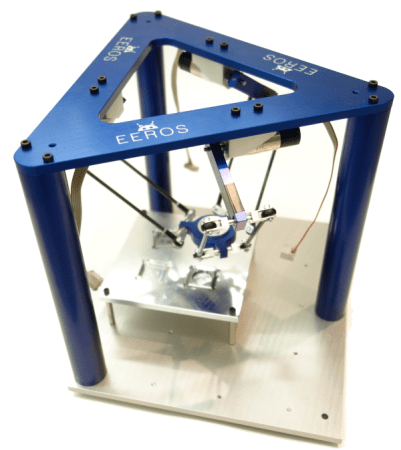
Delta Redesign



Verfasser: Simon Fink

Partner: INF Institut NTB Buchs

Referent: Prof. Dr. sc. Techn. Urs Graf

Korreferent: Prof. Einar Nielsen

Datum: 14.08.2018

Inhalt

[Aufgabenstellung 2](#_Toc521690371)

[Allgemeine Beschreibung 2](#_Toc521690372)

[Evaluation 3](#_Toc521690373)

[Hardware 3](#_Toc521690374)

[EEROS - Software 3](#_Toc521690375)

[Control System 3](#_Toc521690376)

[Safety System 3](#_Toc521690377)

[Sequencer 4](#_Toc521690378)

[HAL 4](#_Toc521690379)

[Theorie 5](#_Toc521690380)

[Berechnung Hardware 5](#_Toc521690381)

[Berechnung der Leistung für die Motoren 5](#_Toc521690382)

[Berechnung Encoder Signale 5](#_Toc521690383)

[Auslegung des Transistors für den Elektromagneten 5](#_Toc521690384)

[Spannungsteiler der Encoder Signale 5](#_Toc521690385)

[Berechnung HAL 5](#_Toc521690386)

[Encoder IEH2-4096 5](#_Toc521690387)

[Encoder HEM3-256 5](#_Toc521690388)

[Auslastung 6](#_Toc521690389)

[Beagle Bone Blue 7](#_Toc521690390)

[Adapterprint 7](#_Toc521690391)

[Literaturverzeichnis 8](#_Toc521690392)

[Eigenständigkeitserklärung 8](#_Toc521690393)

# Aufgabenstellung

Im Rahmen einer MSE Vertiefungsarbeit soll die Software des bestehenden EEDURO-Delta Roboters auf die aktuelle EEROS Version angepasst werden. Da dieser Roboter als Vorzeigebeispiel des EEROS-Frameworks dient, soll ein ausführliches Tutorial auf der EEROS Homepage erstellt werden.

# Allgemeine Beschreibung

Diese Arbeit beinhaltet die Ergänzung und Anpassung des bestehenden Source-Codes des Delta Roboters auf die aktuelle Version von EEROS. Dieser ist auf GitHub unter <https://github.com/simonfing/VTDelta> zu finden.

Die Tutorials befinden sich unter <http://hw.eeros.org/eeduro> . Die Step-by-Step Anleitung um sich einen Delta Roboter zu bauen befindet sich unter <http://hw.eeros.org/eeduro/delta/build_your_own>.

# Evaluation

## Hardware

In dieser Arbeit wird nur die Hardware sowie die Software des Delta Roboters behandelt. Folgende Hardware soll wiederverwendet werden:

1. 3 Stück Faulhaber 1524-012SR Motoren mit IEH2-4096 Encoder und 15-8 76:1 Getriebe.
2. 1 Stück Faulhaber 0816-012SR Motor mit HEM3-256 Encoder und 08/3 120:1 Getriebe
3. Elektromagnet GTO-14-0.5000 12VDC
4. 3 beleuchtete Drucktaster
5. LED Beleuchtung
6. Maus für manuelle Steuerung

Die Berechnungen und Auslegungen für die Punkte 1-3 befinden sich im Kapitel Theorie.

Die drei beleuchteten Drucktaster werden direkt über die GPIOs des Beagle Bone Blue angeschlossen. Dieses besitzt 8 direkt zugängliche GPIOs sowie einige indirekt zugängliche. Indirekt zugänglich bedeutet, dass diese I/Os standardmässig nicht als GPIO konfiguriert sind, sondern andere Funktionen haben, wie z.B. UART, SPI, I²C usw. Mehr zur Umsetzung, wie die Drucktaster an das Beagle Bone Blue angeschlossen werden, im Kapitel Beagle Bone Blue – Adapterprint.

Die verbauten LED Bänder benötigen 12V. Da das Beagle Bone Blue ebenfalls mit 12V betrieben werden kann, werden die LED Bänder direkt an diese angeschlossen.

Da das Beagle Bone Blue über einen USB 2.0 host verfügt, kann eine Maus an diesem angeschlossen werden.

## EEROS - Software

EEROS ist ein open source Real-Time Robotics Software Framework, entwickelt an der Interstaatlichen Hochschule für Technik in Buchs. Dieses Framework wurde und wird für die Regelung des Delta Roboters verwendet.

Es besteht aus dem Control System, dem Sequencer, dem Safety System und dem Hardware Abstraction Layer (HAL). Da der Sourcecode bereits besteht, jedoch mit einer älteren Version von EEROS entwickelt wurde, wird dieser auf den neuesten Stand gebracht. Nachfolgend werden die Änderungen der einzelnen Komponenten erklärt. Die genauen Funktionen der einzelnen Systeme werden nicht erklärt. Die Beschreibung zu diesen sind unter <http://wiki.eeros.org/eeros_architecture/start> zu finden.

### Control System

Im Control System gibt es nur kleinere Änderungen. So wurde der «Board»-Block gegen einen «Mux»-Block und einen «DeMux»-Block ausgetauscht. Der Mux nimmt die vier Encoder Signale entgegen und packt diese in einen AxisVector. Dem entgegengesetzt wandelt der DeMux einen AxisVector in vier «double»-Signale um und gibt diese Werte den entsprechenden Motoren weiter. Das Diagramm sowie die Beschreibung des Control Systems befindet sich unter <http://hw.eeros.org/eeduro/delta/software#control_system>.

### Safety System

Das Safety System wurde aus dem bestehenden Code übernommen. Es wurden die Zuweisungen der Safety Levels und Safety Events angepasst. Das Safety Level “slJoystickTeaching” wurde entfernt, da der Xbox-Controller nicht mehr verwendet wird. Es wurde jedoch ein neues Safety Level “slCalibrating” mit dem entsprechenden Safety Event “doCalibrating” eingefügt. Somit ist es während des laufenden Betriebes möglich eine Neukalibrierung durchzuführen. Die genaue Beschreibung des Safety Systems sowie dessen Diagramm ist unter <http://hw.eeros.org/eeduro/delta/software#safety_system> zu finden.

### Sequencer

Sequenzen waren im bestehenden Code schon vorhanden, jedoch hat sich der Sequencer in letzter Zeit geändert. Die Sequenzen wurden nun unterteilt in “Main Sequence”, “Sort Sequence”, “Shuffle Sequence”, “Mouse Sequence”, “MoveBlock Sequence” und “Calibration Sequence”. Die einzelnen Schritte der Sequenzen wurden in “Steps” unterteilt, hinter welchen ein einzelner Arbeitsschritt steht. Zudem wurden noch Monitore und “Exception Sequences” eingeführt. Der genaue Ablauf der Sequenzen ist dem Diagramm und der Beschreibung unter <http://hw.eeros.org/eeduro/delta/software#sequencer_&_sequences> zu entnehmen.

Die “Main Sequence” wird immer durchlaufen. Je nach aktuellem Safety Level schaltet die “Main Sequence” die entsprechende Sequenz hinzu.

Der “Sort Sequence” sowie der “Shuffle Sequence” wurde ein “Mouse Move Monitor” angehängt. Dieser überprüft laufend die Bewegung der Maus. Sobald diese bewegt wird, ruft der Monitor die “Move Mouse Exception Sequence” auf, welche die laufende Sequenz beendet und das Safety Event “doMouseTeaching” auslöst. Die “Main Sequence” ruft nun die “Mouse Sequence” auf.

Der “Mouse Sequence” wurde ein “Time Out Monitor” angehängt. Dieser überprüft wie lange es dauert, bis die “Mouse Sequence” durchlaufen ist. Bei bewegen der Maus, oder drücken einer Taste, wird dieser zurückgesetzt. Sobald die Zeit überschritten ist, wird die “Mouse Sequence Time Out Exception Sequence” aufgerufen, welche die “Mouse Sequence” beendet und das Safety Event “doAutoMoving” auslöst.

Um in die “Calibration Sequence” zu gelangen muss zuerst das Safety Event “doEmergency” ausgelöst werden. Dies geschieht, indem der rote Drucktaster betätigt wird. Die LED dieses Tasters beginnt nun zu leuchten. Durch 2 Sekunden langes drücken auf den roten Drucktaster wird nun das Safety Event “doCalibrating” ausgelöst. Die “Main Sequence” ruft nun die “Calibration Sequence” auf.

### HAL

Der Hardware Abstraction Layer wurde erst zu einem späteren Zeitpunkt als eigenes Element in EEROS hinzugefügt. Dementsprechend wurde für die neue Software des Delta Roboters ein HAL-Konfigurations-File erzeugt, in welchem die Motoren, Encoder und GPIOs des Delta Roboters beschrieben sind. Die Berechnung des HAL-Konfigurations-File befinden sich in Kapitel Theorie – Berechnung – HAL oder unter <http://hw.eeros.org/eeduro/delta/software#hal>.

# Theorie

## Berechnung Hardware

### Berechnung der Leistung für die Motoren

Die Motoren

### Berechnung Encoder Signale

### Auslegung des Transistors für den Elektromagneten

Da der Elektromagnet mit 12V betrieben werden kann, ein GPIO des Beagle Bone Blue jedoch nur 3.3V liefert, ist eine kleine Schaltung nötig.

Der Elektromagnet wird mit 12V betrieben und hat eine Leistung von 1.3W. Somit ergibt sich für den Kollektorstrom einen Wert von . Der BC337 liefert einen Kollektorstrom von bis zu 800 mA. Für einen Kollektorstrom von 108 mA ist laut Datenblatt ein Basisstrom von 0,8 mA bis 3 mA nötig. Da die GPIOs des Beagle Bone Blue 3.3V liefern ist hierfür ein Vorwiderstand von 1kΩ bis 4kΩ nötig.

Parallel zum Elektromagneten wurde in entgegengesetzter Stromrichtung eine Diode verbaut. Diese dient zum Schutz des Transistors und dem GPIO Port des Beagle Bone Blue. Sobald der Elektromagnet ausgeschaltet wird, erzeugt das noch vorhandene Magnetfeld einen Strom im Elektromagneten, welcher dann über Diode und den Elektromagnet fließt, bis er verschwindet.

### Spannungsteiler der Encoder Signale

Um die Spannung der Encoder Signale von 5V auf 3.3V zu bekommen wurde ein Spannungsteiler, mit je einem 330Ω und einem 680Ω Widerstand, verwendet. Die dadurch am 680Ω Widerstand und am Encoder Eingang des Beagle Bone Blue anliegende Spannung ist .

## Berechnung HAL

### Encoder IEH2-4096

Die Encoder der Achsenmotoren liefern 4096 Linien pro Umdrehung. Durch eine vier Quadranten Auswertung ergibt dies 16.384 Flanken pro Umdrehung. Da auf den Achsenmotoren ein Getriebe mit Übersetzung von 76:1 angebracht ist, entspricht eine Umdrehung der Motorachse 1/76 Umdrehung der Getriebeachse oder 4,737°. Für eine volle Umdrehung der Getriebeachse würde der Encoder 1.245.184 Flanken erhalten. Da sich diese Achse jedoch um höchstens 120° drehen soll wird die «scale» im HAL Konfigurationsfile auf diesen Wert angepasst.

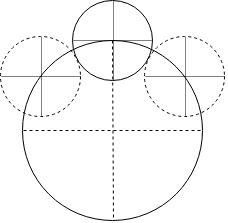
Berechnung maximaler Drehwinkel:

### Encoder HEM3-256

Äquivalent zur Berechnung des ieh2 Encoders. Der hem3-256 liefert 256 Linien pro Umdrehung. Durch FQD ergibt das 1024 Flanken pro Umdrehung des Motors. Das Getriebe hat in diesem Fall jedoch ein Übersetzungsverhältnis von 120:1, somit ergibt dies 122.880 Flanken für eine Umdrehung der Getriebeachse. Der maximale Drehbereich wird jedoch auf 320° beschränkt, welches ebenfalls im HAL Konfigurationsfile mitberücksichtigt wird.

Der genaue maximale Drehwinkel wird wie folgt berechnet:

Aus den Konstruktionszeichnungen des Delta Roboters ist ersichtlich, dass sowohl auf dem «Rotating tool carrier[[1]](#footnote-1)» als auch auf dem «TCP link[[2]](#footnote-2)» ein Stift eingebaut wird, der eine komplette Umdrehung der Teile verhindert.



2\*r

Abbildung 1: "Rotating tool carrier" Beschränkung

D

x

Die Stifte haben einen Durchmesser von 1.5 mm und liegen auf einem Kreis mit Durchmesser 50 mm. Aus den Mittelpunkten dieser Kreise bildet sich ein gleichschenkliges Dreieck. Anhand von Trigonometrie kann der Winkel x bestimmt werden.

Da dieser Anschlag von beiden Seiten gezählt werden muss ergibt sich ein gesamter Drehbereich von

# Auslastung

# Beagle Bone Blue[[3]](#footnote-3)

Das Beagle Bone Blue ist ein Linux basierter “Single-Board-Computer”. Es verfügt über einen 1GHz ARM Cortex-A8 Prozessor, einen 512 MB DDR3 RAM sowie über zwei 32-bit programmierbare Real-Time Units (PRU).

Um den Delta Roboter zu steuern besitzt das Beagle Bone Blue:

* 4 Motorenausgänge,
* 4 Quadratur Encoder,
* 8 GPIOs und
* einen USB Anschluss.

Um dies mit dem bestehenden Delta Roboter verwenden zu können ist jedoch ein zusätzlicher Adapterprint nötig.

## Adapterprint

Die Encoder der Motoren werden mit 5V betrieben. Die Encoder-Eingänge des Beagle Bone Blue dürfen jedoch maximal mit 3,3V gespiesen werden. Um dies zu bewerkstelligen, wurde an jedem Encoder Signal ein Spannungsteiler eingebaut.

Der Elektromagnet des Delta Roboters ist auf eine Spannung von 12V ausgelegt. Die GPIOs des Beagle Bone Blues liefern jedoch nur 3,3V. Um ein genügend großes Magnetfeld aufzubauen, wurde mit einem Transistor 12V auf den Elektromagnet geschaltet. Um den Transistor sowie das Beagle Bone Blue beim Abbau des Magnetfeldes vor Überspannung zu schützen, wurde noch eine Diode eingebaut.

Das Schema des Adapterprints ist unter <http://hw.eeros.org/eeduro/delta/hardware> zu finden.

# Literaturverzeichnis

# Eigenständigkeitserklärung

Hiermit bestätige ich, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Hilfsmittel benutzt habe. Die Stellen der Arbeit, die dem Wortlaut oder dem Sinn nach anderen Werken entnommen sind, wurden unter Angabe der Quelle kenntlich gemacht.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Simon Fink Datum

1. https://github.com/eeduro/delta-mechanics/blob/master/EEDURO-D-011-00.pdf [↑](#footnote-ref-1)
2. https://github.com/eeduro/delta-mechanics/blob/master/EEDURO-D-009-00.pdf [↑](#footnote-ref-2)
3. https://beagleboard.org/blue [↑](#footnote-ref-3)