

- 날짜 : 2013년 12월 19일
- 제목 : PIC16F886 에서 인터럽트 우선순위가 있는경우의 부작용 최소화 기법
- 문서번호 : KR-ES-0178
- 작성자 : 이율삼(charles.lee@microchip.com , Field Application Engineer)

PIC16F886 의 경우 인터럽트 벡터가 1개 있기 때문에 2개이상의 인터럽트 서비스 루틴을 사용하는 경우 낮은 우선순위의 인터럽트 서비스 루틴이 수행중일때 높은 우선순위의 인터럽트가 걸리더라도 서비스 루틴이 끝나기 전에는 높은 우선순위의 인터럽트가 실행될수 없다

이러한 문제는 어떤경우에는 시스템의 인터럽트 레이턴시가 늘어나게 되어 큰 문제가 될수 있다. 아래의 예제는 높은 우선순위의 SPI slave 인터럽트 태스크 와 낮은 우선순위의 타이머를 이용한 4채널 펄스폭 측정 태스크 이 두가지를 높은 우선순위의 SPI slave 인터럽트 태스크에 영향을 최소화 하면서 낮은 우선순위의 타이머 인터럽트가 문제없도록 구현한 예제 이다.

PIC16F886의 인터럽트 서비스 루틴에서 우선순위가 낮은 타이머 인터럽트 서비스 루틴의 실행시간을 최소화 시키고 타이머 서비스 루틴에서 수행해야 할 기능을 메인 루틴에서 구현함으로써 높은 우선순위 태스크의 worst case 인터럽트 레이턴시를 최소화 시킬수 있었다.

```
void interrupt int886() // -- interrupt service
{
    if(SSPIF){ //executed when one byte data received 높은 우선순위의 인터럽트 먼저 체크
        if(receive){
            if(SSPBUF==0xaa & cnt==0){
                buf[0]=SSPBUF;
                en=1;
            }else {
                if(en){
                    cnt++; buf[cnt]=SSPBUF;
                }
                if(cnt==0x06){cnt=0;receive=0;SSPBUF=sbuf[cnt++];en=0;} //change to send mode
            }
        }
    }else {
        if(cnt<5){
            SSPBUF=sbuf[cnt++];
        }else
        {
            SSPBUF=sbuf[cnt++];
            receive=1; //change to receive mode
            cnt=0;
        }
    }
}
```


