



AVR - Chapter 7

류대우

(주)뉴티씨(NEWTC) 대표이사

davidryu@newtc.co.kr

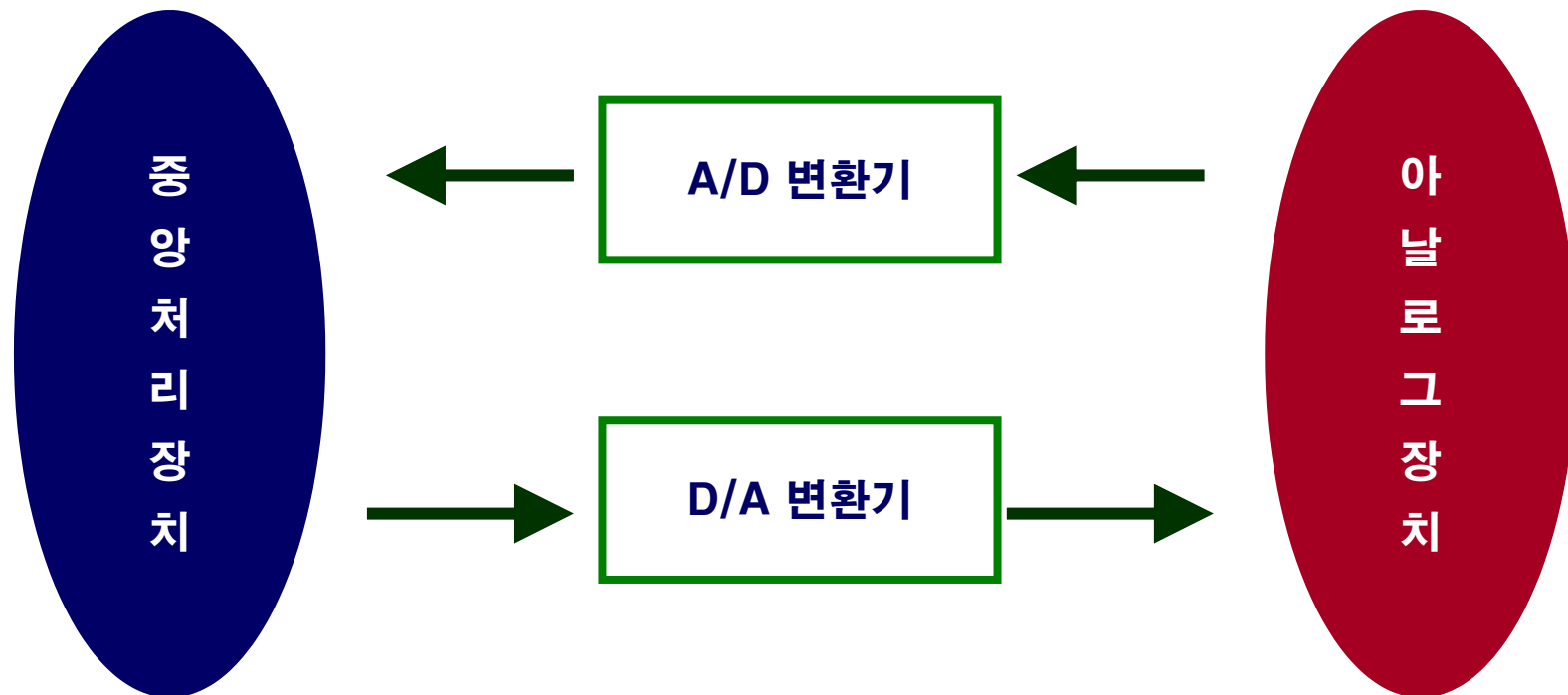
[A/D 컨버터란?



- 연속적인 신호인 아날로그 신호를 부호화된 디지털 신호로 변환하는 일.
- 아날로그 디지털 변환을 수행하는 기계 장치를 아날로그 디지털 변환기(AD Converter)라고 함.
- 이 장치는 온도, 압력, 음성, 영상 신호, 전압 등의 실 생활에서 연속적으로 측정되는 신호를 컴퓨터에 입력 하여 디지털화시키는 장치

[A/D 컨버터

]



[AVR의 내장 A/D 컨버터 특징]

- 8채널
- 10비트 분해능
- 축차비교형(변환시간 수십us의 종속형 A/D Converter)
- 내부 아날로그 멀티플렉서 탑재
- 샘플/홀드회로 탑재로 인하여 A/D 동작 동안 전압 고정화
- 단극성 아날로그 입력 / 차동 입력 선택
- 차동 입력에서 10배~200배의 증폭 A/D 가능
- 포트 F는 아날로그 비교기 기능으로도 사용 가능
- 변환시간 (13us~260us (50kHz~200kHz))

[분해능]

■ 분해능

- 마이컴에서 분해능이란 보통 AD 컨버터에서 주로 사용되는 용어.
- 외부에서 아날로그 신호가 들어 오게 되면 sin파가 입력되게 됨.
- 그런데 예를들어 이 아날로그 신호가 어떻게 변화가 되는지를 1초에 한번씩 체크하는 것과 0.1초마다 체크하는 것, 그리고 0.01 초마다 체크 하는 것은 다름.
- 얼마나 세밀하게 체크를 하는가가 얼마나 "분해"를 하는 것인지와 의미가 같음.
- 좀더 세밀하게 체크할 수 있다면 보다 정확한 데이터를 구할 수가 있음. 이것이 분해할 수 있는 능력이라고 함.

[A/D 컨버터의 구성]

■ 범용 PORTF의 특수 기능

- ADC0 ~ ADC7 : 8채널 10비트 A/D 컨버터의 아날로그 입력단자
- ADC 정확도 성능 향상을 위한 독립 전원 구성
- AVCC : Analog Supply Voltage(VCC의 전압의 $\pm 0.3V$ 유지해야함)
- AGND : Analog Ground (반드시 GND와 연결)
- AREF : Analog Reference Voltage

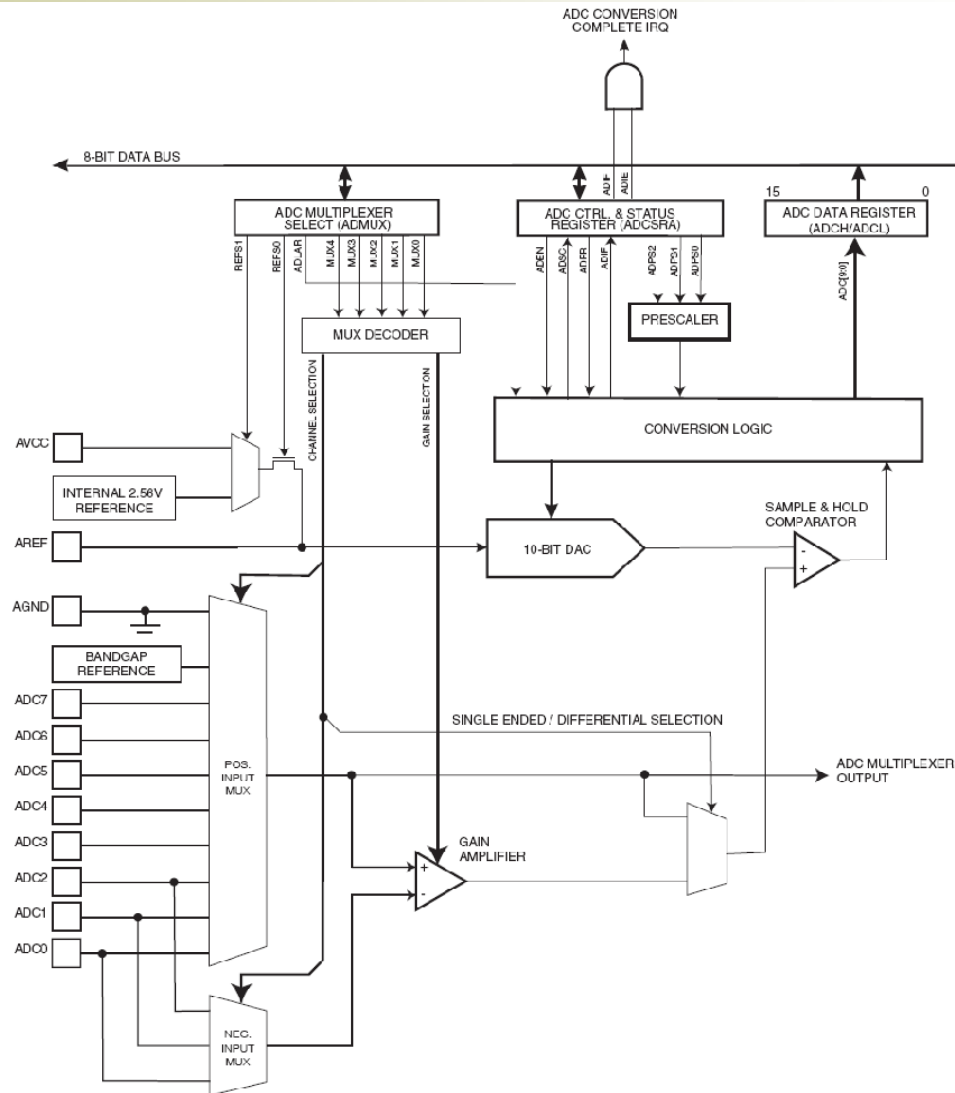
■ 입력 전압의 범위

- 일반 모드 입력전압의 범위($0V \sim V_{ref}$)
- 차동입력 모드 입력전압의 범위($-V_{ref} \sim V_{ref}$)

■ Vref의 범위

- 전원전압 VCC를 초과할 수 없다.($V_{REF}=V_{GND} \sim V_{CC}$, 내부기준전압 2.56V)

[AD Convertor의 블록도]



[A/D 변환 결과]

- ADC 변환 결과

$$ADC = \frac{V_{IN} \cdot 1024}{V_{REF}}$$

[단극성 입력]

$$ADC = \frac{(V_{POS} - V_{NEG}) \cdot GAIN \cdot 512}{V_{REF}}$$

[차동 입력]

[A/D 변환 오차 종류]

- 비선형 오차 (Integral non-linearity error)
 - 변환 결과가 교정될 수 없는 상태
 - 대처방안 없음
- 차동 비선형 오차 (Differential non-linearity error)
 - 변환 결과가 교정될 수 없는 상태
 - 대처방안 없음
- 양자화 오차 (Quantization error)
 - 아날로그 값을 디지털 값으로 변환하면서 생기는 변환의 한계
 - 대처방안 : 분해능이 높은 ADC사용으로 극복
- 오프셋 오차 (Offset error)
 - 변환 결과가 이상적인 디지털 값에서 일정한 양만큼 벗어난 상태
 - 대처방안 : 변환된 디지털 값에 일정치 값을 더하거나 빼서 교정한다.
- 이득 오차 (Gain error)
 - 변환 결과가 이상적인 디지털 값에서 일정한 비율만큼 벗어난 상태
 - 대처방안 : 변환된 디지털 값에 일정치 값을 곱하거나 나누어서 교정한다.

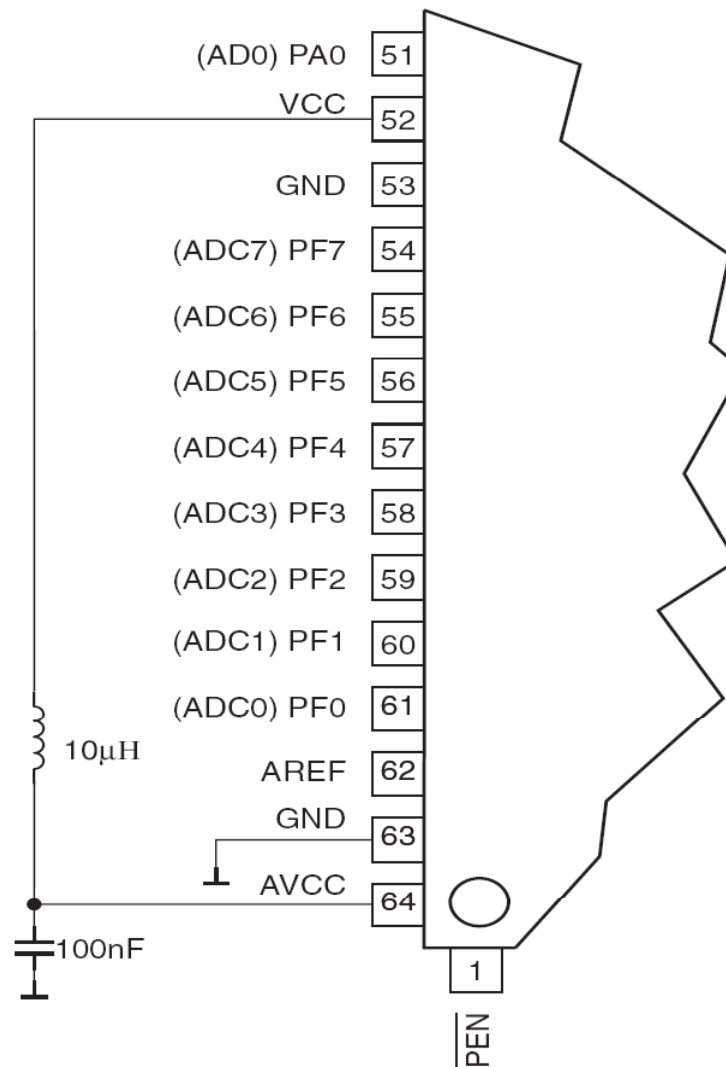
[A/D Convertor 잡음 제거 방법[1]]

- A/D Convertor의 경우에는 노이즈에 매우 민감하기 때문에 ATmega128 내에서도 AVCC, AREF, AGND와 같은 ADC 전원 구성도 따로 하였으며 사용자 또한 몇 가지 사항을 주의하여 사용해야 한다.
 - AVCC = 독립적인 아날로그 회로 전원 단자
 - AREF = 기준 전원 입력 단자
 - AGND = 아날로그 회로 접지 단자

[A/D Convertor 잡음 제거 방법[2]]

- ① 아날로그 입력선은 최소한으로 짧게 하고 잡음의 영향이 없도록 회로를 구성한다.
- ② 아날로그 전원단자 AVCC에 VCC를 인가할 때는 LC필터를 거쳐 안정화 시킨다.
- ③ 아날로그 회로의 모든 접지는 AGND에 접지하여 한 포인트에서만 GND와 접속한다.
- ④ ADC 동작중에는 병렬 I/O 포트의 논리상태를 스위칭하지 않는다.
- ⑤ 잡음에 민감한 아날로그 소자의 ADC의 경우에는 Adc Noise Reduction mode를 사용한다.
- ⑥ 잡음이 심하여 결과 값의 변동이 심하면 디지털 필터를 사용하거나 평균치를 구하여 사용한다.

[A/D Convertor 잡음 제거 방법[3]]



[AREF 관련하여]

- AREF핀에 고정된 전압 소스가 연결된다면, 어플리케이션에 다른 레퍼런스 전압 옵션들을 사용해서는 안됩니다. 이유는 그것들이 외부 전압에 쇼트될 것이기 때문입니다.
- 만약, AREF핀에 아무 외부 전압이 가해지지 않았다면, AVCC 또는 2.56V를 레퍼런스 선택으로 전환할 수 있습니다.
- 레퍼런스 전압 소스를 스위칭한 후 첫번째 ADC 컨버전은 정확하지 않을 수 있으므로,이 값은 버려야 한다.

[A/D 컨버터의 동작]

1. A/D 컨버터 초기화 설정

- ADCSRA 설정 (0x0F → 0x8F)
 - ADC 활성화 안하기(ADEN=0)
 - ADSC 시작 변환 (변환시작 안하기:0)
 - ADFR 동작 모드 설정 (프리런닝(1)/단일변환(0))
 - ADIE설정 : ADC 변환완료 인터럽트 활성화(1)
 - ADPS 분주비 설정 (ADPS 2~0 : 1/128은 111)
 - ADC 활성화 하기(ADEN=1)
- ADMUX 설정(0x00|ch)
 - ADC 기준전압설정 (REF 1~0)
 - ADLAR 설정(좌측정렬:1)
 - ADC 입력채널설정 (MUX 4~0)

2. A/D 컨버터 시작 설정

- ADCSRA 설정
 - ADSC 시작 변환 (변환시작 :1)

3. A/D 컨버터 데이터 받기

- ADC 변환완료 인터럽트 ISR에서 데이터 수신 받기 : 16진수 int 변수 value
 - value=ADCL; //Read 8 low bits first (important)
 - value|=(int)ADCH << 8; //read 2 high bits and shift into top byte

[ADMUX (1/2)]

■ ADMUX(ADC Multiplexer Selection Register)

Bit	7	6	5	4	3	2	1	0	
	REFS1	REFS0	ADLAR	MUX4	MUX3	MUX2	MUX1	MUX0	ADMUX
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

○ Bit 7, 6 – REFS1, REFS0 (Reference Selection Bit)

REFS1	REFS0	기준 접압
0	0	외부의 AREF 단자로 입력된 전압을 사용한다.
0	1	외부의 AVCC 단자로 입력된 전압을 사용한다.
1	0	-
1	1	내부의 2.56V를 사용한다

○ Bit 5 – ADLAR (ADC Left Adjust Result)

- ADLAR = 1 : 변환결과값을 ADCH/L에 저장할 때 좌측으로 끝을 맞추어 저장된다

○ Bit 4,3,2,1,0 – MUXn (Analog Channel and Gain Selection Bit)

MUX4..0	Single Ended Input	Positive Differential Input	Negative Differential Input	Gain
0000	ADC0	N/A		
0001	ADC1			
00010	ADC2			
00011	ADC3			
00100	ADC4			
00101	ADC5			
00110	ADC6			
00111	ADC7			
01000 ⁽¹⁾		ADC0	ADC0	10x
01001		ADC1	ADC0	10x

[ADMUX (2/2)]

MUX4..0	Single Ended Input	Positive Differential Input	Negative Differential Input	Gain
01010 ⁽¹⁾	N/A	ADC0	ADC0	200x
01011		ADC1	ADC0	200x
01100		ADC2	ADC2	10x
01101		ADC3	ADC2	10x
01110		ADC2	ADC2	200x
01111		ADC3	ADC2	200x
10000		ADC0	ADC1	1x
10001		ADC1	ADC1	1x
10010		ADC2	ADC1	1x
10011		ADC3	ADC1	1x
10100		ADC4	ADC1	1x
10101		ADC5	ADC1	1x
10110		ADC6	ADC1	1x
10111		ADC7	ADC1	1x
11000		ADC0	ADC2	1x
11001		ADC1	ADC2	1x
11010		ADC2	ADC2	1x
11011		ADC3	ADC2	1x
11100		ADC4	ADC2	1x
11101		ADC5	ADC2	1x
11110	1.23V (V _{EG})	N/A		
11111	0V (GND)			

[Control and Status Register]

■ ADCSRA(ADC Control and Status Register)

Bit	7	6	5	4	3	2	1	0	
	ADEN	ADSC	ADFR	ADIF	ADIE	ADPS2	ADPS1	ADPS0	ADCSRA
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- Bit 7 – ADEN (ADC Enable)
 - ADEN = 1 : ADC 활성화
- Bit 6 – ADSC (ADC Start Conversion)
 - ADSC = 1 : ADC의 변환이 시작된다.
 - (단일 변환모드일때 단 한번만 작동 / 프리런닝모드일때 변환동작반복)
- Bit 5 – ADFR (ADC Free Running Select)
 - ADFR = 1 : 프리런닝모드 ■프리런닝코드(Free running mode) : ADC 동작을 반복적으로 수행하게 된다
 - ADFR = 0 : 단일변환모드 ■단일변환모드(Single conversion mode) : ADC 동작을 한 번만 수행하게 된다.
- Bit 4 – ADIF (ADC Interrupt Flag)
 - ADC변환완료 인터럽트가 요청되고 그 상태를 이 비트에 표시한다.
- Bit 3 – ADIE (ADC Interrupt Enable)
 - ADIE = 1 : ADC Interrupt 활성화
- Bit 2,1,0 – ADPS 2~0 (ADC Prescaler Select Bit)
 - ADC에 인가되는 클럭의 분주비를 설정한다

ADPS2	ADPS1	ADPS0	Division Factor
0	0	0	2
0	0	1	2
0	1	0	4
0	1	1	8
1	0	0	16
1	0	1	32
1	1	0	64
1	1	1	128

[ADC 실습



ICCAVR Application Builder [M128]

CPU | Memory | Ports | Timer0 | Timer1 | Timer2 | Timer3 | UART | SPI | Analog

Processor

Target CPU: M128

Xtal speed (MHz): 16.000

☐ PLL enable

☐ XDIV enable...

XDIV frequency: 124031Hz

Watchdog timer

☐ Enable Prescale cycles: 16K

☐ Interrupt enable

External interrupts

Trigger on...

☐ INT0 Low level

☐ INT1 Low level

☐ INT2 Low level

Edge ☐

☐ INT3 Low level

☐ INT4 Low level

☐ INT5 Low level

☐ INT6 Low level

☐ INT7 Low level

Comment: new design

EEPROM

☐ ready interrupt

Ok Options Preview Cancel

[ADC 실습



ICCAVR Application Builder [M128]

CPU | Memory | Ports | Timer0 | Timer1 | Timer2 | Timer3 | UART | SPI | Analog

External memory

☒ Enable external memory

External memory page configuration

NA / 0x1100-0xFFFF

☐ Wait states

Lower wait states: None

Upper wait states: None

Absolute address definitions

Address	Name	Type	Count
0x0000		signed char	1

Add

Modify

Remove

Ok Options Preview Cancel

[ADC 실습

]

ICCAVR Application Builder [M128]

CPU | Memory | Ports | Timer0 | Timer1 | Timer2 | Timer3 | UART | SPI | Analog

Port A

	7	6	5	4	3	2	1	0
Direction	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Value	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Change	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Port B

	7	6	5	4	3	2	1	0
Direction	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Value	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Change	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Port C

	7	6	5	4	3	2	1	0
Direction	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Value	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Change	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Port D

	7	6	5	4	3	2	1	0
Direction	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Value	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Change	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Port E

	7	6	5	4	3	2	1	0
Direction	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Value	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Port F

	7	6	5	4	3	2	1	0
Direction	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Value	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Port G

	7	6	5	4	3	2	1	0
Direction	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Value	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

right click on a "value" bit to define signal name

Ok Options Preview Cancel

[ADC 실습

]

ICCAVR Application Builder [M128]

CPU | Memory | Ports | **Timer0** | Timer1 | Timer2 | Timer3 | UART | SPI | Analog

Initialisation

☒ Use Timer0 Power off ☐ Overflow interrupt ☐

Desired value Units Actual value (error%)

10 mSec 9.984mSec (0.2%)

Prescale select ... TCNT0

1024 0x64

Waveform mode

Normal

Compare

OCR0

0x9C

OCO output mode

Disconnected

☐ Clear on compare

☐ Compare interrupt

Asynchronous mode

☐ Enable async mode

External crystal (Hz)

32768

Ok Options Preview Cancel

[ADC 실습



ICCAVR Application Builder [M128]

CPU | Memory | Ports | Timer0 | Timer1 | Timer2 | Timer3 | UART | SPI | Analog

Analog Comparator

☐ Use Analog comparator

☒ Disable

☐ Timer1 input capture enable

☐ Multiplexer enable

☐ Bandgap reference

Interrupt trigger level
Output toggle

☐ Compare interrupt

Analog to Digital Convertor

☒ use ADC Power off ☐

☒ ADC enable

☐ Free-run select

☐ Start conversion

☐ Auto trigger

☐ Bandgap reference

☐ Internal reference

☐ Left adjust

Reference: AREF Trigger source: Free running

Prescale: 128 ☐ High speed

Conversion time: 104uS

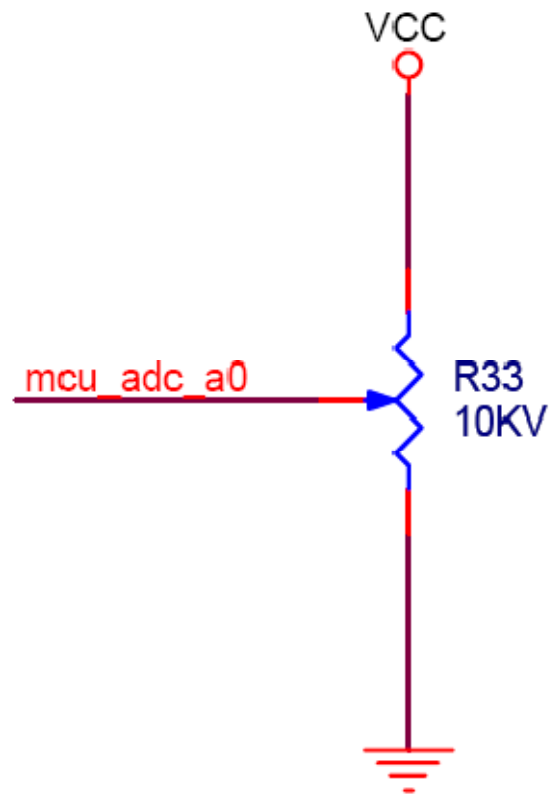
☐ Conversion complete interrupt

Ok Options Preview Cancel

[ADC 실습]

■ 가변 저항

Variable Register



[ADC 실습



- 가변 저항

```
void main(){
```

```
    volatile unsigned int ch0,ah0,al0;
```

```
    init_devices();
```

```
    ktm128_init();
```

```
    while(1){
```

```
        ADCSRA &= 0x7f;
```

```
        ADMUX = 0x40;
```

```
        ADCSRA |= 0xC0;
```

```
        al0 = ADCL;
```

```
        ah0 = ADCH;
```

```
        ch0 += al0+ah0*256;
```

```
        printf( "data : %d|r|n", ch0);
```

```
        delay(500);
```

```
        ch0 = 0;
```

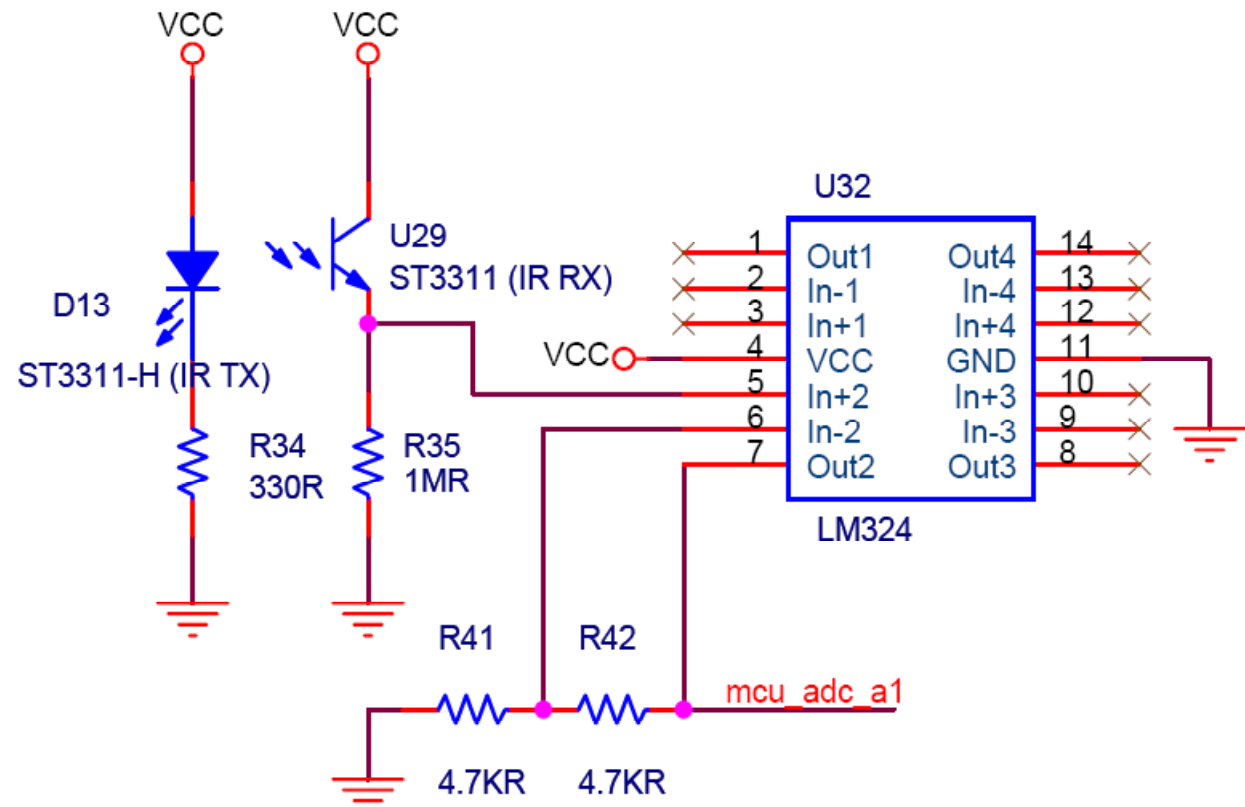
```
    }
```

```
}
```

[ADC 실습]

■ 적외선 센서

IR Sensor



[ADC 실습



- 적외선 센서

```
void main(){
```

```
    volatile unsigned int ch0,ah0,al0;
```

```
    init_devices();
```

```
    ktm128_init();
```

```
    while(1){
```

```
        ADCSRA &= 0x7f;
```

```
        ADMUX = 0x41;
```

```
        ADCSRA |= 0xC0;
```

```
        al0 = ADCL;
```

```
        ah0 = ADCH;
```

```
        ch0 += al0+ah0*256;
```

```
        printf( "data : %d|r|n", ch0);
```

```
        delay(500);
```

```
        ch0 = 0;
```

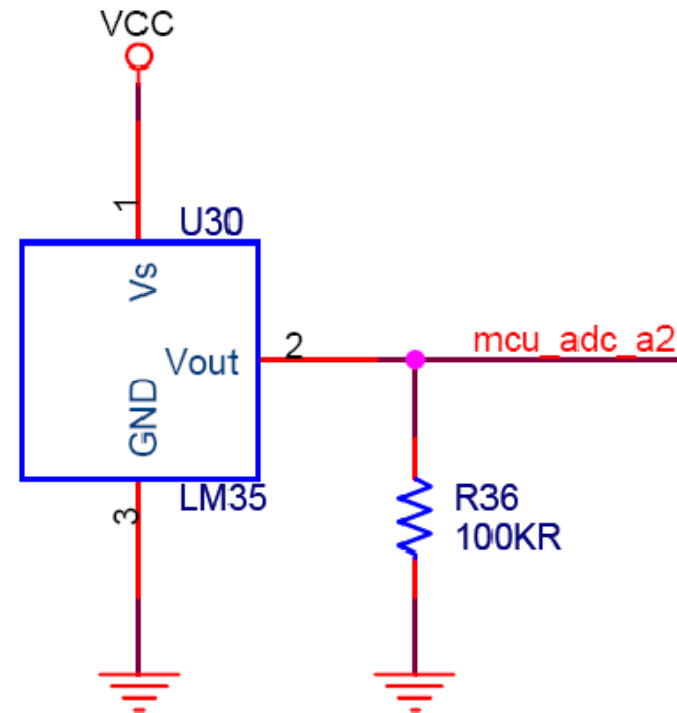
```
    }
```

```
}
```

[ADC 실습]

■ 온도 센서

Temp Sensor



[ADC 실습

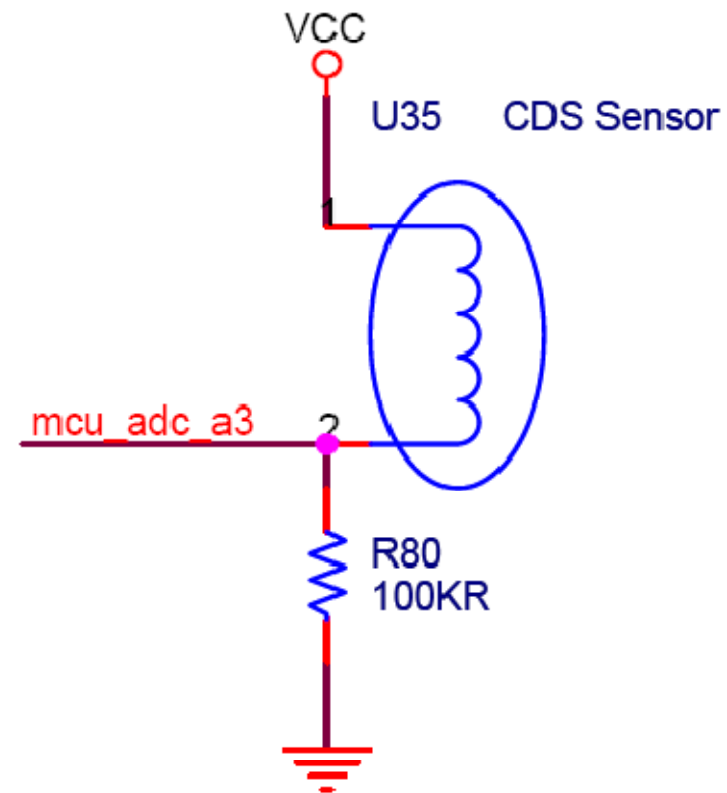


- 온도 센서

```
void main(){  
    volatile unsigned int ch0,ah0,al0;  
  
    init_devices();  
    ktm128_init();  
  
    while(1){  
        ADCSRA &= 0x7f;  
        ADMUX = 0x42;  
        ADCSRA |= 0xC0;  
  
        al0 = ADCL;  
        ah0 = ADCH;  
        ch0 += al0+ah0*256;  
  
        printf("data : %d|r|n", ch0);  
        delay(500);  
        ch0 = 0;  
    }  
}
```

[ADC 실습]

■ 조도 센서



[ADC 실습



- 조도 센서

```
void main(){  
    volatile unsigned int ch0,ah0,a10;  
  
    init_devices();  
    ktm128_init();  
  
    while(1){  
        ADCSRA &= 0x7f;  
        ADMUX = 0x43;  
        ADCSRA |= 0xC0;  
  
        a10 = ADCL;  
        ah0 = ADCH;  
        ch0 += a10+ah0*256;  
  
        printf("data : %d|r|n", ch0);  
        delay(500);  
        ch0 = 0;  
    }  
}
```

[문제

]

- 각 센서(4개) 값을 터미널을 이용하여 출력 할 수 있도록 만드시오.
- 센서 값 중 조도 센서 값을 FND를 통하여 출력 할 수 있도록 만드시오.