DIO-LINK OPTION

Digital Interface for COMM-LINK & HCS II

Rev. 1.0

***** CAUTION *****

PLEASE READ THE ENTIRE MANUAL ** BEFORE ** ATTEMPTING ASSEMBLY !!
Understand the significance of each component and jumper setting
' BEFORE ' assembly and applying power. CHECK AND RECHECK !
FAILURE TO DO SO WILL PUT YOUR WARRANTY AT RISK !!

**** NOTICE *****

It is very important to understand the
function of R15 ** BEFORE ** you install it.
Please read the entire manual
before using the board.

CIRCUIT CELLAR INC.

4 Park Street ● Vernon, CT 06066

TECHNICAL MANUAL

DIO-LINK OPTION

Digital Interface
for COMM-LINK & HCS II

Technical Manual

Release 1.0
4/28/92

Circuit Cellar Inc.
4 Park St.
Vernon, CT 06066

Conditions of Sale and Product Warranty

CIRCUIT CELLAR, Inc. (CCI) and the Buyer agree to the following terms and conditions of Sale and Purchase:

1. CCI 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 option of CCI, be repaired or replaced.

2. A minimum inspection fee must be prepaid for the repair of units that are no longer under warranty. Contact CCI for information on current minimum charges.

3. NO WARRANTY is extended on USER ASSEMBLED systems or kits. However, assembled kits will be inspected and repaired with charges based on the current minimum one hour charge. CCI retains the right to refuse to repair any USER ASSEMBLED item. This right is at the sole discretion of CCI. However, in the event that the repair charge would exceed a reasonable amount, the user may be consulted for a determination. Repairs on user assembled items must be prepaid. Return authorization must be obtained prior to any return.

4. CCI will not be responsible for the repair or replacement of any unit damaged by user modification, negligence, abuse and/or mishandling, or improper installation.

5. CCI is not responsible to the Buyer for any loss or claim of special or consequential damages.

6. All units returned for repair must first receive prior authorization from CCI. A return authorization number may be obtained by phone or letter. Please retain a record of this number, since most subsequent correspondence will refer to this authorization. Under no circumstances should any product be returned to CCI without prior authorization. CCI will assume no responsibility for unauthorized returns. Returns must be shipped prepaid. Insurance is recommended as losses by a shipping carrier are not the responsibility of CCI. Repaired units will be returned postage and insurance paid.

7. CCI 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.

********** Disclaimer **********

While we have attempted to provide accurate and up to date information in this manual, CIRCUIT CELLAR INC. 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 the products of manufacturers other than ourselves. 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 the individual manufacturers.

****

Conditions of Sale and Product Warranty

CIRCUIT CELLAR, Inc. (CCI) and the Buyer agree to the following terms and conditions of Sale and Purchase:

1. CCI 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 option of CCI, be repaired or replaced.

2. A minimum inspection fee must be prepaid for the repair of units that are no longer under warranty. Contact CCI for information on current minimum charges.

3. NO WARRANTY is extended on USER ASSEMBLED systems or kits. However, assembled kits will be inspected and repaired with charges based on the current minimum one hour charge. CCI retains the right to refuse to repair any USER ASSEMBLED item. This right is at the sole discretion of CCI. However, in the event that the repair charge would exceed a reasonable amount, the user may be consulted for a determination. Repairs on user assembled items must be prepaid. Return authorization must be obtained prior to any return.

4. CCI will not be responsible for the repair or replacement of any unit damaged by user modification, negligence, abuse and/or mishandling, or improper installation.

5. CCI is not responsible to the Buyer for any loss or claim of special or consequential damages.

6. All units returned for repair must first receive prior authorization from CCI. A return authorization number may be obtained by phone or letter. Please retain a record of this number, since most subsequent correspondence will refer to this authorization. Under no circumstances should any product be returned to CCI without prior authorization. CCI will assume no responsibility for unauthorized returns. Returns must be shipped prepaid. Insurance is recommended as losses by a shipping carrier are not the responsibility of CCI. Repaired units will be returned postage and insurance paid.

7. CCI 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.

Table of Contents

COMM-LINK, THE BASIC NETWORK NODE FOR THE HCS II

The HCS II is an expandable, network-based, intelligent-node, industrial-oriented supervisory control system which, in its minimal configuration, performs quite suitably as a home control system. HCS II incorporates direct digital inputs and outputs, direct analog inputs and outputs, real time or Boolean decision event triggering, X-10 transmission or reception, infrared remote control transmission and reception, remote displays as well as a master console, and it has the capability to perform as a complete badge monitoring and personnel tracking system.

The HCS II system architecture consists of a central supervisory controller connected to up to 31 other functional modules (called links) via an RS-485 serial network. The system controller and the links can operate independently and do not need the system controller or other links to function. This allows easy testing or incorporation as intelligent subsystems in other control equipment. The subsystem links all share a common 8031 controller board (generically called a COMM-LINK) with the I/O customized for each application.

The supervisory controller is quite literally the brains of the system. While every Link has its own processor, they each rely upon the supervisory controller for timing, control commands, and overall system coordination.

The supervisory controller is a 9.218 MHz 64180-based single board computer. It has two serial ports, up to 96K bytes of memory (32K bytes are battery-backed), an eight channel 8-bit or 10-bit A/D, a real time clock-calendar, and 24-bits of parallel I/O (an additional 48-bits of parallel I/O can be added if required). These parallel I/O bits can be further conditioned by externally connecting them to optoisolators, relays, or drivers. Depending upon your application you may find that the supervisory controller is all you need.

One of the serial ports performs as a dedicated 9600 bps serial RS-485 connection to the various network links. The wire from this port can be up to 4000 feet long (as opposed to 50 feet for RS-232). The second port connects (RS-232) to a PC/AT which functions as the master console. Programs running on the master console allow the user to enter and store control sequences which will be subsequently compiled and downloaded to the supervisory controller's non-volatile memory. The master console also acts as an on-line real time display of all control activity in the system. When these downloading or display functions are not required, the master console can be turned off to reduce system power consumption.

Besides the Supervisory Controller, the HCS II system consists of a multitude (none to 31) of specialized-function network nodes which facilitate remote data acquisition, closed-loop control, and display. With the exception of the ADIO-LINK, all comm-links consist of the same generic 8031 computer with only the external parallel I/O circuitry and operating software being different. The generic COMM-LINK board is configured with minimal components. It accommodates up to 64K bytes of either RAM and/or EPROM memory. The single serial port is configured to operate both RS-232 and RS-485. When connected to the Supervisory Controller in normal operation, the connection is via a twisted pair to the RS-485. Additional line balancing and termination resistors are included. According to the driver chip manufacturers, this line can be 4000 feet long. In normal operation only RS-485 is used and the MAX232 (U7) can be removed to reduce power.

The RS-232 is reserved for local non-system direct testing or use (of course, you could also operate it stand-alone with RS-485). Each link has a unique command set and, for multiple units of the same type, a unique address. By connecting the link to the serial port of a terminal or any PC running a terminal emulation program, you can directly command the comm-link to do any activity which it would normally do for the Supervisory Controller.

The intention is to promote connecting the individual links via a 4-wire cable, rather than just a 2-wire twisted pair, when local power is unavailable. Two wires supply RS-485 while the other pair supplies +12V and ground (actually 9-12V is fine. Higher voltages require a larger heatsink on the regulator). Of course, the board can be operated on +5V.

DIO-LINK - DIGITAL INPUT/OUTPUT INTERFACE

Like other 'LINK' modules, up to 8 DIO-LINKs can be accommodated within a system (DIO-LINKs beyond DIO0 and DIO1 require a battery backed RAM to preserve the address while power to the link is off). It's hardware is the COMM-LINK circuit again with the "DIO" referring to the 8-bits of digital I/O on PORT1 of the 8031. The 8 lines are "bit-programmable": each line can be defined as an input or as an output. Since these are direct processor bits, however, it is suggested that the prototyping area on the link board be used to add custom signal conditioning and/or protection as required.

The typical use for this network link is to provide remote data collection (what is the current logic setting of each relays at the top of the elevator shaft) or closing the control loop without running sensor wires all the way back to the supervisory controller.

DIO Controller

DIO-LINK connections

J1	Signal	Function
2	P1.7	I/O bit 7
4	P1.6	I/O bit 6
6	P1.5	I/O bit 5
8	P1.4	I/O bit 4
10	P1.3	I/O bit 3
12	P1.2	I/O bit 2
14	P1.1	I/O bit 1
16	P1.0	I/O bit 0
1	P3.3(INT1)	INPUT -- OPEN=net addr 0, GND=net addr 1
3	P3.2(INT0)	0.5 Hz heartbeat, GND for interactive mode
5	P3.5(T1)	OUTPUT -- LOW pulse on byte output
7	P3.4(T0)	controls RS-485 driver (LED ON = disabled)

DIO-LINK default jumper setup

JP1	EPROM	U2 is a 32K EPROM
JP2	RAM	U3 is an 8K RAM
JP3	installed	enable external program memory
JP4	open	close to reset board
JP5	RS-485 termination	as needed
JP6	not used	
JP7	relocated	now JP7B (close for interactive mode)
JP8	not used	

To Get Started

Communications is 9600 b/s 8N1. (Remove 75176 transceiver chip, install MAX-232.)

Connect PC to RS-232 header as usual.
Verify commands to satisfy yourself that it works...
Prefix all commands with "! DIOx " (note spaces!)
x must match the net address set by P3.3
the default address is zero, so use "! DIO0 "
There is no command echo in this mode

If you have trouble, ground P3.2 (JP7B/*INT0) and reset the CPU to enter "interactive" mode:
Descriptive banner appears and prefix is not used

For Network Use Install 75176 transceiver (remove MAX-232.)
Connect PC through RS-485 converter.
Your software must enable/disable RTS correctly!
Make sure P3.2 (JP7B/*INT0) is NOT grounded when turning power ON.

Checksums

If you must be absolutely certain that there are no errors in either the commands or responses, you can add a checksum to each command. The firmware will respond only to commands with valid checksums, so your controller code must resend the command if it does not produce the expected response within a specified timeout. The command format is:

```
#xx addr command
```

where:

#	identifies a command with a checksum
xx	checksum byte in hex, described below
blank	a blank (ASCII 32, 20 hex) is required
addr	target node's network address
blank	a blank (ASCII 32, 20 hex) is required
command	command text
<CR>	carriage return (ASCII 13, 0D hex)

The node's response will look like this:

```
$xx addr command
```

where:

$	identifies node response (same as ! response)
xx	checksum byte in hex, described below
blank	a blank (ASCII 32, 20 hex)
addr	target node's network address
blank	a blank (ASCII 32, 20 hex)
response	the response to the command
<CR>	carriage return (ASCII 13, 0D hex)

Note that ordinary and checksummed responses both start with a "$" character. You can distinguish between the two by examining the second character: if it's a blank, there is no checksum.

Compute a checksum using this algorithm:

- assemble the line with ASCII zeros in the checksum hexits
#00 X10 QN01

- add up all the character values, including the zeros

#	23	(all values in hex here)
0	30	
0	30	
blank	20	
X	58	
1	31	
0	30	
blank	20	
Q	51	
N	4E	
0	30	
1	31	
total	7C	(only the low byte is used)

- find the 2's complement of the checksum byte

-(7C) = 84

- convert the byte to ASCII hexits and replace the zeros
#84 X10 QN01

To verify the checksum for an incoming line,

- assemble the entire line
$E7 X10 N01=XX<CR>

- discard the trailing <CR> (and <LF> if present)
$E7 X10 N01=XX

- convert the checksum chars to a single binary byte
$E7 X10 N01=XX (checksum is E7 hex)

- replace the checksum chars with two ASCII zeros (30 hex)
$00 X10 N01=XX (ASCII 48 chars, not binary zeros!)

- add up all the character values, including the zeros

$	24	(all values in hex here)
0	30	
0	30	
blank	20	
X	58	
1	31	
0	30	
blank	20	
N	4E	
0	30	
1	31	
=	3D	
X	58	
X	58	
total	19	(only the low byte is used)

- add the checksum byte
19 + E7 = 00

If the result is zero, the line is (quite likely) valid.

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.

Controller commands

A=str **Set network address to string (default DIO0)**
(16 chars max, capitalized, blanks discarded)
(last character's LSB replaced with bit from P3.3)
(Use DALLAS 1213C smartsocket or DS1225 battery backed RAM in U3 to retain configuration beyond DIO0 and DIO1)
(HCSII requires the address name = DI00 through DI07)
A=DIO0 will be either DIO0 or DIO1
A=DIO4 will be either DIO4 or DIO5
A=DIOA will be either DIOA or DIOB)

D **Display program status (debugging use)**

E **Show and clear error flags (debugging use)**

Ln **Set logging mode (bit mapped)**
L report current mode
L0 disable (default)
L1 shows DP processing
L32 shows device decoding

Nn **Set network/interactive mode**
N report current mode
N0 sets interactive mode
N1 sets network mode (no error messages) (default)
N2 sets network mode with command echo and err msgs
N3 same as N1, with command acknowledgement

Q **Query inputs**
Q DP returns whole port value (byte)
Q DP.3 returns value for bit 3 only (bit)

RESET **Perform power-on reset (must be completely spelled out)**

Rn **Set ouput delay for the T command to n*100uS**
R0 no delay 1mS
R350 (350*100uS)=35mS/character (default)

S **Set outputs**
S DP=34 set port to hex 34 (byte)
S DP.3=0 set port bit 3 to 0 (bit)

T **Send text string to output**
T DP=string

Notes

All data values are in HEX notation.

You can read the digital port at any time, but a 0 (LOW) output bit will override any external data. Therefore, make sure that all the output bits are 1 (HIGH) for any bits you want to read from an external device. For example:

```
DIO0 >S DP=0F       bits 0-3 are HIGH for input
DIO0 >Q DP               read inputs
DP=0F                    if all inputs are HIGH
DIO0 >Q DP               read inputs
DP=00                    if all inputs are LOW
```

The firmware pulses bit P3.5 LOW for about 250 µs after each S command. This bit can serve as a strobe to tell an external device that new data is present. The bit is also strobed when you change a bit using the S DP.b=xx commands.

The firmware strobes P3.5 LOW for each T command text string character. The peak data rate is about 850 µs/char (1200 char/sec), with the strobe pulse lasting about 250 µs for each character. Note that this is only slightly slower than the RS-485 interface's 960 char/sec data rate.

Decoding and executing each T command requires about 30 ms, so the overall average data rate for 40 character messages is about 600 char/sec. The firmware incorporates a 250 byte buffer for incoming network message, but does not buffer the decoded text for the digital port. You should avoid sending large numbers of long messages to the DIO module to avoid overrunning the input buffer.

The T command interprets several C-style escape sequences to simplify sending "unprintable" characters in the ASCII string. All sequences start with a backslash (ASCII 5C)...

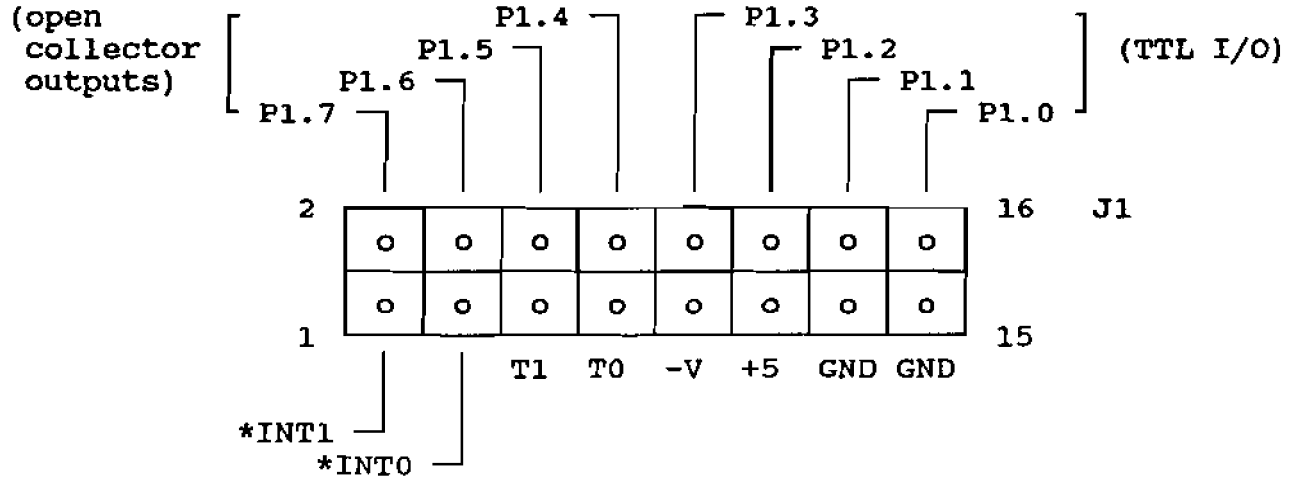
\cn	Control character n (\cZ = Ctrl-Z)
\e	Escape character, ASCII 27
\f	Form feed character, ASCII 12
\n	New line (linefeed and carriage return)
\r	Carriage return (to current line, leftmost column)
\xnn	Send hex char nn (Must have two digits)
\\	Single backslash

A backslash followed by any other character simply sends that character, so the backslash vanishes.

The \x sequence is intended to transmit any character including those above 0x7F.

The \c sequence is a different way to generate control characters that might otherwise use \x. For example, \x01 and \cA are equivalent. Only the low-order five bits of the character are used, so \cA and \ca are equivalent.

I/O CONNECTOR



The DIO-LINK makes use of all of PORT1 as input/outputs, so the special functions of address input, mode input , and heartbeat output are NOT on the PORT1 pins like other COMM boards. Special connections are necessary to easily implement all of these functions.

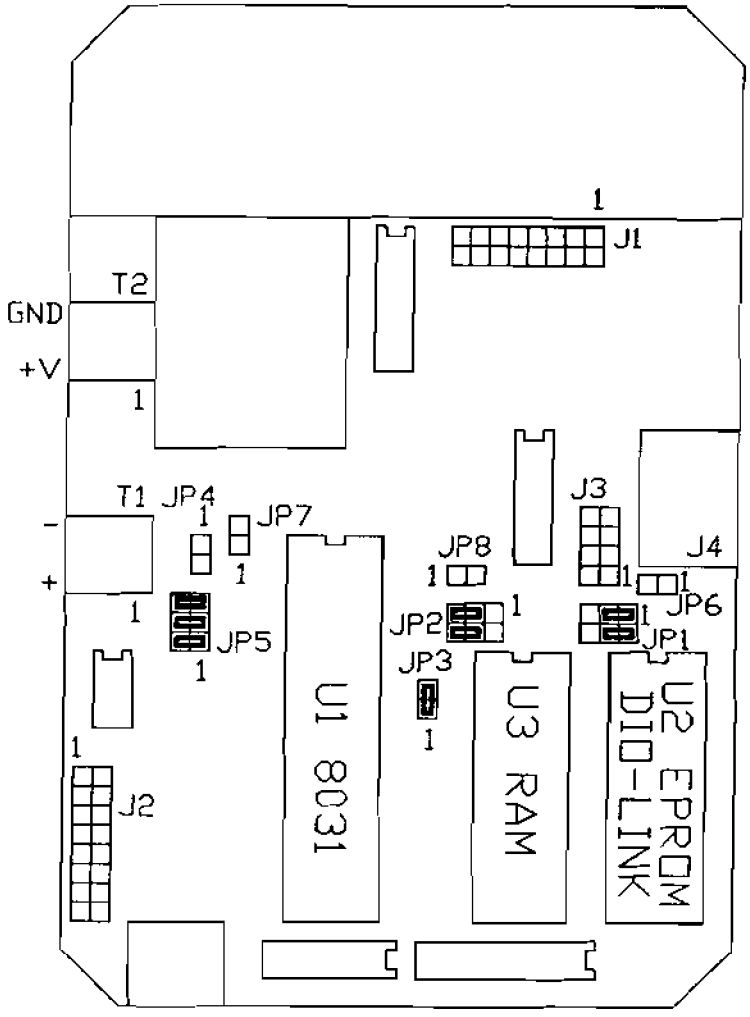
See the special instructions for DIO-LINK kit assembly to add the appropriate parts to the prototype area. Most are optional and are not required.

1. Relocate R10, LED5, and JP7 (now JP7B) so the heartbeat LED runs from U9 pin 10. JP7B pin 1 and U9 pin 11 comes from P3.2 (*INT0). JP7B pin 2 to ground which sets interactive mode when enabled (remove the jumper after reset to allow the heartbeat to function).

2. A two pin jumper connects to P3.3 (*INT1) and ground when enabled, to set the address bit if needed.

3. If bits P1.4-7 are used as outputs, they should be pulled up to VCC with a resister (1K or so). If they are used as inputs, use the following connections for P1.4-7.

P1.4	U9 pin 9
P1.5	U9 pin 5
P1.6	U9 pin 3
P1.7	U9 pin 1



DEFAULT JUMPER CONFIGURATION

INTERACTIVE STANDALONE MODE

Install JP7B the interactive mode jumper and follow the directions for RS232 use under the section "GETTING STARTED - DIO-LINK MONITOR".

Upon reset you should see the sign-on banner similar to:

```
DIO-LINK Networked Digital I/O, Version 1.0e
  Copyright 1992 Circuit Cellar, Inc.
Initializing device information
 ...DP:  ready
DIO0 >
```

Remove the JP7 jumper to enable the heartbeat LED.
You can change the address (or name) of the DIO-LINK module if you would like by using the 'A' command. **(NOTE: The XPRESS programming language require the name DIO0-DIO7 when this module is used with the HCSII supervisory controller.)**

```
A=GARAGE0
GARAGE0 >
```

Bit P3.3 (*INT1) is an address bit. The LSBit of the address is replaced with this bit. Enabling (grounding) the bit would make the address:

```
GARAGE1 >
```

Unless you have installed a non-volatile RAM in U3 this address will revert make to 'DIO0' if the power is lost.

Some commands are used strickly for developmental debuging and offer no practical function.

```
DIO0 >D
Net address [DIO0 ].  Stack used 45/7FH.
T0Ticks 79

DIO0 >E
Error flags 00.  Checksum errors 0.

DIO0 >
```

Logging, which defaults to off, can be used as feedback indicating that a command has been received and recognized.

```
DIO0 >Q DP
DP=FF
DIO0 >L32
```

Turn on the logging and repeat the Query Digital Port command.

```
DIO0 >Q DP
[Device DP]
Read FF from address 0001
DP=FF
DIO0 >
```

Testing individual bits are just as easy.

```
DIO0 >Q DP.3
[Device DP, bit ID = 3]
Read FF from address 0001
DP.3=1
DIO0 >
```

The complete byte is read but only the bit of interest is returned. The bit is in the form off binary '1' or '0'.

The port or any single bit can be written to using the 'S' command.

```
DIO0 >S DP.3=0
[Device DP, bit ID = 3]
Read FF from address 0001
Write F7 to address 0001
DIO0 >
```

Now send a string using the 'T' command.

```
DIO0 >T DP=Test!
[Device DP]
Write 54 to address 0001
Write 65 to address 0001
Write 73 to address 0001
Write 74 to address 0001
Write 21 to address 0001
DIO0 >
```

This is interpreted as a print command. Each character is written to the output port P1.0-7, then P3.5 (T1) is pulsed low as a data ready strobe. The routine is continued until the complete string has been sent.

USING DIO-LINK WITH HCSII AND XPRESS PROGRAMMING LANGUAGE

In order to maintain ease of configuration and setup, we have preassigned port numbers to fixed hardware arrangements. Ports with fixed assignments include all direct inputs and outputs, netbits, A/D converters, and D/A converters. There is no way to reassign port numbers nor is there any way to reassign input/output assignments. All ports are fixed as either input or output. The current state of all digital outputs may be tested within event equations.

NETBIT

Netbits are digital I/O bits found on the DIO-LINK module. Up to eight DIO-LINK modules may be used on the same network for digital I/O. The 'Netbit' XPRESS keyword is used to both set and read netbits.

Each DIO-LINK has 8 netbits which may be either inputs or outputs (four outputs have open collector drivers). To use a bit as an output simply set the bit to either 1 or 0. To use the bit as an input, it must be set to a 1 on reset, then the bit may be read at any time and the current state of the input will be returned.

DIO-LINK ADDRESS	16 inputs/outputs
DIO0	Netbit(0-7)
DIO1	Netbit(8-15)
DIO2	Netbit(16-23)
DIO3	Netbit(24-31)
DIO4	Netbit(32-39)
DIO5	Netbit(40-47)
DIO6	Netbit(48-55)
DIO7	Netbit(56-63)

LPT

The DIO-LINK board can alternately be used as a parallel output device having a 'DATA GOOD' strobe to indicate the presence of valid data. This may be used for event logging using a parallel printer, voice output using a text to speech converter, or any other paralle device which requires a strobed output. No acknowledgement signals are used as the DIO-LINK is asynchronous only. T1 provides a 250uS strobe (the character rate can be adjusted for slow print devices by using the Rn command). The HCSII XPRESS keyword is LPTx="string" where 'x' is the same number as the DIO-LINK module (LPT0 is sent to DIO0).

using DIO-LINK as a strobed output port:

```
DIO0 = LPT(0)
DIO1 = LPT(1)
DIO2 = LPT(2)
DIO3 = LPT(3)
DIO4 = LPT(4)
DIO5 = LPT(5)
DIO6 = LPT(6)
DIO7 = LPT(7)
```

Kit Specific Instructions for the HSC II - DIO-LINK

These are additional specific instructions for adding the DIO-LINK hardware to the COMM-LINK board. Please refer to the COMM-LINK manual for the basic assembly instructions.

The DIO-LINK circuit board has two areas of special concern, the power input section and the DIO open collector driver section. The power for DIO-LINK can be a regulated 5.0 volts DC or an unregulated 9-12 volt DC. The I/O configuration must be defined now so you can make installed the appropriate I/O components on the PC board.

PROVIDING A REGULATED 5.0 VOLT INPUT

If the board will be used with a regulated 5.0 volt DC input, the components Q1, D4, C8, and associated heat sink hardware can be eliminated. Simply insert a wire jumper into location R15 (zero ohm resistor or wire jumper) and attach the regulated 5.0 volts through T2. **NOTE: INSERT R15 ONLY WHEN USING A REGULATED 5 VOLT INPUT, APPLYING MORE THAN 5 VOLTS WITH R15 INSTALLED WILL DAMAGE YOUR BOARD.**

PROVIDING AN UNREGULATED 9-12 VOLT INPUT

If the board will be used with an unregulated 9-12 volt DC input, install Q1, D4, C8, and the associated heat sink hardware during the standard kit assembly procedure. **R15 is not used.** The unregulated 9-12 volt DC input is attached through T2.

I/O CONFIGURATION

VISUAL HEARTBEAT (not necessary for DIO-LINK operation)

The COMM-LINK heartbeat LED5 and R18 normally comes from P1.7. The DIO-LINK board uses all 8 bits of PORT1 for I/O. The heartbeat signal on the DIO-LINK has been moved to P3.2 (*INT0). If you wish to see this heartbeat, install LED5 and R18 in the prototype area, not in the positions designated by the silkscreen. Using any small gage insulated wire (i.e. wire wrap wire) connect one end of the resistor to VCC and the other end to the LED's anode. (NOTE: the lead closest to the flattened side on the LED's rim is the anode) Connect the cathode to an used section of U9 pin 10, the 7407. Finally, connect P3.2 to the input of that same unused gate U9, pin 11.

INTERACTIVE MODE ENABLE JUMPER (only necessary when DIO-LINK must start-up in interactive mode)

The COMM-LINK mode JP7 normally comes into P1.0. The DIO-LINK board uses all 8 bits of PORT1 for I/O. If you wish to use the interactive mode, install the jumper (JP7) in the prototype area, not in the position designated by the silkscreen. Using any small gage insulated wire (i.e. wire wrap wire) connect one pin of the jumper to ground and the other pin to P3.2. (NOTE: Once the board is powered up, the shorting jumper placed on JP7 (which puts the DIO-LINK into interactive mode) must be removed for the heartbeat signal to reach the LED installed in the previous step, since P3.2 performs a dual function.)

ADDRESS BIT INPUT (only necessary when using more than one DIO-LINK with the HCSII)

The COMM-LINK address inputs normally comes into PORT1. The DIO-LINK board uses all 8 bits of PORT1 for I/O. If you wish to use the DIO-LINK address input bit, install a two pin jumper in the prototype area. Using any small gage insulated wire (i.e. wire wrap wire) connect one pin of the jumper to ground and the other pin to P3.3 (*INT1).

PRINTER PORT or 8 TTL OUTPUTS

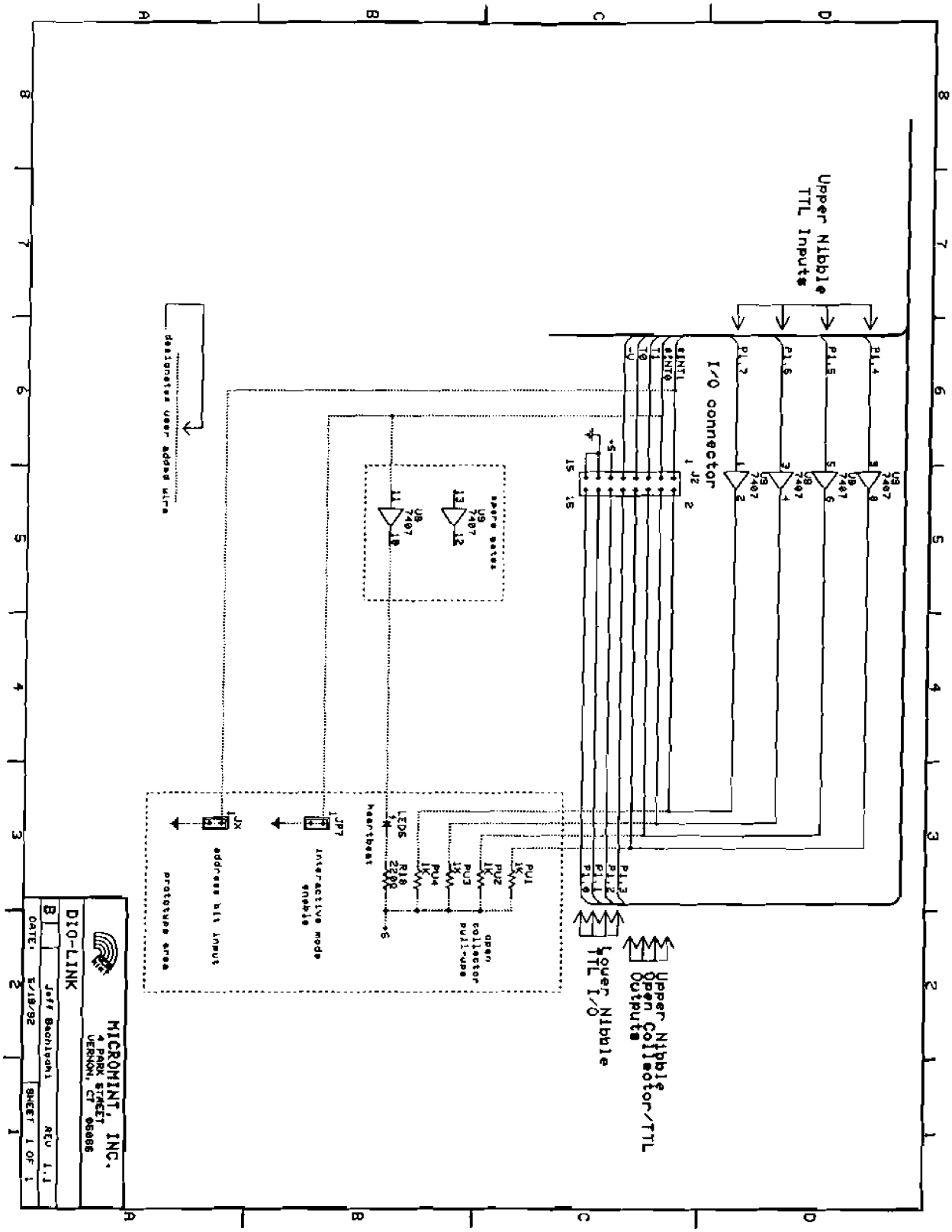
The high nibble, P1.4-7, has open collector outputs. Connect four pull-up resistors from U9 pins 8, 6, 4, and 2 to VCC. Use the outputs on J2.

8 TTL INPUTS

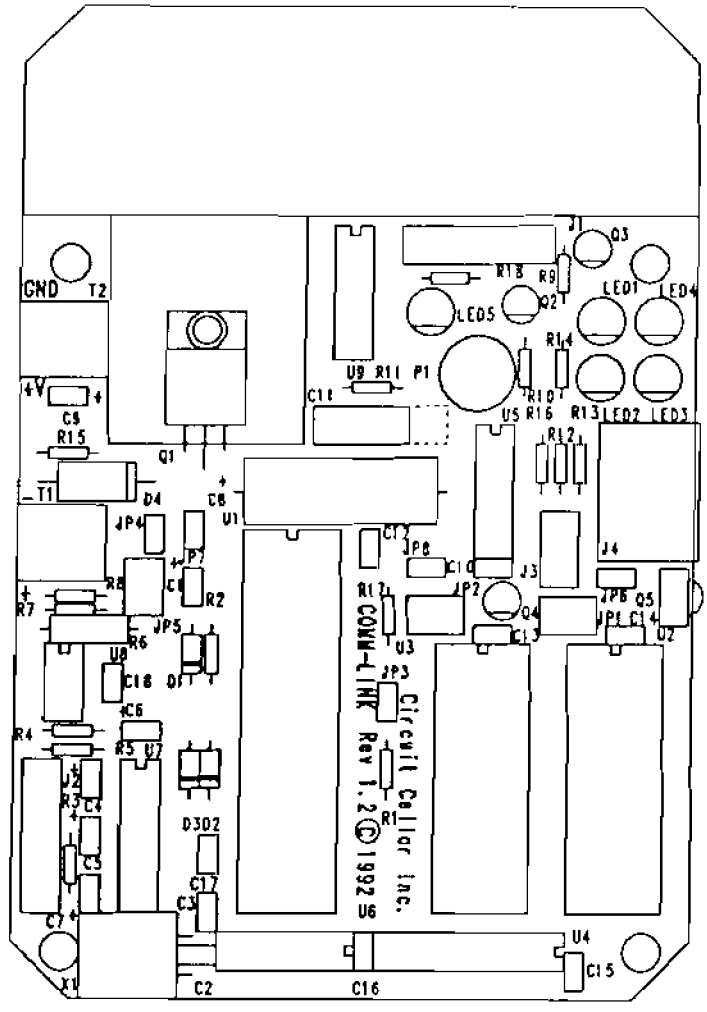
The high nibble, P1.4-7, has open collector outputs. Apply the upper nibble inputs NOT to J2, but to the P1.4-7 signals on U9 or U1.

4 TTL INPUTS and 4 OPEN COLLECTOR OUTPUTS

Use the lower nibble, P1.0-3, as TTL inputs to J2. Use the upper nibble, P1.4-7, as open collector outputs from J2.



DIO-LINK SCHEMATICS (1 of 1)



Silkscreen for the DIO-LINK

PARTS LIST for the DIO-LINK

DESIGNATION	PART#	DESCRIPTION
Printed Circuit Assembly		
COMM-LINK		COMM-LINK assembly
Resistors		
R18	220Ω	1/4 watt, 5%, (red-red-brn) (from COMM-LINK kit, relocate position)
PULL-UP1-4	1K	1/4 watt, 5%, (brn-blk-red) (optional for pull-ups)
CONNECTORS		
J2	2x8	Square Pin Header (optional for external connection)
JX	1x2	Square Pin Header (optional for address enable bit)
Miscellaneous		
U2	27C256	DIO-LINK Firmware in EPROM
LED5	TIL 220	Visible LED (from COMM-LINK kit, relocate position)