EhBASIC for the 2m5 modular SBC with emulated 6502

EhBASIC is a BASIC interpreter for the 65xx family of CPUs written by Lee Davison.

The copy distributed with this document has the necessary modifications to run it on the emulated 6502 SBC. Most changes are in the min_mon.asm file to handle terminal IO, load & save of basic programs and the front end to deal with interrupts. It holds the constants to define RAM usage and IO-vectors. The predefined source and the machine loadable image ehbasic.hex are configured for the default IO-vectors and 32k of SRAM. It was assembled using the Kowalski simulator: http://exifpro.com/utils.html

Two changes were made to the basic.asm file. The RAM scan was removed because EhBASIC itself is running from RAM. RAM must now be configured in the min_mon.asm file. During a cold start of BASIC you may still configure less RAM to reserve some for other purposes. The second change should better accommodate interrupts by controlling the interrupt disable bit in the processor status.

Terminal IO

The emulated ACIA of the SBC is driven by vectors in the min_mon.asm file. During input a delete character is converted to backspace. Any escape sequence sent by the terminal is discarded. During output the backspace character is converted to an erasing backspace.

If you want to download programs as ASCII text to EhBASIC, you should enable the emulator XON/XOFF flow control feature by setting the required watermarks (flowlo=128, flowhi=192). I have made tests with PuTTY and Tera Term. Both work well when the content of the clipboard is dropped (right click) and they use a built in COM port of the PC.

With a USB to serial adapter there is a bigger latency from XOFF until the hardware buffer is exhausted. In Tera Term I could get it to work with smaller flow control settings (flowlo=64, flowhi=128), but PuTTY would still loose portions of the text in longer programs.

As a remedy you should enter a NEW command in EhBASIC before downloading any program and make sure, that the line numbers of the program are in ascending order. Then EhBASIC digests the program faster and it works even without flow control and with a USB adapter.

LOAD and SAVE

LOAD <file#>
SAVE <file#>

For these commands to work an I²C EEPROM must be connected to the SBC. The same numbering system (0-$fe) is used as for saved machine code like EhBASIC itself. To better distinguish a basic program from a machine code program, you should use a separate range of numbers for them. However, attempting to load the wrong format in either EhBASIC or the SBC monitor throws an error message.

A special file number of $ff or 255 exits EhBASIC to the SBCs' monitor, where you can use the EEPROM utility commands to delete programs, view a directory list or set autoload options. Return to EhBASIC with the X command.

Interrupts

The interrupt handling capacity of EhBASIC on my SBC emulator with a ~ 2MHz emulated 6502 is limited. As a rule of thumb I would recommend no more than 20 interrupts per second. You could set up a timer that trips once a second and program a wall clock with it. You can definitely not read a continuous byte stream fast enough from an ACIA with > 600 Baud in an EhBASIC interrupt service routine for RDRF. Whenever the expected interrupt rate is high, think about using a machine code module intercepting the IRQ or NMI vector.

In addition some EhBASIC commands are not suitable to be used with interrupts. The INPUT statement would wait until you hit <CR> and EhBASIC can not handle interrupts until than. Use a loop with a GET statement instead. During a WAIT, interrupts can not be processed until the condition is met. Use DO:LOOP UNTIL instead. Not to mention, that stopping a program in an interrupt driven environment is not a good idea.

I have included my BASIC test program to test the interrupt system of the SBCs' internal interrupt devices as an example of how to utilize interrupts in EhBASIC.

Automating the start of an EhBASIC program during power on

The same way as automating the start of a machine code program with the SBCs' monitor program using the EA command, you can add a set of commands to be run at the start up of EhBASIC.

Configure autoload:

```
1FD D240 00 x00 y09 NV-BDIZC H>EA01"C\rLOAD 6\rRUN\r"
EEPROM autoload 01 "C\rLOAD 6\rRUN\r"
```

At power on you will see:

```
N6502 Emulator V0.81  built Mar 02 2013
Program 01 loaded

6502 EhBASIC [C]old/[W]arm ?

Memory size ?

19711 Bytes free

Enhanced BASIC 2.22

Ready
LOAD 6

Program 06 loaded

Ready
RUN
```