

Busan Software Meister High School

MICROPROCESSOR

2309 양유빈

20230511 마이크로프로세서

OVERVIEW

- GPIO 관련 HAL 라이브러리 1~4
- 실습보조장치_JKIT-Nucleo-64
- LED 4개 동시 점멸 프로그램
- 앰블러스 표시등 프로그램

GPIO 관련 HAL 라이브러리

GPIO related HAL library

 형태: void HAL_GPIO_WritePin(GPIO_TypeDef *GPIOx, uint16_t GPIO_Pin, GPIO_PinState PinState)

설명	GPIOx의 지정된 핀(GPIO_Pin)을 0 또는 1로 설정
파라미터	<ul style="list-style-type: none">• GPIOx : GPIO의 이름(STM32-F103RB의 경우 x는 A~D까지의 값을 가질 수 있음)• GPIO_Pin : GPIO pin을 지정하며, GPIO_PIN_0 ~ GPIO_PIN_15 사이의 값을 가짐• PinState : 핀의 상태 설정<ul style="list-style-type: none">· GPIO_PIN_RESET : Pin을 Reset(0)· GPIO_PIN_SET : Pin을 Set(1)
리턴값	없음
예시	HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, GPIO_PIN_SET);

GPIO 관련 HAL 라이브러리-2

GPIO related HAL library-2

 형태: void HAL_Delay(uint32_t Delay)

설명	지정된 시간만큼 지연(단위: ms)
파라미터	Delay : 지연하고자 하는 시간
리턴값	없음
예시	HAL_Delay(500); // 0.5초 지연

GPIO 관련 HAL 라이브러리-3


GPIO related HAL library-3

 형태: void HAL_GPIO_TogglePin(GPIO_TypeDef *GPIOx, uint16_t GPIO_Pin)

설명	GPIOx의 지정된 핀(GPIO_Pin)을 토글(toggle)
파라미터	<ul style="list-style-type: none">• GPIOx : GPIO의 이름(STM32-F103RB의 경우 x는 A~D까지의 값을 가질 수 있음)• GPIO_Pin : GPIO pin을 지정하며, GPIO_PIN_0 ~ GPIO_PIN_15 사이의 값을 가짐
리턴값	없음
예시	HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_5);

GPIO 관련 HAL 라이브러리-4

GPIO related HAL library-4

 형태: GPIO_PinState HAL_GPIO_ReadPin(GPIO_TypeDef *GPIOx, uint16_t GPIO_Pin)

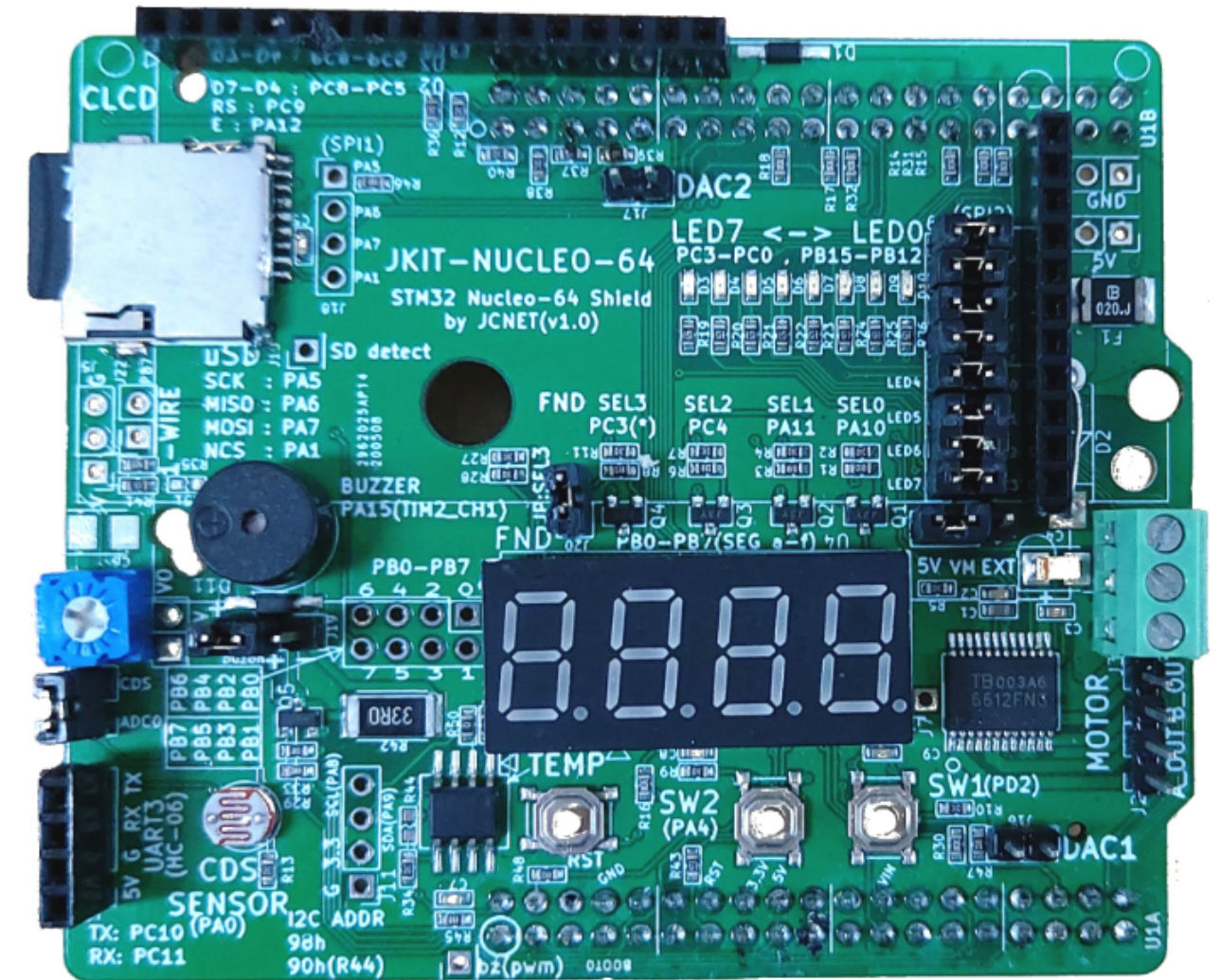
설명	GPIOx의 지정된 핀(GPIO_Pin)에 입력된 값을 읽어옴
파라미터	<ul style="list-style-type: none">GPIOx : GPIO의 이름(STM32-F103RB의 경우 x는 A~D까지의 값을 가질 수 있음)GPIO_Pin : GPIO pin을 지정하며, GPIO_PIN_0 ~ GPIO_PIN_15 사이의 값을 가짐
리턴값	현재 핀으로 입력되는 신호 값 (GPIO_PIN_RESET, GPIO_PIN_SET)
예시	HAL_GPIO_TogglePin(GPIOC, GPIO_PIN_13);

실습보조장치_JKIT-Nucleo-64

Training aid_JKIT-Nucleo-64

JKIT-Nucleo-64 기능 및 규격

- STM32 마이크로컨트롤러 실습 보조 장치
- Nucleo-F103RB와 같은 Nucleo-64 보드와 결합하여 사용
- LED, FND, 스위치, 버저, 온도센서, 광센서, 모터, SD카드 실습 가능
- USART, SPI, I2C 통신 실습 가능 및 커넥터 제공
- CLCD, TFT-LCD 실습 가능(장착용 커넥터 제공)
- GPIO 포트, ADC 포트, DAC 포트, WS2812B 용 커넥터 제공
- Nucleo-64에서 제공하는 전원 사용



LED 4개 동시 점멸 프로그램

Simultaneous flashing program of 4 LEDs

```
1 // LED 8개 중 가운데 4개를 1초에 한 번씩 점멸하도록 프로그래밍.(0.5초 동안은 LED가 ON, 0.5초 동안 LED가 OFF를 무한 반복)
2 // 중간 4개(PC1, PC0, PB15, PB14) LED ON
3 // 0.5초 지연
4 // 중간 4개의 LED를 토글(toggle)
5 // 위의 과정 반복
6 #include "main.h"
7 int main(void) {
8     /* USER CODE BEGIN WHILE */
9     while (1) {
10         HAL_GPIO_WritePin(GPIOC, GPIO_PIN_1, GPIO_PIN_SET);
11         HAL_GPIO_WritePin(GPIOC, GPIO_PIN_0, GPIO_PIN_SET);
12         HAL_GPIO_WritePin(GPIOB, GPIO_PIN_15, GPIO_PIN_SET);
13         HAL_GPIO_WritePin(GPIOB, GPIO_PIN_14, GPIO_PIN_SET);
14         HAL_Delay(500);
15         HAL_GPIO_WritePin(GPIOC, GPIO_PIN_1, GPIO_PIN_RESET);
16         HAL_GPIO_WritePin(GPIOC, GPIO_PIN_0, GPIO_PIN_RESET);
17         HAL_GPIO_WritePin(GPIOB, GPIO_PIN_15, GPIO_PIN_RESET);
18         HAL_GPIO_WritePin(GPIOB, GPIO_PIN_14, GPIO_PIN_RESET);
19         HAL_Delay(500);
20     /* USER CODE END WHILE */
21
22     /* USER CODE BEGIN WHILE */
23     while (1) {
24         HAL_GPIO_WritePin(GPIOC, GPIO_PIN_0 | GPIO_PIN_1, GPIO_PIN_SET);
25         HAL_GPIO_WritePin(GPIOB, GPIO_PIN_15 | GPIO_PIN_14, GPIO_PIN_SET);
26         HAL_Delay(500);
27         HAL_GPIO_WritePin(GPIOC, GPIO_PIN_0 | GPIO_PIN_1, GPIO_PIN_RESET);
28         HAL_GPIO_WritePin(GPIOB, GPIO_PIN_15 | GPIO_PIN_14, GPIO_PIN_RESET);
29         HAL_Delay(500);
30     /* USER CODE END WHILE */
31
32     /* USER CODE BEGIN 2 */
33     HAL_GPIO_WritePin(GPIOC, GPIO_PIN_0 | GPIO_PIN_1, GPIO_PIN_RESET);
34     HAL_GPIO_WritePin(GPIOB, GPIO_PIN_15 | GPIO_PIN_14, GPIO_PIN_RESET);
35     /* USER CODE END 2 */
36
37     /* USER CODE BEGIN WHILE */
38     while (1) {
39         HAL_GPIO_TogglePin(GPIOC, GPIO_PIN_0 | GPIO_PIN_1);
40         HAL_GPIO_TogglePin(GPIOB, GPIO_PIN_15 | GPIO_PIN_14);
41         HAL_Delay(500);
42     /* USER CODE END WHILE */
43     }
```


앰블러스 표시등 프로그램

Ambulus light program

```
// 좌측 LED 4개와 우측 LED 4개가 서로 반대로 0.3초 간격으로 ON, OFF 하도록 프로그래밍.
#include "main.h"
int main(void) {
    /* USER CODE BEGIN 2 */
    HAL_GPIO_WritePin(GPIOC, GPIO_PIN_0 | GPIO_PIN_1 | GPIO_PIN_2 | GPIO_PIN_3, GPIO_PIN_SET);
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_15 | GPIO_PIN_14 | GPIO_PIN_13 | GPIO_PIN_12, GPIO_PIN_RESET);
    /* USER CODE END 2 */

    /* USER CODE BEGIN WHILE */
    while (1) {
        HAL_Delay(300);
        HAL_GPIO_TogglePin(GPIOC, GPIO_PIN_0 | GPIO_PIN_1 | GPIO_PIN_2 | GPIO_PIN_3);
        HAL_GPIO_TogglePin(GPIOB, GPIO_PIN_15 | GPIO_PIN_14 | GPIO_PIN_13 | GPIO_PIN_12);
    /* USER CODE END WHILE */
    }
}
```