

실험 결과 보고서

: 6 주차 Scatter 파일 및 릴레이 모듈

실험날짜: 2024-10-08

실험자: 3 조(강태진, 김기윤, 김도완, 임성표)

목차

1. 실험 목적.....	3
2. 세부 목표.....	3
3. 실험 과정.....	3
4. 실험 결과.....	8
5. 참고 문헌.....	9

1. 실험 목적

- 스캐터 파일의 이해 및 플래시 프로그래밍
- 릴레이 모듈의 이해 및 임베디드 펌웨어를 통한 동작
- 폴링 방식의 이해

2. 세부 목표

- Datasheet 및 Reference Manual 을 참고하여 해당 레지스터 및 주소에 대한 설 정 이해
- 스캐터 파일을 통해 플래시 메모리에 프로그램 다운로드
- 플래시 메모리에 올려진 프로그램 정상적인 동작 확인

3. 실험 과정

- 1) 스캐터 파일(.icf)을 통해 원하는 메모리 위치에 프로그램 다운로드

```
/*-Memory Regions-*/
define symbol __ICFEDIT_region_ROM_start__ = 0x08000000;
define symbol __ICFEDIT_region_ROM_end__   = 0x08080000;
define symbol __ICFEDIT_region_RAM_start__  = 0x20000000;
define symbol __ICFEDIT_region_RAM_end__    = 0x20008000;
```

그림 1. Icf 파일의 메모리 영역 할당 코드

기존 RAM 의 주소 범위(0x2000 0000~ 0x2000 FFFF)에서 0x8000 의 메모리 영역을 할당하기 위해 범위를 0x2000 0000 ~ 0x2000 8000 로 조정한다.

기존 ROM 메모리 영역(0x0800 0000 ~ 0x0803 FFFF)에서 0x80000 의 메모리 영역을 할당하기 위해 범위를 0x0800 0000 ~ 0x0808 0000 로 조정한다.

ROM 의 경우 기존의 256KB 보다 512KB 로 확장해야 하기 때문에, 스캐터 파일에서 메모리 끝을 Reserved(예약된) 영역까지 확장한다. 즉, 0x0803 FFFF 이후의 영역까지 사용하도록 설정한다.

2) 릴레이 모듈을 통한 모터 구동

(1) KEY1: 모터 정방향 회전 (전진)

```
//key1 push -> 모터 정방향 회전
if (!(GPIOC_IDR & 0x10)) { // KEY1 버튼 누름 (PC4)
    GPIOC_BSRR |= 0x0100; // C8 핀 set (HIGH)
    GPIOC_BSRR |= 0x02000000; // C9 핀 reset (LOW)
}
```

그림 2. KEY1 버튼을 눌렀을때의 동작 코드

GPIOC_IDR & 0x10: key1 버튼 상태 확인, 버튼을 누르면 GPIOC_IDR & 0x10 는 0 이 되고 모터 정방향 회전을 위해 c8 핀을 set 하고 c9 핀을 reset

(2) KEY2: 모터 역방향 회전 (후진)

```
//key2 push -> 모터 역방향 회전
if (!(GPIOB_IDR & 0x0400)) { // KEY2 버튼 누름 (PB10)
    GPIOC_BSRR |= 0x01000080; // C8 핀 reset (LOW)
    GPIOC_BSRR |= 0x0200; // C9 핀 set (HIGH)
}
```

그림 3. KEY2 버튼을 눌렀을때의 동작 코드

GPIOC_IDR & 0x400: ke2 버튼 상태 확인, 버튼을 누르면 조건이 참이 되고 c8 핀 reset, c9 핀을 set 하여 역방향 회전

(3) KEY3: 모터 정방향 2 초 역방향 2 초 회전 후 정지

```
//key3 push -> 정방향2초, 역방향 2초 회전 후 정지
if (!(GPIOC_IDR & 0x2000)) { // KEY3 버튼 누름 (PC13)

    // 모터 정지
    GPIOC_BSRR |= 0x01000000; // C8 핀 reset (LOW)
    GPIOC_BSRR |= 0x02000000; // C9 핀 reset (LOW)

    // 정방향 회전
    GPIOC_BSRR |= 0x0100; // C8 핀 set (HIGH)
    GPIOC_BSRR |= 0x02000000; // C9 핀 reset (LOW)
    delay(); // 대기

    // 역방향 회전
    GPIOC_BSRR |= 0x01000080; // C8 핀 reset (LOW)
    GPIOC_BSRR |= 0x0200; // C9 핀 set (HIGH)
    delay(); // 대기

    // 모터 정지
    GPIOC_BSRR |= 0x01000000; // C8 핀 reset (LOW)
    GPIOC_BSRR |= 0x02000000; // C9 핀 reset (LOW)
}
```

그림 4. KEY3 버튼을 눌렀을때의 동작 코드

KEY3 버튼을 눌렀을 때, 모터를 정방향 작동 후 delay() 로 인한 지연 후 다시 역방향 회전, 지연 후 모터 정지

(4) KEY4: 모터 정지

```
//key4 push -> 모터 정지
if (!(GPIOA_IDR & 0x01)) { // KEY4 버튼 누름 (PA0)
    GPIOC_BSRR |= 0x01000000; // C8 핀 reset (LOW)
    GPIOC_BSRR |= 0x02000000; // C9 핀 reset (LOW)
}
```

그림 5. KEY4 버튼을 눌렀을때의 동작 코드

3) 회로 및 전선 연결 설정

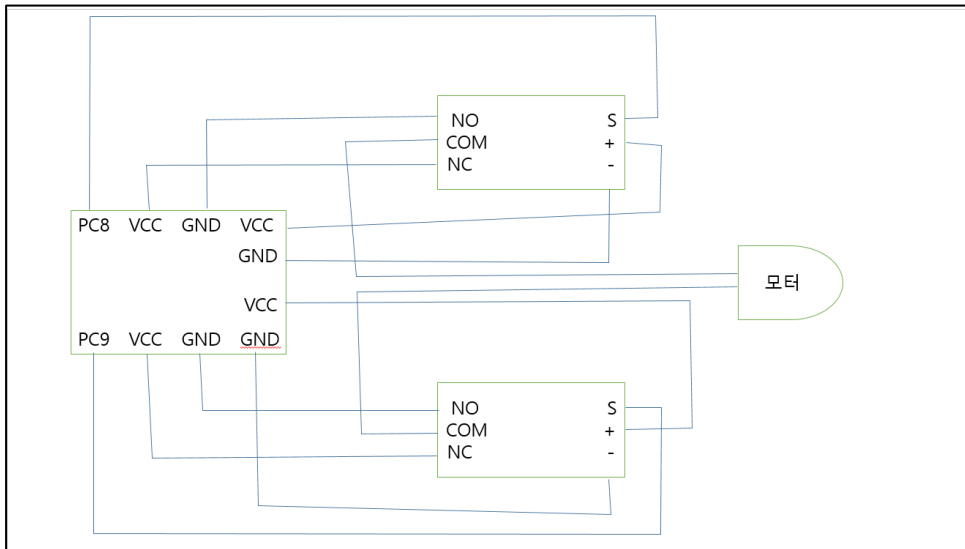


그림 6. 회로도

(1)보드의 Output -> 릴레이 모듈의 Input 에 연결

-보드에서 릴레이 모듈로 제어신호(ON/OFF)를 전달한다.

(2)보드의 VCC -> 릴레이 모듈의 VCC 에 연결

-릴레이 모듈에 작동 전압을 공급한다.

(3)보드의 GND -> 릴레이 모듈의 GND 에 연결

-보드와 릴레이 모듈 간 공통 GND 를 맞춰준다.

(4)NO->GND, NC ->VCC, COM-> 모터에 연결한다.

-릴레이가 OFF -> NC 단자가 연결되어 모터에 VCC 가 공급되지 않는다.

-릴레이가 ON -> NO 단자가 닫히면서 GND 가 연결되고 모터가 동작한다.

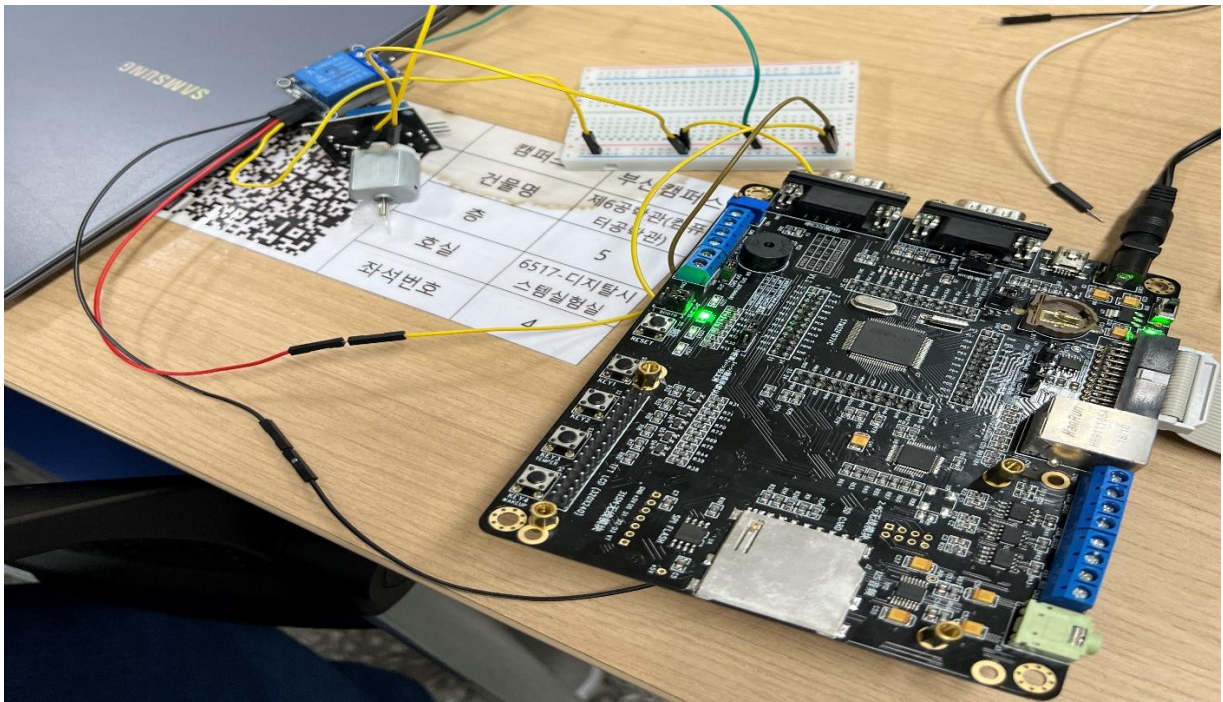


그림 7. 릴레이모듈, 보드, 모터의 회로 연결 사진

4. 실험 결과

(1) 실험 결과 보고

모터가 제대로 작동되지 않아, 코드를 살펴보았지만 문제가 없었다. 보드 교체 후 정상 작동한 것을 보아 보드 문제였던 것을 확인하였다.

KEY1 - 모터 정방향 회전

- KEY1 버튼을 누르면 모터가 정방향으로 회전합니다.
- C8 핀을 set 하고, C9 핀을 reset 하여 모터가 정방향으로 회전합니다.

KEY2 - 모터 역방향 회전

- KEY2 버튼을 누르면 모터가 역방향으로 회전합니다
- C8 핀을 reset 하고, C9 핀을 set 하여 모터가 역방향으로 회전합니다.

KEY3 – 모터 정방향 및 역방향 회전

- KEY3 버튼을 누르면 모터가 정방향으로 2 초 회전한 후, 역방향으로 2 초 동안 회전한 후 정지합니다.
- 정방향 회전 : C8 핀 set, C9 핀 reset
- 역방향 회전 : C8 핀 reset, C9 핀 set
- 정지: C8 핀과 C9 핀 모두 reset
- Delay() 함수를 사용하여 회전시간을 제어

KEY4 - 모터 정지

- KEY4 버튼을 누르면 모터가 정지합니다.
- C8, C9 핀을 모두 reset

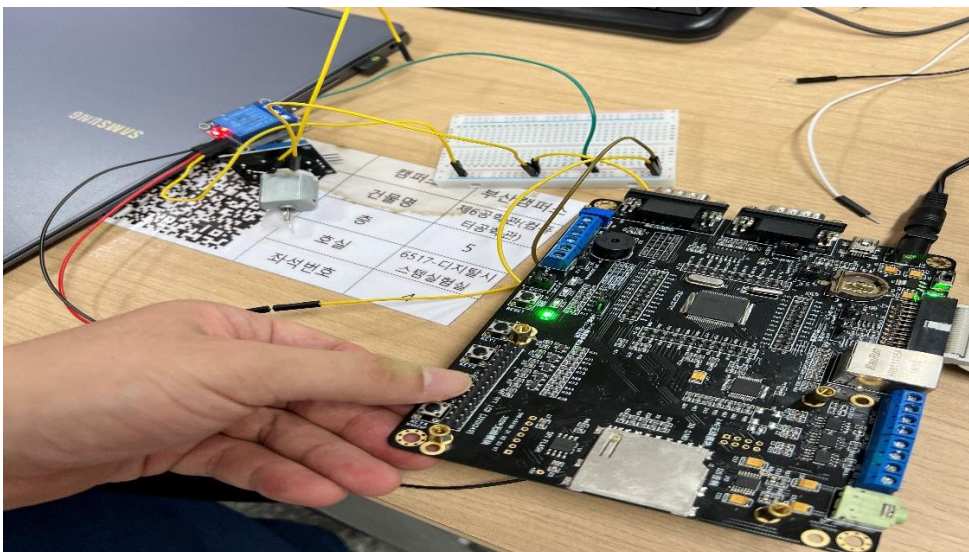


그림 8. KEY3 을 눌렀을 때, 모터의 동작 확인

(2) 폴링 방식의 이해와 문제점

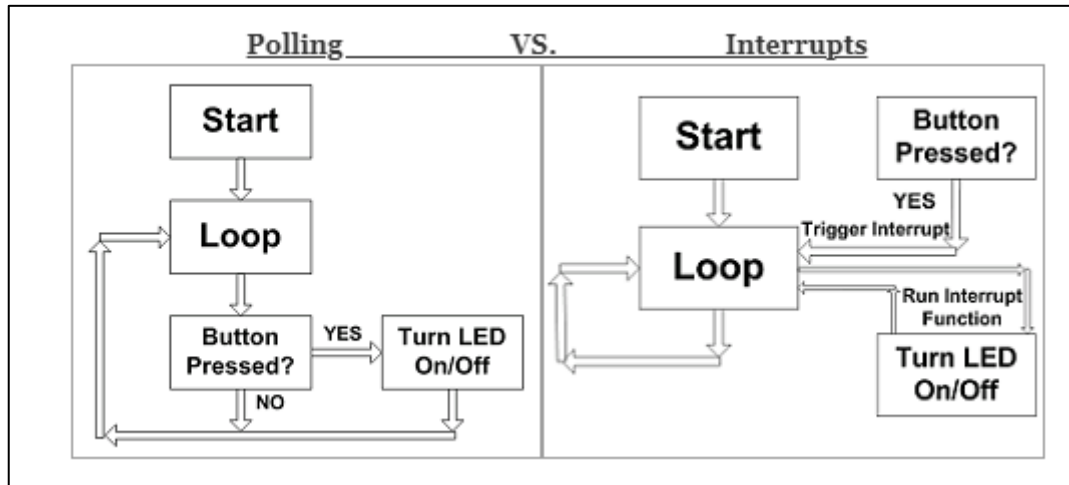


그림 9. 폴링 방식과 인터럽트 방식의 동작 비교

폴링방식은 CPU 가 처리해야할 새로운 데이터가 있는지 주기적으로 장치의 상태를 확인하는 방식이다. 별도의 인터럽트 설정이 필요없으므로 구현이 간단하다.

문제점은 이벤트가 발생할때까지 무의미하게 루프를 돌기 때문에 실시간 처리가 필요한 시스템에서 비효율적이고, 새로운 데이터가 들어오는 시점과 확인하는 시간차에 의해 대기시간이 길어진다.

5. 참고 문헌

stm32_Datasheet.pdf stm32_ReferenceManual.pdf

디바이스 데이터 입출력에 있어서 폴링 방식과

인터럽트 구동방식의 데이터 처리방법-구철회

(한국항공우주학회지, 2005,9)