Creative Control Concepts

PIC-DIO & Mini-DIO

PIC based Digital I/O Nodes for the Circuit Cellar® Home Control System (HCS)

November 1999

v1.13

Disclaimer

- While we have attempted to provide accurate and up-to- date information in this manual, Creative Control Concepts makes no representations or warranties respecting its contents.

- We reserve the right to make periodic changes to the text and to issue new editions of this manual without notification.

- Occasionally in this manual, we refer to other manufacturers' products. Such references do not constitute an endorsement of these products, but are included for the purpose of illustration or clarification. We do not intend such technical information and interface data to supersede information provided by individual manufacturers.

- Devices sold by Creative Control Concepts are covered by the warranty and patent indemnification provisions appearing in its Terms of Sale only. Creative Control Concepts makes no warranty, express, statutory, implied, or by description regarding the information set forth herein or regarding the freedom of the described devices from patent infringement. Creative Control Concepts makes no warranty of merchantability or fitness for any purposes.

- Creative Control Concepts reserves the right to discontinue production and change specifications and prices any time and without notice.

- This product is intended for use in normal commercial applications. Applications requiring extended temperature and unusual environmental requirements, or applications requiring high reliability such as military, medical life support or life-sustaining equipment, are specifically not recommended.

Conditions of Sale and Product Warranty

Creative Control Concepts and the Buyer agree to the following terms and conditions of Sale and Purchase:

1. Creative Control Concepts extends the following warranty: a factory manufactured circuit board or assembly carries with it a one-year warranty covering both parts and labor. Any unit which is found to have a defect in materials or workmanship will, at the discretion of Creative Control Concepts, be repaired or replaced.

2. A minimum inspection fee must be prepaid for the repair of units that are no longer under warranty. Call Creative Control Concepts for a current list of fees.

3. Creative Control Concepts will not be responsible for the repair or replacement of any unit damaged by user modification, negligence, abuse and mishandling, or improper installation.

4. Creative Control Concepts is not responsible to the Buyer for any loss or claim of special or consequential damages.

5. All units returned for repair must first receive prior authorization from Creative Control Concepts. A return authorization number may be obtained by phone, email, or letter. Please retain a record of the return authorization number since most subsequent correspondence will refer to this authorization. Under no circumstances should any product be returned to Creative Control Concepts without such authorization. Creative Control Concepts assumes no responsibility for returns unaccompanied by an authorization number. All returns must be shipped prepaid and ought to be insured. Creative Control Concepts is not responsible for losses or damage during shipment. Repaired units will be returned with postage and insurance paid.

6. Creative Control Concepts reserves the right to alter any feature or specification at any time. This right extends to fees, charges, and any other conditions or warranties contained herein.

Chapter

1

PIC-DIO/Mini-DIO Overview

The PIC-DIO and Mini-DIO provide you with cost-effective 8-bit digital I/O in a small package for your HCS Home Automation System or any other system with a serial port. You can construct an entire DIO node with just a PIC-DIO or Mini-DIO chip, RS-485 or RS-232 Transceiver, and a few discrete parts. Both chips recognize the bit and byte manipulation commands sent from your HCS II controller board. The PIC-DIO also recognizes the string output command generated by the LPT() XPRESS command allowing you easily interface a printer to your HCS. Unlike the original HCS DIO-Link, the PIC-DIO has a BUSY input pin so your PIC-DIO can monitor the status of a connected printer or other device instead of using a predefined delay value between characters. The PIC-DIO also has 3 status LED's which give you useful information about the HCS network and the parallel port.

The PIC-DIO/Mini-DIO can also be used with any system that has serial data capabilities. Need remote digital I/O for your next project? Just connect your controller's serial port to a Mini-DIO using RS-232 or RS-485 transceivers. Generating bit and byte manipulation commands on the serial network is a snap. You can even add checksums to validate the command packet data. Do you need to monitor some I/O with your PC? Just use a Mini-DIO and a MAX232 serial interface chip connected to your PC's serial port. Sample circuits are included later in this manual.

Most information in this manual applies to both the PIC-DIO and the Mini-DIO. Information that applies only to one or the other is noted.

Chapter

2

PIC-DIO/Mini-DIO Protocol Overview

The PIC-DIO and Mini-DIO recognize all the DIO-Link bit and byte control commands used in a Circuit Cellar® HCS II Home Automation system. The PIC-DIO also understands the output string command T DP= for use with devices like printers. However, because these commands are simple ASCII strings, these DIO interfaces can be used on systems other than the HCS II Home Automation System where digital I/O is needed somewhere on a serial network. When used on a multi-drop RS-485 network, up to eight may be connected on the same network.

Commands are sent to the PIC/Mini-DIO via a standard serial port. When connected to a Circuit Cellar® HCS II, they send and receive commands over the HCS II RS-485 multi-drop serial network. To connect a PIC/Mini-DIO to a PC serial port, use a voltage shifter such as a MAX232 from Maxim. When connecting PIC/Mini-DIO's to an RS-485 network, use a converter such as a 75176.

The PIC/Mini-DIO communicate with their host using the following serial settings:

```
9600 Baud
  8 Bits
No Parity
1 Stop Bit
```

Command/Packet Format

When commands are sent to a PIC/Mini-DIO, they must be in one of the following formats:

```
 ! DIO# command\n
#XX DIO# command\n
```

Packets starting with ! do not use checksums. If you are using checksums (see below) all packets sent to the PIC/Mini-DIO must start with #. The computed checksum goes in place of the XX. Replace the # after DIO with the address number of the node which must be between 0 and 7. All packets must end with a carriage return (ASCII 13) and/or line-feed (ASCII 10).

Replies from the PIC/Mini-DIO will always start with $. If the original command packet used a checksum, the response will also have a checksum, otherwise it will not:

```
 $ DIO# reply\n
$XX DIO# reply\n
```

Checksums

To ensure reliable communications between the host computer and the PIC/Mini-DIO, checksums can be used in packets to help ensure the validity of data. If the PIC/Mini-DIO receives a packet with an invalid checksum, the packet is simply ignored and no response is sent. Your controller must recognize the lack of a response and resend the packet after a given timeout period. When connected to an HCS II controller, a received packet with a bad checksum is indicated by an E in the node status window.

Checksums for command packets are generated as follows:

1. Construct the command packet with zeros in the checksum location. Do **NOT** include the trailing carriage return:

   ```
   #00 DIO4 QDP
   ```

2. Add up all ASCII values of all characters in the packet, except for the \n:

   ```
   #         23    (these are HEX values)
   0         30
   0         30
   space     20
   D         44
   I         49
   O         4F
   4         34
   space     20
   Q         51
   D         44
   P         50
   Total     B8    (only the low byte is kept)
   ```

3. Take the 2's complement of the checksum byte computed above:

   ```
   -(B8) = 48
   ```

4. Convert the checksum byte into 2 ASCII chars and replace the zeros in the checksum location:

   ```
   #48 DIO4 QDP
   ```

To validate the checksum of a received packet from a PIC/Mini-DIO, do the following:

1. Discard the trailing carriage return (and linefeed if present):

   ```
   $E7 DIO4 DP=A3
   ```

2. Convert the ASCII checksum into a single hex byte.

3. Replace the checksum characters with zeros in the packet:

   ```
   $00 DIO4 DP=A3
   ```

5. Add all ASCII values of the characters in the packet, minus any trailing \n or \r:

   ```
   $         24    (these are HEX values)
   0         30
   0         30
   space     20
   D         44
   I         49
   O         4F
   4         34
   space     20
   D         44
   P         50
   =         3D
   A         41
   3         33
   Total     19    (only the low byte is kept)
   ```

4. Add the checksum byte to the packet total. If the result is 00, the packet is probably valid:

   ```
   E7 + 19 = 00    (The packet is okay)
   ```

That's all there is to it! Here is a great hint from the original Circuit Cellar® DIO-Link User Manual:

"After you think about it for a while you'll write a single subroutine to calculate or verify the checksum. If you hand the routine a line with a 00 checksum, it'll replace the zeros with the new checksum so you can send the line. If you hand it a (presumably) valid received line it will do the same, but also return the calculated checksum to the caller as an integer."

Chapter

3

PIC-DIO/Mini-DIO Command Overview

The PIC-DIO and Mini-DIO recognize all the DIO-Link bit and byte control commands used in a Circuit Cellar® HCS II System. These commands are all that is needed to query and set the IO port via a byte or bits. The PIC-DIO also supports the DIO-Link's T DP= command allowing you to output a string of characters to an external device like a printer.

All command packets must end with a carriage return (indicated by <CR>).

- **Query Inputs As A Byte**

  This command allows you to query the state of the IO port and have all bit states returned as a hex byte.

```
! DIO2 QDP<CR>
#cc DIO4 QDP<CR>     (cc is computed checksum byte)
```

  This will return the current value of the I/O port. The state of the input bits and the current state of any outputs are returned as an ASCII hex byte in the following format:

```
$ DIO2 DP=XX<CR>
```

  where XX is the current port value. If a checksum was included in the original packet, one will be included in the response.

  In XPRESS, you can query the PIC-DIO port as a byte using the NETBYTE command. For example, if you have your PIC-DIO configured for address 3 (DIO3), the corresponding NETBYTE would be 3. Below is a simple example:

```
IF ADC(4) > 25 THEN
   NETBYTE(3) = 178
END

IF NETBYTE(3)  > 80 THEN
   Console = "Threshold Exceeded"
END
```

  See your XPRESS manual for the list of NetByte numbers that correspond to each DIO-Link address.

- **Query Individual Port Bits**

  You can query individual port bits by specifying the specific bit number (0 .. 7) in the command:

```
! DIO2 QDP.4<CR>
#cc DIO4 QDP.6<CR>  (cc is computed checksum byte)
```

  These will return the current state of the specified port:

```
$ DIO2 DP.6=X<CR>
```

  where X will be a 0 or a 1. If a checksum was included in the original packet, one will be included in the response.

  Note it is often easier to poll the DIO using a byte query command and then check the bit state in software. This way you only have to worry about one response format to parse, not two.

  In XPRESS, individual bits on a PIC-DIO can be queried using the NetBit command. Using the DIO3 example again, the code below will query bit 3 and set bit 6 accordingly:

```
IF NETBIT(27) = ON THEN
   NETBIT(30) = OFF
END
```

  See your XPRESS manual for the list of NetByte numbers that correspond to each DIO-Link address.

- **Output Byte To The IO Port**

  This command allows you to set all the bits at once by sending the bit states as a hex byte:

```
! DIO2 S DP=XX<CR>
#cc DIO4 S DP=XX<CR>       (cc is computed checksum byte)
```

  where XX is the desired byte to be output on the IO port. On both the PIC-DIO and Mini-DIO, the STROBE line will pulse low for 250µs. This allows the DIO to signal a connected device that new data is available.

  NOTE! The PIC/Mini-DIO uses open collector outputs with pull-ups that are built into the PIC. Thus, setting a bit to a 1 will output a high via the pull-up. However, a bit set to 1 can also be used as an input. Thus if a pin set to 1 is pulled to ground externally (i.e. its state is LOW), a 0 will be returned when it is queried. Thus, any port you want to be an input, set to 1. To treat it like an output, set it to either 1 (pin floats high via pull-up) or 0.

  WARNING! *Since these are bi-directional I/O pins, you must be careful when setting a bit to 0. If that pin is driven HIGH externally and you tell the PIC to set it LOW, the pin will short and the resulting currents will damage the PIC!*

  No response is generated to a Set Byte command.

- **Set Bit State On IO Port**

  Individual bits can be manipulated via this command:

```
! DIO2 SDP.Y=X<CR>
#cc DIO4 SDP.Y=X<CR>       (cc is computed checksum byte)
```

  where Y is the bit number to change (0 .. 7) and X is the desired state (0=off 1=on). No response is generated to a Bit Set command. When a bit is set, the STROBE line is pulled LOW for 250µs to signal new data.

  See the NOTE and WARNING above. For XPRESS examples, see the first two sections.

- **Output String Via I/O Port *(PIC-DIO ONLY!)***

  The PIC-DIO has a 128 character buffer, which allows you to send long sequences of data to be output on the IO port. This feature is often used to output text onto a parallel printer. The PIC-DIO has STROBE and BUSY control lines so you can easily interface any Centronics parallel printer to it. However, it is not limited to just printers. You can connect it to any device's data bus and signal that new data is available via the strobe line. When data is sent via the T= command, the strobe line is pulled low for 25µs. Most printers can handle a pulse as small as 1µs so this is plenty.

  To output a string of characters, send the string in this format (the space between T & DP is OPTIONAL!):

```
! DIO6 T DP=Hello World\nThis will print on the next line\n<CR>
```

  Note the use of \n. The PIC-DIO recognizes some C-style escape sequences so you can easily send special characters and control characters using visible ASCII characters:

```
\cn     Control character n (Ctrl-M = \cM)
\e      Escape character (ASCII 27)
\f      Form Feed (ASCII 12)
\t      Tab (ASCII 9)
\n      New Line (Carriage Return + Line Feed)
\r      Carriage Return only (leftmost column, same line)
\xnn    Print specified character using hex ASCII value for nn
        You must use 2 characters! (pad with 0 if necessary)
\\      Print single backslash
```

  Any other character following a single backslash will be printed normally, but the backslash will be ignored and will not be output.

  Note that these escape sequences allow you to send the same data in different ways. For example, \x0C and \cM are the same. Case is ignored for the \c sequence so \cG and \cg are equivalent. However, the character after the slash MUST be in lower case.

In XPRESS, sending text to a connected printer is very easy. The LPT() command is used and the number put into the parenthesis should match the appropriate PIC-DIO address.

```
LPT(3) = "System Status: OK"
```

All of the above control codes can be used in an LPT() statement.

- **Reset PIC-DIO Interface *(PIC-DIO v1.1 only!)***

The PIC-DIO can be reset from a remote location using a simple network command:

```
! DIO4 RESET<CR>
#cc DIO4 RESET<CR>          (cc is computed checksum byte)
```

Note that the PIC-DIO only looks for the R in the command so the rest of the word RESET can be dropped if you wish. The PIC-DIO performs a hard reset by timing out the PIC watchdog. The PIC-DIO will also turn on all the status LEDs for 2 seconds. Thus, it will take your PIC-DIO a little less than 3 seconds to recover from a network reset.

There is no XPRESS command to send the RESET command to a PIC-DIO so you have to do it yourself using the Network command in XPRESS. The example below will reset PIC-DIO #3:

```
NETWORK = "DIO3 RESET"
```

Chapter

4

PIC-DIO/Mini-DIO Addressing

The PIC-DIO and Mini-DIO can be used in a multi-drop configuration where each unit has a unique address. Addresses are 4 characters: DIO# where # is the unique ID of the node which must be between 0 and 7. Up to eight nodes of any combination of PIC-DIOs and Mini-DIOs can be used on the same HCS RS-485 network. If you use a PIC-DIO or Mini-DIO connected to an RS-232 serial port, you can only have one connected at a time so you should use the default node address of DIO0.

Setting the address of a PIC-DIO

The address of a PIC-DIO interface is set using three jumpers on J2. Each pair of pins on J2 is used to set a specific bit of the node address. The pair closest to the bottom of the board are for bit 0 and the pair closest to the 16C63A are for bit 2. Installing a jumper on J2 signifies a 1 while leaving the pins open signifies a 0. Thus, if you install a jumper on the pair of pins closest to the bottom of the PCB, the network address will be DIO1. **NOTE! This is different from how v1.0 handled the address! If you upgrade your PIC-DIO, take this into account when setting your jumpers!** These pins must stay connected at all times since they are referenced each time the PIC-DIO resets.

Setting the address of a Mini-DIO

The Mini-DIO is based on a PIC that has built in EEPROM memory. When you configure the address, it is saved in EEPROM so it is retained even if the power is removed. To configure your Mini-PIC, follow these simple steps. You can do it on a breadboard or with your completed node circuit disconnected from your network:

1. Connect the crystal and 20pf capacitors as shown in the schematic.
2. Connect the Vss and Vdd lines (Pins 5 and 14 respectively) but DON'T power it up yet.
3. Connect the CFG pin (Pin 18) to GND which signals the PIC to enter configuration mode.
4. Set the address using Pins 6-8 (A0-2 respectively). +5V is a 1 and GND is a 0
   Example: Pin 6 -> GND and Pins 7 & 8 -> +5V sets the address to DIO6
5. Power up the PIC. When you do this it will read the set address on Pins 6-8 and store the address in EEPROM. You can optionally connect the serial TX pin (Pin 2) to a serial terminal/port using a voltage shifter like a MAX232. After configuration, the software version and current address will be output on the serial out line.
6. Power down the PIC after 5 seconds and disconnect the connections to Pins 6-8 since these are normal I/O lines during normal operation.

That is all that is required. The address will stay set unless you change it (by powering up the Mini-PIC with the CFG line pulled low.

Chapter

5

PIC-DIO/Mini-DIO Pinouts

Below are the pinouts for the Mini-DIO and the PIC-DIO:

MINI-DIO

Pin 1	Serial Enable
Pin 2	TX
Pin 3	RX
Pin 4	/RESET
Pin 5	Vss
Pin 6	I/O Bit 0
Pin 7	I/O Bit 1
Pin 8	I/O Bit 2
Pin 9	I/O Bit 3
Pin 10	I/O Bit 4
Pin 11	I/O Bit 5
Pin 12	I/O Bit 6
Pin 13	I/O Bit 7
Pin 14	Vdd
Pin 15	OSC2
Pin 16	OSC1
Pin 17	/STROBE
Pin 18	/CFG

PIC-DIO

Pin 1	/RESET
Pin 2	Node Addr Bit 0
Pin 3	Node Addr Bit 1
Pin 4	Node Addr Bit 2
Pin 5	/STROBE
Pin 6	/BUSY
Pin 7	NC
Pin 8	Vss
Pin 9	OSC1
Pin 10	OSC2
Pin 11	/Serial Error
Pin 12	/Network Traffic
Pin 13	/Parallel Out
Pin 14	NC
Pin 15	NC
Pin 16	Serial Enable
Pin 17	TX
Pin 18	RX
Pin 19	Vss
Pin 20	Vdd
Pin 21	I/O Bit 0
Pin 22	I/O Bit 1
Pin 23	I/O Bit 2
Pin 24	I/O Bit 3
Pin 25	I/O Bit 4
Pin 26	I/O Bit 5
Pin 27	I/O Bit 6
Pin 28	I/O Bit 7

Anytime PORTB changes due to an S command, the STROBE line is pulled low for 250µs.
Data that is output due to a T= command causes strobe pulses of 25µs.

The PIC-DIO will timeout and dump the current buffer if the BUSY line stays HIGH for more than 25ms.

Connecting your PIC-DIO to a Printer

If you wish to connect a printer to a PIC-DIO, you should connect the following pins of the PIC-DIO to a normal male Centronics connector:

PIC-DIO J1	Printer
Pin 1	Pin 1 (/STROBE)
Pin 3	Pin 2 (Data 0)
Pin 5	Pin 3 (Data 1)
Pin 7	Pin 4 (Data 2)
Pin 9	Pin 5 (Data 3)
Pin 11	Pin 6 (Data 4)
Pin 13	Pin 7 (Data 5)
Pin 15	Pin 8 (Data 6)
Pin 17	Pin 9 (Data 7)
Pin 21	Pin 11 (Busy)
Pins 2,4,6,8,10,12,14,16,18,20,22	Pin 19-29 (Ground)

The easiest way to connect your PIC-DIO to a printer is to use IDC connectors (a 36-Pin Male Centronics and a 2x13 header connector) and 26 conductor ribbon cable. Align the cable on the Pin 1 side of each connector.

If you bought a kit from CCC, a 3' printer cable is provided. Simply plug the IDC connector onto the PIC-DIO board making sure the Pin 1 mark on the connector is next to the '1' on the circuit board. Next connect the Centronics connector to your parallel printer.

Chapter

6

PIC-DIO/Mini-DIO Miscellaneous

Mini-DIO

The Mini-DIO chip has a 20 character buffer to store incoming commands. This is more than enough to handle both query and set commands. Since the serial input is interrupt driven, the Mini-DIO will handle an incoming packet while processing the current command in the buffer.

You may be curious why the RTCC input is used for the serial in line. Because the INT input is on Port B, it could not be used since all of Port B is needed for I/O since these pins have internal pull-up resistors. By setting the RTCC counter to FF and incrementing on a falling edge, an RTCC overflow interrupt is generated when a new byte's stop bit (H to L) arrives. Once a character is received, the RTCC counter is reset to FF in prep for the next character.

The Mini-DIO I/O ports can sink 25mA each but the combined current of all pins cannot exceed 150mA.

Although the Mini-DIO does not handle string output commands (T=), whenever an I/O Pin changes via a Set Byte or Set Bit command, the STROBE line is pulsed low for 250µs. This allows you to connect a Mini-DIO to another system and signal it when new data is ready.

PIC-DIO

The PIC-DIO has a 128 byte buffer so you can send long lines to a printer or other device like a microcontroller data bus. However, when sending a long command, the PIC-DIO may miss a subsequent packet if the printer is busy for any substantial amount of time when the PIC is trying to send the buffer. If the PIC-DIO times out (after 25ms) waiting for the BUSY line to clear, the current buffer is dropped so the next command can be received. Using a printer with its own buffer will ensure commands are processed quickly and this reduces, if not eliminates the risk of missed packets. Even though the PIC-DIO has a 128 character buffer for incoming serial packets, the HCS has a 96 character limit on network packets which translates to approximately 80 chars of text that can be put into a single LPT() command.

Since the STROBE output is pulsed low for 25µs for each character and the code requires approx. 50-100µs to process each character at 3.6864 MHz, the PIC-DIO can process 8000 characters per second. However, some printers will be slower than this and by holding the BUSY line high during processing, the data rate will be much lower.

If you connect the PIC-DIO to a device which does not have a BUSY line and you wish to use the T= string output command, make sure you tie the BUSY line to GND. Otherwise, all

buffers with T=string commands will timeout and be dumped. Also, ensure the device can handle the data changing every 100ms.

Status LED's

The PIC-DIO has three status LED's which give the user an indication of what is happening on the HCS network and the parallel port.

The Red LED indicates a serial error has occurred on the RS-485 network. Errors can include checksum failures, buffer overflows, and framing errors. The most common error is usually a checksum failure due to a borderline network. If your network is too long, uses cheap wire, or is not properly terminated, this can often cause packet corruption and will cause checksum failures. Thus, your PIC-DIO can help indicate when your network is not operating properly.

The Yellow LED indicates RS-485 network traffic has been received. Each flash indicates the reception of a network packet. Note that the LED flashes for ALL packets, not just packets addressed to the PIC-DIO. Also, note that the length of an LED flash is NOT indicative of the length of the packet. The PIC-DIO checks the node address while the rest of a packet is still being received and thus terminates reception of a packet if the address does not match the preset PIC-DIO address.

The Green LED indicates that data is being output onto the parallel port. The LED stays on during the entire transmission of a string received in a TDP command. This indicator is useful in debugging printer communication problems. For example, if your printer won't print and you wonder if the HCS is sending the string or not, the Green LED will indicate if a valid string was received and sent. If it does, the problem is with your printer, not your HCS.

Chapter

7

Constructing your PIC-DIO

PIC-DIO Parts List

If you purchased a kit from CCC, check the contents to ensure you have all the necessary parts:

Qty	Description	Board Ref
1	PIC-DIO Circuit Board	NA
1	28-Pin PIC-DIO IC (16C63A)	U1
1	75176 RS-485 IC	U2
1	28-Pin Narrow IC Socket	U1
1	8-Pin IC Socket	U2
1	7805 TO-220 1A Voltage Regulator	U3
1	3.6864 MHz Low Profile Crystal (oval metal can)	X1
2	20pf Capacitors (look like green resistors)	C1, C2
1	0.1µF Capacitor (blue teardrop shape or beige)	C3
1	100µF 16V Electrolytic Capacitor (blue can shape)	C4
3	10K ¼ watt Resistors (brown-black-red-gold)	R1, R2, R3
3	180Ω ¼ watt Resistors (red-red-brown-gold)	R4, R5, R6
1	100Ω ¼ watt Resistor (brown-black-brown-gold)	R7
1	Red T-1¾ LED	D1
1	Yellow T-1¾ LED	D2
1	Green T-1¾ LED	D3
1	2x13 Pin Header	J1
1	2x3 Pin Header	J2
1	2 Pin Header	J5
2	2 Position Terminal Blocks	J3, J4
4	Shorting Blocks	NA
1	Ribbon Cable w/IDC & Centronics Connectors	NA

Here are some hints when constructing your PIC-DIO:

- When building the circuit, mark off each connection on the schematic after it is soldered into place.
- Use ESD protection when handling the ICs! If you do not have a strap, touch something grounded before handling the chips.
- Construct the circuit using the IC sockets, leaving the chips alone. When you are done, check that +5V and GND appear on the proper pins in the IC sockets. Then install the chips.
- Locate the Crystal and 20pf capacitors as close to the OSC pins as possible (if you are building on perf board).
- Install the shorting block to terminate the RS-485 network if necessary (i.e. if this module is on the extreme end of the network run). Only 2 modules can have active terminators at the same time!

Construction Steps

1) Orient your circuit board so "Component Side" is on the top surface located near the lower left corner of the board. Use the mask diagrams on the next page to locate the proper positions for the parts.

2) Install Resistors R1-R7 and Capacitors C1 & C2. These parts have no polarity and can be installed either way.

3) Install the crystal (X1). There is no polarity so it can be installed either way.

4) Install the IC sockets. Ensure the cutout on the socket edge matches the cutout on the mask diagram. The 28-pin cutout will be on the left and the 8-pin cutout will be towards the top. **BE CAREFUL when soldering the IC sockets. Make sure you do not create a solder bridge between pins or with a nearby trace. Certain traces run close to the IC socket pins.** Do NOT install the IC's at this time!

5) Install capacitor C3. If it is blue and has a + sign on it, it must be installed so the + signs on the capacitor and circuit board align. If C3 is beige, (usually labeled with 104) it can be installed either way.

6) Install capacitor C4. C4 MUST be installed one way to ensure proper polarity. C4 has a band with minus signs on it to indicate the negative lead. This lead must be towards the top of the board when the part is inserted in the hole without the + sign on the circuit board mask.

7) Install the LEDs D1-D3. These parts must be installed so the lead near the flat edge is installed in the lower LEFT hole. See the mask diagram for more detail.

8) Install the header connectors J1, J2, & J5. Be careful when soldering J1 to avoid creating solder bridges between pins or with nearby traces.

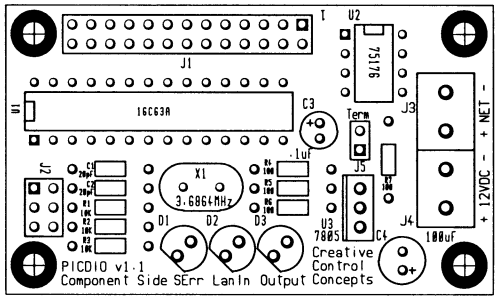
9) Install the Terminal Blocks J3 & J4 so that the terminals point to the right.

10) Install the 7805 voltage regulator (U3) so the side with the metal tab is on the **LEFT**. See the mask diagram for more detail.

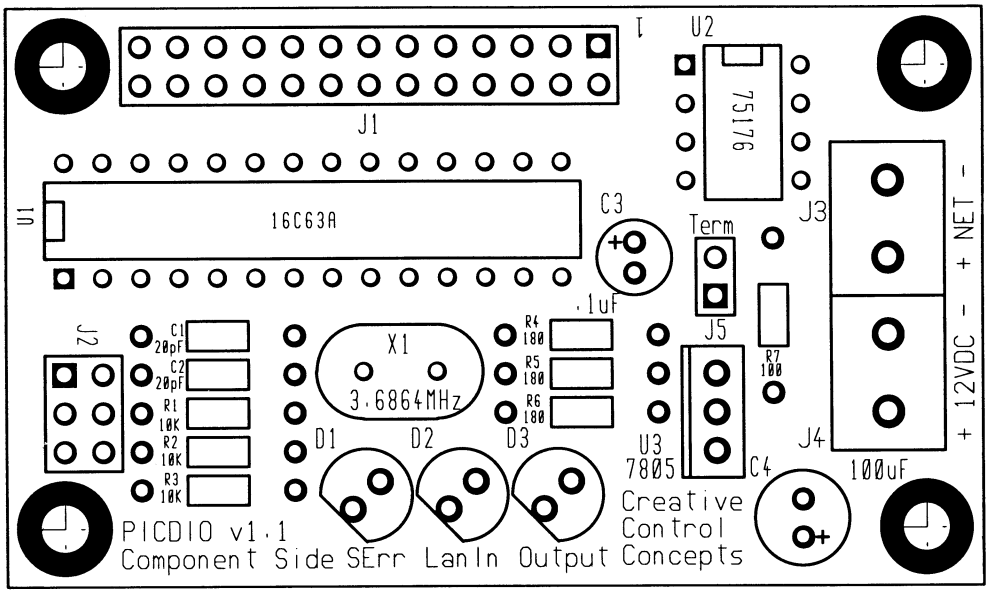
11) Apply power to the board without the IC's installed and check that +5 volts appears at the proper pins (see the schematic). Disconnect power & install the IC's. Make sure Pin 1 of each chip is installed so it is closest to the socket cutout.

12) Install jumpers on J2 to set the module address. A jumper indicates a 1 and J2 represents address bits 0 through 2 from the bottom up. Thus installing a jumper on the two pins CLOSEST to the 16C63A would set the address to DIO4. If network termination is necessary, install a jumper on J5.

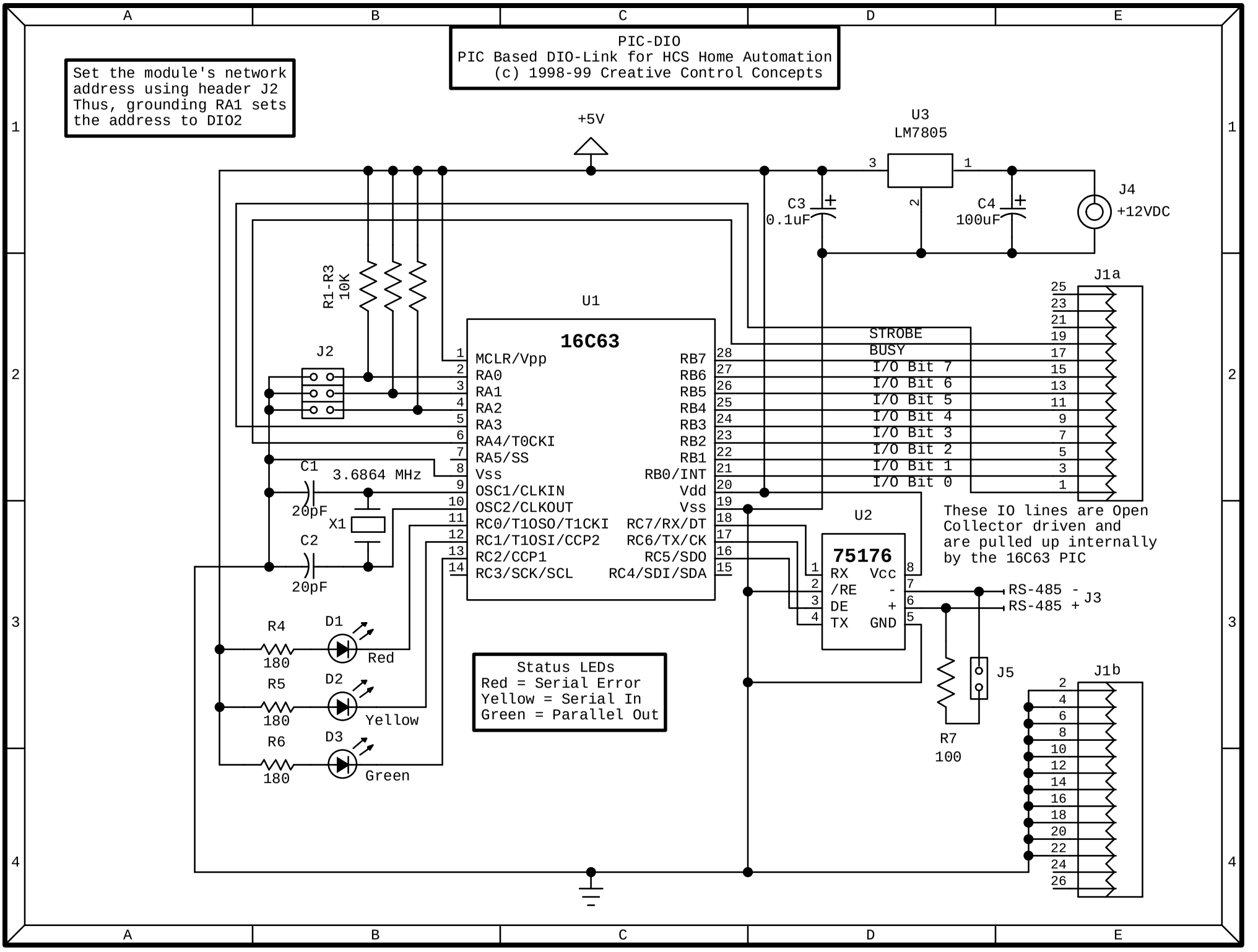
13) Connect your PIC-DIO to your HCS network, +12VDC, and power it up. The Yellow LED should begin flashing as network traffic is received. If you send data to your PIC-DIO from HOST or your XPRESS program, the green LED will flash as that data is outputted.



1X Circuit Board Mask



2X Circuit Board Mask


PIC-DIO
PIC Based DIO-Link for HCS Home Automation
(c) 1998-99 Creative Control Concepts
Set the module's network address using header J2 Thus, grounding RA1 sets the address to DIO2
+5V
U3
LM7805
C3
0.1uF
C4
100uF
J4
+12VDC
R1-R3
10K
J2
U1
16C63
MCLR/Vpp
RA0
RA1
RA2
RA3
RA4/T0CKI
RA5/SS
Vss
OSC1/CLKIN
OSC2/CLKOUT
RC0/T1OSO/T1CKI
RC1/T1OSI/CCP2
RC2/CCP1
RC3/SCK/SCL
RB7
RB6
RB5
RB4
RB3
RB2
RB1
RB0/INT
Vdd
Vss
RC7/RX/DT
RC6/TX/CK
RC5/SDO
RC4/SDI/SDA
C1
3.6864 MHz
20pF
X1
C2
20pF
J1a
STROBE
BUSY
I/O Bit 7
I/O Bit 6
I/O Bit 5
I/O Bit 4
I/O Bit 3
I/O Bit 2
I/O Bit 1
I/O Bit 0
These IO lines are Open Collector driven and are pulled up internally by the 16C63 PIC
U2
75176
RX
/RE
DE
TX
Vcc
-
+
GND
RS-485 -
RS-485 +
J3
R4
180
D1
Red
R5
180
D2
Yellow
R6
180
D3
Green
Status LEDs
Red = Serial Error
Yellow = Serial In
Green = Parallel Out
R7
100
J5
J1b

Chapter

8

Constructing your Mini-DIO

Mini-DIO Parts List

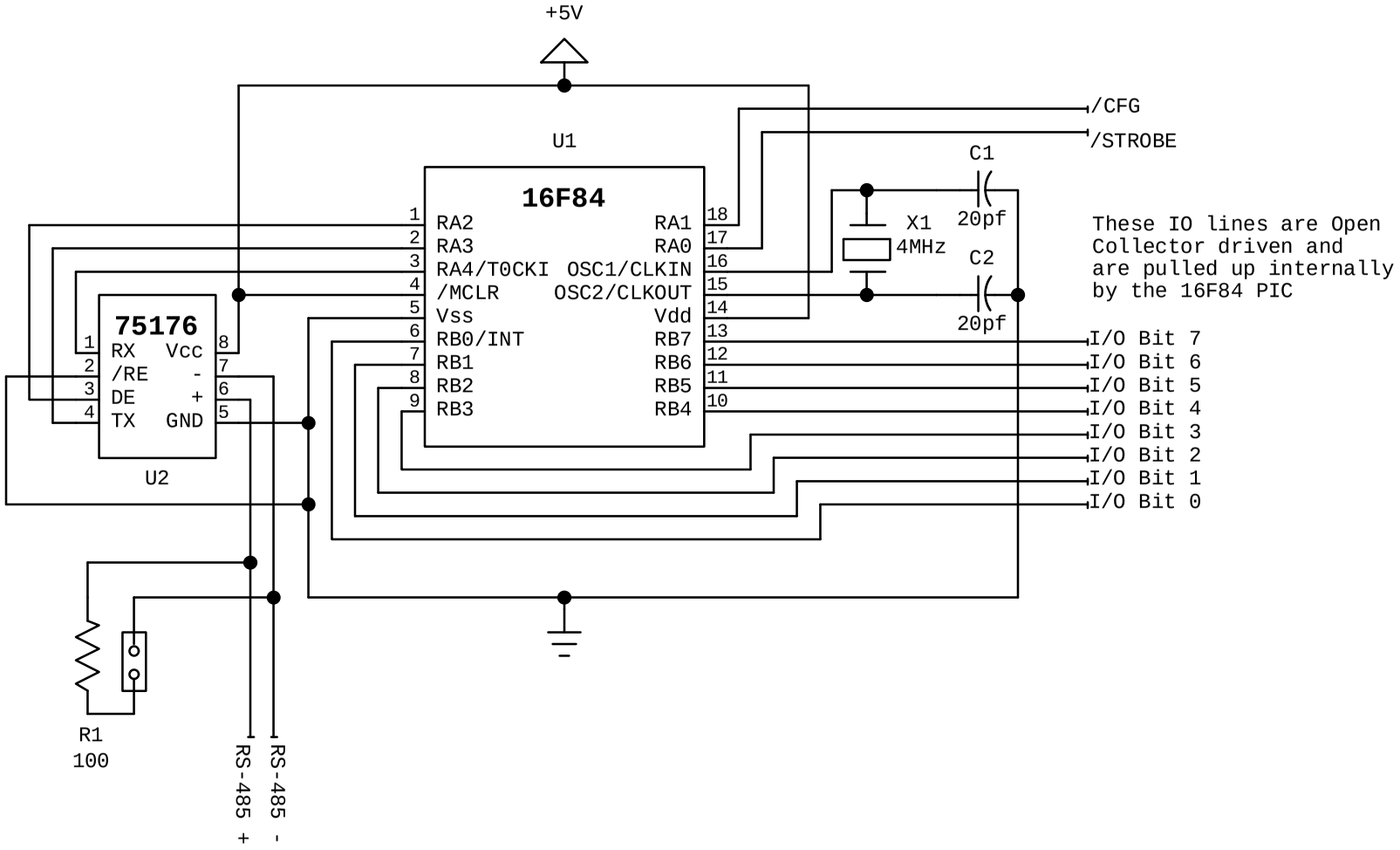
If you purchased a kit from CCC, check the contents to ensure you have all the necessary parts:

Qty	Description
1	18-Pin Mini-DIO IC
1	8-Pin 75176 RS-485 IC
1	100Ω ¼ watt Resistor (brown-black-brown-gold)
1	2 Pin Jumper Block
1	Shorting Block
1	18 Pin IC Socket
1	8-Pin IC Socket
1	4MHz Low Profile Crystal
2	20pf Capacitors (look like green resistors)
1	78L05 100mA Voltage Regulator
1	0.1µF Tantalum Capacitor
1	100µF 16V Electrolytic Capacitor
8	2 Position Terminal Blocks

Here are some hints when constructing your Mini-DIO:

- When building the circuit, mark off each connection on the schematic after it is soldered into place.
- Use ESD protection when handling the ICs! If you do not have a strap, touch something grounded before handling the chips.
- Construct the circuit using the IC sockets, leaving the chips alone. When you are done, check that +5V and GND appear on the proper pins in the IC sockets. Then install the chips.
- Locate the Crystal and 20pf capacitors as close to the OSC pins as possible.
- Install the shorting block to terminate the RS-485 network if necessary (i.e. if this module is on the extreme end of the network run). Only 2 modules can have active terminators!
- The kit includes a 78L05 voltage regulator which can handle up to 100mA. If your Mini-DIO circuit will draw more than 100mA (via high current buffers, opto-isolators, etc.), replace the 78L05 with a normal 1Amp 7805 regulator with a heatsink!

Mini PIC-DIO
PIC Based DIO-Link for HCS Home Automation
(c) 1998 Creative Control Concepts



Chapter

9

PIC-DIO/Mini-DIO I/O Buffering

PIC microcontrollers tend to be pretty robust when it comes to their I/O pins, but they can be damaged if you are not careful. If you want to ensure reliable performance or have a noisy environment, buffer your I/O to protect your PIC/Mini-DIO.

Outputs

- **REMEMBER!** All outputs are open-collector with weak internal pull-ups.
- If you need a high current or voltage output, try using an Allegro driver which can source/sink up to 500mA at up to 48VDC.
- If you connect a relay to an output, make sure you place a 1N4001 diode across the coil to absorb any spikes generated by the coil when the relay de-energizes. If the coil is not 5VDC or draws more than 25mA, use an appropriate driver to drive the coil. You might also look at a solid state relay, which handles a control voltage of 5V. All Electronics has many low current (1-3A) solid state relays available for a good price.

Inputs

- If you expect voltages in between +/- 30VDC, an MC1489 serial receiver connected to a 74LS14 hex buffer makes a great input buffer.
- Other I/O protection can include opto-isolators fed into a 74LS14 schmitt trigger fed to the input or low voltage MOVs.

A great diagram of various I/O protection can be found in Circuit Cellar Ink® Issue #78, Page 52.

Chapter
10

Change History

PIC-DIO

v1.1
Added Network RESET command
Made serial handling more efficient
Changed clock frequency to 3.6864MHz for 0% error @ 9600baud
Switched from an array based buffer to a RAM bank buffer using indirect FSR access
Added watchdog timer and brownout protection
Added Serial Error, RS-485 Traffic, and Parallel Out Status LED's
Changed node address handling so jumpers represent 1's instead of 0's

v1.0
Original Release

Mini-DIO

v1.0
Original Release

Chapter
11

Need Help?

If you get stuck trying to get your system to work, drop us a line at support@cc-concepts.com and we will do our best to help you get your PIC-DIO or Mini-DIO working.

Check out our web site at http://www.cc-concepts.com/ for updates, bug reports, etc.

Here are some useful links for additional info referenced in this manual:

Microchip PICs: http://www.microchip.com

Circuit Cellar® HCS: http://www.cc-concepts.com/products/hcs/

All Electronics: http://www.allcorp.com/