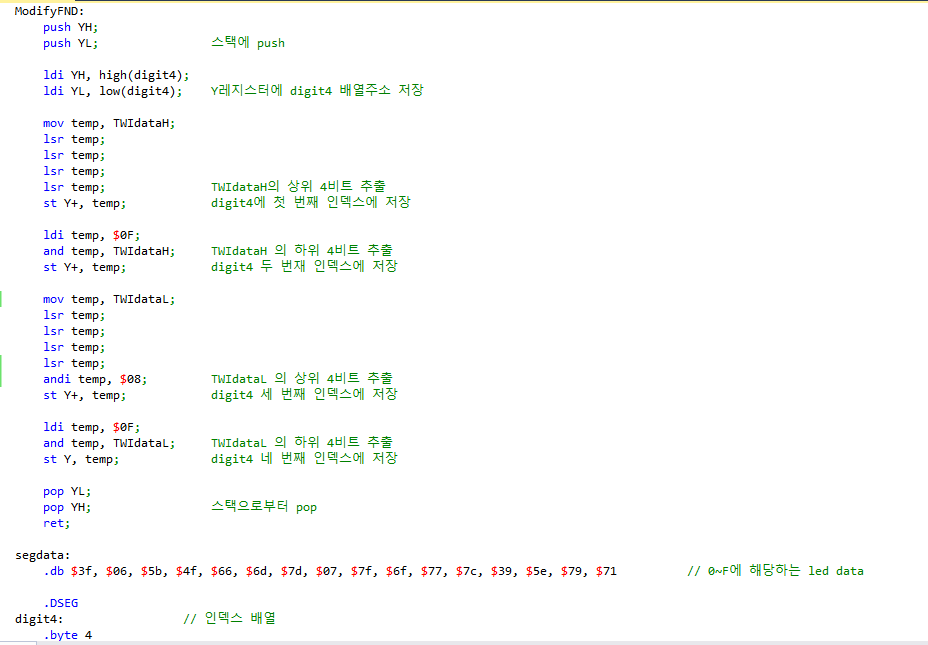
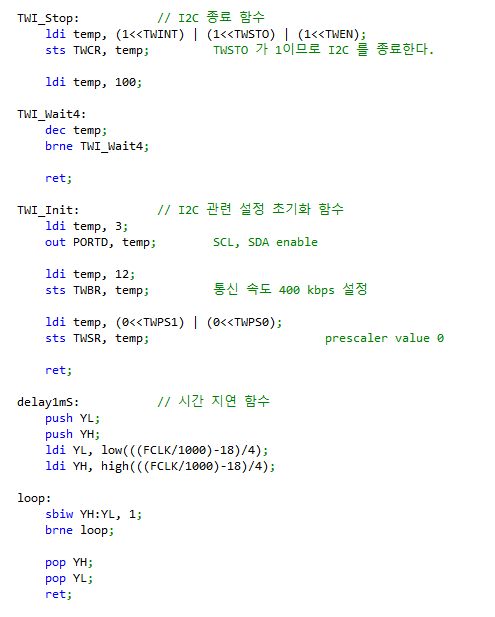
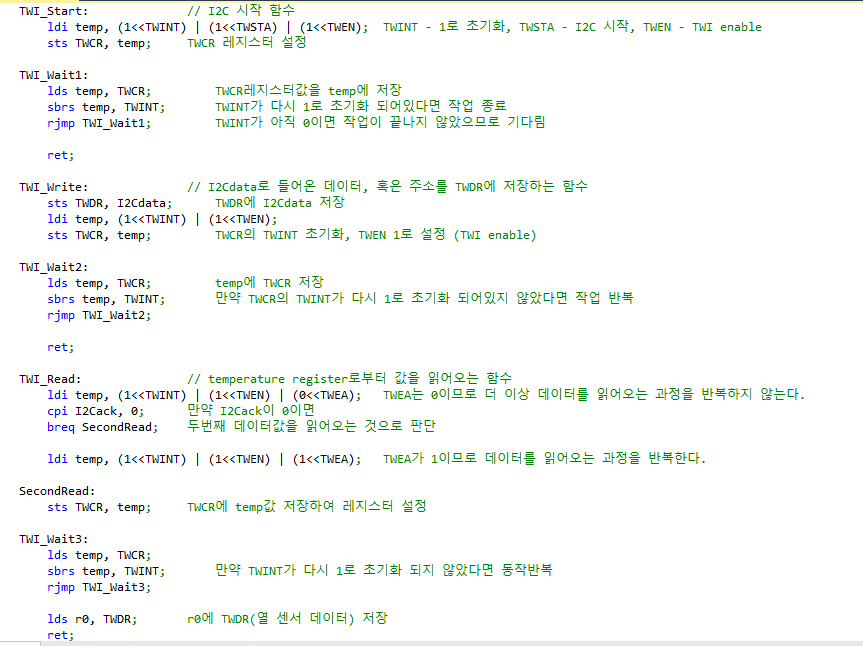
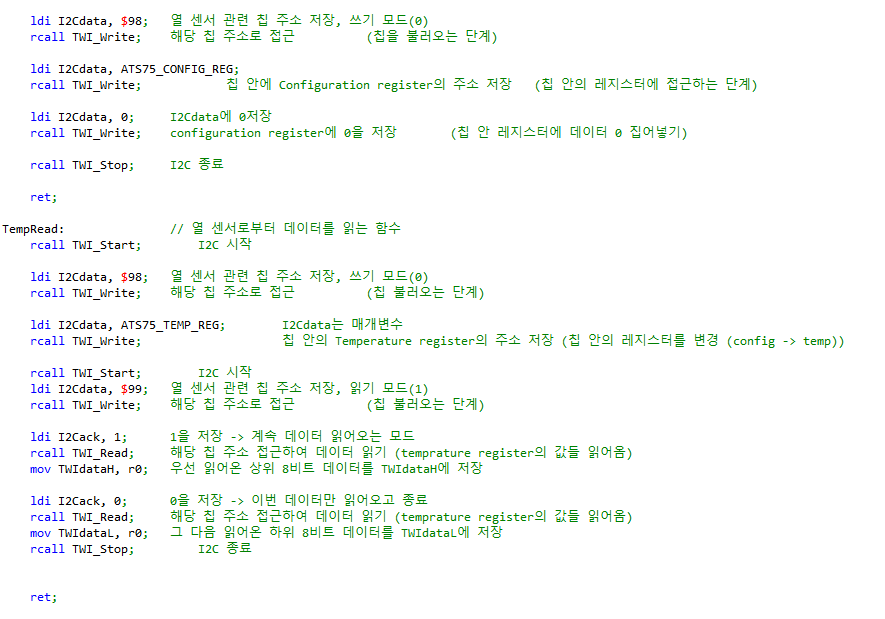
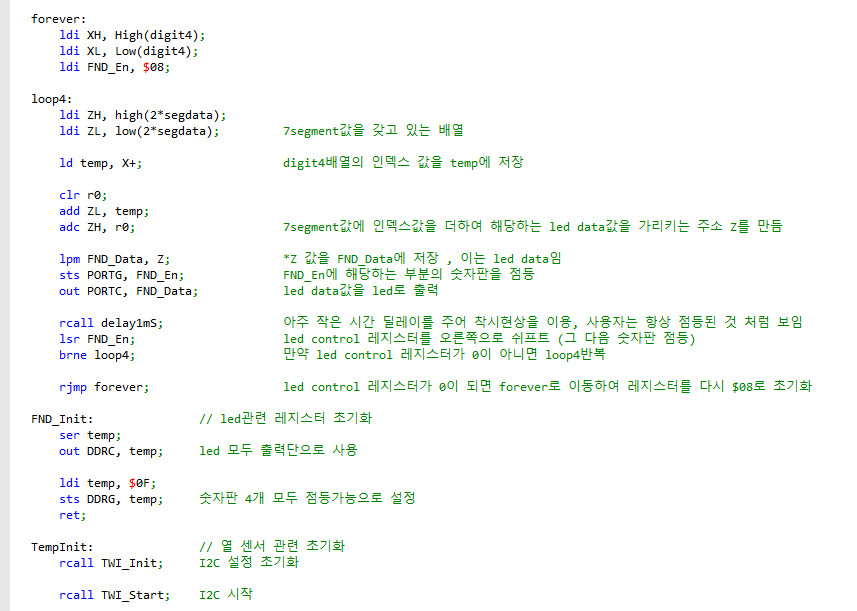
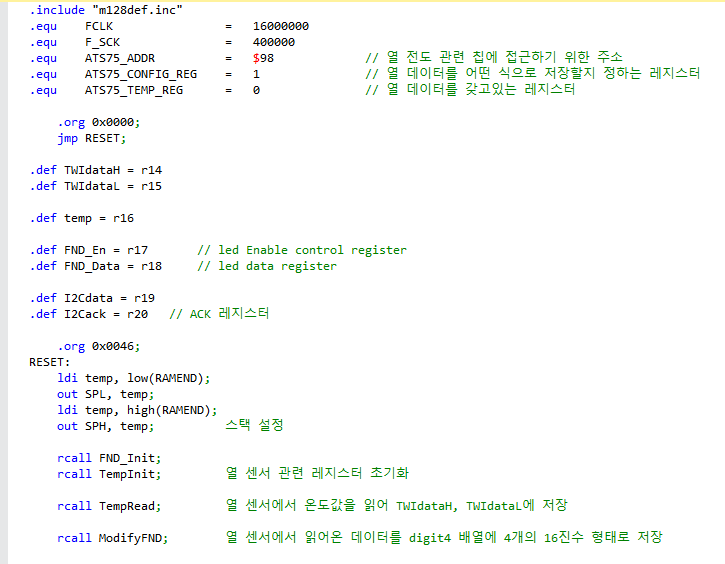
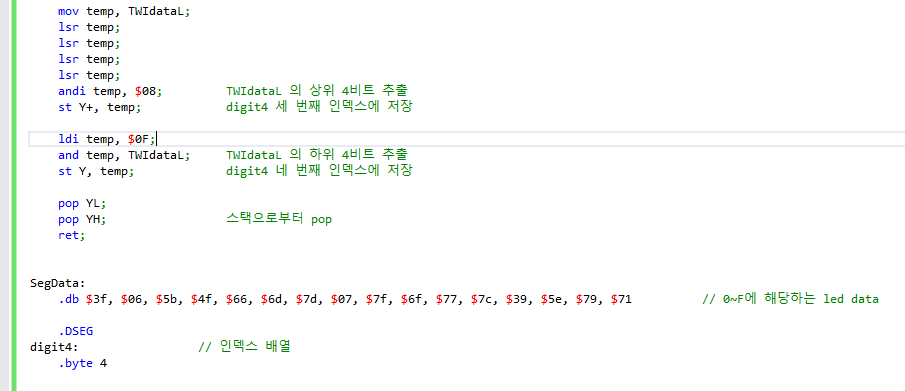
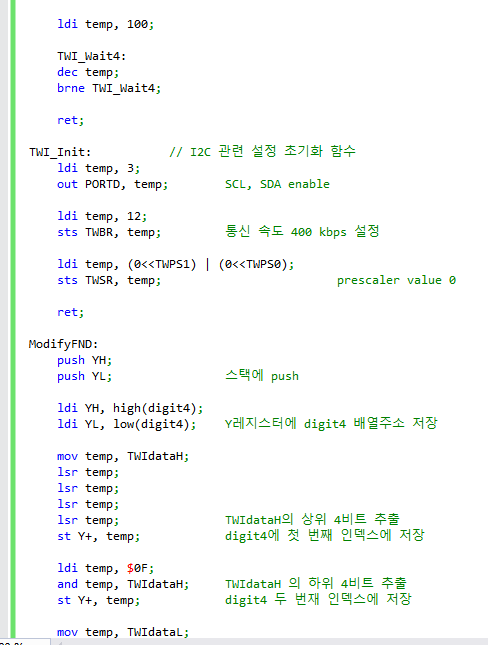
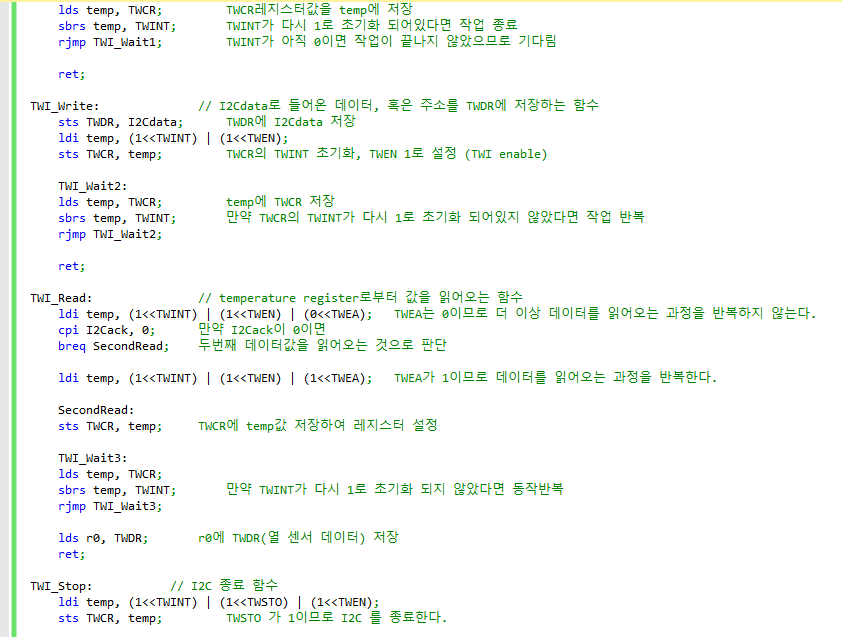
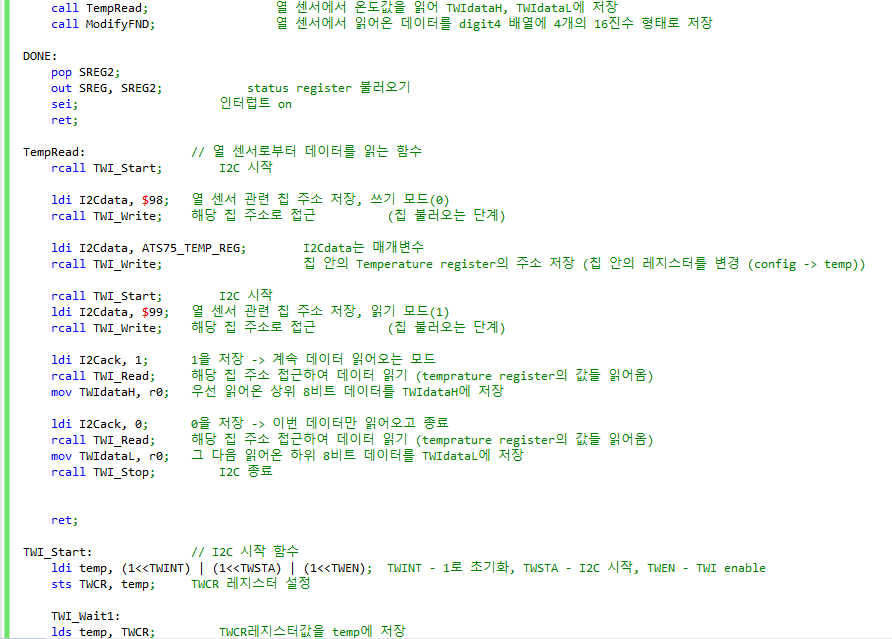
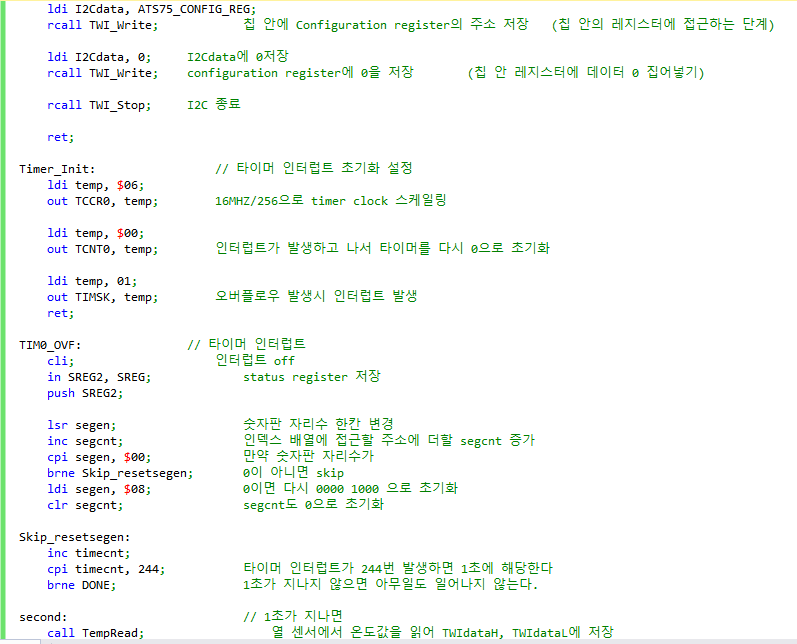
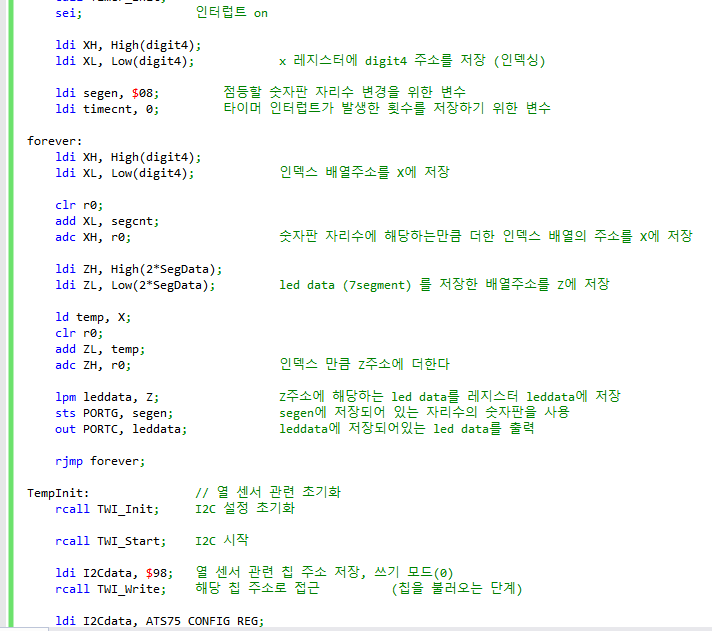
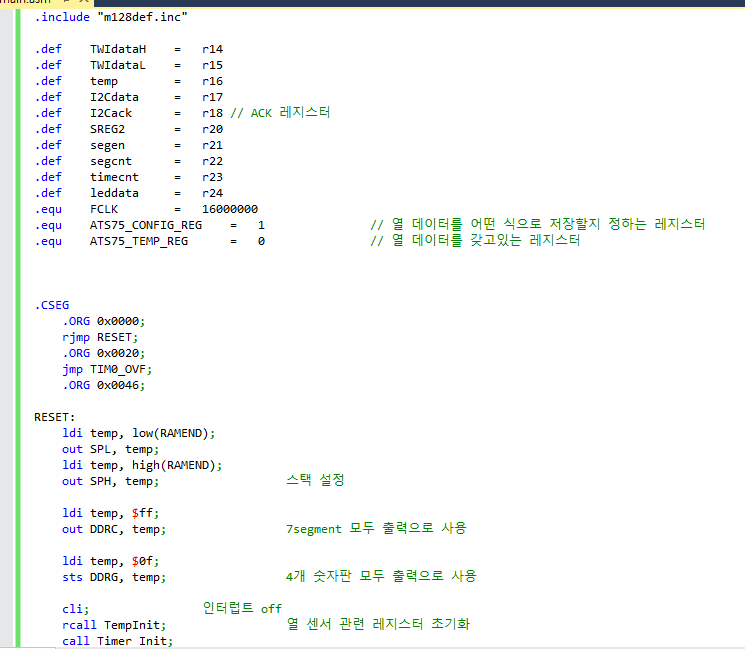
Lab 11-a



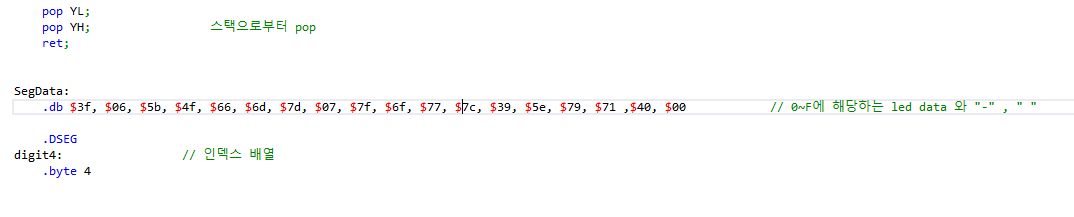
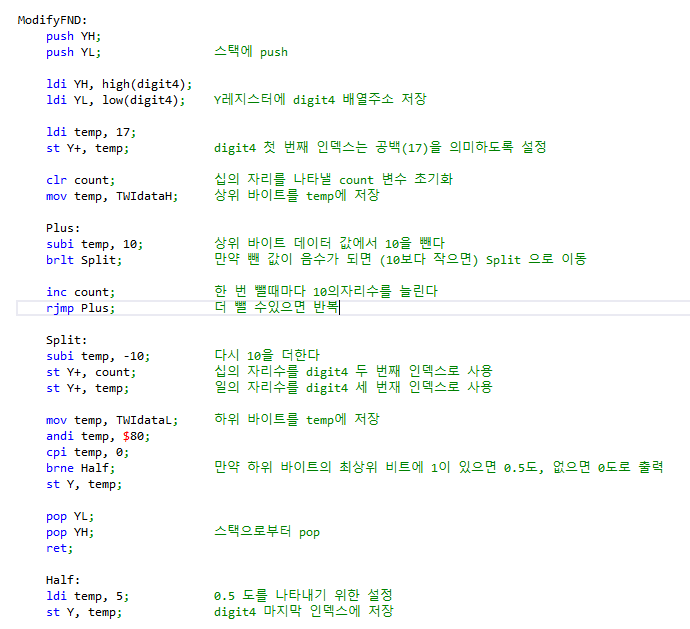
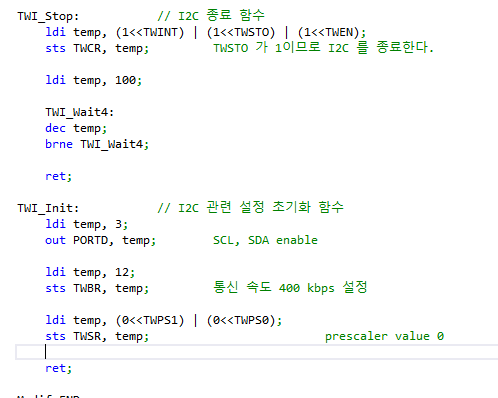
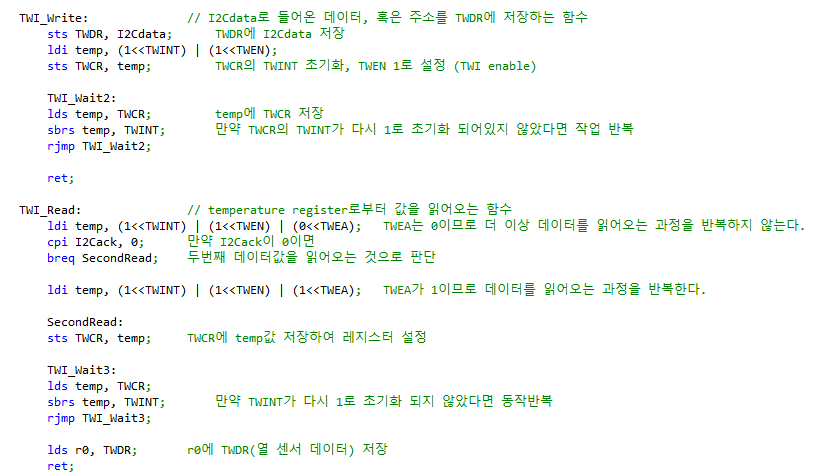
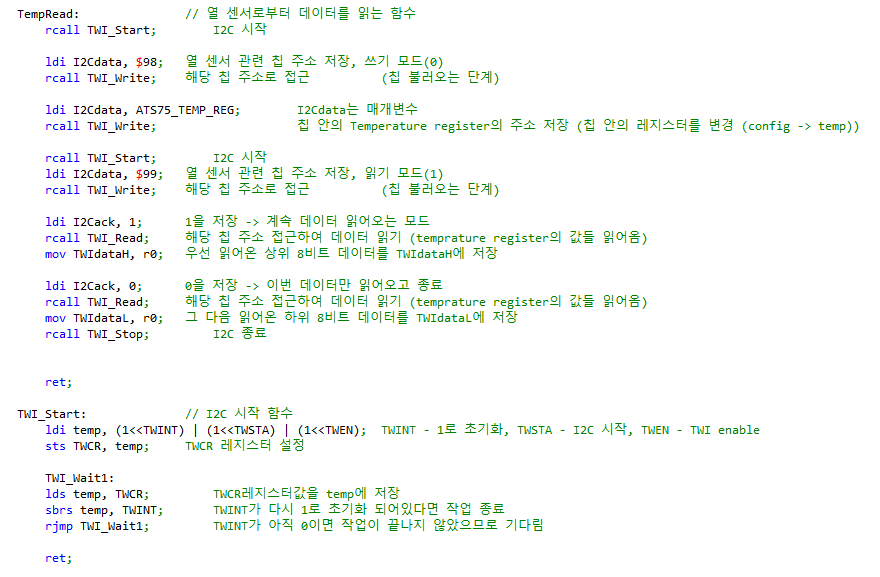
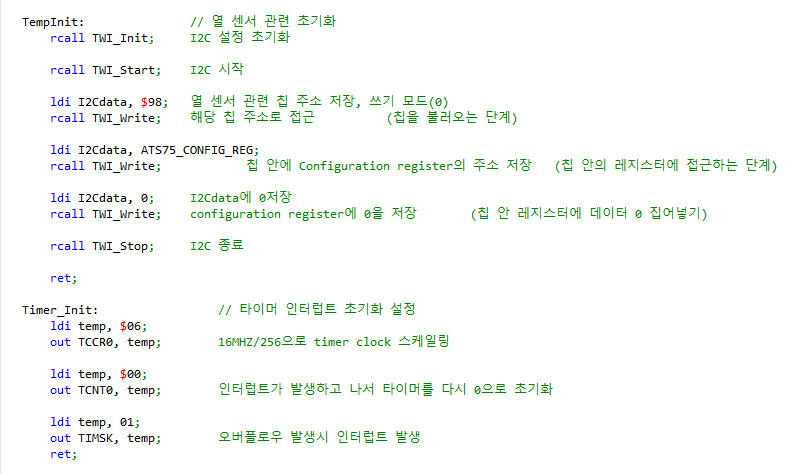
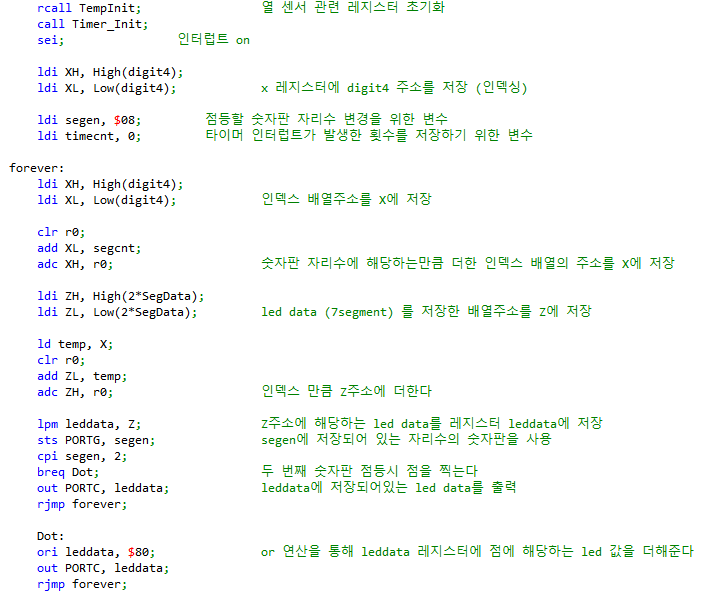
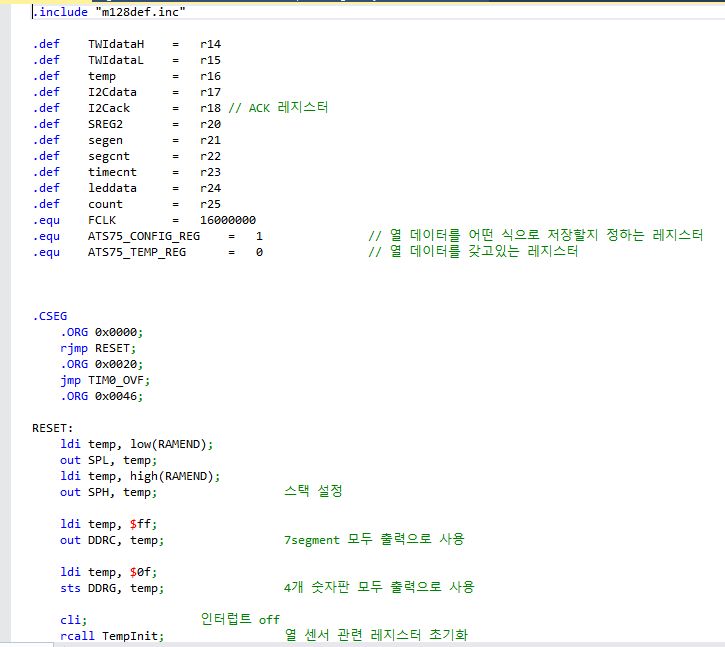
I2C (TWI 통신) 을 이용하여 CPU 와 주변 기기와의 직렬통신이 가능한데 그 중 ADC 변환을 담당하는 칩과 통신을 주고받는 실습이다. 주고 받는 정보는 온도 센서를 이용한 현재 온도 정보이며 상위 8비트 , 하위 8비트의 16비트 데이터를 주고 받지만 실제로는 configuration register 설정을 통해 하위비트의 경우 1~4비트까지 유의미한 정보를 입력받을 수 있다. 현재 실습에선 configuration register에 0을 집어넣었고 이는 하위비트 8비트 중 MSB 1 비트만 유의미한 정보로서 받아들이고 있다. 이 온도정보를 7-segment로 display 하였다.

Lab 11-b



lab 11-b는 매 초마다 주변의 온도를 읽어 LED display를 수행하는 실습이었다. 주변의 온도를 읽어 digit4에 저장하는 과정자체는 11-a 실습과 동일하다. 매 초마다 수행하기 위하여 타이머 인터럽트를 사용하였다. 타이머 인터럽트가 244번 발생하여 1초가 되면 TempRead 함수를 호출하여 주변의 온도 데이터를 얻어오고 ModifyFND를 수행하여 얻어온 온도 데이터를 digit4 에 저장하였다. 이런 과정을 통해 매 초마다 주변의 온도값을 읽는 온도계를 만들 수 있다.

lab 11-c



이전 실습에서 RAW 한 데이터 정보를 얻었다면 이번실습은 이를 10진수의 실제 온도계처럼 변환하여 좀 더 유용성을 높이는 실습이다. 16진수로 얻은 데이터 값을 10의 뺄셈을 통해 10의 자리수와 1의 자리수를 분리, 소수점의 경우 0.5도와 0도를 구별할 수 있도록 프로그래밍하였다.

소수부와 정수부를 분리하기 위해 소수점을 구현하였는데 이는 segcnt (led control) 값이 2가 되면 leddata에 $80을 or 연산하는 것으로 leddata값에 점을 포함하여 출력하도록 설정하였다.

고찰

처음 수업을 듣기 전엔 12째 주에 수행하였던 Photo sensor 실습과 비슷할 것이라 생각했다.

둘다 똑같이 외부의 아날로그 신호를 입력받아 디지털화하여 AVR보드에 나타내기 때문에 그 과정역시 비슷할 것이라 생각했었는데 이번주의 실습은 과정에서 큰 차이를 보였다. 광량 측정 센서를 통해 조도값을 읽어들이는 실습은 조금 더 독립적인 I/O 장치 구동의 느낌이었다면 이번주 실습은 TWI 통신 (I2C) 을 통한 I/O 장치 구동으로 Master – slave 관계의 조금은 종속적인 느낌의 구현이었다. I2C의 사용법은 첫 번째로 칩의 주소를 모드선택과 함께 입력하고, 두 번째로 해당하는 칩의 레지스터 주소를 입력해 레지스터로 접근, 마지막으로 해당하는 레지스터에 데이터값을 넣어주는 3단계의 과정을 거친다. 이러한 점을 잘 숙지한다면 TWI 통신 역시 크게 어려운점 없이 수행할 수 있었다.