KONKURS NA NAJLEPSZE PRACE DYPLOMOWE WIMiR

VI Konkurs Na Najlepsze Prace Dyplomowe Wydziału Inżynierii Mechanicznej i Robotyki 2016/2017 – prace inżynierskie

Patryk Paryło

Mechatronic Engineering

2D plotter design

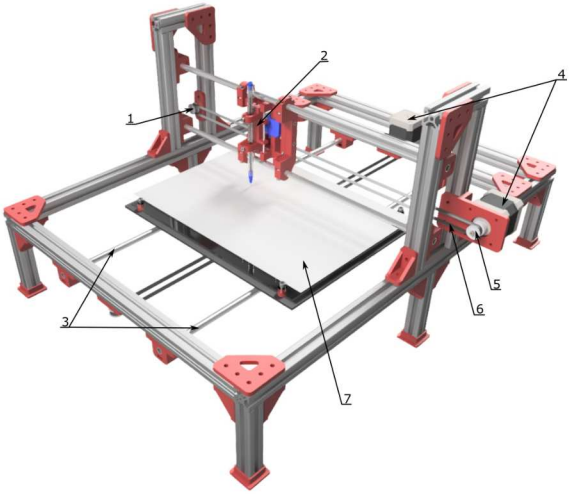
Projekt plotera 2D

promotor: **dr hab. inż.** Krzysztof Mendrok – Katedra Robotyki i Mechatroniki

Abstract: The aim of the thesis is to design 2D plotter, that would be able to draw graphics with the use of ballpoint pen. Device is based on Computer Numerical Control principles. At the beginning of the thesis, basic assumptions are presented. These include: simple mechanic system, good quality plots, low total cost and easy to modify frame. Considering aforementioned assumptions, prototype of the plotter would be built. Following chapter describes fundamental terms of Computer Numerical Control, including CNC machines positioning and programing. In addition, chapter consist of comparison of solution used in contemporary 2D pen plotters. Next chapter focuses on design of the plotter. First 3D model is presented, along with basic concept of machine movements. Additionally, key parameters are calculated, on the basis of which, plotter mechanical components are selected. Moreover, static and modal analysis of the frame is performed. Furthermore electronics components and software are described. Last chapters are compose of field testing of the prototype, results presentation and conclusions.

Proposed Design

The movement along X axis and Y axis is ensured by two stepper motors, then the movement along Z axis by one mini servo motor. Stepper motors was chosen because of its simply construction, high reliability, as well as precise positioning and repeatability of the movement. Moreover, errors do no cumulate from one step to the next. Another important advantages, is capability to accurately operate in an open loop control system. For simple task of moving pen upwards and downwards mini servomotor was selected.



1 - idler pulley 2 - plotting head 3 - linear guides 4 - stepper motors 5 - driving pulley 6 - timing belt 7 - table

Electronic components:

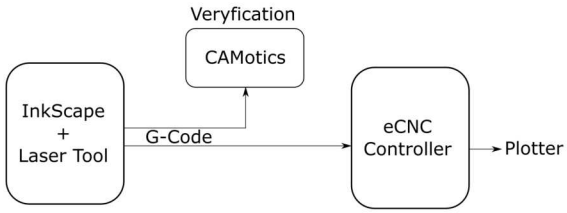
- **Arduino Mega 2560** microcontroller board based on the Atmega2560
- **Arduino Mega Pololu Shield**, which is capable to interface with up to five A4988 stepper motor driven board
- Two **Pololu A4988** stepper motor drivers
- Four limit switches in order to avoid collision between table or plotting head and frame

Software:

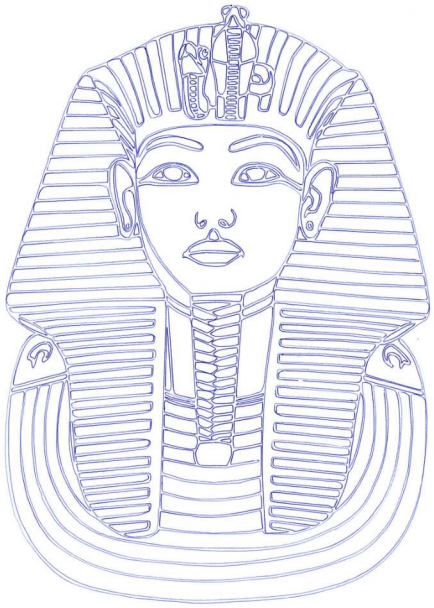
- **Easy CNC** - modular firmware for CNC machine
- **Easy CNC Controller GUI** - machine state control by managing error massages, searching home position, resetting status and position, manual machine movement with three different speeds
- **Inkscape** - open-source vector graphics editor primary operates on Scalable Vector Graphics
- **Inskscape Laser Tool Plug** - generate path for the tool in the form of G-Code
- **CAMotics** - CNC machine tool path simulation

Plotter Workflow

Plotter workflow starts with vector graphic created in Inkscape. To plot simple shapes, including: rectangles, polynomials, ovals etc., user can use built in tools. Shapes consist of vectors would be directly transform to tool path, then with the used of Laser Tool plug-in into G-Code. However, if user would like to plot raster images, case become more complicated. First image have to be import to Inkscape workspace, then, using trace bitmap tool, convert it into path. Mentioned tool has several different methods to accomplish this task: brightness cut-off, edge detection or colour quantization. Selection of the method depends on input image. In addition the higher the resolution of the input image, the better results of output path. Moreover, raster graphics with transparent background, tend to be easier to convert. Next step is to perform G-Code simulation with the used of CAMotics software. If plotting tool operates as intended, user can proceed to final step. At final step, plotter search for home position. Then, Easy CNC Controller starts to send G-Code instruction to the machine and plotting begins.



Test Drawing



Large test drawing: Tutankhamun (7392 lines of code, plotting time: 20 minutes)

Prototype