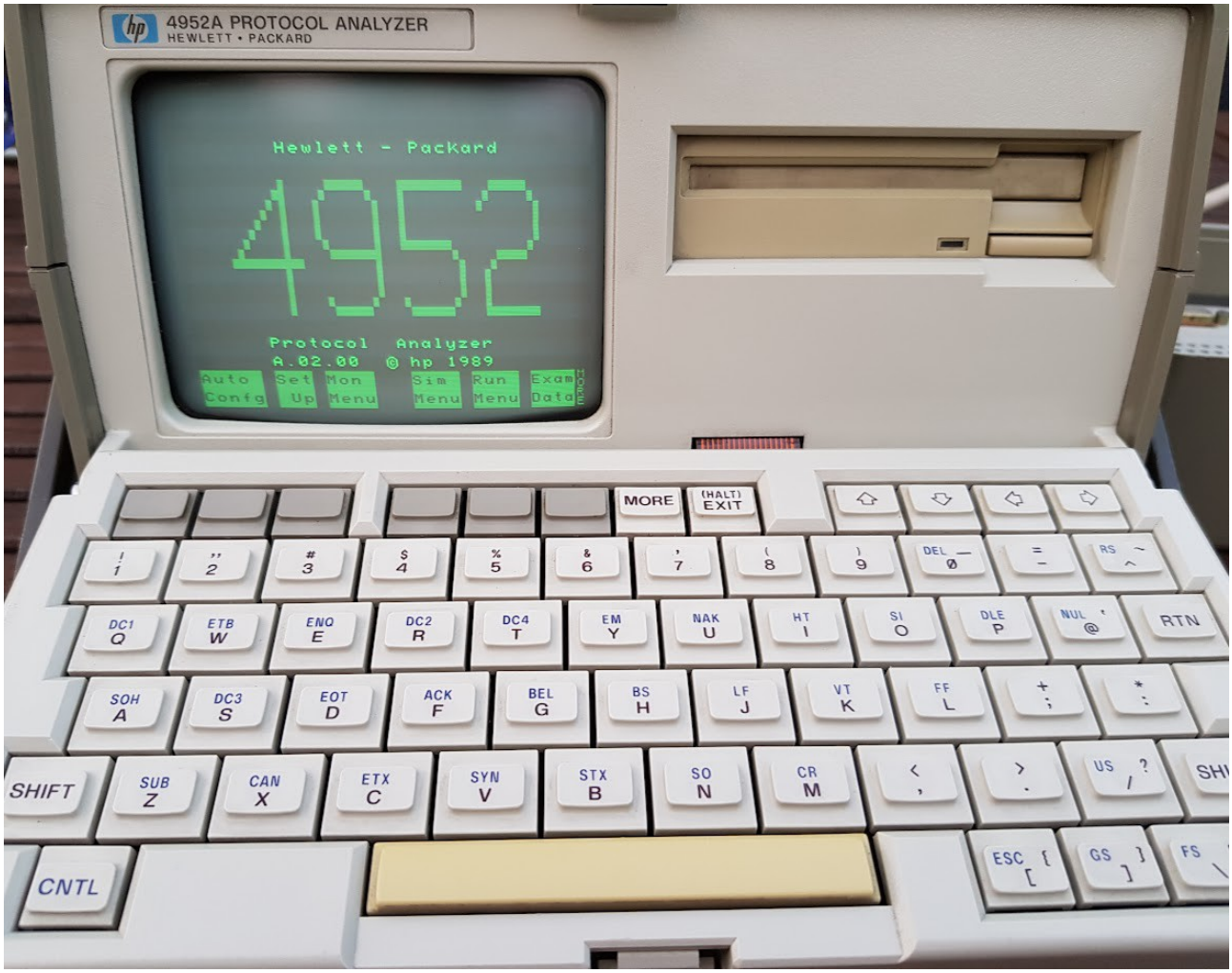
HP 4952A Protocol Analyzer

Repair.

16.08.22



Bernd Längerich, 21.11.2022 v23

Table of contents

Table of figures

Introduction

In the past years, I have worked on commuications a lot. The company I worked for had a communication analyzer to sniff on serial communications, HDLC, X.25 and later ISDN. They were used a lot.

Lately a HP 4952A was acquired on ebay by me. It came in a box and was rattling inside. The seller did a very poor job on packing up the instrument in a suitable box. Oh boy, that was not a good start...

Figure 1: Poorly packaged instrument...

Fortunately the instrument seemed to be undamaged, no shattering glass, no dents or cracks. I have to thank Hermes, they seemed to take good care of the box.

First inspection

There seemed to be no damage, so I inspected the contents. The whole pack consist of

1. The HP 4952A Protocol Analyzer
2. A HP 18179A RS232/V.24 Pod
3. The pod connection cable
4. A RS232 ribbon cable
5. A (new?) mains cable
6. A spare fuse

The CRT was very visibly burned, so it had a lot of working hours. I was not expecting a good display from it.

The mains voltage can be anything between 100 and 240V, 48-66Hz, so I plugged it in and throw the switch. No smoke...

The instrument came up with the main screen and a warning that the NVRAM contents was lost. I confirmed and found the menu to be working. The display was not so bad than I thought, the burned lines are less visible when powered on.

The instrument has no extended memory board.

Figure 2: Contents of pouch

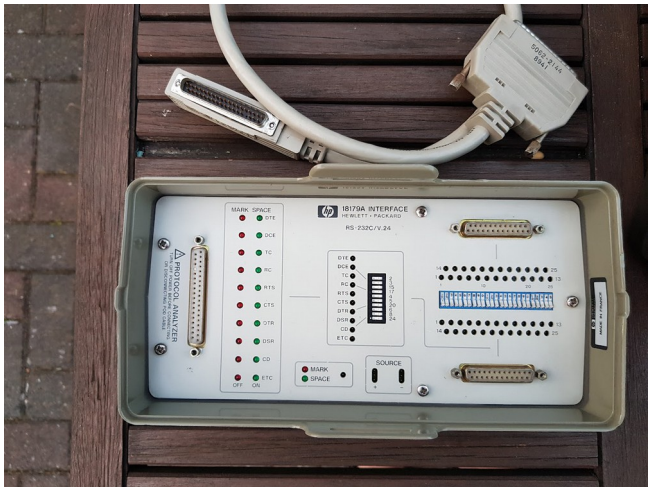


Figure 3: Interface pod and cable

Cleaning

The first thing I did was cleaning the instrument. There was quite some dirt on all the panels and after a wash of all the outside, it looked way better. The disk drive and the space bar were quite yellow, but I couldn't do anything to make it brighter. I refused to do the RetroBright-thing yet.

Dismantling the analyzer

To wash the outside, I opened the analyzer. I removed the pouch from the top and the screws from the bottom, then I removed the top cover.

Inside there was some dust, so I removed the disk drive, the disk controller board, the memory board and cleaned all.

Figure 4: Front view with RS232C Interface Pod

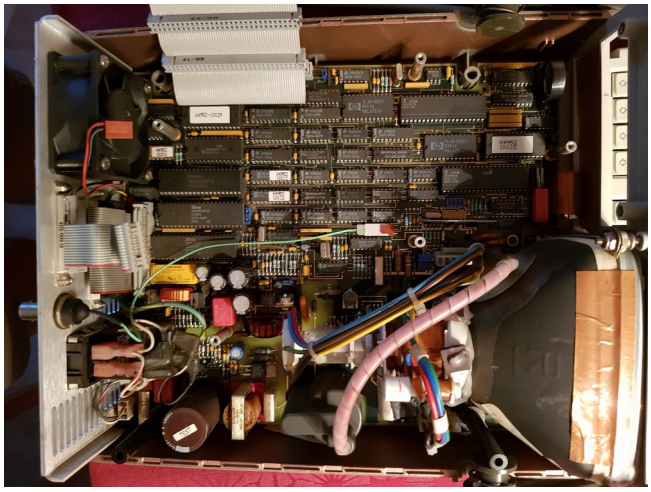


Figure 5: Main board A1 and CRT

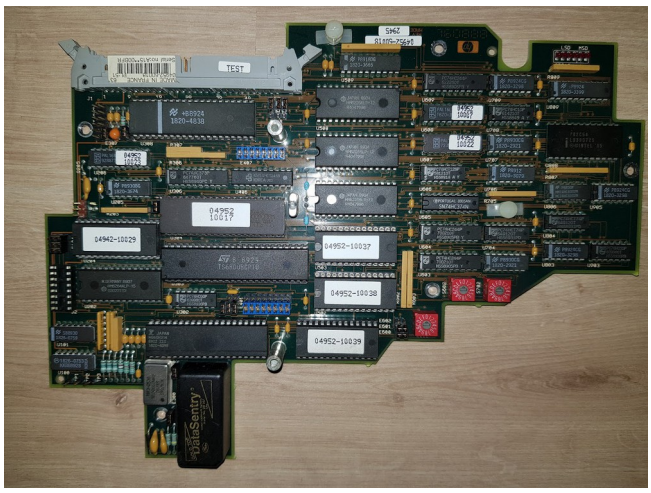


Figure 6: Memory board A2

Figure 7: Disk controller board A3

Inspecting the infamous Rifa X-capacitor, I found it had already released the magic smoke and had been replaced. The replacement doesn't show any cracks, so I left it in.

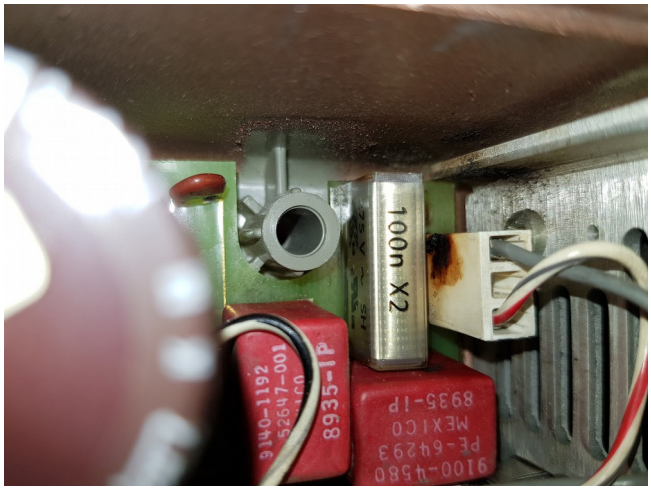


Figure 8: X capacitor with smoke trails

But the filter made by Schaffner already leaked, so I removed the filter section and rewired. I have to find a suitable replacement mains socket, filter, fuse and switch in that form.

Figure 9: Rear viw with combination of IEC plug, filter, fuse and switch

Testing with the boards installed back, I found that all keys are working except the "3" key. So I had to dismantle the keyboard. All keys are made by Cherry, so I unsoldered the key, opened it and gave it a good clean.

Figure 10: Desoldered key

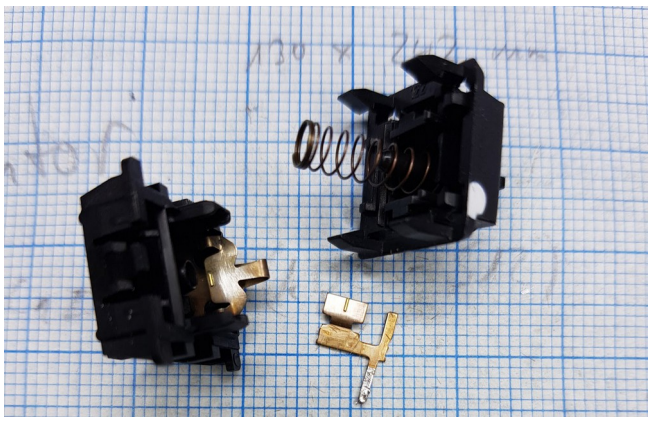


Figure 11: Disassembled and cleaned key

After reassembling it, it tested fine, so I resoldered it.

Memory backup battery

The memory backup battery is a NiCd accumulator. It still works and hasn't leaked yet, so I leave it in now. I will replace it later. This seems to be a good brand, as a lot of other cells already leaked and destroyed several boards.

Disc drive

The disc drive was untested, as I had no DD discs avaliable. After getting some old discs, I tested the drive. It made horrible noises and started to scrape the coating from the disc. Furthermore, the eject mechanism was not working OK, so I wanted to replace the drive.

Further inspection showed me, that the drive uses an unusual interface with only a 34pin ribbon cable. No power cable, it uses some of the ground lines on odd pin numbers to supply +5V and +12V. This is not hard to rewire, but I soon found, it uses a special disk change reset signal. But the show stopper of replacing the drive with a usual 3.5" microdisc drive was: It rotates at 600rpm, that is twice the speed of a "normal" disc. So the data rate is 500k with double density MFM, not 250k.

2-3-2. Transfer Rate

Burst Transfer Rate : 500Kbits/sec for MFM

2-3-3. Access Time

a. Track to Track Slew Rate
: 6msec Min.

b. Track to Track Step Settling Time
: 15msec Max.

The value of 15msec is the time necessary to stabilize the head within 0.035mm of its absolute position.

d. Motor Start Time : 900msec Max.

Motor start time is the time that it takes for the READY signal to become true (low) after the MOTOR ON signal has been activated.

2-3-4. Functional

a. Rotation Speed : 600rpm

The continuous speed variation is within ±1.5%. The instantanuous speed variation is within ±1.0%.

b. Recording Density : 8717BPI
(Side 1, Track 79)

c. Track Density : 0.1875mm
(7.38×10^{-3} in.)
Track - Track

Figure 12: Excerpt of technical data of Sony MP-F52W

3-1. CONNECTOR AND PIN ASSIGNMENTS

3-1-1. Signal Connector Pin Assignment

PIN	SIGNAL DESCRIPTION	PIN	SIGNAL DESCRIPTION
1	DISK CHANGE RESET	2	DISK CHANGE
3	+5V	4	IN USE
5	+5V	6	DRIVE SELECT 3
7	+5V	8	INDEX
9	+5V	10	DRIVE SELECT 0
11	+5V	12	DRIVE SELECT 1
13	RETURN	14	DRIVE SELECT 2
15	RETURN	16	MOTOR ON
17	RETURN	18	DIRECTION
19	RETURN	20	STEP
21	RETURN	22	WRITE DATA
23	RETURN	24	WRITE GATE
25	RETURN	26	TRACK 00
27	RETURN	28	WRITE PROTECT
29	+12V	30	READ DATA
31	+12V	32	HEAD SELECT
33	+12V	34	READY

Figure 13: Floppy pin assignments

So I started to disassemble the drive and began to clean it thoroughly. Meanwhile I started to check my new old discs that came from a Commodore Amiga. Most computers had high density drives, but Atari ST and Commodore Amiga started with double density, so theses are the discs you will mostly get.

Amiga discs are recorded with GCR, not FM or MFM. And it seems they used a higher writing current as a lot of discs where hard to format. I first thought they were all bad, but I found out that they got better after several formatting. So I wrote a script that formats until disc is OK, but at most five times. All discs that failed were discarded. I got a bunch of workings discs formatted to 720k.

The cleaning went well, but the disc insert and eject mechanism is still stuck, it doesn't slide in and out freely. The disc drive is connected with a very short cable, so I used a longer cable for testing. The drive reports "Unformatted disc" when inserting a blank disc, but soon gets stuck when trying to format it with "Bad disc" indication. Sometimes it stops rotating and freezes, but will continue with "Disc out" when removing the disc. So the disc change signal is working fine. Testing with a scope, the index pulses come at every 100ms, which indicates 600rpm, so this is fine. The ready signal is OK. At the read data pin there is data available, so next is to check write gate and write data lines during formatting. The discs are fine when they are formatted using a Linux system.

I made a test adapter with two 34pin headers that separates the supply lines and routes them to a 4pin header to enable usage of a normal 4 pin connector for 3.5" drives. Additionally all signals are routed to pin headers to allow observing and setting signals. So I can monitor signals while being driven by the 4952A or I canset lines and check results without the 4952A.

Using the 4952A did not reveal much information, I was using a scope only. Maybe a logic analyser would have made sense to be used, but then I would have to dig through kilobytes of raw data to check for sync signals, gaps, address marks etc. Some interesting things I found:

1. Write protect is only detected while really trying to write to the disk
2. Formatting reads cylinder 79 first and then seeks back to track 0, writing and reading it.

Testing the Sony drive, I came to the conclusion that either head 0 (lower side) is broken or the surrounding electronics. It always reads garbage. Head 1 is working fine, and I confirmed it by writing 500kHz from a function generator with write gate enabled. I can read the signal back from head 1, but not from head 0. Unfortunately the heads flex cables have different layout (mirrored) for both heads, so I cannot swap both heads as the flex cable is not long enough. This makes it nearly impossible to check wether the head or the amplifier is faulty.

Using a usual 3.5" drive I confirmed that this works for the same disk with 250kHz on both heads.

As these special disc drives are hard to find and rather expensive, I checked for replacing the drive with a USB floppy emulator.

Gotek floppy emulator and FlashFloppy firmware

The Gotek floppy emulator was used with some other gear already, but only to replace "normal" drives. Reading the fine manual, there seems to be configuration options to emulate the data rate and the spindle speed. I have to dig further if this is a feasible solution and how to prepare the disc images.

It should be possible to emulate the 600rpm by setting the spindle speed and data rate. The sector size, number of sectors etc. should be easy, too. What is currently unknown is the different track layout for upper tracks, It seems to use at least some high sector numbers.

First configuration file

I found some information on the internet about a HP logic analyzer, using the same drive with LIF format. It indicated that the special handling would be on cylinder 79. According to one of the disk images made with Teledisk, on track 79 there are sectors numbered 98, 98, 99,..., 112, so they do not start with id=1 as on all other cylinders, but use high sector numbers. This applies to cylinder 79 and both heads, cylinders 77 and 78 seems to be unused. But this information is not confirmed.

This leads to the following IMG.CFG to be tested on FlashFloppy:

```
[4952A]
cyls=80
heads=2
rpm=600
mode=mfm
rate=500
secs=16
bps=256

tracks=0-78
  id=1

tracks=79
  id=97
```

Furthermore a raw image with 655360 bytes as of 80 cylinders * 2 heads * 16 sectors á 256 bytes is created as an "empty floppy":

```
dd if=/dev/zero of=EMPTY.4952A.IMG bs=256 count=2560
```

Special floppy cable

To connect the Gotek drive without modifying it, I assembled a special wire harness, using two 34pin IDC connectors and a ribbon cable. From the second side to the Gotek, I modified the cable as follows:

1. Cut Pin 1 and use a single crimp contact ("Dupont") to jumper it to pin SWCLK of the Gotek (right above the power connector)
2. Cut pins 3, 5, 7, 9, 11 and connect them to a single crimp contact to connect to the power connector (+5V)
3. Cut pins 29, 31, 33

This disconnects some of the odd numbered pins from ground as they are used by HP for power supply and disk change reset.

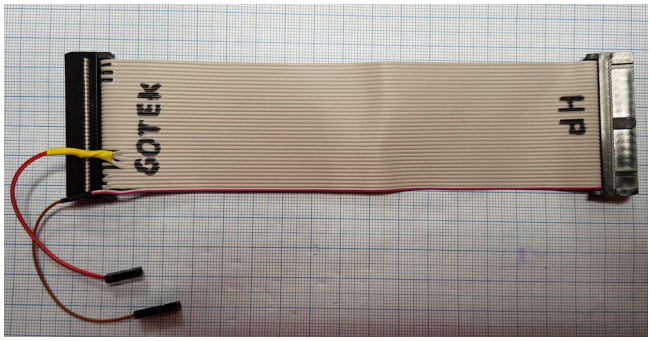


Figure 14: Special connection cable

The Gotek has to be jumpered as DS0 and configured for Shugart interface with pin assignment as follows in file FF.CFG:

```
interface=shugart
pin02=chg
pin34=rdy
chgrst=pa14
```

First test

Using this configuration, the HP indicates "DISC OUT" without an USB stick and changes to "NO LIF DISK" or „DISC NOT FORMATTED“ when inserting it. The display states the drive is seeking to cylinder 78/79 and then back to zero when inserting the disc. This is identical to the original drive. Formatting the disc starts and formats cylinders zero to 76 with lower/upper heads, then switches to cylinder 78 upper head and this fails. Three tracks are ommitted.

Second configuration file and test

I changed the IMG.CFG to include all three upper cylinders 77 to 79 to use start sector 97, but this didn't help either. So I had to debug it more. The upper cylinders are special and some kind of copy protection. One of the images indicates cylinders 77 and 78 to be 9 sectory with 512 bytes, but I doubted this would be true (The remarks indicate you have to copy only the lower 77 cylinders 0 to 76 to a freshly formatted HP disc o get a working copy).

So I updated to a logging FlashFloppy version and used a second image (according to the documentation you should swap image before removing the USB stick, which is not possible with only one image).

The resulting log indicated, that BAD IDAM 0x11 to 0x20 has been selected.

```
IMG IDAM Bad Sector: 11
IMG DAM Unknown
IMG IDAM Bad Sector: 12
IMG DAM Unknown
[...]
IMG IDAM Bad Sector: 19
IMG DAM Unknown
IMG IDAM Bad Sector: 1a
```

```
IMG DAM Unknown
[...]
IMG IDAM Bad Sector: 1f
IMG DAM Unknown
IMG IDAM Bad Sector: 20
IMG DAM Unknown
```

IDAM is index address mark, so I changed the IMG.CFG to use first id=0x11 for cylinder 78.

```
[4952A]
cyls=80
heads=2
rpm=600
mode=mfm
rate=500
secs=16
bps=256
tracks=0-77
  id=1
tracks=78
  id=17
tracks=79
  id=97
```

This did not work either and results as before in "BAD DISC" after some tries on cylinder 78 head 1, but the logfile now has no errors, only successful writes before the disc change.

I am confused.... The resulting image clearly indicates the cylinder 78 head 1 is formatted the same as cylinders 0 to 76 are. I see the same data bytes, the sectors contain a sequence of „6D B6 DB“ in the whole sector. So there must be another trick HP is using.

The data bytes for cylinder 77 heads 0 and 1 and cylinder 78 head 0 are unwritten. I assume they are never addressed.

Third test

So I tested with another approach. The disc image made with Teledisk indicates some sector contents. It includes cylinders 78 and 79 with some data in it. But on cylinders 77 heads 0 and 1 and cylinder 78 head 0 there are 9 sectors with 512 bytes in it, so it was formatted as MSDOS 720kB.

Deeper inspection showed all cylinders 0 to 76 using sector ids of 1 to 16. cyinder 78 head 1 uses sector ids 17 to 32. cylinder 79 uses sector ids 97 to 112. All sectors are 256 bytes. All tracks that are used have 16 sectors. So I made an educated guess that this information is indeed correct and if correctly written to disk this would be the disc contents that was read from the original disc.

So I converted the image to my needs in multiple steps:

1. From Teledisk to RAW image, this too large due to the MSDOS tracks inbetween
2. Copy the first cylinders 0 to 76 from raw image to give part 1
3. Copy a blanc sector

4. Multiply the sector for 3 tracks (48 sectors) to give part 2
5. Copy the last 3 tracks with 48 sectors from the image to give part 3
6. Concatenate all 3 parts to a new raw image, resulting in exactly 655360 bytes

I was not convinced that this would work, but it was worth a try. The resulting raw image fits my assumption of the geometry, so everything I had gathered up to now was included.

I ran all commands on linux as windows is a PITA for those tasks (everything is available there too, but...):

```
dskconv -itype tele -otype raw 4952A-2.TD0 4952A-2.IMG

dd if=4952A-1.IMG of=4952A-1_1.IMG bs=256 count=2464

dd if=4952A-1.IMG of=sector.IMG bs=256 skip=2463 count=1

cat sector.IMG sector.IMG > sector2.IMG
cat sector2.IMG sector2.IMG > sector4.IMG
cat sector4.IMG sector4.IMG > sector8.IMG
cat sector8.IMG sector8.IMG > sector16.IMG
cat sector16.IMG sector16.IMG sector16.IMG > sector48.IMG

dd if=4952A-1.IMG of=4952A-1_2.IMG bs=256 skip=2518

cat 4952A-1_1.IMG sector48.IMG 4952A-1_2.IMG > 4952A-1.IMG
```

The resulting file was copied to the USB stick as the only image along with the configuration file. And when the stick was inserted, the HP 4952A started to scan the disc, seeked to cylinder 78 and 79, back to zero and then scanned the LIF file system. The image was somehow working!!!

I could DIR the discs, load files. So read access seem to be working fine.

That was surprising and I tried to copy files, remove files, pack discs etc. and all went well. So I copied another image and then deleted everything from it, giving an empty disc. That empty disc image can now be used like a freshly formatted disc.

Application denied

Another problem I ran into: I cannot load the application programs from the system disc (image). They are not loaded with „Application denied“ error messages. According to the manual this occurs if the application is copied from the original disc. With the disc copy utility (on the same disc an external application). I assume this has already been done on the image file I got, so I cannot do very much. As the disc controller card uses its own microprocessor 80C88 and the error message is contained in the ROM U606 on the disc controller, I will try to disassemble the code contained in the ROM. Maybe one can find out what causes this message. The message is emitted without any access to the disc, so it may be some attribute in the direcory which is not copied.

Disassembling the FDC controller ROM

The ROM is 32kB in size. I have been using IDA pro, but I had information that Ghidra is a newer tool, running on Windows. So I gave it a try.

After some time with the file in Ghidra, I managed to find the disc format routine. I walked through all that x86 assembler code and found the culprit why formatting was not working:

They used strange track ids on cylinder 78 head 1. The track id was not 78, but 150! As

FlashFloppy can not handle those track ids in a raw image file, it reports back track 78. That is why the format did not work! Stupid me haven't found the information that was already in the TeleDisk image! Copying a disk with TeleDisk would have worked, I assume. And a copy with another copy tool like ImageDisk would have worked too, as well as using a real floppy drive at the magnetic surface records whatever is written to it, not so with the emulation.

Switching image format for the floppy emulation

After some discussions about the issue, Keir, the developer of FlashFloppy, suggested HFE format, as this format stores more metadata. And surprisingly, the empty HFE file I ceated could be formatted! So this was a big success!

Next step was to convert one of the Teledisk images to HFE, the tool from HXC2001.COM did this flawlessly. The only thing to do is patch the data rate, as the images were taken with a usual PC drive. After that the HFE file mounted fine and could be read by the 4952A. And interestingly the applications could be loaded too! So my assumption, that this does not work was wrong and the images taken by Teledisk are fine.

Next step was to convert the IMD files for the optional application discs. They were missing the upper tracks and I had to combine the lower tracks with the upper tracks of a 4952A formatted disc.

Creating a custom front for the Gotek drive

The Sony flooppy drive is higher than an usual PC floppy drive. The Gotek form factor follows the now used PC like flopps drives, so I had to mount the drive with a bracket inside the HP. As the drive was black, I was not satisfied with the optics and decided to create a new front.

On Thingiverse there is a Gotek bracket and front for the Amiga computer, that uses the same form factor. I modified the design to include a HP logo of that time and printed the front with two colors.

Figure 15: hp branded front

The bracket to mount the Gotek hardware was printed too and mounted inside the 4952A:

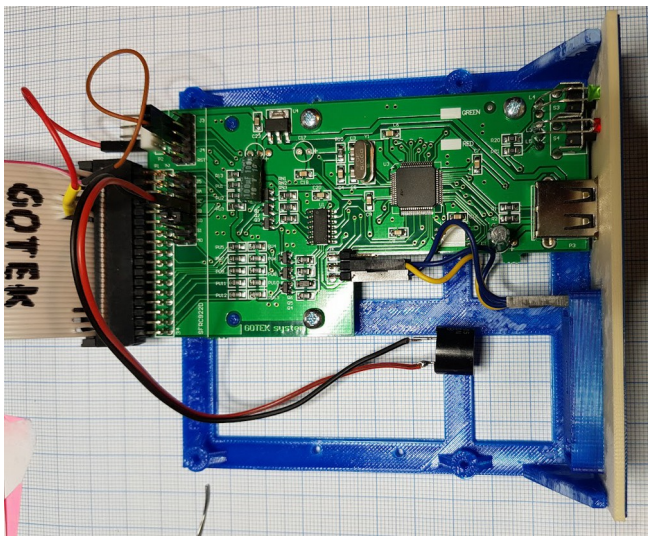


Figure 16: Bracket with Gotek hardware and buzzer

The FlashFloppy software can handle an OLED display instead of the 3 digit LED, so I mounted one to indicate not only the number of the image used, but the name. This greatly enhances usability.

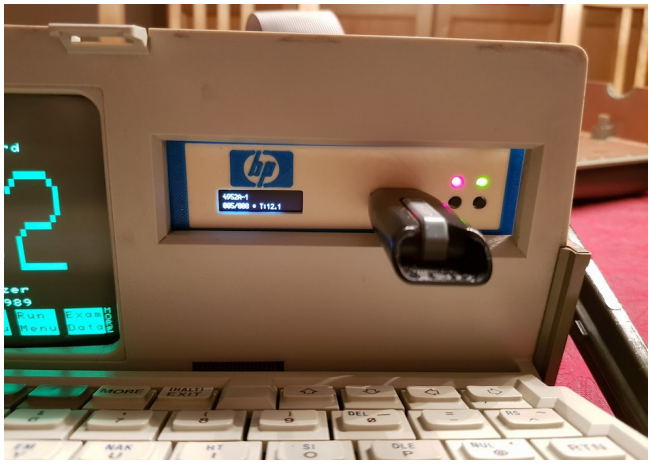


Figure 17: Floppy emulator inside the 4952A with OLED

Surprisingly the HFE images are faster than the raw data images with configuration file, despite the fact that more data is to be read from the USB stick.

Conclusion

Old disc drives are mechanically very delicate devices. They do not work any more if there is corrosion, dust, or they have been abused. It is important to not put discs of value in it before assuming proper operation.

As I had no DD discs, I had to buy some used discs. Not all old discs behave well too, they have mechanical issues (tight fit in their sleeve, crusty surfaces which will hinder rotation, bad spots...) Furthermore discs that were used in Commodore Amiga computers are recorded with a different recording scheme (they use GCR) and seem to be harder to reformat.

HP invented some copy protection scheme, that could be copied (at least the images made with

Teledisk contain all information that is required to create a working copy), and made quite some effort to create strange sectors.

The FlashFloppy software is such a great emulation, it even can handle such odd sectors with the help of the HFE image format. It takes some more space for the images, but this is not a problem with todays USB stick sizes.

What was a little bit annoying was the fact that I already had the information about the strange sector headers in the Teledisk image file, but I did not recognize them. This resulted in some work of disassembling the FDC controller code, but this was an exercise that can be useful with other software reverse engineering.

The HP 4952A is a nice device. Although it is somehow limited, it is still useful when analysing serial communications. It will not handle data rates above 64kbps though.

Acknowledgements and disclaimer

First I like to thank my wife, who is sometimes not very fond of the strange hobbyist she married. Even the music I play is not her style. And test equipment everywhere, mostly defective and/or opened... However some of her husbands skills come in handy when household devices like washing machines etc. become inoperative.

I wrote this report for my personal recording. It is not allowed to copy, reproduce or print it without my approval. No commercial use is allowed without prior approval.

All figures, measurements, conclusions and presentations are made to my best knowledge. There is no guarantee whatsoever for correctness and applicability. There may and will be errors, ommissions and flaws in this report, as this is just a presentation of what I have done.

Working on test equipment, especially those that operate with mains or high tension voltages, may expose you to dangers. All work is done on your own risk. DO NOT TAKE MY ADVICE AS THE ONLY SOURCE OF INFORMATION.

All trademarks and brand names are properties of their owners. They are used within this document solely to name the manufacturer or brand that was available to me.