7146CEM: Automotive Software Engineering - Design and Development

Coursework

Vignesh Babu Aravamuthan

[adnanv@coventry.ac.uk](mailto:adnanv@coventry.ac.uk)

Table of Contents

[1 Introduction 3](#_Toc88847287)

[2 Software Development Life Cycle 3](#_Toc88847288)

[2.1 Requirement Gathering 3](#_Toc88847289)

[2.1.1 PID controller 3](#_Toc88847290)

[2.1.2 Cruise Control 3](#_Toc88847291)

[2.1.3 Motor Speed 4](#_Toc88847292)

[2.1.4 Code 4](#_Toc88847293)

[2.2 Design 4](#_Toc88847294)

[2.2.1 PID Controller 4](#_Toc88847295)

[2.2.2 PID Tuning Algorithm 4](#_Toc88847296)

[2.2.3Cruise Control 4](#_Toc88847297)

[2.2.4 Motor Speed 5](#_Toc88847298)

[2.3 Coding 5](#_Toc88847299)

[2.4 Testing 5](#_Toc88847300)

[2.5 Validation 5](#_Toc88847301)

[3 Advantages of the used SDLC model 5](#_Toc88847302)

[4 GitHub Workflow 5](#_Toc88847303)

[5 References 5](#_Toc88847304)

[Figure 1 Cruise Control Schematic 4](#_Toc88845493)

[Table 1 Technical Requirements 4](#_Toc88847191)

[Table 2 Technical Requirements 4](#_Toc88847192)

[Table 3 Cruise Control Model Validation 5](#_Toc88847193)

[Table 4 Motor Speed Model Validation 6](#_Toc88847194)

Notes: make center alignment.

# 1 Introduction

This document contains information and details regarding the workflow used to create the PID controller, tuning of PID and generation of code for cruise control and motor speed control project. GitHub is used as version control system for the project. It is integrated with MATLAB to easily facilitate the GitHub process.

Control systems are needed when the system deals with continuously varying parameters or external disturbances such as load, friction, wind etc, which will affect the output of the system. In order to maintain the stable output of the system even in the presence of external disturbances a controller is needed to control those output variations. There are lot of controlling algorithms available for the control system, we will use only PID controller for cruise control and motor speed.

# 2 Software Development Life Cycle

This section gives an over all view of the software development process used to develop the cruise control and motor speed model, starting with requirements, design, development, testing and validation.

## 2.1 Requirement Gathering

This is the first stage of the V-development cycle contains the detailed understanding of requirements and expectations for the final product.

### 2.1.1 PID controller

#### 2.1.1.1 Technical Requirements

* To design the PID controller using the following equations

Where,

* PID Controller block should contain discrete function blocks.

#### 2.1.1.2 Non-Functional Requirement

* PID controller model should be designed and made it to referenced model.

### 2.1.2 Cruise Control

#### 2.1.2.1 Function Requirement

The system should have the following functional requirement

* Speed of the car should not fluctuate with respect to the external disturbances.

#### 2.1.2.2 Technical Requirement

The system should have the following technical requirements

|  |  |
| --- | --- |
| S.No | Requirements |
| 1 | Rise time < 10s |
| 2 | Overshoot < 10% |
| 3 | Stead state error <1% |

Table Technical Requirements

### 2.1.3 Motor Speed

#### 2.1.3.1 Technical Requirement

The system should have the following technical requirements

|  |  |
| --- | --- |
| S.No | Requirements |
| 1 | Rise time < 5s |
| 2 | Overshoot < 5% |
| 3 | Stead state error <1% |

Table Technical Requirements

### 2.1.4 Code

Developed code for the controller should have the following requirements:

* Code should be optimized for RAM efficiency.
* To develop the code in accordance with ISO26262.
* Code should follow MISRA C Guidelines.

## 2.2 Design

### 2.2.1 PID Controller

-Mention the process used to design the PID controller like creating the transfer function, converting transfer function to Z Transform

### 2.2.2 PID Tuning Algorithm

-mention the script which is used to Tune the PID for the models

-mentionthe script path to refer.

Script ‘PID\_Turning\_Script.mlx’ should be created to obtain Kp, Ki and Kd values to satisfy the requirements. Script should use trial and error method to find the values. Calculation for overshoot, rise time, steady state error for the output signal is to be implemented in the script to parallelly check whether the requirements are met with corresponding Kp, Ki and Kd values.

Desing uml diagrams- paste it here

### 2.2.3Cruise Control

-UML diagrams, block diagrams

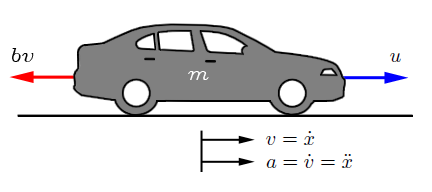


Figure Cruise Control Schematic

### 2.2.4 Motor Speed

-UML diagrams, Block diagrams

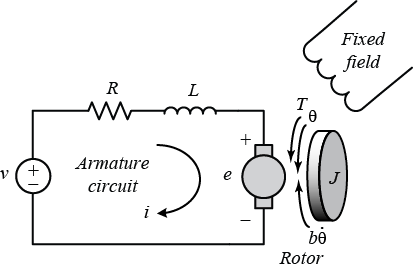


Figure Motor Speed Schematic

## 2.3 Development and Coding

Script for cruise control model is located inside ‘TASK 3/A\_CruiseControl/Scripts’ and for motor speed control it is located inside ‘TASK 3/B\_MotorSpeed/Scripts’.

### 2.3.1 Cruise Control

Block diagram of cruise control paste it here.

### 2.3.2 Motor Speed Control

Bock diagram of Motor speed conrol paste it here

-Autocode generation procedures, Documentation procedures used.

- code advisor report details

## 2.4 Testing

-Unit testing results, Polyspace statics analysis result. Cpp Check result

## 2.5 Validation

-Comparing requirements and output of Testing to validate.

-add risetime graphs

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| S.No | Requirements | Input | Expected Output | Actual Output | Pass/Fail |
| 1 | Rise time < 10s |  |  |  |  |
| 2 | Overshoot < 10% |  |  |  |  |
| 3 | Stead state error <1% |  |  |  |  |

Table Cruise Control Model Validation

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| S.No | Requirements | Input | Expected Output | Actual Output | Pass/Fail |
| 1 | Rise time < 5s |  |  |  |  |
| 2 | Overshoot < 5% |  |  |  |  |
| 3 | Stead state error <1% |  |  |  |  |

Table Motor Speed Model Validation

# 3 Advantages of the used SDLC model

* Mention the advantages of the SDLC model used with the development of the pid.

# 4 GitHub Workflow

* Mention the github links.
* Paste the flow chart of comits and branches as a picture

GitHub is integrated with Matlab, and all the versioning process is done within matlab itself.

# 5 References