Busan Software Meister High School

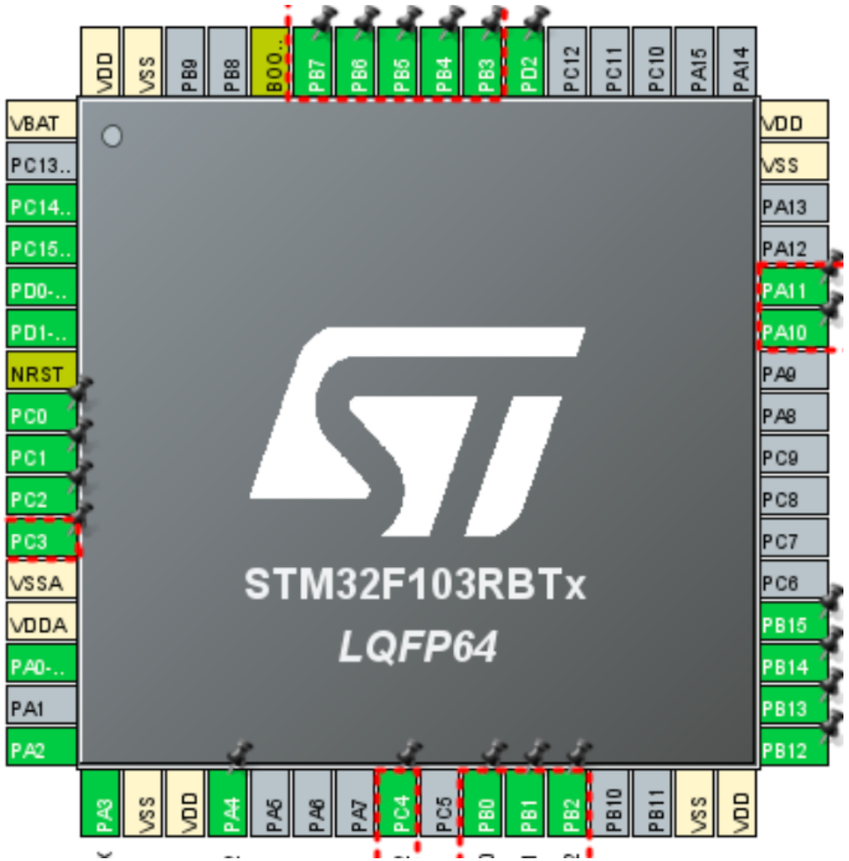
MICROPROCESSOR

2309 양유빈

20231005 마이크로프로세서

FND STM32CubeIDE 설정

FND STM32CubeIDE Settings



- FND 핀 할당
  - PB0 : A
  - PB1 : B
  - PB2 : C
  - PB3 : D
  - PB4 : E
  - PB5 : F
  - PB6 : G
  - PB7 : DP
- 선택 핀(FND_SEL(x)) 할당
  - PA10 : COM0
  - PA11 : COM1
  - PC4 : COM2
  - PC3 : COM3

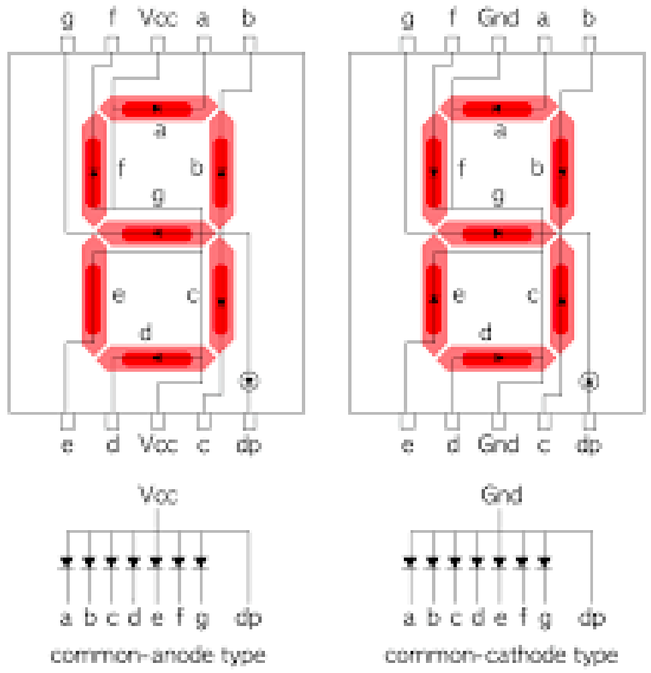
FND 7 표시 코드

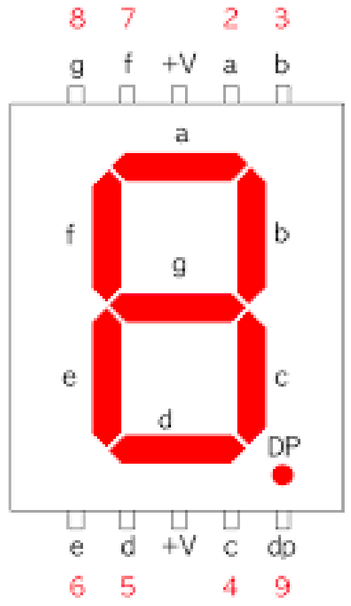
FND 7 display code

```
int main(void) {
/* USER CODE BEGIN WHILE */
  while(1)
  {
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_0, GPIO_PIN_SET); // A
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_1, GPIO_PIN_SET); // B
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_2, GPIO_PIN_SET); // C
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_10, GPIO_PIN_SET); // COM 신호
    /* USER CODE END WHILE */
    /* USER CODE BEGIN 3 */
  }
  /* USER CODE END 3 */
}
```

숫자에 따른 FND 값

FND value according to number





7세그먼트 핀과 오렌지보드 핀(붉은색) 연결

16 진수	16진수 표현 7-세그먼트의 비트값								데이터 값 (HEX)
	H	G	F	E	D	C	B	A	
0	0	0	1	1	1	1	1	1	0X3F
1	0	0	0	0	0	1	1	0	0X06
2	0	1	0	1	1	0	1	1	0X5B
3	0	1	0	0	1	1	1	1	0X4F
4	0	1	1	0	0	1	1	0	0X66
5	0	1	1	0	1	1	0	1	0X6D
6	0	1	1	1	1	1	0	1	0X7D
7	0	0	1	0	0	1	1	1	0X27
8	0	1	1	1	1	1	1	1	0X7F
9	0	1	1	0	1	1	1	1	0X6F
A	0	1	1	1	0	1	1	1	0X77
B	0	1	1	1	1	1	0	0	0X7C
C	0	0	1	1	1	0	0	1	0X39
D	0	1	0	1	1	1	1	0	0X5E
E	0	1	1	1	1	0	0	1	0X79
F	0	1	1	1	0	0	0	1	0X71

FND HELP 표시

Show FND HELP

```
#include "main.h"
/* USER CODE BEGIN PV */
uint8_t help[] = {0x76, 0x79, 0x38, 0x73};
/* USER CODE END PV */
/* USER CODE BEGIN 0 */
typedef struct fnd {
  GPIO_TypeDef *port;
  uint16_t pin;
} FND;
FND value[8] = {
  {GPIOB, GPIO_PIN_0}, {GPIOB, GPIO_PIN_1},
  {GPIOB, GPIO_PIN_2}, {GPIOB, GPIO_PIN_3},
  {GPIOB, GPIO_PIN_4}, {GPIOB, GPIO_PIN_5},
  {GPIOB, GPIO_PIN_6}, {GPIOB, GPIO_PIN_7}
};
FND sel[4] = {
  {GPIOA, GPIO_PIN_10}, {GPIOA, GPIO_PIN_11},
  {GPIOC, GPIO_PIN_4}, {GPIOC, GPIO_PIN_3}
};
```

```
void display_fnd(uint8_t data, uint8_t position, uint32_t time) {
  int i;
  for (i = 0; i < 4; i++) {
    if(i == position)
      HAL_GPIO_WritePin(sel[i].port, sel[i].pin, 1);
    else
      HAL_GPIO_WritePin(sel[i].port, sel[i].pin, 0);
  }
 for(i = 0; i < 8; i++) {
    if((data & (1 << i)) != 0)
      HAL_GPIO_WritePin(value[i].port, value[i].pin, 1);
    else
      HAL_GPIO_WritePin(value[i].port, value[i].pin, 0);
  }
  HAL_Delay(time);
}
/* USER CODE BEGIN 0 */
```

```
int main(void) {
/* USER CODE BEGIN WHILE */
  while(1) {
    display_fnd(help[0], 3, 5);
    display_fnd(help[1], 2, 5);
    display_fnd(help[2], 1, 5);
    display_fnd(help[3], 0, 5);
    /* USER CODE END WHILE */
    /* USER CODE BEGIN 3 */
  }
  /* USER CODE END 3 */
}
```