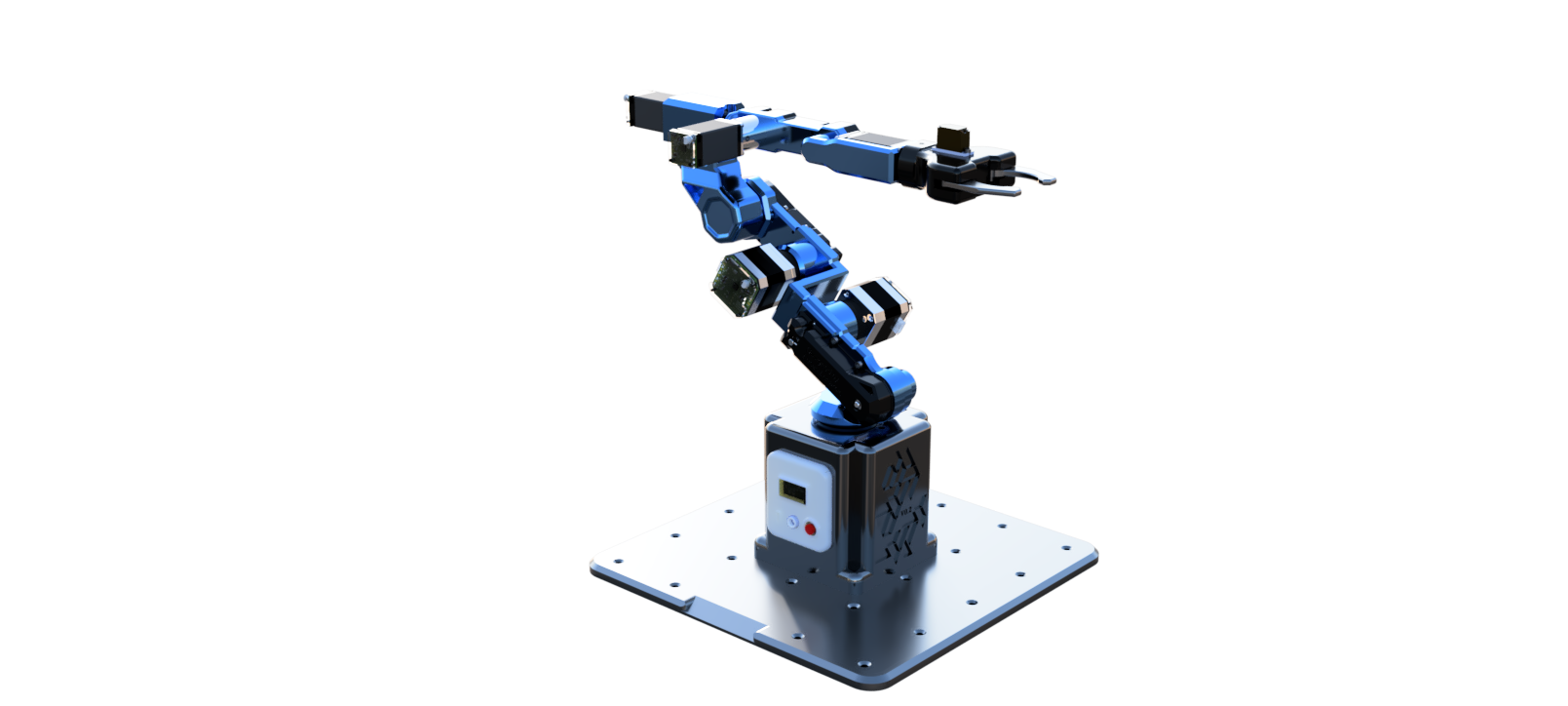
**6 Axis Robotic Arm**

**Technicus Award '22**

GitHub: <https://github.com/Lemme-lab/Robotic-Arm>



Inhaltsangabe

# Einleitung

## 1.1 Konzept

## 1.2 Features

## 1.3 Planned Code

## 1.3.1 General

## 1.3.2 Website

## 1.3.3 Desktop Client

## 1.3.4 Mobile Client

## 1.3.5 Self-Test

### 

# Preislisten

## 2.1 Robotic Arm

## 2.2 Robotic Arm Main Controller

## 2.3 Nema 17 Stepper Motor

## 2.4 Nema 11 Stepper Motor

# 3.0 3D-Modell

## 3.1 Robotic Arm

## 3.2 Motors Nema 17 / Nema 11

## 3.3 Elektronik

## 3.3.1 Main Controller

## 3.3.2 Nema 17 Motor Controller

## 3.3.3 Nema 11 Motor Controller

# 4.0 Elektronik

## 4.1 Robotic Arm Main Controller

## 4.2 Stepper Motor Nema 17

## 4.3 Stepper Motor Nema 11

# Einleitung

## 1.1 Konzept

Für den Technicus Award der HTL, haben wir (Lucas Lenarcic und Matteo Müller) uns,

im Zuge des Werkstättenunterrichts bei Herrn Prof. Grabner Burkhard, überlegt einen 5-Achsen Roboterarm zu bauen.

Der Roboterarm wurde/wird mithilfe von Fusion entworfen. Die einzelnen mechanischen Komponenten des Armgehäuses werden mithilfe eines 3D-Druckers gefertigt. Elektronische Komponenten, wie Motortreiber, Stromversorgung, Steuerung via App oder Web wurden/werden mit der PCB-Design Software Altium entworfen. Für die Herstellung dieser Platinen wird entweder die schulinterne Leiterplattenfertigung hinzugezogen oder ein externer Hersteller.

Es ist geplant mit dem Greifer des Roboterarms Objekte aufzuheben und diese an einem beliebigen, in der Reichweite des Armes gelegenen, Punkt wieder abzulegen.

Die Steuerung soll wie in den folgenden Punkten noch genauer beschrieben über mehrere Varianten möglich gemacht werden.

So soll man den Roboter mithilfe einer Webapplikation, also über eine selbst designte HTML-Website (virtueller Joystick, …) und folglich mit dem ESP32 und dessen eingebautem WLAN-Modul steuern können.  
Die andere Variante wäre bzw. ist eine App, die wir mithilfe von Flutter designen werden.

Ein Bild, das Text, Schiefertafel enthält.

Automatisch generierte Beschreibung

## 1.2 Features

* Self Designed Stepper Motor Controller (A4988)
* Self designed Main Controller with a small footprint
* Compact Size Factor
* Esthetic
* Many other little Features

## 1.3 Planned Code:

## 1.3.1 General:

A few bullet points and explanations for the code part. Different technologies and programming languages such as C++, WEB-Dev, and Flutter will be used. It will go from Basic C and register manipulation to UI-Development on Cross-Platform Mobile Apps.

* Controlling Axis 1- 5
* Save Control sequence
* Remote Connection
* Desktop Application
* Website
* Mobile App
* PHP Database
* C++, HTML, CSS, JS, FLUTTER, JAVA, Python
* Tensorflow, Keras

## 1.3.2 Website:

A HTML Website that must be functional and appealing to use. Features will be send via a ESP32 and the included Wireless module. Cross-platform Data will mean that we will have to save the current data and inputs on a small database. UI and other Features will mean that we have to use JS with or without an additional framework.

* Checking Power Status
* Check Wireless Connection
* Current Voltage/Power Levels
* Slider Controlls for Motors
* Test Motor func
* Switch Mode (Normal Control, Train ,AI)
* Cam Live Stream implementation
* Stop/Start/hold -->pan> Slide buttons
* Speed
* Axis coordinates

## 1.3.3 Desktop Client

A C++ or Java desktop client to be used when wired up to the robot arm. From the User ready interface to debbuging, testing and programming features will be included. This is one of the most important interface to the robot arm and as such it will be programmed first.

* Checking Power Status
* Check Wireless Connection
* Current Voltage/Power Levels
* Slider Controlls for Motors
* Test Motor func
* Switch Mode (Normal Control,AI)
* Cam
* Debug Features / Console
* Test Features
* Stop/Start/Hold --> slidebuttons
* Speed

## 1.3.4 Mobile Client

Last we will do a Mobile Client app for controlling and checking the robot arm on the go. It will be programmed in Flutter or another Mobile Cross Platform Developer Kit.

* Checking Power Status
* Check Wireless Connection
* Current-Voltage/Power Levels
* Settings
* Live Cam

## 1.4.5 Self-Test

Testclasses to exclude code-mistakes. Motors are working in the corract way and also the remote connection and controlling from the web applicatio, if so then we can put everything together and test the whole thing

* Motor
* Remote Connection
* Connections Seriell/Connectors

# Preislisten

## Ein Bild, das Tisch enthält. Automatisch generierte BeschreibungRobotic Arm

Ein Bild, das Tisch enthält.

Automatisch generierte Beschreibung

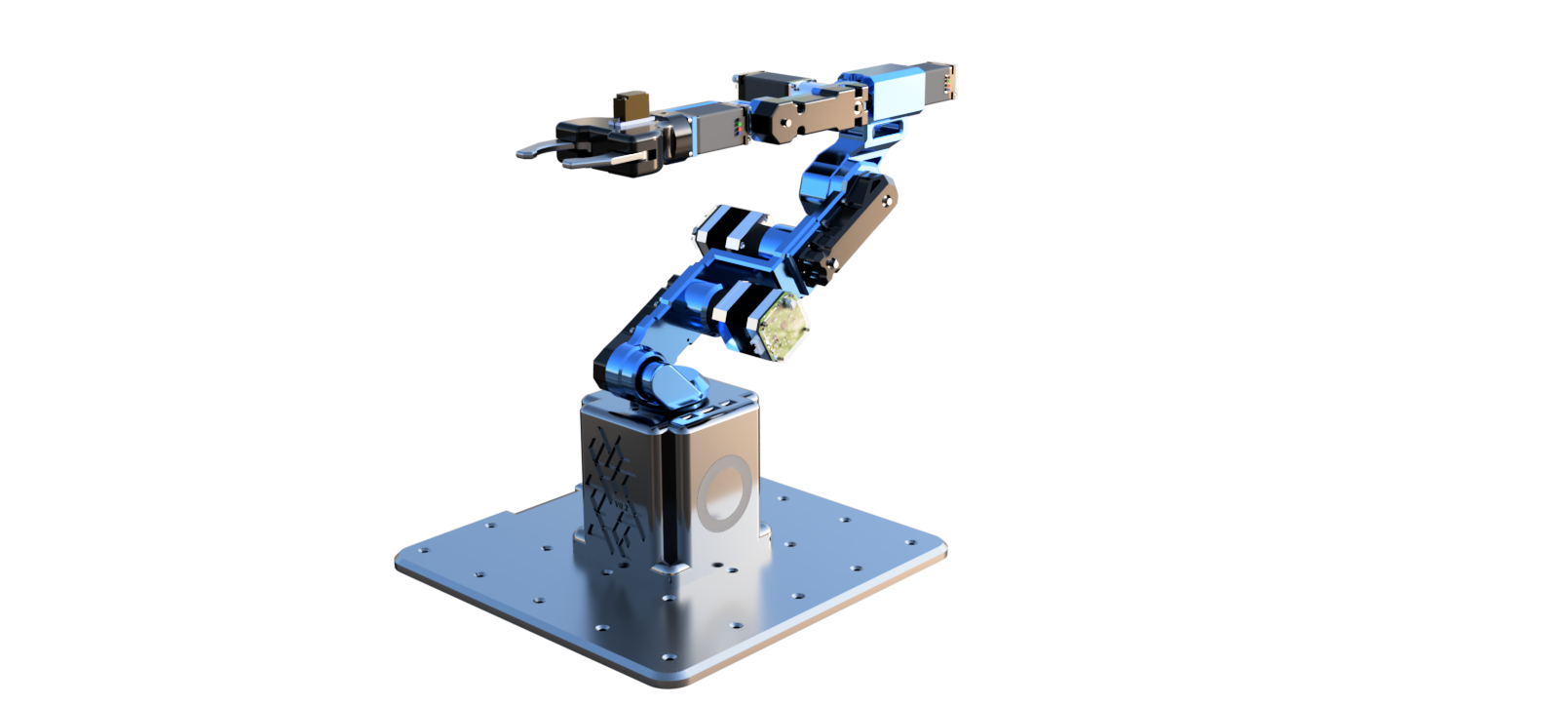
## Robotic Arm Main Controller

## Nema 17 Stepper Motor

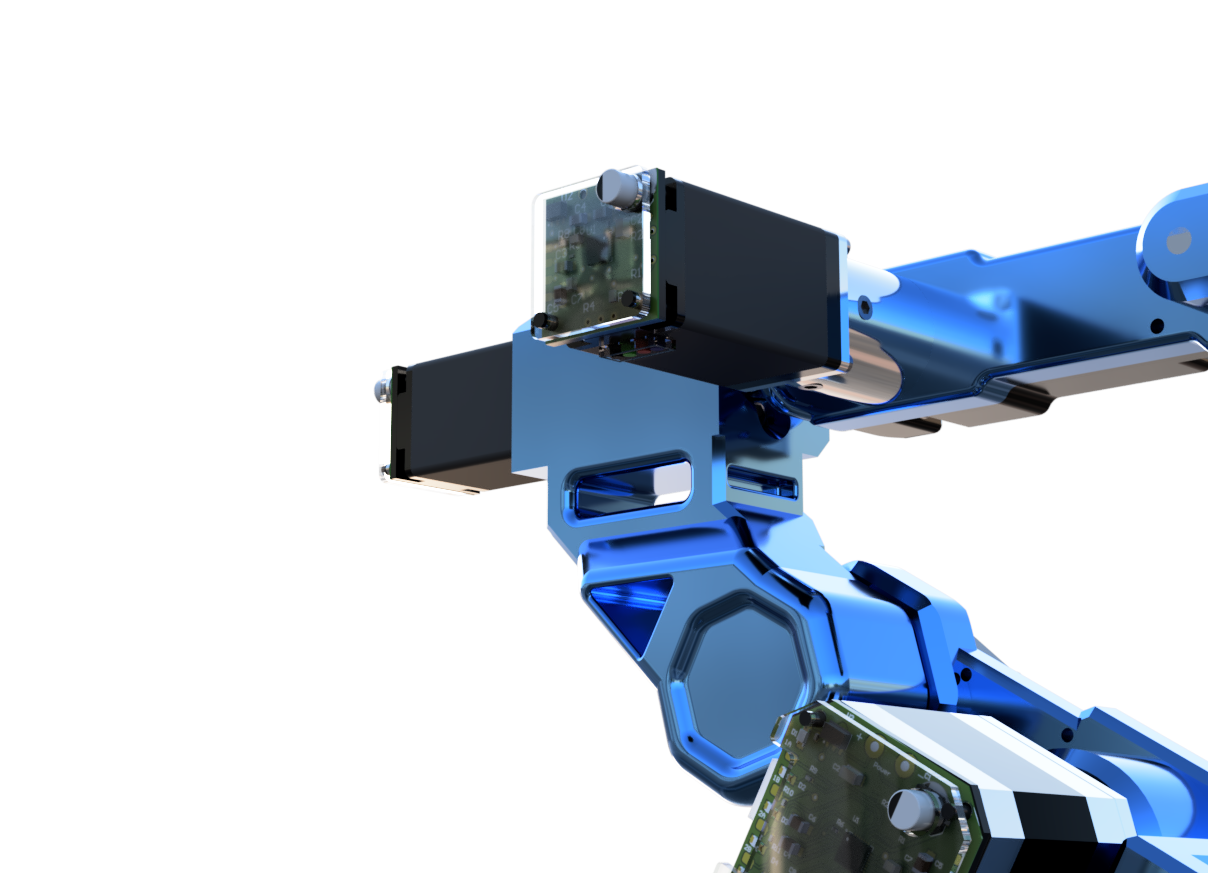
## Nema 11 Stepper Motor

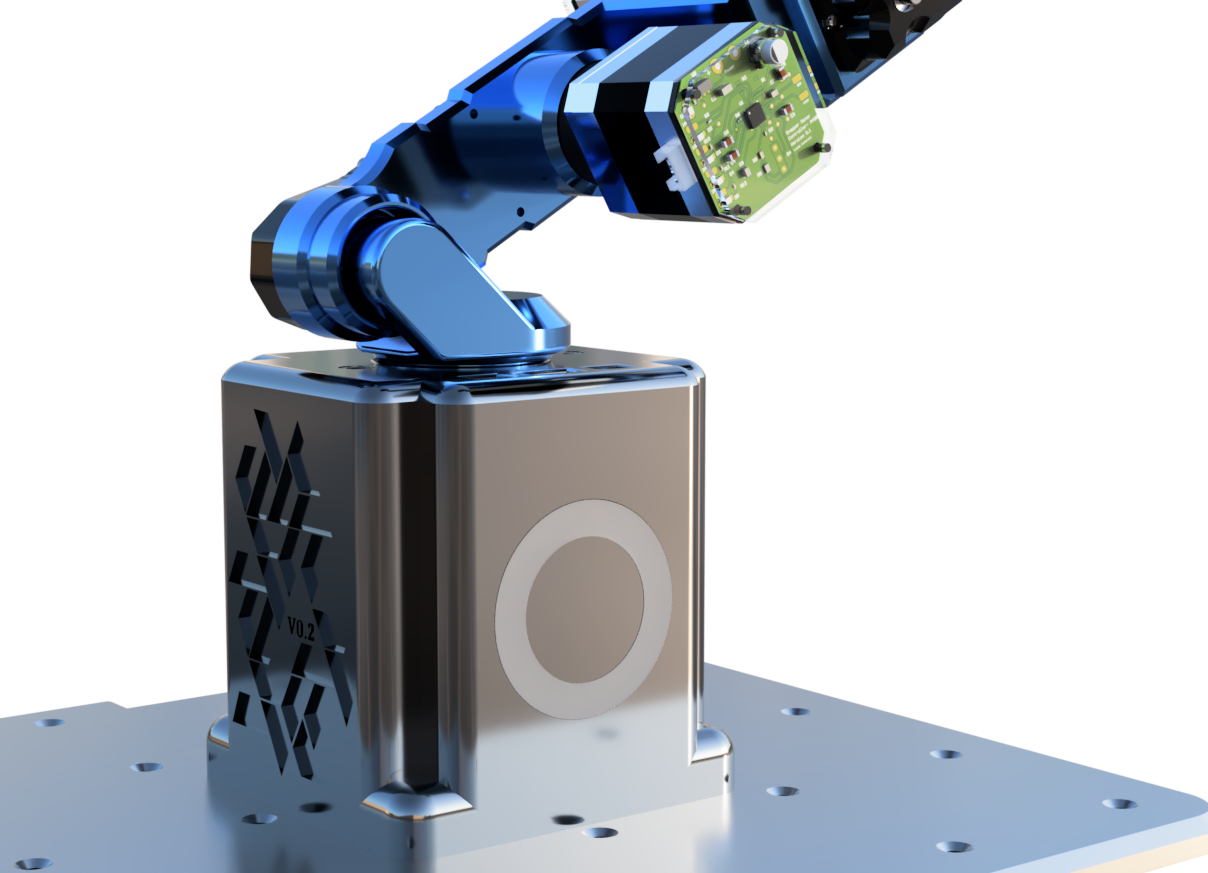
# 3.0 3D-Modells

## 3.1 Robotic Arm



Ein Bild, das dunkel, Automat enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

## 3.2 Motors Nema 17 / Nema 11







## 3.3Elektronik

## Ein Bild, das Text, Elektronik, Schaltkreis enthält. Automatisch generierte Beschreibung3.3.1 Main Controller:

Ein Bild, das Text, Elektronik, Schaltkreis enthält.

Automatisch generierte Beschreibung

## 3.3.2 Nema 17 Motor Controller

Ein Bild, das Text, Elektronik, Schaltkreis enthält.

Automatisch generierte Beschreibung

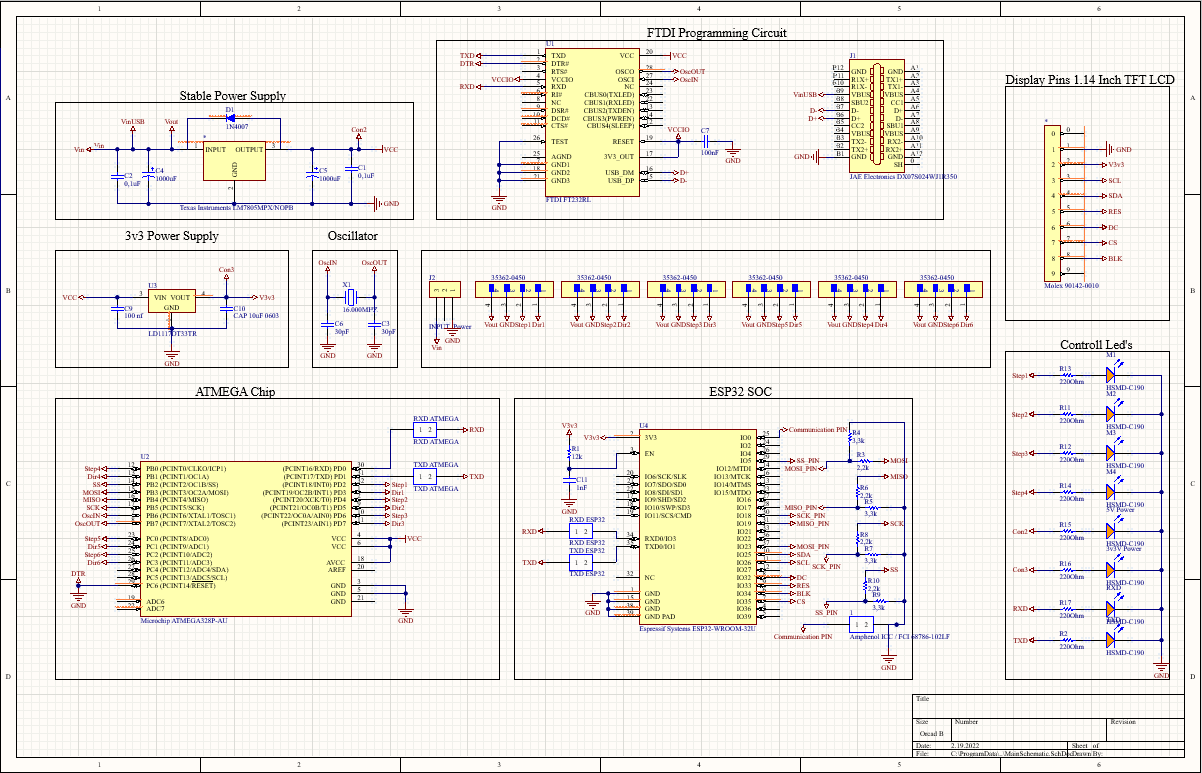
## 3.3.3 Nema 11 Motor Controller

Ein Bild, das Text, Elektronik, Schaltkreis enthält.

Automatisch generierte Beschreibung

# 4.0 Elektronik

# 4.1 Robotic Arm Main Controller



Ein Bild, das Text, Anzeigetafel enthält.

Automatisch generierte Beschreibung

## Ein Bild, das Text, drinnen enthält. Automatisch generierte Beschreibung4.2 Stepper Motor Nema 17

# Ein Bild, das Text enthält. Automatisch generierte Beschreibung

## 4.3 Stepper Motor Nema 12

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung