

THÍ NGHIỆM: VI XỬ LÝ – VI ĐIỀU KHIỂN

BÁO CÁO

LAB1

Giáo viên hướng dẫn	Huỳnh Phúc Nghị
Sinh viên thực hiện	Trương Phi Trường
Mssv	1915749

Link cho các bài mô phỏng:

https://github.com/TRUONGTRUONG2304/Simulation_Proteus_LAB1

Exercise 1: In this exercise, the status of two LEDs are switched every 2 seconds, as demonstrated in the figure bellow.

Source code:

```
while (1)
{
    /* USER CODE END WHILE */
    // Đảo trạng thái chân PA6
    HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_6);
    HAL_Delay(2000);
    // Đảo trạng thái lần lượt chân PA6 và PA5
    HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_6);
    HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_5);
    HAL_Delay(2000);
    // Đảo trạng thái chân PA5
    HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_5);
    /* USER CODE BEGIN 3 */
}
```

Exercise 2: Extend the first exercise to simulate the behavior of a traffic light. A third LED, named **LED-GREEN** is added to the system, which is connected to **PA7**. A cycle in this traffic light is 5 seconds for the RED, 2 seconds for the YELLOW and 3 seconds for the GREEN. The LED-GREEN is also controlled by its negative pin.

Source code:

```
int state = 1;
```

```

while (1)
{
    /* USER CODE END WHILE */
    // Thực hiện set đồng thời cả 3 chân PA5, PA6, PA7 trạng thái tích cực để tắt hết LED
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, GPIO_PIN_SET);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_6, GPIO_PIN_SET);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_7, GPIO_PIN_SET);
    // Dùng switch kiểm tra trạng thái đầu vào để bật sáng LED tương ứng
    // Case 1: Đèn tại chân PA5 sáng
    // Case 2: Đèn tại chân PA6 sáng
    // Case 3: Đèn tại chân PA7 sáng
    switch(state){
        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;
    }
    HAL_Delay(1000);
    // Chuyển sang trạng thái khác cho LED khác sáng
    state += 1;
    // Nếu trạng thái vượt quá 3 thì set lại trạng thái ban đầu là 1
    if(state > 3) state = 1;
    /* USER CODE BEGIN 3 */
}

```

Exercise 3: Extend to the 4-way traffic light. Arrange 12 LEDs in a nice shape to simulate the behaviors of a traffic light. A reference design can be found in the figure bellow.

Source code:

```

int state = 1;
while (1)
{
    /* USER CODE END WHILE */
    // Thực hiện set đồng thời cả các chân thành trạng thái tích cực để tắt hết LED
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_1, GPIO_PIN_SET);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_2, GPIO_PIN_SET);

```

```

    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_3, GPIO_PIN_SET);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_13, GPIO_PIN_SET);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_14, GPIO_PIN_SET);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_15, GPIO_PIN_SET);
//    Dùng switch để kiểm tra trạng thái đầu vào
//    Case1 cho led ở PA1(ngang) và PA15(đọc) sáng
//    Case2 cho led ở PA2(ngang) và PA14(đọc) sáng
//    Case3 cho led ở PA4(ngang) và PA13(đọc) sáng
    switch(state){
    case 1:
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_1, GPIO_PIN_RESET);
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_15, GPIO_PIN_RESET);
        break;
    case 2:
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_2, GPIO_PIN_RESET);
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_14, GPIO_PIN_RESET);
        break;
    case 3:
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_3, GPIO_PIN_RESET);
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_13, GPIO_PIN_RESET);
        break;
    }
//    Chuyển sang trạng thái khác cho LED khác sáng
    state += 1;
//    Nếu trạng thái vượt quá 3 thì set lại trạng thái ban đầu là 1
    if(state > 3) state = 1;
    HAL_Delay(1000);
    /* USER CODE BEGIN 3 */
}

```

Exercise 4: Add **only one 7 led segment** to the schematic in Exercise 3. This component can be found in Proteus by the keyword **7SEG-COM-ANODE**. For this device, the common pin should be connected to the power supply and other pins are supposed to be connected to PB0 to PB6. Therefore, to turn-on a segment in this 7SEG, the STM32 pin should be in logic 0 (0V).

Source code:

```

void display7SEG(int count){
//    Thực hiện reset LED 7 đoạn cho các chân thành 0 để bật sáng hết cả 7 đoạn
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_1, GPIO_PIN_RESET);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_2, GPIO_PIN_RESET);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_3, GPIO_PIN_RESET);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_4, GPIO_PIN_RESET);

```

```
HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, GPIO_PIN_RESET);
HAL_GPIO_WritePin(GPIOA, GPIO_PIN_6, GPIO_PIN_RESET);
HAL_GPIO_WritePin(GPIOA, GPIO_PIN_7, GPIO_PIN_RESET);
// Dùng switch để kiểm tra đầu vào là số nào cần hiển thị
// Sau đó tắt đi các chân không cần thiết để được số như đầu vào
// Thứ tự tương ứng trong mô phỏng 7 6 5 4 3 2 1 (Proteus) -> a b c d e f g (LED 7
đoạn)
```

```
switch(count){
```

```
case 0:
```

```
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_1, GPIO_PIN_SET);
```

```
    break;
```

```
case 1:
```

```
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_1, GPIO_PIN_SET);
```

```
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_2, GPIO_PIN_SET);
```

```
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_3, GPIO_PIN_SET);
```

```
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_4, GPIO_PIN_SET);
```

```
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_7, GPIO_PIN_SET);
```

```
    break;
```

```
case 2:
```

```
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_2, GPIO_PIN_SET);
```

```
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, GPIO_PIN_SET);
```

```
    break;
```

```
case 3:
```

```
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_2, GPIO_PIN_SET);
```

```
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_3, GPIO_PIN_SET);
```

```
    break;
```

```
case 4:
```

```
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_3, GPIO_PIN_SET);
```

```
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_4, GPIO_PIN_SET);
```

```
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_7, GPIO_PIN_SET);
```

```
    break;
```

```
case 5:
```

```
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_3, GPIO_PIN_SET);
```

```
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_6, GPIO_PIN_SET);
```

```
    break;
```

```
case 6:
```

```
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_6, GPIO_PIN_SET);
```

```
    break;
```

```
case 7:
```

```
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_1, GPIO_PIN_SET);
```

```
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_2, GPIO_PIN_SET);
```

```
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_3, GPIO_PIN_SET);
```

```
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_4, GPIO_PIN_SET);
```

```
    break;
```

```
case 8:
```

```
    break;
```

```

        case 9:
            HAL_GPIO_WritePin(GPIOA, GPIO_PIN_3, GPIO_PIN_SET);
            break;
    }
}

```

```

int counter = 0;
while (1)
{
    /* USER CODE END WHILE */
    if(counter >= 10) counter = 0;
    display7SEG(counter++);
    HAL_Delay(1000);
    /* USER CODE BEGIN 3 */
}

```

Exercise 5: Integrate the 7SEG-LED to the 4 ways traffic light. In this case, the 7SEG-LED is used to display countdown value.

In this exercise, only source code is required to present. The function display7SEG in previous exercise can be re-used.

Source code:

```

void trafficLightRow(int state){
    // Tắt cả 3 đèn ở ngã tư
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_1, GPIO_PIN_SET);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_2, GPIO_PIN_SET);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_3, GPIO_PIN_SET);
    // Kiểm tra trạng thái vào là gì bằng switch
    // Nếu 1 thì đèn tại chân PA1 sáng
    // Nếu 2 thì đèn tại chân PA2 sáng
    // Nếu 3 thì đèn tại chân PA3 sáng
    switch(state){
        case 1:
            HAL_GPIO_WritePin(GPIOA, GPIO_PIN_1, GPIO_PIN_RESET);
            break;
        case 2:
            HAL_GPIO_WritePin(GPIOA, GPIO_PIN_2, GPIO_PIN_RESET);
            break;
        case 3:
            HAL_GPIO_WritePin(GPIOA, GPIO_PIN_3, GPIO_PIN_RESET);
            break;
    }
}

```

```

//Hiển thị trạng thái của 3 đèn (hàng dọc ở ngã tư)
void trafficLightCol(int state){
//    Tắt cả 3 đèn ở ngã tư
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_13, GPIO_PIN_SET);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_14, GPIO_PIN_SET);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_15, GPIO_PIN_SET);
//    Kiểm tra trạng thái vào là gì bằng switch
//    Nếu 1 thì đèn tại chân PA13 sáng
//    Nếu 2 thì đèn tại chân PA14 sáng
//    Nếu 3 thì đèn tại chân PA15 sáng
    switch(state){
case 1:
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_13, GPIO_PIN_RESET);
        break;
case 2:
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_14, GPIO_PIN_RESET);
        break;
case 3:
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_15, GPIO_PIN_RESET);
        break;
    }
}

//Hiển thị số đếm trên LED 7 đoạn (hàng ngang ở ngã tư)
void display7SEGrow(int count){
//    Thực hiện reset LED 7 đoạn cho các chân thành 0 để bật sáng hết cả 7 đoạn
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_0, GPIO_PIN_RESET);
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_1, GPIO_PIN_RESET);
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_2, GPIO_PIN_RESET);
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_3, GPIO_PIN_RESET);
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_4, GPIO_PIN_RESET);
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_5, GPIO_PIN_RESET);
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_6, GPIO_PIN_RESET);
//    Dùng switch để kiểm tra đầu vào là số nào cần hiển thị
//    Thứ tự tương ứng trong mô phỏng 0 1 2 3 4 5 6 (Proteus) -> a b c d e f g (LED 7
đoạn)
//    Sau đó tắt đi các chân không cần thiết để được số như đầu vào
    switch(count){
case 0:
        HAL_GPIO_WritePin(GPIOB, GPIO_PIN_6, GPIO_PIN_SET);
        break;
case 1:
        HAL_GPIO_WritePin(GPIOB, GPIO_PIN_0, GPIO_PIN_SET);
        HAL_GPIO_WritePin(GPIOB, GPIO_PIN_3, GPIO_PIN_SET);
        HAL_GPIO_WritePin(GPIOB, GPIO_PIN_4, GPIO_PIN_SET);
        HAL_GPIO_WritePin(GPIOB, GPIO_PIN_5, GPIO_PIN_SET);
        HAL_GPIO_WritePin(GPIOB, GPIO_PIN_6, GPIO_PIN_SET);

```

```

        break;
    case 2:
        HAL_GPIO_WritePin(GPIOB, GPIO_PIN_2, GPIO_PIN_SET);
        HAL_GPIO_WritePin(GPIOB, GPIO_PIN_5, GPIO_PIN_SET);
        break;
    case 3:
        HAL_GPIO_WritePin(GPIOB, GPIO_PIN_4, GPIO_PIN_SET);
        HAL_GPIO_WritePin(GPIOB, GPIO_PIN_5, GPIO_PIN_SET);
        break;
    case 4:
        HAL_GPIO_WritePin(GPIOB, GPIO_PIN_0, GPIO_PIN_SET);
        HAL_GPIO_WritePin(GPIOB, GPIO_PIN_3, GPIO_PIN_SET);
        HAL_GPIO_WritePin(GPIOB, GPIO_PIN_4, GPIO_PIN_SET);
        break;
    case 5:
        HAL_GPIO_WritePin(GPIOB, GPIO_PIN_1, GPIO_PIN_SET);
        HAL_GPIO_WritePin(GPIOB, GPIO_PIN_4, GPIO_PIN_SET);
        break;
    case 6:
        HAL_GPIO_WritePin(GPIOB, GPIO_PIN_1, GPIO_PIN_SET);
        break;
    case 7:
        HAL_GPIO_WritePin(GPIOB, GPIO_PIN_3, GPIO_PIN_SET);
        HAL_GPIO_WritePin(GPIOB, GPIO_PIN_4, GPIO_PIN_SET);
        HAL_GPIO_WritePin(GPIOB, GPIO_PIN_5, GPIO_PIN_SET);
        HAL_GPIO_WritePin(GPIOB, GPIO_PIN_6, GPIO_PIN_SET);
        break;
    case 8:
        break;
    case 9:
        HAL_GPIO_WritePin(GPIOB, GPIO_PIN_4, GPIO_PIN_SET);
        break;
    }
}
//Hiển thị số đếm trên LED 7 đoạn (hàng ngang ở ngã tư)
void display7SEGcol(int count){
//    Thực hiện reset LED 7 đoạn cho các chân thành 0 để bật sáng hết cả 7 đoạn
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_7, GPIO_PIN_RESET);
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_8, GPIO_PIN_RESET);
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_9, GPIO_PIN_RESET);
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_10, GPIO_PIN_RESET);
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_11, GPIO_PIN_RESET);
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_12, GPIO_PIN_RESET);
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_13, GPIO_PIN_RESET);
//    Dùng switch để kiểm tra đầu vào là số nào cần hiển thị

```

//Thứ tự tương ứng trong mô phỏng 7 8 9 10 11 12 13 (Proteus) -> a b c d e f g (LED 7 đoạn)

// Sau đó tắt đi tất cả chân không cần thiết để được số như đầu vào

switch(count){

case 0:

HAL_GPIO_WritePin(GPIOB, GPIO_PIN_13, *GPIO_PIN_SET*);

break;

case 1:

HAL_GPIO_WritePin(GPIOB, GPIO_PIN_7, *GPIO_PIN_SET*);

HAL_GPIO_WritePin(GPIOB, GPIO_PIN_10, *GPIO_PIN_SET*);

HAL_GPIO_WritePin(GPIOB, GPIO_PIN_11, *GPIO_PIN_SET*);

HAL_GPIO_WritePin(GPIOB, GPIO_PIN_12, *GPIO_PIN_SET*);

HAL_GPIO_WritePin(GPIOB, GPIO_PIN_13, *GPIO_PIN_SET*);

break;

case 2:

HAL_GPIO_WritePin(GPIOB, GPIO_PIN_9, *GPIO_PIN_SET*);

HAL_GPIO_WritePin(GPIOB, GPIO_PIN_12, *GPIO_PIN_SET*);

break;

case 3:

HAL_GPIO_WritePin(GPIOB, GPIO_PIN_11, *GPIO_PIN_SET*);

HAL_GPIO_WritePin(GPIOB, GPIO_PIN_12, *GPIO_PIN_SET*);

break;

case 4:

HAL_GPIO_WritePin(GPIOB, GPIO_PIN_7, *GPIO_PIN_SET*);

HAL_GPIO_WritePin(GPIOB, GPIO_PIN_10, *GPIO_PIN_SET*);

HAL_GPIO_WritePin(GPIOB, GPIO_PIN_11, *GPIO_PIN_SET*);

break;

case 5:

HAL_GPIO_WritePin(GPIOB, GPIO_PIN_8, *GPIO_PIN_SET*);

HAL_GPIO_WritePin(GPIOB, GPIO_PIN_11, *GPIO_PIN_SET*);

break;

case 6:

HAL_GPIO_WritePin(GPIOB, GPIO_PIN_8, *GPIO_PIN_SET*);

break;

case 7:

HAL_GPIO_WritePin(GPIOB, GPIO_PIN_10, *GPIO_PIN_SET*);

HAL_GPIO_WritePin(GPIOB, GPIO_PIN_11, *GPIO_PIN_SET*);

HAL_GPIO_WritePin(GPIOB, GPIO_PIN_12, *GPIO_PIN_SET*);

HAL_GPIO_WritePin(GPIOB, GPIO_PIN_13, *GPIO_PIN_SET*);

break;

case 8:

break;

case 9:

HAL_GPIO_WritePin(GPIOB, GPIO_PIN_11, *GPIO_PIN_SET*);**break**;

}

}

Exercise 6: In this exercise, a new Proteus schematic is designed to simulate an analog clock, with 12 different number. The connections for 12 LEDs are supposed from PA4 to PA15 of the STM32. The arrangement of 12 LEDs is depicted as follows

Source code:

```
//Hàm kiểm tra hoạt động
//Thực hiện bật sáng tất cả các đèn (set giá trị âm) theo thứ tự các LED ở chân 4->15
//Với độ trễ mỗi lần bật tắt là 0.5s
void Testing(){
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_4, GPIO_PIN_RESET);
    HAL_Delay(500);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, GPIO_PIN_RESET);
    HAL_Delay(500);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_6, GPIO_PIN_RESET);
    HAL_Delay(500);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_7, GPIO_PIN_RESET);
    HAL_Delay(500);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_8, GPIO_PIN_RESET);
    HAL_Delay(500);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_9, GPIO_PIN_RESET);
    HAL_Delay(500);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_10, GPIO_PIN_RESET);
    HAL_Delay(500);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_11, GPIO_PIN_RESET);
    HAL_Delay(500);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_12, GPIO_PIN_RESET);
    HAL_Delay(500);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_13, GPIO_PIN_RESET);
    HAL_Delay(500);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_14, GPIO_PIN_RESET);
    HAL_Delay(500);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_15, GPIO_PIN_RESET);
    HAL_Delay(500);
}
```

Exercise 7:

Implement a function named **clearAllClock()** to turn off all 12 LEDs. Present the source code of this function.

Source code:

```
//Hàm tắt tất cả 12 LED
```

```
//Set giá trị dương cho 12 chân nối với chân âm của LED (4->15)
void clearAllClock(){
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_4, GPIO_PIN_SET);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, GPIO_PIN_SET);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_6, GPIO_PIN_SET);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_7, GPIO_PIN_SET);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_8, GPIO_PIN_SET);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_9, GPIO_PIN_SET);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_10, GPIO_PIN_SET);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_11, GPIO_PIN_SET);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_12, GPIO_PIN_SET);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_13, GPIO_PIN_SET);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_14, GPIO_PIN_SET);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_15, GPIO_PIN_SET);
}
```

Exercise 8: Implement a function named **setNumberOnClock(int num)**. The input for this function is from **0 to 11** and an appropriate LED is turn on. Present the source code of this function.

Source code:

```
//Hàm làm sáng đèn LED tại các vị trí được chỉ định
void setNumberOnClock(int num){
    // Kiểm tra giá trị đầu vào num bằng switch
    // Tương ứng với từng giá trị num, set 0 cho chân nối với chân âm của LED
    // giá trị âm để đèn sáng
    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;
    }
}
```

```

        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;
    }
}

```

Exercise 9: Implement a function named **clearNumberOnClock(int num)**. The input for this function is from **0 to 11** and an appropriate LED is turn off.

Source code:

```

//Hàm làm tắt LED tại các vị trí được chỉ định
void clearNumberOnClock(int num){
    // Kiểm tra giá trị đầu vào num bằng switch
    // Tương ứng với từng giá trị num, set 1 cho chân nối với chân âm của LED
    // giá trị dương để đèn tắt đi
    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;
}
}

```

Exercise 10: Integrate the whole system and use 12 LEDs to display a clock. At a given time, there are only 3 LEDs are turn on for hour, minute and second information.

Source code:

```

//Hàm hiện thị giờ trên 1 đồng hồ bằng 12 LED đơn
//Hour: Giờ (0->11)
//Minute: Do hạn chế về số LED nên set tạm thời 0->4: LED0, 5->9: LED2, ...
//Second: Do hạn chế về số LED nên set tạm thời 0->4: LED0, 5->9: LED2, ...
//timesecond: Thời gian để tăng second lên 1 để tiện kiểm tra
void setHour(int hour, int minute, int second, int timesecond){
    minute = minute/5;
    second = second/5;
    timesecond = timesecond / 3;
    //Hour
    setNumberOnClock(hour);
    HAL_Delay(timesecond);
}

```

```
clearNumberOnClock(hour);  
//Minute  
setNumberOnClock(minute);  
HAL_Delay(timesecond);  
clearNumberOnClock(minute);  
//Second  
setNumberOnClock(second);  
HAL_Delay(timesecond);  
clearNumberOnClock(second);  
}
```

```
//Kiểm tra thử đồng hồ 12 LED  
int hour = 0, minute = 0, second = 0;  
clearAllClock();  
while (1)  
{  
    /* USER CODE END WHILE */  
    setHour(hour, minute, second, 1000);  
    second++;  
    if(second > 59){  
        second = 0;  
        minute++;  
    }  
    if(minute > 59) {  
        minute = 0;  
        hour++;  
    }  
    if(hour > 11) hour = 0;  
    /* USER CODE BEGIN 3 */  
}
```