**Project Proposal**

Group Members:

* Jyoti Dhiman - 2020csb1092
* Rohit Kinha - 2020csb1118
* Sanyam Walia - 2020csb1122

1. **Title:** Implementation of a Smart Car parking system with Collision prevention and fare calculator
2. **Overall objective:**

Through this project, we aim to implement a car parking system where input of sensors is fed as digital signals and based on them, parking gates open or close.

E.g., Suppose a parking has 4 slots, S1(empty), S2(filled), S3 (filled)and S4 (empty)Then, the information regarding how many of them are occupied can be stored in a register as s = {s1, s2, s3, s4} = 4’b0110

Also, we plan on Using DFF’s to synchronize all sensors. Use of counters to count the no. of clock cycles passed and hence calculate the fare.

Use of registers to store and access data.

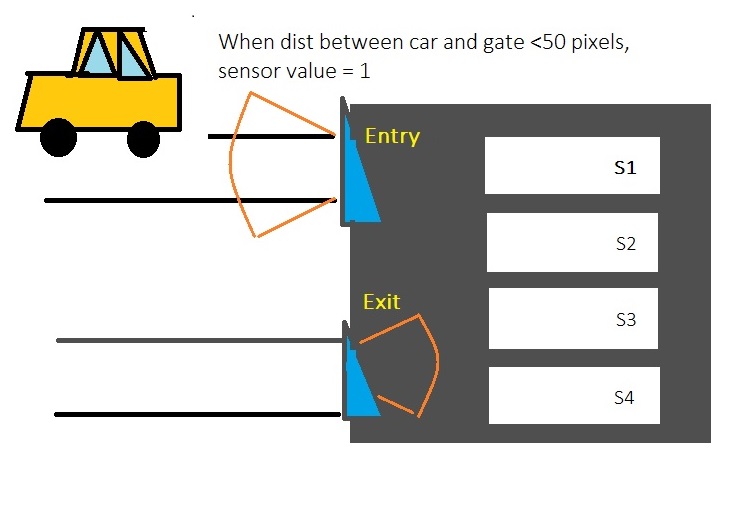
1. **Implementation:**

We plan on Implementing the above-mentioned system through an interactive webpage (tentative else basic Verilog implementation) where the user has the option of moving the car(image) back and forth.

And once the car is close to the gate, sensor value becomes 1 and the system starts working

1. **Functionality:**

Design-



We will basically look at following cases:

1. Entry sensor = 1 and parking slot available and exit sensor = 0

If this is the case, then we simply park the car in the available slot, change its value to 1 and then start the counter

E.g., 1011, we park car at s2, and string becomes 1111

1. Exit sensor 1

We stop the counter, calculate the fare and display it. And then we change the corresponding bit from 0 to 1.

1. Entry sensor 1 but all slots full

So, we display no parking

1. Entry sensor 1 and Exit sensor 1: As, exit sensor is 1, means car is exiting. So, to avoid congestion, we first let the exiting car leave and then we park the new incoming car

So, e.g., we had 1001 and car at s4 leaves and new car comes then, string first changes to 1000 and then to 1001. Counter also stops and starts again.

**Course Components that will be used:**

* Multiplexer with sensor bits as the control signals
* Registers to store and access data regarding the 4 parking slots
* Synchronous counter (Using DFFs) to count the time and thus calculate fare.

---------------------------------------END OF PROPOSAL-------------------------------------------