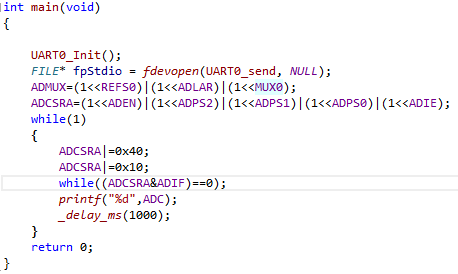
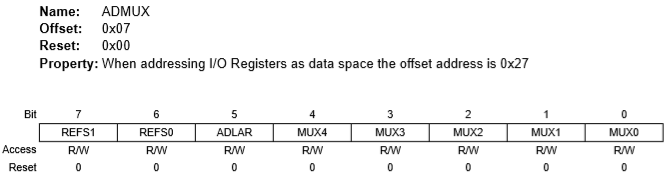
ADC

이번장에서는 ADC에 대해서 다룬다. ADC는 아날로그-디지털 컨버터의 약자로 아날로그 신호를 디지털 값으로 바꿔주는 용도의 기능이다. 이기능을 이용하면 단순한 HI/Low 값이 아닌 파형의 전앖값을 알수 있게 해주는 기능이다.

센서 중에서는 0과 1만이 아닌 전앖값을 출력해주는 센서들이 있다. 주로 아날로그 센서에서 자주 쓰이는 형태인데 이 전앖값을 이용해서 정확한 값을 측정해야 되는 종류들의 센서이다. 이런 센서들을 활용하려면 ADC를 사용하는 방법들에 대해서 배워야 된다.

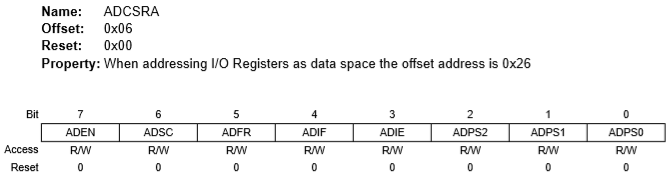
이 코드는 ADC를 세팅해서 ADC1에서 아날로그 데이터 값을 받아와서 출력해준다. printf문을 쓰는데 이에 대한 사용처는 후에 UART에서 다루기 때문에 여기서는 이야기 하지 않는다. 이번에도 처음보는 레지스터가 많은데 이것에 대해서는 차근차근 설명할 것이다.



이 레지스터는 ADC 멀티플렉서 레지스터로 입력받을 ADC의 채널과 레퍼런스 전압 등을 결정한다. 이 레지스터를 기능별로 나누면 다음과 같이 3분류로 나눌 수 있다.

* REFS0,1=ADC가 참고할 기준전압에 대해서 선택한다.
* ADLAR=ADC에서 출력되는 데이터의 기준 방향을 정한다.
* MUX0~4=ADC채널을 고른다. 단일 채널은 총 8개가 존재하며 세팅 여부에 따라서 입력 이득이나 양/음 입력등을 선택할수 있다.

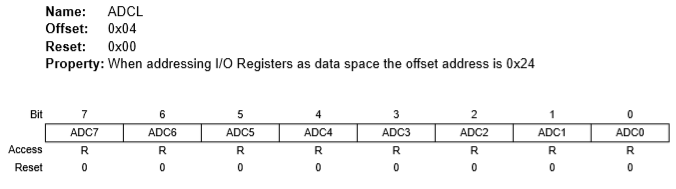
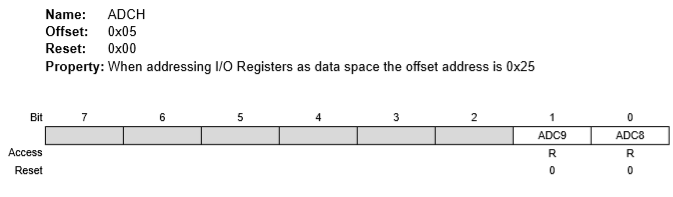
위의 코드에서는 데이터 방향을 왼쪽 정렬로 하고 기준전압은 AVcc로 잡은 다음 ADC채널 1번을 선택해서 PF1에서 입력을 받는다.

ADCSRA레지스터는 ADC 컨트롤/상태 레지스터로 ADC의 동작에 대한 설정이나 ADC의 상태를 확인할수 있다. 각 비트마다 개별적인 기능이 하나씩 있는데 이는 다음과 같다.

* ADEN=ADC동작 비트로 이 비트를 1로 해줘야 ADC가 동작한다
* ADSC=ADC데이터 변환 시작 비트로 이 비트를 1로 해주면 ADC가 데이터 변환을 시작한다.
* ADFR=ADC Free Running모드 선택으로 이 비트를 1로 해놔야 샘플링/변환이 연속적으로 이루어진다
* ADIF=ADC인터럽트 플래그. 변환이 다 되었으면 상태가 1로 올라가면서 ADC변화이 다 되었음을 알리는 플래그다
* ADIE=ADC 인터럽트 동작 플래그. ADC인터럽트를 사용하기 위해서 1로 올려야된다
* ADPS0~2=ADC 프리스케일러. ADC의 샘플링 클럭을 조절하기 위한 레지스터다.

위의 코드에서는 ADEN과 ADIE를 사전에 1로 해주고 프리스케일러 비트틀 전부 올려서 최대 분주비로 놓고 사용하고있다. ADSC와 ADIF는 후에 사용된다.

While 문에 들어서게 되면 먼저 ADCSRA|=0x40으로 ADSC비트를 1로 만들어 주는 것으로 ADC변환을 시작하고 ADCSRA|=0x10으로 ADIF비트를 토글해서 내려준다. 이후 while 문을 하나 더 넣어서 ADIF가 1이 될때까지 대기한다. ADIF가 1이 되면 변환이 끝났다는 이야기이기 때문에 출력문을 이용해서 ADC에 있는 값을 출력한다.



ADC는 원래 ADCH와 ADCL로 나누어져 있는 레지스터다. 이는 8비트를 넘어서는 값을 표현하기 위한 편법으로써 사용되는 방법인데 ATmega128A에 탑재된 ADC는 10비트 출력값을 가진 ADC로 최대 출력값이 1024이다. 하지만 8비트 레지스터로는 안에 1024비트를 담지 못하기 때문에 편법을 쓴다. 남는 2비트를 다른 레지스터에 저장하는 방식으로써 추가적인 데이터를 저장한다.

컴파일러의 발전으로 ADC라고 쓰기만 해도 이 두 레지스터의 값을 총합해서 한 개의 값으로 바꿔주는 방식이기 때문에 코드에서는 ADC라고만 작성해도 정상적인 값이 출력된다. 마이크로 컨트롤러에 따라서 이를 지원할수도 있고 아닐수도 있기 때문에 사용할 때 이에 대해서 주의할 필요가 있다.

ADC를 변환하면 아래와 같은 방식으로 데이터가 출력된다.



ADC의 변환값은 0~1024로 이는 레퍼런스 전압인 5볼트를 1024등분했다고 생각하면 된다. 이 데이터로부터 원래의 전압값을 구하려면 다음과 같은 공식을 써서 구할수 있다

ATmega128A의 ADC분해능은 10비트이고 레퍼런스 전압은 코드 초기에서 외부전압인 5V로 잡아놨기 때문에 이를 통해 식을 세우면

따라서 위에 출력된 4개의 값은 각각 0V, 1.25V, 3.125V, 3.75V라는 것을 알수 있다.

ADC가 완벽하게 정확하진 않다. 분해능에 따라서 ADC데이터의 1 값은 약 0.004V에 해당하는데 이는 이보다 작은값은 실제로 무시 될 수도 있다는 의미이다. 하지만 실제로 이정도의 오차로 문제가 될 일은 아주 정밀한 기계가 아닌 이상 거의 없으니 신경 쓰지 않아도 된다.

마이크로 컨트롤러마다 이는 전부 다르게 나올수 있다. 별도의 ADC기준전압을 잡아줬거나 3.3V를 쓰는 마이크로 컨트롤러를 사용한다 거나 하면 레퍼런스 전압이 바뀌는 관계로 값이 바뀌며 분해능이 더 높은 ADC를 가진 마이크로 컨트롤러의 경우에는 더 세세한 값이 나올 수도 있다.

지금까지 ADC에 대해서 다루었다. ADC는 센서를 활용하기위한 가장 기초적인 기능중 하나로 많은 센서들이 아날로그값 출력을 가지고 있는 만큼 써먹을 일이 많을 것이다.