

---

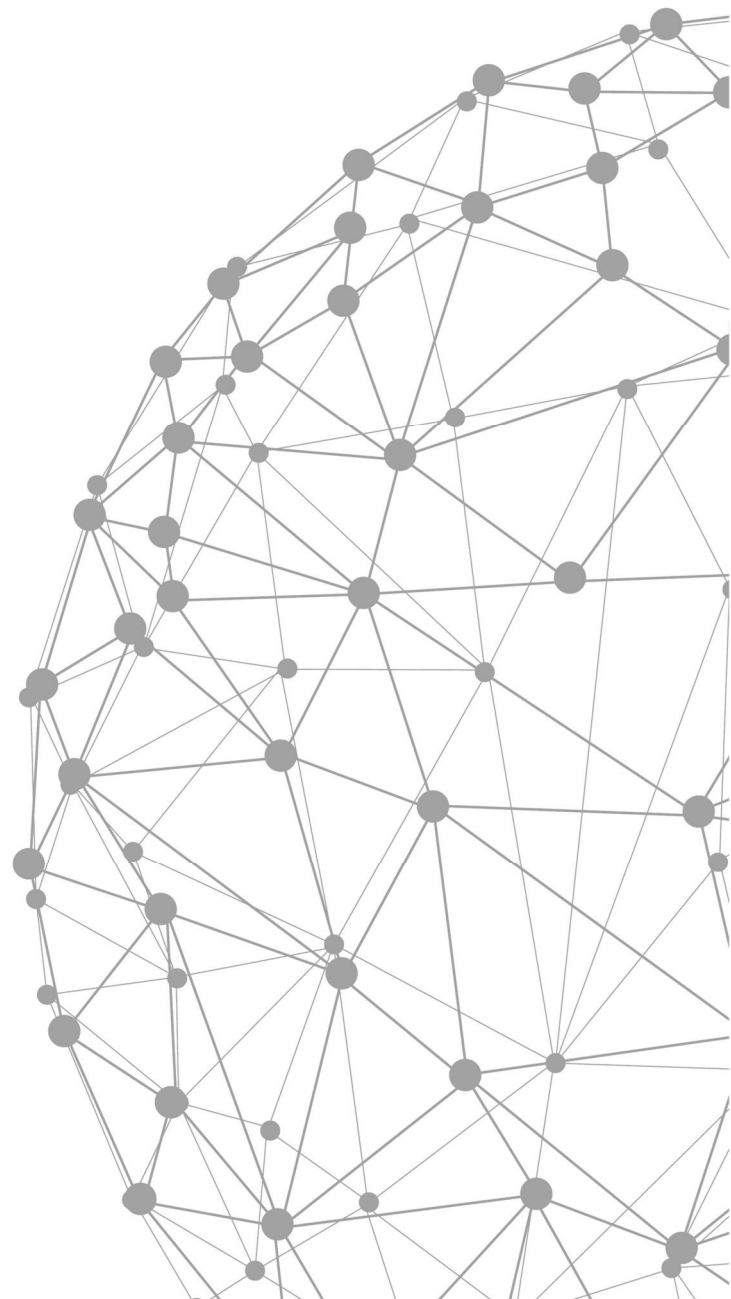
## **Project task 1**

### **Pololu Zumo 32U4**

Doc.-Number: Pflichtenheft\_Draft4\_2.docx  
Doc.-Version: A5  
299792458

Customer:  
NewTec  
Development  
Buchenweg 3  
89284 Pfaffenhofen a. d. Roth  
Germany

Author:  
Team: ~~ o=o\  
Hs Offenburg



Change History

Doc.-Version	Description of Modification	Date
A1	Initial revision	
A2	First version of use case diagrams, use case descriptions and requirements written	20.03.2024
A3	Use case diagrams, use case descriptions and requirements modified	27.03.2024
A4	Use case diagrams, use case descriptions and requirements modified	30.03.2024
A5	Use case diagrams, use case descriptions and requirements modified	03.04.2024

Release

	Name	Responsibility	Date	Signature
Creation				i.A.
Verification				i.A.
Approval				i.A.
Release				i.A.

## Table of Contents

<b>Change History .....</b>	<b>2</b>
<b>Release .....</b>	<b>2</b>
<b>Table of Contents .....</b>	<b>3</b>
<b>List of Tables.....</b>	<b>4</b>
<b>List of Figures .....</b>	<b>4</b>
<b>1 General.....</b>	<b>5</b>
1.1 Abbreviations.....	5
1.2 Terminology.....	6
1.2.1 System scenarios.....	6
1.2.2 Interface scenarios.....	6
1.2.3 System states.....	7
1.2.4 Interface states.....	7
1.2.5 Hardware.....	8
1.2.6 Other .....	9
1.3 Referenced Documents .....	10
1.4 Applicable Standards .....	10
<b>2 Introduction .....</b>	<b>11</b>
2.1 System Overview .....	11
2.2 Interface Overview .....	12
2.3 Scenarios .....	13
2.3.1 System .....	13
2.3.2 Interface .....	18
<b>3 Requirements .....</b>	<b>24</b>
3.1 Functional Requirements .....	24
3.1.1 Prelap.....	24
3.1.2 DuringLap.....	24
3.1.3 Postlap .....	24
3.1.4 Other .....	25
3.2 Non-Functional Requirements .....	25

List of Tables

Table 1: Abbreviations	5
Table 2: Terminology of the system scenarios	6
Table 3: Terminology of the interface scenarios	6
Table 4: Terminology of the system states	7
Table 5: Terminology of the interface states	7
Table 6: Terminology of the hardware	8
Table 7: Other terminology	9
Table 8: Referenced Documents	10
Table 9: Applicable Standards	10

List of Figures

Figure 1 System Overview.....	11
Figure 2 Interface Overview.....	12

# 1 General

## 1.1 Abbreviations

Abbreviation	Description
OLED	Organic light emitting diode
MCU	Microcontroller unit

Table 1: Abbreviations

## 1.2 Terminology

### 1.2.1 System scenarios

Term	Description
CalibrateLineSensors	The <i>LineSensors</i> are calibrated to the current light conditions
DisplayTeamName	The <i>TeamName</i> is shown on the <i>OledDisplay</i>
DriveLap	The <i>Robot</i> shows the <i>CountDown</i> on the <i>OledDisplay</i> , drives to the <i>StartFinishLine</i> and then drives one <i>FullLap</i>
SetParameters	Allows the <i>User</i> to choose between different <i>ParameterSets</i>
HandleError	Displays an error message on the <i>OledDisplay</i> until the <i>User</i> resets it

Table 2: Terminology of the system scenarios

### 1.2.2 Interface scenarios

Term	Description
<i>InitializeMcu</i>	Required system resources and variables are initialized. The <i>TeamName</i> is shown on the <i>OledDisplay</i>
<i>CalibrateLineSensors</i>	The <i>LineSensors</i> are calibrated to the current light conditions
<i>GetReadyForLap</i>	The MCU starts with a <i>CountDown</i> from 3 to 0 and the Robot starts moving.
<i>DriveLap</i>	The <i>Robot</i> follows the <i>GuideLine</i>
<i>DriveOverGap</i>	Allows the robot to drive over an interruption in the <i>GuideLine</i>
<i>MeasureTime</i>	Measures the time of a lap. This use case runs parallel to other use cases.
<i>DisplayLapTime</i>	Displays the completed lap time
<i>HandleError</i>	Displays an error message on the <i>OledDisplay</i>
<i>SetParameters</i>	Allows to choose between different <i>ParameterSets</i>

Table 3: Terminology of the interface scenarios

### 1.2.3 System states

Term	Description
LineSensorCalibrationDone	The calibration of the <i>LineSensors</i> is finished
Ready	The <i>Robot</i> is ready to retrieve a command from the <i>User</i>
Running	The <i>Robot</i> is driving on the <i>PlayField</i>
Setup	The <i>Robot</i> allows to choose between different sets of parameters which affect the <i>Robot's</i> behavior, e.g. a more cautious driving style
Error	An error sets the <i>Robot</i> into an error <i>State</i> . This <i>State</i> must be reset by the <i>User</i>

Table 4: Terminology of the system states

### 1.2.4 Interface states

Term	Description
InitializationDone	The initialization of the system is done
ReadyToDrive	The <i>Robot</i> is ready to drive
DriveToStart	The <i>Robot</i> move to the <i>StartFinishLine</i>
DriveToFinish	The <i>Robot</i> drives a lap on the <i>PlayField</i>
StateDriveOverGap	The <i>Robot</i> drives over a gap in the <i>GuideLine</i>
DisplayTime	The <i>Robot</i> displays the lap time on the <i>OledDisplay</i>
Setup	The <i>Robot</i> shows the parameters and allows them to be adjusted
Error	An error sets the <i>Robot</i> into an error <i>State</i> . This <i>State</i> must be reset by the <i>User</i>

Table 5: Terminology of the interface states

### 1.2.5 Hardware

Term	Description
Robot	Zumo32U4 as described in PololuDataSheet <sup>[2]</sup>
OledDisplay	The OLED display on the top of the Robot that allows visible feedback to the <i>User</i> as described in PololuDataSheet <sup>[2]</sup>
Buzzer	The buzzer on the <i>Robot</i> that allows audio feedback to the <i>User</i> as described in PololuDataSheet <sup>[2]</sup>
LineSensors	The sensor array at the front on the underside of the <i>Robot</i> that is able to recognize the lines of the <i>PlayField</i>
PowerSwitch	The switch at the back of the <i>Robot</i> . The <i>Robot</i> is only supplied with power when the switch is in the “ON” position as described in PololuDataSheet <sup>[2]</sup>
StartButton	The leftmost user button on the back of the <i>Robot</i> as described in PololuDataSheet <sup>[2]</sup>
ParamButton	The middle user button on the back of the <i>Robot</i> as described in PololuDataSheet <sup>[2]</sup>
CalibrateButton	The rightmost user button on the back of the <i>Robot</i> as described in PololuDataSheet <sup>[2]</sup>
ResetButton	The reset button on the back of the <i>Robot</i> with the label “reset” as described in PololuDataSheet <sup>[2]</sup>

Table 6: Terminology of the hardware



### 1.2.6 Other

Term	Description
AlarmSignal	The <i>AlarmSignal</i> is a signal of frequency 440Hz which is played for 333ms and after a pause of 333ms played again for 333ms. The volume is a minimum of 60dB at 10cm distance.
CountDown	A countdown that starts with 3 and counts down to 0. The number is always decremented after 1s and is shown on the <i>OledDisplay</i> in addition to the <i>TeamName</i>
DriveMotors	The two metal gearmotors that are connected to the wheels on the backside of the <i>Robot</i>
FullLap	One lap on the <i>PlayField</i> that starts with the <i>StartFinishLine</i> and ends with the same <i>StartFinishLine</i>
FullStop	The <i>Robot</i> is not moving because the <i>DriveMotors</i> do not receive any power
LapTime	The time the Robot needed to finish a <i>FullLap</i>
GuideLine	The line on the <i>PlayField</i> that marks the lap and shows the Robot where it must drive
ParameterSets	The <i>Robot</i> allows to choose between different sets of parameters which affect the <i>Robot's</i> behavior, e.g. a more cautious driving style
PlayField	The flat surface the <i>Robot</i> drives on
PoweredOn	The <i>Robot</i> is supplied with power for at least 500ms so that all the software of the Mcu had enough time to finish all initialization steps
Redetect	Searching and recognizing the <i>GuideLine</i> after losing <i>GuideLine</i> while driving
ShortBeep	The ShortBeep is a signal of frequency 440Hz with a duration of 1000ms and a minimum volume of 60dB at 10cm distance
StartFinishLine	The line perpendicular to the <i>GuideLine</i> that marks the beginning and end of the lap
State	The state of the Robot. Only one state can be active at the same time
TeamName	The name that identifies the team that owns the Robot. The name is defined by the software of the Mcu and cannot be changed
User	The person that interacts with the <i>Robot</i>

Table 7: Other terminology

### 1.3 Referenced Documents

Reference	Document-Identification	Description
[1]	SpecificationSheet	The document „11001_0099_0088_RD-Product-Specification.pdf“ which describes the specifications of the project
[2]	PololuDataSheet	The datasheet of the <i>Robot</i> with the name „zumo_32u4_oled_robot.pdf“

Table 8: Referenced Documents

### 1.4 Applicable Standards

Reference	Document-Identification	Description
[1]	N/A	N/A

Table 9: Applicable Standards

## 2 Introduction

### 2.1 System Overview

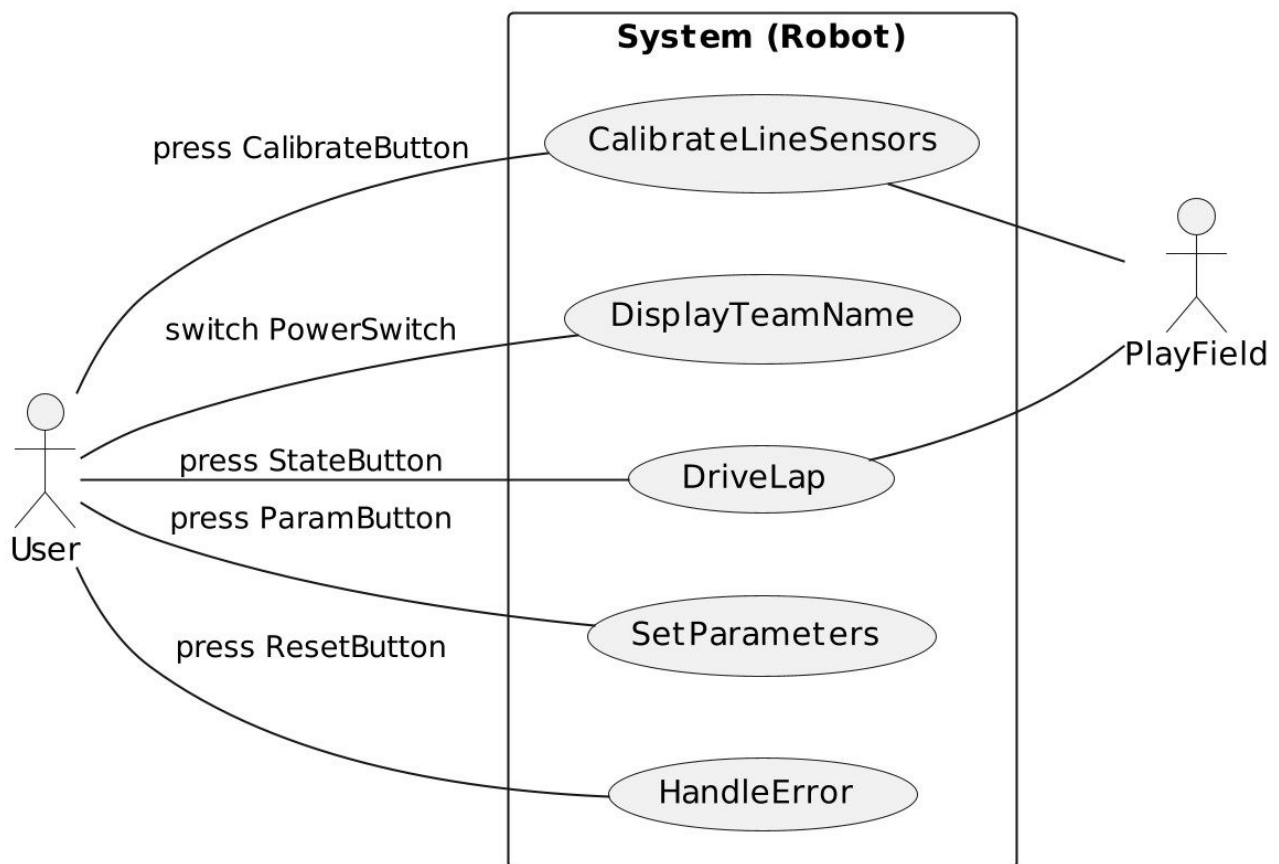


Figure 1 System Overview

## 2.2 Interface Overview

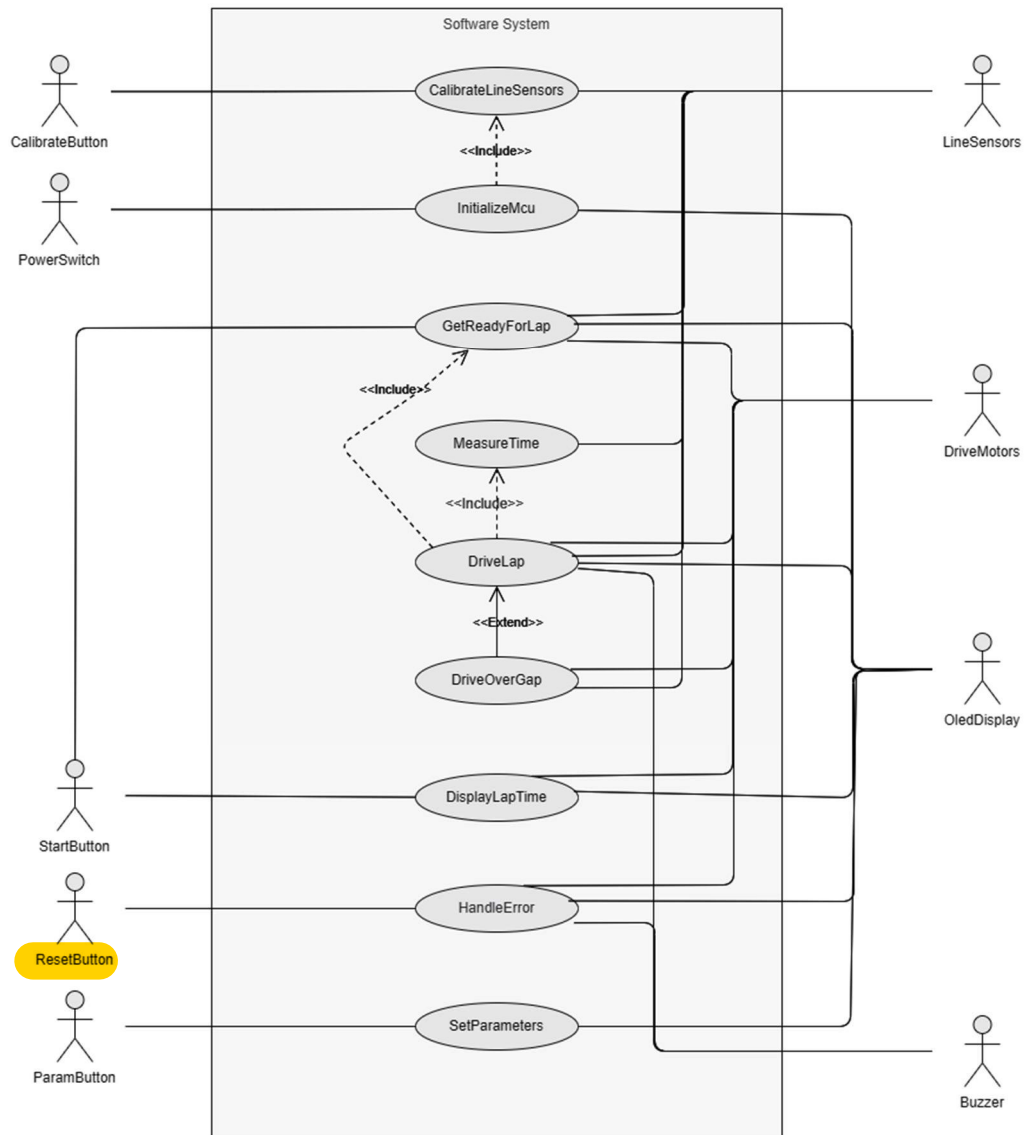



Figure 2 Interface Overview (Visual Paradigma)

## 2.3 Scenarios

### 2.3.1 System

Reference number	2.3.1.1
Name	<i>CalibrateLineSensors</i>
Short description	The <i>LineSensors</i> are calibrated to the current light conditions
Precondition	<p>There are two different sets of preconditions in this use case. All the conditions of one set must be met. The condition sets are mutually exclusive and therefore cannot be both met at the same time.</p> <p>Precondition set 1 (Robot is powered on)</p> <ul style="list-style-type: none"> <li>The <i>Robot</i> is placed on the <i>GuideLine</i> of the <i>PlayField</i></li> <li>The <i>PowerSwitch</i> is in the position "OFF"</li> </ul> <p>Precondition set 2 (Calibration is triggered by the user)</p> <ul style="list-style-type: none"> <li>The <i>Robot</i> is placed on the <i>GuideLine</i> of the <i>PlayField</i></li> <li>The <i>Robot</i> is <i>PoweredOn</i></li> <li>The <i>Robot</i> is in <i>FullStop</i></li> <li>The <i>State Ready</i> is active</li> </ul>
Postcondition	The <i>State LineSensorCalibrationDone</i> is active
Error case	none
Actors	<i>User</i> , <i>PlayField</i>
Trigger	<p>There are two different triggers.</p> <p>The first trigger is only effective if precondition set 1 met:</p> <ul style="list-style-type: none"> <li>The <i>PowerSwitch</i> is switched from "OFF" to "ON"</li> </ul> <p>The second trigger is only effective if precondition set 2 met:</p> <ul style="list-style-type: none"> <li>The <i>User</i> releases the <i>CalibrateButton</i></li> </ul>
Standard sequence	<ol style="list-style-type: none"> <li>The <i>Robot</i> is placed on the <i>GuideLine</i> of the <i>PlayField</i></li> <li>a. The <i>PowerSwitch</i> is switched from "OFF" to "ON"</li> <li>The <i>LineSensors</i> are calibrated to the current light conditions</li> <li>The <i>State LineSensorCalibrationDone</i> is activated</li> </ol>
Alternative sequences	<ol style="list-style-type: none"> <li>2b.1. The <i>State Ready</i> is active</li> <li>The <i>User</i> releases the <i>CalibrateButton</i></li> <li>Go to standard sequence 3.</li> </ol>

Reference number	2.3.1.2
Name	<i>DisplayTeamName</i>
Short description	The <i>TeamName</i> is shown on the <i>OledDisplay</i>
Precondition	<p>There are two different sets of preconditions in this use case. All the conditions of one set must be met. The condition sets are mutually exclusive and therefore cannot be both met at the same time.</p> <p>Precondition set 1:</p> <ul style="list-style-type: none"> <li>• The <i>Robot</i> is <i>PoweredOn</i></li> <li>• The <i>Robot</i> is in <i>FullStop</i></li> <li>• The <i>State LineSensorCalibrationDone</i> is active</li> </ul> <p>Precondition set 2:</p> <ul style="list-style-type: none"> <li>• The <i>Robot</i> is <i>PoweredOn</i></li> <li>• The <i>Robot</i> is in <i>FullStop</i></li> <li>• The <i>State Error</i> is active</li> </ul>
Postcondition	<p>All these conditions must be met:</p> <ul style="list-style-type: none"> <li>• The <i>Robot</i> is <i>PoweredOn</i></li> <li>• The <i>Robot</i> is in <i>FullStop</i></li> <li>• The <i>State Ready</i> is active</li> </ul>
Error case	none
Actors	<i>User</i>
Trigger	<p>There are two different triggers.</p> <p>The first trigger is only effective if precondition set 1 met:</p> <ul style="list-style-type: none"> <li>• The <i>State LineSensorCalibrationDone</i> changes from not active to active</li> </ul> <p>The second trigger is only effective if precondition set 2 met:</p> <ul style="list-style-type: none"> <li>• The use case <i>HandleError</i> is finished</li> </ul>
Standard sequence	<ol style="list-style-type: none"> <li>1. The <i>State LineSensorCalibrationDone</i> changes from not active to active</li> <li>2. The <i>OledDisplay</i> is cleared</li> <li>3. The <i>TeamName</i> is shown on the <i>OledDisplay</i> and not cleared</li> <li>4. The <i>State Ready</i> is activated</li> </ol>
Alternative sequences	none

Reference number	2.3.1.3
Name	<i>DriveLap</i>
Short description	The <i>Robot</i> shows the <i>CountDown</i> on the <i>OledDisplay</i> , drives to the <i>StartFinishLine</i> , and then drives one <i>FullLap</i>
Precondition	All these conditions must be met: <ul style="list-style-type: none"> <li>• The <i>Robot</i> is <i>PoweredOn</i></li> <li>• The <i>Robot</i> is in <i>FullStop</i></li> <li>• The <i>State Ready</i> is active</li> </ul>
Postcondition	All these conditions must be met: <ul style="list-style-type: none"> <li>• The <i>Robot</i> is <i>PoweredOn</i></li> <li>• The <i>Robot</i> is in <i>FullStop</i></li> <li>• The <i>State Ready</i> is active</li> </ul>
Error case	There are two independent error cases: <ul style="list-style-type: none"> <li>• The first <i>StartFinishLine</i> is not recognized within 10s</li> <li>• The <i>FullLap</i> is not finished within 20s</li> </ul> 
Actors	<i>User</i> , <i>PlayField</i>
Trigger	The <i>User</i> releases the <i>StartButton</i>
Standard sequence	<ol style="list-style-type: none"> <li>1. The <i>User</i> releases the <i>StartButton</i></li> <li>2. The <i>State Running</i> is activated</li> <li>3. The <i>Robot</i> shows the <i>CountDown</i> on the <i>OledDisplay</i></li> <li>4. The <i>Robot</i> starts driving, following the line on the <i>PlayField</i></li> <li>5a. The <i>Robot</i> recognizes the <i>StartFinishLine</i></li> <li>6. The <i>ShortBeep</i> is played on the <i>Buzzer</i></li> <li>7. The <i>Robot</i> continues driving, following the line on the <i>PlayField</i></li> <li>8a. The <i>Robot</i> recognizes the <i>StartFinishLine</i></li> <li>9. The <i>ShortBeep</i> is played on the <i>Buzzer</i></li> <li>10. The <i>Robot</i> goes to a <i>FullStop</i></li> <li>11. The <i>LapTime</i> is shown on the <i>OledDisplay</i></li> <li>12. The <i>User</i> releases the <i>StartButton</i></li> <li>13. The <i>State Ready</i> is activated</li> </ol>
Alternative sequences	<ol style="list-style-type: none"> <li>5b.1. The <i>StartFinishLine</i> is not recognized within 10s</li> <li>2. The use case <i>HandleError</i> (2.3.1.5) is activated</li> </ol>
	<ol style="list-style-type: none"> <li>8b.1. The <i>Robot</i> does not recognize the <i>StartFinishLine</i> within 10s</li> <li>2. The use case <i>HandleError</i> (2.3.1.5) is activated</li> </ol>

Reference number	2.3.1.4
Name	<i>SetParameters</i>
Short description	Allows the <i>User</i> to choose between different <i>ParameterSets</i>
Precondition	All these conditions must be met: <ul style="list-style-type: none"> <li>• The <i>Robot</i> is <i>PoweredOn</i></li> <li>• The <i>Robot</i> is in <i>FullStop</i></li> <li>• The <i>State Ready</i> is active</li> </ul>
Postcondition	All these conditions must be met: <ul style="list-style-type: none"> <li>• The <i>Robot</i> is <i>PoweredOn</i></li> <li>• The <i>Robot</i> is in <i>FullStop</i></li> <li>• The <i>State Ready</i> is active</li> </ul>
Error case	none
Actors	<i>User</i>
Trigger	The <i>ParamButton</i> is released
Standard sequence	<ol style="list-style-type: none"> <li>1. The <i>ParamButton</i> is released</li> <li>2. The <i>State Setup</i> is activated</li> <li>3. Cycle to the next <i>ParameterSet</i></li> <li>4. The <i>OledDisplay</i> shows the number of the active <i>ParameterSet</i> for 0.5s</li> <li>5. The <i>State Ready</i> is activated</li> </ol>
Alternative sequences	none



Reference number	2.3.1.5
Name	<i>HandleError</i>
Short description	Displays an error message on the <i>OledDisplay</i> until the <i>User</i> resets it
Precondition	All these conditions must be met: <ul style="list-style-type: none"> <li>• The <i>Robot</i> is <i>PoweredOn</i></li> <li>• An error occurred</li> </ul>
Postcondition	All these conditions must be met: <ul style="list-style-type: none"> <li>• The <i>Robot</i> is <i>PoweredOn</i></li> <li>• The <i>Robot</i> is in <i>FullStop</i></li> <li>• The use case <i>DisplayTeamName</i> (2.3.1.2) is activated</li> </ul>
Error case	none
Actors	<i>User</i>
Trigger	An error occurs
Standard sequence	<ol style="list-style-type: none"> <li>1. An error occurs</li> <li>2. The <i>State Error</i> is activated</li> <li>3. The <i>Robot</i> goes to a <i>FullStop</i></li> <li>4. The <i>AlarmSignal</i> is played on the <i>Buzzer</i></li> <li>5. An error message is displayed on the <i>OledDisplay</i></li> <li>6. The <i>User</i> places the <i>Robot</i> on the <i>GuideLine</i></li> <li>7. The <i>User</i> releases the <i>ResetButton</i></li> <li>8. The use case <i>CalibrateLineSensors</i> (2.3.1.1) is activated</li> </ol>
Alternative sequences	none

### 2.3.2 Interface

Reference number	2.3.2.1
Name	<i>InitializeMcu</i>
Short description	Required system resources and variables are initialized. The <i>TeamName</i> is shown on the <i>OledDisplay</i>
Precondition	All these conditions must be met: <ul style="list-style-type: none"> <li>The MCU must be offline and have no power</li> <li>The <i>PowerSwitch</i> is in the position "OFF"</li> </ul>
Postcondition	The <i>State InitializationDone</i> is active
Error case	none
Actors	<i>PowerSwitch</i> , <i>OledDisplay</i>
Trigger	The <i>PowerSwitch</i> is switched from "OFF" to "ON"
Standard sequence	<ol style="list-style-type: none"> <li>All system resources are initialized</li> <li>All system variables are initialized</li> <li>The <i>TeamName</i> is displayed on the <i>OledDisplay</i></li> <li>The <i>State InitializationDone</i> is activated</li> </ol>
Alternative sequences	none

Reference number	2.3.2.2
Name	<i>CalibrateLineSensors</i>
Short description	The <i>LineSensors</i> are calibrated to the current light conditions
Precondition	The <i>State Error</i> is not active
Postcondition	The <i>State ReadyToDrive</i> is active
Error case	none
Actors	<i>LineSensors</i> , <i>CalibrateButton</i>
Trigger	<p>There are two different triggers. Both triggers execute the same standard sequence.</p> <p>The first trigger is:</p> <ul style="list-style-type: none"> <li>The <i>State InitializationDone</i> is activated</li> </ul> <p>The second trigger is:</p> <ul style="list-style-type: none"> <li>The <i>CalibrateButton</i> is released</li> </ul>
Standard sequence	<ol style="list-style-type: none"> <li>The <i>LineSensors</i> are calibrated</li> <li>The <i>State ReadyToDrive</i> is activated</li> </ol>
Alternative sequences	none

Reference number	2.3.2.3
Name	<i>GetReadyForLap</i>
Short description	The MCU starts with a <i>CountDown</i> from 3 to 0 and the Robot starts moving.
Precondition	The <i>State ReadyToDrive</i> is active
Postcondition	The State <i>DriveToStart</i> is not active
Error case	The <i>DriveToStart</i> is active for more than 10s
Actors	<i>OledDisplay</i> , <i>DriveMotors</i> , <i>LineSensors</i>
Trigger	The <i>StartButton</i> is released
Standard sequence	<ol style="list-style-type: none"><li>1. The <i>CountDown</i> is displayed</li><li>2. As soon as 0 is displayed, the <i>DriveMotors</i> are supplied with power</li><li>3. The State <i>DriveToStart</i> is activated</li><li>4. The <i>Robot</i> follows the <i>GuideLine</i></li><li>5. The <i>StartFinishLine</i> is recognized</li><li>6. The <i>State DriveToFinish</i> is activated</li></ol>
Alternative sequences	none

Reference number	2.3.2.4
Name	<i>DriveLap</i>
Short description	The <i>Robot</i> follows the <i>GuideLine</i>
Precondition	The <i>State DriveToFinish</i> is active
Postcondition	The <i>State DriveToFinish</i> is not active
Error case	The <i>FullLap</i> is not finished within 20s
Actors	<i>OledDisplay</i> , <i>DriveMotors</i> , <i>LineSensors</i>
Trigger	There are two independent triggers: <ul style="list-style-type: none"> <li>The <i>State DriveToFinish</i> changes from not active to active</li> <li>The use case <i>DriveOverGap</i> (2.3.2.5) is finished</li> </ul>
Standard sequence	<ol style="list-style-type: none"> <li>1a. The <i>State DriveToFinish</i> changes from not active to active</li> <li>2. The <i>Buzzer</i> emits the <i>ShortBeep</i></li> <li>3a. The <i>Robot</i> follows the <i>GuideLine</i></li> <li>4a. The <i>StartFinishLine</i> is again</li> <li>5. The <i>Buzzer</i> emits the <i>ShortBeep</i></li> <li>6. The <i>State DisplayTime</i> is activated</li> </ol>
Alternative sequences	<ol style="list-style-type: none"> <li>1b.1. The use case <i>DriveOverGap</i> (2.3.2.5) is finished</li> <li>2. The <i>State DriveToFinish</i> is activated</li> <li>3. Go to standard sequence 3a.</li> </ol>
	<ol style="list-style-type: none"> <li>3b.1. The <i>Robot</i> encounters a gap</li> <li>2. The use case <i>DriveOverGap</i> (2.3.2.5) is entered</li> </ol>
	<ol style="list-style-type: none"> <li>4b.1. The <i>StartFinishLine</i> is not detected after 20s</li> <li>2. The use case <i>HandleError</i> (2.3.2.8) is entered</li> </ol>

Reference number	2.3.2.5
Name	<i>DriveOverGap</i>
Short description	Allows the <i>Robot</i> to drive over an interruption in the <i>GuideLine</i>
Precondition	The <i>State DriveToFinish</i> is active
Postcondition	The <i>State DriveToFinish</i> is active
Error case	<i>GuideLine</i> is not recognized within 5s
Actors	<i>DriveMotors, LineSensors</i>
Trigger	The <i>GuideLine</i> is lost
Standard sequence	<ol style="list-style-type: none"> <li>1. The <i>State StateDriveOverGap</i> is activated</li> <li>2. The <i>Robot</i> searches for the <i>GuideLine</i></li> <li>3a. The <i>GuideLine</i> is recognized</li> <li>4. The use case <i>DriveLap</i> (2.3.2.4) is activated</li> </ol>
Alternative sequences	<ol style="list-style-type: none"> <li>3b.1. The <i>GuideLine</i> is not recognized within 5s</li> <li>2. The use case <i>HandleError</i> (2.3.2.8) is entered</li> </ol>

Reference number	2.3.2.6
Name	<i>MeasureTime</i>
Short description	Measures the time of a <i>FullLap</i> . This use case runs parallel to other use cases.
Precondition	The <i>State DriveToStart</i> is active
Postcondition	The <i>State DriveToFinish</i> is not active
Error case	none
Actors	<i>LineSensors</i>
Trigger	The <i>StartFinishLine</i> is recognized
Standard sequence	<ol style="list-style-type: none"> <li>1. The timer is started</li> <li>2. The <i>StartFinishLine</i> is recognized again</li> <li>3. The timer is stopped</li> </ol>
Alternative sequences	none

Reference number	2.3.2.7
Name	<i>DisplayLapTime</i>
Short description	Displays the completed <i>LapTime</i>
Precondition	The <i>State DriveToFinish</i> is active
Postcondition	The <i>State ReadyToDrive</i> is active
Error case	none
Actors	<i>OledDisplay</i> , <i>DriveMotors</i>
Trigger	The <i>State DisplayTime</i> changes from not active to active
Standard sequence	<ol style="list-style-type: none"> <li>1. Stops the power supply to the <i>DriveMotors</i></li> <li>2. Shows the completed lap time on the <i>OledDisplay</i></li> <li>3. The <i>StartButton</i> is released</li> <li>4. The <i>State ReadyToDrive</i> is activated</li> </ol>
Alternative sequences	none

Reference number	2.3.2.8
Name	<i>HandleError</i>
Short description	Displays an error message on the <i>OledDisplay</i>
Precondition	An error occurred
Postcondition	The <i>State Error</i> is not active
Error case	none
Actors	<i>OledDisplay, Buzzer, DriveMotors, ResetButton</i>
Trigger	An error occurs
Standard sequence	<ol style="list-style-type: none"> <li>1. The <i>State Error</i> is activated</li> <li>2. Stops the power supply to the <i>DriveMotors</i></li> <li>3. The <i>AlarmSignal</i> is played on the <i>Buzzer</i></li> <li>4. An error message is displayed on the <i>OledDisplay</i></li> <li>5. The <i>ResetButton</i> is released</li> <li>6. The use case <i>InitializeMcu</i> (2.3.2.1) is activated</li> </ol>
Alternative sequences	none

Reference number	2.3.2.9
Name	<i>SetParameters</i>
Short description	Allows to choose between different <i>ParameterSets</i>
Precondition	The <i>State Error</i> is not active
Postcondition	The <i>State ReadyToDrive</i> is active
Error case	none
Actors	<i>OLED-Display, ParamButton</i>
Trigger	The <i>ParamButton</i> button has been released
Standard sequence	<ol style="list-style-type: none"> <li>1. The <i>ParamButton</i> is released</li> <li>2. The <i>State Setup</i> is activated</li> <li>3. Cycle to the next <i>ParameterSet</i></li> <li>4. The <i>OledDisplay</i> shows the number of the active <i>ParameterSet</i> for 0.5s</li> <li>5. The <i>State ReadyToDrive</i> is activated</li> </ol>
Alternative sequences	none



## 3 Requirements

### 3.1 Functional Requirements

#### 3.1.1 Before start

3.1.1.1 If the *Robot* is *PoweredOn* the *Robot* shall do ALL of the following steps

- display the *TeamName* on the *OledDisplay* for at least 2s
- calibrate the *LineSensors*

3.1.1.2 Upon releasing the *StartButton* the *Robot* shall *CountDown*

3.1.1.3 Upon releasing the *StartButton* the *Robot* shall detect the *StartFinishLine* in 10s or less

3.1.1.4 If the *Robot* does not detect the *StartFinishLine* in 10s or less the *Robot* shall go into the *Error State*

3.1.1.5 If the *Robot* detects the *StartFinishLine* AND is in state *DriveToStart* the *Robot* shall do ALL of the following steps

- emit a *ShortBeep* via *Buzzer*
- start the time measurement for the *FullLap*



#### 3.1.2 During run

3.1.2.1 If the *Robot* does not complete a *FullLap* in less than 20s the *Robot* shall go into the *Error State*

3.1.2.2 If the *Robot* leaves the *GuideLine*, the *Robot* shall *Redetect* the *GuideLine* in 5s or less

3.1.2.3 If the *Robot* does not *Redetect* the *GuideLine* in less than 5s the *Robot* shall go into *Error State*



#### 3.1.3 After run

3.1.3.1 If the *Robot* detects the *StartFinishLine* AND is in state *DriveToFinish* the *Robot* shall do ALL of the following steps

- Go to *FullStop*
- emit a *ShortBeep* via *Buzzer*
- end the time measurement for the *FullLap*
- display the measured time for the *FullLap* on the *OledDisplay*



### 3.1.4 Others

3.1.4.1 If the *Robot* goes into the *Error State* the *Robot* shall do the following steps in order

- Go to *FullStop*
- emit an *AlarmSignal* via *Buzzer*
- display the error reason on the *OledDisplay*

3.1.4.2 Upon releasing the *ResetButton* the *Robot* shall do ALL of the following

- clear the error message from the *OledDisplay*
- display the *TeamName*

3.1.4.3 Upon releasing the *CalibrateButton* the *Robot* shall do ALL of the following steps

- calibrate the *LineSensors*
- display the *TeamName* on the *OledDisplay* for at least 2s

3.1.4.4 During the competition the only changes to the software shall be the selection of a predefined list of *ParameterSets* via the *ParamButton*

3.1.4.5 Upon releasing the *ParamButton* the next entry in the list of *ParameterSets* is selected

3.1.4.6 Upon selection of the *ParameterSet* the *OledDisplay* shows its corresponding number for 0.5s

## 3.2 Non-Functional Requirements

3.2.1 The software shall be hardware independent

3.2.2 The programmer shall write the software in the programming language C

3.2.3 The *Robot* shall run during daytime- or officelight conditions

3.2.4 The *Robot* shall start the competition on the *GuideLine*

3.2.5 The *Robot* shall be powered on when it is positioned on the *GuideLine*

3.2.6 The software shall at most use 80% of the available flash memory

3.2.7 The dimensions of the *GuideLine* AND the *StartFinishLine* are defined in the SpecificationSheet<sup>[1]</sup>

3.2.8 The *Robot* shall be placed on the *GuideLine* of the *PlayField* with a minimum distance of 1cm in from the *StartFinishLine*