

# 数字逻辑Final Project

---

## 1 开发计划

---

### 1.1 选题：

停车场收费系统

### 1.2 成员分工

| 姓名  | 学号       | 分工        | 贡献    |
|-----|----------|-----------|-------|
| 刘乐奇 | 12011327 | 音乐、键盘、数码管 | 33.3% |
| 何忠荣 | 12011424 | 车的进出场     | 33.3% |
| 廖泽通 | 12011417 | 管理员模式     | 33.3% |

### 1.3 执行记录

第8周，Project选题发布；

12.4， 决定选题；

12.31， 最终测试；

12.31， 答辩。

## 2 设计

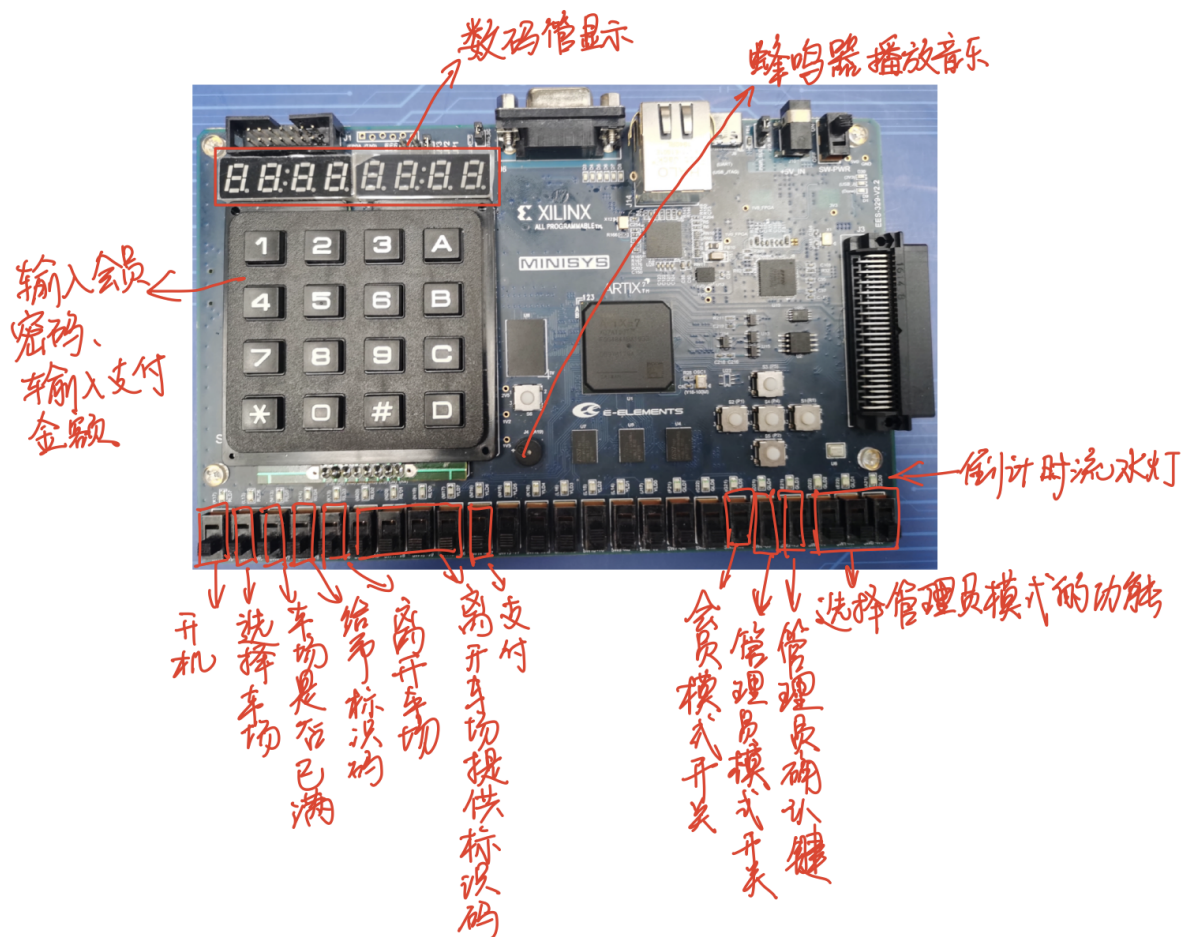
---

### 2.1 需求分析

#### 2.1.1 系统功能

本次Project需要在FPGA板上实现停车场收费系统。其中要有管理员管理车场、判断车场是否为满、车辆进场的标识码分配、车辆出场后的付款。

#### 2.1.2 端口与输入输出设备



## 2.2 系统结构设计

### 2.2.1 各模块接口及功能

- 音乐模块：输入接口clk时钟信号，输出接口beep用于利用蜂鸣器播放音乐

```
module music(
    input clk,
    output beep
);
```

- 数码管模块：输入接口clk时钟信号，rst复位信号，mode用于选择静态显示字符（mode高电平）和动态显示字符（mode低电平），text用于传入要显示的文本。输出接口seg\_en用于选择显示的位数，seg\_out用于显示字符。

```
//数码管显示
module show (
    input clk, rst,
    input mode,
    input [127:0] text,
    output [7:0] seg_en,
    output reg [7:0] seg_out
);
```

- 流水灯模块：输入接口clk时钟信号，st开始流水信号（也即开始计时）。输出接口led用于开发板上流水灯的显示，time\_end用于传出计时结束的信号。

```

module led (
    input clk,
    input st,
    output [23:0] led,
    output time_end
);

```

- 小键盘模块：输入接口clk时钟信号，enable启动小键盘。输出接口keyboard\_val用于传出按下的小键盘值。row和col用于扫描键盘。

```

module key16(
    input          clk,
    input          enable,
    input  [3:0] row,
    output reg [3:0] col,
    output reg [3:0] keyboard_val
);

```

- 停车场模块：

```

) module parking(
    input clk,
    input sw,
    input rst,
    //来车
    input come, place, stop_car,
    output reg full_switch,
    output reg [15:0] stop_place_num_light,
    //出车
    input leave,
    input [3:0] outplace_carnum,
    output reg count_down, empty_switch,
    output reg [6:0] money, stop_time, //传给支付模式
    output reg [127:0] car_leave_light,
    input [3:0] keyboard_val,
    //支付
    input pay,
    //管理员控制
    input empty, change,
    input [3:0] initial_money,
    input [2:0] per_money,
    input [3:0] place_number,
    output reg [119:0] information_light,
    output reg [55:0] pay_light,
    input time_end,
    output reg allpay
);

```

- 管理员模块：

```

module manager(
input clk,rst,sw,next,[3:0]num,[6:0]timeout,[2:0]state,[3:0]moneyget,
//next确认键(进入下一步) num接收小键盘数字 state接收状态(模式选择)
//moneyget接收来的钱 timeout接收出场车的停车时间
output [3:0]initial_money,[2:0]per_money,[3:0]num_of_park,[3:0]initial_vipmoney,
output reg [9:0]averagetime, reg empty, reg [63:0]show
//起始价, 时价, 车位数, 会员优惠, 平均时间, show为开发板显示屏展示内容
); //管理员模式,

```

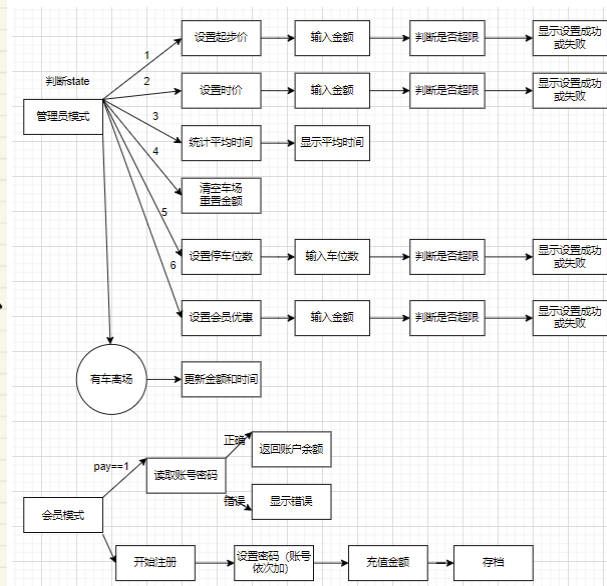
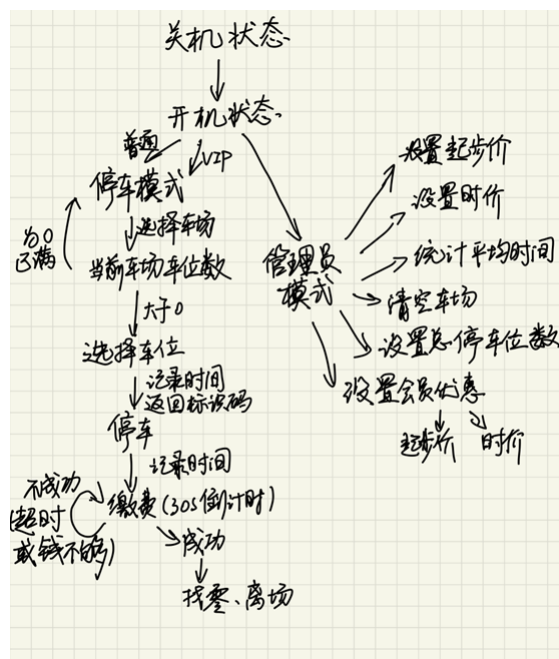
- 会员模块:

```

module vip_apply(
input clk,rst, sw, //开关
input pay, //开始支付同时要把开关sw打开
input num, //输入的数据
input next, //进入下一步
input idcheck, //查找的id
input passwordcheck, //查找的密码
output reg [63:0] show, //开发板上显示,
output reg [9:0] account //支付状态下返回余额
);

```

## 2.2.2 事务处理流程图



## 2.3 详细设计

- 车辆出入车场

首先, 在车辆入场前, 需要对整个车场进行初始化。initial\_mon 是停车起始价, place\_num 是每个车场的车位数 (分有A、B两个车场), per 是停车时价。{A0...A7} 与 {B0...B7} 表示两个车场的车位。

```

        if(sw&&rst&&come ==0&&leave ==0&&stop_car==0)begin
            {A0,A1,A2,A3,A4,A5,A6,A7} = 8'b00000000;
            {B0,B1,B2,B3,B4,B5,B6,B7} = 8'b00000000;
            full_switch = 0;
            initial_mon = 4'b1001;
            place_num = 4'b1000;
            per = 3'b101;
        end
    end

```

在车辆入场时，需要先判断选择的停车场是否有空车位，若无，则会在数码管上显示 FULL；若有，则会在数码管上显示八个 8。

```

| if(sw == 1&&come ==1&&leave ==0&&stop_car == 0) begin
|     full_switch = 0;
|     if(place == 1) begin
|         if(A0 == 0) begin
|             empty_place = 3'b000;
|         end
|         else if(A1 ==0) begin...
|             if(empty_place>=place_num) full_switch = 1;
|         end
|     else begin
|         if(B0 == 0) begin
|             empty_place = 000;
|         end
|         else if(B1 == 0) begin...
|             if(empty_place>=place_num) full_switch = 1;
|         end
|     end
| end

```

然后，会给予该车辆唯一的标识码，用于该车离场时付费。



```

if(sw == 1&&come ==1&&full_switch == 0&&leave ==0&&stop_car == 1) begin
    if(place)
        case(empty_place)
            3'b000:begin stop_place_num_light = 16'b10001000_11000000;A0 = 1:end
            3'b001:begin stop_place_num_light = 16'b10001000_11111001;A1 = 1:end
            3'b010:begin stop_place_num_light = 16'b10001000_10100100;A2 = 1:end
            3'b011:begin stop_place_num_light = 16'b10001000_10110000;A3 = 1:end
            3'b100:begin stop_place_num_light = 16'b10001000_10011001;A4 = 1:end
            3'b101:begin stop_place_num_light = 16'b10001000_10010010;A5 = 1:end
            3'b110:begin stop_place_num_light = 16'b10001000_10000010;A6 = 1:end
            3'b111:begin stop_place_num_light = 16'b10001000_11111000;A7 = 1:end
            default:stop_place_num_light = 16'b11111111_11111111;
        endcase
    else
        case(empty_place)
            3'b000:begin stop_place_num_light = 16'b10000011_11000000;B0 = 1:end
            3'b001:begin stop_place_num_light = 16'b10000011_11111001;B1 = 1:end
            3'b010:begin stop_place_num_light = 16'b10000011_10100100;B2 = 1:end
            3'b011:begin stop_place_num_light = 16'b10000011_10110000;B3 = 1:end
            3'b100:begin stop_place_num_light = 16'b10000011_10011001;B4 = 1:end
            3'b101:begin stop_place_num_light = 16'b10000011_10010010;B5 = 1:end
            3'b110:begin stop_place_num_light = 16'b10000011_10000010;B6 = 1:end
            3'b111:begin stop_place_num_light = 16'b10000011_11111000;B7 = 1:end
            default:stop_place_num_light = 16'b11111111_11111111;
        endcase
    end
end

```

最终，在车辆离场时，会根据其标识码，计算费用，并让车辆付款。

```

    else begin
        case(outplace_carnum)
            3'b000:if(B0 == 1) begin
                stop_time = carplaceB[0];
                if(carplaceB[0]>10) money = initial_mon+per*10;
                else money = initial_mon+stop_time*per;
            end
            else empty_switch = 1;
            3'b001:if(B1 == 1) begin...
            3'b010:if(B2 == 1) begin...
            3'b011:if(B3 == 1) begin...
            3'b100:if(B4 == 1) begin...
            3'b101:if(B5 == 1) begin...
            3'b110:if(B6 == 1) begin...
            3'b111:if(B7 == 1) begin...
        endcase
    end
    case (stop_time/10)...
    case (stop_time-10*(stop_time/10))...
    case (money/10)...
    case (money-(money/10)*10)...
    car_leave_light = {32'b10000111_11110000_11001000_10000110,8'b11111111,stop_time_light,8'b11111111,40'b11001000_10100011_10101011_10000110_10010001,8'b11111111,mon

```

## 3 板上测试

由于bug过多，来不及debug，故非常遗憾没能实现完整的板上测试。

约束文件：

```

set_property IOSTANDARD LVCMOS33 [get_ports {seg_en[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {seg_en[1]}]
set_property IOSTANDARD LVCMOS33 [get_ports {seg_en[2]}]
set_property IOSTANDARD LVCMOS33 [get_ports {seg_en[3]}]
set_property IOSTANDARD LVCMOS33 [get_ports {seg_en[4]}]
set_property IOSTANDARD LVCMOS33 [get_ports {seg_en[5]}]
set_property IOSTANDARD LVCMOS33 [get_ports {seg_en[6]}]
set_property IOSTANDARD LVCMOS33 [get_ports {seg_en[7]}]
set_property IOSTANDARD LVCMOS33 [get_ports {seg_out[0]}]

```

```
set_property IOSTANDARD LVCMOS33 [get_ports {seg_out[1]}]
set_property IOSTANDARD LVCMOS33 [get_ports {seg_out[2]}]
set_property IOSTANDARD LVCMOS33 [get_ports {seg_out[3]}]
set_property IOSTANDARD LVCMOS33 [get_ports {seg_out[4]}]
set_property IOSTANDARD LVCMOS33 [get_ports {seg_out[5]}]
set_property IOSTANDARD LVCMOS33 [get_ports {seg_out[6]}]
set_property IOSTANDARD LVCMOS33 [get_ports {seg_out[7]}]
set_property IOSTANDARD LVCMOS33 [get_ports {outplace_carnum[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {outplace_carnum[1]}]
set_property IOSTANDARD LVCMOS33 [get_ports {outplace_carnum[2]}]
set_property IOSTANDARD LVCMOS33 [get_ports {outplace_carnum[3]}]
set_property IOSTANDARD LVCMOS33 [get_ports sw]
set_property IOSTANDARD LVCMOS33 [get_ports come]
set_property IOSTANDARD LVCMOS33 [get_ports place]
set_property IOSTANDARD LVCMOS33 [get_ports stop_car]
set_property IOSTANDARD LVCMOS33 [get_ports leave]
```

```
set_property PACKAGE_PIN Y9 [get_ports sw]
set_property PACKAGE_PIN Y7 [get_ports come]
set_property PACKAGE_PIN W9 [get_ports place]
set_property PACKAGE_PIN Y8 [get_ports stop_car]
set_property PACKAGE_PIN AB8 [get_ports leave]
set_property PACKAGE_PIN AB6 [get_ports {outplace_carnum[0]}]
set_property PACKAGE_PIN V9 [get_ports {outplace_carnum[1]}]
set_property PACKAGE_PIN V8 [get_ports {outplace_carnum[2]}]
set_property PACKAGE_PIN AA8 [get_ports {outplace_carnum[3]}]
set_property PACKAGE_PIN A18 [get_ports {seg_en[7]}]
set_property PACKAGE_PIN A20 [get_ports {seg_en[6]}]
set_property PACKAGE_PIN B20 [get_ports {seg_en[5]}]
set_property PACKAGE_PIN E18 [get_ports {seg_en[4]}]
set_property PACKAGE_PIN F18 [get_ports {seg_en[3]}]
set_property PACKAGE_PIN D19 [get_ports {seg_en[2]}]
set_property PACKAGE_PIN E19 [get_ports {seg_en[1]}]
set_property PACKAGE_PIN C19 [get_ports {seg_en[0]}]
set_property PACKAGE_PIN F15 [get_ports {seg_out[0]}]
set_property PACKAGE_PIN F13 [get_ports {seg_out[1]}]
set_property PACKAGE_PIN F14 [get_ports {seg_out[2]}]
set_property PACKAGE_PIN F16 [get_ports {seg_out[3]}]
set_property PACKAGE_PIN E17 [get_ports {seg_out[4]}]
set_property PACKAGE_PIN C14 [get_ports {seg_out[5]}]
set_property PACKAGE_PIN C15 [get_ports {seg_out[6]}]
set_property PACKAGE_PIN E13 [get_ports {seg_out[7]}]
```

```
set_property IOSTANDARD LVCMOS33 [get_ports {col[3]}]
set_property IOSTANDARD LVCMOS33 [get_ports {col[2]}]
set_property IOSTANDARD LVCMOS33 [get_ports {col[1]}]
set_property IOSTANDARD LVCMOS33 [get_ports {col[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {row[3]}]
set_property IOSTANDARD LVCMOS33 [get_ports {row[2]}]
set_property IOSTANDARD LVCMOS33 [get_ports {row[1]}]
set_property IOSTANDARD LVCMOS33 [get_ports {row[0]}]
```

```
set_property IOSTANDARD LVCMOS33 [get_ports clk]
set_property IOSTANDARD LVCMOS33 [get_ports rst]
```

```
set_property PACKAGE_PIN M2 [get_ports {col[3]}]
set_property PACKAGE_PIN K6 [get_ports {col[2]}]
set_property PACKAGE_PIN J6 [get_ports {col[1]}]
```



```
set_property PACKAGE_PIN L5 [get_ports {col[0]}]
set_property PACKAGE_PIN K4 [get_ports {row[3]}]
set_property PACKAGE_PIN J4 [get_ports {row[2]}]
set_property PACKAGE_PIN L3 [get_ports {row[1]}]
set_property PACKAGE_PIN K3 [get_ports {row[0]}]

set_property PACKAGE_PIN Y18 [get_ports clk]
set_property PACKAGE_PIN P20 [get_ports rst]

set_property IOSTANDARD LVCMOS33 [get_ports {state[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {state[1]}]
set_property IOSTANDARD LVCMOS33 [get_ports {state[2]}]
set_property IOSTANDARD LVCMOS33 [get_ports sw1]
set_property IOSTANDARD LVCMOS33 [get_ports sw2]
set_property IOSTANDARD LVCMOS33 [get_ports next0]
set_property PACKAGE_PIN W4 [get_ports {state[0]}]
set_property PACKAGE_PIN R4 [get_ports {state[1]}]
set_property PACKAGE_PIN T4 [get_ports {state[2]}]
set_property PACKAGE_PIN T5 [get_ports next0]
set_property PACKAGE_PIN U5 [get_ports sw2]
set_property PACKAGE_PIN W6 [get_ports sw1]

set_property IOSTANDARD LVCMOS33 [get_ports {led[15]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[14]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[13]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[12]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[11]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[10]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[9]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[8]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[7]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[6]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[5]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[4]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[3]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[2]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[1]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[16]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[17]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[18]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[19]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[20]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[21]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[22]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[23]}]
set_property PACKAGE_PIN A21 [get_ports {led[0]}]
set_property PACKAGE_PIN E22 [get_ports {led[1]}]
set_property PACKAGE_PIN D22 [get_ports {led[2]}]
set_property PACKAGE_PIN E21 [get_ports {led[3]}]
set_property PACKAGE_PIN D21 [get_ports {led[4]}]
set_property PACKAGE_PIN G21 [get_ports {led[5]}]
set_property PACKAGE_PIN G22 [get_ports {led[6]}]
set_property PACKAGE_PIN F21 [get_ports {led[7]}]
set_property PACKAGE_PIN J17 [get_ports {led[8]}]
set_property PACKAGE_PIN L14 [get_ports {led[9]}]
set_property PACKAGE_PIN L15 [get_ports {led[10]}]
set_property PACKAGE_PIN L16 [get_ports {led[11]}]
```

```

set_property PACKAGE_PIN K16 [get_ports {led[12]}]
set_property PACKAGE_PIN M15 [get_ports {led[13]}]
set_property PACKAGE_PIN M17 [get_ports {led[15]}]
set_property PACKAGE_PIN N19 [get_ports {led[16]}]
set_property PACKAGE_PIN N20 [get_ports {led[17]}]
set_property PACKAGE_PIN M20 [get_ports {led[18]}]
set_property PACKAGE_PIN K13 [get_ports {led[19]}]
set_property PACKAGE_PIN K14 [get_ports {led[20]}]
set_property PACKAGE_PIN M13 [get_ports {led[21]}]
set_property PACKAGE_PIN L13 [get_ports {led[22]}]
set_property PACKAGE_PIN K17 [get_ports {led[23]}]
set_property PACKAGE_PIN M16 [get_ports {led[14]}]

set_property IOSTANDARD LVCMOS33 [get_ports beep]
set_property PACKAGE_PIN A19 [get_ports beep]

```

## 4 总结及优化

### 4.1 问题及解决方案

- 一开始，对Verilog的规范性不注意，多次将一个变量在多个always块中赋值

### 4.2 提升空间

- 代码可读性较差，期望能有修整。
- 有许多难以寻求原因的bug，难以修复，导致未能成功实现。

