

Digital Logic Project 项目文档

组织与计划

0.1 团队分工

吴鎏亿：设计了状态机，负责按键消抖模块、顶层模块以及控制模块的一部分，也负责主要的 debug

周子涵：设计了数码管输出的逻辑和功放喇叭的输出

王子旭：完成初始状态机模块（状态转换&自清洁模式），负责主要外接键盘bonus实现

0.2 开发计划安排和实施情况

开发计划：选择15周进行答辩，在答辩 4 周前开始，整体按照先分工完成模块再整合debug，再冲击完成bonus

具体实施：

第一周：完成分工，分开完成自己部分module书写

第二周：整合各个模块，对状态进行进一步细化

第三周：进行过程中bug修复，完善细节，尝试完成bonus

第四周答辩前：完成外接输出设备bonus，对外接输入设备进行调试。

Github 入口：

项目细节

1.1 项目简介

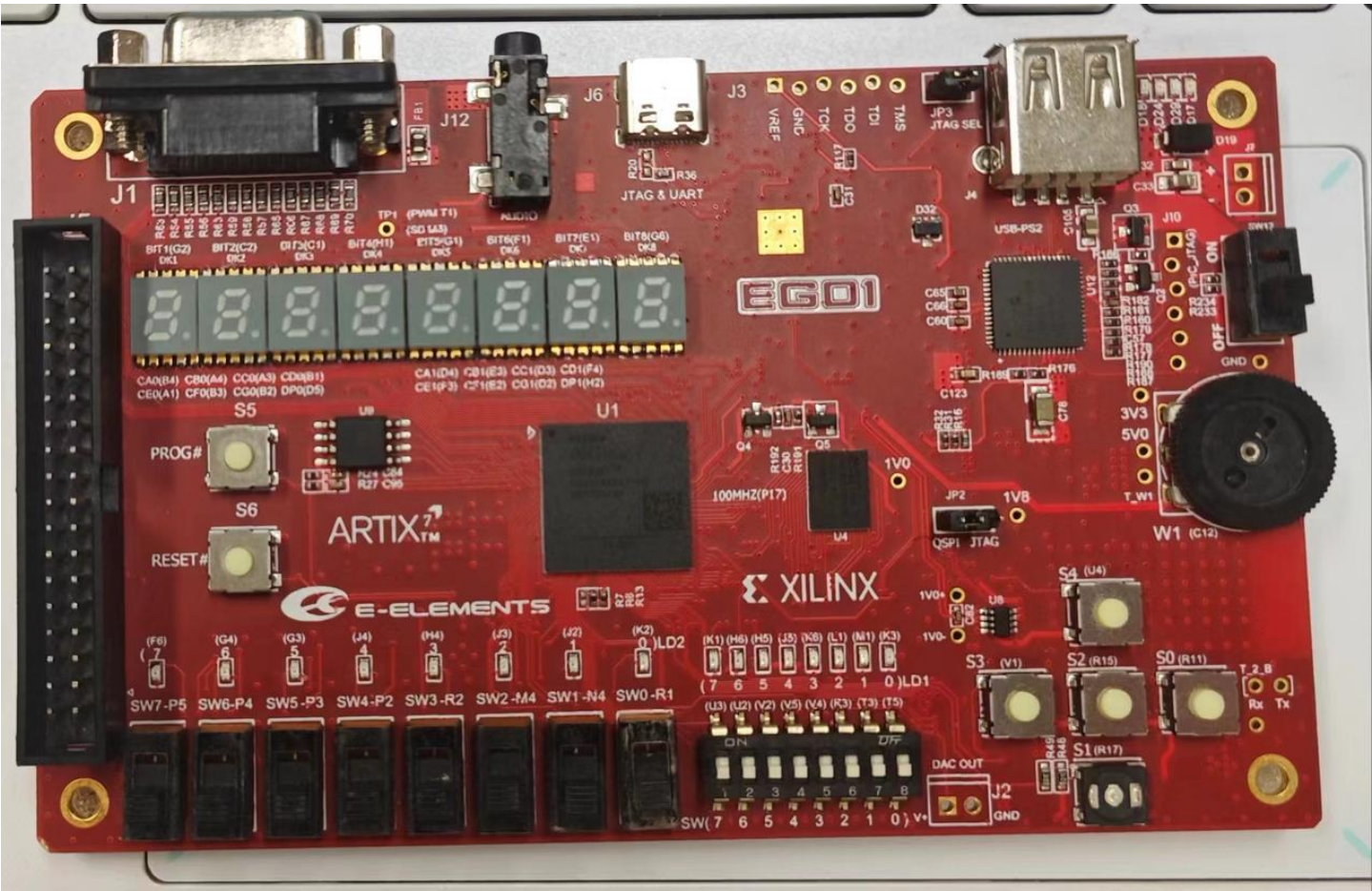
本项目是基于EGO1开发版，verilog语言以及vivado开发工具而实现的一个模拟抽油烟机控制电路的数字逻辑项目。该项目主要采用了模块化设计来实现各个功能，在输入端采用了按键消抖模块，同时在内存中设计了常量池模块，在输出部分有八位七段数码管输出模块和功放喇叭输出模块，最重要的控制逻辑全部集中在控制模块中，所有的模块都在一个顶层模块main.v中被调用。本项目的关键在于状态机，很多功能都与状态的值与变化息息相关。最后在使用时，本项目可以接收按键，开关等输入，通过七段数码管输出当前状态与一些时间辅以声音提醒，以及一些LED灯也会用作状态提醒。

1.2 系统功能列表

本项目完成了所有基础功能，具体包含：

- (1) 两种开关机方式：拨码开关 P 为开时，短按 X 键开机，长按 X 键 3s 关机；在 5s 内 依次按左键、右键即开机，在 5s 内依次按右键、左键，显示倒计时状态，后确认关机。两种开关机方式在任何状态都能生效；
- (2) 能够实现模式转换，包括了：从待机模式进入1档、2档、3档工作模式，1档和2档可以互相切换，累计工作事件；显示飓风模式的60s倒计时，飓风模式强制退出的60s倒计时；
- (3) 完成了自清洁功能，存在手动清洁和自清洁两种方式，都可将累计工作时间清零；自清洁默认为180s倒计时；
- (4) 实现了辅助功能，包括：照明功能、查询功能、时间功能、智能提醒、高级设置；
- (5) 实现了通过八个七段数码管输出状态与时间信息；
- (6) 功放喇叭：实现了不同频率，不同时长声音的输出，且在切换状态时输出用作提醒。

1.3 系统使用说明



将右下角的五个按键分别称作 A,S,D,X,W(根据键盘上的位置定义)，最右边的两个拨码开关依次控制 P,light_dip。

输出包含：1. 8位7段数码管表示当前时间、状态等数据；2. 3个LED灯分别表示“智能提醒”、开机、照明功能；3. 功放喇叭表示状态转换和智能提醒。

关机状态：

在拨码开关P为开时，可以使用短按 X 键直接开机，或者也可以使用手势开关，先按下 A 键，在手势有效时间（默认 5 秒）内按下 D 键实现开机。开机自动进入待机模式，当前时间从 00:00:00 开始计时。

待机模式：

第一次进入待机模式，当前时间、工作时间均为 00:00:00。在油烟机常用模式间的转换中，拨码开关P 均置为关。

在拨码开关P 为关时，W 为菜单键，X 为 查询键。

若要进入工作模式，先按菜单键，后按下 A、S、D、X 分别进入 1档工作、2档工作、飓风模式、自清洁模式。

若要进入查询模式，按下 X 键即可。

工作模式：

在工作模式中，累计工作时间会不断累加，W 键表示返回。7段数码管上会显示当前时间、状态（分别使用 P1,P2,P3 表示）

1档、2档工作状态可自由切换，功放喇叭发出不同声音，可使用 W 键回到待机模式。

飓风模式下，每次开机只能进入一次，自动开始60s倒计时，结束后自动切换至 2 档工作状态。期间按下 W 键会进入强制待机模式，倒计时 60s，不能使用常规按键退出，结束后自动进入待机模式。

如果工作时长超过了智能提醒时间，则对应的警报灯开始闪烁，提醒需要清洁。

自清洁模式：

进入自清洁模式，开始 180s 倒计时，无法使用常规按键退出。自清洁结束回到待机模式，累计工作时间清空。

查询模式：

从待机模式按 查询键进入查询状态，状态栏显示 “FF” ，可按 W 键返回待机模式

分别按下 X、A、S、D 键分别查询 累计工作时间、当前时间、手势开关时间、智能提醒时间。进入具体的查询状态后，可使用 W 键返回查询状态；如果处于可修改的 当前时间、手势开关时间、智能提醒时间 查询状态，可按 X 键进入修改模式。

修改模式：

进入修改模式，7段数码管显示 当前数值和状态，“S3 “、” S2”、“S1” 分别表示正在修改时、分、秒，按 A 表示加1，按 D 表示减1，按 S 表示确认，进入下一个修改单元。如果已经修改完毕，则按 S 自动回到对应的查询状态。

手势有效时间从实际考虑，能设置的值为 1-59 s。特别的，智能提醒时间、手势有效时间**不能设置为0s**，这样的结果会显示 “EE” 状态，且无法保存，保留原来的数值。

手动自清洁按钮：

在开机状态下，在拨码开关P 为开，按下 W 键，会将累计工作时间清空。

恢复出厂设置：

在开机状态下，在拨码开关P 为开，按下 W 键，会强制回到待机状态，当前时间、累计工作时间清空，手势有效时间、智能提醒时间恢复默认设置。

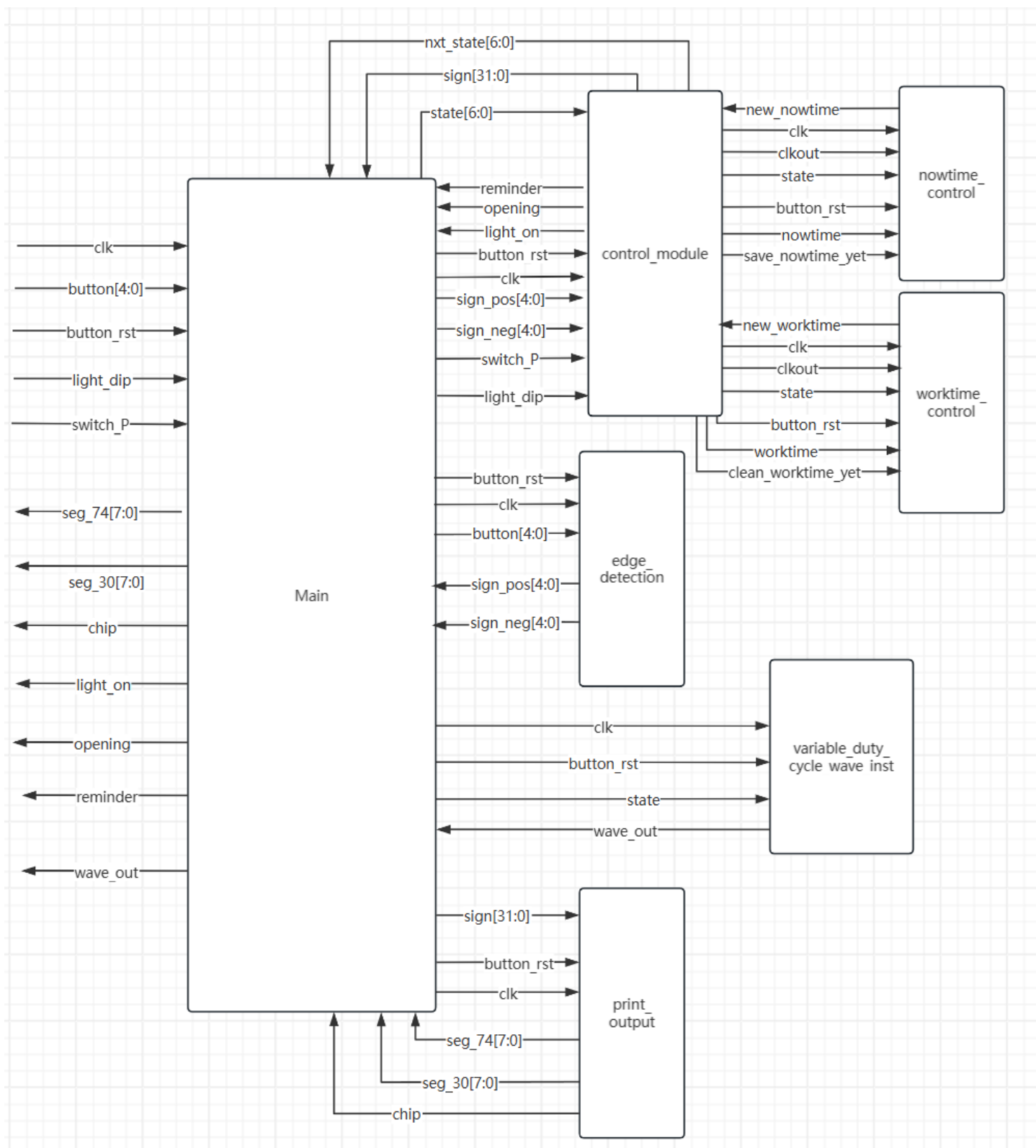
照明模式：

在开机状态，将拨码开关 light_dip 开启，则对应的LED灯亮，不受所有状态影响（除了关机）。

关机方法：

在开机任何状态下，拨码开关P 置为开，长按 X 键 3s 或者在手势有效时间内依次按下 D、A 键均可关机。

1.4 系统结构说明



Main 模块的端口说明：

1. `button[4:0]`: 用于控制状态机的状态转换，分别对应状态机的A、S、W、X、D
2. `switch_P`: 用于控制状态机的状态转换，和按钮结合分别表示不同的状态转换
3. `light_dip`: 用于控制照明模式
4. `clk`: 时钟信号
5. `button_rst`: 用于复位状态机
6. `chip`: 控制7段数码管的片选信号

7. seg_74: 用于控制7段数码管的7-4位数字的段选信号
8. seg_30: 用于控制7段数码管的3-0位数字的段选信号
9. opening: 用于控制油烟机开机状态的信号
10. reminder: 用于控制提醒功能的信号
11. wave_out: 用于控制攻防喇叭的输出

功能描述:Main模块是整个系统的主模块,控制输入信号传入、处理,对状态机和模式进行转换,控制输出信号的输出,实现了整个系统的功能

1.5 子模块功能说明

防抖模块(edge_detection)

输入 `button_A`、`button_S`,以及DWX,还有clk,`button_rst` 等按键,输出对应的信号 `sign_A`,`sign_S` 等6个信号。

控制模块(contaol_module)

接收当前state,并主要根据当前的 `state` 和用户输入的经过消抖后的信号判断输出的模式和状态的切换,最后联系到显示输出模块,声音模块和更新主模块中 `state` 的值

时间模块(nowtime_control & worktime_control)

传入当前的 `state`、`nowtime` (`worktime`)、`clk`、`clkout` 等参数,使用 `new_nowtime``、`new_worktime` 进行修改。将时间模块从控制模块中分离出来,提高系统的模块化程度,使得时间相关的功能更加集中和易于管理。

显示输出模块(print_out)

根据 输入的 `state`,控制 `on_off`、`light` 两个灯,以及根据 `sign` 中的 8 位信号 (32位二进制),分别对应 7段数码管上的8位。设定了一定的周期使得 8个7段数码管轮流显示数字,利用人眼视觉残留实现视觉上的同时发光。

[七段数码管26字母对照表\(附带映射表以及映射数组\)_七段数码管显示字母对应表-CSDN博客](#)

声音模块(variable_duty_cycle_wave_inst)

主要利用了脉冲宽度调制信号。脉冲宽度调制信号是一连串频率固定的脉冲信号,每个脉冲的宽度都可能不同。这种数字信号在通过一个简单的低通滤波器后,被转化为模拟电压信号,电压的大小跟一定区间内的平均脉冲宽度成正比。这个区间由低通滤波器的3dB截止频率和脉冲频率共同决定。根据输入的state,在state切换的时候播放不同频率,不同时长的声音。输出设备为功放喇叭。

PARAMETER: 存放各种常数,包含状态、时间、初始值等常数。

1.6 bonus 的实现

1.手势开关以及手势开关倒计时：本项目实现了手势开关，且在手势开关的关机等待时间内可以随时退出等待并且该过程不影响系统的正常运行。具体为拨码开关P 为开时，可以使用依次按下 D、A 键进行关机，在有效时间内，7段数码管会显示倒计时，若将拨码开关P 置为关，则会显示油烟机目前的状态，二者互不影响。

2.额外输入设备：实现ps2_keyboard.v 模块完成，逻辑上成功实现键盘按下模拟ego1开发板上按键开关按下，但接入顶层设计进行测试时发现：由于ps2协议信号较老，目前很多键盘不支持，同时ps2协议传输不稳定，与键盘规格有关。具体模块内容可见github-src-to complete部分。

3.额外输出设备：本项目实现了功放喇叭不同频率，不同长时的输出，声音在模式切换时会输出，以及在智能提醒状态会持续输出。

项目总结

2.1 项目遇到的问题和解决

1.一开始感觉无从下手，对整体的模块化没有思路。后来，通过画流程图和将功能细分，大量编写代码以及testbench测试，逐渐解决；

2.multi-drive是一个很严重的critical warning，需要及时修改，避免在运行时候的错误。

2.2 本项目的提升空间：

1.能够再使用其他键盘进行测试，彻底完成bonus外接输入设备部分；

2.能够对于模块进一步细化，有助于提升代码可读性和维护能力；

3.声音可以做的更加规范和悦耳。

2.3 团队合作、开发及测试工作的总结

本次项目对于我们每一个队员来说都是很有意义的，首先我们深刻的认识到了verilog语言与其它语言的不同以及它自身运行的一些特性（如multi-drive），也认识到了开发版上的不同接口与不同功能，体会到了如何通过编程语言和模块化设计达成一个电路逻辑以及其中的常见的隔阂。同时我们也尝试很多途径去debug，体会到了找问题在真实工程中的艰难。最后也是感谢每位组员高效，成效地完成了各自的分工和一起解决项目难题，以实现这次Project的提前答辩与较好的展示效果。

2.4 基于 EGO1 的项目构想

1.使用蓝牙或者VGA进行bonus输入以及输出。

2.电子锁系统：本项目旨在使用EGO1开发板和Verilog硬件描述语言实现一个电子锁系统。该电子锁通过输入正确的密码解锁，保证设备的安全性。系统设计包括以下几个核心功能模块：

1. **密码输入模块：**使用外部键盘输入密码，Verilog编写按键扫描程序，读取用户输入。
2. **密码验证模块：**通过硬件逻辑将输入的密码与预设密码进行比对，判断输入是否正确。
3. **控制模块：**如果输入密码正确，控制锁舌的电动机工作，解锁门锁；如果密码错误，保持锁定状态。
4. **指示灯显示：**通过LED指示灯显示当前锁的状态，绿色表示解锁，红色表示密码错误。

硬件平台为EGO1开发板，结合FPGA实现高效的密码验证与锁控逻辑。通过Verilog语言进行数字电路设计，确保系统的稳定性与快速响应。本项目不仅可以提高对硬件编程的理解，同时还将提供一个实用的安全控制系统原型。