

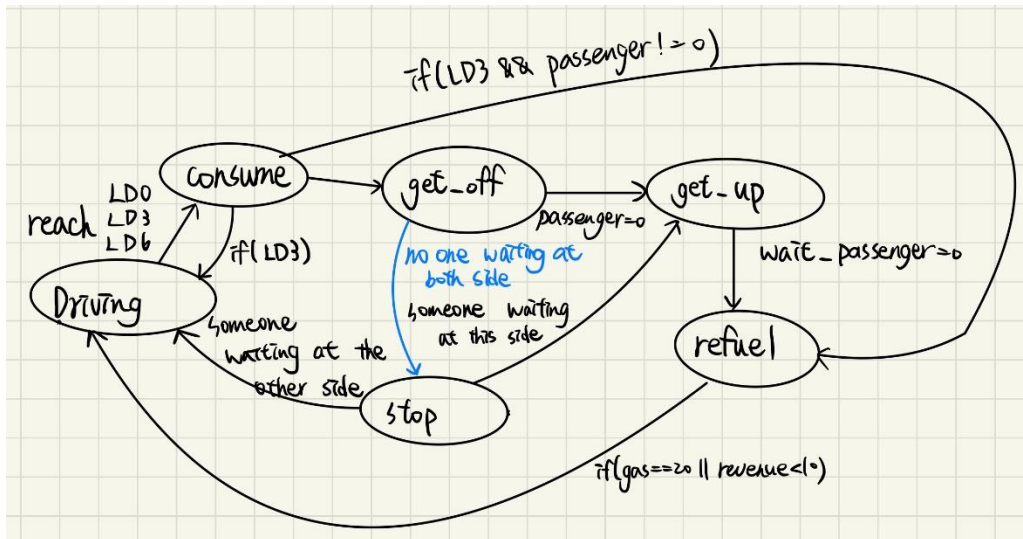
## Lab 6

學號: 109000205

姓名: 蕭皓隆

### 1. 實作過程

#### State\_diagram :



利用 Finite-State Machine 實作這次的 Lab6，設計了 6 個 state，分別是 DRIVING、GET\_OFF、GET\_UP、REFUEL、STOP 及 CONSUME，切換 state 的條件主要是利用 LED 的燈號以及乘客或 gas 等等來判斷。

**DRIVING**：若  $up\_down == 1$  (往上) 時將  $LED << 1$ ， $up\_down == 0$  (往下) 時將  $LED >> 1$ ，到達相對應位置的時候切換 state。

**GET OFF**：讓  $passenger\_next = passenger - 1$ ，當  $passenger == 0$  時切換到 GET\_UP。

**GET UP**：讓  $passenger =$  等車的人數，接著切換到 REFUEL。

**REFUEL**： $gas\_next = gas + 10$ ， $revenue\_next = revenue - 10$ ，直到  $gas == 20$  或  $revenue < 10$  時跳至 DRIVING。

**STOP**：為兩邊車站都沒人等車時會跳到的 state，若兩邊車站開始有人出現的話，判斷為現在這個車站或另一邊的車站，若為現在的就跳到 GET\_UP，為另一邊的就跳到 DRIVING。

**CONSUME**：到 LD0、LD3 以及 LD6 時要先扣 gas， $gas\_next = gas - (passenger * 5)$ ，扣完後跳到 REFUEL 或是 GET\_OFF

偵測鍵盤是否按下並增加等車人數的部分：

```
if(been_ready && key_down[last_change] == 1'b1) begin
    if(key_num == 4'b0001) begin
        wait0_passenger_next = (wait0_passenger==2)?wait0_passenger:wait0_passenger+1;
        wait6_passenger_next = wait6_passenger;
    end
    else if(key_num == 4'b0010) begin
        wait0_passenger_next = wait0_passenger;
        wait6_passenger_next = (wait6_passenger==2)?wait6_passenger:wait6_passenger+1;
    end
    else begin
        wait0_passenger_next = wait0_passenger;
        wait6_passenger_next = wait6_passenger;
    end
end
else begin
    wait0_passenger_next = wait0_passenger;
    wait6_passenger_next = wait6_passenger;
end
end
```

## 2. 學到的東西與遇到的困難

學到的東西：處理很多因為 clk 而 delay 的 bug，學到很多實用的技巧能彌補 delay 帶來的困擾。

遇到的困難：在處理鍵盤的時候要考慮 clk 的問題，若 clk 不對或是參雜其他不同 clk 的變數時容易有鍵盤沒反應抑或是其他部分出錯的問題。

## 3. 想對老師或助教說的話

