

임베디드시스템 설계 및 실험 보고서 1

조번호: 8조

작성자: 201602181 정진성

조원:

201824523 안혜준

201924660 한병정

202055516 김명서

201824483 박진영

1. 실험 목표

실험 목적은 하드웨어의 기본적인 이해도 향상과 그에 따른 ARM Cortex-M3 기반 임베디드 시스템 설계 기술을 습득이다.

그에 따라 이번 회차는 개발 환경을 구축 및 IAR Embedded Worknbench에서의 프로젝트 생성 및 설정을 우선한다. 이후 주어진 데이터 시트에 따라 Reference Manual을 참조해 실험 기기에 대한 이해(레지스터 및 주소에 대한 설정 이해)를 목적으로 한다.

실질적인 실험 내용은 GPIO(general-purpose input/output)을 사용한 LED의 제어이다. 조이스틱의 방향에 따라 점화되고 반전되는 LED 제어를 구현을 목표로 한다.

2. 실험 과정

STM32 보드의 모든 외부 장치는 port, pin을 통해 MCU와 연결되어있다. 입출력을 하려면 원하는 port나 pin에 우선적으로 clock을 부여한 후, 포트 설정을 하여 입출력이 가능하다.

우선 사용하려는 핀과 포트에 clock을 부여하고, 각 핀/포트의 레지스터에 접근하여야 한다. 이후 동작은 reference에 따라 진행한다.

레지스터에 접근할 때에는 volatile선언된 변수를 사용한다. 이 변수를 사용하면 컴파일러가 최적화를 수행하지 않는다.

제어되어야 할 부분과 동작은 아래와 같다.

LED

Up : PD4, PD7 LED On

Down : PD4, PD7 LED Off

Left : PD2, PD3 LED On

Right : PD2, PD3 LED Off

Select : PB8, All LED Toggle

방향키

PB8 : selection

PC2, PC3, PC4, PC5 : 방향

1. 주변 장치에 clock enable

1-1 RCC 레지스터 주소 획득

Figure 5. Memory map

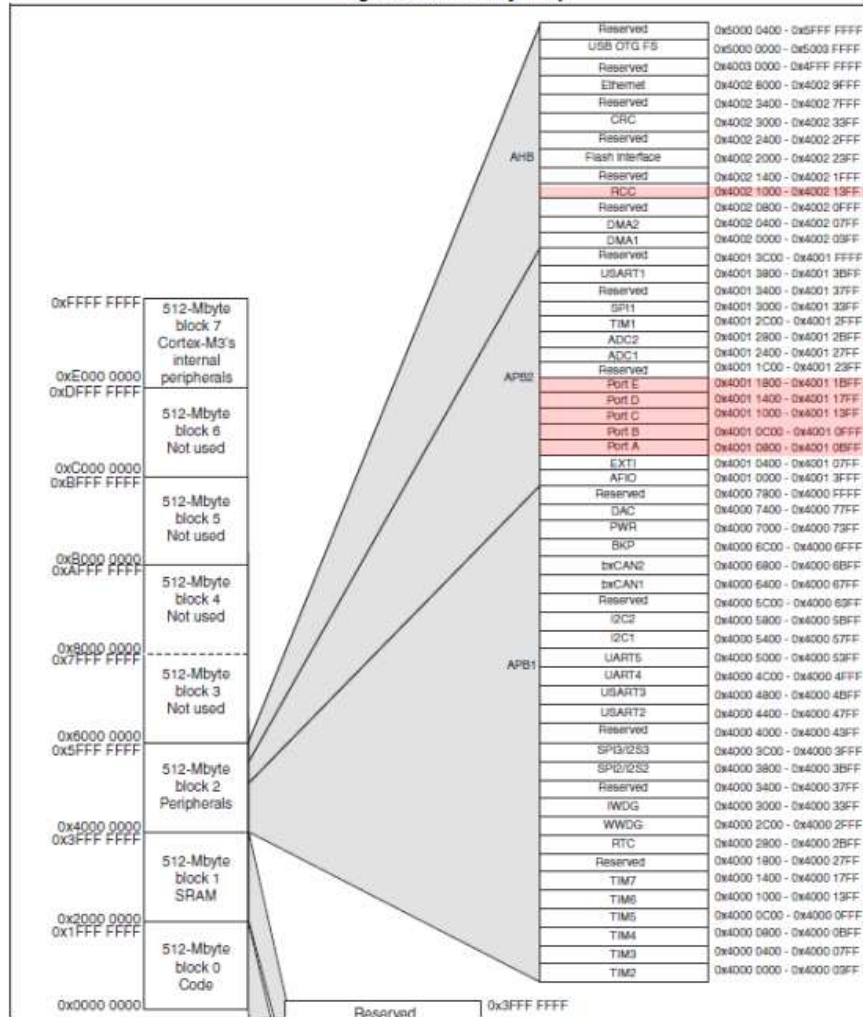


그림 5

우선 포트를 사용하기 위해 클럭을 인가할 필요가 있다. 이를 위해 RCC_APB2ENR 레지스터에 접근한다. 그림 5를 보면 RCC에 해당하는 레지스터 값은 0x4002 1000으로 시작하는 것을 알 수 있다.

7.3.7 APB2 peripheral clock enable register (RCC_APB2ENR)

Address: 0x18

Reset value: 0x0000 0000

Access: word, half-word and byte access

No wait states, except if the access occurs while an access to a peripheral in the APB2 domain is on going. In this case, wait states are inserted until the access to APB2 peripheral is finished.

그림 7.3.7

여기서 RCC_APB2ENR에 해당하는 오프셋인(그림 7.3.7참조) 0x18을 더하면 해당 레지스터의 주소를 얻을 수 있다.

■ RCC_APB2ENR (volatile unsigned int *)0x40021018

1-2 포트 활성화

Reserved										TIM11 EN	TIM10 EN	TIM9 EN	Reserved		
										RW	RW	RW			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADC3 EN	USART 1EN	TIM8 EN	SPI1 EN	TIM1 EN	ADC2 EN	ADC1 EN	IOPG EN	IOPF EN	IOPE EN	IOPD EN	IOPC EN	IOPB EN	IOPA EN	Res.	AFIO EN
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW		RW

이후 포트를 활성화 하기 위해 위 표에 따른 OR연산을 진행한다. 활성화 되어야 할 것은 GPIO D, GPIO C, GPIO B이다. 즉, 번호로 5, 4, 3을 enable하여야한다.

각 레지스터 번호의 값에 비트연산을 하면 아래와 같이 비트 연산을 진행한다.

- *0x40021018 |= 0x20; // GPIO D 활성화
- *0x40021018 |= 0x10; // GPIO C 활성화
- *0x40021018 |= 0x8; // GPIO B 활성화

각각의 16진수는 아래와 같이 2진수에 대응됨을 유의한다.

0x20 = 0010 0000

0x10 = 0001 0000

0x8 = 0000 1000

2. 포트 레지스터 주소 할당

2-1. 매핑된 메모리 주소 확인

위의 그림 5를 참조하여 메모리 주소를 확인한다. Port D에 대한 매핑 주소는 0x4001 1400임을 확인할 수 있다.

또한 Port C와 B에 대한 매핑 주소는 각각 0x4001 1000, 0x4001 0C00임을 역시 확인 가능하다. 해당 값들은 volatile로 변수에 할당한다.

2-2. offset한 주소 확인 및 레지스터 설정

클릭 인가 이후 이번 실험에서 사용할 레지스터와 설명은 아래와 같다.

GPIOx_CRL // RTC control register low

- GPIOC_CRL
- GPIOB_CRL
- GPIOD_CRL

GPIOx_BSRR // Port bit set/reset register

- GPIOD_BSRR

GPIOx_ODR //Port output data register

- GPIOD_ODR

GPIOx_IDR // Port input data register

- GPIOC_IDR
- GPIOB_IDR

이 중 예를 들어, GPIOD로 시작하는 레지스터는 PortD에 매핑된 주소를 기준으로 오프셋을 확인하여 설정이 가능하다.

오프셋에 대한 설정값은 아래 그림 9.5와 같다.

9.5 GPIO and AFIO register maps

Refer to [Table 3 on page 51](#) for the register boundary addresses. The following tables give the GPIO and AFIO register map and the reset values.

Table 59. GPIO register map and reset values

Offset	Register	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	GPIOx_CRL	CNF7 [1:0]	MODE7 [1:0]			CNF6 [1:0]	MODE6 [1:0]			CNF5 [1:0]	MODE5 [1:0]			CNF4 [1:0]	MODE4 [1:0]			CNF3 [1:0]	MODE3 [1:0]			CNF2 [1:0]	MODE2 [1:0]			CNF1 [1:0]	MODE1 [1:0]			CNF0 [1:0]	MODE0 [1:0]		
	Reset value	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0
0x04	GPIOx_CRH	CNF15 [1:0]	MODE15 [1:0]			CNF14 [1:0]	MODE14 [1:0]			CNF13 [1:0]	MODE13 [1:0]			CNF12 [1:0]	MODE12 [1:0]			CNF11 [1:0]	MODE11 [1:0]			CNF10 [1:0]	MODE10 [1:0]			CNF9 [1:0]	MODE9 [1:0]			CNF8 [1:0]	MODE8 [1:0]		
	Reset value	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0
0x08	GPIOx_IDR	Reserved																IDRy															
	Reset value																	0															
0x0C	GPIOx_ODR	Reserved																ODRy															
	Reset value																	0															
0x10	GPIOx_BSRR	BR[15:0]																BSR[15:0]															
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0x14	GPIOx_BRR	Reserved																BR[15:0]															
	Reset value																	0															
0x18	GPIOx_LCKR	Reserved																LCKR	LCK[15:0]														
	Reset value																		0														

그림 9.5

이러한 GPIOB_BSRR는 매핑 주소인 0x4001 1400에 GPIOx_BSRR의 오프셋 0x10를 더하여 0x4001 1410이 된다. 이 방식을 이용하여 할당된 각 레지스터의 주소는 아래와 같다.

- GPIOB_CRL (volatile unsigned *)0x40011400
- GPIOB_BSRR (volatile unsigned *)0x40011410
- GPIOB_ODR (volatile unsigned *)0x4001140C
- GPIOC_CRL (volatile unsigned *)0x40011000
- GPIOC_IDR (volatile unsigned *)0x40011008
- GPIOB_CRL (volatile unsigned *)0x40010C00
- GPIOB_IDR (volatile unsigned *)0x40010C08

3. 기능 구현

앞서 접근한 레지스터의 주소로 각 기능을 구현한다. 각 동작에 대한 방법은 아래와 같다.

3-1. 포트 configuration

9.2.1 Port configuration register low (GPIOx_CRL) (x=A..G)

Address offset: 0x00

Reset value: 0x4444 4444

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CNF7[1:0]		MODE7[1:0]		CNF6[1:0]		MODE6[1:0]		CNF5[1:0]		MODE5[1:0]		CNF4[1:0]		MODE4[1:0]	
r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNF3[1:0]		MODE3[1:0]		CNF2[1:0]		MODE2[1:0]		CNF1[1:0]		MODE1[1:0]		CNF0[1:0]		MODE0[1:0]	
r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

GPIOx_CRL 레지스터의 포트를 설정한다.

예를 들어 GPIOD_CRL의 경우 Port D2, Port D3, Port D4, Port D7를 설정하여야 한다.

래퍼런스를 참조해 GPIOD_CRL 레지스터의 값에 0x30033300을 or연산한다.

이 값을 2진수로 환산하면 0011 0000 0000 0011 0011 0011 0000 0000이므로, MODE2(8, 9), MODE3(12, 13), MODE4(16, 17), MODE7(28, 29)번 인덱스를 활성화한다.

나머지 Port C와 Port B도 같은 과정을 반복한다.

3-2 동작 구현

Up : PD4, PD7 LED On

Down : PD4, PD7 LED Off

Left : PD2, PD3 LED On

Right : PD2, PD3 LED Off

Select : PB8, All LED Toggle

LED를 켜기위해 포트를 set하는 동작을 구현한다. 해당 동작은 GPIOx_BSRR에서 담당한다.

9.2.5 Port bit set/reset register (GPIOx_BSRR) (x=A..G)

Address offset: 0x10

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BS15	BS14	BS13	BS12	BS11	BS10	BS9	BS8	BS7	BS6	BS5	BS4	BS3	BS2	BS1	BS0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

Bits 31:16 **BRy**: Port x Reset bit y (y= 0 .. 15)

These bits are write-only and can be accessed in Word mode only.

0: No action on the corresponding ODRx bit

1: Reset the corresponding ODRx bit

Note: If both BSx and BRx are set, BSx has priority.

Bits 15:0 **BSy**: Port x Set bit y (y= 0 .. 15)

These bits are write-only and can be accessed in Word mode only.

0: No action on the corresponding ODRx bit

1: Set the corresponding ODRx bit

Set:

PD2와 PD3을 set하는 동작에 대한 비트: 0000 1100 = 0xC

PD4와 PD7을 set하는 동작에 대한 비트: 1001 0000 = 0x90

Reset:

PD2와 PD3을 reset하는 동작에 대한 비트: 1100 0000 0000 0000 0000 = 0xC0000

PD4와 PD7을 reset하는 동작에 대한 비트: 1001 0000 0000 0000 0000 0000 = 0x900000

각 레지스터의 값에 or연산을 하여 각 Port를 set하면 해당 LED가 켜지고 꺼지는 동작을 수행하도록 코드를 작성한다.

다음으로 방향키 입력을 확인하기 위하여 GPIOx_IDR의 데이터 그림 9.2.3을 통해 확인한다.

9.2.3 Port input data register (GPIOx_IDR) (x=A..G)

Address offset: 0x08h

Reset value: 0x0000 XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IDR15	IDR14	IDR13	IDR12	IDR11	IDR10	IDR9	IDR8	IDR7	IDR6	IDR5	IDR4	IDR3	IDR2	IDR1	IDR0
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

Bits 31:16 Reserved, must be kept at reset value.

Bits 15:0 **IDRy**: Port input data (y= 0 .. 15)

These bits are read only and can be accessed in Word mode only. They contain the input value of the corresponding I/O port.

그림 9.2.3

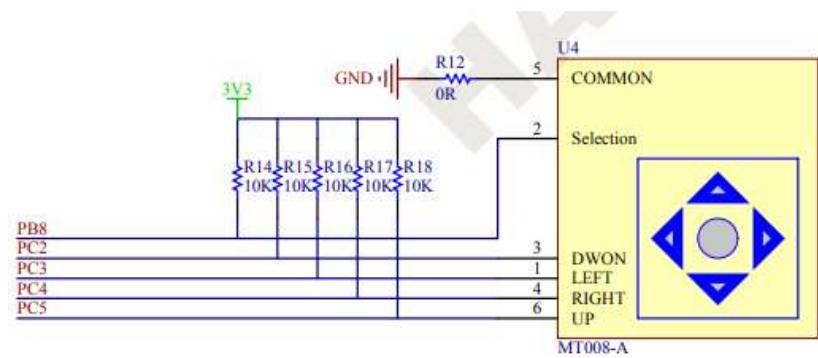
방향키의 각 방향에 맞는 포트 번호를 아래의, 주어진 스키마를 통해 확인한다.

PB8 : selection

PC2, PC3, PC4, PC5 : 방향

임을 확인할 수 있다. 아래의 그림 9.2.3을 확인하여 해당 주소에 값이 있는지 여부를 판단하여 주어진 문제에 대한 코드를 작성한다.

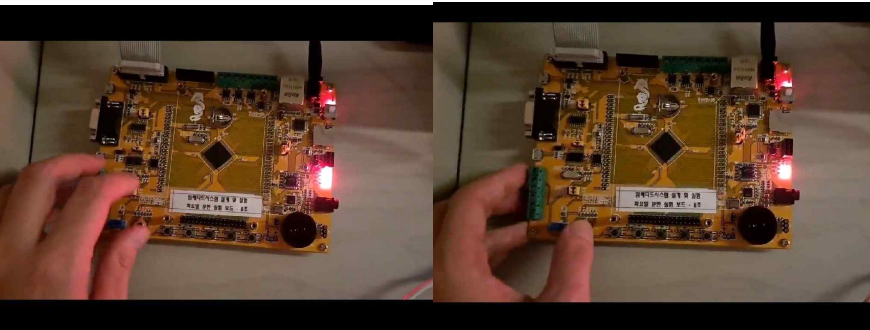
그림 9.2.3

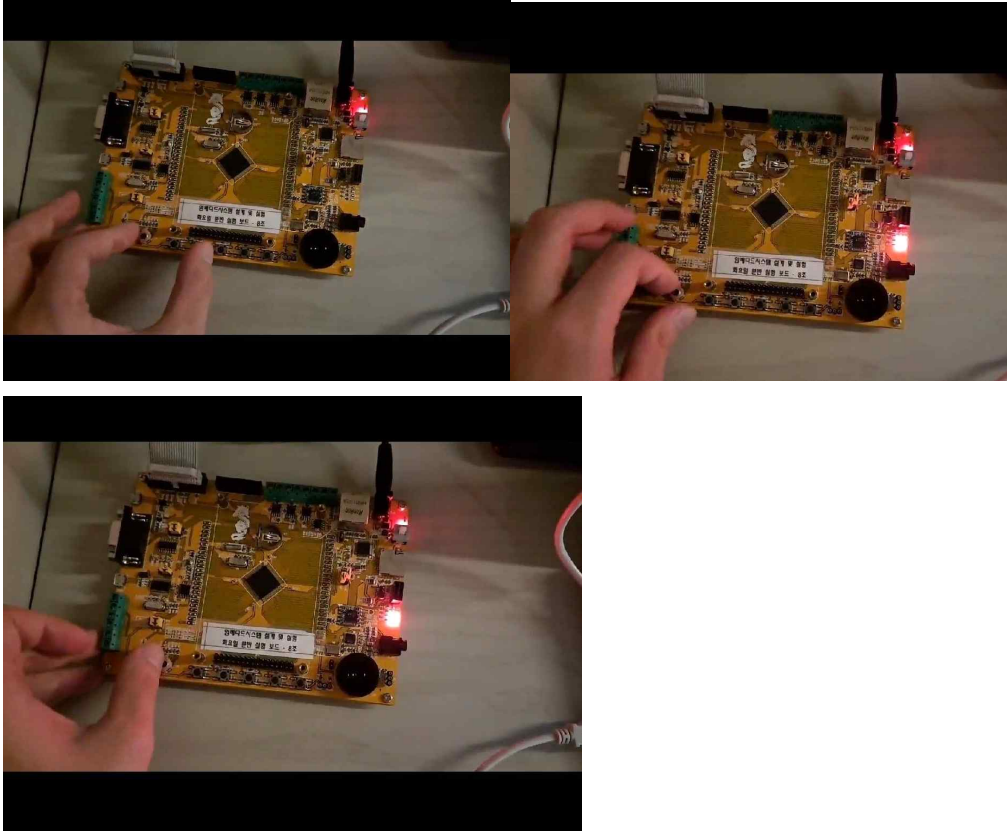


3. 실험 결과

실험 과정에서 주어진 요소들을 종합하여 코드로 구현하였다.

아래 사진에서 보듯 기기가 의도대로 작동함을 확인할 수 있다.





해당 작동에 대한 동영상 링크 : <https://photos.app.goo.gl/kPBmcq18sVXJWGVJ9>