

Recovering data from the Micropolis 1325

Ronny Hansen, May 2024

ST506 Drive - 1325

- ND was using different ST506 drives from Micropolis.
- Micropolis 1304 formatted as 23MB or 45MB
- Micropolis 1325 formatted as 28MB, 45MB or 74MB

- 1325 has 85.3MB unformatted capacity
 - Disks 5
 - Heads 8
 - Cylinders 1024
 - Sectors 17?

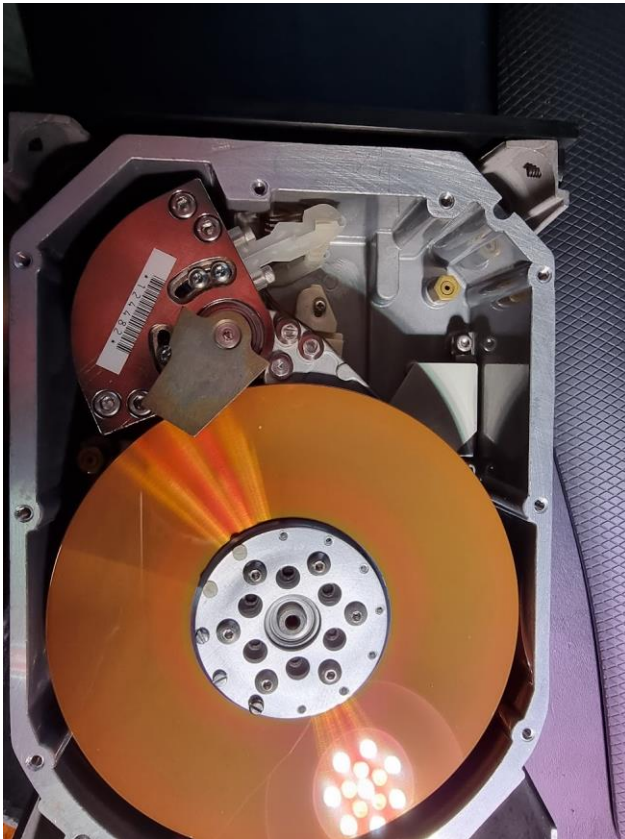
- SINTRAN Page = 1024 Words (2KB)
(1777 oct)



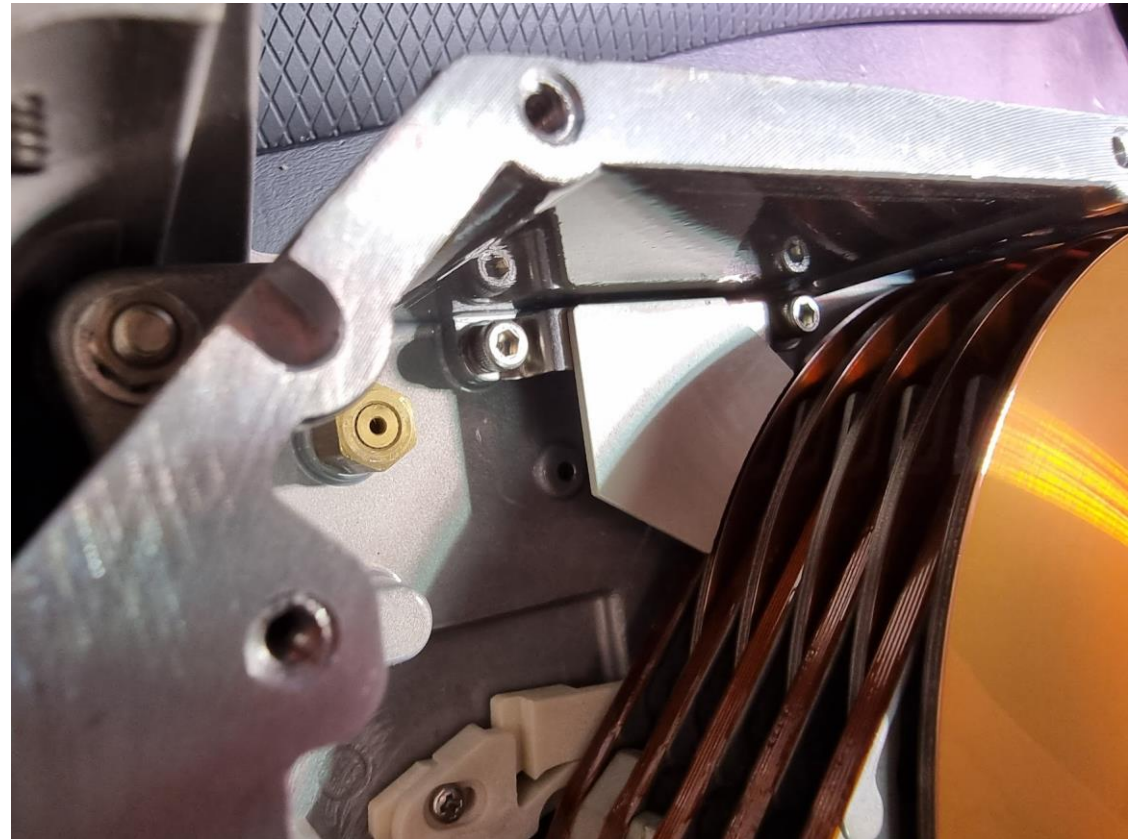
Challenge

- I can read any track/cylinder on the drive if I am accessing head 0-3.
- As soon as I try to access head 4-7 I get "Address mismatch".
- I thought first it could be signalling from the controller
 - Forcing the "head select bit 2" to GND (active low) doesn't improve the "Address mismatch" for head 4-7.
 - However, while forcing it to GND and trying to read head 0-3 now gives "Address mismatch".
 - Removed the GND signal from bit 2, and now I can read 0-3 (as before) but head 4-7
- Conclusion:
 - The drive uses the head select bit 2, but heads 3-7 are still not behaving.

Checked the inside of the drive – Nothing stuck. Nothing special



<https://youtube.com/shorts/Pc21rt4SWuY>



<https://www.youtube.com/shorts/SVTQmTflNP8>

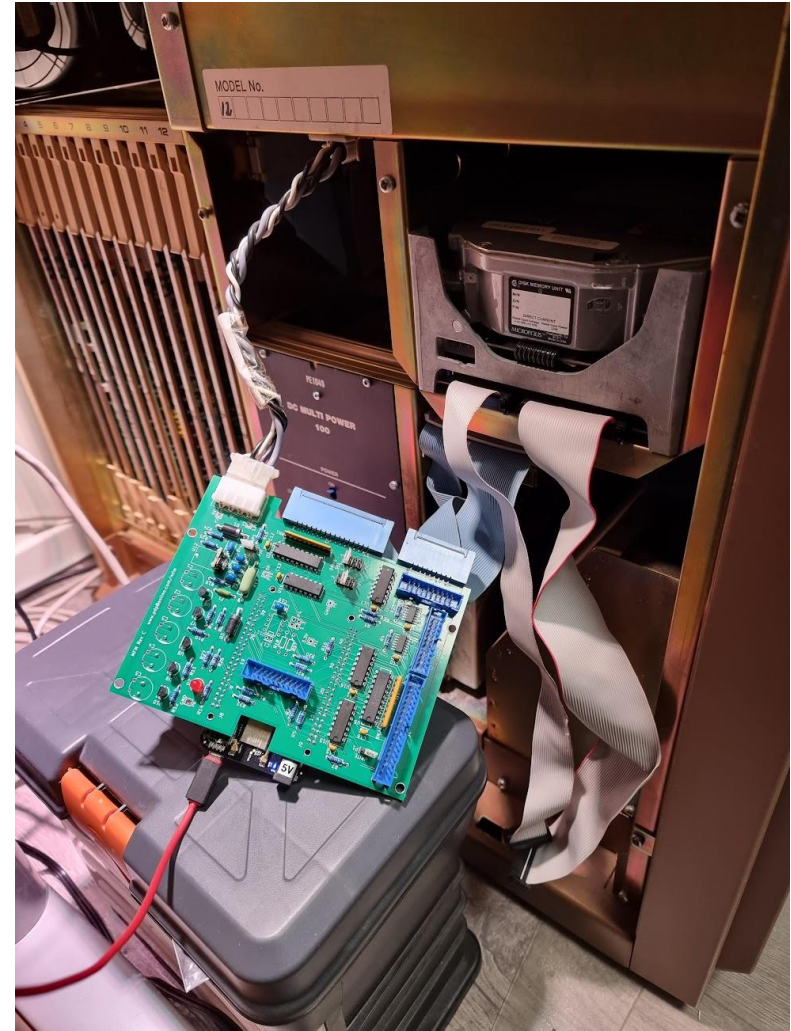
Why is this a challenge

- With drives that have a servo platter a common issue is not all the heads are properly centered over the track which causes the data from some of the heads to not be recoverable.
- The head servo is closed loop positioning which keeps the servo head centred on its track.
- This should keep all the heads on track but mechanical changes cause some of the heads to be out of alignment.

ST506/MFM HDD Emulator



<https://www.pdp8online.com/mfm/>



Extended the SW to understand the drive

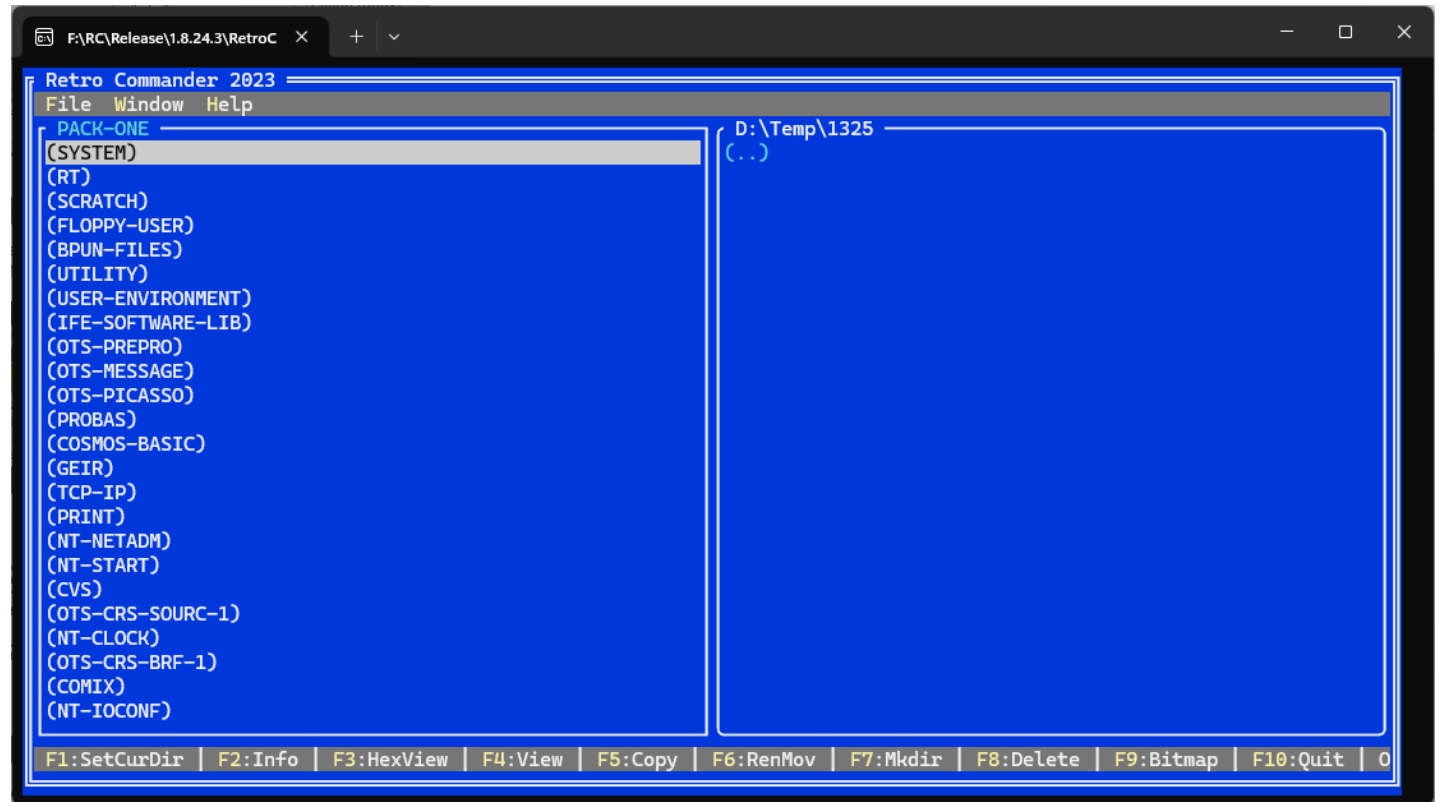
- David was very nice and extended the mfm software to understand the ND format on the Micropolis 1325 drive.

Commands to read an image off the drive

- Reading HDD image

- `mfm_read --format nd100_3041 --drive 1 --cyl 1024 --retries 100 --heads 8 --ignore_seek_errors --tran micropolis_raw --extracted_data_file micropolis_data.img`

Dumped
what I could
from the
drive



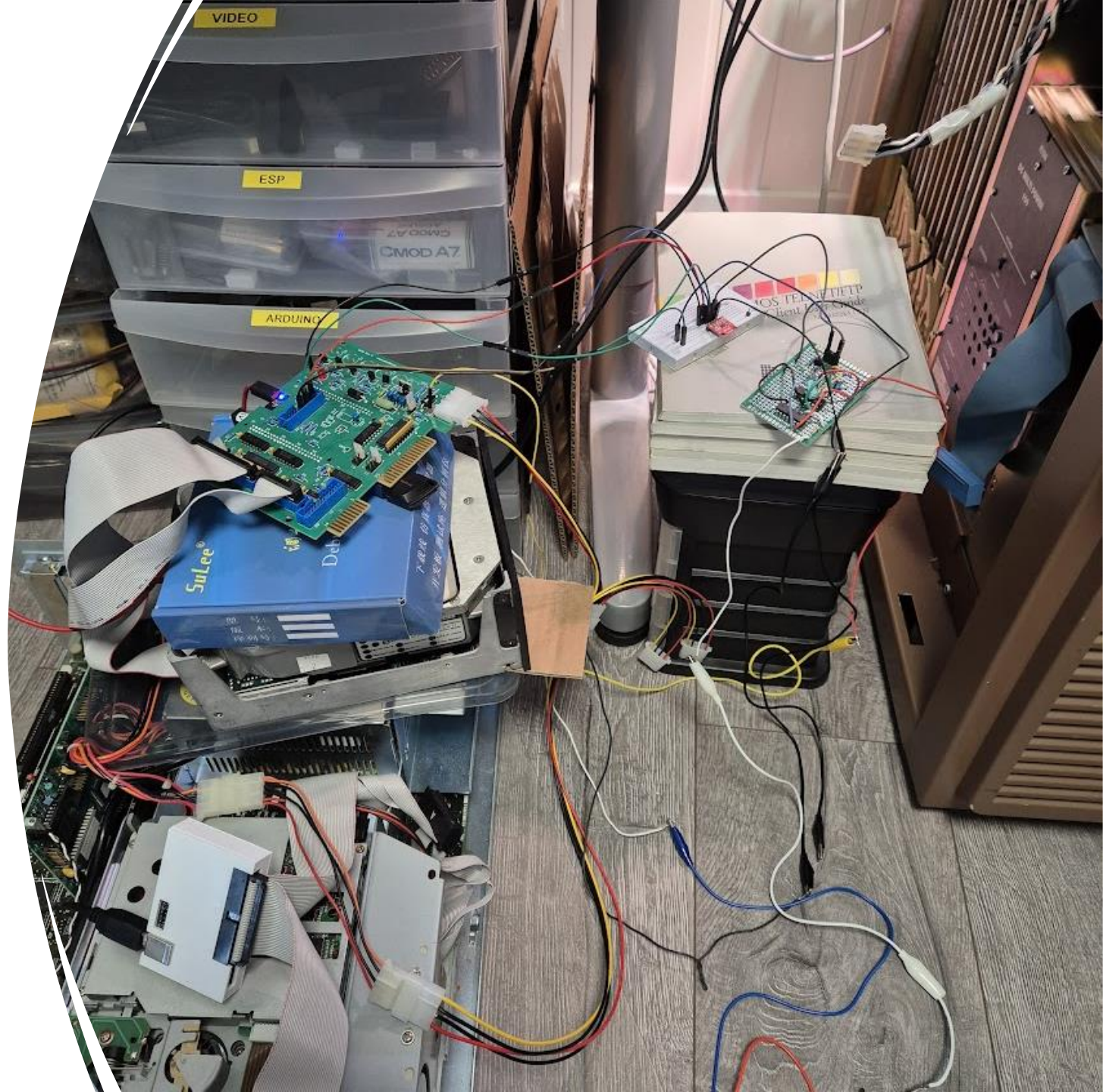
The screenshot shows the Retro Commander 2023 application window. The title bar reads "F:\RC\Release\1.8.24.3\RetroC". The menu bar includes "File", "Window", and "Help". The main window is split into two panes. The left pane shows a directory listing for "PACK-ONE" with the following contents:

- (SYSTEM)
- (RT)
- (SCRATCH)
- (FLOPPY-USER)
- (BPUN-FILES)
- (UTILITY)
- (USER-ENVIRONMENT)
- (IFE-SOFTWARE-LIB)
- (OTS-PREPRO)
- (OTS-MESSAGE)
- (OTS-PICASSO)
- (PROBAS)
- (COSMOS-BASIC)
- (GEIR)
- (TCP-IP)
- (PRINT)
- (NT-NETADM)
- (NT-START)
- (CVS)
- (OTS-CRS-SOURC-1)
- (NT-CLOCK)
- (OTS-CRS-BRF-1)
- (COMIX)
- (NT-IOCONF)

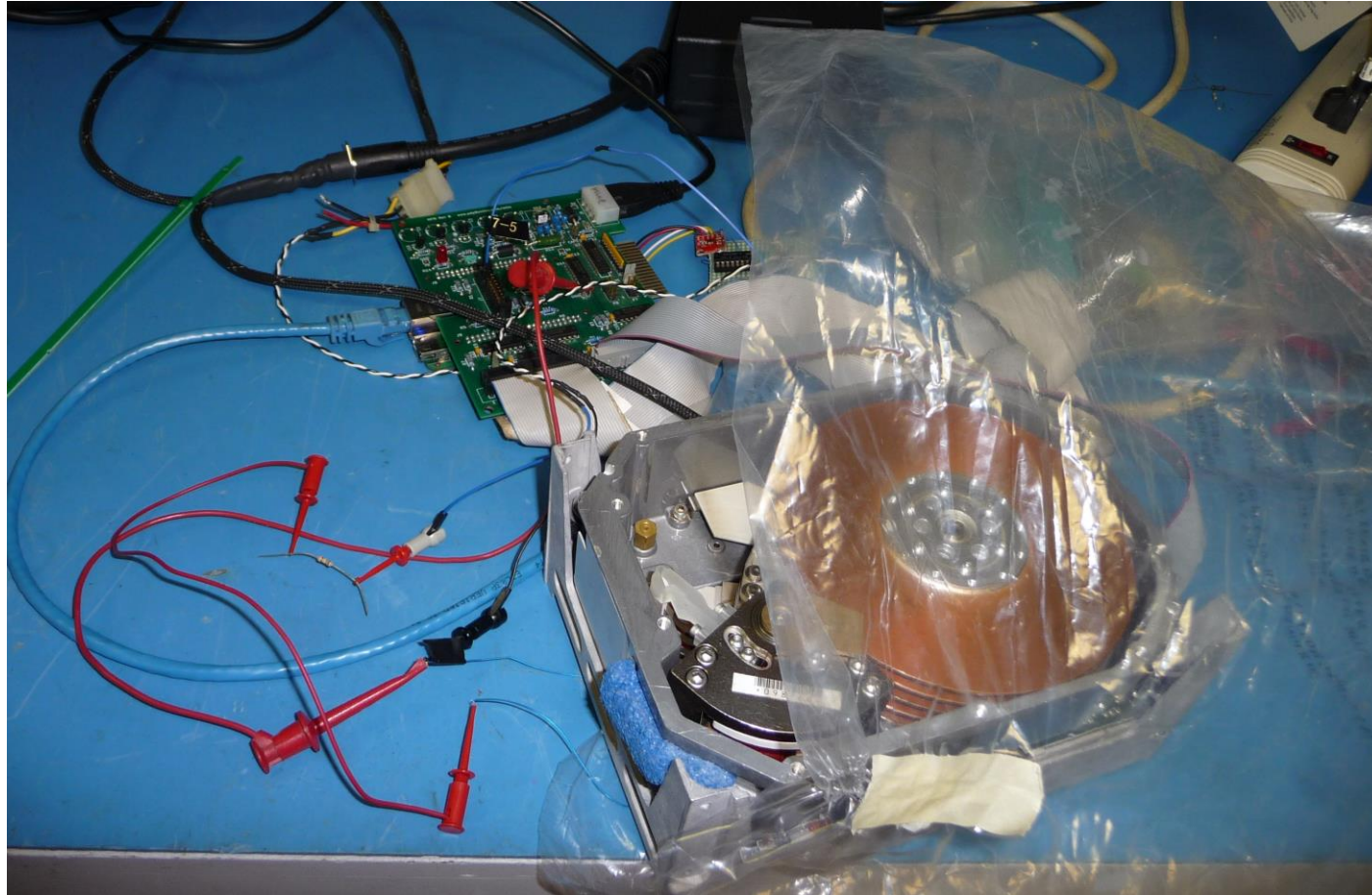
The right pane shows the current directory path "D:\Temp\1325" and its contents "(..)". At the bottom of the window, a status bar displays function key shortcuts: "F1:SetCurDir | F2:Info | F3:HexView | F4:View | F5:Copy | F6:RenMov | F7:Mkdir | F8>Delete | F9:Bitmap | F10:Quit | 0".

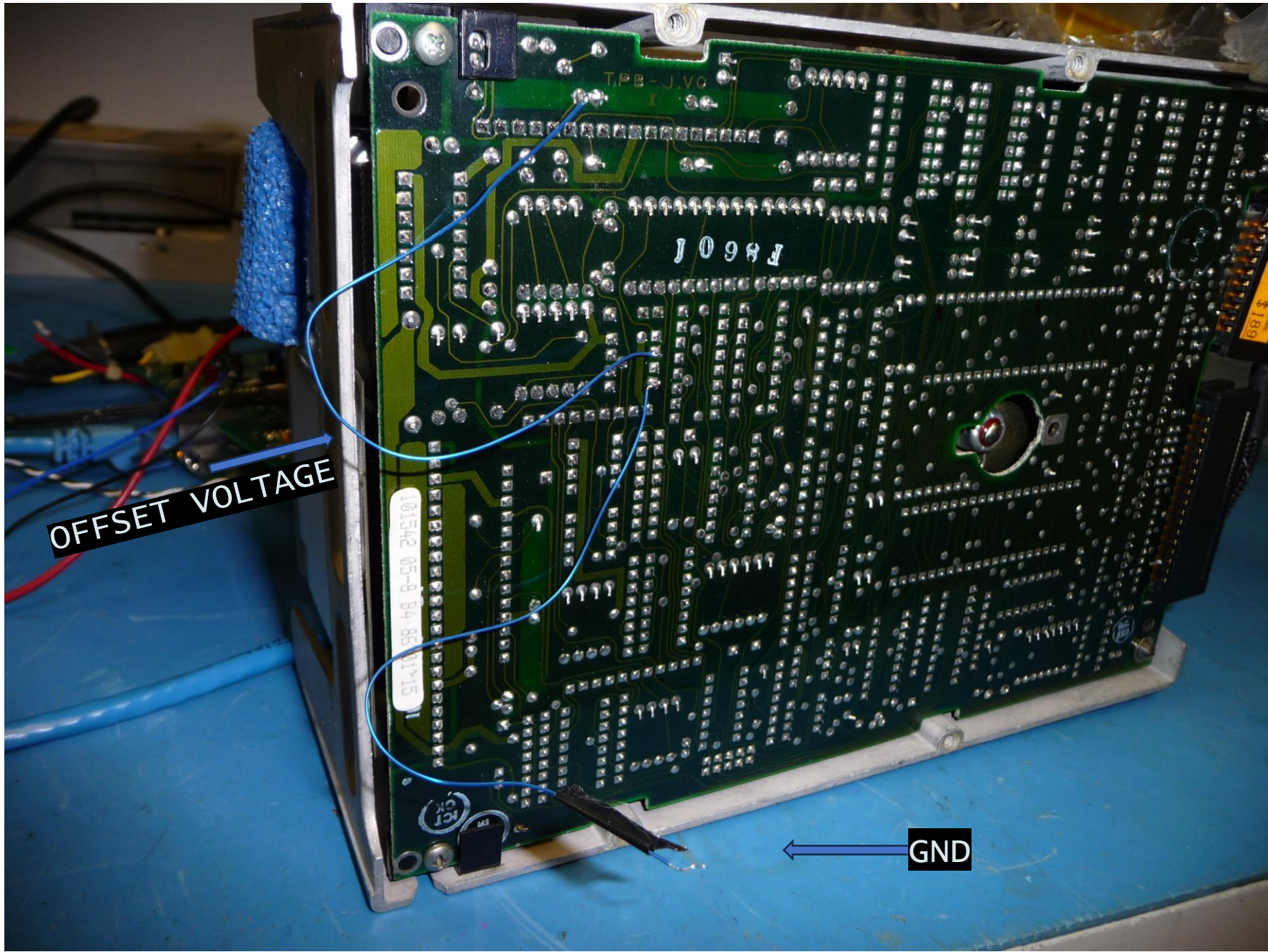
Data recovery by pulling servo head.

Adding a DAC and a buffer.
Controlling the DAC from code and forcing the drive to read the correct cylinder



Data recovery by pulling head servo (David's setup)

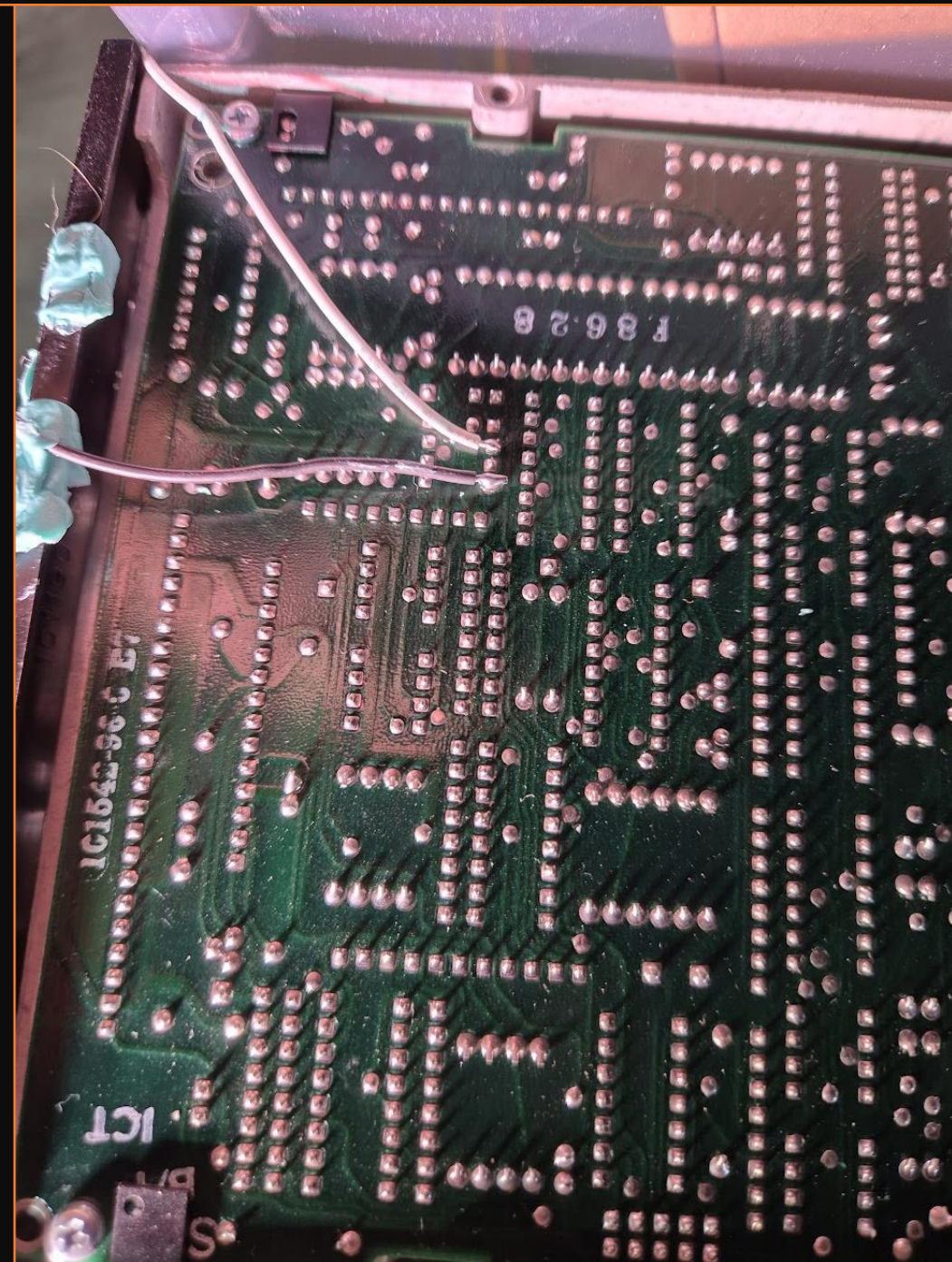




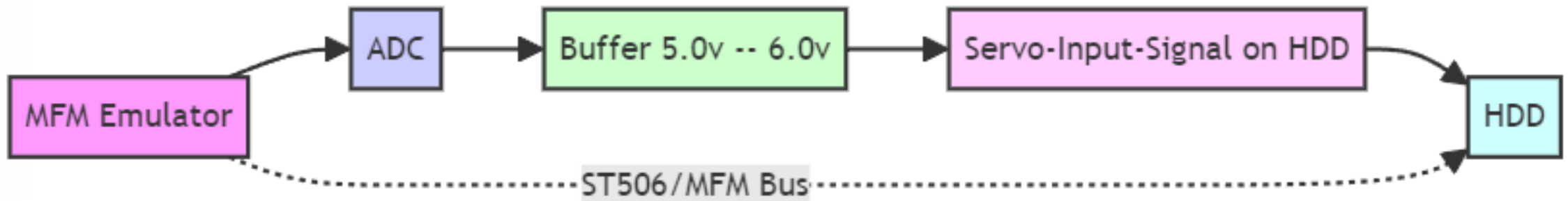
OFFSET VOLTAGE

GND

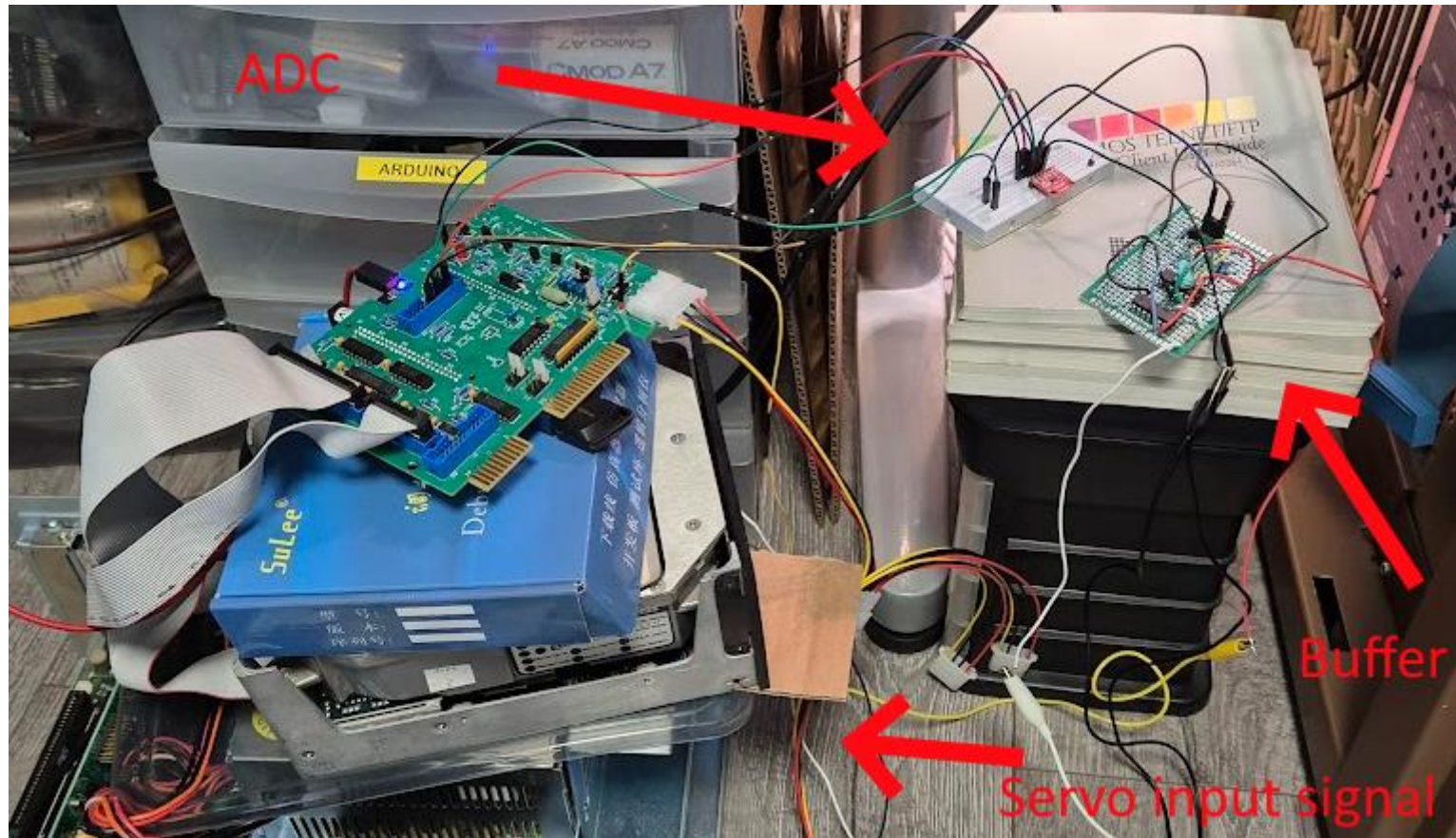
My Micropolis 1325



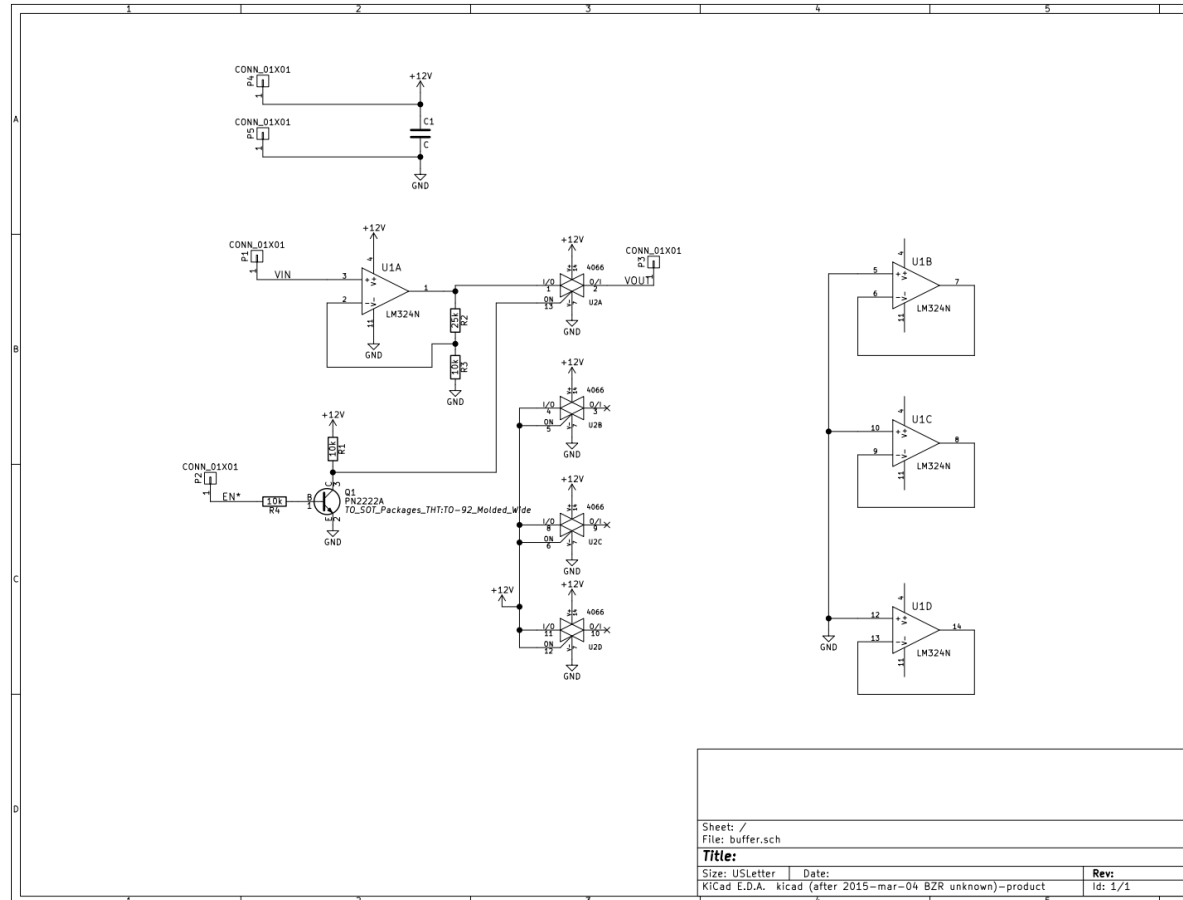
Diagram



Real world setup – very hacky



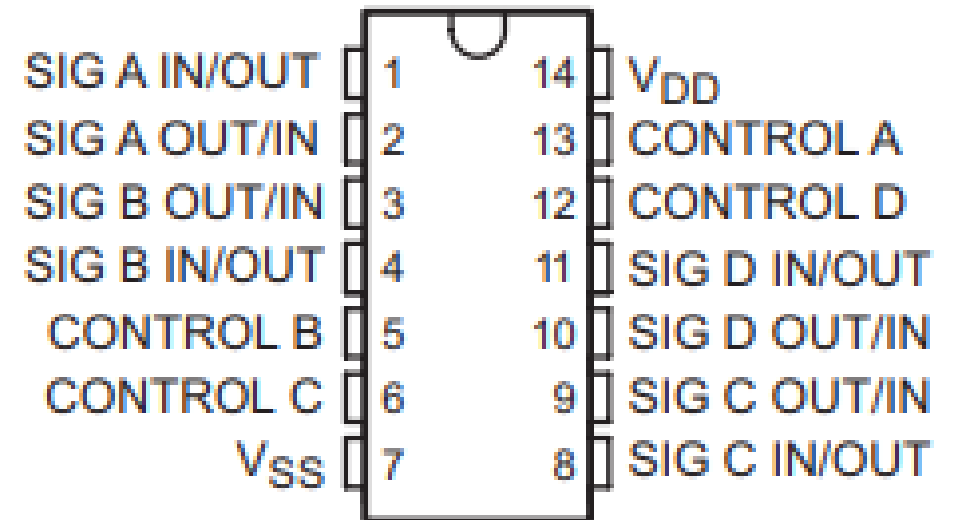
Schematic of buffer board



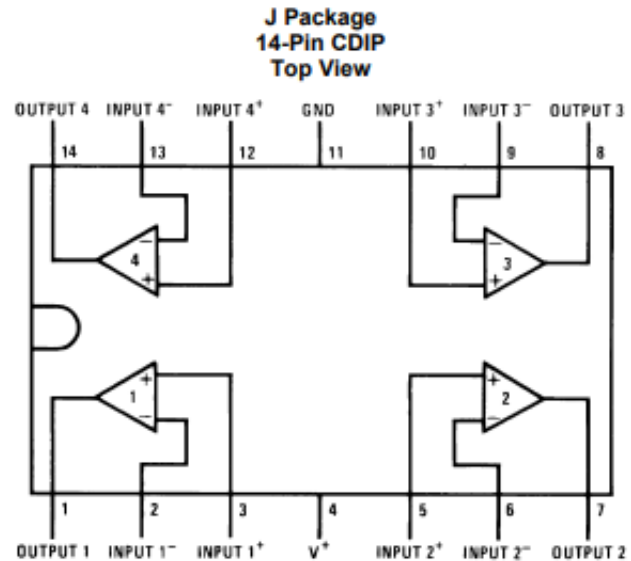
4066 CMOS Quad Bilateral Switch

Pin Functions

PIN		I/O	DESCRIPTION
NO.	NAME		
1	SIG A IN/OUT	I/O	Input/Output for Switch A
2	SIG A OUT/IN	I/O	Output/Input for Switch A
3	SIG B OUT/IN	I/O	Output/Input for Switch B
4	SIG B IN/OUT	I/O	Input/Output for Switch B
5	CONTROL B	I	Control pin for Switch B
6	CONTROL C	I	Control pin for Switch C
7	V _{SS}	—	Low Voltage Power Pin
8	SIG C IN/OUT	I/O	Input/Output for Switch C
9	SIG C OUT/IN	I/O	Output/Input for Switch C
10	SIG D OUT/IN	I/O	Output/Input for Switch D
11	SIG D IN/OUT	I/O	Input/Output for Switch D
12	CONTROL D	I	Control Pin for D
13	CONTROL A	I	Control Pin for A
14	V _{DD}	—	Power Pin



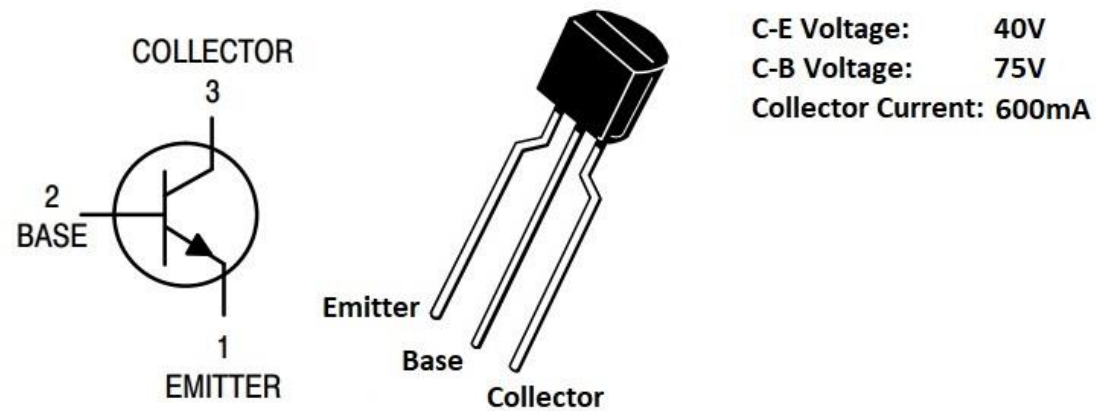
LM324 Quad-Operational Amplifiers



PIN		TYPE	DESCRIPTION
NAME	NO.		
OUTPUT1	1	O	Output, Channel 1
INPUT1-	2	I	Inverting Input, Channel 1
INPUT1+	3	I	Noninverting Input, Channel 1
V+	4	P	Positive Supply Voltage
INPUT2+	5	I	Nonverting Input, Channel 2
INPUT2-	6	I	Inverting Input, Channel 2
OUTPUT2	7	O	Output, Channel 2
OUTPUT3	8	O	Output, Channel 3
INPUT3-	9	I	Inverting Input, Channel 3
INPUT3+	10	I	Noninverting Input, Channel 3
GND	11	P	Ground or Negative Supply Voltage
INPUT4+	12	I	Noninverting Input, Channel 4
INPUT4-	13	I	Inverting Input, Channel 4
OUTPUT4	14	O	Output, Channel 4

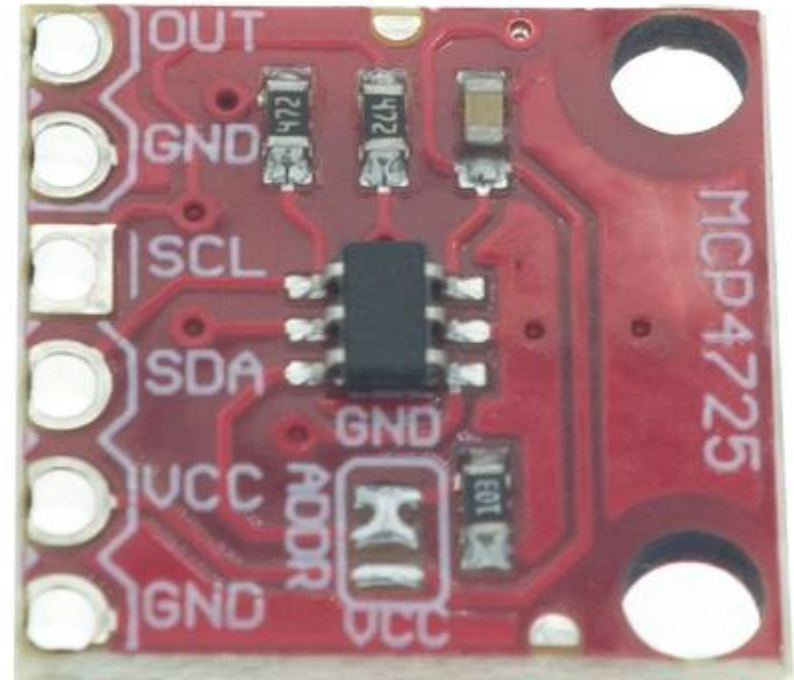
PN2222A NPN Transistor

PN2222A NPN Transistor



Input to Buffer Board pin P1/VINN

- Control signal comes from ADC MPC4724 signal "OUT"
- Power (3.3V), Ground, I2C data, I2C clock are on the grove connection.



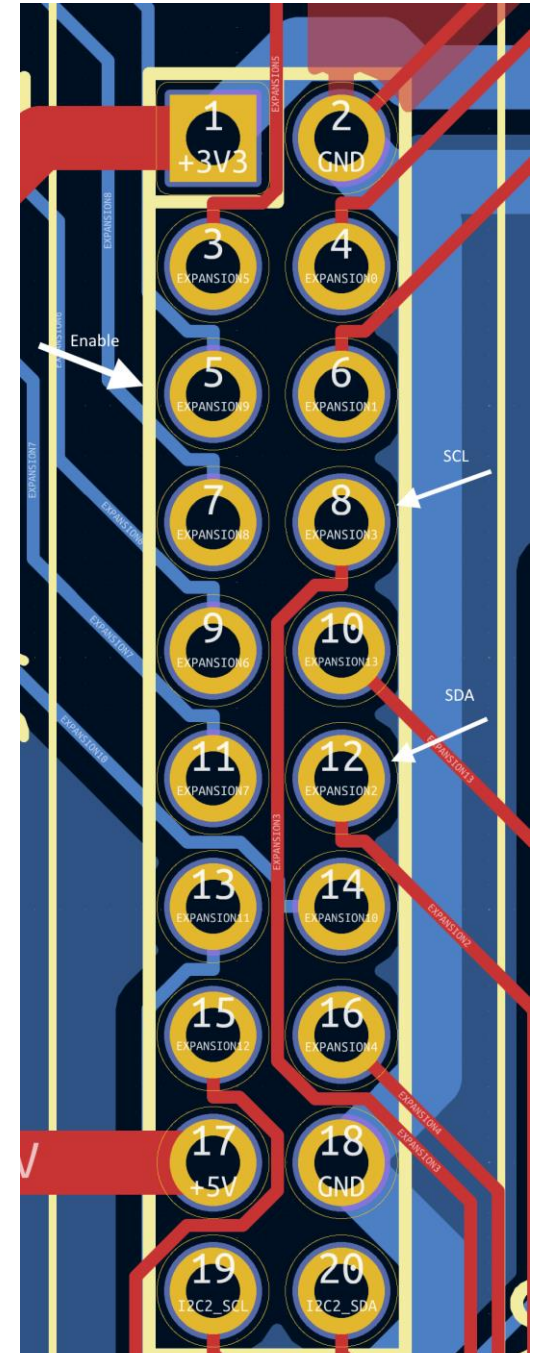
Connecting to J7 to get I2C2

Beagleboard P8_2

- Pin 21 - I2C2_SCL (mode 2)
 - J7 – 8 (Expansion 3)
- Pin 22- I2C2_SDA (mode 2)
 - J7 – 12 (Expansion 2)

Enable signal, Pin 9 (Gpio2[05]) ->P2

- J7 – 5 (Expansion 9)



Set PIN mode in the OS for I2C-2

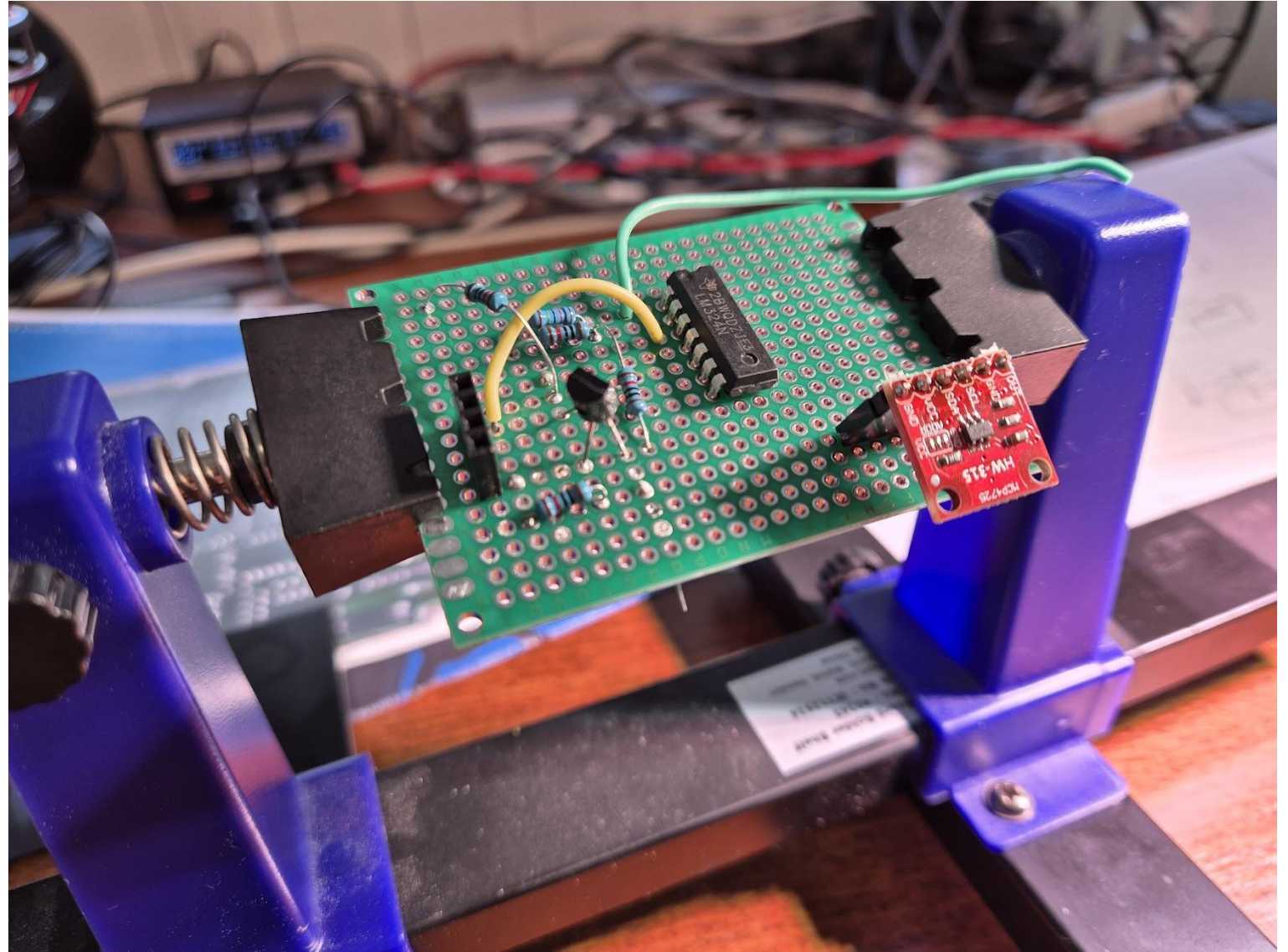
```
$ config-pin P9_21 i2c
```

```
Current mode for P9_21 is:      i2c
```

```
$ config-pin P9_22 i2c
```

```
Current mode for P9_22 is:      i2c
```

Buffer Board and ADC



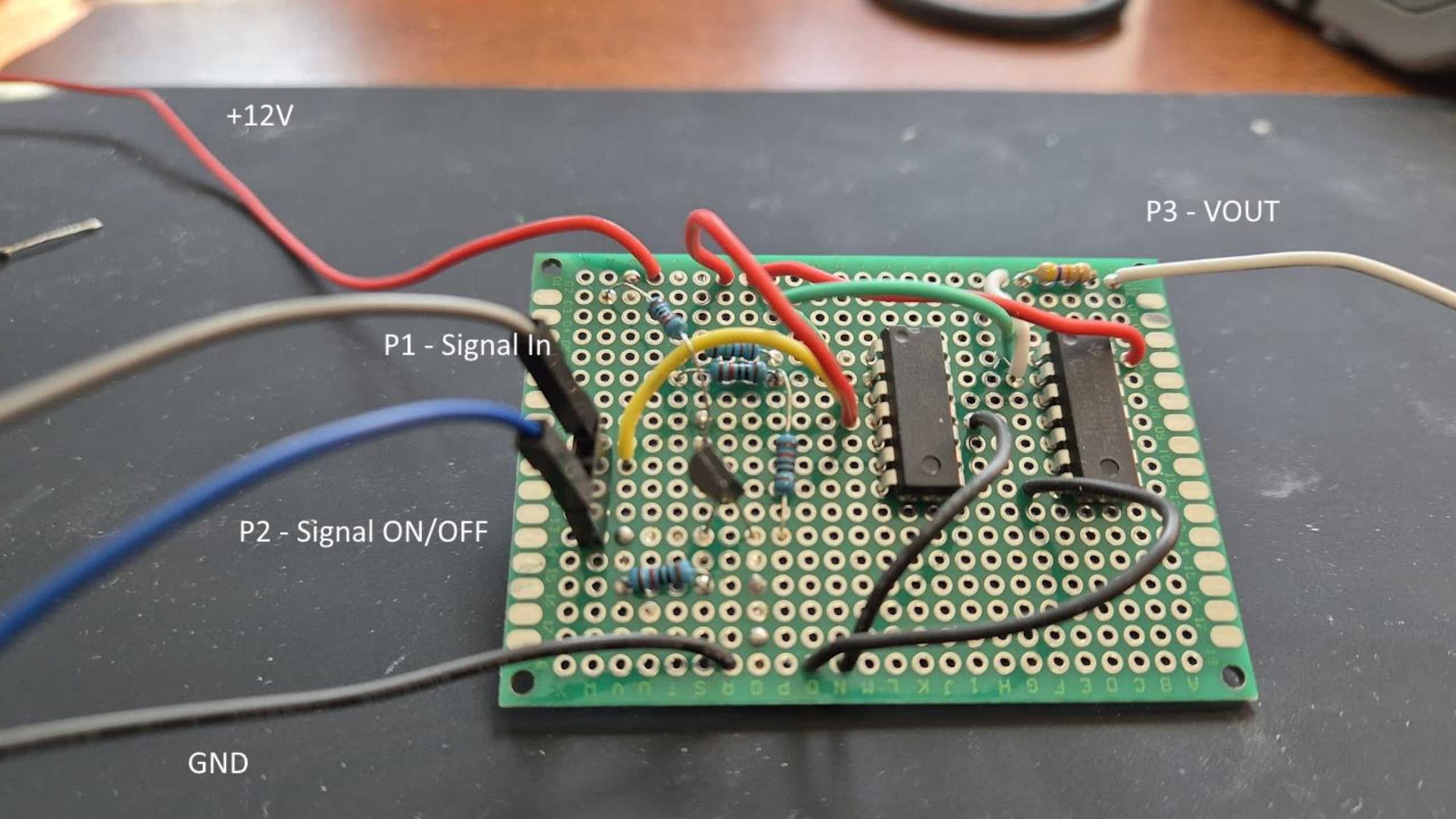
+12V

P3 - VOUT

P1 - Signal In

P2 - Signal ON/OFF

GND



Software – info from David

- It is a branch of 1.40. I don't plan to keep it up to date with current code since this was an infrequent need for me and I don't expect enough other people to be using it to be worth the effort to maintain. The code for controlling the DAC is in **drive_read.c** and likely will need adjustment for other drives. If you drive too low or high an offset voltage the head servo will go unstable.
- The way this works is the servo head has a special pattern that the drive electronics converts to a voltage that represents how far the head is from the track center.
 - An error amplifier compares the voltage to the nominal center of track voltage to drive the head servo to bring the servo voltage to the nominal voltage.
 - This keeps the head centred on the servo pattern. I drive an offset voltage into the error amplifier to effectively change the nominal voltage which shifts the head position.
 - The code steps through the specified voltage range trying to find a voltage that will allow all the sectors to be read on that head.
 - It keeps track for each head the offset voltage to try as the starting voltage when it starts on the next cylinder.

From David

- The old code doesn't work with the 11.8 OS.
- Code works fine with rev C board.
- Try using the 7.11 OS images.
 - <https://www.pdp8online.com/mfm/revb/software.shtml>
- Conclusion:
 - Unable to boot 7.11 OS image on BeagleBoard WIFI as it panics when there is NO Ethernet.
- Solution:
 - Merging the special `mfm_read.cs` with the new release. Add support for DAC by including the `mcp4725.c/h` files

Reading the drive

- `/home/ronny/repos/mfm/mfm/mfm_read --format nd100_3041 --drive 1 --cyl 1024 --retries 100 --heads 8 --ignore_seek_errors --tranc3_raw --extracted_data_file micropolis_c3.img | tee -a mfm_log.txt`

Read status = 98.7% recovered

- Expected 73728 sectors got 72802 good sectors, 887 bad header, 39 bad data
- 0 sectors marked bad or spare
- 0 sectors corrected with ECC. Max bits in burst corrected 0

Source code available

- GitHub - <https://github.com/RonnyA/mfm/tree/servo>
- Tar file
 - <https://hackercorp.blob.core.windows.net/upload/mfm/mfm.tgz>
 - Contains also runnable executables