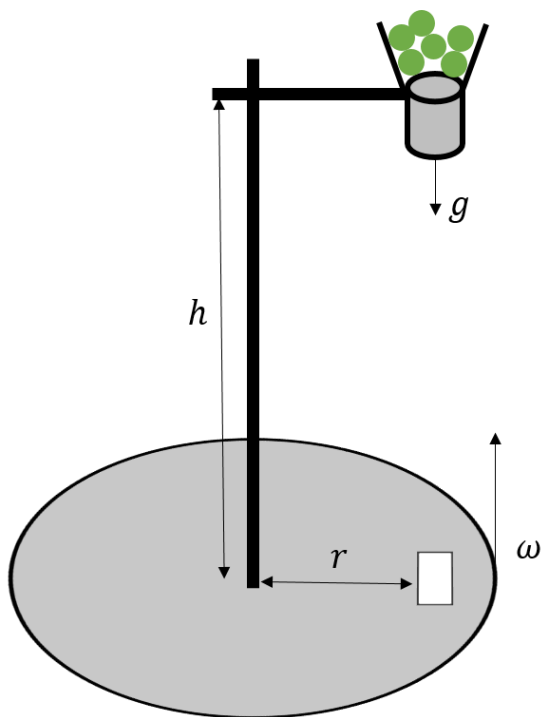


Laborbericht KIS Versuchsaufbau Kugelfall

Aufgabe 1 – Systemanalyse

Aufbau:



Sensoren:

- Lichtsensor
- Hallsensor
- Taster

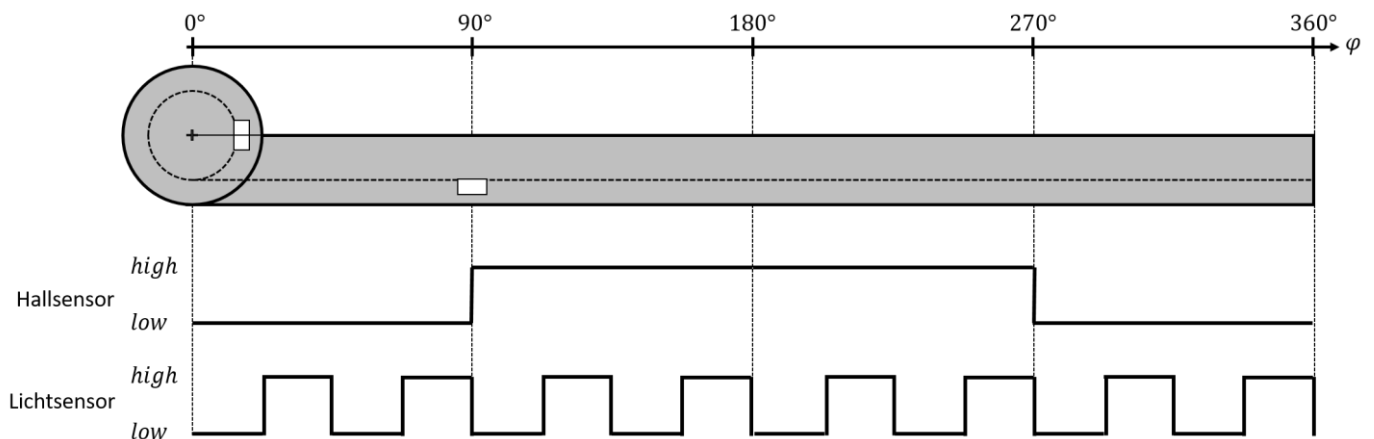
Aktuatoren:

- Servomotor

Identifizierte Probleme:

- Drehscheibe besitzt Reibung
- Kugel ist keine Punktemasse
Scheibe darf nicht zu schnell sein -> sonst kann die Kugel nicht passieren
- Scheibe darf sich nicht zu langsam drehen, da die Berechnungen dann fehlerhaft sind
- Fallzeit der Kugel ist nicht exakt bestimmbar
- Motor benötigt eine gewisse Zeit bevor die Kugel freigegeben wird
- Motor muss exakt eingestellt werden, da sonst mehrere Kugeln freigegeben werden
- Aufbau ist nicht symmetrisch

Signalverläufe:



Lösungsvorgehen:

1. Der Hallsensor identifiziert die halben Umdrehungen, sodass erkannt werden kann, wo sich das Loch befindet
2. Der Lichtsensor liefert für jede Umdrehung deutlich mehr Pegeländerungen, die gemessen werden können. Darüber ist es möglich deutlich genauer die aktuelle Geschwindigkeit abzuleiten, als an dem Hallsensor.
3. Über die Geschwindigkeit kann in Kombination mit dem Hallsensor ermittelt werden, wie lange es theoretisch noch dauert, bis der Auslass für die Kugeln das nächste Mal das Loch passiert.
 - a. Das größte Problem, das an dieser Stelle gelöst werden muss, ist die Interpolation der Geschwindigkeit
 - b. Mathematisch kann kein exaktes Modell ermittelt werden, da keine Laborbedingungen vorliegen und es keine Möglichkeit gibt, die Reibung zu charakterisieren
 - c. Annahme: Reibung ist nicht linear, also hängt in irgendeiner Weise von der Geschwindigkeit der Scheibe ab, weswegen die Geschwindigkeit stückweise linear interpoliert werden kann
4. Falls sich die Geschwindigkeit in zu kurzer Zeit zu stark ändert, sollte keine Kugelfreigabe durchgeführt werden, da ansonsten die lineare Interpolation keine guten Ergebnisse liefert.
5. Die Fallzeit der Kugel und die Zeit, die der Motor benötigt die Kugel freizugeben, muss zuvor experimentell bestimmt werden. Anschließend, kann bestimmt werden, wie lange im Voraus die Kugel freigegeben werden muss.
6. Da der Aufbau nicht symmetrisch ist, muss zu Beginn die Drehrichtung der Scheibe festgelegt werden.

Ermittelte Größen:

1. Radius bis zum Loch: 0,170m
2. Maße des Loches: 0,065m
3. Fallhöhe: 0,750m

Aufgabe 2 - Design

Genereller Lösungsansatz

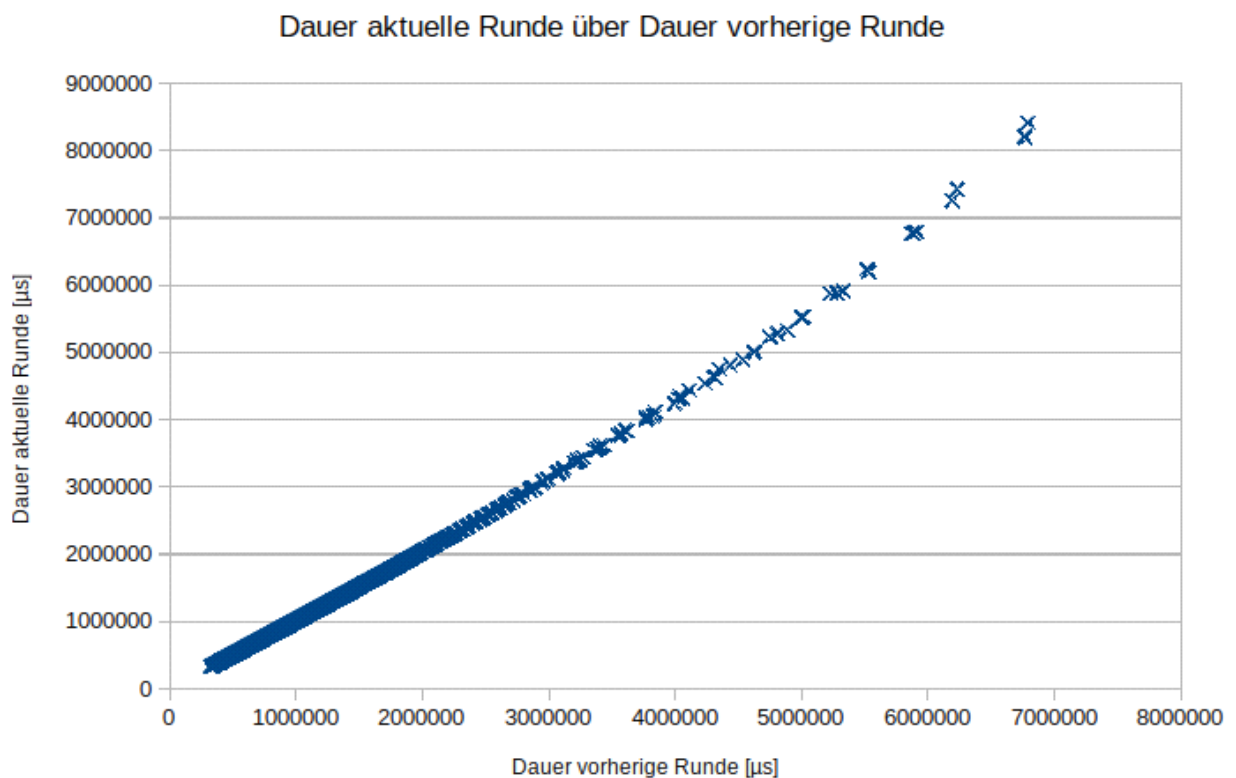
Die Fallzeit der Kugel aus der gemessenen Fallhöhe von 0,75m wird wie folgt bestimmt:

$$t_{fall} = \sqrt{\frac{2h}{g}} = 0,391s$$

Sobald die letzte Umdrehungsdauer der Scheibe bekannt ist, kann eine Schätzung für die nächste Umdrehung erstellt werden.

Aus einer gemessenen Rundendauer kann die darauf folgende Rundenzeit approximiert werden. Dazu haben wir zunächst überprüft, wie sehr sich die Rundendauer im Vergleich zu der vorhergehenden verändert. Wie man an der nachfolgenden Grafik erkennen kann, nimmt die Rundendauer von Runde zu Runde, näherungsweise linear ab. Daraus ergibt sich, dass die Rundendauer von Runde zu Runde gut approximierbar ist.

Weiterhin ist zu erkennen, dass bei langsamer Geschwindigkeit, also langer Rundendauer, die Approximation von ihrem linearen Verhalten abweicht. Für eine gute Näherung ist also eine Mindestgeschwindigkeit erforderlich.



Über unterschiedliche Rundenzeiten, war es nicht möglich die darauffolgende Rundenzeit zu approximieren, dass bei jeder Geschwindigkeit richtig ausgelöst wurde. Daher wurden für unterschiedliche Rundendauern verschiedene Approximationspolynome ermittelt. Der Verlauf ist in folgende Intervalle unterteilt:

p1 für [0s ;0.5s)

p2 für [0.5s ; 2s)

p3 für [2s ; inf)

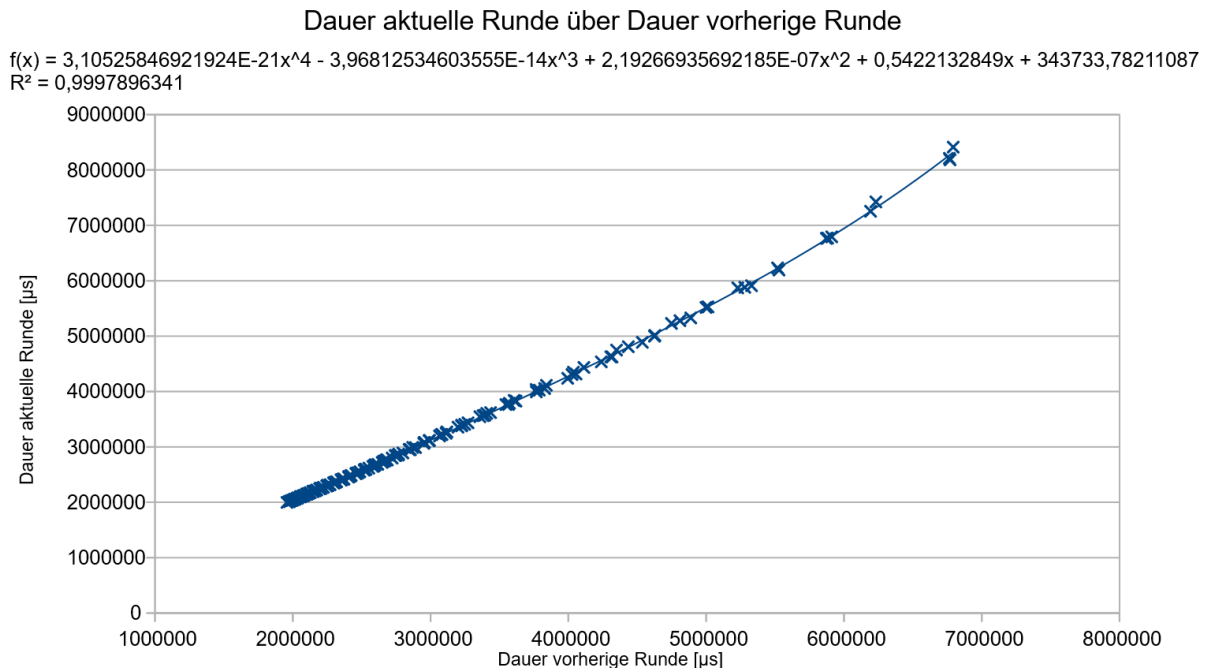
$$p_1(t) = 7,73699143114947 \cdot 10^{-9} \cdot t^2 + 0,998685885 \cdot t + 268,2445651096$$

$$p_2(t) = 1,2897880394751110 \cdot 10^{-8} \cdot t^2 + 0,9910487474 \cdot t + 2585,5449062138$$

$$p_3(t) = 3,10525846921924 \cdot 10^{-21} \cdot t^4 - 3,96812534603555 \cdot 10^{-14} \cdot t^3 + 2,19266935692185 \cdot 10^{-7} \cdot t^2 + 0,5422132849 \cdot t + 343733,782110879$$

$$t_{next}(t) = \begin{cases} p_1(t) & \text{für } t < 500\text{ms} \\ p_2(t) & \text{für } 500\text{ms} \leq t < 2\text{s} \\ p_3(t) & \text{für } t \geq 2\text{s} \end{cases}$$

Die Werte der Polynomkoeffizienten wurden durch eine lineare Regression aus aufgenommenen Messwerten ermittelt. Beispielhaft wird das ermittelte Polynom für den Abschnitt A3 in folgender Grafik skizziert:

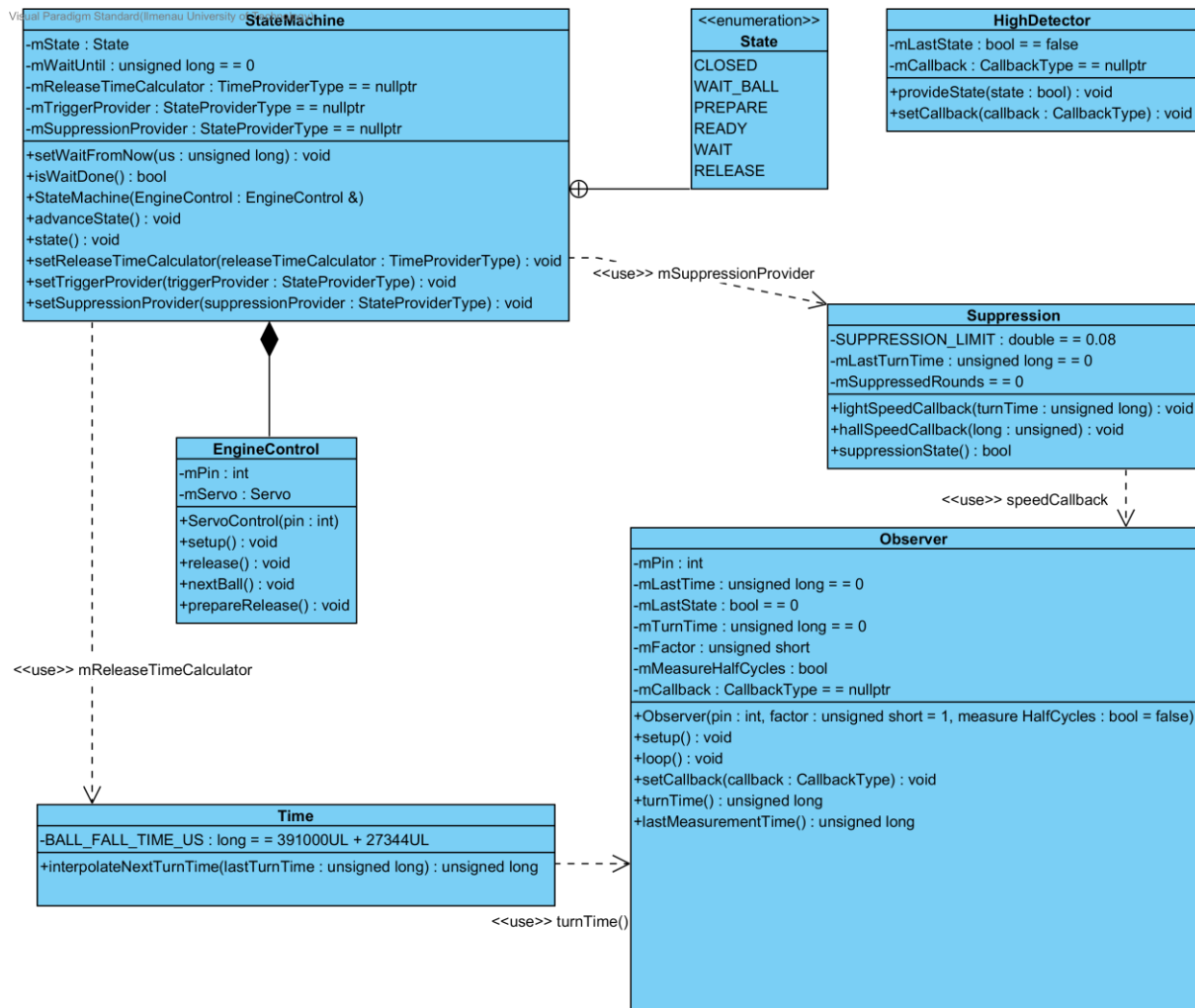


Mit der zuvor errechneten Falldauer und der Approximation der Dauer der nachfolgenden Runde, kann ein ausreichend genauer Zeitpunkt bestimmt werden, wann die Kugel freigegeben werden soll. Zusätzlich zu muss noch betrachtet werden, dass der Motor für die Bewegung eine gewisse Zeit benötigt und somit zwischen Tastendruck und Freigabe eine gewisse Zeit vergeht. Zur Minimierung der Verzögerung durch den Auslösemechanismus am Stellmotor, wird dieser nicht nur komplett geschlossen bzw. geöffnet, sondern auch in einen Zwischenzustand (halb-offen)überführt. Darüber hinaus muss bei der Implementierung beachtet werden, dass bei hohen Drehgeschwindigkeiten, die Scheibe innerhalb der Fallzeit der Kugel, ggf. mehrere Umdrehungen macht. Um die Position des Loches auf der Scheibe zu identifizieren, wird der Hall-Sensor verwendet, welcher die jeweils halben Umdrehungen markiert.

Weiterhin ist es eine Anforderung, dass sprunghafte bzw. untypische Änderungen der Drehbewegung erkannt werden und eine Freigabe der Kugel unterdrückt wird. Um dieses Verhalten umzusetzen wird von Runde zu Runde die Geschwindigkeitsänderung verfolgt. Ist diese Änderung zu groß, würde der Kugelfall zu stark vom ermittelten Fallmodell abweichen und das Loch verfehlen. Daher wird bei einem Überschreiten eines Schwellwertes der Geschwindigkeitsänderung, die Freigabe der Kugel unterdrückt.

Softwareentwurf

Im Folgenden ist ein Entwurf in Form eines Klassendiagramms abgebildet:



Folgende Programmteile wurden als Kernkomponenten identifiziert:

- **StateMachine:** Zustandsmaschine die den Ablauf steuert
- **EngineControl:** Steuerung des Motors mit den Zuständen:
 - **nextBall():** vollständig schließen, nächste Kugel fällt durch
 - **prepareRelease():** Ist ein Zwischenzustand, um die Zeit bis zur Freigabe der Kugel zu minimieren
 - **release():** vollständig öffnen, Kugel fällt herunter

- Time: Berechnung der Wartezeit bis zum Abwurf aus der Dauer der letzten Runde und der in der momentanen Runde schon verstrichenen Zeit
- Observer: Überwachen der Flankenwechsel eines Signals und daraus Errechnung der Rundendauer
- Suppressor: Überwachen der Rundendauern und wenn nötig Ausgabe eines Sperrsignals bei zu starker Änderung
- HighDetector: Erkennt High-Flanken im Signal der Sensoren

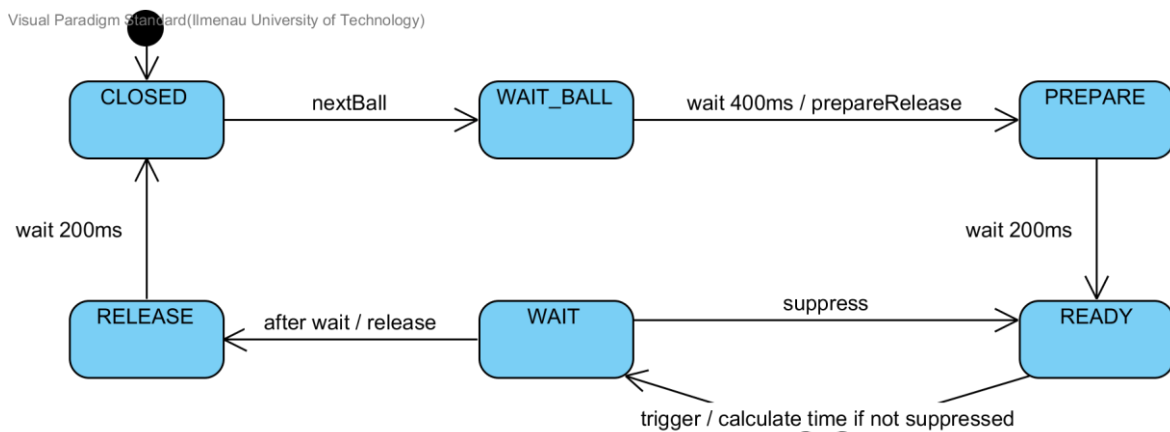
Jede Klasse wird global einmal instanziiert. Eine Ausnahme bildet lediglich die Observer-Klasse mit 2 Instanzen, die je eine Instanz für Hall- und Photosensor besitzt.

Den zentralen Start des Programmablaufs bildet die setup() Funktion, die zu beginn durch das Arduino Framework aufgerufen wird. Dort werden die verwendeten Pins des Arduino Boards konfiguriert und alle Instanzen über Function-Pointer erstellt. Anschließend an das Setup geht das Programm in die Arduino-typische Programmschleife, in die loop() Funktion über.

In der loop()-Funktion werden sowohl die Observer, als auch die Statemachine periodisch abgearbeitet.

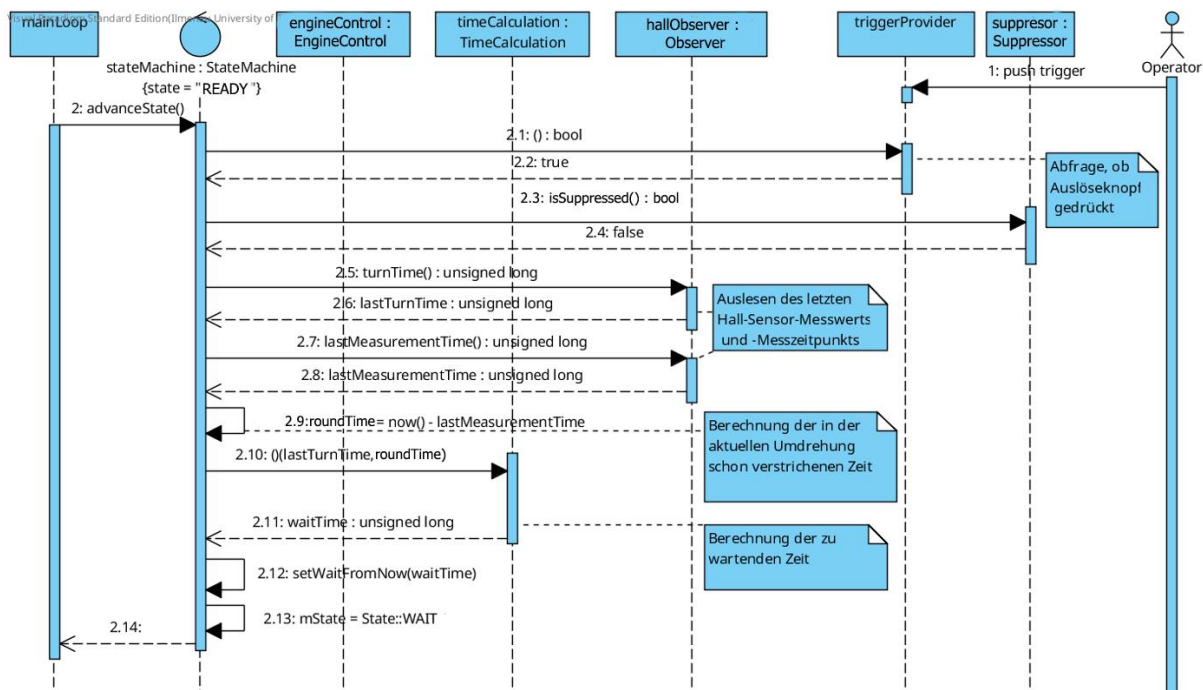
Zustandsmaschine

Die Zustandsmaschine besitzt 6 Zustände und folgende Übergänge:



Jeder Zustand der Statemachine repräsentiert dabei auch einen bestimmten Zustand des Motors, der bei den Übergängen eingestellt wird. Da der Motor eine gewisse Zeit benötigt, um die einzelnen Befehle auszuführen, wurden Wartezeiten in die Statemachine eingebaut, um sicherzugehen, dass die gewünschte Motorposition auch erreicht wurde, bevor der Programmablauf fortgesetzt wird.

Beispielhaft wurde für den Zustand "READY" die vollständige Interaktion zwischen den anderen Klassen und auch dem User visualisiert:



Machbarkeitsanalyse

Das Loch besitzt eine mittlere Länge von 0,065m und befindet sich in einem Abstand von 0,17m zum Mittelpunkt. Damit ergibt sich, dass das Loch etwa 6% einer vollen Umdrehung einnimmt.

Bei einem kleinen Sicherheitsabstand, damit die Kugel nicht von den Kanten abprallt, werden 5%, anstatt 6% angenommen. Für eine beispielhafte Umdrehungsdauer von 400ms ergibt sich damit ein Zeitfenster von 20ms, in der die Kugel das Loch passieren kann. Aus diesem Zeitfenster ergibt sich, dass sich die Scheibe nicht zu schnell drehen darf, da die Kugel ansonsten das Loch nicht in dieser Zeit passieren kann.

Aus den Anforderungen ergibt sich ebenfalls eine Wahrscheinlichkeit von 80% mit der das Loch getroffen werden soll.

Aufgabe 3 - Implementierung

Um die Fallzeit der Kugel annäherungsweise exakt zu bestimmen, wurde eine Suche zwischen 391 und 491ms durchgeführt. Dabei ergab sich in diesem Zeitraum der beste Wert bei ~418ms.