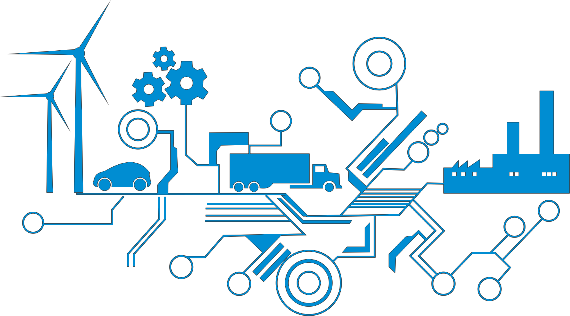
POWERTRAIN



Group

Title

Group Members:

|  |  |  |
| --- | --- | --- |
| Sl. No. | Name | Employee ID |
| 1 |  | 123456 |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |
| 5 |  |  |

© KPIT Technologies Ltd. All rights reserved. No part of this document may be reproduced, stored in a retrieval system or transmitted in any form, or by any means, electronic, mechanical, photocopying, recording or otherwise without the prior permission in writing of KPIT Technologies Ltd. Confidential & for internal circulation only.

**Revision History**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version No. | Date | Prepared By/  Modified By | Overview of Changes | Approved By |
| 0.1 | 24-10-16 | Team members names |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table of Contents

[1Introduction 6](#_Toc509570475)

[1.1 About Powertrain 6](#_Toc509570476)

[1.2 Requirement 6](#_Toc509570477)

[1.3 Plan 6](#_Toc509570478)

[1.4 Feature Comparison 6](#_Toc509570479)

[2 Feature Name of Armstrong/ Pico 7](#_Toc509570480)

[2.1 Detailed Requirements 7](#_Toc509570481)

[2.1.1 Comparison of Existing (Infineon, Renesas , Atmel etc ) 7](#_Toc509570482)

[2.1.2 Proposed Design 7](#_Toc509570483)

[2.2 Functional requirements 7](#_Toc509570484)

[2.2.1 Systems and Subsystems 7](#_Toc509570485)

[2.2.2 Inputs and outputs of the Systems 7](#_Toc509570486)

[2.3 Test Cases 7](#_Toc509570487)

[2.4 UML Diagrams 7](#_Toc509570488)

[2.5 Model Implementation 7](#_Toc509570489)

[2.6 Code Generation of the Model 7](#_Toc509570490)

[2.6.1 Autocode Generation 7](#_Toc509570491)

[2.6.2 AUTOSAR complaint code generation 7](#_Toc509570492)

[2.7 Model Implementation 8](#_Toc509570493)

[2.8 Test case Validation 8](#_Toc509570494)

[2.9 Results 8](#_Toc509570495)

[2.10 Sprayer Model 8](#_Toc509570496)

[2.10.1 Data Dictionary 8](#_Toc509570497)

[2.10.2 Code Generation 8](#_Toc509570498)

[2.10.3 Testing 8](#_Toc509570499)

[3 Feature 2 8](#_Toc509570500)

[4 Feature 3 8](#_Toc509570501)

[5 References 8](#_Toc509570502)

Table of Figures (Please auto insert)

[Figure 1‑1- Features and requirements analysis 6](#_Toc465958486)

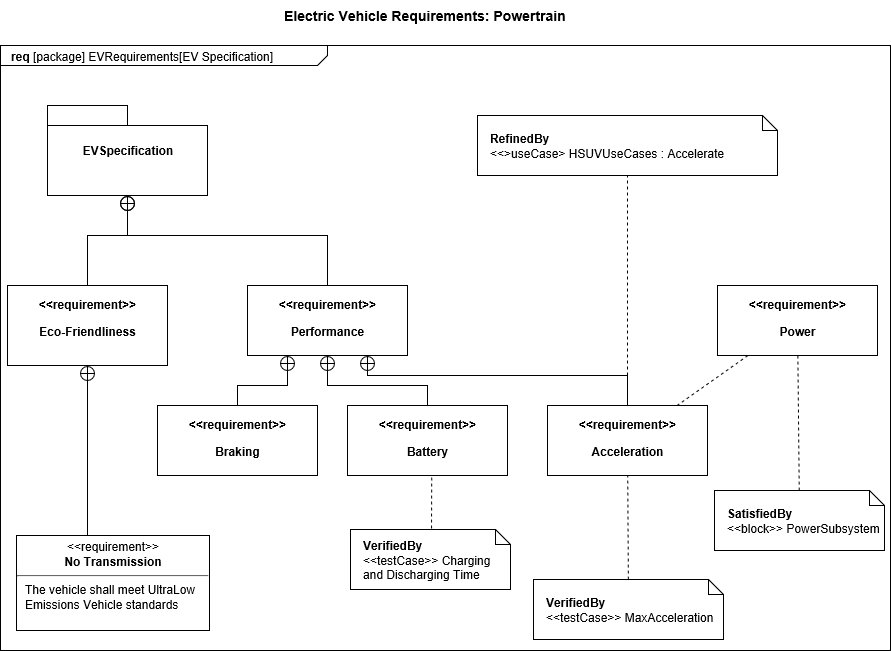
Table of Tables (Please auto insert)

**No table of figures entries found.**

# 1 Introduction

## Powertrain in Battery Electric Vehicle (BEV)

## Requirement



## Plan

The idea is to conceive a system that completely satisfies the all the requirements.

# Battery Management System

## Battery Recharging System

The system receives an input analog signal from the pedal, due the movement of the brake or pedal. The analog signal is then processed through an electronic control unit (ECU) that produces a pulse width modulation as output. The output generated by the ECU is then used as input by a generator to produce energy electric that will be use to charge the battery. At the same time, the battery is used to provide energy to the motor, which will be powering the system as long as the car is on. The flowchart of the system is shown below.

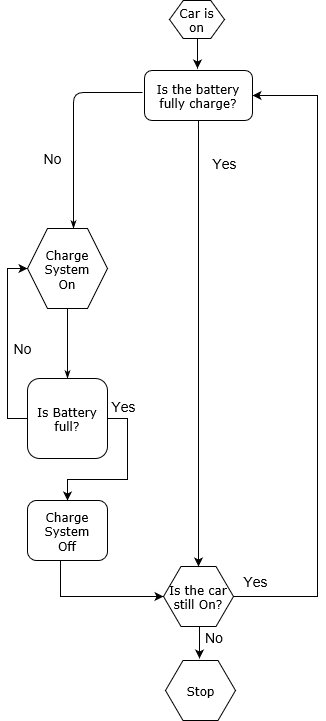


Figure 1- Flowchart

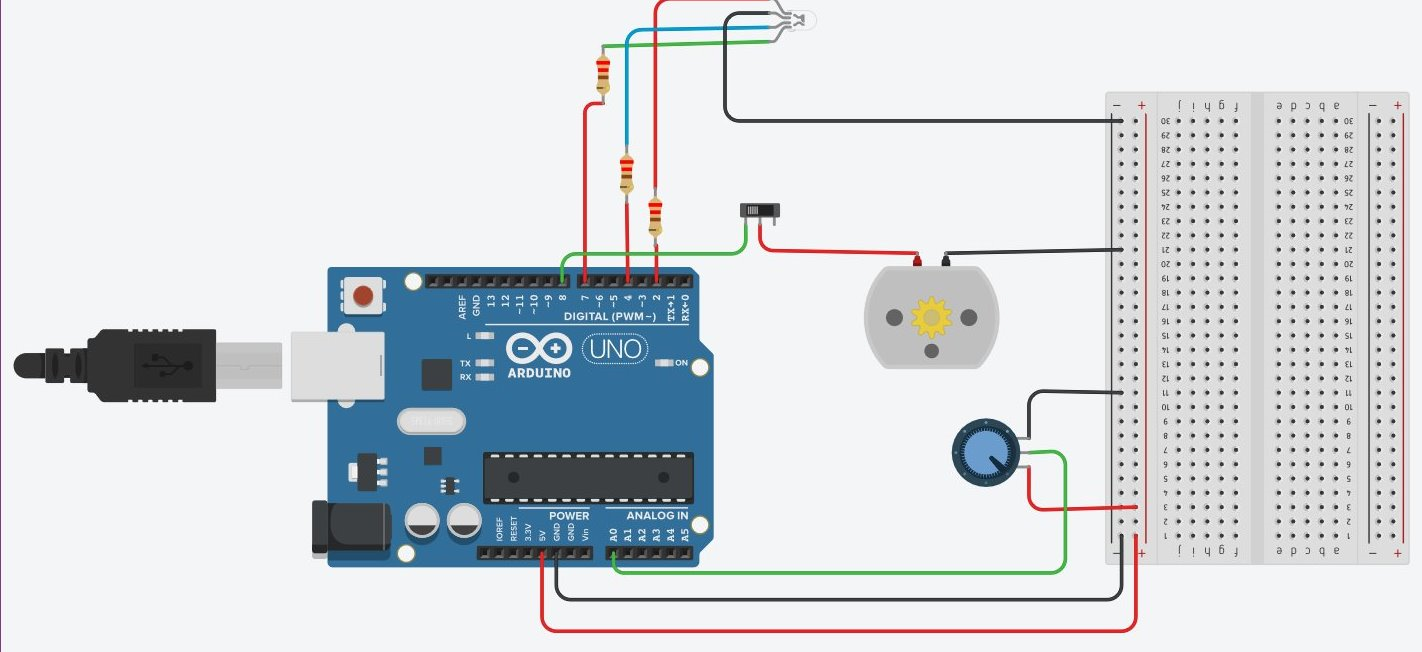
### Detailed Requirements

|  |  |
| --- | --- |
| Requirement ID | Requirement Description |
| BRS100 | An analog signal shall be generated when the driver pushes on the brake or accelerator. |
| BRS200 | The analog signal generated by the pedal shall be transformed using a generator into DC power. |
| BRS300 | The driver shall be able to see the level of energy recovered every time the pedal is pushed. |

### Functional requirements

|  |  |
| --- | --- |
| Requirement ID | Requirement Description |
| BRS001 | The system requires 5V DC power supply. |
| BRS101 | Once the loss of energy reaches a given threshold, the system is automatically awaked. |
| BRS102 | Whenever energy is recovered, the system will blink a Red, Green or Blue LED according to the amount of energy restored. |
| BRS201 | Once the battery is fully charged, the charging system goes to sleep. |
| BRS202 | Every 5 minutes, the system check the level of the battery. |

### Systems and Subsystems



|  |  |  |  |
| --- | --- | --- | --- |
| Name | Quantity |  | Component |
| U1 | 1 |  | Arduino Uno R3 |
| R1, R2,R3 | 3 |  | 220 ohm Resistor |
| RGB | 1 |  | RGB LED |
| P | 1 |  | 250 kOhm Potentiometer |
| S | 1 |  | Slide switch |
| M | 1 |  | D C Motor |

### Inputs and outputs of the Systems

Inputs: Analog Signal

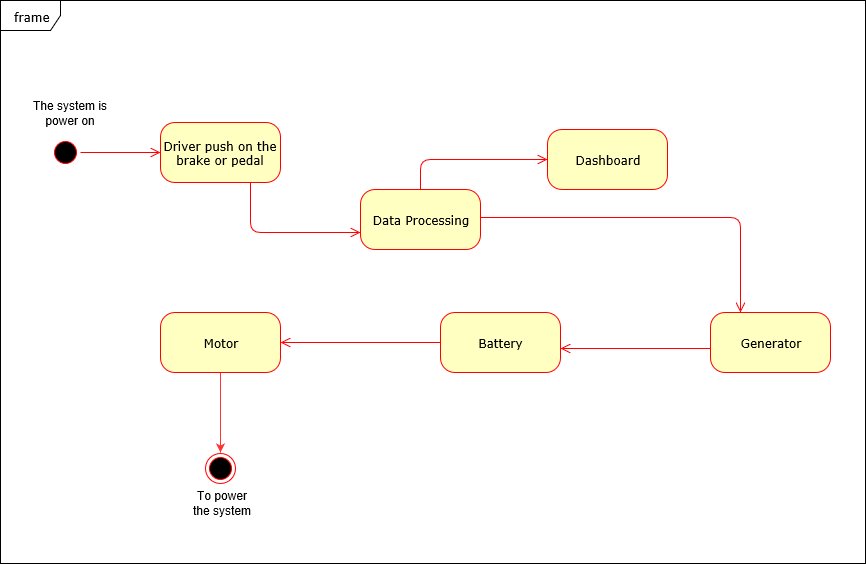
Outputs: DC power

### Test Cases

|  |  |  |
| --- | --- | --- |
| Test Case ID | Test Case Description | Expected Result |
| BRST001 | Battery is full, car is on, and the driver pushes on the brake or accelerator. | No LEDs blink |
| BRST002 | Car is on and battery isn’t fully charged but the threshold is not reached yet. | No LEDs blink |
| BRST003 | Car is on, battery not fully and threshold is reached | Charging System is On |
| BRST004 | After test BRST003, driver pushes on the brake or accelerator. | Green, Yellow or Red LED will blink |
| BRST005 | After test BRST004, the battery is fully charged and the car still on | Charging System Off |

### UML Diagrams

#### Functional Diagram



### Model Implementation

Describe ..

### Model Implementation

Describe ..

### Test case Validation

Describe ..

### Results

Describe ..

### Testing

Describe ..

Simulink Test/harness

## Battery Temperature Warning

### Detailed Requirements

In electric powertrain system, energy from the battery will convert to kinetic energy to power the vehicle or thermal energy that will otherwise be wasted to ambient environment. Also sudden temperature increase may indicate the undesired reaction happened in the battery.

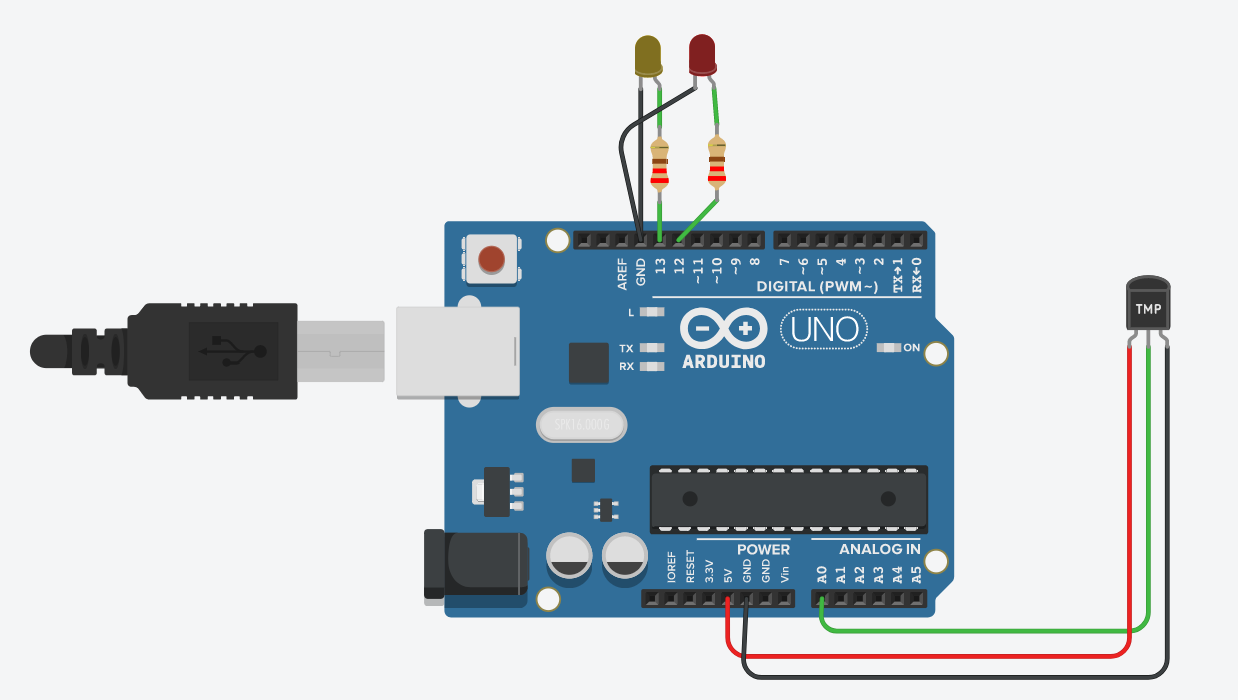
### Detailed Requirements

|  |  |
| --- | --- |
| Requirement ID | Requirement Description |
| BTW100 | Warning shall be given when battery temperature reaches critical. |
| BTW200 | Warning shall be given when sudden battery temperature increase occurs. |

### Functional requirements

|  |  |
| --- | --- |
| Requirement ID | Requirement Description |
| BTW001 | The system requires 5V DC power supply. |
| BTW002 | The system will measure battery temperature every 2 seconds. |
| BTW101 | Upon battery temperature reaches critical, red LED trigger counter will be reset to 10. |
| BTW102 | When red LED trigger counter is greater than 0, red LED will blink and the red LED trigger counter by 1, for every 2 seconds. |
| BTW201 | Upon sudden temperature increase, yellow LED trigger counter will be reset to 10. |
| BTW202 | When yellow LED trigger counter is greater than 0, yellow red LED will blink and the yellow LED trigger counter by 1, for every 2 seconds. |

### Systems and Subsystems



|  |  |  |  |
| --- | --- | --- | --- |
| Name | Quantity |  | Component |
| U1 | 1 |  | Arduino Uno R3 |
| R1, R2 | 2 |  | 220 ohm Resistor |
| D1 | 1 |  | Yellow LED |
| U2 | 1 |  | Temperature Sensor [TMP36] |
| D2 | 1 |  | Red LED |

### Inputs and outputs of the Systems

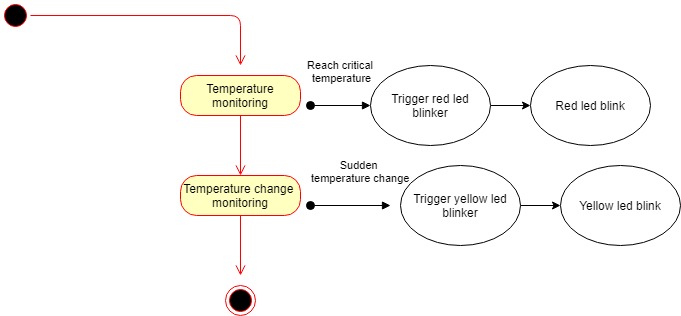
Inputs: Battery temperature

Outputs: LED warnings

## Test Cases

|  |  |  |
| --- | --- | --- |
| Test Case ID | Test Case Description | Expected Result |
| BTWT001 | Set to room temperature(25C) for 2 seconds | No LEDs blink |
| BTWT002 | After test BTWT001, increase temperature by 10C by 2 seconds (to 35C) | No LEDs blink |
| BTWT003 | After test BTWT002, increase temperature by 25C by 2 seconds (to 60C) | Yellow LEDs blinks 10 times after |
| BTWT004 | After test BTWT003, increase temperature by 40C by 2 seconds (to 100C) | Red and Yellow LEDs blinks 10 times after |
| BTWT005 | Set the temperature by 0 and Increase temperature by 25 C | Yellow LEDs blinks 10 times after |

## UML Diagrams



## Model Implementation

Describe ..

## Code Generation of the Model

Describe ..

### Autocode Generation

Describe ..

### AUTOSAR complaint code generation

Describe ..

## Model Implementation

Describe ..

## Test case Validation

Describe ..

## Results

Describe ..

## Sprayer Model

Describe ..

### Data Dictionary

Describe ..

### Code Generation

Describe ..

#### Autocode Generation

Describe ..

#### AUTOSAR Code Generation

Describe ..

### Testing

Describe ..

Simulink Test/harness

# Feature 3

Same template as above .. etc

Each feature will be updated by the corresponding feature owner

More .. .based on users

# References

IEEE Format of Referencing