

Índice

Sistema de adquisición de datos RS485AD.

1. Descripción	3
2. Señales analógicas.	3
3. Señales digitales.	3
4. Comunicaciones.	3
5. Comandos.	4
6. Funciones en DLL.	4
7. Alimentación de la placa.	5
8. Distribución de elementos	5
9. Parámetros de fábrica	5

RS485AD

1. Descripción

El sistema de adquisición de datos RS485AD consta de 8 entradas A/D y 2 salidas D/A, configurables en rango a partir de un par de resistencias, así como 4 entradas digitales y 16 salidas digitales. Este módulo ha sido diseñado pensando en la compatibilidad con otros productos de la gama permitiendo conectar las salidas digitales a las placas Interfase de potencia para el control de relés, transistores, triacs, etc. Además tal y como indica su nombre permite la comunicación a través de RS232 (el puerto COM de un PC) y RS485. Esta última da la posibilidad de conectar múltiples placas (hasta un máximo de 8) en forma de bus RS485 serie, optimizando así la utilización de los puertos COM del ordenador.

2. Señales analógicas.

El módulo cuenta con ocho entradas analógicas en modo común (AD0 .. AD7), así como dos salidas analógicas (DA0 y DA1). El convertidor A/D y D/A es de 12 bits por lo que se puede obtener una resolución de hasta 4096 valores distintos de tensión. El rango de medida de las entradas es configurable a partir del divisor de tensión formado por las resistencias RA y RB que se encuentran junto a cada entrada analógica. Además dispone de un condensador para la eliminación de ruidos que puedan aparecer en la señal. El divisor ha de calcularse de forma que la tensión máxima a la entrada del conversor A/D quede en 2.45V, valor de referencia que dará un máximo de resolución. El esquema queda reflejado en la figura 1.

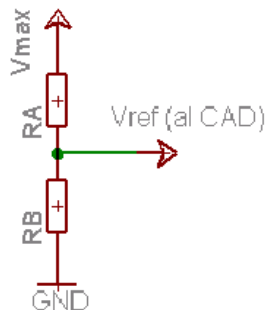


Figura 1

$$V_{ref} = \frac{V_{max}}{RA + RB} * RB \rightarrow RA = \frac{RB * (V_{max} - V_{ref})}{V_{ref}}$$

Si se toma un valor para RB y se conoce el valor máximo de la señal de entrada se puede deducir RA de la ecuación, pues $V_{ref}=2.45V$.

Las salidas analógicas disponen al igual que las entradas de la posibilidad de variar el factor de escala, debido al amplificador operacional. La relación de las resistencias R1 y R2 viene dada por la ecuación.

$$V_{max} = \left(\frac{R2}{R1} + 1 \right) * V_{ref} \rightarrow R2 = \left(\frac{V_{max}}{V_{ref}} - 1 \right) * R1$$

Donde V_{max} es el valor de tensión máximo que se quiere sacar por la salida analógica.

3. Señales digitales.

Dispone de cuatro líneas I/O digitales que permiten la lectura y la escritura en estas (E/S 0 .. E/S 3), así como 16 líneas (E/S) agrupadas en un conector de cable plano de 20 pines situado en la parte superior derecha de la placa. Este conector está pensado para conectarse con las placas de Interfase de potencia, o las Interfase de otros módulos similares.

4. Comunicaciones.

Las comunicaciones se realizan a través del conector Sub DB9 hembra que permite la conexión directa al puerto COM del PC a través de un cable de puerto serie estándar pin a pin. También dispone de la posibilidad de comunicación RS485 para lo que se han habilitado los pines 1 y 8 del conector Sub DB9 como D+ y D- respectivamente. El blindaje se toma en la carcasa del conector.

Con el fin de diferenciar las placas en el bus RS485 se ha dotado a cada una con un selector de dirección configurable (JP1) a través de jumpers para indicar de que dispositivo se trata a la hora de hacer llamadas a estos. JP1 se puede considerar como un número binario de tres bits en el que el par de patillas serigrafiadas con 3 es el bit

de mayor peso. Por tanto, para configurar la dirección hay que poner un jumper en las patillas ó bit que se quiera poner a 0. Por poner un ejemplo, si se quiere configurar el dispositivo con dirección 0 (en binario 000), hay que poner tres jumpers, una en cada par de patillas. Si se quiere la dirección 5 (en binario 101), se pondría sólo un jumper en el par de patillas serigrafiadas con un 2.

5. Comandos.

Mediante el envío de sencillos comandos a través del puerto serie se puede obtener el estado de las entradas así como actuar sobre las salidas.

Todos los comandos siguen el mismo esquema:

Un carácter de inicio de comando ‘#’.

Dirección de la placa (configurable con los jumpers) ‘d’.

El comando a ejecutar ‘C’.

El canal analógico o la línea digital sobre la que se actúa ‘nn’.

El valor si se trata de una salida analógica o digital ‘vv’.

Carácter de fin de comando ‘\$’:

“#dCnnvv\$”

Los posibles valores de “C” son:

Entradas analógicas: Comando “A”. Devuelve el valor de 0 a 4096 del canal analógico seleccionado. Ejemplo: #1A5\$ devuelve el valor del canal 5 de la placa con dirección 1.

Entradas digitales: Comando “E”. Devuelve el valor 0 o 1 de la entrada digital seleccionada. Ejemplo: #2E1\$ devuelve el estado de la línea 1 de la placa con dirección 2. Con el comando “U” es posible acceder a las 16 líneas digitales. Ejemplo #1U05\$ devuelve el estado de la línea 5 del módulo 1. También existe la posibilidad de obtener el estado de todas las líneas en dos grupos: el comando “F” devuelve el estado de las 4 entradas digitales mientras que el comando “V” devuelve el estado de las 16 líneas

digitales (00003 indica que las líneas 0 y 1 están a 1 y el resto a 0).

Salidas analógicas: Comando “D”. Escribe el valor indicado en el canal analógico seleccionado. Ejemplo #3D10100\$, escribe 0100 en el canal DAC1 de la placa con dirección 3.

Salidas digitales: Comando “C”. Escribe sobre las líneas de entrada salida. Ejemplo: #5C031\$ pone a 1 la salida 03 de la placa con dirección 5. Comando “S”. Escribe sobre las líneas de sólo salida. Ejemplo: #6S140\$ pone a 0 la salida 14 de la placa con dirección 6.

6. Funciones en DLL.

Para facilitar la integración en otras aplicaciones de este producto, se ha creado una librería con funciones que directamente se encargan de pedir datos o enviarlos a la placa, sin que la aplicación principal tenga que preocuparse de esta labor. Las funciones permiten introducir el factor de escala para que así el valor que se devuelve sea el valor de la magnitud deseada y no el valor digital obtenido de la conversión.

A continuación se describe cada una de ellas:

InitPort: Inicializa el puerto serie en el que se encuentran conectados los módulos de adquisición de datos, indicando el puerto (p) y la velocidad de transmisión (v). En caso de no poder abrir el puerto por estar ocupado la función devuelve -1.

function InitPort (p,v:integer): integer

ClosePort: Cierra el puerto serie abierto mediante la función anterior. Se le pasa como parámetro el número de puerto. Devuelve 0 si se ha abierto correctamente el puerto o -1 en caso contrario.

function ClosePort (p:integer): integer

Digin: Devuelve en state el valor del estado del canal digital indicado en channel del número de dispositivo idnum.

function Digin (var idnum: integer; channel: integer; var state:integer): integer

DiginExt: Permite obtener el valor de una

línea del grupo de 16 líneas digitales.

*function DigInExt(var idnum:integer;
channel:integer; var state:integer)*

Dig4In: Devuelve el valor de las cuatro entradas digitales del dispositivo número idnum, en un número de cuatro dígitos (state) de forma que cada dígito indica con 0 ó 1 el estado de cada entrada identificada por la posición del dígito dentro del número.

*function Dig4In (var idnum: integer; var
state: integer):integer*

Dig16In: Al igual que la anterior devuelve el estado de las 16 líneas digitales en un número de 5 dígitos

*function Dig16In(var idnum: integer; var
state: integer):integer*

ADIn: devuelve el valor (voltage) del canal seleccionado(channel) del dispositivo idnum. Si el factor de escala que se indica es 0 devuelve el valor digital del convertidor.

*function ADIn (var idnum:i nteger;
channel: integer; var voltage: single): integer*

AD8In: Devuelve en un array de 8 posiciones el valor de los 8 canales de entrada analógica del dispositivo idnum.

*function AD8In (var idnum: integer; var
voltage: array of integer): integer*

DAOut: escribe el valor indicado sobre el canal DAC seleccionado.

*function DAOut (var idnum: integer;
channel: integer; analogOut: single): integer*

7. Alimentación de la placa.

Para que el RS485AD funcione correctamente, se debe conectar a una fuente de alimentación comprendida entre +9V y +15V debidamente rectificadas y filtradas, al conector X2, donde la serigrafía X2 indica el positivo.

Hay que tener en cuenta que la máxima salida analógica va a ser igual a la tensión de alimentación.

8. Distribución de elementos.

La figura 2 muestra un esquema de los elementos constituyentes del RS485AD.

Para tener acceso a las resistencias que configuran la ganancia del amplificador para las salidas analógicas, se debe quitar la plaquita que contiene el microcontrolador. Para ello sujetar firmemente la placa del RS485AD, y tirar en dirección perpendicular a ella de la placa que contiene el microcontrolador.

ADVERTENCIA: Siempre que manipule la placa electrónica para cambiar configuraciones de resistencias o sustraer el microcontrolador, desconecte la misma de la fuente de alimentación.

9. Parámetros de fábrica.

Por defecto la placa RS485AD viene configurada con una velocidad de 38400 bps, una ganancia de 4 en los canales DA ($R3=30K$ y $R1=10K$), y la dirección de placa 0.

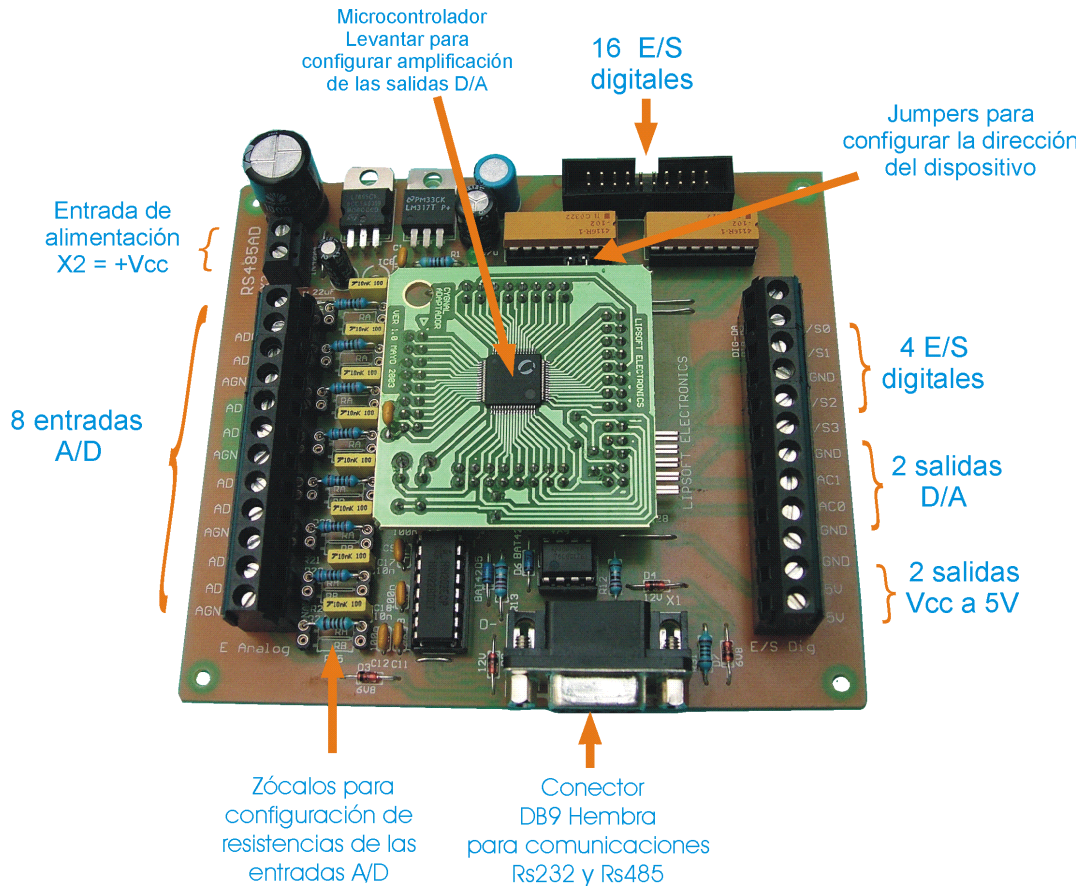


Fig 2.