Kugelfall

Aufbau des KIS Labors SS2018

**Professor Dr.-Ing. habil. Armin Zimmermann**

Lehrbeauftragter an der TU-Ilmenau

Fachbereich Mechatronik

Bearbeitung von

**Mohamad Akkad**

**Yutong Xiao**

**Inhaltsverzeichnis**

1. Einführung ...............................................................................................................04
2. Vorbemerkungen......................................................................................................04

2.1 Zielgruppe .........................................................................................................04

1. Beschreibung der Elemente .....................................................................................05
   1. Arduino Uno......................................................................................................05
   2. Photoelektrischer Näherungsschalter (LRS 1120- 304) ...................................05
   3. Hall Effect Gear Tooth Speed Sensors CYGTS211/212 ….............................06
   4. Drehscheibe ......................................................................................................06
   5. Servo Motor ......................................................................................................07
   6. "Blackbox" mit der elektronischen Verschaltung ............................................08
   7. Trigger/ Switch .................................................................................................08
   8. Arduino IDE ......................................................................................................08
2. Interpretation der Arbeitsweise der Sensoren ..........................................................09
3. Algorithmus Design ..................................................................................................10

5.1.1 Mathematische Grundlage ...............................................................................10

5.1.2 Klasse Design .................................................................................................11

5.1.3 Beschreibung vom Algorithmus ......................................................................12

1. Fehleranalyse.............................................................................................................13

**1 Einführung**

Wir werden in unserm Bericht über die Tools, die wir verwendet haben,

Welche Algurithmus haben wir erfolgt und Entwickelt, um under Zweck

Welche Anwendungssoftware haben wir verwendet und welche Probleme, Schwierigkeiten und die Fehler haben uns konfrontiert? Wie haben wir damit umgegangen?

Der Versuch besteht aus Verschaltung, Sensoren, Aktoren, Körper des Systems und Arduino Uno.

**2 Vorbemerkungen**

**2.1 Zielgruppe**

In unserem Versuch geht es darum, mit Ardoinu Uno , Programmierung und Entwicklungsumgebung vertraut zu machen und erfolgreiche Steuerung der Kugelfallen mit der Erfolgsquote von über 80% in einem Loch auf einer Drehscheibe durchzuführen.

**Elemente des Projetes:**

1. Arduino Uno
2. Photoelektrischer Näherungsschalter (LRS 1120- 304)
3. Hall Effect Gear Tooth Speed Sensors CYGTS211/212
4. Cam Kamera
5. Drehscheibe
6. Servo Motor
7. "Blackbox" mit der elektronischen Verschaltung
8. Trigger/ Switch
9. Und schlussendlich die Software Arduino IDE

**3 Beschreibung der Elemente**

**3.1 Arduino Uno**

Arduino ist eine Open-Source-Elektronik–Plattform, wurde ausgelegt und umgesetzt, um, zuallererst, zwischen Software und Hardware zu integrieren und zusammenzuhängen und danach, um alle Altergruppen zu passen. Also es ist nicht lediglich für die Amateure oder die profi von Programmierer/ Programmiererinnen geziehlt sondern auch für die Jugendlichen, die Interesse an der Programmierung haben.

**Abbildung (1)**

Man kann auch Arduino Uno als Entwicklungsplatine betrachten.

Arduino Uno Plattform verwendet den Microcontroller Atmega328, der 14(Ein/ Ausgänge) hat, die man als Digital In/out benüzen kann. 6 von diesen 14 könnte man als PMW(Pulse-Width modulation) also als Analog In/ Out verwenden.

Arduino enthlät einen Kristal Oszilator mit 16MHz, USB Eingang, um mit dem Computer zu kommunizieren und auch einen zusätzlichen Zugang für die Energieversorgung.

**3.2 Photoelektrischer Näherungsschalter (LRS 1120- 304)**

Der Sensor ist nicht auf unmittelbare Berührung angewiesen sondern auf die Annährung von einer Oberfläche.

Man verwendet ihn, um Z.B die Geschwindigkeit der Drehscheibe zu ermitteln.

In unserem Versuch ist der Sensor ein optischer Näherungsschalter, der auf Lichtreflexion reagieren.

**Abbildung (2)**

* 1. **Hall Effect Gear Tooth Speed Sensors CYGTS211/212**

Der Hall-Sensor wird als Positionierungssensor verwendet. Grundidee ist, dass der Sensor eine Ausgangsspannungs hat, wenn er sich in ein verlaufendes konstantes magnetisches Feld befindet, das senkrecht auf dem Sensor ist.

**Abbildung (3)**

* 1. **Drehscheibe**

Die Drehscheibe hat von der Oberseite ein Loch, wodurch die fallenden Kugeln fallen dürfen.

Von der unterseite ist die Drehseiben aus 12 Sektorn geteilt. Jeder Sektor ist entweder Weiß oder Schwarz jedoch farbig unterschiedlich sequenziell.

Die Drehscheibe verfügen über 2 kleine Magneten, die symmetrisch zueinander sind und nützlich zur Bestimmung des Loches.

* 1. **Servo Motor**

Ein sehr effizientes Element des Projekts, da er die Möglichkeit erlaubt, seine Geschwindigkeit, Beschleunigung und die Winkelposition anzusteuern.

Die Aufgabe des Motors ist, eine Kugel ins Loch lassen zu fallen.

**Abbildung (4)**

* 1. **"Blackbox" mit der elektronischen Verschaltung**

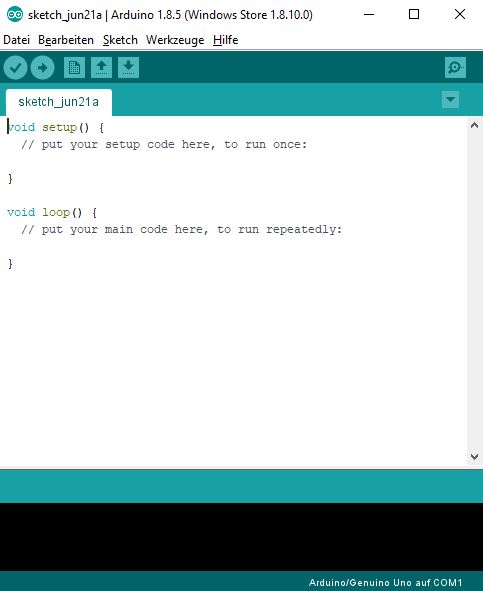
Die elektronische Verschaltung ist das Mittel der Verbindung zwischen dem elektonischen Teil(Arduino Uno) und dem mechanischen Teil, stellt dem Nutzer aber auch zwei Varianten zur Verfügung. Die erste ist manuel und die zweite ist automatisch, die von Arduino gesteuert wird.

**3.7 Trigger/ Switch**

Man kann dadurch von dem Kugelfall manuell steuern.

Bei unserem Experiment, wenn man einmal Den drückt, dann fällt nur eine Kugel und wenn man zweimal drückt, dann fallen 5 Kugeln nach ein ander jedoch entsprechend der Geschwindigkeit der Drehscheibe und der Position des Loches.

* 1. **Arduino IDE**

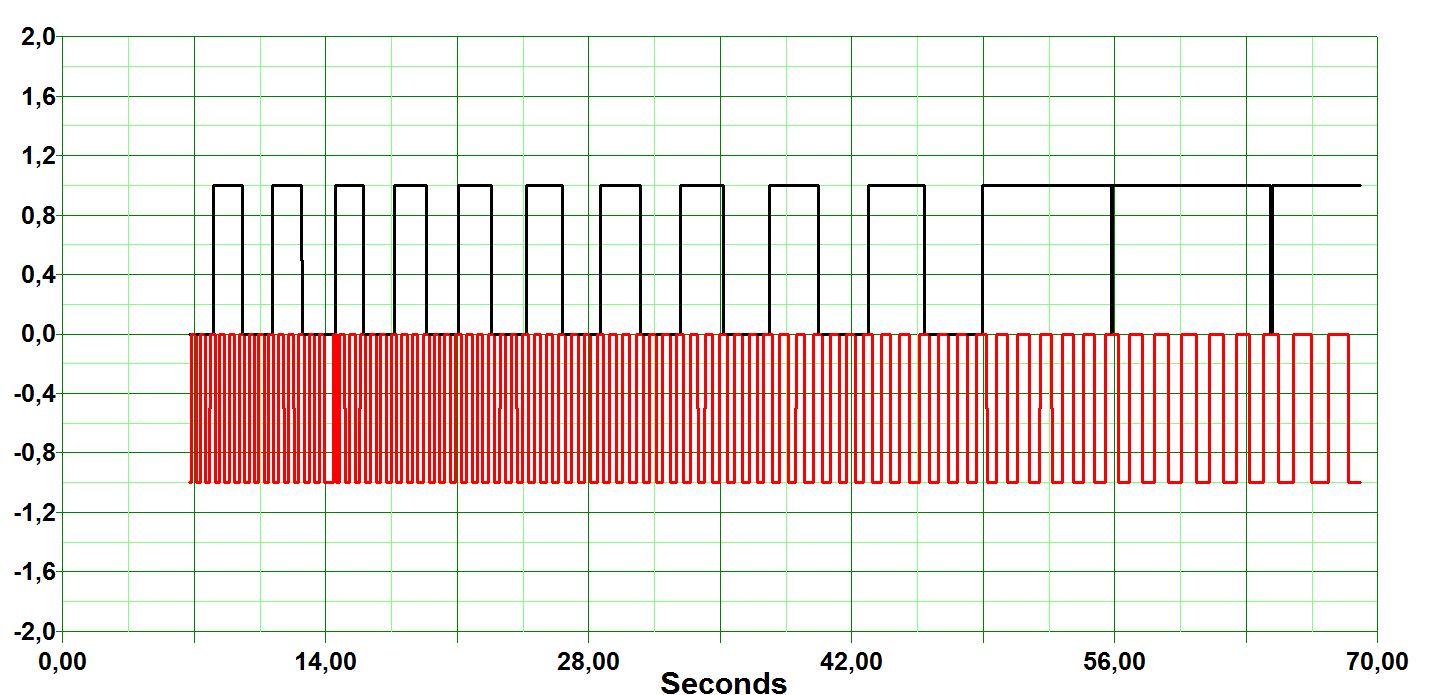
Es ist eine integrierte  Entwicklungsumgebung, die den Programmierern und Programmiererinnen ein sehr einfaches Interface hat und steht kostenlos im Internet zur Verfügung.

Die Programmiersprache ist fast wie C++.

Die Sensorwerte(Ausgaben) haben wir schon erfasst, um nachzuvollziehen, welche Ausgaben, die die Sensoren geben können und dann ein Alguritmos der Steuerung des Kugelfalls auszulegen.

**Abbildung (5)**

In der vorliegenden Abbildung beziehen sich die roten Impulse auf die Werte, die wir von Photosensor bekommen haben, und die schwarzen Impulse beziehen sich auf Hall-Sensorwerte.



**Abbildung (6)**

**Voraussetzungen:**

Die Drehscheibe ist auf 12 weißen und schwarzen Segmenten nacheinander geteilt.

Unter dem Loch gibt es ein Magnet und gegenüber liegt der Andere.

**4 Interpretation der Arbeitsweise der Sensoren:**

Bei Photosensor(**Photodiode**) haben wir 2 Zustände, und zwar:

**Zustand 0:** Der Photosensor befindet sich auf einem weißen Segment und gibt die Ausgabe 1.

**Zustand 1:** Der Photosensor befindet sich auf schwarzen Segment und gibt die serielle Ausgabe 0.

Beim Hall Sensor wird der Wert sich ändern, wenn das Magnet unter der Scheibe den Hallsensor vorbeigehen. Dann gibt es eine kurze ansteigende und abfallende Flanke --> '1'.

Wenn der Magnet am Loch in der Näher des Hallsensors wird der Wert Rising. In anderem Fall wird das Falling.

Vom vorliegenden Bild kann man das feststellen, dass es sechs veränderliche Werte der Photoelektronischen Verschlatung zwischen zwei veränderliche Werte des Hallsensors.

Und wenn die Geschwindigkeit der Drehscheibe zu klein ist (im Bild ist ungefähr 0.19 rps), ist der Wert vom Hallsensor sinnlos.

**5 Algorithmus Design**

**5.1.1 Mathematische Grundlage**

Variablen: Fall-Zeit (konstant)

Nach der freie Fall Formel:

Nach den 3 Messungen von der Hohe und den Durchschnitt auszurechnen, erhalten wir die Hohe und die Fall-Zeit

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| 78 cm | 78.3 cm | 77.8 cm |

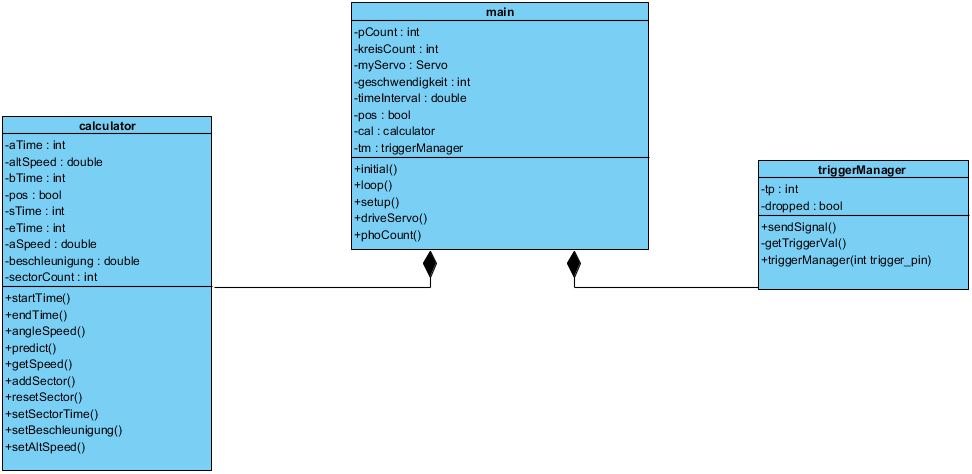
Der Durchschnitt der Höhe ist 78.03cm und die Fall-Zeit ist danach 0,399 sekunde.

Die Winkelgeschwindigkeitsformel:



**5.1.2 Klasse Design**

Das Class-digramm von Visual-paradigm



**Abbildung (7)**

**5.1.3 Beschreibung vom Algorithmus**

## C:\Users\Administrator\Desktop\Speedcal.pngWinkelgeschwindigkeitsberechnung

## C:\Users\Administrator\Desktop\PredictTime.pngWartezeitberechnung

## C:\Users\Administrator\Desktop\triggerLogik.pngKugelfalllogik

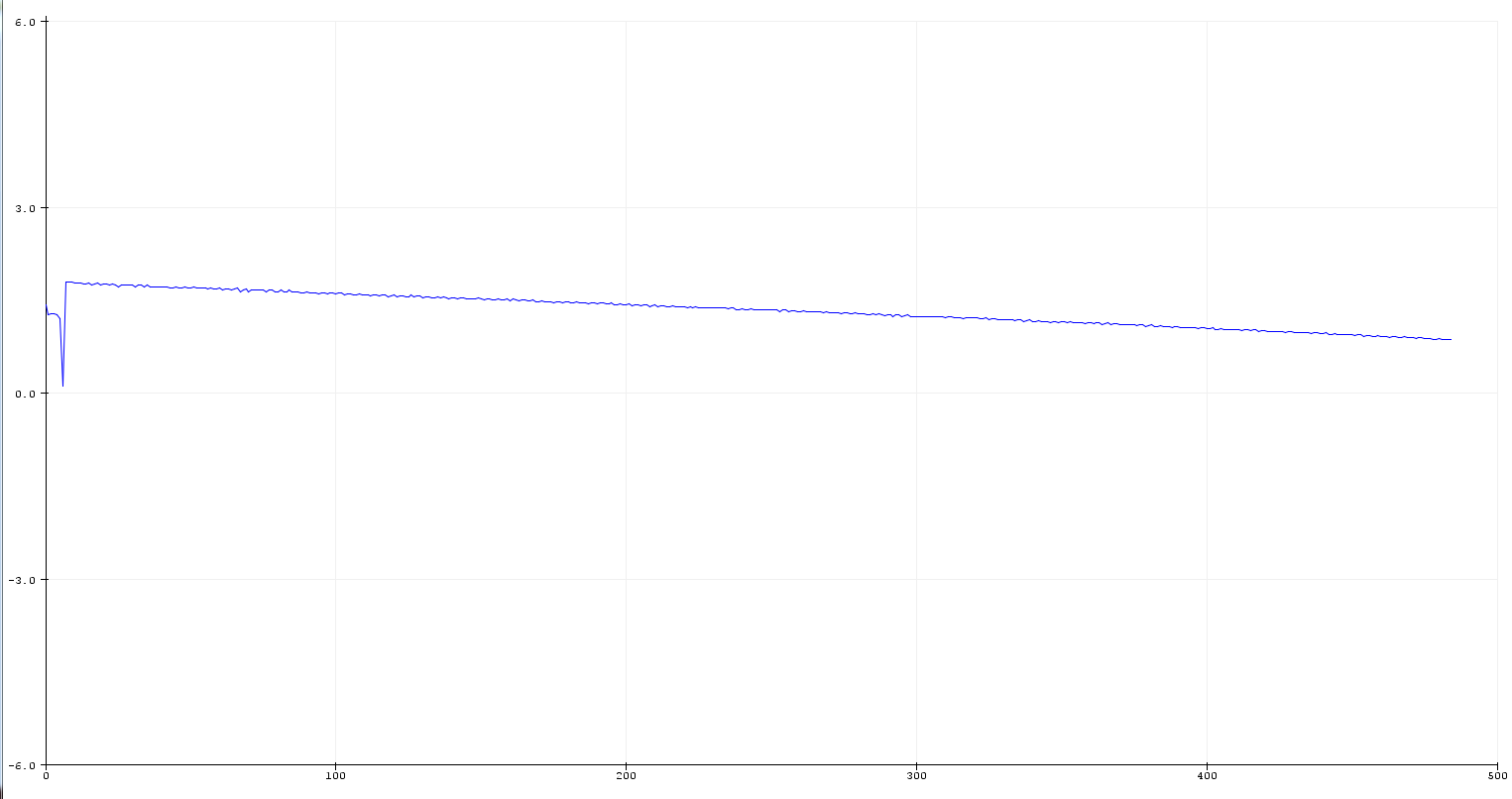
1. **Fehleranalyse**

Verspätung vom Servo Motor:

Die Drehgeschwindigkeit des Motors ist langsamer als das Programm. Dann gibt es eine Verspätung t, die ein Konstant ist. Der Wert jetzt ist 143 ms (beim Fenster), 116(bei der Tür) Es gibt auch Verspätung vom Trigger aber sie hat keinen Einluss auf die Genauigkeit.

**Die Mathematische Fehler:**

Wir benutzen ein lineares Model, um die Änderung der Geschwindigkeit zu beschreiben. Aber tatsächlich ist die Änderung nicht linear.



Abbildung

**Messungsfehler:**

Nach der Untersuchung finden wir ,wenn die Winkelgeschwindigkeit jede 6 Sektoren ein mal ausgerechnet wird und die Beschleunigung jede 2 Runde ausgerechnet wird, ist die Messung fast linear.

Einmal hineinzoomen, um das Bild eindeutiger zu werden



Abbildung

**Die Programmlaufzeit des Systems:**

Die Laufzeit vom Programm beeinflusst auch die vorhersagende Zeit. Nach der Messung finden wir diese Wert ist kleiner als 1ms und kann man das vernachlässigen.

**Mechanische Mängel:**

1. Der Pfad, in dem die Kugel rein fallen soll, ist nicht fest, so dass manschmal die Kugel fällt und schlägt auf den Grenzen des Loches.
2. Die Reibungen von der Gräten sind offensichtlich voneinander unterschiedlich, die Änderung der Geschwindigkeit bei der Tür unstabiler ist.
3. Das Rohr, in dem sich die Kugeln befinden, wird zufällig verschoben wegen der Unaufmerksamkeit oder Unerfahrenheit, dass wenn der Winkel des Rohres nur einen Grad verschoben wird, wird das System nicht ideal funktionieren. Das ist genau was wir bemerkt haben. Wir stellen jedes mal unsere Parameter, um mit dem aktuellen Systemzustand übereinzustimmen, und dann wenn wir nächstes mal kommen, funktioniert das System gar nicht gut wie letztes Mal.