

Abschlussarbeit zur Erlangung des akademischen Grades
Bachelor of Science

Zentrale Steuerung der Maschine zur Fasermattenproduktion für den LHCb SciFi Tracker

David Hannes

24. Juni 2014

Lehrstuhl für Experimentelle Physik V
Fakultät für Physik
Technische Universität Dortmund

Erstgutachter dieser Arbeit ist Prof. Dr. Bernhard Spaan.
Zweitgutachter ist Dr. Reiner Klingenberg.

Kurzfassung

Diese Bachelorarbeit beschäftigt sich mit der Steuerung einer Maschine zur Herstellung der Fasermatten für das Upgrade der Tracking-Stationen am LHCb-Detektor. Aufgrund eines Upgrades des LHCb-Experiments werden neue Detektoren benötigt, die aus scintillierenden Fasern hergestellt werden. Diese Fasern werden mithilfe eines Wickelrades zu Matten gelegt und verklebt. Anschließend werden die Matten zu Modulen verarbeitet, die am LHCb-Experiment als Detektoren verbaut werden. Die Wickelung der Matten wird durch einen Prototypen für eine Wickelmaschine, auf den in Kapitel 2 genauer eingegangen wird, ermöglicht. Die wesentliche Schwierigkeit des Prototyps ist, dass er sich aus vielen einzelnen Komponenten zusammensetzt, die alle autonom voneinander funktionieren. Im Rahmen dieser Bachelorarbeit werden die Komponenten über eine zentrale Steuereinheit verbunden, um somit die Produktion der Fasermatten zu verbessern. Dies sieht in einem ersten Schritt vor, die bisherigen Möglichkeiten des Prototyps bereitzustellen. Als zweiter Schritt folgen kleine Optimierungen. Mit der zentralen Steuerung werden zukünftig gewünschte Veränderungen leicht softwareseitig zu implementieren sein, sodass kein aufwändiger Umbau der Hardware erforderlich sein wird.

Abstract

This bachelor thesis deals with the control of a prototype for the production of fibre mats which are required for the upgrade of the tracking stations at the LHCb detector. Due to an upgrade of the LHCb experiment new detectors made of scintillating fibres are required. These fibres are assembled to mats using a winding wheel. Afterwards these mats are processed to modules which build the detector at the LHCb experiment. The prototype described later is in charge of winding single fibres to mats. The major difficulty of this prototype is the fact that there are many different components working autonomously. In the context of this bachelor thesis the components will be brought together in order to improve the production. This improvement will be realised in two steps. The first step is to implement all previous functions and giving moderate advantages to the existing system. In the second step the centralised control will provide a basis for the future by simplifying the whole production and being able to implement further functions more easily.

Abbildungsverzeichnis

1.1	Schematischer Aufbau des LHCb-Detektors nach dem geplanten Upgrade 2018/19.	3
1.2	Die drei Stationen des SciFi Trackers.	4
1.3	Schematischer Aufbau des oberen Abschnitts eines Teilmoduls.	5
1.4	Fasermatte bestehend aus fünf Lagen szintillierender Fasern.	6
2.1	Prototyp für eine Maschine zur Fasermattenproduktion der TU Dortmund.	8
2.2	Querschnitt einer Fasermatte mit Markierungen zum Zuschneiden.	10
3.1	Schematischer Aufbau des Prototyps vor dem Umbau.	13
3.2	Schematischer Aufbau des Prototyps nach dem Umbau.	13
3.3	Schaltplan für das Spiegel-Transistor-System.	15
3.4	Verschaltung des Raspberry Pis.	15
3.5	Schaltplan für das Potentiometer (Spannungsteiler).	16
3.6	Signalausgang des Raspberry Pis für verschiedene Eingangsspannungen beginnend bei maximaler Spannung.	17
3.7	Signalausgang des Raspberry Pis für verschiedene Eingangsspannungen beginnend bei minimaler Spannung.	18
3.8	Signalausgang des Raspberry Pis für verschiedene Eingangsspannungen beginnend bei maximaler Spannung auf den Bereich 1,15 V bis 1,33 V fokussiert.	18
3.9	Signalausgang des Raspberry Pis für verschiedene Eingangsspannungen beginnend bei minimaler Spannung auf den Bereich 1,18 V bis 1,33 V fokussiert.	19
3.10	Beurteilung der Qualität des ADC durch den Vergleich von Eingangsspannung zur ausgegebenen Spannung.	20
3.11	Struktureller Aufbau des Softwarekonzepts.	21

Inhaltsverzeichnis

1	Einführung	1
1.1	Large Hadron Collider	1
1.2	Large Hadron Collider beauty	2
1.2.1	Aufbau des LHCb-Experiments	2
1.2.2	Scintillating Fibre Tracker	4
2	Fasermattenproduktion	7
2.1	Anforderungen	7
2.2	Prototyp	7
2.3	Produktion	9
3	Zentrale Steuerung des Prototyps	11
3.1	Motivation	11
3.2	Voraussetzungen	11
3.2.1	Hardware	11
3.2.2	Software	12
3.3	Zielsetzung	13
3.3.1	Aufbau	13
3.3.2	Funktionen	14
3.4	Umsetzung	14
3.4.1	Aufbau und Verschaltung der Hardware	14
3.4.2	Testmessungen an der Hardware	16
3.4.3	Software	21
3.5	Erweiterungsmöglichkeiten	24
4	Zusammenfassung	25
	Literaturverzeichnis	27

1 Einführung

Die aktuelle Teilchenphysik beschäftigt sich mit Fragen über die Zusammensetzung der Materie und ihrer kleinsten Bestandteile. Hierfür sind Teilchenkollisionen die Methode, um immer kleinere Strukturen aufzulösen, neue Teilchen zu produzieren und physikalische Eigenschaften auf diesen Größenskalen zu untersuchen. Der weltweit größte Verbund von Instituten, Hochschulen und Forschern zur Untersuchung von (Elementar-)Teilchen ist das CERN¹ in Genf [2]. Es besteht aus einem Komplex von verschiedenen Beschleunigern, an denen unterschiedliche Experimente angesiedelt sind. Ziele der einzelnen Experimente sind unter anderem die Entdeckung neuer Teilchen. Außerdem werden neue Gesetzmäßigkeiten und bereits bekannte Teilchen genauer untersucht, um das Standardmodell der Teilchenphysik zu erweitern und zu verifizieren [1].

Im Folgenden wird zunächst der LHC² als Teilchenbeschleuniger am CERN kurz erläutert werden, bevor das für die Bachelorarbeit relevante Experiment, das LHCb³-Experiment, näher betrachtet wird und damit verbunden die Grundlagen zum Verständnis der vorliegenden Bachelorarbeit erläutert werden.

Die Beschreibung des LHCb und seiner Komponenten bezieht sich auf den Stand nach dem geplanten Upgrade im Jahre 2018/19 [4]. Das Upgrade des LHCb ermöglicht die Nutzung einer höheren Luminosität⁴, die durch den LHC gegeben ist. Die höhere Luminosität ergibt mehr Ereignisse pro Zeit in den Detektoren, die entsprechend dafür aufgerüstet werden.

1.1 Large Hadron Collider

Der LHC ist mit 26,7 km der größte Teilchenbeschleuniger der Welt [2], welcher gegenläufige Hadronenpakete (entweder Protonen oder Bleikerne) führt und an ausgewählten Punkten zur Kollisionen bringt. An diesen Kollisionenpunkten befinden sich die Experimente der verschiedenen Forschungsgruppen. Die größten Experimente sind ATLAS⁵, CMS⁶, ALICE⁷ und LHCb [1]. Das ATLAS- und CMS-

¹Conseil Européen pour la Recherche Nucléaire [1]

²Large Hadron Collider [2]

³Large Hadron Collider beauty [3]

⁴Anzahl der Teilchenkollisionen pro Fläche und Zeit [5]

⁵A Toroidal LHC ApparatuS [6]

⁶Compact Muon Solenoid [7]

⁷A Large Ion Collider Experiment [8]

Experiment verfolgen ähnliche Ziele und versuchen unter anderem die Existenz des Higgs-Teilchens nachzuweisen [1]. ALICE beschäftigt sich mit der Frage des Zustands des Universums kurz nach dem Urknall, dafür werden Quark-Gluonen Plasmen erforscht [1]. Das für die Bachelorarbeit relevante Experiment ist das LHCb-Experiment, was im Folgenden genauer erklärt wird.

1.2 Large Hadron Collider beauty

Das LHCb-Experiment verfolgt zwei Hauptziele. Zum einen werden seltene Zerfälle untersucht, um die Vorhersagen des Standardmodells (SM) zu überprüfen und zum anderen wird CP -Verletzung erforscht, die die Asymmetrie der Materie-Antimaterie-Verteilung im Universum erklären soll [3]. Das „beauty“ im Namen des Experiments leitet sich aus den dafür untersuchten Teilchen ab. Am LHCb wird der Zerfall von B-Mesonen⁸ untersucht. Die Vermessung dieser - durch die starke Wechselwirkung hervorgerufenen - Teilchen stellt besondere Anforderungen an die benutzten Detektoren. Deswegen werden der Aufbau und die Funktionsweise des Experiments genauer beschrieben, wobei ein Augenmerk auf die für die Bachelorarbeit relevanten Aspekte gelegt wird.

1.2.1 Aufbau des LHCb-Experiments

Der schematische Aufbau des LHCb-Experiments nach dem Upgrade ist in Abbildung 1.1 dargestellt.

Es ist zu erkennen, dass es sich dabei um ein sogenanntes Vorwärtsspektrometer handelt [9]. Dies liegt daran, dass die bei den Proton-Proton-Kollisionen entstehenden B-Mesonen vorrangig in kleinen Raumwinkeln um die Strahlrichtung fliegen und es deswegen am sinnvollsten ist diesen Bereich zu detektieren.

Der Kollisionspunkt wird von dem ersten Detektor umschlossen. Hierbei handelt es sich um den VELO⁹, der den Zerfallsort, der in der Proton-Proton-Kollision entstehenden B-Mesonen, bestimmt. Der VELO ist ein Halbleiterspurdetektor, der über Siliziumpixel [4] die Teilchenspuren rekonstruiert. Der Ort des VELOs wird durch die geringe Lebenszeit ($< 1,65 \cdot 10^{-12}$ s [10]) der B-Mesonen und die damit verbundene geringe Flugstrecke von wenigen Zentimetern motiviert.

Als nächstes folgt RICH1¹⁰, der mit Hilfe des Cherenkov-Effekts den Teilchenimpuls in einem Bereich von 1 bis $50 \frac{\text{GeV}}{c}$ [3] misst.

Hinter dem RICH1 und vor dem Magneten befindet sich der UT¹¹, der aus Siliziumstreifen besteht und die Bahn der durchfliegenden Teilchen darstellt [4].

⁸Mesonen, die beauty/bottom-Quarks enthalten.

⁹Vertex Locator

¹⁰Ring-Imaging Cherenkov detector

¹¹Upstream Tracker

Anschließend werden die Teilchen, abhängig von Ladung und Impuls vom Magneten abgelenkt. Mit Hilfe des SciFi¹² Trackers, hinter dem Magneten kann über die Ablenkung der Teilchen im Magnetfeld eine Aussage über deren Impuls und Ladung getroffen werden. Im Anschluss an den SciFi Tracker folgt der RICH2 Detektor, der Impulse bis etwa $150 \frac{\text{GeV}}{c}$ [3] auflösen kann.

Dahinter befinden sich das ECAL¹³ und das HCAL¹⁴. Die beiden Kalorimeter sind für die Messung der Gesamtenergie der entsprechenden Teilchen zuständig.

Beide Kalorimeter bestehen aus mehreren Schichten Blei (ECAL) oder Eisen (HCAL) und szintillierenden Kacheln [3], die hintereinander aufgebaut sind. Beim Teilchendurchtritt durch die Blei-/Eisenplatten werden Sekundärteilchen erzeugt, die das szintillierende Material anregen, worüber indirekt die Energie der Primärteilchen gemessen werden kann [3]. Als letzte Station ist das Myonen-System aufgebaut, das die Myonen nachweisen soll, die bei den zu untersuchenden seltenen Zerfällen auftreten. Dafür sind fünf Stationen aufgebaut, die mit einer Kombination der Gase Kohlenstoffdioxid, Argon und Tetrafluormethan gefüllt sind. Über die Wechselwirkung der Myonen mit den Gasen können die Myonen nachgewiesen werden [3].

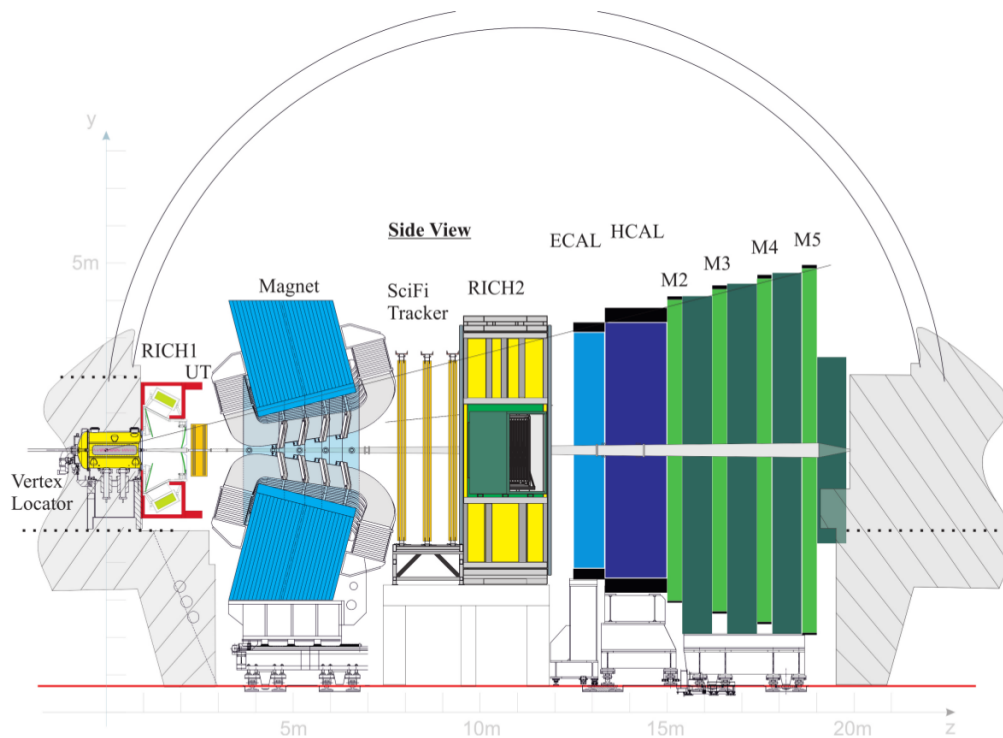


Abb. 1.1: Schematischer Aufbau des LHCb-Detektors nach dem geplanten Upgrade 2018/19 [4].

¹²Scintillating Fibre

¹³Electromagnetic CALorimeter

¹⁴Hadronic CALorimeter

1.2.2 Scintillating Fibre Tracker

Der Scintillating Fibre Tracker ist ein Spurdetektor aus szintillierenden Fasern. Aufgebaut ist er aus drei Stationen mit jeweils vier Detektorlagen (vergleiche Abbildung 1.2). Jede dieser Detektorlagen ist aus zwölf Modulen aufgebaut. Die beiden mittleren Detektorlagen sind entlang der Strahlachse um $\pm 5^\circ$ bezüglich der ersten und letzten Lage versetzt, um eine bessere Ortsauflösung zu erreichen [4].

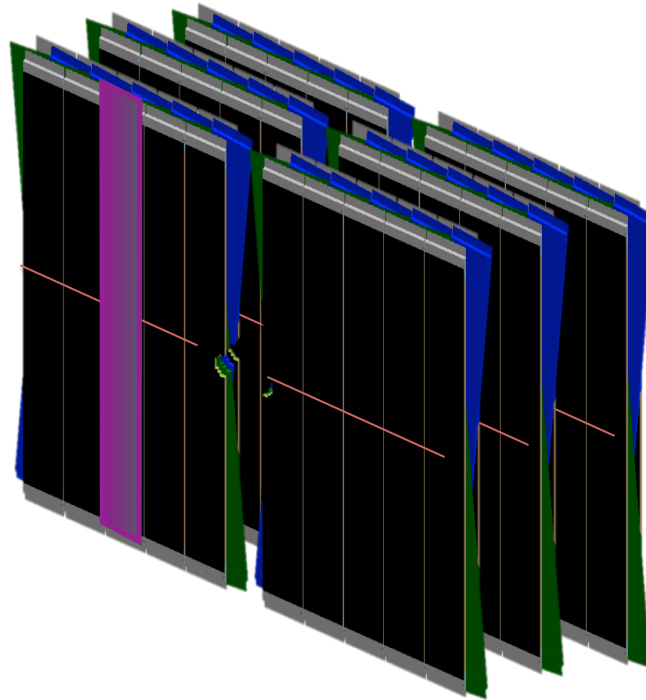


Abb. 1.2: Die drei Stationen des SciFi Trackers mit vertikalem Schnitt durch das Strahlrohr. Violett hervorgehoben ist eine Fasermatte. In blau und grün sind die beiden mittleren Detektorlagen dargestellt. Die orange horizontale Linie deutet die Position der Spiegel an. Die Abbildung wurde nach Vorlage von [11] erstellt.

Jedes Modul ist in zwei Teile aufgeteilt, die unabhängig voneinander funktionieren. Ein Teil des Moduls befindet sich unterhalb des Strahlrohres und das andere oberhalb. An einem Ende eines Teilmoduls werden vier SiPMs¹⁵, die jeweils 128 Aufnahmekanäle besitzen, Kühlelemente und Ausleseelektronik verbaut. Das andere Ende des Teilmoduls wird mit Spiegeln bestückt, welche die Photonen reflektieren, die nicht in Richtung des SiPMs geführt wurden.

¹⁵Silizium PhotoMultiplier

Die Maße eines Moduls betragen $0,52\text{ m} \times 5\text{ m}$, eine einzelne Fasermatte besitzt allerdings nur die Ausmaße $0,13\text{ m} \times 2,5\text{ m}$ [4]. So ergeben sich die zwei Reihen mit jeweils vier nebeneinander liegenden Matten pro Modul. Ein schematischer Überblick eines Teilmoduls ist in Abbildung 1.3 dargestellt.

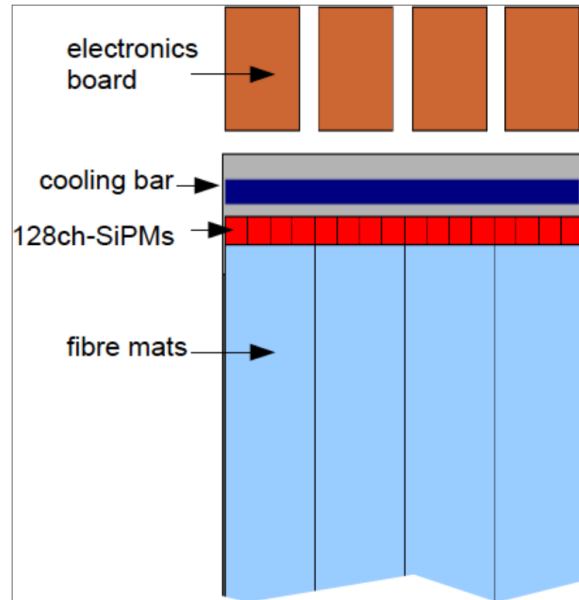


Abb. 1.3: Schematischer Aufbau des oberen Abschnitts eines Teilmoduls [12].

Jedes einzelne Modul wird von einem Kohlenstofffaser verstärkten Polymer, das durch zwei Nomex[®] honeycomb¹⁶ Schichten getrennt ist [4], umgeben, sodass es einfacher eingebaut und genutzt werden kann. In Abbildung 1.4 ist eine Fasermatte im Querschnitt dargestellt.

¹⁶Nomex ist ein eingetragenes Warenzeichen der E.I. du Pont de Nemours and Company (DuPont[®]).

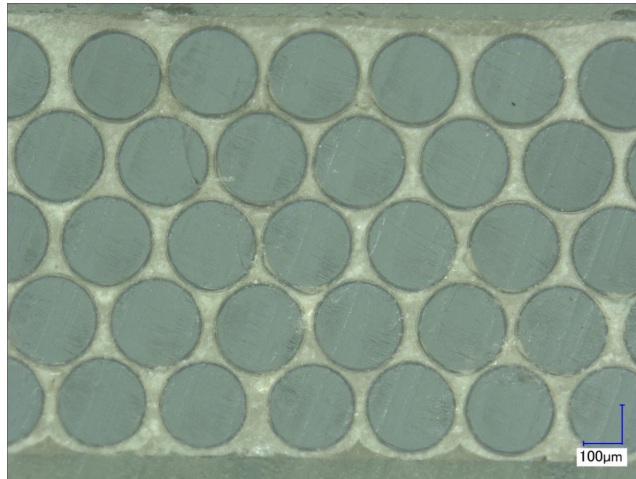


Abb. 1.4: Fasermatte bestehend aus fünf Lagen szintillierender Fasern [12].

Die Fasermatte besteht aus fünf Lagen übereinander gelegter Fasern, die miteinander verklebt sind. Jede einzelne Faser hat einen Durchmesser von $250\ \mu\text{m}$. Durchtritt ein Teilchen eine Faser, so werden die Polymermoleküle angeregt und senden optische Photonen aus. Diese werden über Totalreflexion zum SiPM geführt, wo sie registriert werden. Totalreflexion wird über das sogenannte „Cladding“ [4] ermöglicht, wobei der Kern der Faser mit zwei Schichten Kunststoff mit absteigenden Brechungsindices umgeben wird [4]. Die innere Claddingschicht besteht aus Polymethylmethacrylat mit einem Brechungsindex $n = 1,49$, während der Kern einen Brechungsindex von $n = 1,60$ aufweist. Die äußere Claddingschicht besteht aus einem flourhaltigen Polymer mit einem Brechungsindex von $n = 1,42$ [4]. Da die Fasermatten und die Optimierung ihrer das Thema der vorliegenden Bachelorarbeit sind, wird auf die Themen gesondert in Kapitel 2 eingegangen.

2 Fasermattenproduktion

2.1 Anforderungen

Zur Klärung der Anforderungen an die Fasermattenproduktion, wird hier zunächst auf die Eigenschaften einer Fasermatte eingegangen. Eine Fasermatte besteht aus fünf bis sechs Schichten einzelner szintillierender Fasern, die einen Durchmesser von 250 μm besitzen und miteinander verklebt werden. Da die szintillierenden Fasern vom Hersteller Kuraray [13] eine gewisse Toleranz bezüglich ihres Durchmessers aufweisen, kommt es zu Abweichungen der Dicke der Fasern über längere Strecken, die bei der Produktion berücksichtigt werden müssen. Zusätzlich zu diesen Schwankungen treten sogenannte Blobs¹ auf. Faserabschnitte mit Blobs können teilweise nicht zum Wickeln verwendet werden. Dabei ist nicht allein der Faserdurchmesser an einem Punkt entscheidend, sondern auch die Abschnitte vor und hinter der Verdickung. Als Richtwert für einen brauchbaren Faserabschnitt hat sich ein Faserdurchmesser von $<300 \mu\text{m}$ herausgestellt.

Die Zahl der Faserschichten richtet sich nach dem Einsatzort im SciFi Tracker. So werden Module, die in der Nähe des Strahlganges eingebaut werden, mit sechs Schichten produziert; bei weiter außen liegenden Modulen ist die Strahlenbelastung geringer und es genügen fünf Schichten. Grund dafür ist, dass Strahlenexposition die einzelnen Fasern beschädigt, sodass es zu einer geringeren Lichtausbeute kommt. Die extra Schicht wird benutzt, damit es trotz der Beschädigung zu einem verwertbaren Signal kommt.

2.2 Prototyp

Zur Produktion der Matten, wurde am Lehrstuhl E5 ein Prototyp für eine Maschine zur Fasermattenproduktion entwickelt, der in Abbildung 2.1 dargestellt ist. Beginnend mit der Abwickelspule (rechts im Bild) wird eine einzelne Faser über eine lose Rolle und einen Positioniermotor auf ein sich drehendes Rad gewickelt.

¹Blobs sind **lokale** Verdickungen der Faser.

2 Fasermattenproduktion

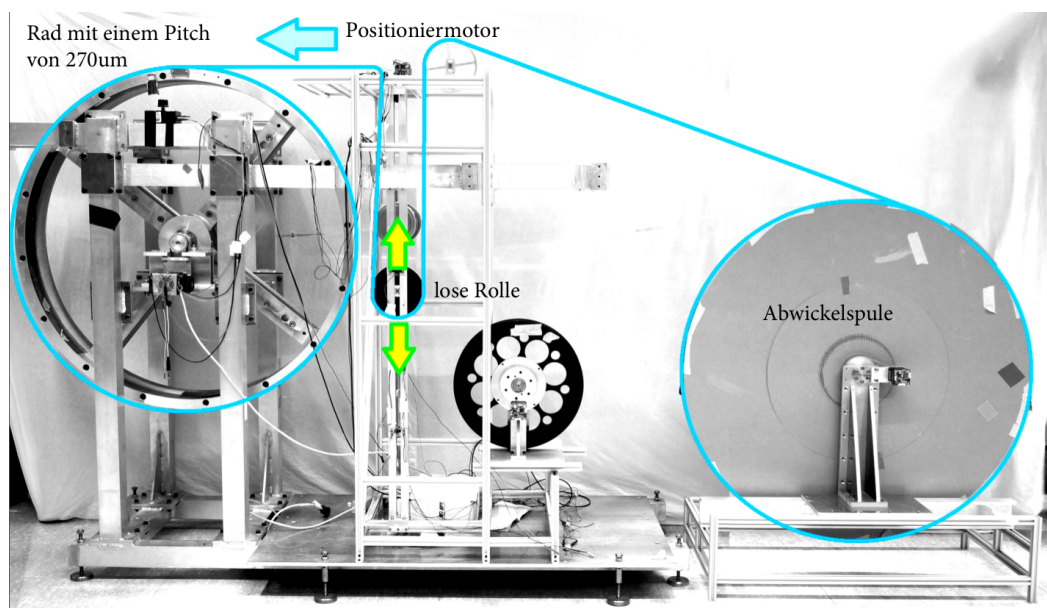


Abb. 2.1: Prototyp für eine Maschine zur Fasermattenproduktion der TU Dortmund.
Nach Vorlage von [4].

Das Wickelrad (links im Bild) ist ein Metallrad mit einem Durchmesser von 1 m, in das kleine Nuten eingefräst sind, sodass es sich um ein Gewinde mit einer Steigung $270\ \mu\text{m}$ handelt. Die Nuten dienen zur Führung der ersten Lage der zu wickelnden Fasermatte.

Eine Fasermatte wird zunächst auf eine Breite von ca. 16 cm gewickelt, bevor sie anschließend auf die gewünschten 13 cm zugeschnitten wird. Der Grund für dieses Vorgehen wird in Abschnitt 2.3 erläutert.

Die Geschwindigkeit des Wickelrades wird über ein Potentiometer reguliert, welches ein entsprechendes Signal an den angeschlossenen Schrittmotor leitet und ihn sowohl vorwärts als auch rückwärts drehen lässt. So sind manuelle Korrekturen möglich, wenn zum Beispiel eine Faser in die falsche Nut springt.

Die Faser benötigt für die Mattenbreite von ca. 16 cm eine seitliche Führung durch einen Positioniermotor, um die Position auf dem Wickelrad zu kontrollieren und eine zu hohe seitliche Zugspannung zu verhindern.

Für jede Umdrehung die das Wickelrad vollführt, bewegt sich der Positioniermotor $270\ \mu\text{m}$ weiter vom Startpunkt weg, sodass die Fasern nebeneinander liegen. Ob das Wickelrad sich gedreht hat, wird über eine Lichtschranke überprüft, die aus einem Spiegel, einer LED und einem Fototransistor besteht.

An einer festen Position neben dem Wickelrad befinden sich LED und Transistor nebeneinander. Fährt der, an dem Wickelrad angebrachte, Spiegel an diesem Aufbau vorbei, wird das von der LED ausgesendete Infrarotlicht vom Spiegel reflektiert und der Fototransistor schaltet durch. Das Signal wird auf den motoreigenen Eingang des Positioniermotors gelegt und lässt diesen einen seitlichen Schritt gehen.

Die Faser wird über eine Abwickelspule und eine lose Rolle zum Positioniermotor geführt. Die Abwickelspule wird von einem Schrittmotor betrieben, der über die motoreigenen Ein- und Ausgänge in Kontakt mit den LED-Sensorstationen steht, die sich ober- und unterhalb der Rolle befinden. Diese Sensorstationen bestehen jeweils aus einer LED und einem gegenüberliegenden Fototransistor. Die Rolle bewegt sich zwischen den LEDs.

Wird die Faser abgewickelt, so verändert sich, je nach Geschwindigkeit mit der die Faser vom Wickelmotor gezogen wird, die Position der losen Rolle. In diesem Zuge schaltet ein Fototransistor durch, was ein Signal an den Eingang des Motors der Abwickelspule liefert. Dieses Signal bedeutet der Abwickelspule langsamer oder schneller zu wickeln, sodass immer die gleiche mechanische Zugspannung auf die am Wickelrad ankommende Faser wirkt und diese dort optimal verarbeitet werden kann.

2.3 Produktion

Hier soll die konkrete Produktion einer Fasermatte erklärt werden, wobei bereits erste Hinweise auf Probleme und Komplikationen gegeben werden. Eine Qualitätskontrolle der Faser ist aufgrund der in Abschnitt 2.1 genannten variierenden Faserdurchmesser notwendig. Die Qualitätsüberprüfung wird durch die Faserkontrolle durchgeführt, bei welcher der Faserdurchmesser mit Hilfe eines 2-Achsen-Laser-Messsystems (BETALasermike Accuscan 5010 [14]) bestimmt wird.

Zudem wird die Faser auf Beschädigungen untersucht. Als Konsequenz werden unbrauchbare Faserabschnitte entfernt, sodass schlussendlich eine Faserrolle vorliegt, die für die Produktion genutzt werden kann. Diese Rolle wird als Abwickelspule verwendet und die Faser entsprechend der Beschreibung von Abschnitt 2.2 in den Prototypen eingesetzt.

Bevor die Matte gewickelt werden kann ist erst eine Vorbereitung des Wickelrades notwendig. Dafür werden das Rad und die Führungselemente gereinigt, um sie von alten Kleberresten sowie Schmutz und Staub zu befreien. Ist der Prototyp gereinigt, wird das Rad versiegelt, um Unebenheiten auszugleichen, bevor es mehrmals (5-6 Schichten) mit Trennmittel bestrichen wird. Das Trennmittel verhindert das Anhaften der mit Epoxiharz verklebten Fasermatte am Rad.

Die Faser wird zu Beginn des Wickelprozesses mit einer Schraube am Rad fixiert und in die erste Nut gelegt. Der Wickelradmotor wird langsam über das Potentiometer zum Starten gebracht und die Position der Faser mit Hilfe einer Digitalkamera kontrolliert. Verspringt die Faser und gerät aus der für sie vorgesehenen Nut, wird das Rad angehalten, die Lichtschranke ausgeschaltet und die Faser wieder in die richtige Nut gelegt, bevor die Produktion weiterlaufen kann.

Die Lichtschranke muss ausgeschaltet werden, da beim Rückwärtsfahren des Wickelrades der Spiegel an der Steuereinheit vorbei kommen könnte und damit der

3 Zentrale Steuerung des Prototyps

3.1 Motivation

Der in Kapitel 2 beschriebene Aufbau weist in der Praxis Probleme auf, welche die Produktion von Fasermatten erschweren.

Prinzipiell ist die dezentralisierte Steuerung der einzelnen Motoren, die nur auf gewisse Ereignisse reagieren, für eine flexible Nutzung des Prototyps suboptimal. Die dezentralisierte Steuerung besteht aus drei autonomen Schrittmotoren (Spule, Positionierung, Rad), dem Spiegel-Transistor-System und einem Potentiometer. Dies verhindert die Möglichkeit der Dokumentation, die verschiedene Aspekte festhalten kann. Beispielsweise welche Strecke an Faser schon gewickelt worden ist oder zu sehen wie breit die aktuelle Matte ist. Ein weiteres Problem ist die Steuerung des Positioniermotors über die Lichtschranke. Der aktuelle Aufbau ist nicht in der Lage zu unterscheiden, ob sich das Wickelrad vor- oder rückwärts dreht.

Eine zentrale Steuerung ermöglicht eine Dokumentation und somit einen einfacheren und flexibleren Umgang mit dem Prototypen. Wird eine zentrale Steuereinheit verbaut, die die anderen Komponenten kontrolliert und ihnen Anweisungen gibt, können über diese Steuereinheit auch die Informationen, die jedes einzelnen Prototypelement bereitstellt, verarbeitet und zielgerichtet genutzt werden. Beispielsweise kann die Information der Drehrichtung des Rades, welche durch das Potentiometer gegeben ist, von der zentralen Einheit verarbeitet werden und in Kombination mit der Information des Spiegel-Transistor-Systems zur Steuerung des Positioniermotors dienen. Die zentrale Steuerung kann erkennen in welche Richtung sich das Rad dreht und bewegt den Positioniermotor entsprechend nach links oder rechts.

3.2 Voraussetzungen

3.2.1 Hardware

Die vorliegende Hardware besteht aus den zwei zu steuernden Schrittmotoren PANdrive PD3-110-42-232 (Positioniermotor) und PANdrive PD3-109-57-RS (Wickelradmotor) von Trinamic [15]. Jeder Motor besitzt einen Controller mit einem analogen und zwei digitale Eingängen, an die Sensoren angeschlossen werden können.

3 Zentrale Steuerung des Prototyps

Zusätzlich kann ein Programm auf den Controller überspielt werden, sodass der Motor autonom arbeiten kann. Weitere Hardwarekomponenten sind das Potentiometer und die Lichtschranke, welche in Abschnitt 3.4 näher spezifiziert werden. Die Geräte zur Steuerung der losen Rolle und des Abwickelspule werden nicht erwähnt, da deren Aufbau nicht verändert wird. Das wichtigste Element des neuen Konzepts ist die zentrale Steuerung. Hierfür wird ein Raspberry Pi benutzt, welches ein günstiger Minicomputer ist, der viele Schnittstellen, wie USB- oder HDMI-Anschlüsse bietet und über GPIO¹-Anschlüsse vielseitig genutzt werden kann [16].

Für die Bachelorarbeit wird ein Raspberry Pi Model B Revision 2 verwendet, das mit dem System-On Chip Broadcom BCM2835 [17] arbeitet. Das Betriebssystem des Raspberry Pis ist das vom Hersteller empfohlene [16] Raspbian (Debian Wheezy) Version January 2014. Ziel ist es, dass das Raspberry Pi die Signale des Spiegel-Transistor-Systems und des Potentiometers ausliest und mit Hilfe dieser Informationen die Schrittmotoren steuert. Somit kann eine zentralisierte Steuerung erfolgen, wobei das Raspberry Pi als zentrales Element alle Signale empfängt und alle Motoren steuert. Gründe, für das Raspberry Pi und gegen ein ähnliches System (zum Beispiel Arduino), finden sich vor allem im erhöhten Funktionsumfang (diverse Schnittstellen, Möglichkeit eines Betriebssystems etc.) und der sehr guten Dokumentation im Internet. Es sind bereits viele Bibliotheken vorbereitet, die ohne große Komplikationen eingebunden werden können.

3.2.2 Software

Als Programmiersprache wird C++ verwendet, da sie systemunabhängig, sehr gut dokumentiert und weit verbreitet ist. Bei der Auswahl der Software werden erste Probleme deutlich: Die Steuerung der Schrittmotoren erfolgt über die vom Hersteller selbst entwickelte Programmiersprache TMCL², die in Kombination mit einer Entwicklungsumgebung geliefert wird, welche nur auf Windows Systemen arbeitet [15]. Da das verwendete Betriebssystem Raspbian eine Linux Distribution ist, muss hier eine Lösung gefunden werden. Demzufolge muss erst die Funktionsweise von TMCL verstanden werden, damit die gleichen Funktionen in C++ aufgebaut werden können, so dass das Raspberry Pi die Schrittmotoren steuern kann.

Die Programmierung der GPIO-Anschlüsse ist mit C++ möglich, da es eine sehr gut dokumentierte und aktuelle Bibliothek [18] gibt. Zudem kann das Raspberry Pi per SSH angesprochen werden, so können auch noch Änderungen am Raspberry Pi vorgenommen werden, obwohl es schon im Prototypen eingebaut ist.

¹General Purpose Input Output

²Trinamic Motion Control Language

3.3 Zielsetzung

3.3.1 Aufbau

Erstes Ziel ist es, wie in Abschnitt 3.1 beschrieben, den Aufbau mit der oben beschriebenen Hardware so zu verändern, dass die Steuerung in Zukunft zentral über das Raspberry Pi abläuft. Um dieser Veränderung zu erklären, werden ein Ist-Schema (Abbildung 3.1) und ein Soll-Schema (Abbildung 3.2) herangezogen, die den Umbau verdeutlichen.

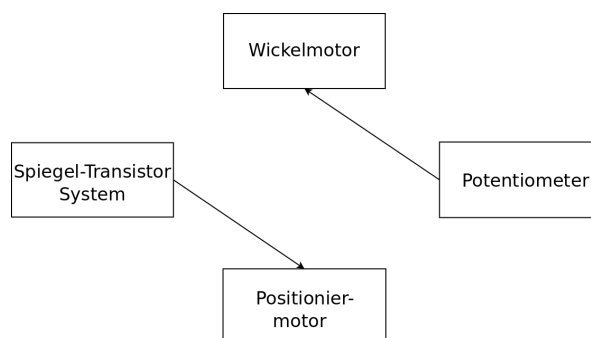


Abb. 3.1: Schematischer Aufbau des Prototyps vor dem Umbau.

Im Ist-Schema (Abbildung 3.1) ist der Aufbau der Komponenten vor dem Umbau dargestellt. Es ist zu sehen, dass das Spiegel-Transistor-System nur ein Signal an den Positioniermotor gibt, der darauf reagiert. Analog gibt das Potentiometer einen Spannungswert an den Wickelmotor weiter. Eine Verbindung zwischen den beiden Gruppen besteht nicht.

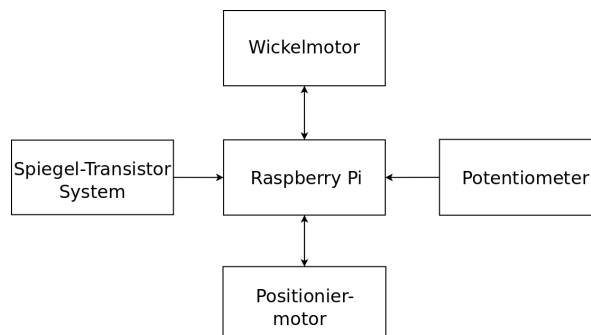


Abb. 3.2: Schematischer Aufbau des Prototyps nach dem Umbau.

Das Soll-Schema (Abbildung 3.2) zeigt die Verbindung der Komponenten nach dem Umbau. In 3.2 ist das Raspberry Pi das zentrale Element, das Signale vom Spiegel-Transistor-System und vom Potentiometer erhält. Zudem gibt es Steuerbefehle an den Positionier- und Wickelmotor. Weitere Verbindungen der einzelnen

Komponenten gibt es nicht. Damit wird gewährleistet, dass die gesamte Steuerung vom Raspberry Pi übernommen wird.

3.3.2 Funktionen

Da mit der Zentralisierung eine Erleichterung und Verbesserung der Steuerung erfolgen soll, sind auch einige funktionale Weiterentwicklungen formuliert. Zunächst sollen alle bisherigen Funktionen möglich sein. Diese sind:

- Die Steuerung (Geschwindigkeit, Vor- und Rückwärtsbewegung, Stoppen) des Wickelrades mittels eines Drehreglers (zum Beispiel ein Potentiometer).
- Das Spiegel-Transistor-System erkennt den Spiegel beim Vorbeifahren und bewegt den Positioniermotor einen Schritt weiter.
- Die Steuerung über ein Programm³.

Bei einem Prototypen können sich schnell die Anforderungen ändern, auf die unkompliziert eingegangen werden muss. Deswegen ist es nicht von primärer Bedeutung, konkrete Aufgaben durch die Steuerung zu erledigen, sondern vielmehr steht im Vordergrund ein Grundgerüst bereitzustellen, das flexibel auf zukünftige Aufgabenstellungen reagieren kann. Nichtsdestotrotz sollen einige zusätzliche Funktionen realisiert werden, um die Funktionalität dieses Grundgerüsts zu beweisen. Diese sind:

- Das Spiegel-Transistor-System kann entscheiden, ob der Positioniermotor einen Schritt vor oder zurück bewegt werden muss.
- Matten einer bestimmten Breite sollen automatisiert gewickelt werden können.

Weitere Möglichkeiten werden später in Abschnitt 3.5 diskutiert.

3.4 Umsetzung

3.4.1 Aufbau und Verschaltung der Hardware

Mit der in Abschnitt 3.2.1 beschriebenen Hardware wird eine Umsetzung des Aufbaus realisiert. Die Schaltpläne werden mit der Software Eagle [19] erstellt. Als erstes wird das Spiegel-Transistor-System betrachtet. Der Schaltplan für das System ist in Abbildung 3.3 dargestellt.

³Positioniermotor und Wickelradmotor müssen einzeln gestoppt werden können. Korrekturschritte beim Positioniermotor müssen möglich sein. Eine Geschwindigkeitsabfrage des Wickelradmotors und die Abfrage der aktuellen Position des Positioniermotors müssen gewährleistet sein.

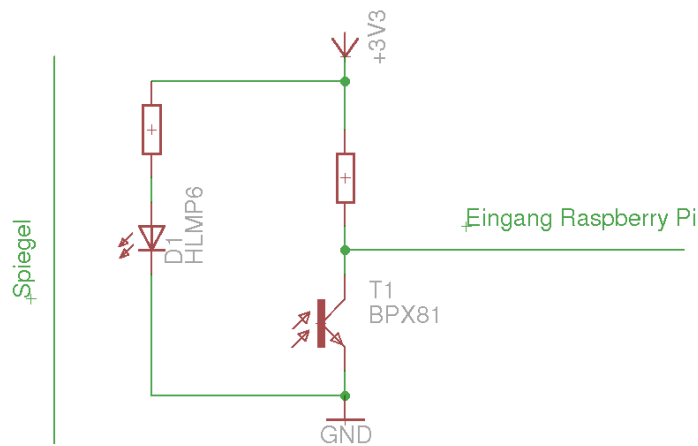


Abb. 3.3: Schaltplan für das Spiegel-Transistor-System.

Der Schaltplan verdeutlicht die Funktionsweise des Systems. Die LED leuchtet dauerhaft und sobald der Spiegel an der LED vorbei läuft, reflektiert dieser das Infrarotlicht, welches auf den Fototransistor trifft. Dort schaltet der Transistor durch und die Spannung in Höhe 3,3 V, die beim Sperren des Transistors anliegt, fällt auf 0 V ab. Das Spannungssignal wird direkt an einen GPIO-Pin des Raspberry Pis gelegt (vergleiche Abbildung 3.4), der softwareseitig ausgelesen wird.

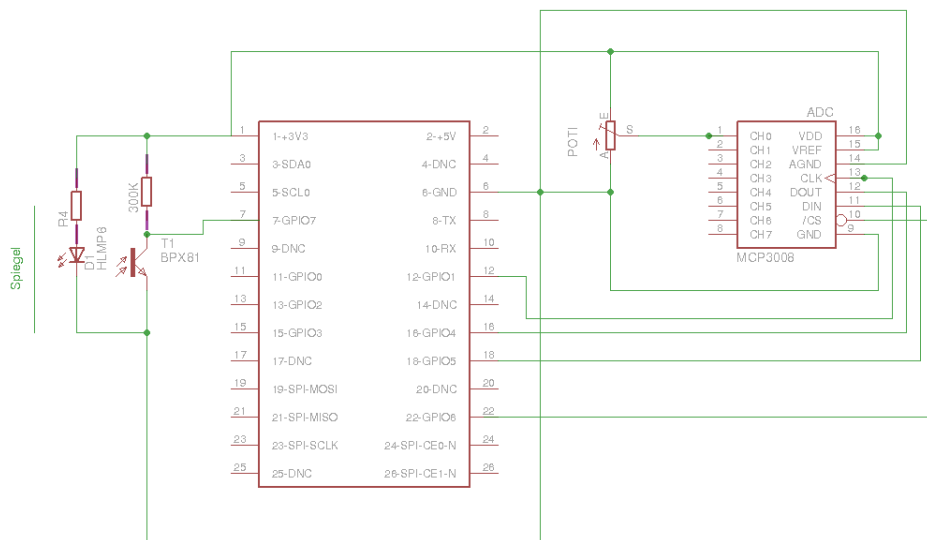


Abb. 3.4: Verschaltung des Raspberry Pis, die die Verbindung von Raspberry Pi, ADC und Spiegel-Transistor System darstellt. Die Bezeichnung für die Eingänge des Raspberry Pis beziehen sich auf eine ältere Version des Raspberry Pis. Dies ist für den vorliegenden Fall unproblematisch, da die benutzen Eingänge in beiden Versionen identisch sind. Beide Fassungen sind in [20] dargestellt.

3 Zentrale Steuerung des Prototyps

Die Signalaufnahme des Potentiometers gestaltet sich schwieriger, da ein analoges Signal interpretiert werden muss. Das Potentiometer funktioniert wie ein Spannungsteiler, der in Abbildung 3.5 dargestellt ist.

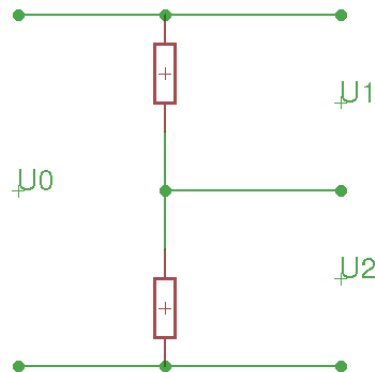


Abb. 3.5: Schaltplan für das Potentiometer (Spannungsteiler).

Dieser Spannungsbereich kann von den GPIOs nicht verarbeitet werden, da es sich dabei um digitale Eingänge handelt. Das Raspberry Pi stellt keine analogen Eingänge zur Verfügung, weswegen eine externe Schaltung benötigt wird, die das analoge Signal des Potentiometers in ein digitales Signal umwandelt. Die Schaltung und der Anschluss an das Raspberry Pi sind in Abbildung 3.4 dargestellt.

Als Analog-Digital-Wandler (ADC) wird ein MCP3008 von Microchip [21] verwendet. Dieser hat eine Auflösung von 10 Bit und verfügt über eine SPI⁴ Schnittstelle, welche mit dem Raspberry Pi kompatibel ist. Der Aufbau, der in Abbildung 3.4 dargestellt ist, wird nach einer Vorlage [23] erstellt. Damit ist die Umrechnung der analogen Eingangsspannung auf ein digitales Signal, das an das Raspberry Pi gelegt wird, möglich.

Die Steuerung der Schrittmotoren erfolgt über eine RS232 Schnittstelle. Dafür werden die beiden Schrittmotoren an einen RS232-Hub angeschlossen, welcher per USB mit dem Raspberry Pi verbunden ist. Somit können die Schrittmotoren softwareseitig gesteuert werden.

3.4.2 Testmessungen an der Hardware

Um eine korrekte Funktionsweise zu garantieren, werden mögliche Fehlerquellen untersucht. Zum einen wird das Signal des Spiegel-Transistor-Systems direkt an einen GPIO Pin gelegt, weswegen die Funktionsweise von diesem überprüft werden muss. Zudem ist es sinnvoll den ADC zu überprüfen, sodass es keine Fehler beim Auslesen des Potentiometers gibt.

⁴Serial Peripheral Interface [22]

Der GPIO des Raspberry Pi unterscheidet zwischen den Zuständen HIGH und LOW, abhängig von der am Pin anliegenden Spannung. Dabei ist laut Spezifikation [24] ein HIGH Signal ab einer Spannung von 1,3 V und ein LOW Signal unterhalb einer Spannung von 0,8 V garantiert.

Diese Aussage wird validiert, indem eine variable Eingangsspannung, die mit einem Multimeter kontrolliert wird, an einen GPIO Pin gelegt und das vom Raspberry Pi generierte Signal (HIGH oder LOW) ausgelesen wird. Dabei wird jeweils einmal von einer hohen Spannung kommend die Spannung verringert (HIGH zu LOW) und andersherum von einer niedrigen Spannung kommend die Spannung erhöht (LOW zu HIGH). Die Ergebnisse für jeweils 3 Messreihen sind in Abbildung 3.6 für HIGH zu LOW zu LOW zu HIGH dargestellt.

Der Bereich, indem der GPIO-Pin instabil ist wird in Abbildungen 3.8 und 3.9 genauer betrachtet. Es lässt sich erkennen, dass die vorausgesagten garantierten HIGH und LOW Werte für die entsprechenden Spannungen in jeder Messung erreicht werden. Zum LOW Signal ist die angegebene Grenze sogar großzügig gewählt, hier wäre eine Spannung von 1,1 V als untere Grenze vertretbar. Ein direkter Anschluss eines Spannungssignals an einen GPIO-Pin liefert mit den angegebenen Grenzen ein reproduzierbares und korrektes Ergebnis.

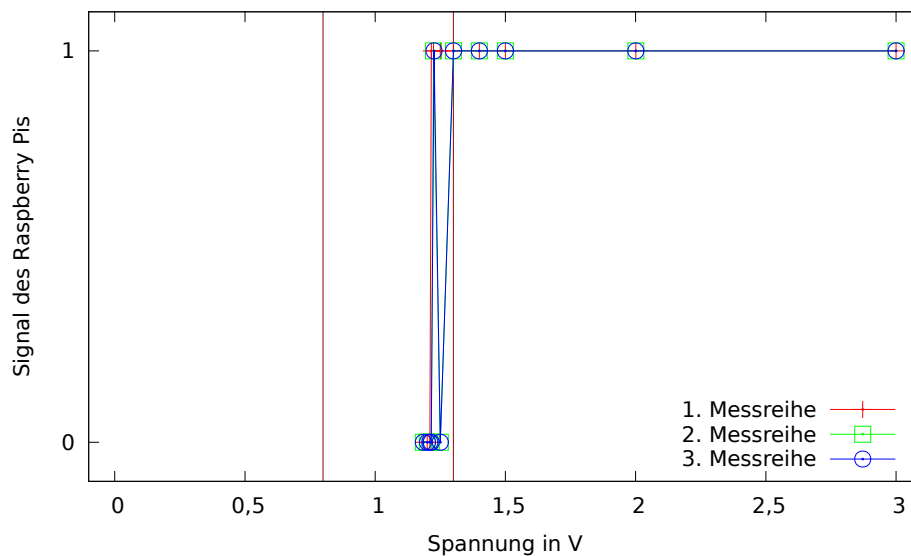


Abb. 3.6: Signalausgang des Raspberry Pis für verschiedene Eingangsspannungen beginnend bei maximaler Spannung. Die vertikalen Markierungen zeigen die, durch die Spezifikation gegebenen, Schwellenwerte. Für eine bessere Übersicht werden die Datenpunkte verbunden.

3 Zentrale Steuerung des Prototyps

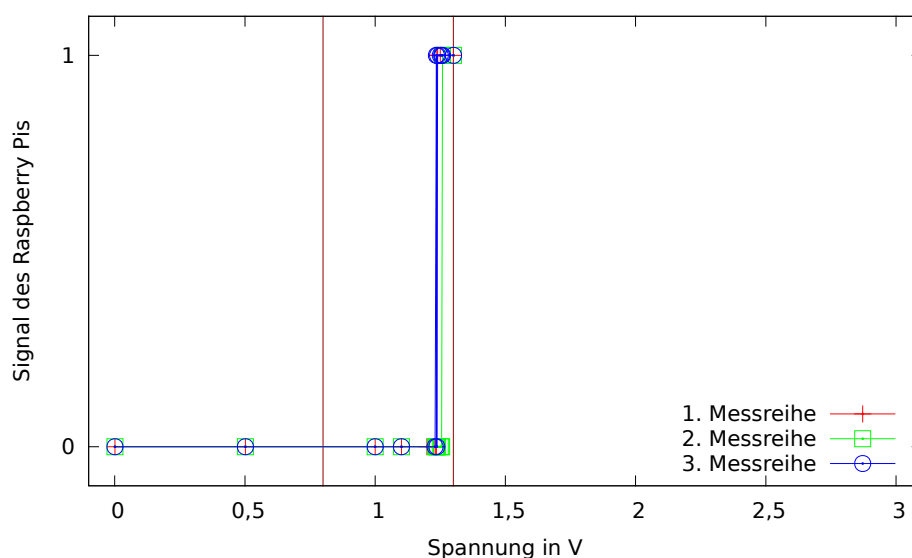


Abb. 3.7: Signalausgang des Raspberry Pis für verschiedene Eingangsspannungen beginnend bei minimaler Spannung. Die vertikalen Markierungen zeigen die, durch die Spezifikation gegebenen, Schwellenwerte. Für eine bessere Übersicht werden die Datenpunkte verbunden.

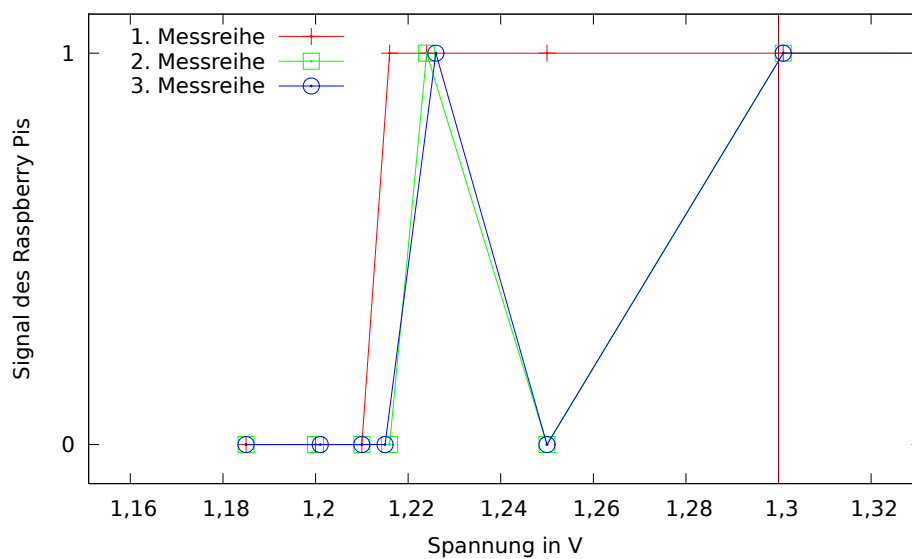


Abb. 3.8: Signalausgang des Raspberry Pis für verschiedene Eingangsspannungen beginnend bei maximaler Spannung auf den Bereich 1,15 V bis 1,33 V fokussiert. Die vertikale Markierung zeigt den, durch die Spezifikation gegebenen, Schwellenwert. Für eine bessere Übersicht werden die Datenpunkte verbunden.

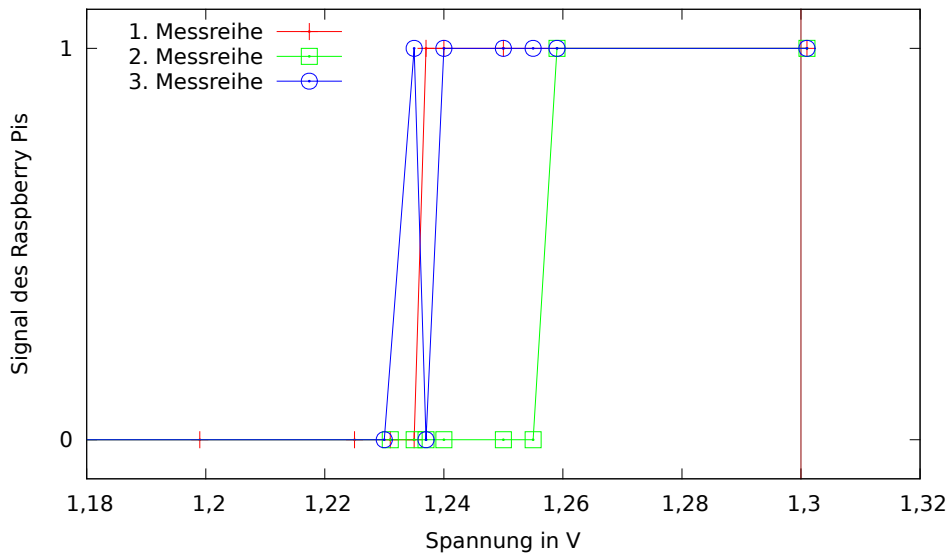


Abb. 3.9: Signalausgang des Raspberry Pis für verschiedene Eingangsspannungen beginnend bei minimaler Spannung auf den Bereich 1,18 V bis 1,33 V fokussiert. Die vertikale Markierung zeigen den, durch die Spezifikation gegebenen, Schwellenwert. Für eine bessere Übersicht werden die Datenpunkte verbunden.

Das zweite Signal, das verarbeitet wird, ist das des ADC. Die Funktionsweise des ADC wird überprüft, indem eine variable Eingangsspannung (über ein Multimeter kontrolliert) an den ADC gelegt wird. Diese und das digitale Ausgangssignal des ADCs, welches in einen Spannungswert umgerechnet wird, werden gegeneinander aufgetragen und eine Ausgleichsrechnung liefert eine Aussage über die Qualität des ADC.

Die Umrechnung vom digitalen Signal S_{digi} zur analogen Spannung $V_{\text{gerechnet}}$ erfolgt über die Formel

$$V_{\text{gerechnet}} = S_{\text{digi}} \frac{3,3 \text{ V}}{1024}.$$

Sie ergibt sich über die Auflösung des ADCs, die 10 Bit beträgt und somit 1024 verschiedene Werte zulässt und dem Normierungsfaktor 3,3 V. Die 3,3 V sind durch die Spannungsversorgung des Raspberry Pi gegeben. Das wichtigste Kriterium für die Verwendung von 3,3 V ist, dass es bei höheren Spannungen zur Zerstörung von elektronischen Bauteilen, die an die GPIO-Pins angeschlossen sind, kommen kann [16].

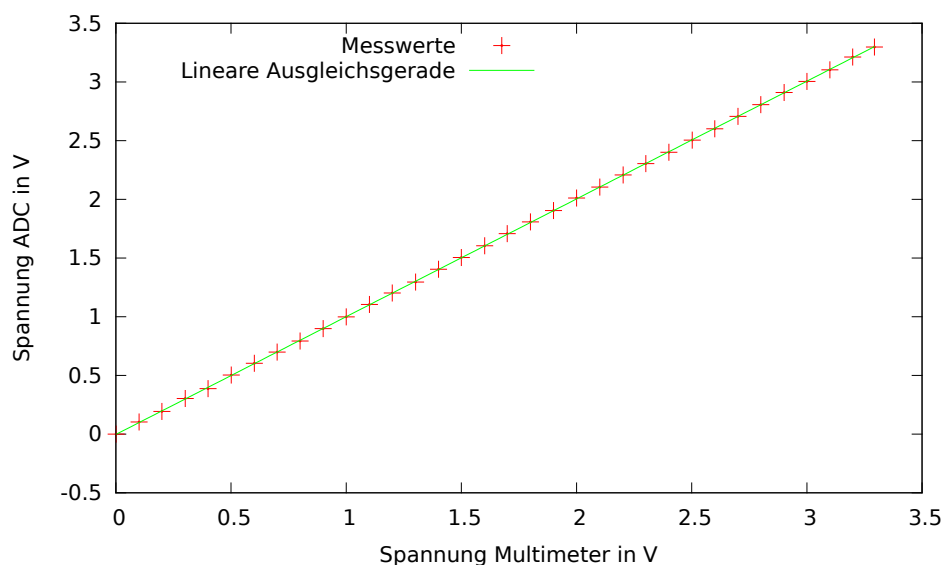


Abb. 3.10: Beurteilung der Qualität des ADC durch den Vergleich von Eingangsspannung zur ausgegebenen Spannung.

Abbildung 3.10 zeigt einen linearen Zusammenhang der beiden aufgetragenen Spannungen, der mittels GnuPlot [25] bestimmte lineare Zusammenhang ist:

$$V_{\text{gerechnet}}(V_{\text{Eingang}}) = (1,0032 \pm 0,0007) \cdot V_{\text{Eingang}} - (0,003 \pm 0,001) \text{ V}$$

Eine ideal funktionierender ADC liefert den gleichen Spannungswert, wie die Spannungsquelle. Im Diagramm ergäbe sich eine Gerade mit Steigung

$$m = \frac{V_{\text{gerechnet}}}{V_{\text{Eingang}}} = 1$$

Analog wäre der Y-Achsenabschnitt $V_{\text{gerechnet}}(0 \text{ V}) = 0$. Demnach reproduziert der ADC die anliegende Spannung mit guter Genauigkeit (Abweichungen im Promillebereich).

3.4.3 Software

Der strukturelle Aufbau des Softwarekonzepts ist in Abbildung 3.11 dargestellt.

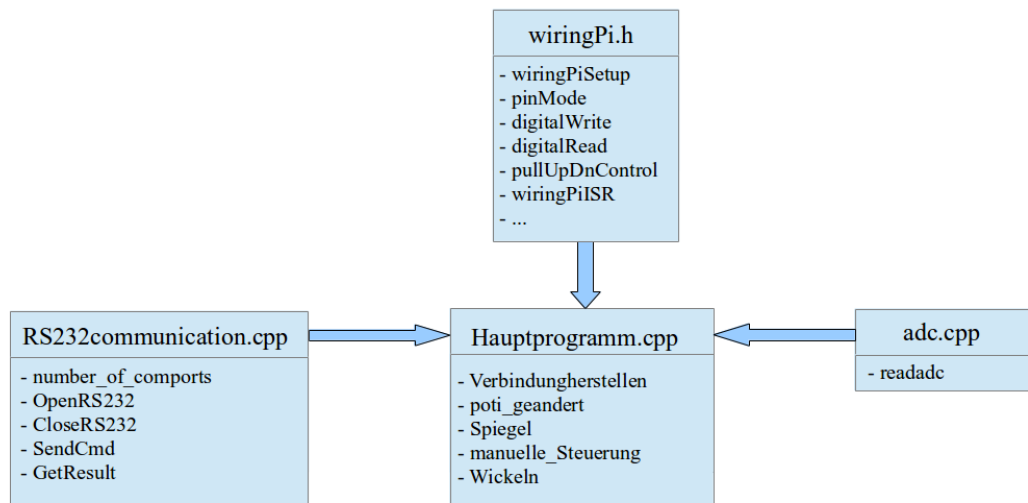


Abb. 3.11: Struktureller Aufbau des Softwarekonzepts.

Aus der Grafik geht hervor, dass die im Rahmen dieser Arbeit erstellte `Hauptprogramm.cpp` zentraler Baustein der Software ist, der auf verschiedene andere Elemente zurückgreift. Dabei ist im Kopf eines jeden Elements der Grafik der Name der Datei geschrieben und der zweite Abschnitt zählt die, in dem Programm definierten, Funktionen auf. Die Dateien sind im Einzelnen:

- **RS232communication.cpp**

Diese Bibliothek ermöglicht die Kommunikation mit den Schrittmotoren über die RS232-Schnittstelle. Zudem werden hier die wichtigsten Funktionen zur Motorsteuerung definiert, diese sind `SendCmd` und `GetResult`. Die `SendCmd`-Funktion leistet die Umwandlung des C++-Befehls in eine Bitfolge, wie sie durch einen TMCL-Befehl erzeugt werden würde und ermöglicht somit die Steuerung der Motoren. Zu jedem Befehl muss eine Antwort des Motors über `GetResult` ausgelesen werden, bevor ein neuer Befehl gesendet wird, um Fehlinterpretationen zu vermeiden. Der Motor wird entsprechend über `GetResult` ausgelesen, was nicht nur den sauberen Programmablauf ermöglicht sondern auch Informationen bezüglich Geschwindigkeit, Position und weiteren Eigenschaften⁵ des Motors erhebt.

`RS232communication.cpp` wurde nach einer Vorlage [26] von Trinamic entwickelt.

⁵Vollständige Aufzählung abfragbaren Informationen sind in der Bedienungsanleitung [15] zu finden.

- `adc.cpp`

In der, im Zusammenhang mit dieser Arbeit entwickelten, Bibliothek `adc.cpp` wird die Funktion `readadc` definiert, die das Auslesen des ADCs und damit die Benutzung des Potentiometers ermöglicht. Da der ADC laut Datenblatt [21] acht mögliche Kanäle (Channel 0-7) besitzt, muss hier der zu benutzende Kanal ausgesucht werden, damit das Signal richtig verarbeitet wird. Entsprechend dieser Verarbeitung wird das analoge Signal zu einem digitalen gewandelt und diesem in Form eines Integer-Wertes zurückgegeben. Damit eine Kommunikation zwischen Raspberry Pi und ADC funktioniert, bedarf es der Bibliothek `wiringPi.h`.
- `wiringPi.h`

Die Bibliothek `wiringPi.h` stellt alle Funktionen zur Arbeit mit den GPIOs zur Verfügung. Insbesondere sind dies `digitalWrite` und `digitalRead`, die das Schreiben und Auslesen der Pins ermöglichen. Hinzu kommen nützliche Zusatzfunktionen wie die Möglichkeit Pull-Down-Widerstände zu aktivieren, was zum Beispiel für die schnelle Senkung der Spannung des Spiegelpins genutzt wird. Damit kann auch bei hohen Drehgeschwindigkeiten (dies entspricht einer kurzen Zeit, in der der Transistor den Spiegel sieht) das Spiegel-Transistor-System zuverlässig ausgelesen werden. `wiringPi.h` ist sehr gut dokumentiert und wird aktiv betreut [18].
- `Hauptprogramm.cpp`

Die `Hauptprogramm.cpp` besteht aus mehreren Funktionen. Diese sind:

 - `poti_geandert`

`poti_geandert` nutzt die eingangs erläuterte `readadc`-Funktion aus der Bibliothek `adc.cpp`, um zu entscheiden, ob sich der Wert des Potentiometers geändert hat. Dafür werden Grenzen eingeführt, die überschritten werden müssen beziehungsweise nicht überschritten werden dürfen, damit eine Änderung des Potentiometers akzeptiert wird. Damit sind zwei weitere Anforderungen erfüllt: Zum einen soll verhindert werden, dass sich, durch Instabilitäten des Potentiometers und der Kabelverbindungen, kleine Änderungen direkt auf die Geschwindigkeit des Motors auswirken und es somit zu unkontrollierbaren Geschwindigkeitsänderungen kommt. Zum anderen soll mit der oberen Grenze für die Änderung des Potentiometerwertes eine Übersteuerung des Wickelradmotors verhindert werden. Will der Anwender den Motor zu schnell beschleunigen, müssten aufgrund des schweren Rads Drehmomente realisiert werden, die die Mechanik des Motors überfordern würde. Befindet sich die Änderung des Wertes innerhalb dieser Grenzen, so reagiert der Schrittmotor auf den veränderten Wert und steuert die so neu eingestellte Geschwindigkeit an. Andernfalls wird die Veränderung des Potentiometers ignoriert.

– Spiegel

Die Funktion `Spiegel` gibt an, ob der Fototransistor geschaltet hat und erhöht im Falle der Durchschaltung einen Zähler. Damit kann später die Anzahl der Umdrehungen gemessen werden, was einen Rückschluss auf die Breite der Matte zulässt.

– manuelle_Steuerung

Die Funktion `manuelle_Steuerung` ist die erste der beiden Funktionen, die in der `main`-Funktion in einem separaten Thread aufgerufen werden. Ein Thread beschreibt die Aufteilung bestimmter Ressourcen in einem Programm, die es ermöglicht, dass mehrere Funktionen gleichzeitig laufen können. In einem Programm, das keine Threads enthält, wird die `main`-Funktion sequentiell ausgeführt. Dies führt zu Problemen, wenn mehrere Funktionen gleichzeitig arbeiten. Hier erfolgt die Steuerung des Wickelradmotors über das Potentiometer. Um jederzeit eingreifen zu können, ist eine manuelle Steuerung gewünscht, die die Steuerung des Motors übernimmt. Dafür werden die beiden Steuerungen in zwei Threads gestartet. Beide Programme laufen nun parallel und das Eingreifen via manueller Steuerung ist jederzeit möglich [27].

Die manuelle Steuerung wird durch ein Kommandozeilenmenü realisiert, das bestimmte Aktionen und Funktionen vereinigt. Es kann die aktuellen Positionen der Motoren auslesen, sie stoppen lassen und bestimmte Aktionen (zum Beispiel Matten wickeln) automatisiert durchführen. Sie stellt den größten Fortschritt der Zentralisierung dar, weil durch sie eine bequeme Steuerung vieler Aktionen möglich ist. Zudem lassen sich weitere Funktionen in Zukunft einfach integrieren. Nähere Erläuterungen werden in Kapitel 4 gegeben.

– Wickeln

Die Funktion `Wickeln` ermöglicht die Steuerung des Wickelradmotors mittels Potentiometer. Dazu wird eine Endlosschleife benutzt, in der ständig das Potentiometer ausgelesen und die Geschwindigkeit des Motors entsprechend angepasst wird. An dieser Stelle ist zu erwähnen, dass die Funktion `Spiegel` hier über einen sogenannten Interrupt aufgerufen wird. Ein Interrupt ist eine kurzfristige Unterbrechung des laufenden Programms mit dem Ziel, eine zeitkritische Funktion aufzurufen [28]. Zeitkritisch ist die Funktion `Spiegel` deswegen, weil der Transistor bei hohen Drehgeschwindigkeiten nur sehr kurz schaltet. Dieses Vorgehen hat sich im Laufe der Bachelorarbeit als zielführend erwiesen. Zunächst wurde der Spiegel über eine einfache if-Abfrage kontrolliert, die bei hohen Drehgeschwindigkeiten des Motors versagte. Unter Benutzung des Interrupts wird das Spiegelsignal zuverlässig erkannt.

3.5 Erweiterungsmöglichkeiten

Mit der, im Rahmen der Bachelorarbeit entwickelten, Steuerung lassen sich Motorbefehle mit wenig Programmieraufwand benutzen und sind somit flexibel einsetzbar. Ebenso können Motorbefehle, die noch nicht eingebaut sind, bequem hinzugefügt werden.

Neben der einfachen Steuerung der Motoren ergibt sich durch die programmgestützte Überwachung aller Komponente die Möglichkeit der Dokumentation. Programmiertechnisch ist die Dokumentation unproblematisch. Ein Schreiben und Lesen von Dateien ist über C++ schnell realisiert und stellt ein probates Mittel dar, um aktuelle Vorgänge aufzuzeichnen. Es könnte zum Beispiel im Programm errechnet werden wie viel Kilometer Faser vom Anfangsbestand noch auf der Rolle sind. Da durch die Faserkontrolle Stellen mit Blobs bekannt sind, kann mit dieser Information das Raspberry Pi die Geschwindigkeit an der entsprechenden Stelle verlangsamen und somit das Herausspringen der Faser aus der Nut vermeiden. Über die Internetverbindung wäre es auch möglich, Personen über den aktuellen Fortschritt per Mail zu informieren oder bei Problemen zu kontaktieren.

Vorstellbar ist auch eine kontinuierliche Bewegung des Positioniermotors anhand der Geschwindigkeit des Wickelrades, sodass die Faser bei einem Schritt nicht kurzfristig stark mechanisch belastet wird, sondern dauerhaft nur eine geringe mechanische Belastung erfährt.

Die Steuerung könnte zudem komfortabler gestaltet werden, indem die manuelle Steuerung, die aktuell über die Kommandozeile benutzt wird, eine grafische Oberfläche erhält und somit intuitiver benutzt werden kann.

Es ist zu sehen ist, dass sich durch die neue Steuerung eine Vielzahl von Möglichkeiten ergeben, die in Zukunft implementiert werden können und somit die Fasermattenproduktion am Prototypen erheblich vereinfachen und flexibler gestalten können.

4 Zusammenfassung

Die Veränderungen im Aufbau wurden nach Schema 3.2 umgesetzt, womit die gewünschte Zentralisierung des Aufbaus erreicht ist. Wo vor dem Umbau die Motoren jeweils nur mit ihrer Peripherie kommunizierten, läuft jetzt der Informationsfluss über das Raspberry Pi, das die entsprechenden Befehle an die Motoren sendet und sie kontrolliert.

Alle Funktionen, die vor dem Umbau vorhanden waren, sind auch nach dem Umbau verfügbar. Zudem konnten die gewünschten Sonderfunktionen aus Abschnitt 3.3.2, realisiert werden.

Beim Auslesen des Spiegels wird die aktuelle Geschwindigkeit des Wickelrads abgefragt und entsprechend bewegt sich der Positioniermotor einen Schritt weiter (positive Geschwindigkeit) oder zurück (negative Geschwindigkeit). Auch dem Wunsch nach einer automatisierten Wickelung von Matten bestimmter Breite wurde Rechnung getragen¹.

Die neue Steuerung, dessen Erweiterungsmöglichkeiten in Abschnitt 3.5 diskutiert wurden, erlaubt die einfache Umsetzung neuer Funktionen. Das Hauptziel die Steuerung so zu vereinfachen, dass zukünftig der Prototyp flexibel auf neue Anforderungen reagieren kann ist erreicht und erste neue Funktionen sind implementiert.

¹Die Funktion konnte aufgrund einer Fehlfunktion am Rad bisher nicht überprüft werden. Dies sollte aufgrund der neuen Steuerung schnell realisiert sein, sobald das Wickelrad wieder einsatzfähig ist.

Literaturverzeichnis

- [1] "Offizielle Homepage CERN," June, 2014. <http://home.web.cern.ch>.
- [2] "FACTBOX-Five facts about CERN's Large Hadron Collider," June, 2014. <http://uk.reuters.com/article/2008/09/09/uk-science-cern-lhc-idUKL810212720080909>.
- [3] "Homepage LHCb-Experiment," June, 2014. <http://lhcb-public.web.cern.ch/lhcb-public/>.
- [4] LHCb Collaboration, "LHCb Tracker Upgrade Technical Design Report," Tech. Rep. CERN-LHCC-2014-001. LHCb-TDR-015, CERN, Geneva, Feb, 2014.
- [5] S. Y. Lee, *Accelerator Physics*. World Scientific Pub Co Inc, 2 ed., 12, 2004. <http://amazon.com/o/ASIN/981256182X/>.
- [6] "International CERN Masterclasses - Hands on Particle Physics," June, 2014. <https://kjende.web.cern.ch/kjende/en/atlas.htm>.
- [7] "Homepage CMS-Experiment," June, 2014. <http://cms.web.cern.ch>.
- [8] "Homepage ALICE-Experiment," June, 2014. <http://aliceinfo.cern.ch/>.
- [9] "LHCb Masterclass Dokument," June, 2014. <http://lhcb-public.web.cern.ch/lhcb-public/en/LHCb-outreach/masterclasses/DEinstructions.pdf>.
- [10] Particle Data Group J. Beringer et al., "The review of particle physics," *Phys. Rev.* **D86** (2012) 010001. <http://pdg.lbl.gov>.
- [11] Fred Blanc, "Status of the Scintillating Fibre Tracker." Vortrag im Rahmen der LHCb week, Dec, 2013.
- [12] Lehrstuhl E5 TU Dortmund, June, 2014. Diese bisher unveröffentlichten Dokumente sind während der Forschungsarbeit von Robert Ekelhof und Janine Müller entstanden und wurden freundlicherweise für die Bachelorarbeit zur Verfügung gestellt.
- [13] "Herstellerhomepage Kuraray," June, 2014. <http://www.kuraray.eu/>.
- [14] Beta LaserMike, *AccuScan 5010/5025/5040/5080 Instruction Handbook*, Mar, 2009.

- [15] "Homepage Trinamic," June, 2014.
<http://www.trinamic.com/software-tools/tmcl-ide>.
- [16] "Homepage Raspberry Pi," June, 2014. <http://www.raspberrypi.org>.
- [17] "Mikroprozessor des Raspberry Pi," June, 2014. <http://www.raspberrypi.org/wp-content/uploads/2012/02/BCM2835-ARM-Peripherals.pdf>.
- [18] drogon, "Wiring Pi - GPIO Interface library for the Raspberry Pi," June, 2014. <http://wiringpi.com/>.
- [19] "Homepage Eagle," June, 2014. <http://www.cadsoft.de/>.
- [20] "Board Revisions 1 und 2 im Überblick," June, 2014.
http://erik-bartmann.de/download/PiMeUp_ModelB_Rev2.pdf.
- [21] "Herstellerseite MCP3008," June, 2014. <http://www.microchip.com/wwwproducts/Devices.aspx?product=MCP30081>.
- [22] "Erklärung SPI auf mikrocontroller.net," June, 2014. http://www.mikrocontroller.net/articles/Serial_Peripheral_Interface.
- [23] "Anleitung zur Implementierung des ADC," June, 2014.
http://erik-bartmann.de/download/PiMeUp_MCP3008.pdf.
- [24] "Raspberry Pi Specification," June, 2014.
<http://de.scribd.com/doc/101830961/GPIO-Pads-Control2>.
- [25] "Homepage Gnuplot," June, 2014. <http://www.gnuplot.info/>.
- [26] "Testcode von Trinamic," June, 2014.
www.trinamic.com/_scripts/download.php?file=_articles%2Fsoftware-tools%2Ftmcl-ide%2F_appnotes%2FTMCLTestVC%2B%2B.zip.
- [27] "C++ Reference - Thread," June, 2014.
<http://www.cplusplus.com/reference/thread/thread/>.
- [28] A. Metzlar, *BIOS - das Praxisbuch*. Franzis, München, 2004.