

**BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC BÁCH KHOA  
ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH  
KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH  
---o0o---**

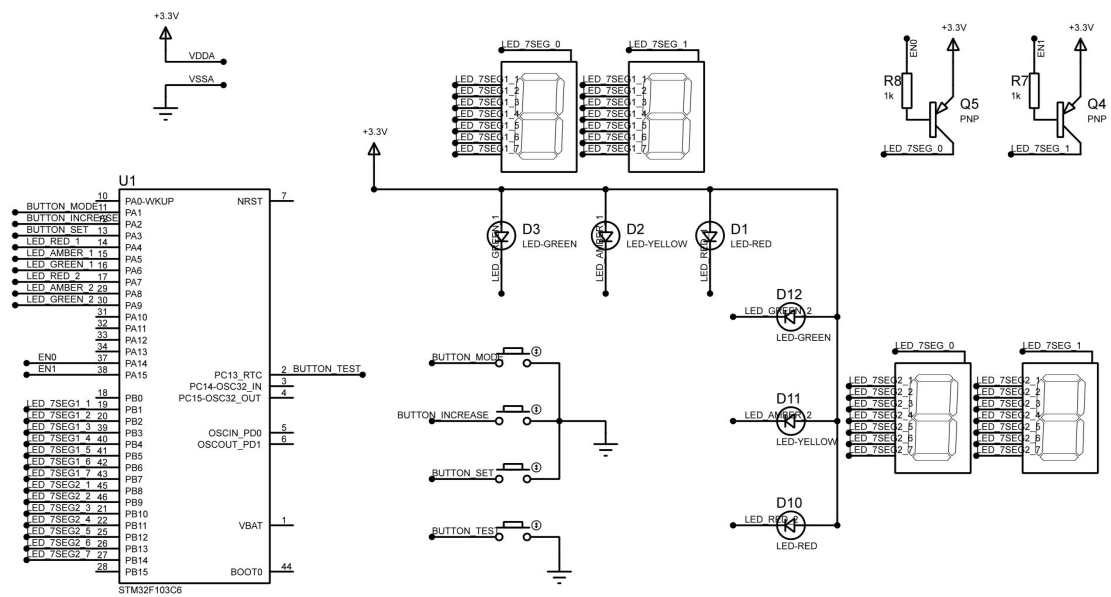
**LAB 3**



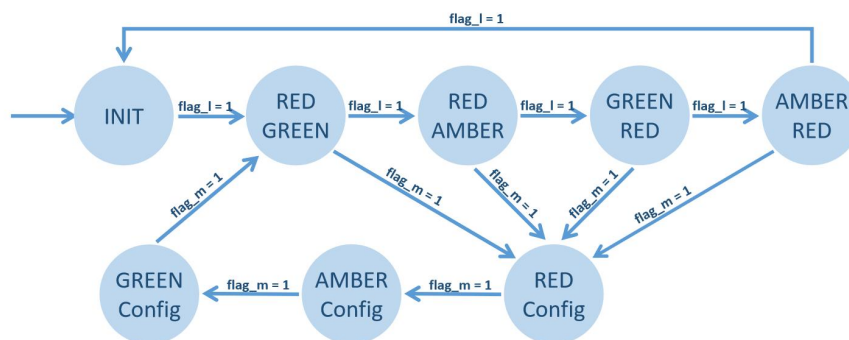
**Giáo viên hướng dẫn: Cao Tiến Đạt  
Sinh viên thực hiện: Phan Minh Trọng**

*22, tháng 12 năm 2024.*

### Schematic use for exercise 1-10



### Exercise 1:



### Exercise 4:

```
void init_SoftwareTimer(uint32_t HCLK, uint32_t Prescaler, uint32_t Period){
    TIMECYCLE = ((1000 * (Prescaler + 1) * (Period + 1)) / (HCLK));
}
```

## Exercise 5:

```
void button_reading(void){
    for(int i = BUTTON_MODE; i < N0_OF_BUTTONS; ++i){
        debounceButtonBuffer2[i] = debounceButtonBuffer1[i];
        debounceButtonBuffer1[i] = HAL_GPIO_ReadPin(BUTTON_MODE_GPIO_Port, (BUTTON_MODE_Pin << i));
        if(debounceButtonBuffer1[i] == debounceButtonBuffer2[i]){
            flagForButtonPress[i] = debounceButtonBuffer1[i];
            if(flagForButtonPress[i] == BUTTON_STATUS_PRESSED){
                if(counterForButtonPress1s[i] < (DEBOUNCE_TIME/TIMECYCLE)){
                    counterForButtonPress1s[i]++;
                } else {
                    flagForButtonPress1s[i] = 1;
                    counterForButtonPress1s[i] = 0;
                }
            } else {
                counterForButtonPress1s[i] = 0;
                flagForButtonPress1s[i] = 0;
            }
        }
    }
}
```

```
unsigned char is_button_pressed(unsigned char index){
    if(index >= N0_OF_BUTTONS) return 0;
    return (flagForButtonPress[index] == BUTTON_STATUS_PRESSED);
}

unsigned char is_button_pressed_1s(unsigned char index){
    if(index >= N0_OF_BUTTONS) return 0xff;
    return (flagForButtonPress1s[index] == 1);
}
```

```

void fsm_input_processing(){
    for(uint8_t i = BUTTON_MODE; i<N0_OF_BUTTONS; ++i){
        switch(buttonState[i]){
            case BUTTON_STATE_RELEASE:
                if(is_button_pressed(i)){
                    buttonState[i] = BUTTON_STATE_PRESSED;
                    flag_m = (i == BUTTON_MODE);
                    flag_i = (i == BUTTON_INCREASE);
                    flag_s = (i == BUTTON_SET);
                }
                break;
            case BUTTON_STATE_PRESSED:
                if(!is_button_pressed(i)){
                    buttonState[i] = BUTTON_STATE_RELEASE;
                    break;
                }
                // if(is_button_pressed_1s(i)){
                //     buttonState[i] = BUTTON_STATE_PRESSED_1S;
                // }
                break;
            case BUTTON_STATE_PRESSED_1S:
                if(!is_button_pressed(i)){
                    buttonState[i] = BUTTON_STATE_RELEASE;
                }
                //TODO
                if(is_button_pressed_1s(INPUT_MODE_ID)){
                    flag_m = 1;
                }
                break;
            default:
                break;
        }
    }
}

```

### Exercise 6-10:

Source code: <https://github.com/FanMinkJonk/MCU-LAB3.git>