

Diseño de un sistema de control CAN

Lidia Granados Lopez, Fèlix Coll Carrera

Tercer curso de Ingeniería Técnica Industrial, especialidad en Electrónica Industrial

lgranados86@gmail.com

felix.coll85@gmail.com

Resumen— Este artículo describe la implementación de un sistema de control sencillo, que controla la temperatura y el nivel de un depósito.

I. INTRODUCCIÓN

El sistema que se implementará se puede observar en la figura 1. Este consta de 3 nodos (A, B y C) cada uno de los cuales tiene una función diferente, 2 válvulas, que se abren y se cierran cuando es necesario y finalmente 2 sensores que marcan el nivel de la sustancia que contiene el depósito y su temperatura.

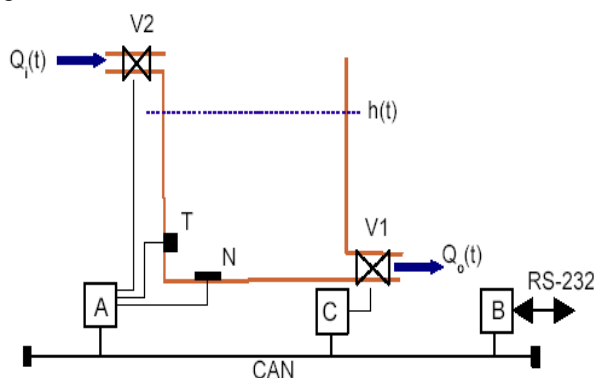


Fig. 1 Esquema del sistema de control de nivel y Temperatura de un depósito.

II. ESPECIFICACIÓN DEL SISTEMA

A. Función de los nodos

1) Nodo A

El nodo A está conectado al sensor de temperatura (T), al sensor de nivel (N) y a la válvula (V2). Este nodo tiene la función de tomar los datos de los dos sensores y enviarlos mediante el bus CAN. Además, también tiene la capacidad de abrir y cerrar V2.

Finalmente hay que destacar que el nodo tiene dos modos de actuar, en Sondeo y en Productor. En Sondeo, el nodo envía los datos de los sensores solo si ha recibido una solicitud de transmisión de estos datos. En modo Productor, en cambio, envía los datos de manera periódica, sin ningún tipo de solicitud.

2) Nodo B

El nodo B es el controlador de todo el sistema, principalmente porque conoce el valor del nivel deseado (consigna), y el rango de valores de la temperatura. Así pues, cuando recibe el valor del nivel, lo compara con la

consigna y envía las órdenes a V2 para que se abra o se cierre.

Cuando recibe el valor de la temperatura, si está fuera del rango entre 20°C y 90°C, activa la alarma (cerrando V1 y V2). También cabe destacar que enseña por pantalla el estado actual de la consigna, el nivel, la temperatura y las válvulas V1 y V2.

3) *Nodo C*: Este nodo únicamente abre o cierra (alarma) V1, dependiendo de las órdenes que reciba del nodo B por el bus can.

B. Función de los sensores

1) *Sensor de temperatura*: Este sensor registra la temperatura del líquido. Tiene un rango de -55 °C a 200 °C y su precisión es de 1 °C.

2) *Sensor de nivel*: El sensor registra el nivel del líquido del tanque, con una precisión de 0.1 mm. Una vez que el nodo B lee estos valores.

C. Función de los actuadores

1) *Válvula V1*: La válvula tiene dos estados: abierta/cerrada. El caudal de ésta cuando está abierta es de $Q_0 = 50 \text{ litros/s} = 0.005 \text{ m}^3/\text{s}$.

2) *Válvula V2*: El caudal de esta válvula cuando está abierta es de $Q_i = 100 \text{ litros/s} = 0.1 \text{ m}^3/\text{s}$.

D. Modos y estados

Los nodos tienen dos modos de operación, que pueden ser cambiados por el usuario en el nodo B mediante un pulsador:

1) *Sondeo*: El nodo B es el que se encarga de hacer la solicitud de datos al nodo A, mientras este último espera.

2) *Productor*: El nodo A envía periódicamente los datos de los sensores al nodo B. El periodo de muestreo es configurado por el nodo B.

E. Estados

El nodo B puede encontrarse en dos estados diferentes dependiendo de la temperatura del depósito:

1) *Alarma*: Permanece en este estado mientras la temperatura del depósito no está dentro de los rangos adecuados.

2) *Control*: Es el estado normal de operación, donde se ajusta el nivel al de consigna con la apertura y cierre de V2.

III. CORRESPONDENCIA CON LA PLACA

Como hemos dicho anteriormente, este sistema de control será implementado sobre la placa CCS CAN Bus compuesta por 4 nodos CAN(A, B, C y D).

Principalmente nos centraremos más en la comunicación entre nodos, por eso dejaremos de lado la utilización de sensores y los sustituiremos por componentes y funciones de la placa. Así pues, los sensores y actuadores se emularán mediante:

- Sensor de temperatura: Potenciómetro del nodo A.
- Sensor de nivel: Cálculo periódico mediante programación en función de la apertura o cierre de las válvulas.
- V1: Abierto → Led verde en el nodo C.
Cerrado → Led rojo en el nodo C.
- V2: Abierto → Led verde en el nodo A.
Cerrado → Led rojo en el nodo A.
- Estado de Alarma: Led rojo nodo B.
- Estado de control: Led verde nodo B.
- Selección de consigna: Potenciómetro del nodo B.
- Selección de modo: Botón del nodo B.

Los únicos nodos de la placa que podemos programar directamente son el A y el B. Por lo que cuando tenemos que acceder a los LEDs del nodo C lo haremos enviándole las tramas específicas para ello desde el nodo B.

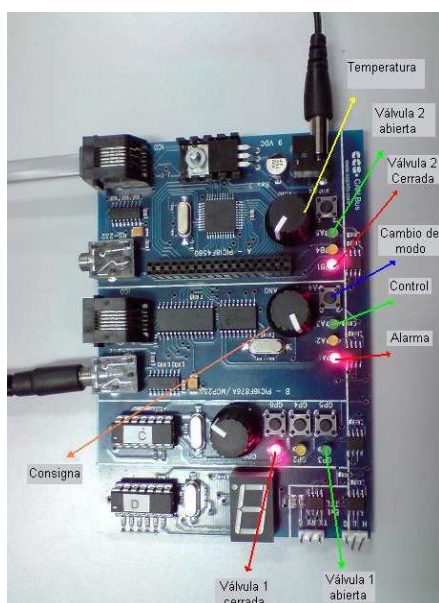


Fig. 2 Esquema de la placa CCS CAN

En la figura 2 se puede observar detalladamente la placa con la que se ha implementado el sistema de control, así como sus características principales.

IV. DEFINICIÓN DEL PROTOCOLO DE COMUNICACIONES

Para poder realizar su función, los nodos del sistema ejecutan el protocolo de comunicaciones que se describe a continuación.

En el diseño de protocolos se ha hecho una serie de suposiciones al iniciar la comunicación. Entre ellas tenemos que las válvulas comienzan estando abiertas, que el modo de operación es el de sondeo y que se está en estado de control.

En la tabla 1 se puede ver el vocabulario, la codificación y el intercambio de mensajes entre los nodos.

Nombre del mensaje	Descripción	ID (hex)	Datos enviados	Intercambio
Productor	Cambio a productor	20	(1byte) T/100*	Nodo B → Nodo A
Sondeo	Cambio a sondeo	30		Nodo B → Nodo A
Confir.	Confirmación de cambio	40		Nodo A → Nodo B
Peti	Petición de datos	50		Nodo B → Nodo A
Valores	Datos de los sensores	60	Temperatura y nivel	Nodo A → Nodo B
Abrir V2	Orden de abrir V2	100		Nodo B → Nodo A
Cerrar V2	Orden de cerrar V2	110		Nodo B → Nodo A
Alarma	CerrarV2 y v1_abierta=0	120		Nodo B → Nodo A
Control	V1_abierta=1	130		Nodo B → Nodo A
Abrir V1	Abrir V1	300	Ledverde=ON Ledrojo=OFF	Nodo B → Nodo C
Cerrar V1	Cerrar V1	300	Ledverde=OFF Ledrojo=ON	Nodo B → Nodo C

Tab.1 Mensajes e Identificadores

*En el modo Productor, el nodo B envía el nuevo periodo de muestreo (T) al nodo A como dato de 1byte en unidades de 0.1s.

A. Nodo B

El nodo B está formado solo por una espera, que siempre está pendiente de los siguientes eventos:

1) *Time out*: Tiempo que espera para recibir la confirmación “confir” del cambio de modo. Al pasar este tiempo vuelve a enviar la petición de cambio de modo al nodo A.

2) *Botón*: Espera que se pulse el botón para cambiar del modo “Sondeo” al modo “Productor” y viceversa.

3) *Confir*: recibe el mensaje de confirmación, señal de que el nodo A a cambiado de modo. Este es el único mensaje que consideramos que se puede “perder”, y no llegar al nodo A, por eso se ha añadido el time out.

4) *Tsondeo*: Es el tiempo del modo “Sondeo”, que cuando finaliza, el nodo B vuelve a enviar una petición de datos.

5) *Valores*: Espera los valores (temperatura y nivel) tomados por el nodo A, los compara en las condiciones pertinentes, y enciende estado de alarma si es necesario.

6) *Otros*: Recibe otros mensajes del sistema que no tenemos en cuenta.

B. Nodo A

El nodo A está formado también solo por una espera que esta pendiente de los siguientes eventos:

1) *Productor*: Espera el mensaje “Productor” para el cambio a este modo. Envía confirmación, y comprueba si estaba anteriormente en modo “Sondeo”.

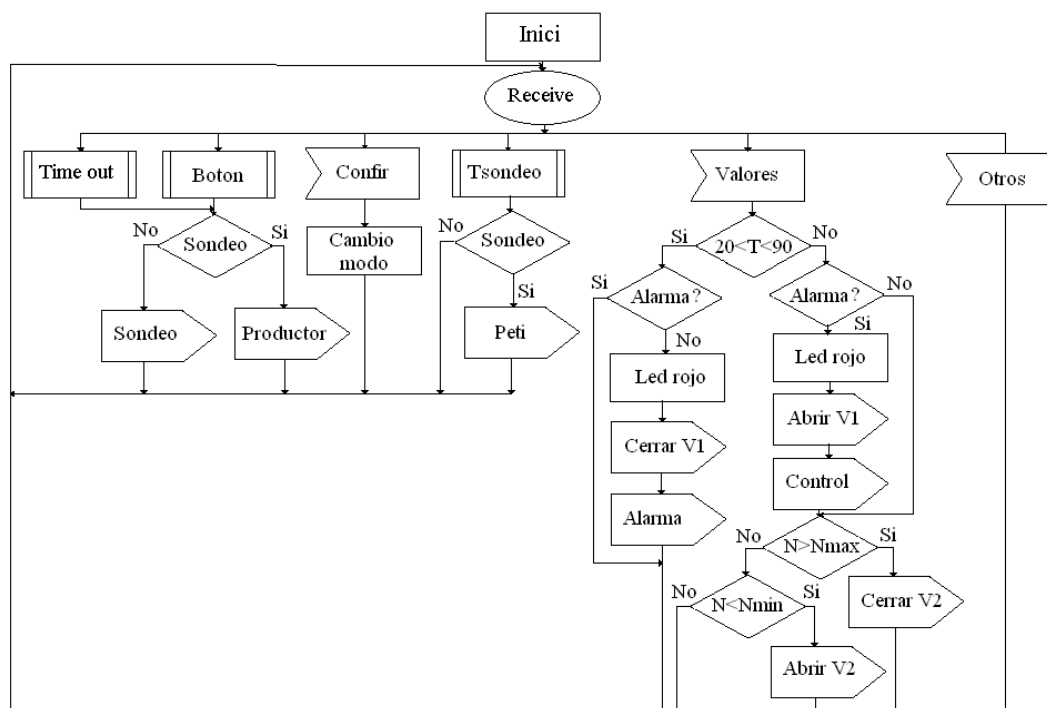


Fig.3 Diagrama SDL del nodo B

2) *Sondeo*: Espera el mensaje “Sondeo” para el cambio a este modo. Envía confirmación, y comprueba si estaba anteriormente en modo “Productor”.

3) *Peti*: Espera mensaje de petición de datos, y posteriormente los envía.

4) *Tproduct*: Periodo de muestreo para enviar los valores si esta en modo “Productor”.

5) *Alarma*: Espera el mensaje de activación de estado de alarma.

6) *Control*: Espera el mensaje para cambio al estado de control.

7) *Abrir V2*: Espera el mensaje de “Abrir V2”.

8) *Cerrar V2*: Espera el mensaje de “Cerrar V2”.

9) *Otros*: Espera otros mensajes que no le interesan.

C. Nodo C

Este nodo recibe los mensajes Abrir V1 y Cerrar V1 que le envía el nodo B y enciende su LED verde y su LED rojo respectivamente.

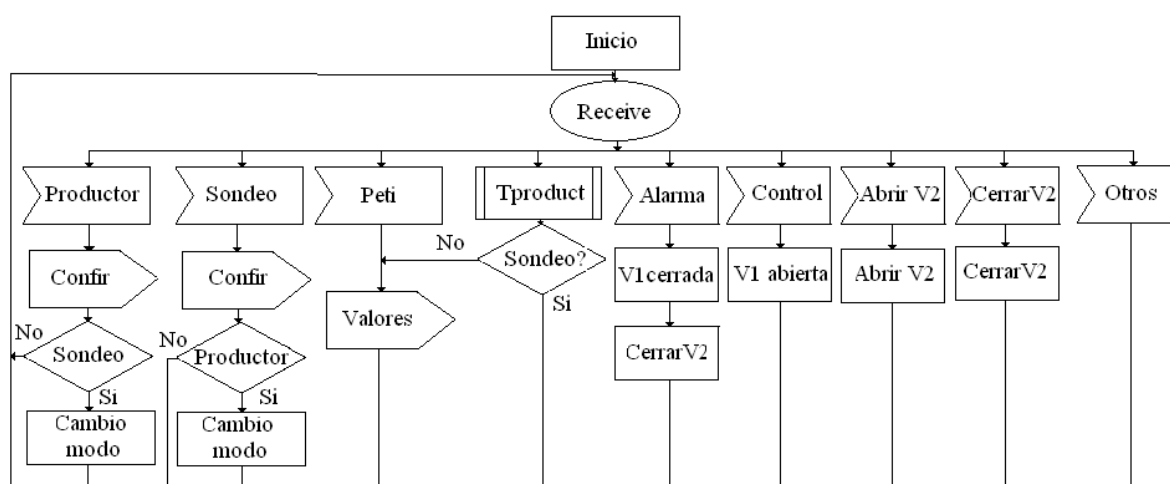


Fig.4 Diagrama SDL del nodo A

V. IMPLEMENTACIÓN

A. Estructura del programa

La programación de los dos nodos se ha intentado estructurar de forma clara y ordenada, utilizando funciones para aquellas partes que se repiten dentro del programa o que tienen una misión más específica y relativamente independiente del programa principal.

Se han utilizado las siguientes funciones:

1) *Nodo A*

- **configuración()**: Configuración de la interrupción del timer y de la conversión del potenciómetro.
- **enviar_trama(id,buffer,tam)**: Espera a que esté libre el bus y envía la trama con los parámetros indicados.

2) *Nodo B*

- **configuración()**: Tiene la misma función que en el A.
- **enviar_trama(id,buffer,tam)**: Tiene la misma función que en el A.
- **write_leds_c(on)**: Envía al nodo C la trama adecuada para encender y apagar los LEDs dependiendo de si el que queremos encender(on) es verde o rojo.
- **comprobar_temp(temperatura)**: Según la temperatura actualiza el estado(variable global) a Control o Alarma.
- **comp_nivel(niv_sensor)**: Según el valor recibido del sensor de nivel, envía mensajes de abrir o cerrar V2.
- **cal_consigna()**: Lee el potenciómetro de consigna y devuelve el valor del nivel elegido en milímetros.

El programa principal de ambos nodos está estructurado siguiendo el SDL, es decir, esta compuesto por un bucle infinito (while(TRUE)), dentro del cual se puede entrar en las diferentes ramas si se cumple la condición (if()) para cada una de ellas.

B. Pruebas de funcionamiento

Para comprobar el correcto funcionamiento del sistema utilizamos el puerto RS-232 del nodo B para mostrar por pantalla los datos que nos interesan y, el envío y recepción de tramas en éste nodo. En la Figura 3 se puede ver una muestra de la visualización por pantalla, con ejemplos del cambio de modo y el envío y recepción de tramas.

Por otra parte, en las Figuras 4 y 5 se pueden ver los resultados de las pruebas en la placa sobre el cambio de estado y cierre/apertura de válvulas.

```

CAN_PUTD(): BUFF=0 ID=00000110 LEN=0 PRI=3 EXT=1 RTR=0
VALVULA V2 CERRADA
CAN_PUTD(): BUFF=0 ID=00000020 LEN=1 PRI=3 EXT=1 RTR=0
DATA = 05
CAN_GETD(): BUFF=0 ID=00000040 LEN=0 OVF=0 FILT=0 RTR=0 EXT=1 INV=0
CAMBIO A PRODUCTOR
CAN_GETD(): BUFF=0 ID=00000060 LEN=3 OVF=0 FILT=0 RTR=0 EXT=1 INV=0
DATA = 73 B6 0F
Temperatura=60 Grados
Nivel=402 mm
Consigna=400
CAN_GETD(): BUFF=0 ID=00000060 LEN=3 OVF=0 FILT=0 RTR=0 EXT=1 INV=0
DATA = 73 73 00
Temperatura=60 Grados
Nivel=344 mm
Consigna=400

```

Fig.3 Cambio a productor

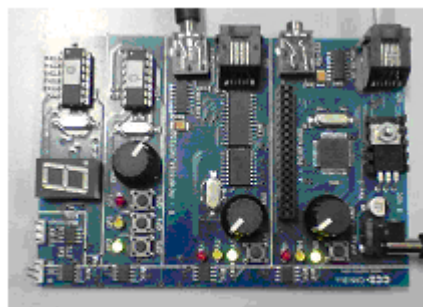


Fig.4 Placa en Control y V2 abierta

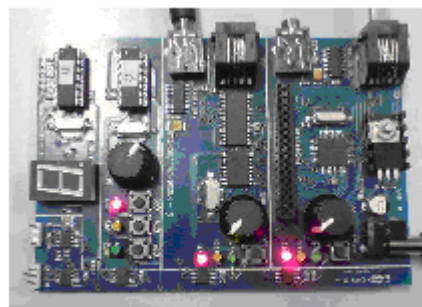


Fig.5 Placa en Alarma

VI. CONCLUSIÓN

Como se puede observar, el sistema se ha podido implementar con éxito. Con él hemos aprendido a comunicar nodos, con una cierta distancia entre sí a través del bus de campo CAN.

Los buses de campo son muy utilizados en la automatización de los procesos de producción. Simplifican bastante la instalación y operación de máquinas y equipamientos industriales porque convierte los dispositivos en inteligentes a un bajo coste.

El desarrollo de este sistema sencillo se podría aplicar al diseño de un proceso industrial más amplio.

Asignatura: Redes de Comunicación Industriales.

Impartida por Guillermo Rodríguez-Navas González.



Félix Coll Carrera. Estudiante de tercer curso de Ingeniería Técnica Industrial (especialidad en electrónica industrial) de la UIB.



Lidia Granados López. Estudiante de tercer curso de Ingeniería Técnica Industrial (especialidad en electrónica industrial) de la UIB.