

Aufgabe ESEP

Prof. Dr. Zhen Ru Dai
Prof. Dr. Franz Korf
Prof. Dr. Thomas Lehmann
M. Sc. Enrico Christophers

Sommersemester 2020

Achtung: Wegen der derzeitigen Corona-Situation werden die ESEP-relevanten Veranstaltungen dieses Semester teilweise online gehalten. Wir berücksichtigen diese Situation und haben die Aufgaben dementsprechend angepasst. Achten Sie bitte auf weitere Hinweise in diesem Dokument. Das gilt ausdrücklich nur für dieses Semester!

1 Ziel des Praktikums

In diesem Praktikum entwerfen Sie Steuerungs-Software im Team mit den aktuellen Methoden und Techniken der Softwareentwicklung und führen diesen Entwurf in funktionierende Software über. Hierdurch sollen Sie lernen, geeignete Methoden und Techniken der Softwareentwicklung auszuwählen, anzuwenden und ein funktionierendes Software-Produkt zu erstellen. Weiterhin soll die Koordinierung und das Arbeiten im Team geübt werden.

2 Aufgabenstellung

Bauen Sie aus zwei Förderbandmodulen (Festo-Transfersystem) eine **Werkstück-Sortieranlage**. Jedes Förderbandmodul wird durch einen eigenen Steuerungscomputer (Embedded System) gesteuert. Die beiden Computer sind über eine serielle Schnittstelle gekoppelt. Alternativ kann das Qnet Protokoll von QNX auch verwendet werden.

Für die Implementierung der Grundfunktionalität wird ein 4er-Team benötigt. Bei einem größeren Team sind, in Absprache mit der Betreuung, Zusatzfunktionalitäten zu realisieren.

3 Problembeschreibung/Anforderungen

Im Folgenden sind die zu erfüllenden Anforderungen aufgeführt. Zur besseren Referenzierung sind die Aussagen am Ende mit einer fortlaufenden Nummer versehen.

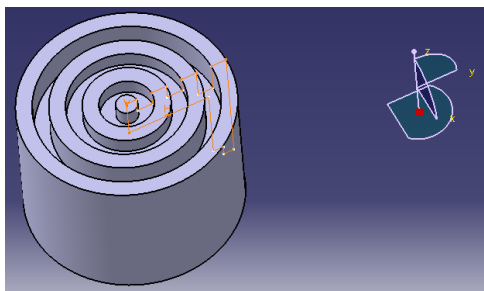
Trotz des Umfangs dieses Dokumentes sind die Anforderungen *unterspezifiziert* oder vielleicht auch *widersprüchlich*. Ein Teil Ihrer Tätigkeit besteht also darin, mit Ihrer Betreuung die Spezifikation zu *präzisieren*.

Alle Absprachen sind schriftlich festzuhalten!

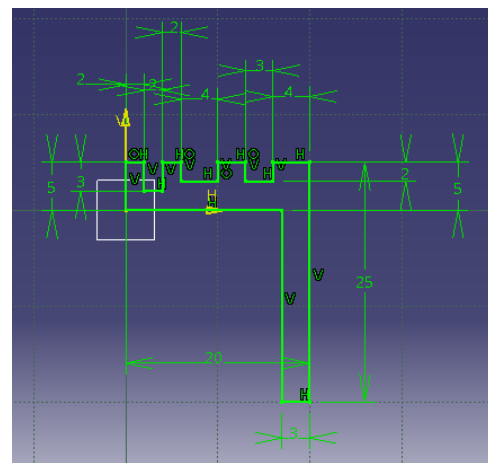
3.1 Grundfunktionalität Sortieren

Die Sortieranlage soll Werkstücke so sortieren, dass Werkstücke mit bestimmten Eigenschaften in die Rutschen aussortiert werden und am Ende von Förderbandmodul 2 Werkstücke in einer vorgegebenen Reihenfolge ankommen⁽¹⁾. Es gibt dabei vier Typen von Werkstücken, die auch unterschiedlich auf das Band des Förderbandmodul 1 gelegt werden können⁽²⁾.

- Flache Werkstücke⁽³⁾
- Werkstücke mit Bohrung und Metalleinsatz⁽⁴⁾.
- Werkstücke mit Bohrung ohne Metalleinsatz⁽⁵⁾.
- Werkstücke mit Ringen auf der Oberseite⁽⁶⁾. Die Ringe stellen eine binär-codierte Typkennung dar (vgl. Abb. 1)⁽⁷⁾. Ein mitteltiefer Ring stellt eine logische 0 dar, ein tiefer Ring eine logische 1⁽⁸⁾. Ringe auf gleicher Höhe wie der Außenring können als Trennzeichen angesehen werden⁽⁹⁾.



(a) 3D-Ansicht



(b) Schnitt durch Werkstück

Abbildung 1: Prinzipieller Aufbau der codierten Werkstücke

Ziel der Sortierung ist es, dass am Ende von Förderbandmodul 2 die uncodierte Werkstücke vereinzelt ankommen⁽¹⁰⁾. Für die Sortierung gibt es zwei Modi⁽¹¹⁾:

- Sortiermode A: Es kommen am Ende von Förderbandmodul 2 Werkstücke in der Reihenfolge Bohrung oben mit Metall → keine Bohrung (nicht flach) an⁽¹²⁾.
- Sortiermode B: Es kommen am Ende von Förderbandmodul 2 Werkstücke in der Reihenfolge Bohrung oben mit Metall → Bohrung oben ohne Metall → keine Bohrung (nicht flach) an⁽¹³⁾.

Die Sortieranlage befindet sich nach dem Start im Sortiermode $A_{(14)}$. Sind beide Bänder leer und es wird ein flaches Werkstück zugeführt, so wird der Sortiermode gewechselt₍₁₅₎. Für die anderen Werkstücke gelten folgende Sortierregeln₍₁₆₎:

1. Flache Werkstücke sollen auf Förderbandmodul 1 erkannt und aussortiert werden₍₁₇₎.
2. Werkstücke, die nicht der gewünschten Reihung entsprechen, sollen auf Förderbandmodul 2 erkannt und aussortiert werden₍₁₈₎.
3. Bei binär-codierten Werkstücken sollen die Werkstücke mit der Typkennung 1 und 4 auf dem Förderbandmodul 1 aussortiert werden₍₁₉₎.
4. Bei binär-codierten Werkstücken sollen die Werkstücke mit der Typkennung 2 und 7 auf dem Förderbandmodul 2 aussortiert werden₍₂₀₎.

Weiterhin ist zu beachten:

- Die Zuführung erfolgt durch Einlegen des Werkstücks am Anfang des Bandes von Förderbandmodul 1₍₂₁₎. Die Lichtschranke wird hierdurch unterbrochen₍₂₂₎.
- Es dürfen stets Werkstücke auf das Band gelegt werden, wenn der Anfang von Förderbandmodul 1 frei ist₍₂₃₎. Auf diesem Förderbandmodul können sich also mehrere Werkstücke befinden₍₂₄₎.
- Die Übergabe an das Förderbandmodul 2 soll *vereinzelte* erfolgen. Es wird nur ein Werkstück auf Förderbandmodul 2 transferiert, wenn ein weiteres Werkstück auf Förderbandmodul 2 mindestens den Abstand von 25 cm hat₍₂₅₎. Es dürfen sich auf Förderbandmodul 2 maximal zwei Werkstücke auf dem Transportband befinden₍₂₆₎.
- Bei der Übergabe des Werkstücks von Förderbandmodul 1 auf Förderbandmodul 2 kann es passieren, dass die Werkstücke sich "überschlagen", d. h. eine oben liegende Bohrung liegt dann auf Förderbandmodul 2 unten₍₂₇₎.
- Auf beiden Bändern sollen die Werkstücke *langsam* durch die Höhenmessung transportiert werden₍₂₈₎. Der gleichmäßig langsame Transport ist gegebenenfalls für die Oberflächenabtastung erforderlich₍₂₉₎.
- Es darf kein Werkstück von den Bändern fallen₍₃₀₎.
- Wenn ein Werkstück das Ende von Förderbandmodul 2 erreicht, sollen auf der Konsole folgende Werkstückdaten ausgegeben werden₍₃₁₎:
 - Werkstück-ID₍₃₂₎
 - Typ₍₃₃₎
 - Höhenmesswert an Förderbandmodul 1₍₃₄₎
 - Höhenmesswert an Förderbandmodul 2₍₃₅₎

Sollte sich das Werkstück "überschlagen" haben, so soll dieses zusätzlich ausgegeben werden₍₃₆₎.

Die Werkstück-ID vergibt Ihr System beim Erkennen des Werkstücks am Anfang von Förderbandmodul 1₍₃₇₎. Im Regelfall (wenn das Werkstück mit der Bohrung

nach oben eingelegt wurde), ist auch der Typ schon nach Durchlauf von Förderbandmodul 1 bekannt₍₃₈₎.

Wird ein Werkstück mit einer binär-codierten Typkennung erkannt, so ist auf der Konsole der jeweiligen Anlage folgende Daten auszugeben₍₃₉₎:

- Zeitstempel der Erkennung₍₄₀₎
 - Werkstück-ID₍₄₁₎
 - Binärcode₍₄₂₎
 - Höhen-Messwert₍₄₃₎
- Beide Bänder sollen jeweils stoppen, wenn sich kein Werkstück auf ihnen befindet₍₄₄₎.
 - Ist die Rutsche auf Förderbandmodul 1 voll, so soll die Aussortierung über Förderbandmodul 2 erfolgen₍₄₅₎. Umgekehrt, ist die Rutsche auf Förderbandmodul 2 voll, so soll die Aussortierung bereits auf Förderbandmodul 1 erfolgen₍₄₆₎. Diese Situation ist dem Bediener zu signalisieren₍₄₇₎. Die Bedingung an die Reihenfolge am Ende von Förderbandmodul 2 muss weiterhin eingehalten werden₍₄₈₎.
 - Insgesamt soll ein möglichst hoher Durchsatz an Werkstücken erreicht werden₍₄₉₎.

SIL Test: Damit Sie Ihre Implementierung jederzeit ohne Hardware testen können, ermöglichen Sie das SIL Testen₍₅₀₎. Achten Sie darauf, dass Sie die Testklassen getrennt von den Implementierungsklassen halten. Definieren Sie sinnvolle Testdaten für Ihre SIL Tests₍₅₁₎. Die Testergebnisse sind zu protokollieren und bei den Abgaben vorzuzeigen.

Ansteuerung der Sortiermechanik: Bezüglich der Umleitung von Werkstücken gibt es zwei Varianten von Förderbandmodulen₍₅₂₎. Die eine Variante verfügt über eine Weiche zum Aussortieren der Werkstücke, die zweite Variante über einen Auswerfer₍₅₃₎.

Die Weichen sind im stromlosen Zustand *geschlossen*₍₅₄₎. Beim Öffnen fließt Strom durch die Magnetspule, die die Weiche betätigt₍₅₅₎. Wenn der Strom zu lange fließt, überhitzt die Spule, und die Weiche wird beschädigt₍₅₆₎. **Daher bitte die Weichen *nicht minutenlang* auf Durchgang stellen**₍₅₇₎!

Der Auswerfer lässt im stromlosen Zustand Werkstücke passieren₍₅₈₎. Fließt Strom durch den Auswerfer, so fährt dieser aus und drückt ein Werkstück aktiv in die Rutsche₍₅₉₎. Im ausgefahrenen Zustand können keine Werkstücke passieren und werden auch nicht in die Rutsche befördert₍₆₀₎.

Ihre Lösung soll mit beiden Varianten, Weiche und Auswerfer, arbeiten können₍₆₁₎. Die Abnahme erfolgt mit einer beliebigen Kombination an Anlagenvarianten₍₆₂₎.

Höhenmessung: Der Laser der Höhenmessung ist gegenüber der Senkrechten in Richtung Bandanfang verkippt₍₆₃₎. Die Verkipfung ist für das Funktionieren des Messverfahrens erforderlich (Blendung)₍₆₄₎.

Überlegen Sie selber ein Vorgehen, wie die Anlage nach Erkennung und Behebung eines Fehlers weitermachen soll₍₆₅₎. Die Bänder sollen im Fehlerfall nach Möglichkeit nicht geräumt werden₍₆₆₎. Falls eine Erweiterung der Fehlererfassung technisch sinnvoll ist oder wesentlich besser in das Lösungskonzept passt, so ist dieses in Absprache möglich₍₆₇₎.

3.2 Bedientaster

Die Bedientaster haben folgende Funktion₍₆₈₎:

- **Start** Anlage wechselt in den Betriebszustand₍₆₉₎. Wird der Taster lange gedrückt, kommt die Anlage in einen Service-Mode und führt beispielsweise Kalibrierungen oder Selbsttests durch₍₇₀₎.
- **Stop** Die Anlage wechselt wieder in den Ruhezustand₍₇₁₎. Der Wechsel ist nur möglich, wenn keine Fehler oder Warnungen vorliegen₍₇₂₎.
- **Reset** Fehlerquittierung (siehe unten)₍₇₃₎.
- **E-Stopp** Schnellabschaltung. Wird der E-Stopp-Schalter gedrückt, steht die *ganze* Anlage still, ähnlich einem Not-Aus₍₇₄₎. Es steht also auch das Förderbandmodul still, an dessen Bedienpanel der E-Stopp-Schalter *nicht* gedrückt wurde₍₇₅₎. Wenn der Schalter anschließend wieder herausgezogen wird, bleibt die Anlage weiterhin so lange stehen, bis der Reset-Taster gedrückt wurde₍₇₆₎.

Geben Sie dem Benutzer Hinweise zur Bedienung der Anlage über die LEDs an den Tastern₍₇₇₎.

3.3 Zustandsanzeigen

Die farbigen Anzeigeleuchten signalisieren folgende Anlagenzustände₍₇₈₎:

Grün Betriebszustand soll die Lampe dauerhaft leuchten₍₇₉₎. Im Service-Mode soll die Lampe blinken₍₈₀₎.

Gelb Blinklicht bei Warnungen₍₈₁₎.

Rot Fehler- und Quittierungszustand₍₈₂₎:

Bei Fehlermeldesystemen müssen unterschiedliche Zustände signalisiert werden₍₈₃₎. Wenn ein Fehler neu aufgetreten ist, hat er den Zustand *anstehend unquittiert*₍₈₄₎. Nun sieht ein Bediener den Fehler (der immer noch ansteht), und teilt dies dem Meldesystem durch Drücken der *Quittierungstaste* mit₍₈₅₎. Der Fehler wechselt in den Zustand *anstehend quittiert*₍₈₆₎. Wenn nun der Fehler behoben wird, der Fehler-Signaleingang (siehe unten) also den Wert *OK* hat, wechselt der Fehler in den Zustand *OK*₍₈₇₎. Dieser Zustand wird nicht gesondert angezeigt₍₈₈₎. Weiterhin ist noch denkbar, dass sich der Fehler von selbst repariert, ohne dass das Bedienpersonal ihn zur Kenntnis genommen hat₍₈₉₎. Dies ist der Zustand *gegangen unquittiert*₍₉₀₎.

In dieser vollen Breite lässt sich die Fehlersignalisierung nur durchführen, wenn es einen allgemeinen *Signaleingang* gibt, der einen Fehler signalisiert₍₉₁₎.

Die Anzeige für aufgetauchte/verschwundene Werkstücke geht nach dem Drücken der Quittungstaste direkt in den Zustand *OK*₍₉₂₎.

Die Fehlerzustände sollen folgendermaßen signalisiert werden₍₉₃₎:

kein Fehler Leuchte aus₍₉₄₎

anstehend unquittiert schnelles Blinken (1 Hz)₍₉₅₎

anstehend quittiert dauerhaft leuchten₍₉₆₎

gegangen unquittiert langsames Blinken (0,5 Hz)₍₉₇₎

4 Erweiterte Anforderungen

Ist ein Team größer als vier Studierende, so wird die zusätzliche Kapazität durch Zusatzfunktionalität kompensiert. Die folgenden Anforderungen stellen mögliche Erweiterungen dar. Die jeweils für das Team relevante Funktionalität ist mit der Betreuung abzusprechen.

Auch hier gilt, alle Absprachen sind *schriftlich* festzuhalten!

4.1 Embedded Recorder/Replay

Alle Eingangssignale in ihre Software sollen durch den Embedded Recorder aufgezeichnet werden₍₉₈₎. Die Abspeicherung soll für Menschen lesbar erfolgen₍₉₉₎. Hierdurch kann im Fehlerfall eine genaue Analyse der Signalreihenfolge durchgeführt werden₍₁₀₀₎.

Ihre Software soll dann durch ein Zusatzelement diese Aufzeichnung einlesen und die Daten in der Reihenfolge, und falls erforderlich Timing, in die Software einspeisen (Replay)₍₁₀₁₎. Hierdurch kann dann mittels Debugger die Software beobachtet und Fehler gefunden werden₍₁₀₂₎.

Die Codierung der aufgezeichneten Sequenzen sollen so aufgebaut werden, dass auch Sequenzen von Hand erstellt werden können₍₁₀₃₎. Die Funktionalität der Software soll dann über den Replay-Mechanismus mit diesen künstlichen Sequenzen getestet werden₍₁₀₄₎.

Hinweis: Setzen Sie den Embedded Recorder für das Testen ein. Daher ist es sinnvoll, ihn frühzeitig im Projekt zu realisieren.

4.2 Remote Control

Die Sortieranlage soll wie auf einem Maschinenleitstand von einem remote PC gesteuert und überwacht werden. Über die Remote Control läßt sich die Anlage ein- und ausschalten₍₁₀₅₎. Auf einer graphischen Oberfläche ist das Button-Panel der Anlage abgebildet und kann diese aus der Entfernung bedienen₍₁₀₆₎. Die graphische Oberfläche zeigt auch den aktuellen Status der Anlage an₍₁₀₇₎.

4.3 Drittes Förderbandmodul über MQTT

Bei dieser Aufgaben werden zwei Teams zusammenarbeiten. Team A soll für Team B ein drittes Förderbandmodul implementieren, das alle Werkstücke, die am Ende des zweiten Förderbandmoduls ankommen, in der schnellen Geschwindigkeit durchtransportiert. Dabei darf immer nur ein Werkstück auf dem dritten Förderbandmodul befinden. Zusätzlich soll währenddessen ein Disco-Licht (d.h. wechselndes Blinken der Anzeigeleuchten) abgespielt werden. Die Kommunikation soll mit dem MQTT Protokoll realisiert werden.

5 Projektablauf

Zu den Praktikumsterminen soll der Projektfortschritt ihrer Betreuung demonstriert werden. Auch wenn Sie vermutlich nach einem agilen Prozess vorgehen, so sind zu bestimmten Terminen Mindestfunktionalitäten **am Beginn des Praktikumstermins** vorzuweisen. Weiterhin sind zu bestimmten Terminen zum Beginn des Praktikumstermins Berichte abzuliefern. Wenn Sie hiervon abweichen wollen, so ist das mit der Betreuung schriftlich zu vereinbaren.

Die Aufwände der Teams sollen erfasst werden. Die Tracking-Daten sollen zu jedem Praktikumstermin aktuell sein.

Achtung: Während der Online Vorlesungszeit haben Sie leider keinen Zugang zur Hardware. Alle hardware-relevanten Aufgaben werden daher bei den Online-Abnahmen von den Betreuern nicht abgefragt. Diese sind jedoch nach der Online-Zeit baldmöglichst abzugeben. Sinnvollerweise sollten Sie auch nicht hardware-relevanten Aufgaben vorziehen, damit Sie später Zeit haben für die Implementierungen an der Hardware.

5.1 Projektabnahme

Folgende Leistungen müssen für den erfolgreichen Abschluss erbracht werden:

- Die Dokumente liegen pünktlich vor und sind abgenommen.
- Projektplan updaten, Anpassung der Schätzungen und der geleisteten Aufwänden.
- Implementierung entspricht dem dokumentierten Design.
- Ihre Implementierung ist ausführlich getestet und protokolliert.
- Alle Funktionalitäten der Anlage sind am Ende des Praktikums erfolgreich umgesetzt.

Das Praktikum kann durch die Betreuung ggf. vorzeitig abgebrochen werden, wenn der Projektfortschritt als nicht ausreichend angesehen wird. Es soll hierdurch verhindert werden, dass weiterhin Zeit investiert wird, die nicht zu einem erfolgreichen Abschluss führen wird.

5.2 Praktikumstermine

Hier finden Sie die Pflichtaufgaben und die empfohlenen Aufgaben zu den jeweiligen Praktikumsterminen. Beachten Sie bitte, dass die Zusatzaufgaben nicht aufgelistet sind. Diese müssen Sie bei der Projektplanung mit berücksichtigen.

Wenn Sie die Reihenfolge der Aufgaben umstellen wollen, so ist dies mit dem Betreuer abzusprechen und schriftlich festzuhalten. Wir empfehlen, eine Abnahmeliste zu den jeweiligen Terminen zu erstellen und diese bei der Abnahme gemeinsam mit dem Betreuer durchzugehen. Die Betreuer behalten sich vor, ihre eigene Abnahmeliste zu erstellen.

1. Praktikumstermin

Pflicht:

- Aus den Workshops zu vervollständigen:
 - Sie haben ein Team gebildet.
 - Jedes Teammitglied hat eine verantwortliche Rolle innerhalb des Projekts bekommen.
 - Sie haben sich intensiv mit der Aufgabenstellung beschäftigt und haben die ersten System- und Anforderungsanalyse gemacht.

- Sie haben mit einem Projektplan angefangen und eine grobe Projektstruktur erstellt.
- Sie haben sich mit eventuellen Zusatzaufgaben beschäftigt.
- Sie haben eine grobe Software Architektur für Ihr Projekt erstellt und haben System-Schnittstellen definiert.
- Sie haben ein Projekt in Momentics aufgesetzt.
- Die Codeverwaltung ist in Gitlab der Informatik aufgesetzt. Ihre Betreuung hat Zugriff auf das Repository. Im Repository ist ein README.md angelegt, in dem die Namen der Teammitglieder mit E-Mail-Adresse eingetragen sind.

Empfehlung:

- Sie können eine Anlage vom Beaglebone Black aus ansprechen (z.B. Ampel ansteuern).

2. Praktikumstermin

Pflicht:

- Die vollständige Anforderungsanalyse liegt als Dokument vor. Eventuelle individuelle Abmachungen mit dem Betreuer sind ausführlich zu dokumentieren.
- Die Aktorik kann über einen Hardware Abstraction Layer (HAL) angesteuert werden.
- Beispiel-Code für die serielle Schnittstelle bzw. Qnet ist lauffähig.

Empfehlung:

- Ein Test prüft die korrekte Datenübertragung über die serielle Schnittstelle bzw. Qnet.
- Abnahmetests anhand der Systemanforderungen definieren.
- Architektur und Design ausarbeiten.

3. Praktikumstermin

Pflicht (Quality Gate):

- Eine Sensorik-HAL basierend auf ISRs und Messages ist realisiert.
- Eine semi-finale Software Architektur mit wohldefinierten Schnittstellen ist ausgearbeitet und dokumentiert. Sie diskutieren diese mit Ihrer Betreuung.

Empfehlung:

- Sensorik-HAL testen
- Anlagensteuerung modellieren.
- Patterns für die Implementierung auswählen.

4. Praktikumstermin

Pflicht:

- Modellierung der Anlagensteuerung ohne Fehlerbehandlung abgeschlossen.
- Kommunikation zwischen zwei Anlagen implementiert und getestet.
- (SIL) Tests mit Abdeckungskriterien (z.B. Zustandsüberdeckung, Transitionsüberdeckung, Verzweigungsüberdeckung etc.) definiert.

Empfehlung:

- Modellierung der Anlagensteuerung beider Förderbänder mit Ausnahmebehandlung.
- Implementierung der ersten Anlagensteuerung.

5. Praktikumstermin

Pflicht (Quality Gate):

- Die Modellierung der Anlagensteuerung inkl. Fehlerbehandlung ist abgeschlossen und ist zu Beginn des Praktikumstermins vorzustellen.

Empfehlung:

- Eine Anlage sortiert bereits Werkstücke.

6. Praktikumstermin

Empfehlung:

- Die Ablaufsteuerung über beide Förderbänder inkl. Fehlerbehandlung vollständig implementieren.
- Ablaufsteuerung ausführlich testen. Test Protokolle anfertigen.
- Alle Dokumente und Projektplan aktualisieren.

7. Praktikumstermin (Endabnahme)

An diesem Termin erfolgt die Endabnahme durch die Betreuer und es erfolgt eine Reflexion über das Projekt. Pflicht:

- Gesamtanlage ist bereit für die Abnahmetests durch den Kunden.
- Alle nicht realisierten Funktionalitäten sind dokumentiert und begründet.
- Bekannte Fehler oder offene Punkte sind dokumentiert.
- Lessons Learned dokumentiert.
- Abgabe aller Artefakten des Projekts (d.h. Dokumente, Plan, Code, Protokolle etc.).

6 Zuordnung der Projektkomponenten zu den Vorlesungen

Hier sind die wichtigsten Projektkomponenten und eine Zuordnung zu den Vorlesungen aufgelistet, in denen Sie Näheres über Entwurf und Implementierung dieser Komponenten lernen:

- I/O-Hilfsfunktionen für die analogen und binären Ein- und Ausgaben einschließlich ISRs für die Interruptbehandlung (Vorlesung SY, EP).
- Hilfsfunktionen für das Message Passing über die serielle Schnittstelle (Vorlesung SY).
- Hilfsfunktionen für das Qnet Protokoll von QNX (Vorlesung SY).
- Eine *Laufzeitumgebung* für die Automaten (Vorlesung EP).
- Implementierung der Automaten (Vorlesung EP).
- Projektplanung, Entwicklungsprozess, Projektkontrolle (Vorlesung SE2).
- Konfigurationsmanagement, Qualitätsmanagement (Vorlesung SE2)
- Architektur (Vorlesung SE2)
- Implementierung von Design Patterns, Architektur Patterns (Vorlesung EP).
- Testlevels, Testkonfiguration, Testmethoden, Testcoverage, Heuristiken (Vorlesung SE2).
- Anforderungsanalyse, Software Architektur, Design Patterns, Test Überdeckungskriterien (Vorlesung SE1).

7 Hilfen

- Alle Vorlesungsskripte von SE2, SY und EP, sowie praktikumsrelevante Informationen finden Sie im EMIL-Lernraum.
- Vorlesungsskripte von SE1 finden Sie ebenfalls im EMIL.
- Jede der beteiligten Professorinnen bietet Unterstützung für das zuständige Fachgebiet an, außerdem hat jede Praktikumsgruppe eine Betreuung, die am Ende des Projekts die Schlussabnahme macht. Weiterhin unterstützen Sie die LabormitarbeiterInnen bei technischen Fragen.

Festo Signale am Beaglebone Black

GPIO 0, folgende Bits sind Eingänge

Bit	Funktion
2	Werkstück im Einlauf = low when true
3	Werkstück in Höhenmessung = low when true
4	Werkstück Höhe OK = high when true
5	Werkstück in Weiche = low when true
7	Werkstück Metall = high when true
14	Weiche offen = high when true
15	Rutsche voll = low when true
20	Werkstück im Auslauf = low when true
22	TasteStart = high, wenn betätigt
23	TasteStop = low, wenn betätigt
26	TasteReset = high, wenn betätigt
27	E-Stop = low, wenn betätigt

GPIO 1, folgende Bits sind Ausgänge:

Bit	Funktion
12	Motor Rechtslauf
13	Motor Linkslauf
14	Motor langsam
15	Motor Stop
16	Rote Lampe an
17	Gelbe Lampe an
18	Grüne Lampe an
19	Weiche öffnen

GPIO 2, folgende Bits sind Ausgänge:

Bit	Funktion
2	LED Taste Start
3	LED Taste Reset
4	Signalleuchte Q1
5	Signalleuchte Q2