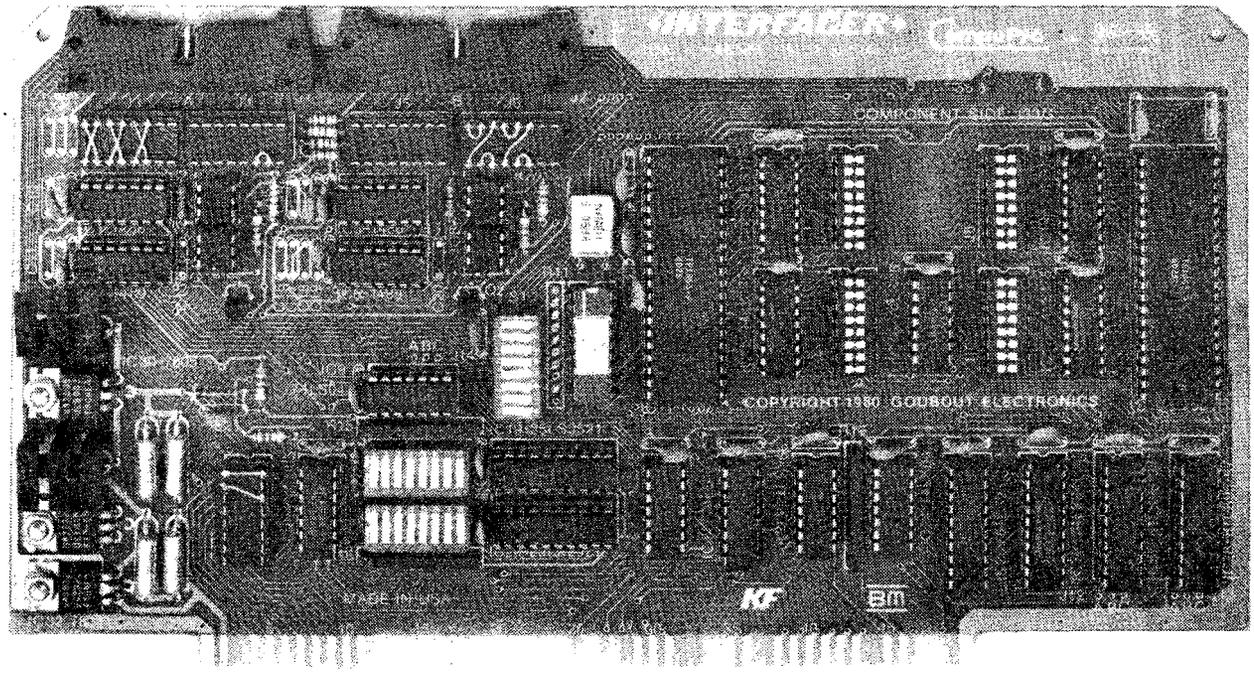
INTERFACER 1™

TECHNICAL MANUAL



IEEE 696 / S-100

DUAL SERIAL RS 232C I/O
with full handshake

133H

mac

CompuPro®

COMPUPRO division, GODBOUT ELECTRONICS

7/82

Table of contents

ABOUT INTERFACER

Congratulations on your decision to purchase the INTERFACER, a dual channel RS-232 serial I/O board designed specifically for full electrical and mechanical compatibility with the IEEE 696/S-100 standard. The S-100 bus is currently one of the most popular in the industry and by far the most prolific. We believe that this board with the rest of the S-100 portion of the CompuPro family, is one of the best boards available for that bus.

Features, such as reliable hardware UARTs, independent channel port selection, crystal controlled dual baud rate generator, full hardware/software programmability, and a convenient interrupt structure allow for maximum flexibility at a reasonable price.

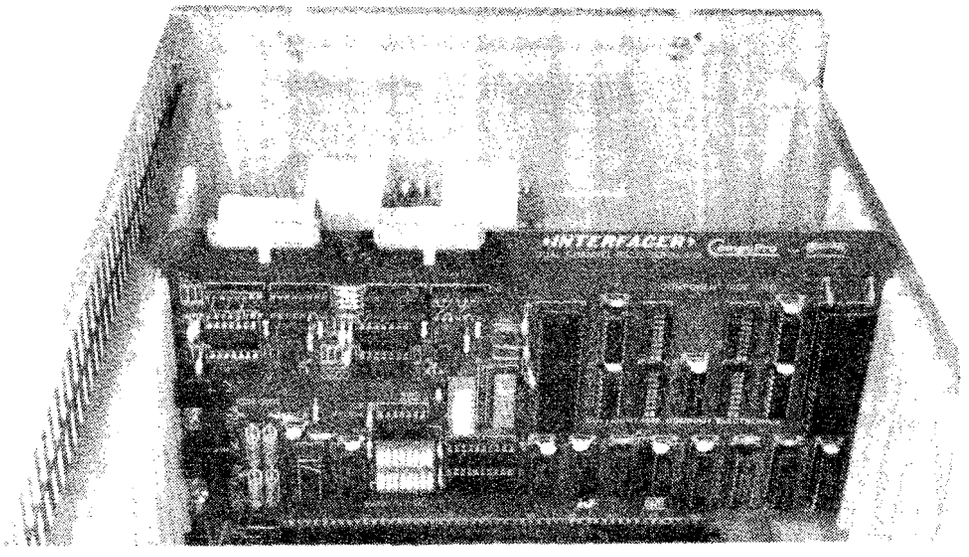
Thank you for choosing a CompuPro product....welcome to the club.

TECHNICAL OVERVIEW

This board incorporates reliable LSI technology UART chips to perform the basic serial-to-parallel and parallel-to-serial conversions necessary for this S-100 dual channel I/O board. By using these UARTs, the CPU is freed of the time consuming drudgery of performing serial I/O operations by software techniques. Because of this, I/O operations are far more reliable and easier to implement.

In addition to the UARTs, several other features are included to make this board extremely flexible and easy to use. These features include an on board crystal controlled dual baud rate generator for reliable baud rates independent of your CPU clock speed, conversion to TTL, current loop and RS-232 levels for interfacing to almost every kind of serial device, and hardware/software programmability for power-on operation and easy parameter modification.

Other features standard to all CompuPro boards include thorough bypassing of all supply lines to suppress transients, on board regulators, and low power Schottky TTL and MOS technology integrated circuits for reliably cool operation. All this and sockets for all ICs go onto a double-sided, solder masked printed circuit board with a complete component layout legend.



Shown above is a typical installation using (2) optional Interfacer cable assemblies. These cables consist of an Ansley 609-25S 25-pin "D" connector and an Ansley 609-2601M 26-pin female transition connector.

Individual cable parts or assembled cables in 18 and 30 inch lengths are available from Godbout Electronics.

I/O ADDRESS ASSIGNMENT

Each Channel on this board is addressable as a single 2 port block anywhere through the 256 port I/O space. In addition, either or both Channels can be disabled and electrically removed from the I/O space. DIP switch S2 addresses Channel "A" and DIP switch S3 addresses Channel "B". As shipped, both Channels will have the data port residing at the port address, and the status port residing at the port address + 1. If desired, this board may be jumpered so that both Channels have their status ports at the selected address and the data ports at the selected address + 1. This may be accomplished by cutting the trace between points "B" and "C" at J14 (on the solder side of the board) and installing a jumper between points "A" and "B". Both Channels are addressed as follows:

SWITCH POSITION	FUNCTION	
1	ADDRESS A1	
2	ADDRESS A2	
3	ADDRESS A3	"ON" = "0"
4	ADDRESS A4	
5	ADDRESS A5	"OFF" = "1"
6	ADDRESS A6	
7	ADDRESS A7	
8	CHANNEL DIS.	- "ON" = DISABLED - "OFF" = ENABLED

Example: To address Channel "A" at the first I/O port address pair 00H and 01H, positions 1 through 7 of switch S2 would be ON and position 8 would be OFF so that the Channel "A" is enabled. This configuration places the data port at 00H and the status port at 01H - assuming that J14 has not been altered from the factory setting.

CONTROL PORT BIT ASSIGNMENT

Outputs to the CONTROL PORT from the CPU are defined as follows:

DATA BIT	NAME	SIGNAL
D0	RxINT E	Receiver interrupt enable
D1	TxINT E	Transmitter interrupt enable
D2	CD	RS232 CD output
D3	CA	RS232 CA output
D4	TSB	Number of stop bits
D5	NP	No parity
D6	EPS	Even parity select
D7	NBI	Number of bit/character

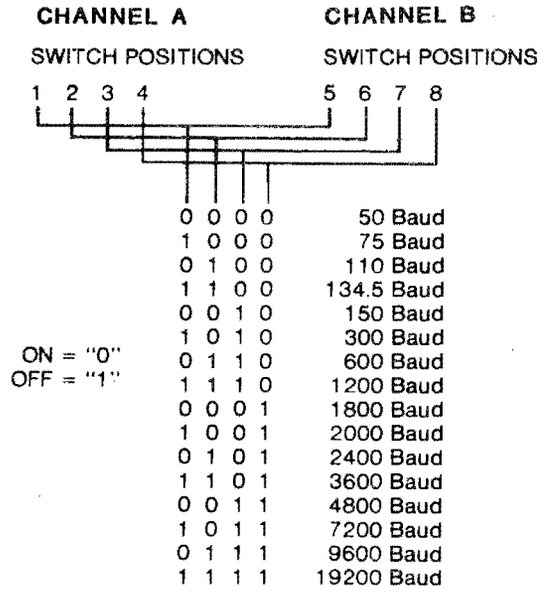
STATUS PORT BIT ASSIGNMENT

Inputs from the STATUS PORT to the CPU ar defined as follows:

DATA BIT	NAME	SIGNAL
D0	TBMT	Transmitter buffer empty
D1	DAV	Data available
D2	OPT	Optional status line
D3	PE	Parity error
D4	OR	Over run
D5	FE	Framing error
D6	CC	RS232 CC input
D7	CB	RS232 CB input

BAUD RATE SELECTION

Dip switch S1 is used to select the baud rate for both Channels A and B. Switch positions 1 - 4 set the baud rate for Channel A and positions 5 - 8 set the baud rate for Channel B as shown below.

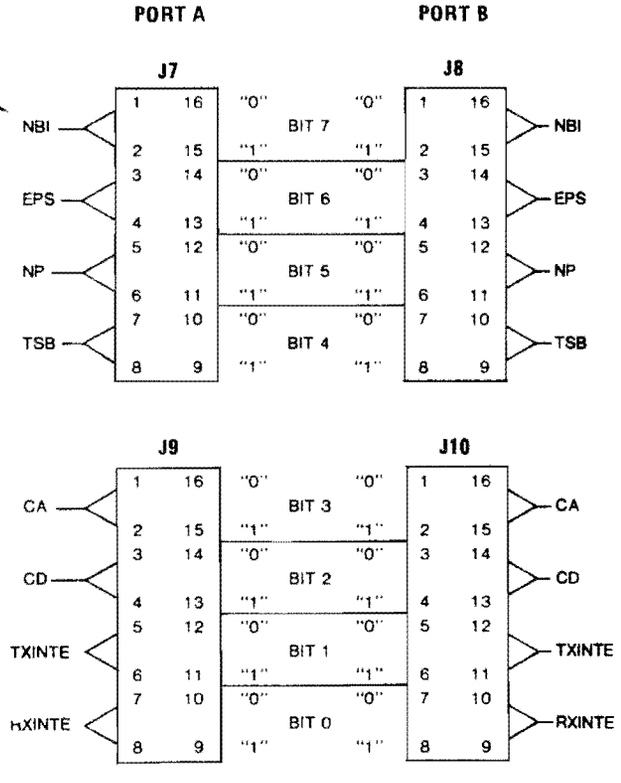


EXAMPLE: To set Channel "A" to 9600 baud, DIP switch S1 would have position 1 ON and positions 2, 3, and 4 would be OFF. To set Channel "B" to 110 baud, DIP switch S1 would have positions 5, 7, and 8 ON and position 6 OFF.

UART PROGRAMMING

On power-up the control latches are all reset to a defined state as shown below. If the UART parameters for your system configuration are set properly for the reset state, no software initialization is necessary for proper operation of this board. To set the power-up parameters, the proper logic level for the programming header should be chosen from the table and the corresponding header trace should be left intact while the trace with the opposite level should be cut (an X-ACTO knife works great). For example, if odd parity is desired, the table shows that EPS should be a logic "0" and the header diagram indicates that the trace between pins 3 and 14 should remain while the trace between pins 4 and 13 should be cut. NOTE: never leave both traces to a signal line unbroken as this could lead to heating and possible damage to the 74LS175's.

UART PROGRAMMING

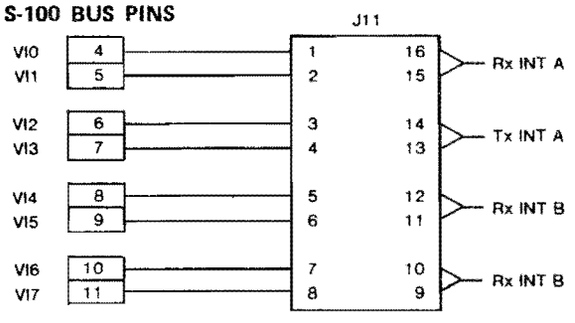


SIGNAL	"0"	"1"
EPS	ODD PARITY	EVEN PARITY
NBI	7 BITS	8 BITS
TSB	1 STOP BIT	2 STOP BITS
NP	PARITY	NO PARITY
TXINTE	DISABLE	ENABLE
RXINTE	DISABLE	ENABLE
CA	"SPACING"	"MARKING"
CD	"SPACING"	"MARKING"

The state of this board may be altered at any time by outputting a new control word to the Control Port. To change the mask under software control, simply output a "1" to the proper data bit of the Control Port corresponding to the signal that you desire to change. This will flip the logic level of the particular signal to the opposite state (i.e. odd parity will become even parity). Outputting a "0" to a bit will return it to the power-up setting.

VECTORED INTERRUPTS

When enabled and jumpered to the appropriate interrupt pin on the S-100 bus, each port has both a transmit and receive interrupt line. The receive interrupt line (RxINT A or B) is driven low when data is available from the UART and the RXINTE line has been enabled as described in the UART PROGRAMMING section. This line resets to a high impedance state after the data has been fetched from the UART. The transmit interrupt line (TxINT A or B) is driven low when the UART is ready to accept a character from the CPU and the TXINTE line has been enabled as described in the UART PROGRAMMING section. This line resets to a high impedance state when the UART transmitter buffer is full and it cannot accept another character. Note that "sINTA", S-100 bus pin 96, is not monitored by this board and is not needed to implement a useful interrupt scheme.



As with the UART parameters, the vectored interrupts are enabled and disabled as described previously. Outputting a "1" to bits 0 or 1 will disable an enabled interrupt or enable a disabled interrupt. Outputting a "0" will return them to their power-up or reset setting.

SERIAL MODE JUMPERS

The INTERFACER board with its unique serial programming jumpers, allows the user to adapt his board to all standard RS-232 pin configurations and to non-standard current loop configurations. In current loop mode, this board may be set to use the on board current source or an external current source. For example, a teletype requires that the on board current source be used, so the serial mode jumpers (J3/J5 and J4/J6) should be set like the example shown on the following page.

In RS-232 mode, these jumpers may be set so that this board operates in a "master" mode where it behaves as the Data Terminal Equipment (DTE), or it may be set so that the board operates in a "slave" mode where it behaves as the Data Communication Equipment (DCE). Since almost all CRT terminals and serial interface printers operate as the "master" or as the Data Terminal Equipment, then the INTERFACER board must operate as the "slave" or Data Communication Equipment. For example, to connect the INTERFACER to a terminal like an ADM 3A or a Hazeltine, the serial mode jumpers (J3/J5 and J4/J6) should be set in "slave" mode as shown on the following page. To connect the INTERFACER to a Modem is a different set-up because Modems are set to operate as "slaves". When connected to a Modem, the INTERFACER should be set in the "master" mode as shown on the following page.

RS-232C CONTROL LINES

The RS-232 control and data lines are defined as shown below.

PIN#	CIRCUIT	DIR.	DESCRIPTION
1	AA		PROTECTIVE GROUND
2	BA	TO DCE	TRANSMITTED DATA
3	BB	TO DTE	RECEIVED DATA
4	CA	TO DCE	REQUEST TO SEND
5	CB	TO DTE	CLEAR TO SEND
6	CC	TO DTE	DATA SET READY
7	AB		SIGNAL GROUND
8	CF	TO DTE	REC'D LINE SIGNAL DET.
20	CD	TO DCE	DATA TERMINAL READY

Four hardwired RS-232 handshaking signals are provided for interfacing to equipment needing these lines as shown below. These lines may be set to power-up either marking or spacing, and their state may be altered by software commands as described in the UART PROGRAMMING Section.

OUTPUT LINES

DATA BIT	RS-232 LINE	DB25 PIN CONNECTION
D2	CD	20 OR 6 *
D3	CA	4 OR 5 *

INPUT LINES

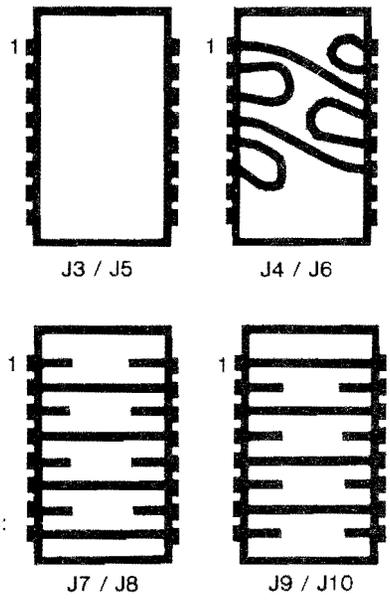
DATA BIT	RS-232 LINE	DB25 PIN CONNECTION
D6	CC	6 OR 20 *
D7	CB	5 OR 4 *

NOTE: Non-starred pin numbers indicate the DB25 pin number when the Serial Mode Jumpers are set for "master" mode. The starred pin numbers indicate the DB25 pin number when the Serial Mode Jumpers are set for "slave" mode.

One optional input line (CF or Rec'd Line Signal Detect) is provided at locations J12 for Channel A and J13 for Channel B. If point "B" is jumpered to point "A", this line will appear on Status Bit D2. By jumpering point "B" to point "C", the "End of Character" output (EOC) from the UART will appear on Status Bit D2.

TYPICAL PROGRAMMING JUMPERS

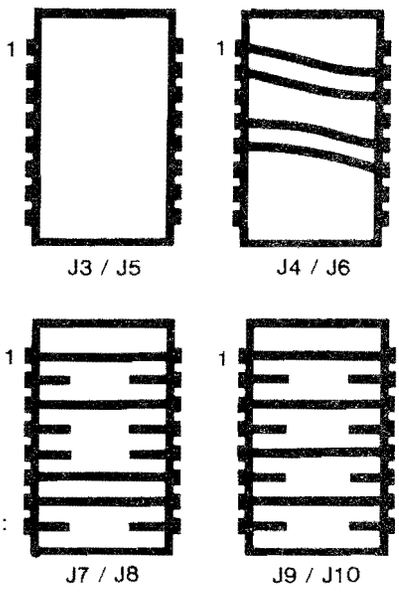
Current loop - on board current sources EXAMPLE: TTY



This configuration yields:

Even parity
8 data bits
2 stop bits
No parity
Interrupts disabled

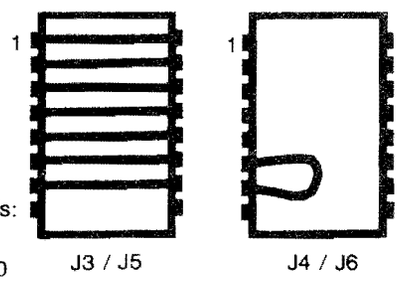
Current loop - external current sources



This configuration yields:

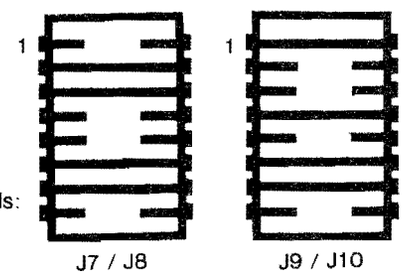
Odd parity
7 data bits
1 stop bit
No parity
Interrupts disabled

RS - 232C - Master mode EXAMPLE: Modem



This configuration yields:

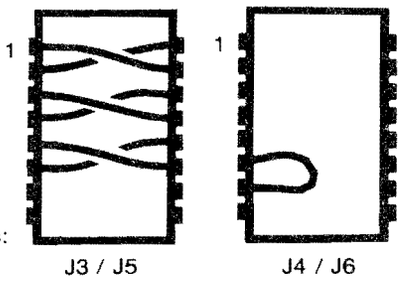
CA "spacing" on pin 4
CD "marking" on pin 20
BA (Tx data) on pin 2
BB (Rx data) on pin 3
CB Status to I/O board on pin 5
CC Status to I/O board on pin 6
CF (Received line signal detector on pin 8)



This configuration yields:

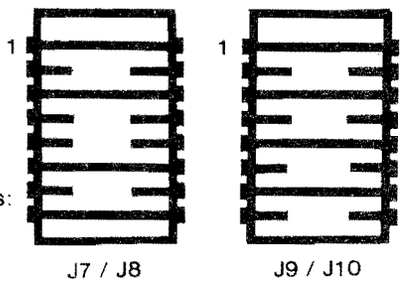
Odd parity
8 data bits
1 stop bit
No parity
Tx Inte enabled
Rx Inte disabled

RS - 232C - Slave mode EXAMPLE: CRT Terminal, printer



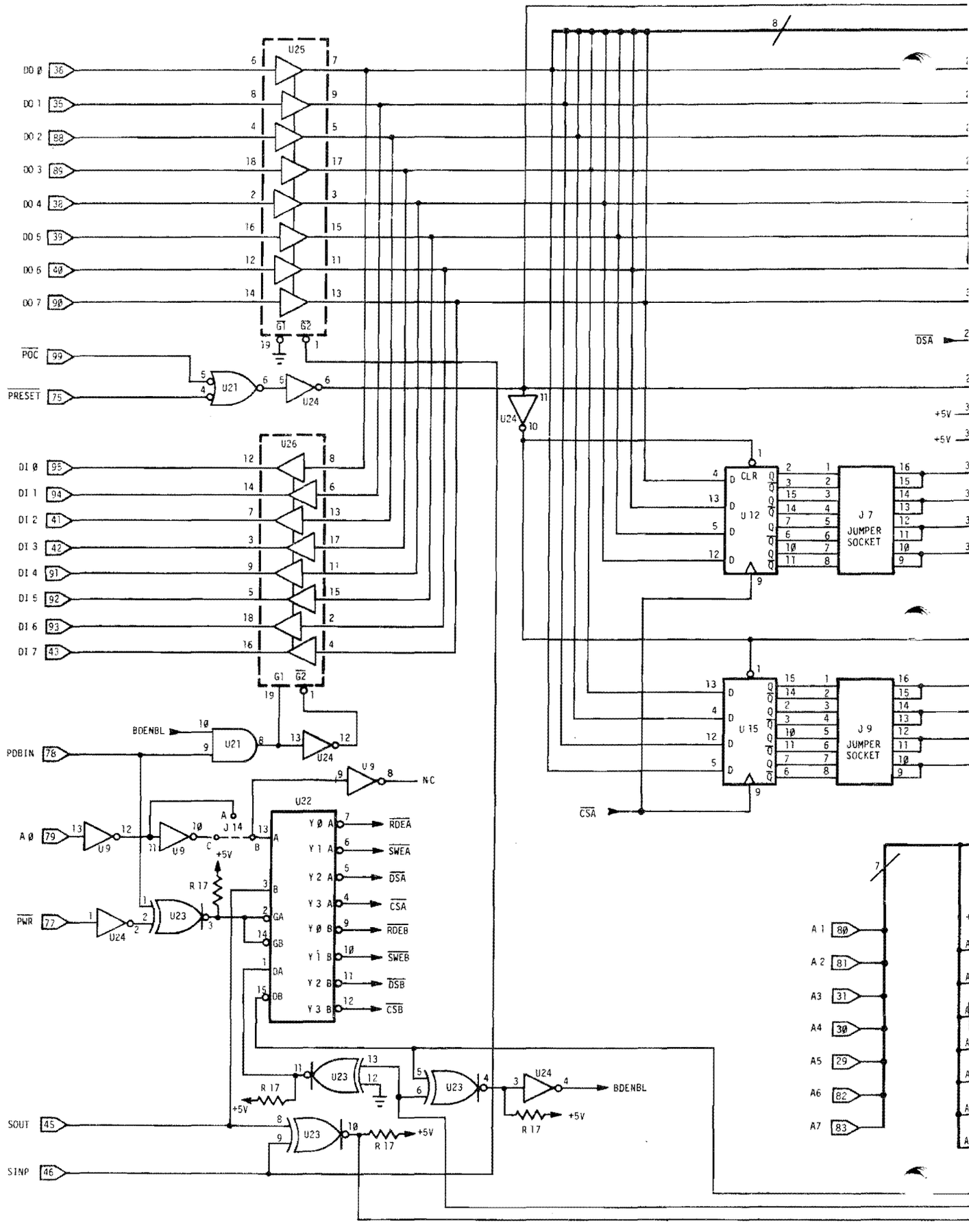
This configuration yields:

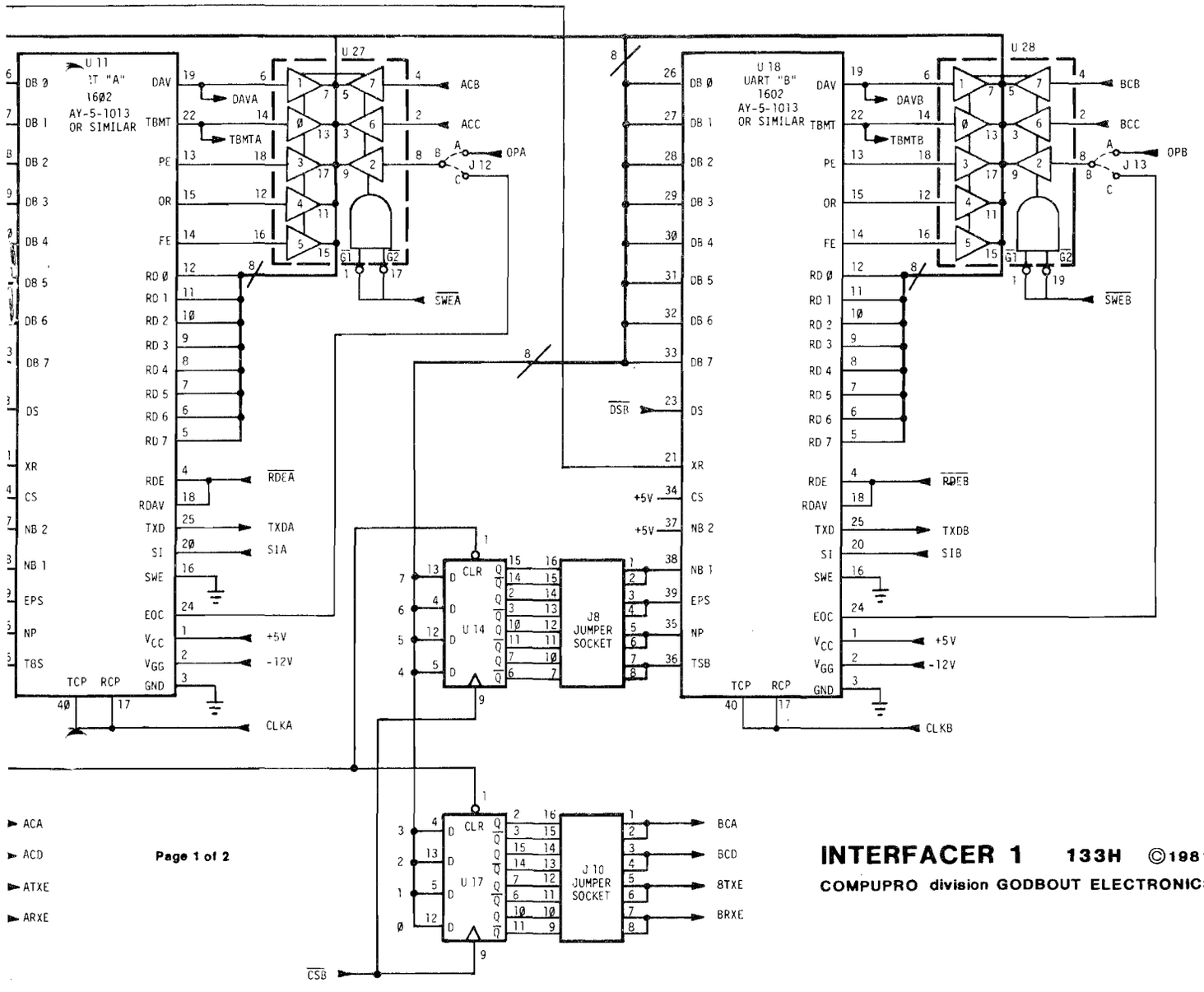
CA "spacing" on pin 5
CD "spacing" on pin 6
BA (Tx data) on pin 3
BB (Rx data) on pin 2
CB Status to I/O board on pin 4
CC Status to I/O board on pin 20



This configuration yields:

Odd parity
7 data bits
2 stop bit
No parity
Interrupts disabled

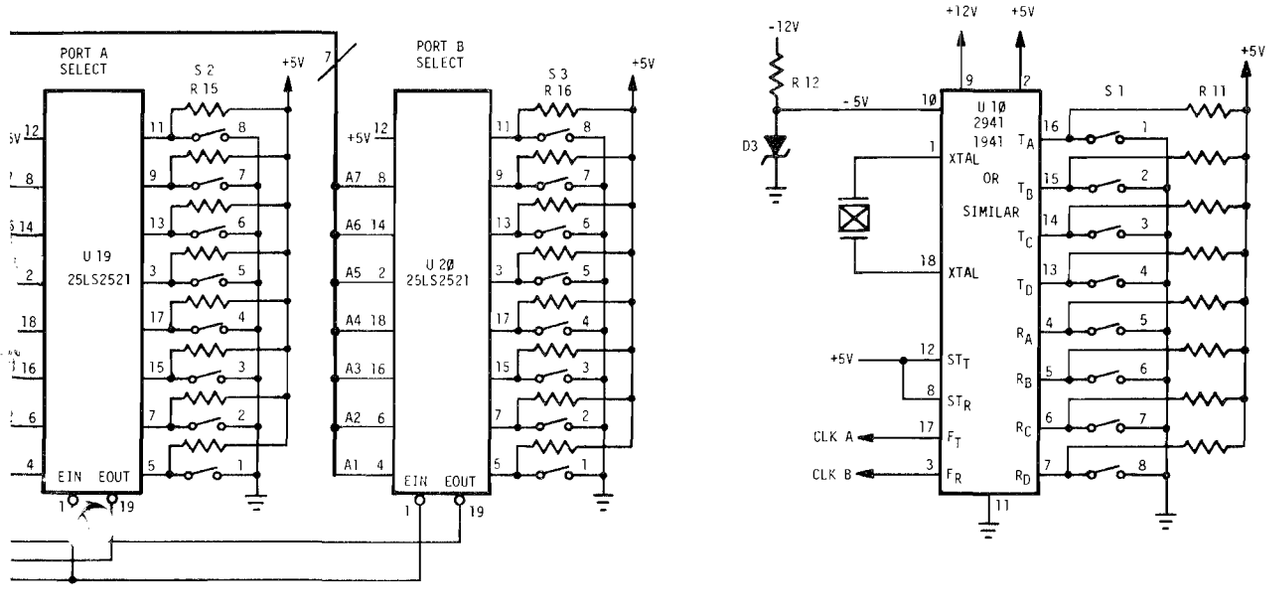

U25
DO Ø
DO 1
DO 2
DO 3
DO 4
DO 5
DO 6
DO 7
G1
G2
POC
PRESET
U21
U24
U26
DI Ø
DI 1
DI 2
DI 3
DI 4
DI 5
DI 6
DI 7
BDENBL
PDBIN
U9
NC
U22
A Ø
J 14
PWR
R 17
+5V
U23
Y Ø A
Y 1 A
Y 2 A
Y 3 A
Y Ø B
Y 1 B
Y 2 B
Y 3 B
RDEA
SWEA
DSA
CSA
RDEB
SWEB
DSB
CSB
SOUT
SINP
BDENBL
U12
U15
J 7
JUMPER
SOCKET
J 9
JUMPER
SOCKET
A 1
A 2
A3
A4
A5
A6
A7

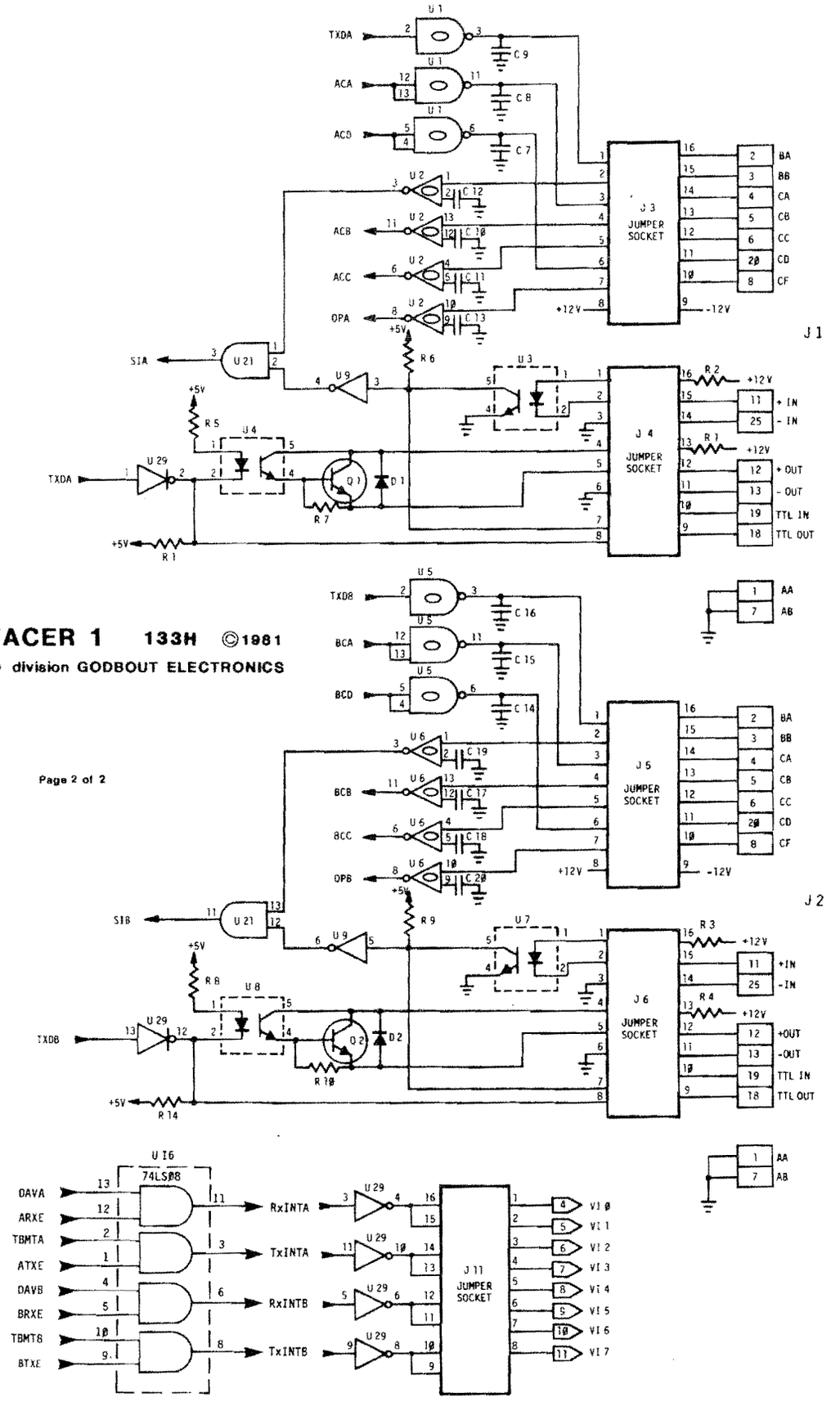


Page 1 of 2

INTERFACER 1 133H ©1981

COMPUPRO division GODBOUT ELECTRONICS




INTERFACER 1 133H ©1981
COMPUPRO division GODBOUT ELECTRONICS
Page 2 of 2
TXDA
ACA
ACB
ACC
ACD
OPA
SIA
TXDB
BCA
BCB
BCC
BCD
OPB
SIB
OAVA
ARXE
TBMTA
ATXE
OAVB
BRXE
TBMTB
BTXE
74LS08
RxINTA
TxINTA
RxINTB
TxINTB
J1
J2
J3 JUMPER SOCKET
J4 JUMPER SOCKET
J5 JUMPER SOCKET
J6 JUMPER SOCKET
J11 JUMPER SOCKET
+12V
-12V
+5V
+IN
-IN
+OUT
-OUT
TTL IN
TTL OUT
BA
BB
CA
CB
CC
CD
CF
AA
AB
VI Ø
VI 1
VI 2
VI 3
VI 4
VI 5
VI 6
VI 7

BOARD POWER UP AND TESTING

Now the fun part . . . before you install the board in your system, take the time to check these few things over:

1. _____ ARE ALL IC'S PROPERLY INSERTED?
2. _____ ARE THE BAUD RATE SWITCHES SET TO THE PROPER BAUD RATE FOR YOUR TERMINAL?
3. _____ ARE THE PORTS YOU WANT TO USE PROPERLY ADDRESSED AND ENABLED?
4. _____ IF YOU DON'T WANT TO USE A PORT, IS IT DISABLED?
5. _____ ARE THE HARDWARE/SOFTWARE PROGRAMMING JUMPERS SET THE WAY YOU REALLY WANT THEM?
6. _____ ARE YOU SURE YOU AREN'T TYING TWO OR MORE OF THE PROGRAMMING OUTPUTS TOGETHER?
7. _____ ARE THE INTERRUPT JUMPERS SET TO THE PROPER INTERRUPT LINE OR ARE THEY COMPLETELY REMOVED IF NOT BEING USED?
8. _____ IF USING CURRENT LOOP - ARE THE PROGRAMMING LINES SET PROPERLY FOR USING EITHER THE ONBOARD CURRENT SOURCE OR THE EXTERNAL SOURCE? (SEE FIGURES A AND B)
9. _____ IF USING RS232 - ARE THE PROGRAMMING LINES SET PROPERLY FOR MASTER OR SLAVE MODE? (SEE FIGURE C AND D).
10. _____ ARE THE OPTION JUMPERS SET THE WAY YOU WANT?

If you have answered 'YES' to all of the above - you are ready to proceed.

It's now time to power up your board. With the power on, verify that the voltages specified earlier are still at their proper levels and that no chips are getting excessively hot (U1, U2, U5, U6 and U10 should be warm). If everything is OK up to this point, enter the I/O board testing routine via either front panel switches or editor/assembler.

If the I/O board is working, characters typed on the terminal will be echoed back.

I/O BOARD TESTING ROUTINE

THIS PROGRAM INPUTS CHARACTERS FROM A TERMINAL DEVICE AND ECHOES THEM BACK

```
0000                0000 *
0000                0010 *
0000                0020 *
0000                0030 *
0000                0040 *
0000                0050 CONTROL EQU  01H      CONTROL PORT ADDRESS
0000                0060 STATUS  EQU  01H      STATUS PORT ADDRESS
0000                0070 DATA    EQU  00H      DATA PORT ADDRESS
0000                0080 *
0000                0090 *
0000                0100         ORG  0000H
0000                0110 *
0000 3E 00          0120 START   MVI  A,00H    INITIALIZE THE
0002 D3 01          0130         OUT  CONTROL  CONTROL PORT
0004 DB 01          0140 BEGIN   IN   STATUS   INPUT STATUS AND
0006 E6 02          0150         ANI  02H      MASK FOR DAV
0008 CA 04 00       0160         JZ   BEGIN    IF NOTHING/CHECK AGAIN
000B DB 00          0170         IN   DATA     IF DAV -1, INPUT DATA AND
000D 47             0180         MOV  B,A      MOVE TO REGISTER B
000E DB 01          0190 LOOP    IN   STATUS   INPUT STATUS AND
0010 E6 01          0200         ANI  01H      MASK FOR TBMT
0012 CA 0E 00       0210         JZ   LOOP     IF NOT READY, CHECK AGAIN
0015 78             0220         MOV  A,B      IF TBMT -1, MOVE DATA INTO
0016 D3 00          0230         OUT  DATA     REGISTER A, THEN OUTPUT
0018 C3 04 00       0240         JMP  BEGIN    JUMP FOR NEXT CHARACTER
```

TROUBLESHOOTING

Assuming that your board is having problems and the testing program is not running - *DON'T PANIC!!!* A logical approach to troubleshooting will show you that an I/O board is nothing to be frightened of. To troubleshoot this board, you will either need a scope or a logic probe. A frequency counter is handy too.

1. Verify that the Baud rate you want is being delivered to the UART. With a scope or frequency counter, measure the frequency at pins 40 and 17 of the UART. Remember that the UART needs a 16x clock, so the frequency will be 16 times the Baud rate.

   example: 1200 Baud x 16 = 19.2 KHz
   for people with scopes:
   $$\text{Frequency (Hz)} = \frac{1}{\text{cycle period (seconds)}}$$

2. Verify with scope or logic probe that data reaches pin 20 of the UART when you type a character on your terminal. If not - trace back and see where it's being lost.

3. Verify that the status bit DAV (pin 19 of the UART) goes high after striking a character from the terminal. If it does not, and data is getting to the UART, you may have a bad UART.

4. Verify that your computer is getting the status of the board by running a simple program like the one below

   ```
   LOOP    IN   STATUS   (your status port)
           JMP  LOOP
   ```

   Verify that the line SWE (A or B) pulses low while this program is running (pins 1 and 19 of the status gate U11 or U19), and that the data is getting onto the buss from the buss driver U30 as pin 1 and 19 pulse low and high respectively.

5. Verify that the programming latches (U12, 14, 15, 17) and the UARTS are receiving their control strobes when the following program is run:

   ```
   LOOP    OUT  CONTROL   (your control port)
           JMP  LOOP
   ```

   The line CS (A or B) should strobe low while this program is running (check pin 9 of U12 and U15 *or* U14 and U17)depending on which channel you are currently strobing.

6. Verify that data is reaching the UARTS from the host computer by running the following program and observing pin 34 of the UARTS (DS, A or B) strobes low. *note:* only garbage will be transmitted.

   ```
   LOOP    OUT  DATA      (your data port)
           JMP  LOOP
   ```

7. Verify that data is reaching the computer from the UARTS by running the following program and observing that RDE (A or B) strobes low (pins 4 and 18 of the UARTS) and that pins 1 and 19 of U26 strobe low and high respectively.

   ```
   LOOP    IN   DATA      (your data port)
           JMP  LOOP
   ```

8. Verify that serial data is exiting from UART pin 25 and that it is propigated to the output connector pin through the output level translators. (test this while running the I/O board testing routine).

9. If you don't seem to be getting any of the above mentioned control strobes, check that either pin 1 of U22 is strobing high or pin 15 is strobing low when running any of the above programs. If not, either the address is set wrong or there is a defective switch, IC or board in the I/O address decoding circuitry. Also pin 2 and 14 of U22 should be strobing low to indicate either a pDBIN or PWR.

CIRCUIT DESCRIPTION

The heart of this serial I/O board is a MOS/LSI UART similar to the TR1602-B or the TR1863 by Western Digital. These UARTs perform the complete parallel-to-serial and serial-to-parallel conversion, error detection and serial format modification necessary for reliable and flexible serial communication.

This I/O board can be roughly divided into seven major sections; I/O Port Select Logic, Bus Driver Logic, Hardware/Software Programming Logic, Baud Rate Select Logic, Interrupt Logic, Serial Line Level Conversion Logic, and the UARTs.

I/O PORT SELECT LOGIC

Address lines A7-A1 each feed into one input of the 25LS2521 octal comparators (U19, U20) with the other input connected to one switch position of the port select DIP switch. The output of the comparator is gated with the I/O detect logic consisting of X-NOR gate (U23) on sINP and sOUT. When an I/O operation is decoded, and the selected address is on the bus, the control line decoder (U22) is enabled and the proper control signal is output to the UARTs dependent on the 2 select inputs, address A0, and sOUT, and when either pDBIN or pWR* occurs. At the same time, board enable is decoded and the input buffer (U25) is enabled or disabled with sINP.

BUS DRIVER LOGIC

In this section, there will be reference to 2 different buses, the S-100 bus and the internal bi-directional board bus.

The S-100 Data Out line (DO0-DO7) are buffered through octal buffer U25 to drive the internal board bus, which presents the S-100 bus information to the UARTs and the Hardware/ Software Programming Logic to be latched when the processor is not performing an input operation.

The S-100 Data In lines (DI0-DI7) are driven by the 74LS241 octal bus driver (U26) from the internal board bus. The board bus is driven by the UARTs for data and the 81LS95/97 status control buffers. This bus is a TRI-STATE bus, and therefore only one device (either the UARTs or the status buffers) controls the bus at any one time.

HARDWARE/SOFTWARE PROGRAMMING LOGIC

The Hardware/Software Programming Logi consists of U12, U15, J7, and J9 for Channel A and U14, U17, J8 and J10 for Channel B along with one AND gate (U21) and two inverters (U24) for the reset logic. On power-up and reset, all sixteen of the quad 'D' latches (74LS175- U12, U14, U15, U17) are cleared by either power on clear (POC*) or reset (pRESET*). This establishes defined levels on one side of each of the jumper sockets (J7, J8, J9 and J10) which can be wired to each of the four signals on the opposite side of the jumper socket. Since the latches bring out both Q and Q* outputs, each of the 8 control lines per Channel has its own control bit and each bit is brought out both inverted and non-inverted at the same time. This allows each of the control bits to be set either high or low for the particular system configuration needed on power-up. By outputting a '1' to any of the control bits, the particular control line's level will be inverted. This occurs when the clock inputs of the control latches are toggled by the I/O decoder (U22) causing the current data bus data to be latched. This new data causes the jumper sockets levels to change and the control bit to change along with them. One note of caution: NEVER never wire any output of a latch to another latch output or leave both programming shunts unbroken as this could lead to heating and possible damage to the 74LS175s.

BAUD RATE SELECT LOGIC

The Baud Rate Select Logic is by far the simplest section of the board. The complete circuitry consists of the baud rate generator (BR2941-L or BR1941-L, U10), a 5.0688 MHz crystal (X1), DIP switch S1, and the resistor/ zener minus 5 volt regulator (R12 and D3). The baud rate generator contains a crystal oscillator and two programmable dividers to obtain the two independant baud clocks for the UARTs (CLKA and CLKB). The baud rate is set in a binary fashion on DIP switch S1 with positions 1 - 4 for Channel A and positions 5 - 8 for Channel B. Resistor R12 and zener diode D3 provide the - 5 volt substrate bias for the BR2941-L (not needed for the BR1941-L). NOTE: Holes for SIP R11 have been provided in the event that a baud rate generator without internal pullups is provided. R11 will only be provided if needed.

INTERRUPT LOGIC

The interrupt logic consists of four AND gates (U16) and four open collector inverters (U29). When a particular interrupt has been enabled by bringing the necessary enable bit high either by power-up programming or altering the mask under software control, the AND gates can pass their status onto the inverters. When either the Data Available (DAV) or Transmitter Buffer Empty (TBMT) signals go high from the UART, this level is passed by the AND gates to the inverters. The high level on the inverter input pulls the output out of the high impedance state and drives the bus low. The processor acknowledges this interrupt by either reading from or writing data to the UART, which returns the inverter output and the bus line to the high impedance state.

SERIAL LINE LEVEL CONVERSION LOGIC

The Level Conversion Logic consists of two main parts; the RS232 - TTL level converters and the current loop and TTL level converters. Since both Channels are the same, we will only discuss Channel A. All the signals from the Channel A connector (J1) are brought out to jumper sockets J3 and J4 for user programming.

All RS232 level signals are taken from J3 and are either fed from or fed to the two level converters U1 and U2 (1488 - 1489). The 1488 converts TTL level signals to the +12 and -12 volt levels of an RS232 line. Holes for disc capacitors (C7-C9 and C14-C16) are provided on the outputs of these drivers to limit slew rate to less than 30V/uSEC for compatibility with the EIA RS-232-C specifications. For very short cable lengths, you might want to install disc caps with a value no greater than 400pf. For very long cables, we advise that no caps be installed due to the cable capacitance already limiting slew rate. The 1489 converts the RS232 levels back to TTL levels for the UARTs and the status bits. Holes are provided on the board for adding the response control capacitors (C10-C13 and C17-C20) for increased noise immunity.

The current loop and TTL signals all enter and exit through J4. The +IN and -IN signals are designed to feed opto-isolator U3 either through the on board current source (R2) or directly from an external current source. The output of the opto-isolator is combined with the TTL IN signal, inverted (U9) and ANDed (U21) with the level converted RS232 data and fed to the UART (U11) via SIA. Transmit data "A" (TxDA) is inverted (U29), fed through the jumper socket J4 to the TTL out line, and drives the LED of opto-isolator U4. The output of U4 is buffered by transistor Q1 (with diode D1 to clip negative spikes) and fed to J4 where it user either the internal current source (R1) or an external source and drives the +OUT and -OUT lines.

UARTs

The UARTs (U11, U18) take the data from the internal board bus, the serial parameters from the Hardware/Software Programming Logic, The baud clock from the Baud Rate Select Logic and the control signals from the I/O Port Select Logic and outputs serial data to the Serial Line Level Conversion Logic.

The UARTs also take serial data from the Conversion Logic and with the previously mentioned sections yield data and status to the internal board bus for driving the S-100 bus. In summary, as stated at the beginning of this section, the UARTs are the heart of this board.

Parts List

Upon receipt of your kit, check your parts against the list below.

☐ 1 - Circuit board

INTEGRATED CIRCUITS (NOTE: the following parts may have letter suffixes & prefixes along with the key numbers given below.)

☐ 2 - TR1863-P UARTs (U11, U18)
☐ 1 - BR2941-L or BR1941-L Baud rate generator (U10)
☐ 2 - 1488/75488 TTL to RS232 converter (U1, U5)
☐ 2 - 1489/75489 RS232 to TTL converter (U2, U6)
☐ 1 - 74LS241 octal buss driver (U26)
☐ 1 - 81LS95 octal buss driver (U25)
☐ 2 - 81LS95/97 octal buss driver (U27, U28)
☐ 4 - 74LS175 quad "D" latches (U12, U14, U15, U17)
☐ 1 - 74LS266 quad X-NOR with o.c. outputs (U23)
☐ 1 - 74LS05 hex o.c. inverter (U29)
☐ 2 - 74LS04 hex inverters (U9, U24)
☐ 1 - 74LS155 decoder (U22)
☐ 2 - 25LS2521 octal comparators (U19, U20)
☐ 2 - 74LS08 quad 2 input AND (U21, U13)
☐ 4 - MCT-2 or similar opto-isolator (U3, U4, U7, U8)
☐ 1 - 7805 positive 5V regulator (U30)
☐ 1 - 7812 positive 12V regulator (U32)
☐ 1 - 7912 negative 12V regulator (U31)

OTHER ELECTRONIC COMPONENTS

☐ 2 - NPN transistors (Q1,Q2)**
☐ 2 - Signal diodes, 1N914, 809-36 or similar (D1,D2)**
☐ 1 - 5V zener diode, 1N751A or similar (D3)**
☐ 4 - Single inline resistor packs (R11,* R15, R16, R16)**
☐ 4 - 560 ohm ¼ watt resistors (R1, R2, R3, R4)**
☐ 2 - 470 ohm ¼ watt resistors (R5, R8)**
☐ 2 - 4.7K ohm ¼ watt resistors (R7, R10)**
☐ 5 - 2.7K ohm ¼ watt resistors (R12, R13, R14, R6, R9)**
☐ 1 - 5.0688 MHz crystal (Z1)
☐ 2 - 39uF 10V tantalum capacitors (C1, C2)
☐ 4 - 1.8uF 35V tantalum capacitors (C3, C4, C5, C6)
☐ 31 - ceramic disc bypass capacitors**

MECHANICAL COMPONENTS

☐ 36- low profile sockets**
☐ 3 - 8 pole DIP switches**
☐ 1 - heat sink for U30
☐ 3 - sets of 6-32 hardware
☐ 2 - 26 pin I/O connectors
☐ 5 - 16 pin DIP platforms
☐ 4 - 16 pin AMP dip shunts
☐ 1 - Instruction booklet

*Note: R11 may not be installed or needed on some boards.
**Supplied already soldered to the board.

Component Layout

IF YOU NEED ASSISTANCE ALWAYS CONTACT
YOUR **COMPUPRO DEALER FIRST**

CUSTOMER SERVICE INFORMATION

Our paramount concern is that you be satisfied with any Godbout CompuPro product. If this product fails to operate properly, it may be returned to us for service; see warranty information below.
If you need further information feel free to write us at:

Box 2355, Oakland Airport, CA 94614-0355

LIMITED WARRANTY INFORMATION

Godbout Electronics will repair or replace, at our option, any parts found to be defective in either materials or workmanship for a period of 1 year from date of invoice. Defective parts *MUST* be returned for replacement.

If a defective part causes a Godbout Electronics product to operate improperly during the 1 year warranty period, we will service it free (original owner only) if delivered and shipped at owner's expense to and from Godbout Electronics. If improper operation is due to an error or errors on the part of the purchaser, there may be a repair charge. Purchaser will be notified if this charge exceeds $50.00.

We are not responsible for damage caused by the use of solder intended for purposes other than electronic equipment construction, failure to follow printed instructions, misuse or abuse, unauthorized modifications, use of our products in applications other than those intended by Godbout Electronics, theft, fire, or accidents.

Return to purchaser of a fully functioning unit meeting all advertised specifications in effect as of date of purchase is considered to be complete fulfillment of all warranty obligations assumed by Godbout Electronics. This warranty covers only products marketed by Godbout Electronics and does not cover other equipment used in conjunction with said products. We are not responsible for incidental or consequential damages.

Prices and specifications are subject to change without notice, owing to the volatile nature and pricing structure of the electronics industry.