RS232-to-Centronics interface

Geoff Nicholls

Most microcomputers worth owning have an 'RS232' connector, or port, through which serial communications (input/output) is conducted. It is a convention that, for listing on a printer, the BASIC LLIST or LPRINT command assumes a printer is connected to the RS232 port. Problem is, serial interface printers are more expensive than parallel 'Centronics' interface printers. Save some money, build this interface.

WHILE I designed and constructed this project to drive an Admate or TI 850 80-column dot-matrix printer from the lab. Microbee, it is 'universal' enough to suit any application requiring an interface between an RS232C port and a Centronics interface.

Printers with a parallel, or 'Centronics', interface are around $80 to $150 cheaper than with a serial, or RS232C, interface. However, the 'default' printer output on most microcomputers is via the serial interface and a serial interface printer is assumed when LLISTing or LPRINTing from BASIC. This project can be constructed for considerably less than the difference between the cost of a printer with a serial or a parallel interface.

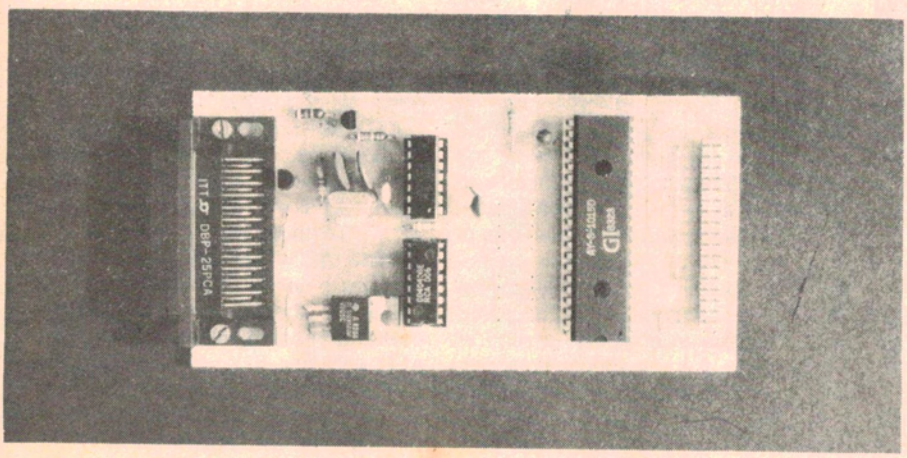
Features

The project simply plugs directly between the computer's RS232C socket and the printer's Centronics connector. It is powered from the +12 V line on the RS232C interface. It is preset to operate at a speed of 1200 baud, but provision has been made for selectable baud rates of 300, 600, 2400 and 4800 (depending on choice of one IC), in addition to that. The data format is also preset, to eight data bits one stop bit/even parity, but other formats can be selected. Tracks etched on the pc board preset the speed and data format, but provision has been made to use either links or DIL switches.

The interface is built around a single supply rail UART ('universal asynchronous, receiver-transmitter) from General Instruments, the AY-3-1015D. This chip pretty well does the whole job, even supplying the acknowledge signal (handshake) for the RS232C port. A 4.9152 MHz crystal is divided down to provide baud rate clock outputs. Either of two IC types can be used here — a 4020B or a 4040B. The 4040B provides only the lower three baud rates (300, 600 & 1200) while the 4020B provides the full complement. However, whichever type is used, it must be capable of running at 5 MHz on a 5 V supply. The minimum speed spec. for Philips and Fairchild devices equals this, but it is lower for National Semiconductor devices — though some chips may run at this speed.

Construction

Assembling the project is quite straightforward. However, there are a few points to watch. Firstly, whether you're using a ready-made pc board or have etched your own, check that all the tracks are intact and that there are no bridges, particularly where tracks run between IC pins. Also see that all the holes are drilled correctly. ▶

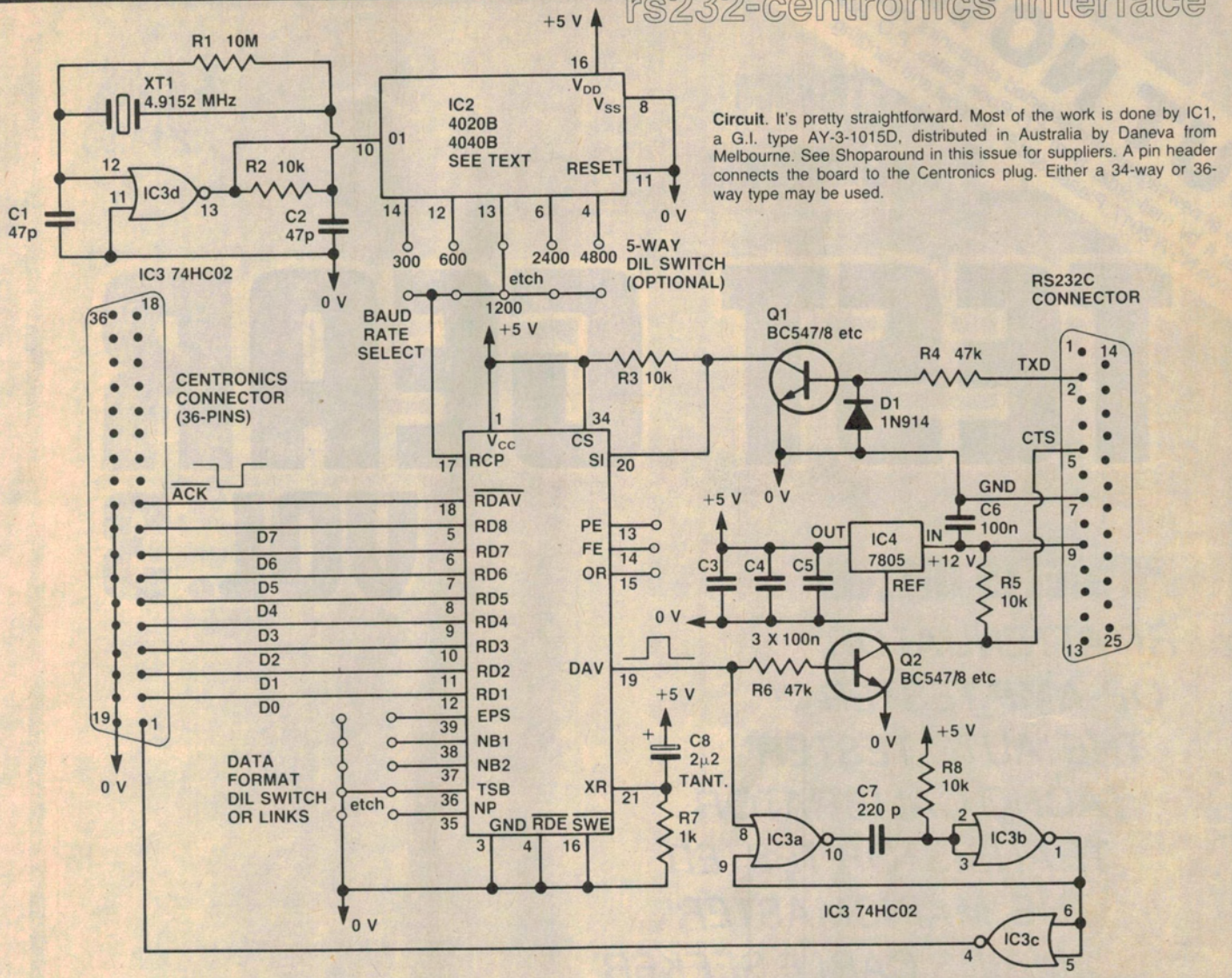


Plug in and go. There's not much to this project. The prototype here is configured to plug straight into a Microbee, but the project's suitable for any micro with an RS232C port.

HOW IT WORKS ETI-675

Most of the circuitry in the interface is inside IC1, a universal asynchronous receiver/transmitter, or UART. I have only utilised the receiver section, hence the large number of unused pins. Serial data enters the UART on pin 20 after being inverted and level shifted by Q1 to convert from RS232 voltages to TTL levels. When a complete word has been received the Data Available line (DAV) goes high, indicating that the word is ready to be read out. The DAV low-to-high transition triggers the monostable made from IC3 a, b, c. This generates the Data Strobe signal for the Centronics cable. The DAV signal also drives the Clear To Send (CTS) line on the RS232C side via inverting buffer Q2. This inhibits any further characters from the sending device (i.e: Microbee) until the printer has read the last one sent. When the printer is ready for another character it momentarily lowers the ACKnowledge line on the Centronics cable, which resets the DAV line via the UART input RDAV. This returns CTS to the high voltage and we are back where we started, ready to receive a new character.

The baud rate clock comprises IC3d, a standard crystal oscillator using a high speed CMOS gate, and IC2, a multistage divider. The divided outputs are brought to a



Circuit. It's pretty straightforward. Most of the work is done by IC1, a G.I. type AY-3-1015D, distributed in Australia by Daneva from Melbourne. See Shoparound in this issue for suppliers. A pin header connects the board to the Centronics plug. Either a 34-way or 36-way type may be used.

pad array, which can be used with a 5-position DIL switch to change rates if you wish. The PC board is etched to only use 1200 baud, so that Microbee users need not change anything.

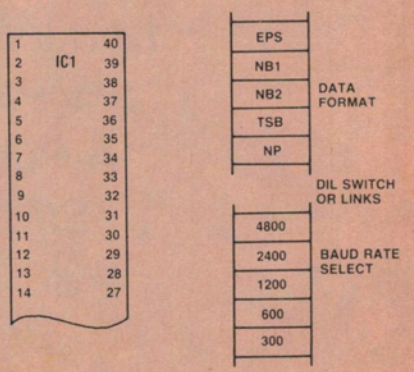
Either a 4020B or 4040B can be used for IC2, but the highest two baud rates will be different (1200 baud is unaffected). The overlay shows the rates for a 4020B i.e: 300, 600, 1200, 2400 and 4800 baud. If a 4040 is used then the lowest three rates will be the same but extra links will have to run to get 2400 and 4800 baud, owing to the different pinouts.

IC2 must be able to run with a 4.9 MHz clock at 5 V, which is slightly faster than the typical spec. for National devices, but is the minimum spec. for Fairchild or Philips (HEF4020B, HEF4040B). The symptom of a slow device is that the first stage divides by three instead of two, so look at pin 9 with a counter to make sure all is well.

The CR network of C8 and R7 reset the UART on power-up while IC4 and associated components develop the +5 V rail from the RS232C's positive supply. The pads near pins 35-39 of the UART set up the parity and bit length of the serial conversion, they are preset by board tracks to suit the Microbee.

OPTIONAL DIP SWITCHES

The PC board has been laid out to allow the option of fitting two 5-way DIL switch banks, between the UART and ICs 2 and 3. This allows the selection of different baud rates, parity and stop bits. The PC board comes with tracks etched to set eight data bits, one stop bit, no parity bit and 1200 baud. These suit most uses, but if you want to run some other combination then you will have to cut the two tracks and install wire links or DIL switches. The signal definitions are as per the diagram here and Table 1.

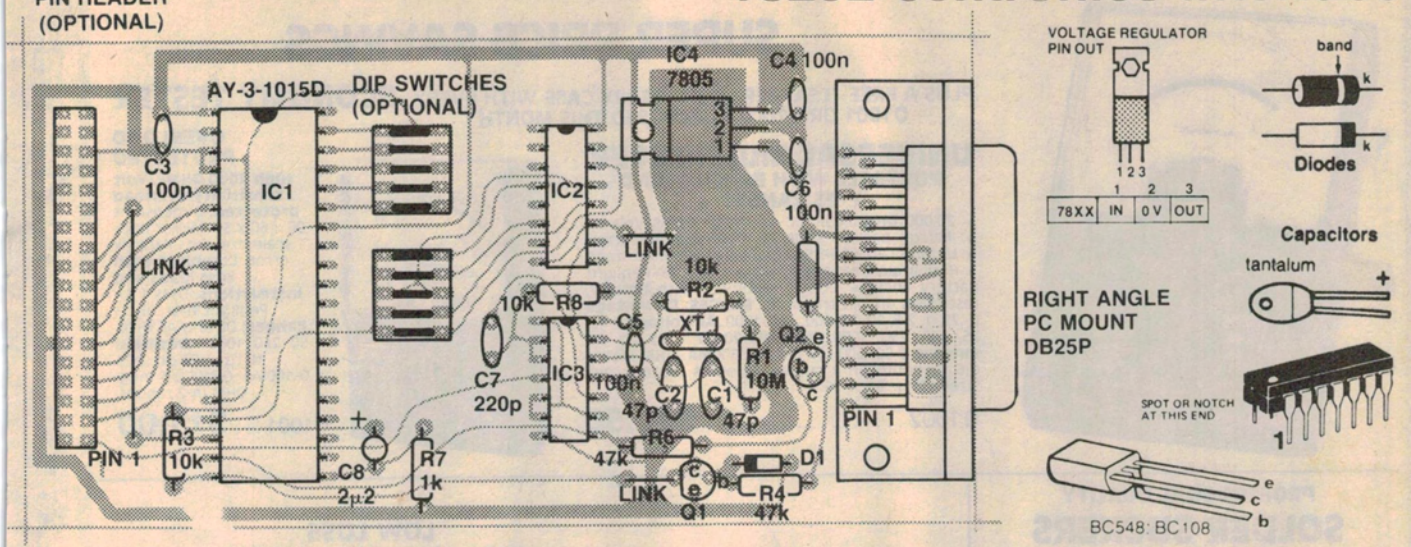


NOTE THAT YOU CAN ONLY GET 2400 AND 4800 BAUD WHEN USING THE 4020B FOR IC2

TABLE 1

SIGNAL	FUNCTION		
EPS	ON — ODD PARITY	OFF — EVEN PARITY	
NB1, NB2	NUMBER OF BITS PER CHARACTER		
	NB2	NB1	BITS
	ON	ON	5
	ON	OFF	6
	OFF	ON	7
	OFF	OFF	8
TSB	ON — 1 STOP BIT	OFF — 2 STOP BITS	
NP	ON — NO PARITY	OFF — PARITY EXPECTED	

Only one switch in the baud rate bank may be on at one time, otherwise improper operation will occur.



PARTS LIST — ETI-675

Resistors ..... all ¼W, 5%
R1 ..... 10M (10% OK)
R2, 3, 5, 8 ..... 10k
R4, R6 ..... 47k
R7 ..... 1k

Capacitors
C1, C2 ..... 47p NPO ceramic
C3, 4, 5, 6 ..... 100n 'bluechip' ceramic
C7 ..... 220p ceramic
C8 ..... 2µ2/6 V tantalum

Semiconductors
IC1 ..... AY-3-1015D (G.I.) or exact equiv.
IC2 ..... 4020B or 4040B (must run at 4.9 MHz on 5 V supply)*
IC3 ..... 74HC02
Q1, Q2 ..... BC547, BC548 etc.
D1 ..... 1N914

Miscellaneous
XT1 ..... 4.9152 MHz crystal, HC18/U

ETI-674 pc board; right angle pc mount DB25 plug; 34-way pin header (0.1 x 0.1") — optional; 34-way female IDC plug (optional); IDC Centronics plug (if required); 34-way ribbon cable; two 5-way DIL switches (optional); etc.

Price estimate: $28-$58

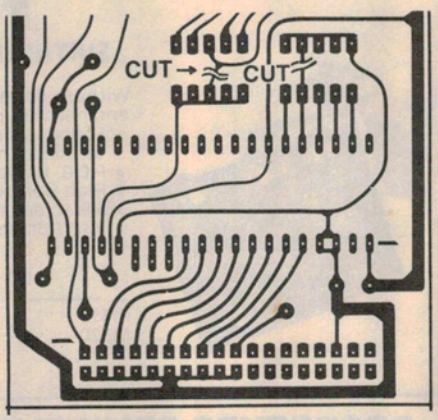
*Philips HEF4020B/HEF4040B and Fairchild 4020B/4040B known to work.

Solder all the resistors and capacitors in place first. Make sure you get the tantalum capacitor (C8) the right way round. Next solder D1 in place, making sure you get it the right way round, too. Follow with the two transistors, Q1 and Q2.

Solder the three links in place next, followed by the three IC sockets and the crystal. The three-terminal regulator (IC4) can be soldered in now. Note that a heatsink is not required for it. If you are using DIL switches for baud rate and data selection, these should be soldered in place next.

Last of all, solder the 34-pin header strip and DB25P right angle connector in place. The latter should be bolted to the board firmly before soldering so that no stress is placed on the solder joints. Insert the ICs now. The UART is an NMOS device, while the other two are CMOS. Take the usual precautions against static damage.

Make up interconnecting cables to suit your individual requirements (see next section). If you're using a Centronics plug, I strongly recommend you use an insulation displacement type as the solder pin type is much harder to assemble, with a chance of errors and poor connections.



A cut above. To obtain other baud rates and data formats, the two tracks marked above must be cut. Links or DIL switches can be used to select the required configuration.

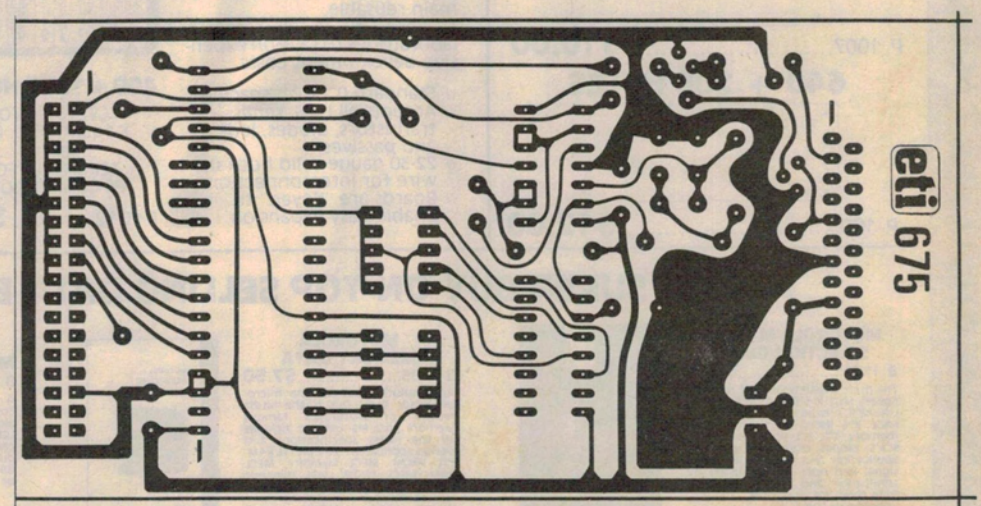
Mounting

Actual mounting details are left up to you, as requirements will vary widely. The board can be mounted inside your printer, in which case the DP25P plug and 34-pin header may be dispensed with and the board wired-in directly. Alternatively, the board may be mounted inside a zippy box, or other suitable case, and cables wired to it with suitable connectors on the end. Or, if you have a Microbee, you can do as I did and plug it straight in to the DB25 socket on the rear of the cabinet and let it hang out the back. A couple of 'feet' might be useful, though. These could consist of two standoff pillars with rubber grommets attached, bolted to the end of the board either side of the 34-pin header. The overall price of the project depends on the connectors and case used — or not used.

Using it

With a Microbee, just plug it in and go! With other micros, setting the baud rate and data format is simply a matter of cutting the two tracks (as indicated in the diagram here) and either installing links or DIP switches and linking the appropriate outputs across as per Table 1.

Happy LLISTing! ●



Board artwork. Full-size reproduction of the pc board. ▶