

Prof. Dr. Christoph Scholl  
M. Sc. Tobias Seufert  
Tobias Faller, Timo Fritsch, Stefan Wittemer

Freiburg, 20. Mai 2020

# Praktikum Embedded Systems Engineering

## Teil 2 Steuerung eines Ottomotors

In dieser Aufgabe werden Sie die Steuerlogik für einen Ottomotor mit vier Zylindern entwerfen. Die Fallstudie besteht im Wesentlichen aus zwei Teilen:

- Eine zeitkontinuierliche Simulink-Umgebung, um das physikalische Verhalten des Motors zu modellieren.
- Zeitdiskrete Stateflow-Blöcke, welche die Steuerlogik des Motors implementieren.

Wir werden hier einen Ottomotor betrachten, der mit einem einfachen Vergaser ausgerüstet ist. Die Menge des Kraftstoff-Luft-Gemisches wird durch eine Drosselklappe im Vergaser gesteuert.

## 1 Grundlagen des Motormodells

Ein Ottomotor mit vier Zylindern ist typischerweise als Viertaktmotor ausgelegt, d.h. jeder Zylinder durchläuft immer wieder nacheinander vier Phasen:

- Ansaugen:** Ein Benzin-Luft-Gemisch wird in die Brennkammer gesaugt. Die Menge wird durch die Stellung der Drosselklappe bestimmt ( $0^\circ$ : geschlossen;  $90^\circ$ : komplett offen), welche beim Auto durch das Gaspedal beeinflusst wird.
- Verdichten:** Das Gasgemisch wird stark komprimiert.
- Arbeitshub:** Ein Zündfunke löst die Verbrennung des Gemisches (Explosion) aus. Es entsteht ein hoher Druck, der Kolben wird nach unten gedrückt und die Kurbelwelle dreht sich durch diesen Impuls weiter.
- Ausstoß:** Die Verbrennungsrückstände (Abgase) werden aus dem Zylinder gedrückt.

Eine Illustration des Ablaufs sehen Sie in Abbildung 1. Wenn die Kurbelwelle zwei volle Umdrehungen durchlaufen hat ( $720^\circ$ ), beginnt der Zyklus wieder von vorne. Damit dauert jede der vier Phasen eine halbe Kurbelwellenumdrehung ( $180^\circ$ ).

Bei einem Motor mit vier Zylindern befindet sich jeweils einer der Zylinder in einer der vier Phasen. Deshalb ist auch immer einer der Zylinder in der Verbrennungsphase und erzeugt dadurch ein Drehmoment der Kurbelwelle.

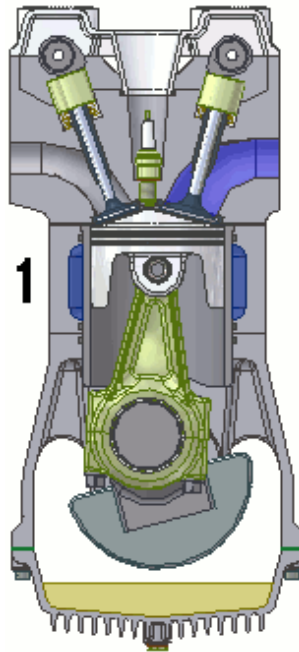


Abbildung 1: Die vier Phasen des Ottomotors. Animation aus der Wikipedia

## 1.1 Der richtige Zündzeitpunkt

Der Zündfunke wird mit Hilfe einer Zündkerze erzeugt, welche wiederum von der Zündsteuerung aktiviert wird. Der genaue Zeitpunkt ist dabei sehr wichtig, weil:

- Wenn man zu lange wartet, könnte sich das Gemisch durch den hohen Druck selbst entzünden (Adiabatische Zündung, Klopfen).
- Eine zu frühe Zündung würde den Kolben in die entgegengesetzte Richtung drücken und damit entgegen der Kurbelwellendrehrichtung wirken. Dies hätte einen bremsenden Effekt und würde mechanisch sehr hohe Belastungen erzeugen, wodurch Schäden am Motor entstehen würden.
- Es dauert einen (sehr kurzen) Augenblick, bis sich das Gemisch vollständig entzündet hat und damit die volle Wirkung entfalten kann.

Idealerweise sollte der vollständige Brand des Benzin-Luft-Gemisches beim Erreichen des oberen Totpunkts<sup>1</sup> einsetzen. Daraus folgt, dass man bereits vor Erreichen des oberen Totpunktes den Zündfunken auslösen muss, um die kurze Verzögerung bei der Verbrennung auszugleichen. Typischerweise gibt man den Zündpunkt nicht als zeitliche Größe an, sondern als Winkel der Kurbelwellenstellung. Dabei entspricht  $0^\circ$  dem oberen Totpunkt des 1. Zylinders am Übergang zwischen der Verdichtungsphase und des Arbeitshubs.

<sup>1</sup> Oberer Totpunkt: höchste Position des Kolbens im Zylinder,  $0^\circ$  bei Zylinder 1

### Zündposition $\psi_i$ :

Die Zündposition  $\psi_i$  eines Zylinders mit der Nummer  $i$  ( $i = 1, \dots, 4$ ) ergibt sich aus folgender Gleichung:

$$\psi_i = \frac{720^\circ}{4}(i - 1) - \phi ,$$

wobei  $\phi$  den Zündwinkel relativ zum oberen Totpunkt beschreibt. Je größer  $\phi$  ist, desto früher wird gezündet. Mit  $\phi = 0^\circ$  beschreibt  $\psi_i$  die obere Totpunktstellung des Kolbens im Zylinder  $i$ .

Gehen Sie für diese Aufgabe davon aus, dass es 1,5 ms dauert, bis der Zündfunke das komplette Gasgemisch entzündet hat, d.h. der ideale Zündzeitpunkt liegt 1,5 ms vor dem Erreichen des oberen Totpunkts.

Dadurch ist der ideale Wert für  $\phi$  von der aktuellen Drehzahl abhängig und muss dynamisch bestimmt werden. Würde man ein festes  $\phi$  wählen, so hätte der Motor nur für eine bestimmte Drehzahl ein optimales Zündverhalten.

## 1.2 Positionsbestimmung

Für die Steuerung der Zündung ist es unerlässlich, die genaue Position der Kolben zu kennen. Da diese fest mit der Kurbelwelle verbunden sind, reicht es aus, den Drehwinkel der Kurbelwelle zu kennen. Jeder Ottomotor ist deshalb mit Sensoren ausgestattet, die bestimmte Positionen der Kurbelwelle erkennen können.

Ein erster Sensor erkennt, wenn der 1. Zylinder den oberen Totpunkt erreicht hat. Der Wert des ausgehenden Signals geht in diesem Moment von 0 auf 1. Dieses Signal hat also nur alle  $720^\circ$  (2 Umdrehungen) eine steigende Flanke und wird *major mark* genannt.

Zusätzlich erzeugt ein zweiter Sensor 60 Mal pro Kurbelwellenumdrehung eine steigende Flanke, was einer Einteilung in  $6^\circ$ -Schritte entspricht. Dieses Signal wird mit *minor mark* bezeichnet.

Mit Hilfe dieser beiden Signale kann man nun den Drehwinkel der Kurbelwelle und damit auch die Position der Kolben bestimmen. Eine Genauigkeit von  $0.5^\circ$  ist problemlos möglich, wenn man zwischen den *minor marks*-Flanken geschickt interpoliert.

### Aufgabe, Schriftliche Ausarbeitung bis spätestens 10.06.2020:

Gehen Sie von Umdrehungsgeschwindigkeiten der Kurbelwelle zwischen 500 U/min und 7000 U/min aus. Berechnen Sie für diese beiden Extremwerte:

- a) Die Anzahl der Umdrehungen pro Sekunde.
- b) Die Winkelgeschwindigkeit in der Einheit  $^{\circ}/s$ .
- c) Die Zeit, die zwischen zwei *major marks* vergeht.
- d) Die Zeit, die zwischen zwei *minor marks* vergeht.
- e) Die zeitliche Differenz zwischen
  - 1) zwei aufeinanderfolgenden Zündungen im gleichen Zylinder.
  - 2) den Zündungen zweier aufeinanderfolgender Zylinder in der Zündreihenfolge.Gehen Sie hier davon aus, dass  $\phi$  konstant ist.
- f) Die maximale zeitliche Abweichung des Zündzeitpunktes vom Idealwert, wenn bei der Positionbestimmung der Kurbelwelle ein Fehler von maximal  $0,5^{\circ}$  toleriert wird.
- g) Die Drehwinkel der Kurbelwelle nach  $10\mu s$ .

Benutzen sie als Einheiten für Winkel Grad und für die Zeit Sekunden (bzw.  $ms/\mu s$ ).

**Die Ausarbeitung muss digital als PDF erstellt und bis zum 10.06.2020 14:00 per E-Mail abgegeben werden.**

Tobias Faller: fallert@tf.uni-freiburg.de,

Timo Fritsch: timo.fritsch@uranus.uni-freiburg.de,

Stefan Wittmer: stefan.wittmer@uranus.uni-freiburg.de

## 2 Das Simulink Modell

Öffnen Sie die Datei `motorsteuerung.mdl` (Download über die Praktikumsseite). Es öffnet sich das Simulink-Modell aus Abb. 2.

Die Blöcke sind farblich in die folgenden Gruppen eingeteilt:

**Orange:** In diesen Blöcken befindet sich das physikalische Modell des Motors. Bitte verändern Sie diese Blöcke nicht.

**Grün:** Das sind die Stateflow-Blöcke, welche Sie in dieser Aufgabe bearbeiten werden.

**Hellblau:** Dies sind Ausgaben (Displays, Scopes), mit ihnen können Sie alle wichtigen Werte während der Simulation beobachten. Wenn nötig, können Sie diese gerne erweitern oder zusätzliche Ausgaben hinzufügen.

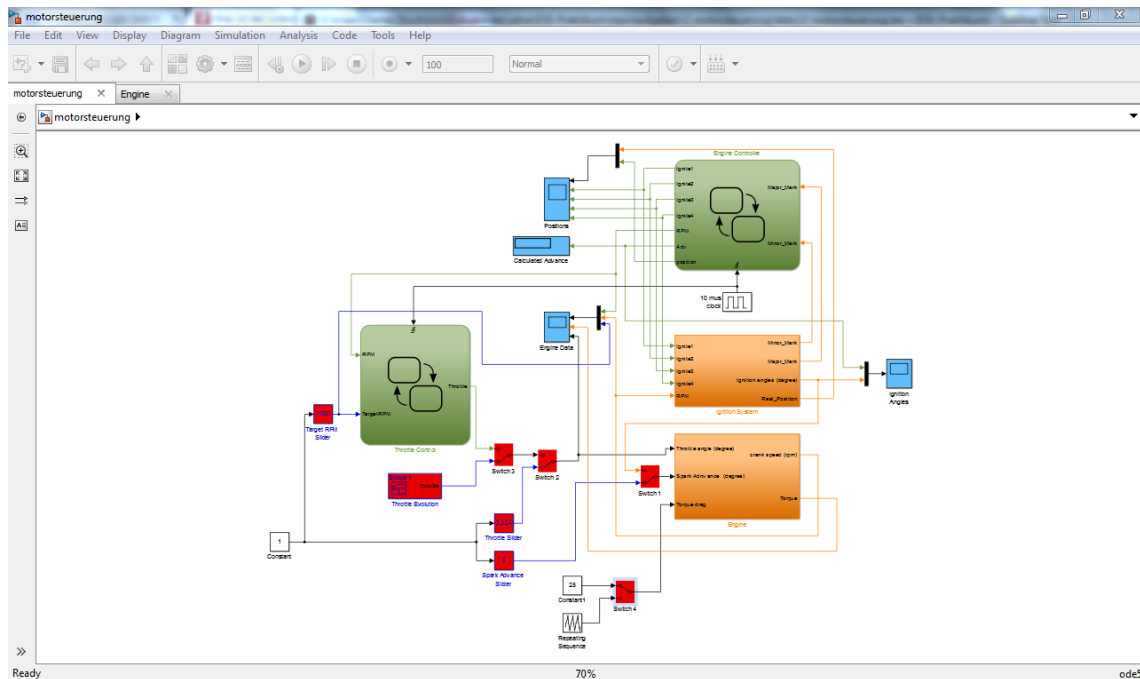


Abbildung 2: Das Simulink Motormodell

**Rot:** Hiermit können Sie die Eingaben an das Motormodell kontrollieren, dies ist insbesondere wichtig für den Anfang, wenn Ihre automatische Steuerung noch nicht fertig ist, bzw. noch nicht korrekt funktioniert.

Im den folgenden Abschnitten werde die einzelnen Blöcke mit ihren Ein- und Ausgängen genauer beschrieben.

## Engine

Dieser Block enthält die Simulink-Modellierung des Motors. Hier werden alle relevanten physikalischen Parameter berechnet. Dieser Block wertet die Zündsignale aus und berechnet daraus den aktuellen Zündwinkel  $\phi$ . Dieser ergibt sich aus der Position der Kurbelwelle zum Zeitpunkt der Zündung.

## Eingänge

**IgniteX** Über diese Eingänge werden die Zündsignale abgefragt. Sollten die Zündsignale zu spät oder gar nicht erzeugt werden, erfolgt eine adiabatische Selbstentzündung aufgrund der Kompression zufällig zwischen  $0^\circ$  und  $15^\circ$  nach dem Totpunkt.

**Throttle Angle** Dies ist die Stellung der Drosselklappe. Der Wert muss zwischen 0 und 90 liegen. Dieser Eingang kann während der Simulation von Ihnen beeinflusst werden. Später entwerfen Sie hierfür eine automatische Regelung.

**Spark Advance** Dieser Eingang entspricht dem Winkel  $\phi$ . Der Wert ergibt sich automatisch durch die Zündsignale, die Sie im Laufe dieser Aufgabe erzeugen werden. Für Testzwecke können Sie diesen Wert auch über einen Schieberegler einstellen.

**Torque Drag** Der letzte Eingang beschreibt die Last, die dem Motordrehmoment entgegenwirkt, beispielsweise durch mechanische Widerstände des Motors selbst oder durch die zu verrichtende Arbeit. In der realen Welt ist die Last vermutlich selten konstant, deshalb kann über einen Schalter zwischen einer konstanten und einer veränderlichen Last umgeschaltet werden.

## Ausgänge

**Real Position** Der tatsächliche Drehwinkel der Kurbelwelle.

**Major Mark** Das Positionssignal der Kurbelwelle für die  $0^\circ$  Stellung.

**Minor Mark** Das Positionssignal der Kurbelwelle für die  $6^\circ$  Unterteilung.

**Torque** Das berechnete Drehmoment. Dieses bestimmt sich aus der Kraft, mit der der Kolben während des Arbeitshubs über den Pleuel auf den Exzenter der Kurbelwelle drückt.

**Crank Speed** Die aktuelle tatsächliche Drehzahl des Motors.

**Calc. Spark Advance** Dies ist der berechnete Zündwinkel, der sich aus den Zündsignalen ergibt.

### Hinweis:

Die Signale **Real Position**, **Torque**, **Crank Speed** und **Calc. Spark Advance** dürfen nicht in der Steuerlogik (grüne Blöcke) benutzt werden. Diese Signale dienen nur der Visualisierung und Ergebniskontrolle.

Sollte die Drehzahl auf 500 U/min oder weniger absinken, stoppt die Simulation. Dies entspricht einem Absterben ("Abwürgen") des Motors bei zu geringer Drehzahl.

## Steuerung des Modells

Wie bereits erwähnt, kann mit den roten Blöcken die Simulation beeinflusst werden. Es gibt folgende Möglichkeiten:

**Switch 1:** Hiermit kann gesteuert werden, ob der Zündwinkel manuell oder dynamisch aus den Zündsignalen bestimmt wird. Solange Sie die Zündsignale noch nicht erzeugen, empfiehlt sich die Benutzung der manuellen Steuerung durch den Schieberegler.

**Switch 2:** Mit diesem Schalter kann man zwischen einem manuell eingestellten Winkel der Drosselklappe oder einem sich selbständig verändernden Wert wählen.

**Switch 3:** Dieser Schalter wählt zwischen einer automatischen Drosselklappensteuerung (wird später implementiert) oder einer vorgegebenen Kurve aus.

**Spark Advance Slider:** Dies ist ein Schieberegler, um einen Zündwinkel manuell einzustellen.

**Throttle Slider:** Hiermit können Sie die Stellung der Drosselklappen verändern. Bei  $90^\circ$  ist die Klappe vollständig geöffnet und der Motor sollte die volle Leistung erbringen.

**Target RPM Slider:** Im zweiten Teil dieser Aufgabe können Sie mit diesem Regler die Zieldrehzahl bestimmen. Ihre Regelung muss dann versuchen, diese zu erreichen.

**Throttle Evolution:** Hiermit kann man einen vorgegebenen Verlauf des Drosselklappenwinkels definieren.

## Ausgaben

**Engine Data:** Hier werden die drei wichtigsten Motorparameter dargestellt. Das erste Scope zeigt die relevanten Drehzahlen (real, gemessen und Vorgabe). Im zweiten Scope folgt das Drehmoment des Motors und die Last. Sind beide Werte gleich groß befinden Sie sich das System im Gleichgewicht und die Drehzahl verändert sich nicht. Das dritte Scope zeigt die Stellung der Drosselklappe.

**Ignition Angles:** Dieses Scope vergleicht den von Ihnen berechneten Zündwinkel mit dem Zündwinkel, der sich aus den Zündsignalen ergibt.

**Positions:** Diese Anzeige ist für den ersten Teil Ihrer Aufgabe relevant. Im obersten Diagramm sehen die tatsächliche und von Ihnen berechnete Position der Kurbelwelle im direkten Vergleich. Darunter folgen die vier Zündsignale. Sie können damit genau erkennen zu welchem Zeitpunkt (Winkel) ein Zündsignal von 0 auf 1 geht.

### Aufgabe:

Starten Sie nun die Simulation und öffnen Sie das Scope **Engine Data**. Sie sehen, dass zunächst das Drehmoment über der Last liegt und dadurch die Drehzahl ansteigt, doch schnell ist ein Gleichgewicht erreicht und die Drehzahl verändert sich nicht mehr.

Bitte ignorieren Sie momentan die Warnungen über nicht benutzte Signale. Die Warnungen werden erzeugt, weil die Stateflow-Blöcke noch nicht implementiert sind und deshalb einige Signale nicht verwendet werden. Beachten Sie auch, dass das Modell einen kurzen Moment benötigt, bis überall sinnvolle Werte anliegen.

Ein ähnliches Problem gibt es auch bei realen Motoren: Der Anlasser muss den Motor erst einmal komplett durchdrehen, damit die Steuerlogik die Position der Kolben berechnen kann. Vorher wird keine Zündung ausgelöst, da eine Fehlzündung den Motor beschädigen könnte. Im vorliegenden Modell ignorieren wir dieses Problem.

Verändern Sie nun nacheinander den **Throttle Slider** und den **Spark Advance Slider**. Beobachten Sie den Einfluss dieser beiden Regler auf das Verhalten der Drehzahl.

Machen Sie sich mit dem Modell vertraut, bevor Sie mit der Aufgabe fortfahren.

### 3 Entwurf der Zündsteuerung

#### Aufgabe:

Implementieren Sie nun den Stateflow-Block **Engine Controller**. Lesen Sie sich die Aufgabenstellung erst vollständig durch und überlegen Sie sich zunächst, wie Sie das Zustandsdiagramm organisieren möchten (z.B. Aufgabenverteilung, Hierarchie). **Es ist sehr hilfreich, Ihre Überlegungen schriftlich zu dokumentieren (evtl. auch durch eine Grafik), bevor Sie mit der Implementierung anfangen. Sie müssen nicht abgegeben werden.**

Das Zustandsdiagramm wird durch eine `clock` mit einer Zykluslänge von  $10\mu\text{s}$  getriggert, d.h. alle  $10\mu\text{s}$  kann ein Zustandsübergang stattfinden.

Der Controller besitzt zwei Eingänge, `Major_Mark` und `Minor_Mark`. Mit Hilfe dieser beiden Signale müssen Sie die Position der Kurbelwelle und die Drehzahl bestimmen.

#### Hinweis:

Nutzen Sie bei beiden Signalen bitte nur die steigende Flanke. Es wäre prinzipiell zwar möglich mit Hilfe der fallenden Flanke die Präzision zu verdoppeln, dass soll hier aber nicht ausgenutzt werden. Gehen Sie einfach davon aus, dass die Signale zu einem zufälligen Zeitpunkt wieder auf 0 gesetzt werden.

Ihre ermittelte Drehzahl übergeben Sie bitte dem Ausgang `RPM`, damit diese mit der tatsächlichen Drehzahl verglichen werden kann. Ebenso übergeben Sie bitte Ihre berechnete Position dem Ausgang `position`. Normieren Sie diesen Wert auf  $720^\circ$ .

Mit Hilfe Ihrer Drehzahl können Sie nun den Winkel  $\phi$  bestimmen und damit auch für jeden Zylinder ein Zündsignal erzeugen (Ausgänge `Ignite1`, ..., `Ignite4`). Das Zündsignal soll genau zum Zündzeitpunkt von 0 auf 1 wechseln (steigende Flanke). Belassen Sie das Signal einige Clockzyklen auf 1, bevor Sie es wieder auf 0 setzen. Schreiben Sie den berechneten Wert des Winkels  $\phi$  auf den Ausgang `Adv`, damit er zur Kontrolle visualisiert werden kann.

Beachten Sie, dass schlecht gewählte Zündzeitpunkte einen negativen Effekt auf die Motorleistung haben. Versuchen sie die Position der Kurbelwelle möglichst exakt zu bestimmen. Sie müssen hierzu zwischen zwei aufeinanderfolgenden Minor-Marks-Signalen interpolieren. **Berechnen Sie Ihren maximalen Fehler.**



## Checkpoint:

Führen Sie Ihren Entwurf der Zündsteuerung einem Betreuer vor und besprechen Sie mit ihm die bislang gestellten Aufgaben.

Kriterien für den Checkpoint:

- Position der Kurbelwelle muss auf mindestens  $0,5^\circ$  genau bestimmt werden (Abweichung zu **Real Position**).
- Die Drehzahl muss mindestens 6000 U/m erreichen. Besser: 6300.
- Welche maximale Drehzahl können Sie mit Ihrer Lösung erreichen, wenn Sie die Drosselklappe voll öffnen ( $90^\circ$ )?

## 4 Entwurf einer Drehzahlregelung

Nehmen Sie an, Sie möchten den Motor in einer Umgebung betreiben, bei der es darauf ankommt, dass der Motor eine bestimmte Zieldrehzahl einhält (z.B. Seilwinde oder Stromaggregat). Eine Regelung soll automatisch dafür sorgen, dass der Motor die vorgegebene Drehzahl zunächst erreicht und dann einhält. Die Zieldrehzahl kann über den Schieberegler (**Target RPM Slider**) eingestellt und während der Simulation verändert werden.

Die Drehzahl wird im **Engine Controller** bestimmt und hier verwendet. Sie kann nur über die Stellung der Drosselklappe beeinflusst werden.<sup>2</sup>

### Aufgabe:

Implementieren Sie den Stateflow-Block **Throttle Control**. Ausgehend von der aktuellen Zieldrehzahl (sie kann sich ändern) und der gemessenen Momentandrehzahl muss ständig die Stellung der Drosselklappe korrigiert werden, Versuchen Sie eine Regelung zu entwerfen, die die Zieldrehzahl möglichst schnell erreicht und dann einhält.

Beachten Sie, dass sich das Motormodell träge verhält, d.h. Änderungen an der Position der Drosselklappe führen nur mit etwas Verzögerung zu einer entsprechenden Reaktion des Drehmoments bzw. der Drehzahl. Dies resultiert u.a. aus der Massenträgheit des gesamten Systems. Dadurch kann es vorkommen, dass die Motordrehzahl um die Zieldrehzahl oszilliert, weil die Regelung auf Veränderungen zu langsam reagiert. Versuchen Sie solch ein Verhalten möglichst zu vermeiden, bzw. gering zu halten. Sie müssen sich hierzu überlegen, wie man Änderungen am Motorverhalten möglichst frühzeitig erkennen kann und wie groß die Änderungen an der Drosselklappenstellung jeweils sein sollen.

Durch **Switch 4** können Sie zwischen einer konstanten und einer veränderlichen Last wählen. Die Last wirkt dem Drehmoment des Motors entgegen und beeinflusst dadurch die Drehzahlentwicklung.

---

<sup>2</sup> Die Zündzeitpunkte haben natürlich auch einen Einfluss auf die Drehzahl. Dies wird in modernen Motoren u.a. dafür ausgenutzt, um ggf. schnell das Drehmoment zu verringern. In diesem Modell wollen wir dies aber nicht tun.

### Schriftliche Ausarbeitung bis 24.06.2020 14:00

Beschreiben Sie die wesentliche Funktionsweise Ihrer Motorsteuerung. Der Umfang darf vier Seiten nicht überschreiten. Beschränken Sie sich auf das Wesentliche, benutzen Sie Grafiken und schematische Flowcharts. Vermeiden Sie direkte Screenshots aus ihrer Implementierung, erstellen Sie die Grafiken selbst. Wir empfehlen die Verwendung von  $\text{\LaTeX}$ .

**Die Ausarbeitung muss digital als PDF erstellt und bis zum 24.06.2020 14:00 per E-Mail abgegeben werden.**

Tobias Faller: fallert@tf.uni-freiburg.de,

Timo Fritsch: timo.fritsch@uranus.uni-freiburg.de,

Stefan Wittemer: stefan.wittemer@uranus.uni-freiburg.de

Beispielhafte Schwerpunkte:

- Wie funktioniert ihre Drehzahlmessung, wie genau ist sie?
- Für welche Reglerart haben Sie sich entschieden, warum?
- Wie haben Sie Ihren Regler konfiguriert (Trial-and-Error/Systematisch)?
- Wie verhält sich Ihre Regelung wenn sich die Last verändert?
- Wie schnell reagiert sie auf die Veränderungen?
- Haben Sie zusätzliche Features implementiert? Wenn nicht, welche wären sinnvoll?
- Wo gab es Probleme, wie haben Sie sie gelöst?

Beschreiben Sie abschließend in wenigen Sätzen Ihre bisherigen Erfahrungen im Praktikum, ihren subjektiv empfundenen und tatsächlichen Arbeitsaufwand sowie Verbesserungsvorschläge und Anregungen.

### Checkpoint:

Bevor Sie mit Teil 3 des Praktikums fortfahren, führen Sie bitte einem Betreuer die Drehzahlregelung vor. Ihre Regelung muss eine gegebene Drehzahl erreichen und einigermaßen stabil halten können.