VIETNAM NATIONAL UNIVERSITY, HO CHI MINH CITY
HO CHI MINH CITY UNIVERSITY OF TECHNOLOGY
Faculty of Computer Science and Engineering

CC02 — Lab Report

Microprocessor - Microcontroller
Lab 1

Supervisors: Nguyen Thien An
Students: Nguyen Thanh Tai 2252722

Ho Chi Minh City, September 17, 2024

Contents

1 Exercise

The GitHub link for the lab schematics is at here or in this link: https://github.com/ThanhTaiNguyen24/mcu-mpu-lab1.

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

```
while(1) {
    // THE FUNCTION INPUT INSERTED HERE
    HAL_Delay(1000);
}
```

The schematic for the exercises from 1 to 5 is located here:

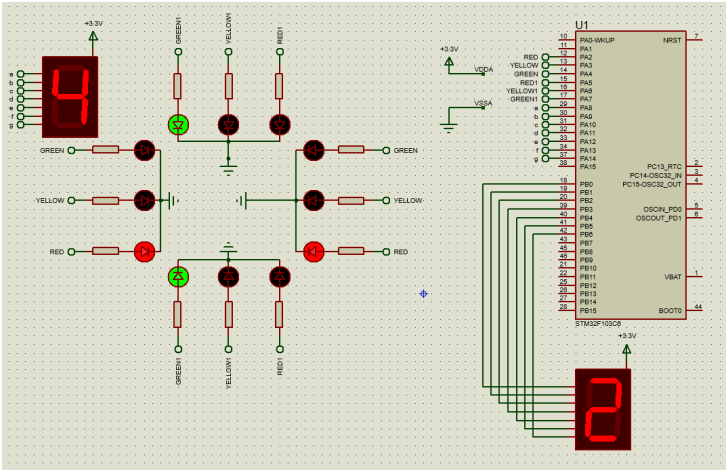

Figure 1: *The schematic for the exercises from 1 to 5.*

The schematic for the exercises from 6 to 10 is located here:

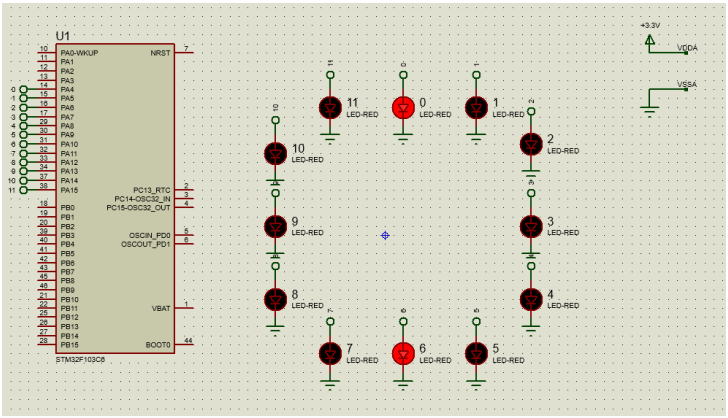

Figure 2: *The schematic for the exercises from 6 to 10.*

1.1 Exercise 1

1.1.1 Report 1

Can be found at 1.

1.1.2 Report 2

This is the header file library for the exercise 1 :

```
while (1)
  {
      HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_5);
      HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_6);
      HAL_Delay(2000);
  }
```

Pre-Setup GPIO output level for both LED:

- RED-LED: High
- RED-YELLOW: Low

1.2 Exercise 2

1.2.1 Report 1

Can be found at 1.

1.2.2 Report 2

This is the source code for the exercise 2:

```
int count = 3;
int counter1 = 3;
while (1)
  {
    //xanh
    if (counter1 > 1 && count == 3){
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_7, GPIO_PIN_RESET);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_6, GPIO_PIN_SET);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, GPIO_PIN_SET);
    }

    //vang
    if (counter1 > 1 && count == 2){
      HAL_GPIO_WritePin(GPIOA, GPIO_PIN_6, GPIO_PIN_RESET);
      HAL_GPIO_WritePin(GPIOA, GPIO_PIN_7, GPIO_PIN_SET);
      HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, GPIO_PIN_SET);
    }
    //do
    if (counter1 > 1 && count == 5){
      HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, GPIO_PIN_RESET);
      HAL_GPIO_WritePin(GPIOA, GPIO_PIN_6, GPIO_PIN_SET);
      HAL_GPIO_WritePin(GPIOA, GPIO_PIN_7, GPIO_PIN_SET);
    }
      counter1--;
      if (counter1 <= 0 && count == 3) {
          count = 2;
          counter1 = 2;
      }
      if (counter1 <= 0 && count == 2) {
          count = 5;
          counter1 = 5;
      }
      if (counter1 <= 0 && count == 5) {
          count = 3;
          counter1 = 3;
      }
      HAL_Delay(1000);
  }
```

1.3 Exercise 3

1.3.1 Report 1

Can be found at 1.

1.3.2 Report 2

This is the source code for the exercise 3:

```
int counter1 = 3;
int count = 3;
while (1)
  {
    if (counter1 > 1 && count == 3){
       HAL_GPIO_WritePin(GPIOA, GPIO_PIN_7, GPIO_PIN_SET);
       HAL_GPIO_WritePin(GPIOA, GPIO_PIN_6, GPIO_PIN_RESET);
       HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, GPIO_PIN_RESET);
       HAL_GPIO_WritePin(GPIOA, GPIO_PIN_2, GPIO_PIN_SET);
       HAL_GPIO_WritePin(GPIOA, GPIO_PIN_3, GPIO_PIN_RESET);
       HAL_GPIO_WritePin(GPIOA, GPIO_PIN_4, GPIO_PIN_RESET);
    }

    if (counter1 > 1 && count == 2){
       HAL_GPIO_WritePin(GPIOA, GPIO_PIN_6, GPIO_PIN_SET);
       HAL_GPIO_WritePin(GPIOA, GPIO_PIN_7, GPIO_PIN_RESET);
       HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, GPIO_PIN_RESET);
       HAL_GPIO_WritePin(GPIOA, GPIO_PIN_2, GPIO_PIN_SET);
       HAL_GPIO_WritePin(GPIOA, GPIO_PIN_3, GPIO_PIN_RESET);
       HAL_GPIO_WritePin(GPIOA, GPIO_PIN_4, GPIO_PIN_RESET);
    }

    if (counter1 > 1 && count == 5){
       if (counter1 > 2){
          HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, GPIO_PIN_SET); //do
          HAL_GPIO_WritePin(GPIOA, GPIO_PIN_6, GPIO_PIN_RESET); //
          HAL_GPIO_WritePin(GPIOA, GPIO_PIN_7, GPIO_PIN_RESET); //
          HAL_GPIO_WritePin(GPIOA, GPIO_PIN_4, GPIO_PIN_SET); //xanh
          HAL_GPIO_WritePin(GPIOA, GPIO_PIN_2, GPIO_PIN_RESET); //
          HAL_GPIO_WritePin(GPIOA, GPIO_PIN_3, GPIO_PIN_RESET); //
       } else {
          HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, GPIO_PIN_SET); //do
          HAL_GPIO_WritePin(GPIOA, GPIO_PIN_6, GPIO_PIN_RESET); //
          HAL_GPIO_WritePin(GPIOA, GPIO_PIN_7, GPIO_PIN_RESET); //
          HAL_GPIO_WritePin(GPIOA, GPIO_PIN_3, GPIO_PIN_SET); //xanh
          HAL_GPIO_WritePin(GPIOA, GPIO_PIN_2, GPIO_PIN_RESET); //
          HAL_GPIO_WritePin(GPIOA, GPIO_PIN_4, GPIO_PIN_RESET); //
       }
      }
        counter1--;
```

```
41              if (counter1 <= 0 && count == 3) {
42                count = 2;
43                counter1 = 2;
44              }
45              if (counter1 <= 0 && count == 2) {
46                count = 5;
47                counter1 = 5;
48              }
49              if (counter1 <= 0 && count == 5) {
50                count = 3;
51                counter1 = 3;
52              }
53              HAL_Delay(1000);
54  }
```

1.4 Exercise 4

1.4.1 Report 1

Can be found at 1.

1.4.2 Report 2

This is the source code for the exercise 4:

```
uint8_t led7segment[10] = {0xC0,0xF9,0xA4,0xB0,0x99,0x92,0x82,0xF8,0x80,0x90};
uint8_t count = 0;
while (1)
  {
    led7segmentdisplay(count);
    count++;
    if (count >= 10) count = 0;
    HAL_Delay(1000);
  }
  void led7segmentdisplay (uint8_t n){
  uint8_t numberCode = led7segment[n];
  uint32_t pin0 = GPIO_PIN_0;
  for(uint8_t i = 0; i<8 ; i++)
  {
    uint32_t pin = pin0 << i;
    uint8_t value = (numberCode >> i) & 0x01;
    HAL_GPIO_WritePin(GPIOB, pin , value);
  }
}
```

1.5 Exercise 5

1.5.1 Report 1

Can be found at 1.

1.5.2 Report 2

This is the source code for the exercise 5:

```
uint8_t led7segment[10] = {0xC0,0xF9,0xA4,0xB0,0x99,0x92};
int counter1 = 3;
int counter2 = 5;
int count = 3;
while (1)
  {

        if (counter1 > 1 && count == 3 && counter2 > 1){
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_7, GPIO_PIN_SET);
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_6, GPIO_PIN_RESET);
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, GPIO_PIN_RESET);
          HAL_GPIO_WritePin(GPIOA, GPIO_PIN_2, GPIO_PIN_SET);
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_3, GPIO_PIN_RESET);
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_4, GPIO_PIN_RESET);

        }
        //vang
        if (counter1 > 1 && count == 2 && counter2 > 1){
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_6, GPIO_PIN_SET);
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_7, GPIO_PIN_RESET);
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, GPIO_PIN_RESET);
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_2, GPIO_PIN_SET);
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_3, GPIO_PIN_RESET);
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_4, GPIO_PIN_RESET);

        }

                      //do
        if (counter1 > 1 && count == 5 && counter2 > 1){

          if (counter1 > 2){
            HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, GPIO_PIN_SET); //do
            HAL_GPIO_WritePin(GPIOA, GPIO_PIN_6, GPIO_PIN_RESET); //
            HAL_GPIO_WritePin(GPIOA, GPIO_PIN_7, GPIO_PIN_RESET); //
            HAL_GPIO_WritePin(GPIOA, GPIO_PIN_4, GPIO_PIN_SET); //xanh
            HAL_GPIO_WritePin(GPIOA, GPIO_PIN_2, GPIO_PIN_RESET); //
            HAL_GPIO_WritePin(GPIOA, GPIO_PIN_3, GPIO_PIN_RESET); //

          } else {
            HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, GPIO_PIN_SET); //do
```

```
            HAL_GPIO_WritePin(GPIOA, GPIO_PIN_6, GPIO_PIN_RESET); //
            HAL_GPIO_WritePin(GPIOA, GPIO_PIN_7, GPIO_PIN_RESET); //
            HAL_GPIO_WritePin(GPIOA, GPIO_PIN_3, GPIO_PIN_SET); //xanh
            HAL_GPIO_WritePin(GPIOA, GPIO_PIN_2, GPIO_PIN_RESET); //
            HAL_GPIO_WritePin(GPIOA, GPIO_PIN_4, GPIO_PIN_RESET); //

          }
        }
        led7segmentdisplay(counter1);
        led7segmentdisplay1(counter2);
          counter1--;
          counter2--;
          if (counter1 <= 0 && count == 3) {
            count = 2;
            counter1 = 2;
          }


          if (counter1 <= 0 && count == 2) {
            count = 5;
            counter1 = 5;
          }
          if (counter1 <= 0 && count == 5) {
            count = 3;
            counter1 = 3;
          }
          if (counter2 <= 0 && count == 3){
            counter2 = 5;
          }
          if (counter2 <= 0 && count == 2) {
            counter2 = 5;
          }
          if (counter2 <= 0 && count == 5){
            if (counter1 > 2){
              counter2 = 3;
            } else {
              counter2 = 2;
            }

          }


          HAL_Delay(1000);
    }
void led7segmentdisplay (uint8_t n){
  uint8_t numberCode = led7segment[n];
  uint32_t pin0 = GPIO_PIN_0;
  for(uint8_t i = 0; i < 8 ; i++)
  {
```

```
90      uint32_t pin = pin0 << i;
91      uint8_t value = (numberCode >> i) & 0x01;
92      HAL_GPIO_WritePin(GPIOB, pin , value);
93    }
94  }
95  void led7segmentdisplay1 (uint8_t n){
96    uint8_t numberCode = led7segment[n];
97      uint32_t pin8 = GPIO_PIN_8;
98      for(uint8_t i = 0; i < 8 ; i++)
99      {
100       uint32_t pin = pin8 << i;
101       uint8_t value = (numberCode >> i) & 0x01;
102       HAL_GPIO_WritePin(GPIOA, pin , value);
103     }
104 }
```

1.6 Exercise 6

1.6.1 Report 1

Can be found at 2.

1.6.2 Report 2

This is the source code for the exercise 6:

```
uint16_t ledNumber[] = {GPIO_PIN_4, GPIO_PIN_5, GPIO_PIN_6, GPIO_PIN_7, GPIO_PIN_8
    , GPIO_PIN_9, GPIO_PIN_10, GPIO_PIN_11, GPIO_PIN_12, GPIO_PIN_13, GPIO_PIN_14,
     GPIO_PIN_15};
while (1)
  {
    for (int i = 0; i < 12; i++){
      HAL_GPIO_TogglePin(GPIOA, ledNumber[i]);
    }
    HAL_Delay(1000);
  }
```

1.7 Exercise 7

This is the code for the exercise 7 `void clearAllClock()`:

```
uint16_t ledNumber[] = {GPIO_PIN_4, GPIO_PIN_5, GPIO_PIN_6, GPIO_PIN_7, GPIO_PIN_8
    , GPIO_PIN_9, GPIO_PIN_10, GPIO_PIN_11, GPIO_PIN_12, GPIO_PIN_13, GPIO_PIN_14,
     GPIO_PIN_15};
void clearAllClock(void){
  for (int i = 0; i < 12; i++) HAL_GPIO_WritePin(GPIOA, ledNumber[i],
    GPIO_PIN_RESET);
}
```

1.8 Exercise 8

This is the code for the exercise 8 `void setNumberOnClock()`:

```
void setNumberOnClock(int num){
    switch(num)
      {
        case 0:
          HAL_GPIO_WritePin(GPIOA, GPIO_PIN_4, GPIO_PIN_SET);
          break;
        case 1:
          HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, GPIO_PIN_SET);
          break;
        case 2:
          HAL_GPIO_WritePin(GPIOA, GPIO_PIN_6, GPIO_PIN_SET);
          break;
        case 3:
          HAL_GPIO_WritePin(GPIOA, GPIO_PIN_7, GPIO_PIN_SET);
          break;
        case 4:
          HAL_GPIO_WritePin(GPIOA, GPIO_PIN_8, GPIO_PIN_SET);
          break;
        case 5:
          HAL_GPIO_WritePin(GPIOA, GPIO_PIN_9, GPIO_PIN_SET);
          break;
        case 6:
          HAL_GPIO_WritePin(GPIOA, GPIO_PIN_10, GPIO_PIN_SET);
          break;
        case 7:
          HAL_GPIO_WritePin(GPIOA, GPIO_PIN_11, GPIO_PIN_SET);
          break;
        case 8:
          HAL_GPIO_WritePin(GPIOA, GPIO_PIN_12, GPIO_PIN_SET);
          break;
        case 9:
          HAL_GPIO_WritePin(GPIOA, GPIO_PIN_13, GPIO_PIN_SET);
          break;
        case 10:
          HAL_GPIO_WritePin(GPIOA, GPIO_PIN_14, GPIO_PIN_SET);
          break;
        case 11:
          HAL_GPIO_WritePin(GPIOA, GPIO_PIN_15, GPIO_PIN_SET);
          break;
      }
}
```

1.9 Exercise 9

This is the code for the exercise 9 `void clearNumberOnClock(int num)`:

```
void clearNumberOnClock(int num){
  switch(num)
        {
          case 0:
            HAL_GPIO_WritePin(GPIOA, GPIO_PIN_4, GPIO_PIN_RESET);
            break;
          case 1:
            HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, GPIO_PIN_RESET);
            break;
          case 2:
            HAL_GPIO_WritePin(GPIOA, GPIO_PIN_6, GPIO_PIN_RESET);
            break;
          case 3:
            HAL_GPIO_WritePin(GPIOA, GPIO_PIN_7, GPIO_PIN_RESET);
            break;
          case 4:
            HAL_GPIO_WritePin(GPIOA, GPIO_PIN_8, GPIO_PIN_RESET);
            break;
          case 5:
            HAL_GPIO_WritePin(GPIOA, GPIO_PIN_9, GPIO_PIN_RESET);
            break;
          case 6:
            HAL_GPIO_WritePin(GPIOA, GPIO_PIN_10, GPIO_PIN_RESET);
            break;
          case 7:
            HAL_GPIO_WritePin(GPIOA, GPIO_PIN_11, GPIO_PIN_RESET);
            break;
          case 8:
            HAL_GPIO_WritePin(GPIOA, GPIO_PIN_12, GPIO_PIN_RESET);
            break;
          case 9:
            HAL_GPIO_WritePin(GPIOA, GPIO_PIN_13, GPIO_PIN_RESET);
            break;
          case 10:
            HAL_GPIO_WritePin(GPIOA, GPIO_PIN_14, GPIO_PIN_RESET);
            break;
          case 11:
            HAL_GPIO_WritePin(GPIOA, GPIO_PIN_15, GPIO_PIN_RESET);
            break;
        }
}
```

1.10 Exercise 10

This is the code for the exercise 10:

```
int second = 0;
int minute = 0;
int hour = 0;
    while (1)
  {
   HAL_Delay(10);

   second++;
   if (second >= 60){
     clearNumberOnClock(second/5 - 1);
     second = 0;
     minute++;
     if (minute >= 60){
       clearNumberOnClock(minute/5 - 1);
       minute = 0;
       hour++;
       if(hour >= 12){
         clearNumberOnClock(hour - 1);
         hour = 0;
       }
     }
   }
   clearNumberOnClock(hour - 1);
   clearNumberOnClock(minute/5 - 1);
   clearNumberOnClock(second/5 - 1);
   setNumberOnClock(hour);
   setNumberOnClock(minute/5);
   setNumberOnClock(second/5);

  }
```

References