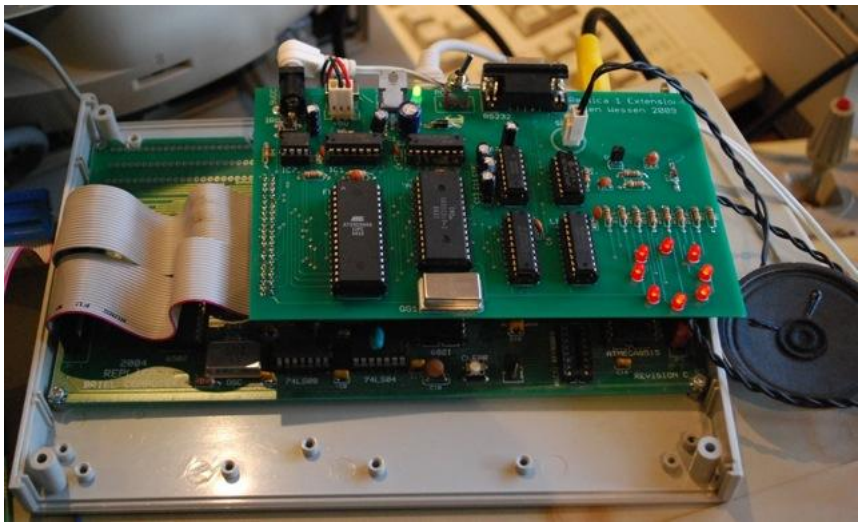
REBUS: A board with many Replica 1 extensions

15 August 2009

Features | Design & Programming | Pictures

For the first year or so of owning my Replica 1, my endeavours to enhance its capabilities were almost entirely software oriented, and thus KRUSADER was born. However, the better I understood the hardware, the more I wanted to utilise this knowledge and expand my machine from a hardware perspective as well. Based on various information and discussions on the web and on the Briel support and 6502 forums, there were a number of things that I had tried on breadboards and protoboards and it made sense to integrate them tidily on a single extension board.

The work presented here shows the outcome of this effort; what I am calling REBUS for Replica Extensions Board... (I will have to think of something for the U and S later.)



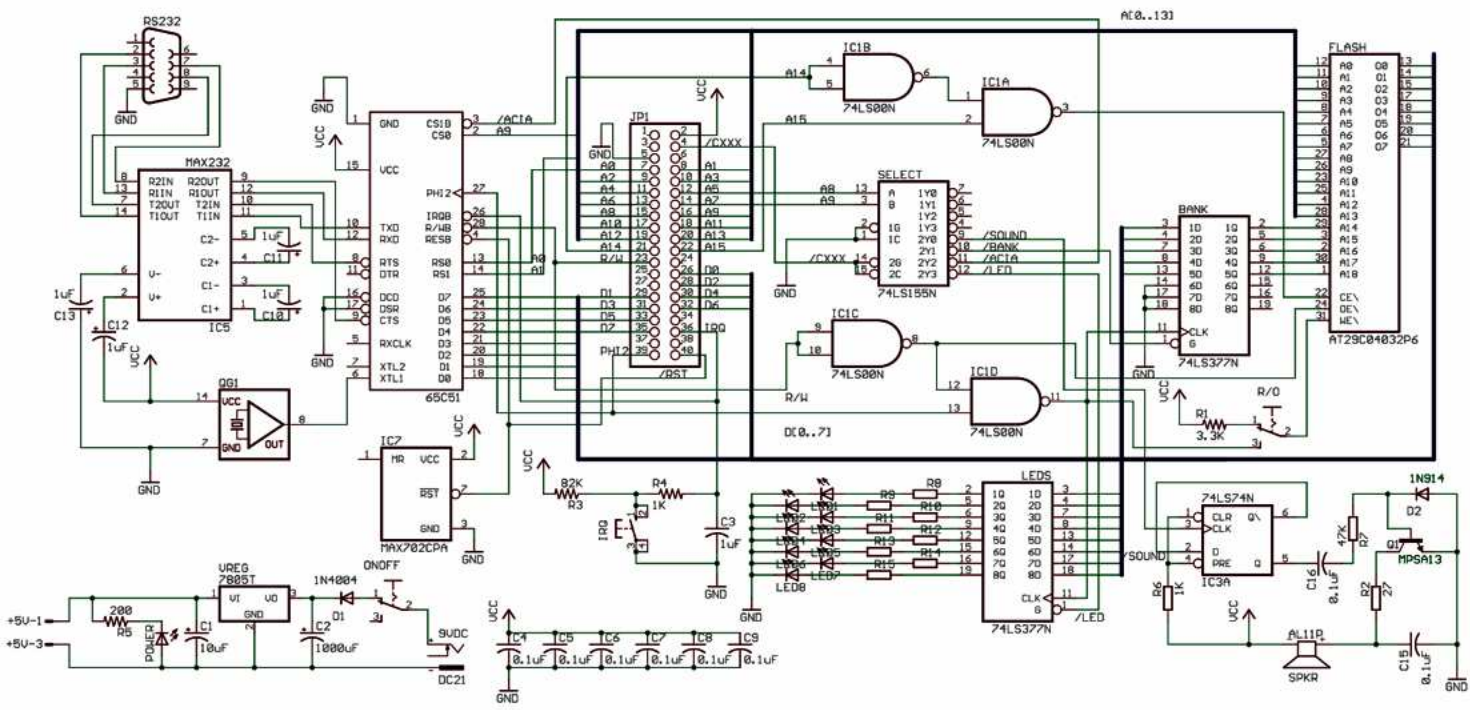
An extension board running connected to the Replica 1

Features

Power	Before the Replica 1 SE, power was delivered via an AT power supply, and since these supplies come with a power switch, it was not necessary to have one on the Replica 1 board. However, I don't have an AT power supply, and neither do I want once since they are bulky and unattractive. Instead I built a small regulator board and powered my Replica using an old mobile phone charger. This little board also had no switch or indicator LED, so I made sure that my extensions board added these for me. I also included a MAX702 so the machine would auto reset on startup.
Fast serial	One of the more painful aspects of the Replica 1 is the 2400 baud serial I/O speed. Bamse (fsastrom) showed a simple way to increase this to 19.2K on the Replica 1 support forum (Simple Comm Port). However, this has some implications for the operating system if you want both standard I/O as well as fast serial I/O supported. In order to run using standard keyboard in and video out but still use the full serial I/O speed for data transfer, it must be possible to switch between sources for character input and output, and this requires changes in the ROM. I patched the Woz monitor, Apple 1 BASIC and KRUSADER and built an ACIA ROM that supports switching between standard I/O, fast serial I/O, or running both concurrently, based on a soft switch at location $F7. (Note that when operating in concurrent mode, the serial I/O is by necessity much slower than when the pure fast serial mode is employed.)
Flash memory	The first thing I built and added to my Replica 1 was the simple 1 chip EEPROM burner from Vince Briel. This enabled me to burn new ROMs as I updated KRUSADER, and also to have 8K of ROM for storing programs. The memory circuit I use here is an extension of this idea, using instead 512K of Flash memory plus a latch to control paging as 32 banks of 16K each. The flash memory chip I use (an Atmel AT29C040A) supports writing in 256 byte chunks and I am currently writing the code to allow it to function as a hard disk. Until this is done I just use an external burner to write to it and treat it as a read only device on the Replica.
Sound	Cowgod (Apple II-style audio on the Replica-1) showed a simple way to generate sound on the Replica 1 using the same method employed by the Apple II - i.e. toggling a flip-flop at a desired rate to click a speaker. Pitch is controlled by changing the delay between toggles, and duration by the number of toggles. Some clever tricks involving damping the vibrations allow software support for volume control. Sharing the Apple II mechanism means the many Apple II sound generation techniques work, such as those described in the Apple Assembly Line article Making Noise and Other Sounds, the simple tone generation described at 8-bit Sound & Fury, and this 1991 USENET post by Steve Hawley Apple II sound hacking.
Blinkenlights	After completing the circuit design and layout for the above features, I was left an unused select line and an empty section of board, so what better to fill it with than some blinkenlights? So I added 8 LEDs. Rather than give them any preset diagnostic purpose, I simply arranged them in a circle and let their state be controlled by writing an 8-bit value to memory location $C300. Add some interesting sounds, and it makes a nice litle show for the kids.

Design and Programming

The figure below shows the circuit diagram for the Replica 1 extensions board (click for a pdf version).



The central component of the circuit is the 74LS155 SELECT decoder, taking in the $C000 signal from the Replica 1 (active low) as well as address lines A8 and A9, and outputting 4 control lines, one for each subsystem: sound ($C000), flash bank select ($C100), serial I/O control ($C200), and the LEDs ($C300).

Controlling the LEDs is by writing an 8-bit value to $C300 that causes the LEDs to light according to the binary representation of the value. A 74LS377 8-bit octal flip-flop is used as memory. As described above, control of sound is as for the Apple II with any read or write to $C000 toggling the 74LS74 and generating a click of the speaker. (The Apple II memory location for clicking the speaker is $C030, and since the last 8 bits are ignored by the select logic, Apple II sound code will work unchanged). The following code gives a simple demonstration of synchronised sound and LED control.

```
C       LIGHT AND SOUND SAMPLE

C       MEMORY MAP
SPKR    .=  $C000
LEDS    .=  $C300

C       CONSTANTS FOR NUMBER OF
C       LOOPS AND PULSE WIDTH
LOOPS   .=  $A
IWIDTH  .=  $FF

C       ZERO PAGE
COUNT   .=  $0
WIDTH   .=  $1
LOOP    .=  $2
LEDMSK  .=  $3
LEDCNT  .=  $4

C       ENTRY POINT
MAIN    .M
        LDA #LOOPS      INIT LOOPS
        STA LOOP
        LDA #$0
        STA LEDMSK
        JSR UPDATE
.1      JSR SWOOP
        DEC LOOP
        BNE .1
        RTS
```

```
C       UPDATE LED STATE
UPDATE  .M
        LDA LEDMSK
        LSR
        BNE .1
        LDA #$80
.1      STA LEDMSK      RESET MASK
        STA LEDS        LIGHT NEW LED
        LDA #$20        RESET COUNT
        STA LEDCNT
        RTS

C       ONE SWOOP SOUND
SWOOP   .M
        LDA #$1
        STA COUNT
        LDA #IWIDTH
        STA WIDTH
.1      LDY COUNT
.2      LDA SPKR
        LDX WIDTH
        DEC LEDCNT
        BNE .3          BIT IS SET
        JSR UPDATE      SET LEDS
.3      DEX
        BNE .3
        DEY
        BNE .2
        DEC WIDTH
        BNE .1
        RTS
```

The serial I/O is handled by the 6551 Asynchronous Communications Interface Adapter (ACIA), with a MAX232 to set the voltage levels for RS-232 I/O. This chip was designed as a companion to the 6502 and hence the integration is very simple. The register select pins are tied to A0 and A1, and the chip selects to A9 and /ACIA from the 155 decoder. Hence the chip register memory locations are $C200 for data, $C201 for status, $C202 for command and $C203 for control. Programming the ACIA is as shown in the following table.

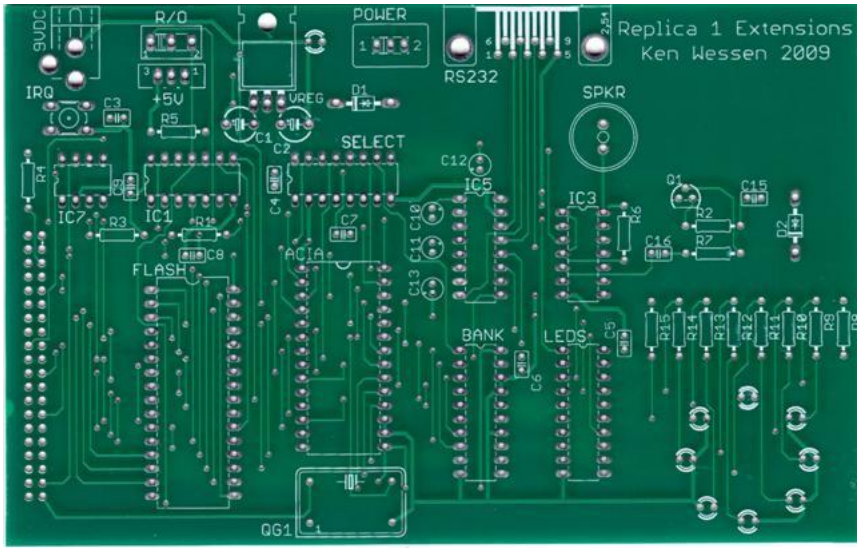
ACIA Initialisation (19.2K baud, 8 bits, no parity, 1 stop bit)	CMD .= $C202 CTRL .= $C203 LDA #$0B STA CMD LDA #$1F STA CTRL
ACIA Output code	DATA .= $C200 STATUS .= $C201

	STA DATA SEND DATA .WAIT LDA STATUS LOAD STATUS AND #$10 CHECK BIT 4 BEQ .WAIT ACIA DONE?
ACIA Input code	DATA .= $C200 STATUS .= $C201 .WAIT LDA STATUS LOAD STATUS AND #$08 CHECK BIT 3 BEQ .WAIT NO CHAR YET LDA DATA LOAD CHAR

The other 74LS377 provides the memory for the flash memory bank selection, with the 5 low order bits of any value written to memory location $C100 being mapped to the 5 high order bits of the AT29C040A. Since the block size for this chip is only 256 bytes, there is no great overhead to implementing a simple file system and supporting read-write access. However, for now I am burning it externally and using it in read mode only. For the future, a read only switch is provided for data security.

The remaining parts of the circuit include a debounced IRQ button (this also needs some changes to the mini-monitor in KRUSADER ROM to properly handle the interrupt), a MAX702 to provide the auto-reset on startup functionality, and an independent 5V regulator circuit with power switch and LED.

Pictures



The Replica 1 extension circuit board

A completed extension board

Back to KRUSADER or Computer Collection