VIETNAM NATIONAL UNIVERSITY HO CHI MINH CITY
HO CHI MINH CITY UNIVERSITY OF TECHNOLOGY
FACULTY OF COMPUTER SCIENCE AND ENGINEERING

Microprocessor - Microcontroller

Lab Report - CO3010

Lab 1

Advisor(s): Phan Văn Sỹ
Student(s): Lương Đức Hiếu 2352324

HO CHI MINH CITY, SEPTEMBER 2025

Contents

1 Overall

Lab schematics are submitted via GitHub link: https://github.com/hieuld1003/MPU-MCU

The default while(1) code for most of the exercise is:

```
while(1){
    //Insert function for each exercise
    HAL_Delay(1000);
    }
```

The schematic for the exercises from 1 to 5 and 6 to 10:

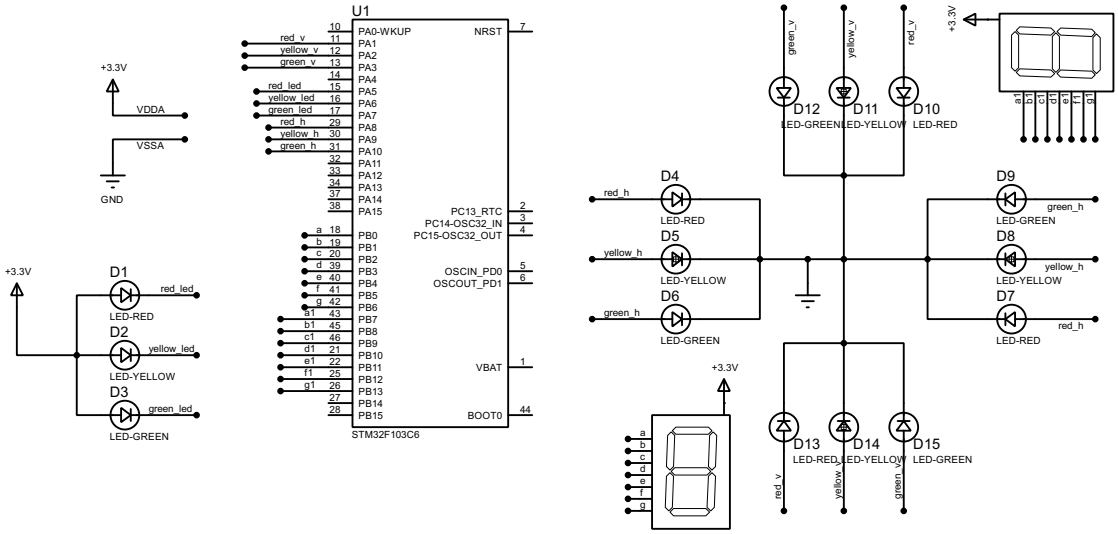

Figure 1.1: Exercise 1 to 5 schematic

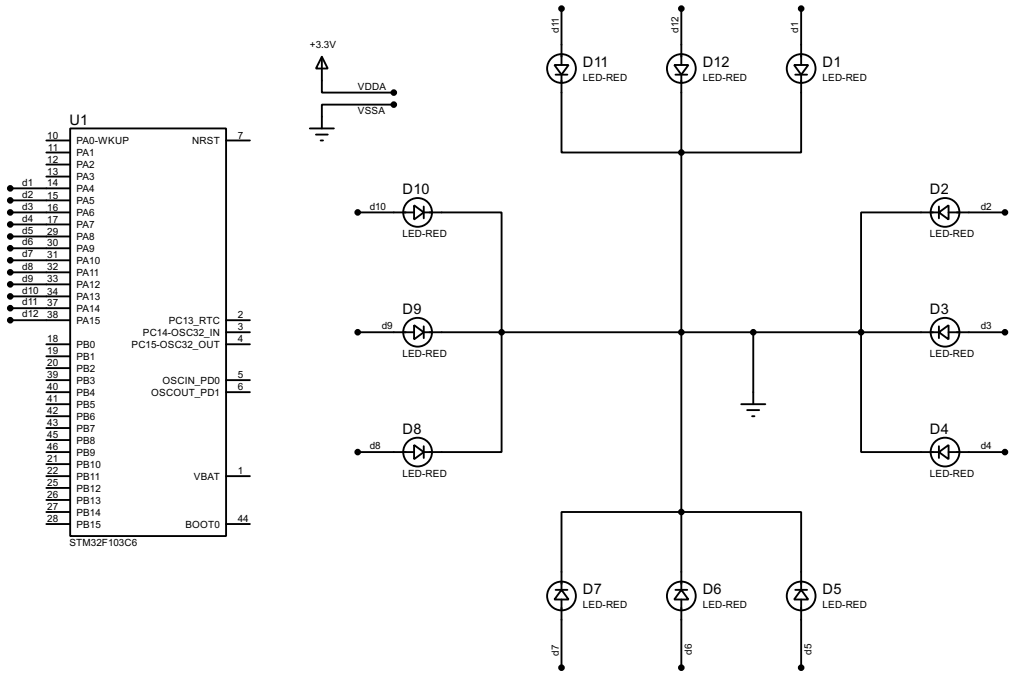

Figure 1.2: Exercise 6 to 10 schematic

2 Exercise 1

2.1 Report 1

Schematic is in **Overall** section.

2.2 Report 2

This is the pseudocode from exercise 1 header file:

```
Initialize:
setInit = 0
count = 0

Main Loop:
while(1) {
    if (setInit == 0) {
        // Initialize LEDs: red OFF, yellow ON
```

```
        setInitLED()
        setInit = 1
    }

    if (count == 1) {
        count = 0
        // Toggle both red and yellow LEDs
        toggleLED()
    } else {
        count = count + 1
    }

    // Wait for next iteration
}
```

3 Exercise 2

3.1 Report 1

Schematic is in **Overall** section.

3.2 Report 2

This is the pseudocode from exercise 2 header file:

```
Initialize:
init = 0
count = 0
phase = 0

Main Loop:
while(1) {
    if (init == 0) {
        // Initial state: red ON, yellow OFF, green OFF
        initial()
        phase = 1
        init = 1
    }
```

```
14
15      switch(phase) {
16          case 1: // Red phase (5 time units)
17              if (count == 5) {
18                  Phase2()  // Toggle red OFF, yellow ON
19                  phase = 2
20                  count = 0
21              }
22              break
23
24          case 2: // Yellow phase (3 time units)
25              if (count == 3) {
26                  Phase3()  // Toggle yellow OFF, green ON
27                  phase = 3
28                  count = 0
29              }
30              break
31
32          case 3: // Green phase (2 time units)
33              if (count == 2) {
34                  Phase1()  // Toggle green OFF, red ON
35                  phase = 1
36                  count = 0
37              }
38              break
39      }
40
41      count = count + 1
42      // Wait for next iteration
43  }
```

4 Exercise 3

4.1 Report 1

Schematic is in **Overall** section.

4.2 Report 2

This is the pseudocode from exercise 3 header file:

```
Initialize:
init = 0
ver_phase = 0, hor_phase = 0
ver_count = 0, hor_count = 0

Main Loop:
while(1) {
    // Initial setup (runs only once)
    if (init == 0) {
        Set vertical: green ON, others OFF
        Set horizontal: red ON, others OFF
        ver_phase = 1, hor_phase = 1
        init = 1
    }

    // Vertical light state machine
    if (ver_count reaches threshold for current phase) {
        ver_count = 0
        Switch to next phase:
        - If phase 1 (green, 3s): change to phase 2 (yellow)
        - If phase 2 (yellow, 2s): change to phase 3 (red)
        - If phase 3 (red, 5s): change to phase 1 (green)
    }

    // Horizontal light state machine
    if (hor_count reaches threshold for current phase) {
        hor_count = 0
        Switch to next phase:
        - If phase 1 (red, 5s): change to phase 2 (green)
```

```
        - If phase 2 (green, 3s): change to phase 3 (yellow)
        - If phase 3 (yellow, 2s): change to phase 1 (red)
    }

    ver_count++
    hor_count++
}
```

5 Exercise 4

5.1 Report 1

Schematic is in **Overall** section.

5.2 Report 2

This is the pseudocode from exercise 4 header file:

```
int arr[10][7] = {
    {0, 0, 0, 0, 0, 0, 1}, //0
    {1, 0, 0, 1, 1, 1, 1}, //1
    {0, 0, 1, 0, 0, 1, 0}, //2
    {0, 0, 0, 0, 1, 1, 0}, //3
    {1, 0, 0, 1, 1, 0, 0}, //4
    {0, 1, 0, 0, 1, 0, 0}, //5
    {0, 1, 0, 0, 0, 0, 0}, //6
    {0, 0, 0, 1, 1, 1, 1}, //7
    {0, 0, 0, 0, 0, 0, 0}, //8
    {0, 0, 0, 0, 1, 0, 0}  //9
};
//   a, b, c, d, e, f, g


Function display7SEG(num):
if (0 <= num <= 9) {
    for (state = 0 to 6) {
        // Set each segment according to the pattern in arr
        Set GPIO pin (GPIOB_PIN_state) to arr[num][state]
    }
} else {
    // Invalid number, turn off all segments
    Turn off all pins on GPIOB
}
```

6 Exercise 5

6.1 Report 1

Schematic is in **Overall** section.

6.2 Report 2

This is the pseudocode from exercise 5 header file:

```
Initialize:
// Same as Exercise 3

Main Loop:
while(1) {
    // Run the traffic light control system
    exercise_3()

    // Display countdown on first 7-segment display for vertical
    switch(ver_phase) {
        case 1: // Green phase
            display7SEG(3 - ver_count)  // Countdown from 3 to 0
            break

        case 2: // Yellow phase
            display7SEG(2 - ver_count)  // Countdown from 2 to 0
            break

        case 3: // Red phase
            display7SEG(5 - ver_count)  // Countdown from 5 to 0
            break
    }

    // Display countdown on second 7-segment display for
       horizontal
    switch(hor_phase) {
        case 1: // Red phase
            display7SEG_2(5 - hor_count)  // Countdown from 5 to 0
            break
```

```
29
30          case 2: // Green phase
31              display7SEG_2(3 - hor_count)  // Countdown from 3 to 0
32              break
33
34          case 3: // Yellow phase
35              display7SEG_2(2 - hor_count)  // Countdown from 2 to 0
36              break
37      }
38
39      // Wait for next iteration
40  }
```

7 Exercise 6

7.1 Report 1

Schematic is in **Overall** section.

7.2 Report 2

This is the pseudocode from exercise 6 header file:

```
//initialize the clock pins
uint16_t clockPins[12] = {
    d12_Pin, d1_Pin, d2_Pin, d3_Pin, d4_Pin, d5_Pin,
    d6_Pin, d7_Pin, d8_Pin, d9_Pin, d10_Pin, d11_Pin
};

void testLEDs() {
    for (int i = 0; i < 12; i++) {
        HAL_GPIO_WritePin(GPIOA, clockPins[i], 1);
        HAL_Delay(500);
    }
}
```

8 Exercise 7

8.1 Report 1

Schematic is in **Overall** section.

8.2 Report 2

This is the pseudocode from exercise 7 header file:

```
void clearAllClock() {
    for (int i = 0; i < 12; i++) {
        HAL_GPIO_WritePin(GPIOA, clockPins[i], 0);
    }
}
```

9 Exercise 8

9.1 Report 1

Schematic is in **Overall** section.

9.2 Report 2

This is the pseudocode from exercise 8 header file:

```
void setNumberOnClock(int num) {
    if (num >= 0 && num < 12) {
        HAL_GPIO_WritePin(GPIOA, clockPins[num], 1);
    }
}
```

10 Exercise 9

10.1 Report 1

Schematic is in **Overall** section.

10.2 Report 2

This is the pseudocode from exercise 9 header file:

```
void clearNumberOnClock(int num) {
    if (num >= 0 && num < 12) {
        HAL_GPIO_WritePin(GPIOA, clockPins[num], 0);
    }
}
```

11 Exercise 10

11.1 Report 1

Schematic is in **Overall** section.

11.2 Report 2

This is the pseudocode from exercise 10 header file

```
int second = 0,
minute = 0,
hour = 0;
void displayClock() {

    ++second;
    if (second > 59){
        second = 0;
        ++minute;
    }
    if (minute > 59){
        minute = 0;
        ++hour;
    }
    if (hour > 23){
        hour = 0;
    }

    clearAllClock();
    hour = hour % 12;
    int minuteLED = minute / 5;
    int secondLED = second / 5;

    setNumberOnClock(hour);
    setNumberOnClock(minuteLED);
    setNumberOnClock(secondLED);
}
```