

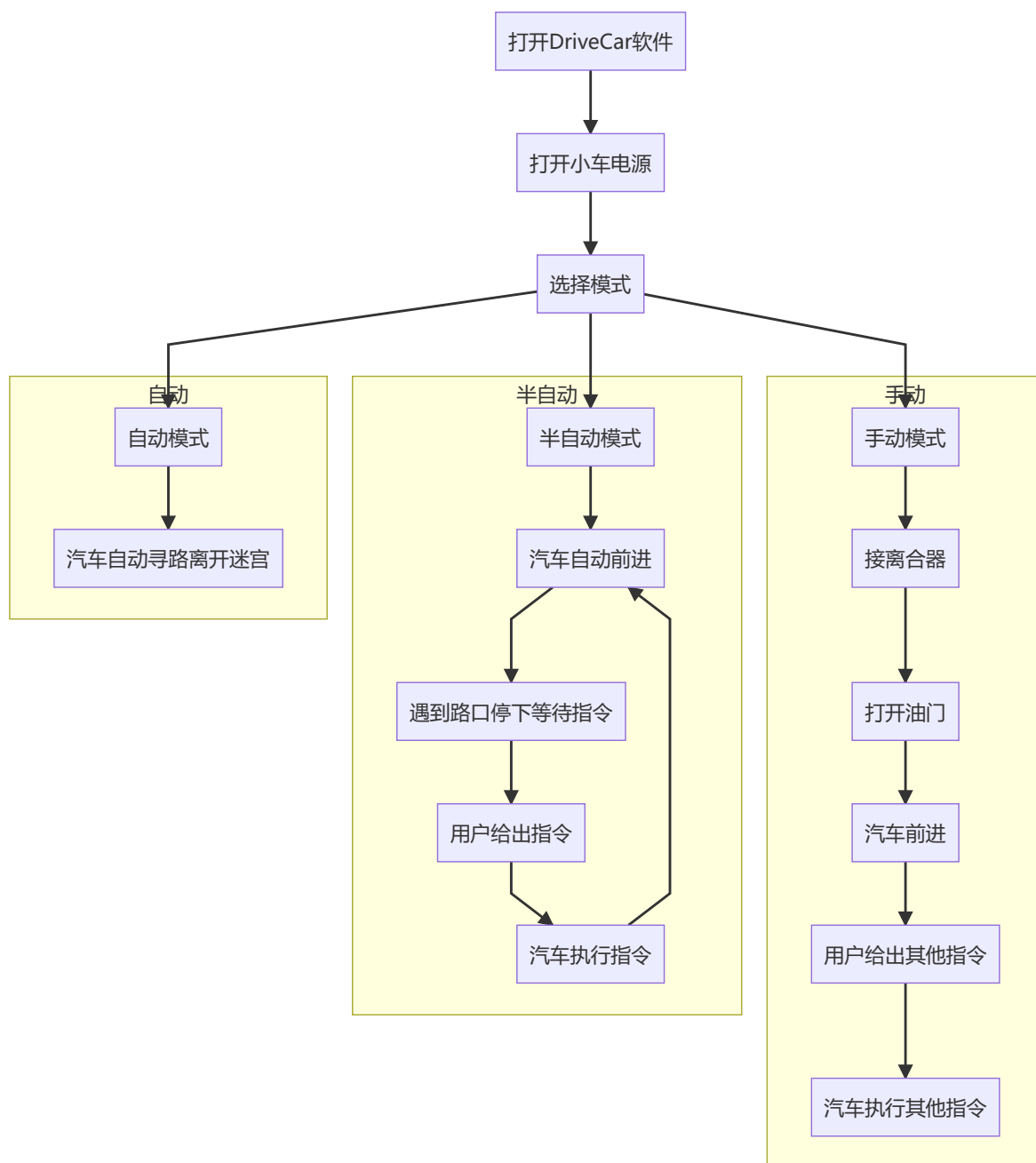
Project - A real car

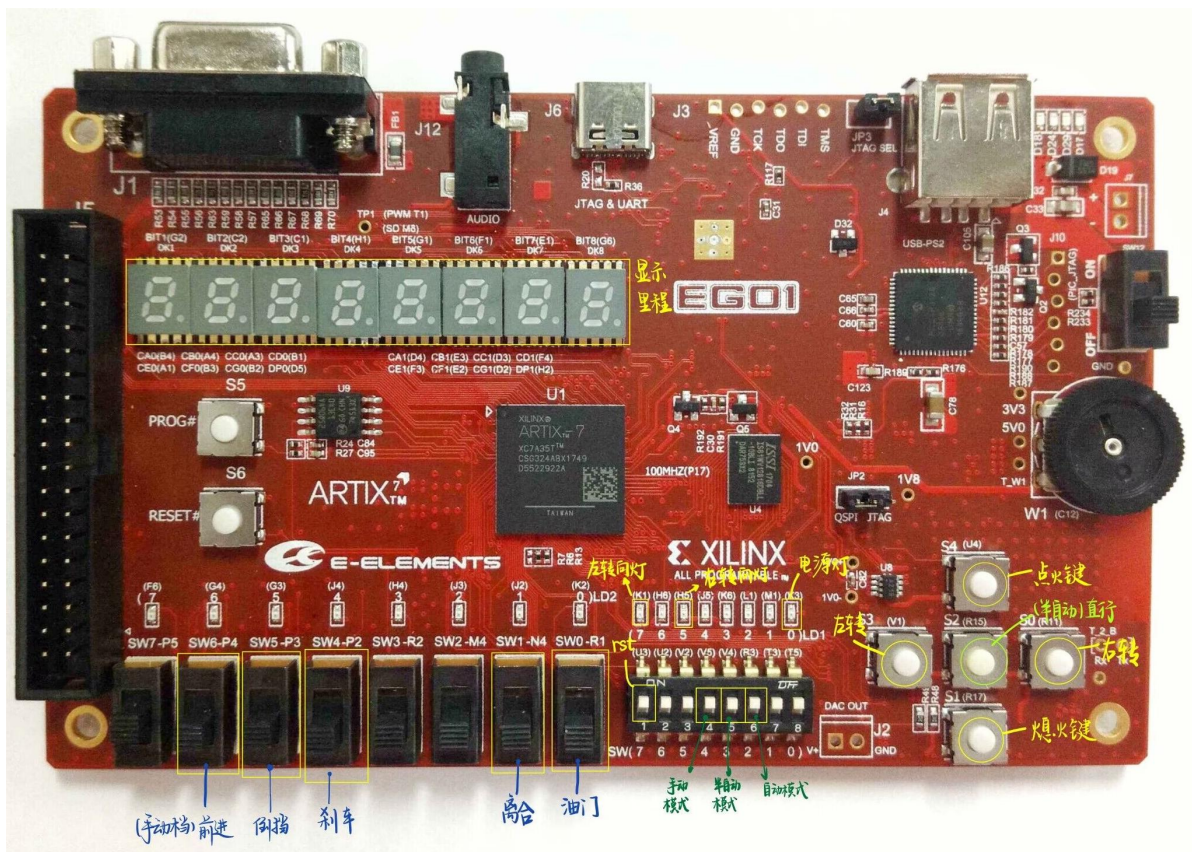
零、分工

- 成员：江睿翔（12111611），张雨菁（12111944）
- 贡献：江睿翔（50%），张雨菁（50%）
- 具体分工

任务	完成
总体框架设计	江睿翔，张雨菁
手动、半自动、自动模块状态转移的设计和实现	张雨菁
电源灯模块、手动模式转向灯和里程记录模块	江睿翔
撰写报告	江睿翔

一、使用说明





- S4是点火键，对应引脚U4
- S1是熄火键，对应引脚R17
- S3和S0是左右转向键，对应引脚V1和R11
- SW7是系统复位信号，低电平有效，对应引脚U3
- SW4, SW3, SW2是选择模式键，分别为手动模式、半自动模式和自动模式，对应引脚V5, V4和R3
- LD1-0是电源灯，对应引脚K3
- LD1-7和LD1-5是左右转向灯，对应引脚K1和H5
- 数码管用于显示里程
- SW6-P4是手动挡前进指令，对应引脚G4
- SW5-P3是倒挡指令，对应引脚G3
- SW4-P2是手动挡刹车指令，对应引脚J4
- SW1-N4是手动挡离合指令，对应引脚J2
- SW0-R1是手动挡油门指令，对应引脚K2

二、新增模块功能和输入输出说明

- 本部分将说明所有新增模块的功能和输入输出规格，demo中已给出的模块不作说明，出现多次的输入输出仅作一次说明

```

module SimulatedDevice(
    input sys_clk, //bind to P17 pin (100MHz system clock)
    input rx, //bind to N5 pin
    output tx, //bind to T4 pin
    input manual_driving, //手动模式选择信号
    input semi_auto_driving, //半自动模式选择信号
    input auto_driving, //自动模式选择信号
    input turn_left_signal, //左转按钮
    input turn_right_signal, //右转按钮
    input go_straight_signal, //直行按钮（半自动）
    input move_forward_signal, //直行按钮（手动）

```

```

input move_backward_signal, //倒挡按钮
input place_barrier_signal, //放置障碍按钮
input destroy_barrier_signal, //摧毁障碍按钮
input Poweron_signal, //电源开启信号
input clutch_signal, //离合按钮
input throttle_signal, //油门按钮
input brake_signal, //刹车按钮
input Poweroff_signal, //电源关闭信号
input rst,

output front_detector, //下面四个都用于探测障碍
output left_detector,
output right_detector,
output back_detector,
output Led_power, //控制电源灯是否开启
output ledL, //左转向灯
output ledR, //右转向灯
output[7:0] seg, //下面三个都用于里程记录
output[7:0] seg1,
output[7:0] an
);

```

- ```

module ledPower(//电源灯
 input clk, //时钟信号
 input rst, //复位信号
 input on, //开启电源按钮
 input off, //关闭电源按钮

 output led //控制电源灯是否开启
);

```

- ```

module Manual_driving( //手动模式
    input manual_driving,
    input semi_auto_driving,
    input auto_driving,
    input turn_left_signal,
    input turn_right_signal,
    input move_forward_signal,
    input move_backward_signal,
    input clutch_signal,
    input throttle_signal,
    input brake_signal,
    input clk,
    input Rst_n,
    input poweron_signal, //电源是否应该开启

    output reg[7:0] out, //给软件传递的信号
    output reg[2:0] state, //小车当前状态
    output wire[7:0] seg,
    output wire[7:0] seg1,
    output wire[7:0] an
);

```

- ```

module ledDesign(//左右转向灯
 input wire clk,
 input wire rst_n,
 input wire left, //小车是否左转
 input wire right, //小车是否右转
 input wire[2:0] state, //小车在手动驾驶模块中的状态

 output reg ledL, //左转向灯是否亮
 output reg ledR //右转向灯是否亮
);

```

- ```

module CountDesign( //里程记录
    input clk,
    input rst,
    input moving, //小车是否在形势
    input poweron, //小车电源是否打开

    output[7:0] seg, //下面三个都用于控制数码管的显示
    output[7:0] seg1,
    output[7:0] an
);

```

- ```

module SemiAuto_Driving(//半自动模式
 input manual_driving,
 input semi_auto_driving,
 input auto_driving,
 input poweron,
 input go_straight_signal,
 input turn_left_signal,
 input turn_right_signal,
 input front_detector,
 input back_detector,
 input left_detector,
 input right_detector,
 input rst,
 input clk,

 output reg[7:0] out
);

```

- ```

module Auto_Driving(//自动模式
    input manual_driving,
    input semi_auto_driving,
    input auto_driving,
    input poweron,
    input place_barrier_signal,
    input destroy_barrier_signal,
    input front_detector,
    input back_detector,
    input left_detector,
    input right_detector,
    input rst,
    input clk,

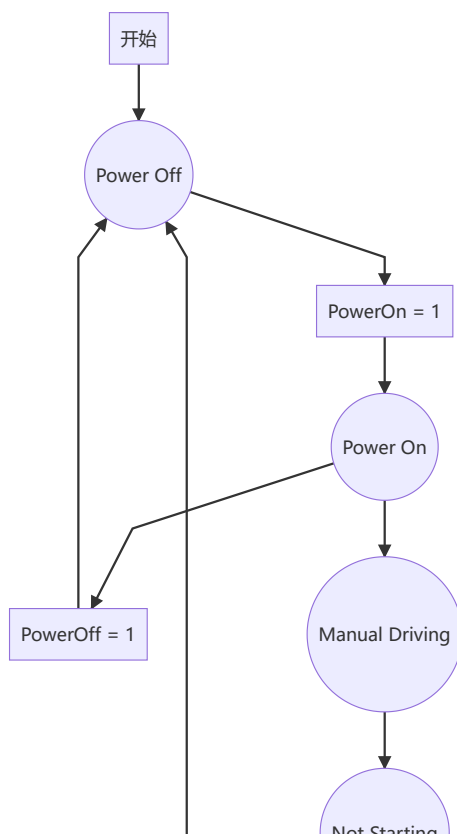
```

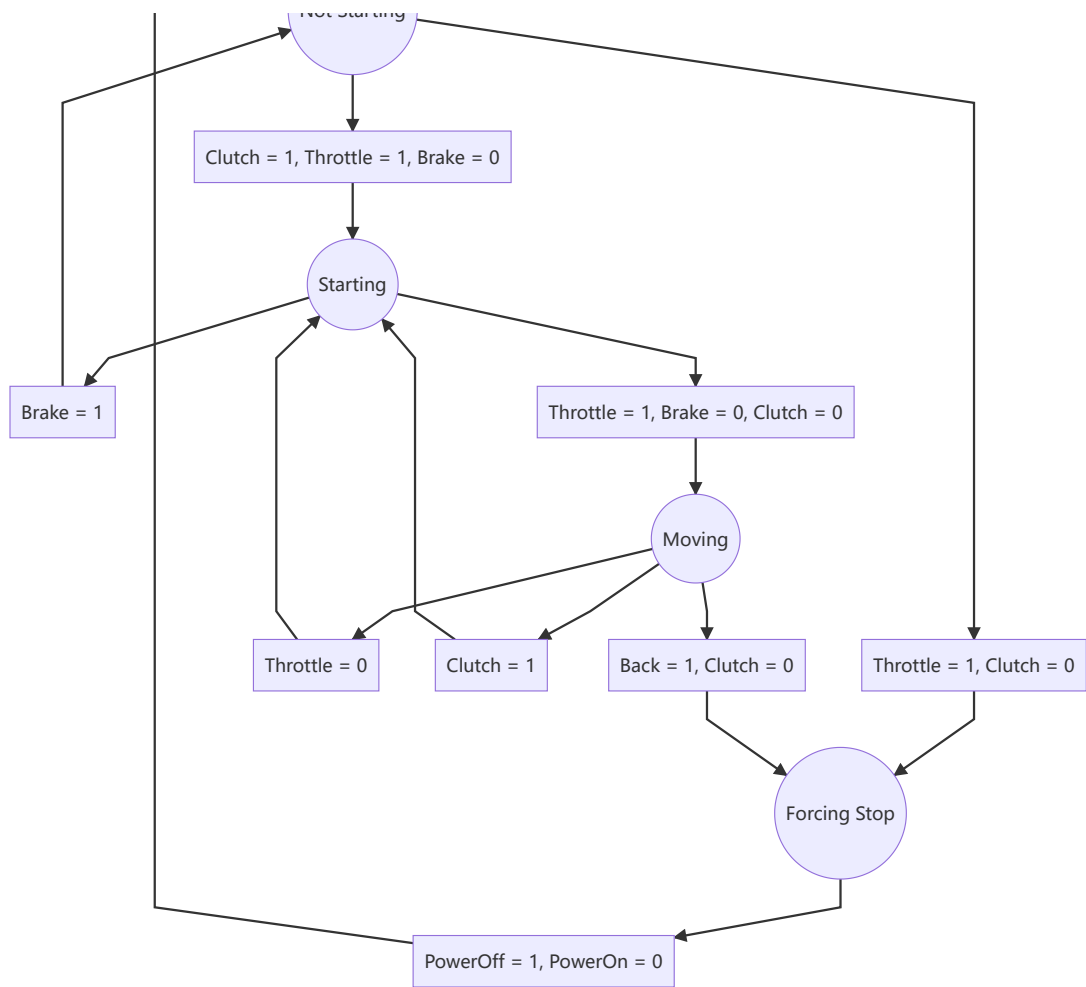
```
output reg[7:0] out
);
```

- ```
module song(sys_clk,beep); //蜂鸣器演奏
input sys_clk; //系统时钟50MHz
output beep; //蜂鸣器输出端
```

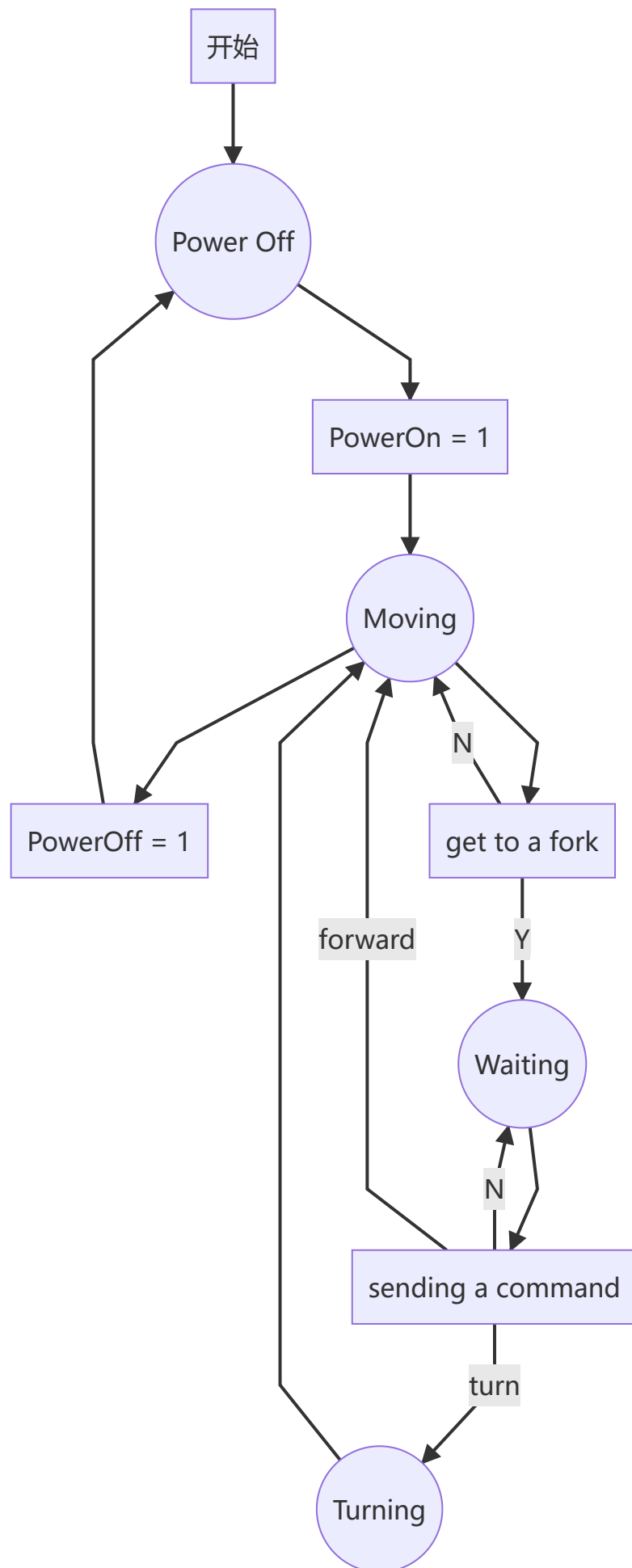
### 三、状态流程图

- 本部分仅阐述需要状态转移的模块的工作机制
- `module Manual_driving`



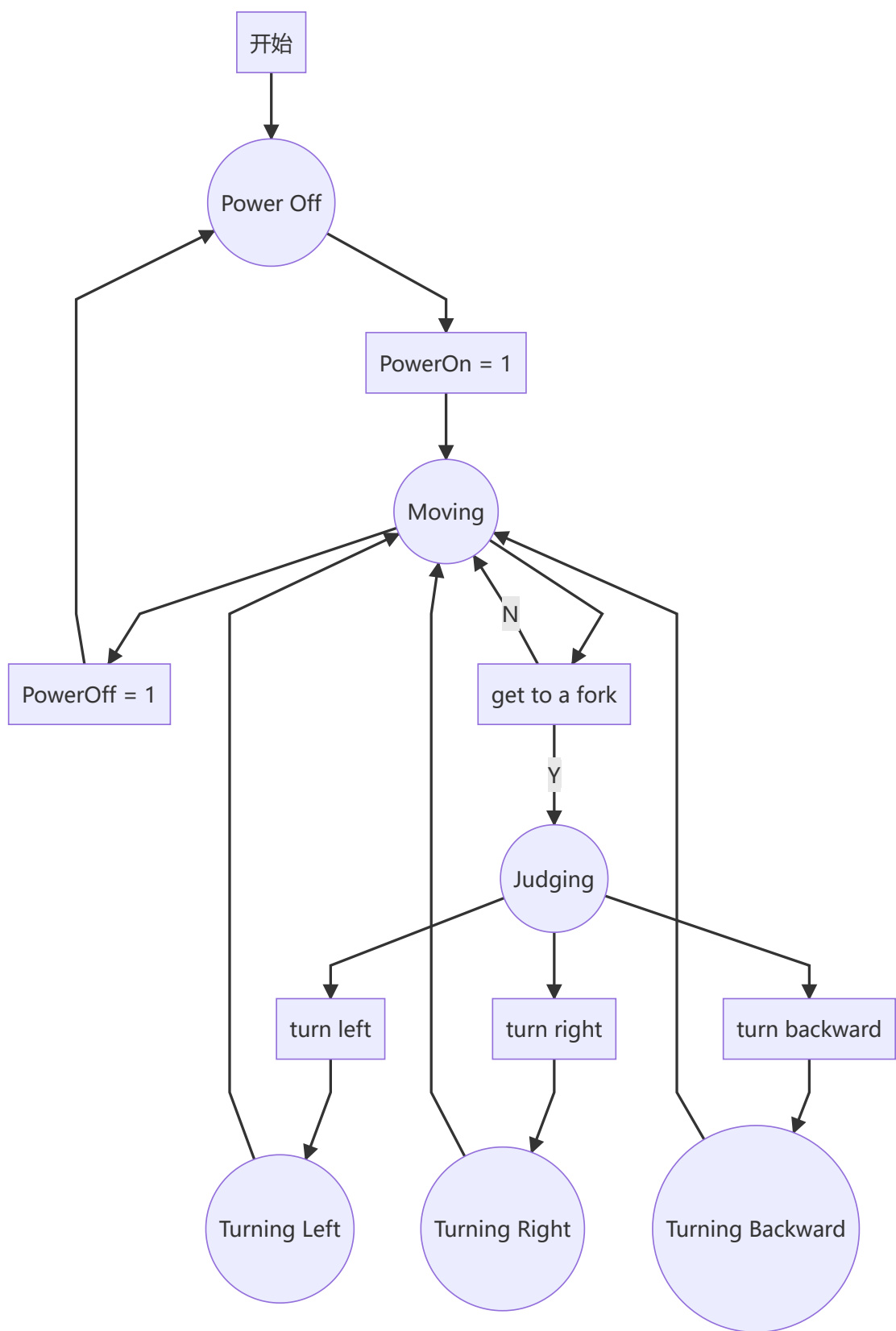


- `module semiAuto_Driving`

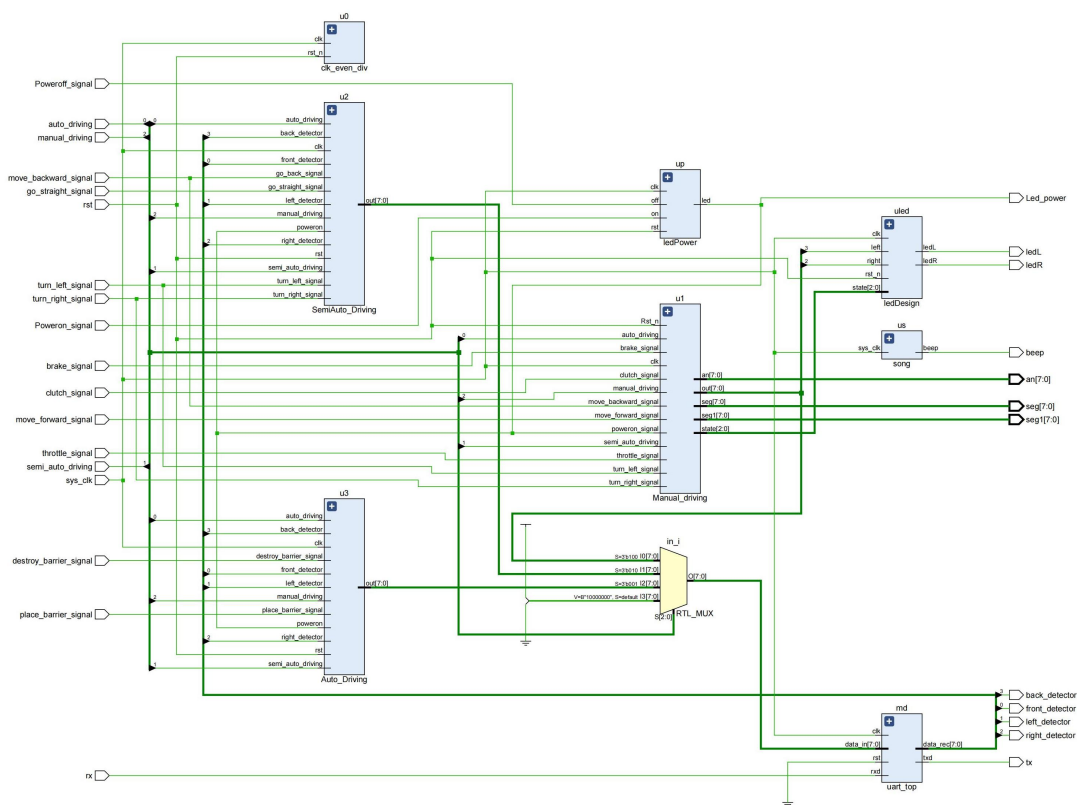


- module auto\_Driving





## 四、系统结构设计



顶层模块 `SimulTcdDevice` 中使用的子模块有：

- `Manual_Driving`
- `semiAuto_Driving`
- `Auto_Driving`
- `ledDesign`
- `ledPower`
- `song`

模块 `Manual_Driving` 中使用的子模块有：

- `CountDesign`

以上各个模块的输入输出和功能均已在第二部分说明，此处不再赘述。

## 五、经验总结

江睿翔：

- 先设计好不同模式的状态，再分别实现状态
- 基本电路的设计尽可能固定化、标准化
- 代码需要规范书写，使用合适的变量名和缩进格式
- 认真对待vivado软件给出的警告

张雨菁：

- 在开始写代码之前做好总设计

- 不规范的代码可能会牵扯出奇怪的bug（比如说当时因为case没列全导致detector一直是高电平，让我一直以为是detector出问题了）
- 不能用软件的思维写硬件
- 小组合作多沟通
- 有解决不了的问题不要死磕，可以问助教或者看看其他人有没有遇见过这样的问题

从最开始拿到小车demo时觉得新奇有趣但不知道怎么上手，到后来每天烧板子几乎玩腻了这个小车，总体还是收获到了很多东西。

最后，感谢教学团队！（尤其是王薇老师对我提出的每一个问题都在很短的时间回复我并且很认真负责地向我解释，感动）