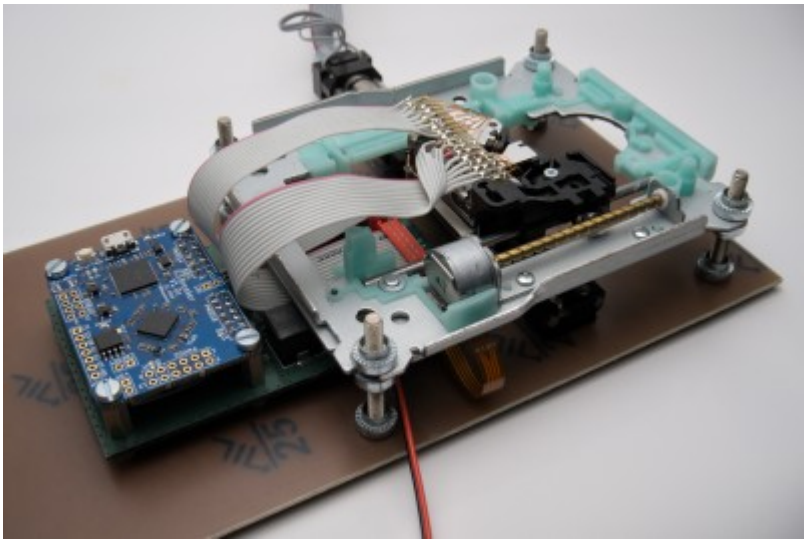
DVD-Drive-Based Laser Scanning Microscope



Due to an abundance of one specific type of DVD drives, plans were made to give a useful afterlife to these devices. A laser scanning microscope seemed to be the most useful project.

Preface

This project was inspired by the work done by Hannes Zöllner towards his Ph.D., during which he evaluated commercial DVD drives for fault injection in semiconductor devices. At some point, his system was also repurposed for imaging, with surprisingly remarkable results. An effort was made to optimize the system used in this project for imaging rather than for fault injection.

A DVD drive is a remarkable device, being able to discern the height of tiny pits of about 440 nm in length, laid out in tracks separated by only 780 nm. Of course, Blu-Ray technology decreased both these numbers further, but DVD drives tend to be thrown away in large quantities these days, which makes them available to the hobbyist basically free of charge.

Concept

The OPU (Optical Pickup Unit) of a DVD drive normally contains all the optical and most of the electrical components required to read data from a disk. Normally, it includes laser diodes (and sometimes their driver electronics), a multi-quadrant detector diode (PDIC or OPIC), the required optical elements (lenses, mirrors, prisms, gratings) and the actors for focus, tilt and tracking control (typically electromagnetic actuators moving the primary lens close to the disk). Normally, the OPU is mounted on a linear rail equipped with a stepper motor for one-axis position control.

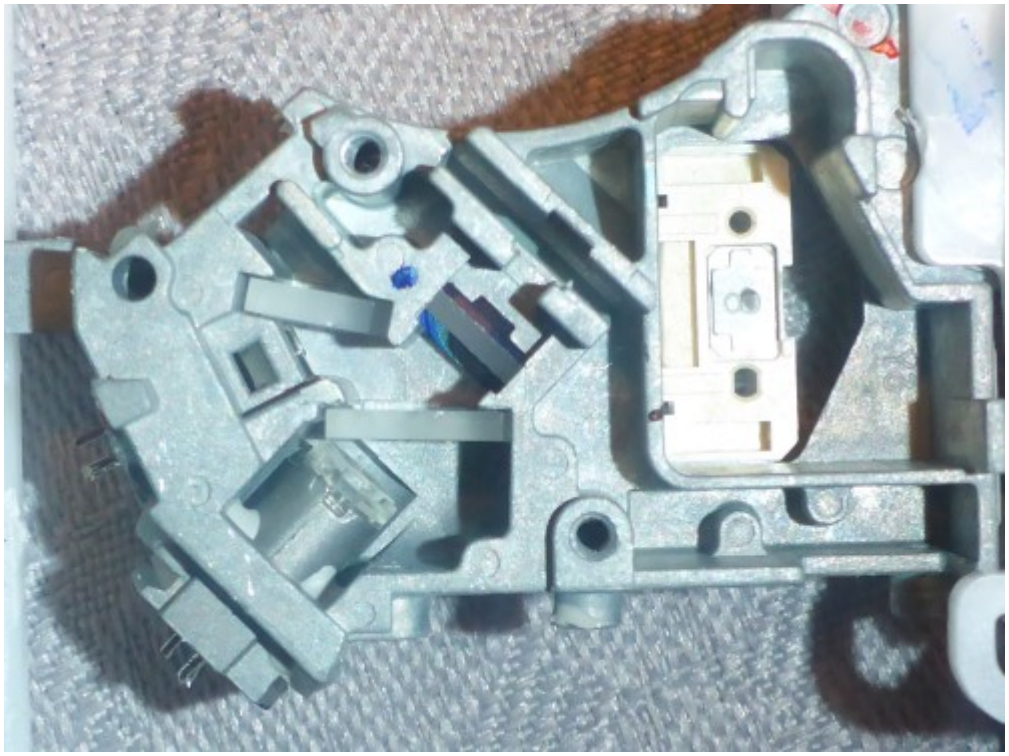
In the available drives, all OPU components are interfaced using a single FFC (flat cable) to a custom board that houses a proprietary microprocessor and various components not quite available to the average consumer. Therefore a replacement for these electronics that utilizes commercially available components open source hardware/software should be created.

This replacement should provide the following basic functionalities:

- Interface to PC for data acquisition
- Driving of OPU actuators (lens coils)
- Driving of laser diode(s)
- Readout of the PDIC for focusing and image acquisition

OPU Architecture

The OPU in the available drives is a tightly integrated component, mostly consisting of a single cast-aluminum block, in which various optical components are glued. A top-down-view of the OPU after removing the cover can be seen below.



Laser Diodes, Lens and Actuators

The OPU houses two laser diodes. As this is just a DVD/CD reader, the lasers are not very powerful but provide sufficient power for microscopic applications. The (red) DVD diode starts laser operation above approx. 60 mA and requires about 2-2.5 V of forward voltage. Its polarity can be easily determined by measurements.

The primary lens is positioned using electromagnetic actuators. These come in the form of three coils, which tilt or move the coil proportionally to the current flowing though them. Their polarity and required full-scale DC actuation current can be easily found by using a lab power supply.

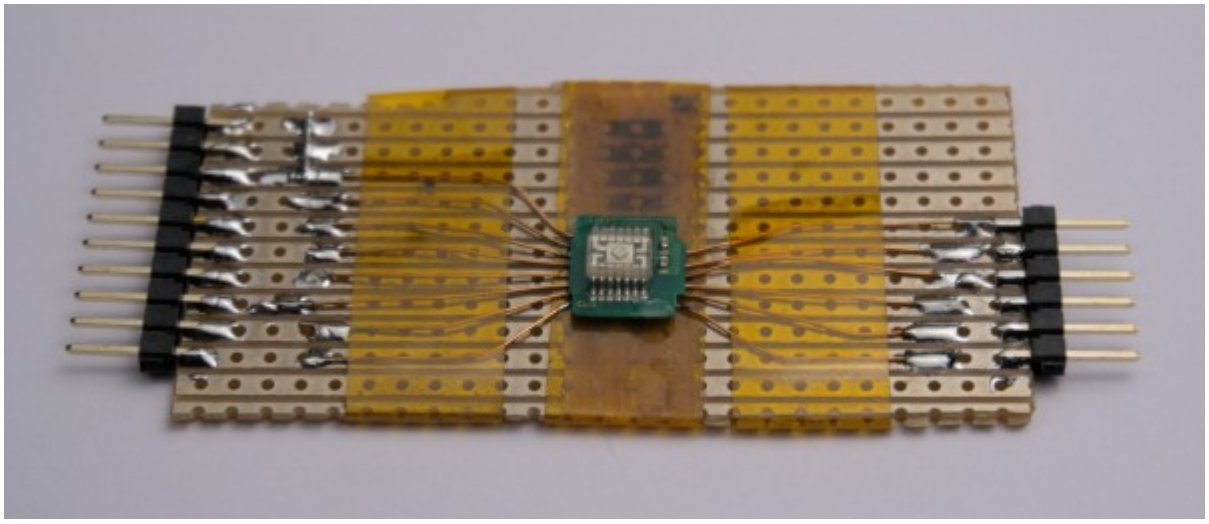
The focusing control loop driving these coils is normally deriving its error signal from the astigmatic focus error detection method. It requires (in its basic form) a four-quadrant diode, and focus is optimized when all quadrants of the diode are equally illuminated. Deviation from this point will cause the circular illumiation to become elliptical and therefore illuminate two diagonal quadrants more

strongly. A schematic explanation of that scheme can be found here and in many textbooks.

PDIC

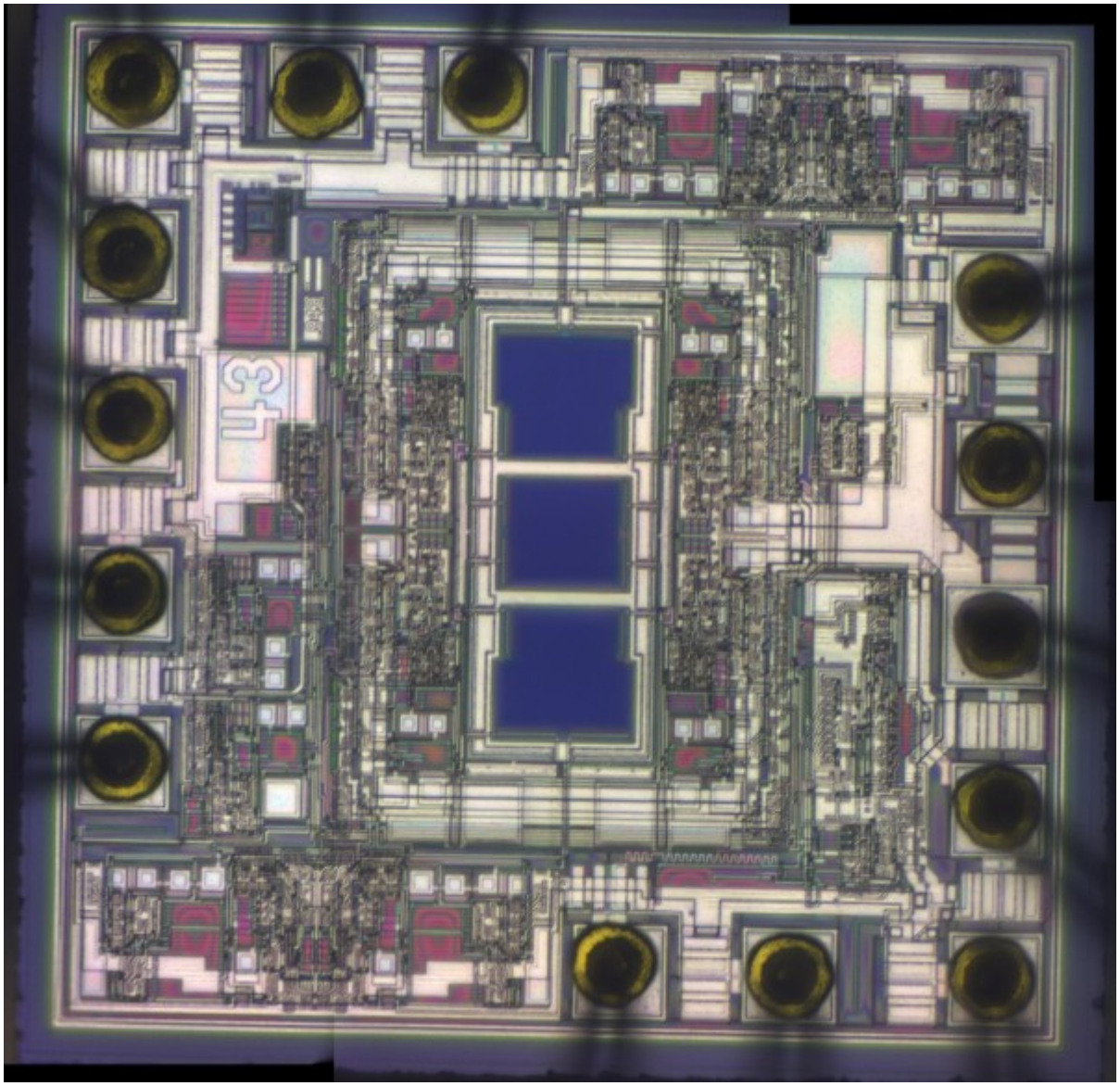
The PDIC (Photo Diode Integrated Circuit), in a DVD reader application, serves multiple purposes. These include gathering focus, tilt and tracking information from the disk as well as recovery of the RF data stored on the disk tracks. For this reason, PDICs typically consist of at least 6 active diode areas, sometimes even more. The PDIC used in the available drive type is seen below. Given its a very niche device, oftentimes custom-designed for a specific drive vendor, datasheets are hard to come by. Initial pinout probing revealed only the power supply pin locations and a number of analog output pins, but I was unable to find any matching ICs datasheets, so some reverse engineering effort was required to figure out its secrets.

First, the device was placed on a little break-out PCB to allow easy access to all of its pins, especially during microscope inspections.



PDIC Reverse Engineering: Optical

Using an IC inspection microscope, a die shot of the PDIC was produced. This was useful to get a better idea of the internal structure of the device. This process is made way easier by the fact that it is an optical integrated circuit, and is therefore housed inside a clear plastic package with good optical properties, suitable for direct microscope inspection without decapsulation. The picture below could be used to identify some functional blocks of the circuit, however full tracing from the bondpads through internal circuits to the diode quadrants didn't really go anywhere for me. However, it became apparent that the PDIC apparently houses at least independent eight photo diode channels. (Four quadrants in the center square, and at least two segments for each of the outer squares).



PDIC Reverse Engineering: Electro-Optical

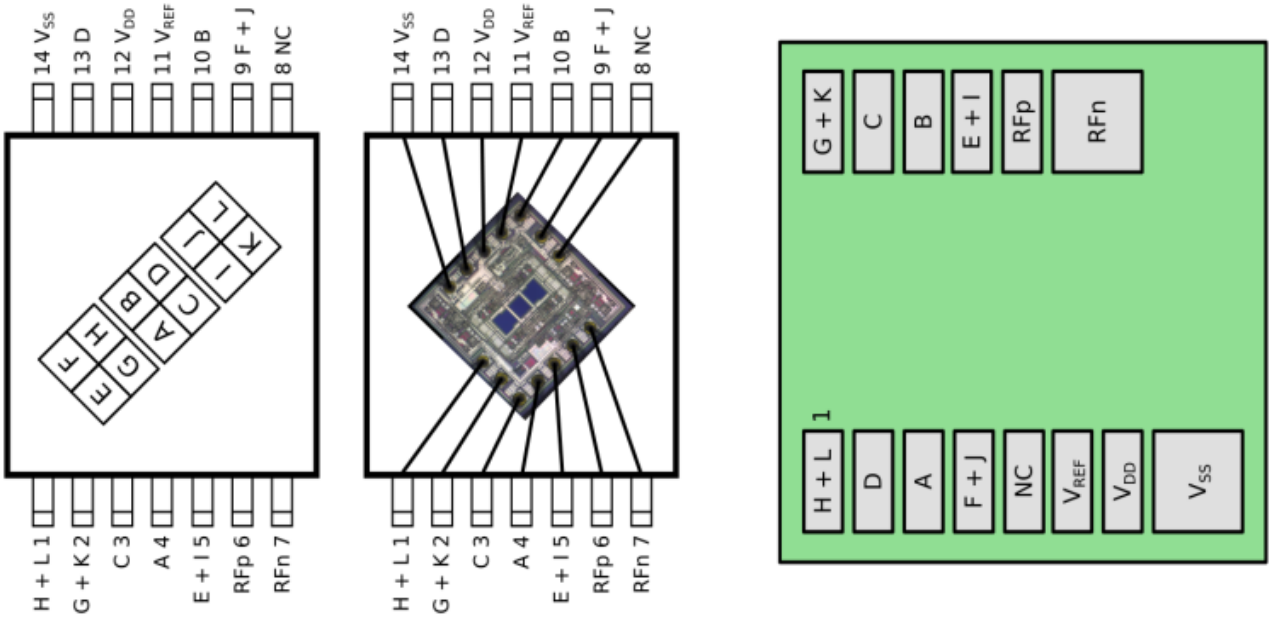
On most of the pins, some change in output voltage could be seen when illuminating the device with a bright source of light. At first, the output behavior of each pin was simply put into different categories:

- Low light sensitivity: 4 outputs
- High light sensitivity: 4 outputs
- Same magnitude, but opposite polarity outputs (differential pair): 2 pins
- Negative output voltage: 1 pin

It was conjectured that the smaller inner quadrants were responsible for the low-sensitivity outputs, while the outer segments would produce a large amplitude (due to their larger light collection area).

In a second reverse engineering session, the exact pinout was determined by means of selective illumination: For a given test, two IC output pins were compared using two DVMs, and an aperture was slowly moved along the surface, either vertically or horizontally. This would in a change of the output voltage for one of the outputs slightly earlier than for the other one. In this way, the positional relationships between all output pins could be resolved, forming a final assignment of diode

quadrants to output pins.

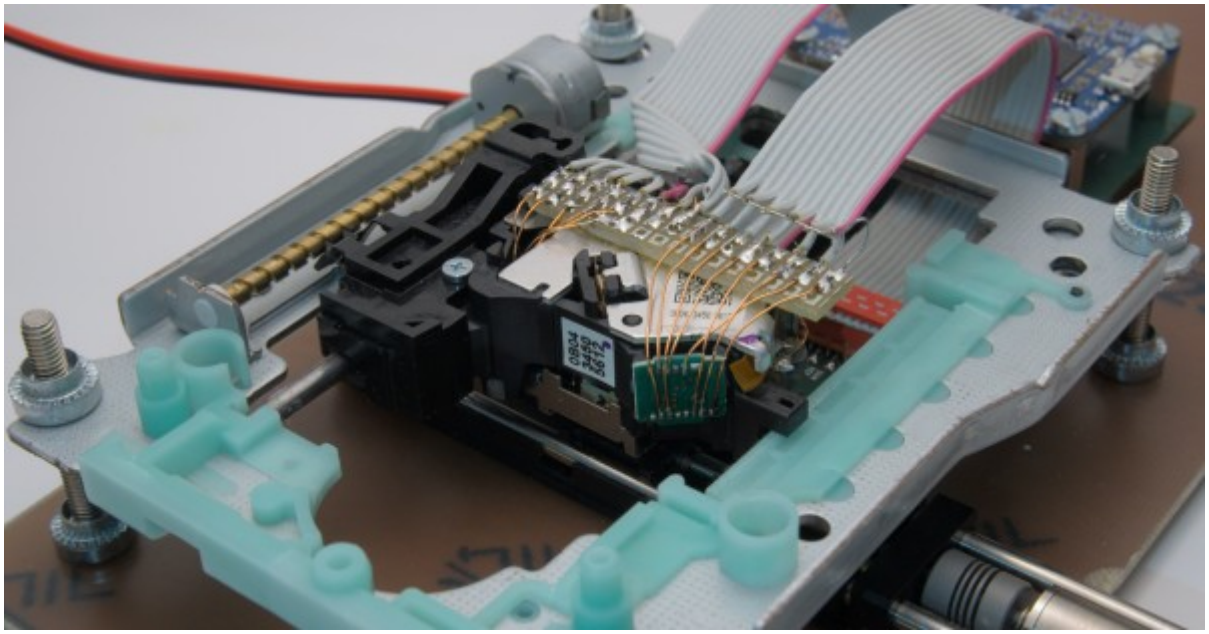


The output producing negative voltages is not considered to be useful in the final application, while the differential output pair is likely a summed output of the four inner quadrants (the high bandwidth RF output). With the assignment of pins to diode quadrants done, the OPU could be modified to fan-out the required connections to custom electronics

OPU Modification

The whole top assembly of the DVD drive is reused. This includes not only the OPU, but also the linear (worm gear) axis driven by a stepper motor and its base plate. A breadboard header was added to the OPU, fanning out all required signals to the custom electronics on a ribbon cable. The analog output signals are separated on one side, while the current inputs for the laser diode and the lens actuators are kept to the other side. This should help mitigate electrical coupling between these signals, even though the diode outputs are actively driven by the PDIC.

The other axis (below the drive mechanism) is built from a commercial precision linear actuator that was donated to the project. Other people have simply used another DVD drive in its place, which would certainly be a possibility here.



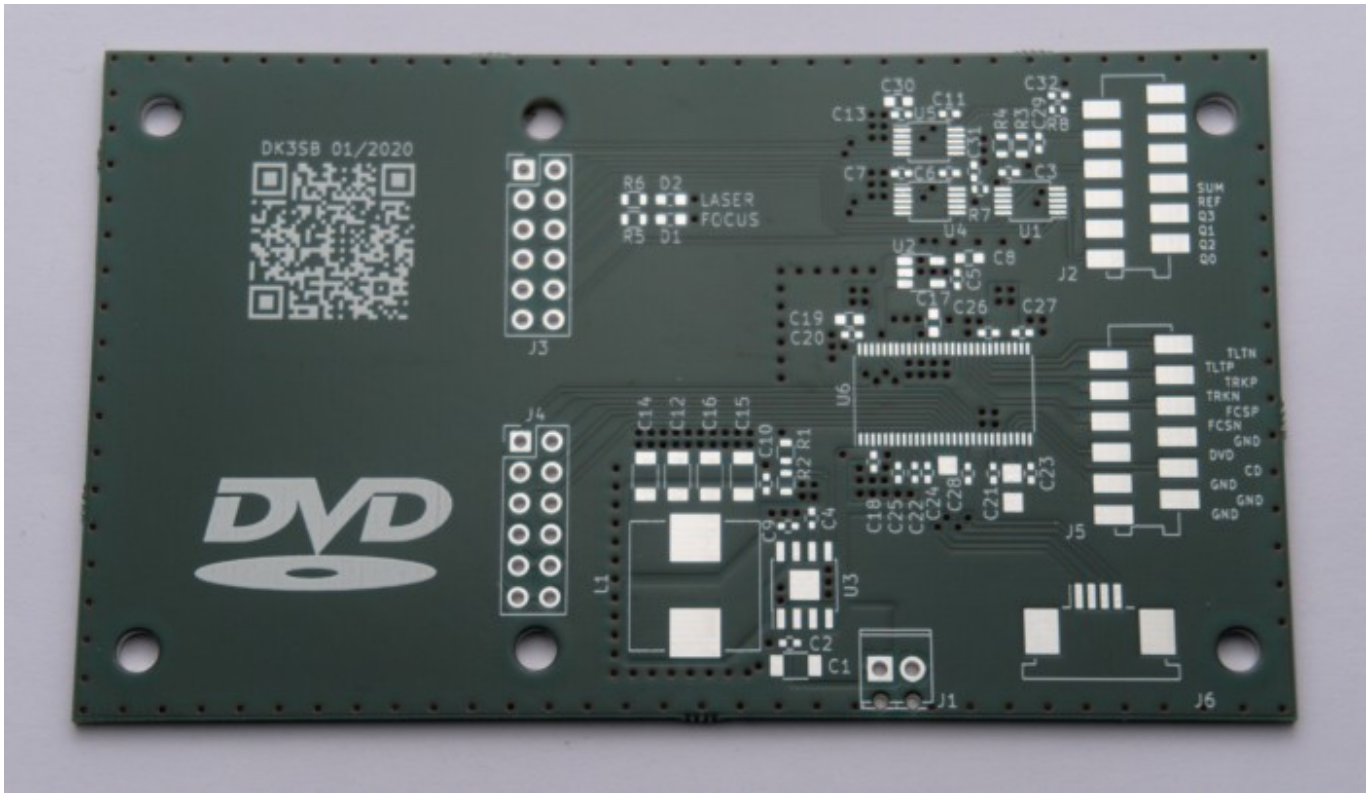
Electronics

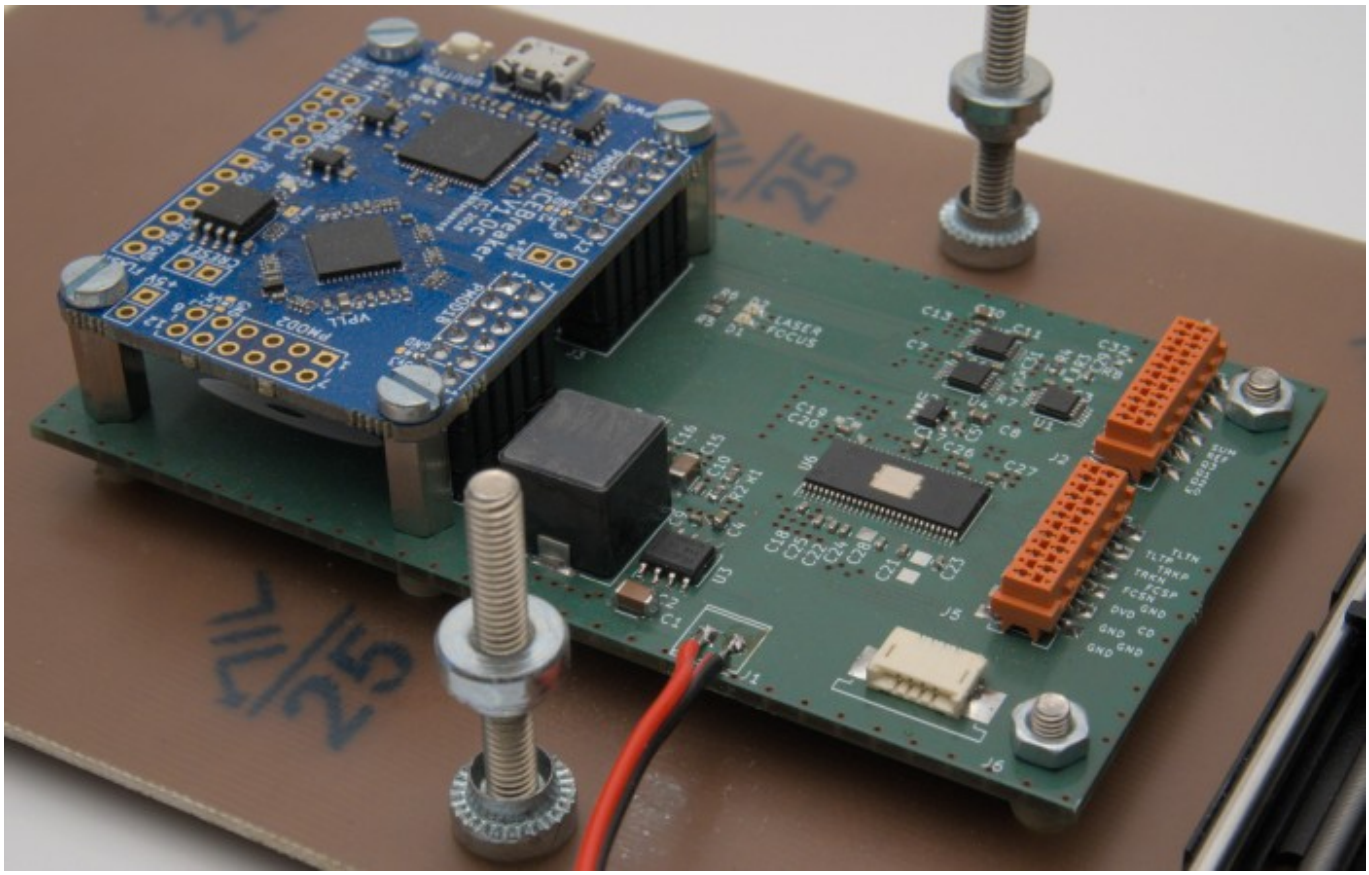
A PCB was designed that would fulfill all necessary interfacing between PC and the optical components. Those would be:

- Drive motors for X/Y axes
- Drive the lens actuator coils (constant current/voltage)
- Drive the laser LEDs (constant current)
- Provide power supply

It was pretty obvious that at least all the mechanical actuation and laser driving could be handled by an IC foreseen for an optical drive. Texas Instruments offers a line-up of such chips, which are one of the few types that can easily be sourced from common distributors. The TI TPIC2050 was chosen due to its all-digital interface and flexible output options. Given it's a pretty high-quantity component, its cost is very reasonable (€2.50 in single quantities) and can hardly be beaten by any other integrated options.

A PCB was designed around this IC, with an additional 16 bit ADC plus analog MUX for the PDIC outputs. The PCB was built to house an iceBreaker FPGA board that would take care of all the interfacing. All electronics were designed in KiCad and are available publicly in its GitLab repository.





FPGA Design

The FPGA had to fulfill a couple of interfacing tasks, and the design uses the FOSS FPGA toolchain for the Lattice ice40 UP5k FPGA on the iceBreaker board. It mainly does the following things:

- Automatically cycle through ADC MUX channels and read out photo diode currents
- Provide register-level access to the TPIC2050 from the host software
- Provide fast write access to the TPIC2050 DACs
- Allow communications to the host PC using a 1Mbps USB-Serial connection

The design itself is quite straight forward, but I had a bit of fun with the verification environment. I used cocotb and iverilog for all the simulations. The testbenches contain proper hardware device models for all ICs as well as the optical chain. The simulated UART interface can be connected to a virtual serial port, which can then be stimulated by the real host software. This allows running a full „simulated hardware in the loop“ test, which is quite a neat thing to have.

Setting a focus actuator current would alter the optical chain, resulting in different values digitized by the ADC for the various photo diode current channels. Using some guesstimates for the actual optical properties of the system, the whole software focusing algorithm could be prototyped without any actual hardware.

All the gateware and verification environment is also available in the corresponding GitLab repository.

Software

Architecture

The software essentially only needs to consist of a couple of drivers for all the hardware components. It give the user some higher-level access to the TPIC2050 functionality (stepper motor driving, laser current settings, end stop detection) and abstracts away a few of the optical system properties: Scripts can simply get the focus and tilt error terms already computed from the raw photo current ADC readings and implement their own focus control algorithms on top of that.

X-Axis Linear Actuator

Another driver takes care of running the X-Axis, which was implemented by a linear actuator that was kindly donated to the project by Szymon. It uses a somewhat old (and buggy!) serial-based controller unit (National Aperture MC-4B Series), for which I managed to dig up the manual from the Internet Archive, see here:

mc-4bmanual2_1.pdf

. Sadly, its firmware is terribly buggy and somehow manages to misunderstand some of the simple commands it should understand from time to time. This makes it go to the wrong position once in a while, which results in nasty artifacts in acquired images. At some point, this controller will need to be replaced…

Main Software and Algorithms

The main laser microscope class provides a few methods for focus control and scanning a predefined area to an image file. The focusing is the most interesting aspect here. Generally, the focusing first performs a wide scan of the Z-height to find the linear region of the focus error detector. Following that, it implements a simple PI control scheme for successively minimizing the focusing error.

As relatively large scan areas should be possible, focus tracking during the scan is required. Initially, the software uses a relatively simple algorithm to achieve this:

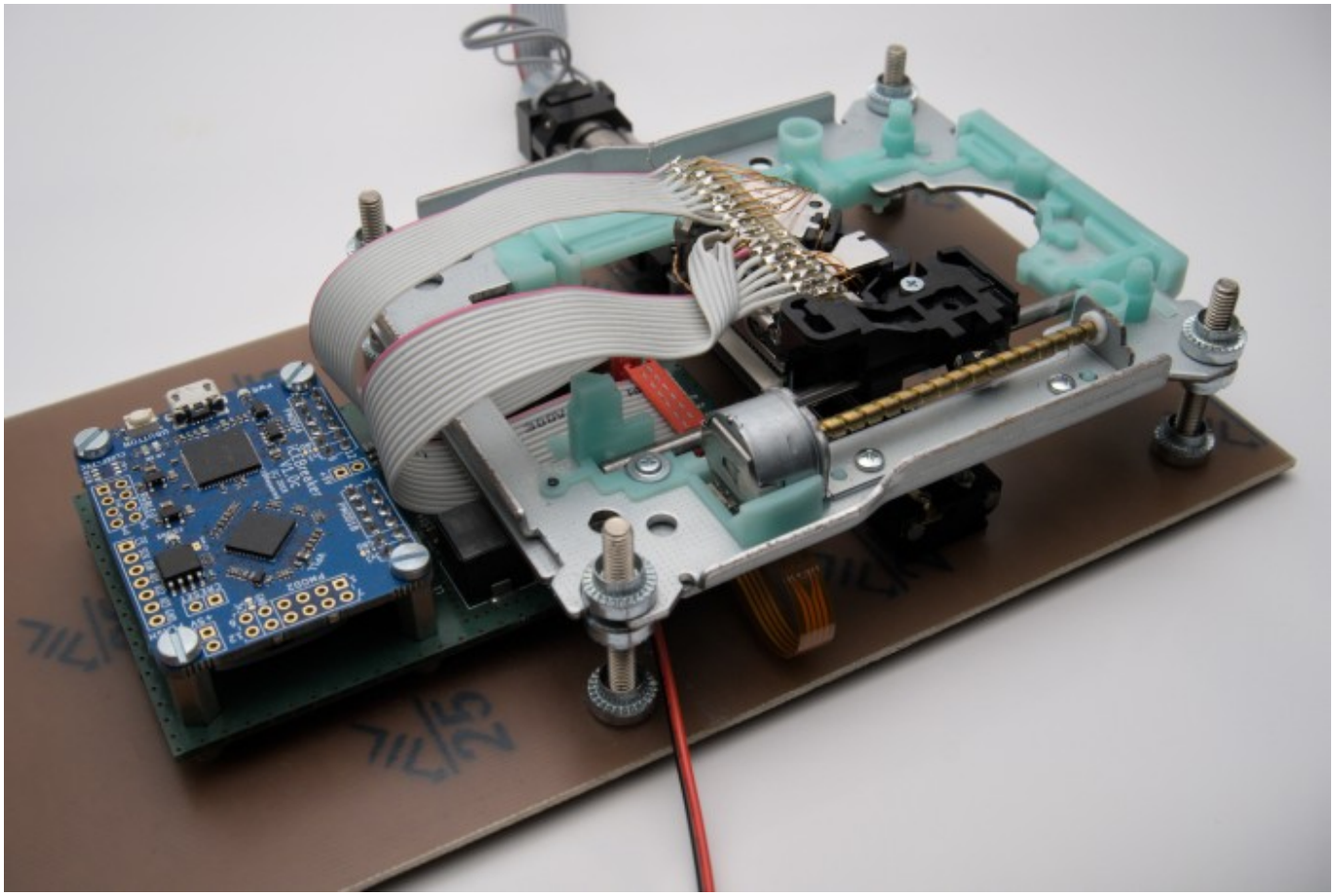
- Inside the rectangular scan area, place a virtual grid with at least N points per millimetre
- At each grid point, perform one focus acquisition and store the focused Z-height
- Least-Squares-Fit a 3D plane through all the points on the grid
- During the actual scan, calculate the optimal Z-height for a given position by evaluating the Z-Coordinate of this plane

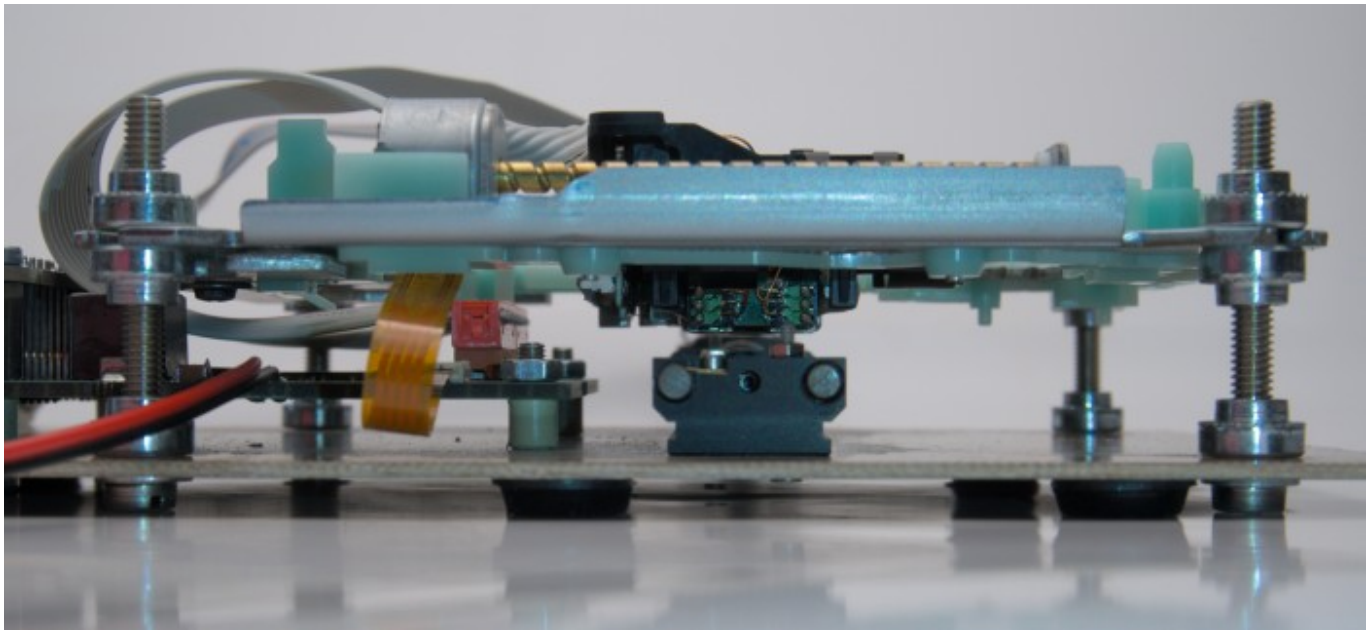
Another noteworthy aspect is the Y-Axis (implemented using the DVD drive motor and mechanism). Given this whole mechanism is a made from a few cheap plastic pieces, it has some backlash to it. This could be reliably solved by only approaching positions from one side. This yields surprisingly good repeatability (sub-µm-range), but the software needs to account for that and will intentionally overshoot any positioning targets in the non-preferred direction before approaching them from the preferred side. Also, this axis was found not to be terribly linear. No big surprise here either, as its driven from a low-precision worm gear. With large scan areas, this will reflect into periodic stretching artifacts of the images (to be seen below).

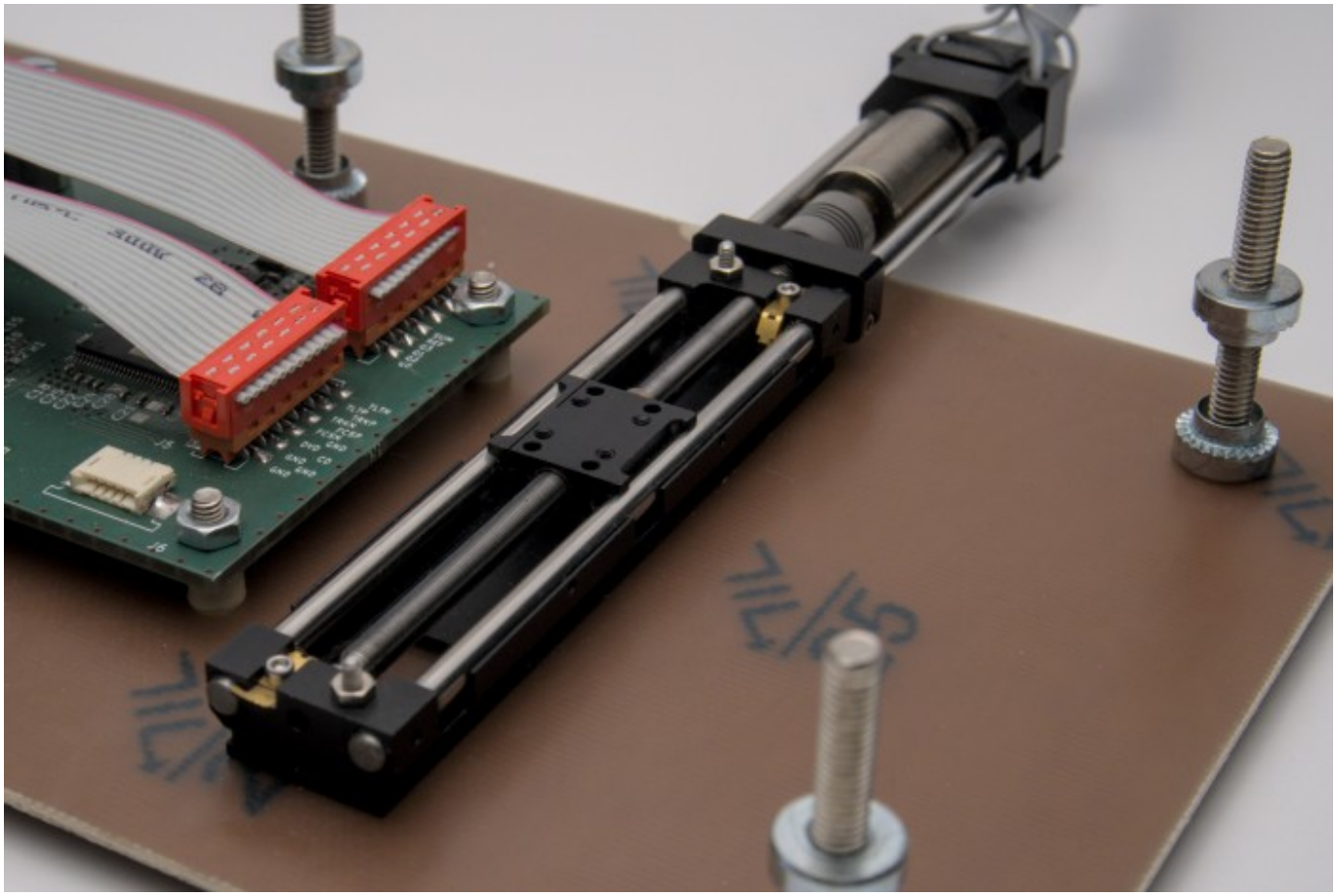
The software is as open as everything else (though a little more hacky), and can be found in its GitLab Repository as well.

Full System and Mechanical Assembly

The following two pictures show the system fully assembled. Also below is a picture of the linear actuator used for the X-Axis. This is a wonderfully precise instrument (National Aperture linear stage with sub 100 nm positioning accuracy). Its only downfall is the controller (see my ramblings above in the Software section).

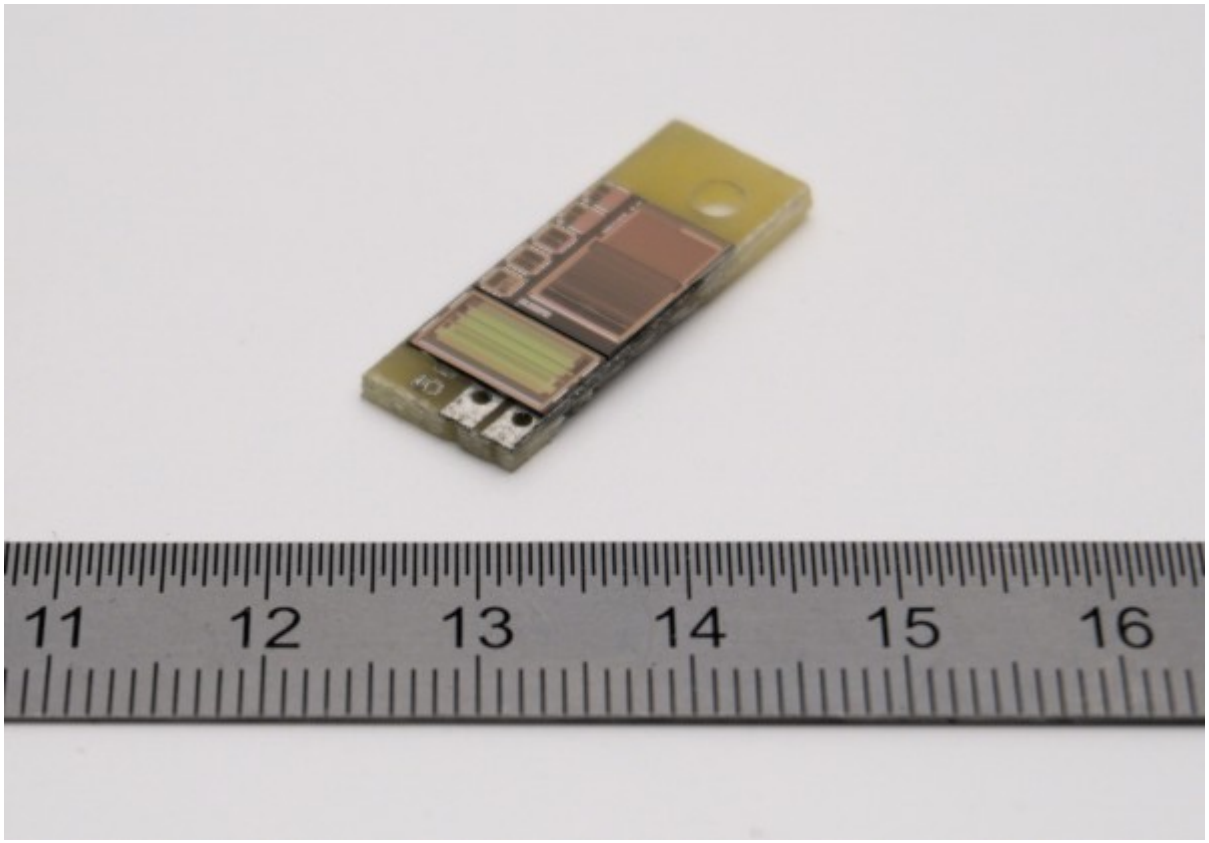






Sample Preparation and Imaging

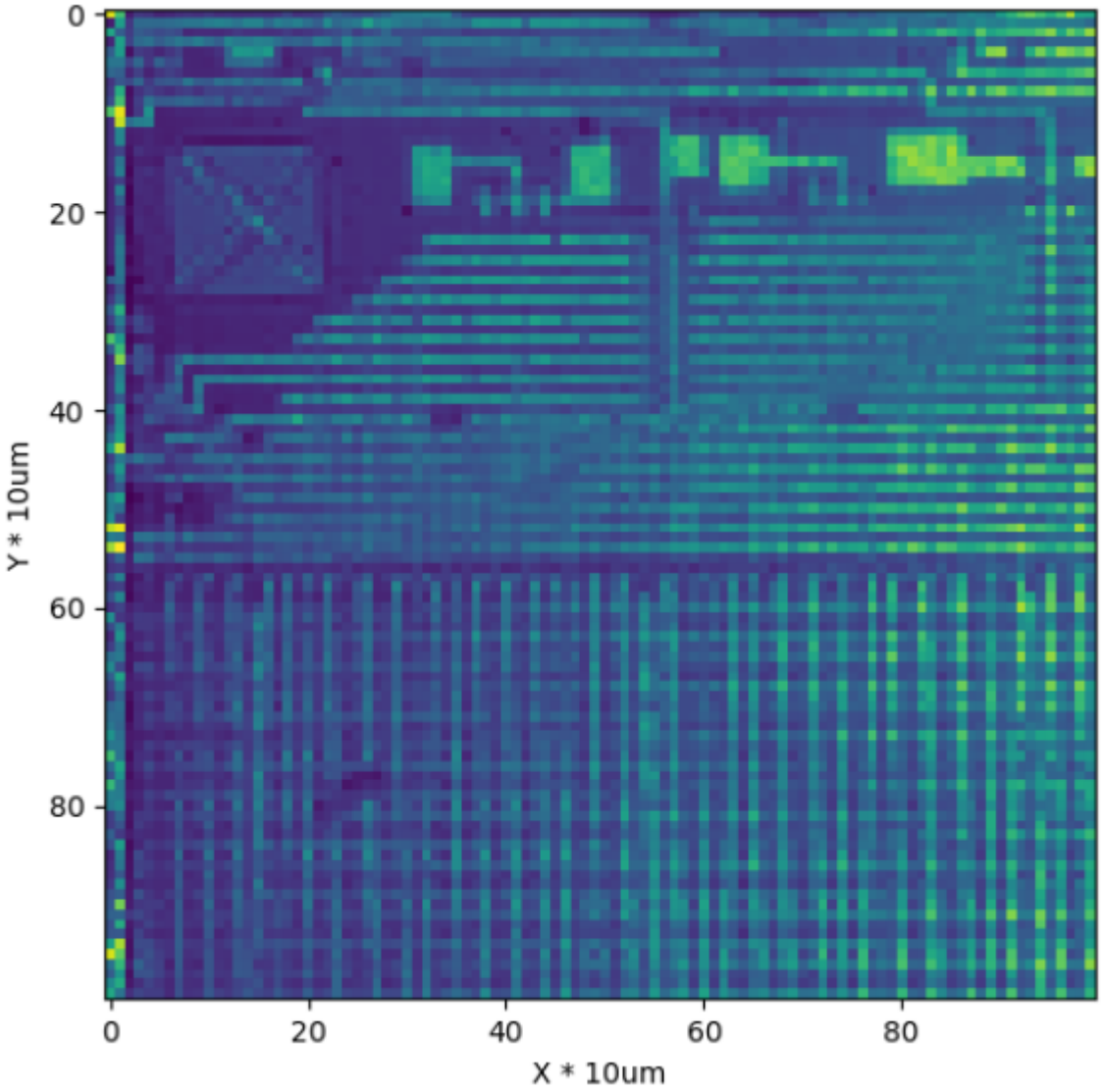
With the full system in place, all that remains is to place a sample on the X-stage and start alignment and scanning. For sample preparation, I simply glue samples on a more manageable piece of perf board, like I did below for some random 250 nm CMOS chips.



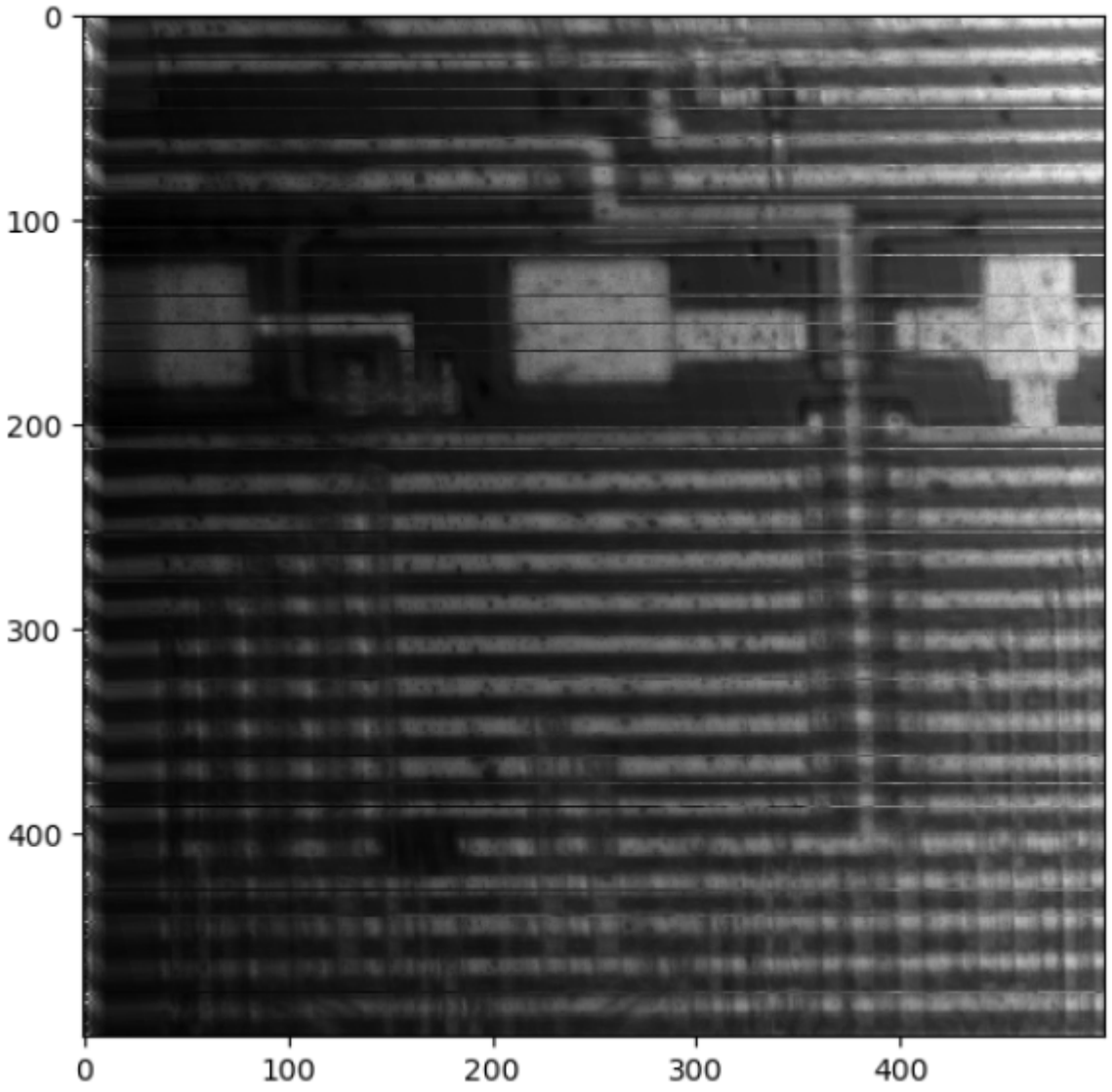
Alignment is a somewhat tricky process, but using the laser at low current (to produce a just-visible dot on the sample without causing any laser safety concerns) helps a great deal.

The scanning process can last from a few minutes to a few hours, depending on the required step size (quality) and area of the scan. Here's a few examples of resulting images:

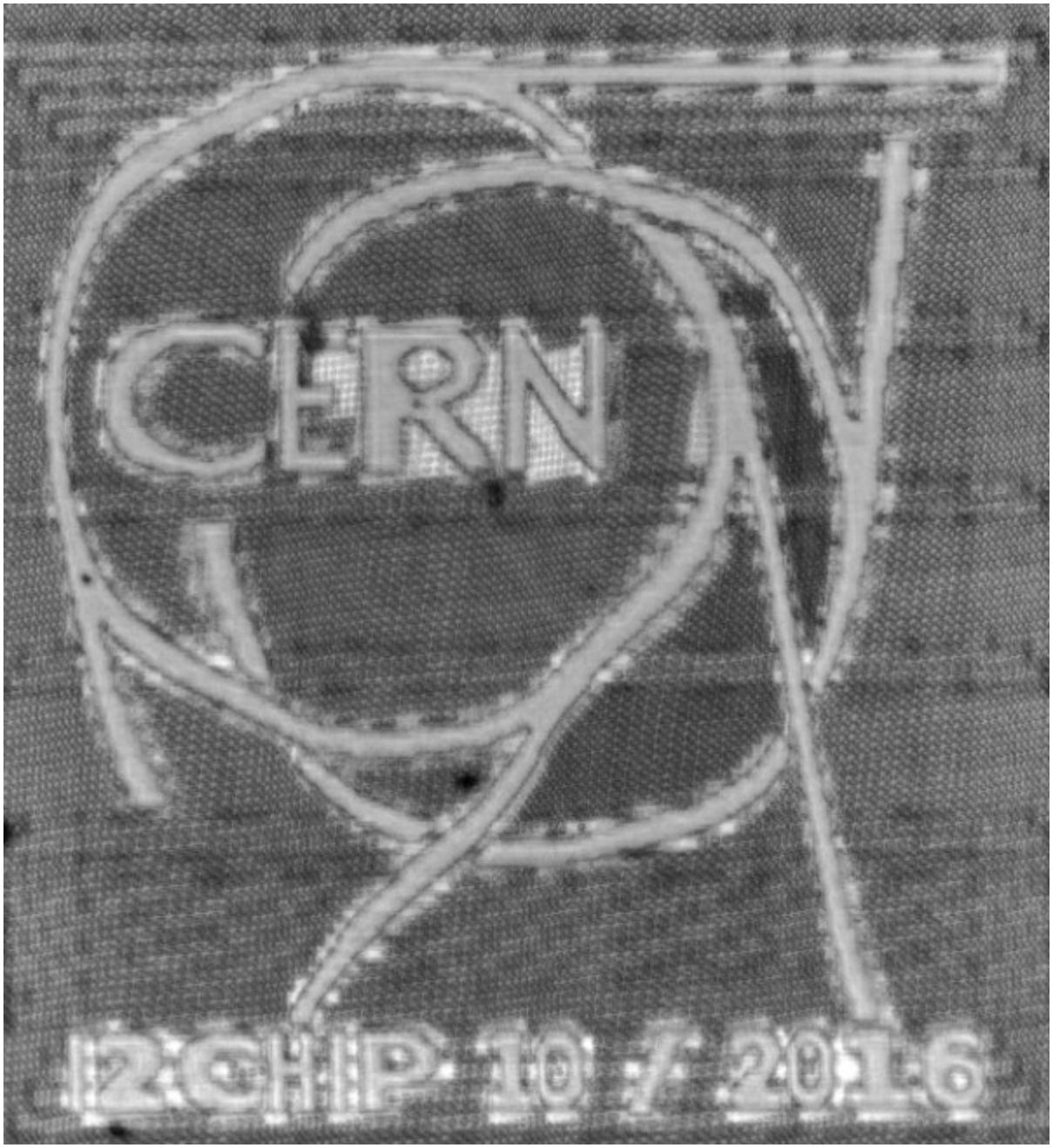
Coarse imaging of some antique EPROM device, scan area 1x1mm. (Picture Below)



Higher resoultion scan of the same device EPROM, scan area 500×500 um. The two routing layers can be seen above one another. (Picture Below)

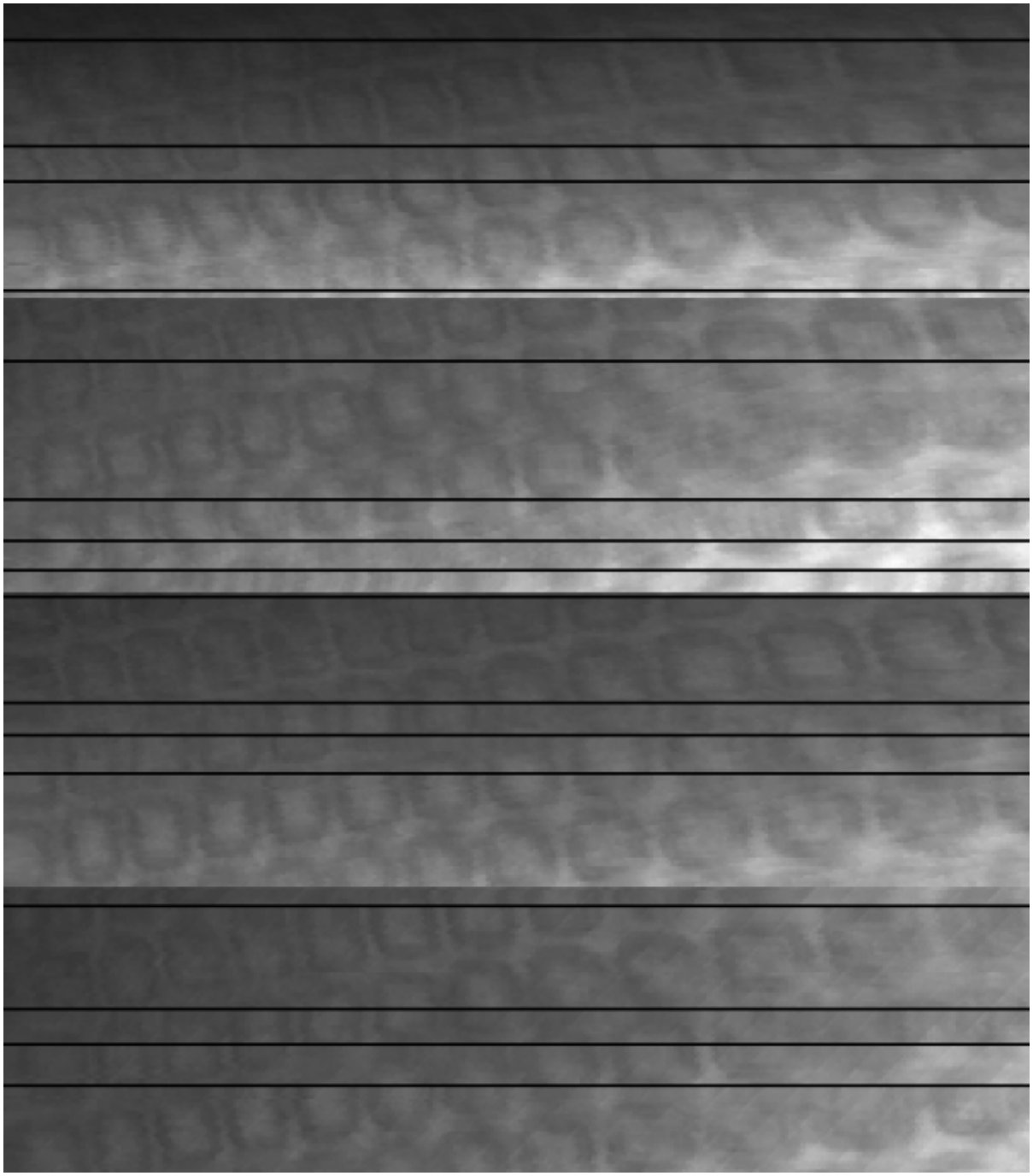


Scan of an I2C test chip from my colleage Szymon (fabricated in 65nm CMOS, picture below).



This picture shows some of the aforementioned nonlinearity in one axis, which introduces periodical distortions in the image. These could be corrected in principle, but I didn't bother. The nice CERN logo was placed on the top metal layer, and is composed of many small pixels that are on the order of a few micrometers large.

Also from this chip, here's a few very small scans performed at 200 nm step size. Small metal filling patterns on the top metal layer are imaged, each of which is only about a two micron wide square. Lithographic imperferfections can clearly resolved using this imaging technique. The horizontal (black) lines were caused by the erratic X-axis stage, sometimes going to places it was not told to go...



Summary

For not a lot of money, a rather capable little laser microscope could be built. At some point, more scanned objects will be added to the collection above. If you are interested in replicating this project or a part of it, all developments are publicly available in the following repositories:

- Hardware: https://git.loetlabor-jena.de/dvdlaser/dvdlaser_hw
- FPGA Gateware: https://git.loetlabor-jena.de/dvdlaser/dvdlaser_fpga
- PC software: https://git.loetlabor-jena.de/dvdlaser/dvdlaser_sw

From:
https://loetlabor-jena.de/ - **Lötlabor Jena**

Permanent link:
https://loetlabor-jena.de/doku.php?id=projekte:dvdlsm:start&rev=1612657689

Last update: **2021/02/07 00:28**