



AVR - HW2

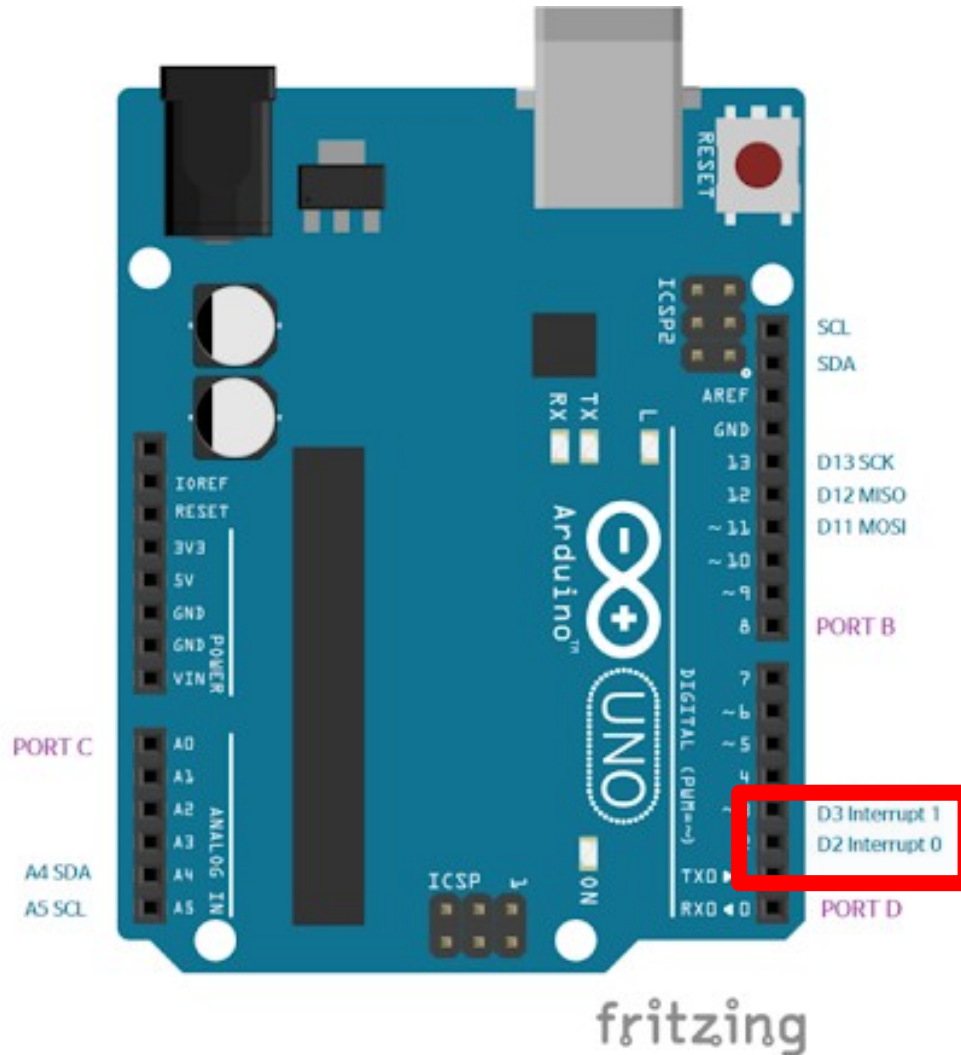
임베디드스쿨1기

Lv1과정

2020. 09. 25

김인겸

1. 아두이노 우노 인터럽트 핀맵



이 부분이 interrupt와 관련된 핀이다.
아두이노 우노는 2개의 인터럽트를
제어할 수 있다.

2. 인터럽트 발생과정

1. 인터럽트 발생 시 현재 진행 중인 명령어까지 처리를 완료한다.
2. PC(Program Counter)를 메모리의 스택에 저장한다
3. 인터럽트 벡터가 있는 프로그램메모리 상단으로 점프한다
4. 인터럽트 벡터를 통해 ISR(Interrupt Service Routine)으로 점프해서 인터럽트를 처리한다
5. 인터럽트 처리 후 메모리의 스택에 저장했던 PC를 꺼내와서 인터럽트 처리 전의 작업을 다시 수행한다.

2. 인터럽트 발생과정

1. 인터럽트 벡터 테이블은 플래시 프로그램메모리 상단에 고정적으로 존재한다.

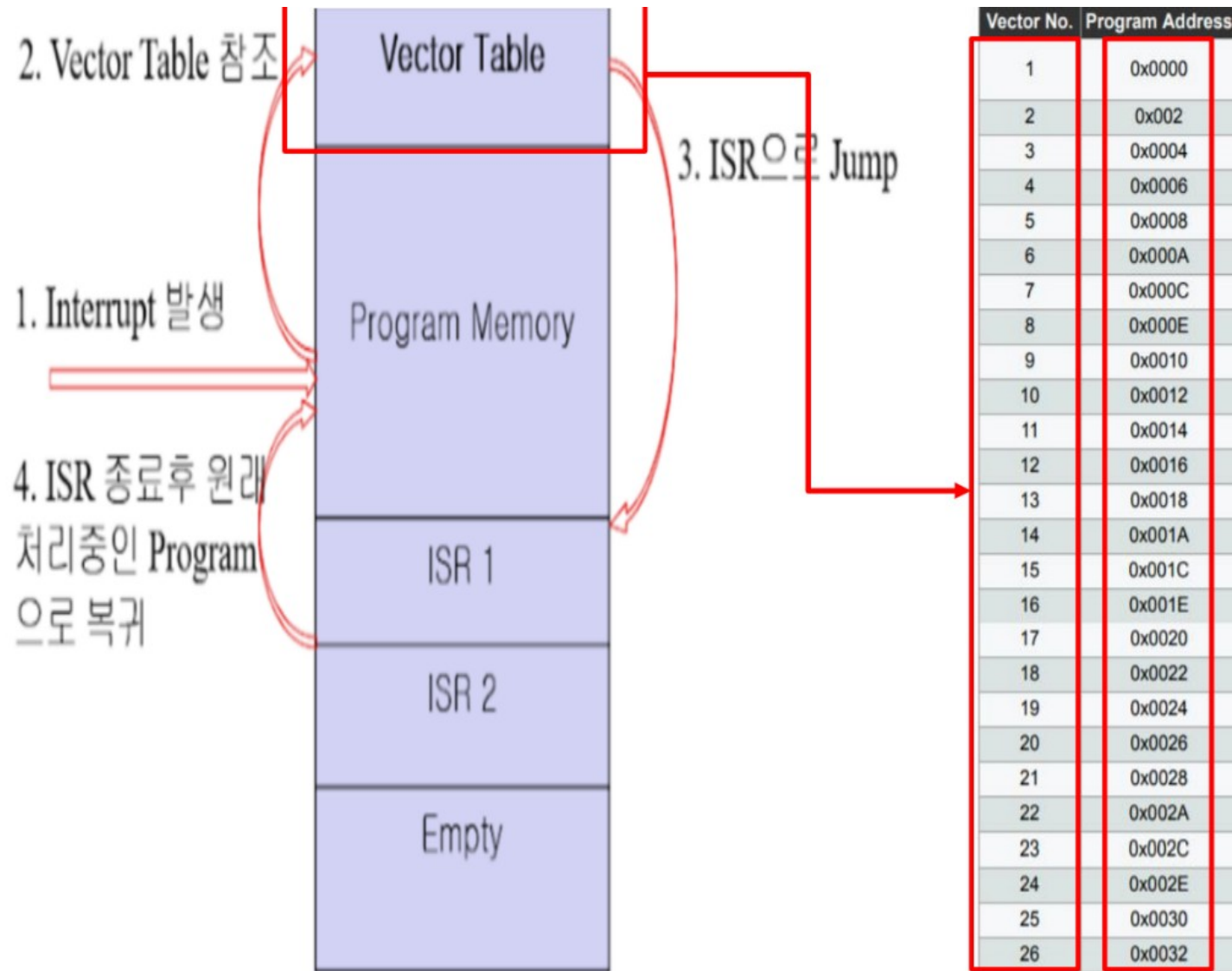
Table 11-1. Reset and Interrupt Vectors in ATmega328P

| Vector No. | Program Address | Source | Interrupt Definition |
|------------|-----------------|--------------|---|
| 1 | 0x0000 | RESET | External pin, power-on reset, brown-out reset and watchdog system reset |
| 2 | 0x0002 | INT0 | External interrupt request 0 |
| 3 | 0x0004 | INT1 | External interrupt request 1 |
| 4 | 0x0006 | PCINT0 | Pin change interrupt request 0 |
| 5 | 0x0008 | PCINT1 | Pin change interrupt request 1 |
| 6 | 0x000A | PCINT2 | Pin change interrupt request 2 |
| 7 | 0x000C | WDT | Watchdog time-out interrupt |
| 8 | 0x000E | TIMER2 COMPA | Timer/Counter2 compare match A |
| 9 | 0x0010 | TIMER2 COMPB | Timer/Counter2 compare match B |
| 10 | 0x0012 | TIMER2 OVF | Timer/Counter2 overflow |
| 11 | 0x0014 | TIMER1 CAPT | Timer/Counter1 capture event |
| 12 | 0x0016 | TIMER1 COMPA | Timer/Counter1 compare match A |
| 13 | 0x0018 | TIMER1 COMPB | Timer/Counter1 compare match B |
| 14 | 0x001A | TIMER1 OVF | Timer/Counter1 overflow |
| 15 | 0x001C | TIMER0 COMPA | Timer/Counter0 compare match A |
| 16 | 0x001E | TIMER0 COMPB | Timer/Counter0 compare match B |
| 17 | 0x0020 | TIMER0 OVF | Timer/Counter0 overflow |
| 18 | 0x0022 | SPI, STC | SPI serial transfer complete |

```
0x0002      jmp     EXT_INT0      ; IRQ0 Handler
0x0004      jmp     EXT_INT1      ; IRQ1 Handler
```

점프 명령어로 구성되어 있음을 확인할 수 있다.
2Byte씩 차이가 나는 이유 : jmp명령어(16비트),
인터럽트 서비스루틴을 실행할 주소(EXT_INT0, 16비트)
라고 추측해봄

2. 인터럽트 발생과정



표훈님 자료 참고 굳굳

3. 인터럽트 관련 레지스터

1. SREG – AVR Status Register

The AVR status register – SREG – is defined as:

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---------------|----------|----------|----------|----------|----------|----------|----------|----------|------|
| 0x3F (0x5F) | I | T | H | S | V | N | Z | C | SREG |
| 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 | |

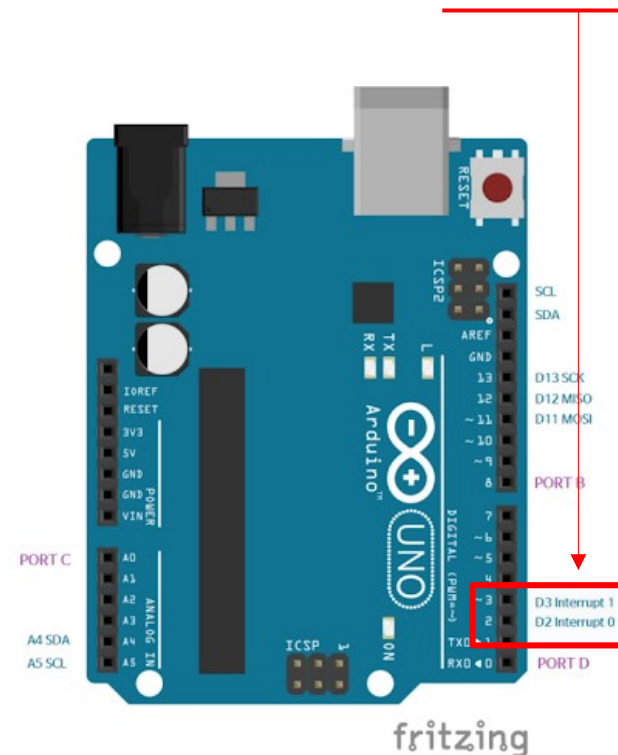
I비트를 활성화 시켜줘야 모든 인터럽트 기능을 허용한다.
반대로 I비트가 0이면 모든 인터럽트 기능을 차단한다.

3. 인터럽트 관련 레지스터

2. EIMSK – External Interrupt Mask Register

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---------------|---|---|---|---|---|---|------|------|-------|
| 0x1D (0x3D) | – | – | – | – | – | – | INT1 | INT0 | EIMSK |
| Read/Write | R | R | R | R | R | R | R/W | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

어느 핀의 인터럽트를 활성화 시킬지 결정
1일 경우 인터럽트 허용.
0일 경우 인터럽트 차단.



3. 인터럽트 관련 레지스터

3. EICRA – External Interrupt Control Register A

The external interrupt control register A contains control bits for interrupt sense control.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---------------|---|---|---|---|-------|-------|-------|-------|-------|
| (0x69) | – | – | – | – | ISC11 | ISC10 | ISC01 | ISC00 | EICRA |
| Read/Write | R | R | R | R | R/W | R/W | R/W | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

ISC(Interrupt Sence Control)

: 인터럽트를 어떤 시점에서 잡을지를 결정한다.

| ISC01 | ISC00 | INT0핀 |
|-------|-------|--------------|
| 0 | 0 | Low level |
| 0 | 1 | 모든 논리적인 변화 |
| 1 | 0 | Falling edge |
| 1 | 1 | Rising edge |

| ISC11 | ISC10 | INT1핀 |
|-------|-------|--------------|
| 0 | 0 | Low level |
| 0 | 1 | 모든 논리적인 변화 |
| 1 | 0 | Falling edge |
| 1 | 1 | Rising edge |

* pull-up저항이 연결될 경우 스위치 on → low level
스위치 off → high level

* 모든 논리적인 변화 : high → low
low → high

3. 인터럽트 관련 레지스터

4.(덜 중요)

EIFR – External Interrupt Flag Register

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---------------|---|---|---|---|---|---|-------|-------|------|
| 0x1C (0x3C) | – | – | – | – | – | – | INTF1 | INTF0 | EIFR |
| Read/Write | R | R | R | R | R | R | R/W | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

프로그램 수행 도중 인터럽트 요청이 발생할 경우 해당 장치의 인터럽트 플래그가 세트 된다.
인터럽트 서비스 루틴으로 점프하게 되면 클리어 된다.
인터럽트의 발생상황을 확인할 수 있다.

3. 인터럽트 관련 레지스터(추가)

추가로 인터럽트를 사용하고 싶을 경우

PCICR – Pin Change Interrupt Control Register

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---------------|---|---|---|---|---|-------|-------|-------|-------|
| (0x68) | – | – | – | – | – | PCIE2 | PCIE1 | PCIE0 | PCICR |
| Read/Write | R | R | R | R | R | R/W | R/W | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

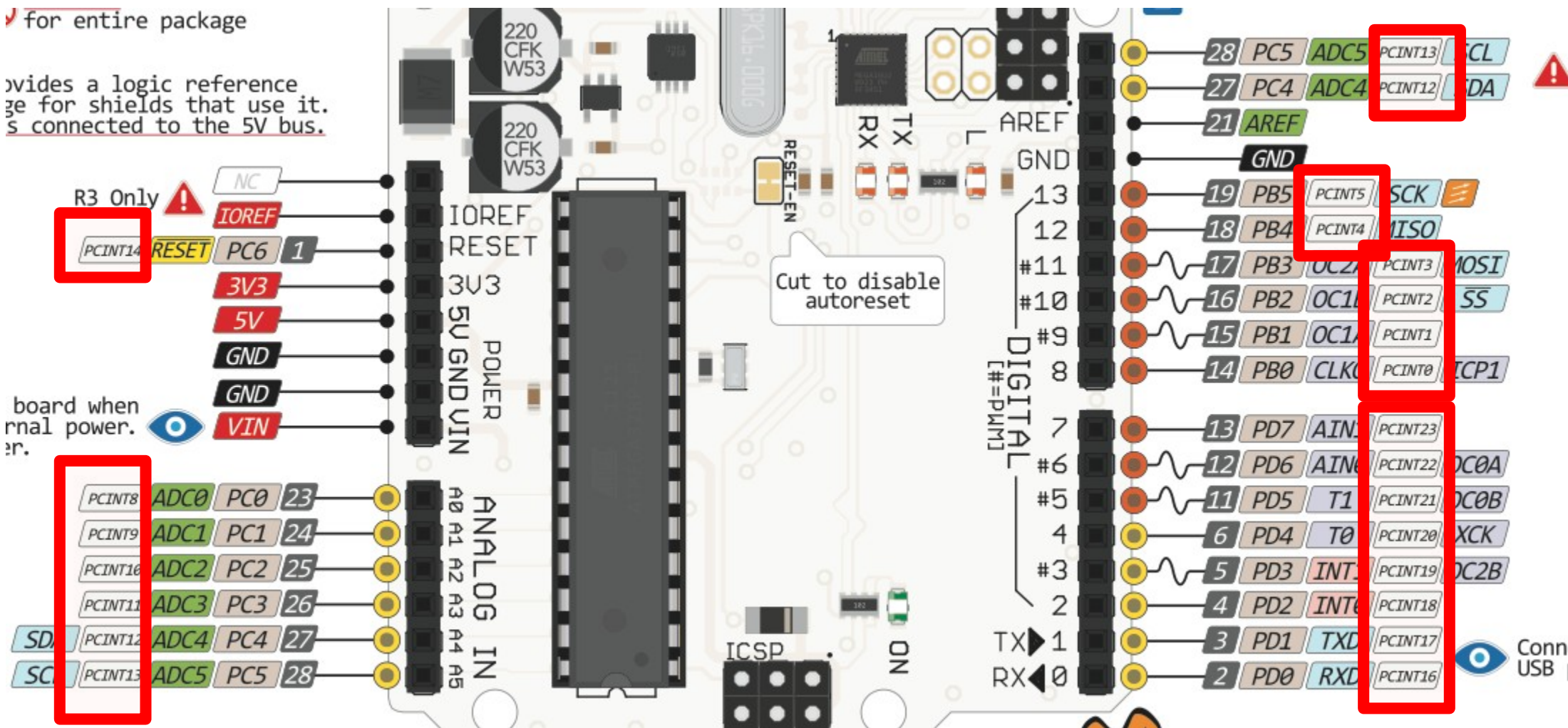
| | |
|-------|----------------|
| PCIE2 | PCINT23~16 활성화 |
| PCIE1 | PCINT 14~8 활성화 |
| PCIE0 | PCINT 7~0 활성화 |

PCIFR – Pin Change Interrupt Flag Register

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---------------|---|---|---|---|---|-------|-------|-------|-------|
| 0x1B (0x3B) | – | – | – | – | – | PCIF2 | PCIF1 | PCIF0 | PCIFR |
| Read/Write | R | R | R | R | R | R/W | R/W | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

PCINT0~23 위치를 나타내는 아두이노 우노 핀맵

PCINT0~23 위치를 나타내는 아두이노 우노 핀맵



3. 인터럽트 관련 레지스터(추가)

PCMSK2 – Pin Change Mask Register 2

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---------------|---------|---------|---------|---------|---------|---------|---------|---------|--------|
| (0x6D) | PCINT23 | PCINT22 | PCINT21 | PCINT20 | PCINT19 | PCINT18 | PCINT17 | PCINT16 | PCMSK2 |
| 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 | |

PCMSK1 – Pin Change Mask Register 1

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---------------|---|---------|---------|---------|---------|---------|--------|--------|--------|
| (0x6C) | – | PCINT14 | PCINT13 | PCINT12 | PCINT11 | PCINT10 | PCINT9 | PCINT8 | PCMSK1 |
| Read/Write | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

PCMSK0 – Pin Change Mask Register 0

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---------------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| (0x6B) | PCINT7 | PCINT6 | PCINT5 | PCINT4 | PCINT3 | PCINT2 | PCINT1 | PCINT0 | PCMSK0 |
| 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 | |

3. 인터럽트 관련 레지스터(추가)

Table 11-1. Reset and Interrupt Vectors in ATmega328P

| Vector No. | Program Address | Source | Interrupt Definition |
|------------|-----------------|--------------|---|
| 1 | 0x0000 | RESET | External pin, power-on reset, brown-out reset and watchdog system reset |
| 2 | 0x0002 | INT0 | External interrupt request 0 |
| 3 | 0x0004 | INT1 | External interrupt request 1 |
| 4 | 0x0006 | PCINT0 | Pin change interrupt request 0 |
| 5 | 0x0008 | PCINT1 | Pin change interrupt request 1 |
| 6 | 0x000A | PCINT2 | Pin change interrupt request 2 |
| 7 | 0x000C | WDI | Watchdog time-out interrupt |
| 8 | 0x000E | TIMER2 COMPA | Timer/Counter2 compare match A |
| 9 | 0x0010 | TIMER2 COMPB | Timer/Counter2 compare match B |
| 10 | 0x0012 | TIMER2 OVF | Timer/Counter2 overflow |
| 11 | 0x0014 | TIMER1 CAPT | Timer/Counter1 capture event |
| 12 | 0x0016 | TIMER1 COMPA | Timer/Counter1 compare match A |
| 13 | 0x0018 | TIMER1 COMPB | Timer/Counter1 compare match B |
| 14 | 0x001A | TIMER1 OVF | Timer/Counter1 overflow |
| 15 | 0x001C | TIMER0 COMPA | Timer/Counter0 compare match A |
| 16 | 0x001E | TIMER0 COMPB | Timer/Counter0 compare match B |
| 17 | 0x0020 | TIMER0 OVF | Timer/Counter0 overflow |
| 18 | 0x0022 | SPI, STC | SPI serial transfer complete |

공용으로 사용??

rising / falling에서 모두 인터럽트 발생??

예제 코드

```
#define F_CPU 16000000UL
#include <avr/io.h>
#include <util/delay.h>
#include <avr/interrupt.h>

#define sbi(PORTX, BitX)    ( PORTX |= (1 << BitX) )
#define cbi(PORTX, BitX)    ( PORTX &= ~(1 << BitX) )

SIGNAL (INT1_vect){

    PORTB = 0X20;

    _delay_ms(200);

}

int main(void)
{
    sbi(SREG, 7);
    sbi(EIMSK, INT1);
    EICRA = 0X00;
    DDRB = 0X20;
    DDRD = 0X00;
    PORTD = 0Xff;

    while (1)
    {
        PORTB = 0X00;
    }
}
```


ADC

ADC(Analog / Digital Converter)

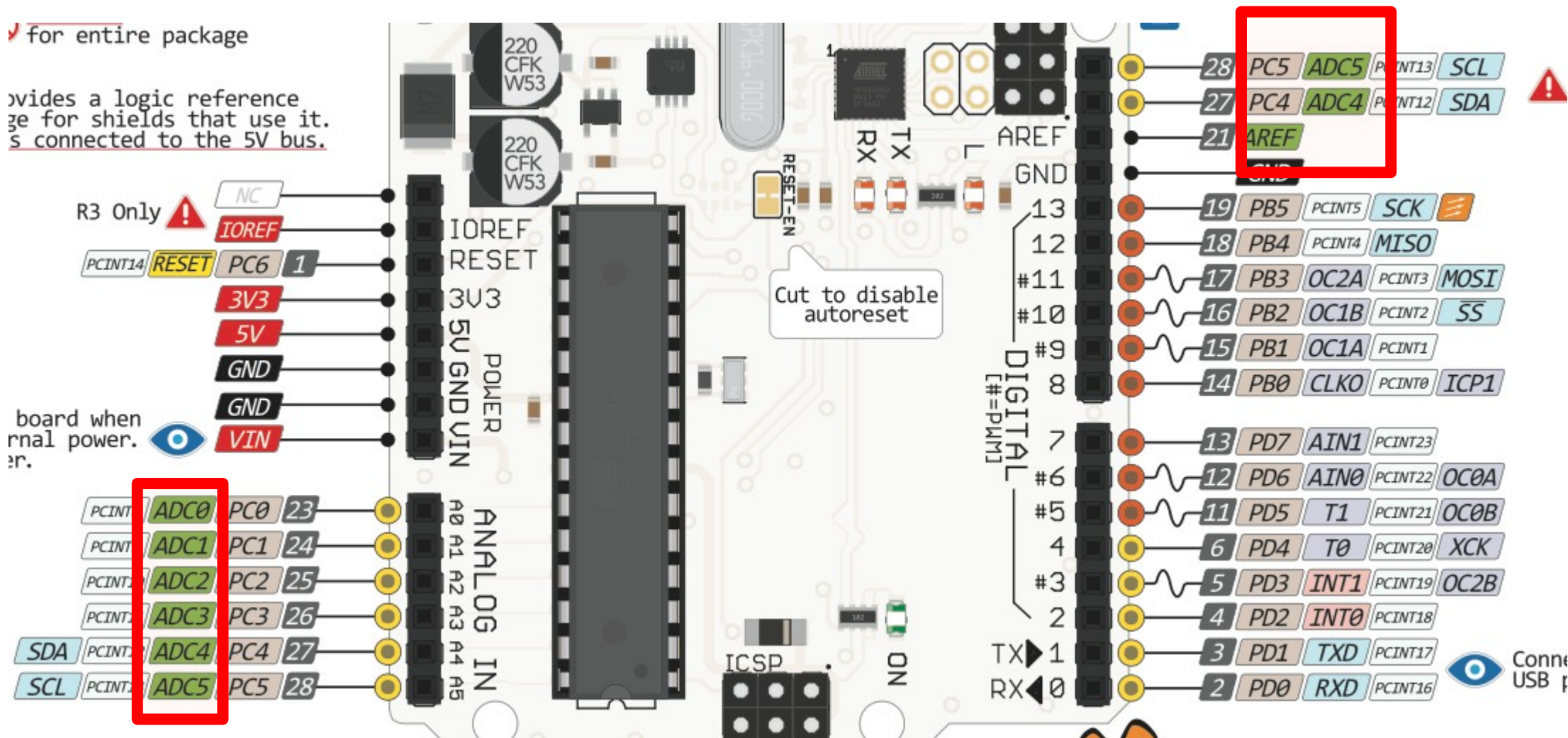
Features

- 10-bit resolution 10비트 분해능
- 0.5 LSB integral non-linearity
- ± 2 LSB absolute accuracy 65~260us 변환 시간
- 65 to 260 μ s conversion time
- Up to 15kSPS 6채널의 멀티플렉스된 단일 입력
- 6 multiplexed single ended input channels
- 2 additional multiplexed single ended input channels 온도센서 입력 채널
- Temperature sensor input channel
- Optional left adjustment for ADC result readout 0~V_{CC}ADC입력 전압 범위
- 0 to V_{CC} ADC input voltage range
- Selectable 1.1V ADC reference voltage
- Free running or single conversion mode
- Interrupt on ADC conversion complete
- Sleep mode noise canceler

ADC 관련 아두이노 우노 핀맵

for entire package

Provides a logic reference
for shields that use it.
is connected to the 5V bus.



ADC 관련 레지스터

1. ADMUX – ADC Multiplexer Selection Register

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---------------|-------------|---|-------|---|---------------------|---|---|---|-------|
| (0x7C) | REFS1 REFS0 | | ADLAR | – | MUX3 MUX2 MUX1 MUX0 | | | | ADMUX |
| Read/Write | R/W | | R/W | R | R/W | | | | |
| Initial Value | 0 | | 0 | 0 | 0 | | | | |

ADC에서 사용하는 기준 전압 선택

| REFS1 | REFS2 | |
|-------|-------|---------------|
| 0 | 0 | AREF의 입력전압 사용 |
| 0 | 1 | AVCC의 입력전압 사용 |
| 1 | 0 | 예약됨 |
| 1 | 1 | 내부 1.1V 사용 |

A/D 컨버터의 입력 채널을 선택

| MUX3..0 | Single Ended Input |
|---------|-------------------------|
| 0000 | ADC0 |
| 0001 | ADC1 |
| 0010 | ADC2 |
| 0011 | ADC3 |
| 0100 | ADC4 |
| 0101 | ADC5 |
| 0110 | ADC6 |
| 0111 | ADC7 |
| 1000 | ADC8 ⁽¹⁾ |
| 1001 | (reserved) |
| 1010 | (reserved) |
| 1011 | (reserved) |
| 1100 | (reserved) |
| 1101 | (reserved) |
| 1110 | 1.1V (V _{BG}) |
| 1111 | 0V (GND) |

포기하면 얻는 건 아무것도 없다.

ADC 관련 레지스터

1. ADMUX – ADC Multiplexer Selection Register

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---------------|-------|-------|-------|---|------|------|------|------|-------|
| (0x7C) | REFS1 | REFS0 | ADLAR | – | MUX3 | MUX2 | MUX1 | MUX0 | ADMUX |
| Read/Write | R/W | R/W | R/W | R | R/W | R/W | R/W | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

ADCH, ADCL 레지스터
우측 정렬

1 ADLAR = 0

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | |
|---------------|------|------|------|------|------|------|------|------|------|
| (0x79) | – | – | – | – | – | – | ADC9 | ADC8 | ADCH |
| (0x78) | ADC7 | ADC6 | ADC5 | ADC4 | ADC3 | ADC2 | ADC1 | ADC0 | ADCL |
| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| Read/Write | R | R | R | R | R | R | R | R | |
| | R | R | R | R | R | R | R | R | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

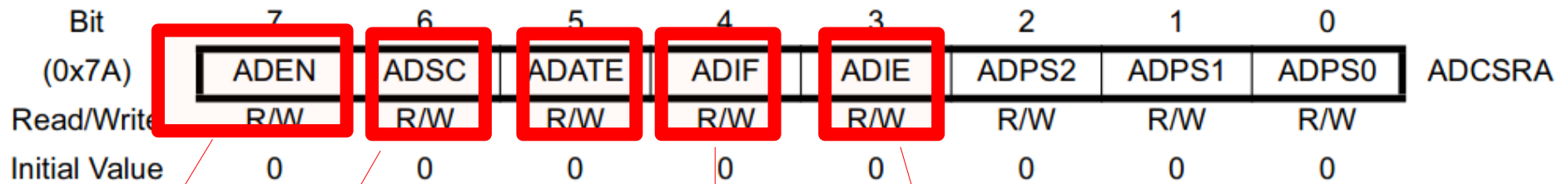
ADCH, ADCL 레지스터
좌측 정렬

2 ADLAR = 1

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | |
|---------------|------|------|------|------|------|------|------|------|------|
| (0x79) | ADC9 | ADC8 | ADC7 | ADC6 | ADC5 | ADC4 | ADC3 | ADC2 | ADCH |
| (0x78) | ADC1 | ADC0 | – | – | – | – | – | – | ADCL |
| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| Read/Write | R | R | R | R | R | R | R | R | |
| | R | R | R | R | R | R | R | R | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

ADC 관련 레지스터

2. ADCSRA – ADC Control and Status Register A



1 : ADC 활성화
0 : ADC 비활성화

ADC Interrupt Flag
: 변환이 완료되면 1로
설정되면서 인터럽트를 요청함.
인터럽트 벡터가 실행되면
지워짐.

ADC Interrupt Enable
: SREF의 I비트가 활성화 되고
이 비트가 활성화 되면 ADC완료 인터럽트가
활성화됨(뒷페이지 참고)

ADC Start Conversion
단일 변환 모드에서 1로 설정하면 변환을
시작. 변환 완료되면 0으로 설정됨(?)
프리러닝 모드에서 1로 설정하면 첫 번째
변환을 시작하고 그 다음부터는 자동으로
변환이 반복됨.

*Atmega128은 5번째 비트가 ADFR(ADC Free
Running) 인데 328p는 좀 다른가 보다

ADC 관련 레지스터

ADC 변환이 완료될 때
실행되는 인터럽트를
인터럽트 벡터 테이블에서
확인할 수 있다.

| Vector No. | Program Address | Source | Interrupt Definition |
|------------|-----------------|--------------|---|
| 1 | 0x0000 | RESET | External pin, power-on reset, brown-out reset and watchdog system reset |
| 2 | 0x0002 | INT0 | External interrupt request 0 |
| 3 | 0x0004 | INT1 | External interrupt request 1 |
| 4 | 0x0006 | PCINT0 | Pin change interrupt request 0 |
| 5 | 0x0008 | PCINT1 | Pin change interrupt request 1 |
| 6 | 0x000A | PCINT2 | Pin change interrupt request 2 |
| 7 | 0x000C | WDT | Watchdog time-out interrupt |
| 8 | 0x000E | TIMER2 COMPA | Timer/Counter2 compare match A |
| 9 | 0x0010 | TIMER2 COMPB | Timer/Counter2 compare match B |
| 10 | 0x0012 | TIMER2 OVF | Timer/Counter2 overflow |
| 11 | 0x0014 | TIMER1 CAPT | Timer/Counter1 capture event |
| 12 | 0x0016 | TIMER1 COMPA | Timer/Counter1 compare match A |
| 13 | 0x0018 | TIMER1 COMPB | Timer/Counter1 compare match B |
| 14 | 0x001A | TIMER1 OVF | Timer/Counter1 overflow |
| 15 | 0x001C | TIMER0 COMPA | Timer/Counter0 compare match A |
| 16 | 0x001E | TIMER0 COMPB | Timer/Counter0 compare match B |
| 17 | 0x0020 | TIMER0 OVF | Timer/Counter0 overflow |
| 18 | 0x0022 | SPI, STC | SPI serial transfer complete |
| 19 | 0x0024 | USART, RX | USART Rx complete |
| 20 | 0x0026 | USART, UDRE | USART, data register empty |
| 21 | 0x0028 | USART, TX | USART Tx complete |
| 22 | 0x002A | ADC | ADC conversion complete |
| 23 | 0x002C | EE READY | EEPROM ready |
| 24 | 0x002E | ANALOG COMP | Analog comparator |
| 25 | 0x0030 | TWI | 2-wire serial interface |
| 26 | 0x0032 | SPM READY | Store program memory ready |

ADC 관련 레지스터

2. ADCSRA – ADC Control and Status Register A

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---------------|------|------|-------|------|------|-------|-------|-------|--------|
| (0x7A) | ADEN | ADSC | ADATE | 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 | |

Table 23-5. ADC Prescaler Selections

| 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 |

: 시스템 클록 주파수와 ADC에 대한 입력 클록 간의 비율을 결정함

- ADC는 50kHz ~ 200kHz 사이에서 동작함

- Atmega328p의 cpu 클록은 16MHz이므로 Division Factor를 64로 설정하면(1 1 0)
ADC에 대한 입력 클록이 $16\text{MHz}/64 = 250\text{kHz}$ 가 된다.

ADC 관련 레지스터

3. ADCH(ADC data Register High): A/D 컨버터의 변환 결과를 저장하는 레지스터.
 ADCL(ADC data Register Low) - 10비트의 분해능 성능을 가지므로 10비트가 필요하다.

1 ADLAR = 0

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | |
|---------------|------|------|------|------|------|------|------|------|------|
| (0x79) | — | — | — | — | — | — | ADC9 | ADC8 | ADCH |
| (0x78) | ADC7 | ADC6 | ADC5 | ADC4 | ADC3 | ADC2 | ADC1 | ADC0 | ADCL |
| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| Read/Write | R | R | R | R | R | R | R | R | |
| | R | R | R | R | R | R | R | R | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

2 ADLAR = 1

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | |
|---------------|------|------|------|------|------|------|------|------|------|
| (0x79) | ADC9 | ADC8 | ADC7 | ADC6 | ADC5 | ADC4 | ADC3 | ADC2 | ADCH |
| (0x78) | ADC1 | ADC0 | — | — | — | — | — | — | ADCL |
| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| Read/Write | R | R | R | R | R | R | R | R | |
| | R | R | R | R | R | R | R | R | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

USART

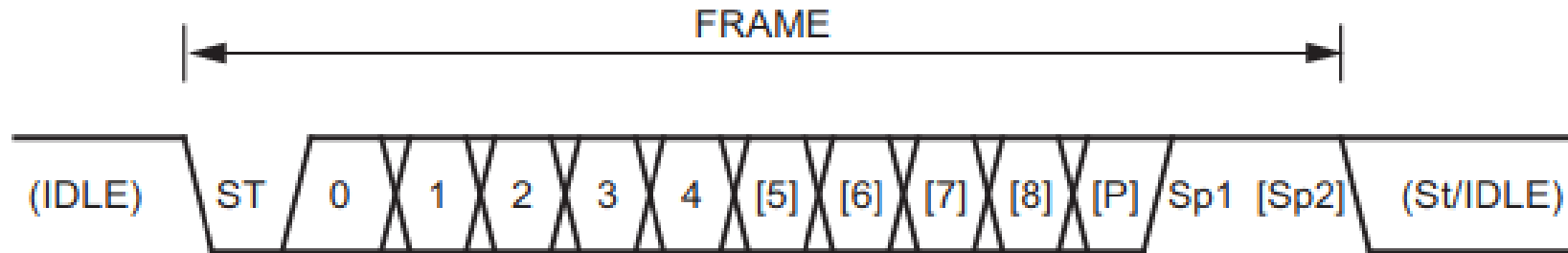
universal synchronous/asynchronous receiver transmitter

범용 동기 비동기 수신기 송신기

Features

- Full duplex operation (independent serial receive and transmit registers) 전이중 작동
- Asynchronous or synchronous operation 비동기 또는 동기 작동
- Master or slave clocked synchronous operation
- High resolution baud rate generator
- Supports serial frames with 5, 6, 7, 8, or 9 data bits and 1 or 2 stop bits 패리티 생성 및 검출 가능
- Odd or even parity generation and parity check supported by hardware
- Data overrun detection
- Framing error detection
- Noise filtering includes false start bit detection and digital low pass filter
- Three separate interrupts on TX complete, TX data register empty and RX complete
- Multi-processor communication mode
- Double speed asynchronous communication mode

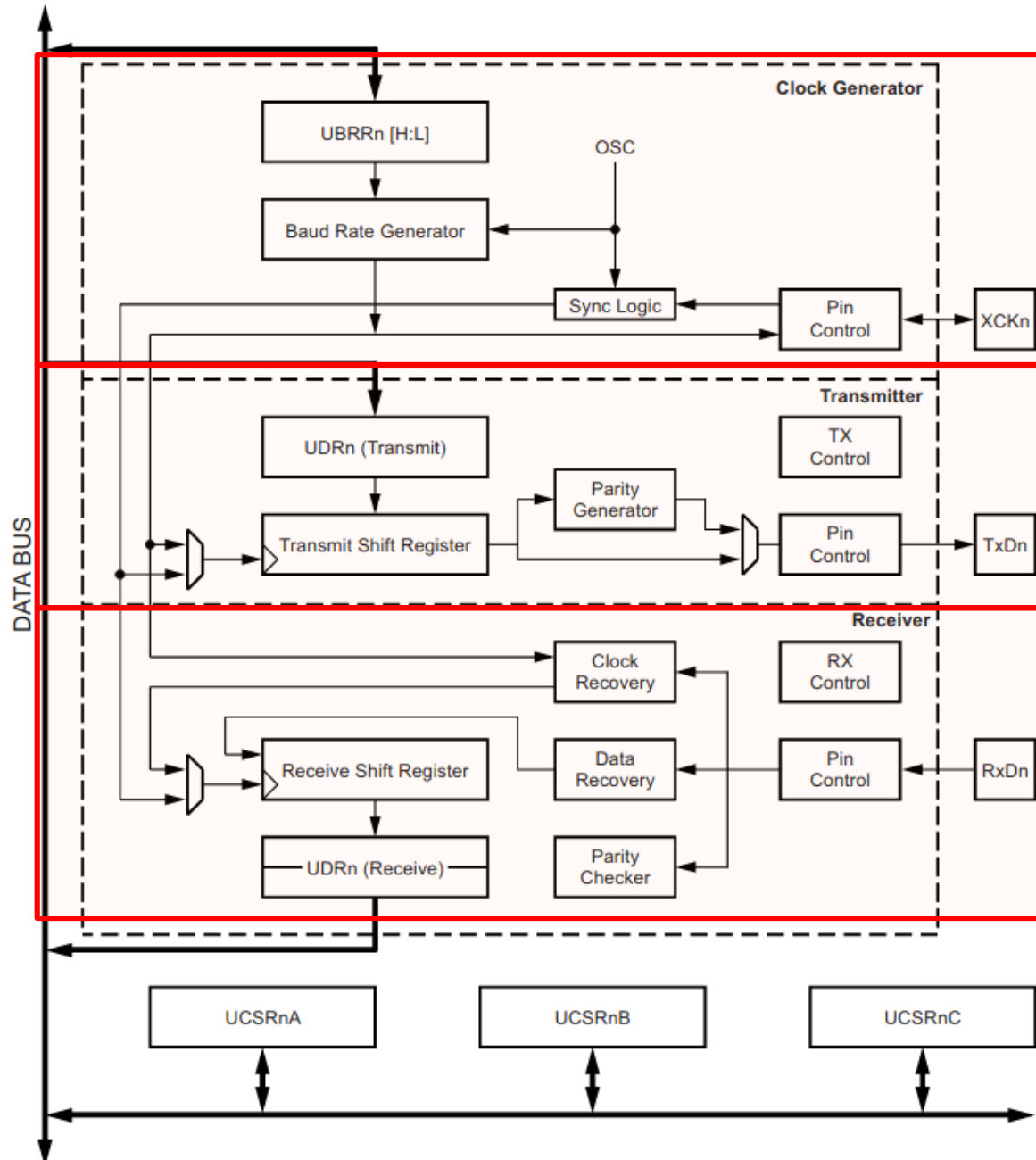
USART 데이터 프레임



- Start (시작 비트) : 항상 LOW이며 데이터의 시작을 알림
- 0~8 (데이터 비트) : 전송되는 데이터 비트이며 5,6,7,8,9bit로 설정할 수 있다.
- Parity (패리티 비트) : 오류 검출 기능, 사용할 수도 안 할 수도, 홀수 패리티 또는 짝수 패리티.
- Stop (정지 비트) : 항상 HIGH이며 데이터의 끝을 알림

USART

USART Block Diagram⁽¹⁾



클럭 발생부 - 일반 비동기 모드
- 2배속 비동기 모드
- 마스터 동기 모드
- 슬레이브 동기 모드

송신부

수신부

USART 관련 인터럽트 세 가지

| Vector No. | Program Address | Source | Interrupt Definition |
|------------|-----------------|--------------|---|
| 1 | 0x0000 | RESET | External pin, power-on reset, brown-out reset and watchdog system reset |
| 2 | 0x0002 | INT0 | External interrupt request 0 |
| 3 | 0x0004 | INT1 | External interrupt request 1 |
| 4 | 0x0006 | PCINT0 | Pin change interrupt request 0 |
| 5 | 0x0008 | PCINT1 | Pin change interrupt request 1 |
| 6 | 0x000A | PCINT2 | Pin change interrupt request 2 |
| 7 | 0x000C | WDT | Watchdog time-out interrupt |
| 8 | 0x000E | TIMER2 COMPA | Timer/Counter2 compare match A |
| 9 | 0x0010 | TIMER2 COMPB | Timer/Counter2 compare match B |
| 10 | 0x0012 | TIMER2 OVF | Timer/Counter2 overflow |
| 11 | 0x0014 | TIMER1 CAPT | Timer/Counter1 capture event |
| 12 | 0x0016 | TIMER1 COMPA | Timer/Counter1 compare match A |
| 13 | 0x0018 | TIMER1 COMPB | Timer/Counter1 compare match B |
| 14 | 0x001A | TIMER1 OVF | Timer/Counter1 overflow |
| 15 | 0x001C | TIMER0 COMPA | Timer/Counter0 compare match A |
| 16 | 0x001E | TIMER0 COMPB | Timer/Counter0 compare match B |
| 17 | 0x0020 | TIMER0 OVF | Timer/Counter0 overflow |
| 18 | 0x0022 | SPI STC | SPI serial transfer complete |
| 19 | 0x0024 | USART, RX | USART Rx complete |
| 20 | 0x0026 | USART, UDRE | USART, data register empty |
| 21 | 0x0028 | USART, TX | USART, Tx complete |
| 22 | 0x002A | ADC | ADC conversion complete |
| 23 | 0x002C | EE READY | EEPROM ready |
| 24 | 0x002E | ANALOG COMP | Analog comparator |
| 25 | 0x0030 | TWI | 2-wire serial interface |
| 26 | 0x0032 | SPM READY | Store program memory ready |

USART 관련 레지스터

1. UDRn – USART I/O Data Register n

| | | | | | | | | | |
|---------------|----------|-----|-----|-----|-----|-----|-----|-----|--------------|
| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | RXB[7:0] | | | | | | | | UDRn (Read) |
| | TXB[7:0] | | | | | | | | UDRn (Write) |
| 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 | |

USART모듈의 송수신 데이터 버퍼의 기능을 수행하는 레지스터

송신 시 TXB에 데이터를 저장.

수신 시 RXB에 데이터를 저장.

USART 관련 레지스터

2-1. UCSRnA – USART Control and Status Register n A

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---------------|------|------|-------|-----|------|------|------|-------|--------|
| | RXCn | TXCn | UDREN | FEn | DORn | UPEn | U2Xn | MPCMn | UCSRnA |
| Read/Write | R | R/W | R | R | R | R | R/W | R/W | |
| Initial Value | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | |

UDRn의 수신 버퍼를 읽지 않으면 1.
읽으면 0.

1로 세트되면 → 전송속도 2배 증가
(UBRRn레지스터 참고)

USART 관련 레지스터

2-2. UCSRnB – USART Control and Status Register n B

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---------------|--------------------------|--------------------------|--------------------------|-------------------------|-------------------------|--------------------------|-------------------------|-------------------------|--------------------|
| | RXCIE_n | TXCIE_n | UDRIE_n | RXEN_n | TXEN_n | UCSZ_{n2} | RXB8_n | TXB8_n | UCSR _{nB} |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

USART 관련 레지스터

2-3. UCSRnC – USART Control and Status Register n C

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---------------|---------|---------|-------|-------|-------|--------|--------|--------|--------|
| | UMSELn1 | UMSELn0 | UPMn1 | UPMn0 | USBSn | UCSZn1 | UCSZn0 | UCPOLn | UCSRnC |
| 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 | 1 | 1 | 0 | |

| UMSELn1 | UMSELn0 | Mode |
|---------|---------|-----------------------------------|
| 0 | 0 | Asynchronous USART |
| 0 | 1 | Synchronous USART |
| 1 | 0 | (Reserved) |
| 1 | 1 | Master SPI (MSPIM) ⁽¹⁾ |

USART 통신모드 설정

Table 19-5. UPMn Bits Settings

| UPMn1 | UPMn0 | Parity Mode |
|-------|-------|----------------------|
| 0 | 0 | Disabled |
| 0 | 1 | Reserved |
| 1 | 0 | Enabled, even parity |
| 1 | 1 | Enabled, odd parity |

패리티 설정

| USBSn | Stop Bit(s) |
|-------|-------------|
| 0 | 1-bit |
| 1 | 2-bit |

Stop비트 설정

USART 관련 레지스터

2-3. UCSRnC – USART Control and Status Register n C

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---------------|---------|---------|-------|-------|-------|--------|--------|--------|--------|
| | UMSELn1 | UMSELn0 | UPMn1 | UPMn0 | USBSn | UCSZn1 | UCSZn0 | UCPOLn | UCSRnC |
| 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 | 1 | 1 | 0 | |

| UCSZn2 | UCSZn1 | UCSZn0 | Character Size |
|--------|--------|--------|----------------|
| 0 | 0 | 0 | 5-bit |
| 0 | 0 | 1 | 6-bit |
| 0 | 1 | 0 | 7-bit |
| 0 | 1 | 1 | 8-bit |
| 1 | 0 | 0 | Reserved |
| 1 | 0 | 1 | Reserved |
| 1 | 1 | 0 | Reserved |
| 1 | 1 | 1 | 9-bit |

전송 문자의
비트 수를 결정

| UCPOLn | Transmitted Data Changed (Output of TxDn Pin) | Received Data Sampled (Input on RxDn Pin) |
|--------|---|---|
| 0 | Rising XCKn edge | Falling XCKn edge |
| 1 | Falling XCKn edge | Rising XCKn edge |

USART 관련 레지스터

3. UBRRnL and UBRRnH – USART Baud Rate Registers

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | |
|---------------|------------|-----|-----|-----|-------------|-----|-----|-----|--------|
| | – | – | – | – | UBRRn[11:8] | | | | UBRRnH |
| | UBRRn[7:0] | | | | | | | | UBRRnL |
| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| Read/Write | R | R | R | R | R/W | R/W | R/W | R/W | |
| | 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 | |
| | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

USART의 통신 속도(baud rate)를 설정하는 레지스터

bps = baud rate * 한번에 변조되는 비트 수

Table 19-12. Examples of UBRRn Settings for Commonly Used Oscillator Frequencies

| Baud Rate (bps) | $f_{osc} = 16.0000\text{MHz}$ | | | |
|---------------------|-------------------------------|-------|----------|-------|
| | U2Xn = 0 | | U2Xn = 1 | |
| | UBRRn | Error | UBRRn | Error |
| 2400 | 416 | –0.1% | 832 | 0.0% |
| 4800 | 207 | 0.2% | 416 | –0.1% |
| 9600 | 103 | 0.2% | 207 | 0.2% |
| 14.4k | 68 | 0.6% | 138 | –0.1% |
| 19.2k | 51 | 0.2% | 103 | 0.2% |
| 28.8k | 34 | –0.8% | 68 | 0.6% |
| 38.4k | 25 | 0.2% | 51 | 0.2% |
| 57.6k | 16 | 2.1% | 34 | –0.8% |
| 76.8k | 12 | 0.2% | 25 | 0.2% |
| 115.2k | 8 | –3.5% | 16 | 2.1% |
| 230.4k | 3 | 8.5% | 8 | –3.5% |
| 250k | 3 | 0.0% | 7 | 0.0% |
| 0.5M | 1 | 0.0% | 3 | 0.0% |
| 1M | 0 | 0.0% | 1 | 0.0% |
| Max. ⁽¹⁾ | 1Mbps | | 2Mbps | |

Note: 1. UBRRn = 0, error = 0.0%

뭔말인지 잘 모르겠다

아무튼 이 레지스터를
통해서 전송 속도를
조절하나보다

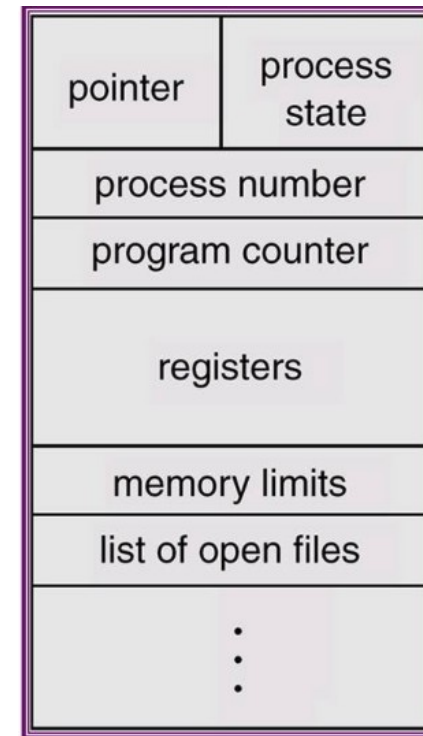
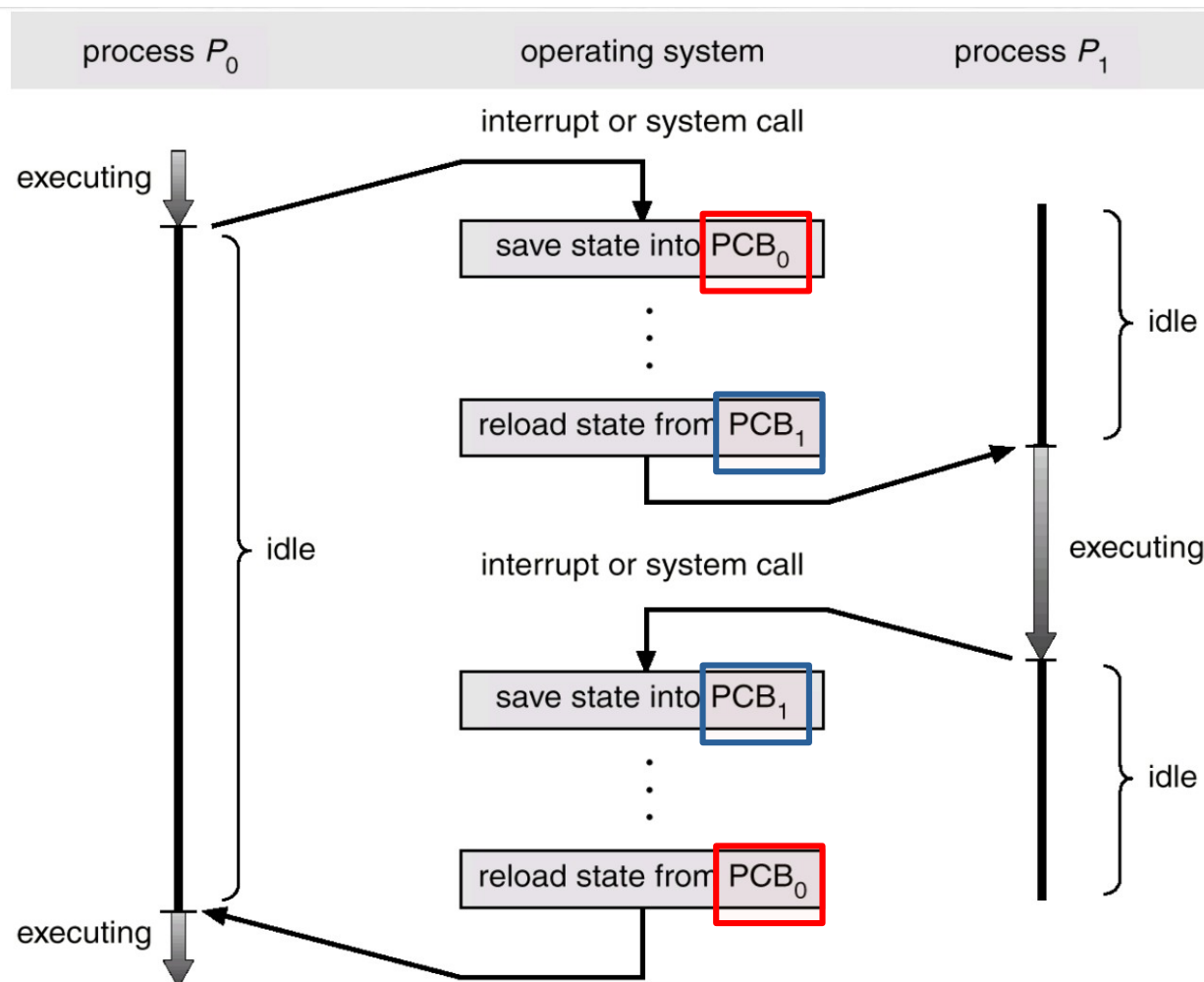
포기하면 얻는 건 아무것도 없다.

Context Switching

1. Context Switching이란?

- 두 개 이상의 프로세스가 진행될 때 os는 스케줄링을 통해 각각의 프로세스에 얼마 만큼의 클럭을 할당 할 지 결정한다.
- 프로세스가 중요할 수록 우선순위가 높아 더 긴 시간 간격으로 실행될 수 있다
- 이렇게 여러 개의 프로세스들이 일정한 시간 간격으로 매우 빠르게 번갈아가며 실행되므로 사용자는 멀티태스킹을 하는 것처럼 느낄 수 있다.
- Context Switching이란 현재 진행하고 있는 프로세스의 레지스터 상태를 PCB(Process Control Block)에 저장하고 다음 진행 할 프로세스의 상태 값을 읽어서 실행하는 과정을 말한다.
- 현재 상태를 PCB에 저장하는 이유는 다른 프로세서가 실행될 때 레지스터 값이 바뀔으로써 현재 상태로 돌아왔을 때 데이터 손실이 발생하기 때문이다.
- 프로세스 사이의 전환 뿐만 아니라 인터럽트와 프로세스 전환에도 context switching이 이루어진다.

Context Switching



PCB구조

Process number : 프로세스는 각각 고유의 번호를 가지고 있다

Program counter : 프로세스가 다음 실행할 주소

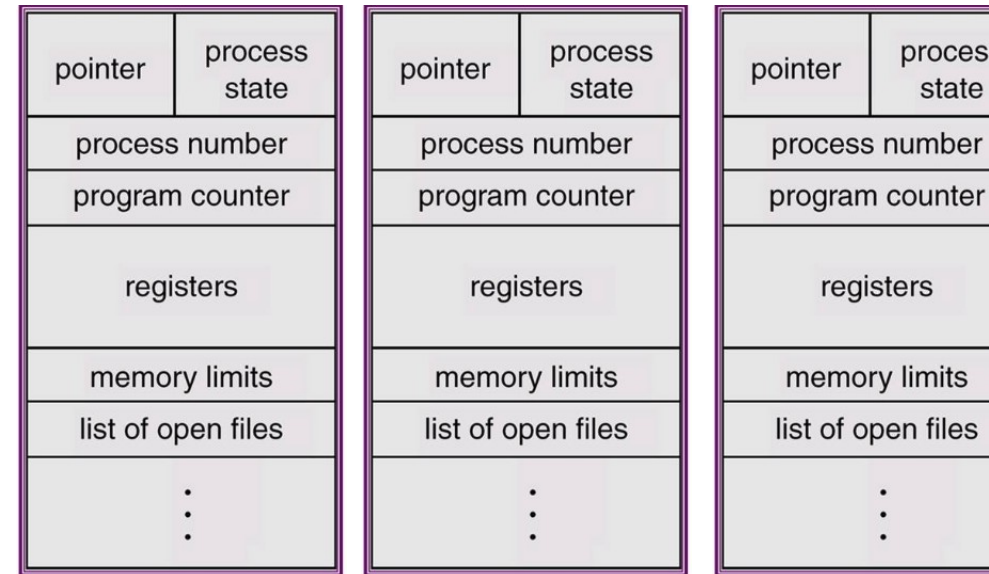
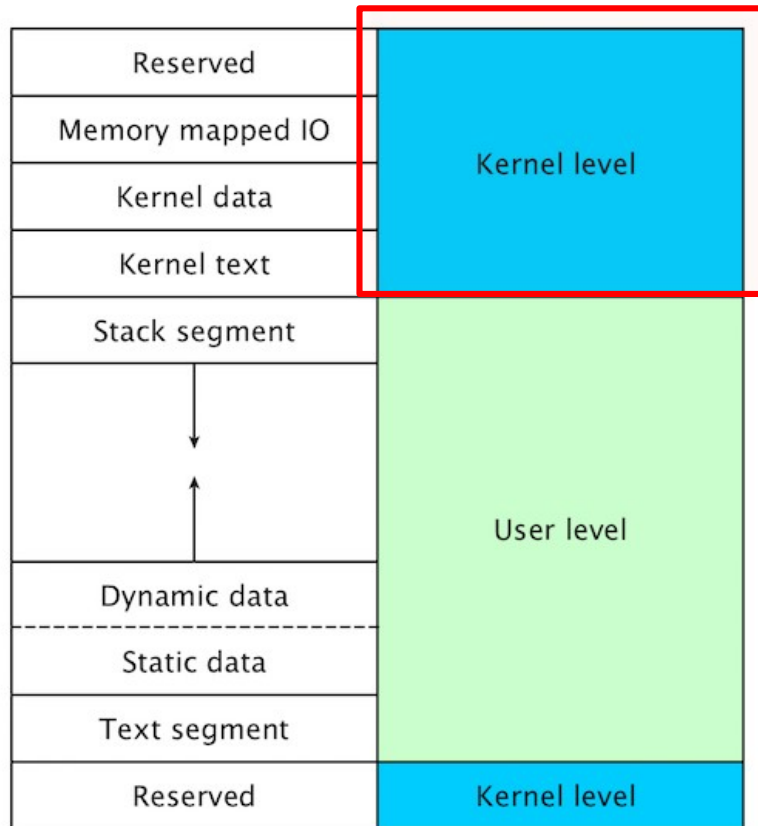
Registers : 레지스터들의 정보가 저장됨을 확인할 수 있다.

Context Switching

- 실행중인 프로세스는 모두 각각의 PCB를 가지고 있다.
- PCB는 프로세스가 실행될 때 같이 생기고 프로세스가 완료되면 사라진다.
- PCB는 메모리의 커널 영역에 위치한다.(커널 스택???)

0xffffffff
0xffff0010
0xffff0000
0x90000000
0x80000000

0x10000000
0x04000000
0x00000000



프로세스1 PCB

프로세스2 PCB

프로세스3 PCB

Context Switching

2. AVR에는 운영체제가 없는데? → MicroC/OS(uc/os)

- MicroC/OS-II(uC/OS-II) : uC/OS-II는 RTOS의 성격을 그대로 가지되, 다른 OS들에 비하여 매우 경량화된 구조와 크기를 가지고 있다. -위키백과
- RTOS(실시간 운영 체제) 프로세스간의 스케줄링을 도와줌.
- 30kByte 미만