
Subject: Newbie with Tiny68K

Posted by [smp](#) on Fri, 26 Jan 2018 16:57:29 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hello all,

I'm new here - I recently purchased a Tiny68K board from Bill Shen, and while I was waiting to be admitted here onto the Forum, I've been e-mailing with Bill. I'd like to make a few posts to recount my experiences for the benefit of others.

My board arrived safe & sound, and came up right away. I'm having difficulty with the USB-to-TTL converter trying to get CTS & RTS working. The cables I already had were all only 3-wire connections, with no CTS & RTS. Those all have worked fine for the terminal interface with CTS & RTS jumper together. Next, I tried one from eBay, but it only worked for the TX & RX, but not for CTS & RTS. I see that I screwed up and got a 3.3V version, so I am hopeful that my next try from Amazon will work. That one should arrive today. I'll give it a try and report back.

Thanks for listening!

smp

Subject: Re: Newbie with Tiny68K

Posted by [smp](#) on Fri, 26 Jan 2018 16:58:53 GMT

[View Forum Message](#) <> [Reply to Message](#)

Now that I've posted my first message, hopefully my links will now work...

I'm having difficulty with the USB-to-TTL converter trying to get CTS & RTS working. The cables I already had were all only 3-wire connections, with no CTS & RTS. Those all have worked fine for the terminal interface with CTS & RTS jumper together. Next, I tried this, from eBay:

https://www.ebay.com/itm/FTDI-FT232RL-USB-Cable-to-Serial-adaptmodule-USB-TO-TTL-RS232-Arduino-Cable-PR/302571041695?ssPageName=STRK%3AMEBIDX%3AIT&_trksid=p2060353.m2749.l2649

Again, this works for the TX & RX, but not for CTS & RTS. I see that I screwed up and got a 3.3V version, so I am hopeful that my next try will work:

https://www.amazon.com/gp/product/B010KII6VG/ref=oh_aui_detailpage_o00_s00?ie=UTF8&psc=1

This unit should arrive today. I'll give it a try and report back.

Thanks for listening!

smp

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Fri, 26 Jan 2018 17:09:13 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi all,

I have the source code for a 68K Tiny Basic that is configured to work on an early Motorola trainer. The code is well documented and indicates the portion that would need to be modified for another system:

```
*****
* The following routines are the only ones that need *
* to be changed for a different I/O environment.   *
*****

*
* ===== Output character to the console (Port 1) from register D0
* (Preserves all registers.)
*
OUTC BTST #1,$10040 is port 1 ready for a character?
BEQ OUTC if not, wait for it
MOVE.B D0,$10042 out it goes.
RTS

*
* ===== Input a character from the console into register D0 (or
* return Zero status if there's no character available).
*
INC BTST #0,$10040 is character ready?
BEQ INCRET if not, return Zero status
MOVE.B $10042,D0 else get the character
AND.B #$7F,D0 zero out the high bit
INCRET RTS

*
* ===== Output character to the host (Port 2) from register D0
* (Preserves all registers.)
*
AUXOUT BTST #1,$10041 is port 2 ready for a character?
BEQ AUXOUT if not, wait for it
MOVE.B D0,$10043 out it goes.
RTS

*
* ===== Input a character from the host into register D0 (or
* return Zero status if there's no character available).
*
AUXIN BTST #0,$10041 is character ready?
BEQ AXIRET if not, return Zero status
```

```

MOVE.B $10043,D0 else get the character
AND.B #$7F,D0 zero out the high bit
AXIRET RTS

```

```

*
* ===== Return to the resident monitor, operating system, etc.
*

```

```

BYEBYE MOVE.B #228,D7 return to Tutor
TRAP #14

```

Since I am not yet a 68K programmer, I asked Bill what changes I need to make to the OUTC and INC routines to connect properly to the Tiny68K Monitor.

Bill replied with this:

Tiny68K uses the trap services of EASy68K. So for console input it is task 5, console output is task 6 and console input pending is task 7. You can also write directly to the 68681 registers, but the advantage of trap is you can simulate your code with EASy68K simulator.

The EASy68K compatible replacement routine for OUTC & INC:

OUTC:

```

movem.l d0-d1,-(sp) * save reg d0,d1
move.b d0,d1 * d1 contains the character to send
move.b #6,d0 * console output trap service
trap #15
movem.l (sp)+,d0-d1 * restore register
rts

```

INC:

```

movem.l d1,-(sp) * save reg d1
move.b #0,d1 * assuming no input pending
move.b #7,d0 * console ready?
trap #15
move.b d1,d0 * if zero, no input pending, exit
beq done
move.b #5,d0 * otherwise, get data into d1.b
trap #15
move.b d1,d0 * transfer to d0.b

```

done:

```

move.l (sp)+,d1 * restore d1
rts

```

There are no equivalent EASy68K trap services with the auxiliary port, so replace the host I/O with direct write to 68681 registers:

(I'm assuming the serial port B is already properly initialized...)

```

SRB equ $FFF013 * status reg of serial port B (read)

```

```
RHRB equ $FFF017 * Rx holding reg of serial port B (read)
THRb equ $FFF017 * Tx holding reg of serial port B (write)
```

AUXOUT:

```
btst.b #2,SRB * wait for transmit ready
beq AUXOUT
move.b d0,THRb * put char out to serial port B
rts
```

AUXIN:

```
clr.b d0 * assuming no char input ready
btst.b #0,SRB * does receiver has a char input?
beq hostdone * no input pending, exit with d0.b = zero
move.b RHRb,d0 * get character into d0.b
```

hostdone:

```
rts
```

As you can see the trap services are less efficient, but having the ability to simulate code is worth the overhead.

Bill

With those I/O changes, I was able to get the Tiny Basic to start running, but quite a few more problems have arisen. Stay tuned!

smp

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Fri, 26 Jan 2018 17:15:09 GMT
[View Forum Message](#) <> [Reply to Message](#)

Continuing my effort to get the Tiny Basic to run properly, I re-ORG'd the code to 4000H, and then it was apparently working. At this point, all characters that I type are echoed twice, but the BASIC is receiving the correct entries, and acting upon them. I tried BYE, NEW, and PRINT 1+2, and they all worked appropriately.

It appears that the code is sensitive to where it resides in memory, for some reason. From the beginning, I saw that the author expected the code to be in page 0, because he starts right out assuming the top of memory is only 16 bits:

*

* Modifiable system constants:

*

```
TXTBGN DC.L TXT beginning of program memory
ENDMEM DC.L $FFFE end of available memory
```

*

* The main interpreter starts here:

*

```
CSTART MOVE.L ENDMEM,SP initialize stack pointer
LEA INITMSG,A6 tell who we are
BSR.L PRMMSG
MOVE.L TXTBGN,TXTUNF init. end-of-program pointer
MOVE.L ENDMEM,D0 get address of end of memory
```

That's why I've tried to stay in page 0. ORG at 4000H and top of memory at FFFEh seems to be working at this point, so, hopefully I don't have to crawl through all the code looking for the 16 bit restrictions and opening them up.

Thanks again for listening!

smp

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Fri, 26 Jan 2018 17:26:28 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi again all,

I decided that the Tiny68K monitor TRAP must echo characters, so I found the one line in Tiny Basic to comment out within the GETLN routine. Double character echo fixed.

It gets a bit bizarre after that. It appears that everything in the Tiny Basic works in direct input mode. It also will let me enter a program, line by line, and LIST will print it all back to me. So far, so good. Then things start to get funny. After typing in one of my favorite test programs, and listing it out, the RUN command hangs. Rats!

I've spent a couple of hours skulking through the code, and nothing obvious (to me, anyway) stands out. I read the reprinted article from Dr. Dobbs, and that shed some light:

"One warning: the DIRECT and EXEC routines were written assuming that the interpreter itself would be somewhere in the first 64K of memory (\$0 to \$FFFF). If you move it above 64K, you will have to modify the EXEC routine and check the rest of the code carefully to make sure the addressing modes are correct."

So, the author admits to coding this to run only in page 0, and he doesn't know all the bits that will have to be changed to run in a higher page of memory.

I've played around quite a bit with getting the Basic loaded and then executing one or more direct commands, and typing in a program and try to run it. Most all of this playing around results with the RUN command hanging up. I can't recount all the different combinations that I tried, but I finally found this:

Reset the Tiny68K board.
Load the Tiny Basic s-record file.
Type in your program - don't do anything else but type in your program.
Type RUN.

This sequence will get your program to run successfully (if you typed everything in the program correctly).

The next LIST or RUN or PRINT command will hang!

I have Tera Term, so it's pretty easy to load the s-record file to get Tiny Basic running, and then modify the character and line delays to send in my program from a text file. That's how I've been able to try dozens of things and get to this point. I'm starting to doubt the robustness of the Tiny Basic code. Since I am a novice programmer on the 68K, I'd love to have you folks take a look. Maybe someone with more experience programming a 68000 will do better than I have on figuring this out.

Thanks for listening!

smp

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Fri, 26 Jan 2018 17:34:18 GMT
[View Forum Message](#) <> [Reply to Message](#)

Here are the files for your inspection.

tbi68k12.ZIP is the original package I downloaded from the Internet.

TinyBasic68K.X68 is the original source code file with changes only to make it assemble without errors using EASy68K.

TinyBasic68Ksmp.X68 is the file with all my modifications in it, that is operating as described previously.

Folks, I'm interested in getting this Tiny Basic to work correctly on the Tiny68K board. I would greatly appreciate your inspection of what I've done here, and whatever observations, comments or ideas you may have will be very welcome!

smp

File Attachments

- 1) [tbi68k12.zip](#), downloaded 334 times
 - 2) [TinyBasic68K.X68](#), downloaded 442 times
 - 3) [TinyBasic68Ksmp.X68](#), downloaded 441 times
-
-

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Fri, 26 Jan 2018 21:33:36 GMT
[View Forum Message](#) <> [Reply to Message](#)

smp wrote on Fri, 26 January 2018 11:58 I'm having difficulty with the USB-to-TTL converter trying to get CTS & RTS working...

https://www.amazon.com/gp/product/B010KII6VG/ref=oh_aui_detailpage_o00_s00?ie=UTF8&psc=1

This unit should arrive today. I'll give it a try and report back.

My cable arrived today, and it did not work for me at all. SIGH. It is now on it's way back to Amazon for a refund. This is starting to get frustrating. My TX & RX USB-to-TTL cables work fine. The 3.3V cable that I purchased by mistake works for TX & RX, but the CTS & RTS don't seem to be working. Now this \$20 cable that had good reviews doesn't work for me at all.

I have a good working USB-to-RS232 interface - maybe I'll look for a RS232-to-TTL converter to attach to that.

Thanks for listening.

smp

Subject: Re: Newbie with Tiny68K
Posted by [plasmo](#) on Sat, 27 Jan 2018 00:06:08 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Stephen,
Welcome to the retrobrew forum.

I tried the TinyBasic68ksmp.X68 you uploaded and I believe have observed the same problem:

```
100 PRINT 1
RUN <-- run OK with correct answer
LIST <-- hang here
```

I do have a serial connection with working handshake, so the problem is not due to the handshake problem. I like to tackle the TinyBasic problem first. Perhaps other users may have answer to the serial port issue over the weekend. If not, I'll ship you a CP2102 adapter Monday. So let me see if I can figure out the TinyBasic...

Bill

Subject: Re: Newbie with Tiny68K
Posted by [plasmo](#) on Sat, 27 Jan 2018 01:29:40 GMT
[View Forum Message](#) <> [Reply to Message](#)

Stephen,

The way the INC routine deals with character-not-ready is unusual. So instead of fighting it and making it works with EASy68K trap services, I just do the direct I/O method. You can see the I/O routines are similar to that of the auxiliary port.

I restored the echo routine you've commented out (BSR GOOUT).

Search for the string **X, those are my modifications to your TinyBasic68Ksmp.X68

I'm not familiar with TinyBasic, but here is my quick test after the modification:

Gordo's MC68000 Tiny BASIC, v1.3
modified to operate on the Tiny68K board by Bill Shen

```
OK
> PRINT 23
    23
```

```
OK
> 100 PRINT 1
> 200 PRINT 2
> 300 PRINT 1+2
> RUN
    1
    2
    3
```

```
OK
> LIST
  100 PRINT 1
  200 PRINT 2
  300 PRINT 1+2
```

```
OK
> RUN
    1
    2
    3
```

```
OK
>
```

File Attachments

1) [TinyBasic68Ksmphcs.X68](#), downloaded 426 times

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Sat, 27 Jan 2018 13:42:25 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Bill,

Well, that certainly turned out to be kind of anticlimactic, didn't it?

Thank you for your attention on this. I'll be trying out that direct I/O code later on today, and I'll try and put the Tiny Basic through some more exhaustive testing.

It remains a mystery to me why the I/O code was the culprit causing the hang up like that...

smp

Subject: Re: Newbie with Tiny68K
Posted by [mikemac](#) on Sat, 27 Jan 2018 16:03:47 GMT
[View Forum Message](#) <> [Reply to Message](#)

smp wrote on Sat, 27 January 2018 06:42Hi Bill,

Well, that certainly turned out to be kind of anticlimactic, didn't it?

Thank you for your attention on this. I'll be trying out that direct I/O code later on today, and I'll try and put the Tiny Basic through some more exhaustive testing.

It remains a mystery to me why the I/O code was the culprit causing the hang up like that...

smp

Basic interpreters often expect the keyboard routine to return even if there is no key pressed. They use that so you can interrupt the running of a program, otherwise you'd have to reset the whole board to break out of the program. I wouldn't be surprised if the routine you tried to use waits for a key to be pressed, thereby hanging the Basic interpreter.

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Sat, 27 Jan 2018 18:12:45 GMT
[View Forum Message](#) <> [Reply to Message](#)

mikemac wrote on Sat, 27 January 2018 11:03 Basic interpreters often expect the keyboard routine to return even if there is no key pressed. They use that so you can interrupt the running of a program, otherwise you'd have to reset the whole board to break out of the program. I wouldn't be surprised if the routine you tried to use waits for a key to be pressed, thereby hanging the Basic interpreter.

Hi Mike, thanks for your thought.

I agree with what you said, but the mystery for me is why would the TRAP service routines for I/O allow one direct command, like PRINT SIZE, and then hang on the second direct command? Or, why would the TRAP service routines allow me to enter a program line by line, and then RUN the program, and then hang on the next direct command? If it were as you said, then wouldn't the TRAP service routines cause a hang right away?

Oh, well, it doesn't really matter, I suppose. I'm just sorry that a new-to-me way of performing I/O didn't work out in this case.

smp

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Sat, 27 Jan 2018 18:59:04 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello again, all,

I've had the Tiny BASIC up and running for a few hours and it seems to be operating as advertised, once I put in Bill's changes to the I/O routines. So, for anyone who's interested, the code that Bill uploaded here is available for anyone who may want to run Tiny BASIC on their system.

One of my test programs that I use across my vintage machines is a routine to find all the ways to make \$1.00 (US) from change (pennies, nickels, dimes & quarters). Here is the code I wrote for the 68K Tiny BASIC::

```
10 C=0
20 FOR P=0 TO 100 STEP 5
30 FOR N=0 TO 20
40 FOR D=0 TO 10
50 FOR Q=0 TO 4
60 IF P+(N*5)+(D*10)+(Q*25)<>100 GOTO 90
70 PRINT #3,"p=",P," n=",N," d=",D," q=",Q
80 C=C+1
90 NEXT Q
100 NEXT D
110 NEXT N
120 NEXT P
130 PRINT
```

140 PRINT C," ways to make \$1.00 from change"
150 PRINT

With this program, I've "benchmarked" a variety of machines I have, for the fun of seeing how fast or slow they might be. Even the differences between BASICs on the same machine. I was a bit surprised about the difference between the Tiny BASIC and the CB68 compiling BASIC on CP/M 68K:

Tiny BASIC: 2 minutes 6 seconds
CB68: 4 minutes 6 seconds

My timing was by eye on a running digital clock, so the times are probably +/- a second or so, but, wow, I didn't expect 2 minutes difference between the two. The reason probably lies in CB68 performing floating point arithmetic, and Tiny BASIC performing integer arithmetic. I've taken measurements on several BASICs on several machines, and I have results running from 30 seconds to 26 minutes 21 seconds. So, the 8 MHz 68000 here is performing in the better-than-average category, as far as I'm concerned.

I hope that others will try out the Tiny BASIC for themselves. The original .ZIP file has a documentation file, as well as a few programs to tryout.

smp

Subject: Re: Newbie with Tiny68K
Posted by [plasmo](#) on Sat, 27 Jan 2018 20:42:09 GMT
[View Forum Message](#) <> [Reply to Message](#)

Looks like TinyBasic is running well. Benchmarks are interesting. Here is another benchmark of various basic interpreter/compiler running on different machines based on one program:
https://www.retrobrewcomputers.org/forum/index.php?t=msg&th=201&goto=3306&#msg_3306

I'll try your program on the few machines I have.

Subject: Re: Newbie with Tiny68K
Posted by [plasmo](#) on Sat, 27 Jan 2018 23:33:13 GMT
[View Forum Message](#) <> [Reply to Message](#)

Of course you can speed up performance by putting in a faster CPU with the corresponding oscillator. As a standard procedure to establish performance margin of every Tiny68K I built, I replace the 8MHz oscillator in every Tiny68K with 10 Mhz and 12 MHz oscillator and run memory diagnostics. All board will pass the 10MHz and 3 out of 4 will pass 12MHz. So your board will run at 10Mhz and most likely 12MHz. If you replace the CPU with 12Mhz part (MC68000P12 or MCHC68000P12), it is likely to run at 16MHz. The DRAM is fast enough to running at zero wait

state at 12MHz but according to my paper calculation a wait state is needed at 16MHz, depending on the DRAM speed grade. I know I've ran Tiny68K at 16MHz with P12 part before. I tried it again just now with a MC68000P12 and 60nS DRAM SIMM, and it works just fine.

In term of availability, you may find DIP64 part at 12MHz, but DIP64@16MHz is hard to find. I'd be leery of the 16MHz part as well, it may be a slow part re-labelled so I'd only buy from reputable sellers. Faster 68000 are more likely packaged in the PLCC format. I have redesigned Tiny68K with PLCC package like the one in the picture and I generally run it at 16MHz (it is a 12MHz 68HC000 if you look closely).

File Attachments

1) [DSC_33310127_F.jpg](#), downloaded 501 times

Subject: Re: Newbie with Tiny68K

Posted by [plasmo](#) on Sun, 28 Jan 2018 03:35:39 GMT

[View Forum Message](#) <> [Reply to Message](#)

smp,

I figured out why the trap services are not working. My mistake!

Just like Mikemac said that console input does not want to spinning forever waiting for input, so the INC routine checks for input, if no input returns to calling program with Zero flag set. My mistake is when restoring registers just before returning to the calling program, I used MOVE.L instruction which affects the condition flags. What I should use is MOVEM.L instruction which does not change the condition flags.

So attached is the TinyBasic using EASy68K trap services. My changes are marked with **X and they are:

1. turn off echo with task #12
2. restore the echo back routine you commented out (BSR GOOUT)
3. get rid of one unnecessary line in INC routine
4. change MOVE.L to MOVEM.L just before return

File Attachments

1) [TinyBasic68Ksmphcs2.X68](#), downloaded 401 times

Subject: Re: Newbie with Tiny68K

Posted by [smp](#) on Sun, 28 Jan 2018 13:43:26 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi Bill,

Thanks very much for the speed upgrade info, and also for your effort to get the original TRAP

service I/O to work.

smp

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Sun, 28 Jan 2018 13:47:49 GMT
[View Forum Message](#) <> [Reply to Message](#)

For the benefit of anyone who is interested, here is a complete set of files for the Tiny BASIC with the direct I/O routines.

TinyBasic68Ksmp.X68 - assembler source text file
TinyBasic68Ksmp.S68 - s-record file ready to load
TinyBasic68Ksmp.L68 - assembly listing text file

All these files were generated from the EASy68K assembler application.

Enjoy!

smp

File Attachments

- 1) [TinyBasic68Ksmp.L68](#), downloaded 355 times
- 2) [TinyBasic68Ksmp.S68](#), downloaded 380 times
- 3) [TinyBasic68Ksmp.X68](#), downloaded 359 times

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Mon, 29 Jan 2018 19:56:05 GMT
[View Forum Message](#) <> [Reply to Message](#)

plasma wrote on Sat, 27 January 2018 18:33Of course you can speed up performance by putting in a faster CPU with the corresponding oscillator. <...snip...>

I must be doing something wrong. I always try to find parts at several places like Jameco, DigiKey and Mouser. Sometimes I have to experiment to determine how they describe the part that I'm looking for. I determined that I'm looking for a 12 MHz half-can oscillator. I've also tried half-can crystal oscillator, and half-DIP oscillator. I can find some half-can oscillators, but none at 12 MHz or 16 MHz.

Can anyone give me a pointer to a place that I can purchase the required 12 MHz and 16 MHz half-can oscillators to experiment with my board?

Thanks in advance, for any pointers or links that you may be able to provide.

smp

Subject: Re: Newbie with Tiny68K
Posted by [Andrew B](#) on Mon, 29 Jan 2018 20:03:41 GMT
[View Forum Message](#) <> [Reply to Message](#)

Try these P/Ns (both as 5V parts, standard pinout):
ECS-2100AX-120
ECS-2100AX-160

Digi-Key has them both.

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Mon, 29 Jan 2018 20:13:46 GMT
[View Forum Message](#) <> [Reply to Message](#)

Andrew B wrote on Mon, 29 January 2018 15:03Try these P/Ns (both as 5V parts, standard pinout):
ECS-2100AX-120
ECS-2100AX-160

Digi-Key has them both.

Thanks a million, Andrew! I now have one of each on order.

smp

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Tue, 30 Jan 2018 16:47:46 GMT
[View Forum Message](#) <> [Reply to Message](#)

smp wrote on Mon, 29 January 2018 15:13Andrew B wrote on Mon, 29 January 2018 15:03Try these P/Ns (both as 5V parts, standard pinout):
ECS-2100AX-120
ECS-2100AX-160

Digi-Key has them both.

Thanks a million, Andrew! I now have one of each on order.

smp

Along with those two oscillators, I've also purchased this:

https://www.ebay.com/itm/1x-MOTOROLA-MC68000P12F-16MHZ-DIP-6-4-16-32-BIT-16Mbyte-ADDRESS-RANGE/112636890643?ssPageName=STRK%3AMEBIDX%3AIT&_trksid=p2060353.m2749.l2649

This eBay Seller has good feedback, so I thought it might be worth the risk. I'll report back on how it works (or not).

smp

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Tue, 30 Jan 2018 16:53:34 GMT
[View Forum Message](#) <> [Reply to Message](#)

smp wrote on Fri, 26 January 2018 16:33smp wrote on Fri, 26 January 2018 11:58 I'm having difficulty with the USB-to-TTL converter trying to get CTS & RTS working...

https://www.amazon.com/gp/product/B010KII6VG/ref=oh_aui_detailpage_o00_s00?ie=UTF8&psc=1

This unit should arrive today. I'll give it a try and report back.

My cable arrived today, and it did not work for me at all. SIGH. It is now on it's way back to Amazon for a refund. This is starting to get frustrating. My TX & RX USB-to-TTL cables work fine. The 3.3V cable that I purchased by mistake works for TX & RX, but the CTS & RTS don't seem to be working. Now this \$20 cable that had good reviews doesn't work for me at all.

I have a good working USB-to-RS232 interface - maybe I'll look for a RS232-to-TTL converter to attach to that.

Thanks for listening.

smp

Just for the record, I am still dead in the water when it comes to hitching up a working SIO connection to my terminal that also has CTS & RTS. I can only use the cables I already have for TX & RX only with CTS & RTS jumper together on the Tiny68K board.

I'm not the brightest when it comes to serial connections. I also have a good USB-to-RS232 cable. Can anyone provide me a step-by-step procedure to acquire a RS232-to-TTL adapter and get it wired up properly to provide the needed CTS & RTS? I would greatly appreciate some

detailed advice with pictures if possible.

Thanks in advance!

smp

Subject: Re: Newbie with Tiny68K
Posted by [plasmo](#) on Tue, 30 Jan 2018 17:49:27 GMT
[View Forum Message](#) <> [Reply to Message](#)

smp,

I corresponded via email with another user who also experienced RTS/CTS handshake problem. He was using the FTDI brand of USB adapter. He received a CP2102 USB adapter over the past weekend and the handshake problem is now resolved. I think the FTDI brand of USB adapters have problems for some (myself included).

I have a spare CP2102 adapter that I wired up with an adapter board so they plug straight into the 6-pin Tiny68K serial port without additional wiring. I can mail it to you if you like. You'll need a USB 2.0 extension cable (not USB A-to-B which has a square end) to connect this.

Your board should run at 12 MHz without changing the 68000. When changing the 68000 you need a tool with a bend like this:

<https://www.ebay.com/itm/New-2-Pairs-Cross-Lock-Tweezers-Self-Closing-Fiber-Grip-4-Bent-Straight-tip/321808761311>, slide the tool under the IC and gently pry up the 68000. Move forward a little bit and pry up again and repeat until the chip comes clean off. When inserting a new 68000, the row spacing of a new part is too wide for the socket, so you need to bend each row against a hard surface to get the proper row spacing. Align them carefully with the sockets before press it down. Aligning 2 long rows of pins can be tricky. Take your time.

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Tue, 30 Jan 2018 23:08:12 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Bill,

I found these CP2102 adapters. Is any of these what I need?

<https://www.amazon.com/KEDSUM-CP2102-Module-Download-Converter/dp/B009T2ZR6W>
https://www.amazon.com/WINGONEER-CP2102-Module-Serial-Converter/dp/B01LRVQIFQ/ref=pd_sbs_147_1?_encoding=UTF8&pd_rd_i=B01LRVQIFQ&pd_rd_r=ZTQV432JGPYBX20PWDFB&pd_rd_w=a8wBp&pd_rd_wg=uoJGW&psc=1&refRID=ZTQV432JGPYBX20PWDFB

<https://www.amazon.com/CP2102-UART-6PIN-Serial-Converter/dp/B00CD264HG>

Can you please tell me how you get the RTS signal instead of the DTR signal off the board? Actually, I think the WINGONEER one has the RTS hole labeled. Perhaps that's the one I ought to get? I can solder a post into the RTS hole and then use the jumper cables to attach to the Tiny68K board, and the USB extension cable to attach to my computer. Right?

Thanks very much for any advice you can offer.

smp

Subject: Re: Newbie with Tiny68K

Posted by [smp](#) on Tue, 30 Jan 2018 23:36:21 GMT

[View Forum Message](#) <> [Reply to Message](#)

plasma wrote on Tue, 30 January 2018 12:49 Your board should run at 12 MHz without changing the 68000. When changing the 68000 you need a tool with a bend like this:
<https://www.ebay.com/itm/New-2-Pairs-Cross-Lock-Tweezers-Self-Closing-Fiber-Grip-4-Bent-Straight-tip/321808761311>, slide the tool under the IC and gently pry up the 68000. Move forward a little bit and pry up again and repeat until the chip comes clean off. When inserting a new 68000, the row spacing of a new part is too wide for the socket, so you need to bend each row against a hard surface to get the proper row spacing. Align them carefully with the sockets before press it down. Aligning 2 long rows of pins can be tricky. Take your time.

Hi again, Bill, and thanks very much for your advice on the speedup process.

I plan to first try the 12 MHz oscillator. I had hoped that the P8 processor would maybe work. Thanks for verifying that it will.

I had trouble with the link you provided, but I found this:

<https://www.amazon.com/TWEEZERS-POINTED-CLOSING-SOLDERING-NO-VELTOOLS/dp/B010EAQQFS>

Is this what I need?

Thanks a million for all your attention and advice!

smp

Subject: Re: Newbie with Tiny68K

Posted by [plasma](#) on Wed, 31 Jan 2018 01:34:25 GMT

[View Forum Message](#) <> [Reply to Message](#)

This is the picture of the tweezers I used to remove 68000. You need a way of getting under the device and lift up gently and repeat the process. A small spoon may work as well.

This is the CP2102 board and how I wired in the RTS and CTS. The RTS, CTS are not present on the 6-pin header so I need to build a small adapter board and bring the signals out from the pc board. I think this one from amazon is like mine.

https://www.amazon.com/Honbay-CP2102-Module-Download-Converter/dp/B01A0BOGHG/ref=sr_1_9?s=electronics&ie=UTF8&qid=1517362187&sr=1-9&keywords=cp2102+usb

File Attachments

- 1) [DSC_33330130.jpg](#), downloaded 1227 times
 - 2) [DSC_33350130_F.jpg](#), downloaded 368 times
 - 3) [DSC_33340130_F.jpg](#), downloaded 372 times
-

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Wed, 31 Jan 2018 01:49:16 GMT
[View Forum Message](#) <> [Reply to Message](#)

plasm0 wrote on Tue, 30 January 2018 20:34 This is the picture of the tweezers I used to remove 68000. You need a way of getting under the device and lift up gently and repeat the process. A small spoon may work as well. <...snip...>

This is the CP2102 board and how I wired in the RTS and CTS. The RTS, CTS are not present on the 6-pin header so I need to build a small adapter board and bring the signals out from the pc board. I think this one from amazon is like mine. <...snip...>

Thanks very much, Bill. I think I have the picture now. One of your pictures shows that there is a silk screen on the underneath of the board - that explains how the description of some of these boards indicates that all the signals are clearly marked, even though in the picture shown they certainly are not.

I'm going to go ahead and order the CP2102 from Amazon. As far as the tool to lift the 68000 out of the socket goes, I understand what you are saying, and I'll see what implement I have that can do the job. I get it - slow and careful will get the job done.

Thanks again!

smp

Subject: Re: Newbie with Tiny68K
Posted by [googlefish](#) on Wed, 31 Jan 2018 08:44:19 GMT

[View Forum Message](#) <> [Reply to Message](#)

I'm the other guy who had problems with CTS/RTS signals and the FTDI adapter.

Now I have two working adapters from amazon:

WINGONEER USB 2.0 to TTL UART 6PIN CP2102 Module Serial Converter

Diymore 2Pcs CP2102 USB 2.0 to TTL UART Module 6-Pin Serial Converter STC Replace FT232 Module

both are working.

They have slight different pin layout, but it's documented on the board.

I use 1:1 connection with these 5 pins, nothing more:

GND

Txd

RxD

CTS

RTS

thanks

Stephan

P.S. sorry no real links, I'm not yet allowed to send them.

Subject: Re: Newbie with Tiny68K

Posted by [plasmo](#) on Thu, 01 Feb 2018 01:54:54 GMT

[View Forum Message](#) <> [Reply to Message](#)

Thanks for the list of CP2102 that worked for you. This is the link to Diymore CP2102:

https://www.amazon.com/Diymore-CP2102-Module-Converter-Repla ce/dp/B0755DBGJM/ref=sr_1_1?ie=UTF8&qid=1517449844&s r=8-1&keywords=Diymore+2Pcs+CP2102+USB+2.0+to+TTL+UART+M odule+6-Pin+Serial+Converter+STC+Replace+FT232+Module

2 pcs for \$6.88 and free shipping. They look just like my CP2102.

Subject: Re: Newbie with Tiny68K

Posted by [adx](#) on Thu, 01 Feb 2018 02:21:44 GMT

[View Forum Message](#) <> [Reply to Message](#)

I was successful with an FTDI based adapter that I have from Sparkfun. It's this model

<https://www.sparkfun.com/products/9716>. I also have a known genuine ftdi based cable at work that I'll try once I find it. I wonder if the people who are having difficulties with them have ran into the counterfeit chips.

Subject: Re: Newbie with Tiny68K
Posted by [plasmo](#) on Thu, 01 Feb 2018 05:20:53 GMT
[View Forum Message](#) <> [Reply to Message](#)

adx,
What is your computer's operating system? I'm running Windows Vista and I know the latest FTDI driver does not support Windows Vista, so I'm using an older driver.
Including myself there are 4 users with FTDI RTS/CTS handshake problems. Two bought the adapters from China, the other two are from Amazon. The problem MAY be due to counterfeit parts, but the distribution channels are pretty far apart. I guess I should contact FTDI tech supports to see if they know about this issue.

Subject: Re: Newbie with Tiny68K
Posted by [adx](#) on Thu, 01 Feb 2018 11:59:15 GMT
[View Forum Message](#) <> [Reply to Message](#)

I'm using OS X. It could be a driver issue on Vista.

Subject: Re: Newbie with Tiny68K
Posted by [googlefish](#) on Thu, 01 Feb 2018 13:55:18 GMT
[View Forum Message](#) <> [Reply to Message](#)

hm, I'm using OS X too.
Have you tried GKermit?
That was my problem with the FTDI cable.

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Thu, 01 Feb 2018 16:29:57 GMT
[View Forum Message](#) <> [Reply to Message](#)

I have two different CP2102 units on order:

https://www.ebay.com/itm/Black-5V-CP2102-USB-to-TTL-Serial-Adapter-Module-for-LilyPad-Arduino-Pro-Mini/362018308267?ssPageName=STRK%3AMEBIDX%3AIT&_trksid=p2060353.m2749.l2649
https://www.amazon.com/gp/product/B01LRVQIFQ/ref=od_aui_detailpages00?ie=UTF8&psc=1

The second one is the WINGONEER that matches with Stephan, I think. Since I also am using Mac OSX, I am hopeful this will do the trick for me. The ability to use GKermit with the Tiny68K board is pretty critical.

Along with those, I have a number of bits & bobs coming along. I'll continue to post about my experiences, as soon as I have them.

smp

Subject: Re: Newbie with Tiny68K
Posted by [adx](#) on Thu, 01 Feb 2018 17:01:20 GMT
[View Forum Message](#) <> [Reply to Message](#)

Quote:
hm, I'm using OS X too.
Have you tried GKermit?
That was my problem with the FTDI cable.

I don't think I'd waste my time trying gkermit. You're just as likely to hit tty layer incompatibilities has hardware issues. I use minicom installed via Homebrew. Another option would be to try out picocom. It's super light weight.

Subject: Re: Newbie with Tiny68K
Posted by [googlefish](#) on Thu, 01 Feb 2018 17:02:29 GMT
[View Forum Message](#) <> [Reply to Message](#)

I have both and both are working.
Little different pinlayout but clearly printed on the board.

I'm sure it works for you, too.

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Thu, 01 Feb 2018 18:24:51 GMT
[View Forum Message](#) <> [Reply to Message](#)

adx wrote on Thu, 01 February 2018 12:01 I don't think I'd waste my time trying gkermit. You're just as likely to hit tty layer incompatibilities has hardware issues. I use minicom installed via Homebrew. Another option would be to try out picocom. It's super light weight.

Hi, and thanks very much for your thoughts.

To this point, the only way I see to get anything other than a plain text file into CP/M-68K is via GKermit. Can you elaborate a little on your suggestion of Minicom and Picocom? I Googled a little bit, but I have not yet been able to find out any information. Are these CP/M-68K applications that you can somehow get into the filesystem on the Tiny68K board? I would definitely like to know more about these.

Thanks!

smp

Subject: Re: Newbie with Tiny68K
Posted by [plasmo](#) on Thu, 01 Feb 2018 18:48:27 GMT
[View Forum Message](#) <> [Reply to Message](#)

The main reason Tiny68K needs RTS/CTS handshakes is to work with gkermit. The software is already installed so all you need is type "gkermit -r" to receive files from PC or "gkermit -t filename" to send filename to PC. I'll be interested in alternative file transfer methods as well because if we don't need gkermit, then I can remove the RTS/CTS requirements which has caused more problems than all other issues combined.

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Thu, 01 Feb 2018 19:59:39 GMT
[View Forum Message](#) <> [Reply to Message](#)

smp wrote on Thu, 01 February 2018 13:24<...snip...>To this point, the only way I see to get anything other than a plain text file into CP/M-68K is via GKermit. Can you elaborate a little on your suggestion of Minicom and Picocom? I Googled a little bit, but I have not yet been able to find out any information. Are these CP/M-68K applications that you can somehow get into the filesystem on the Tiny68K board? I would definitely like to know more about these.<...snip...>

More Googling got me the information that both Minicom and Picocom are minimalist terminal emulation programs that run on Linux. They seem to be good substitutes for Hyper Terminal or Tera Term. They do not substitute for the CP/M application that can send and receive files from inside the CP/M-68K filesystem.

smp

Subject: Re: Newbie with Tiny68K
Posted by [rhkoolstar](#) on Thu, 01 Feb 2018 20:32:09 GMT
[View Forum Message](#) <> [Reply to Message](#)

Maybe I am missing something, but looking at the schematics and the pcb, the tiny68k terminal is only 4 wires. Rx, Tx, VCC and Gnd.

I'm not surprised when CTS and RTS don't work, they seem not not be wired up at all.

Edit disregard, I was looking at the rev 0 board.

Rienk

Subject: Re: Newbie with Tiny68K

Posted by [smp](#) on Thu, 01 Feb 2018 21:26:33 GMT

[View Forum Message](#) <> [Reply to Message](#)

rhkoolstar wrote on Thu, 01 February 2018 15:32 Maybe I am missing something, but looking at the schematics and the pcb, the tiny68k terminal is only 4 wires. Rx, Tx, VCC and Gnd.

I'm not surprised when CTS and RTS don't work, they seem not not be wired up at all.

Rienk

On the schematic, check out the 68681. You'll see that Rx & Tx go to the terminal connector pins T7 & T8. You'll also see that OP0 goes to pin T6 (RTS) and IP0 goes to pin T5 (CTS). Finally, +5V & GND go to pins T9 & T10. The terminal connector pins are pointed out in one of Bill's photos.

I can attest that not connecting anything to CTS & RTS means that the Tiny68K board will hang, waiting for RTS. Only if you connect CTS & RTS together will the board start up properly. Of course, if I had a device with the proper handling of CTS & RTS, that would work, too.

smp

Subject: Re: Newbie with Tiny68K

Posted by [adx](#) on Thu, 01 Feb 2018 21:34:46 GMT

[View Forum Message](#) <> [Reply to Message](#)

plasma wrote on Thu, 01 February 2018 10:48 The main reason Tiny68K needs RTS/CTS handshakes is to work with gkermit. The software is already installed so all you need is type "gkermit -r" to receive files from PC or "gkermit -t filename" to send filename to PC. I'll be interested in alternative file transfer methods as well because if we don't need gkermit, then I can remove the RTS/CTS requirements which has caused more problems than all other issues combined.

Some of the other CP/M ROMs have XMODEM installed to do the file transfers. That's always been easier to use than Kermit.

Subject: Re: Newbie with Tiny68K

Posted by [smp](#) on Thu, 01 Feb 2018 21:40:03 GMT

[View Forum Message](#) <> [Reply to Message](#)

adx wrote on Thu, 01 February 2018 16:34 <...snip> Some of the other CP/M ROMs have XMODEM installed to do the file transfers. That's always been easier to use than Kermit.

Ha! I've never seen CP/M in a ROM. I have seen the XMODEM application added into CP/M file systems, but it was always added in after the fact. It never came standard with any CP/M file system. Do you know of an XMODEM application coded for the 68K, or a source file that can be

assembled for the 68K?

Thanks!

smp

Subject: Re: Newbie with Tiny68K
Posted by [rhkoolstar](#) on Thu, 01 Feb 2018 23:01:14 GMT
[View Forum Message](#) <> [Reply to Message](#)

CP/M in ROM? try Epson PX-8. No XMODEM, but TERM and FILINK

Back to serial communications. You might suffer from framing errors, when the clocks are not equal enough. You can try adding an extra stopbit to the sender, to give the receiver more time to catch up.

Rienk

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Fri, 02 Feb 2018 21:42:11 GMT
[View Forum Message](#) <> [Reply to Message](#)

OMG it's working!!!

I received my WINGONEER CP2102 device today. I set it up and managed to get it to work kinda with Minicom on my Macintosh. Then I tried a free trial of a Macintosh application called Serial - and, holy cow! it's working! I have been able to send and receive files using GKERMIT. YAY!

I'll spend some time fooling around, but eventually I want to pull in a Forth-83 application and get it running.

Thanks for listening to all my whining, and thanks very much for all your advice and assistance!

smp

Subject: Re: Newbie with Tiny68K
Posted by [plasmo](#) on Sat, 03 Feb 2018 05:09:11 GMT
[View Forum Message](#) <> [Reply to Message](#)

Excellent!

I think I'll recommend the CP2102 USB-to-serial adapters for Tiny68K from now on.

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Sat, 03 Feb 2018 13:53:39 GMT
[View Forum Message](#) <> [Reply to Message](#)

Well, here's the rest of the story:

I tried it first on my Windows XP machine running on Parallels in my Mac. No go. Windows could not find drivers for the device.

Second, I tried directly on my Mac. As I said, I had to try a couple of Serial terminal options, but the device was recognized right off.

Certainly for Mac users, the CP2102 is the right way to go. For Windows, I don't have the full story.

smp

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Sat, 03 Feb 2018 19:36:41 GMT
[View Forum Message](#) <> [Reply to Message](#)

Can anyone tell me what the memory map for CP/M-68K is ?

Now that I finally can get programs into my machine, I cleared disk B: and loaded in Forth83 from Gaby's CP/M site. The disappointment came when I tried to execute F83, and I get an "insufficient memory or bad file header" message. For the time being, I am betting on "insufficient memory" being the error because the CP/M memory map is constrained to 64K, or something small like that.

Thanks!

smp

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Sat, 03 Feb 2018 20:00:50 GMT
[View Forum Message](#) <> [Reply to Message](#)

I'm looking at the CP/M-68K System Guide, and when I execute the command:

```
A>SIZE68 CP/M.REL
```

it returns:

```
cpm.rel:26284+2786+16076=45146 (B05A ) stack size = 0
```

The System Guide says this about the SIZE68 CP/M.REL command:

This gives you the total size of the system in both decimal and hex byte counts. Subtract this number from the highest memory address in your system and add one to get the highest possible

address at which CPM.REL can be relocated.
The System Guide goes on to say:
Assuming that the result is aaaaaa, type this command:

```
A>RELOC -Baaaaaa CPM.REL CPM.SYS
```

The result is the CPM.SYS file, relocated to load at memory address aaaaaa. If you want CPM.SYS to reside at some other memory address, such as immediately above the exception vector area, you can use RELOC to place the system at that address.

When you perform the relocation, verify that the resulting system does not overlap the TPA as defined in the BIOS. The boundaries of the system are determined by taking the relocation address of CPM.SYS as the base, and adding the size of the system (use SIZE68 on CPM.SYS) to get the upper bound. This address range must not overlap the TPA that the BIOS defines in the Memory Region Table.

So, I see the Transient Program Area (TPA) must be defined in the BIOS. That's where I'll go looking next...

smp

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Sat, 03 Feb 2018 20:36:53 GMT
[View Forum Message](#) <> [Reply to Message](#)

I found this in the R5 BIOS:
memrgn: dc.w 1 * 1 memory region
 dc.l \$20000 * starts above CP/M
 dc.l \$6e0000 * goes until \$7000000. RAM drive above it
* dc.l \$dd0000 * goes until \$df0000, TPA is ~14meg bytes
Bill, can you please decipher this for me? I can see that the ~14 MB TPA line is commented out. I also remember that you have set up disk drive E: as a large RAM drive. I cannot understand where the TPA is or what size it is by this table.

Thanks very much in advance for your assistance!

smp

Subject: Re: Newbie with Tiny68K
Posted by [plasmo](#) on Sat, 03 Feb 2018 21:07:25 GMT
[View Forum Message](#) <> [Reply to Message](#)

I think you have a bad header file with f83.68k. I downloaded forth 83 from cpm.z80.de, unzip it, gkermit them all to CF disk and execute f83.68k and have the same error message as you did. I'm looking for a relocatable executable such as f83.rel. This is because f83.68k may need to

reside at different memory to run correctly.

The TPA for Tiny68K is quite large. The size of memrgn (i.e., TPA) is 0x6E0000 or 7208960 bytes (over 7 meg). You can confirm it with the utility that you can download from Gaby's site, TPALEN

<http://cpm.z80.de/download/util68k.zip>

Subject: Re: Newbie with Tiny68K

Posted by [smp](#) on Sat, 03 Feb 2018 21:11:20 GMT

[View Forum Message](#) <> [Reply to Message](#)

plasm0 wrote on Sat, 03 February 2018 16:07: I think you have a bad header file with f83.68k. I downloaded forth 83 from cpm.z80.de, unzip it, gkermit them all to hard disk and execute f83.68k and have the same error message as you did. I'm looking for a relocatable executable such as f83.rel. This is because f83.68k may need to reside at different memory to run correctly.

The TPA for Tiny68K is quite large. The size of memrgn (i.e., TPA) is 0x6E0000 or 7208960 bytes (over 7 meg). You can confirm it with the utility that you can download from Gaby's site, TPALEN

<http://cpm.z80.de/download/util68k.zip>

RATS! I was hoping that it was not the bad header file. If you find another Forth83, please let me know. Thanks very much for the information and the pointer, Bill!

smp

Subject: Re: Newbie with Tiny68K

Posted by [lowen](#) on Sat, 03 Feb 2018 21:12:07 GMT

[View Forum Message](#) <> [Reply to Message](#)

smp wrote on Thu, 01 February 2018 16:40...

Ha! I've never seen CP/M in a ROM.

Well, the CPU280 boots CP/M 3 out of EPROM, all the way to CCP. The ROM isn't accessible as a disk device, but it does have CP/M 3 in ROM....

Quote: I have seen the XMODEM application added into CP/M file systems, but it was always added in after the fact. It never came standard with any CP/M file system. Do you know of an XMODEM application coded for the 68K, or a source file that can be assembled for the 68K?

Well, CP/M 68K has a C compiler, and it might not be too hard to get an rzsz derivative to build and run. I don't think YAM was ever ported, but that was as I recall the gold standard for CP/M X/YMODEM-supporting terminal programs back in the day. But rzsz will do everything, X/Y/ZMODEM, and it's simple to use.

I think Kermit was used simply because there is a binary already available from Gaby and others. Kermit is very portable, too.

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Sat, 03 Feb 2018 22:00:37 GMT
[View Forum Message](#) <> [Reply to Message](#)

plasm0 wrote on Sat, 03 February 2018 16:07<...snip...>
The TPA for Tiny68K is quite large. The size of memrgn (i.e., TPA) is 0x6E0000 or 7208960 bytes (over 7 meg). You can confirm it with the utility that you can download from Gaby's site, TPALEN
<http://cpm.z80.de/download/util68k.zip>

I found those utilities already on my CP/M file system. Thanks!

I managed to get the TPALEN file assembled, and it verifies what you say:
TPA LENGHT = 7208960 BYTS (7040 KB)
It even looks like an old spelling error has been preserved over the years!

smp

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Sat, 03 Feb 2018 22:25:45 GMT
[View Forum Message](#) <> [Reply to Message](#)

lowen wrote on Sat, 03 February 2018 16:12smp wrote on Thu, 01 February 2018 16:40...
Ha! I've never seen CP/M in a ROM.

Well, the CPU280 boots CP/M 3 out of EPROM, all the way to CCP. The ROM isn't accessible as a disk device, but it does have CP/M 3 in ROM....

Quote:I have seen the XMODEM application added into CP/M file systems, but it was always added in after the fact. It never came standard with any CP/M file system. Do you know of an XMODEM application coded for the 68K, or a source file that can be assembled for the 68K?

Well, CP/M 68K has a C compiler, and it might not be too hard to get an rzsZ derivative to build and run. I don't think YAM was ever ported, but that was as I recall the gold standard for CP/M X/YMODEM-supporting terminal programs back in the day. But rzsZ will do everything, X/Y/ZMODEM, and it's simple to use.

I think Kermit was used simply because there is a binary already available from Gaby and others. Kermit is very portable, too.

Hi Lowen, and thanks very much for your info.

Both you and rhkoolstar have now pointed out systems with a version of CP/M in ROM. Thanks to you both. I guess that even though I have never seen one, they do exist out there.

Yes, if source files can be found for C code or 68K assembly, those files can easily be brought into the CP/M-68K file system for use. As you know, X/Y/ZMODEM/Kermit is critical for importing already compiled/assembled applications. I'm not sure why Kermit is so often found. Perhaps simply because it was earlier than the others? I like the GKermit here on the CP/M-68K because it is the stripped down version made to perform send & receive only, and not all the other bloat often found in the other versions. Kermit tried to be all things to all users, and so I find many different versions across many different machines quite confusing at times. Do I have to set the baud rate? Do I have to indicate that I'm doing binary files? Etc., sometimes for several settings. Oh, well, since I finally have the correct interface device to get GKermit running I'm in good shape now, but I would love to try another method to see if we can get away from needing to use CTS & RTS in the terminal I/O. If you find some source code to try, please share it here!

smp

Subject: Re: Newbie with Tiny68K
Posted by [b1ackmai1er](#) on Sun, 04 Feb 2018 13:06:52 GMT
[View Forum Message](#) <> [Reply to Message](#)

smp,

Have you seen Lee Davison's(RIP) 68K EHBASIC?

<https://github.com/jefftranter/68000/tree/master/ehbasic>

Regards Phil.

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Sun, 04 Feb 2018 13:49:02 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Phil,

Thanks very much for the pointer. This has been suggested to me a couple of times, but I've been too lazy to go through the effort of getting the required assembler, etc., etc. Maybe one of these days.

Thanks!

smp

Subject: Re: Newbie with Tiny68K
Posted by [plasm0](#) on Sun, 04 Feb 2018 14:58:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

I've already ported Lee Davison's EhBasic to Tiny68K. It was written for EASy68K so used the same EASy68K trap services so the porting was trivia (org to 0x4000 and turn off interrupts). EhBasic is relocatable, so you can move it anywhere in memory without recompiling. I ran your make-\$1-from-changes BASIC program and it took 90 seconds to complete (I have to change line 70 because EhBasic didn't understand 'PRINT #3'). You may recall that it took TinyBASIC 2 minutes to complete.

The program ported to Tiny68K is attached below, search for "***X" for my changes to the original code.

File Attachments

1) [EhBasic.zip](#), downloaded 279 times

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Sun, 04 Feb 2018 16:52:29 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hey, thanks a million, Bill! Greatly appreciated. Cool to have another BASIC in my arsenal.

smp

UPDATE: WHOO HOO! This is my 35th message, so now I'm a Member, rather than a Junior Member. Yay!

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Sun, 04 Feb 2018 18:56:20 GMT
[View Forum Message](#) <> [Reply to Message](#)

Today, I've been playing around with EhBASIC. I also decided to try out my 12 MHz and 16 MHz oscillators.

Here are my eyeball times for my DOLLAR.BAS program with the three different oscillators installed:

8 MHz	12 MHz	16 MHz
-----	-----	-----
1:45	1:11	0:53

The easiest check on this performance is to see that at 8 MHz the run took 105 seconds and at 16

MHz the run took 53 seconds, pretty much right on at half the time.

This is excellent. I am still using the MC68000P8 that came installed with my board. I have not yet dared to pull the processor and try the MC68000P12F 16 MHz chip I purchased on eBay. Maybe another time when I have my courage up.

Is there any risk to keep using the MC68000P8 at 16 MHz? It seems to me no, but I'd like to hear other opinions on this.

Thanks!

smp

UPDATE: I also tried out the ASCIIART.BAS program at 16 MHz - only 43 seconds!

Subject: Re: Newbie with Tiny68K
Posted by [plasmo](#) on Sun, 04 Feb 2018 19:44:21 GMT
[View Forum Message](#) <> [Reply to Message](#)

I'm surprised that -P8 part can run at 16 MHz. Now, I've never tried with Tiny68K so I have no empirical data to show how unusual that is. At 16Mhz the DRAM access is probably out of spec, on paper, but not as far out as the 68000. The primary danger to overclocking is heat; your part is rated at 70 degree C and they are probably around 50 degree C now, so no, there are no danger. Another trick of getting more performance is to raise the voltage; 5.25V is the spec, but they can go as high as 5.5V. Parts also get hotter at higher voltage, so risk of damage increases with voltage. Put your finger on the part, 50 degree C is when you are barely able to keep your finger on the part. You can always put a fan on it if you are really worry, but I wouldn't bother.

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Sun, 04 Feb 2018 20:09:40 GMT
[View Forum Message](#) <> [Reply to Message](#)

Ah, yes, heat. I've been checking this afternoon, and the 68000 and the Altera device, and the memory are all just barely warm to the touch. I have a 5V @ 2A wall wart supply, and my 5V is reading 4.94V. I'm not going to go for a higher voltage at this point. I was just trying to see what the board can do with a few component changes. If the DRAM is now out of spec, that will probably bother my conscience, but until I see any real flakey stuff going on, I'll probably stay with 16 MHz for the time being.

Fun stuff! Please let me know if you know of any other 68K Forth available anywhere out there. Or any other interesting programs I ought to be trying.

I found the original reference manual for the 6502 EhBASIC, and I extracted the language reference pages in the attached document, if anyone is interested.

smp

File Attachments

1) [EhBASIC_Reference.pdf](#), downloaded 327 times

Subject: Re: Newbie with Tiny68K

Posted by [norwestrzh](#) on Mon, 05 Feb 2018 00:02:10 GMT

[View Forum Message](#) <> [Reply to Message](#)

> Please let me know if you know of any other 68K Forth available anywhere out there.

Have you tried: <http://home.earthlink.net/~schulldw/cpm68/simulator.html>

There is a F83 in his simulator disk that is pretty nice -- big vocabulary.

Roger

P.S. It is fairly common to be able to overclock quality Motorola 68k CPU's. I've seen 8 MHz parts go 10 or 12, but I've never seen an 8 MHz part run at twice the nominal clock!

Subject: Re: Newbie with Tiny68K

Posted by [plasmo](#) on Mon, 05 Feb 2018 15:18:17 GMT

[View Forum Message](#) <> [Reply to Message](#)

Also take a look at CPM68K in <http://www.retroarchive.org/cpm/cdrom/CPM/CPM68K/>

zamp has written up instruction on how to unsqueeze the files:

[https://www.retrobrewcomputers.org/forum/index.php?t=msg&](https://www.retrobrewcomputers.org/forum/index.php?t=msg&th=222&goto=3757&#msg_3757)

[;th=222&goto=3757&#msg_3757](https://www.retrobrewcomputers.org/forum/index.php?t=msg&th=222&goto=3757&#msg_3757)

I tried the F83V2-68.LBR. It unsqueeze OK, but still won't execute on Tiny68K

Subject: Re: Newbie with Tiny68K

Posted by [lowen](#) on Mon, 05 Feb 2018 16:38:33 GMT

[View Forum Message](#) <> [Reply to Message](#)

[norwestrzh](#) wrote on Sun, 04 February 2018 19:02>...

P.S. It is fairly common to be able to overclock quality Motorola 68k CPU's. I've seen 8 MHz

parts go 10 or 12, but I've never seen an 8 MHz part run at twice the nominal clock!

A really fun read about the 68K overclock situation can be found here. Summary: 32MHz bursts with the 12F part, wired-up with a DIP64 and PC board with PLCC. Later he gets 38MHz with the PLCC. His page is a fascinating read on what he had to do to get different things working, doing clock speed switching during bus cycles, etc.

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Mon, 05 Feb 2018 21:26:51 GMT
[View Forum Message](#) <> [Reply to Message](#)

plasma wrote on Mon, 05 February 2018 10:18 Also take a look at CPM68K in
<http://www.retroarchive.org/cpm/cdrom/CPM/CPM68K/>

zamp has written up instruction on how to unsqueeze the files:

https://www.retrobrewcomputers.org/forum/index.php?t=msg& ;amp ;amp ;amp ;amp ;amp ;th=222&goto=3757&#msg_3757

I tried the F83V2-68.LBR. It unsqueeze OK, but still won't execute on Tiny68K

Hi Bill,

Thanks for the pointers. I downloaded that stuff. I see that the files in F83v2-68.LBR are almost all files that have been compacted by other methods, but apparently not the F83.68K file. I extracted it using LU, and I verify that I got the same "Insufficient memory or bad file header" error message.

I also downloaded UTILS.LBR. Here is a listing of whats in it:

-C > -o utils

lu: old library file UTILS.LBR has 32 entries, 0 free

-O > -l

Name Type Start Length CRC

```
=====
DIRECTORY      0  8 0000
CONFIG .HQ      8  5 A504
FROMHEX .CQ    13 12 AD04
HEX .MQN       25  8 EA44
IBMREAD .6QK   33 158 2584
IBMREAD .CQ   191 36 F859
IBMREAD .MQN   227  8 9C92
INDENT .CQ    235 164 DA99
INDENT .DQC   399 29 5402
INDNTCOD.HQ    428  8 D5C5
INDNTGLO.HQ   436 28 C97C
IO .CQ        464 62 CE6F
KILLDU .6QK   526 142 256D
```

```
KILLDU .C      668  5 D331
KILLDU .MAN    673  4 80E2
LEXI  .CQ     677  58 933A
LKINDENT.SUB   735  1 2276
PARSE .CQ     736  51 9EEC
PRCOMMEN.CQ   787  48 5C4C
PTOC  .6QK    835 133 A46B
PTOC  .CQ     968  21 E984
PTOC  .MQN    989  7 6FF4
SORTFL .6QK   996 149 6DC9
SORTFL .C     1145  6 3EC9
SORTFL .MAN   1151  3 289A
TOHEX .C     1154  9 2C76
UTFILES .SQB  1163  6 D37E
WEED  .MAN   1169  3 F0D7
WEED  .SUB   1172  1 CDA4
XDIR  .CQ   1173  76 53C4
XDIR  .MQN   1249 11 9D78
XDIRLINK.SUB  1260  1 379D
```

```
=====
Used sectors      1261
Deleted sectors   0
Total sectors     1261
Active entries = 32, deleted = 0, free = 0, total = 32
-L > -c
```

lu: old library UTILS.LBR closed

The files on retroarchive.org are going to be difficult to retrieve if there are a number of unpacking methods that have to be used. It appears that we will need additional tools besides the LU tool. See how the file extensions have been changed from 68K to 6QK, and from SUB to SQB, and from H to HQ, and from C to CQ, etc. Some other file compacter will be needed to unpack those files after they've been extracted from the LBR file.

Does anyone have any advice to offer on this? Are those additional tools actually in the XLU68K.LBR file? Or, will the -eu command get the job done from the LU tool? I definitely need another tutorial like the one that zamp posted for how to use the LU tool!

Thanks for listening!

smp

UPDATE: BTW, I'm doing all of this with my 16 MHz oscillator still installed. Everything seems to be operating OK.

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Mon, 05 Feb 2018 22:12:00 GMT

[View Forum Message](#) <> [Reply to Message](#)

norwestrzh wrote on Sun, 04 February 2018 19:02> Please let me know if you know of any other 68K Forth available anywhere out there.

Have you tried: <http://home.earthlink.net/~schultdw/cpm68/simulator.html>

There is a F83 in his simulator disk that is pretty nice -- big vocabulary.

Roger

P.S. It is fairly common to be able to overclock quality Motorola 68k CPU's. I've seen 8 MHz parts go 10 or 12, but I've never seen an 8 MHz part run at twice the nominal clock!

Hi Roger, and thanks very much for this pointer.

I am not a very competent 68K guy at this point, so please bear with me. It appears to me that the F83 you refer to is probably in the file DISKC.CPM.FS. Is there any way to extract it from that file, short of make-ing the simulator and then manually pulling the file out somehow? I'm not very familiar with running simulators at all, so I have no experience here.

Thanks in advance for your advice!

smp

Subject: Re: Newbie with Tiny68K
Posted by [plasmo](#) on Tue, 06 Feb 2018 12:09:06 GMT

[View Forum Message](#) <> [Reply to Message](#)

The -eu command in LU.68K should take care of the extract & unsqueezing. So all you need to do is open the file (-o utils), extract & unsqueezing (-eu *.*), and exit (-x). You should see all the files extracted.

This is the DRAM diagnostic I used to test out the 16 meg memory. It should take about 3 minutes to run each iteration at 16MHz. It will repeat the test with different seed values indefinitely. To test the serial port, you can use gkermit to send a file to PC and read it back and compare the two files using fcomp. To test the CF, you can initialize drive E (RAMdrive) and pipe e:=d:*.*[v] (copy with verify). This suite of tests should provide pretty good fault coverage and give you more confidence in running at 16MHz.

File Attachments

1) [tstDRAM2.zip](#), downloaded 261 times

Subject: Re: Newbie with Tiny68K

Posted by [smp](#) on Tue, 06 Feb 2018 14:44:14 GMT

[View Forum Message](#) <> [Reply to Message](#)

plasm0 wrote on Tue, 06 February 2018 07:09The -eu command in LU.68K should take care of the extract & unsqueezing. So all you need to do is open the file (-o utils), extract & unsqueezing (-eu *.*), and exit (-x). You should see all the files extracted.

This is the DRAM diagnostic I used to test out the 16 meg memory. It should take about 3 minutes to run each iteration at 16MHz. It will repeat the test with different seed values indefinitely. To test the serial port, you can use gkermit to send a file to PC and read it back and compare the two files using fcomp. To test the CF, you can initialize drive E (RAMdrive) and pip e:=d:.*[v] (copy with verify). This suite of tests should provide pretty good fault coverage and give you more confidence in running at 16MHz.

Thanks a million for all your help and advice, Bill!

I am running the memory test now - one iteration has completed successfully already. I have been using GKermit, mostly pulling things into the Tiny68K board, but I've gone both ways without any hiccups. After a while on the memory test, I'll check the CF, too. This is great. Thanks very much.

Thanks also for the advice on the LU tool operation. While I am not new to vintage computers, I haven't used some of these tools before and it's nice to hear that I'm not wandering in the weeds too badly.

smp

Subject: Re: Newbie with Tiny68K

Posted by [norwestrzh](#) on Tue, 06 Feb 2018 18:10:29 GMT

[View Forum Message](#) <> [Reply to Message](#)

>> It appears to me that the F83 you refer to is probably in the file DISKC.CPM.FS. Is there any way

>> to extract it from that file, short of make-ing the simulator and then manually pulling the file
>> out somehow? I'm not very familiar with running simulators at all, so I have no experience here.

Sure!! Are you familiar with cpmtools?? Do you use it??? Easy to extract it that way.

It is a two part application: F83.68k loads F83.BIN. If you don't use cpmtools or if it is too much of an effort to build and use it, I can make you copies -- either binary, or S-record. Need a way of getting the two files to you.

Roger

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Tue, 06 Feb 2018 19:19:57 GMT
[View Forum Message](#) <> [Reply to Message](#)

norwestrzh wrote on Tue, 06 February 2018 13:10>> It appears to me that the F83 you refer to is probably in the file DISKC.CPM.FS. Is there any way
>> to extract it from that file, short of make-ing the simulator and then manually pulling the file
>> out somehow? I'm not very familiar with running simulators at all, so I have no experience here.

Sure!! Are you familiar with cpmttools?? Do you use it??? Easy to extract it that way.

It is a two part application: F83.68k loads F83.BIN. If you don't use cpmttools or if it is too much of an effort to build and use it, I can make you copies -- either binary, or S-record. Need a way of getting the two files to you.

Roger

Hi again, Roger,

I obtained a win32 copy of the cpmttools. when I run CPMLS on the file DISKC.CPM.FS, I get a 0: followed by about 30 blank lines, and then I'm back at the command prompt. I assume that's the directory listing for user 0, but all blank lines? Huh?

As you can see, I'm not familiar with what I should be doing here. If you would be so kind as to provide me copies of f83.68k and f83.bin when you have some spare time, I would greatly appreciate that. I will use GKermite to pull these files into CP/M on my system, so binary files would be fine. Can you post them here, so anyone else who's interested can also benefit?

Thanks very much!

smp

Subject: Re: Newbie with Tiny68K
Posted by [rhkoolstar](#) on Tue, 06 Feb 2018 19:29:49 GMT
[View Forum Message](#) <> [Reply to Message](#)

You are probably using the wrong disk format with CPMTTOOLS (-f format)
Look in the diskdefs file and pick one that fits the geometry of the disk you are using
Or make one that fits.

Rienk

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Tue, 06 Feb 2018 21:24:03 GMT
[View Forum Message](#) <> [Reply to Message](#)

plasma wrote on Tue, 06 February 2018 07:09 ... This is the DRAM diagnostic I used to test out the 16 meg memory. It should take about 3 minutes to run each iteration at 16MHz. It will repeat the test with different seed values indefinitely. ...

I tested the DRAM for 6+ hours. All OK. I guess that gives me some confidence in my 16 MHz setup!

smp

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Tue, 06 Feb 2018 21:27:02 GMT
[View Forum Message](#) <> [Reply to Message](#)

rhkoolstar wrote on Tue, 06 February 2018 14:29 You are probably using the wrong disk format with CPMTTOOLS (-f format)
Look in the diskdefs file and pick one that fits the geometry of the disk you are using
Or make one that fits.

Rienk

Hi Rienk,

Thanks for your advice.

The web site for the 68K CP/M simulator:

<http://home.earthlink.net/~schultdw/cpm68/simulator.html>

does not indicate what the disk image format is (at least as far as I can tell anyway). I'll try some of the diskdefs types and see if I can make anything out.

smp

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Tue, 06 Feb 2018 22:13:32 GMT
[View Forum Message](#) <> [Reply to Message](#)

plasma wrote on Tue, 06 February 2018 07:09 ... To test the CF, you can initialize drive E

(RAMdrive) and pip e:=d:*.*[v] (copy with verify). This suite of tests should provide pretty good fault coverage and give you more confidence in running at 16MHz.

OK! I performed the CF test as described a few times with no errors reported.

It sure looks to me like I have a pretty solid 16 MHz Tiny68K system! Do you think perhaps the MC68000P8 is actually a faster part labelled as a P8? I know that chip makers sometimes do things like that for efficiency. Since the 16 MHz parts are being made, rather than change production lines, just label the faster parts with the slower label?

I don't know, but I'm certainly happy that my board is running at 16 MHz and seems to be nice and stable. That was a pretty cheap upgrade, too. The 16 MHz oscillator cost \$2.47 (plus mailing) at Digi-Key.

smp

Subject: Re: Newbie with Tiny68K
Posted by [rhkoolstar](#) on Tue, 06 Feb 2018 22:47:36 GMT
[View Forum Message](#) <> [Reply to Message](#)

To find the disk format, check the Disk parameter block in BIOS

Subject: Re: Newbie with Tiny68K
Posted by [norwestrzh](#) on Tue, 06 Feb 2018 23:24:48 GMT
[View Forum Message](#) <> [Reply to Message](#)

Here's the disk definition that worked for me (for the 68k simulator file system):

```
diskdef sim
  seclen 128
  tracks 512
  sectrk 256
  blocksize 2048
  maxdir 4096
  skew 0
  boottrk 1
  os 2.2
end
```

Roger

Subject: Re: Newbie with Tiny68K
Posted by [norwestrzh](#) on Tue, 06 Feb 2018 23:41:40 GMT
[View Forum Message](#) <> [Reply to Message](#)

>> Can you post them here, so anyone else who's interested can also benefit?

Just in case cpmttools fails you, here is a file containing the S-records for both parts of F83.

See the "SEND68" command at the beginning of each part for the proper file names.

Roger

File Attachments

1) [f83.68k.sr](#), downloaded 338 times

Subject: Re: Newbie with Tiny68K
Posted by [plasmo](#) on Wed, 07 Feb 2018 00:29:38 GMT
[View Forum Message](#) <> [Reply to Message](#)

smp wrote on Tue, 06 February 2018 15:13plasmo wrote on Tue, 06 February 2018 07:09 ... To test the CF, you can initialize drive E (RAMdrive) and pip e:=d:*.[v] (copy with verify). This suite of tests should provide pretty good fault coverage and give you more confidence in running at 16MHz.

OK! I performed the CF test as described a few times with no errors reported.

It sure looks to me like I have a pretty solid 16 MHz Tiny68K system! Do you think perhaps the MC68000P8 is actually a faster part labelled as a P8? I know that chip makers sometimes do things like that for efficiency. Since the 16 MHz parts are being made, rather than change production lines, just label the faster parts with the slower label?

I don't know, but I'm certainly happy that my board is running at 16 MHz and seems to be nice and stable. That was a pretty cheap upgrade, too. The 16 MHz oscillator cost \$2.47 (plus mailing) at Digi-Key.

smp

I certainly don't expect your board to run at 16MHz. In my standard margin test I only test to 12MHz and never even thought about trying 16MHz because I don't expect it to pass. Now you've got me curious and I tried 16MHz on a couple of my boards. One passed and other failed. Hmm, that's pretty interesting. Now I wonder just how close to the edge the 16MHz board is. One way to find out is to raise the clock by 10% and run through the test suite or lower the voltage 5% and see if it still pass the tests. If so, I'll continue to operate it at 16MHz. If not, I'd roll back the clock by 10% and run at 14.3MHz or chose to run at 16MHz anyway. We are not talking about mission critical component so running it up close to the edge is just for fun.

Subject: Re: Newbie with Tiny68K
Posted by [plasmo](#) on Wed, 07 Feb 2018 04:09:48 GMT
[View Forum Message](#) <> [Reply to Message](#)

smp,

To transfer the Srecords that norwestrzh provided, you need a program called SRIN. You can find it here:

https://www.retrobrewcomputers.org/forum/index.php?t=msg&th=201&goto=3332&#msg_3332

Since cpmtools is so useful, you should learn how to use it to transfer files in and out of cpm disk image. Here is how I would use cpmtools to extract files from diskc.cpm.fs image:

copy the disk definition that norwestrzh provided:

```
diskdef sim
seclen 128
tracks 512
sectrk 256
blocksize 2048
maxdir 4096
skew 0
boottrk 1
os 2.2
end
```

and append to file 'diskdefs' in cpmtools. diskdefs is collection of various Disk Parameter Block. copy the simulation disk image, diskc.cpm.fs, to the cpmtools directory execute this two commands:

```
cpmcp -f sim diskc.cpm.fs 0:f83.bin f83.bin <--(copy user 0 file named 'f83.bin' in disk image 'diskc.cpm.fs' of the format 'sim' to 'f83.bin')
cpmcp -f sim diskc.cpm.fs 0:f83.68k f83.68k <--(copy user 0 file named 'f83.68k' in disk image 'diskc.cpm.fs' of the format 'sim' to 'f83.68k')
```

Now, you should see two files named f83.68k and f83.bin in cpmtools directory. You can gkermit them to Tiny68K. Once they are transferred to Tiny68K, type 'f83' and you should see this message:

```
68000 Forth 83 Model
Version 2.1.0
Modified 03Jun84
```

You are on your own now because I know nothing of Forth

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Wed, 07 Feb 2018 18:01:38 GMT

plasma wrote on Tue, 06 February 2018 23:09 ... Since cpmtools is so useful, you should learn how to use it to transfer files in and out of cpm disk image. Here is how I would use cpmtools to extract files from diskc.cpm.fs image: ...

Hi Bill,

Thank you *very* much for the instructions. I agree that getting some experience using cpm tools will be quite beneficial. I noodled my way through it and I managed to get access to the disk image file, and then extract the two F83 files. I pulled the files into CP/M using GKermit, and I have the F83 running! I found that F83 requires several of the BLK files from before, so I extracted them (again) from the LBR file. I now have everything together on my disk B: and it all seems to be running. I have lots more experimenting to do with F83, for sure.

Thanks also to Roger and Rienk for your advice and assistance, too! You folks are all awesome, with all your advice and assistance. Being new to the 68K is interesting, as I can often get myself confused due to my past experience with 8080 & Z80 CP/M machines. I get thinking that I know what I'm doing and before I know it, I'm wandering in the weeds with these new tools that I haven't used before.

Thanks again, guys, I really appreciate you all.

smp

Subject: Re: Newbie with Tiny68K
Posted by [smp](#) on Wed, 07 Feb 2018 19:53:37 GMT
[View Forum Message](#) <> [Reply to Message](#)

I've been playing around with F83. I typed in my Forth version of my favorite test program, how many ways to make \$1 from change:

```
VARIABLE P  
VARIABLE N  
VARIABLE D  
VARIABLE Q  
VARIABLE C
```

```
: DOLLAR  
DECIMAL  
CR 0 C!  
101 0 DO  
  I P!  
  21 0 DO  
    I N!  
    11 0 DO  
      I D!
```

```

5 0 DO
  I Q!
  P @ N @ 5 * D @ 10 * Q @ 25 *
  + + + 100 =
  IF
    ." P=" P @ ." N=" N @ .
    ." D=" D @ ." Q=" Q @ . CR
    C @ 1 + C!
  THEN
  LOOP
LOOP
LOOP
5 +LOOP
CR C @ ." WAYS TO MAKE $1.00"
CR ;

```

In order to get it to fit on one screen of 16 lines, I needed to use multiple commands per line, but I printed it out here as clearly as I can, for ease of reading.

When I run this program in F83, it takes about 5 seconds. You may recall that on my system at 16 MHz, the Tiny BASIC took about 1:04 and the CP/M BASIC took about 2:04. This puts F83 at about 12-13 times faster than the Tiny BASIC version (and ~25 times faster than the CP/M BASIC). If you don't need floating point arithmetic, F83 seems to be an excellent alternative programming language.

smp

Subject: Re: Newbie with Tiny68K
 Posted by [smp](#) on Wed, 07 Feb 2018 21:51:46 GMT
[View Forum Message](#) <> [Reply to Message](#)

Here is the Sieve of Eratosthenes for F83:

```

Scr # 2      B:SMP.BLK
0 \ Sieve of Eratosthenes          07FEB18smp
1 : prime? ( n -- ? ) here + c@ 0= ;
2 : composite! ( n -- ) here + 1 swap c! ;
3 : sieve ( n -- )
4 here over erase 2
5 begin
6 2dup dup * >
7 while
8   dup prime?
9   if
10    2dup dup * do i composite! dup +loop
11  then
12  1+
13 repeat

```

14 drop cr

15 ." Primes: " 2 do i prime? if i . then loop ;

The largest integer it will handle is 32100. Anything larger than that and funny things start to happen. I assume that the size of the stack that it builds gets large enough to interfere with something else in memory.

smp

Subject: Re: Newbie with Tiny68K

Posted by [adx](#) on Thu, 08 Feb 2018 12:38:33 GMT

[View Forum Message](#) <> [Reply to Message](#)

Quote:When I run this program in F83, it takes about 5 seconds. You may recall that on my system at 16 MHz, the Tiny BASIC took about 1:04 and the CP/M BASIC took about 2:04. This puts F83 at about 12-13 times faster than the Tiny BASIC version (and ~25 times faster than the CP/M BASIC). If you don't need floating point arithmetic, F83 seems to be an excellent alternative programming language.

The wonderful thing about Forth is that entire program is going to be ran as a bunch of machine coded moves, jumps, adds, and muls basically inline with how you've coded it. Depending on the implementation you're not even having the overhead of using bsr calls. I'm willing to bet that most of the time in your code is being spent in the CP/M I/O output routines.

Subject: Re: Newbie with Tiny68K

Posted by [etchedpixels](#) on Sun, 11 Feb 2018 19:15:42 GMT

[View Forum Message](#) <> [Reply to Message](#)

31000 sounds a bit close to 32767 which is the 16bit max integer - is your forth using 16bit maths, if so you just need to write/find a bignum library

Subject: Re: Newbie with Tiny68K

Posted by [smp](#) on Sun, 11 Feb 2018 19:38:01 GMT

[View Forum Message](#) <> [Reply to Message](#)

Yes, F83 is doing 16 bit integer math. You're correct, 32767 is the maximum positive signed integer.

Since the SIEVE returns what appears to be a rational list for 30099, and starts to fail somewhere near but above that number, I figured that there may be some collision in memory where the list the the SIEVE builds grows to collide with something else.

Thanks for your thoughts.

smp

Subject: Re: Newbie with Tiny68K
Posted by [UhClem](#) on Mon, 12 Feb 2018 00:00:15 GMT
[View Forum Message](#) <> [Reply to Message](#)

F83 is a 16 bit Forth which limits its address space. There are a couple of ways to handle the addresses:

1) Use the feature of the 68000 where loading a 16 bit value into an address register triggers an automatic sign extension. This is fast but limits you to using the 32K at the bottom of the address space and 32K at the top. Assuming you have RAM at the top.

2) Load the 16 bit address into a data register and then do something to bypass the sign extension. In KERNEL68.BLK NEXT is defined as:

```
LABEL >NEXT  
  IP )+ D7 MOVE    LONG 0 D7 BP DI) W LEA  
  WORD W )+ D7 MOVE  LONG 0 D7 BP DI) JMP
```

Either way, F83 is limited to 64KB. You can use the rest of the address space in other ways but the F83 kernel and code don't know about it.

Subject: Re: Newbie with Tiny68K
Posted by [newjes250](#) on Wed, 10 Jul 2024 04:32:46 GMT
[View Forum Message](#) <> [Reply to Message](#)

To Plasm0:

I saw you describe a version of Tiny68k using PLCC 68HC000, with two CPLDs.
Have you posted anywhere about this specific version?

Always interested in your many projects.

Jesse (newjes250)

Subject: Re: Newbie with Tiny68K
Posted by [plasm0](#) on Wed, 10 Jul 2024 19:48:12 GMT
[View Forum Message](#) <> [Reply to Message](#)

Thanks for the reminder; I've actually forgotten about rev3 of Tiny68K which used PLCC 68000

and 2 CPLD on 100x100mm 2-layer pc board. Adding to existing Tiny68K, I wanted to have Ethernet (ENC28J60) and fast parallel USB interface with FT245. The board worked, but was unstable. I think it is because it was 2-layer PCB with circuitous routing and poor grounding. I have not pursued it further. If I were to do it again, I'd go with 4-layer pc board.

Bill

File Attachments

1) [DSC_33310127_F.jpg](#), downloaded 183 times
