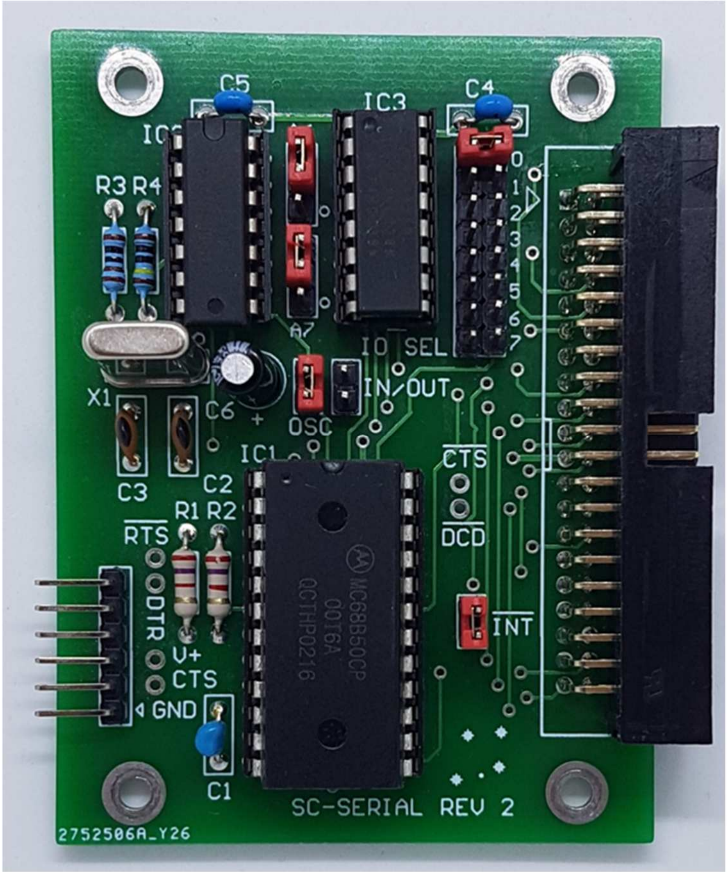
A 6850 ACIA Serial add-on board for the Southern Cross Computer

By Craig RS Jones



Bit-banging a serial port has a few drawbacks. Whether you are transmitting or receiving you just can't do anything else. Get the bit delay timing wrong when transmitting and the receiving device will not understand, miss the falling edge of the start bit when receiving and you may synchronise to a data bit or miss an entire character.
To relieve the microprocessor of this burden the reception and transmission of serial data can be off-loaded to purpose built hardware, enter the Universal Asynchronous Receiver Transmitter (UART).

The transmitting side of the UART handles the serialisation and clocking out of the serial data adding the necessary start and stop bits, generating the parity bit as necessary.
The receiver synchronises to the start bit of the incoming serial data and shifts the data bits in, checking the stop bits and parity before setting a flag to show that it has received data.

The design presented here is based on Grant Searle's 'Grant's 7-chip Z80 computer', I have added I/O decoding and have included an oscillator for the baud rate clock instead of using the Z80 clock like Grant so you can set the Baud rate independently of the Southern Cross (SC) clock.

The MC6850 ACIA (Asynchronous Communications Interface Adapter) used is obsolete and no longer manufactured but you can still get it on online from multiple vendors.
Available in 3 speed grades 1Mhz (MC6850) 1.5Mhz (MC68A50) and 2Mhz (MC68B50) the original part was fabricated in NMOS. Multiple vendors produced the part as a second source and Hitachi made a CMOS version, HD6350. My device of choice is the CMOS Hitachi HD63B50P.

Circuit Description

The circuit has 3 distinct parts, I/O address decoding, a crystal oscillator, and the ACIA.

I/O Address Decoding

The I/O address decoding uses the same idea as the SC. Using the combination of the two highest I/O address lines, A7 and A6 to enable the 74HC138 and the lower I/O address lines to give 8 possible decoded I/O addresses of two bytes each.

Crytal Oscillator

The ACIA requires an external Baud Rate clock for both the Receiver and Transmitter, this clock can be programmatically divided by 1,16 and 64 inside the ACIA to derive the Baud Rate.

The 7.3728MHz crystal oscillator is divided by 64 to give a 115200 Baud rate clock. You can change the crystal and load capacitors to select other Baud rates, i.e. a 1.84320MHz crystal with the internal divider set to divide by 16 will also yield a 115200 Baud Rate.

The OSC and IN/OUT straps are used to select the Baud rate clock source, installing both the OSC and IN/OUT strap enables the oscillator on the serial board to be used for both the Baud Rate Generator and the Z80 CPU Clock on the SC main board. If you have used a strap for the SPEED selector on the SC you can remove it or you can just remove IC10, the 74HCU04 IC to disconnect the SC clock oscillator if you have used a switch as your speed selector.

Installing only the OSC strap selects the on-board oscillator as the Baud rate clock source, similarly installing only the IN/OUT strap allows the clock oscillator on the SC main board to be used as the Baud Rate Clock.

ACIA

The MC6850 ACIA uses the Motorola bus timing where data is read written on the falling edge of the E clock. /IORQ is inverted to provide this signal along with /WR as the R/W signal.
Chip selects are provided by /M1, to prevent spurious I/O during an interrupt acknowledge cycle if interrupts are used, along with the IO Select from the 74HC138.
RSO is connected to AO to select between the Control/Status registers and the Transmit/Receive registers.

The /IRQ output is an active low open drain output that connects to the /INT pin on the Z80 which has a pullup resistor on it, this type of connection is known as a 'wired-OR', any device connected in this way can pull-down the /INT line and cause an interrupt.

Modem control signals RTS,CTS and DCD

The RTS output can be programmed High or Low without affecting the reception or transmission of data,
CTS and DCD are connected to ground for normal operation, there are holes provided for the pins if you want to use them for flow control, there are some small tracks that can be cut to disconnect the pins from Ground.
You can use these signals for hardware flow control, stopping the transmitter from sending until the receiver catches up but I haven't tried this yet...

Serial Port Connector / USB to Serial Converter

The board was designed with the top USB to Serial adapter in mind but the middle one, which has the connections reversed, seems to be more available now but has to be plugged in upside down. You could always go 'old school' and wire in the MAX202 serial port adapter.

Reference:

http://searle.wales/