Korai Viktor - koraiviktor@hotmail.com

Bindics Gergely - bindicsgeri@gmail.com

[E-mail cím]

Epitome

We are building a spectacular drawing unit that is easily controllable and programmable for everyone without any relevant professional knowledge via the self-designed image processing software developed specifically for this machine.

Project I.

Drawing machine

Table of contents

[Introduction 2](#_Toc450667293)

[Requirements 2](#_Toc450667294)

[Features 3](#_Toc450667295)

[Input 3](#_Toc450667296)

[Output 3](#_Toc450667297)

[Block diagram 3](#_Toc450667298)

[Implementation description 4](#_Toc450667299)

[Verification against requirements 10](#_Toc450667300)

[Timetable 10](#_Toc450667301)

[Cost estimation 11](#_Toc450667302)

[Milestones 11](#_Toc450667304)

[Difficulties and solutions 12](#_Toc450667305)

[Summary 13](#_Toc450667306)

***Specification***

# Introduction

We are going to implement an automatic drawing machine, with a user friendly image processing software designed and written for this specific task.

The aim of this project is to make this machine usable for people with no related programming knowledge. Also, with this robotic arm, we can draw figures very precisely to avoid human mistakes, or to be able to draw under different extreme environmental circumstances. For example, using an air-brush combined with this tool, no worker has to suffer from the negative side effects of the toxic gases emitted during the painting.

We are splitting up the project described above into 2 parts.

In the first part we are going to assemble a self-designed manipulator which can move along 2 dimensions. It working with a pre-defined coordinate array which contains integer numbers.

In the second part, we are going to develop the user friendly driver software for this hardware and implement the third axis which is responsible for the movement of the plotting head. (Touches the drawing surfaces or lifts it up.) We are also planning to design a monitoring feature which will display graphically the current state of the ongoing process. Finally, we are going to change the USB communication to Bluetooth.

# Requirements

We have the following requirements stated against the final product: the arm can paint and reproduce the same picture that we loaded into our software. Then, this software should be able to calculate the drawing routes from the input image and save them as individual projects, so that we can load them later in case we want to reuse them. Finally, anyone without related programming knowledge should be able to program the robotic arm with the help of our image processing software.

In the end, the manipulator should reach 1mm precision.

As for the time requirements, the drawing machine will be able to draw a 500x500 pixel image in 5 minutes.

At the end of the whole project, after the 2 semester, it will be able to draw a picture using one color. The plan is to make it easy to implement an improvement, where the unit will be able to use several colors.

# Features

The robotic arm is portable, and can reproduce the input image under extreme environmental circumstances. Also, with this software we can observe and monitor the ongoing process.

# Input

At the end of the first semester for the input we will load a pre-defined coordinate map which will contain the colored points of the final image that the unit will draw.

By the end of the second semester the input of our tool will be a single BMP image, that we have to load into our software, which will define the final picture.

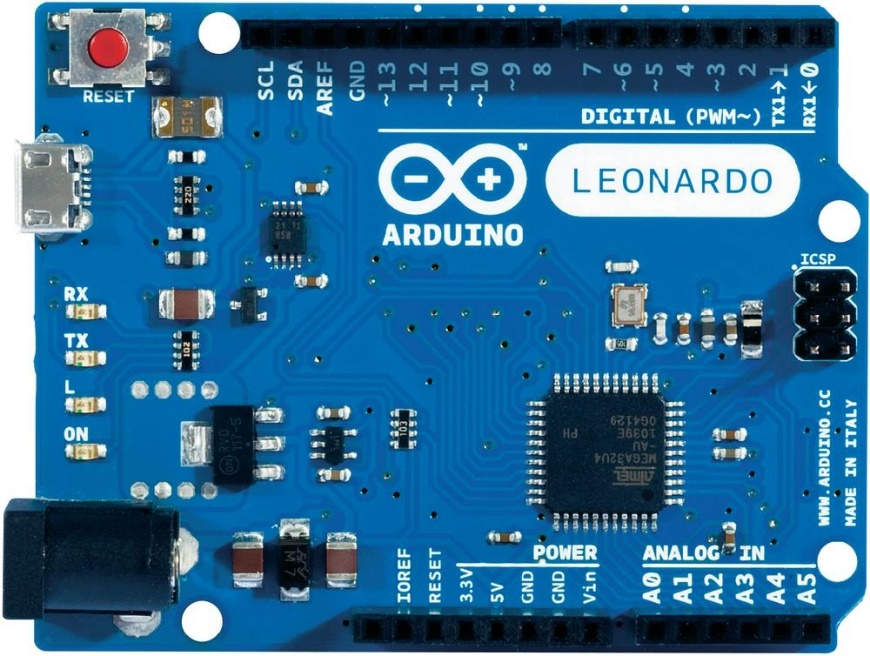
# Output

The drawn black and white picture of the robotic arm on a piece of paper. For now, the device doesn’t contain the 3rd motor which will be responsible for lifting up the drawing head and release it down onto the paper. Thus, we can only draw an image which is made up by one continuous line. The precision we could reach at the moment is 2mm. The maximum size of the image is 200mmx200mm.

# Block diagram



# Implementation description

1. **Hardware list**
   * Microcontroller
     + Arduino Leonardo
       - Microcontroller ATmega32u4
       - Operating Voltage 5V
       - Input Voltage (recommended) 7-12V
       - Input Voltage (limits) 6-20V
       - Digital I/O Pins 20
       - PWM Channels 7
       - Analog Input Channels 12
       - DC Current per I/O Pin 40 mA
       - DC Current for 3.3V Pin 50 mA
       - Flash Memory 32 KB (ATmega32u4) of which 4 KB used by bootloader
       - SRAM 2.5 KB (ATmega32u4)
       - EEPROM 1 KB (ATmega32u4)
       - Clock Speed 16 MHz
       - Length 68.6 mm
       - Width 53.3 mm
       - Weight 20g

We chose this type of microcontroller, because it has several PWM outputs, so it can control up to 3 individual stepper motor drivers without any workaround. As the input voltage is between 7 and 12 Volts, and it has a micro USB input port and a normal DC input port, it can be supplied by our laptop while we program and test the code, so that we don’t have to use any special cable nor individual voltage supply to make it work. On the other hand, when we are finished with the development and the test phase we can use a simple 9V battery as a power supply from within the wooden skeleton itself. This way the robot can work on its own and it doesn’t have to be connected to any external periphery.

* + Laptop



We decided to use laptop because the portability is one of the main features considering our project and it is capable to operate under windows operation system which is indispensable to be able to run our self-designed image processing unit.

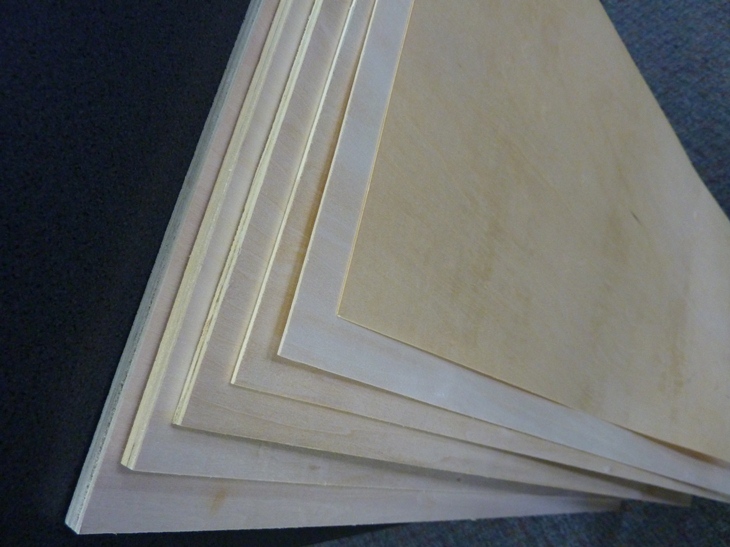
* + Wooden skeleton and other components (gears etc.)



The wooden skeleton is a critical part of our implementation. We considered a lot of materials to use, including different kinds of plastics and metals. Finally, we stuck to the wood because it is far more cost efficient, which is also an important aim of our project. Besides, to manufacture the wooden parts, we only needed manufacturing tools that are easily obtainable at low cost, and also many of them can be found in most of the households. If we used plastic or metal, we would have to buy more special tools, resulting in a lot of extra cost. Finally, it’s light weight is also important from the portability point of view and the motors have limited torque to move the different parts as well.

The wood as a material caused difficulties in exchange for the low price, but we will get back to it in a later section.

* + - Birchplywood4 mm (1525x1525 mm BB/CP)



* + - * Density:Medium density, but excellent strength
      * Values (kg/m3):air dry: 650-830, absolute dry: 610-800
      * Its strength characteristics are similar to beech, but it has lower density, so it is more optimal to manufacture furniture construction
      * It has extraordinary flexibility and toughness
      * Quality: BB/CP  
          BB side: smaller, healthy ingrown knots and slight discoloration may occur  
          CP: smaller branch failures and slight discoloration may occur  
        Easily workable, it is possible to manufacture it rapidly

Because of the density and the thickness of the birch it is a light weighted material but the structure of the plywood makes it strong enough to stabilize and hold the whole frame together.

* + Self-designed manipulator

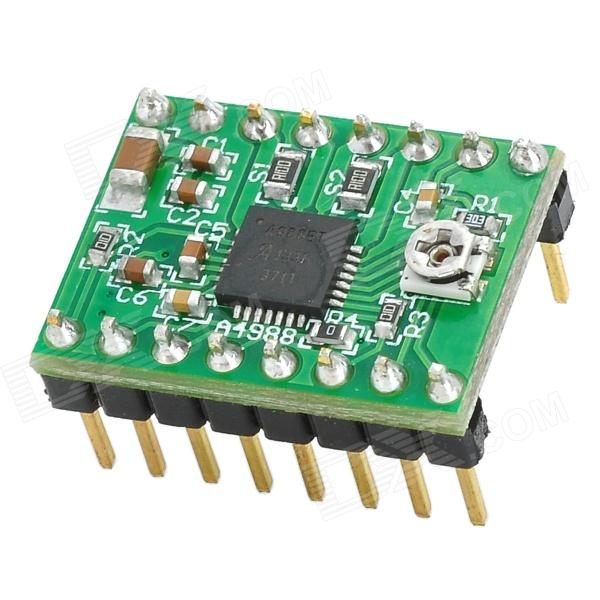


* + - 3 NEMA 17 stepper motor
      * 1.8o step angle
      * 200 steps per revolution
      * Size: 42.3 mm square × 48 mm, not including the shaft (NEMA 17)
      * Weight: 350 g (13 oz.)
      * Shaft diameter: 5 mm “D”
      * Steps per revolution: 200
      * Current rating: 1.2 A per coil
      * Voltage rating: 4 V
      * Resistance: 3.3 Ω per coil
      * Holding torque: 3.2 kg-cm (44 oz.-in)
      * Inductance: 2.8 mH per coil
      * Lead length: 30 cm (12″)
      * Output shaft supported by two ball bearings

We had a hard time to decide whether we use stepper, DC or servo motors.

First, we excluded the servo motor from the possibilities, because it cannot rotate continuously, the maximum was 270 degrees that we could reach, which was not suitable for us, since we didn’t want to include an additional gearing. Then the stepper motor turned out as a winner, because we couldn’t control the DC motor with enough precision.

* + - 3 stepper motor driver



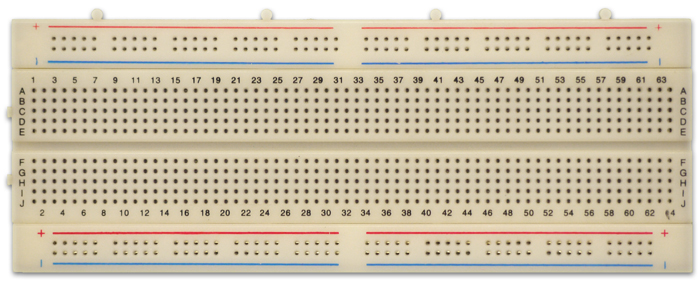
* + - * Dimensions
      * Size: 0.6″ × 0.8″
      * Weight: 1.3 g1
      * General specifications
      * Minimum operating voltage: 8 V
      * Maximum operating voltage: 35 V
      * Continuous current per phase: 1 A
      * Maximum current per phase: 2 A
      * Minimum logic voltage: 3 V
      * Maximum logic voltage: 5.5 V
      * Micro step resolutions: full, 1/2, 1/4, 1/8, and 1/16
      * Reverse voltage protection: No
      * Bulk packaged: No
      * Header pins soldered: No

Choosing this type of driver was an easy decision. It perfectly fits our needs, easy to use and integrate. An important factor was that we could get it from Hungary, so we didn’t have to wait weeks with the shipping, and it was very cheap as well. Additionally, we got a free heatsink for every piece, which solved the overheating issue instantly.

For controlling the driver, we only need a power supply and 2 signals from the Arduino board. As a power supply we used 2 9V batteries to provide enough current for the circuit. It was important, because the torque of the motor depends on the incoming current level. We could set the desired output current of the driver for the motors with the help of the potentiometer placed on the driver. Furthermore, we drove the motor with 4-signal solution. (6-signal solution would be a possibility as well, but we wanted the simpler option.)

Basically, we wanted to put every electrical component into the lower part of the robot which is a container box, these drivers are small so that they don’t take up a lot of space on the breadboard. This is essential in case we want to use additional motors in the future. (For example for the third axis.)

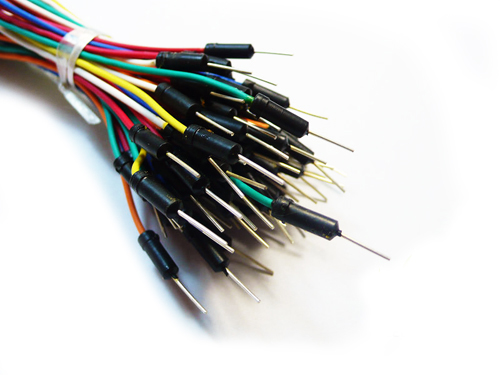
* + - Breadboard



All of our components can be placed onto it without any difficulties, and even we have some spare space for extra components in the future.

It is easy to test the elements individually and we need less cables to provide the sufficient voltage for all parts.

* + - Jumper wires



This is the simplest way to make connections on the breadboard, we bought three kind of jumper wires: female-female, male-male, female-male. The ends of these cables are removable, so we have the final version of the circuit we can solder them together.

* + - 2x 9V battery



This is one of the cheapest power supplies that we could use, and can provide enough power for our circuit design, and it is easily expandable, because we can buy it in almost every shop.

**Software list**

* + Windows operation system

It is one of the most common operation systems, and later we want to develop our image processing software in visual basic, because we already have a lot of experience in it.

* + Graphic design software for e.g. Paint

It is not necessary to use Paint; the user can use any kind of graphic design program or a simple BMP formatted picture instead of creating a new one. It is just an offer because it is included in the windows operation system.

* + Self-developed image processing software

This is one of the main part of our project, in our future plans it will have a duplex communication channel with the microcontroller, thus it can monitor the drawing process meanwhile it provides the needed information about the image that is under construction.

* + Arduino IDE

It is free and designed for our microcontroller so that it is very simple to use to this kind of software developments.

* + Manipulator controlling software developed in Arduino IDE

This will be on the other side of the communication channel mentioned above. This software will take control over the stepper motors and drive them according to the input data which came from the control unit.

1. **Control logic**



# Verification against requirements

We would like to show the unit while it is recreating the image from the pre-defined coordinate map that we loaded into the image processing software.

We would like to measure the precision with a ruler, and the time taken to draw the image with a timer. Another option is to use a millimeter paper, and we make the machine drawing on it, this way we can see the deviation.

# Timetable

# Cost estimation

# 

# Milestones

* Connecting the circuit including the Arduino and 1 stepper motor and successfully drive them
* Manufacturing and assembling the wooden skeleton
* Attaching 1 gear-rack combination to a motor and moving it along 1 dimension
* Assembling the second motor and its gear(s), and being able to move to the desired coordinate in a 2D coordinate system
* Reach the target resolution
* Testing against requirements

# Difficulties and solutions

1. **Friction**

Problem:

Our biggest challenge was to minimize the friction between the sliding surfaces.

Solution:

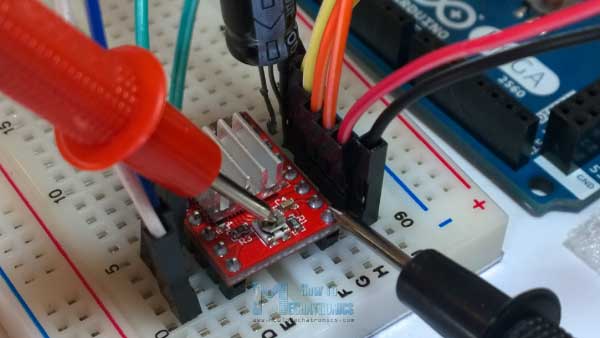


We manufactured sliding surface extensions from birch plywood and covered it with soap. Also, lightened the wood structure where it was possible.

1. **Motor performance**

Problem:

The two motor couldn’t reach the maximum torque with a single 9V battery.

Solution: 

We adjusted the pot meters on the drivers and connected another 9V battery in series.

# Summary

We had hard time realizing our project, but it was a great experience. We enjoyed every moment of the process, even if we had our ups and downs. We encountered a lot of new problems and solutions that we haven’t met before, so after all we could learn a lot from this project. It was quite a big challenge to build up a project from scratch, and select the most suitable materials and components considering the different factors like smooth sliding of the surfaces, the light weight and the overall cost reduction. We are really proud of our machine, and we are happy that we could make it work at the end!

We hope that in the future many people will be able to easily reconstruct the robot at home based on our project, which is uploaded to GitHub so that anybody can download it freely.