello.

Se ha conseguido que el RCE funcione a una velocidad de scan dentro de los márgenes previstos de tal forma que resulte mucho más veloz que los SAD's hasta ahora existentes y, con ello, poder tener un mayor detalle de la procedencia de las fallas ocurridas dentro de una instalación de AT / BT de forma concreta.

Trasportar los datos recogidos en el buffer del PLC a la memoria interna del PC y la posterior creación de una base de datos en base excel, permite a los consultores y analistas encargados de la supervisión de los sistemas de control de la planta, tener una visión más detallada debido que el detalle de señales recogidas es mucho mayor al actual y, por lo tanto, se puede analizar con más detalle cual ha sido realmente el origen de la falla en el sistema eléctrico, con lo que el posterior tratamiento, no tan sólo de la señal sino también de las protecciones implicadas, sea más sencillo debido al conocimiento real causante de la caída de dicho dispositivo.

En conclusión, verificado el buen resultado de este sistema, el paso siguiente sería llevar este sistema a la totalidad de las señales de la planta, es decir, la elaboración de un cuadro de control más sofisticado dotado de un receptor de señales tanto digitales como analógicas dentro del sistema eléctrico así como el termodinámico. Con ello se conseguirá un control total de la planta de forma totalmente específica, detallada y de fácil análisis para los controladores de planta.



## Agradecimientos

Este proyecto no hubiera podido ser posible sin la ayuda de todo el equipo de **SIGE S.A** (Gestión Informática) que han estado apoyándome dentro de la elaboración de este proyecto, y que con sus consejos, conocimientos y, sobretodo, con su tiempo han hecho posible la creación de lo que ahora en adelante se integrará en todos los futuros proyectos tanto de ampliación de plantas, como de nueva implantación.

También agradecer a la empresa **AESA S.A** (Asesoría Energética) toda la colaboración de su equipo para la especificación del sistema eléctrico, así como la descripción del cuadro de protecciones del sistema eléctrico. AESA ha proporcionado toda esta información así como los planos de los esquemas unifilares de la planta de cogeneración, del sistema de protecciones y demás sistema eléctricos.

Finalmente, sin la colaboración de estas dos empresas no hubiera sido posible la realización de este proyecto y, con lo cual, debo agradecer a todos los componentes de las diversas secciones tanto de **SIGE** como de **AESA** la colaboración prestada, y a todos aquellos que puntualmente han incidido con comentarios, observaciones o simplemente con críticas constructivas que tan bien han hecho en este proyecto y que ha colaborado su éxito.



## Bibliografía

### Bibliografía complementaria

- [1] WEIDMÜLLER. *Bornes*. Catálogo 2005 / 06
- [2] GE POWER MANAGEMENT. *Sistema Numérico de Protección, Monitorización, Análisis y control de la energía. SMOR – B. Instrucciones*.
- [3] GE POWER MANAGEMENT. *Sistema integrado de Protección y Control. DDS . Instrucciones*
- [4] GE POWER MANAGEMENT. *Moduladores de tensión . TOV serie 1000. Instrucciones*
- [5] GE POWER MANAGEMENT. *Protección Digital de Frecuencia . MFF serie 1000 . Instruc*
- [6] GE POWER MANAGEMENT. *SEG XG2 relee de salto de vector*
- [7] GE POWER MANAGEMENT. *Protección Digital de Alimentador . MIF . Instr*
- [8] GE POWER MANAGEMENT. *Protección Dig Tal de alimentador monofásica . MIF N . inst*
- [9] BECKHOFF. *New Automation Technology / 2006*
- [10] BECKHOFF. *Documentation for BC9000 and BC9100*
- [11] MICROSOFT PRESS. *Microsoft Visual Basic 6.0 Guide. . Microsoft Corporation 1995*

