"Inside_RetroSurfer.txt"

```
                                                    v0.3, last changed
16.12.2001

------------------------------------------------------------------------
---
Programming docs for the Silversurfer serial port (specific to when used
with
the Retro Replay freezer cartridge for the C-64)       (w)2001
Groepaz/Hitmen
------------------------------------------------------------------------
---

The Retro Replay has an accessory connector that can carry Amiga 1200
hardware.
I tend to call this connector the "Silversurfer port", as it will not be
able
to carry bigger expansions of the 1200. The connector uses the spare_CS
signal,
not the RTC_CS signal. This lets you use add-ons like the Silversurfer to
add a
serial port to the C-64.

The 16 registers of the clock-port are apped to $de02-$de0f (lower two
registers not available!). The IRQ of that port is connected to the NMI line
of the 6510 processor. The two missing bytes of the Spare_CS space in non-
REU
compatible mode will be no problem, because the Silversurfer is mirrored
over
that area twice.

To use the Silversurfer, just write $01 to $de01 to activate the accessory
connector and use $de08-$de0f for the eight registers of the 16c550 UART.

Do _NOT_ use the registers shadowed at $de02-$de07, these may no more work
in
future versions of the Silversurfer and the Retroreplay hardware.

Also to preserve C128 compatibility, the cpu should always be set to 1MHz
mode
when the Silversurfer is accessed.

------------------------------------------------------------------------
---
16c550 UART - Universal Asynchronous Receiver/Transmitter Overview
------------------------------------------------------------------------
---
```

```
Port-Address  Access        Description

N/A     de08 (r/w) RXD/TXD Transmit/Receive Buffer
             (r/w)   DLL       if bit 7 of LCR is set: Baud Rate Divisor
LSB
N/A     de09 (r/w)   IER - Interrupt Enable Register
             (r/w)   DLM       if bit 7 of LCR is set: Baud Rate Divisor
MSB
(de02) & de0a (r)     IIR - Interrupt Identification Register
             (w)     FCR - FIFO Control Register
(de03) & de0b (r/w)   LCR - Line Control Register
(de04) & de0c (r/w)   MCR - Modem Control Register
(de05) & de0d (r)     LSR - Line Status Register
(de06) & de0e (r)     MSR - Modem Status Register
(de07) & de0f (r/w)         Scratch Pad Register

------------------------------------------------------------------------
---
16c550 UART - Universal Asynchronous Receiver/Transmitter Detailed
Description
------------------------------------------------------------------------
---

if bit 7 of LCR is 1 :

$de08 - (r/w) DLL     - Baud Rate Divisor LSB
$de09 - (r/w) DLM     - Baud Rate Divisor MSB

        - Baud rate divisors can be calculated by taking the oscillating
          frequency (7,372,800) and dividing by the quantity of the desired
          baud rate times the UART clocking factor (16).  Use the following
          formula:
                      divisor = 7372800 / (BaudRate * 16);




        - it is not recommended to use zero as divisor.

            Baud Rate       Divisor          Baud Rate      Divisor

                  1                             4800      96    $0060
                 15                             9600      48    $0030
                 75                            19200      24    $0018
                 50    9216                    38400      12    $000c
                150    3072                    57600       8    $0008
                300    1536                   115200       4    $0004
                600     768                   153600       3    $0003
               1200     384                   230400       2    $0002
               1800     256    $0100          460800       1    $0001
               2400     192    $00c0
```

```
        note: baudrates higher 230k are unknown to work on the retroreplay
              hardware. (most pc's wont do more than 115k anyway ;=P)

if bit 7 of LCR is 0 :

$de08 - (r/w) RXD/TXD - Receive/Transmit Buffer

$de09 - (r/w) IER     - Interrupt Enable Register

         7 6 5 4 3 2 1 0
                           +---- 1 = enable data available interrupt (and
16550 Timeout)
                         +------ 1 = enable Transmit Holding Register empty
(THRE) interrupt
                       +-------- 1 = enable Reviever lines status interrupt
                     +---------- 1 = enable modem-status-change interrupt
            +------------------ reserved (zero)

        - 16550 will interrupt if data exists in the FIFO and isn't read
          within the time it takes to receive four bytes or if no data is
          received within the time it takes to receive four bytes.

$de0a - (r) IIR - Interrupt Identification Register

         7 6 5 4 3 2 1 0
                       +---- 1 = no int. pending, 0=int. pending
                  +-------- Interrupt ID bits (see below)
                +---------- 16550  1 = timeout int. pending, 0 for
8250/16450
             +------------- reserved (zero)
          +----------------- 16550  set to 1 if FIFO queues are enabled

        Interrupt ID Bits

         21       Meaning            Priority           To reset
         00  modem-status-change      lowest      read MSR
         01  transmit-register-empty  low         read IIR / write THR
         10  data-available           high        read rec buffer reg
         11  line-status              highest     read LSR

        - interrupt pending flag uses reverse logic, 0 = pending, 1 = none
        - interrupt will occur if any of the line status bits are set
        - THRE bit is set when THRE register is emptied into the TSR


$de0a - (w) FCR - FIFO Control Register

         7 6 5 4 3 2 1 0
                       +---- 1 = enable FIFO and clear XMIT and RCVR FIFO
queues
                     +------ 1 = clear RCVR FIFO
```

```
                  +-------- 1 = clear XMIT FIFO
                +---------- 1 = change RXRDY & TXRDY pins from mode 0 to
mode 1 (DMA Mode select)
             +------------- reserved (zero)
          +----------------- trigger level for RCVR FIFO interrupt

        Bits      RCVR FIFO
         76     Trigger Level
         00        1 byte
         01        4 bytes
         10        8 bytes
         11       14 bytes

        - Bit 0 must be set in order to write to any other FCR bits
        - Bit 1 when set to 1 the RCVR FIFO is cleared and this bit is
reset.
          The receiver shift register is not cleared.
        - Bit 2 when set to 1 the XMIT FIFO is cleared and this bit is
reset.
          The transmit shift register is not cleared.


$de0b - (r/w) LCR - Line Control Register

         7 6 5 4 3 2 1 0
                   +------ word length select bits (see below)
                 +-------- 0 = 1 stop bit, 1 = 1.5 or 2  (see note)
               +---------- 0 = no parity, 1 = parity (PEN)
             +------------ 0 = odd parity, 1 = even (EPS)
           +-------------- 0 = parity disabled, 1 = enabled
         +---------------- 0 = turn break off, 1 = force spacing break
state
          +----------------- 1 = baud rate divisor latch (DLAB); 0 = RBR,
THR or IER

        Bits
         10     Word length bits
         00 = 5 bits per character
         01 = 6 bits per character
         10 = 7 bits per character
         11 = 8 bits per character

        - stop bits = 1.5 for 5 bit words or 2 for 6, 7 or 8 bit words
        - bit 7 changes the mode of registers $de08 and $de09  If set these
          registers become the LSB and MSB of the baud rate divisor.
          Otherwise $de08 is the Transmit/Receive Buffer Register and $de09
is
          the Interrupt Enable Register.


$de0c - (r/w) MCR - Modem Control Register
```

```
         7 6 5 4 3 2 1 0
                     +---- 1 = activate DTR
                   +------ 1 = activate RTS
                 +-------- OUT1
               +---------- OUT2
             +------------ 0 = normal, 1 = loop back test
         +----------------- reserved (zero)

        - If bit 4 is set, data from the Transmit Shift Register is received
          in the Receiver Shift Register.  The SOUT line is set to logic
          high, the SIN line and control lines are disconnected.   CTS, DSR,
          RI and CD inputs are disconnected.  DTR, RTS, OUT1 and OUT2 are
          then connected internally.

$de0d - (r) LSR - Line Status Register

         7 6 5 4 3 2 1 0
                     +---- 1 = data ready
                   +------ 1 = overrun error (OE)
                 +-------- 1 = parity error (PE)
               +---------- 1 = framing error (FE)
             +------------ 1 = break interrupt  (BI)
           +-------------- 1 = transmitter holding register empty (THRE)
         +---------------- 1 = transmitter shift register empty (TSRE)
         +----------------- 1 = 16550 PE/FE/Break in FIFO queue, 0 for 8250
& 16450

        - Bit 0 is set when a byte is placed in the Receiver Buffer Register
          and cleared when the byte is read by the CPU (or when the CPU
          clears the FIFO for the 16550).  Results in Receive Data Available
          Interrupts if enabled.
        - Bits 1-4 indicate errors and result in Line Status Interrupts
          if enabled.
            - Bit 1 is set when a second byte is received before the byte
            in the Receiver Buffer Register is read by the CPU (the 16550 in
            FIFO mode sets this bit when the queue is full and the byte in
the
            Receiver Shift Register hasn't been moved into the queue).  This
            bit is reset when the CPU reads the LSR
            - Bit 2 is set whenever a byte is received that doesn't match
the
            requested parity.  Reset upon reading the LSR.  (The 16550
maintains
            parity information with each byte and sets bit 2 only when the
byte
            is at the top of the FIFO queue.)
            - Bit 3 is set when a character is received without proper stop
            bits.  Upon detecting a framing error the UART attempts to
            resynchronize.  Reset by reading the LSR.  (The 16550 maintains
            this information with each byte and sets bit 3 only when the
byte
```

```
            is at the top of the FIFO queue.)
            - Bit 4 is set when a break condition is sensed (when space is
            detected for longer than 1 fullword).  A zero byte is placed in
            the Receiver Buffer Register (or 16550 FIFO).  Reset by reading
            the LSR.  (The 16550 maintains this information with each byte
and
            sets bit 4 only when the byte is at the top of the FIFO queue.)
          - Bit 5 is set when the Transmit Holding Register shifts a byte
            into the Transmit Shift Register (or XMIT FIFO queue is empty for
            16550) and is cleared when a byte is written to the THR (or the
            XMIT FIFO).   Results in Transmit Holding Register Empty
interrupts
            if enabled.
          - Bit 6 is set when both the Transmitter Holding Register and the
            Transmitter Shift Register are empty. On the 16550, when the XMIT
            FIFO and Transmitter Shift Register are empty.
          - Bit 7 is 16550 specific and indicates there is a byte in the FIFO
            queue that was received with a Parity, Framing or Break error.


$de0e - (r) MSR - Modem Status Register

           7 6 5 4 3 2 1 0
                         +---- 1 = DCTS (Delta CTS) (CTS changed)
                       +------ 1 = DDSR (Delta DSR) (DSR changed)
                     +-------- 1 = TERI (ring indicator changed)
                   +---------- 1 = DDCD (Delta Data Carrier Detect) (DCD
changed)
                 +------------ 1 = CTS  (clear to send)
               +-------------- 1 = DSR  (data set ready)
             +---------------- 1 = RI   (ring indicator )
           +------------------ 1 = DCD  (data carrier detect)

          - Bits 0-3 are reset when the CPU reads the MSR
          - Bit 4 is the Modem Control Register RTS during loopback test
          - Bit 5 is the Modem Control Register DTR during loopback test
          - Bit 6 is the Modem Control Register OUT1 during loopback test
          - Bit 7 is the Modem Control Register OUT2 during loopback test

$de0f - (r/w) Scratch Pad Register

    This 8-bit Read/Write Register does not control the UART in anyway. It
is
    intended as a scratchpad register to be used by the programmer to hold
data
    temporarily.

------------------------------------------------------------------------
---

FIFO Interrupt Mode Operation
```

```
When the RCVR FIFO and receiver interrupts are enabled (FCR0=1, IER0=1) RCVR
interrupts will occur as follows:

A. The receive data available interrupt will be issued to the CPU when the
FIFO
   has reached its programmed trigger level; it will be cleared as soon as
the
   FIFO drops below its programmed trigger level.
B. The IIR receive data available indication also occurs when the FIFO
trigger
   level is reached, and like the interrupt it is cleared when the FIFO
drops
   below the trigger level.
C. The receiver line status interrupt (IIR=06), as before, has higher
priority
   than the received data available (IIR=04) interrupt.
D. The data ready bit (LSR0) is set as soon as a character is transferred
from
   the shift register to the RCVR FIFO. It is reset when the FIFO is empty.

When RCVR FIFO and receiver interrupts are enabled, RCVR FIFO timeout
interrupts will occur as follows:

A. A FIFO timeout interrupt will occur, if the following conditions exist:

    - at least one character is in the FIFO
    - the most recent serial character received was longer than 4 continuous
      character times ago (if 2 stop bits are programmed the second one is
in-
      cluded in this time delay).
    - the most recent CPU read of the FIFO was longer than 4 continuous
      character times ago.

   The maximum time between a received character and a timeout interrupt
will
   be 160 ms at 300 baud with a 12-bit receive character (i.e., 1 Start,
   8 Data, 1 Parity and 2 Stop Bits).

B. Character times are calculated by using the RCLK input for a clock signal
   (this makes the delay proportional to the baudrate).

C. When a timeout interrupt has occurred it is cleared and the timer reset
when
   the CPU reads one character from the RCVR FIFO.

D. When a timeout interrupt has not occurred the timeout timer is reset
after a
   new character is received or after the CPU reads the RCVR FIFO.

When the XMIT FIFO and transmitter interrupts are enabled (FCR0=1, IER1=1),
XMIT interrupts will occur as follows:
```

```
A. The transmitter holding register interrupt (02) occurs when the XMIT FIFO
is
   empty; it is cleared as soon as the transmitter holding register is
written
   to (1 to 16 characters may be written to the XMIT FIFO while servicing
this
   interrupt) or the IIR is read.
B. The transmitter FIFO empty indications will be delayed 1 character time
   minus the last stop bit time whenever the following occurs: THRE=1 and
there
   have not been at least two bytes at the same time in the transmit FIFO,
   since the last THRE=1. The first transmitter interrupt after changing
FCR0
   will be immediate, if it is enabled.

Character timeout and RCVR FIFO trigger level interrupts have the same
priority
as the current received data available interrupt; XMIT FIFO empty has the
same
priority as the current transmitter holding register empty interrupt.

-----------------------------------------------------------------------
---

FIFO Polled Mode Operation

With FCR0=1 resetting IER0, IER1, IER2, IER3 or all to zero puts the UART in
the FIFO Polled Mode of operation. Since the RCVR and XMITTER are controlled
separately either one or both can be in the polled mode of operation. In
this
mode the user's program will check RCVR and XMITTER status via the LSR.

As stated previously:

- LSR0 will be set as long as there is one byte in the RCVR FIFO.
- LSR1 to LSR4 will specify which error(s) has occurred.
- Character error status is handled the same way as when in the interrupt
mode,
  the IIR is not affected since IER2=0.
- LSR5 will indicate when the XMIT FIFO is empty.
- LSR6 will indicate that both the XMIT FIFO and shift register are empty.
- LSR7 will indicate whether there are any errors in the RCVR FIFO.
- There is no trigger level reached or timeout condition indicated in the
FIFO
  Polled Mode, however, the RCVR and XMIT FIFOs are still fully capable of
  holding characters.

-----------------------------------------------------------------------
---

general programming considerations:
```

```
- 16550's is pin and software compatible with the 16450 but has an internal
  16-byte FIFO queue that may be enabled/disabled by software
- PCs(XT) are capable of 38.4Kbaud, while AT's are capable of 115.2Kbaud ...
a
  stock c64 using the SilverSurver should pull ~80Kbaud, the standard c64
  kernal implementation will do 2400baud.
- receiver checks only the first stop bit of each character regardless
  of the number of stop bits specified
- data loss can occur without overrun or framing errors if the interrupts
are
  serviced too slowly
- reserved bits are usually set to zero.  Code should NOT rely on this being
  the case since future enhancement may use these bits

------------------------------------------------------------------------------
---
sources:

UART info:
    register info taken from my favourite ms-dos/pc quickreference "help-
pc",
    completed and updated from various other sources, register adresses and
    other related stuff adapted for the retro-replay hardware

retroreplay info:
    taken from the original inside_replay.txt written by Jens Schoenfeld

silversurfer info:
    taken from the original inside_surfer.txt written by Jens Schoenfeld,
also
    adapted to fit.

------------------------------------------------------------------------------
---
```