

DOS/65 Richard Leary

[Post Reply](#)

64 posts



1

2

3

4

5

...

7

**MK14HAK**

Re: DOS/65 Richard Leary

Mon Jul 27, 2015 9:49 am

The Monitor ROM source needs a correction: keyboard = \$df00 keyboard array.

[600RevB:16K,2MHz,64x32,470,CEGMON](#)
[SuperKit:502,540B,542B,CEGMON, 8" and 5" FDDs](#)
[Cards:PE IO,6522 D-A-D, AY3-8910,ProgramGraphics,Color,UK101](#)
[WIP:HexDOS,FDD Emulator](#)

**RichLeary**

Re: DOS/65 Richard Leary

Tue Jul 28, 2015 3:01 pm

Mike - I'll take a look at that ASAP. Wonder why it works?

Right now I have a few hardware issues or tasks I am working:

- Drive B not working right. I swapped the drive with another drive and was seeing same symptoms, e. g., could not format without errors, could not copy system or data area without error. I checked the drive cable and saw some insulation breaks and conductor breaks at the drive edge connectors so I cut the cable off (it was a long cable) and crimped on two new drive edge connectors - same symptoms. It occurs to me that I did not check the floppy drive power supply so that is next step later today. A little concerned that my step rate, settling time, etc. may be off enough to be marginal.
- Video cable connection at J2 was in poor condition so I just reworked that and securely anchored the cable to the plug with lacing tape.
- RS232 cable connection at J3 is in similar shape so that is another afternoon task.
- I have a very small Murata 5V to 12V converter (isolated) with associated filter inductor and capacitor that I will RTV to the big ground plane area to provide a -12V supply for the RS232 TX circuitry.

As soon as I have the C1P/600 "fixed" and the RS232 interface working I will get back on the software side.

Thanks,

Rich





Re: DOS/65 Richard Leary

📅 Wed Jul 29, 2015 4:20 pm

An update on my fixes:

- a. Drive B working a little better but I need to test and relook at timing.
- b. Video cable connection to J2 tested and good.
- c. RS232 cable connection rework to J3 completed and tested good.
- d. Murata small 5V to -12V converter installed and tested. Rated output is 63mA, more than enough to supply the TXD output buffer even after an 1800 ohm resistor is placed across the -12V lines to meet the Murata 10% of rated load minimum load requirement.

Ultimate test of RS232 interface including the -12V power supply as used on the TXD output buffer was successful XMODEM transfer to one of my development PC's.

I still have to reinstall the C1P/600 in it's case but progress has been good.

I am nearly complete implementing RAWDUMP as a DOS/65 transient. Biggest issue is adjusting the timing loops to achieve the desired delays with my standard 0.98304 MHz system. May get that working this weekend.

Thanks,

Rich Leary



Re: DOS/65 Richard Leary

📅 Wed Aug 05, 2015 3:46 am

It has been an interesting few days with very limited time for DOS/65 and OSI C1P tasks but progress was made:

1. I have concluded that the problem with drive B was most likely invisible oxidation on some of the IC pins and most likely the PIA. A little movement of the pins in the sockets (a little bit towards out then a firm reseating) has cleared the problem. Since the hardware had not been used in 10-20 years this is expected especially for socket-IC connections.
2. The mystery of \$dc00 versus \$df00 for the keyboard has been solved. The 600/C1P board only uses A10 to A15 for address decoding of several devices including the keyboard. So while some documentation says the keyboard is at \$df00 it can be addressed at \$dc00, \$dd00, and \$de00 (and probably everything between \$dc00 and \$dfff) and will work OK. However the WinOSI emulator is coded to only accept \$df00. Mike, thanks for finding this disconnect. I have already created a V2.02 that has that change and some other more minor changes and will post it shortly. By the way - I have no recollection of why I used \$dc00 rather than \$df00 30+ years ago. Maybe I used the schematic as the address source.
3. I have a DOS/65 transient working that does the same thing as RAWDUMP but from within DOS/65. Now my question is how to get that 3x sized (compared to binary) hex file that I capture in a PC at the other end of the RS232 connection into a format that WinOSI will accept as a suitable disk image file? Suggestions?

Thanks,

Rich



dave
Site Admin



Re: DOS/65 Richard Leary

📅 Wed Aug 05, 2015 7:52 am

Hi Rich,

That's fantastic!

An easy way to convert the hex dump to a binary file is to use the free SRECORD utility to convert it. The command line would look like:

Assuming you want to convert a file called "dump.hex" to a binary file called "dump.img", you would use a command line like:

```
srec_cat dump.hex -hexdump -o dump.img -binary
```

The utility is available from <http://srecord.sourceforge.net>

Of course, it would be even better if the dump program used a more robust format, such as intel hex. It's not perfect, using a checksum rather than a CRC, but better than nothing. And by removing the spaces, the density would be comparable to--perhaps better than--a dump of 2-digit hex values separated by spaces. The SRECORD utility could still be used to produce the image file:

```
srec_cat dump.hex -intel -o dump.img -binary
```

I hope that helps

Good luck!

Dave



MK14HAK



Re: DOS/65 Richard Leary

📅 Wed Aug 05, 2015 10:01 am

Or hexbin util provided with WinOSI will convert two, hex, CR to bin. Follow it up with CLEAN for those noise bytes! Good to see your progress Rich. I have a C4 version of your boot ROM done and a diskloader version that allows Exmon useage for snooping around DOS65.

600RevB:16K,2MHz,64x32,470,CEGMON
SuperKit:502,540B,542B,CEGMON, 8" and 5" FDDs
Cards:PE IO,6522 D-A-D, AY3-8910,ProgramGraphics,Color,UK101
WIP:HexDOS,FDD Emulator



RichLeary



Re: DOS/65 Richard Leary

Thu Aug 06, 2015 4:45 am

Hmmmm! Things are not going too well. I ran my DOS/65 RAWDUMP routine capturing the serial stream and then ran HEXBIN. That seemed to work fine but when I ran CLEAN it failed on the second track.

I mounted the pre-CLEAN bin file in WINOSI configured with 40K of RAM and only my C1MON ROM (V2.02 with the keyboard fix).

C1MON comes up fine and Q starts the boot but only the first track is read and the monitor prompt reappears. What is read looks correct but it fails to load the other tracks.

Possible causes and actions:

- a. C1MON V2.02 - I may have introduced an error not with the simple keyboard address change but with some of the minor changes I made. Not likely but I will investigate.
- b. DOS/65 RAWDUMP - most likely problem source as it is new. Plenty to look at. The fact that CLEAN failed may be a hint.
- c. Other - wide open!

I have not tried the alternate approach but before I do that I need to make the checks noted.

Suggestions or test files you could post?

Thanks,

Rich



dave
Site Admin



Re: DOS/65 Richard Leary

Thu Aug 06, 2015 6:30 am

Hi Rich,

What was the file size of the hex file and the binary?

I haven't seen the source of CLEAN, but I imagine that it looks for track and sector headers to extract the meaningful data from the raw data, ignoring garbage bytes between sectors. That way, if you want to verify you have a good image, you can run rawdump twice, and compare the CLEANed images. If you have a match, then you have a good dump. This way, garbage bytes between sectors (which may include partially overwritten data with parity errors) won't make two good dumps look as if they don't match up.

Of course, if DOS/65 doesn't use the OSI sector format, then CLEAN may not work as expected. If that's the case, perhaps you could just try making 2 dumps and comparing the raw data. That may not mean the dumps are good, but at least it will tell you if they are consistent.

Also, note that the RAWDUMP format allocates 9*256 bytes per track (for 5.25")

One thing that occurs to me: Since the data between sectors is ignored, it would be easy to add in a CRC at the end of a track (or even a CRC at the end of each sector) to verify a correctly written/read track. This could be precomputed, and then just dumped out at the end of a sector or track write, and would improve confidence in the data. It would be completely compatible with the OS65D format, simply ignored by pre-existing software, and can even be inserted into

emulator disk images without breaking anything. That may help improve confidence in the quality of disk reproductions.

And, again, using something slightly better than the raw hex format, such as an intel hex format, or even a custom format with CRC, would help ensure that once tracks are dumped, they are not corrupted. There's no real compelling reason to stick with the RAWDUMP hex format, since the end goal is the binary image anyway.

Dave



MK14HAK



Re: DOS/65 Richard Leary

Thu Aug 06, 2015 8:45 am

May I suggest you check your system and directory tk headers:

43 57 TK 58 NB 76 01 08

Remove any "NB" noise bytes in your image and restore the length of each edited track by adding a byte at the end of the track.

If that doesnt give you a bootable image then use RAWDUMP and clean manually as above.

RAWDUMP should work fine as it dumps the track header + 2K if I remember correctly. The track headers are the same and no sectors within so it should work. Hexbin the 2K only and your done. No crc etc required.

I havent created a C1 image as yet Rich otherwise I would post for you. As far as I recall the kybd was the only glitch I found and I have a C4 version of your ROM booting images fine so I think your ROM should be good.

Does your dumped image of tk1 compare with tk 1 bin of compiled version ?

Mike

Last edited by MK14HAK on Tue Aug 11, 2015 7:38 am, edited 2 times in total.

600RevB:16K,2MHz,64x32,470,CEGMON

SuperKit:502,540B,542B,CEGMON, 8" and 5" FDDs

Cards:PE IO,6522 D-A-D, AY3-8910,ProgramGraphics,Color,UK101

WIP:HexDOS,FDD Emulator



bxdanny



Re: DOS/65 Richard Leary

Thu Aug 06, 2015 3:32 pm

“

May I suggest you check your system and directory tk headers:

43 57 TK NB 58 76 01 08

The noise bytes certainly wouldn't be between the track number and the 58 (hex) "track type" code. If present, they would be between the 58 (last byte of the track header) and the 76 (first byte of the sector header). And they really shouldn't do any harm in that position, since any byte other than 76 should be ignored there.

One unavoidable shortcoming of the image-file format is that there is no way of encoding a delay *between* bytes. That shouldn't matter when image files are run in the emulator, but it could make a difference in some circumstances when writing actual disks, if a delay that is supposed to be there between two sequences of bytes isn't there.

Dave's idea for adding a checksum or CRC at the end of each sector, even though it would normally be ignored , seems like a good one, UNLESS it should happen to be 76 02, or whatever the OS might be looking for to signal the start of the next

sector (if there are multiple sectors on a track).

As for the current images that don't work, one very simple initial check on the format would be to see if the 65D "sector directory" command (e.g. DI 01) shows a single 8-page sector on each track or not (RUN "SECDIR" to check a range of tracks).

No current OSI hardware

Former programmer for Dwo Quong Fok Lok Sow and Orion Software Associates

Former owner of C1P MF (original version) and C2-8P DF (502-based)



Post Reply    

64 posts



1

2

3

4

5

...

7



< [Return to "Files \(software, documentation, images, projects\)"](#)

Jump to 

[🏠 OSIweb.org](#) < [Board index](#)

 [Delete cookies](#) All times are UTC

Powered by [phpBB®](#) Forum Software © phpBB Limited

[Privacy](#) | [Terms](#)