

# MANUAL uPLC V4

**Versión manual 1.00**

© 2002 Ibercomp SA – Rev 1.00 Julio 2002

## 1) INTRODUCCION

En primer lugar, deseamos expresar nuestro agradecimiento por haber adquirido nuestro producto .PLC V4, confiamos que el equipo cumpla con sus expectativas.

Antes de comenzar a describir esta placa vamos a intentar explicar el por qué de este diseño. Nuestra andadura en la fabricación de cartas electrónicas estándar comenzó seis años después de la fundación de nuestra empresa, un verano de 1994 y fue con la primera versión de nuestro ALTAIR 535.

Este microcontrolador, de propósito general, estaba destinado principalmente al desarrollo de prototipos y al aprendizaje. Desde un primer momento los usuarios nos demandaban un equipo que fuese más sencillo de utilizar, que no requiriese al mismo tiempo conocimientos avanzados de electrónica y programación.

Por ello desarrollamos nuestro equipo uPLC, el cual estaba dotado de dos microcontroladores ZILOG funcionando en paralelo. Este nuevo equipo no exigía al usuario desarrollarse su propia electrónica, puesto que disponía de entradas y salidas acondicionadas. Para facilitar su labor se desarrolló un lenguaje de alto nivel tipo BASIC con numerosas instrucciones dedicadas al control.

El siguiente paso fue sacar un equipo dotado de mayor potencia, mayor número de entradas/salidas y una consola basada en un display de 4 dígitos y 4 pulsadores. Este equipo, denominado uPLC II, con unas prestaciones superiores al anterior se le implementó una versión más avanzada del BASIC.

En la mayoría de los casos no se necesitaba un equipo tan potente, sino de una máquina que fuese compacta y a la vez flexible a la hora de programarla. Para resolver este problema retomamos nuestro equipo uPLC y sustituimos sus dos microprocesadores en paralelo por un microcontrolador AT89S53.

La gran ventaja de esta solución frente a las anteriores es que el micro se puede reprogramar sin extraerlo de la placa, lo que nos permitía que el usuario desarrollara su aplicación en su lenguaje de programación favorito. Ya no era necesario utilizar nuestro BASIC interpretado. Este punto es muy importante, porque permite integrar su aplicación por primera vez dentro del microcontrolador, lo da al equipo una indiscutible mayor robustez.

Este equipo fue el primero en salir de nuestra primera línea de producción SMD, lo que marcaría un paso significativo hacia el abaratamiento de nuestros productos, al mismo tiempo que supondría un importante incremento en la calidad.

Para complementar nuestro uPLC versión 2 comenzamos a comercializar el lenguaje BASCOM 8051 y el lenguaje KEIL C. El primero es ideal para aplicaciones sencillas y enseñanza mientras que el segundo es casi una obligación, a pesar de su precio, para desarrollos profesionales.

Unos meses después mejoramos nuestro uPLC, comercializando bajo la denominación uPLC versión 3, en esta añadimos un reloj en tiempo real, memoria E2PROM para almacenamiento de parámetros y la posibilidad de configurar el puerto serie como RS232c o RS485.

Esta versión, sigue comercializándose en la actualidad bajo la denominación uPLC digital, y es la solución ideal a muchos proyectos.

A pesar de disponer de un equipo económico, compacto y bastante robusto, decidimos dar un paso más en nuestro continuo desarrollo. En este paso rediseñamos por completo el equipo destacando las siguientes mejoras:

- . Alimentación 220VAC o 12/24 DC.
- . La capacidad de memoria de programa se ha multiplicado por cinco.
- . La velocidad de ejecución se ha duplicado.
- . Consola con 6 pulsadores y display retroiluminado de 2x16 caracteres.
- . Entrada/salida analógica.
- . Integrado en caja para cuadro eléctrico estándar.

Estamos convencidos que el uPLC IV es un buen producto, con características sobresalientes si lo comparamos con otros equipos de la competencia. No queremos que este sea el final de nuestro camino, por lo cual agradeceríamos que nos hiciera llegar sus sugerencias. Quede seguro que estas serán estudiadas y tenidas en cuenta en el próximo desarrollo.

## 2) ESPECIFICACIONES ELECTRONICAS

- . Alimentación: 220VAC 50Hz 100mA  
12VDC 500 mA (depende de modelo)  
24VDC 300 mA (depende de modelo)
- . Duración Batería: 2 años en ausencia de corriente  
(tiempo de recarga 3 semanas)
- . Microcontrolador: 89c51RD funcionando a 22.1184 MHz (64K + 1K)  
89c668 funcionando a 22.1184 MHz (64K + 8K)
- . Entrada analógica: 0..20 mA ó 0..10VDC  
Impedancia de entrada 500 Ohmios o 15K
- . Salida a transistor: Máx 1.5 Amperios/50VDC
- . Salidas a relé: 5 Amperios a 250VAC
- . Entradas optoacopladas: 6 a 24 VAC/VDC
- . Temperatura funcionamiento: 0°C a 50°C funcionamiento normal  
50°C a 70°C funciona automática con display  
ennegrecido.

### 3) ALIMENTACION

El equipo se puede alimentar directamente con corriente 220VAC, para lo cual dispone de las dos bornas del conector J1. A estas bornas está conectado el primario de un transformador de 5VA, el cual está protegido mediante un fusible estándar de 200 mA.

La salida del transformador es rectificadora mediante el puente de diodos G1, después del cual se obtiene una tensión sin estabilizar de 12V o 24V dependiendo del modelo. Esta tensión está disponible en el conector J2.

Esta tensión directamente alimenta los relés, que deberán ser siempre de la misma tensión que el transformador. Por otra parte la tensión se utiliza para alimentar un reductor de tensión conmutado basado en el integrado L4962.

A la salida de este se obtienen 5.1 V de corriente continua estabilizada que tiene como objetivo alimentar toda la parte electrónica del microcontrolador y la consola.

Si lo desea puede “extraer” energía del conector J2, pero teniendo cuidado de no sobrepasar la capacidad máxima del transformador, que es de unos 400 mA. Debe tener presente que el consumo del equipo es de unos 60mA y el de la retroiluminación es de unos 100 mA.

**RECOMENDACIÓN:** Le aconsejamos que en cualquier aplicación que realice la retroiluminación del display esté el mayor tiempo posible apagada, y que en la salida J2 no conecte cargas de más de 40 mA.

Si no dispone de alimentación 220VAC también puede alimentar el equipo a través de este conector J2, pero siempre respetando la tensión de funcionamiento (la indicada en los relés) y la polaridad. Esta entrada/salida de tensión está protegida mediante un fusible térmico rearmable de 500 mA.

### 4) PROGRAMACION

Como todo autómatas programable, el uPLC V4 puede ser reprogramado. Para realizar sobre él un programa es necesario en primer lugar disponer de un lenguaje de programación, unos conocimientos mínimos y tiempo.

No intención de nuestro manual introductorio explicar como se programa un microcontrolador de la familia 51 ni listar todos los lenguajes que se pueden utilizar. Tan solo en nuestro disco uPLC V4 hallará algunos ejemplos escritos en los lenguajes que nosotros, Ibercomp, comercializamos.

En el mercado existen otros muchos lenguajes, y puede encontrar bastante documentación relacionada con los microcontroladores de la familia 51 en la página [www.8052.com](http://www.8052.com).

Si bien, lo que usted desea es realizar aplicaciones profesionales competitivas en precio/calidad háganos caso, lo que usted necesita es el KEIL C. Somos conscientes que su alto precio es muy disuasorio, pero haga números, compare productos, valore su tiempo y verá como al final es la opción más económica.

Una vez que disponga de su lenguaje de programación, haya adquirido unos conocimientos mínimos “ojeando” la documentación del microcontrolador/manuales de programación y disponga de tiempo necesitará un programa cargador.

PHILIPS, el fabricante del microcontrolador ofrece dos programas de este tipo gratuitos, ambos están disponibles en nuestro disco uPLC V4. El primer programa es el WINISP y el segundo FLASH MAGIC.

Ambos programas, sin llegar ninguno de los dos a la perfección funcionan y son válidos para trabajar. Yo particularmente utilizo el WINISP, aunque la mayoría de mis compañeros prefiere el otro.

**IMPORTANTISIMO:** En ambos programas se permite modificar el VECTOR y el registro de ESTADO del microcontrolador. NUNCA DEBE MODIFICARLOS SIN SABER LO QUE ESTA HACIENDO. El valor del vector debe ser **SIEMPRE** FC (en hexadecimal) mientras que el valor del registro de estado debe ser 0 para un funcionamiento normal.

**LEA EN EL MANUAL LA DESCRIPCION DE AMBOS REGISTROS.** Si modifica el valor del vector, y reinicializa el microcontrolador, tiene todas las papeletas para desoldar el integrado y sustituirlo por otro si desea cambiar el programa que hay en él.

El motivo de esta desgracia es que PHILIPS ha dejado una posibilidad para que el usuario pueda modificarse el programa monitor que hay dentro del microcontrolador. Dicho programa está en la dirección apuntada por vector.

En este sentido es más seguro utilizar el programa FLASH MAGIC, dado que la modificación de estos registros hay que hacerlo mucho más “aposta”.

Tal y como se describe en el manual de PHILIPS relativo al microcontrolador hay dos modos, uno que podríamos rebautizar de ejecución (al arrancar se ejecuta el programa de usuario) y otro que podríamos denominar de programación (el micro arranca un programa monitor que permite reprogramarse a si mismo).

El microcontrolador arrancará en uno u otro modo dependiendo de la posición del conmutador SW2. Si este está apuntando hacia arriba, el equipo arrancará en modo programación mientras que si apunta hacia abajo lo hará en ejecución.

Una vez movido el interruptor, para que este tenga efecto habrá de pulsarse el botón de RESET.

Para que comunique el PC con el autómatas será necesario realizar un cable que los una. En un lado deberá disponer de un conector DB9 que irá conectado al COM del

PC y en el otro unos hilos que irán a las bornas correspondientes o bien un conector hembra paso 1.27 mm.

Para mayor información acerca de la realización de un cable lea el capítulo dedicado al puerto serie.

## 5) MICROCONTROLADORES

El uPLC V4 puede montarse en principio con dos microcontroladores, el 89c51RD2 y el 89c668. Ambos microcontroladores, en principio, son compatibles, lo que quiere decir que en la mayor parte de los casos el software o aplicación desarrollado en uno de ellos funcionará en el otro.

Aun así existen algunas diferencias a tener en cuenta:

- . El 668 dispone de una memoria de 8 Kbytes en lugar de 1Kbyte del RD2.
- . El 668 dispone de un controlador I2C por hardware, con lo que es posible “acelerar” algunas aplicaciones. También sería posible realizar una aplicación en la que el microcontrolador funcionase como un dispositivo I2C esclavo. Esto último es muy difícil, casi imposible, realizarlo mediante software.
- . Los periféricos e interrupciones que no son propios del 8052, como pueden ser el PCA, TIMER2, ..., tienen direcciones distintas. Si realiza una aplicación en uno de los dos micros, y luego decide cambiar al otro, debe estudiar detenidamente estas diferencias.

No hemos encontrado más diferencias entre ambos microcontroladores.

Si para programar un equipo dotado de 668 utiliza el BASCOM puede encontrarse el problema de no poder utilizar alguna de las interrupciones. Estos inconvenientes serán solucionados en versiones futuras.

## 6) PERRO GUARDIAN – INMUNIDAD AL RUIDO

Cuando se realiza una aplicación sobre cualquier microcontrolador hay que tener en cuenta que este puede fallar en cualquier momento por causas externas, como puede ser un ruido eléctrico.

Estos fallos pueden tener efectos impredecibles, ya que de forma aleatoria pueden cambiarse registros o posiciones de memoria del microcontrolador, de modo que se alteran datos o incluso el programa pasa a ejecutar un código extraño.

Evitarlo, es literalmente imposible, aunque si es conveniente tomar algunas precauciones. Nosotros al diseñar la electrónica pusimos especial cuidado en lograr un equipo robusto, pero el software también puede colaborar en el logro de un sistema más estable.

Con este fin el equipo dispone de perros guardianes, que son temporizadores que se encargan de vigilar que el software funciona correctamente. Si algo falla, simplemente resetean el equipo y el programa vuelve a comenzar de nuevo.

El funcionamiento es sencillo, el temporizador va incrementándose, y cuando alcanza determinado valor reinicializa el equipo. El software debe encargarse de poner a cero el temporizador.

Internamente los microcontroladores de PHILIPS disponen de un perro guardián asociado al modulo 4 del PCA. Este tiene el inconveniente de ser demasiado rápido, esto es, el software debe refrescarlo. Funcionan bien pero es complicado crear una aplicación que los maneje correctamente.

Manejar correctamente implica que la aplicación solo llame al refresco del perro guardián en un punto, de modo que el software deberá pasar por dicho punto con mucha frecuencia.

Como facilidad hemos añadido un perro guardián externo, el MAX1232 ó LTC1232. Este integrado resetea el microcontrolador si la patilla /ST no cambia de estado al menos una vez cada segundo y medio.

Mediante el JUMPER JP1 podemos seleccionar que este refresco se haga automáticamente con la oscilación de la señal ALE o bien que se realice mediante software utilizando para ello el puerto P3.7



Debe tener presente que cuando el microcontrolador está en modo programación, JP1 debe estar de forma que el refresco de /ST sea automático. El motivo no es otro que el programa BOOTLOADER creado por PHILIPS desconoce que exista el perro guardián externo.

Un dato más, el MAX1232 además de disponer de perro guardián, monitoriza el estado de la alimentación, de forma que si esta baja de 4.5 voltios, el equipo es reseteado. Con ello se evita que el microcontrolador pueda seguir funcionando mientras alguno de los periféricos deja de funcionar.

Los problemas que hemos detectado en entornos muy ruidosos (ej. Montar el equipo justo al lado de un variador de frecuencia de 10 Kw) son:

PROBLEMA	POSIBLE SOLUCION
Que el microcontrolador comience a realizar operaciones aparentemente aleatorias.	Revise que el software sea correcto  Apoyarse en los perros guardianes
Que desaparezca el programa del microcontrolador de forma permanente.	Regrabe el programa y aislar/alejarse del equipo de fuentes de radiaciones electromagnéticas fuertes.
Que se borre parcialmente el contenido de la memoria EEPROM	Escriba el software de manera que se mantengan dos copias de los datos en la memoria. Ambas copias deberán tener un

	<p>checksum o CRC, de modo que el software al inicializarse pueda conocer que copia está intacta.</p> <p>Este borrado parcial suele deberse a un corte de corriente en mitad de un ciclo de lectura/escritura.</p>
Que se borre parcialmente el contenido de la memoria del reloj en tiempo real o cambien sus registros.	<p>En el caso de la memoria proceda igual que en el caso de EEPROM.</p> <p>Compruebe que la tensión de la batería es superior a 3 Voltios.</p> <p>Escribir el software de forma que reajuste el reloj en el caso de leer valores extraños (ej: hora 165, año 48, ...)</p>
Que el display LCD comience a mostrar caracteres extraños, mientras la aplicación parece que funciona correctamente.	<p>Es posible que microcontrolador interno del display le pase lo mismo que al de nuestro software, esto es, que se vuelva loco.</p> <p>La única solución es reinicializar el display. Nuestras rutinas comprueban el estado del display cada vez que se accede a él, si no responde reinicializa el display.</p> <p>Aun así puede haber casos que sea necesario permitir que el usuario se reinicialice, por ejemplo cuando pulse dos teclas alejadas simultáneamente.</p>

## 7) LEDS

El equipo dispone de dos LEDs que pueden ser controlados por software. Estos LEDs son visibles en el frontal del autómatas, habiendo uno de color verde (CPU) y otro de color rojo (STATUS).

Ambos LEDs están conectados directamente a los puertos P0.6 y P0.7. Son de lógica invertida, esto es, cuando se pone el puerto a 0, el LED se enciende, y cuando se pone a 1 se apaga.

Si observa el esquema del uPLC IV, observará que hay una “replica” de estos LEDs en la placa base. Estos están situados ahí únicamente para facilitar la depuración del programa (por si trabaja con el equipo abierto y con la consola mirando hacia abajo).

## 8) DISPLAY LCD

Cuando desarrollamos este proyecto, nos encontramos con el dilema de cómo conectar el DISPLAY LCD. Lo más sencillo, y habitual, como aparece en casi todos los ejemplos que podemos encontrar en Internet, es conectarlo directamente a un puerto del microcontrolador.

Tomando esa solución nos encontrábamos con el problema de que no había puertos suficientes para todos los elementos que deseábamos incorporar en el equipo. Podíamos haber conectado los relés a un multiplexor PCF8574 o incluso a un económico 4094.

Pero en este proyecto había un par de puntos que nos condicionaban:

- Nuestra intención era que al arrancar el equipo el estado de todos los relés de salida fuese siempre apagado. Los puertos del microcontrolador al reinicializar el equipo se ponen siempre a estado 1.
- Para que el equipo fuese económico debía ser fácil de construir, por lo que nos era indispensable simplificar el conexionado entre la placa base y la placa frontal o consola. Llevar 8 líneas para el manejo del display, más cinco para el teclado, más 2 para leds más las alimentaciones era un problema.

La solución era conectar el display a través de un integrado PCF8574, de forma que se pudiera manejar a través del BUS I2C. De acuerdo que el acceso es mucho más complicado y sobre todo lento, pero nos ahorra cableado al mismo tiempo que se subsana el problema de las entradas.

Como ejemplo de acceso al display, todos los listados que suministramos tanto en KEIL C como en BASCOM 8051 disponen de código que accede de una manera u otra al display LCD.

También es posible controlar mediante software el estado de la retroiluminación, recomendamos que este permanezca siempre que sea posible desconectado. Con ello se reduce el consumo del equipo en más de 100 mA (a 12 Voltios), calentándose menos el transformador interno.

Lo ideal es realizar una aplicación que encienda la retroiluminación cuando el usuario presione una tecla (está delante) y apagarlo transcurrido un tiempo. Puede ver como realizar dicha operación en los ejemplos de acceso al teclado.

## 9) SALIDAS A RELES

El uPLC V4 dispone de 8 salidas a relé de un contacto normalmente abierto. Estos relés de salida están agrupados en un bloque de 4 relés con un común y dos bloques con un común. Esta configuración permite reducir el espacio en la placa, al requerir menos bornas de conexión.

Todos los relés están optoacoplados respecto a la electrónica del autómeta, lo cual dota al equipo de una inmunidad alta frente a ruidos eléctricos que puedan surgir a consecuencia de la apertura/cierre de los mismos.

En la propia placa, con el fin de facilitar la depuración del sistema, el equipo dispone de un LED testigo de estado. Este se enciende cuando el relé es activado, y por tanto se cierran sus contactos.

Los 8 relés son controlados directamente por el puerto P2 del microcontrolador, utilizando lógica invertida. Esto quiere decir que para activar un relé es necesario poner la salida a nivel bajo (0).

De este modo, cuando se reinicializa el microcontrolador, todos los relés pasan a un estado definido de apagado.

## 10) ENTRADAS OPTOACOPLADAS

Nuestro autómeta incorpora 8 entradas digitales optoacopladas, agrupadas en dos bloques de 4 más un común. Todos los optoacopladores utilizados disponen a la entrada de dos diodos en antiparalelo, lo que permite no tener en cuenta polaridad, o incluso alimentarlos con corriente alterna.

Tal y como suministramos el equipo, las entradas pueden activarse con tensiones comprendidas entre 7 y 24 voltios corriente continua. Para tensiones mayores será necesario añadir una resistencia en serie para limitar la corriente y dejarla dentro de los valores permitidos del optoacoplador (ver especificaciones en PDF).

Por ejemplo puede activar la entrada directamente a 220 VAC, si conecta en serie a la misma (NUNCA EN EL COMUN) una resistencia de 56K / 1W. En este caso deberá realizar además un pequeño filtrado por software con el fin de eliminar la activación/desactivación debido a la corriente alterna.

Recomendamos que dicha rutina de filtrado se realice utilizando a una interrupción por temporizador con frecuencia superior a 200 Hz, de forma que si en una de las lecturas la entrada está activa, es que hay corriente en ella.

Internamente en el circuito, cada una de las entradas dispone de un LED testigo de estado que se alimenta cuando hay tensión en la misma.

**NOTA:** No es habitual, pero si es posible que el LED testigo de estado se encienda y en el microcontrolador no se lea que la entrada está activa. Es debido a que la tensión en la entrada es demasiado débil para indicar al microcontrolador que está activa y lo suficientemente fuerte como para iluminar tenuemente el LED. Este comportamiento no representa ningún problema en la mayoría de las aplicaciones.

A diferencia que las salidas, las entradas no están conectadas a un mismo puerto. Un bloque de cuatro entradas está conectado a las líneas P1.0, P1.1, P1.2 y P1.3. Con las 3 primeras se puede realizar un control externo del temporizador 2 del microcontrolador, controlando su frecuencia, dirección y estado.

El puerto P1.3 es la entrada del modulo 0 del PCA.

El segundo bloque está conectado a las líneas P33, P34, P35 y P36. La línea P33 corresponde a la interrupción externa 1 (INT1) y las líneas P34 y P35 a las entradas de los módulos 3 y 4 del PCA.

En este apartado es interesante que estudie a fondo el manejo del PCA, se trata de un periférico simple de manejar y sobre todo muy potente que permite realizar algoritmos realmente curiosos.

Por último, en este apartado, queda añadir que las entradas optoacopladas utilizan lógica negativa, lo cual quiere decir que cuando externamente se las activa dándoles tensión, sus puertos asociados del microcontrolador se ponen a nivel bajo.

Es así para dotarlas de una mayor funcionalidad cuando se utilizan como entradas del PCA, control del timer 2 e interrupción externa 1.

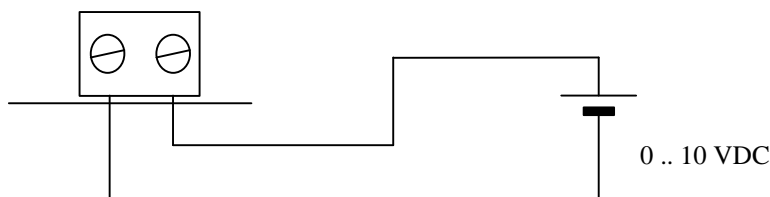
## 11) ENTRADA ANALOGICA

El autómatas dispone de una entrada analógica a través de un convertor de 8 bits TLC549. La entrada de este convertor dispone de una impedancia de 500 Ohmios o 15K. La selección de una u otra impedancia depende si está cerrado o abierto el JUMPER JP4 (cerrado=500 Ohms, abierto=15K).

Con una impedancia de 15K disponemos de una entrada de tensión con rango 0 a 10 Voltios, mientras que si seleccionamos una impedancia de 500 Ohmios podemos conectar al equipo directamente una sonda 0 a 20 mA o 4 a 20mA.

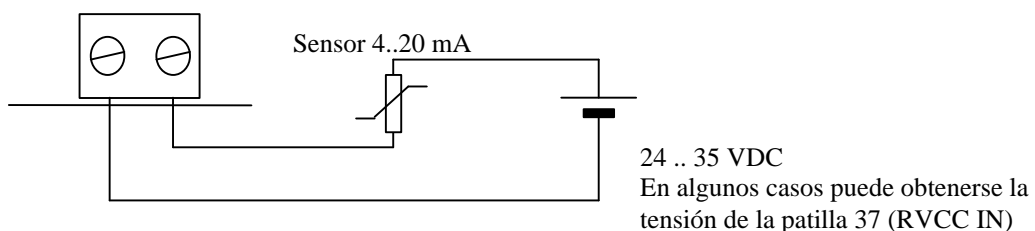
Si lo desea puede aumentar el rango de entrada para medir tensiones mayores, aunque tenga presente que estas, siempre estarán referenciadas a masa. A continuación algunos ejemplos:

0 a 10 Voltios



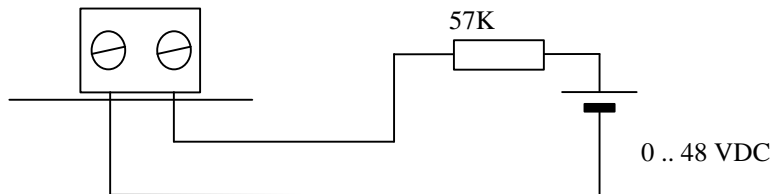
0 a 20 mA

Debe ajustarse la impedancia a 500 Ohmios, y es necesario asegurarse que la tensión de alimentación del sensor 4..20mA sea suficiente para una caída de 10 voltios.



0 a 48 Voltios

Se debe seleccionar una impedancia de entrada de 15K ( Jumper JP4 abierto).



En general la resistencia en serie puede calcular con:

$$R_s = (15 * V_{max} / 10) - 15$$

Donde  $R_s$  es el valor de la resistencia expresado Kohmios y  $V_{max}$  la tensión máxima que va a recibir la entrada analógica.

Si requiere de mayor resolución puede sustituir el integrado TLC549 por un integrado TLC1549, este ultimo dispone de 1024 puntos en lugar de 256. El software deberá ser retocado levemente.

En el disco dispone de un ejemplo de acceso a este integrado escrito en KEIL C y otro escrito en BASCOM 8051.

## 12) SALIDA A TRANSISTOR (PWM)

Además de las 8 salidas a relé, el equipo dispone de una salida a transistor. La ventaja de utilizar un transistor es que se puede actuar sobre él a una frecuencia muy superior a la que se puede actuar sobre un relé.

El transistor utilizado es más que un transistor, ya que en su interior incorpora la lógica necesaria para protegerlo contra sobreintensidades y sobretemperaturas. Se trata por tanto de un MOSFET especial, cuya puerta es compatible CMOS. Activando el puerto P1.3 se activa el transistor.

Si hubiese algún problema de exceso de tensión o temperatura, la propia puerta del transistor pondría a nivel bajo el puerto del microcontrolador, pudiendo este detectar que hay un problema en la salida.

La detección de fallo en el transistor no es difícil de detectar utilizando la entrada del modulo 0 del PCA.

Aunque el uso más frecuente de esta salida es generar una señal PWM utilizando el modulo 0 del PCA como salida. De esta forma la salida a transistor se convierte en una salida analógica de potencia.

Es factible el control de un servo, o directamente de un motor de corriente continua de pequeña potencia.

### 13) TECLADO

El autómatas dispone de un teclado matricial de seis teclas. La palabra matricial nos indica que las teclas no están individualmente conectadas a los puertos del microcontrolador, y que por tanto no pueden ser leídas directamente.



El equipo dispone de dos filas y tres columnas. Por una parte las filas están conectadas a los puertos P0.0 y P0.1 y por otra las columnas están conectadas a los puertos P0.3, P0.4 y P0.5.

Todo el teclado funciona con lógica negativa y su funcionamiento es el siguiente. Si deseamos leer la primera fila, debemos poner P0.0 a nivel bajo y P0.1 a nivel alto. Presionando una tecla de la primera fila se pondrá su columna asociada a nivel bajo, indicándonos que la tecla está pulsada.

Si deseamos leer la segunda fila, pondremos P0.0 a nivel alto y P0.1 a nivel bajo. Ahora las columnas están “conectadas” a los puertos asociados. No es una tarea complicada realizar unas rutinas que gestionen el teclado, basta mirar alternativamente el estado de una y otra fila.

### 14) ZUMBADOR

En la consola del autómatas hay un pequeño zumbador que puede dar avisos acústicos. Este se activa poniendo a nivel bajo la salida P0.2 del microcontrolador. Cuando está activo se emite una señal audible a 4 KHz.

Combinándolo con la lectura de teclado se puede conseguir que aumente la sensación táctil del teclado.

### 15) RELOJ EN TIEMPO REAL/MEMORIA EEPROM

Internamente al autómatas se le ha dotado de un bus I2C, como ya comentamos anteriormente, a este está conectado el display LCD de la consola. Además dispone de una memoria EEPROM de 8 Kbytes para almacenamiento de datos y de un reloj en tiempo real tipo MK41T56.

En los ejemplos que suministramos el bus I2C se simula por software a través de las patillas P16 y P17 (SCL y SDA respectivamente). Esta simulación por software

funciona correctamente en todos los microcontroladores, pero si dispone del 89c668 puede utilizar un BUS I2C por hardware.

Programando adecuadamente podrá realizar rutinas que consuman menos tiempo de procesador, disponiendo de más tiempo para realizar otras tareas. Lo que no es posible es acceder a este bus a más velocidad, puesto que esta está limitada por PHILIPS a 400 Kbaudios/segundo.

## 16) PUERTO SERIE RS232c

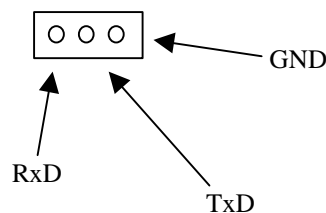
El autómeta que hemos desarrollado dispone de un puerto serie, que es manejado por la UART interna del microcontrolador. En la placa base hay un transceptor RS232c, esto es, un integrado que transforma las señales CMOS del microcontrolador en señales RS232c.

Este integrado es el MAX202/AD202, del cual puede leer su documentación en un PDF. Este puerto estándar permite que el microcontrolador pueda comunicarse con dispositivos externos tales como un MODEM telefónico o bien un ordenador PC con un programa de diagnóstico.

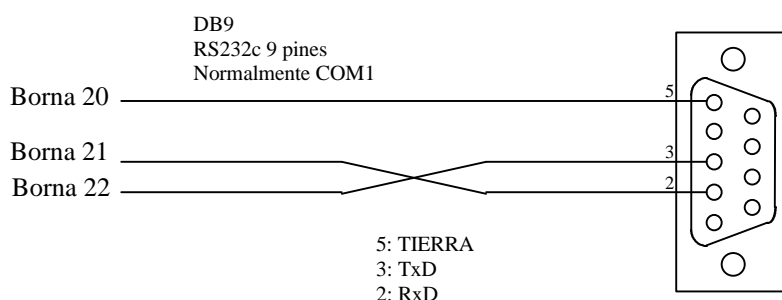
Normalmente todos los dispositivos dotados de puerto serie disponen de un conector SUB D estándar, ya sea de 9 pines o bien de 25 pines. Nuestro equipo no dispone de dicho conector.

Las señales están disponibles directamente sobre las bornas exteriores del equipo o bien a través del conector interno J4. Externamente sobre las bornas es mucho más sencillo de instalar para un electricista, ya que la utilización de cables estándares suele llevar problemas a la hora de pasar cables.

En cambio el conector interno J4 está pensado para ser utilizado durante la programación o bien para la instalación de algún periférico interno, como puede ser un MODEM GSM.



En cualquier caso no será complicado realizar un cable que conecte el equipo del J4 a un PC, o bien desde las bornas externas a un PC. Citamos el PC debido a que es necesario para reprogramar el microcontrolador.



Debe observarse que ambas masas van unidas, y que las señales RxD y TxD van cruzadas.

En cualquier caso este puerto serie puede comunicar dos equipos entre si a una distancia comprendida entre los 20 metros (mínimo garantizado por la norma) y unos 200 metros (máxima que hemos conseguido nosotros en condiciones muy favorables).

Es una buena solución para comunicar dos autómatas entre si, o bien un autómata con un terminal.

### 17) PUERTO SERIE RS485

En algunas ocasiones puede ser necesario conectar varios equipos en RED, con lo que la comunicación RS232c estándar no suele ser adecuada. Por ello se desarrolló el RS485, que mediante un económico cable trenzado + masa puede intercomunicar un mínimo de 32 equipos a una distancia superior a los 1300 metros.

A diferencia del RS232c la transmisión RS485 es half duplex, lo que quiere decir que no se puede transmitir y recibir información al mismo tiempo. Haciendo una analogía se podría decir que el RS232c es, en cierto modo, similar a un teléfono convencional de línea fija. Ambos interlocutores, si se empeñan, pueden hablar y escuchar al mismo tiempo.

En cambio el RS485 lo tendríamos que comparar con una red de Walky Talkys, puede haber decenas de ellos a la escucha, pero solo uno transmitiendo. Aquí a diferencia del teléfono hay que controlar el flujo, así que para hablar (transmitir) debe pulsarse normalmente un botón.

En el RS485 este flujo es controlado por el puerto P1.4, si este está a nivel alto el equipo transmite, mientras que si está a nivel bajo solo escucha. Por tanto durante la comunicación el software debe manejar esta señal.

Dado que el autómata dispone de dos transceptores (uno RS232 y otro RS485) y solo una UART, debemos seleccionar cual de los dos deseamos utilizar. Podremos seleccionar uno u otro con la ayuda del jumper JP3, tal y como se muestra en la siguiente figura:



En nuestros ejemplos BASCOM y KEIL hacemos una comunicación RS485 con nuestra tarjeta termómetro RS485. No sería difícil controlar varios termómetros u otros periféricos utilizando las mismas rutinas.

Tampoco es complicado utilizar cualquier tipo de tarjeta estándar RS485 o acceder a otros autómatas que dispongan de protocolos RS485 (por ejemplo MODBUS).

## **18) PERSONALIZACION DEL EQUIPO**

Si usted es fabricante de una determinada maquinaria, o bien realiza un determinado tipo de instalaciones, es posible que desee personalizar una serie de equipos. Esta personalización puede realizarse por dos motivos:

- a) Corporativo/funcional: Este punto consistiría básicamente en la modificación de la serigrafía del teclado, con el fin de incorporar su marca y sus colores o bien para adaptar las teclas a su aplicación.

Esta modificación no tiene un coste demasiado elevado, incluso para series pequeñas de 25 unidades, siempre y cuando las teclas o display no se desplacen del lugar.

- b) Económico: Se trata de modificar el proceso de montaje para eliminar todo lo que es superfluo en su aplicación, por ejemplo si usted no necesita el tranceptor RS485, relés, entradas, ..., es evidente que no montarlas es un ahorro.

Queremos añadir, que si lo desea es posible montar sobre el equipo cualquier microcontrolador de la familia 51 con encapsulado PLCC 44. Si consulta en la documentación PDF verá una amplia lista de ellos.

Solo recomendaríamos cambiar el microcontrolador en una serie grande, en la que una reducción de costo pueda ser significativa, pero nunca en pequeñas series o en prototipos, ya que lo idóneo es hallar un micro que no sea ISP y que se adapte a sus necesidades.

Por poner un ejemplo, si se desarrolla una aplicación que utiliza tan solo 8 Kbytes de memoria de programa, no necesita memoria externa, ni PCA, y se ha de construir una serie de 500 unidades. En este caso, utilizando un microcontrolador 87c52 puede suponer un ahorro aproximado de 4.500 euros. Quizás algo a tener en cuenta.

Una modificación más profunda, sería rehacer la electrónica para que esta se adapte a sus necesidades, por ejemplo añadir entradas analógicas, más relés, comunicación por red eléctrica, ...

Aunque esta modificación en realidad sería el desarrollo de un circuito a medida, que es el fuerte de nuestra actividad.

## **19) GARANTIA**

El uPLC V4 ha sido diseñado para que en condiciones normales pueda funcionar todo el tiempo, esto es 24 horas al día durante los 365 días del año. No debe estropearse.

Si esto ocurriera, el equipo está garantizado durante un año. Este año comenzará a computar el día que el equipo salga de nuestros talleres.

Durante este periodo de garantía el equipo será reparado o sustituido por uno nuevo sin coste alguno dentro del territorio nacional y/o Portugal. Queda excluida de garantía cualquier avería derivada de una mala utilización (ej: Conectar 220V directamente a la entrada analógica) o por fenómenos meteorológicos (ej: tormentas).

Del mismo modo quedan excluidos de garantía los equipos cuyo microcontrolador haya sido bloqueado como consecuencia de cambiar el vector del BOOTLOADER.

Quedan también excluidos de garantía equipos manipulados, , esto es, componentes cambiados, pistas cortadas, cables soldados, soldaduras adicionales, taladros, ...

El procedimiento de garantía será informar previamente a nuestras oficinas que se va a remitir un equipo averiado. Nosotros le informaremos por que agencia debe enviarnos el equipo, esta pasará por su domicilio a recogerla a una hora acordada.

Con el equipo averiado debe adjuntarse un escrito detallando la avería y como se detectó esta. Es importante para nosotros diferenciar el “equipo no funcionó nunca” del “equipo dejó de funcionar”.

Si después de estudiar el equipo, este funciona, y el problema se debía a un software mal desarrollado, se procederá a reenviar el equipo a contrarembolso de los gastos de ida y vuelta.

En el caso de estar realmente averiado, nosotros le remitiremos sin cargo el equipo junto un boletín de reparación en el que se indicará el motivo de avería.

Ibercomp S.A. no se hace responsable de ninguno de los daños que pueda ocasionar el equipo. Es su obligación, como instalador, cumplir las normativas de seguridad vigentes y evaluar si se adapta a su aplicación. Nosotros simplemente suministramos el equipo como un componente más de su conjunto.

## **20) CONSEJOS**

Somos conscientes que un apartado de este tipo está fuera de lugar en un manual técnico, y que incluso puede molestar a algunas personas. Pero dado que el objetivo de nuestra empresa nunca ha sido, ni será, puramente mercantil sino que es la ilusión del prosperar técnico de nuestros clientes y en definitiva del nuestro, consideramos que un fracaso de nuestros clientes es también nuestro.

Este es el motivo, por el cual nos hemos decidido incluir esta reflexión y compartir, a modo de confidencia, lo que pensamos desde hace mucho tiempo con usted. Confiamos que esta aportación sea constructiva.

Quisiéramos poder decir que todas las buenas ideas tienen éxito pero algunas veces fracasan por una mala gestión, por desarrollarse en el momento inoportuno o simplemente por insuficiencia de recursos.

Por desgracia en la mayoría de los casos el no llegar a buen puerto es una consecuencia de una incapacidad técnica, de una falta de dedicación o de la inexperiencia.

Durante el desarrollo recomendamos que pruebe todo lo posible cada paso que da al escribir el software y que realice los proyectos con tiempo. Con la experiencia obtenida al observar cientos de clientes que han realizado proyectos con nuestros equipos, hemos llegado a la conclusión que cualquier proyecto realizado con prisas, sin experiencia previa, sin testear debidamente o que no se le pueda dar posteriormente un mantenimiento: **está condenado de ante mano al más estrepitoso de los fracasos.**

Si tiene dudas si es capaz de acometer el desarrollo de un proyecto **no lo acepte**. Tenga presente que en nuestro gremio el riesgo es negligencia. Así que deberá seguir trabajando duro adquiriendo conocimientos y experiencia como resultado de su trabajo. Seguro que en un futuro si tendrá seguridad de poderlo afrontar, y lo hará bien.

Si ha decidido en esta vida por el duro camino de ser empresario, no se fíe de las personas solo por sus palabras o sus títulos académicos, lamentablemente en el mundo informático/electrónico es muy fácil dar el pego pronunciando un discurso ininteligible lleno de acrónimos.

Tampoco se deje cegar por la ilusión de alumnos becarios y en prácticas, proyectos de fin de carrera, ... en la mayoría de los casos acaban siendo soluciones mediocres e inacabadas.

Por último, queremos añadir, que disponer de unas herramientas y un entorno de trabajo adecuado, facilita enormemente cualquier desarrollo.

Si lo desea, puede hacernos cualquier tipo de sugerencia, estaremos encantados de escucharle.