

파워온 보고서 작성법

1. <https://poweron.co/login/> 에 접속하여 로그인

- ID: **poweron**
- 비밀번호: **Poweron770972**
- (Recommended) 기억하기 체크
- 이미 로그인한 경우 <https://poweron.co/wp-admin/> 으로 접속

2. 왼쪽 툴바에서 글을 클릭하고 새로 추가

- 이미 작성한 글이 있을 경우 모든 글 리스트에서 선택
- 툴바가 안보이면 하단 좌측에 있는 화살표 모양의 버튼 (▶) 클릭

3. (중요) 글 편집창에 진입하면 우선 오른쪽 툴바에서 다음을 선택

- 카테고리 - Project
- 글 속성 - 템플릿 - Project
- 선택하면 보고서 양식이 나타남
- 툴바가 안보이면 상단 우측에 있는 톱니바퀴 모양의 버튼 (⚙) 클릭

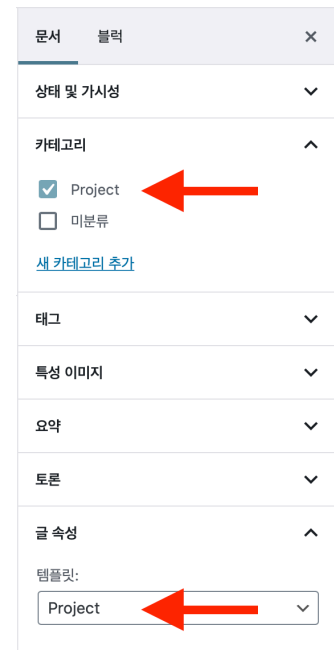
4. 양식에 맞추어 보고서 작성. 첨부한 보고서 예시를 참고할 것.

- 제목 추가
- 작성자 정보 입력
- 개요 입력
- 요약 입력 (오른쪽 툴바에 있는 요약 칸에 입력)
- 본문 입력 (제목 바로 밑의 칸에 입력, 지나친 html 사용 자제)
 - 에디터 사용 방법은 아래 참고
- 참고자료 입력
- 개발환경 입력
- 비공개 옵션 선택 (비공개를 선택할 경우 제목, 작성자, 요약만 공개됨)
- GitHub에 프로젝트를 업로드한 경우 링크 입력
- **임시 글로 저장** 수시로 누르기

5. (Recommended) 임시 글로 저장 이후 추가 옵션 수정

- 제목에 한글이 포함된 경우 오른쪽 툴바에서 **고유 주소**를 영어로 변경
- 나중에 보고서를 찾기 쉽게 오른쪽 툴바에서 **태그**를 추가하기
- 오른쪽 툴바에서 **특성 이미지** 설정하기 (검색 결과나 홈페이지 피쳐 시 사용됨)

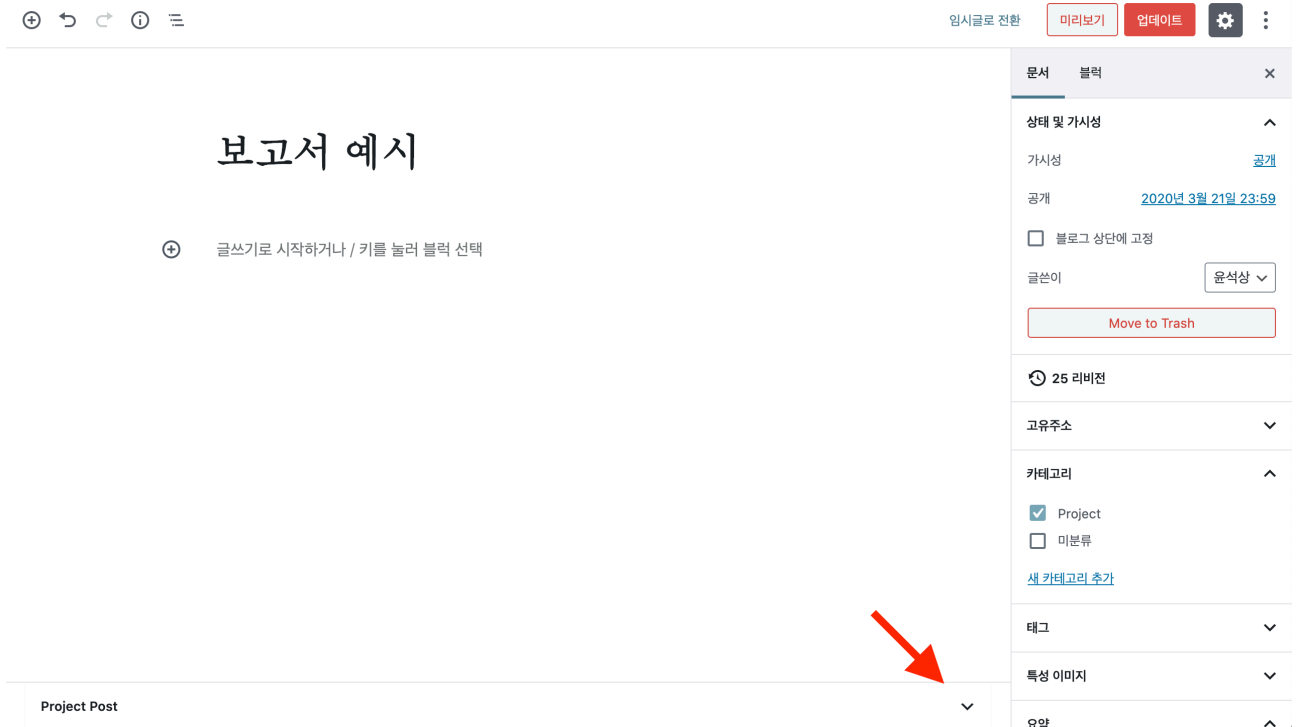
6. 홈페이지 관리자가 보고서를 확인한 후 공개



The image shows a sidebar with various settings. Under '카테고리' (Category), 'Project' is selected with a checkbox. A red arrow points to this selection. Under '글 속성' (Post Properties), '템플릿' (Template) is set to 'Project', also indicated by a red arrow.

본문 작성 & 에디터 사용 방법

우선 Project Post 옆에 v를 클릭하여 불필요한 정보를 숨긴다.



보고서는 특별한 텍스트 스타일 없이 사진과 표, 코드를 위주로 작성하므로 이것 위주로 설명하겠다. 이 에디터는 마크다운(Markdown)을 지원하므로 아는 사람은 사용하면 된다.

+ 표시 옆에 바로 글을 쓰면 기본 문단 (Paragraph) 이 들어간다.

그냥 Enter 키를 누르면 문단을 종료하고 새 문단을 연다. 문단과 문단 사이에는 작은 공간이 생긴다.

문단 내부에서 줄바꿈은 Shift + Enter 키를 누르면 된다.

이미지를 삽입하기 위해서는 / 키를 누르고 이미지를 선택하거나 + 를 누르고 공통 블록-이미지를 선택하면 된다.

이미지를 업로드할 때 캡션을 선택할 수 있으며, 이는 이미지를 삽입한 후에도 수정할 수 있다.

표를 삽입하기 위해서는 / 키를 누르고 테이블을 선택하거나 + 를 누르고 포맷-테이블을 선택하면 된다. (테이블 스타일은 아직 작업중이므로 지금은 우선 내용만 채워놓으면 된다.)

코드를 삽입하기 위해서는 / 키를 누르고 코드를 선택하거나 + 를 누르고 포맷-코드를 선택하면 된다.

■ 제목: 7 segment(FND) 사용법

작성자	
이름	김현성
학과	전자전기공학과
입학 년도	15
최종 수정	2017.02.20

■ 개요

선행 기술	- 없음
목표 기술	- 7 segment 사용 - Multiplexing - 7 segment driver logic IC 사용
연관 기술	- Multiplexing

■ 요약

기초 수준. 7segment 사용법을 다룸. Multiplexing을 이용.

■ 본문

FND라고도 불리는 7segment는 주로 숫자를 표시하기 위한 디스플레이 장치입니다. 7개로 나누어진 막대 모양의 LED를 이용해 0부터 9까지의 숫자를 표시할 수 있습니다..



Figure 1 2-digit FND

7segment의 내부 구조는 다음과 같습니다.

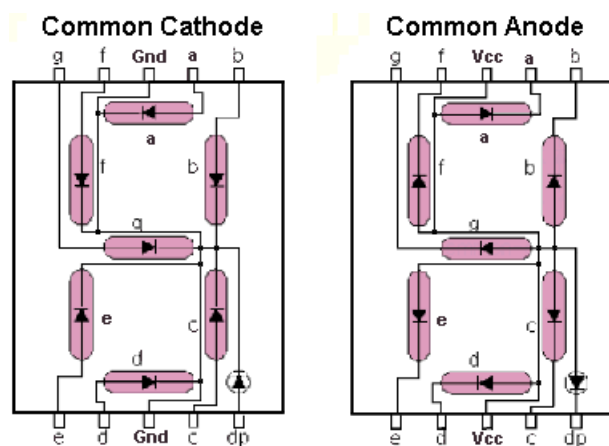


Figure 2 7segment 구조

복잡해 보이지만 원리는 복잡하지 않습니다. LED의 방향에 따라 Anode타입과 Cathode타입으로 구분됩니다. 여기서는 좌측의 Cathode 타입으로 설명하겠습니다. 7 segment는 전류가 흐르는 막대에서 빛이 나오고, 흐르지 않는 곳에서는 빛이 나오지 않습니다. 각 막대에는 시계방향으로 a부터 g까지의 이름이 부여되어있습니다. 만약 숫자 '2'를 표시하고 싶다면 "a, b, d, e, g" 막대에 불이 들어오도록 하면 되겠죠.

Common cathod FND의 데이터시트를 보면 다음과 같은 그림을 찾을 수 있습니다.

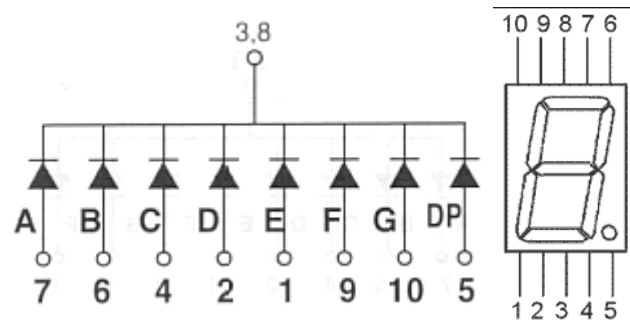


Figure 3 SR-1056K

A막대에 불이 들어오게 하고 싶다면 7번 핀에 HIGH를 입력하고 3번 핀(혹은 8번)을 GND에 연결하면 A막대에 전류가 흘러 불이 들어옵니다.

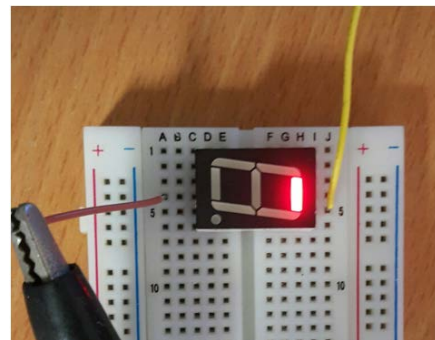
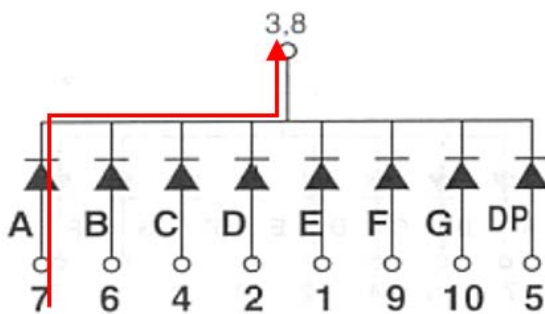


Figure 4 A막대기에서 불이 뽕

그렇다면 2를 표시하고 싶다면 7,6,2,1,10번 핀에 HIGH를 입력하고 3번 핀을 GND에 연결하면 되겠죠.

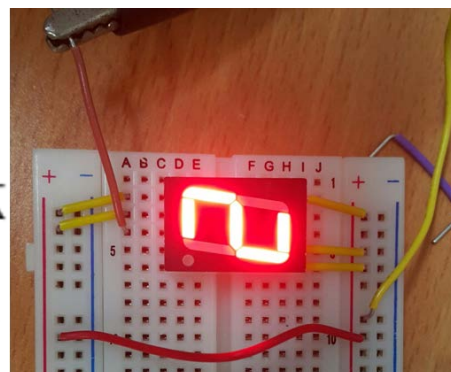
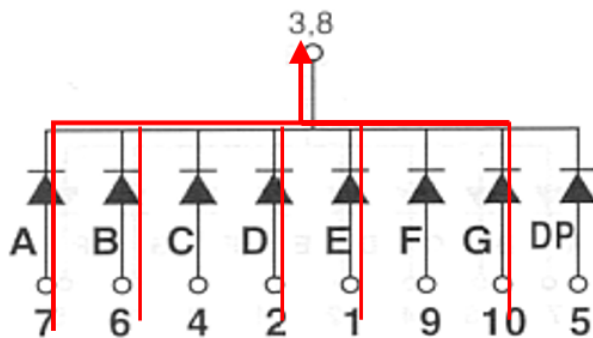


Figure 5 숫자 2가 뽕

이 정도만 안다면 ATmega8과 같은 장치로 숫자를 표시하는 것은 쉽게 할 수 있겠죠?

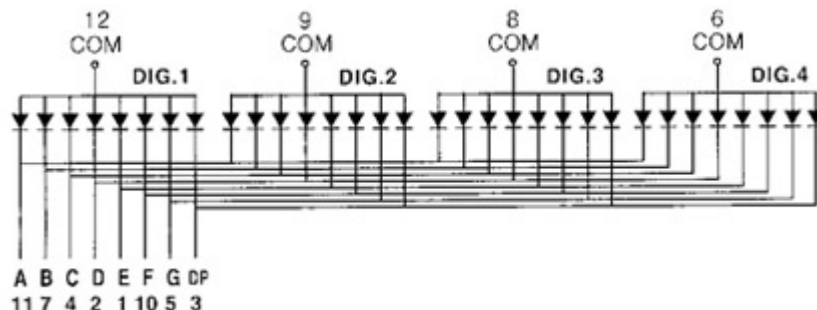
아래의 표는 각 숫자 별로 PORTB 값을 나타낸 표입니다.

AT8 pin	FND pin	0	1	2	3	4	5	6	7	8	9
PB0	A	1	0	1	1	0	1	1	1	1	1
PB1	B	1	1	1	1	1	0	0	1	1	1
PB2	C	1	1	0	1	1	1	1	1	1	1
PB3	D	1	0	1	1	0	1	1	0	1	1
PB4	E	1	0	1	0	0	0	1	0	1	0
PB5	F	1	0	0	0	1	1	1	1	1	1
PB6	G	0	0	1	1	1	1	1	0	1	1
PB7	DP	0	0	0	0	0	0	0	0	0	0
HEX		7E	06	6D	79	66	6D	7D	27	7F	6F

AT8 pin과 FND pin을 표대로 연결하고, 3번 핀을 GND에 연결한 후 PORTB에 HEX값을 넣어주면 됩니다. 아래와 같이 함수를 만들어 사용하면 편하겠죠.

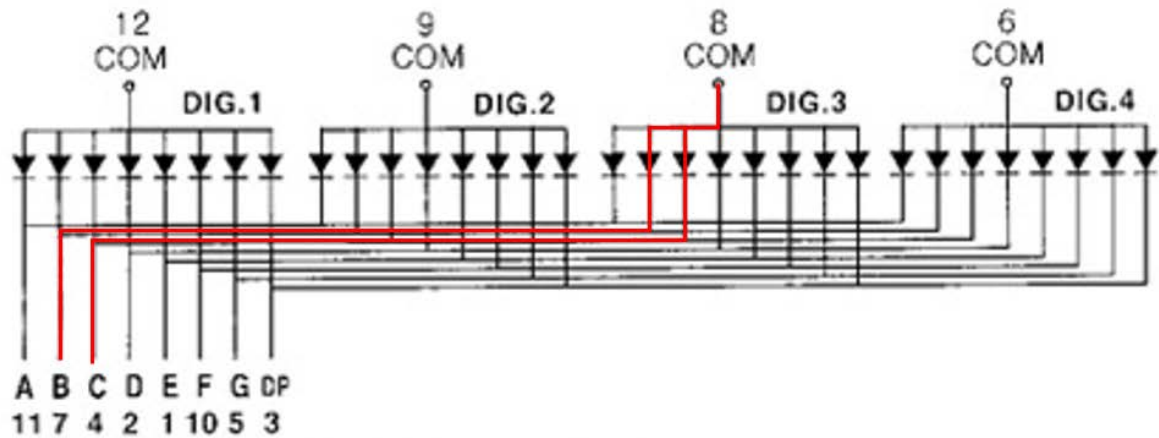
```
void showSegment(int digit)
{
    switch(digit)
    {
        case 0: PORTB = 0x7F; break;
        case 1: PORTB = 0x06; break;
        case 2: PORTB = 0x6D; break;
        case 3: PORTB = 0x79; break;
        case 4: PORTB = 0x66; break;
        case 5: PORTB = 0x6D; break;
        case 6: PORTB = 0x7D; break;
        case 7: PORTB = 0x27; break;
        case 8: PORTB = 0x7F; break;
        case 9: PORTB = 0x6F; break;
    }
}
```

그럼 이제 4 자리 7 segment를 사용해봅시다. Anode 타입의 세그먼트를 사용할게요.

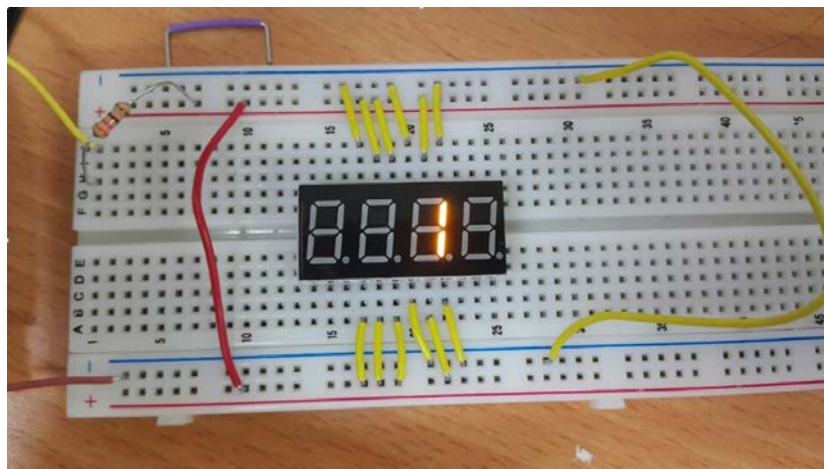


12, 9, 8, 6번 핀은 자리를 결정합니다. 나머지 핀은 한 자리 수일 때와 마찬가지로 어떤 막대에 불을 켤지를 결정합니다. 만약 세번째 자리에 숫자 '1'을 표시하고 싶으면 다

음과 같이 전류가 흘러야 합니다.



일단 세 번째 자리에 표시할 것이므로 8번 핀을 HIGH level에 연결시킵니다. 그리고 숫자를 표시하기 위해 7, 4번 핀을 GND에, 나머지를 HIGH level에 연결시킵니다. 그러면 정방향의 전위차가 있는 [8번→7,4번] 경로로만 전류가 흐르겠죠.



그런데, 네 자리에 모두 다른 숫자를 한번에 표시하는 것은 불가능합니다. 만약 위의 상태에서 9번 핀에 HIGH level을 입력하면 두 번째 자리에는 똑같이 '1'이 출력될 것입니다. 그래서 우리는 'multiplexing'을 이용합니다. 각각의 자리를 동시에 출력하는 것이 아니라, 빠른 속도로 번갈아 가며 출력하는 것입니다. **참고 음짤:** <http://goo.gl/oD4VbJ>

1234를 출력하고자 한다면 다음의 네 과정을 빠르게 반복하는 것입니다. 깜빡이는 속도가 빠르면 사람은 마치 네 자리가 한번에 표시된 것 같이 인지합니다.

- (내용

그럼 아래와 타이머를 이용해 주기적으로 네 자리를 번갈아가면서 출력하도록 하면 되겠죠?

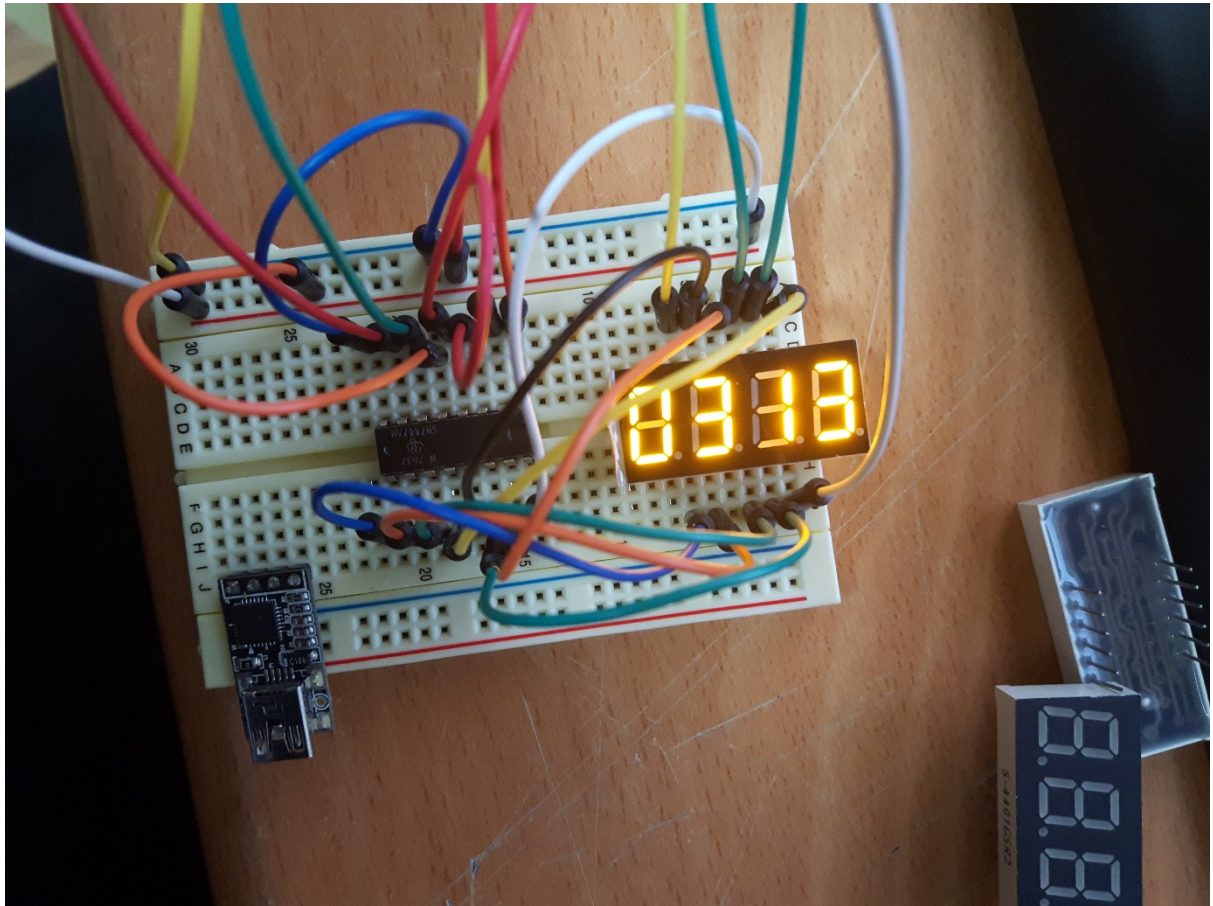
```
int count = 0;
int index = 0;
int value = 2017;

ISR(TIMER0_OVF_vect)
{
    count++;

    if(count == 31)
    {
        count = 0;
        switch(index)
        {
            case 3: showSegment(value/1000); index = 2; break;
            case 2: showSegment(value/100%10); index = 1; break;
            case 1: showSegment(value/10%10); index = 0; break;
            case 0: showSegment(value%10); index = 3; break;
        }
    }
}

void showSegment(int digit)
{
    switch(digit)
    {
        case 0: PORTB = 0x7F; break;
        case 1: PORTB = 0x06; break;
        case 2: PORTB = 0x6D; break;
        case 3: PORTB = 0x79; break;
        case 4: PORTB = 0x66; break;
        case 5: PORTB = 0x6D; break;
        case 6: PORTB = 0x7D; break;
        case 7: PORTB = 0x27; break;
        case 8: PORTB = 0x7F; break;
        case 9: PORTB = 0x6F; break;
    }
}
```


7segment를 사용하면서 느꼈겠지만 상당히 많은 GPIO를 소모한다는 것을 알 수 있습니다. Logic IC를 사용하면 어느정도 핀 수를 줄일 수 있습니다. 7447과 같은 logic IC를 사용하는 것도 좋은 방법입니다. 아래는 사용 예시입니다.



■ 참고자료

본 문서를 작성하는데 참고한 모든 자료를 명시함.

"Program to implement 7-Segment Display with 8051 microcontroller":

<http://blowtech.blogspot.kr/2013/08/program-to-implement-7-segment-display.html>

■ 개발환경

- Microsoft Windows7
- AVR Studio 7.0
- avr 다운로드: AVRISP mkII
- Microsoft Word 2013