
마이크로컨트롤러 기능

제 4장 Internal Memory의 이해



ICAT

Integrated Circuits for Advanced Technology Lab.

Internal Memory의 이해

1. ATmega128A 데이터 메모리 구조
2. AVR Studio 디버깅 모드 시작
3. AVR Studio로 내부 메모리와 레지스터
이해하기

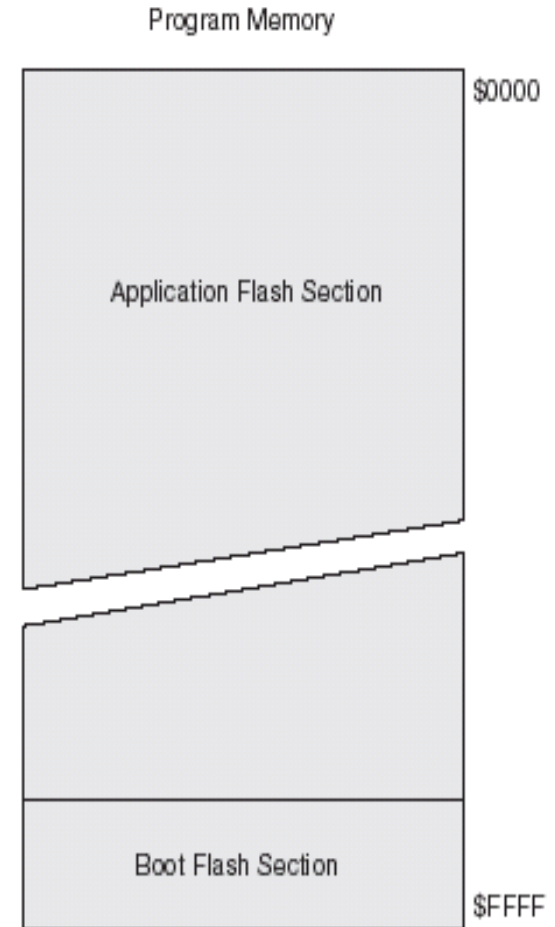


ATMega128A 프로그램 메모리 구조

□ ATMega128A 내장 프로그램 메모리

❖ ATmega 128A는 프로그램 저장을 위해 재 프로그램이 가능한 128[K]바이트의 플래시 메모리를 갖고 있다.

- 8비트 마이크로컨트롤러 이지만, 프로그램 메모리는 각 번지가 16비트 구조로 되어 있어서 $64[K] \times 16\text{비트} = 128[K]\text{바이트}$ 용량을 가진다.
- 이 플래시 메모리는 SPI 통신 방식을 이용한 ISP(In-System Programming) 기능을 가지고 있어서 사용자가 작성한 프로그램을 직접 다운로드 가능
- 플래시 메모리에 프로그램을 기입하는 방법은 ISP 방식 이외에도 JTAG 에뮬레이터를 사용하는 방법과 병렬 프로그래밍 모드를 사용하는 방법이 있다.



ICAT

Integrated Circuits for Advanced Technology Lab.

ATMega128A 데이터 메모리 구조

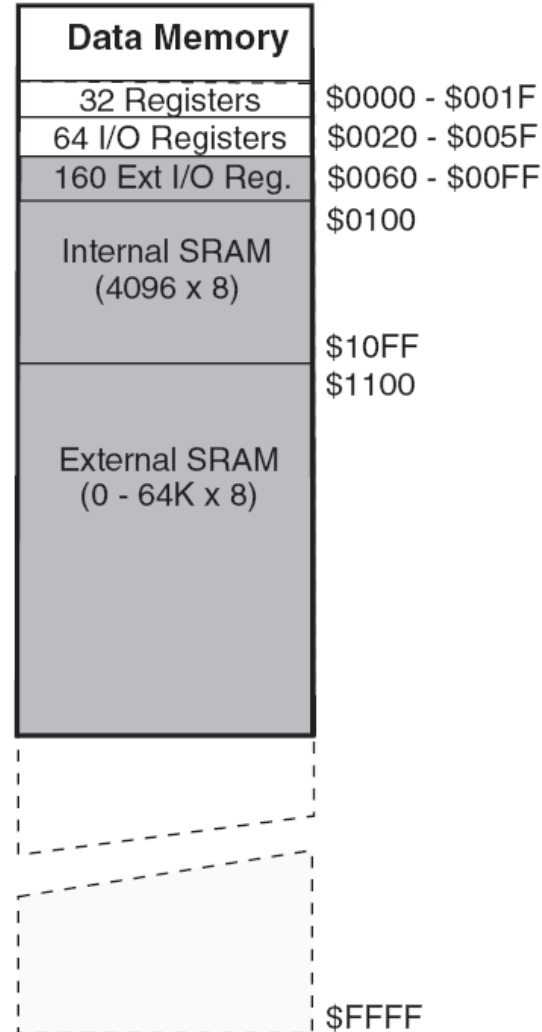
□ ATMega128A 내장 데이터 메모리

❖ RAM

- 레지스터
 - 범용 레지스터
 - I/O 레지스터
 - 확장 I/O 레지스터
- SRAM

❖ ROM

- EEPROM



ATMega128A 데이터 메모리 구조

□ I/O 레지스터와 확장 I/O 레지스터

- ❖ 내장된 각종 I/O 장치를 제어하기 위한 레지스터

주소	레지스터명	주소	레지스터명
\$00(\$20)	PINF	\$01(\$21)	PINE
\$02(\$22)	DDRE	\$03(\$23)	PORTE
\$10(\$30)	PIND	\$11(\$31)	DDRD
\$12(\$32)	PORTD	\$13(\$33)	PINC
\$14(\$34)	DDRC	\$15(\$35)	PORTC
\$16(\$36)	PINB	\$17(\$37)	DDRB
\$18(\$38)	PORTB	\$19(\$39)	PINA
\$1A(\$3A)	DDRA	\$1B(\$3B)	PORTA
\$34(\$54)	MCUCSR	\$35(\$55)	MCUCR
\$3F(\$3F)	SREG		



ATmega128A 데이터 메모리 구조

□ 확장 I/O 레지스터

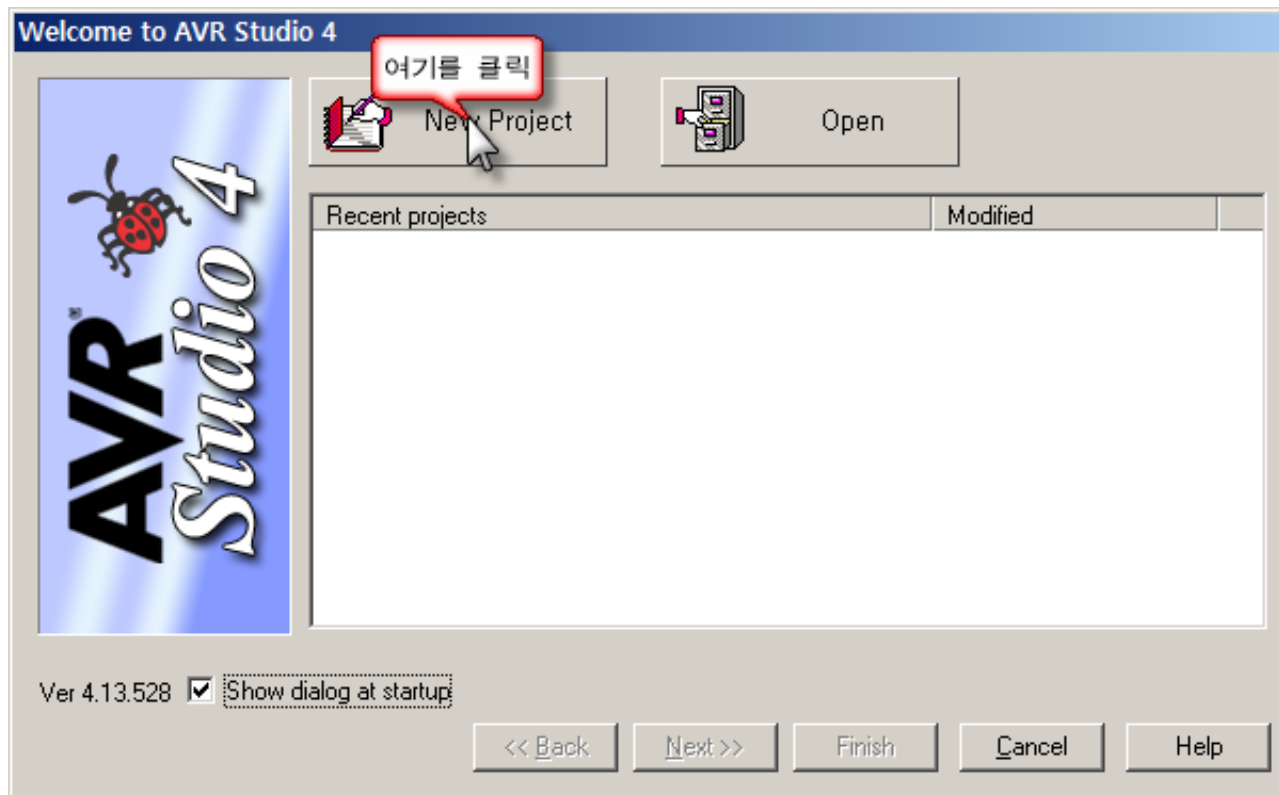
- ❖ ATmega128A에 추가된 각종 I/O 장치를 제어하기 위한 레지스터

주소	레지스터명	주소	레지스터명
\$61	DDRF	\$62	PORTF
\$63	PING	\$64	DDRG
\$65	PORTG	\$6A	EICRA
\$7C	ETIFR	\$7D	ETIMSK



AVR Studio 디버깅 모드 시작

- AVR Studio 디버깅 모드 : 프로젝트 만들기

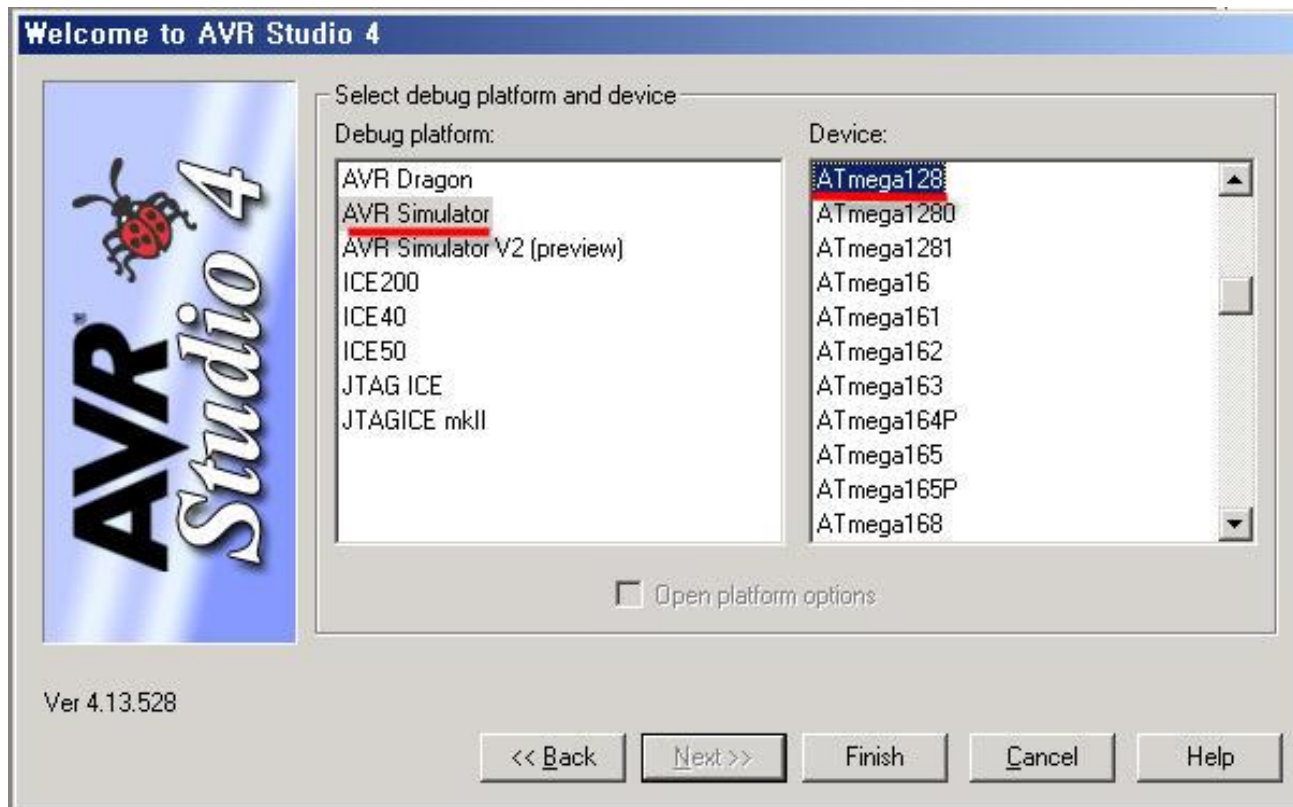


ICAT

Integrated Circuits for Advanced Technology Lab.

AVR Studio 디버깅 모드 시작

□ AVR Studio 디버깅 모드 : Project 생성에 필요한 항목 지정



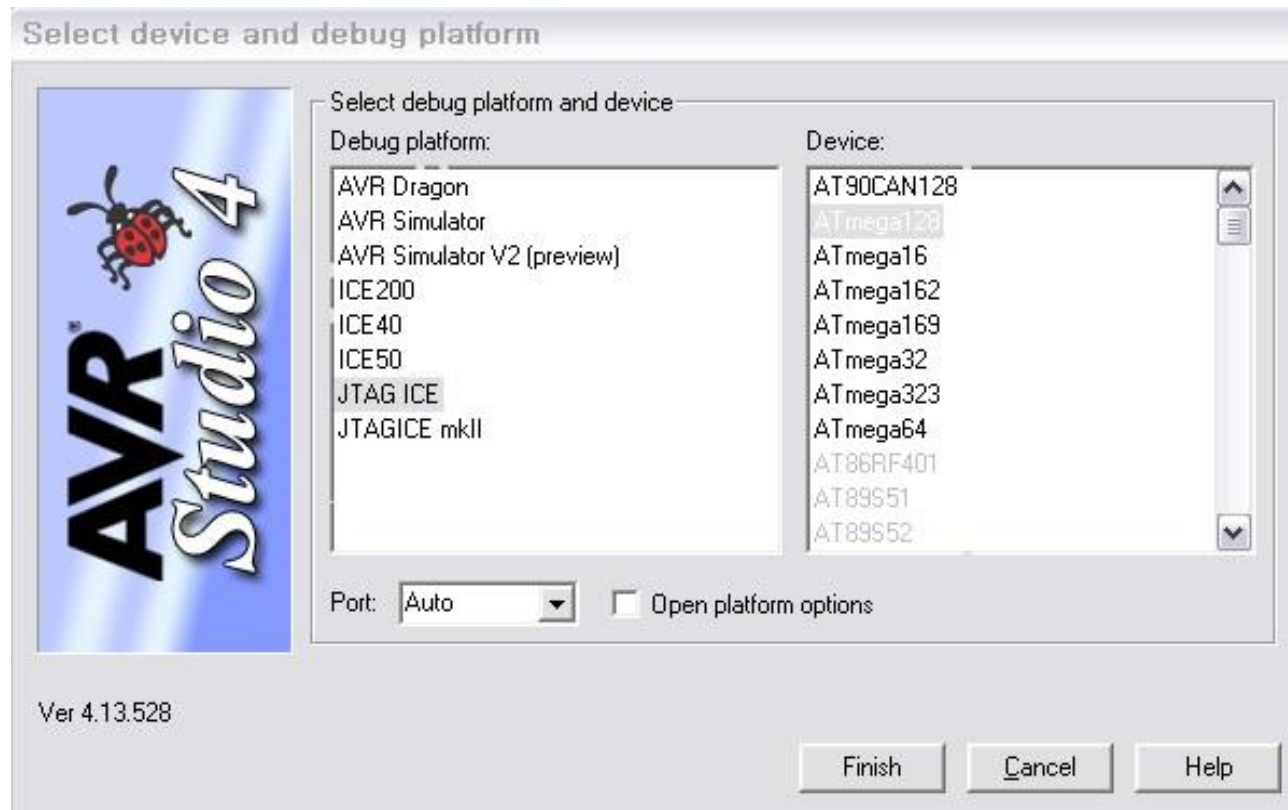
ICAT

Integrated Circuits for Advanced Technology Lab.

AVR Studio 디버깅 모드 시작

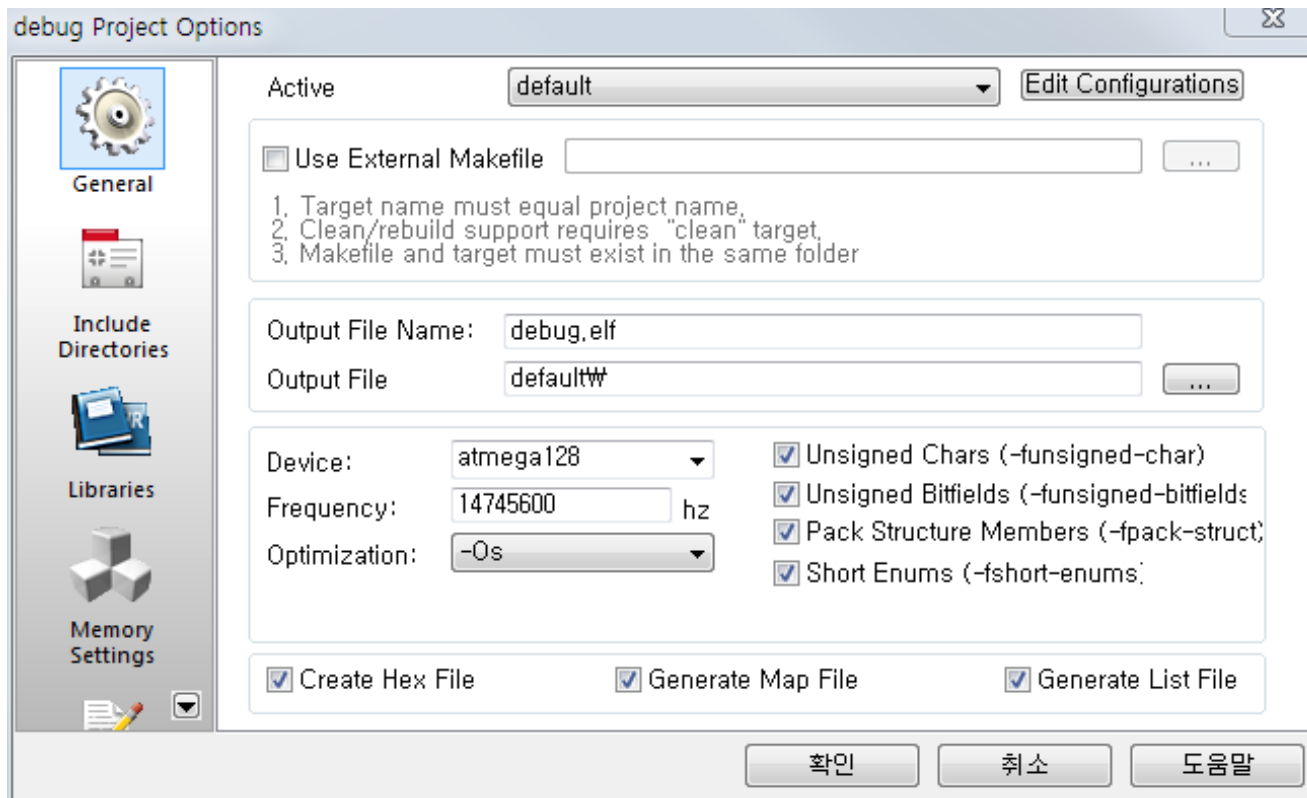
□ AVR Studio 디버깅 모드 :

Project 생성에 필요한 항목 지정 (**JTAG 사용시**)



AVR Studio 디버깅 모드 시작

- AVR Studio 디버깅 모드 : Configuration Option 설정
 - ❖ AVR Studio의 Project -> Configuration Option Select



AVR Studio 디버깅 모드 시작

```
#include<avr/io.h>

int main(void){

    unsigned char data = 0x01 ;
    unsigned long i;

    DDRA = 0xFF;
    PORTA = 0x00;

    while(1){

        PORTA = data;

        data<<= 1;

        for(i=0;i<1000;i++);

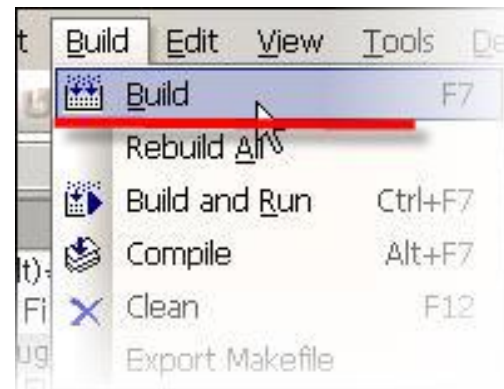
        if(data == 0x00)
            data = 0x01;

    }

    return 0;
}
```

□ 예제 소스 작성 및 이미지 빌드

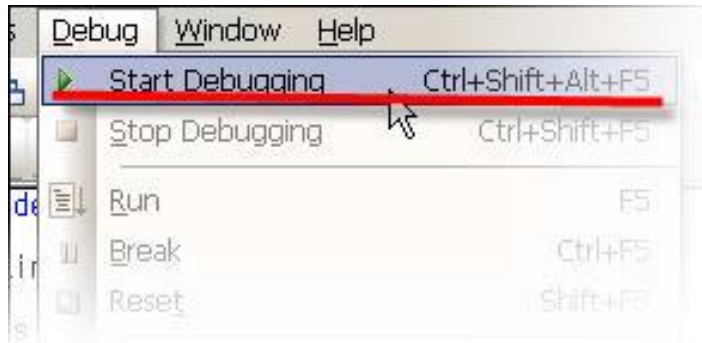
- ❖ 소스 작성
- ❖ 프로그램 컴파일 (Build -> Build)



AVR Studio 디버깅 모드 시작

□ 디버깅 모드 시작

- ❖ Debug 메뉴에서 Start Debugging 선택.



AVR Studio 디버깅 모드 시작

□ 디버깅
모드 화
면

Processor 창

Editor 창

I/O View 창

AVR Studio - [I:\W\Mcu_Module_Opt\WDebug_test\WDebug.c]

File Project Build Edit View Tools Debug Window Help

Trace Disabled

Processor

- Program Counter: 0x000067
- Stack Pointer: 0x10FD
- X pointer: 0x0100
- Y pointer: 0x10FF
- Z pointer: 0x014A
- Cycle Counter: 33
- Frequency: 4,000 MHz
- Stop Watch: 8.25 us
- SREG: [0] [1] [2] [3]

```
#include<avr/io.h>

int main(void){
    unsigned char data = 0x01 ;
    unsigned long i;
    DDRA = 0xFF;
    PORTA = 0x00;

    while(1){
        PORTA = data;
        data<<= 1;
        for(i=0;i<1000;i++);
        if(data == 0x00)
            data = 0x01;
    }

    return 0;
}
```

I/O View

- Cpu and System ...
- EEPROM
- Miscellaneous
- Watchdog Timer
- I/O Port A
- I/O Port B
- I/O Port C
- I/O Port D
- I/O Port E
- I/O Port F
- I/O Port G
- External Interrupts
- 8-bit Timer/Count...
- 16-bit Timer/Coun...
- 8-bit Timer/Coun...
- 16-bit Timer/Cou...
- Serial Peripheral
- USART 0
- USART 1
- Two-wire Serial
- Analog Compara...
- Analog to Digital
- Jtag Interface
- Boot Loader Support

Name	Address	Value	Bits
------	---------	-------	------

Message

Loaded plugin AVR GCC
Loaded partfile: C:\Program Files\Atmel\AVR Tools\PartDescriptionFiles\ATmega128.xml
gcc plug-in: Output directory I:\W\Mcu_Module_Opt\WDebug_test\default\ does not exist
gcc plug-in: Created directory I:\W\Mcu_Module_Opt\WDebug_test\default\
AVR Simulator: Please wait while configuring simulator...
AVR Simulator: ATmega128 Configured OK
Loaded objectfile: I:\W\Mcu_Module_Opt\WDebug_test\default\test.elf

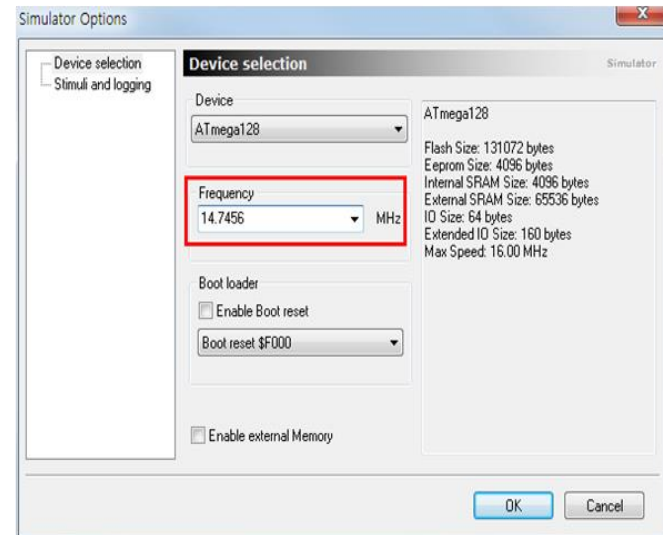
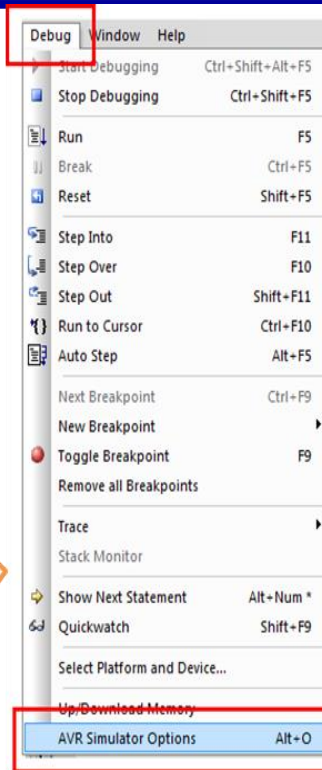
Build Message Find in Files Breakpoints and Tracepoints

ATmega128 AVR Simulator Auto Stopped Ln 3, Col 1 CAP_NUM SCRL

AVR Studio 디버깅 모드 시작

□ 주파수 변경 방법

❖ 14.7456MHz



Name	Value
Program Coun...	0x00005F
Stack Pointer	0x10FD
X pointer	0x0100
Y pointer	0x10FF
Z pointer	0x00D8
Cycle Counter	25
Frequency	4,000 MHz
Stop Watch	0.00 us
SREG	□□□□□□□□
Registers	
R00	0x00
R01	0x00

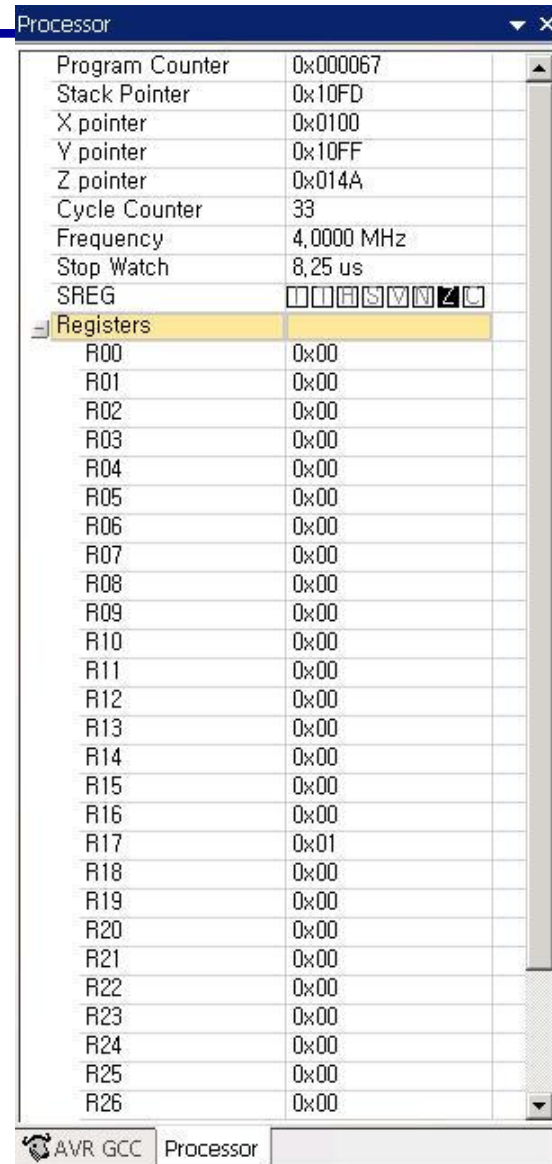
Name	Value
Program Coun...	0x00005F
Stack Pointer	0x10FD
X pointer	0x0100
Y pointer	0x10FF
Z pointer	0x00D8
Cycle Counter	25
Frequency	14,7456 MHz
Stop Watch	0.00 us
SREG	□□□□□□□□
Registers	
R00	0x00
R01	0x00



AVR Studio 디버깅 모드 시작

□ 디버깅 모드 화면 : Processor 창

- ❖ AVR 내부 레지스터 값을 표시
- ❖ 프로그램 카운터와 스택 포인터, 범용 레지스터의 값 확인 가능.



The screenshot shows the 'Processor' window in AVR Studio. It displays various system registers and their current values. The 'Registers' section is expanded, showing R00 through R26. The values for R00-R16 are 0x00, R17 is 0x01, and R18-R26 are 0x00. Other registers like Program Counter, Stack Pointer, X pointer, Y pointer, Z pointer, Cycle Counter, Frequency, Stop Watch, and SREG are also visible.

Register Name	Value
Program Counter	0x000067
Stack Pointer	0x10FD
X pointer	0x0100
Y pointer	0x10FF
Z pointer	0x014A
Cycle Counter	33
Frequency	4,0000 MHz
Stop Watch	8,25 us
SREG	00000000
Registers	
R00	0x00
R01	0x00
R02	0x00
R03	0x00
R04	0x00
R05	0x00
R06	0x00
R07	0x00
R08	0x00
R09	0x00
R10	0x00
R11	0x00
R12	0x00
R13	0x00
R14	0x00
R15	0x00
R16	0x00
R17	0x01
R18	0x00
R19	0x00
R20	0x00
R21	0x00
R22	0x00
R23	0x00
R24	0x00
R25	0x00
R26	0x00

AVR Studio 디버깅 모드 시작

□ 디버깅 모드 화면 : I/O View 창

- ❖ 입출력 레지스터 및 타이머 등의 레지스터 값을 표시
- ❖ 포트A~G 레지스터 및 타이머, UART 레지스터가 표시되는 걸 볼 수 있음.

The screenshot shows the I/O View window in AVR Studio. The window title is 'I/O View' and the selected component is 'I/O Port A'. The left pane shows a tree view of hardware components, including 'I/O Port A', 'I/O Port B', 'I/O Port C', 'I/O Port D', 'I/O Port E', 'I/O Port F', 'I/O Port G', 'External Interrupts', '8-bit Timer/Count...', '16-bit Timer/Coun...', 'Serial Peripheral I...', 'USART 0', 'USART 1', 'Two-wire Serial I...', and 'Analog Comparator'. The right pane shows the selected component's registers and their values:

Name	Address	Value	Bits
DATA REGISTER	0x00	0x00	
DIRECTION REGISTER	0x00	0x00	
INPUT PINS	0x00	0x00	
PORTA	0x1B	0x00	□□□□□□□□
DDRA	0x1A	0x00	□□□□□□□□
PINA	0x19	0x00	□□□□□□□□

AVR Studio 디버깅 모드 시작

□ 디버깅 모드 화면 : Editor 창

❖ 프로그램 소스 표시

```
#include<avr/io.h>
int main(void){
    unsigned char data = 0x01 ;
    unsigned long i;

    DDRA = 0xFF;
    PORTA = 0x00;

    while(1){
        PORTA = data;
        data<<= 1;
        for(i=0;i<1000;i++);
        if(data == 0x00)
            data = 0x01;
    }

    return 0;
}
```



ICAT

Integrated Circuits for Advanced Technology Lab.

AVR Studio 디버깅 모드 시작

□ 프로그램 실행 : 레지스터 값 변화 관찰

❖ Step Into(F11)



❖ Step Into(F11) 반복

```
#include<avr/io.h>

int main(void){

    unsigned char data = 0x01;
    unsigned long i;

    DDRA = 0xFF;
    PORTA = 0x00;
```

❖ Step Into(F11) 한번 더

Name	Address	Value	Bits
PORTA	0x1B	0x00	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
DDRA	0x1A	0x00	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
PINA	0x19	0x00	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>



Name	Address	Value	Bits
PORTA	0x1B	0x00	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
DDRA	0x1A	0xFF	<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/>
PINA	0x19	0x00	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>

AVR Studio 디버깅 모드 시작

□ 디버깅 모드 화면 : Break Point

❖ 디버깅을 해보고 싶은 부분부터 디버깅을 할 수 있게 해주는 기능

The screenshot shows the AVR Studio interface. The code editor displays the following code:

```
#include<avr/io.h>

int main(void){
    unsigned char data = 0x01 ;
    unsigned long i;
    DDRA = 0xFF;
    PORTA = 0x00;
    while(1){
        PORTA = data;
        data<<= 1;
        for(i=0;i<1000;i++);
        if(data == 0x00)
            data = 0x01;
    }
    return 0;
}
```

A yellow arrow points to the line `PORTA = 0x00;`, with a callout box containing the text "현재 실행라인" (Current execution line).

A red dot and vertical line mark a breakpoint on the line `data = 0x01;`, with a callout box containing the text "Break Point".

A tooltip for the "Toggle Breakpoint (F9)" button is shown, with the text "단축키 F9" (Shortcut key F9) next to it.

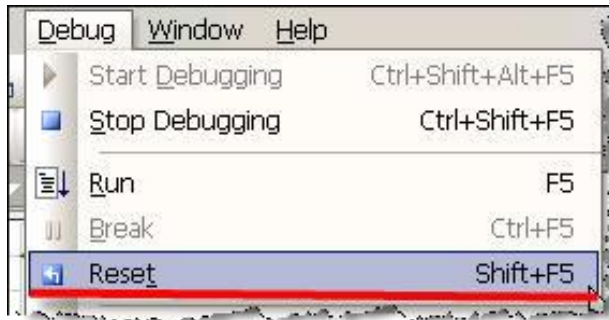
The right sidebar shows the hardware components of the AVR microcontroller, including CPU and System, EEPROM, Miscellaneous, Watchdog Timer, I/O Port A through G, External Interrupts, and Timers.

The ICAT logo and "Integrated Circuits for Advanced Technology Lab." are visible in the bottom right corner.

AVR Studio 디버깅 모드 시작

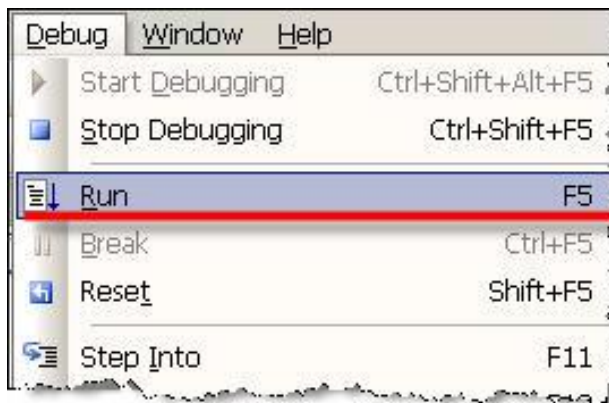
□ 디버깅 모드 화면 : Break Point

❖ 프로그램 리셋(Debug -> Reset)



노란 화살표 main()에 위치

❖ Run (Debug -> Run (F5))



노란 화살표 Break Point 에 위치



ICAT

Integrated Circuits for Advanced Technology Lab.

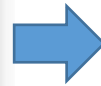
AVR Studio 디버깅 모드 시작

□ 디버깅 모드 화면 : Break Point

Name	Address	Value	Bits
PORTA	0x1B	0x80	■ □ □ □ □ □ □ □
DDRA	0x1A	0xFF	■ ■ ■ ■ ■ ■ ■ ■
PINA	0x19	0x80	■ □ □ □ □ □ □ □

❖ F11

```
PORTA = data;
data<<= 1;
for(i=0;i<1000;i++);
if(data == 0x00)
    data = 0x01;
}
return 0;
```



Name	Address	Value	Bits
PORTA	0x1B	0x01	□ □ □ □ □ □ □ ■
DDRA	0x1A	0xFF	■ ■ ■ ■ ■ ■ ■ ■
PINA	0x19	0x80	■ □ □ □ □ □ □ □



ICAT

Integrated Circuits for Advanced Technology Lab.

실습 4 : 내부 메모리와 레지스터 이해

□ 실습 개요

- ❖ 하나의 예제 프로그램 작성
- ❖ AVR Studio의 메모리 Dump 및 변수 확인 창 등을 활용하여 내부 메모리와 레지스터의 변화를 관찰

□ 실습 목표

- ❖ 내부 메모리의 동작 방식을 이해하고
- ❖ 프로그램이 실행되면서 내부 메모리들을 어떻게 사용하는지를 이해한다.



ICAT

Integrated Circuits for Advanced Technology Lab.

실습 4 : 내부 메모리와 레지스터 이해

□ 예제 프로그램 작성

- ❖ AVR Studio4 실행
- ❖ New Project 생성
- ❖ 예제 프로그램 작성

□ 소스 분석(변수 선언)

- ❖ A,B,C : 전역 변수
 - 프로그램시작과 함께 일정영역을 할당
 - 할당받은 영역에 초기값 저장(0xAA, 0xBB, 0xCC)
 - Unsigned char이므로 1byte 할당
- ❖ ADD : 포인터 변수
 - 역시 내부 일정영역 할당
 - 메모리 영역에 데이터가 아닌 주소를 저장(0x130)
 - 0x130은 SRAM의 영역

```
D:\WAVR\WMemTest\WMemTest.c *
#include<avr/io.h>

unsigned char A=0xAA;
unsigned char B=0xBB;
unsigned char C=0xCC;
unsigned char *ADD=0x130;

int main(){

DDRA = 0xFF;
PORTA = 0x00;

DDRF = 0xFF;
PORTF = 0x00;

ADD = 0x140;
*ADD = 0x11;
*(ADD + 1) = 0x22;
*(ADD + 2) = 0x33;
*(ADD + 3) = 0x44;

A=0xBB;
B=C-A;

PORTA = A;
PORTF = B;

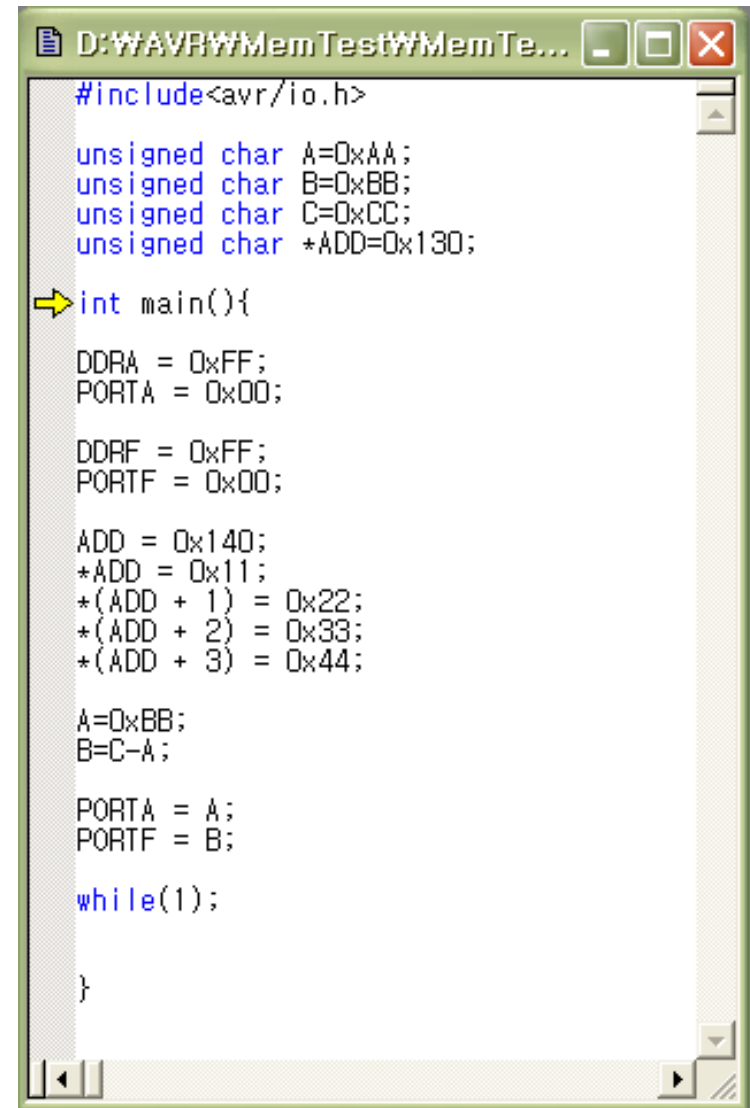
while(1);

}
```

실습 4 : 내부 메모리와 레지스터 이해

□ 프로그램 실행

- ❖ 프로그램을 빌드
- ❖ 디버깅 모드 시작
- ❖ 노란 화살표는 main()에 놓임.



```
D:\WAVR\MemTest\MemTe...
#include<avr/io.h>

unsigned char A=0xAA;
unsigned char B=0xBB;
unsigned char C=0xCC;
unsigned char *ADD=0x130;

→ int main(){

  DDRA = 0xFF;
  PORTA = 0x00;

  DDRF = 0xFF;
  PORTF = 0x00;

  ADD = 0x140;
  *ADD = 0x11;
  *(ADD + 1) = 0x22;
  *(ADD + 2) = 0x33;
  *(ADD + 3) = 0x44;

  A=0xBB;
  B=C-A;

  PORTA = A;
  PORTF = B;

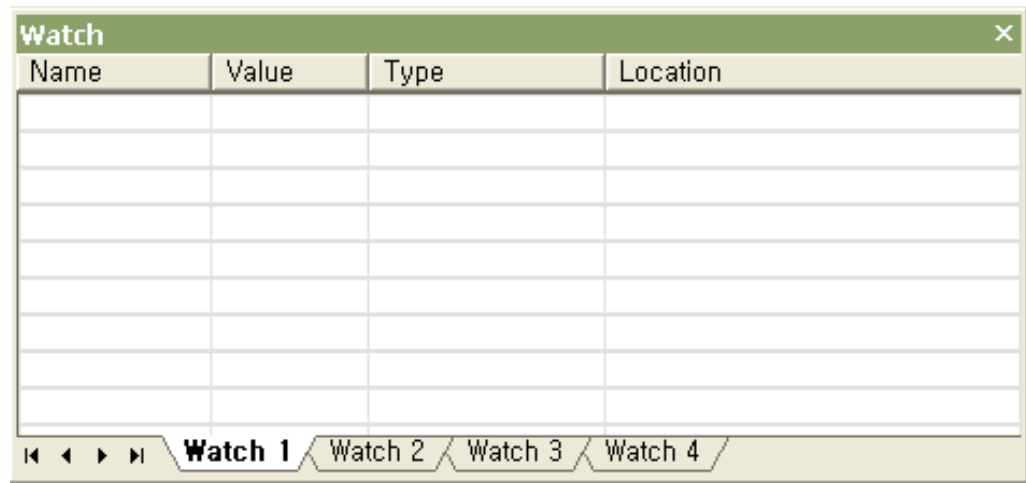
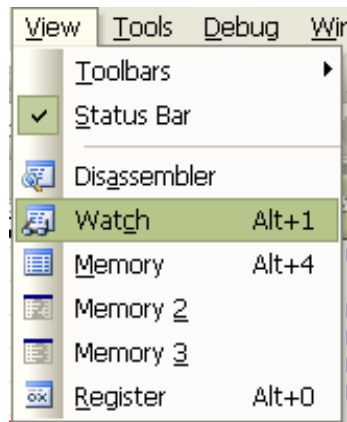
  while(1);

}
```


실습 4 : 내부 메모리와 레지스터 이해

□ 변수 관찰 창(watch)

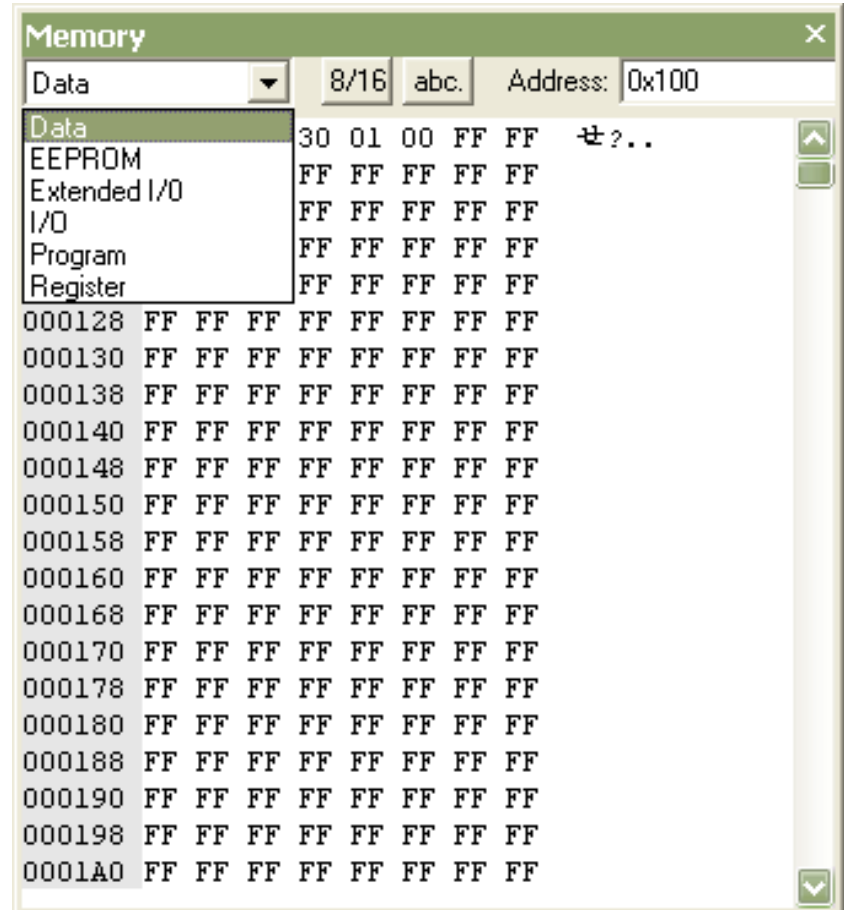
❖ 메뉴 창의 View > Watch



실습 4 : 내부 메모리와 레지스터 이해

□ 내부메모리 관찰 창

- ❖ Data 선택
- ❖ 데이터 메모리 영역을 보여줌.



ICAT

Integrated Circuits for Advanced Technology Lab.

실습 4 : 내부 메모리와 레지스터 이해

□ 변수 관찰 창

- ❖ 변수 관찰 창의 Name부분 빈칸을 더블 클릭하여 변수 이름을 적는다.
- ❖ 여기서는 'A'를 넣었음.

변수명

변수값

변수에 할당된
메모리 주소 :
SRAM 영역

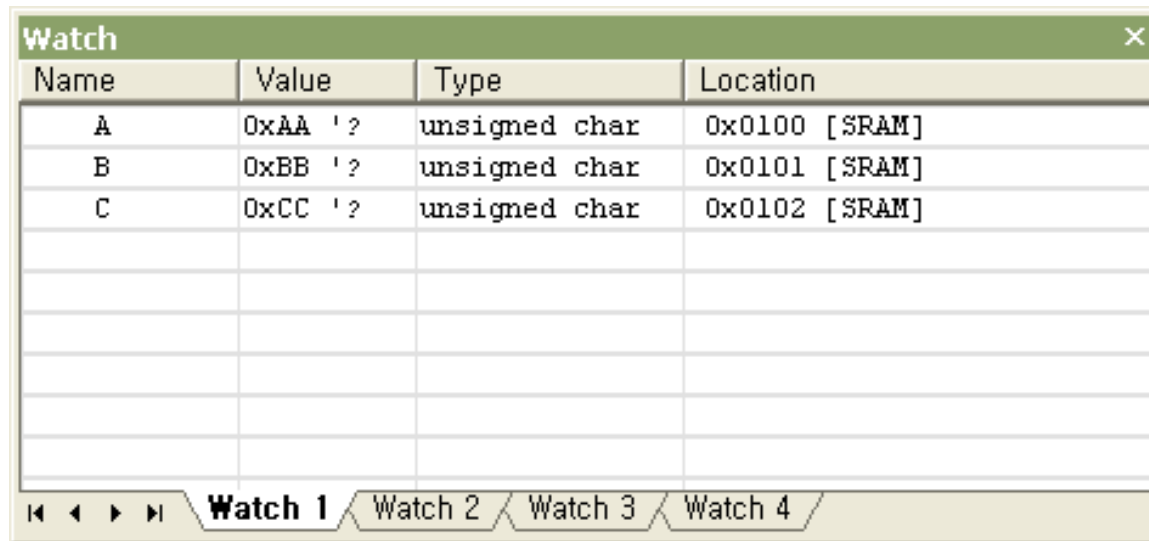
변수 데이터형

Name	Value	Type	Location
A	0xAA '?'	unsigned char	0x0100 [SRAM]

실습 4 : 내부 메모리와 레지스터 이해

□ 변수 관찰 창

❖ 같은 방법으로 B와 C를 입력.



The screenshot shows a 'Watch' window with a table of variables. The table has four columns: Name, Value, Type, and Location. Three rows are populated with variables A, B, and C. Variable A has a value of 0xAA, type of unsigned char, and location 0x0100 [SRAM]. Variable B has a value of 0xBB, type of unsigned char, and location 0x0101 [SRAM]. Variable C has a value of 0xCC, type of unsigned char, and location 0x0102 [SRAM]. The window also features a tabbed interface at the bottom with tabs for Watch 1, Watch 2, Watch 3, and Watch 4, and navigation arrows on the left.

Name	Value	Type	Location
A	0xAA '?'	unsigned char	0x0100 [SRAM]
B	0xBB '?'	unsigned char	0x0101 [SRAM]
C	0xCC '?'	unsigned char	0x0102 [SRAM]

실습 4 : 내부 메모리와 레지스터 이해

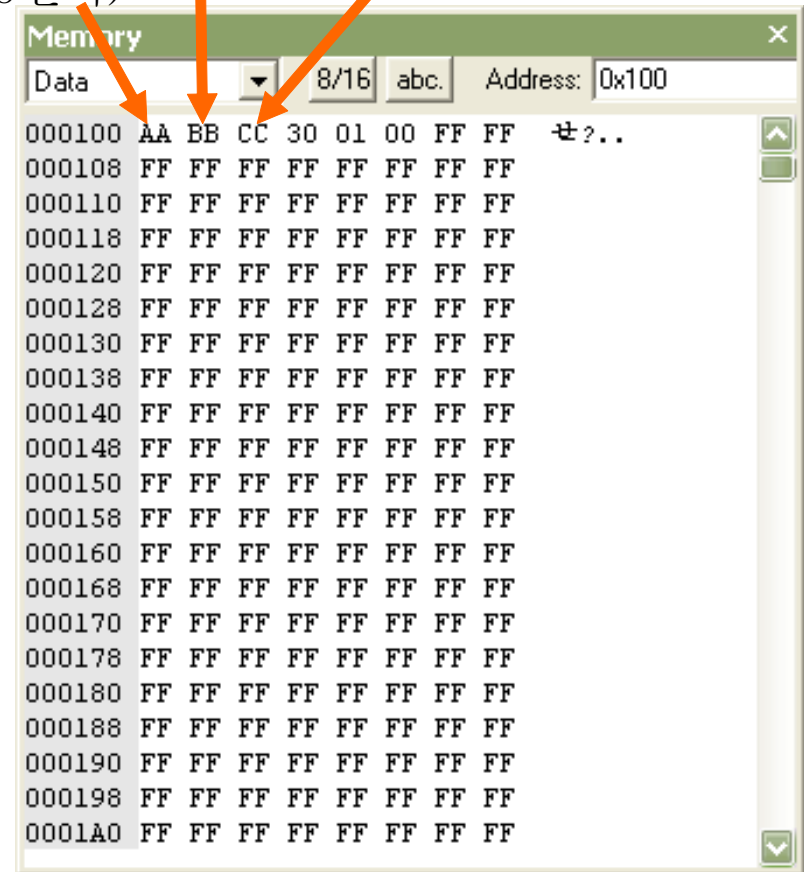
□ 내부메모리 관찰 창

- ❖ 내부메모리 관찰 창에서 Data선택.

변수 B 영역
(0x101번지)

변수 A 영역
(0x100번지)

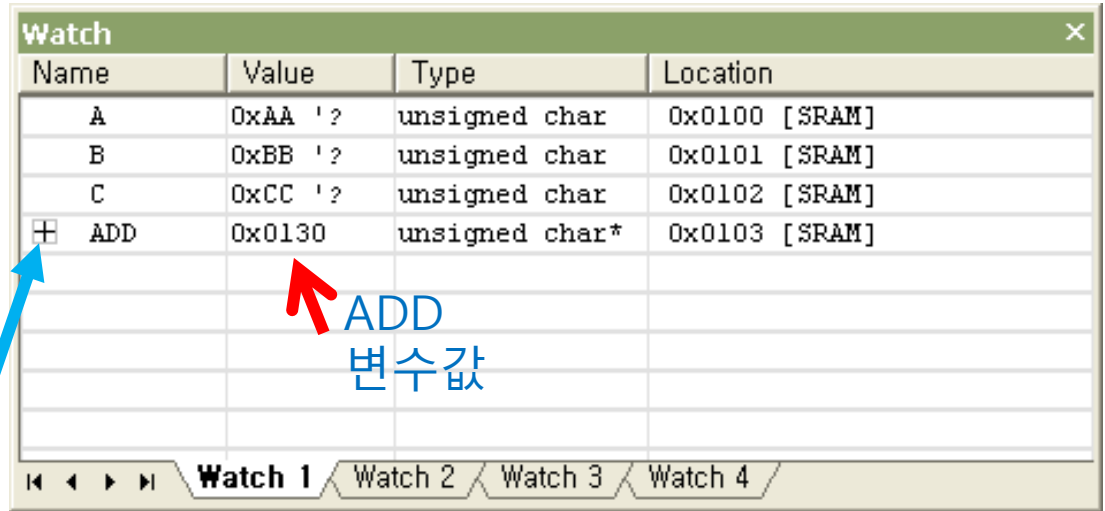
변수 C 영역
(0x102번지)



실습 4 : 내부 메모리와 레지스터 이해

□ 변수 관찰 창

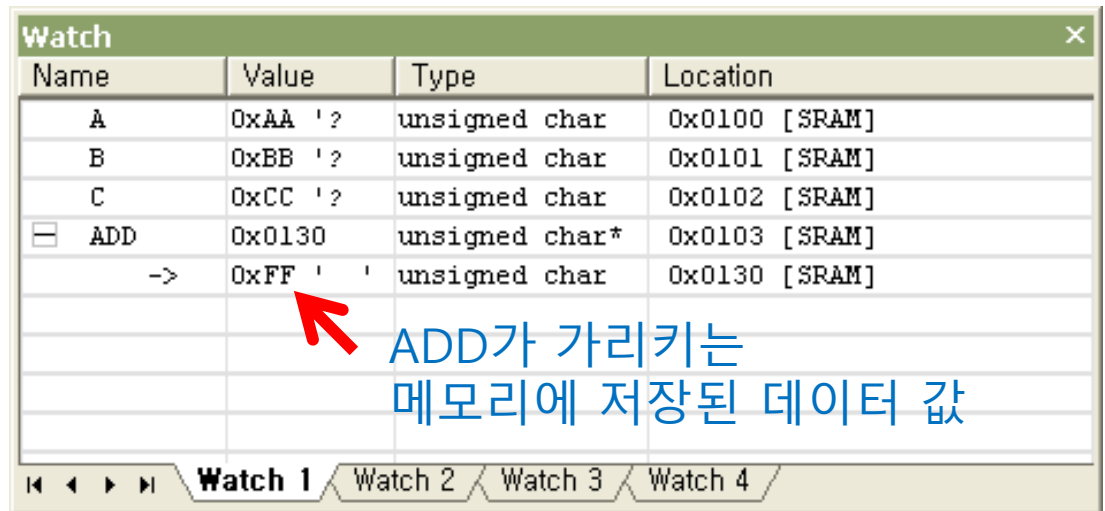
- ❖ 변수 관찰 창에 ADD 변수를 입력.



Name	Value	Type	Location
A	0xAA '?'	unsigned char	0x0100 [SRAM]
B	0xBB '?'	unsigned char	0x0101 [SRAM]
C	0xCC '?'	unsigned char	0x0102 [SRAM]
+	ADD	0x0130	unsigned char* 0x0103 [SRAM]

ADD 변수값

- ❖ ADD 변수 왼쪽의 + 기호를 누름



Name	Value	Type	Location
A	0xAA '?'	unsigned char	0x0100 [SRAM]
B	0xBB '?'	unsigned char	0x0101 [SRAM]
C	0xCC '?'	unsigned char	0x0102 [SRAM]
->	0xFF ' '	unsigned char	0x0130 [SRAM]

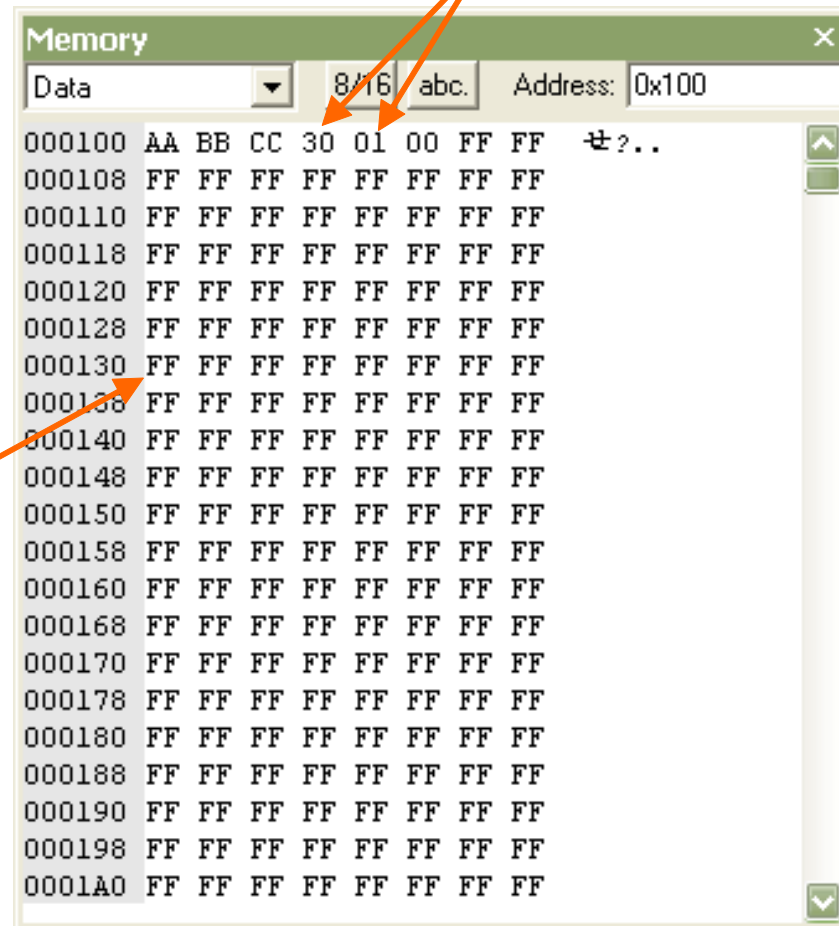
ADD가 가리키는 메모리에 저장된 데이터 값

실습 4 : 내부 메모리와 레지스터 이해

□ 내부메모리 관찰 창

변수 ADD 영역
(0x102번지)

변수 ADD가
가리키는
주소 영역
(0x130번지)



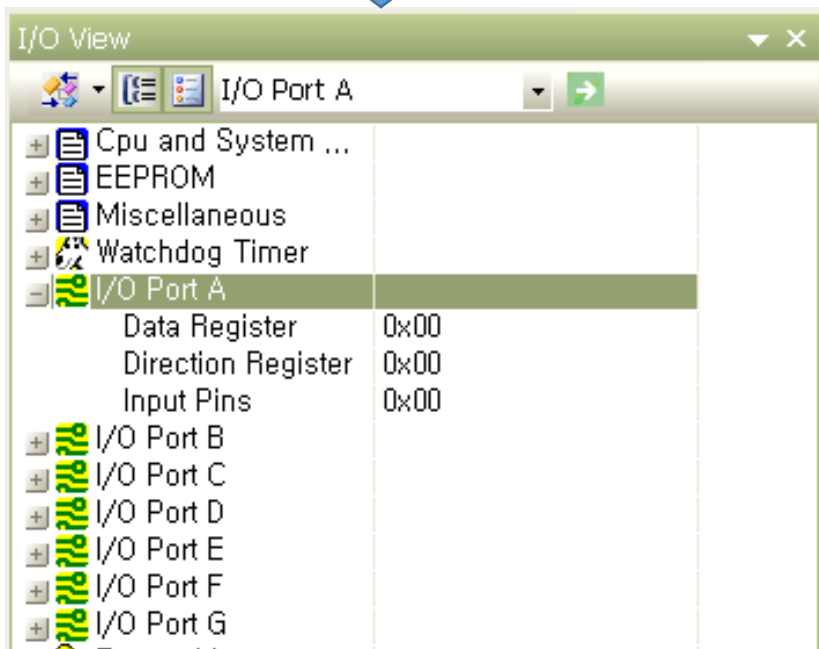
실습 4 : 내부 메모리와 레지스터 이해

- 프로그램 실행
 - ❖ Step Into 명령(F11)



```
unsigned char *ADDR=0x130,  
  
int main(){  
→ DDRA = 0xFF;  
  PORTA = 0x00;  
  
  DDRF = 0xFF;  
  PORTF = 0x00;  
  
  ADDR = 0x140;  
}
```

- I/O 레지스터 관찰 창
 - ❖ PORTA 레지스터 관찰



Name	Ad...	Value	Bits
PORTA	0x1B	0x00	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
DDRA	0x1A	0x00	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
PINA	0x19	0x00	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>



ICAT

Integrated Circuits for Advanced Technology Lab.

실습 4 : 내부 메모리와 레지스터 이해

- 프로그램 실행
 - ❖ Step Into 명령(F11)

- I/O 레지스터 관찰 창
 - ❖ PORTA 레지스터 관찰

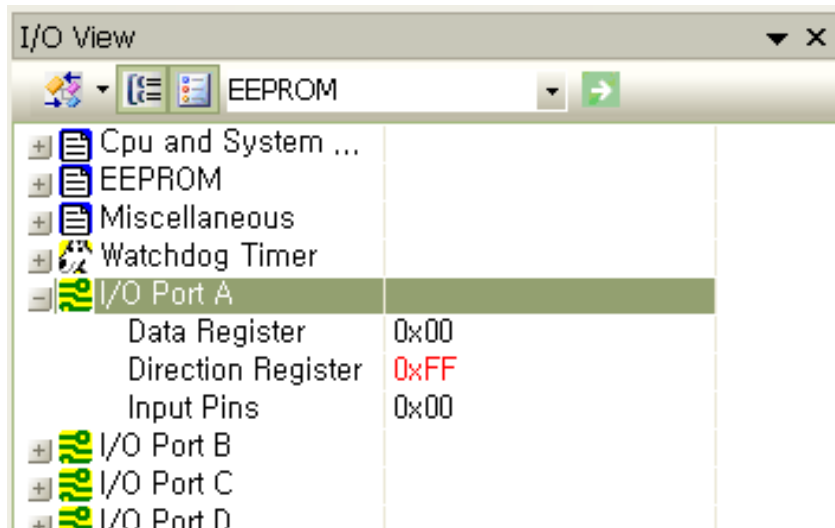


```

unsigned char 0x00;
unsigned char *ADD=0x130;

int main(){
    DDRA = 0xFF;
    → PORTA = 0x00;

    DDRF = 0xFF;
    PORTF = 0x00;
    }
    
```



Name	Ad...	Value	Bits
PORTA	0x1B	0x00	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
DDRA	0x1A	0xFF	<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/>
PINA	0x19	0x00	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>



ICAT

Integrated Circuits for Advanced Technology Lab.

실습 4 : 내부 메모리와 레지스터 이해

- 메모리 관찰 창
 - ❖ I/O register 선택

Address	Value
000000	00 00 00 00 00 00 00 00
000008	00 00 00 20 00 00 00 00
000010	00 00 00 00 00 00 00 00
000018	00 00 FF 00 00 00 00 00
000020	00 00 00 00 00 00 00 00
000028	00 00 00 00 00 00 00 00
000030	00 00 00 00 00 00 00 00
000038	00 00 00 00 00 00 FB 10 02

0x1A : DDRA 레지스터 주소



실습 4 : 내부 메모리와 레지스터 이해

□ 프로그램 실행

- ❖ Step Into 명령(F11) 2회



□ I/O 레지스터 관찰 창

- ❖ PORTA 레지스터 관찰



```
DDRA = 0xFF;  
PORTA = 0x00;
```

```
DDRF = 0xFF;  
PORTF = 0x00;
```

```
→ ADD = 0x140;  
+ADD = 0x11;  
*(ADD + 1) = 0x22;  
*(ADD + 2) = 0x33;  
*(ADD + 3) = 0x44;
```

I/O View

I/O Port F

- Cpu and System ...
- EEPROM
- Miscellaneous
- Watchdog Timer
- I/O Port A
 - Data Register 0x00
 - Direction Register 0xFF
 - Input Pins 0x00
- I/O Port B
- I/O Port C
- I/O Port D
- I/O Port E
- I/O Port F
 - Data Register 0x00
 - Direction Register 0xFF
 - Input Pins 0x00
- I/O Port G

Name	Ad...	Value	Bits
PORTF	0x42	0x00	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
DDRF	0x41	0xFF	<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/>
PINF	0x00	0x00	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>



ICAT

Integrated Circuits for Advanced Technology Lab.

실습 4 : 내부 메모리와 레지스터 이해

□ 메모리 관찰 창

- ❖ Extended I/O 선택

0x61 : DDRF
레지스터 주소

Address	Value
000060	00 FF 00 00 00 00 00 00
000068	00 00 00 00 00 00 00 00
000070	00 00 00 00 00 00 00 00
000078	00 00 00 00 00 00 00 00
000080	00 00 00 00 00 00 00 00
000088	00 00 00 00 00 00 00 00
000090	00 00 00 00 00 06 00 00
000098	00 00 00 20 00 06 00 00
0000A0	00 00 00 00 00 00 00 00
0000A8	00 00 00 00 00 00 00 00
0000B0	00 00 00 00 00 00 00 00
0000B8	00 00 00 00 00 00 00 00
0000C0	00 00 00 00 00 00 00 00
0000C8	00 00 00 00 00 00 00 00
0000D0	00 00 00 00 00 00 00 00
0000D8	00 00 00 00 00 00 00 00
0000E0	00 00 00 00 00 00 00 00
0000E8	00 00 00 00 00 00 00 00
0000F0	00 00 00 00 00 00 00 00
0000F8	00 00 00 00 00 00 00 00

실습 4 : 내부 메모리와 레지스터 이해

□ 프로그램 실행

❖ Step Into 명령(F11)



```
DDRA = 0xFF;
PORTA = 0x00;

DDRF = 0xFF;
PORTF = 0x00;

ADD = 0x140;
→ *ADD = 0x11;
*(ADD + 1) = 0x22;
*(ADD + 2) = 0x33;
*(ADD + 3) = 0x44;

A=0xBB;
B=C-A;
```

□ 변수 관찰 창

❖ ADD 변수 관찰



Name	Value	Type	Location
A	0xAA ' ?	unsigned char	0x0100 [SRAM]
B	0xBB ' ?	unsigned char	0x0101 [SRAM]
C	0xCC ' ?	unsigned char	0x0102 [SRAM]
ADD	0x0140	unsigned char*	0x0103 [SRAM]
->	0xFF ' '	unsigned char	0x0140 [SRAM]



ICAT

Integrated Circuits for Advanced Technology Lab.

실습 4 : 내부 메모리와 레지스터 이해

□ 메모리 관찰 창

❖ Data Memory (SRAM) 선택

ADD 주소

ADD가 가리키는 메모리 주소

Address	Data
000100	AA BB CC 40 01 00 FF FF
000108	FF FF FF FF FF FF FF FF
000110	FF FF FF FF FF FF FF FF
000118	FF FF FF FF FF FF FF FF
000120	FF FF FF FF FF FF FF FF
000128	FF FF FF FF FF FF FF FF
000130	FF FF FF FF FF FF FF FF
000138	FF FF FF FF FF FF FF FF
000140	FF FF FF FF FF FF FF FF
000148	FF FF FF FF FF FF FF FF
000150	FF FF FF FF FF FF FF FF

실습 4 : 내부 메모리와 레지스터 이해

□ 프로그램 실행

- ❖ Step Into 명령(F11)



```
DDRF = 0xFF;
PORTF = 0x00;

ADD = 0x140;
+ADD = 0x11;
→ *(ADD + 1) = 0x22;
*(ADD + 2) = 0x33;
*(ADD + 3) = 0x44;

↓ = 0xRR
```

□ 변수 관찰 창

- ❖ ADD 변수 관찰



Name	Value	Type	Location
A	0xAA '?'	unsigned char	0x0100 [SRAM]
B	0xBB '?'	unsigned char	0x0101 [SRAM]
C	0xCC '?'	unsigned char	0x0102 [SRAM]
ADD	0x0140	unsigned char*	0x0103 [SRAM]
->	0x11 '◀'	unsigned char	0x0140 [SRAM]



ICAT

Integrated Circuits for Advanced Technology Lab.

실습 4 : 내부 메모리와 레지스터 이해

□ 메모리 관찰 창

❖ Data Memory (SRAM) 선택

ADD 주소

ADD가 가리키는 메모리 주소

Address	Data
000100	AA BB CC 40 01 00 FF FF
000108	FF FF FF FF FF FF FF FF
000110	FF FF FF FF FF FF FF FF
000118	FF FF FF FF FF FF FF FF
000120	FF FF FF FF FF FF FF FF
000128	FF FF FF FF FF FF FF FF
000130	FF FF FF FF FF FF FF FF
000138	FF FF FF FF FF FF FF FF
000140	11 FF FF FF FF FF FF
000148	FF FF FF FF FF FF FF FF
000150	FF FF FF FF FF FF FF FF

실습 4 : 내부 메모리와 레지스터 이해

□ 프로그램 실행

- ❖ Step Into 명령(F11) 3회



```
ADD = 0x140;  
*ADD = 0x11;  
*(ADD + 1) = 0x22;  
*(ADD + 2) = 0x33;  
*(ADD + 3) = 0x44;  
→ A=0xBB;  
B=C-A;
```

□ 변수 관찰 창

- ❖ ADD 변수 관찰

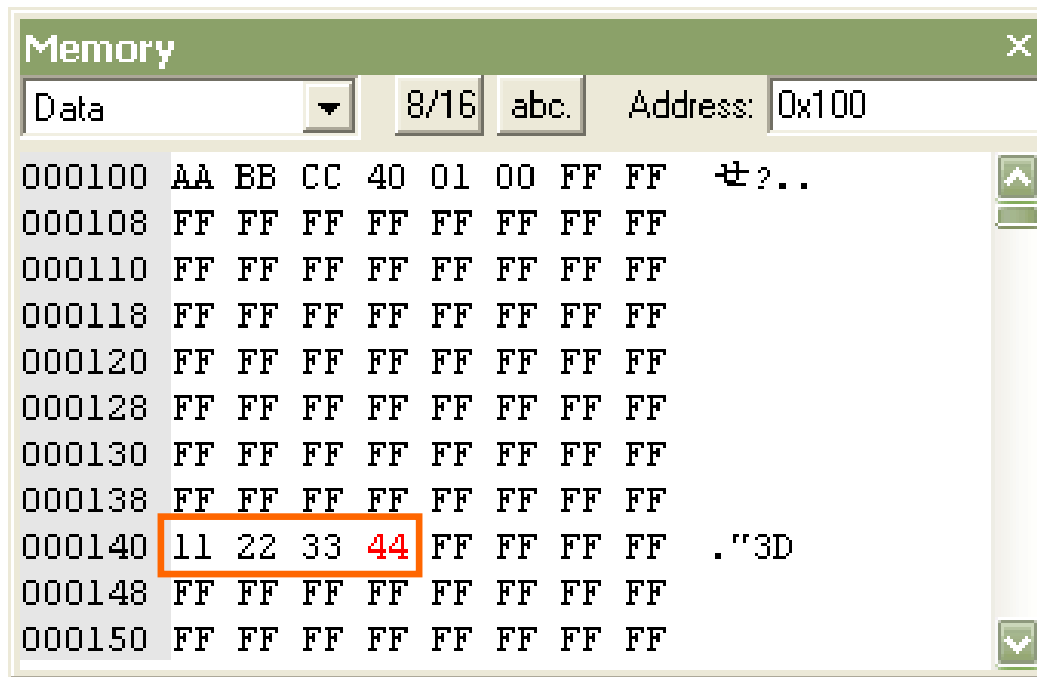


Watch			
Name	Value	Type	Location
A	0xAA ' ?	unsigned char	0x0100 [SRAM]
B	0xBB ' ?	unsigned char	0x0101 [SRAM]
C	0xCC ' ?	unsigned char	0x0102 [SRAM]
ADD	0x0140	unsigned char*	0x0103 [SRAM]
->	0x11 ' ◀	unsigned char	0x0140 [SRAM]

실습 4 : 내부 메모리와 레지스터 이해

□ 메모리 관찰 창

❖ Data Memory (SRAM) 선택



ICAT

Integrated Circuits for Advanced Technology Lab.

실습 4 : 내부 메모리와 레지스터 이해

□ 프로그램 실행

❖ Step Into 명령(F11)



```
*(ADD + 1) = 0x22;  
*(ADD + 2) = 0x33;  
*(ADD + 3) = 0x44;  
  
A=0xBB;  
→ B=C-A;  
  
PORTA = A;  
PORTF = B;
```

□ 변수 관찰 창

❖ A 변수 관찰



Name	Value	Type	Location
A	0xBB '?'	unsigned char	0x0100 [SRAM]
B	0xBB '?'	unsigned char	0x0101 [SRAM]
C	0xCC '?'	unsigned char	0x0102 [SRAM]
ADD	0x0140	unsigned char*	0x0103 [SRAM]
->	0x11 '←'	unsigned char	0x0140 [SRAM]



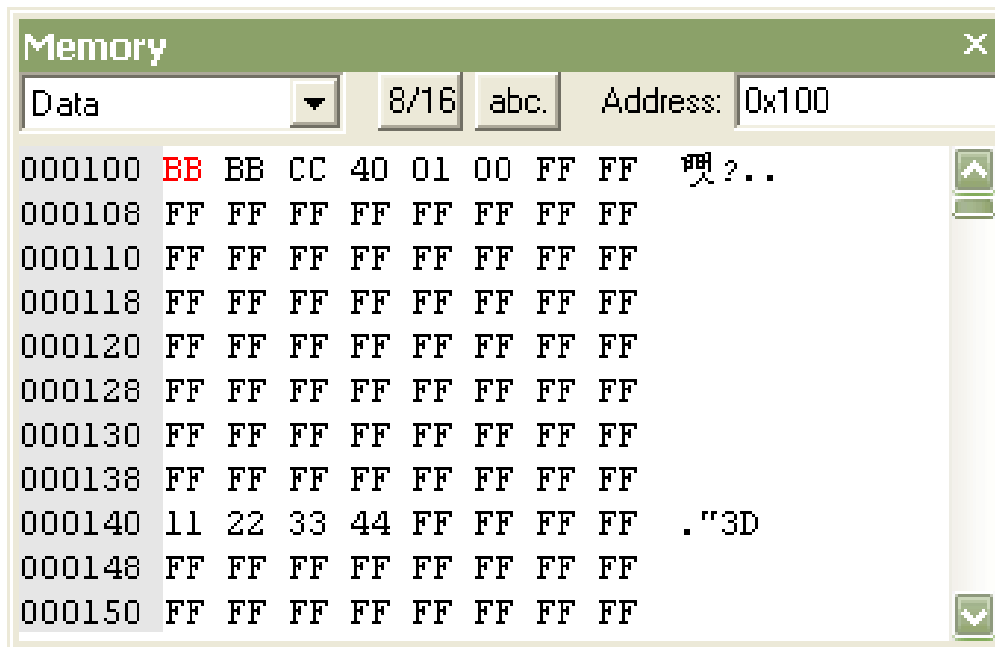
ICAT

Integrated Circuits for Advanced Technology Lab.

실습 4 : 내부 메모리와 레지스터 이해

□ 메모리 관찰 창

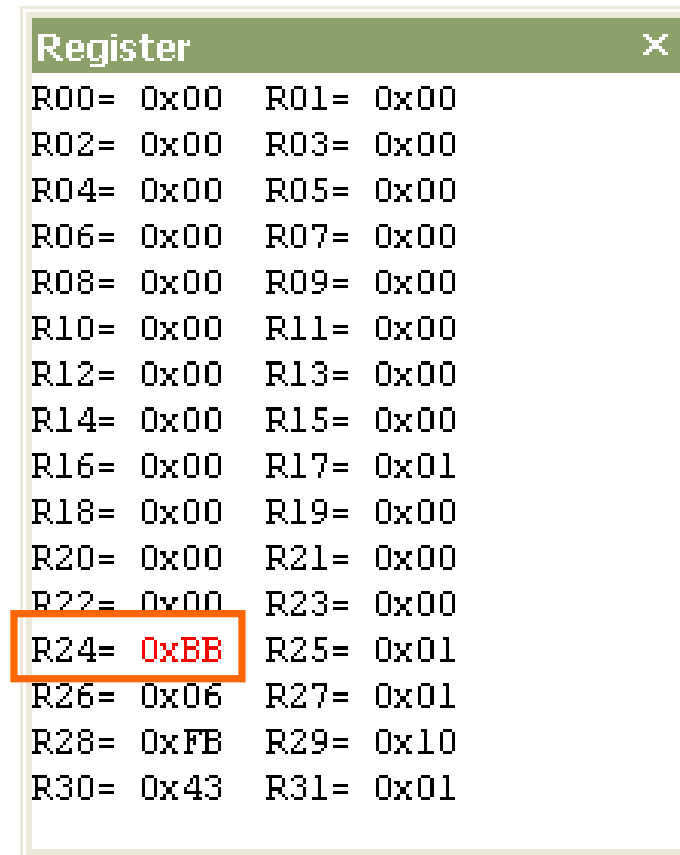
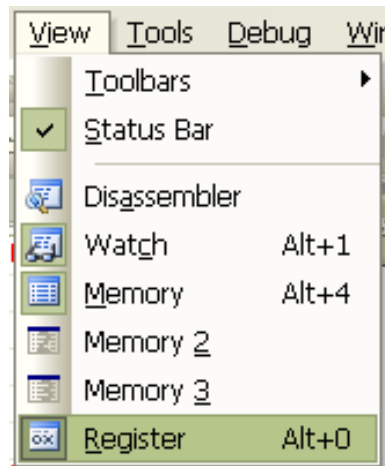
❖ Data Memory (SRAM) 선택



실습 4 : 내부 메모리와 레지스터 이해

□ 레지스터 관찰 창

❖ 메뉴의 View>Register



실습 4 : 내부 메모리와 레지스터 이해

□ 프로그램 실행

- ❖ Step Into 명령(F11)



```
A=0xBB;  
B=C-A;  
→ PORTA = A;  
   PORTF = B;
```

□ 변수 관찰 창

- ❖ B 변수 관찰



Watch			
Name	Value	Type	Location
A	0xBB '?'	unsigned char	0x0100 [SRAM]
B	0x11 '◀'	unsigned char	0x0101 [SRAM]
C	0xCC '?'	unsigned char	0x0102 [SRAM]
ADD	0x0140	unsigned char*	0x0103 [SRAM]
->	0x11 '◀'	unsigned char	0x0140 [SRAM]



실습 4 : 내부 메모리와 레지스터 이해

□ 레지스터 관찰 창

❖ R24, R25 관찰

□ 메모리 관찰 창

❖ Data Memory (SRAM) 선택



Address	Data
000100	BB 11 CC 40 01 00 FF FF ??..
000108	FF FF FF FF FF FF FF FF
000110	FF FF FF FF FF FF FF FF
000118	FF FF FF FF FF FF FF FF
000120	FF FF FF FF FF FF FF FF
000128	FF FF FF FF FF FF FF FF
000130	FF FF FF FF FF FF FF FF
000138	FF FF FF FF FF FF FF FF
000140	11 22 33 44 FF FF FF FF ."3D"
000148	FF FF FF FF FF FF FF FF
000150	FF FF FF FF FF FF FF FF

Register	Value
R00	0x00
R01	0x00
R02	0x00
R03	0x00
R04	0x00
R05	0x00
R06	0x00
R07	0x00
R08	0x00
R09	0x00
R10	0x00
R11	0x00
R12	0x00
R13	0x00
R14	0x00
R15	0x00
R16	0x00
R17	0x01
R18	0x11
R19	0x00
R20	0x00
R21	0x00
R22	0x00
R23	0x00
R24	0x11
R25	0xCC
R26	0x06
R27	0x01
R28	0xFB
R29	0x10
R30	0x43
R31	0x01



ICAT

Integrated Circuits for Advanced Technology Lab.

실습 4 : 내부 메모리와 레지스터 이해

□ 프로그램 실행

- ❖ Step Into 명령(F11)



```
A=0xBB;  
B=C-A;  
  
PORTA = A;  
PORTF = B;  
  
while(1);
```

□ 메모리 관찰 창

- ❖ I/O 레지스터 선택
- ❖ 0x1B 번지 관찰

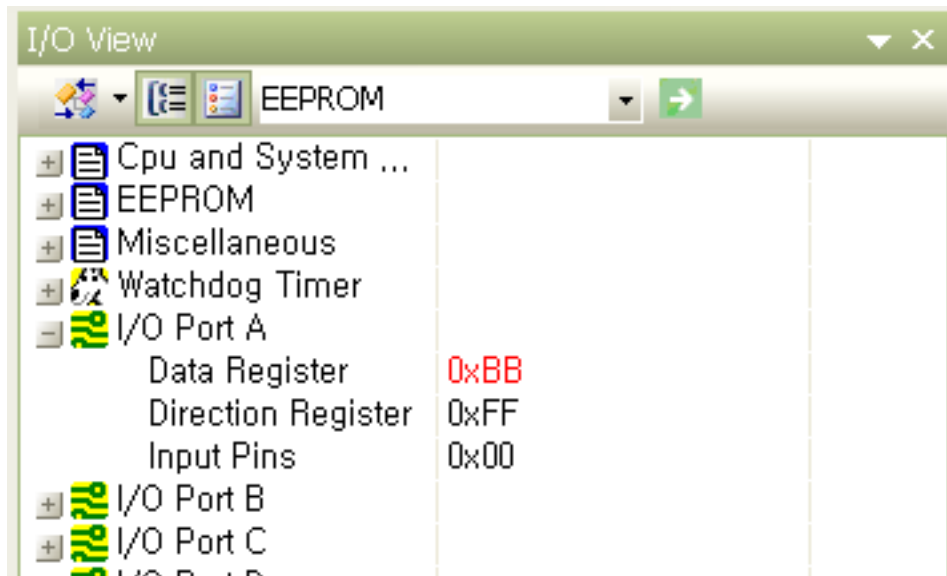
I/O	8/16	abc.	Address:	0x00					
000000	00	00	00	00	00	00	00	00
000008	00	00	00	20	00	00	00	00
000010	00	00	00	00	00	00	00	00
000018	00	00	FF	BB	00	00	00	00	.. ?..
000020	00	00	00	00	00	00	00	00
000028	00	00	00	00	00	00	00	00
000030	00	00	00	00	00	00	00	00
000038	00	00	00	00	00	FB	10	00?.

0x1B : PORTA 레지스터 주소



실습 4 : 내부 메모리와 레지스터 이해

- I/O 레지스터 관찰 창
 - ❖ PORTA 레지스터 관찰



Name	Ad...	Value	Bits
PORTA	0x1B	0xBB	■ □ ■ ■ ■ □ ■ ■
DDRA	0x1A	0xFF	■ ■ ■ ■ ■ ■ ■ ■
PINA	0x19	0x00	□ □ □ □ □ □ □ □



ICAT

Integrated Circuits for Advanced Technology Lab.

실습 4 : 내부 메모리와 레지스터 이해

□ 프로그램 실행

- ❖ Step Into 명령(F11)



```
A=0xBB;  
B=C-A;  
  
PORTA = A;  
PORTF = B;  
  
→ while(1);
```

□ 메모리 관찰 창

- ❖ Extended I/O 선택
- ❖ 0x1B 번지 관찰

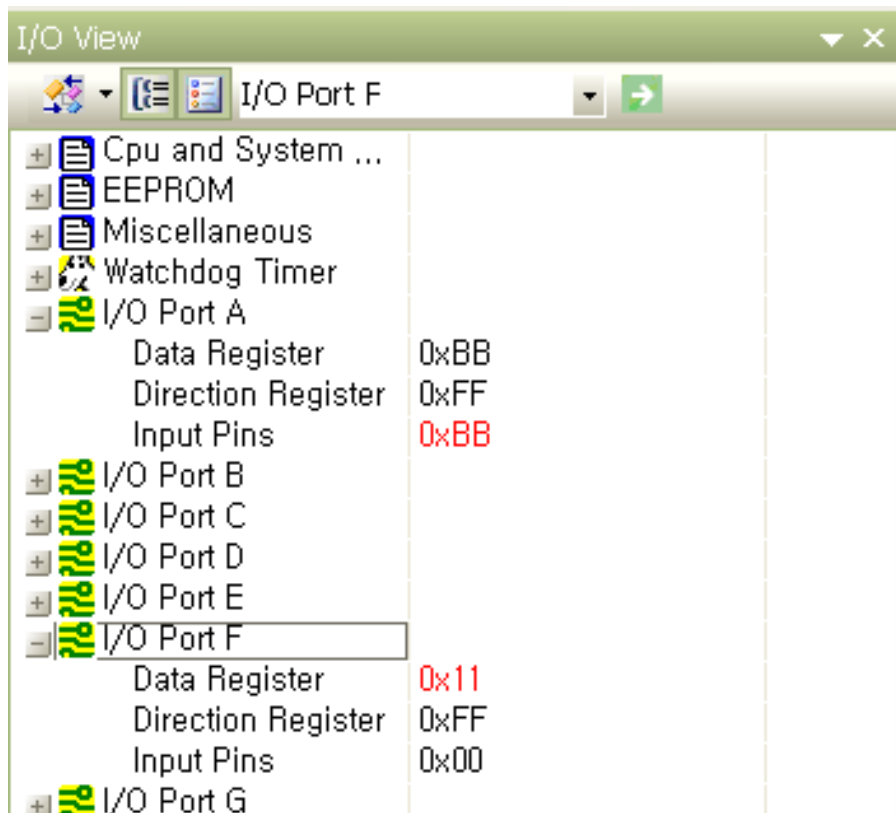


Address	Hex	Hex	Hex	Hex	Hex	Hex	Hex	Hex	Hex	Hex
000060	00	FF	11	00	00	00	00	00	00	00
000068	00	00	00	00	00	00	00	00	00	00
000070	00	00	00	00	00	00	00	00	00	00
000078	00	00	00	00	00	00	00	00	00	00
000080	00	00	00	00	00	00	00	00	00	00
000088	00	00	00	00	00	00	00	00	00	00
000090	00	00	00	00	00	06	00	00	00	00
000098	00	00	00	20	00	06	00	00	00	00
0000A0	00	00	00	00	00	00	00	00	00	00
0000A8	00	00	00	00	00	00	00	00	00	00
0000B0	00	00	00	00	00	00	00	00	00	00

0x62 : PORTF
레지스터 주소

실습 4 : 내부 메모리와 레지스터 이해

- I/O 레지스터 관찰 창
 - ❖ PORTF 레지스터 관찰



Name	Ad...	Value	Bits
PORTF	0x42	0x11	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
DDRF	0x41	0xFF	<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/>
PINF	0x00	0x00	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>



ICAT

Integrated Circuits for Advanced Technology Lab.