HOCHSCHULE RHEIN-WAAL
Rhine-Waal University of Applied Sciences

Faculty of Technology and Bionics

Developing Computer-Aided Laser Lithography for PCB Prototyping

Bachelor Thesis

Arthur Grilo da Silva Santos

Supervisors:
Prof. Ronny Hartanto
M. Eng. Frank Kremer

Kleve, Thursday 13th May, 2021

Declaration of authorship

I hereby declare that the work presented in this thesis is solely my work and that to the best of my knowledge this work is original, except where indicated by references to other authors. No part of this work has been submitted for any other degree or diploma.

Arthur Grilo da Silva Santos, 13.05.2021

Contents

List of Figures

List of Tables

List of Abbreviations

AM	additive manufacturing
CAD	Computer-aided Design
CAE	Computer-aided Engineering
CNC	computer numerical control
DFSM	dry-film photoimageable solder mask
EMI	electromagnetic interference
GUI	graphical user interface
HDI	High-Density Interconnect
LDI	Laser Direct Imaging
NC	numerical control
PCB	printed circuit board
PCBA	printed circuit board assembly
PWM	pulse width modulation
PWB	printed wiring board
RF	radio frequency
SM	subtractive manufacturing
SMT	surface-mount technology
THT	through-hole technology
UV	ultraviolet
UVL	Ultraviolet Lithography

1 Introduction

1.1 Motivation

Automate operations are a trend in any manufacturing process. This can be observed particularly clearly in the electronics sector, where smaller products are developed with high precision calls. This demand implies a product that is hard to be achieved even by the most skilled professionals, at large-scale productions but easily achieved by robots. Therefore, the industry invests in research and improvement of the right manufacturing tools for large-scale production.

The university’s workshop actually has the capability to print a circuit design on the board using the photoengraver masks technology. The photomasks, however, are stamped with a common office laser printer, hence, the design printed on the transparency film is not dark enough to block the ultraviolet (UV) light at the UV exposure unit. It leads to the necessity to print the circuit design, at least, twice.

Miss alignment occurs between the first and second printed design. It occurs because there is no need, for office printers, to print aligned multiple layers. This unprecise process may cause problems at the final circuit design printed on the board, resulting in a final product with poor quality. In addition to the miss alignment issue, the lacking of equipment to align the bottom and top photomasks, which is a necessary step for a double-sided design, may add further errors to the printed circuit, decreasing the quality of the printed circuit board (PCB).

Those issues can be overcome, for low precision PCBs only, with some hours of dedication and re-work until the desired design be achieved, which is at least frustrating. Therefore, it would be valuable for the university’s workshop the acquisition of an affordable tool to photoengrave the circuit design minimizing the before cited concerns.

1.2 Thesis Purpose

The aim of this thesis is to deliver to the university's workshop an alternative solution to the actual photoengraving equipment used for photoengraved circuit board prototype. The new hardware must have its parameters derived and be able to photoengrave a PCB and photoexpose a solder mask media to generate required PCB design.

The machine's user will be able to print the desired design inputted as a Gerber file without previous programming knowledge and without understanding how the machine is controlled. This new hardware will photoengrave the circuit design or the solder mask design precisely and autonomously using the Laser Direct Imaging technique.

1.3 Thesis Contribution

This thesis describes how to convert a 2 axis CNC machine into proper computer-aided laser lithography for PCB prototyping. The principal contributions of this work are:

- The development of an application to translate CAD files to laser photoengrave CAM files.
- The steps to configure and parametrize the main CNC machine settings using Grbl
- Understanding how the lithography parameters, such as laser power, exposure time, feed rate, etc. contributes to the quality of the PCB prototyping.

1.4 Thesis Overview

This work is organized as follows: In section 2, a general introduction of the PCB manufacturing for a better understanding for the reader about the process used in the experiments. Section 3 introduces the necessary theoretical background about the hardware components used in this work. The software tools, APIs, and machine language are covered in section 4. Section 5 defines and discusses the problems to be solved in the scope of this work. The hardware and software technology, hardware configuration and parametrization, and experiments are covered in section 6. Finally, the experiment's discussion, conclusion, and possible improvements are given out in section 7.

2 Printed Circuit Board Manufacturing

Printed circuit board (PCB), also named as printed wiring board (PWB), is an electrical circuit assembled at laminated panels, made of insulating material, in between layers of conductive material. This board supports and electrically link circuit components utilizing conductive features as copper traces and pads.

The number of layers with conductive material depends on the PBC complexity. Simple PCBs can have all tracks designed in one single layer. More complex PCBs, however, may need multiple layers design. Therefore, vias must be used to connect the pads in different layers as Figure 2.1 illustrates.

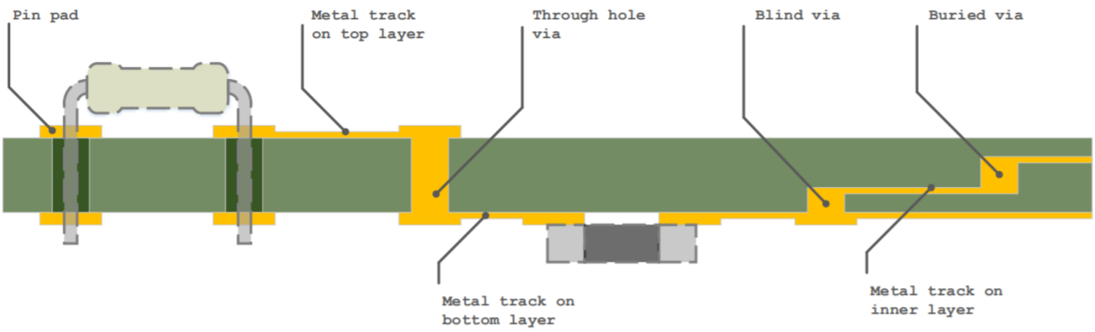


Figure 2.1 – PCB cross-section (Source: Automatic Generation of In-Circuit Tests on Prodrive's AET System for Board Assembly Defects (2017), page 3)

Design, copper patterning, chemical etching, and assembly are some of the manufacturing processes to make a PCB. Multi-layer PCBs are also laminated and drilled before being assembled and after being chemically etched. Those manufacturing processes will be discussed further in more detail in the next sections of this work.

2.1 Historical Background and Overview

The invention of a printed circuit was fomented by the technology improvement in World War II for military use. Before the invention of the PCBs, circuits were made by connecting the component leads point-to-point with soldered wires, crimps, and other

methods. This method was large, bulky, and fragile since the components were fixed on metal chassis, sometimes with a wooden bottom.
Researches inform that the first invented PCB was made in the United Kingdom in the middles 1930s by the Austrian engineer Paul Eisler. Further development and usability were done in the 1940s during World War II in the United States and Germany.

Even that the method of electroplate circuit patterns was patented in 1927 and used in commercial inventions in 1948, the old way to produce electronics kept in the market until the late 1960s when it started to lose market share for the reduced size, weight, and cost advantages of the PCBs [1].

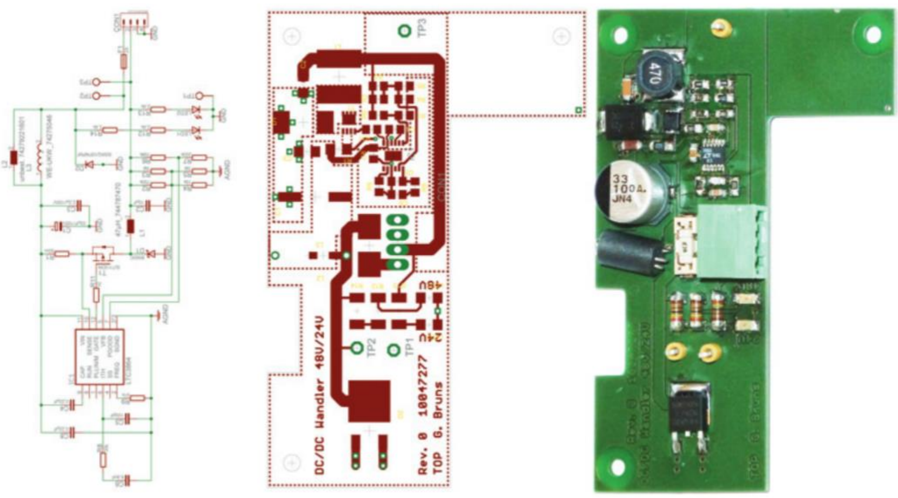
With the advance of technologies as High-Density Interconnect (HDI) PCBs, the boards became smaller and with a higher density of components. Those characteristics help the new invention to rule the electronics market and to be present in almost all electronic assemblies. The PCB's manufacturing technology has also improved and changed since its invention and its production method may vary according to the budget and the complexity of the circuit.

2.2 Computer-Aided Design

Computer-aided design (CAD) is a category of Computer-aided engineering (CAE) related to the technical layout of a design. The CAD software enables faster development and reduces costs by simulating and analyzing the real behavior of the product before purchasing material and manufacturing the hardware [2].

The first step for any printed circuit is to develop the circuit diagram. It describes the electronic functions that the final product should supply. The layout of the PCB is defined based on its circuit diagram and done at CAD software. An efficient design is defined by arranging the components according to their functions and connect them with conductive paths (routing).

The CAD layout may contain assembly issues and missing data or components. Therefore, the plausibility of the file should be checked before delivery. The data delivered for the PCB manufacturing is generated as the output of the internal CAD database provided at the layout design. It is exported into a data format that can be read by the manufacturing and assembly equipment, such as Excellon and Gerber formats. Those files can describe the path of contour milling, the positions where to drill, or the copper pattern in a lithography process. [3]



(a) Circuit diagram (b) CAD layout (c) Final product

Figure 2.2 – from circuit diagram to the CAD layout to the final product (Source: Leiterplatten-Prototyping (2015), page 20)

Multiple PCBs can be designed in a single production blank, making optimum use of the board area and accelerating the production, in most cases, when compared with the manufacturing of individual prototypes one after the other.

2.3 Subtractive, Additive and semi-additive processes

There are basically three manufacturing techniques to transform raw material into the final product. The most conventional technique is subtractive manufacturing (SM), which consists of machining materials, removing undesired matter. Additive manufacturing (AM) is a more recent technique that, in contrast to SM, joins materials to make the final product. The waste produced at AM is substantially reduced if compared with the waste of the SM. It is, however, a more complex technique and not appliable depending on the material, like ceramics [4]. Semi-additive manufacturing is the combination of AM and SM.

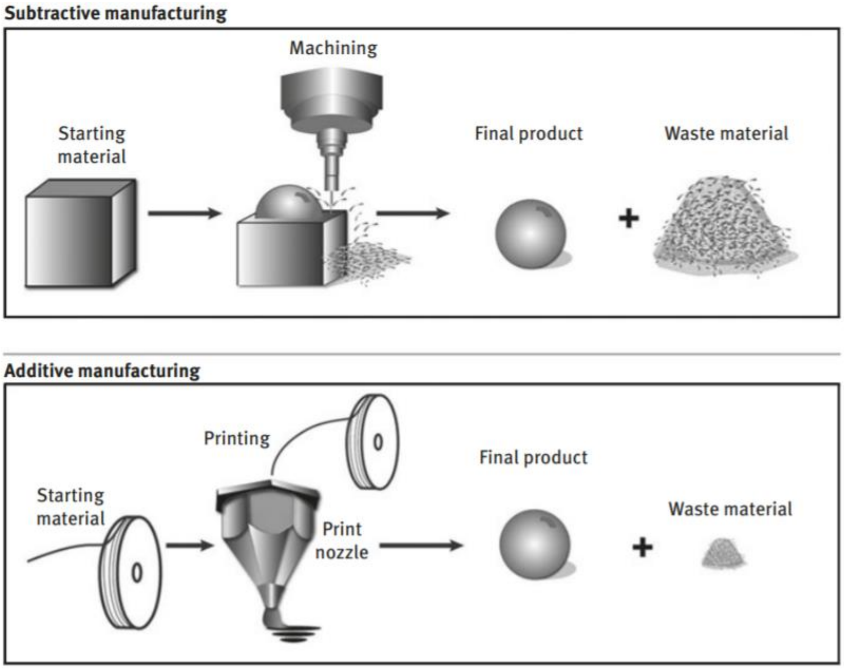


Figure 2.3 – Basic principles of AM and SM (source: GAO analysis)

Not differently from the other products manufacturing process, the layer of conductive material of the PCB can be applied on the insulant layer, also known as laminate, as an additive process, subtractive process, or a combination of both.

The additive process forms conductive patterns by adding conductive material over the laminate using electroless plating or electroplating. This process reduces significantly the chemicals used in PCB production. However, it is a recent and not tough method since it is not able to yield properly the conductive material to resist the handling of rework and normal assembly, which makes it a method not commonly used in the printed circuit industry.

The subtractive process forms conductive patterns by etching away the parts of the continuous conductive foil over the laminate that is not protected with a etch resist. The etcher removes (subtracts) the undesired conductive material, resulting in the wished conductive pattern. Today, the subtractive process is almost the only method used in the printed circuit industry with methods as, silk screen printing, milling, laser etching, and ultraviolet lithography (UVL) [5]. The last will be discussed further in more detail since it is part of the subtractive process used in this work.

2.4 Lithography

Lithography (from Ancient Greek λίθος, lithos 'stone', and γράφειν, graphein 'to write') [6] is the art and science of building a design in a surface and can be translated, quite precisely, as “to write in stones”. In the case of PCBs, the stone is the laminate covered with a layer of conductive material and the objective is to write the patterns on it with a light-sensitive polymer called photoresist.

Since the beginning of microelectronics manufacturing, lithography has been the most used method to produce the high numbers of integrated circuits of the mass-production industry. With lithography, the requirements of resolution can be achieved in the shortest time in comparison with other conductive pattern techniques.

To better define the lithography technique used to generate the patterns, additional identifiers can come before the name such as imprint lithography, with imprinted or pressed mold into the material, e-beam lithography, using electro beans, and optical lithography or UV lithography (UVL), that uses UV light [7].

2.4.1 Ultraviolet Lithography

UVL is a photographic procedure where a photoresist is exposed and developed generating patterns in the layer. This final pattern should be binary (part of the conductive layer is still covered with the photoresist while other parts are not covered). The part covered is kept after a pattern transfer mechanism, as chemical etching, while the rest is etched away. [8]

The general UVL process consists of first laminate the production blank with positive or negative photoresist. The now coated production blank is covered with a photomask, which selectively allows light through, and then exposed to the UV light. In the case of coating with a positive photoresist, the parts exposed to the UV (i.e. the gaps between the tracks) light are weakened and dissolved at the development step while the not exposed parts are kept. For coating with negative photoresist, the parts exposed to the UV light (i.e. the tracks) are hardened and kept and the not exposed ones are dissolved at the development step. After development, the board is then etched, where the parts that are not covered by the photoresist are removed. The last process, stripping, consists of removing the layer of photoresist that covers the not etched conductive layer.

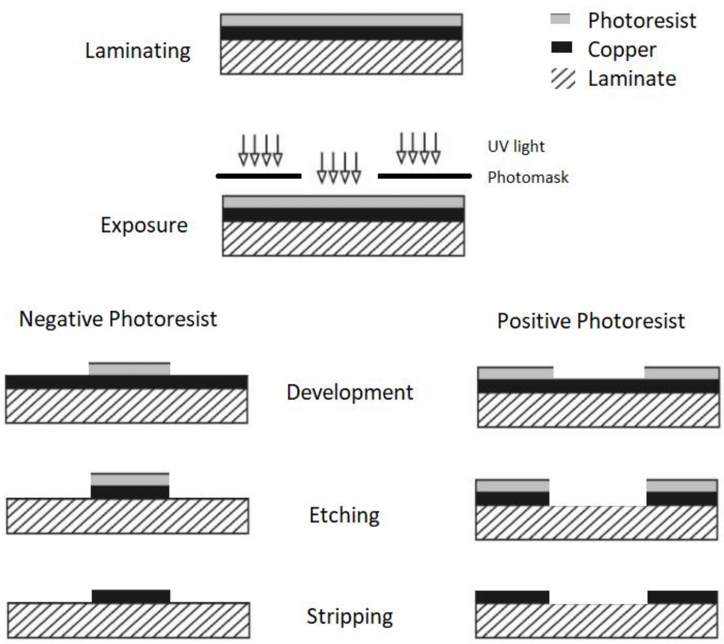


Figure 2.4 – Sequence of copper patterning with photomask lithography technology

2.4.2 Laser Direct Imaging

Laser Direct Imaging (LDI), also known as maskless lithography, is proving to the best imaging solution for HDI boards when compared to photomask lithography. At LDI, a computer-controlled laser beam is scanned across the PCB exposing, often with UV light, the photoresist just at the desired parts. The PCB is often exposed to UV light while using LDI technology.

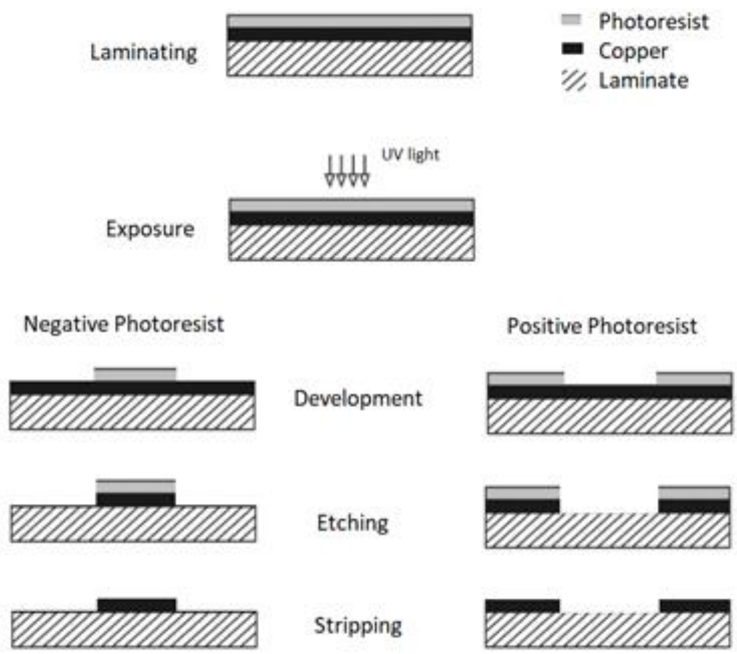


Figure 2.5 – Sequence of copper patterning with LDI technology

LDI is used as an alternative to the photomask lithography technology due to the following deep-rooted problems: [9]

- Change of material between panels that belongs to the same batch.
- Defects due to photomask operation and over-exposure due to lack of contact in between the PCB and the mask.
- Photomasks are not isotropic, which may lead to inaccuracies if the same scaling is used in both directions.
- Unexpected changes at the photomask dimension due to temperature and humidity.

2.5 Etching

The etching is one of the crucial steps in the chemical procedure to reach the subtracted PCB. At this procedure, the unwanted conductive material is removed generating the desired traces and patterns. To selective remove just de unwished part, the desired pattern is covered with an etching resist. Many different etching chemicals are industrially used at chemical etching, some examples are ferric chloride ammonium Persulphate and chromic/sulphuric acid.

The etching is not a complex process but it has many branches. It may vary not only according to the etching chemical used, but also according to the etching arrangements (i.e. simple batch production etching and continuous feeding etching), to the etching technique (i.e. bubble etching and spray etching, and to the etching method (i.e. mechanical etching, chemical etching, and electromechanical etching.

Since etching is one of the most important steps in PCB manufacturing, the defects that occurred in this process may have a high impact on the quality of the final PCB, especially for HDI circuit boards. One of the most common is undercut or under-etching. To avoid those kinds of defects, the immersion etching technique is preferable since it leads to a lower etch factor. [10]

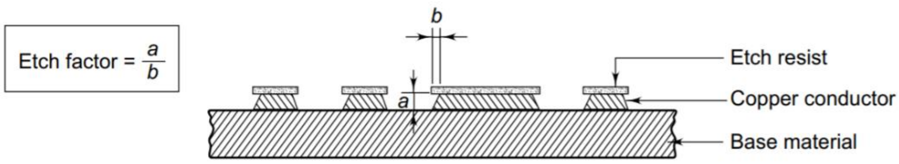


Figure 2.6 – Under-etching defect and etch factor (Source: Printed circuit boards: design, fabrication, assembly and testing (2005), page 381)

Some other common defects as non-etched material (spots where unwished conductive material was not removed possibly causing short circuits) and Pits (spots where wished conductive material was erroneously removed) may appear due to inaccuracy in the past subtractive or development process.

2.6 Lamination

A common characteristic of modern manufacturing PCB technology is the presence of small components and complex trace arrangements. To bypass it, a process named lamination stacks layers of conductive material and insulant material to produce what is known as multilayer PCBs.

Figure 2.7 – 4-layer PCB

Beyond the reduction of the traces design complexity, multilayer PCBs with embedded traces may increase immunity to radio frequency (RF) fields and reduce electromagnetic interference (EMI) emissions by a factor of 10 or more when compared to double-sided PCBs. This improvement comes from the possibility of a continuous conductive layer to be used as the ground while for double-sided circuit boards the ground plane is generally unordered with signal crossovers, etc. [11]

a) Not Embedded Traces b) Embedded Traces

Figure 2.8 – Embedded and not embedded stack scheme

2.7 Drilling

Drilling is a necessary step just for boards using the through-hole technology (THT) components, since surface-mount technology (SMT) components, as the name indicates, are mounted exclusively on the surface of the PCB.

The main purpose of drilling at PCB manufacturing is to create holes that connect its bottom, top and internal layers where the vias will be placed, and to mount through-hole components [12]. There are basically three types of holes. Blind holes, which is a hole drilled at one side of the board to a defined depth without breaking through it. Buried holes, which is a hole in the inner layers that cannot be seen from outside. And through holes, which is a hole that breakthrough one side until the other side of the board.

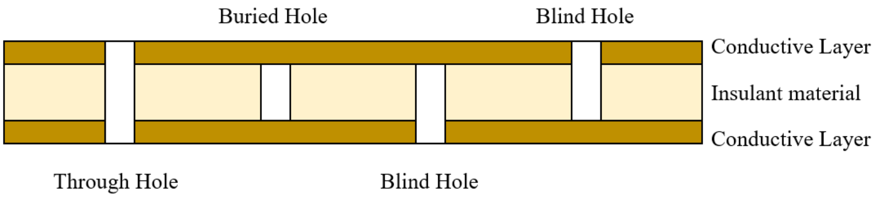


Figure 2.9 – Types of drilled holes

Besides the assembly proposal, drilling can also be a pre-process of PCB manufacturing. Before the conductive pattern printing, the production blank may be drilled to generate reference holes for the pattern printing tool. The holes drilled with the aim of fixing the PCB at the known location are called fiducials.

The fiducials are a common solution when it is made necessary to transfer the global coordinates, (machine's coordinates), to local coordinates (production blank's coordinates) during the manufacturing process. If used correctly, they can guarantee the precision of the printing, when it is made necessary to flip or move the PCB in-between steps, like for double-sided PCB manufacturing and solder mask application.

Generally, the drilling process is computer-controlled. The information with the XY coordinates of the holes and the specific drilling tool can be found in a drill file or an Excellon file.

2.8 Solder Mask Application

The solder mask or solder resist, is a thin layer (about 0.1mm thick) of epoxy, half-epoxy, or fully aqueous material. This layer covers the whole board surface but the areas where components have to be soldered during the assembly process [13]. As its name implies, his main function is to prevent the covered areas from solder shorts that may occur during the soldering process. Although, it is not the only function of the solder resist. The coating also protects the untinned conductive material from abrasive and chemical damage caused by plating and etchant. Likewise, it also ends up protecting the boards from fingerprints, dirt and insulating them from other closely placed ones.

Solder resists may be temporary (present just during the assembly and soldering process) or permanent (being an integral part of the final PCB) and applied by screen printing or photo printing method.

2.9 Testing and Inspection

The last step of PCB manufacturing is the testing and inspection process. At this operation, the printed circuit must be submitted to a final control to verify the reliability and quality.

The most common and influent PCB defects are associated with electrical defects, such as open tracks, bridges in between the traces that should not exist causing a short circuit, RF impedance, and voltage breakdown. Assembled PCBs may also manifest assembly defects such as the installation of the wrong component, missing component, and component with bad contact or mounted backward. [14]

However, electrical and assembly defects are not the only ones to be tested. There is a long list of defects such as mechanical, chemical, long and short-term reliability, optical, etc. The combination of tests to be done must be defined according to how critical it is to avoid such defects.

2.9.1 Bare-board Testing

The test and inspection of the printed circuit can be done after or before assemble the components. Nowadays, however, PCBs are becoming more dense and tiny, which becomes hard to monitor all traces and pads, especially for SMT boards.

The huge majority of PCB defects can be detected before the assembly step. Perform a test and inspection before assembly may seem expensive and twice the work. However, do it upfront may avoid waste of components and maintenance costs due to bare-board defects, saving time, effort, and money that would be spent on a defected bare-board. [15]

2.10 Assembly

It is common to confuse and wrongly use the PCB. This term should be used to refer to a printed circuit board before the assembly process only. Therefore, PCB fabrication is related just to manufacturing steps before PCB assembly (bare board). The combination of bare PCB and mounted components is technically known as printed circuit board assembly (PCBA). This assembly process is vast and may vary according to the technique used (manually or automatized process), and the technology to be assembled (SMT components, THT components, or a combination of both components technology).

3 Hardware

The hardware developed in this thesis is a planar robot. A planar robot, also known as linear robots, or cartesian coordinates robot, has three linear axes of control (X, Y, Z) where each axis makes a right angle with the remaining axes. Due to the high accuracy and precision when operating in rectangular or planar workspaces, and effectiveness for plane travel and stacking, the principal applications of this sort of robot are computer numeric control (CNC) machines, and pick and place machines. [16]

The hardware, however, has a not movable Z-axis, instead of a linear one, since the lithography process to print the patterns in the photoresist is executed in a planar workspace. The robot has a low-power UV laser at the fixed Z-axis, two stepper motors moving the laser module in the Y-axis, and one stepper motor moving the laser module in the X-axis. The linear movement of each axis is done with a timing belt and a toothed pulley, which the last one is fixed to the stepper motor’s shaft.

A Raspberry Pi is used as an embedded system to stream the necessary commands to print the expected circuit design in the production blank to an Arduino nano. This Arduino controls the laser and stepper motors through a controller board containing the necessary electronics components to adjust the laser intensity and motor drivers to send the signals to drive the stepper motors.

Due to the risks to the eyes and skin of the UV-light exposure, the final machine design must be encapsulated in a case containing an external control panel. The intention of this case is to allow the user to control the machine and to observe the process without harming risk. A safety switch is located at the opening mechanism of the machine assuring that no current is being delivered to the machine while open, preventing the possibility of the wrong manipulation of the hardware from an unenlightened user.

The aim of this chapter is to explain and contextualize the reader with the components in the lithography hardware developed in this work.

3.1 What is a Laser

The term laser has been used so often in the late days that is common that we associate it more to a substantive than to its acronym "light amplification by stimulated emission of radiation". Its denomination was done based on the maser acronym "microwave amplification by stimulated emission of radiation", its previously researched "parent". [17]

The first laser was built in 1960 by Theodore H. Maiman and designated as "optical maser" or "infrared maser" [18]. However, its system was first described in Albert Einstein's work "On the Quantum Theory of Radiation " [19] and further researched by other scientists as Rudolf W. Ladenburg, which proved the negative absorption and stimulated emission phenomena in 1928 [20] and Alfred Kastler that proposed the optical pumping method in 1950. [21]

Currently, a large number of different laser types can be found in the market and industry. The different kinds of laser differentiate themselves from each other by various parameters as the laser-active medium, power, beam quality, and wavelength.

3.1.1 How Lasers Work

According to Albert Einstein's research, there are two types of interactions between electromagnetic radiation and matter with two stable energy levels (low-energy state 1 and high-energy state 2), which are absorption and emission of photons. The emission interactions can be either spontaneous as stimulated, while in the absorption interaction it is exclusively stimulated. [22]

Emission is characterized by the decay of the energy state from an atom, initially at high-energy state 2, to a low-energy state 1. The energy gap E_1 - E_2, is then released by the atom in form of a photon of frequency. The case that the phenomenon occurs without external radiation with a suitable frequency is called spontaneous emission. However, if it occurs due to external radiation, it is known as stimulated emission. Absorption, nevertheless, is characterized by the rise of the energy state, initially at low-energy state

1, to high-energy state 2 by external radiation with suitable frequency forcing an electronic transition takes place. [23]

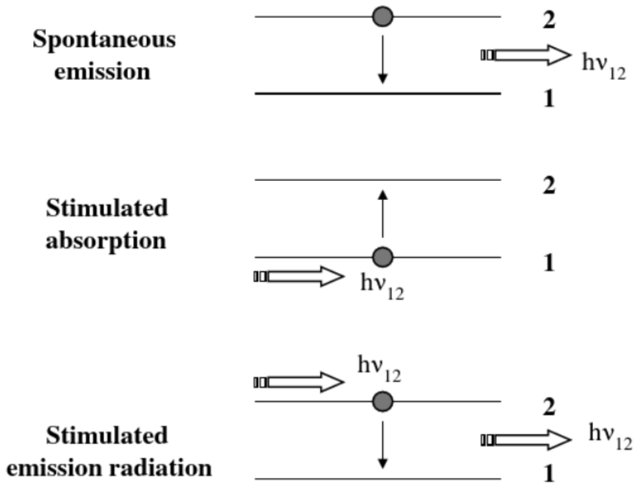


Figure 3.1 – Energy state transition at absorption and emission cases, where hv_{12} is the energy gap between the electronic energy states 1 and 2. (Source: Atomic Absorption Spectrometry: An Introduction (2014), page 3)

Initially, the laser-active medium is in thermodynamic equilibrium. This active medium is then excited increasing the energy level of the electrons located there. These energized electrons have then a certain probability to be transferred under the emission of a photon and once that one atom spontaneously emits a photon, it will stimulate the other ones to do the same. The torrent of illumination caused by the photons' emission is then bounced back through atoms between mirrors, stimulating other atoms to emit even more light. A blinding stream of coherent light is then left out through a small hole generating a laser beam.

3.1.2 Structure of a Laser

In general, there are five laser components: the pump source, the laser-active medium, the resonator, the cooler, and an optical element (such as fibers, mirrors, or lenses).

A pump source generates an electrical field transmitting energy to the laser-active medium and exciting it. This laser-active medium may be a liquid, a gas, or a solid that, by stimulated emission, has the light wave's amplitude amplified. The laser-active medium is placed in between a resonator (structure containing one mirror with partial reflection and another mirror with high reflection), which assures that the light oscillating among the mirrors passes constantly through the active medium. As part of this energy transmission process, heat is released since only a small proportion is converted into light radiation, which is dissipated through a cooler. Last but not least, the emitted radiation passes through an element that adjusts the radiation beam focus. [24]

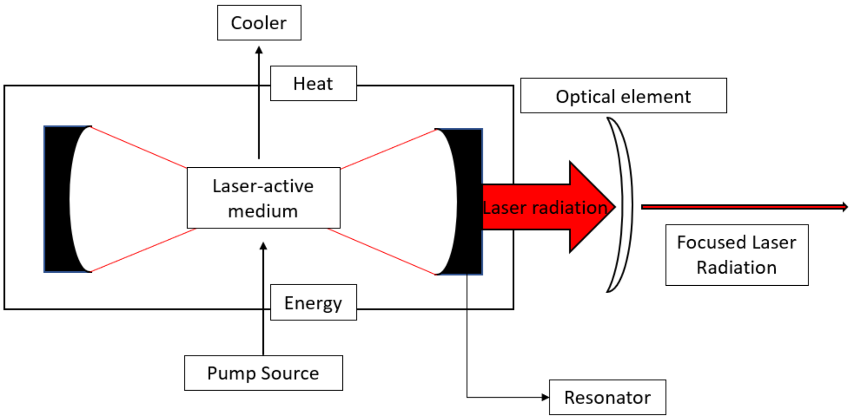


Figure 3.2 – Laser components

3.1.3 Properties of Laser Beams

Directionality, monochromaticity, brightness, and coherence are the four properties that distinguish the laser radiation: [25]

- Since **directionality** is inverse proportional to divergence, they can be treated as one laser radiation property. The directionality describes the angle of divergency made by the light beam, which in the case of laser beams, occur in the resonant cavity. Due to the characteristic that just waves that propagate onwards the optical axis can persist in this cavity, the laser beams are highly directional, which implies the tiny divergence of the laser light.

- The **monochromaticity** describes how pure is the wavelength of the laser light. With laser beams, oscillations happen just in the resonance frequency of the laser cavity due to its resonant system, which further narrows the laser linewidth (frequency range). Another factor that influences the purity of the laser light's wavelength is that only electromagnetic waves with suitable frequencies can be amplified, resulting in a very small linewidth when compared with ordinary lights.

- The **brightness** of a light source is determined by the emitted power per unit surface per unit of solid angle. Since the spatial coherence of the beam has a direct proportionality with the laser brightness, the max brightness is achieved at a perfect spatial coherent beam.

- An electromagnetic wave can be namely spatial and temporally **coherent**. The two sorts of coherence are independent, which means that a wave can be both, namely spatial coherent and temporally coherent, at the same time.

  An electromagnetic wave has perfect coherence if the measured phase difference, at an initial time $t_0 = 0$, of two points in this wave remains the same for any time $t > 0$. The wave has perfect spatial coherence if it occurs for any two points along with itself. In reality, waves are partial spatial coherent since the spatial coherence occurs just in a limited region.

  The electromagnetic wave is temporal coherent if the phase difference between time t and time t + delay period dt remains the same. The temporal coherence is perfect if the delay period dt can be any delay period. If the delay time is bounded ($0 < dt < t_o$), the wave has partial temporal coherence.

3.1.4 Diode Lasers

Diode Lasers, also known as semiconductor lasers, can create lasing conditions at the diode's boundary by pumping directly a diode with an electrical current. Due to this characteristic, diode lasers have spread widely in comparison with the most important classes of lasers, as gas lasers, dye lasers, solid-state, and fiber laser since their capacity to be pumped straight by an electrical current generally leads to more efficient performance.

Traditionally, plasma excitation lasers, such as gas lasers, or incoherent optical flashlamp source lasers, such as solid-state ones, have an efficiency in the order of 1%. Some gas lasers, such as CO2 gas ones, can typically achieve over 10% efficiency. Diode lasers, however, commonly achieve an overall power conversion efficiency of around 50%. Nevertheless, the coherence of gas, dye, fiber, solid-state lasers is better than simple diode lasers due to cavity structures. [26]

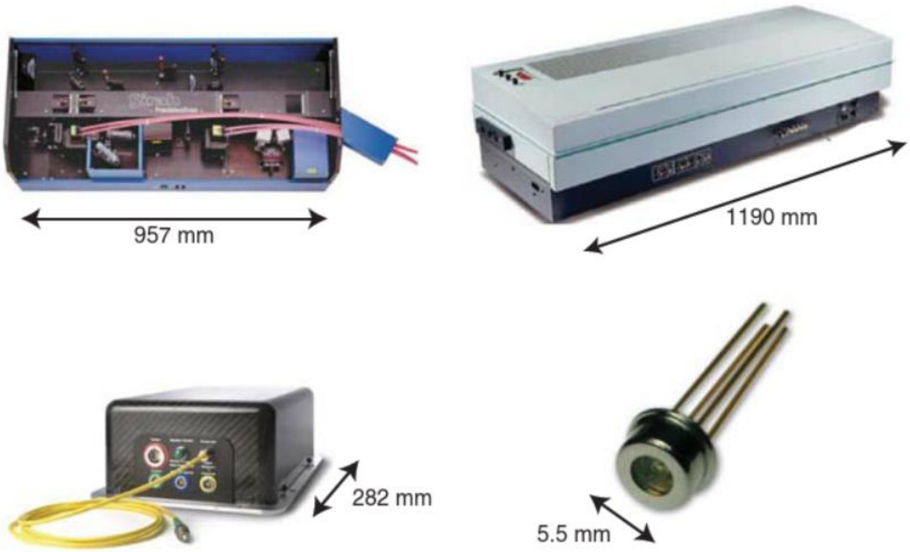


Figure 3.3 – Example of solid-state (upper left), dye (upper right), and fiber laser (bottom left) systems compared to a packed diode laser chip (bottom right). (Source: Diode Lasers and Photonic Integrated Circuits (2012), page 3)

Net size and high reliability are other advantages of the diode laser in comparison to the other lasers. Diode laser chips are as small as a grain of salt and have a volume of around a cubic centimeter for the complete hardware. Gas, solid-state, and fiber lasers, however, are typically some decimeters in length and have a useful lifetime of some thousand hours while carefully qualified diode ones last for some hundred years.

3.2 CNC Machines

The invention of manufacturing machines was a milestone in the industry's automatization. Since the first machines that substituted the hand tools, which made available a more accurate and standardized production, going through the first steam and electrical machines, which enlarged the lots of product produced at the day, until the late years with machines that work at extremely low units of resolution keeping a high precision, which substituted numerous hand tools processes to automatized ones, the manufacturing industry has increased its production capacity with a huge variety of products and technologies.

The CNC technology started to appear after World War II, where equipment was highly needed to effectively reduce the critical demand for skilled workers in the production line. Therefore, several scientific and technical developments were done including the most notable computer-related inventions, biding the computer development, as a pilar, to the improvement of several industrial technologies and digital control. [27]

Figure 3.4 - 3-axis CNC routers for hobbyist (Source: CNC-Fräsen für Maker und Modellbauer (2020), page 4)

A digital control machine is a connection between the computer and the machine through a device (interface) that is able to translate the information provided by the computer to the machine and vice-versa. Those machines were controlled by number, symbol, and letter commands, therefore it has been named computer numerical control (CNC) machines. There is a big variety of CNC machines available today in the industry as the famous 3D printers, milling machines, plasma and water jet cutters, CNC routers, etc.

3.2.1 Benefits of CNC machines

A rule of thumb says that the faster and more accurate the machinery of the industry, the more successful it is, both characteristics are highly present in CNC machines.

The capacity to deliver a faster job due to the need to equip the machine just at the first operation, the facility to move between different product forms and the automatized process allowing the worker to do parallel jobs at the same time that the machine is manufacturing the product are some of the various benefits from a CNC machine.

Furthermore, the use of the same program allows for the high-precision for all produced pieces and the ability to repeat the process to a large number of standardized products. In addition to all those benefits, CNC machines can produce highly complex products, generally too difficult to produce in traditional machines, due to their ability to control the appropriate cutting conditions. [28]

3.2.2 Stepper Motor

There are two types of motors typically used in CNC machinery: stepper and servo motors. The motors in these kinds of machines are used to move the end effector upwards and downwards or to drive the guide systems.

Figure 3.5 – Stepper Motor

The motors driving the guide systems of the machine are responsible to move a considerable portion of the whole structure. It may happen at higher speeds, for the case that the tool is simply moving from one point to another and not cutting, and at lower speeds, when moving from one point to another while cutting material.

One of the most important characteristics of the stepper motor is the high torque while at lower revolutions per minute, which makes it ideal for the application of driving the guide system. In general stepper motors are simple, cheap, very accurate, and do not require tunning, which makes it even more interesting to prototyping or hobby CNC machines.

3.2.3 Timing Belt and Pulleys as Transmission System

The function of the transmission system of a CNC machine is to transform the output work of the motor into linear motion. There are many methods to transform motor power into movement such as chain, cable, belt, rack and pinion, pneumatic, hydraulic, and screws. However, for the CNC application, the application must deliver resolution and accuracy and move bidirectionally. Therefore, CNC transmission systems have a tendency to be either rack and pinion or some type of screw and nut mechanism, sometimes both.

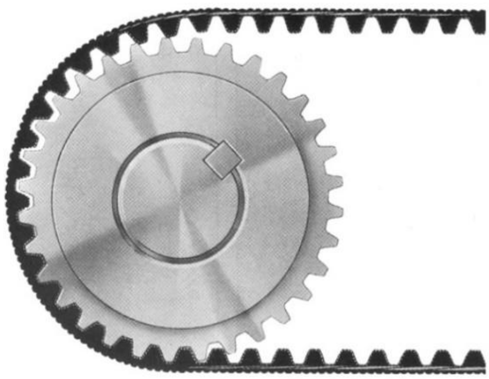


Figure 3.6 - Synchronous belts rely on tooth engagement between the belt and pulley to transmit power. (Source: Gates Corporation)

The timing belt and pulley are in general simple to build, operates very silent, are the least expensive, relevant characteristics for prototyping or CNC hobby machines.

3.2.4 Encoders, Drives and Spindle-Speed Controllers

The combination of motor and transmission systems cannot deliver, by itself, the desired resolution and accuracy for a CNC machine. Therefore, elements as encoders, drives, and spindle-speed controller assist the controlling the machine for a more precise operation.

Encoders are responsible to provide feedback about the actual position, either optically, magnetically, or electronically, in almost real-time. With this information, the device is able to identify if the actual position agrees with the calculated position that the G-code command intended.

To achieve a higher resolution, the stepper motor can make use of a drive device to amplify the signaling frequency rate increasing the stepper-revolution from 200 to 2000 making it possible to work with a resolution of fractions of millimeters.

Differently from the encoder and drive, a spindle-speed controller does not control the basic motion, but a frequency-driven spindle, a relevant variable for rotary cutting or engraving. It transfers the control of the variable from the drive's panel to the software used to control the machine. Some commands as rotation speed, on/off, and forward/backward rotation can then be controlled with G-code or manually at the software panel.

3.3 Arduino

Before the dramatic change that was the introduction of the Arduino platform in 2005, many electronic enthusiasts were not able to perceive their passion due to an expensive and complex hobby that was to design prototypes with microelectronics. Arduino broke the barriers to prototype robots and microelectronics with its accessible price, no necessity of extensive previous knowledge, and learning by doing method leading to the rapid growth of its community. The Arduino platform consists of an integrated development environment (IDE) for its own board and a large selection of hardware. [29]

Figure 3.7 – Arduino hardware “Arduino Uno Rev3” (at the left) and Arduino IDE (at the right) (Source: Arduino’s official website (2021))

Arduino can be used in the development process of interactive projects. It can read signals from a wide range of switches or sensors and can send signals to several electronic components such as motors, lights, displays, etc. Furthermore, Arduino projects can be fully embedded and autonomous or work exclusively like a microcontroller while it is communicating and switching information with software running on another system.

3.3.1 Why Arduino?

Netmedia’s BX-24, MIT’s Handyboard, Parallax Basic Stamp are some of the platforms and microcontrollers available for embedded electronics as an alternative for Arduino. However, one of the clear Arduino advantages is its price (the Arduino’s flagship, Arduino Uno Rev3, is available for €20.00 [30]). Along with it, its software is multi-platform running on Windows, Macintosh, and Linux, while most of the microcontroller systems are limited to windows. In addition, Arduino has a simple and clear programming environment making it easy for beginners but flexible enough for advanced users, has an open-source license, and has extensible software and hardware.

The platform is completely open-source which allows the community to develop and adapt the board for specific applications and to clone its architecture. The open-source license allows the manufacturing of Arduino clones, which are generally cheaper and of lower but enough quality to run prototypes. Moreover, the development software, as many libraries and complete projects code developed by the community, can be downloaded for free.

The size of the Arduino community supports the new enthusiast with lots of answered questions, projects, and solved logic problems available at the forums or with experienced colleagues ready to help to solve your specific problem.

3.3.2 Choosing the Proper Arduino Board

Arduino was released as a unique board option, Arduino Duemilanove (equivalent to the Arduino Uno nowadays), so there was no necessity to select which board, from the more than 15 core hardware options now available, best fits the project. The Arduino hardware may vary in size and capabilities. Some are more powerful but also much larger than the standard model (Arduino Uno Rev3) while some are tiny and inexpensive but less powerful and with limited memory.

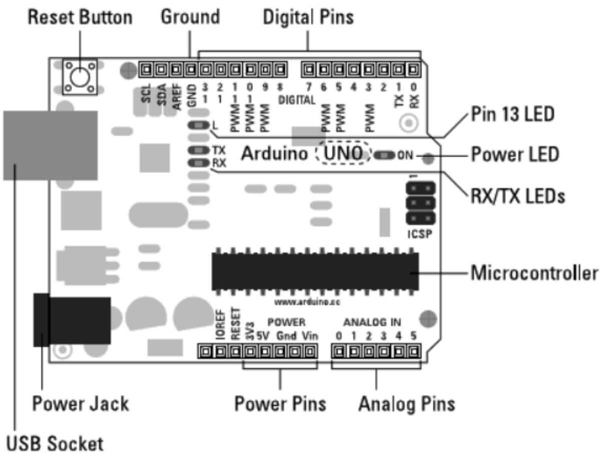


Figure 3.8 – Arduino Uno Rev3 and its relevant parts labeled (Source: Arduino for Dummies (2013), page 21)

The models are generally described based on the standard model Arduino Uno Rev3, which is a microcontroller board based on the ATmega328P running at 16 MHz. It has 6 analog inputs, 14 digital I/O, a USB connection, a power jack, a reset button, and comes with all components needed to support the microcontroller, making it necessary just to connect it to a computer with a USB cable or power the board with an AC/DC adapter or battery to start prototyping. [31]

The Arduino Nano (board model used in this thesis) is tiny, complete, and based on the ATmega328 (Arduino Nano 3.x). It has more or less the same functionality as the Arduino Uno Rev3 but in a different bundle arrangement and size. It has a Mini-B USB socket instead of a standard one and it does not come with an AC/DC power jack. [32]

Figure 3.9 – Arduino Nano (Source: Arduino's Official Store (2021))

3.4 Raspberry Pi

In 2006, at the Computer Laboratory of the University of Cambridge, a group of computer scientists led by Eben Upton wanted to produce an inexpensive educational micro-computer for fresh computer students and enthusiasts. The idea of the project was to make the home computers affordable to help the users of the platform to develop computer science skills before or without access to a graduation course. Two years after looking for a way to make a tiny and cheap computer viable, Eben and his team were able to start developing their low-cost, credit-card size computer. In 2012, it finally became available to the community. [33]

3.4.1 The Board

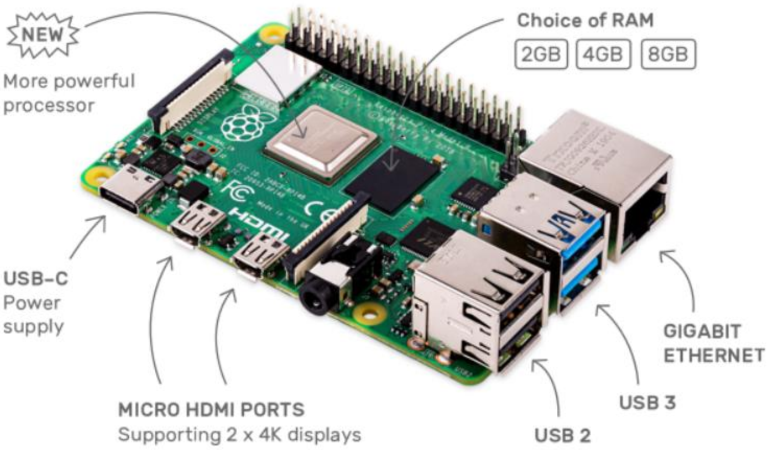


Figure 3.10 – Raspberry Pi 4 Model B (Source: Raspberry Pi's official website (2021))

The Raspberry Pi is a cheap and tiny system, which consists of a board of about 90 x 60 mm with all necessary components to use as a computer, just by inserting an SD card as storage and connecting it to a 5V power supply. Depending on the intention of use, the single-board computer can work as an embedded system, if previously programmed accordingly.

It is compatible with many computer input accessories as monitors, keyboards, and mouses. The operating system is a special Linux distribution, which, in contrast to other systems, like Windows, fits the inherent single-board computer limited hardware resources in terms of processor performance, memory size, and graphics system. [34]

3.4.2 What Raspberry Pi can do?

The Raspberry Pi is nothing less than a very tiny general-purpose computer, therefore anything that can be done on a common computer can be done with a Raspberry Pi as well. The access to the GPIO, that many machines do not have, gives the opportunity to build a dedicated device such as deploy a robot, controlled by software that you wrote.

Anyone who wants to convert an engineering idea into a physical interactive electronics project, prototype, or work of art should consider the Raspberry Pi as an option. Its platform also enables further development and customization, to the users with more advanced computing or electronics understanding, for projects with specific needs. Besides its similarity with a common computer, which allows internet-connected work coded with your preferred language, the Raspberry Pi can perfectly integrate high-level software and low-level electronics as vending machines, robots, and Internet of Things applications. [35]

4 Software

The present thesis looks forward to assisting the users of the university's laboratory by automating the lithography process. Therefore, if possible, it is crucial to not drastically change the steps followed in the existing process, to not add further steps, and reduce some of them by executing it potentially effortless. Nevertheless, it is a challenge to keep the actual process untouched, since direct laser lithography and photomask lithography have different inputs and manufacturing processes.

The software should not just be user-friendly and able to control the machine to deliver the desired printed circuit, but also script all necessary steps in between the computer-aided design and the computer-aided lithography. Therefore, it means that the software input should be kept the same as the existing lithography process (a Gerber file containing the circuit design) and the same output (a production blank exposed by UV light according to the requested design, ready to be developed). In addition, the software should also be able to guide the machine to expose the solder mask media to generate the desired solder mask print.

For this purpose, the software, written in Python, needs to transform the CAM information at the Gerber files to manufacturing information that suits the direct image operation requirements, and translate this information to G-code, the language that the in-built Arduino's program (Grbl) can decipher to control the hardware.

The aim of this chapter is to explain and contextualize the reader with the terms and auxiliary software used for direct imaging software development.

4.1 Python

Python is a contemporary object-oriented programming language. Its handling is supported by an interpreter, in which a program can be piecewise developed and independent instructions can be tried out. Furthermore, its syntax is clear and quick to learn, and its kernel is small.

The fact that Python has a small kernel does not necessarily mean that the language cannot be universally used. Its capability can be easily extended through numerous available modules allowing Python to handle a wide variety of development branches such from web programming and XML processing to complex mathematical calculations. Another Python characteristic is that it was developed to be used in a way that suits best the user, either for classic procedural programming or for elements of functional. [36]

Python is mature, flexible, and seen as a multiparadigm language. For instance, Java developers will not find difficulties programming in Python, since it supports oriented-object paradigm, or C developers could use Cython, a mix of C and Python. There is no need to compile the code to the machine language since Python is an interpreted language, which leads to a code that is executed immediately. Furthermore, Python typing and memory usage are dynamic, and the non-mandatory delimiters are suppressed, which makes its code easy to read. [37]

4.2 Gerber File Format

Gerber format is a PCB image format invented decades ago. In the 1960s and the 1970s, images were printed on lithography film by a precision optical numerical control (NC) machine, also known as vector photoplotter, that selected the appropriate aperture rotating a wheel containing etched holes.

The data containing the commands for the exposure process was contained in a Standard Gerber file driving the plotter and controlling the correct aperture of the wheel to plot the desired image. Since it competently suited this kind of task Standard Gerber became an industry standard. [38]

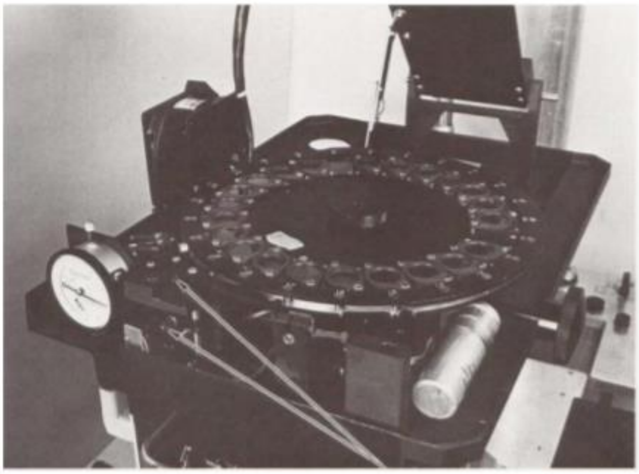


Figure 4.1 – Vector photoplotter (Source: Ucamco: Open Letter to the Gerber User Community (2014), page 2)

When not specified, this work refers to Gerber file format as the current one known as Extended Gerber or RS-274X, an image format. This thesis uses the current Gerber file format rather than the Standard Gerber (RS-274-D), an NC format, since the Extended Gerber supports coordinate format and aperture shapes with the codes G36 and G37, and it has the wheel file embedded to its file.

Wheel files are responsible for the defined data coordinate and aperture definitions, which leads to defined interpolations and meaning of flashes. Therefore, the Standard Gerber file, supported by non-standardized information contained in the wheel file, turns harder the operator's work due to the huge potential for error and misunderstanding brought with the not standard and not embedded wheel file.

Since less than 1% of the Gerber files contain an error that may lead to scrap, over 90% of PCBs, either simple or complex circuit designs, are manufactured based on Gerber file formats, turning Gerber almost the universal choice in the PCB manufacturing industry. [39]

4.2.1 The Gerber Parsing Expression Grammar

Gerber is actually the simplest and most secure format for exchanging data between CAD and CAM systems. The geometrical description of points and vectors is done in a twodimensional Cartesian coordinate system, where it uniquely defines a point P in the plane by specifying its X and Y coordinates followed by its control function (D-code).

Only the “D1” (light on), the “D2” (light off), and the “D3” (flash) commands are defined D-codes. The remaining commands, larger than D10, are reserved for the supplementary information about the mechanical tool. This information describes if the position is to become a specific form, such as holes and pads. The lines bounded by the “%” symbol are lines containing the wheel file information and the symbol “*” indicates the end of the command line.

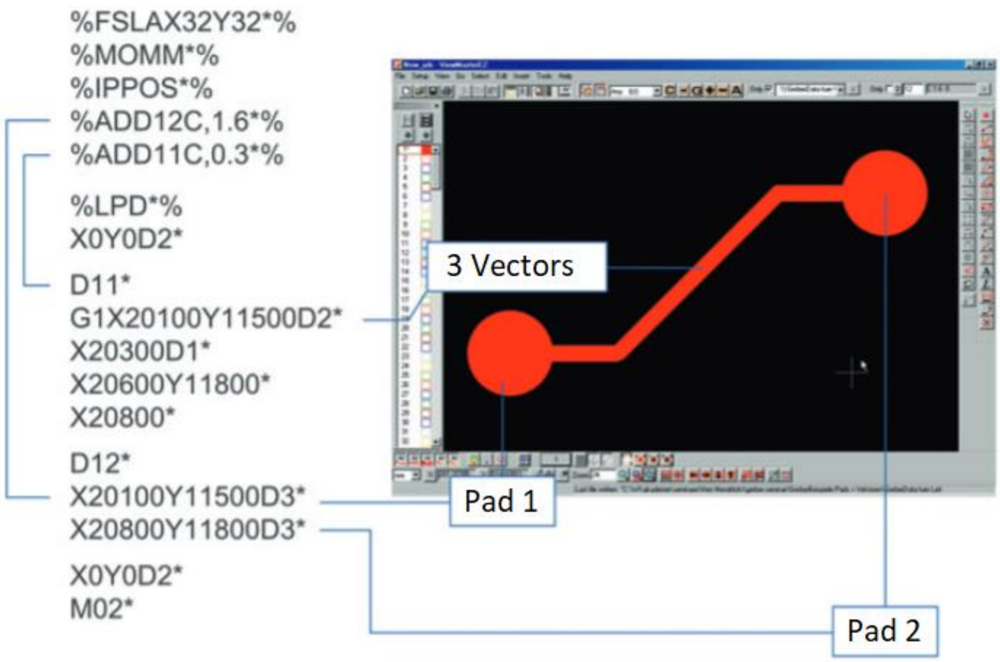


Figure 4.2 - Description of pads and vectors in Extended Gerber format (Source: Leiterplatten-Prototyping (2015), page 143)

The Gerber format is set up on an uncomplicated mathematical model. The rules of which the Gerber format must follow should also be obeyed by any other alternative format that wants to describe a PCB design as an image format. The syntax of the Gerber format is minimal, necessary, and sufficient since it is not possible to describe the element with less information. [40]

4.3 NC programming (G-code)

G-code programming language, also known as RS-274 or simply G-code, is a tool-based manufacturing programming language and the most common and widely used for NC programming. It tells CNC machines, such as 3D printers, CNC mills, and laser engravers, how to move the end effector at the workspace and when to turn it on and off.

Furthermore, G-code is a modal language, which means that once that a command is given it remains active until receiving a new command. The file contains the machine's destination location in terms of the new stop axes position, moving the end effector of the machine from the current position to this destination position. [41]

4.3.1 NC Words

NC words are control system instructions, written as a combination of alphanumeric characters arranged in rows. Those words are interpreted as commands such as to move the machine's end effector through its X-Y-Z axis, to adjust the feed rates, and many other functions: [42]

- **G-words** are preparatory functions represented by the letter "G" followed by two digits such as G01, G20, etc.
- **X, Y, and Z-words** generally succeed G-words and own the X, Y, and Z final positions of the machine tool. For circular interpolation, the I, J, and K-words are used in addition to the X, Y, and Z-words containing the arc center position.
- **F-words** contain the feed rate of the machine operation in millimeters per minute (mm/min), in case of G94 configuration, or millimeters per revolution (mm/rev), in case of G85 configuration. For example, F300 would represent a feed rate of 300 mm/min or mm/rev according to the appropriate G code.
- **S-words** specify the spindle speed in meters per minute (m/min) or in revolutions per minute (rpm) depending on the selected G code configuration. As for F-words, S1000 would represent the spindle speed of the machine at 1000 rpm or m/min. For laser machines, however, the S-word represents the percentage of power used

by the laser. In the case of a laser machine with max spindle speed set as 8000 rpm, the S1000 command would use 1/8, or 12.5%, of the laser max power.

- **T-words** are applied just for CNC machines with an automatic tool changer system. The machine's tools are listed and identified by NC words from T00 to T99.
- **M-words** are miscellaneous functions representing a collection of non-dimensional movements such as, turn a tool ON/OFF, START/STOP a tool, etc. The operator must pay higher attention to M-words since not all M codes' numbers are standardized by ISO specifications as occurs for G-words.

```
G0 X0 Y0 Z300   ; select rapid motion, move spindle to a safe position 30 cm over
                ; the middle of our material
T1001 M6        ; pre-select tool 1001 and perform the automated tool change into the spindle
X-50 Y-100 Z5   ; move to (-50,-100,5), i.e. 5 mm over the first corner
S2000 M3 F200   ; set spindle speed to 2000 rpm and activate spindle with clockwise rotation
                ; set feed rate to 200 mm/min
G1 Z-10         ; select linear interpolation, move down into the material to 10 mm depth
X50             ; move to ( 50,   0,-10)
Y100            ; move to ( 50, 100,-10)
X-50            ; move to (-50, 100,-10)
Y-100           ; move to (-50,-100,-10)
Z5              ; move to (-50,-100,  5), i.e. retract tool to 5 mm over the first corner
M5              ; turn the spindle off
G0 X0 Y0 Z300   ; move rapidly back to the safe
T0 M6           ; remove the tool from the spindle
M30             ; end of program
```

Figure 4.3 – G-code example and its commands description (Source: First steps in G-code at Benjamin Jurke's website)

4.3.2 Coordinate System

The coordinate system at G-code can be either an absolute system or an incremental system. In the case of an **absolute system**, all coordinates refer to the origin distance. For example, for a system with the origin of the coordinate system set at zero, receiving a first command P1 (X10 Y5) followed by a second command P2 (X15 Y15) would have as end effector final position (15, 15). However, for an **incremental system** receiving the same command sequence, the final position of the end effector would be (25, 20).

4.3.3 Reference Points

The machine can save some coordinate positions in the memory of the NC program as reference points:

- The **machine zero**, also known as home position, indicates the starting point and the origin of the coordinate system.
- The **zero shift** option enables to shift the machine zero position to any other point inside the CNC machine's working area. It allows coinciding the zero of the machine with the origin of the workpiece.

4.4 FlatCAM

FlatCAM is a 2D Computer-Aided PCB manufacturing software. It prepares CNC jobs to make PCBs on a CNC router, as the one developed in this work. Furthermore, FlatCAM has the necessary features for the most common PCB design operations, such as isolation routing, drilling, copper area clear, and board cutout for single-sided and double-sided boards.

The software support Gerber, Excellon, G-code, and SVG files as input. The input files will be accordingly treated as one of four different types of objects: Gerber, Excellon, Geometry (for SVG and intermediate step), or CNC job. The input and output dynamic from FlatCAM can be better visualized in Figure 4.4. [43]

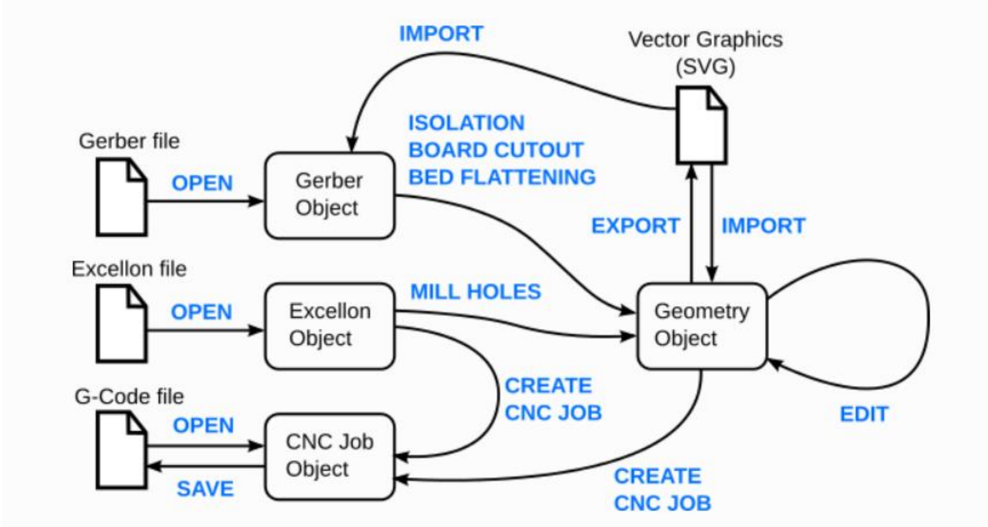


Figure 4.4 – Object and tasks that can be performed in FlatCAM (Source: FlatCAM's official documentation website)

FlatCAM provides a shell command-line interface that uses TCL language. This Shell interface provides access to FlatCAM's functionalities and makes available the possibility of automating large and repetitive sequences of commands with a script, which is an important tool for the design processing step used in the software developed in this thesis.

4.5 Grbl

Directly speaking, Grbl is an open-source, high-performance, and low-cost CNC machine alternative that simply turns Arduinos with a 328p processor into a three-axis parallel-port-based motion CNC controller. It reads the G-code command and converts it into pulses which control stepper motors connected to the Arduino.

Grbl interprets standard G-code commands such as circles, arcs, helical motion, and all other primary commands, but it does not read variables, macro functions, and most canned cycles. The controller is written in optimized C supporting asynchronous operations and timing precision keeping up to 30 kHz of solid, jerk-free control pulses, and making available smooth acceleration delivery due to the management of its 16 future motions' velocities. [44]

5 Problem Definition

The tool developed in this work photoengraves the PCB based on the design described by a Gerber file, which means that any steps in between the image format file to the NC format file, that may be done automatically, should be done so. Furthermore, the software should be able to differentiate and modify the process according to the default/inputted parameters and features.

Furthermore, it is a challenging task to print the photomask with admissible miss alignment in between the printings, even for the less complex boards. Hence, the existing process will always face a misalignment issue since it is virtually impossible to avoid it with the available tools. This task, however, does not make part of the necessary steps for the LDI solution. Therefore, this kind of miss alignment is a problem that is automatically solved while using the purposed application.

Another requirement of this work is to deliver hardware and software designed according to the laboratory and user’s needs. Therefore, time is clearly one of the crucial resources for professors and the scientific staff of the university’s laboratory.

Although the proposed solution has fewer steps than the actual laboratory manufacturing process, it does not necessarily mean that the printing time is lower than the photomask solution. Small and simple boards will demand less time to be engraved using LDI instead of photomask lithography, however, big and complex boards may demand more time to be engraved using LDI instead of photomask technology. For the user, nevertheless, it needs just a few seconds to autonomously engrave the production blank with the purposed hardware, independently of the board size and complexity, while it actually takes a relevant amount of time for the user to prepare the photomask and engrave the board, independently of the board size and complexity, if using the existing solution.

In addition, the laboratory has faced difficulties to cover the developed PCB prototypes with solder masks. Therefore, is also the intention of this work to use the developed solution to assist the laboratory in the solder mask procedure since the LDI technology can also be used at the photoexposure step for solder mask application.

The main and sub research questions are:

- **For which circuit design would be better to print using the developed LDI solution in comparison with the photomask solution available at the university’s laboratory?**

  I. How the configuration parameters, such as spindle speed, feed rate, acceleration, etc., would influence the line profile?

  II. How the physical limitations of the hardware could influence the printed design?

  III. How can dedicated software development improve hardware usability?

5.1 Objective

As mentioned in Section 1.1 the actual solution demands plenty of time for the user and uses tools that were not developed to print photomasks. Section 2.4.2 subsequently describes the limitations of photomask lithography that can be passed over with LDI technology. Therefore, the main purpose of the developed tool is to propose effective pattern printing for the given PCB design (either for photoengraving or solder mask application), not only allowing imaging the circuit accurately and with fewer steps, but also allowing the user to focus on a side task while the hardware prints autonomously the circuit pattern.

To properly control the hardware, this work also intends to deliver software that attends hardware, laboratory, and user specifications. The desired result of this work is to deliver a tested, parametrized, and safe tool to the university’s laboratory that brings benefits regarding the consumed time and accuracy when compared with the technology actually used.

5.2 Software Tool

The information of traces, pads and other elements' location inside the Gerber files must be processed and translated to manufacturing information in the form of G-code files (a language which the CNC machine can interpret) with the assist of software. Section 2.4.1 describes that the part exposed to the UV light is the one that will be removed after the development and etching steps. Therefore, the necessary information to generate the inputted design becomes the areas where the copper should be removed and not the area where it should be kept, which is informed in the Gerber files.

Some software is able to isolate the copper pattern or even able to generate multiple passes to remove the undesired copper on the board. Hence, the translation from CAD, contained in the Gerber file, to CAM would be an intermediary step in between the image format file and the NC format file, which would require familiarity and certain expertise of the user with the selected software. Although, once that the user gets used to the software, it would become relatively simple since the steps to translate from CAD to CAM are the same for all PCBs independent of the size and complexity.

Those characteristics can be found in FlatCAM's software, which makes it an excellent API to work as a format translator for the software developed in this work. FlatCAM is open-source, written in Python, and scriptable, which means that the commands can be streamed to the software in a sequence. The FlatCAM command-line interface can run a script containing the FlatCAM commands eliminating the necessity of a CAD/CAM designer with some familiarity with the software to do the format translation. In addition, automatizing the translation process leads to a translation time span of about 30 seconds, depending on the system processing capacity and PCB complexity. The same process would take some minutes if manually done by a user.

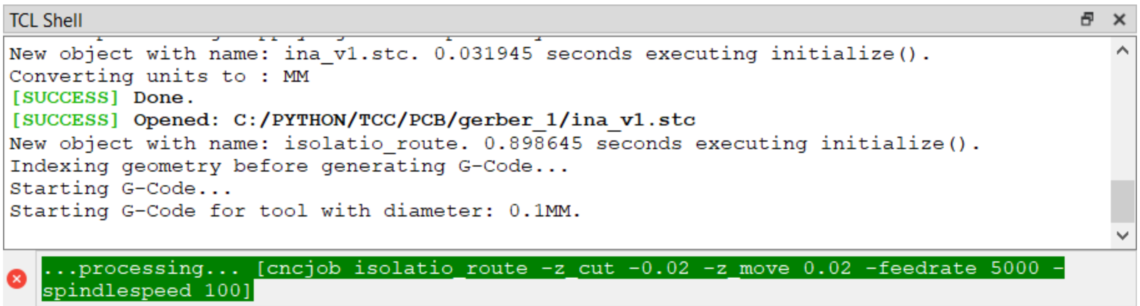


Figure 5.1 – FlatCAM command-line interface running a script

5.3 Layers Alignment

As the actual photoengraving laboratory's solution, this work is not limited to single-sided board manufacturing. Since, for photomask lithography, the bottom and top side of the PCB must be simultaneously printed, both photomasks must be positioned aligned on the production blank. This alignment should be done in a way that a hole done in the top layer pad would go through the board and connect to the correct pad on the bottom layer. Again, it is already a demanding task for simple circuit designs, considering that the alignment is done with naked eyes and without the assistance of any tool, which for complex designs become, once again, virtually impossible.

In the case of CNC machines, each side is printed at a time. Therefore, the PCB must be flipped to print the opposite side. However, just arbitrarily flipping and reallocating the PCB to print the contrary side is not enough since the machine will lose the position and orientation of the previously printed circuit design.

Hence, a common solution for the positioning trouble is a pin table. A pin table, illustrated in Figure 5.2, is a plate with holes organized as a matrix where the board can be pinned to the table. With the assist of this table, the machine is then able to locate where the PCB is according to the pins used to place the board.

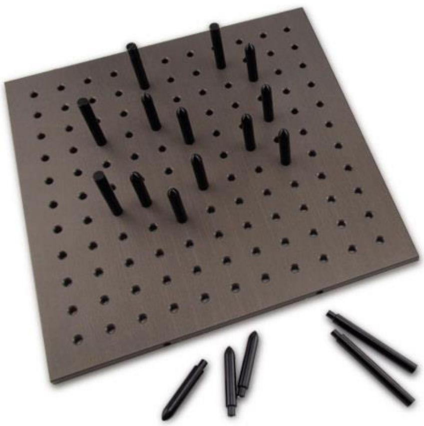


Figure 5.2 – Pin table

Always flip the PCB around a fixed axis is a better practice than arbitrarily flipping the board. Of course, the axis of reference for the board must agree with the axis of reference used by the machine. Another solution is to inform the machine which axis was used as a reference to rotating the PCB.

For the solution developed in this work, the axis of reference used for flipping is the X-axis.

5.4 Project Scope

The context of the project is to adapt an available CNC hardware solution to a computer-aided laser lithography machine for PCB prototyping. The research will therefore focus on understanding how the physical characteristics of the hardware can be controlled by software for the desire application. Nevertheless, the results of this project are the understanding of the machine parameters, to correctly configure it for PCB photoengraving and solder mask application, besides the understanding of the machine for further possible modifications or adaptations.

All the tests have the purpose to explain the reason why the value was defined as the default parameter for the machine. All incoming component values and initial parameters are assumed to be within the specifications of the supplier. Of course, it would be possible to examine possible fluctuations of those values due to damaged or poor-quality components supplied, but this is considered incidental as far it has a not relevant deviation and does not influence hardly at the experiments conclusion.

Due to the time given to write this work, it must limit itself to deliver a lithography solution with a clear advantage to convince the user to utilize the new tool, always that feasible, rather than the old one. Its software is not developed to be failure-proof, but a functional interface to allow the user to set and configure his design. It is clear that further optimization of the software and hardware can/will be necessary after completing this work. Necessary and advised upgrades can be found in section 7.2.

6 Prototype

According to Moggride, the prototype is an existing representation of the final solution design [45]. Houde and Hill complement this definition by distinguishing the prototype from the final solution due to two functions – exploration and demonstration [46]. The importance of prototype as a learning medium is explained by Lim and Stolterman: "Prototypes are the means by which designers organically and evolutionarily learn, discover, generate, and refine designs" [47]. To summarize, the manufacturing of prototypes is important to understand the final application and experiment with it.

Prototyping may bring several benefits for the final design, such as [48]

- **Design relevance**: when there are no secure experimental results about the final product, prototyping conducts to early identification of issues and dead ends, fixing them at an early stage, and reducing the costs when compared to find the same issue at late stages.
- **Cognitive**: thought and action are profoundly linked and co-produce learning. "thinking and doing" increases the chance of the design to succeed when comparing to just hard thinking about the problem, since visualizing the problem helps to solve what was abstract before. As well, the opportunity of offloading tasks to the prototype helps the designer to be more productive.
- **Reflective practice**: the interchange between creator and created vehicle helps to expose issues or new design suggestions since surprises that were not expected by the creator can come from this interchange.
- **Teaching**: Inventors tends to develop a better understanding of which ideas may work or not by the constant interchange, driving them to overlook the consequences of their intentions through prototype test.

Since the tool developed in this work has the intention to understand a computer-aided laser lithography machine, it is considered as a prototype of the final tool. The prototype will be able to perform computer-aided laser lithography, but it is expected to engrave design with some errors that may be not feasible to overcome due to hardware limitations.

With the understanding accumulated with this research, the prototype tool can be enhanced, adapted, or even utilized as a reference for the development of another laser lithography machine.

This chapter will be dedicated to delineating how the prototype was idealized, describing what was learned about the tool, and its scope.

6.1 Technology Selection

Some of the important steps of prototype and product development are: correctly select the technologies to use, decide which parts should be developed, and which parts should be bought from suppliers. This section intends to briefly describe how and why the technologies used for hardware and software were selected.

6.1.1 Hardware

The first decision to be done was about the CNC machine. It was necessary to decide between a design as part of the scope of the project or a design bought from a supplier and adapt the supplier's solution for the purpose of the work.

The design process of a CNC machine, such as calculation of the structure, motors, development of electronics, could be considered a whole thesis itself. Therefore, since the goal of this work is to develop a computer-aided lithography solution for the electronics laboratory of the university, and not develop a CNC machine for them, it was decided to opt for a supplier solution that offers a feasible product and adjust this solution for the purpose of this work.

The product should have the following characteristics:

- **Minimum engraving area of 297 mm x 210 mm** to agree with the possible PCB size that other laboratory tools can work.
- **High engraving accuracy**, at least 0.01 mm to be able to manufacture more complex PCBs.

- **In a budget between 200 and 300 Euro**, reasonable value agreed upon for the development of the prototype.
- **Supplied by a German store**, to avoid possible bureaucracy at the order.
- **Available**, to be able to work with the equipment as soon as possible

From several market options, the EleksMaker® EleksLaser A3 Pro was chosen to be adapted. It has a working range of 380 mm x 300 mm, an engraving accuracy of 0.01 mm, and was available at Real.de for less than 140 Euro. [49]

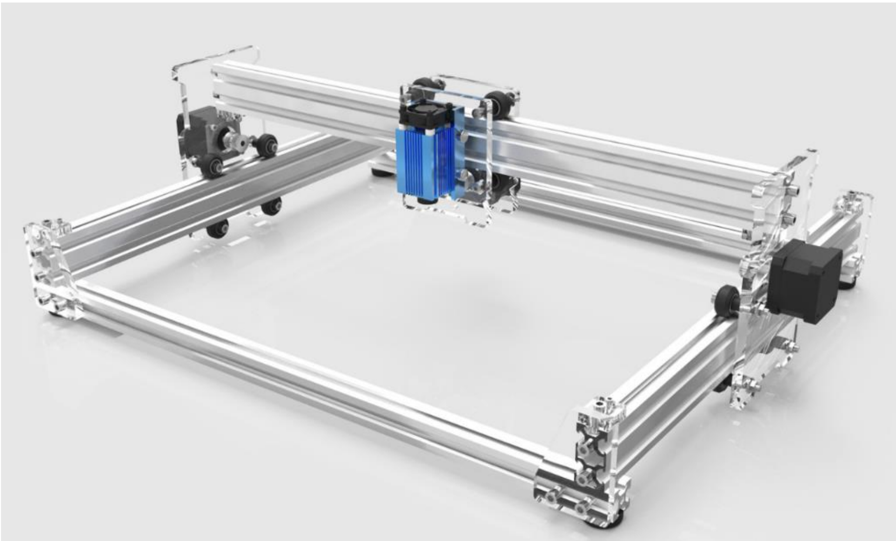


Figure 6.1 - EleksMaker® EleksLaser A3 Pro (Source: EleksMaker® official website)

This engraver machine, however, does not come with a laser module, therefore it should be bought apart. Since the intention of the direct imaging technology is to expose the production blank covered by a photoresist layer, and not burn the photoresist or copper, a laser that works in a relative low-power range fits better the application. Furthermore, the laser also needs to emit a violet/blue light to generate the photoreaction in the photoresist. Hence, it is preferable to select a laser module that emits a laser beam with a wavelength close to 400 nm, to produce UV light.

The selected laser module was a 500 mW blue laser (405 nm) also from EleksMaker®. The laser module comes already with a TLL/PWM modulation circuit, which allows controlling the percentage of power used. Therefore, the select laser would allow to make experiments in a power range of 5 mW (low power) to 500mW (moderate power).

There was no urge to substitute any component or part from the laser engraver, with exception of the microcontroller. The EleksMaker microcontroller is an Arduino nano clone that is not compatible with Grbl v1.1. Hence, another Arduino nano clone with compatibility with Grbl v1.1 substituted the original microcontroller.

Another important part of the hardware is the element responsible to run the software and stream the information to the Arduino that controls the CNC machine. Since it would be unproductive for the user to have the software installed on his personal or office computer to run the software, it was decided that the machine would have a dedicated computer. Considering that the application does not need an apparatus with a high processing capacity, it would be excessive to place a computer exclusively for this hardware. Thus, a reasonable solution is to place a Raspberry Pi with a touch screen display available to do this function, which also improves the tool's usability.

6.1.2 Software

As for the hardware selection, the software selection could also be provided by a supplier or developed. EleksCAM is the application provided by EleksMaker for the EleksLaser A3 Pro. Perhaps, the application is limited and is not able to open Gerber files.

It is a requirement of this thesis to work with Gerber files as the photoengraving process input because of three main reasons. First, it is the standard industrial image format for PCB design. Second, the huge majority of circuit design software can export the CAD information as a Gerber file. Last but not least, the new and old Gerber format designs would not need to be redesigned to work exclusively with the new tool.

Therefore, as pointed out in section 5.2, FlatCAM is the CAM software that better fits the application due to the available features and its scriptability. However, script content changes according to the circuit design. Hence, it is reasonable to develop software to collect design information from the user, create a FlatCAM script and stream the generated G-code to control the hardware.

The software developed in this work is fully written in Python. Due to the advantages already mentioned in section 4.1, and the fact that it is the language where I, as a developer, felt comfortable creating the required application.

6.1.3 System Architecture

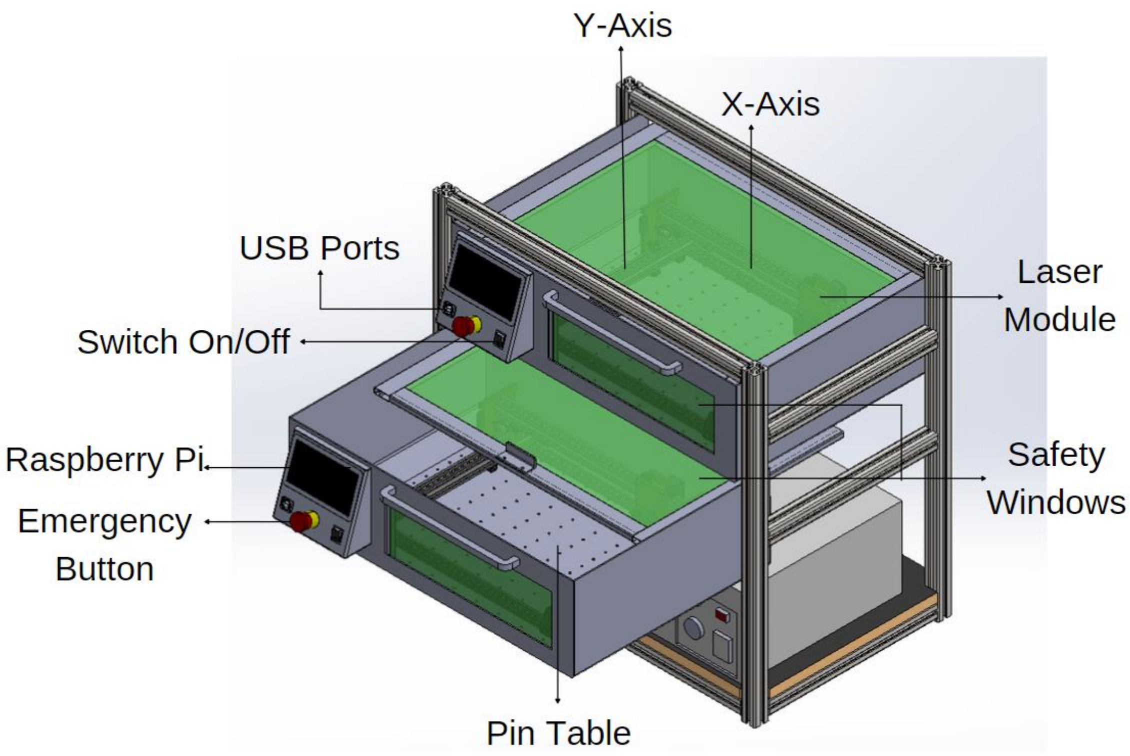


Figure 6.2 – Laser tool system

The university decides to have two machines, one for everyday usage and another to explore new parameter combinations. For safety reasons, the prototype developed in this work will be encapsulated in a metallic structure that slides to allow the user to place and remove the PCB for engraving/solder mask application. The machine contains some areas with laser safety windows to allow the user to visualize the process and align/calibrate the machine.

The control of the machine is done externally with the Raspberry Pi and the file to be processed should be in a USB stick that is connected to the machine USB port. The energy supply of the machine is controlled with a switch button. In addition, an emergency button is available in case of unpredicted malfunction of the machine.

Table 6.1 – System specifications

Transmission system		Control board	
Type	Belt and pulley	**Microcontroller**	Arduino Nano
Motor	42BYG Stepper Motor	**Microchip**	ATmega328P
Detent Torque	40N.cm Min	**Communication Port**	Micro USB
Holding Torque	2.2N.cm Max	**Supported Motor**	42 Stepper Motor within 2A 2-phase 4-wire
Step Angle	1.8˚	**Driver**	A4988
Laser Module		**Embedded System**	
Type	Diode	**Computer**	Raspberry Pi 3B
Laser Power	500 mW	**CPU**	Quad Core 1.2GHz
Wavelength	405 nm	**RAM**	1GB

6.2 Software Design

The HSRW Laser Tool software is a graphical user interface (GUI) that allows the user to easily access the features and parameters input to generate the desired PCB design. The software consisting of three main parts, as shown in Figure 6.3.

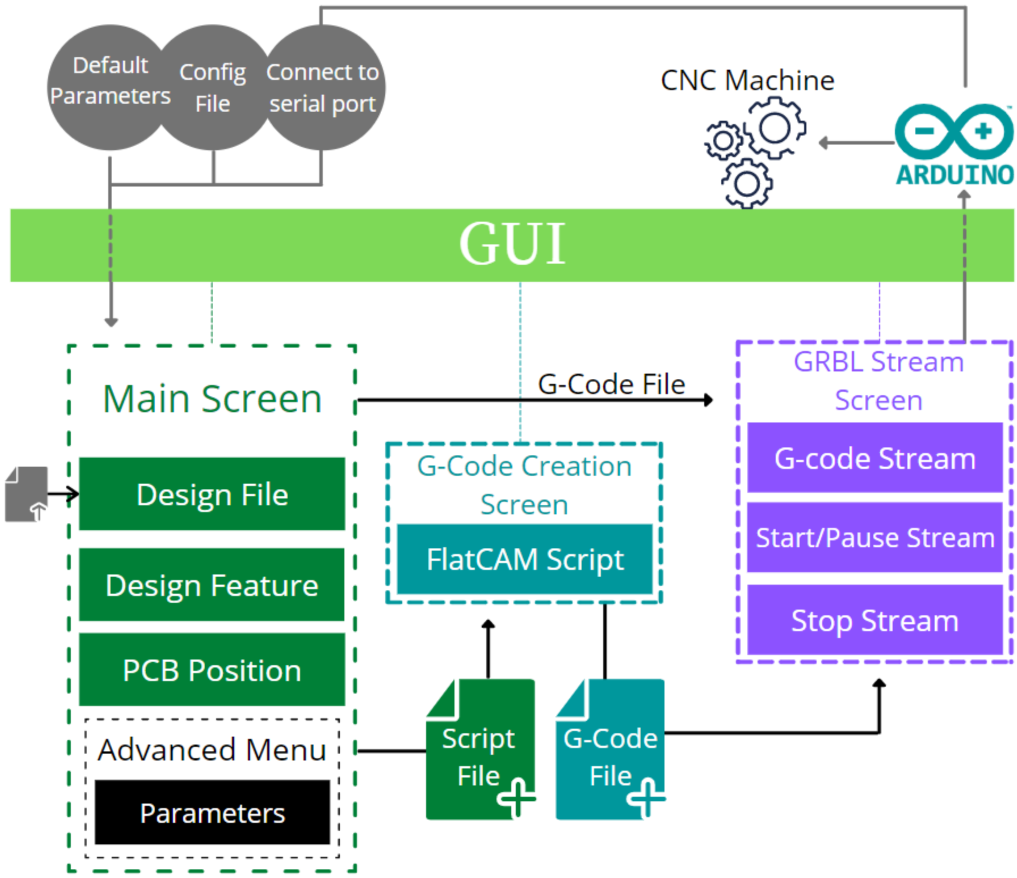


Figure 6.3 – Schematic overview of the software tool

1. The default parameters and configuration files need to be parsed, and a connection with the serial port must be established. After the parsing and connection of the serial port, the main screen will be available containing two menus.

   1.1. The **main menu** containing a frame to select the Gerber file or G-code that will be used, a frame with a pin matrix (as explained in section 5.3 and section 3), a frame with the design feature options (G-code, isolation, removal, and mask), and a start button.

   1.2. The **advanced menu** giving the user the option to change the browse root directory, change the default parameters, and save the changes done or just keep it as default until the next software initialization.

2. Gerber files must be transformed into G-code according to the selected design feature and parameters. After pressing the start button, the G-code Generation screen is

displayed and a script file containing the features and parameters that must be streamed to the FlatCAM command-line interface is generated. Finally, the FlatCAM is initialized and after successful scripting, a G-code file is created containing the CAM information to print the desired PCB design.

3. Finally, at the Grbl stream screen, the lines of the G-code file (generated by the FlatCAM scripting or inputted as design file at the main menu) are streamed to the Arduino running the Grbl. The streamed lines will be interpreted by the Grbl which controls the machine accordingly. The stream of lines can be started, paused, and stopped while on this screen.

The following sections are dedicated to explaining with more detail each screen and its activities using the flowchart diagrammatic representation.

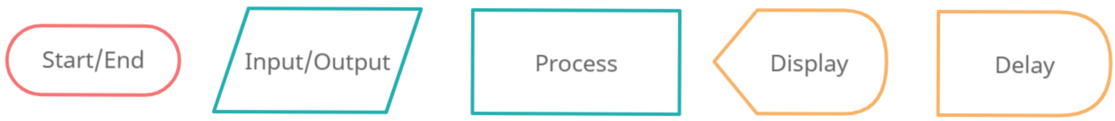


Figure 6.4 – Flowchart Legend

6.2.1 Splash Screen

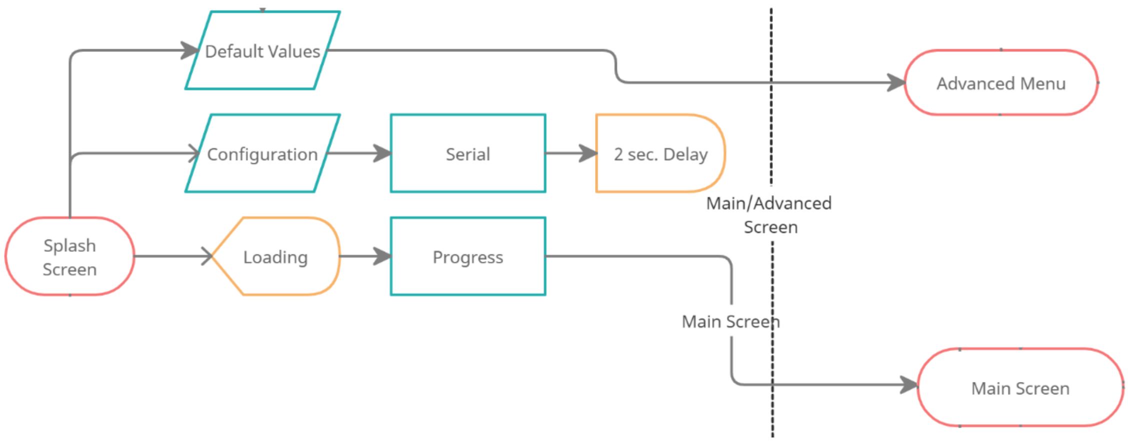


Figure 6.5 – Splash screen flowchart

The Splash screen has the function of initializing the software. It loads the default values of the parameters, which will be displayed at the Advanced Menu screen, and the configuration file containing the port to which the Arduino is connected.

Two seconds delay is used to assure the connection of the Arduino before streaming any command. To communicate to the user that the software is importing the necessary files and making the connection with the Arduino, a progress function is used to what is being loaded in the back-end. After the conclusion of the progress function, which takes 3 seconds, the Main/Advanced Menu screen is called and the Splash screen closes.

Figure 6.6 – Splash screen

6.2.2 Main/Advanced Menu Screen

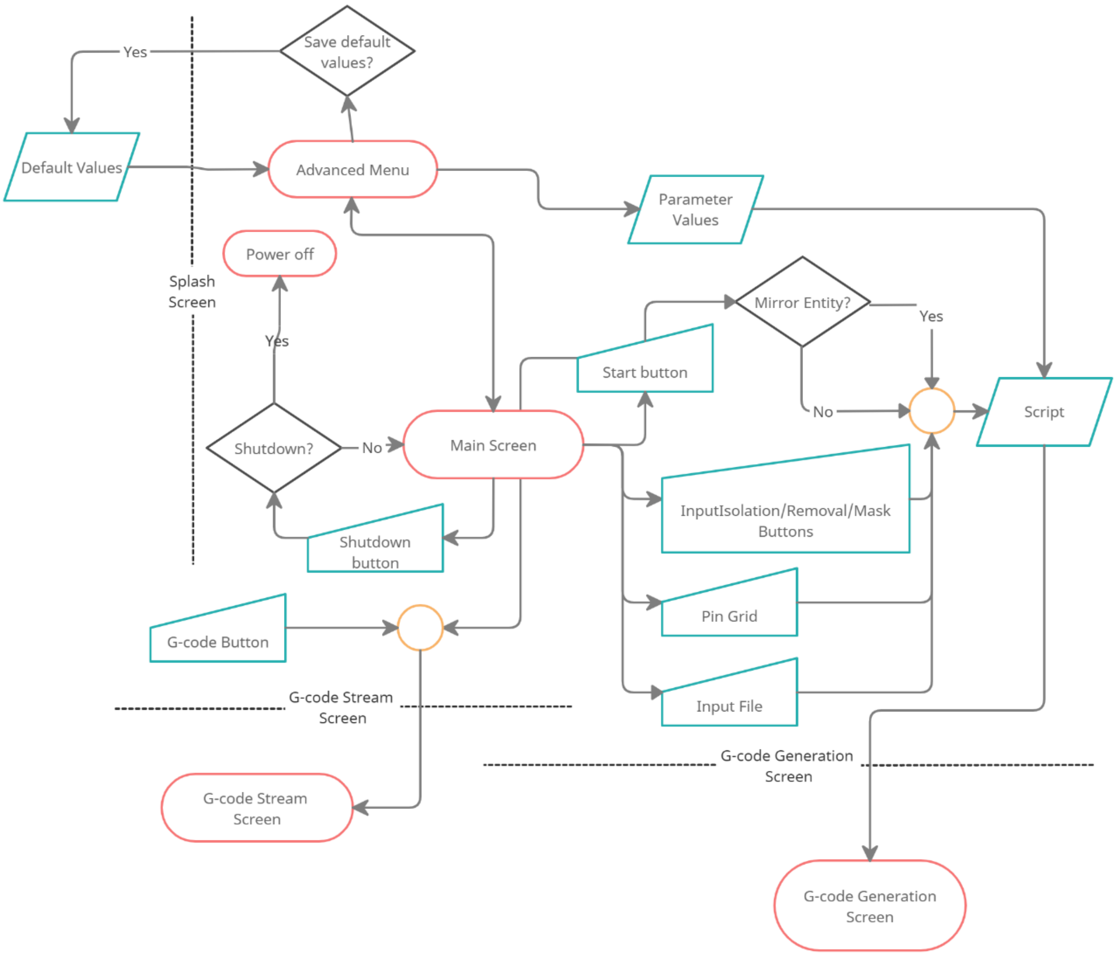


Figure 6.7 – Main/Advanced Menu screen flowchart

The main window and the advanced menu shares the same screen and its display can be switched using the advanced/return button at the top of the screen.

At the Advanced Menu screen, the user has access to the parameter values, the possibility to change them, and to save the changes done as default or just keep the changes until the application is shut down as before explained in section 6.2.

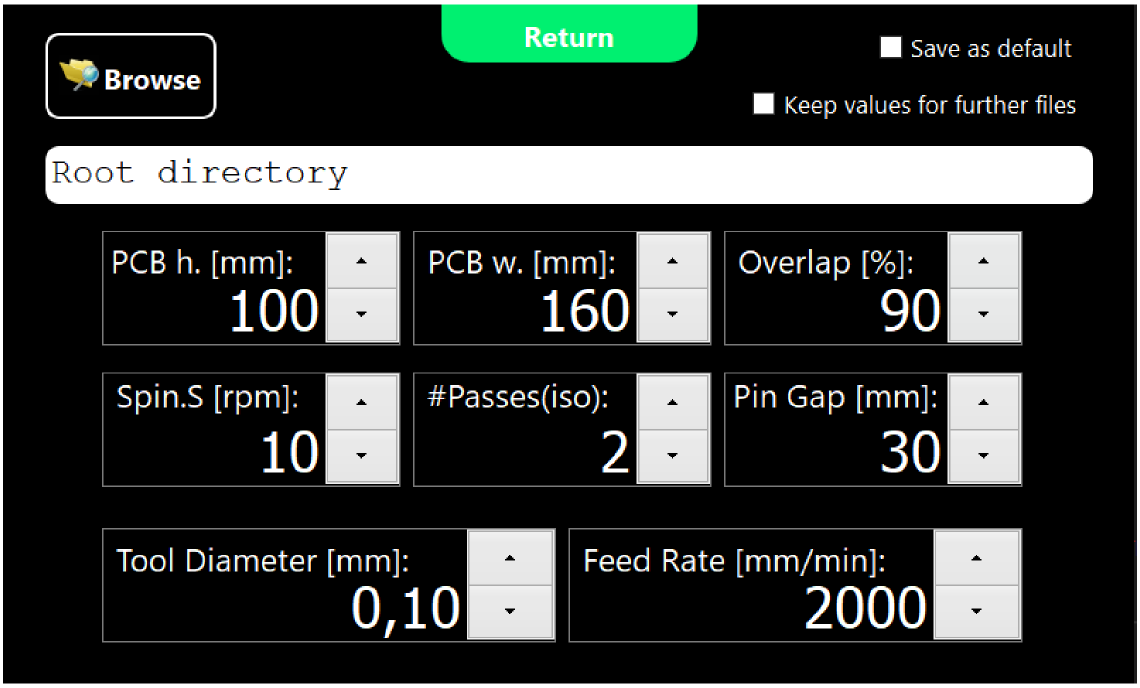


Figure 6.8 – Advanced Menu screen

At the Main screen, the user has access to various buttons such as the shutdown button, the browser button to inform which design file the user wants to use, the features buttons which give the feature options that the user can apply, and the pin checkbox buttons which indicates the pins that were employed to place the production blank.

After press start the software checks which feature button is selected and do one of the following actions:

1. If the G-code feature button is selected the Main/Advanced Menu screen closes and the G-code Stream screen is called.

2. Else, a pop-up is displayed asking if the user wants to mirror the feature and add to the user inputs information (feature button selected, parameter values, production blank location in the grid, design file, and its directory) to create the FlatCAM script.

The script is generated using a standard script file as a reference and replacing the standard information by the user input information on its proper location. After editing the script file, the Main/Advanced Menu closes and the G-code Generation screen is called.

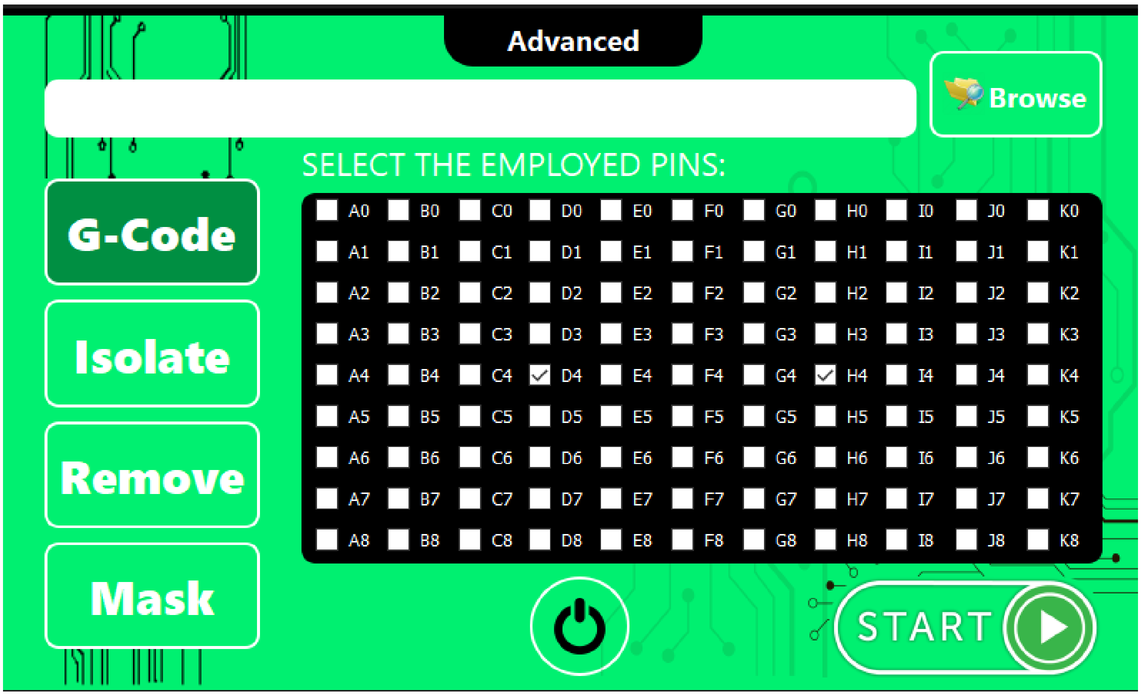


Figure 6.9 – Main screen

6.2.3 G-Code Generation Screen

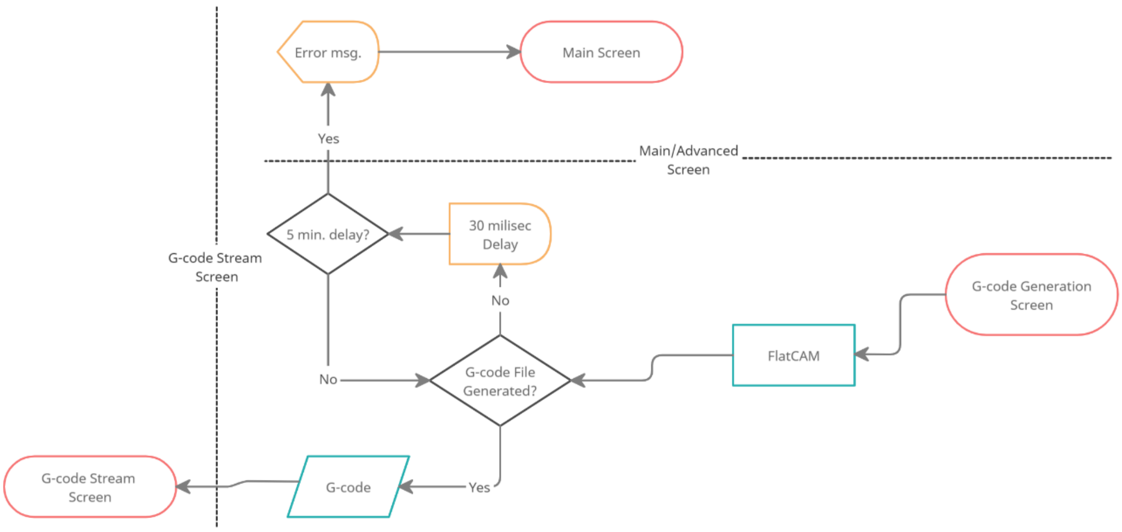


Figure 6.10 – G-code Generation screen flowchart

The G-code Generation screen is basically a loading interface that is running a script in the FlatCAM command shell as a subprocess. The software looks for a “gcode.txt” file in the directory once that the G-code Generation screen is called and if there is a file in the software's directory with this name, it will be deleted. Then, the software starts the FlatCAM as a child process and checks every 30 milliseconds if exists a "gcode.txt" file in the directory. The loop is repeated, at a maximum, 10000 times (5 minutes). If a "gcode.txt" file is generated before the loop runs for 5 minutes the G-code Stream screen is called, and the G-code Generation screen, and the FlatCAM shell command, closes. Else, an error is popped up indicating timeout, closing the FlatCAM command shell and the G-code Generation screen, while calls the Main/Advanced Menu screen.

Is important to inform that the G-code may take a longer time to be generated if a large and complex design is given. Even that no G-code generated during this work took more than one minute, five minutes was decided as timeout to give the program enough time to process the designs that may take longer to be concluded.

The format of the design file and the content of the script file should be checked in case that simple and non-complex design reaches the timeout. The FlatCAM will interrupt the scripting and not generate a G-code if an error occurs during the process, which will hold the software in the G-code Generation screen since no "gcode.txt" file was not founded in the directory until it reaches the timeout.

It is important to mention that FlatCAM supports just computer-friendly syntax for files and directory, such as no blank space in the folders and file name and use of "/" instead of "\". Therefore, it is important to check if the file directory and file name informed in the script file follows this syntax in case that a non-complex design reaches its timeout.

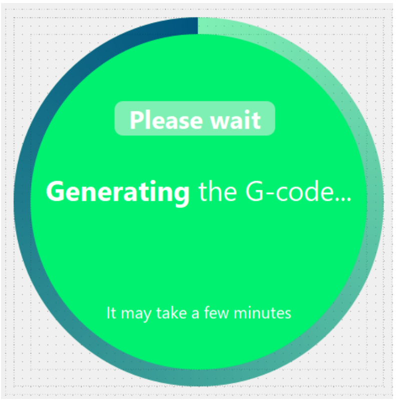


Figure 6.11 – G-code Generation screen

6.2.4 G-code Stream Screen

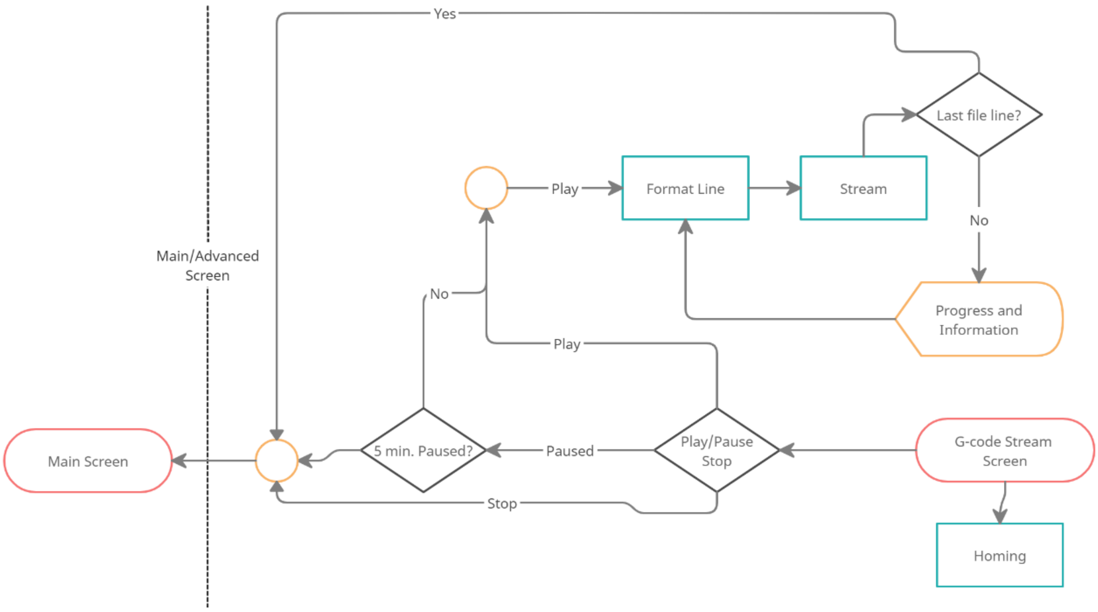


Figure 6.12 – G-code Stream flowchart

The G-code Stream screen can be called directly by the main menu, selecting the G-code featuring, or by the G-code Generation screen after generating the “gcode.txt” file as mentioned in section 6.2.2 and section 6.2.3. Once that this screen is called the homing function is invoked sending the machine to its home position. Once that the machine is in its home position the user can play, pause or stop the streaming.

After starting to stream, the software reads the “gcode.txt” file line by line and set up them to the correct stream format with format line function, which changes Z-words (used at milling) to laser mode words (M3, laser on, or M5, laser off), ignores the comment and empty lines, removes spaces at the beginning and at the end of the line, and separates the NC-words with space. Once that all the lines were streamed, the G-code Stream screen is closed and the Main screen is called.

The stream can be held by clicking the pause button. If it stays for more than 5 minutes paused the G-code Stream screen is closed and the Main screen is called. The software continues to stream the lines again, from where it was stopped if the play button is pressed while the stream is paused.

The stop button interrupts the stream of lines once that is pressed, closing the G-code Stream screen and calling the Main screen.

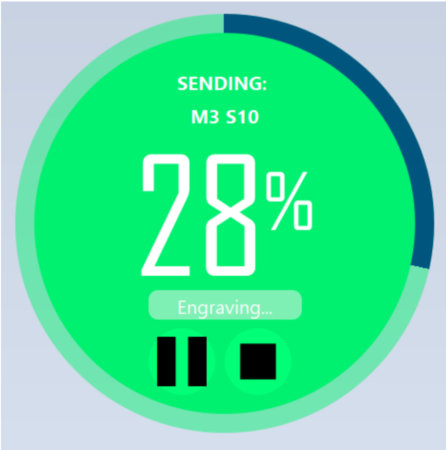


Figure 6.13 - G-code Stream screen

6.2.5 Benefits of the Software Design

The following are the major benefits of the software design:

- The software developed delivers a platform that unites the steps of manufacturing design and hardware control in one single application.
- It requires no coding skills.
- It removes the necessity of the user to understand how to convert a CAD into a CAM design.
- It displays to the user just the relevant features for the translation process.

6.3 Grbl Settings

Grbl is a software developed to be able to work with various CNC machines containing distinct component configurations, sort of transmission systems, and motors, which makes it highly possible that the standard-setting values of Grbl do not agree with the setting values from the tool developed in this work.

There is a list of settings that can be accessed by typing $$ at Arduino's serial monitor running Grbl v1.1. The list of configuration commands and a brief explanation can be found in Table 6.2.

Table 6.2 - Grbl settings

Settings	Description	Settings	Description
$0	Step pulse, microseconds	**$26**	Homing debounce, milliseconds
$1	Step idle delay, milliseconds	**$27**	Homing pull-off, mm
$2	Step port invert, mask	**$30**	Max spindle speed, RPM
$3	Direction port invert, mask	**$31**	Min spindle speed, RPM
$4	Step enable invert, boolean	**$32**	Laser mode, boolean
$5	Limit pins invert, boolean	**$100**	X steps/mm
$6	Probe pin invert, boolean	**$101**	Y steps/mm
$10	Status report, mask	**$102**	Z steps/mm
$11	Junction deviation, mm	**$110**	X Max rate, mm/min
$12	Arc tolerance, mm	**$111**	Y Max rate, mm/min
$13	Report inches, boolean	**$112**	Z Max rate, mm/min
$20	Soft limits, boolean	**$120**	X Acceleration, mm/sec^2
$21	Hard limits, boolean	**$121**	Y Acceleration, mm/sec^2
$22	Homing cycle, boolean	**$122**	Z Acceleration, mm/sec^2
$23	Homing dir invert, mask	**$130**	X Max travel, mm
$24	Homing feed, mm/min	**$131**	Y Max travel, mm
$25	Homing seek, mm/min	**$132**	Z Max travel, mm

Since the machine does not move in the Z-axis direction, there is no necessity to change the standard values and no necessity to discuss the settings related to the Z-axis. This section will focus on discussing the relevant settings for the application and finding the correct value to adapt the EleksMaker A3 pro to the application desired with this work.

6.3.1 $1 - Step Idle Delay

This command locks the stepper motor for an amount given in milliseconds before disabling the motor. If $1=200, for example, the motor will be lock in the position for 200 milliseconds before get disable. The maximum possible value given to this setting is 255 milliseconds.

It is recommended to set this setting to 25-50 to guarantee that the motor will not move after get disabled due to the inertia of the movement. To guarantee that the motor stops but without stay for too long at the corners creating, the step idle delay selected for the tool was 25 milliseconds with the intention to reduce the diameter of the undesired dot patterns as illustrated in Figure 6.14.

Figure 6.14 – Undesired dot pattern at the corners due to a long laser exposure

6.3.2 $30 - Max Spindle Speed and $31 - Min Spindle Speed

Grbl linearly relates the max-min spindle speeds to a max-min pulse width modulation (PWM) with a range of 5V - 0.02V. In the case of a maximum spindle speed of 1000 rpm, it is necessary to set $30=1000 corresponding to 5V PWM. If the G-code contains a spindle speed command higher than the set max spindle speed, Grbl will still output 5V or 1000 rpm. The minimum spindle speed corresponds to 0.02V and generally is set as $31=0. It is also important to inform that a G-code indicating a spindle speed equals 0 “S0” will not stop the rotation but rotates it at the lowest possible velocity.

When the laser mode is active, the max-min spindle speeds represent fractions of the laser power and not rotation speed as described in section 4.3.1. The max spindle speed selected for the tool was 1000 and its min spindle speed set to 0 for an easier identification of the laser power used. In this case, each spindle speed unit represents 0.1% of the laser power.

6.3.3 $32 - Laser Mode

Initially, Grbl was created to turn Arduinos microcontrollers into a CNC milling machine's controller. Therefore, at the Grbl's default mode the machine stops every time an M3, M4, or M5 spindle state or an Sxxx spindle speed is called, which leads to scorching and uneven cutting/engraving for laser CNC machines. To avoid it, Grbl released a new model for laser machines preventing needless stops resulting in a clean outcome, even for low-acceleration machines. [50]

Since the tool developed in this work is a laser machine it was set $32=1 to turn the laser mode on.

6.3.4 $100 and $101 – X and Y steps/mm

This setting indicates how many steps the motor must realize to move the axis 1 mm. Since the transmission system and the motor from the X-axis and Y-axis are the same, it is expected that the $100 value is equal to the $101 value. Furthermore, it is important to configure this setting correctly to be able to print a PCB with the designed dimensions. If the correct setting would be 100 steps/mm and it is set as 80 steps/mm a square that should be 1 mm x 1 mm would be printed as a 0.8 mm x 0.8 mm square instead due to the wrong setting.

This value can be calculated using the equation:

$$steps\ per\ mm = \frac{number\ of\ steps\ per\ revolution}{\frac{Timing\ pulley\ diameter\ \times\ \pi}{1\ rev}} \times\ micro\ stepping \tag{1}$$

Where the number of steps per revolution is 200 steps/rev, the timing pulley diameter is 12.7 mm, and the micro-step is 16, which leads to steps per mm of 80.2 mm. Therefore, the values of X steps/mm and the Y steps/mm were set $100=80=$101.

To find the error for the calculated setting was streamed to the machine command to draw three lines in the X-axis direction and the other three lines in the Y-axis direction, all containing 250 mm.

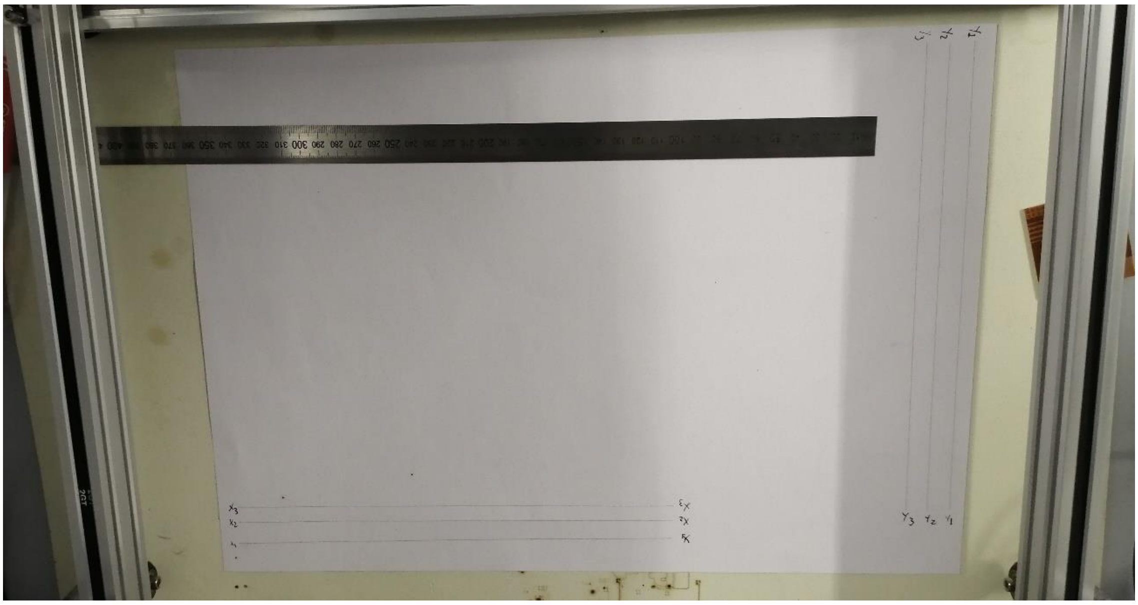
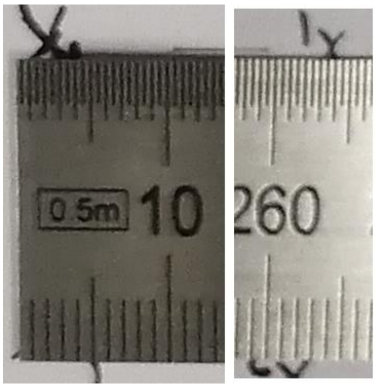
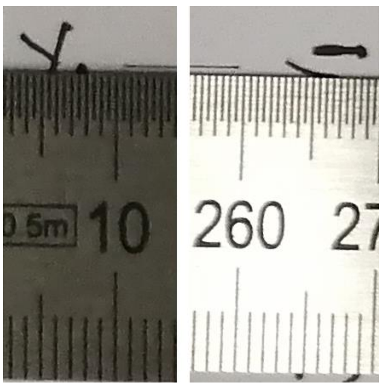


Figure 6.15 – 250 mm lines printed in the X and Y-axis direction

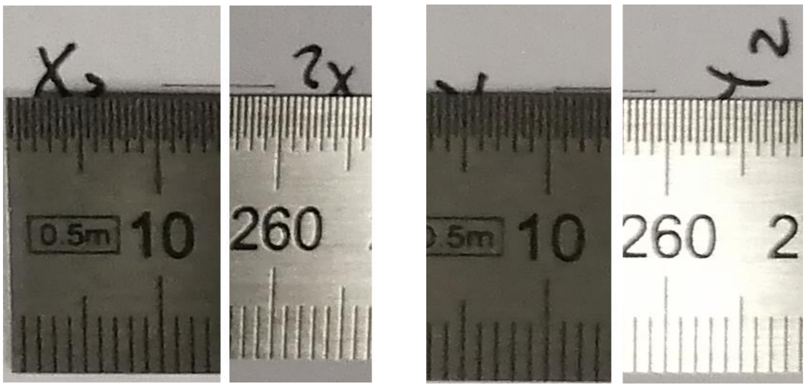
The lines printed had an average length of 249 mm, for both axis directions, which leads to a relative error of 0.4%. This error can be considered irrelevant since the lines printed in a PCB are shorter, leading to an error close or smaller than the engraving accuracy of the machine.



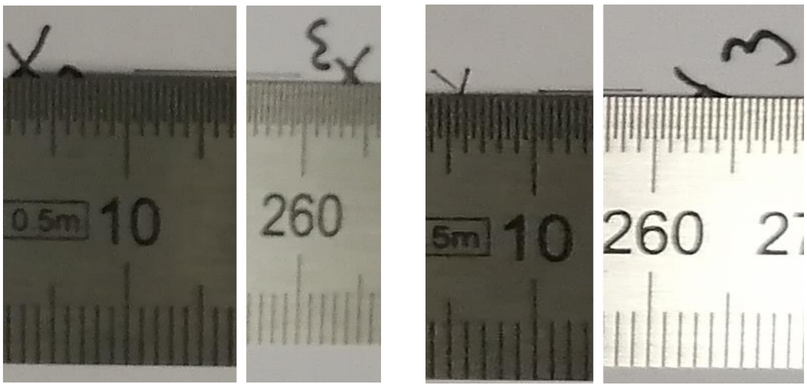
(a) 1st line in the X direction



(b) 1st line in the Y direction



(c) 2nd line in the X direction (d) 2nd line in the Y direction



(e) 3rd line in the X direction (f) 3rd line in the Y direction

Figure 6.16 – Printed lines measurement

6.3.5 $110 and $111 – X and Y Max Rate, mm/min

The X and Y max rate limits the max speed rate of the motor. As for max spindle speed, in the case that the value is set as 1000 mm/min any F-word indicating a speed higher than 1000 mm/min will be interpreted as 1000 mm/min.

The official supplier, EleksMaker, informs that the max rate of the hardware is 400 mm/s [51], or 24000 mm/min, which is an odd value for such a machine. A way to roughly identify the max rate and validate this value is by moving the axis at high speeds. In the case that the motor is commanded to rotate at higher speeds than it was designed to deploy, it will stall making a bit of noise. The stall should not damage the motor, however, it is preferable to increase the max rate (starting from a low value) than to decrease it (starting from a high value) to avoid unnecessary risk. It would also reduce the range of values that should be carefully evaluated in case that the max rate is much lower than 24000 mm/min as informed by the supplier, which is expected.

To identify the characteristic noise, the motor was commanded to move 50 mm, in the positive direction, and return to its initial position, for both axes directions. As for X and Y steps/mm in section 6.3.4, it is expected that both axes have the same max rate. It was decided to start with an initial max rate of 1000 mm/min and increase it in steps of 1000 mm/min until the identification of the noise, which was perceived at 6000 mm/min. Therefore, speed rates until 5000 should be considered for a deeper evaluation.

The X and Y max rate was set $110=2000=$111. The reason for it will be further discussed and derived in section 6.4.2 by comparing how the line thickness changes according to the feed rate.

6.3.6 $120 and $121 – X and Y Acceleration, mm/sec^2

The X and Y acceleration defines how fast the motor will reach the commanded feed rate. As shown in Figure 6.17, higher accelerations drive the motor to arrive at its maximum speed faster, while lower accelerations would take a longer time to arrive at the same maximum speed.

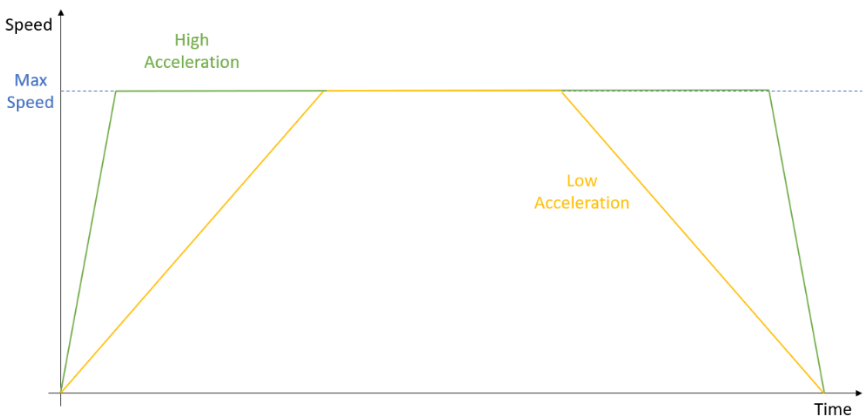


Figure 6.17 – Time vs speed diagram for low and high acceleration

For laser machines, the acceleration factor influences directly the quality of the line since the line thickness increases directly proportional to the amount of energy absorbed by the photoresist. It means that motors accelerating slowly will have a shorter time at constant speed than motors with high acceleration. Therefore, it is interesting to accelerate the motor as fast as possible to make the line thickness constant for as long as possible.

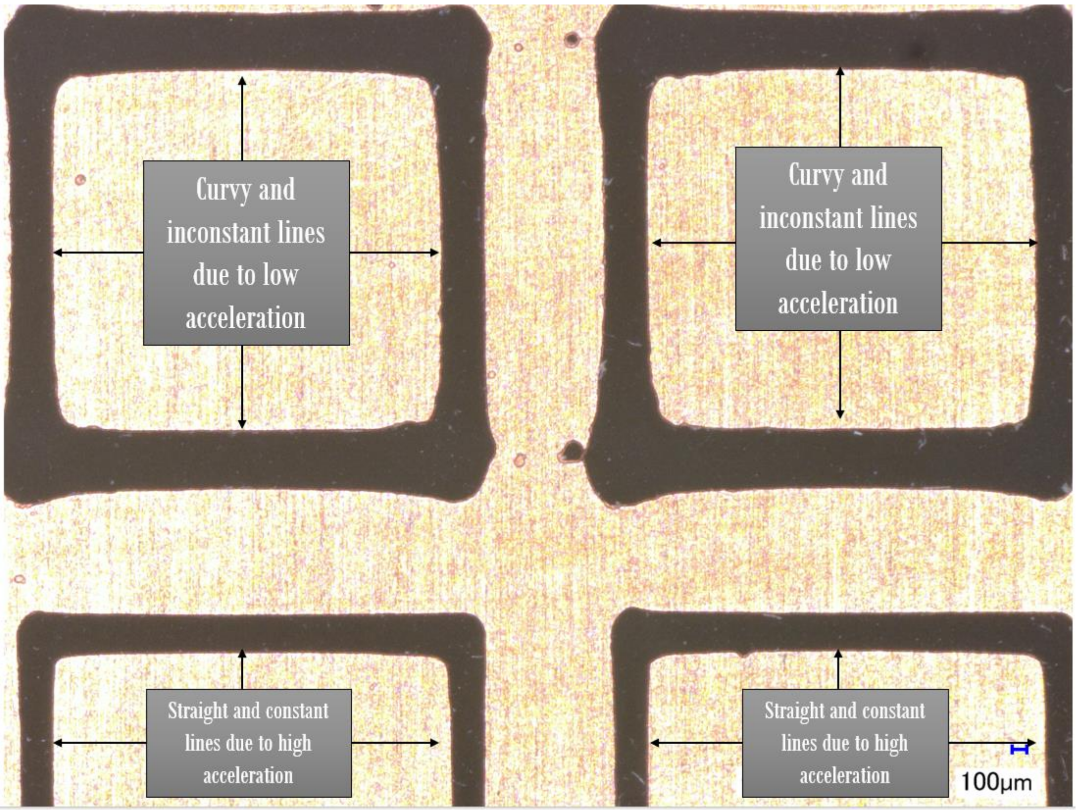


Figure 6.18 – Low and high acceleration effects on the line thickness for laser machines

However, for really high acceleration values the motor starts to behave asynchronously and starts to lose some steps along with the movement. Therefore, it is necessary to test some different acceleration values and see how it influences on the printed design.

To do so, a design containing a circle and vertical, horizontal, and diagonal lines was printed at a speed of 1000 mm/min while varying the acceleration values. Since it is easier to identify the acceleration effects while engraving paper if compared with engraving a production blank, it was decided to evaluate the acceleration effects not only at the production blank but also at a paper sheet.

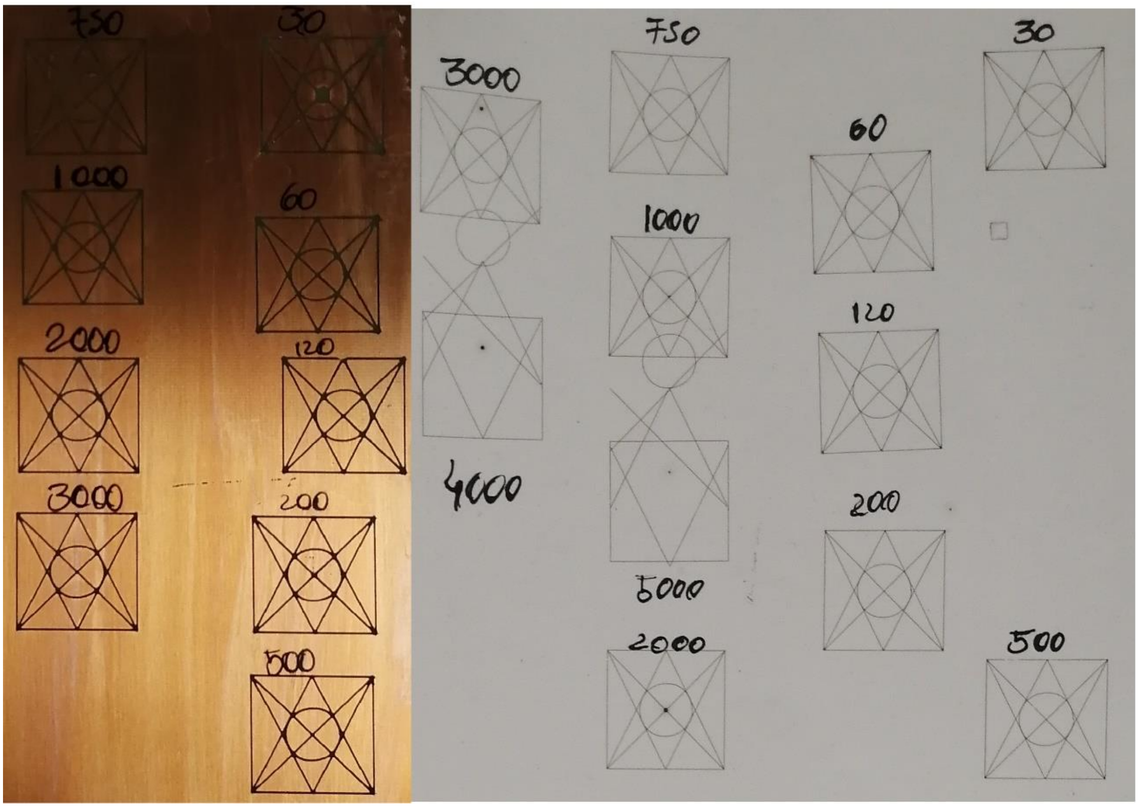


Figure 6.19 – Geometry printed at distinct acceleration values

30 mm/s^2	60 mm/s^2	120 mm/s^2	200 mm/s^2	500 mm/s^2	750 mm/s^2	1000 mm/s^2	2000 mm/s^2	3000 mm/s^2

Figure 6.20 – Zoomed view of the top-right corners of the geometry

The burnt vertices started to disappear, for both materials, after 200mm/s^2. However, the lines, in the paper, started to present open tracks caused by the loss of steps for accelerations equals to or higher than 500mm/s^2. After 4000mm/s^2 the motor started to behave asynchronously printing lines at completely wrong places. Therefore, the X and Y acceleration selected for the laser tool was $120=200=$121.

6.3.7 $130 and $131 – X and Y Max Travel, mm

The X and Y max travel indicates the limits of the machine, which means that once that the axis travels the indicated value it will stop and not be driven further forward. In a case that a CNC machine, with absolute system and $130=300, receives the command G00 X310, it will drive the X-axis until 300mm, stop, and ignore any additional command that intended to move the axis further but accept any command that moves the axis to an X-axis position lower than 300mm.

This command works as software limit switches, which is not recommended but useful if the machine has no limit switch component. Another utility of this setting is to limit the machine to a working ranger lower than its real limitation, which is the case of the application discussed in this work.

The maximum working range of the EleksLaser-A3 Pro 380mm x 300mm. However, as explained in section 6.1.1, the necessary working range of the machine is 297mm x 210mm (A4 size). Hence, to avoid that the axes get driven further than the desired working range the X max travel was set $130=297, and Y max travel was set $131=210.

6.4 Derived Parameters

There are many hardware parameters that influence the circuit design printing quality. Parameters such as the capacity to repeat accurately the same commands, power of the laser, feed rate. the precision of the travel commands may strongly influences how the machine prints the lines in the PCB and therefore the final result of the PCB design. Thus, those parameters must be carefully checked and derived.

Since the precision of the travel commands was already discussed in section 6.3.4 while setting the X and Y steps/mm setting of the machine, this section will discuss the lacking parameters and answer questions such as:

- How reliable is the machine for sequential cyclical commands?
- How does the line thickness change according to the laser power?

- How does the line thickness change according to the feed rate?
- What is the X and Y max rate?

6.4.1 Repeatability

Repeatability is the proximity of successive measurements containing the same magnitude under the same conditions [52]. It is certainly important for the CNC machine to be able to repeat the same commands as accurately and precisely as possible.

The repeatability of the machine was a check for each axis at a time using a dial indicator with a magnetic base. The dial was positioned as indicated in Figure 6.21 to check the repeatability of the X-axis and as indicated in Figure 6.22 to check the repeatability of the Y-axis.

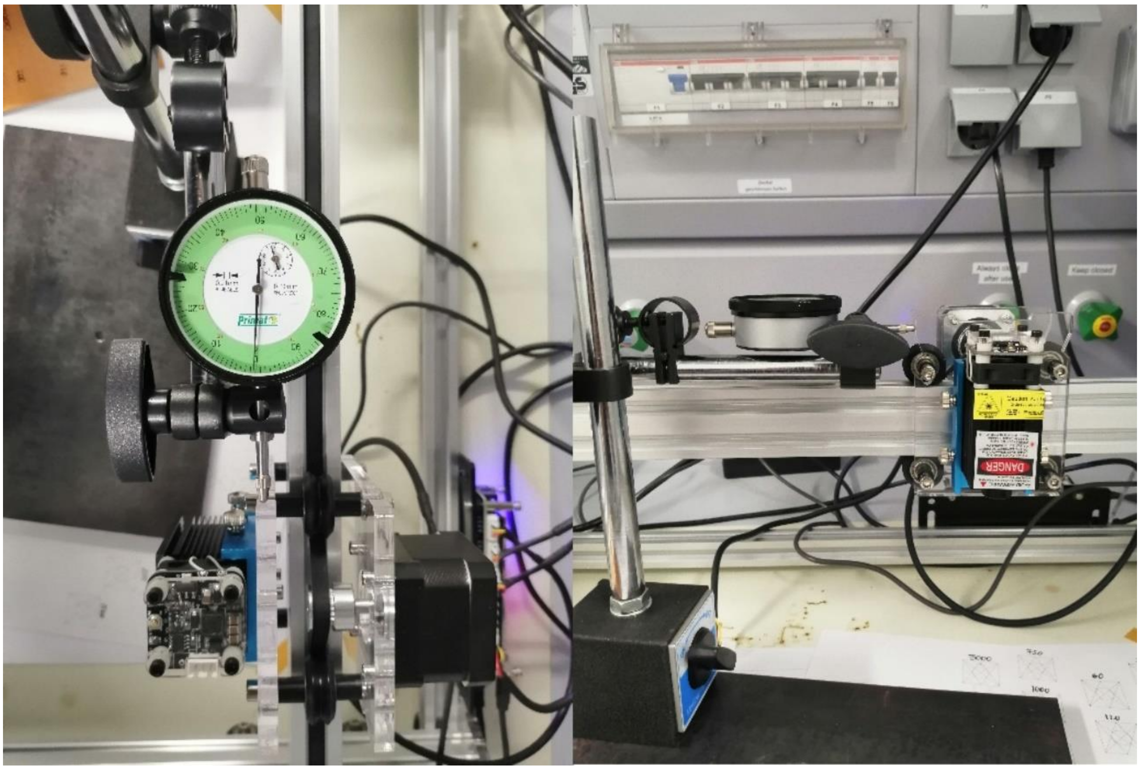


Figure 6.21 – Dial indicator setup for the X-axis

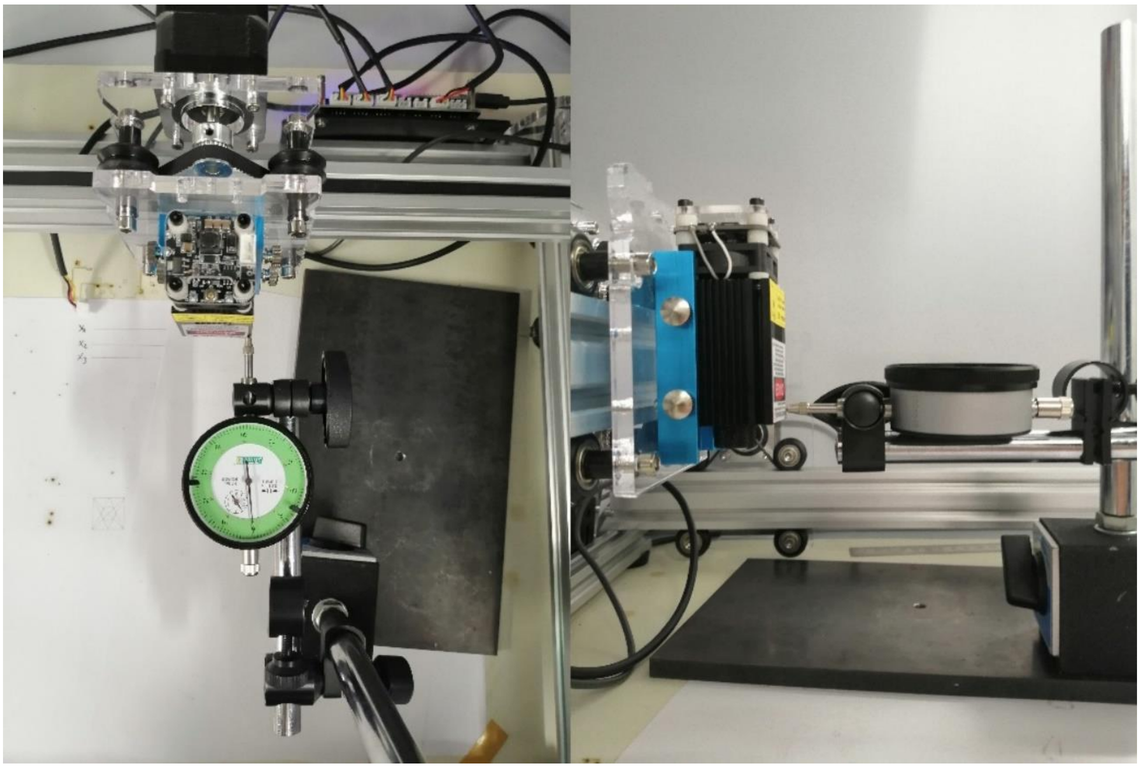


Figure 6.22 – Dial indicator setup for the Y-axis

Besides the way that the dial indicator was positioned, the procedure to check the repeatability of the machine was the same for both axes. The dial was positioned applying slight pressure to the laser module. After proper alignment, the dial's needle was set to zero as shown in Figure 6.23s (a) and (b). The machine then starts a cycle moving forward 1mm and returning to its initial position repeating this cycle 100 times. This procedure was repeated three times at each axis to guarantee that the data was not a fluke but the normal case. It is important to mention that before each cycle the needle was set to zero again. Hence the values displayed in Figure 6.23 (c) to (f) represent the absolute error of each cycle.

The average absolute error was around ±0.01 mm for the X and Y-axis, which is acceptable since it is substantially below the engraving accuracy. Furthermore, this error can also come from bad alignment or due to the presence of a small clearance allowing the parts to slightly move.

(a) Initial needle position (X-axis)

(b) Initial needle position (Y-axis)

(c) Needle position after 1st cycle (X-axis)

(d) Needle position after 1st cycle (Y-axis)

(e) Needle position after 2nd cycle (X-axis)

(f) Needle position after 2nd cycle (Y-axis)

(g) Needle position after 3rd cycle (X-axis)

(h) Needle position after 3rd cycle (Y-axis)

Figure 6.23 - Needle position before and after each axis cycle

6.4.2 Line Thickness

As mentioned in section 2.4.1 a positive photoresist is weakened while exposed to UV light. Section 2.4.2 describes the difference from photomask lithography, which exposes the whole board at once, to the LDI technology, which exposes areas of the PCB punctually. Therefore, for LDI technology, the area weakened depends on the tool diameter, the energy transferred per unit time (laser power), and the exposure duration. The tool diameter, in this case, the laser beam, will limit the minimum line thickness, i.e. a laser beam of 0.1 mm will allow print lines on the board with a thickness equal to or bigger than 0.1 mm. Hence the combination of the laser power and the exposure duration specifies how thicker the line will be.

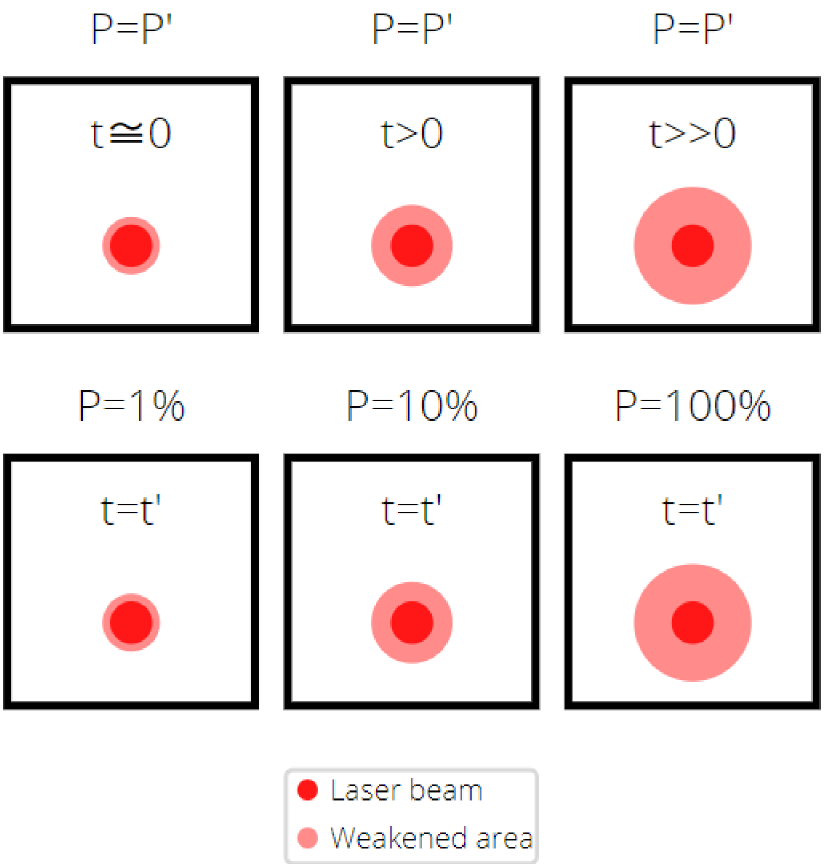


Figure 6.24 – Influence of exposure duration and laser power on the weakened area

A G-code containing multiple squares was created to quantify the influence of those variables. This arrangement was selected to help to track the line thickness of the squares for distinct combinations of feed rate and spindle speed.

The G-code structure has 2 piles, one containing 10 blocks and another containing 5 blocks. Each block represents an increment on the feed rate of 100 mm/min for the first pile, while for the second pile each block represents an increment of 1000 mm/min on the feed rate.

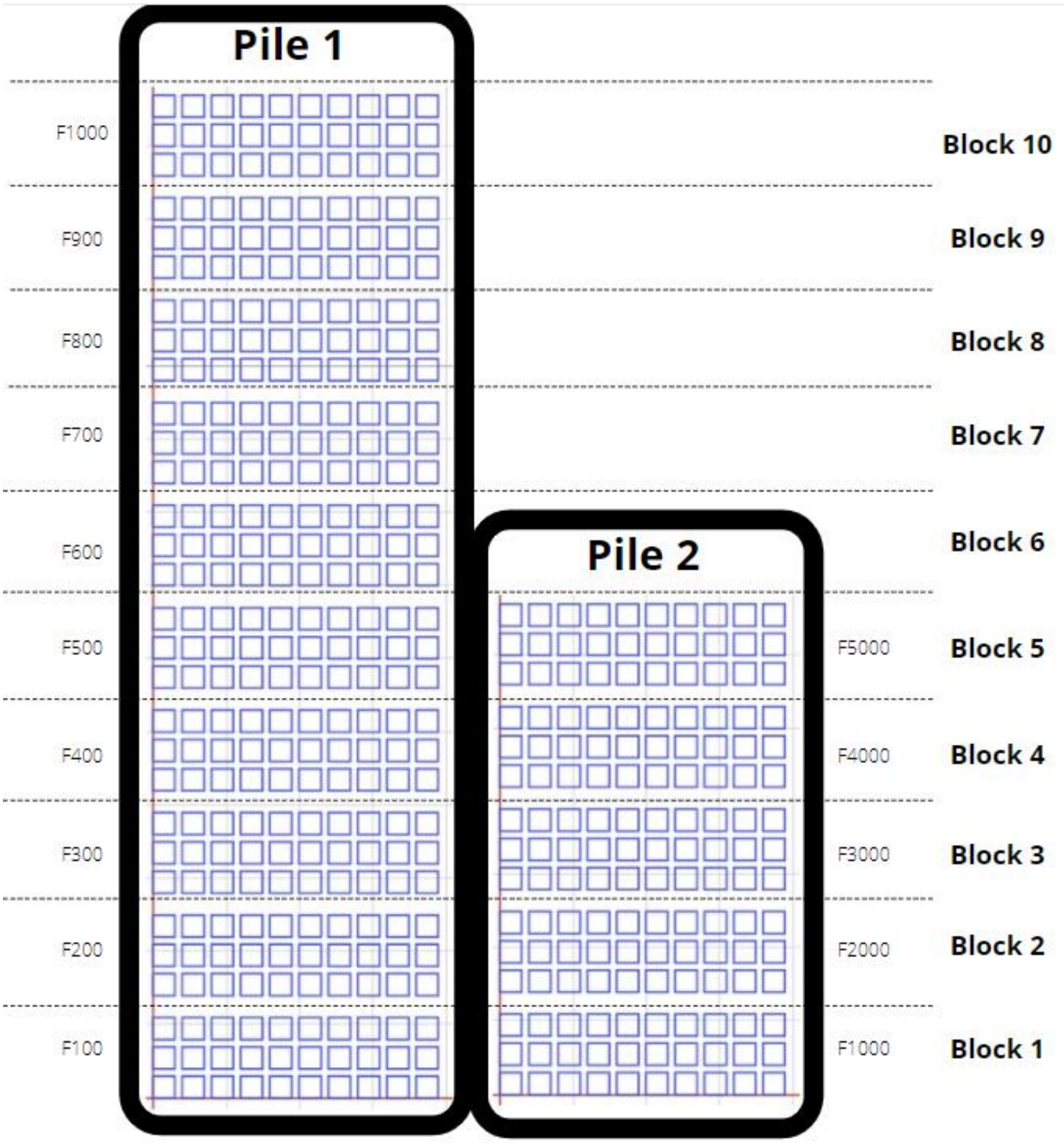


Figure 6.25 – G-code block and pile structure

All the blocks, from both piles, has the same following structure:

- The spindle speed used for each square changes according to the row and to the column that it is located, following the equation:

$$spindle\ speed = column \times 10^{row-1} \tag{2}$$

I.e., a square located at column four and row two will be printed with a spindle speed of 40, or 4% of the laser power.

- All the squares were spaced within a 1 mm gap in all directions.

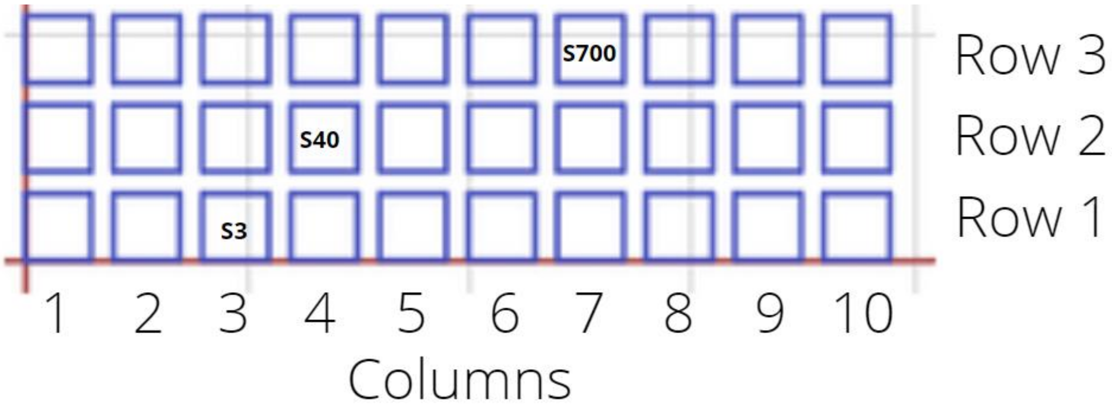


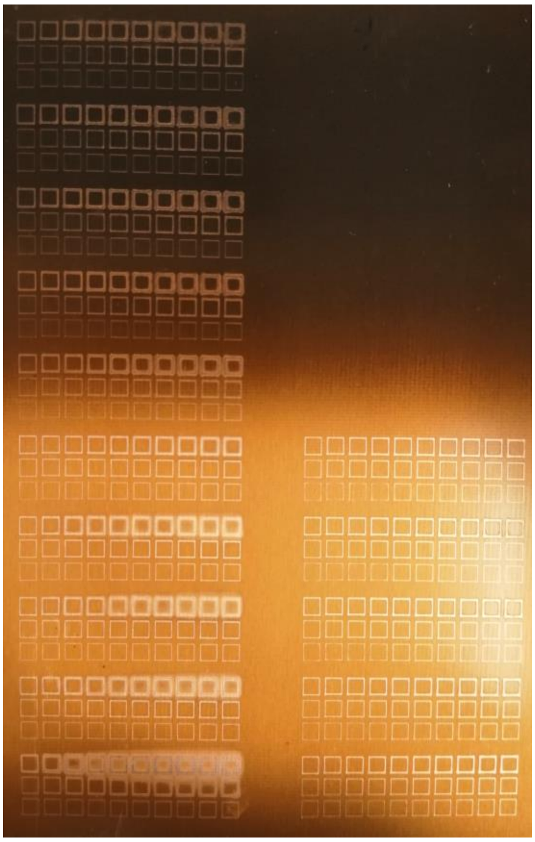
Figure 6.26 – Spindle speed calculation logic

The influence of the spindle speed and the feed rate can be noted even before development (Figure 6.27 (a)). This stage gives the impression that the laser power plays a more important role in the line thickness than the feed rate since all the blocks look similar while there is a clear change in the printed squares when comparing each column or each row.

It is presumed that the squares with clear copper color are areas of the photoresist that was somehow harmed, while the grey color squares are areas where the copper was somehow harmed. It may happen because of the laser beam intensity that etches the photoresist/copper instead of exposing it. Those damages do not affect the final design result since the harmed photoresist will be dissolved during the development process and the harmed copper will be etched at the etching process.

After development (Figure 6.27 (b)), it becomes clear that the feed rate strongly influences the weakened area by comparing the first block of pile 1 (100 mm/min feed rate) with the first block of pile 2 (1000 mm/min feed rate).

(a) Exposed PCB



(b) Developed PCB

Figure 6.27 – Comparison of exposed PCB before and after the development stage

Even that the line thickness becomes clear after development it is still not ideal to measure it at this stage due to the blurred areas and the uncertainty about the region that will be actually etched.

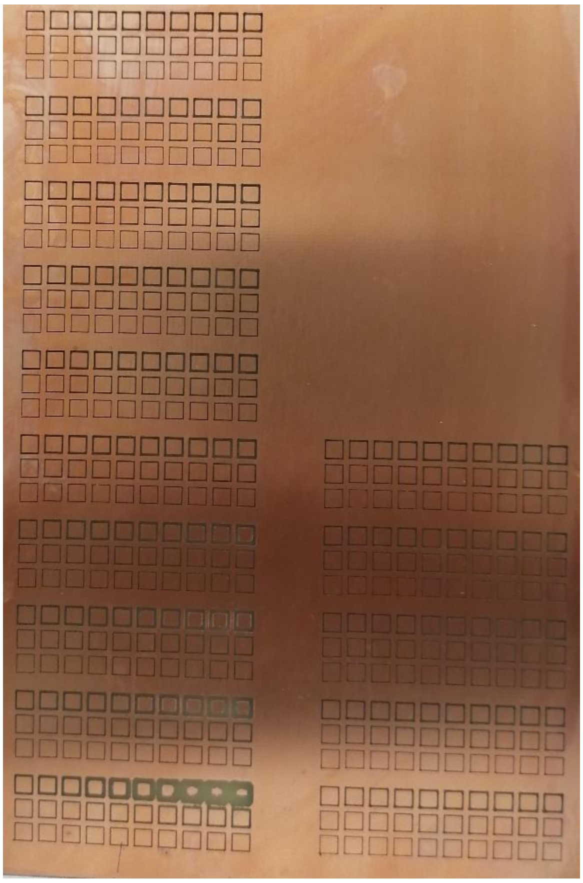


Figure 6.28 – Etched PCB

After etching the line thickness was measured and analyzed with the assistance of a microscope able to amplify the image 200 times. The line thickness measurements were done by taking the average value of the parallel distance between the isolated copper regions for the four sides of the square as shown in Figure 6.30. For the squares with touching sides, as shown in Figure 6.29, it was used the average value for the two not touching sides.

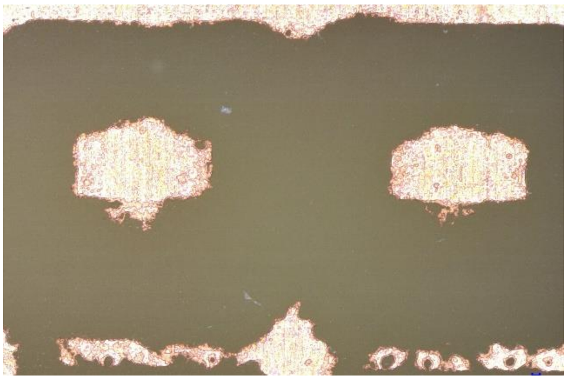


Figure 6.29 - Example of touchline found in pile 1, block 1, row 3

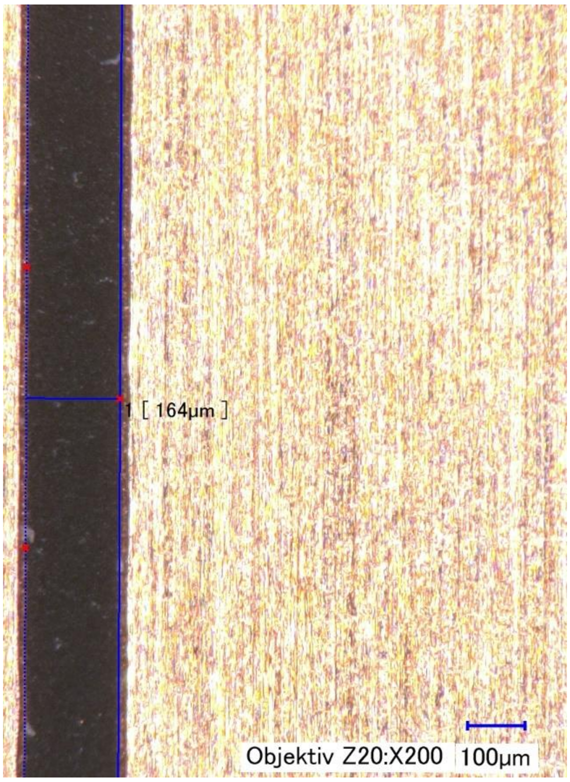


Figure 6.30 – Line thickness measurement

Table 6.3 – Average line thickness in micrometers (Pile 1)

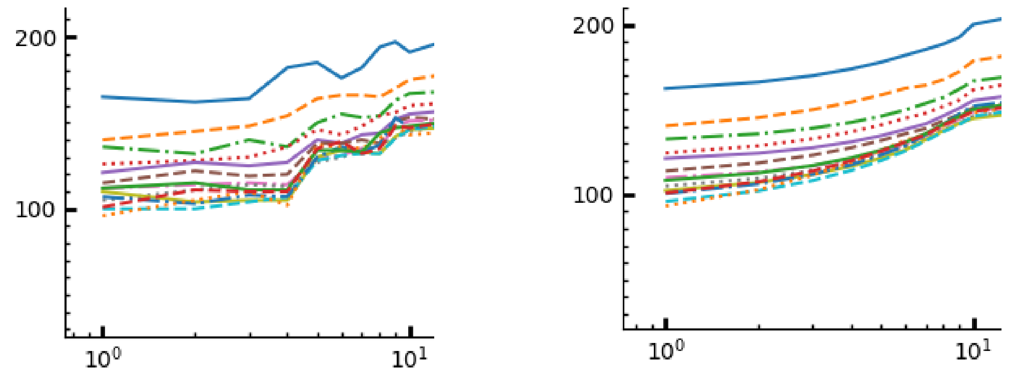
Spindle Speed	F100	F200	F300	F400	F500	F600	F700	F800	F900	F1000
1	165	140	136	126	121	115	112	105	110	100
2	162	145	132	128	127	122	114	111	104	100
3	164	148	140	130	125	119	115	114	105	104
4	182	154	136	136	127	120	114	113	105	107
5	185	164	150	146	140	138	129	127	130	128
6	176	166	155	143	138	138	131	130	134	131
7	182	166	153	148	143	140	135	134	133	132
8	194	165	154	154	144	138	136	138	132	132
9	197	170	163	156	152	151	148	141	142	141
10	191	175	167	160	155	153	151	145	146	146
20	208	183	170	164	160	150	153	154	149	153
30	222	204	183	182	172	167	160	159	159	161
40	236	212	192	188	178	171	169	159	167	168
50	247	220	202	206	188	186	181	177	175	184
60	260	228	208	211	199	188	189	182	185	184
70	272	241	215	215	206	198	191	186	190	187
80	283	246	222	220	206	203	195	188	196	188
90	299	250	230	226	212	202	197	195	195	194
100	312	258	230	229	215	204	205	204	203	200

200	367	308	263	256	235	228	221	215	215	215
300	435	353	318	3002	258	242	239	235	234	222
400	616	384	335	321	271	262	253	246	242	243
500	800	405	370	350	284	278	265	268	252	252
600	935	449	410	369	316	291	269	276	262	263
700	1054	496	431	372	326	307	289	272	271	268
800	1211	558	450	380	338	322	301	286	282	277
900	1357	670	483	402	354	331	303	296	283	279
1000	1502	900	480	434	364	342	317	314	301	284

Table 6.4 – Average line thickness in micrometers (Pile 2)

Spindle Speed	F1000	F2000	F3000	F4000	F5000
1	100	107	96	112	101
2	100	103	105	115	111
3	104	108	110	111	110
4	107	107	102	111	110
5	128	130	137	134	135
6	131	136	136	134	139
7	132	133	135	133	132
8	132	137	141	144	135
9	141	153	148	147	148
10	146	147	143	148	147
20	153	157	147	154	155
30	161	169	167	164	166
40	168	176	175	173	169
50	184	181	181	184	182
60	184	188	187	188	185
70	187	192	195	188	189
80	188	195	202	193	194
90	194	207	192	198	195
100	200	199	196	201	195
200	215	216	215	209	212
300	222	222	233	224	230
400	243	241	237	235	243
500	252	240	247	247	243
600	263	252	250	255	247
700	268	261	258	259	253
800	277	271	268	261	260
900	279	268	273	265	262
1000	284	269	274	265	266

Since the lines engraved to the PCB are sometimes wavy and curvy, the data collected contains some noise. The noise can be clearly seeing in row one of Table 6.3, where the line thickness decrease from F700 to F800, increases from F800 to F900 (but still lower than F700), and then decreases once again from F900 to F1000. Furthermore, considering that the idea of plotting the data is just to support the interpretation of the data, and not to derive intermediate values, filtering the data would not disturb the analysis.



(a) Noise Plot (b) Filtered Plot

Figure 6.31 – Benefits of filtering the data before plotting

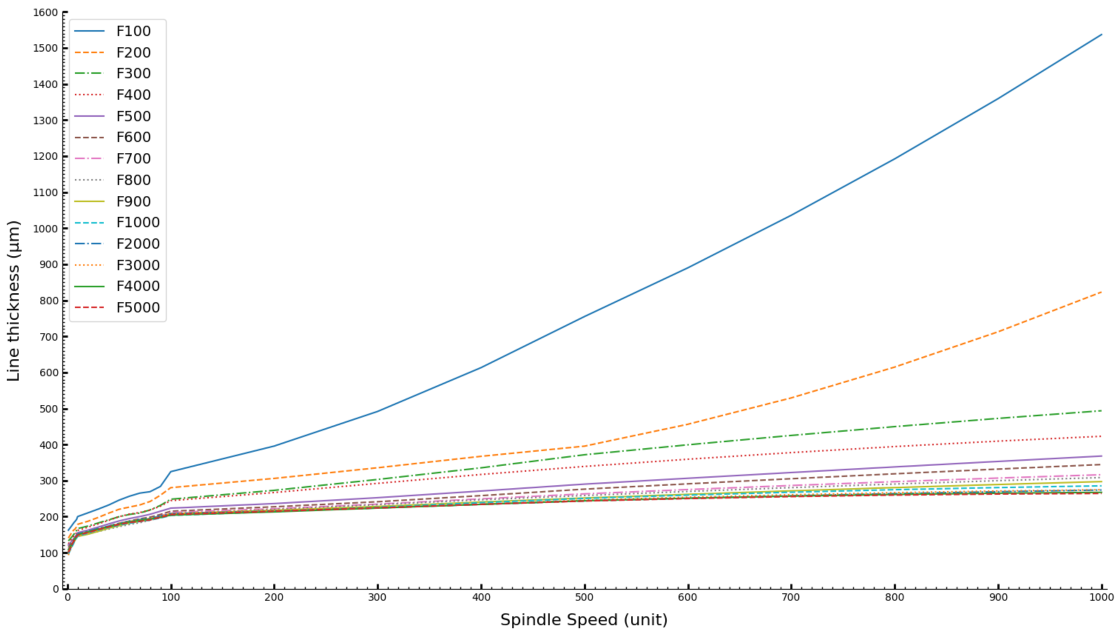
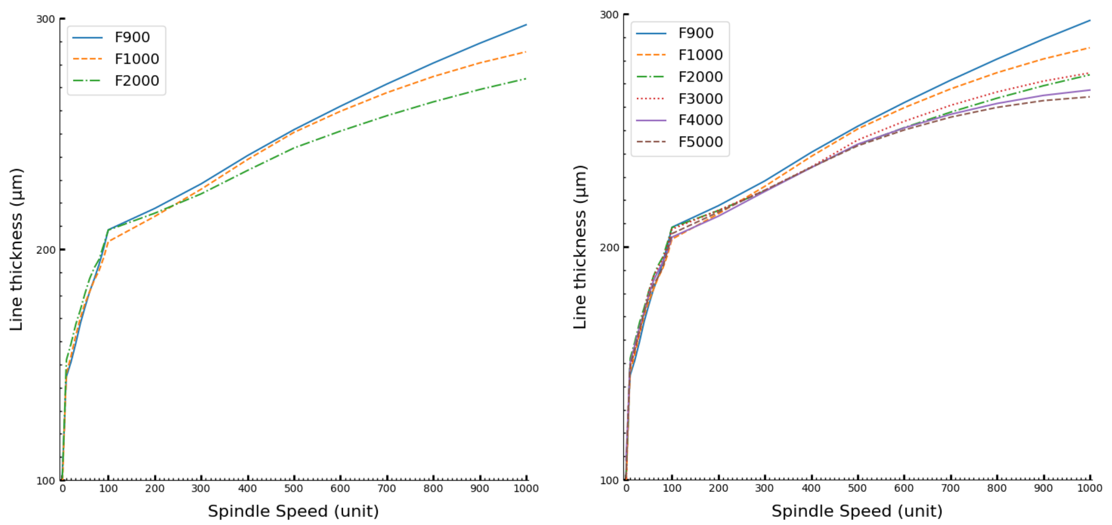


Figure 6.32 – Spindle speed vs line thickness for distinct feed rates

Analyzing the chart, it is possible to conclude the spindle speed and the line thickness have a nearly linear relationship, and the slope of its linear equation is inverse proportional to the feed rate.

Since the machine's engraving accuracy is 0.1 mm, it would be ideal to select a setting that would print a line with a thickness near 0.1 mm to be able to use a diameter tool with the same unit of engraving accuracy. A value close to this desired line thickness can be achieved by any feed rate with spindle speed S1. Therefore, the highest feed rate possible would be the best option for a faster engraving process, since all high feed rates deliver a line thickness close to the hardware's engraving accuracy at low power. Hence, the highest feed rate that the machine can deliver needs to be derived to complement the information collected in section 6.3.5.

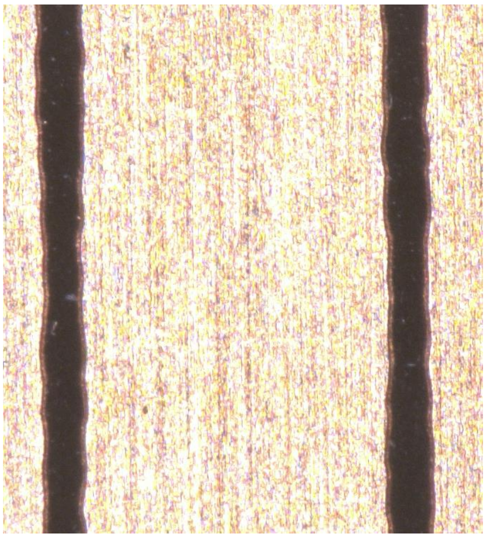


(a) Gap between the F900-F2000 curves (b) Curves F2000-F5000 merging

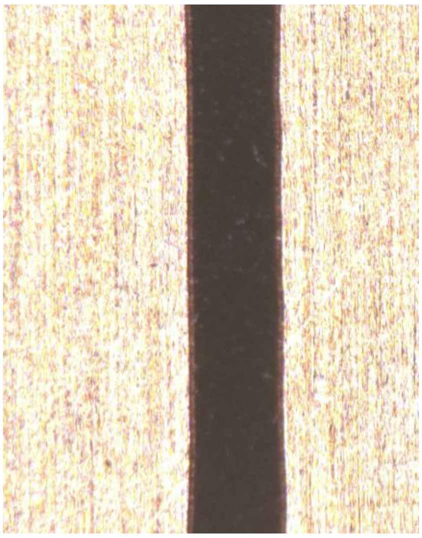
Figure 6.33 – Line thickness values at high feed rates

It is possible to see in the table that the values for F2000 to F5000 are near the same for all spindle speeds. It may indicate that the motor reached its maximum feed rate at 2000 mm/min, therefore any command sent to the hardware indicating the axis to travel at a speed higher than 2000 mm/min would not be delivered since the motor achieved its physical limit delivering just a feed rate close to 2000 mm/min. It can be better perceived in the chart by comparing how the curves F2000 to F5000 cross each other for all the along with all spindle speed values. Therefore, the derived max rate was set to 2000 mm/min.

When the machine is set spindle speed 1 and feed rate 2000 mm/min the printed line has an average thickness of 107 µm, which fits the application. However, the lines printed at low power and high speed are irregular as shown in Figure 6.34. This defect should be avoided since the removal and the multi-passes features may experience not exposed areas due to the irregularity of the lines. Hence, the derived spindle speed value was 10, or 1% of the laser power (50 mW), and the derived feed rate was 2000 mm/min, which deliver a straight line with 147 µm thickness.



(a) Irregular line (F2000 and S1 setting)



(b) Ideal straight line (F2000 and S10 setting)

Figure 6.34 – Example of irregular and straight lines

6.4.3 Design Defects

During the inspection and measurement of the line thickness it was identified some design defects besides the irregular lines:

1. **Unexposed line regions**: Similar to what happened by combining high speed and low power, the combination of high power (S100 or more) and low speed (F100) also presented irregular printing of the lines as shown in Figure 6.35. The area was weakened unevenly, leading to areas that should be completely etched away but contained some copper remains.

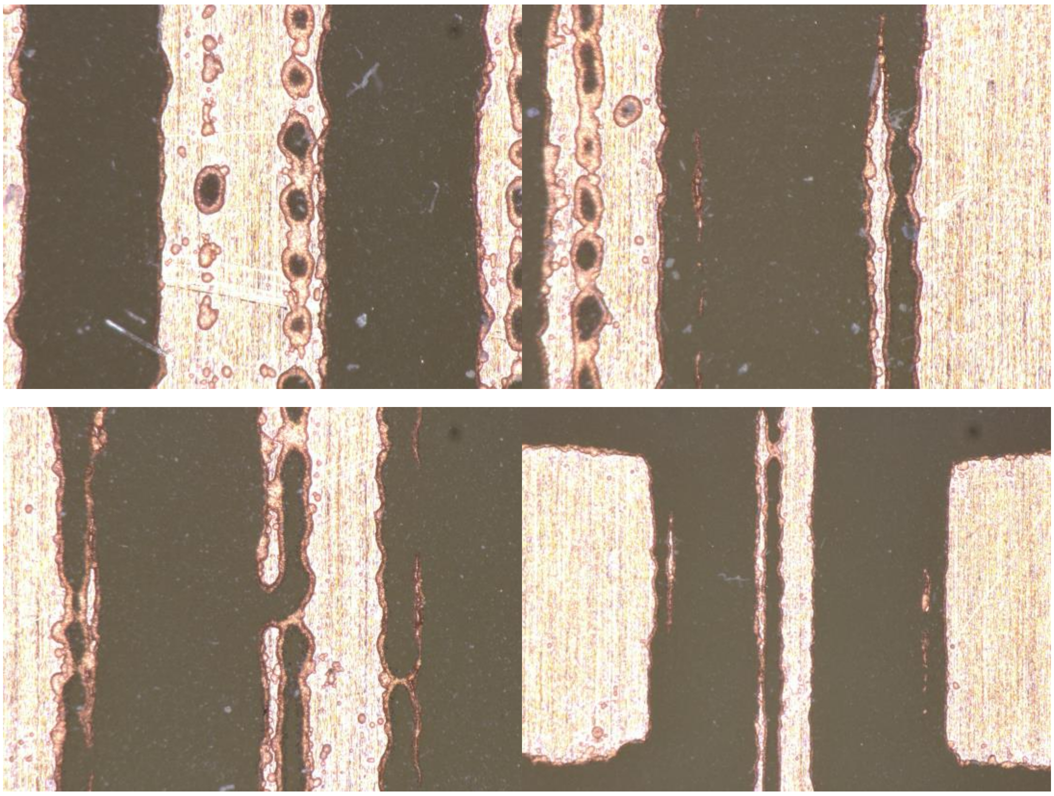
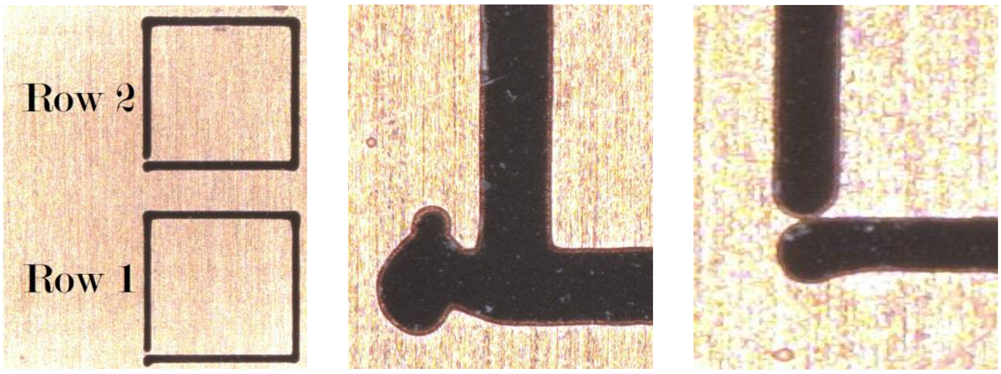


Figure 6.35 – Defects caused by a combination of high power and low speed

2. **Miss alignment**: There was a constant miss alignment defect in the starting corner of some squares located at the first row and first column as shown in Figure 6.36. The same defect was also found once to a square in the first column but in the second row. It probably happens due to the loss of steps in long diagonal movements since this defect appears uniquely while diagonally moving the end effector from the last square of a row to the first square of the next row.



(a) Miss alignment in the first column (row 1 and 2)

(b) Miss positioned starting point in the X direction

(c) Miss positioned starting point in the Y direction

Figure 6.36 – Miss alignment defects

3. **Bridges**: Another identified defect was the presence of bridges. There was no standard identified regarding this defect since it happened at distinct rows, columns, blocks, and sides of the squares. This defect was peculiar since it was probably caused by a sudden stall of the motor or asynchrony at the pulses sent to the laser without clear reasons.

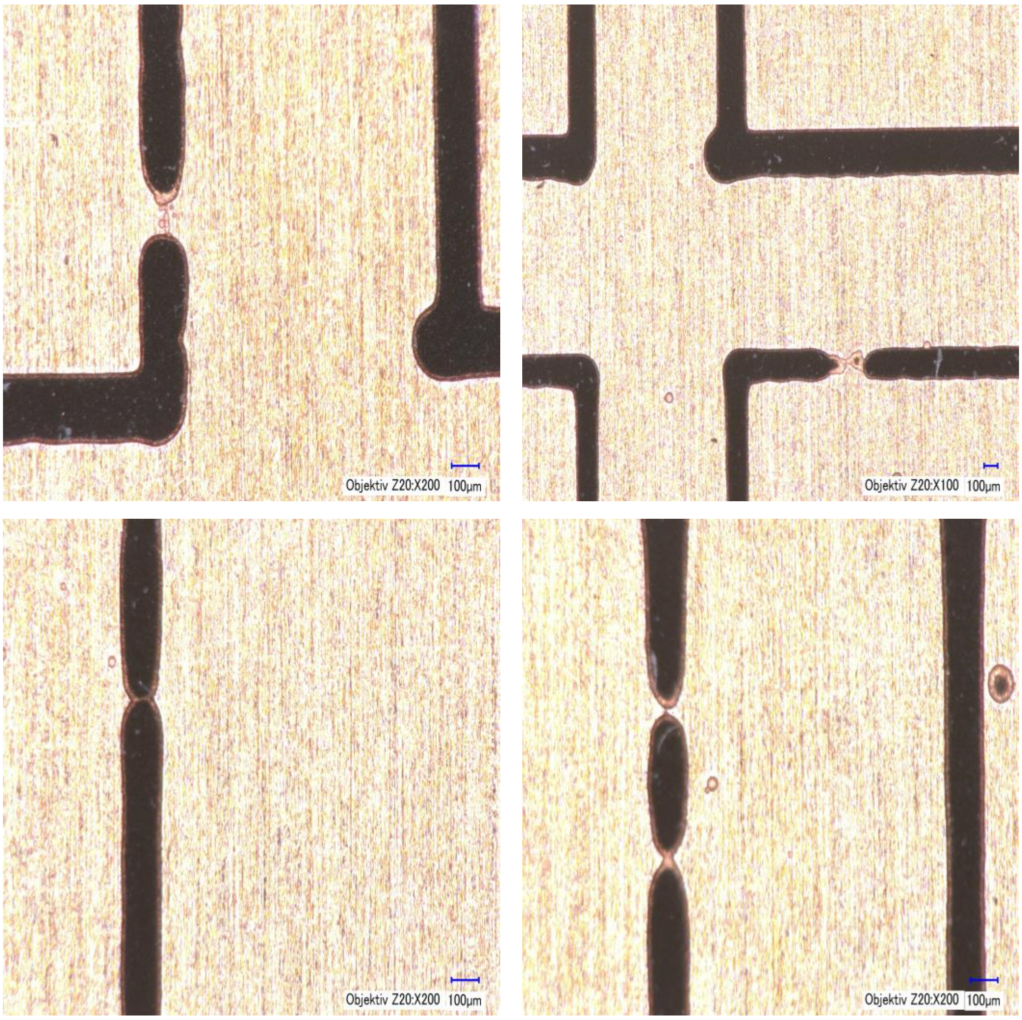


Figure 6.37 – Bridge defect

The unexposed line region defect from Figure 6.35 can be easily avoided by not setting the machine at low speed and high power. The miss alignment and bridge defects from Figure 6.36 and Figure 6.37 can be overcome by setting two isolation passes with an overlap of 90% (avoiding thick lines) instead of one isolation pass. This second pass will guarantee that its starting point coincides with the ending point of the first pass, closing the pattern. In addition, since the path is done twice, it is highly probable that the indiscriminate bridges will be covered due to the low probability of the defect take place in the same location. This solution will result in a long time to print the circuit design but will avoid those two defects.

Another possible solution is to increase the line thickness and diameter tool to 0.2 mm, avoiding working at the engrave accuracy limit, reducing the possibilities of stop loss. It would reduce the resolution of the hardware, but make it more reliable.

It is important to highlight that it is necessary to change the spindle speed from 10 to 100 to be able to print a line with 0.2 mm at 2000 mm/min.

6.5 Experimental Results

The intention of the following experiments is to validate the capacity of the machine to print an authentic PCB design, explore the limits of the hardware, figure out how to overcome printing issues, and what should be done next to improve the hardware and software solution.

The experiments covered in this section were done to test and validate the isolation, removal, and mask features, the grid function, and the capacity of the machine to produce a double-sided PCB.

All the printed designs in this section were developed and etched after engraving to collect results as accurately as possible.

6.5.1 Experimental Setup

During the experiment, all parameters are kept the same for all the attempts, with exception of the diameter tool and the number of passes. The diameter tool was set as 0.1 mm or 0.2 mm, while the number of passes was set as 1 or 2, which brings a total of four setups. Setup 1 uses 0.1 mm tool diameter and 1 pass, setup 2 uses 0.1 mm tool diameter and 2 passes, setup 3 uses 0.2 mm tool diameter and 1 pass, and setup 4 uses 0.2 tool diameter and 2 passes.

It was expected that all setups could deliver a reasonable result at a specific resolution range, but an unreasonable result out of this range. Therefore, the same experiment was done at different setups to derivate which one delivers the best high-resolution PCB designs, which one delivers the best low-resolution PCB designs, and which one should be used for general purposes.

The experiment was split into four stages, an attempt to validate the capacity of manufacturing real PCB designs, an attempt to validate the mirror function, an attempt to derivate the resolution range, and an attempt to make a solder mask application.

6.5.2 PCB Design Validation

This stage of the experiment was dedicated to validating the capacity of the developed solution to print real PCB designs. The same design was printed with the four different setups for the isolation feature and the two different line thickness setups for the removal feature (number of passes is not an input for the removal feature).

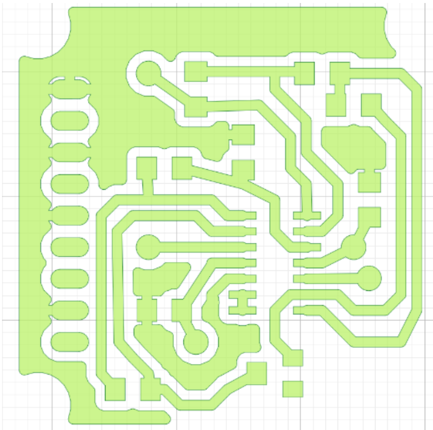
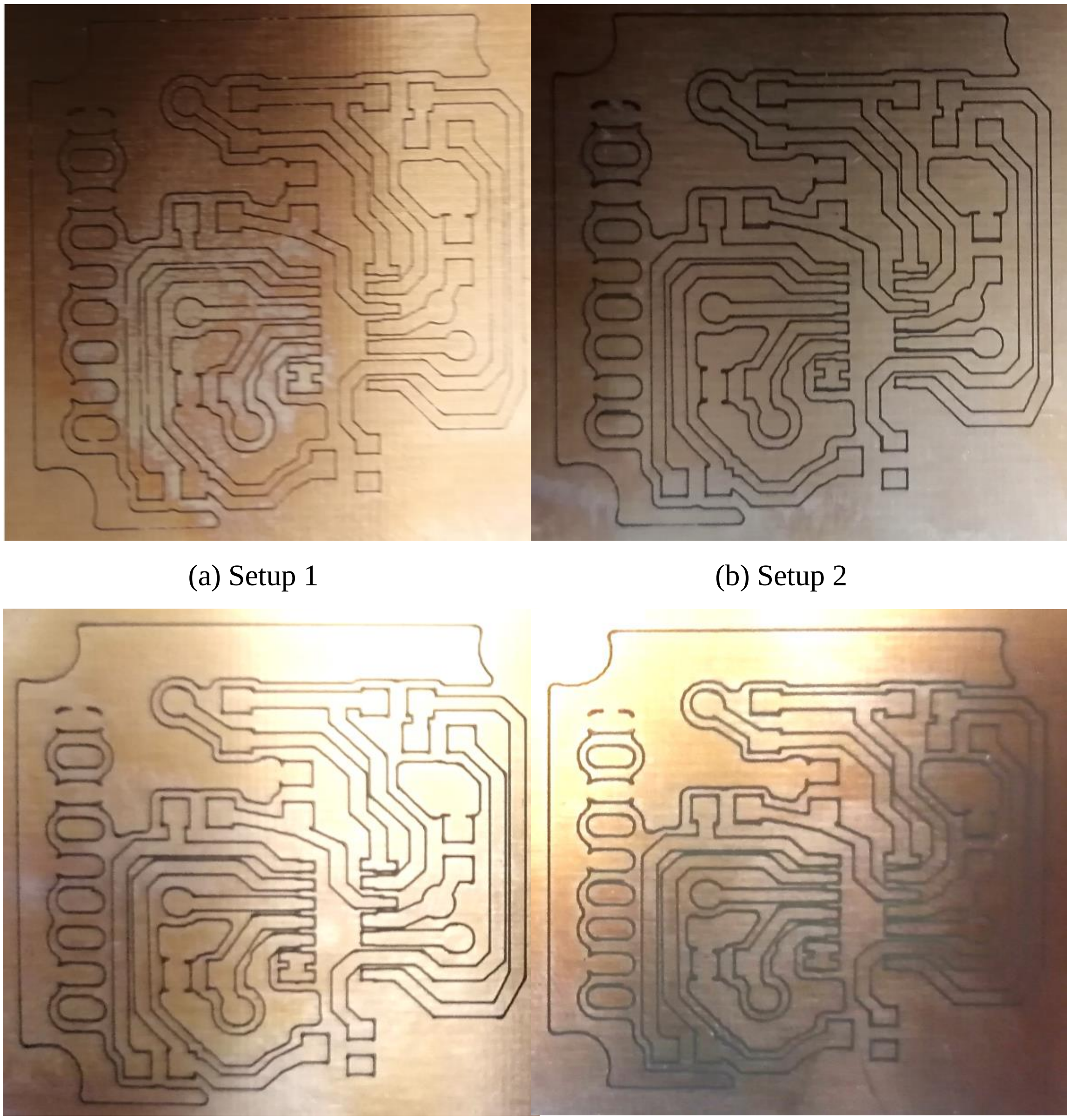


Figure 6.38 – PCB design



(a) Setup 1 (b) Setup 2

(c) Setup 3 (d) Setup 4

Figure 6.39 – Design printed with the isolation feature

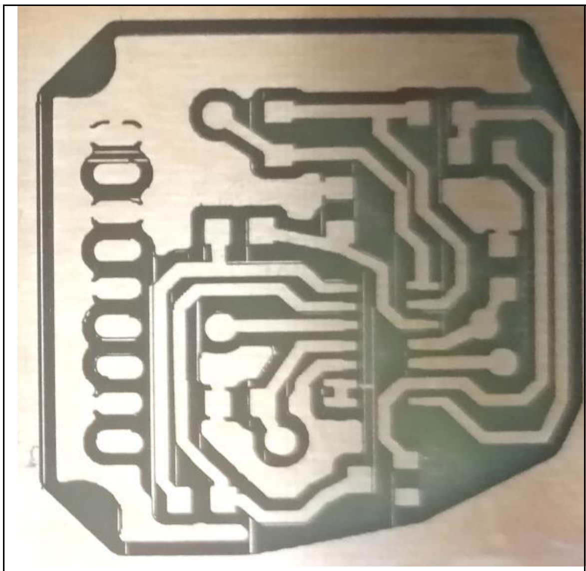
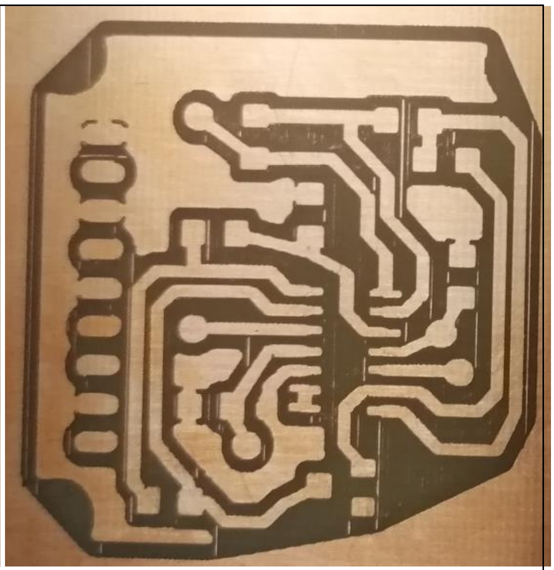
	
(a) 0.1 mm tool diameter	(a) 0.2 mm tool diameter

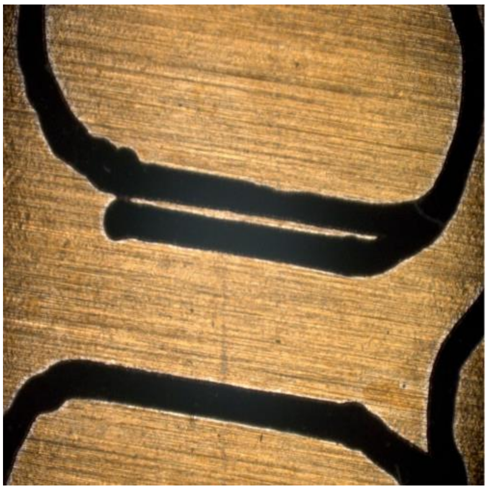
Figure 6.40 – Design printed with the removal feature

Both isolation and removal, features presented expected imperfections due to loss of passes.

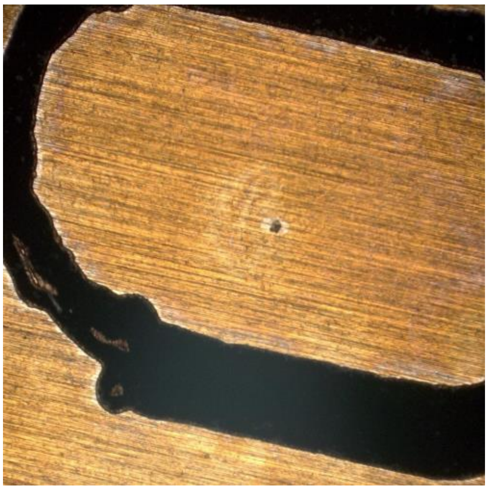
The imperfections present at the PCBs using the isolation feature were similar to the one discussed in section 6.4.3 and were present at all four setups. Those imperfections were minimal and do not compromise the functionality of the PCB, with exception of the one found in the design printed using setup 1 (Figure 6.41 (a)), which cannot be seeing as an imperfection but defect since it compromises the functionality of the PCB.

(a) Setup 1

(b) Setup 3



(c) Setup 2



(d) Setup 4

Figure 6.41 – Printing defects at the isolation feature

As shown in Figure 6.42, the imperfections present in the PCB printed with the removal feature were all minimal and do not affect the functionality of the board. Since the removal feature removes line by line of the non-copper region it was expected the presence of not removed lines due to loss of steps. However, the not removed copper is not connected to the pads and traces since the algorithm used first isolate the desired patterns and then remove the undesired copper, which leads to a disconnection of the erroneous not removed copper lines with the desired pattern.

There was no relevant difference between the PCBs using 0.2 mm or 0.1 mm diameter tool setup, just thinner undesired lines but still present.

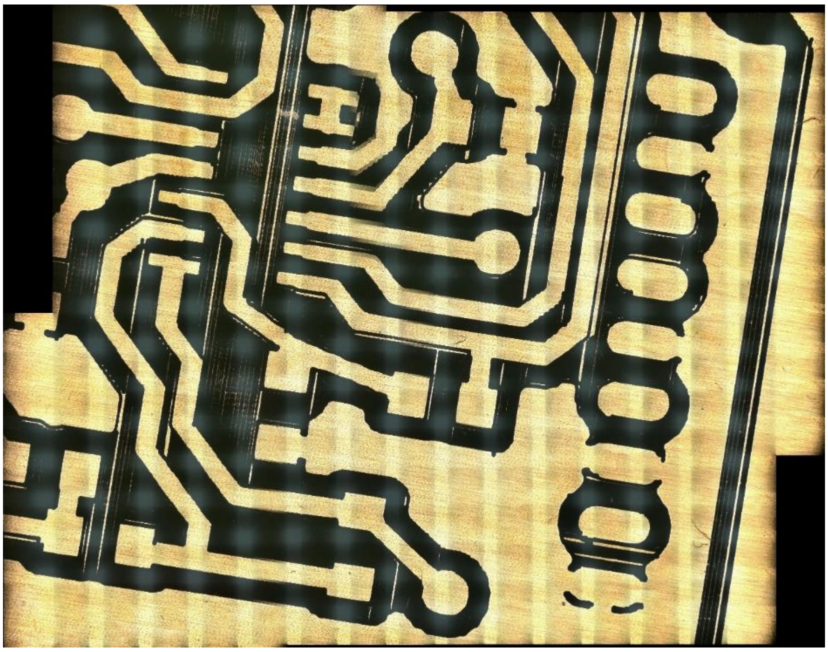


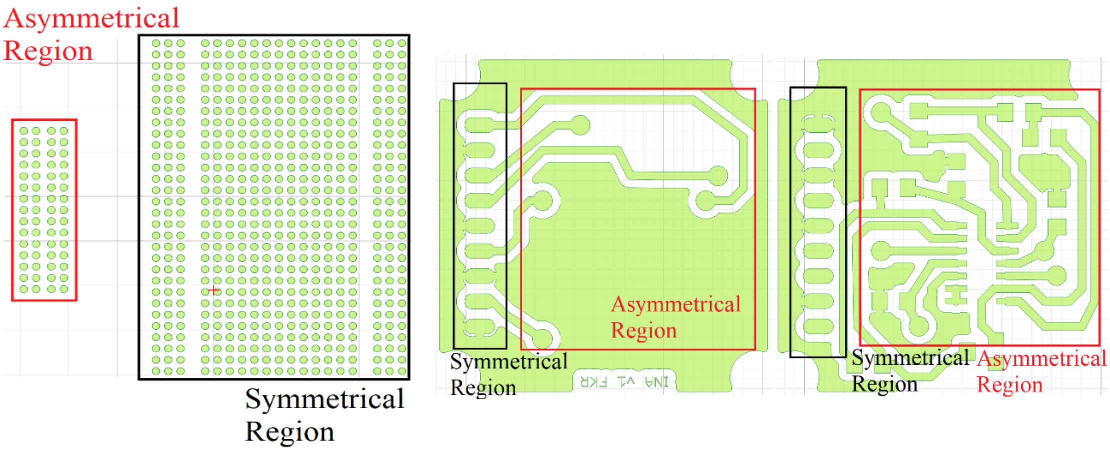
Figure 6.42 – High-resolution image of the PCB printed using the removal feature

This stage of the experiment validates that that is not reliable to work at the highest possible engraving accuracy of the hardware with just one pass. Furthermore, both solutions proposed in section 6.4.3 (setup 2 and setup 3) overcome the previously identified defect in the same section.

6.5.3 Mirror Function Validation

This stage of the experiment was dedicated to validating the capacity of the developed solution to produce double-sided PCB. Unfortunately, the hardware compartment containing the projected pin table was not manufactured at a time to be tested since it was still in the development stage. To overcome this matter both sides of the design were printed at the same side of the production blank to avoid misalignment during the flipping process due to the lack of an accurate positioning tool.

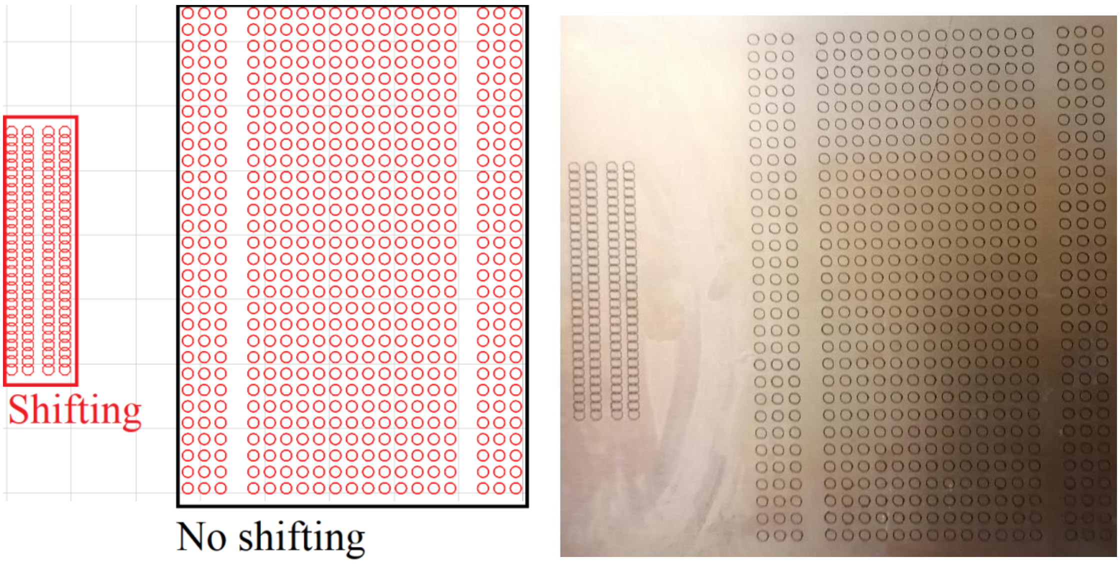
Two design was selected to test the capacity of mirroring with the isolation featuring and the removal featuring. It was relevant that the design contained a region with horizontal symmetry and another region where there was no horizontal symmetry for better visualization of how accurately the symmetrical region was printed after mirroring.



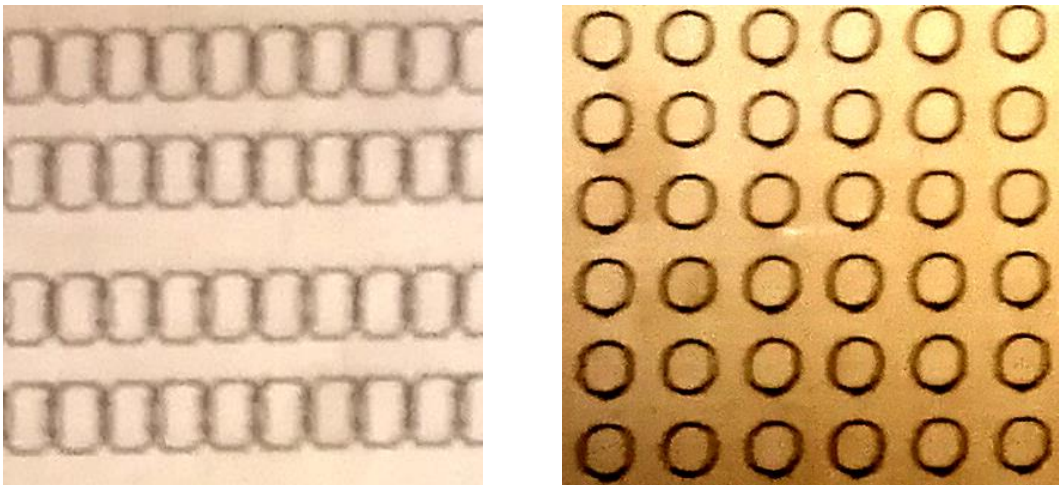
(a) Isolation Design (b) Removal Design

Figure 6.43 – Symmetrical and Asymmetrical regions of the CAD used at the mirror function validation

As expected, this stage of the experiment validates the mirror function by printing the asymmetrical regions with shifting (not overlaying the first print), while ideally printing the symmetrical region by overlaying the first printed layer with a mirrored design.

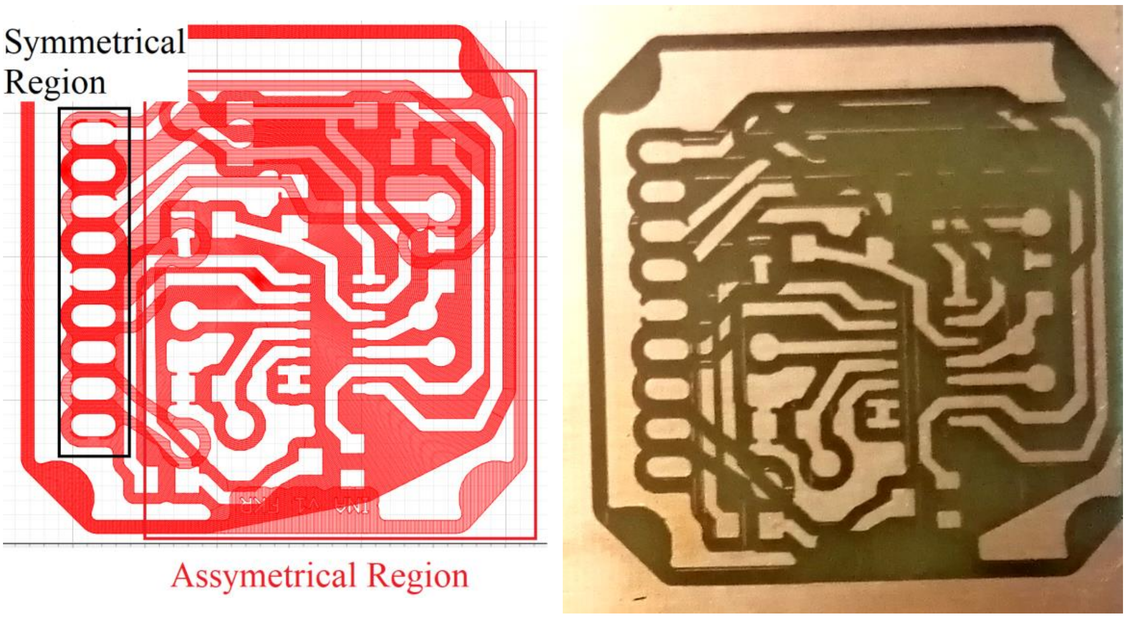


(a) Expected design pattern (b) Printed design pattern



(c) Zoomed asymmetrical region (d) Zoomed symmetrical region

Figure 6.44 – Overlayed isolation design



(a) Expected design pattern (b) Printed design pattern

Figure 6.45 – Overlayed removal design

6.5.4 Resolution Range Derivation

To define the resolution range of the selected setups, a ruler containing diverse components from different sizes was printed with three of the four proposed setups, since setup 1 was already concluded as not reliable in section 6.5.2.

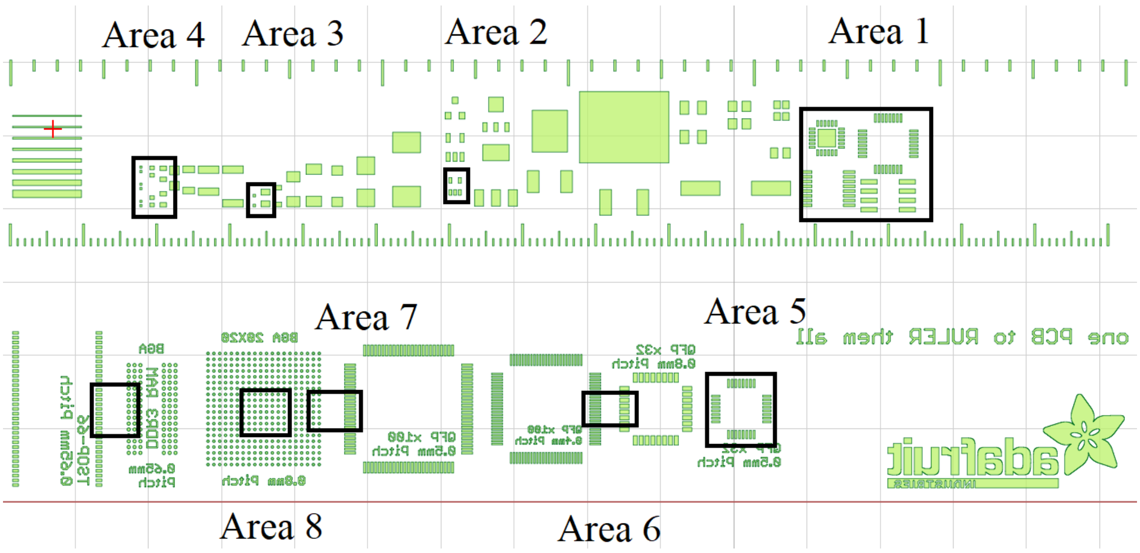


Figure 6.46 – Ruler design and zoomed areas

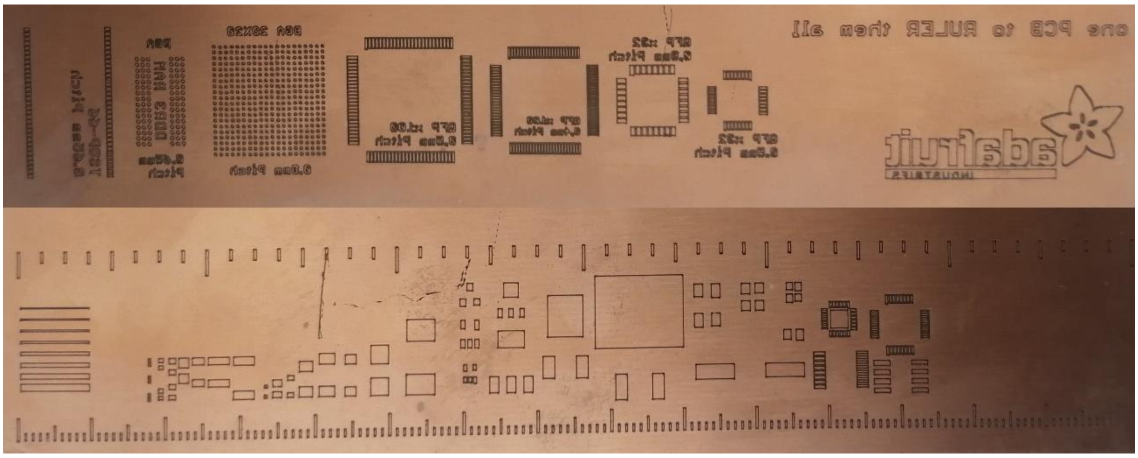
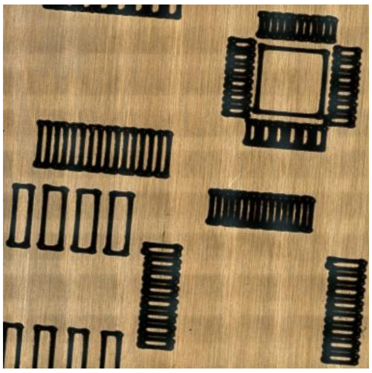


Figure 6.47 – Printed ruler sample (160 mm x 30 mm)

The low-resolution components were printed without relevant imperfections for all three setups. For the High-resolution components, however, it was realized that the setups with two passes did not deliver a reliable result. The comparison of the printing quality for the three different setups can be seen in Figure 6.48.



Area 1 – 0.2 mm diameter tool 1 pass

Area 1 - 0.2 mm diameter tool 2 passes

Area 1 - 0.1 mm diameter tool 2 passes

Area 2 - 0.2 mm diameter tool 1 pass

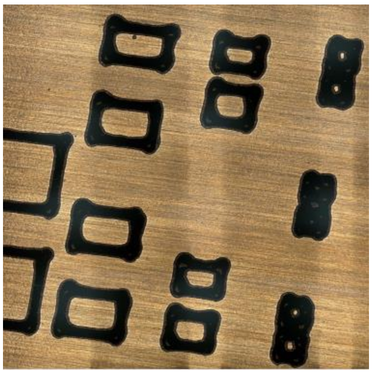
Area 2 - 0.2 mm diameter tool 2 passes

Area 2 - 0.1 mm diameter tool 2 passes

Area 3 - 0.2 mm diameter tool 1 pass

Area 3 - 0.2 mm diameter tool 2 passes

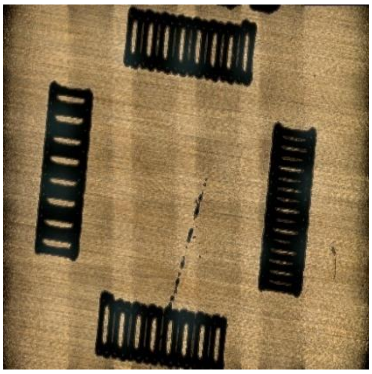
Area 3 - 0.1 mm diameter tool 2 passes



Area 4 - 0.2 mm diameter tool 1 pass

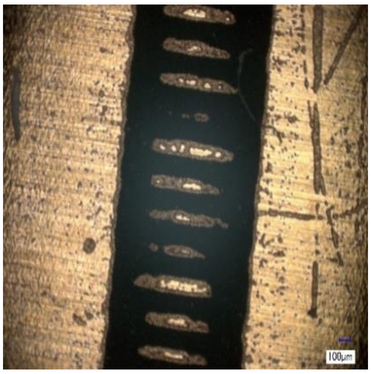
Area 4 - 0.2 mm diameter tool 2 passes

Area 4 - 0.1 mm diameter tool 2 passes

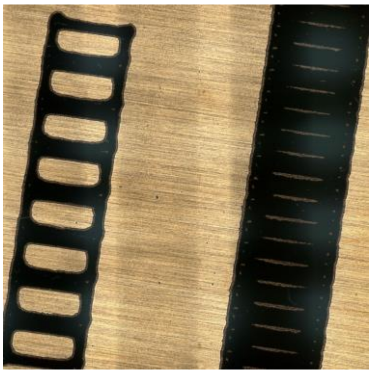


Area 5 - 0.2 mm diameter tool 1 pass

Area 5 - 0.2 mm diameter tool 2 passes



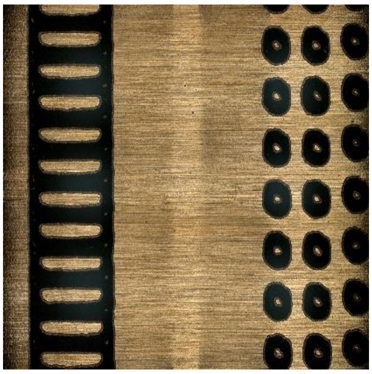
Area 5 - 0.1 mm diameter tool 2 passes



Area 6 - 0.2 mm diameter tool 1 pass

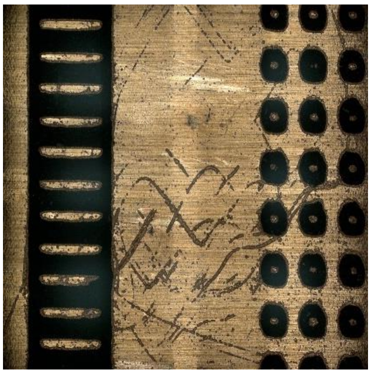
Area 6 - 0.2 mm diameter tool 2 passes

Area 6 - 0.1 mm diameter tool 2 passes

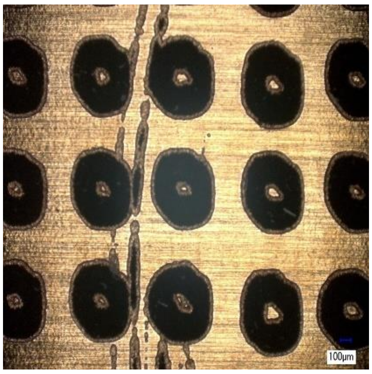


Area 7 - 0.2 mm diameter tool 1 pass

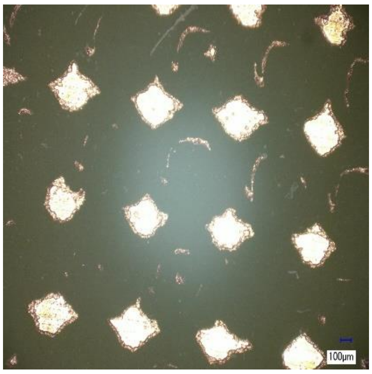
Area 7 - 0.2 mm diameter tool 2 passes



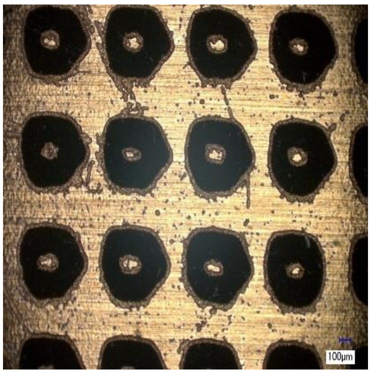
Area 7 - 0.1 mm diameter tool 2 passes



Area 8 - 0.2 mm diameter tool 1 pass



Area 8 - 0.2 mm diameter tool 2 passes



Area 8 - 0.1 mm diameter tool 2 passes

Area 9 - 0.2 mm diameter tool 1 pass

Area 9 - 0.2 mm diameter tool 2 passes

Area 9 - 0.1 mm diameter tool 2 passes

Figure 6.48 – Comparison of the printed design for the three different setups

6.5.5 Solder Mask Application

As mentioned in section 0, photoengraving and solder mask exposure has a similar process. Therefore, it was of interest of this work to transfer the knowledge used to develop an LDI technology for photoengraving to develop an LDI technology for solder mask exposure using the same hardware and software solution.

The development of a solder mask application was tried with two different solder mask media, silkscreened epoxy liquid and dry-film photoimageable solder mask (DFSM). The results were not feasible and satisfactory for silkscreened epoxy liquid, Due to the lack of reputability while applying it.

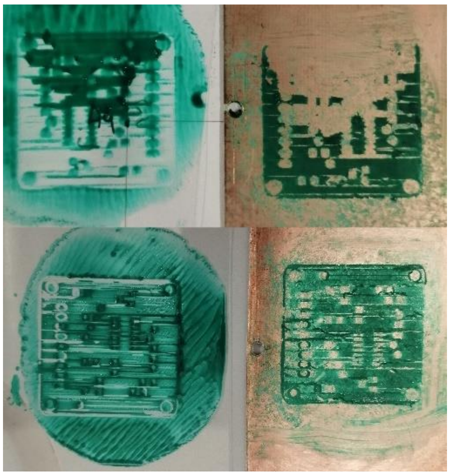


Figure 6.49 – silkscreened epoxy liquid solder mask

With DFSM, however, the results were adequate since its application was repeatable. The same structure used in section 6.4.2 was used to understand how the spindle speed and feed rate influences the line thickness and printing quality of the solder mask.

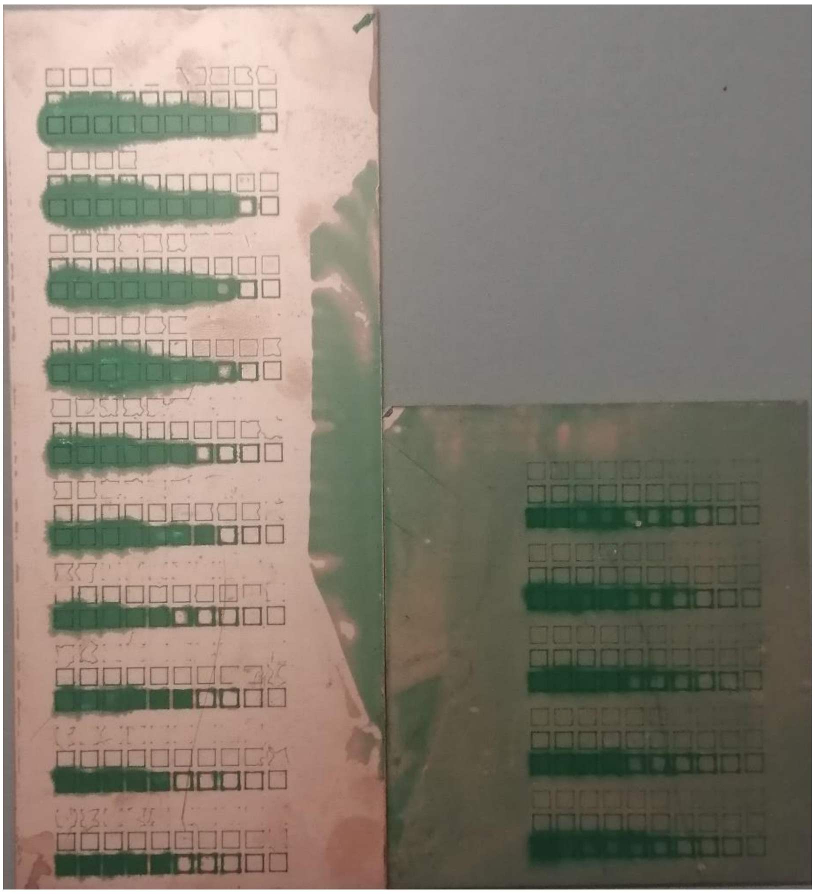


Figure 6.50 - DFSM solder mask printed pattern after development

Similar to the photoengraving process in section 6.4.2, solder mask exposure presents undesired defects at low power and high power for all feed rates. The low power defects happen because the laser beam delivers enough energy to boil the thin plastic film over the solder mask, hence, the small unconnected squares are not pulled out with the rest of the plastic film covering and avoiding the contact of the solder mask with the development solution. At high power, however, the energy not only boils the plastic but also damages the solder mask, hence, this damaged solder mask is erroneously removed with the undesired solder mask. Therefore, the best spindle speed working range for the solder mask printed at 2000 mm/min is between 10 and 100.

It would be ideal if the solder mask application works with the same parameters from the photoengraving, avoiding the necessity of changing the default values always that a solder mask application was necessary. Therefore, it was tried to print a solder mask using the same parameters for the photoengraving (feed rate 2000 mm/min, tool diameter 0.1 mm or 0.2 mm, and spindle speed 10 or 100 according to the tool diameter).

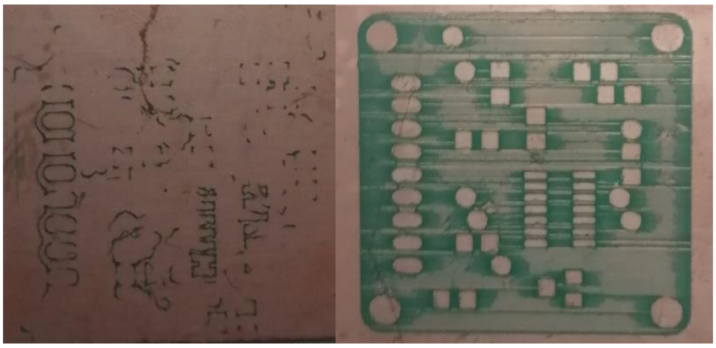


Figure 6.51 – solder mask print with spindle speed 10

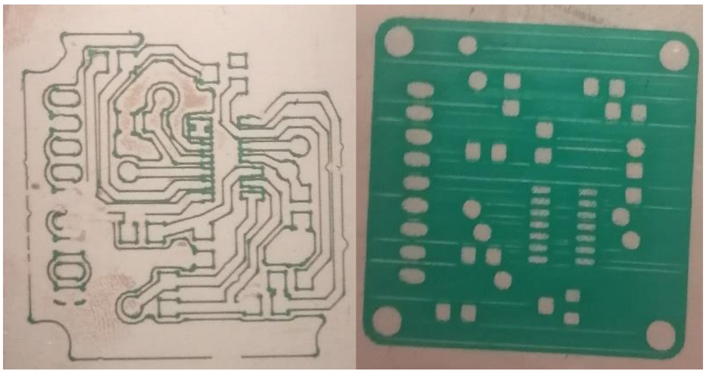


Figure 6.52 – solder mask print with spindle speed 100

The solder mask printed with spindle speed 10 were not as satisfactory as with spindle speed 100, therefore, the ideal spindle speed for solder mask would be 0.2 mm diameter tool and 100 spindle speed.

It is important to mention that the same imperfections while printing using the removal feature discussed in section 6.5.2 were exhibit for the mask feature as well. As for the removal feature, these imperfections do not compromise the functionality of the solder mask.

7 Discussion, Conclusion and Future Work

From the four proposed setups, three were able to reliably deliver a functional PCB. Setup 2 is more reliable for low-resolution PCBs than setup 3 since it guarantees that the patterns will be closed and the miss-aligned between the passes do not affect the printing quality. Setup 3 is more reliable for high-resolution PCBs than the previous setup since there is no miss alignment between passes that causes the etching of desired copper. Setup 4 delivers a feasible result but brings the pitfalls from the two previous setups together, hence, this solution has no advantages over them.

Setup 3 was able to deliver low-resolution PCBs with as much quality as setup 2. Furthermore, it has the same parameters as the optimal solder mask parameters. Therefore, setup 3 (0.2 mm diameter tool, 1 pass, and 100 spindle speed) was defined as the default setup.

At all, the experiments were able to validate all futures proposed by the software, show the hardware limits, bring solutions for the printing issues, understand the best setup ranges, and highlight further work to be done to improve the solution purposed in this work.

Regarding to time, it took 1 min to print the design in Figure 6.38 with the isolation feature using a single pass, 3.5 min to print it with the isolation feature using two passes, and 11 min using the removal feature. It took 8.5 min to print one side of the ruler design in Figure 6.46 and 15 min to print the other side using a single pass, while it took 38.5 min to print one side and 21 min to print the other side with two passes.

7.1 Conclusion

As proposed, the LDI solution delivers to the university's electronics workshop an alternative for the actual photoengraving process and the possibility to do solder mask application without changing the previous input, without adding any intermediary step in the process, and removing the mask printing step. Furthermore, no extensive training is necessary to use this tool since its usability was carefully designed to bring a user experience as smooth as possible with self-explanatory features.

As explained in section 0, time varies according to the feature and PCB complexity. Some experiments took less than 5 min to print both sides while some others took almost 1 h to print both sides. However, after starting the process, there was no necessity for the user to stay close monitoring it. The solution runs fully autonomously allowing the user to focus on other tasks as proposed.

This work was also able to answer which and how the physical limitations could influence the printing design and how configuration parameters influence the quality of the printing and the thickness of the line, showing the path and allowing the solution to evolve.

7.2 Future Work

While this work has demonstrated the potential of the LDI solution developed for the University’s laboratory, there is potential to improve the hardware and software solution even further. Some opportunities that need further research are presented in this section.

Transmission System

The transmission system of the hardware exhibit some imperfection due to step loss in basically all designs. The hardware used in this thesis was intended to be a prototype, and therefore, its transmission system does not perfectly fit the application. It would be relevant to develop a CNC router thinking on this application and its resolution requirements to allow the solution to print PCBs with less or no imperfection.

Homing function

Since the case of the hardware was still in development solution, there was not possible to test and configure a homing function using limit switches. However, the software has a space dedicated to this function, and once that the case gets delivered the homing function can be tested and improved in case of unexpected behavior.

Calibration

After running the hardware several times, it may become less accurate, which is not a specific characteristic from this solution but from all CNC machines. Therefore, it is important to offer the solution some sort of calibration system, either manually or automatically.

The software developed in this work does not have a button or function for calibration but it is recommended to di implementation directly in the software with a button that would call a calibration screen where the user would be able to calibrate the hardware.

Tool exchanging

A pertinent advantage for PCB milling is the possibility to set different tool diameters for different component resolutions in the same board. For laser machines, however, there is no real necessity of changing tools. To do so, it is just necessary to modify the spindle speed parameter used to change the line thickness. Hence, it would be interesting to develop an application that could inform the machine of different S-words according to the resolution of the component that is going to be printed.

Laser module

It was interesting for this work to have a PMW laser to be able to work with a power range and derivate the one that best fits the application. Once that is already defined it would be better to use a laser with a maximum power of 50mW (standard power derived in this work) without PMW to assure that the laser will not print a line by printing a sequence of dots, but by constant feeding. Furthermore, use a 50 mW laser instead of a 500 mW laser increase the safety of the tool in case that, by mistake, the laser uses its maximum power.

Computer vision

An alternative to the pin table solution for the alignment problem would be a system to locate the production blank/PCB with a camera. The advantage of the usage of computer vision for alignment is that PCBs manufactured externally can be used in the solution for further manufacturing or/and solder mask application. In addition, the usage of computer vision for alignment removes the drilling of alignment holes as a preparation step, since the position and the alignment of the PCB will be inputted automatically.

It is important to mention that a system using computer vision as an alignment solution will need to use a rotation matrix to rotate the G-code coordinates (global axes) to local axes since the vertical and horizontal side of the PCB will probably not be aligned with the vertical and horizontal axis of the machine.

References

[1] A. Harper, Charles. Electronic materials and processes handbook. 4th ed. McGraw-Hill: 2003.

[2] Mitzner, Kraig, Bob Doe, Alexander Akulin, Anton Suponin, und Dirk Müller. Complete PCB Design Using OrCAD Capture and PCB Editor. 2. Aufl. San Diego: Elsevier Science & Technology, 2019.

[3] Mitzner, Kraig. Complete PCB Design Using Orcad Capture and Layout. Oxford: Elsevier Science & Technology, 2007.

[4] Davim, J. Paulo. Additive and Subtractive Manufacturing, Emergent Technologies. 4th ed. De Gruyter: Berlin/Boston, 2020.

[5] F. Coombs Jr., Clyde. Printed Circuits Handbook. 6th ed. McGraw-Hill: 2008

[6] Brooks, Frederick Vincent. Lithography, In Chisholm, Hugh Encyclopædia Britannica. 16. 11th ed. Cambridge University Press: 1911.

[7] Robinson, Alex, and Richard Lawson. Materials and Processes for Next Generation Lithography. Elsevier: London, 2016.

[8] Mack, Chris. Fundamental Principles of Optical Lithography: The Science of Microfabrication. 2nd ed. Aufl. Newy York: John Wiley & Sons, Incorporated, 2007.

[9] B. Barclay, and M. Morrell. Laser Direct Imaging – A User Perspective. Leiterplatten magazine. 7-8, 2001.

[10] R. S. Khandpur. Printed circuit boards: design, fabrication, assembly and testing. Tata-McGraw Hill: 2005.

[11] Zumbahlen, Hank and Analog Devices Inc. Engineeri. Linear Circuit Design Handbook. 1st ed. Analog Devices: 2008.

[12] C. Coombs. Printed Circuits Handbook. 7th ed. McGraw-Hill: 2016.

[13] Iordache, Octavian. Evolvable Designs of Experiments: Applications for Circuits. 1st ed. Wiley-VCH. 2009.

[14] GenRad. Introduction to In-Circuit Testing. 1st ed. GenRad, Inc: Massachusetts, 1984.

[15] Cadence PCB Solutions: What Is Bare Board Testing?, in: https://resources.pcb.cadence.com/blog/what-is-bare-board-testing-2, 16.03.2021

[16] Zhang, Dan, and Bin Wei. Mechatronics and Robotics Engineering for Advanced and Intelligent Manufacturing. Cham: Springer International Publishing, Imprint: Springer, 2017.

[17] Taylor, Nick. Laser: The Inventor, The Nobel Laureate, and The Thirty-Year Patent War. Simon & Schuster: 2000.

[18] Fritz Kurt Kneubühl, Markus Werner Sigrist. Laser. 7th ed. Springer-Verlag: 2015.

[19] Einstein, Albert. Zur Quantentheorie der Strahlung. Physikalische Zeitschrift. 18: 1917.

[20] Steen, W.M. Laser Materials Processing. 2nd Ed. 1998.

[21] The Nobel Prize in Physics 1966 Presentation Speech by Professor Ivar Waller. Retrieved January 1, 2007.

[22] Pedrotti, Pedrotti, Bausch, Schmidt. Optik für Ingeneure. Heidelberg: Springer, 2002.

[23] Sanz-Medel, Alfredo, and Pereiro, Rosario. Atomic Absorption Spectrometry : An Introduction, 2nd edition. New York: Momentum Press, 2014.

[24] Singh, S. C., Zeng, H. B., Guo, Chunlei, and Cai, Weiping, eds. Nanomaterials : Processing and Characterization with Lasers : Processing and Characterization with Lasers. Somerset: John Wiley & Sons, Incorporated, 2012.

[25] Svelto, Orazio. Principles of lasers. 5. ed. New York [u.a.]: Springer, 2010.

[26] Coldren, Larry A., Scott W. Corzine, und Milan L. Mashanovitch. Diode Lasers and Photonic Integrated Circuits. 2nd ed. Aufl. Hoboken: John Wiley & Sons, Incorporated, 2012.

[27] Steeds, D. A history of machine tools 1700-1910. 1st ed. Oxford University Press: 1969.

[28] Albert, Alain and FPInnovations – Forintek Division. Understanding CNC Routers. 1st ed. 2010.

[29] Arduino Handbuch für Einsteiger. Landshut: BMU Verlag, 2017.

[30] Arduino’s official store, in: https://store.arduino.cc/arduino-uno-rev3, 12.03.2021

[31] Grimmett, Richard. Arduino Robotic Projects. Olton: Packt Publishing, Limited, 2014. Accessed March 12, 2021.

[32] Arduino’s official store, in: https://store.arduino.cc/arduino-nano, 12.03.2021

[33] Dennis, Andrew K.. Raspberry Pi Home Automation with Arduino. Olton: Packt Publishing, Limited, 2013

[34] Dembowski, Klaus. Raspberry Pi - Das technische Handbuch: Konfiguration, Hardware, Applikationserstellung. 2., erw. und überarb. Aufl. 2015. Wiesbaden: Springer Fachmedien Wiesbaden, 2015.

[35] Molloy, Derek. Exploring Raspberry Pi : Interfacing to the Real World with Embedded Linux : Interfacing to the Real World with Embedded Linux, John Wiley & Sons, Incorporated, 2016.

[36] Kaminski, Steffan, Python 3. München, Wien:: De Gruyter Oldenbourg, 2016.

[37] Igual, Laura, und Santi Seguí. Introduction to Data Science: A Python Approach to Concepts, Techniques and Applications. Cham: Springer International Publishing, Imprint: Springer, 2017.

[38] Tavanier, Karel: Open Letter to the Gerber User Community, in: Ucamco. 2014. Pages 2 and 3.

[39] Tavanier, Karel: Homage to the Gerber Format, in: https://www.ucamco.com/en/news/homage-to-the-gerber-format, 14.03.2021

[40] Borges, Malte, Lars Führmann, Arnold Wiemers, Wojciech Wozny, und LPKF. Leiterplatten-Prototyping. 1. Aufl. Würzburg: Vogel Communications Group, 2015.

[41] Jurke, Benjamin: First steps in G-code: An introduction to the G-code programming language for CNC applications, in: https://benjaminjurke.com/content/articles/2015/gcode-primer/, 15.03.2021

[42] Kumar, Kaushik, Chikesh Ranjan, und J. Paulo Davim. CNC Programming for Machining. 1st ed. 2020. Cham: Springer International Publishing, Imprint: Springer, 2020.

[43] FlatCam: Welcome to FlatCAM's documentation!: 3. Basics, in: http://flatcam.org/manual/index.html, 14.03.2021

[44] gnea's Github repository: grbl, in: https://github.com/gnea/grbl, 15.03.2021

[45] Moggridge, B. Designing Interactions. The MIT Press, 2007

[46] Houde, S. and Hill, C. What do Prototypes Prototype? In M. Helander, T.K. Landauer and P. Prabhu, eds., Handbook of Human-Computer Interaction. Elsevier Science BV, 1997.

[47] Lim, Y., Stolterman, E., and Tenenberg, J. The anatomy of prototypes: Prototypes as filters, prototypes as manifestations of design ideas. ACM Trans. Comput.-Hum. Interact. 15, 2 (2008), 1-27.

[48] Hartmann, Björn. Doctor dissertation. Stanford University: California. Dissertation in partial fulfillment of the requirements for the degree of doctor of philosophy. 2009

[49] Real.de: EleksMaker® EleksLaser-A3, in: https://www.real.de/product/344575344/?kwd=&source=pla&sid=29308881&gclid=Cj0KCQiA0-6ABhDMARIsAFVdQv-EyGNdjHEliNY2nAouxWbj5X07IsJEuguUexv_Pv9Jpp1EAfD_ySQaAsa0EALw_wcB, 04.02.2021

[50] gnea's Github repository wiki: Grbl v1.1 Laser Mode, in: https://github.com/gnea/grbl/wiki/Grbl-v1.1-Laser-Mode, 15.03.2021

[51] EleksMaker: EleksLaser-A3 Pro Specifications, in: http://elekslaser.eleksmaker.com/, 05.04.2021

[52] Joint Committee for Guides in Metrology. Evaluation of measurement data – Guide to the expression of uncertainty in measurement. 1st ed. JCGM: 2008.