

Requirements and Design Documentation (RDD)

Version 0.3

ESEP - Praktikum - Wintersemester 2016

| | | | |
|------------|-----------|---------|------------------------------------|
| Lüdemann | Mona | 2212744 | mona.luedemann1@haw-hamburg.de |
| Butkerei | Marvin | 2247550 | marvin.butkerei@haw-hamburg.de |
| Schumacher | Wilhelm | 2245216 | wilhelm.schumacher@haw-hamburg.de |
| Melkonyan | Anushavan | 2243668 | anushavan.melkonyan@haw-hamburg.de |
| Colbow | Marco | 2177095 | marco.colbow@haw-hamburg.de |
| Cakir | Mehmet | 2195657 | mehmet.cakir@haw-hamburg.de |

16. November 2016

Änderungshistorie:

| Version | Author | Datum | Anmerkungen/Änderungen |
|---------|--------------|------------|---|
| 0.1 | Mehmet Cakir | 2016-10-18 | Kapitel 1-4 und Testkonzept |
| 0.2 | Mehmet Cakir | 2016-10-26 | Korrekturen an Formulierung, Visualisierungen noch nicht festgelegt |
| 0.3 | Mehmet Cakir | 2016-11-03 | Testtabellen umformatiert. Tests zu Grundfunktionen, HAL-UML, Systemgrenzen, Systemarchitektur und Visualisierungsentscheidung sowie entsprechend kurzen Text hinzugefügt. |
| 0.4 | Mehmet Cakir | 2016-11-16 | Neugliederung der Kapitel 4 und 7, Systemkontexte zusammengeführt, verwendete Werkzeuge ergänzt, Zeitmessung und FSM/HSM eingepflegt, Abbildung 6 zur Zeiterfassung aktualisiert, diverse Umformulierungen. |

Inhaltsverzeichnis

| | |
|---|-----------|
| Inhaltsverzeichnis | 2 |
| 1 Teamorganisation | 3 |
| 1.1 Verantwortlichkeiten | 3 |
| 1.2 Absprachen | 3 |
| 1.3 Repository-Konzept | 4 |
| 2 Projektmanagement | 4 |
| 2.1 Prozess | 4 |
| 2.2 PSP/Zeitplan/Tracking | 4 |
| 2.3 Qualitätssicherung | 4 |
| 3 Randbedingungen | 5 |
| 3.1 Entwicklungsumgebung | 5 |
| 3.2 Werkzeuge | 5 |
| 3.3 Sprachen | 5 |
| 4 Requirements and Use Cases | 6 |
| 4.1 Stakeholder | 6 |
| 4.2 Anforderungen | 7 |
| 4.3 Systemkontext | 10 |
| 4.3.1 Softwareebene | 10 |
| 4.3.2 Systemebene | 13 |
| 4.4 Use Cases | 14 |
| 5 Design | 15 |
| 5.1 Systemarchitektur | 15 |
| 5.1.1 Förderband intern | 15 |
| 5.1.2 Gesamtsystem | 16 |
| 5.2 Datenmodellierung | 17 |
| 5.2.1 HAL | 17 |
| 5.3 Zeitmessung | 19 |
| 5.3.1 Vorbedingungen | 19 |
| 5.3.2 Ausführung | 19 |
| 5.4 Verhaltensmodellierung | 21 |
| 6 Implementierung | 25 |
| 7 Testen | 25 |
| 7.1 HAL der Aktorik | 25 |
| 7.2 HAL der Sensorik | 25 |
| 7.3 Serielle Schnittstelle | 25 |
| 7.4 Anlagensteuerung pro Band | 25 |
| 7.5 Ablaufsteuerung über alle drei Bänder | 26 |
| 7.6 Testkonzept für die Abnahme | 26 |
| 8 Lessons Learned | 27 |
| 9 Anhang | 27 |
| 9.1 Glossar | 27 |
| 9.2 Abkürzungen | 27 |

1 Teamorganisation

Grundsätzlich kann jedes Teammitglied eine Aufgabe seiner Wahl übernehmen. Bei jedem Meeting werden die Aufgaben verteilt, worüber im folgenden Meeting über den Fortschritt diskutiert wird. Falls ein Mitglied seine Aufgabe fertiggestellt hat, übernimmt er eine Neue. Bei Nichteinhaltung des Zeitplans werden entsprechend der Zeitpuffer andere Aufgaben zurückgestellt. Die Aufgaben richten sich nach den zu bewältigenden Milestones(siehe [?]) zum jeweiligen Praktikumstermin. Für die Projektleitung und die Pflege des RDD-Dokuments wurde jeweils eine Person bestimmt, welche im Unterkapitel 1.1 eingesehen werden können.

1.1 Verantwortlichkeiten

| Aufgabe | Zuständige/r | Bemerkung |
|------------------|---------------------|--|
| Projektleitung | Mona | Die Projektleitung überwacht den Projektfortschritt und benachrichtigt insbesondere bei Nichteinhalten des Zeitplans alle Teammitglieder. Außerdem hat die Projektleitung bei Unstimmigkeiten immer das letzte Wort. |
| RDD-Pflege | Mehmet | Der Zuständige ist für die Gestaltung und für die Vollständigkeit des RDDs verantwortlich. Er kann andere Gruppenmitglieder dazu auffordern Inhalte für das Dokument zu erarbeiten und ihm bereit zu stellen. |
| Protokollführung | Alle Teammitglieder | Die Protokollführung wird reihum von Gruppenmitgliedern übernommen. Dabei wird folgende Reihenfolge eingehalten: <i>Mona</i> → <i>Marvin</i> → <i>Marco</i> → <i>Wilhelm</i> → <i>Mehmet</i> → <i>Anushavan</i> |

Tabelle 1: Zuteilung von Verantwortlichkeiten

1.2 Absprachen

Zur Kommunikation außerhalb der Praktikumstermine werden Slack und WhatsApp verwendet. Unstimmigkeiten, Fragen und Inkennntnissetzung können somit interaktiv geklärt bzw. mitgeteilt werden. Es wird erwartet, dass jedes Teammitglied in einem Zeitfenster von 24 Stunden darauf reagiert. In folgender Abbildung 1 werden die Termine der Meetings dargestellt:

| Terminplan für Meetings | | | | |
|--|--------------|--------------|--------------|--------------|
| Oktober | Mi, 05.10. | Do, 13.10. | Mi, 19.10. | Mi, 26.10. |
| | ab 16:00 Uhr | ab 12:00 Uhr | ab 16:00 Uhr | ab 16:00 Uhr |
| November | Do, 03.11. | Do, 10.11. | Mi, 16.11. | Mi, 23.11. |
| | ab 12:00 Uhr | ab 12:00 Uhr | ab 16:00 Uhr | ab 16:00 Uhr |
| Dezember | Do, 01.12. | Mi, 07.12. | Mi, 14.12. | Do, 22.12. |
| | ab 12:00 Uhr | ab 16:00 Uhr | ab 16:00 Uhr | ab 12:00 Uhr |
| <i>Weitere Termine können/müssen je nach Bedarf in der Gruppe vereinbart werden.</i> | | | | |

Abbildung 1: Terminplan der Meetings

1.3 Repository-Konzept

Das Projekt wird mit dem Versionskontrollsystem Git verwaltet. Zentral wurde ein Repository auf GitHub angelegt. Erreichbar ist das Repository unter <https://github.com/mbutkerei/conveyor>. Änderungen werden lokal auf einem Branch vorgenommen, jedoch nicht auf dem Master. Sind die Änderungen erfolgreich abgeschlossen, kann der Master mit dem lokalen Branch zusammengeführt werden. Bevor ein push durchgeführt wird, muss gepullt werden. Nachdem ggf. Mergekonflikte gelöst wurden, kann vom Masterbranch aus auf das Repository gepusht werden.

2 Projektmanagement

Für die Gewährleistung eines guten Managements, werden in den folgenden Kapiteln erklärt wie die Teammitglieder mit ihren Aufgaben umgehen bzw. wann eine gegenseitige Benachrichtigung über ihren Fortschritt spätestens stattfinden sollte.

2.1 Prozess

Das Projekt wird auf Grundlage der geforderten Milestones umgesetzt. Für jede Implementierung ist zuvor ein geeignetes, sowie größtenteils selbsterklärendes bzw. verständliches, aber auch möglichst vollständiges Diagramm anzufertigen. Bestenfalls sollte die Visualisierung vor der Implementierung allen anderen Teammitgliedern vorgestellt werden, um mögliche Verbesserungen einzuholen und ggf. Konflikte früh zu erkennen sowie sie zu lösen. Die nachfolgende Tabelle 2 listet für die jeweiligen Spezifikationen die im Team beschlossene Modellierung.

| Spezifikation | Modellierung |
|---------------------------------|---------------------|
| Codestruktur | UML Diagramm |
| Verhalten bzw. logische Abläufe | Zustandsautomat |
| Systemarchitektur | Komponentendiagramm |

Tabelle 2: Festgelegte Modellierung zur jeweiligen Spezifikation

2.2 PSP/Zeitplan/Tracking

Zu jedem Praktikumstermin wird erwartet, dass die verteilten Aufgaben bzw. Milestones erfüllt werden. Um dies zu gewährleisten, muss jedes Teammitglied bei Schwierigkeiten andere Teammitglieder darüber sofort in Kenntnis setzen, damit frühzeitig ausgeholfen werden kann. Dazu wurden Arbeitspakete definiert und als Milestones in einem Gantt-Diagramm festgehalten.

2.3 Qualitätssicherung

Hinsichtlich der Qualitätssicherung, werden die vier Punkte Team, Modellierung, Code und Förderband herangezogen.

1. **Team:** Jedes Teammitglied sollte über seine eigenen Fähigkeiten im Klaren sein und möglichst nur Aufgaben übernehmen, wofür es sich am besten geeignet fühlt. Darüber hinaus muss jedes Teammitglied bei Möglichkeit stets seine Unterstützung anbieten. Bei Problemen oder Überforderung müssen alle anderen Teammitglieder darüber unterrichtet und Aufgaben ggf. neu verteilt werden.
2. **Modellierung:** Vor der Implementierung muss eine geeignete Visualisierung erstellt, anderen Teammitgliedern vorgestellt und diskutiert werden.

3. **Code:** Der Code wird nach beschlossenen Konventionen gefertigt. Dabei werden bekannte Pattern eingesetzt und verständliche sowie übersichtliche Realisierungen angestrebt. Den Maßstab hierfür setzen die Teammitglieder. Treten beim Code Review keine schwerwiegenden Anmerkungen bzw. Verständnisprobleme auf, gilt der Code als verständlich und übersichtlich.
4. **Förderband:** Um hohen Durchsatz sowie Effizienz bei der Aussortierung zu erzielen, werden die Komponenten mit der höchstmöglichen Leistung für die jeweilige Situation angetrieben, während die Sicherheit des Bedieners im Vordergrund steht. Dabei werden Fehler- bzw. Ausnahmezustände ggf. durch einfache Signalcodes mithilfe der Ampel dem Bediener mitgeteilt.

3 Randbedingungen

In diesem Kapitel werden die Bedingungen genannt unter denen das Projekt umgesetzt wird und die Mittel, die für die Umsetzung herangezogen werden.

3.1 Entwicklungsumgebung

Die drei Förderbänder werden über drei QNX Systeme gesteuert, die über eine serielle Schnittstelle verbunden sind. Als IDE wird QNX Momentics auf Windows 7 verwendet.

3.2 Werkzeuge

- QNX Momentics IDE 5.0
- Latex(MiKTeX 2.9, Texmaker 4.5)
- Git 2.8.1
- Visual Paradigm 13.2
- Gantt Project 2.8.1
- Microsoft Visio 2016

3.3 Sprachen

Das System wird in C++ 03 programmiert. Dabei werden vorgegebene Bibliotheken verwendet, welche in folgender Tabelle 3 aufgelistet sind:

| Name | Version | Autor |
|-------------|---------|----------------------------|
| HWaccess.h | Unknown | Prof. Dr. Stephan Pareigis |
| HAWThread.h | Unknown | Prof. Dr. Stephan Pareigis |
| Lock.h | 0.1 | Simon Brummer |

Tabelle 3: Verwendete Programmierbibliotheken

4 Requirements and Use Cases

Mithilfe der Requirements werden die Anforderungen an die einzelnen Komponenten des Förderbandes ermittelt. Dabei werden die Interessen der Stakeholder berücksichtigt.

4.1 Stakeholder

| Stakeholder | Interessen |
|---|---|
| Kunde | <ul style="list-style-type: none">- fehlerfreie Umsetzung der Anforderungen- erfolgreiche Beendigung des Projektes |
| Designer | <ul style="list-style-type: none">- übersichtliches, leicht erweiterbares Design- sorgfältige Dokumentation |
| Entwickler | <ul style="list-style-type: none">- präzises Design- sinnvolle Kommentare- lesbarer Code |
| Tester | <ul style="list-style-type: none">- übersichtliches, vollständiges Testkonzept |
| Bediener (Mitarbeiter, die das Laufband später bedienen sollen) | <ul style="list-style-type: none">- einfache und intuitive Bedienung |
| Instandhalter | <ul style="list-style-type: none">- robustes System |
| Andere Mitarbeiter | <ul style="list-style-type: none">- Kenntnis über System und Funktionsweise |

Tabelle 4: Stakeholder und ihre Interessen

4.2 Anforderungen

| Titel | Beschreibung |
|--|--|
| Ansteuerung der Ampeln | Die Software soll die Ampel für folgende Fälle entsprechend ansteuern können: - grünes Licht bei Normalbetrieb, fehlerfrei - gelbes Licht bei Warnungen - rotes Licht bei Fehler |
| Ansteuerung der Motoren der Förderbänder | Die Motoren der Förderbänder sollen in folgenden Varianten ansteuerbar sein: - Rechtslauf langsam/schnell - Linkslauf langsam/schnell - Stopp |
| Ansteuerung der Weichen | Die Stellungen „offen“ und „geschlossen“ der Weichen müssen angesteuert werden. Außerdem soll beachtet werden, dass die Weichen nur für kurze Zeit die Stellung „offen“ halten, um eine Beschädigung der Weichen zu vermeiden. |
| Erkennung von Werkstücken | Das System muss drei Arten von Werkstücke zuordnen können: - Flache Werkstücke - Werkstücke mit Metalleinsatz (Bohrung liegt nach oben oder unten) - Werkstücke ohne Metalleinsatz (Bohrung liegt nach oben oder unten) |
| Aussortierung von Werkstücken | Flache Werkstücke und Werkstücke, bei der die Bohrung nach unten liegt, sollen aussortiert werden. |
| Reihenfolge der Werkstücke | Am Ende von Band 2 sollen die Werkstücke vereinzelt ankommen in der Reihenfolge: Bohrung oben ohne Metall → Bohrung oben ohne Metall → Bohrung oben mit Metall |

Tabelle 5: Anforderungen(Teil 1)

| Titel | Beschreibung |
|---|---|
| Erkennung von Überschlagen der Werkstücke + Aussortierung des betreffenden Werkstücks | Das System soll erkennen, wenn sich Werkstücke bei der Übergabe von Band 1 zu Band 2 überschlagen und das betreffende Werkstück soll anschließend auf Band 2 aussortiert werden. |
| Langsamer Transport bei Höhenmessung | Wenn ein Werkstück durch die Höhenmessung transportiert wird, soll das Förderband langsam laufen. |
| Konsolenausgabe am Ende von Band 2 | Wenn ein Werkstück das Ende von Band 2 erreicht, sollen auf der Konsole folgende Werkstückdaten ausgegeben werden: - ID - Typ - Höhen-Messwert von Band 1 - Höhen-Messwert von Band 2 |
| Konsolenausgabe am Ende von Band 3 | Am Ende des dritten Bandes sollen die Werkstückdaten ankommender Werkstücke ausgegeben werden. |
| Stopp der Bänder bei keinen Werkstücken | Alle drei Bänder sollen jeweils stoppen, wenn sich kein Werkstück auf ihnen befindet. |
| Erkennung voller Rutschen | Volle Rutschen müssen mithilfe des Sensors am Rutscheneingang erkannt werden. |
| Rutschen koordinieren | Ist die Rutsche auf Band 1 voll, so soll die Aussortierung über Band 2 erfolgen. Umgekehrt, ist die Rutsche auf Band 2 voll, so soll die Aussortierung bereits auf Band 1 erfolgen. |
| Gebündelter Transport von Werkstückgruppen auf Band 3 | Die drei sortierten Werkstücke sollen gebündelt (im Abstand von 1,5cm) an das Ende des dritten Bandes transportiert werden. |
| Fehlererfassung: Verschwinden von Werkstücken + Reaktion | Mittels Zeitmessung soll das Verschwinden von Werkstücken erfasst werden. Wenn zwischen zwei benachbarten Lichtschranken zuviel Zeit vergeht, in der kein Werkstück erfasst wurde, tritt folgende Reaktion auf: Bandstopp, Fehlermeldung. |

Tabelle 6: Anforderungen(Teil 2)

| Titel | Beschreibung |
|--|--|
| Fehlererfassung: Hinzufügen von Werkstücken + Reaktion | Mittels Zeitmessung soll das zu schnelle oder fehlerhafte Hinzufügen von Werkstücken erfasst werden. Wenn zwischen zwei benachbarten Lichtschranken die erwartete Zeit unterschritten wird, in der ein Werkstück erfasst werden müsste, dann tritt folgende Reaktion auf: Bandstopp, Fehlermeldung |
| Fehlererfassung: Beide Rutschen voll + Reaktion | Es soll erkannt werden, wenn beide Rutschen voll sind. Reaktion: Bandstopp, Fehlermeldung |

Tabelle 7: Anforderungen(Teil 3)

4.3 Systemkontext

Zum Systemkontext fallen Anforderungen aus Sicht der Software und des Systems an. Während bei der Software die Ansteuerung der Komponenten eines Förderbandes und die dazugehörigen Schnittstellen anfallen, ist aus Sicht der Systemebene die Kommunikation der Komponenten eines Förderbandes unter sich aber auch die der drei Förderbänder miteinander nötig.

4.3.1 Softwareebene

Im Folgenden sind die Schnittstellen der Ereignisse aufgelistet, die zur Ansteuerung der einzelnen Komponenten ausgelöst sowie Ereignisse, die mithilfe der Sensoren erfasst werden. Die Methodennamen der erfassbaren Ereignisse beginnen mit „is“. **Port A (Ausgabeport)**

| Ereignis | Methodenname |
|------------------|---|
| Motor Rechtslauf | <code>right()</code> |
| Motor Linkslauf | <code>left()</code> |
| Motor langsam | <code>slow()</code> |
| Motor schnell | <code>fast()</code> |
| Motor Stopp | <code>stop()</code> |
| Weiche auf/zu | <code>switchOpen()</code> <code>switchClosed()</code> |
| Ampel Grün | <code>turnGreenOn()</code> <code>turnGreenOff()</code> |
| Ampel Gelb | <code>turnYellowOn()</code> <code>turnYellowOff()</code> |
| Ampel Rot | <code>turnRedOn()</code> <code>turnRedOff()</code> |

Tabelle 8: API auf Port A(Ausgabeport) - auslösbare Ereignisse

Port B (Eingabeport)

| Ereignis | Methodenname |
|---------------------------|--|
| Einlauf Werkstück | <code>isItemRunningIn()</code> |
| Werkstück in Höhenmessung | <code>isItemAltimetry()</code> |
| Höhenmessung | <code>isItemInAltimetryToleranceRange()</code> |
| Werkstück in Weiche | <code>isItemSwitch()</code> |
| Werkstück Metall | <code>isItemMetal()</code> |
| Weiche offen | <code>isSwitchOpen()</code> |
| Rutsche voll | <code>isSkidFull()</code> |
| Auslauf Werkstück | <code>isItemRunningOut()</code> |

Tabelle 9: API auf Port B (Eingabeport) - erfassbare Ereignisse

Port C (Ein-/Ausgabeport)

| Ereignis | Methodenname |
|----------------|---------------------------------------|
| LED Starttaste | turnLedStartOn() turnLedStartOff() |
| LED Resettaste | turnLedResetOn() turnLedResetOff() |
| LED Q1 | turnLedQ1On() turnLedQ1Off() |
| LED Q2 | turnLedQ2On() turnLedQ2Off() |
| Taste Start | isButtonStartPressed() |
| Taste Stopp | isButtonStopPressed() |
| Taste Reset | isButtonResetPressed() |
| Taste E-Stopp | isButtonEStopPressed() |

Tabelle 10: API auf Port C (Ein-/Ausgabeport) - auslösbare/erfassbare Ereignisse

4.3.2 Systemebene

Die nachfolgende Abbildung 2 visualisiert die Systemgrenzen der Förderbandanlage. Dabei sind die dazugehörigen Sensoren und Aktoren abgebildet durch welche eine Förderbandanlage mit der Umwelt und seiner Nachbarsysteme kommuniziert. Die Tabellen 11 und 12 listen die Aufgaben der Sensoren und Aktoren auf.

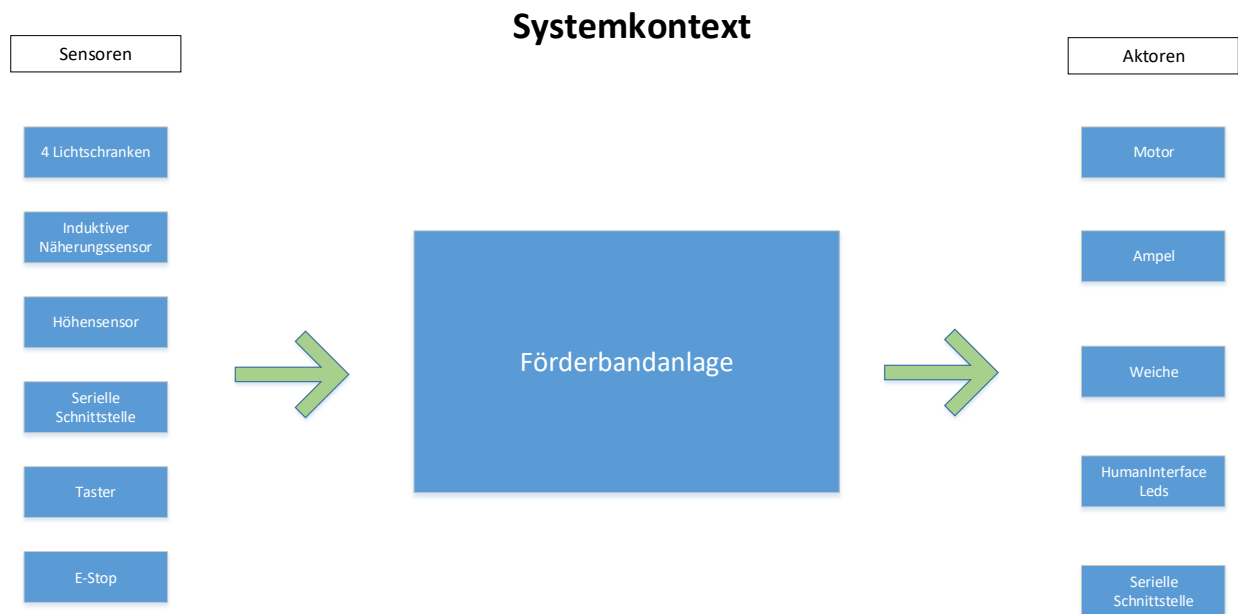


Abbildung 2: Systemgrenzen der Förderbandanlage mit Sensoren und Aktoren

| Sensor | Aufgabe |
|----------------------------|---|
| 4 Lichtschranken | Erfassen, ob sich gerade ein Werkstück sich auf der Höhe der jeweiligen Lichtschranke befindet. |
| Induktiver Näherungssensor | Stellt fest, ob es sich um ein metallisches Werkstück handelt. |
| Höhensensor | Misst die Höhe des Werkstücks. |
| Serielle Schnittstelle | Ermöglicht den Empfang von Datenpaketen anderer Nachbarsysteme. |
| Taster | Löst je nach Programmierung entsprechende Aktion aus. |
| E-Stop | Schaltet alle Förderbänderbänder bedingungslos aus. |

Tabelle 11: Sensoren und deren Aufgaben

| Aktor | Aufgabe |
|------------------------|--|
| Motor | Treibt das band der Förderbandanlagen an. |
| Ampel | Signalisiert entsprechend anliegender Ereignisse. |
| Weiche | Sortiert bei falscher Reihung Werkstücke aus. |
| HumanInterface Leds | Signalisieren dem Bediener Sensorereignisse |
| Serielle Schnittstelle | Ermöglicht das Versenden von Datenpaketen an andere Nachbarsysteme |

Tabelle 12: Aktoren und deren Aufgaben

4.4 Use Cases

1. Flache Werkstücke aussortieren

Akteure: Mitarbeiter (legt die Werkstücke auf das Band), Höhenmessung, Weiche

Auslösendes Ereignis: Höhenmessung erkennt das flache Werkstück.

Kurzbeschreibung: Die flachen Werkstücke werden auf Band 1 mit der Höhenmessung erkannt und über die Weiche aussortiert.

2. Werkstückdaten ausgeben

Akteure: Lichtschranke, Display

Auslösendes Ereignis: Die Lichtschranke auf Band 2 wird durchquert.

Kurzbeschreibung: Wenn ein Werkstück das Ende von Band 2 erreicht, werden die Werkstückdaten auf dem Display ausgegeben.

3. Ausgabe der Werkstücke auf Band 3 in der richtigen Reihenfolge

Akteure: Lichtschranke, Mitarbeiter (nimmt die Werkstücke in Empfang), Weiche

Auslösendes Ereignis: Es sind die drei richtigen Werkstücke auf Band 3 vorhanden.

Kurzbeschreibung: Auf Band 3 werden jeweils 3 Werkstücke gebündelt in der richtigen Reihenfolge (Bohrung oben ohne Metall → Bohrung oben ohne Metall → Bohrung oben mit Metall) ausgegeben.

5 Design

Im Designentwurf sind Systemarchitektur, Datenmodellierung und Verhaltensmodellierung der Förderbänder enthalten.

5.1 Systemarchitektur

Die Systemarchitektur setzt sich aus den internen Architekturen der drei Förderbänder und der Architektur des Gesamtsystems, welche die Schnittstellen der drei Förderbänder zueinander darstellt, zusammen.

5.1.1 Förderband intern

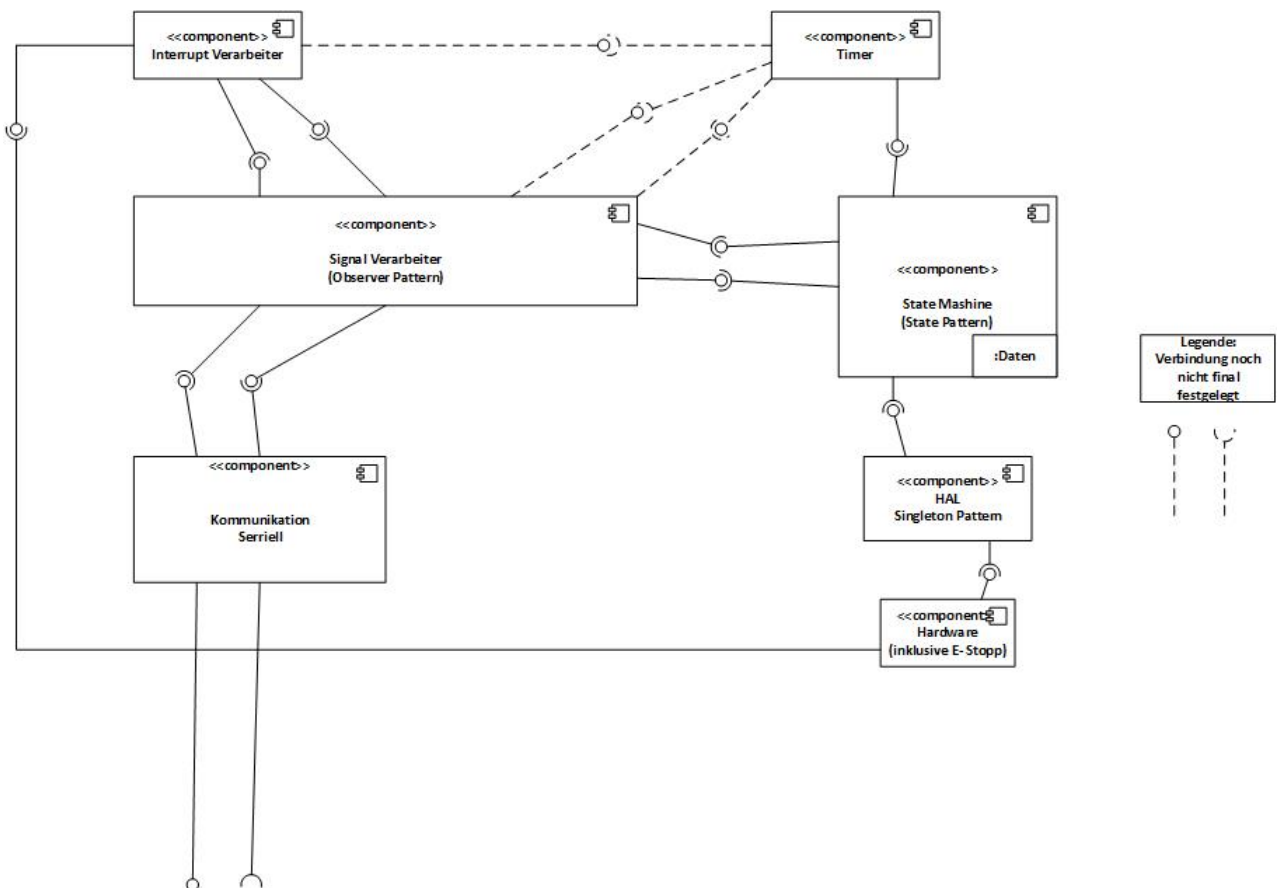


Abbildung 3: Interne Systemarchitektur eines Förderbandes

| Komponente | Aufgabe |
|-----------------------|--|
| Interrupt Verarbeiter | Verarbeitet Interrupts aus Timer, Signal Verarbeiter und Hardware. |
| Timer | Zeiterfassung für zeitkritische Abläufe. |
| Signal Verarbeiter | Verarbeitet Signale aus Interrupt Verarbeiter, Timer, State-machine und Kommunikation seriell. |
| Statemachine | Steuert den logischen Ablauf. |
| Kommunikation seriell | Bildet die Schnittstelle zwischen Förderband und Gesamtsystem. |
| HAL | Hardwareabstraktionsschicht zur Ansteuerung der Komponenten eines Förderbandes. |
| Hardware | Hardware des Förderbandes |

Tabelle 13: Aufgaben der Komponenten eines Förderbandes

5.1.2 Gesamtsystem

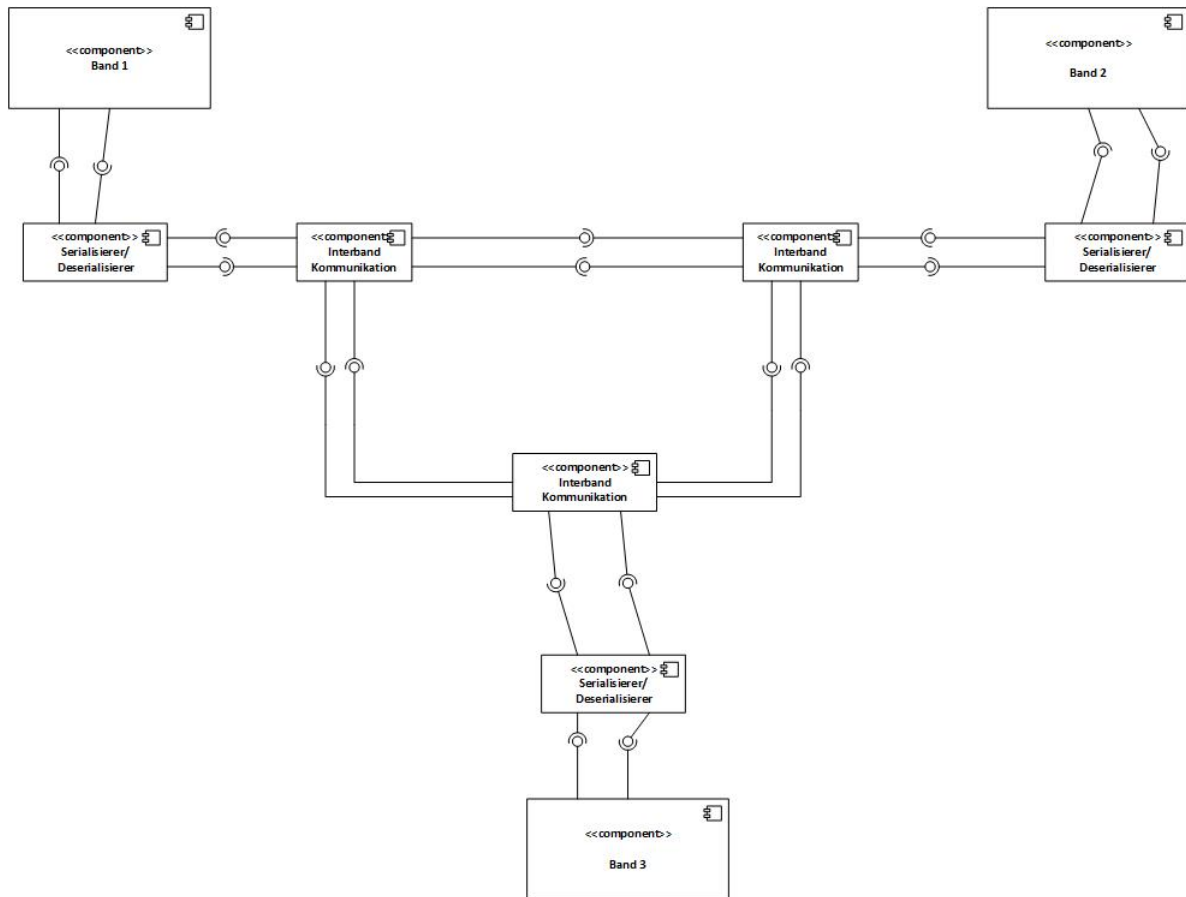


Abbildung 4: Systemarchitektur des Gesamtsystems

| Komponente | Aufgabe |
|-------------------------------|---|
| Band 1 | Erstes Förderband, welches die Sortierung entsprechend der Reihung durchführt. |
| Band 2 | Zweites Förderband, welches die Sortierung entsprechend der Reihung durchführt. |
| Band 3 | Drittes Förderband, welches die Gruppierung der Werkstücke übernimmt und anschließend übergibt. |
| Serialisierer/Deserialisierer | Serialisiert bzw. deserialisiert Datenpakete zur Kommunikation. |
| Interband Kommunikation | Empfängt bzw. versendet serialisierte Datenpakete. |

Tabelle 14: Aufgaben der Komponenten des Gesamtsystems

5.2 Datenmodellierung

Die Modellierung der Klassen und dessen Methoden sind mithilfe von UML-Diagrammen realisiert.

5.2.1 HAL

Mit Der Klasse HAL werden die Hardwarekomponenten eines Förderbandes angesteuert. Dabei wird jede Hardwarekomponente nach dem Singleton-Pattern instanziiert. Die nachfolgende Abbildung 5 stellt alle Klassen zur HAL mit ihren Methoden dar.

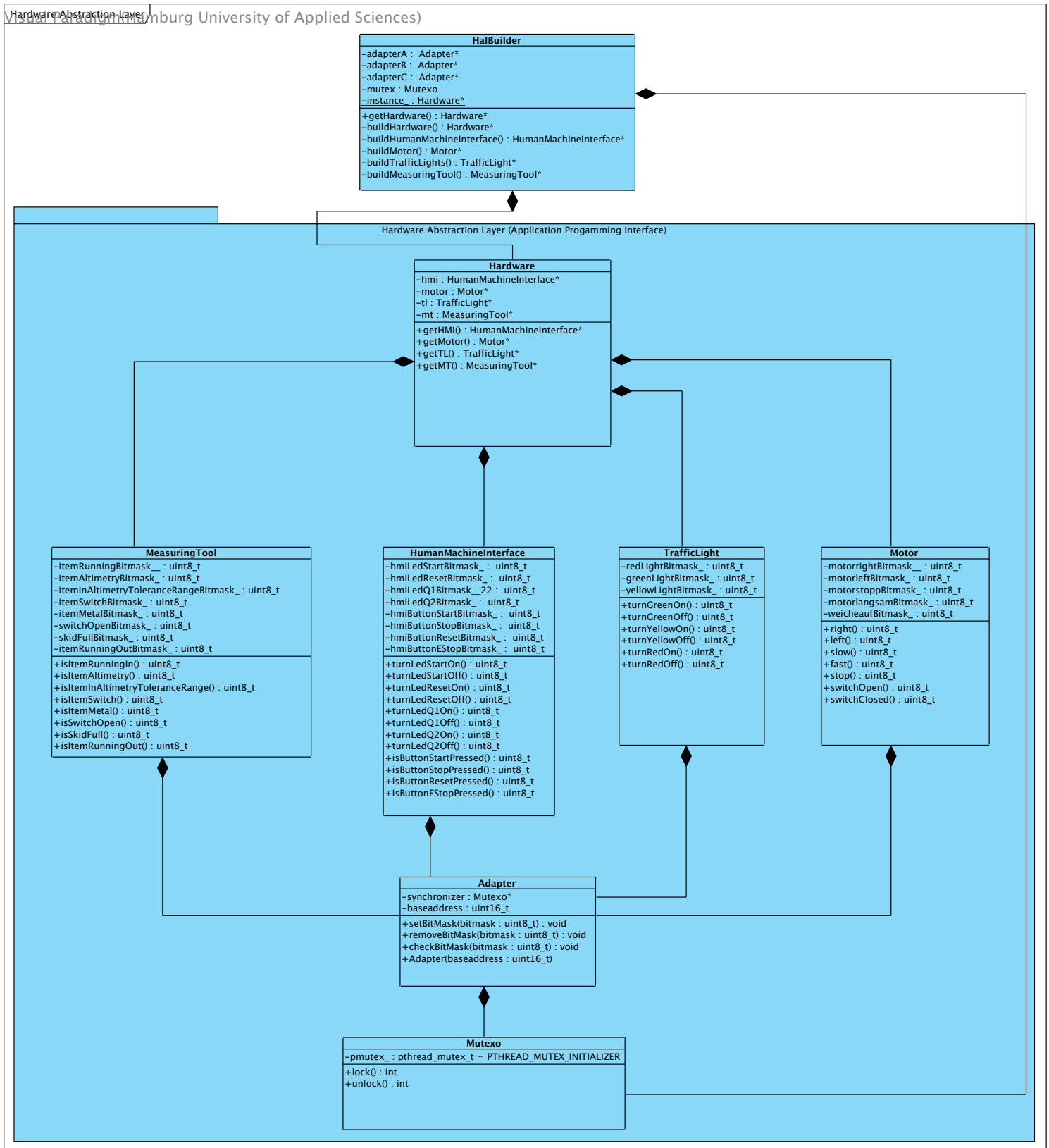


Abbildung 5: UML-Diagramm zur HAL

5.3 Zeitmessung

Auf allen Förderbändern muss jeweils eine Zeitmessung durchgeführt werden, um Zeiten zu erfassen, mit denen im Normalbetrieb unsachgemäß hinzugefügte Werkstücke oder das Verschwinden von Werkstücken erkannt werden kann.

5.3.1 Vorbedingungen

Für die Zeitmessung muss eine ähnliche Routine laufen, die auch im Normalbetrieb für die Förderbänder ablaufen wird, wobei Messvorgänge ausgelassen werden und ggf. das Werkstück bedingungslos durchgelassen wird. Falls in der Routine Geschwindigkeitsänderungen des Bandes vorliegen, müssen diese jedoch miteinbezogen werden, nur die aktive Messung nicht. Da das Band der jeweiligen Förderbänder nicht perfekt geführt wird und der Timer eine relativ hohe Zeitauflösung hat, müssen **best** und **worst case** Zeiten ermittelt werden.

5.3.2 Ausführung

Für den **best case** wird das Werkstück beim Hinzufügen von der ersten Lichtschranke aus gesehen am **linken Rand** der Führung angelegt, während beim **worst case** am **rechten Rand** angelegt werden muss. Dabei muss das Werkstück die erste Lichtschranke auch unterbrechen, was trivialerweise auch im Normalbetrieb beim Hinzufügen weiterer Werkstücke beachtet werden muss! Die Zeitmessung beginnt, nachdem das Werkstück die Lichtschranke am Anfang nicht mehr unterbricht. Unterbricht das Werkstück die nachfolgenden Lichtschranken, wird jeweils die Zeit gestoppt. Die nachfolgende Abbildung 6 zeigt wann die Zeiten t_0 , t_H , t_W und t_E erfasst werden.

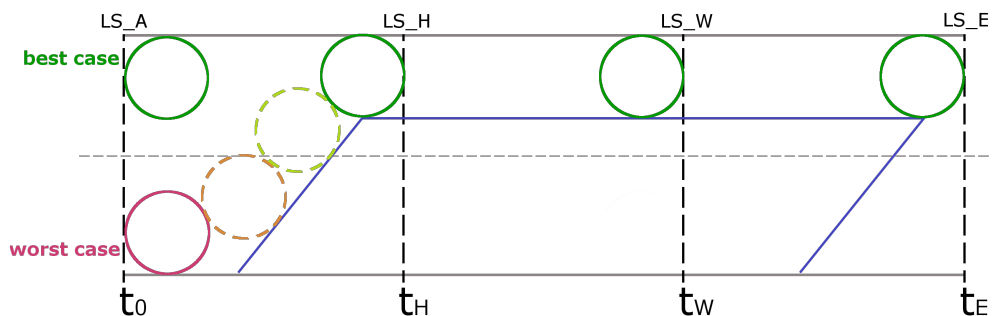


Abbildung 6: Stoppen der Zeit an den Lichtschranken: LS_A , LS_H , LS_W , LS_E

LS_A : Lichtschranke-Anfang, LS_H : Lichtschranke-Höhenmessung, LS_W : Lichtschranke-Weiche, LS_E : Lichtschranke-Ende.

Auf der nachfolgenden Seite ist der endliche Automat in Abbildung 7 zur Zeitmessung abgebildet.

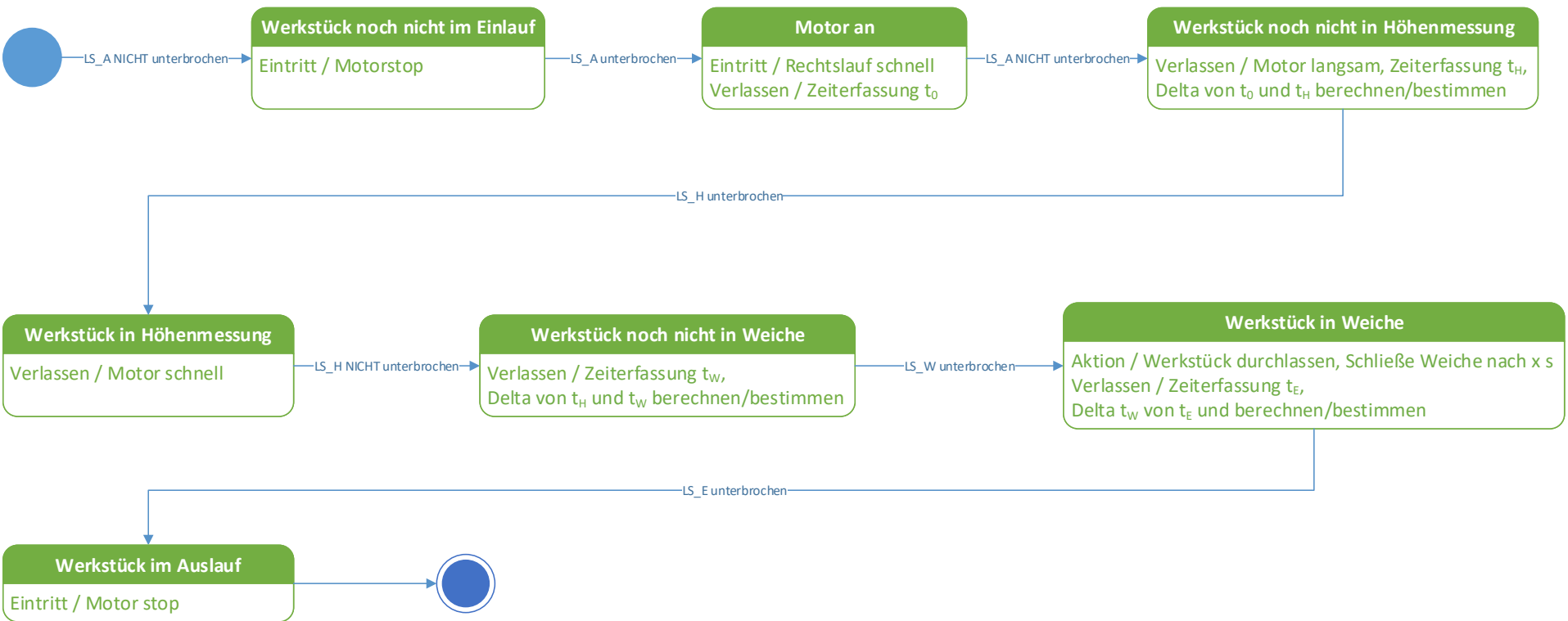


Abbildung 7: FSM zur Zeitmessung

5.4 Verhaltensmodellierung

Das Verhalten der drei Förderbänder ist mithilfe von hierarchischen Zustandsautomaten realisiert. Nach der Durchführung einer Zeitmessung, sind alle drei Förderbänder bereit für den Normalbetrieb. Die Abbildungen 8, 9 und 10 auf den folgenden Seiten bilde die Förderbänder 1, 2 und 3 ab.



22



Abbildung 9: HSM zu Förderband 2

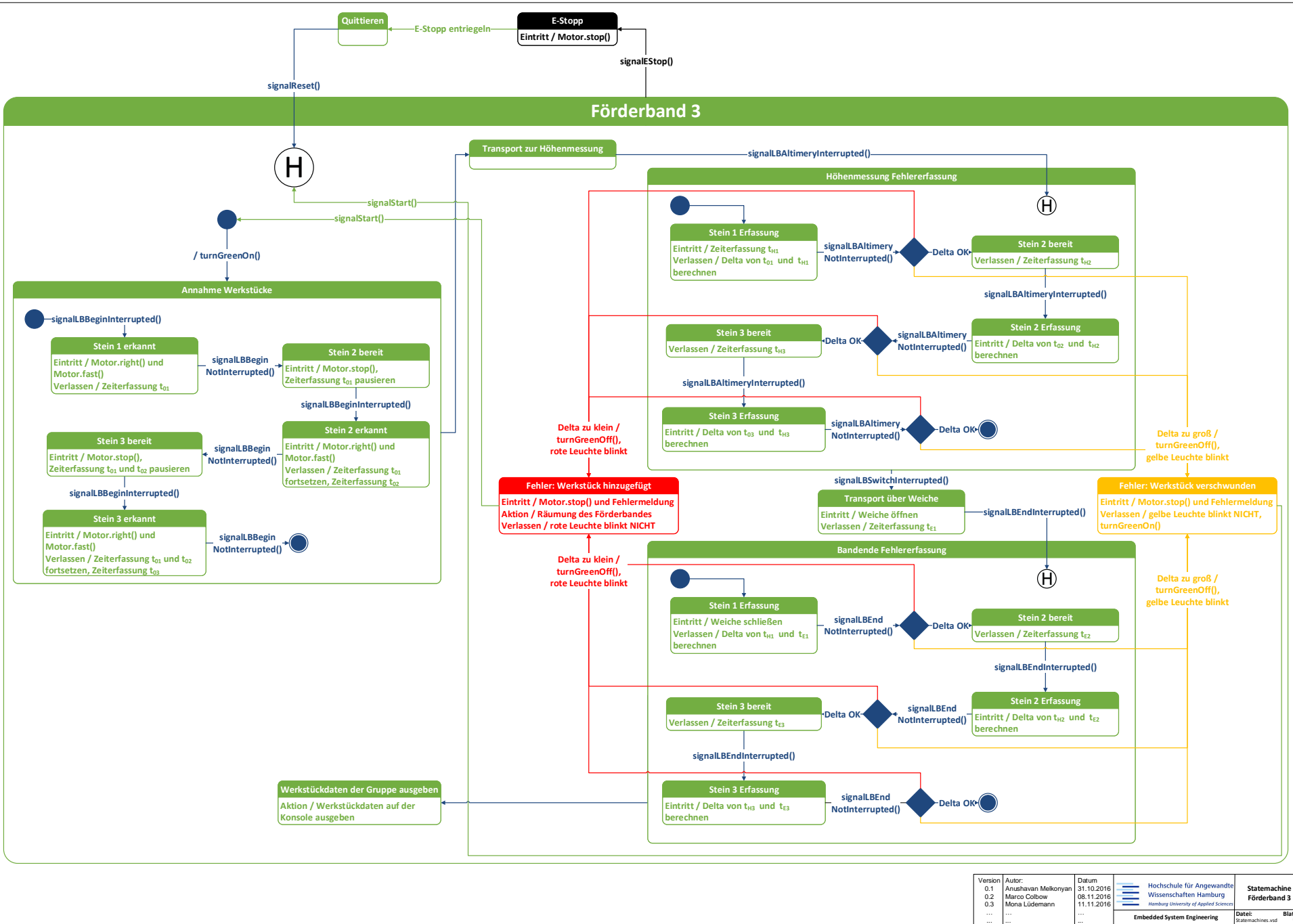


Abbildung 10: HSM zu Förderband 3

6 Implementierung

Anmerkung: Nur wichtige Implementierungsdetails sollen hier erklärt werden. Code-Beispiele (snippets) können hier aufgelistet werden, um der Erklärung zu dienen. Anmerkung: Bitte KEINE ganze Programme hierhin kopieren!

7 Testen

Das Testen der Förderbänder findet nach Fertigstellung von Milestones statt, welche die Ansteuerung der Förderbänder sowie deren Kommunikation untereinander enthalten. Die Tests reichen von essentiellen Ansteuerungstests einzelner Komponenten der Förderbänder bis hin zur Ablaufsteuerung über alle drei Bänder. Für die Abnahme werden speziell komplexere Tests durchgeführt.

7.1 HAL der Aktorik

| ID | Funktion | Test erfolgreich | Anmerkung |
|----|-----------------|------------------|-----------|
| 1 | Rote Lampe an | Ja | |
| 2 | Rote Lampe aus | Ja | |
| 3 | Gelbe Lampe an | Ja | |
| 4 | Gelbe Lampe aus | Ja | |
| 5 | Grüne Lampe an | Ja | |
| 6 | Grüne Lampe aus | Ja | |
| 7 | Motor langsam | Ja | |
| 8 | Motor schnell | Ja | |
| 9 | Motor links | Ja | |
| 10 | Motor rechts | Ja | |
| 11 | Motor stoppen | Ja | |
| 12 | Weiche auf | Ja | |
| 13 | Weiche zu | Ja | |
| 14 | Led Start an | Ja | |
| 15 | Led Start aus | Ja | |
| 16 | Led Reset an | Ja | |
| 17 | Led Reset aus | Ja | |
| 18 | Led Q1 an | Ja | |
| 19 | Led Q1 aus | Ja | |
| 20 | Led Q2 an | Ja | |
| 21 | Led Q2 aus | Ja | |

Tabelle 15: Testauswertung zur HAL der Aktorik

7.2 HAL der Sensorik

Hier erscheinen bald die Tests zu der HAL der Sensorik sowie deren Auswertung.

7.3 Serielle Schnittstelle

Hier erscheinen bald die Tests zu der seriellen Schnittstelle sowie deren Auswertung.

7.4 Anlagensteuerung pro Band

Hier erscheinen bald die Tests zur Anlagensteuerung pro Band sowie deren Auswertung.

7.5 Ablaufsteuerung über alle drei Bänder

Hier erscheinen bald die Tests zur Ablaufsteuerung über alle drei Bänder sowie deren Auswertung.

7.6 Testkonzept für die Abnahme

| ID | Funktion | Test erfolgreich | Anmerkung |
|----|--|------------------|-----------|
| 1 | Erkennung der Werkstücke am Anfang des Förderbandes | | |
| 2 | Flache Werkstücke werden aussortiert | | |
| 3 | Bei der Aussortierung der flachen Werkstücke blinkt die gelbe Leuchte | | |
| 4 | Werkstücke mit der Bohrung nach unten werden aussortiert | | |
| 5 | Bei Förderband 1, Fehlermeldung bei voller Rutsche | | |
| 6 | Bei Förderband 2, Fehlermeldung und Stopp von Förderband 1 und Förderband 2 bei voller Rutsche | | |
| 7 | Stopp beim leeren Förderband | | |
| 8 | Beim Verschwinden von Werkstücken wird eine Fehlermeldung ausgegeben und das Förderband stoppt | | |
| 9 | Beim Hinzufügen von Werkstücken mitten auf dem Förderband wird eine Fehlermeldung ausgegeben und das Förderband stoppt | | |
| 10 | Am Ende von Band 2 soll die gewünschte Reihenfolge der Werkstücke entstehen | | |

Tabelle 16: Testauswertung der Abnahmetests(Teil 1)

| ID | Funktion | Test erfolgreich | Anmerkung |
|----|--|------------------|-----------|
| 12 | Am Ende vom Band 3 werden die Werkstückdaten als 3er Gruppe ausgegeben auf der Konsole | | |
| 13 | Förderband 3 transportiert die Werkstücke erst dann bis zum Ende des Bandes wenn die 3er Gruppe vollständig ist. | | |

Tabelle 17: Testauswertung der Abnahmetests(Teil 2)

8 Lessons Learned

Führen sie ein Teammeeting durch in dem gesammelt wird, was gut gelaufen war, was schlecht gelaufen war und was man im nächsten Projekt (z.B. im PO) besser machen will. Listen sie für die Aspekte jeweils mindestens drei Punkte auf. Weitere Erfahrungen und Erkenntnisse können hier ebenso kommentiert werden, auch Anregungen für die Weiterentwicklung des Praktikums.

9 Anhang

9.1 Glossar

Eindeutige Begriffserklärungen

9.2 Abkürzungen

Listen sie alle Abkürzungen auf, die sie in diesem Dokument benutzt haben.