**Parking Lot System**

*Cypher*

*25.05.2022*

***Important Notes:***

* ***The descriptions in italics in this document (except for some section headings) are exemplary and explanatory and must be removed from the completed report.***
* ***Identify which section of this report was created by which team member***
* ***Your documentation should have ca. 8 pages (content! Without cover sheet, references, appendix etc.).***

# Team members

1. Sheikh Muhammad Adib Bin Sh Abu Bakar
2. Zafirul Izzat Bin Mohamad Zaidi
3. Muhammad Farid Izwan Bin Shabri
4. Muhammad Iqbal Bin Mohd Fauzi

# Introduction

*Parking lot is a common place visited by thousands of people every day. Looking for a parking space could take a lot of time. It is frustrating drive around looking for a parking lot and end up with no place for parking. This problem is our big interest in our project. We plan to create a system that could help people reduce time in looking for a place for parking by indicate a parking area either it is already full or not. Means that our system capable of counting free space by counting the number of entering and leaving car. The number of free spaces will be display at entrance. When there is no free space left, red light indicator, located near the display unit will be turned on and the entering gate will be closed.*

*To reduce the cost of building the solution, we use FPGA to prototype our solution since the solution will go through many iterations. Means that a lot of change will be made without spending a lot of money.*

*In the next section, we will go through how we design our solution followed by how we organize the project and eventually, the steps of the implementation process*

# Concept description

*To build our parking lot system, we reduce the complexity by modelling the structure and behavior of the parking lot in a few blocks diagram. Our main target that we could have a concreate system architecture so that we could split the system into a few components. Modeling diagram not only help us in building a system in more organized way, it also helps us as a team to work together parallelly.*

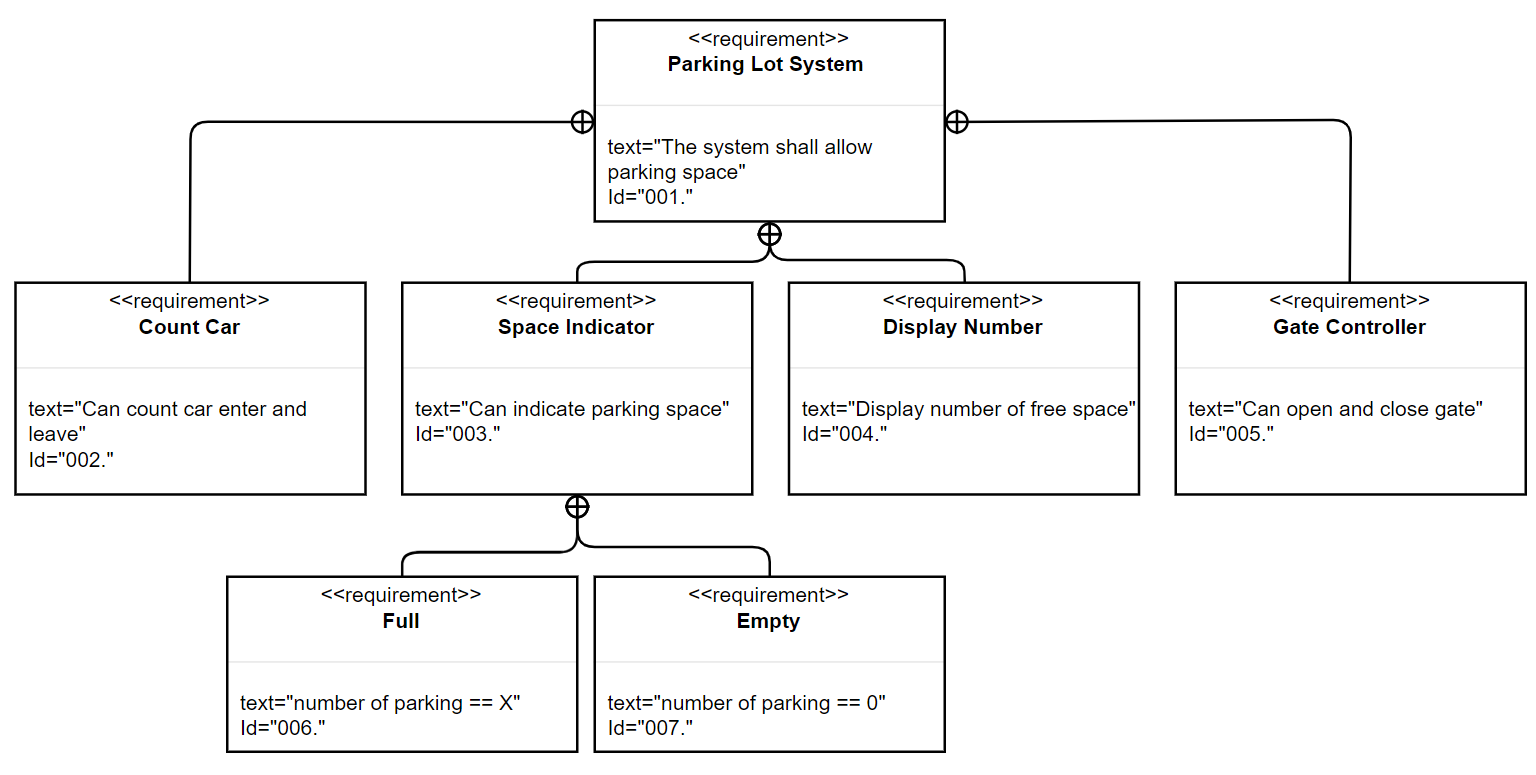
*To do that we first start with requirement diagram to put every requirement that we want to achieve either functional or non-functional related to the calculator that we want to build.*

*We then create a use case diagram to set the boundaries of the parking lot system. With use case diagram we can analyst the component that could exist in the system. After that we model each action in use case using activity diagram with possible scenario. Next, we analyze all the diagram to build a concrete parking lot system architecture, means that we are ready to do the state chart and state machine diagram so that later we can synthesis them using VHDL.*

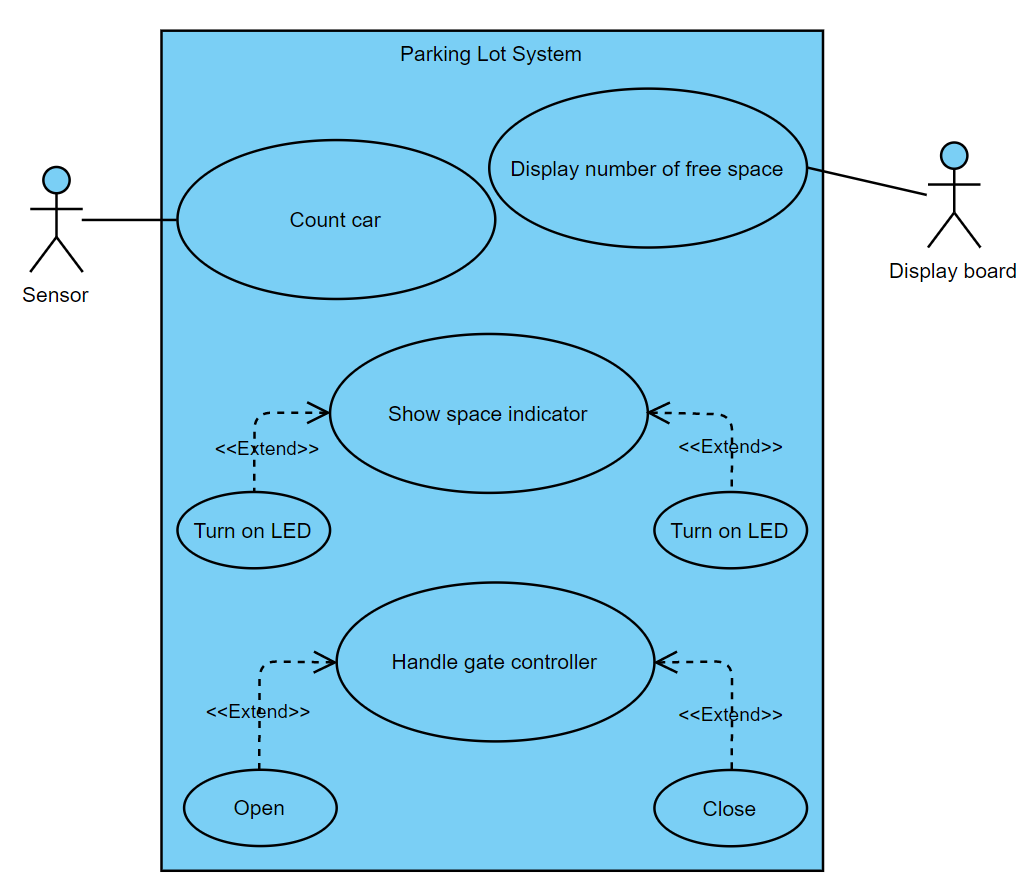
1. Requirement Diagram

The first phase of our project began with a pre-study where basic knowledge on Parking lot was gathered. The aim of the pre-study was to get a summary of our system project in general. The pre-study also enabled the search for content to be used as a guide in this research, such as details on calculator.

Now, we documented the requirements for this project with the functionalities. The important requirements of our system are operation that it can do. The requirement of our system can do are count car where it can count whenever car enter and leave. The system also may have space indicator to help driver know whether the space is empty or full. Display number of free space is also crucial and lastly the gate controller can either open or close.

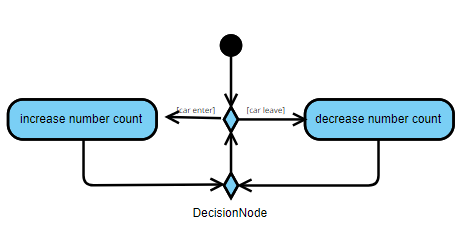


1. Use Case Diagram

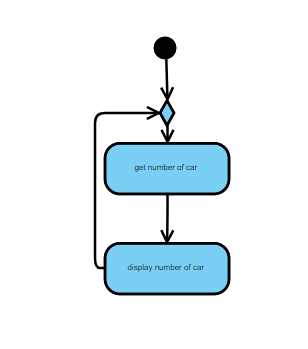


The use case diagram for a parking lot system is shown in the diagram below. There are two actors in this diagram: the sensor and the display board. The sensor in this system oversees counting cars as they enter and exit the parking lot. Furthermore, a display board will show the number of available parking spaces for the car. This system also includes an indicator that displays available space. For instance, if there is free space, green LED will turn on and red LED will be turn off; otherwise, if there is not free space, green LED will turn off and green LED will turn on. This system will also include a gate controller handle, which will allow the gate to be closed or opened.

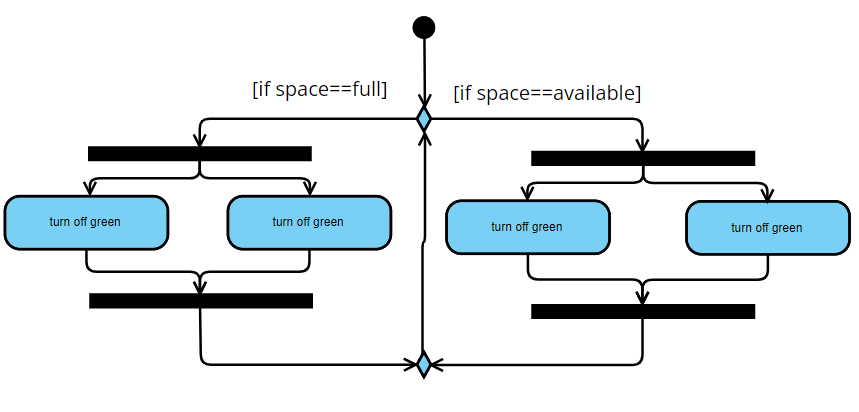
1. Activity Diagram



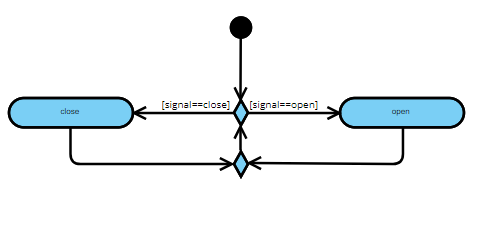
*Figure (a) above shows class diagram when the car is count. When the car is entering, the counter will add the number of counter. When the car is leaving, the counter number will decrease. Then it will continously count when the car is either leave or enter.*



*Figure (b)above explains the scenario when display the number of empty parking space. Class diagrams describes when the display will get amount of cars entering, then it will be displayed to display board.*

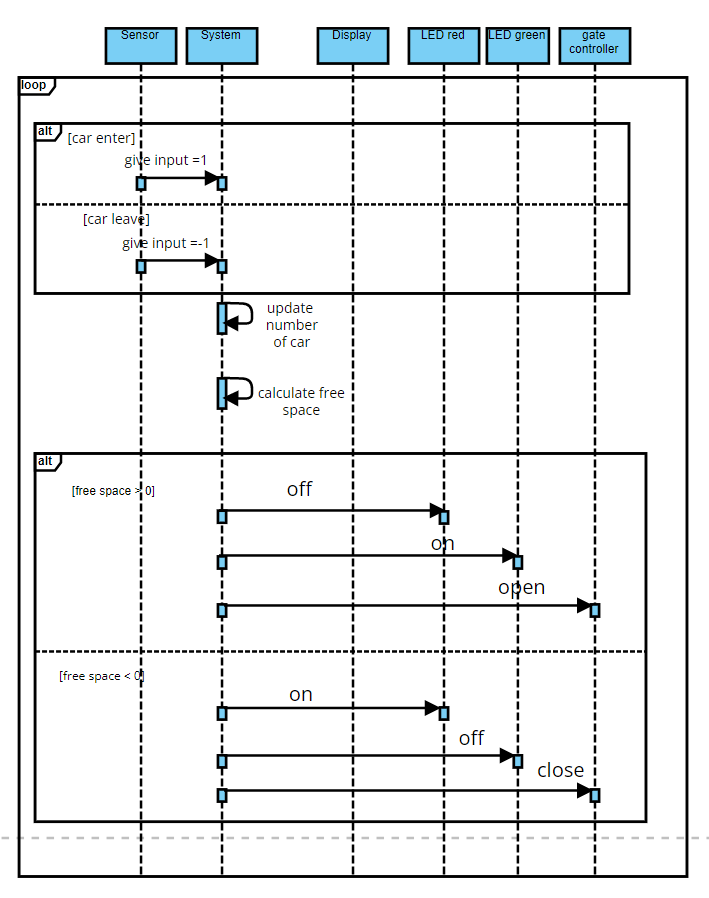


*Class diagram (c) explains space indicator works. When there is available parking space in a parking area, the lamp will turn green and red lamp will be switched off, so that people knows that they can find empty parking space. If the parking space is already fully occupied, then the red lamp will automatically switched on while the green lamp will turn off.*



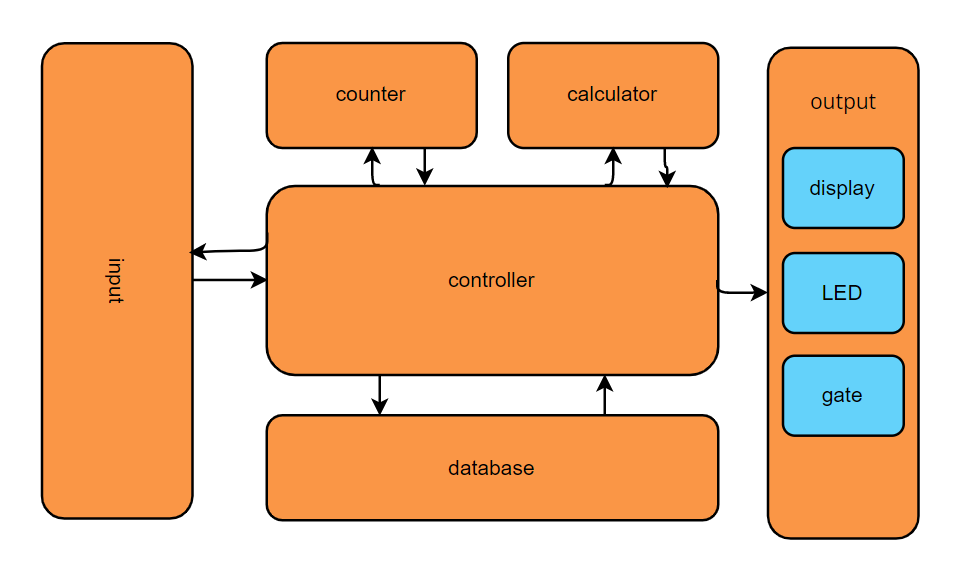
*Figure (d) explains how the gate contoller is being handled. When open signal is received, the gate will open . If close signal is received, the gate is closed. The system will keep reading if there is any changes to the signal.*

1. Sequen diagram



The flow of the system will be depicted in this sequence diagram. Sensor, system, display, led green, led red, and gate controller are the six required objects. Initially, the sensor will inform the system whether the car is entering or exiting the parking lot. If the car enters the parking lot, the system will receive the input value is 1. If the car drives away, the input value will be 0. The system will then update the number of cars and calculate free space. Following the completion of the process of updating and calculating the number of cars, the system will display the number of available spaces on the display board. If there is free space available, the green LED will turn on, the red LED will turn off, and the system will open the gate. If there is no free space, the green LED will turn off, the red LED will light up, and the gate will close.

1. System architecture



*Figure below explains the system architecture of parking lot system. There are 6 main system components which are input, controller, data base, counter, calculator and output. For output, we can see there are three different outputs which are display, LED and lastly gate.*

# Project/Team management

*Which project methods you used in your project?*

*Breakdown: How you managed your tasks?*

*What are the different tasks/roles of the team members in the project?*

*Describe which team member did which tasks.*

# Technologies

# *Describe the technological approaches you will use to implement your project.*

* *VHDL*
* *Eagle*
* *FPGA*
* *7 segment LED*

# VHDL Implementation

*Describe the implementation of your digital design in VHDL/FPGA*

*Provide a detailed block diagram for this purpose and briefly explain the used modules.*

*Describe how you verified your solution. Testbench!!*

*Provide the results for your FPGA Implementation (Results summary + Hardware results if necessary)*

# PCB Design

*Describe the implementation of your schematic and PCB design*

*Give a summary about your PCB design results (BOM, Costs, Size usw.)*

# Sources/References

*Provide the sources on the technologies and algorithms you used in your project (Github).*