



AVR Digital Voltmeter

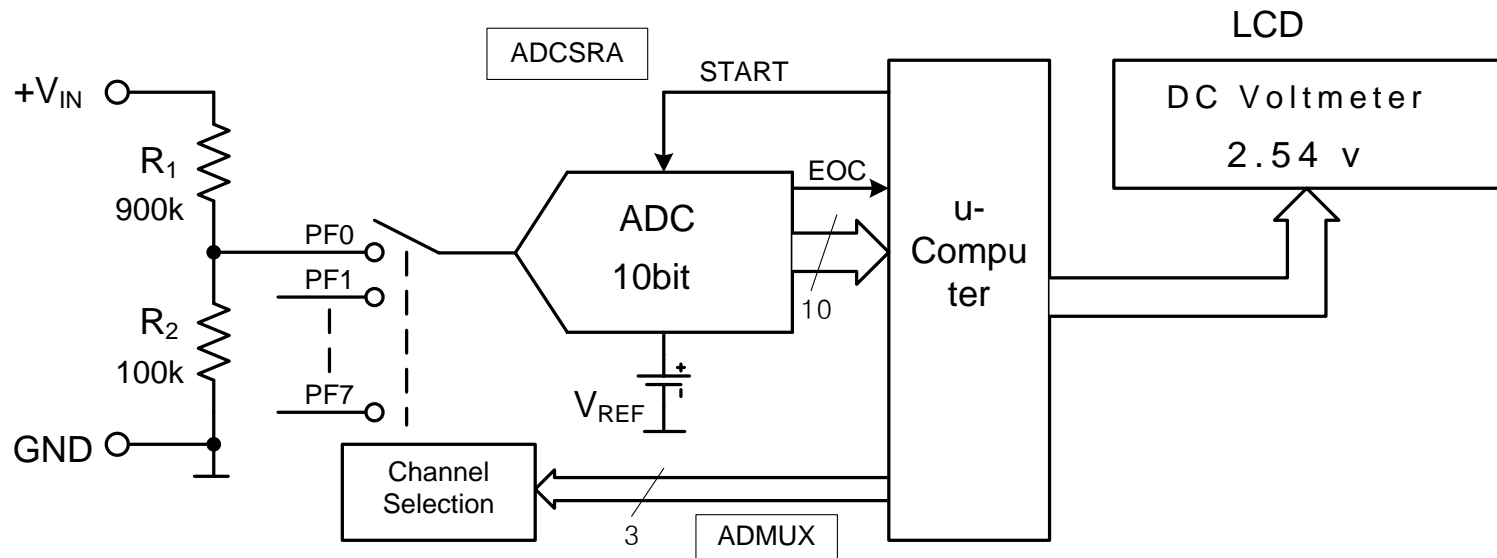
- ❖ ATmega128을 이용한 Digital Voltmeter

- ❖ 개발 목적 :

ATmega 128의 10bit A/D convert 특성을 활용하여 외부 인가 전압을 C프로그램 설계를 통하여 Digital Voltmeter로 구현하므로 U-Com 동작원리와 C프로그램 응용 능력을 향상시킨다.

Digital Voltmeter

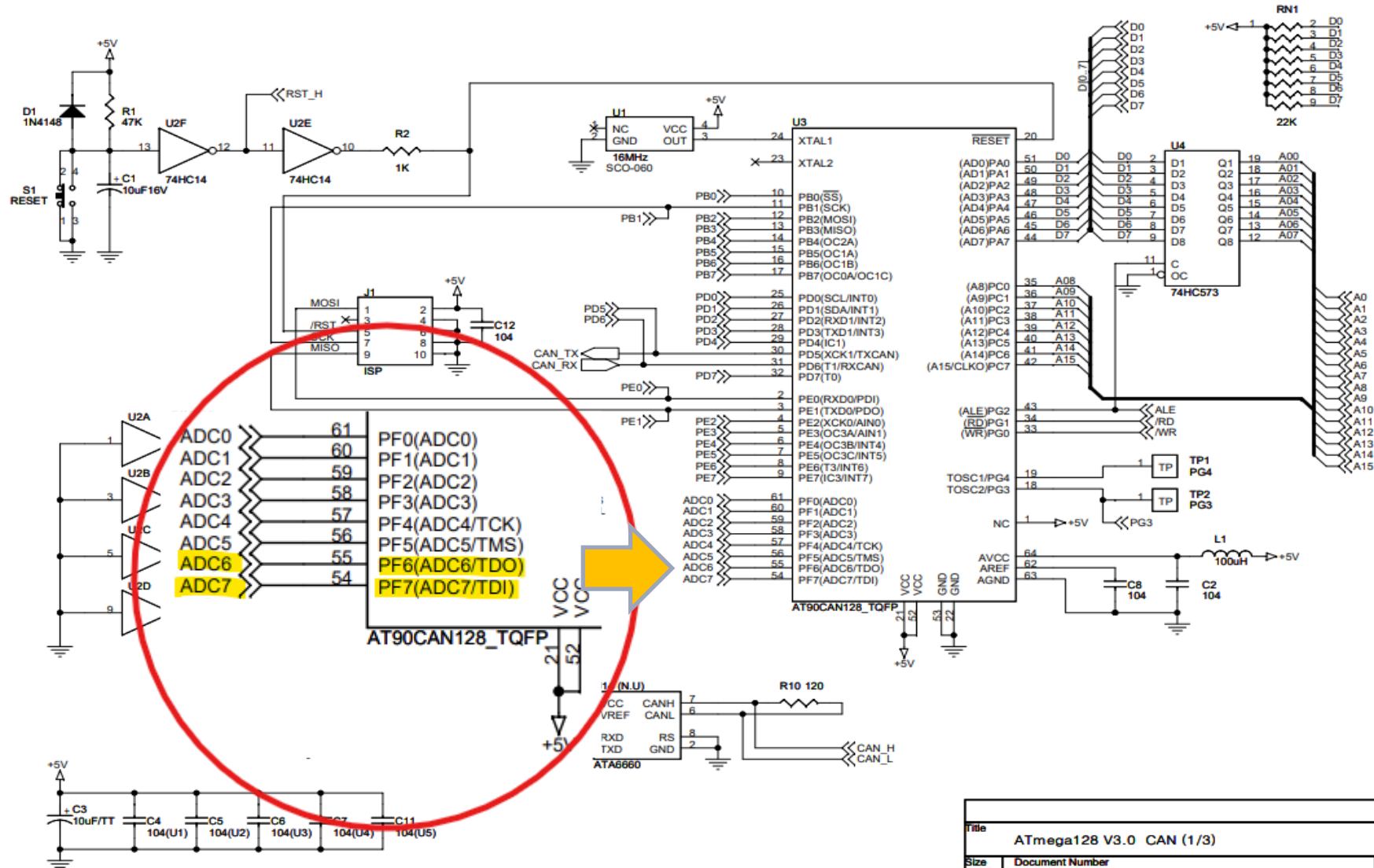
- ATmega 128의 10bit A/D 변환기를 이용한 Digital 전압계 설계



$$\diamond V_{IN} = \frac{AD\ Count}{FSR} * VREF\ [V]$$

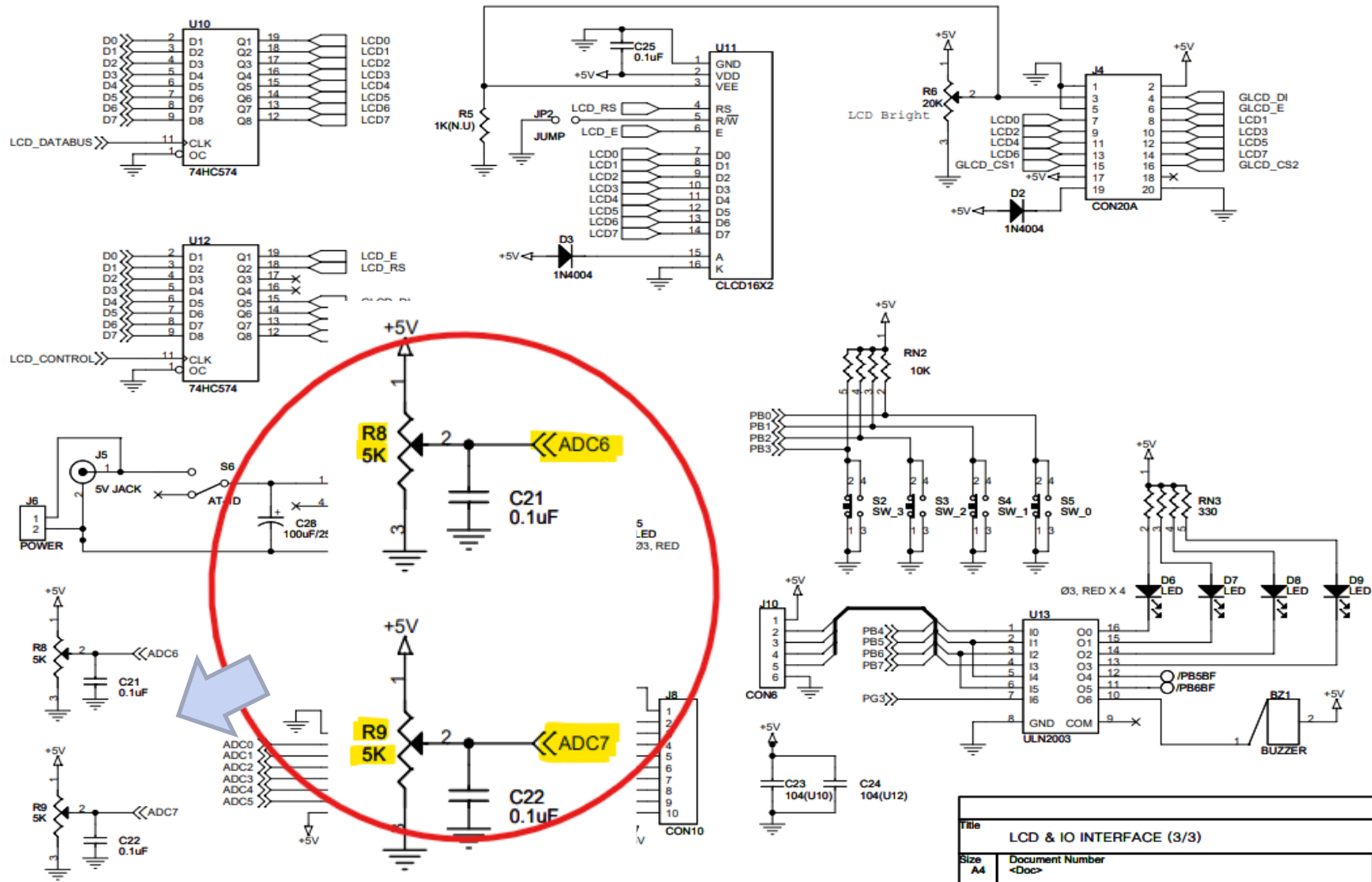
Digital Voltmeter

● ATmega 128의 10bit A/D 변환부분 회로도



Digital Voltmeter

● ATmega 128의 10bit A/D 변환부분 회로도



Digital Voltmeter

- ATmega 128의 PCB Assy



Digital Voltmeter

- Digital Voltmeter 동영상(예시)



Digital Voltmeter

- ATmega 128의 LCD Display 사양



(초기 부팅 화면)



(Digital Voltmeter 초기 화면)



(Digital Voltmeter 가변 화면)

Digital Voltmeter

- ATmega 128의 10bit A/D 변환기 Register 설정

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	

Table 97. Voltage Reference Selections for ADC

REFS1	REFS0	Voltage Reference Selection
0	0	AREF, Internal Vref turned off
0	1	AVCC with external capacitor at AREF pin
1	0	Reserved
1	1	Internal 2.56V Voltage Reference with external capacitor at AREF pin

- Bit 5 – ADLAR: ADC Left Adjust Result
- Bits 4:0 – MUX4:0: Analog Channel and Gain Selection Bits

Digital Voltmeter

● ATmega 128의 10bit A/D 변환기 Register 설정

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
- Bit 6 – ADSC: ADC Start Conversion
- Bit 5 – ADFR: ADC Free Running Select
- Bit 4 – ADIF: ADC Interrupt Flag
- Bit 3 – ADIE: ADC Interrupt Enable
- Bits 2:0 – ADPS2:0: ADC Prescaler Select Bits

MUX4..0	Single Ended Input
00000	ADC0
00001	ADC1
00010	ADC2
00011	ADC3
00100	ADC4
00101	ADC5
00110	ADC6
00111	ADC7
01000	
01001	



AVR - Programming

● MCU(Micro-Control Unit) 초기화

```
void MCU_initialize(void)          /* initialize ATmega128 MCU */
{
    MCUCR = 0x80; // Enable external memory and I/O, Disable SLEEP mode
    XMCRA = 0x44; // 0x1100 - 0x7FFF (1 wait), 0x8000 - 0xFFFF (0 wait)
    XMCRB = 0x80; // Enable bus keeper. Use PC0-PC7 as high byte of address bus

    DDRB = 0xF0; // 입력/출력 방향결정, PORTB7..4 = output, PORTB3..0 = Input
    PORTB = 0x00; // I/O pin 초기값 지정

    DDRD = 0xFD; // Port D7..2 = output, "1= Output, 0= Input"
    PORTD = 0x00; // Port D1..0 = Input

    DDRE = 0x0A; // Port E1,3 = output, All other PORTs are Input
    PORTE = 0xE0;

    DDRF = 0x00; // Port F = Input
    PORTF = 0x00;

    DDRG = 0x1F;

    LCD_CONTROL = 0x00; // LCD 제어
    LCD_DATABUS = 0x00; // LCD Data
}
```



AVR - Programming

- Display X.XX number on LCD

```
void LCD_1d2(float number) /* display X.XX number on LCD */
{
    int i, j;
    j = (int)(number * 100. + 0.5);

    i = j / 100;
    LCD_data(i + '0');

    LCD_data('.');

    i = (j % 100) / 10;
    LCD_data(i + '0');

    i = j % 10;
    LCD_data(i + '0');
}
```