

2) **Conversión A/D**: etapas que la constituyen, descripción de las mismas. ¿cuáles son las especificaciones más relevantes de un A/D?

Todo conversor A/D, incorpora las siguientes etapas:

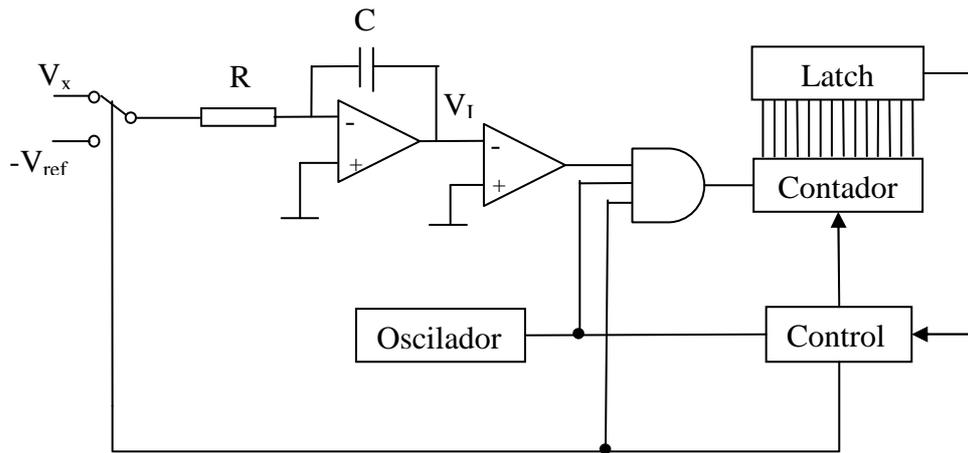
- **Muestreo**: es la encargada de tomar, a intervalos regulares, una muestra de la señal analógica de entrada que va a ser convertida a binario.
- **Cuantificación**: es la encargada de decidir en qué franja de valores analógicos comprendidos entre el mínimo y el correspondiente a plena escala, debe ser ubicado el valor analógico muestreado en la etapa anterior.
- **Codificación**: es la etapa que asigna un código binario de tantos bits como posea el conversor, a la franja de valores en la que se encuentra el valor muestreado.

Especificaciones:

- **Resolución**: Es el menor cambio en la entrada que incrementa o decrementa el código de salida al próximo consecutivo. Depende de la cantidad de bits que maneje el conversor.
- **Error de cuantificación**: Es una consecuencia directa de la resolución del conversor.
Todas las tensiones analógicas dentro de un cierto rango, son representadas por el mismo código binario de salida.
Si el punto medio del rango se asume como el valor nominal, habrá un error de cuantificación de $\pm 1/2$ del bit menos significativo (LSB).
- **Error de offset**: Diferencia entre la salida de un conversor ideal y la del real para 0V de tensión de entrada.
- **Error de plena escala**: Diferencia entre la salida de un conversor ideal y la del real, para una tensión de entrada de plena escala.
- **Error de linealidad**: Máxima desviación respecto de una línea recta que pase por los puntos extremos de las curvas de transferencia.
- **Tiempo de conversión**: Tiempo necesario para efectuar una conversión. Cuanto más bits se utilicen para un determinado rango analógico, menor será el error de cuantificación o sea que será mayor la resolución.

3) Explicar un método de **conversión A/D**.

El método de conversión más frecuentemente usado es el de **integración con doble rampa** cuyo esquema es el siguiente:



Durante la primera rampa, el condensador se carga con una pendiente proporcional al valor de la tensión de entrada (V_x). Durante este tiempo el comparador habilita el paso de pulsos del oscilador hacia el contador. La carga del capacitor se interrumpe cuando el contador llega a su alcance máximo.

La salida del integrador será:

$$V_I = -V_x \cdot \frac{t}{R \cdot C}$$

Finalizando la rampa en el tiempo t_0 en que el contador alcanza el valor N_0 .

Durante la segunda rampa, el condensador se descarga desde el valor alcanzado hasta 0V con una pendiente proporcional a la tensión de referencia ($-V_{ref}$).

$$V_I = -V_x \cdot \frac{t_0}{R \cdot C} + V_{REF} \cdot \frac{t}{R \cdot C}$$

Durante el tiempo de descarga, el contador cuenta los pulsos del oscilador, siendo el valor final de la tensión de salida del integrador: $V_I=0$

Por lo que:

$$-V_x \cdot \frac{t_0}{R \cdot C} + V_{REF} \cdot \frac{t_N}{R \cdot C} = 0 \implies \text{De donde: } t_N = \frac{V_x}{V_{REF}} \cdot t_0$$

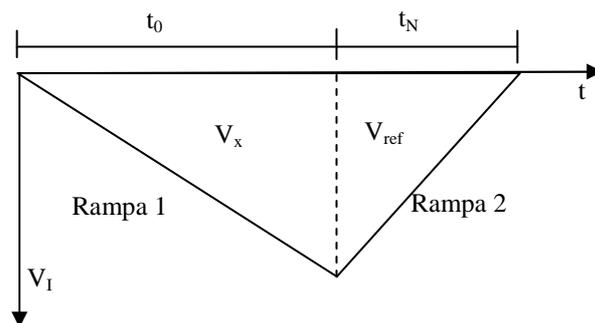
t_N y t_0 , son proporcionales al período del clock y al nº de pulsos contados por el contador durante esos tiempos, o sea:

$$t_N = N \cdot T_{ck} \quad \text{y} \quad t_0 = N_0 \cdot T_{ck}$$

Reemplazando:

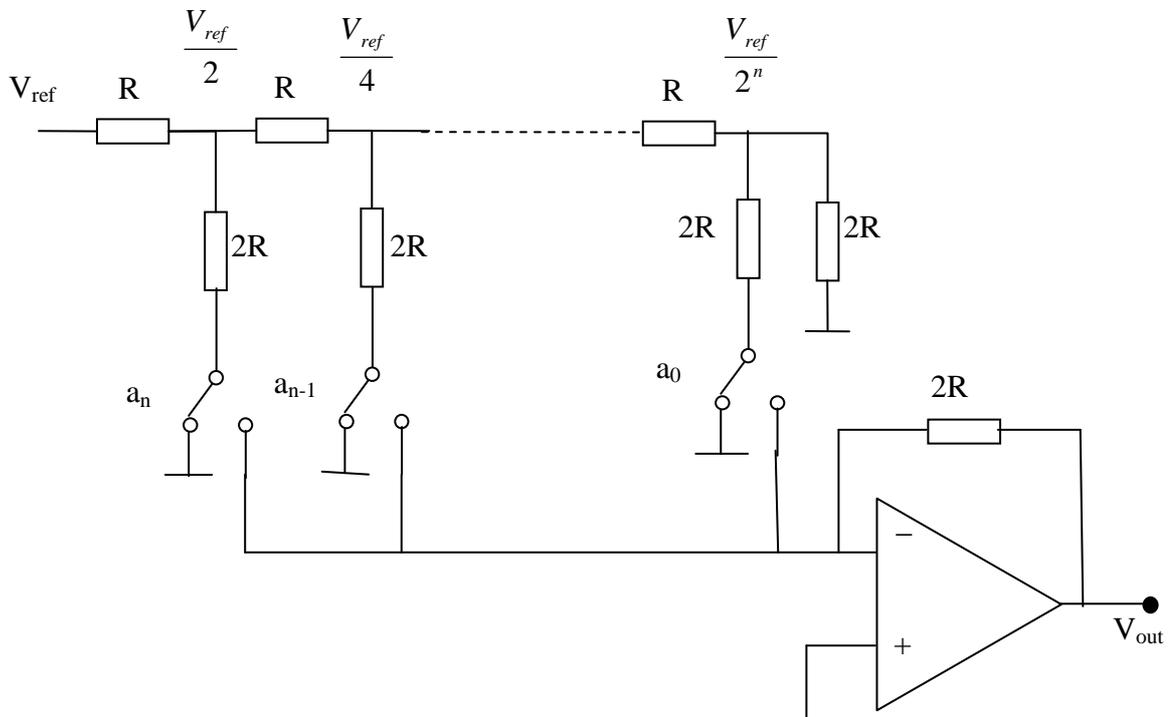
$$N = \frac{V_x}{V_{REF}} \cdot N_0$$

Expresión que nos muestra que el valor alcanzado por el contador, es proporcional a la tensión de entrada.



4) Conversión D/A: explicar un método de conversión.

Método R-2R:



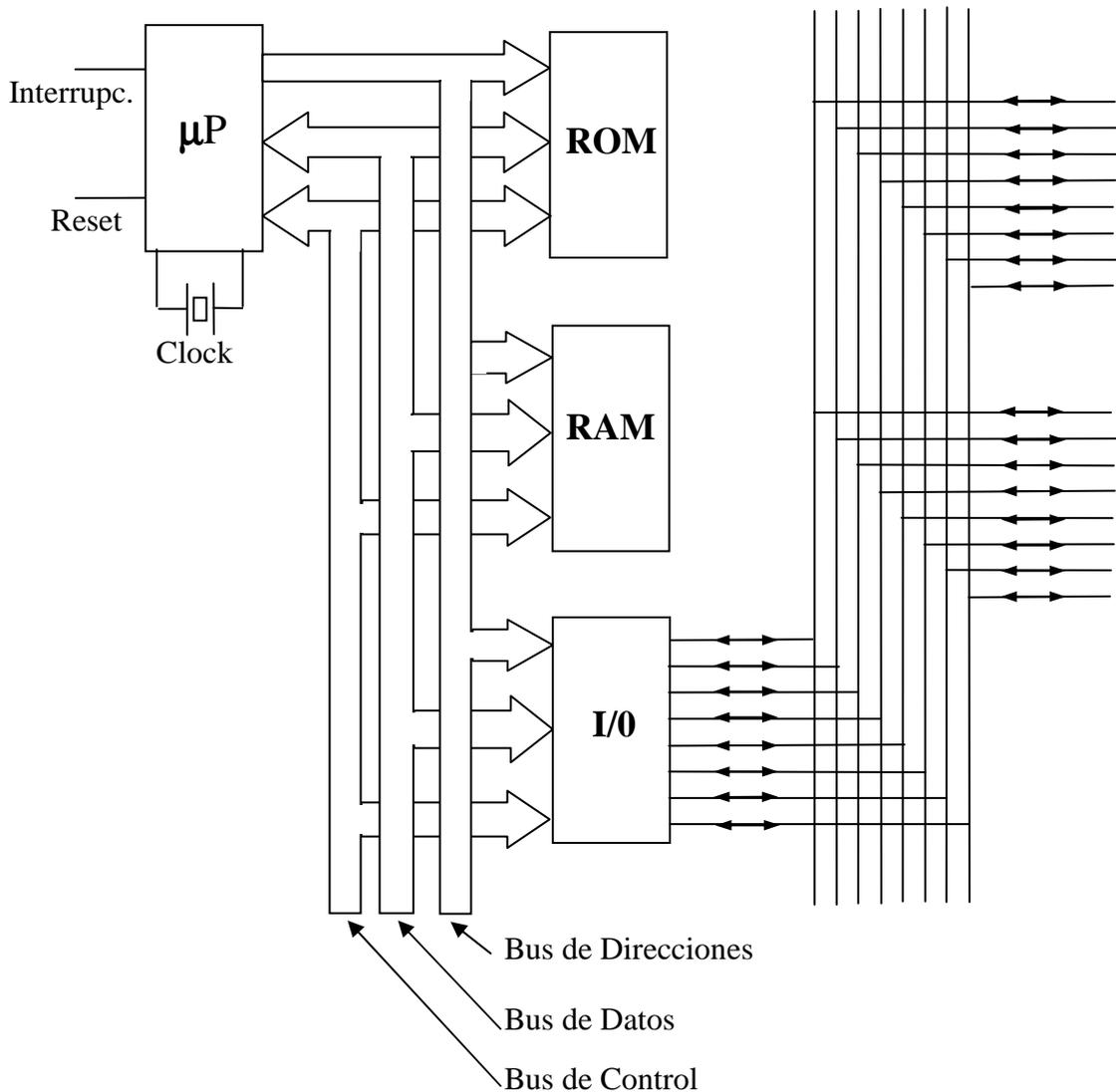
La resistencia equivalente vista hacia la derecha de cualquier resistencia R es siempre igual a R por lo que la tensión se divide por dos en cada nodo. Cada llave electrónica es comandada por un bit del número binario a convertir y se activarán cuando el bit correspondiente esté en "1". La tensión de salida será igual a:

$$V_0 = V_{ref} (1/2 \cdot a_n + 1/4 \cdot a_{n-1} + \dots + 1/2^n \cdot a_0)$$

Siendo n la cantidad de bits del conversor y los coeficientes a_n valdrán "0" o "1" según el estado del bit asociado.

5) Estructura interna de un PLC: diagrama en bloques y descripción.

Como puede apreciarse en el siguiente esquema, un PLC posee todos los bloques existentes en toda computadora, ya que el mismo, no es otra cosa que una computadora diseñada para aplicaciones de control industrial.



6) Tipos de memorias y sus usos en un PLC.

- **ROM:** Su contenido es grabado durante su fabricación y es inmodificable. Sólo puede ser leído. Su uso está restringido al almacenaje del programa del fabricante.
- **RAM:** Su contenido puede ser leído y modificado por el microprocesador en cualquier momento pero se borra al quedar sin alimentación. Se usa normalmente para almacenar variables del proceso como estados de E/S, conteo de tiempos y contadores, operandos y resultados de operaciones matemáticas, etc. Soportada por baterías, puede almacenar el programa del usuario y variables que deban conservarse ante la falta de tensión de red.
- **EPROM:** Son memorias ROM programables y borrables. Una vez grabadas conservan su información aún sin alimentación. Para modificar su contenido deben ser previamente borradas con luz ultravioleta. Se usan normalmente para al-

macenaje externo del programa del usuario como copia de respaldo. Lo más común, hoy en día, para esta aplicación, es salvar el programa del usuario en algún medio de almacenamiento magnético u óptico (diskettes, CD's, etc.)

- **EEPROM:** Ídem anterior pero no es necesaria la luz ultravioleta para su borrado pero sí de un circuito eléctrico especial para su reprogramación, el cual puede estar incluido en el PLC. Esto permite guardar el programa del usuario en el mismo PLC y mantenerlo indefinidamente o hasta tanto se decida cambiarlo o actualizarlo. Existen en la actualidad PLC's que permiten almacenar varios programas de usuario simultáneamente.

7) Definir programa y explicar las diferencias entre programa del fabricante y programa del usuario.

El conjunto de instrucciones que puede interpretar un microprocesador con el código binario que le corresponde a cada instrucción es lo que se llama el **set de instrucciones** del mismo.

El programa será una serie de dichas instrucciones (lenguaje máquina) que determinará la tarea que debe realizar el microprocesador. El programa debe estar previamente almacenado en memoria para poder ser ejecutado.

El **programa del fabricante** es el encargado de procesar al programa del usuario y administrar el flujo de información desde y hacia los módulos de entrada/salida. Reside normalmente **en ROM**.

El **programa del usuario** es el encargado de determinar qué debe hacer el PLC con la información de las entradas/salidas. Reside normalmente **en RAM** soportada por batería, **en EPROM o EEPROM**.

8) ¿Cuál es la función del **watch-dog** y cómo lo logra?

Todo programa real se cierra sobre sí mismo ejecutando el mismo conjunto de instrucciones una y otra vez.

Ésto determina que haya un tiempo máximo dentro del cual una determinada instrucción debe ser ejecutada cíclicamente, lo que puede ser usado para determinar si el programa está siendo procesado en la secuencia correcta.

Si una determinada instrucción estratégicamente ubicada en el programa, no es ejecutada dentro de un tiempo preestablecido, el PLC asume que **el programa salió de su secuencia** (se colgó), por lo que detiene el funcionamiento del PLC y desactiva sus salidas.

9) ¿Cómo funciona la **protección de memoria**?

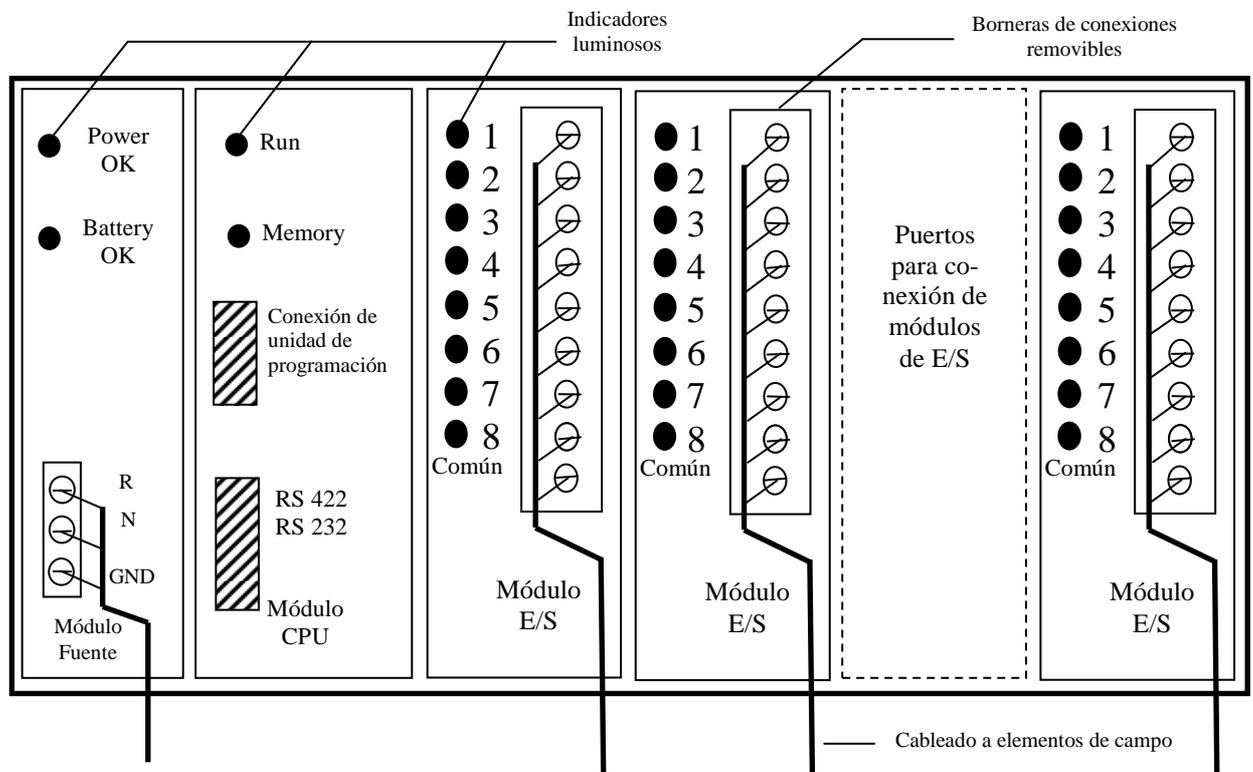
Si el contenido de la memoria de **programa de usuario se altera** por cualquier circunstancia, el mismo sería ejecutado en forma errónea con consecuencias impredecibles.

Para evitar esto, junto con el programa de usuario, el PLC calcula y almacena un **código de detección de error** resultante del mismo.

Durante la rutina de puesta en marcha que reside en el programa del fabricante, este código es recalculado y comparado con el original. De no coincidir ambos, **el programa del usuario no será ejecutado** (será necesario recargar el programa).

10) **Esquema modular** del PLC y su descripción.

La siguiente figura es un bosquejo de la estructura típica de un PLC.



Todos los elementos del sistema controlado son conectados a los módulos de entrada o salida del controlador.

Los indicadores luminosos (**LEDs**) indican, en los módulos de entrada, si se recibe tensión en la entrada correspondiente, y en los módulos de salida, aquellas que estén activas.

El **módulo de fuente** contiene todos los circuitos necesarios para obtener las tensiones requeridas por el equipo (normalmente +5V, +12V, -12V), reguladas para mantenerlas dentro de las tolerancias necesarias y filtrar los ruidos de red. Usualmente la regulación se realiza por conmutación (switching).

El LED "POWER" encendido implica no sólo que el equipo está encendido sino que además las tensiones internas están comprendidas dentro de las tolerancias.

El LED "BATERÍA" permite supervisar el estado de la batería de soporte de la memoria RAM (si el PLC no cuenta con memoria EEPROM) con tiempo suficiente para su reemplazo antes de que se pierda la información en memoria.

Algunos PLC no usan batería sino capacitares de gran valor. Esto implica una menor autonomía ante falta de tensión de red, del orden de 48 hs. comparada con meses o años de las baterías. La corriente de fuga de los capacitares aumenta con la temperatura por lo que su autonomía disminuye al aumentar ésta.

El módulo de CPU (Central Processor Unit) contiene al microprocesador, las memorias, circuitos para la comunicación con la unidad de programación y con otros equipos, etc.

El LED "RUN" (funcionando) encendido implica que el programa se está ejecutando correctamente. Si está apagado, el PLC está detenido con las salidas desactivadas. La causa puede deberse a la acción del watch-dog (en este caso el problema desaparece al apagar y reencender el equipo) o a una falla en el chequeo de la memoria del programa.

11) ¿Qué es el mapa de memoria? Explicar cómo lo emplea el PLC mediante un ejemplo.

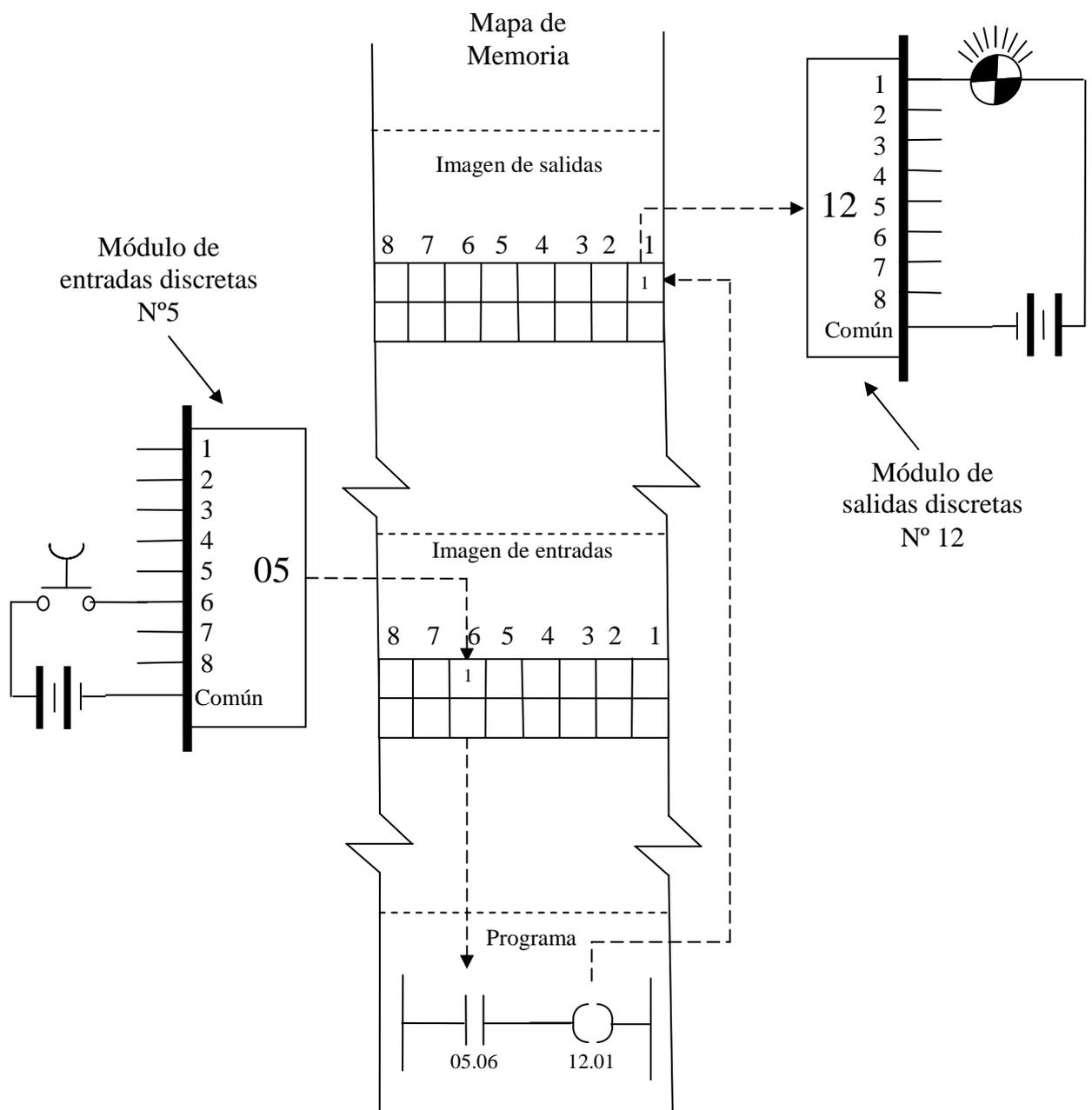
Toda la información sobre el estado de entradas/salidas se encuentra almacenada en memoria y el programa accede a ella a través de su dirección simbólica.

El estado de cada entrada o salida discreta (si/no) se corresponderá con el estado de un bit en memoria.

Las variables numéricas como las provenientes de módulos analógicos, estarán representadas por un conjunto de bits denominado palabra (word) donde la cantidad de bits por palabra dependerá del PLC. A cada palabra le corresponderá un único código de dirección.

Un ejemplo sería, como se ve en la figura siguiente, el módulo de entrada sensando la presencia o ausencia de tensión en cada uno de sus bornes.

Estos estados son transferidos a las direcciones de memorias asociadas al módulo de entrada en cada ciclo de lectura de entradas.

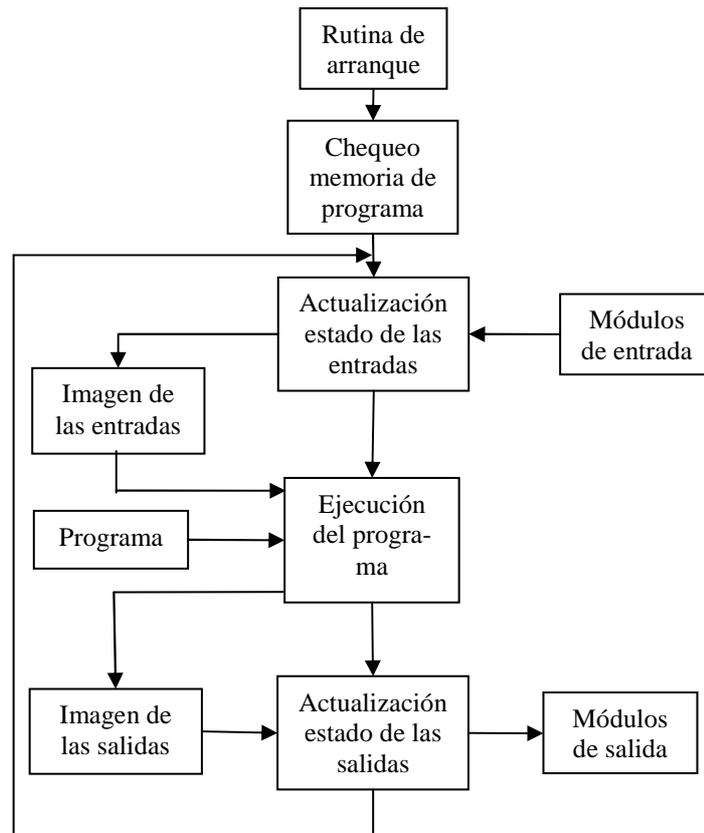


Particularmente, la entrada 6 del módulo 5 está recibiendo tensión en su borne a través del pulsador conectado a él. Este hecho se refleja con un "1" en la dirección de memoria que le corresponde.

Cuando se ejecuta el programa de usuario, se consulta el estado de este bit en memoria, lo que determina que la salida 1 del módulo 12 debe estar activada. Esto resulta en un "1" en la dirección de memoria correspondiente.

Por último, una vez procesado todo el programa, el estado de esta memoria es transferido al módulo propiamente dicho resultando en el encendido de la lámpara conectada al borne 1.

12) Descripción del funcionamiento interno de un PLC mediante un flujograma.



13) ¿Qué es el tiempo de barrido (scan) y de qué depende?

Es el tiempo que demora el PLC en cumplir un ciclo del programa de usuario (incluyendo lecturas de entradas y actualización de salidas). Este ciclo tiene un tiempo propio que dependerá de la cantidad de módulos de entrada/salida, de la longitud del programa y de la velocidad del PLC. Si por alguna razón este tiempo es excedido, actuará el watch-dog deteniendo al controlador.

El máximo tiempo que puede transcurrir entre la activación de una entrada y su reflejo en las salidas (tiempo de scan) será igual al tiempo de retardo del módulo de entrada más 2 veces el tiempo de ejecución del programa (en la primera vez lee las imágenes de entradas y salidas y en la segunda ejecuta el programa propiamente) más el retardo del módulo de salida.

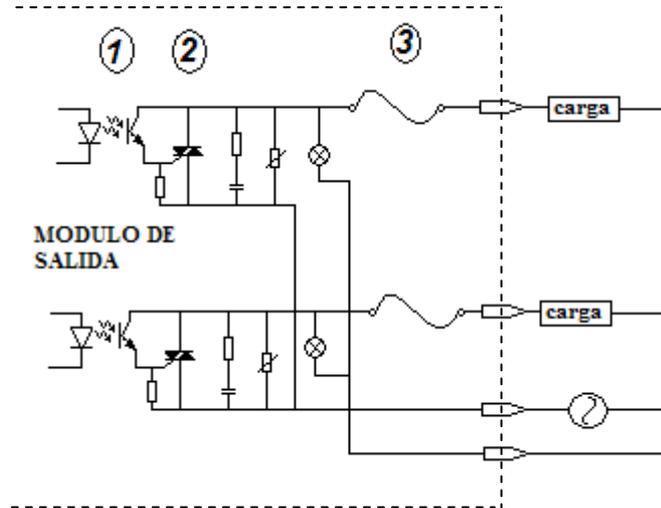
Se tendrá una seria limitación para aplicar un PLC a procesos que requieran un tiempo de respuesta menor al tiempo de scan.

14) Mencionar qué opciones comerciales existen para módulos de E/S discretas, qué etapas los constituyen y su función.

Las opciones comerciales para módulos de E/S discretas son de 5Vcc, 24 Vcc, 48 Vcc, 110 Vca, 220 Vca, etc.

Módulos de salidas discretas:

En la siguiente figura vemos el esquema de un módulo de salidas a Triac:



Las etapas que constituyen a un módulo de salidas discretas son:

- 1) Aislación galvánica (Circuito optoacoplador)
- 2) Elemento de conmutación (relé, triac, transistor)
- 3) Elementos de protección contra sobretensiones y sobrecorrientes y de señalización.

En el módulo se pueden distinguir dos sectores. Uno de ellos es aquel que tiene una conexión con la carga y que maneja la tensión proveniente de campo. El otro es el que se conecta con el bus de E/S del PLC y que está alimentado por la fuente del equipo.

El acoplamiento más frecuente entre los dos sectores es óptico. Cuando el controlador debe activar la carga, enciende un LED que está encapsulado formando un único dispositivo con un fototransistor el que recibe la luz emitida por el LED. Esto activa el tercer terminal (compuerta) del Triac habilitando la conducción entre los otros dos terminales y activando la carga.

El elemento que maneja la carga normalmente es el Triac para tensiones alternas o un transistor para continua o relés para ambas tensiones. Ambos pueden ser vistos como un dispositivo de tres terminales donde dos de ellos se comportan como una llave comandada por el tercer terminal.

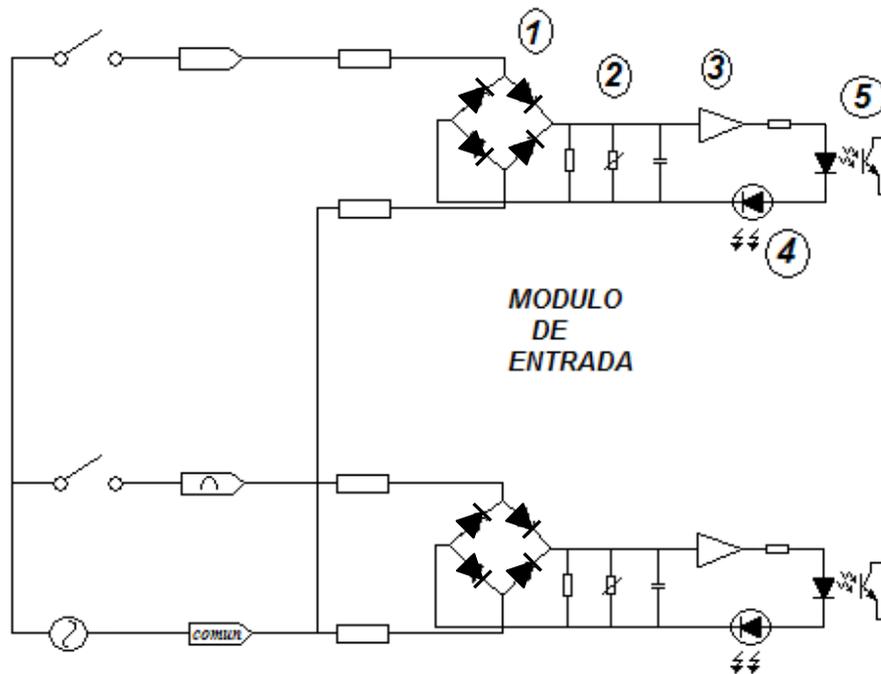
El Triac y el transistor son más sensibles que un relay a sobretensiones y sobrecorrientes. Cualquiera de estas causas puede provocar un cortocircuito permanente en el dispositivo.

Las sobretensiones pueden ser el resultado de ruidos en el cableado a campo o por la desconexión de cargas inductivas.

El varistor y la red RC (snoober) minimizan la probabilidad de que sea superada la máxima tensión de pico del triac

Módulos de entrada discretos:

Los circuitos de entrada son menos susceptibles de ser dañados pero son sensibles al ruido e interferencias.

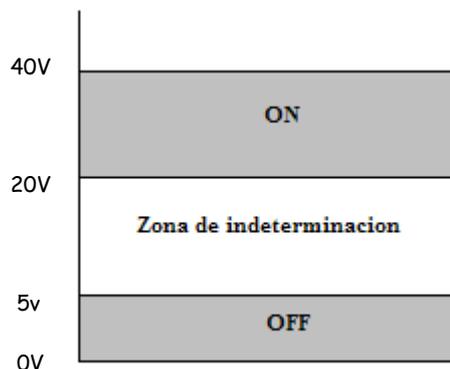


Las etapas que constituyen a un módulo de entradas discretas son:

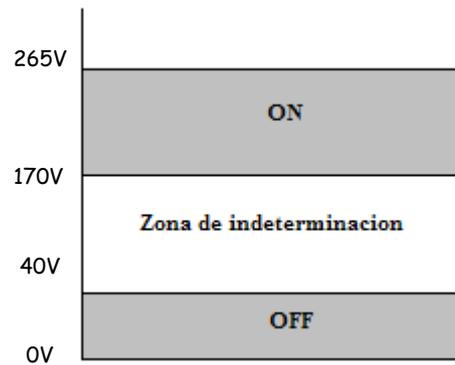
- 1) Circuito **rectificador** tipo puente, para aceptar tanto señales de CA como de CC y en este último caso, sin importar su polaridad.
- 2) Elementos de **protección y filtrado** de señal
- 3) Circuito de **adaptación de señal** (Schmitt Trigger)
- 4) Elementos de **señalización**
- 5) **Optoacoplador**: Conexión al bus E/S del PLC mediante aislación galvánica.

Otro factor a tener en cuenta, son los **valores de tensión admisibles** en la entrada. A título de ejemplo se dan las siguientes figuras:

Módulo de entrada de 24 Vcc:



Módulo de entrada de 220 Vac:

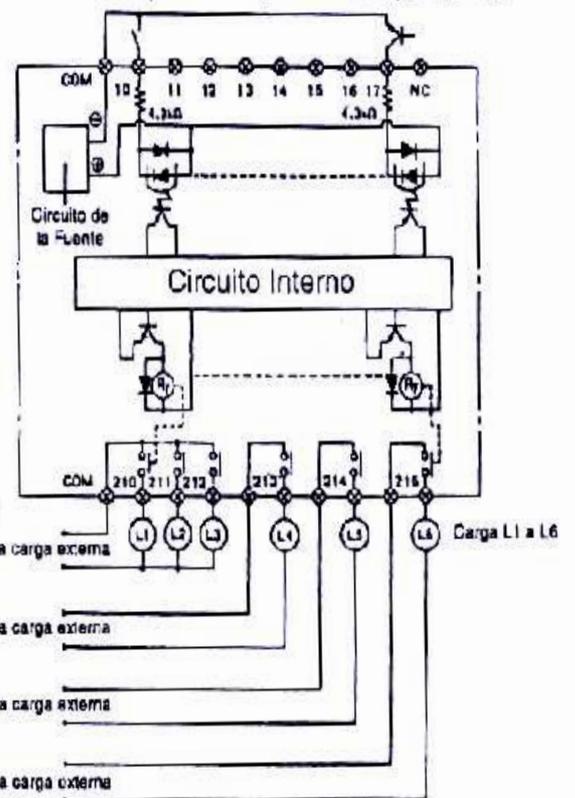
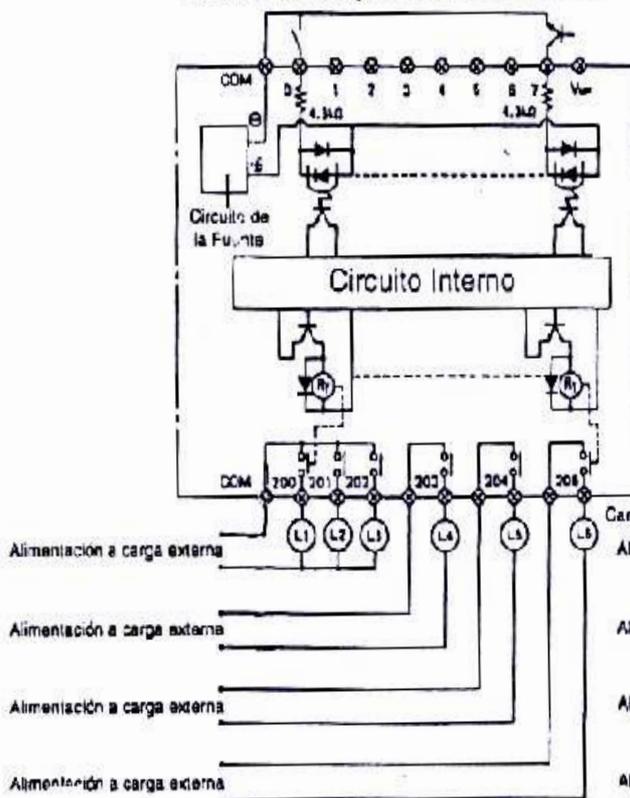


Módulos típicos de entradas y salidas discretas:

MICRO-1

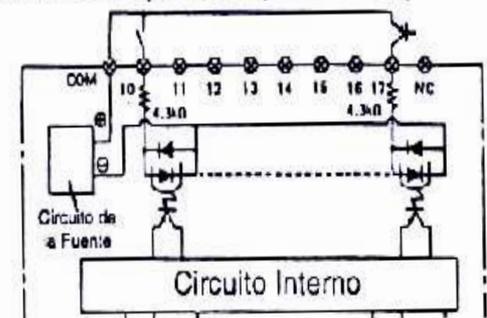
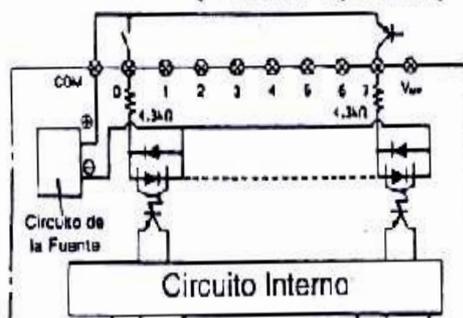
Unidad base (Entrada Tipo NPN)

Unidad de expansión (entrada tipo NPN)



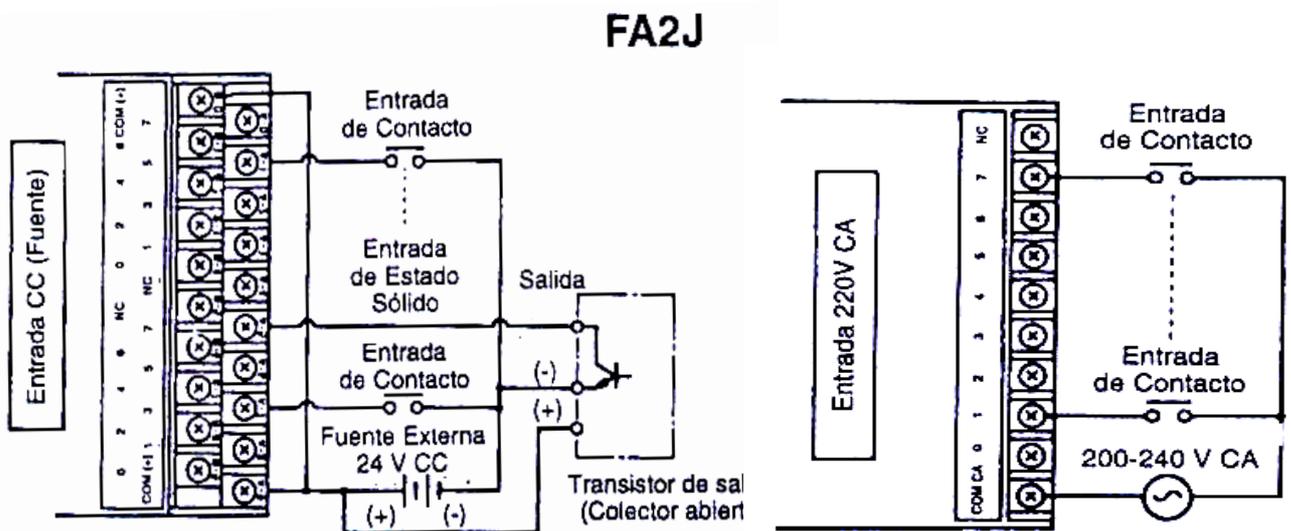
Unidad Base (Entrada Tipo PNP)

Unidad de Expansión (Entrada Tipo PNP)



En este tipo de PLC (MICRO-1) lo que se ve es que en las entradas tipo NPN la corriente que sale del positivo circula a través de los diodos y llega a la salida pasando antes por una resistencia. El colector del transistor va conectado a la salida del PLC que llega desde el positivo y el emisor va conectado al común. En las entradas vemos que tenemos la posibilidad de conectar llaves de distintos tipos. La unidad de expansión trabaja de la misma manera que la unidad base.

En la entrada tipo PNP vemos que se invierte la manera de estar conectada la fuente, ya que el positivo de la fuente va hacia la salida común del PLC, y el negativo se dirige a los diodos y pasando por una resistencia llega a la salida, en donde en este caso tenemos conectado el colector del transistor y una llave de contacto. El emisor del transistor ira conectado al positivo, o sea al común del PLC.



Nota 1: No conecte la salida u otro cable al terminal NC.

Nota 2: Los terminales COM (+) no están conectados entre sí internamente.

En este tipo de PLC (FA2J) vemos que la salida puede ser de CC (corriente continua) como de CA (corriente alterna). En el caso de que en el PLC de CC no nos sirva utilizar la fuente interna de 12V, tenemos la posibilidad de conectarle una fuente externa del voltaje que necesitemos (como se puede ver en el gráfico), la cual tendrá conectado el positivo al común del PLC y el negativo de la fuente va a los elementos que deseemos conectar al PLC.

En el PLC de corriente continua vemos que en una de las salidas hay conectado un transistor NPN, el cual tiene conectado el emisor al negativo de la fuente externa y el colector a la salida del PLC. También está la posibilidad de que el transistor sea PNP, por lo que tendremos que conectar el emisor del transistor al positivo y el colector a la salida del PLC.

Direccionamiento de elementos discretos - Ejemplos comerciales:

Simatic S105-R5:

Cada puesto para conexión de módulo de entrada/salida (slot) dentro del rack se numera de izquierda a derecha con los números 0 a 7.

Dentro de cada módulo los terminales se numeran de arriba hacia abajo comenzando por el número 0.

El código comienza con una letra que indica si el terminal correspondiente es una entrada o una salida.

En la versión europea corresponde una A para salidas y una E para entradas. En la versión americana corresponde una O para salidas y una I para entradas.

A continuación va el número que identifica al slot dentro del rack y luego separado por un punto, el número que identifica al terminal dentro del módulo.

Ejemplo: E3.5 (I3.5) es una entrada que corresponde al cuarto módulo comenzando por la izquierda y al sexto terminal dentro de ese módulo.

Texas TI525C:

Este PLC permite asignar por programa la numeración de la primer entrada/salida de cada módulo al configurar el sistema.

Las siguientes entradas/salidas de un determinado módulo tendrán una numeración correlativa ascendente hasta completar la capacidad del mismo.

Asimismo se antepone al número mencionado una letra que indica si es entrada o salida correspondiendo la X para entradas y la Y para salidas.

Ejemplo: Si a una determinado módulo de 32 entradas se le asigno como primera dirección el número 50, las direcciones de dichas entradas irán desde la X50 a la X82.

Modicon 884:

De forma similar al anterior, se puede asignar el primer número de cada módulo durante la configuración del sistema y las siguientes entradas/salida del mismo módulo seguirán una numeración correlativa ascendente desde dicho número.

La dirección completa estará compuesta por cinco dígitos donde el primero determina si el elemento es entrada o salida correspondiendo el cero para salidas y el uno para entradas.

Los restantes cuatro dígitos corresponden al número de elemento dado en la configuración del sistema.

Ejemplo: Si se ha asignado la dirección 15 al primer elemento de un módulo de entradas de 16 elementos, las direcciones de dichas entradas irán desde 10015 hasta 10030.

Saia PCA14:

Las entradas/salidas se designan simplemente con un número de 0 a 63 en correspondencia con la ubicación del terminal de conexión.

Telemecanique TSX 17-20:

La dirección comienza con una letra correspondiendo la "I" a las entradas y la "O" a las salidas. A continuación va un número que será 0 si es una entrada/salida del rack base y un 1 si corresponde a un rack de extensión. Separado de lo anterior por una coma va un número de dos dígitos que es el número de entrada o salida.

Hitachi P500E:

Cada módulo tiene hasta 16 salidas discretas y se numeran de izquierda a derecha comenzando por cero.

Las entradas o salidas dentro de cada módulo se numeran de arriba hacia abajo de 0 a 15.

La dirección comienza con una letra correspondiendo X para entradas e Y para salidas.

La dirección completa estará dada por la letra inicial, luego el número de módulo y por último el número de entrada/salida.

Ejemplo: Y202 es una salida que corresponde al tercer módulo y a la tercera salida dentro del módulo.

15) ¿Para qué puede emplearse un módulo de cuenta rápida y qué etapa elimina?

Los filtros introducidos para brindar inmunidad, provocan una respuesta lenta (del orden de 10ms).

En los módulos de entrada denominados rápidos, este filtro es eliminado o tiene una constante de tiempo mucho menor. Lógicamente esto implica una menor inmunidad al ruido lo que obligará a tener especial cuidado en el cableado al sensor (cable blindado).

16) ¿Qué precauciones se deben tener con un módulo de salida discreto?

Habrá que tener cuidado con cargas tipo lámparas donde la resistencia en frío es una centésima de la resistencia en caliente con lo que en cada encendido se produce una sobrecorriente de magnitud suficiente para provocar la falla del semiconductor o su degradación progresiva.

En las salidas a triac con cargas puramente resistivas, se deberá tener en cuenta la velocidad de crecimiento de la corriente (dI/dt en A/ μ seg) que puede provocar calentamientos localizados con degradación progresiva del mismo.

El fusible de protección deberá respetar el I^2t del original si se pretende proteger al semiconductor (fusibles ultrarrápidos).

La corriente de fuga del dispositivo sumada a la red de RC y del varistor provocan una circulación de corriente aún en condiciones de circuito abierto que pueden llegar a activar cargas de muy bajo consumo. En estos casos se deberá agregar una resis-

tencia en paralelo con la carga tal que multiplicada por la corriente de fuga (corriente en estado OFF), arroje un valor de tensión suficientemente baja.

Frente a **sobrecargas, sobrecalentamiento o sobretensiones**, puede fundirse la pastilla semiconductora quedando el **dispositivo en cortocircuito**. En este caso, el PLC se verá imposibilitado de cortar la alimentación a la carga. Esto deberá ser tenido en cuenta cuando se manejen cargas cuya falla a una condición no segura pueda provocar daño a las personas.

Si el semiconductor trabaja en condiciones **fuera de tolerancia**, puede no fallar en forma inmediata, pero su tiempo de vida se reduce drásticamente provocando que la salida involucrada quede periódicamente fuera de servicio y deba ser reparada.

17) ¿Qué etapas constituyen a los módulos de E y S analógicos o numéricos?

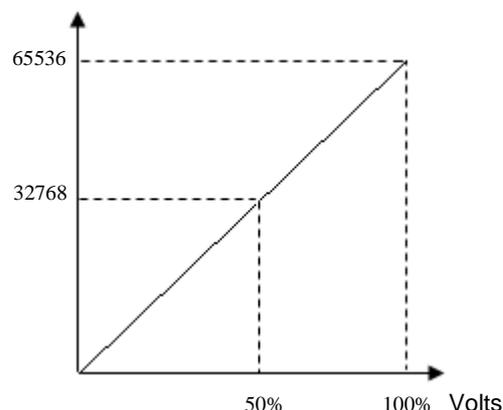
Módulos de entrada analógicos:

Cada módulo permite la conexión de varias entradas analógicas del mismo tipo y se multiplexan para entrar al único convertor A/D del módulo. Por lo tanto, estos módulos poseen una **etapa de conversión A/D**, de una determinada cantidad de bits y un **multiplexor** cuyo N° de entradas depende de la cantidad de entradas del módulo.

El valor de tensión de cada entrada es almacenado en la **palabra de memoria** correspondiente (un registro) y el programa accede a este último valor mediante su dirección.

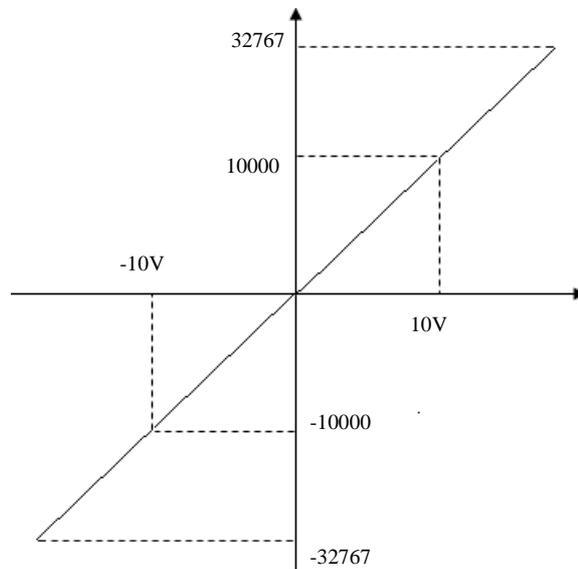
La correspondencia entre el valor de tensión de entrada y el número que la representa internamente puede responder a dos criterios. En el primero, se aprovecha todo el rango numérico que permite la cantidad de bits que posee la palabra donde se almacenará el valor, asignando el menor número posible al menor valor de tensión manejado por el módulo y el máximo número al máximo valor de tensión:

Registro de 16bits sin signo



Otra posibilidad es la asignación a cada valor de la tensión de entrada, de un número que exprese su valor en forma directa:

Registro de 16bits con signo

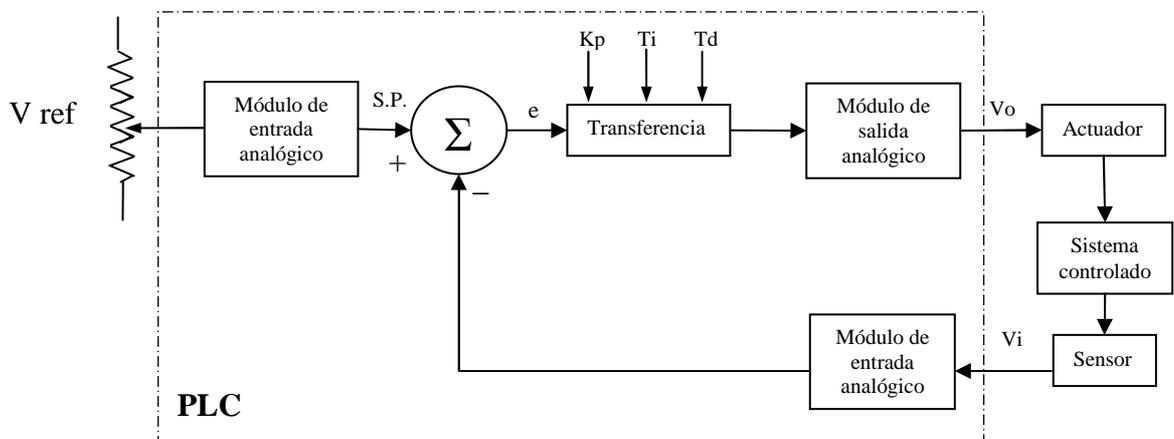


Módulos de salida analógicas:

Cada módulo permite la conexión de varias salidas analógicas del mismo tipo las que se multiplexan para usar el único convertor D/A del módulo. Por lo tanto, estos módulos poseen una **etapa de conversión D/A**, de una determinada cantidad de bits y un **demultiplexor** cuyo N° de salidas depende de la cantidad de salidas del módulo. El valor de tensión de cada salida corresponde al valor numérico almacenado en la palabra de memoria correspondiente, la que es escrita por el programa mediante su dirección.

18) Diagrama en bloques de un SCLC que incorpora un PLC (explicación).

Este control se basa en un sistema de lazo cerrado ya que la acción de control depende de la salida real. Como vemos en el diagrama de bloques, la señal de realimentación es comparada con la entrada, para producir la señal de error.



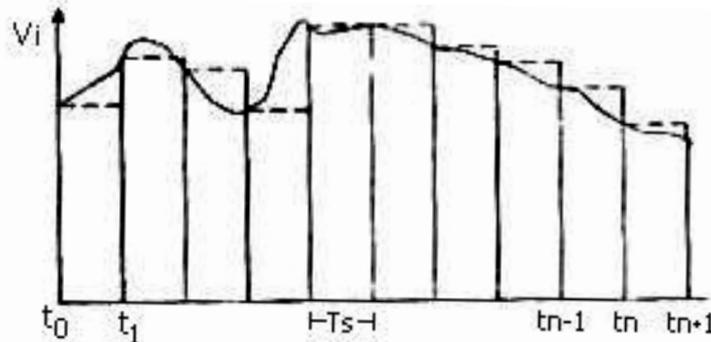
SP = Set point (Valor deseado de la variable controlada) = V_{ref}

V_i = Realimentación del verdadero valor de la variable controlada

V_o = Salida de control

e = error = $SP - V_i$

Dada la característica digital del control donde la conversión A/D, la conversión D/A y el procesamiento del algoritmo de control llevan un tiempo distinto de cero, la señal de entrada es muestreada con un intervalo de tiempo regular llamado **tiempo de muestreo (Sampling Time)**.



Como vemos en la Fig., el control asume que **la señal realimentada se mantiene constante entre dos muestras consecutivas**.

Esto implica que no se podrá controlar un proceso que varíe significativamente durante dicho intervalo. Dicho en otras palabras, **los tiempos de reacción del sistema controlado deberán ser mucho mayores que el tiempo de muestreo (T_s)**.

Asimismo, la salida del control variará por escalones manteniéndose constante entre dos muestras consecutivas.

19) ¿Qué algoritmos de control se conocen y cómo influye cada uno sobre la salida controlada?

a) Control Proporcional:

En este tipo de control, la salida en un momento dado será proporcional al valor del error en ese instante.

$$V_{on} = K_p \cdot e_n + K_o$$

V_{on} = tensión de salida en el instante n

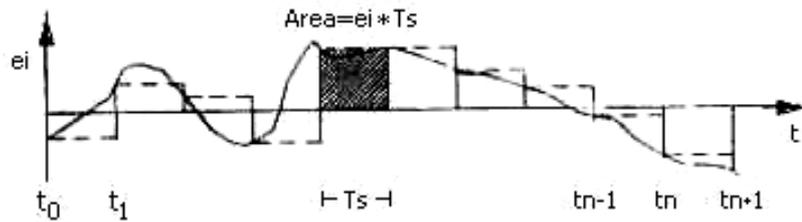
e_n = valor del error en el instante n

K_p = Ganancia de la transferencia

K_o = Constante correspondiente al valor de la salida para error cero

b) Control Integral:

En este caso, la salida en un momento dado será proporcional a la integral del error en el tiempo desde cero hasta ese instante



Dado que el error se considera constante entre dos muestras, la integral será la sumatoria de las áreas de los rectángulos formados entre muestras.

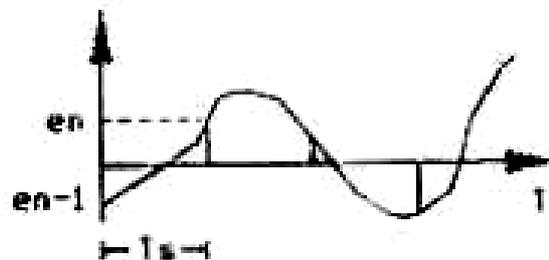
$$I = \sum_{i=0}^n Ts \cdot e_i = Ts \cdot \sum_{i=0}^n e_i$$

Para obtener la tensión de salida, se afecta al valor de la integral por un factor llamado Tiempo Integral (TI) que determinará su peso dentro del control.

$$V_{on} = (TS/TI) \cdot \sum e_i$$

c) Control derivativo:

La salida en un instante dado será proporcional al valor de la derivada del error en el tiempo en ese instante.



Se considera que el error ha variado linealmente durante el intervalo entre dos muestras consecutivas. En este caso, la derivada será:

$$D = (e_n - e_{n-1}) / Ts$$

La tensión de salida se obtiene afectando este término por un factor llamado tiempo derivativo (TD) que determinará el peso de la parte derivativa dentro del control.

$$V_{on} = (e_n - e_{n-1}) \cdot TD / Ts$$

d) Control PID:

El algoritmo completo para un lazo de control Proporcional-Integral-Derivativo (PID) será:

$$V_{on} = K_p \left[e_n + (T_s / T_I) \cdot \sum_{i=0}^n e_i + (e_n - e_{n-1}) \cdot (T_D / T_s) \right] + K_0$$

Para la implementación del control deberán definirse las direcciones de los siguientes registros cuyas palabras representan:

SP: Set Point

Vi: tensión realimentada

Vo: tensión de salida

Kp: ganancia proporcional

TI: tiempo integral

TD: tiempo derivativo

Todos estos valores salvo Vi y Vo pueden definirse como constantes durante la programación o ser ingresados desde fuera del PLC vía módulos analógicos o módulos BCD

El tiempo de muestreo (Ts) queda fijado por el sistema y normalmente dependerá de la cantidad de lazos activos que maneje el controlador.

El control P-I-D es el tipo de control más general y más utilizado. Proporciona una respuesta rápida, un buen control de la estabilidad del sistema y un bajo error de régimen permanente.

Características de los controladores P, I, y D:

El controlador proporcional tendrá el efecto de reducir el tiempo de crecimiento y reduce (pero no elimina) el error de estado estable. El control integrativo tendrá el efecto de eliminar el error de estado estable, pero sin embargo podría empeorar la respuesta transitoria. El control derivativo tendrá el efecto de disminuir el sobreimpulso, mejorando la respuesta transitoria del sistema aunque puede aumentar su inestabilidad (agrega un cero que reduce el "tipo de sistema").

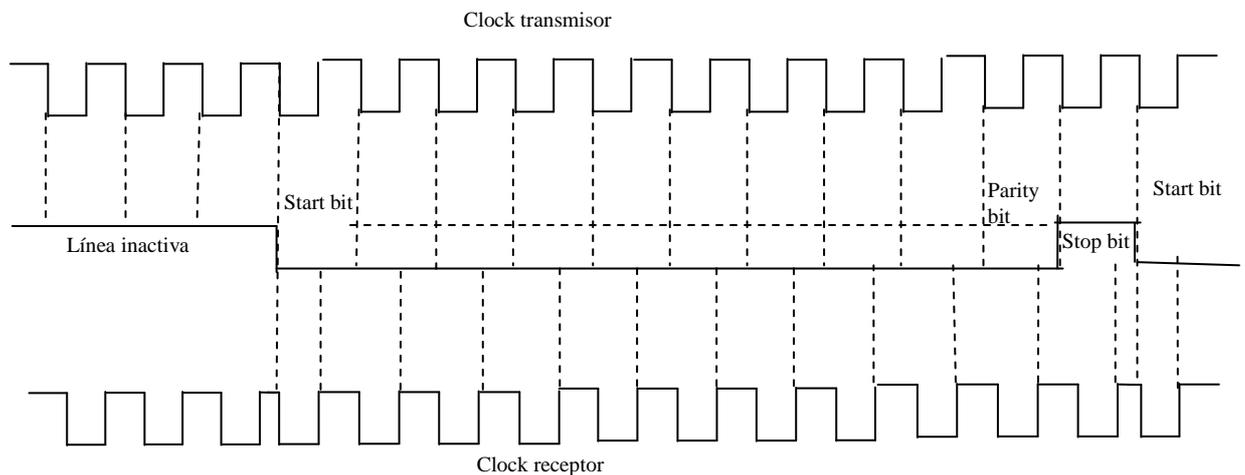
20) En una comunicación serie asincrónica, explicar cómo se sincronizan Tx y Rx y en qué parámetros se deben poner de acuerdo.

La información en el PLC es almacenada en forma binaria. Para transmitir un byte o una palabra binaria a través de un sólo cable, los bits deben ser enviados o recibidos uno por uno distribuidos en el tiempo.

Para transmitir, tanto el transmisor como el receptor ajustan sus tiempos con un oscilador de precisión del que se deriva el clock, determinando la velocidad de transmisión de datos.

La velocidad de transmisión se mide en baudios que se puede definir como la cantidad de variaciones de la señal en el canal por unidad de tiempo. En nuestras aplicaciones coincidirá con los bits por segundo.

Las velocidades más usuales son 300, 600, 1200, 2400, 4800, 9600 y 19200 baudios. La siguiente figura muestra la forma en que se mandan los datos desde el transmisor en sincronismo con el clock y la forma en que los mismos son leídos en el receptor.



Mientras no hay tráfico el transmisor mantiene la línea en "1". Antes de comenzar el envío de los bits de datos, coloca la línea a "0" durante un clock. Esto es lo que se denomina bit de comienzo (start bit) y el receptor utiliza este flanco para sincronizar su clock con el del transmisor.

A continuación se envía una tanda de **7 u 8 bits de datos**, uno por cada flanco de clock. La cantidad de bits de datos enviados por cada tanda una vez seleccionada en ambos extremos permanece constante. La división de la información en tandas es propia de la comunicación asincrónica y se procede de esta forma para permitir la resincronización de ambos osciladores al comienzo de cada una de ellas.

Después de cada tanda se manda un bit de detección de errores llamado bit de paridad (Parity bit). El transmisor determina el valor que debe tener este bit de acuerdo a la cantidad de "1" que se hayan enviado en la tanda. El bit agregado debe mantener la paridad elegida.

Se puede elegir entre dos **tipo de paridad**:

- Paridad par (Even) : Si la cantidad de "1" enviados fue par, se envía un "0" como bit de paridad para mantener la cantidad de unos par.
- Paridad impar (Odd) : Si la cantidad de "1" enviados fue impar, se envía un "1" como bit de paridad para mantener la cantidad de unos impar.

El receptor recalcula este bit luego de recibir la tanda y lo compara con el recibido. De no coincidir, rechaza la información y pide retransmisión.

Por último se vuelve a poner la línea en "1" durante uno o dos clocks. Esto se denomina bits de parada (stop bits) y asegura que, independientemente de los datos a transmitir, siempre exista un flanco en línea formado por el bit de parada y el próximo bit de comienzo que permita la resincronización del receptor.

21) Normas de comunicación serie: ¿para qué sirven y cuáles son sus características más importantes?

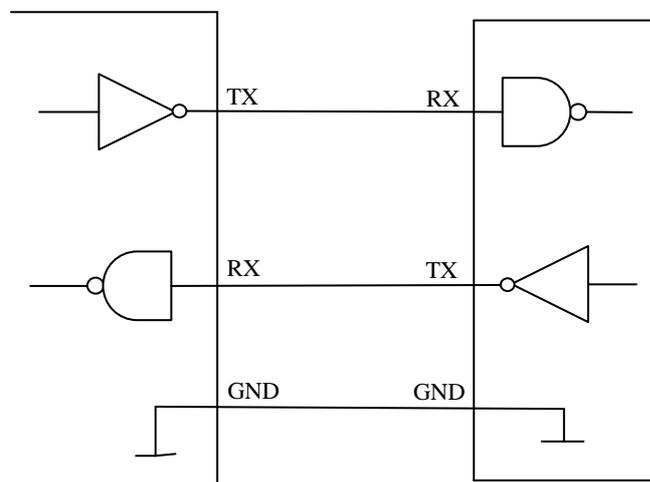
Las normas de comunicación sirven para normalizar o estandarizar la conexión física entre transmisor y receptor, o sea las señales que se manejan, los tipos de conectores y de cableado necesarios y todo lo que respecta a lograr que se pueda establecer una buena comunicación, con la menor tasa de errores posible.

Norma RS232:

El cableado es de tipo desbalanceado. Las tensiones están referidas al terminal común de ambas estaciones y están comprendidas dentro de los siguientes valores:

"1": $-15V < V < -3V$

"0": $15V > V > 3V$



El alcance posible dependerá del nivel de ruido ambiente y de las características del cable usado, aunque la norma establece que no debe usarse para distancias mayores de 15mts quedando su uso limitado a comunicaciones a pie del equipo.

Existen otras señales especificadas en la norma que no hemos mencionado y que cumplen funciones de diálogo entre transmisor y receptor (handshaking). Normalmente estas funciones no son usadas en PLC ya que se implementan por soft.

Norma RS 422/RS 485:

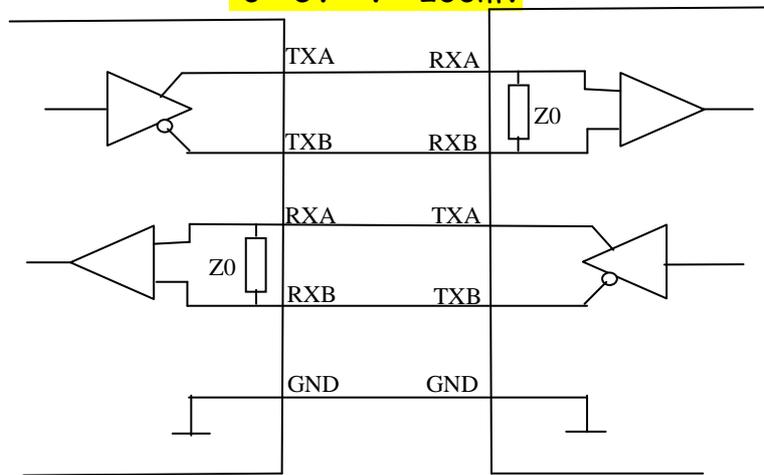
La característica principal de esta norma es que especifica líneas balanceadas. En transmisión diferencial se usan dos cables para transmisión, dos para recepción y un terminal común.

Las tensiones no están referidas a masa sino como diferencia de tensión entre terminales. Dado que ambos terminales están levantados de masa, la señal de ruido aparece como señal de modo común las que son rechazadas por el amplificador de entrada.

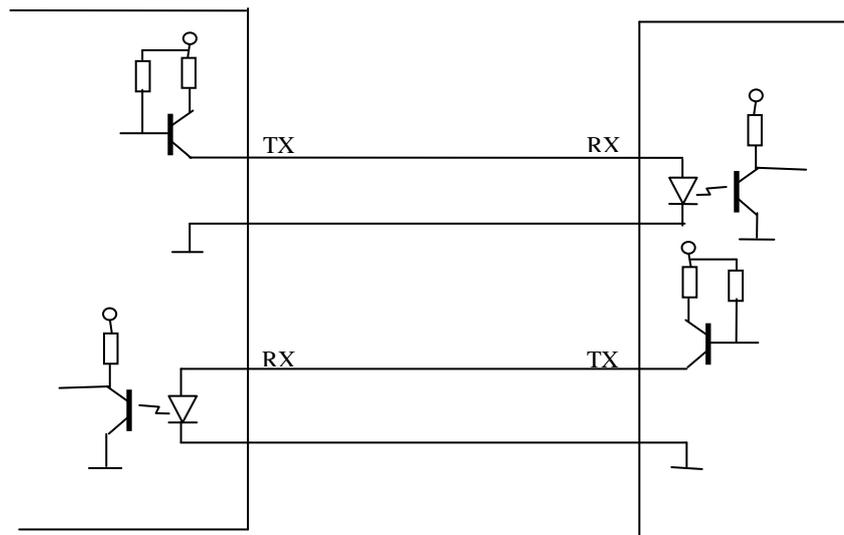
Esto, unido a la baja impedancia de la línea brinda una alta inmunidad al ruido y permite un alcance mucho mayor que en el caso anterior, llegando a los 1,5Km.

La impedancia de entrada Z_0 es del orden de los 100 ohms.
 Las tensiones diferenciales son:

"1": $-5V < V < -200mV$
 "0": $5V > V > 200mV$



Norma 0/20mA - 4/20mA:



La línea se maneja con fuentes de corriente, la impedancia del receptor es baja ($Z = 250\Omega$), tenemos alta inmunidad al ruido con alcances similares a la RS 422/RS 485 (1,5 Km) en velocidades bajas.

22) ¿Qué aseguran los lenguajes o protocolos de comunicación?

Aseguran que los mensajes sean interpretados "correctamente" por ambas partes (PC - PLC o PLC maestro - PLCs esclavos), haciendo que el mensaje quede estructurado de una manera comprensible por ambos. En este caso, por ejemplo, para el PLC SAIA PCA14 modo P, sería (código ASCII):

$STX + W/D + Z + XXX + YY + ETX + BCC$

STX => Comienzo del texto.

W/D => (W) Escribir en memoria del PLC; (D) si se quiere leer la información contenida en la memoria del PLC.

Z => En este punto iría la letra que identifica al elemento a leer o escribir.

E => Elementos discretos (entradas, salidas, etc.).

T => Registro numérico de temporizadores.

C => Registro numérico de contadores.

XXX => Números de la dirección del elemento a leer o escribir.

YY => Valor numérico que se pretende escribir.

ETX => Fin del texto.

BCC => Block Check Code.

Cada uno de los caracteres que componen el mensaje se enviará por la línea de comunicación de la forma previamente mencionada o sea con un bit de start, 7 u 8 bits correspondientes al caracter a enviar, un bit de paridad y un bit de Stop.

Supongamos querer activar la salida 60 desde una PC vía el canal de comunicación. El mensaje que esta última deberá enviar será:

STX + "W" + "E" + "060" + "1" + ETX

Si se pretende conocer el estado del contador 280

STX + "D" + "C" + "280" + ETX

En este último caso, inmediatamente se recibirá por el canal de comunicación el código ASCII del contenido numérico de dicho contador.

23) Concepto de sistema de control distribuido.

Los sistemas basados en PLC permiten un crecimiento por pasos en el nivel de integración al funcionamiento global de la planta (son **escalables**).

Cada isla de producción tendrá su controlador con capacidad de **funcionamiento autónomo** dedicado al control y manejo del subsistema.

Este controlador podrá **comunicarse con otro u otros en el mismo nivel** para la coordinación de áreas entre ellos.

Estos últimos **se conectarán en red** con un centro de comando, con funciones de programación de alto nivel el que ejercerá tareas de supervisión, monitoreo y coordinación de trabajos y podrá realizar diagnósticos del sistema (ej.: sistemas **SCADA**).

Los fabricantes de PLC proveen software para el desarrollo de pantallas mímicas interactivas. En este caso, sobre el gráfico del proceso definido por el usuario, aparecerá información proveniente de los controladores de la red como puntos de medición, condiciones de alarma, mensajes al operador, etc.

El centro de comando podrá a su vez conectarse a través de una red Standard con el **Host de la planta** donde se planifica la política de producción y se recaban del proceso **datos económicos y estadísticos**.

24) Mencionar algunos aspectos a considerar en la selección de un PLC para un proyecto.

Antes de encarar la selección de un PLC debemos definir nuestras necesidades. Para esto será necesario elaborar un **preproyecto** que incluya un programa. Esto puede parecer un contrasentido ya que en la etapa de proyecto no tenemos definido el PLC a utilizar y por lo tanto se empleará el lenguaje **ladder** que es el más extendido, estando disponible en la mayoría de los PLC y es fácilmente trasladable a otros lenguajes como el booleano.

De este preproyecto, podremos definir los siguientes requerimientos:

1. Cantidad de **Entradas/Salidas discretas** discriminadas por los valores de tensión que deban manejar.
2. Cantidad de bits auxiliares (bobinas auxiliares)
3. Cantidad de **temporizadores**. Precisión requerida. Rangos de tiempo.
4. Cantidad de **contadores**. Requerimientos de cuenta ascendente y descendente. Alcances
5. Necesidad de **tratamiento matemático**
6. **Entradas/Salidas analógicas**. Cantidad. Rangos de tensión o corriente. Precisión. Tiempo de conversión.
7. **Entradas/Salidas BCD**
8. **Contadores rápidos**
9. **Módulos especiales** (para Termocuplas, Pt-100, Posicionadores, Motores paso a paso, etc.)
10. Cantidad de palabras auxiliares
11. **Tiempo máximo de barrido** (Scan time).
12. Necesidad de **subrutinas de respuestas**
13. Necesidad de **lazos de control**. Cantidad de lazos. Tiempo de muestreo máximo.
14. **Canales de comunicación**. Velocidad de transferencia. Distancia. Nivel de ruido. Tráfico esperado.
15. **Lenguaje** a utilizar. En este caso, deberá tenerse en cuenta que el mismo deberá ser accesible al personal de mantenimiento más que al de proyecto. Hay que considerar que el programa final cumplirá el mismo papel que el plano en una instalación convencional.
16. Cantidad aproximada de **memoria de programa** necesaria. Este es un punto de muy difícil resolución ya que la memoria consumida por cada tipo de instrucción depende del PLC específico. De todos modos, si en la especificación se in-

cluyen todos los requerimientos mencionados anteriormente, el proveedor puede evaluar la necesidad de memoria en la cotización.

17. **Unidad de programación:** Considerando que para un mismo modelo de PLC hay generalmente varias unidades de programación compatibles, deberemos considerar lo siguiente:

La programación propiamente dicha puede ser realizada por el proveedor del equipo en cuyo caso no sería estrictamente necesario la adquisición de la unidad de programación. De todos modos, la misma tiene una enorme aplicación durante el mantenimiento poniendo a disposición del usuario una herramienta que se traducirá en menores tiempos de parada para reparación. Mientras mayor sea la potencia de la unidad, mayor será la reducción de los tiempos de reparación por lo que la decisión de compra y en caso afirmativo el modelo a adquirir deberá ser contrastado contra los costos del tiempo improductivo.