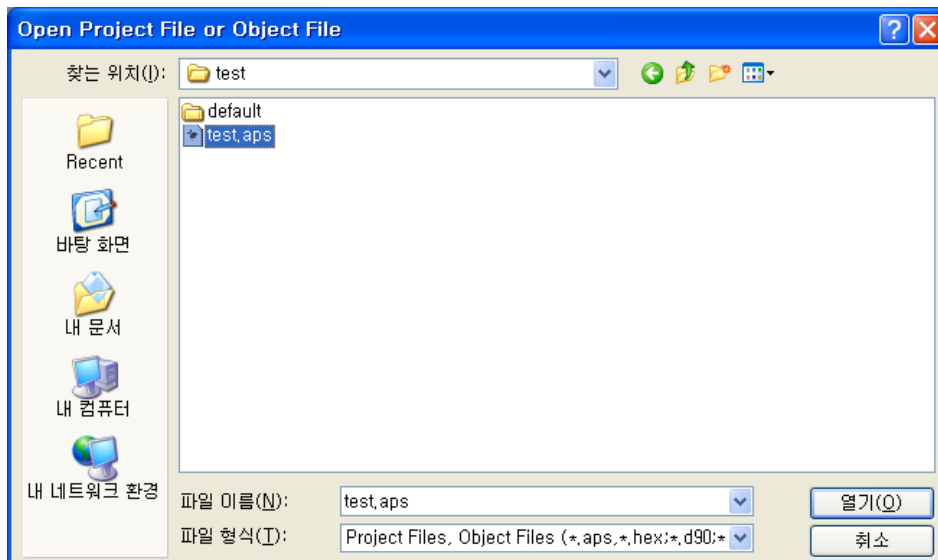
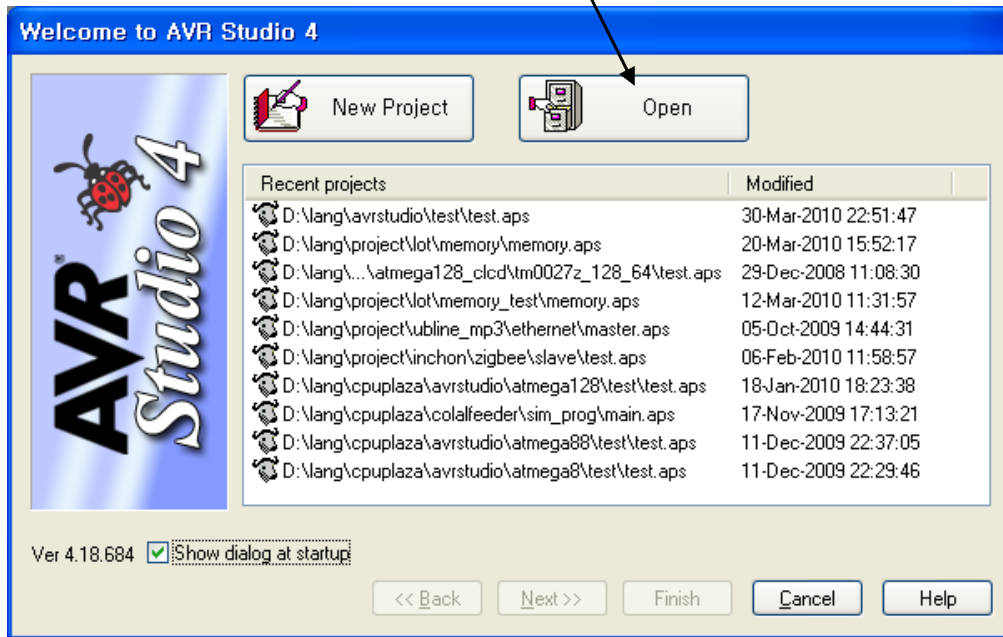


## \* AVR Studio4.18 JTAG ICE로 디버그 하기

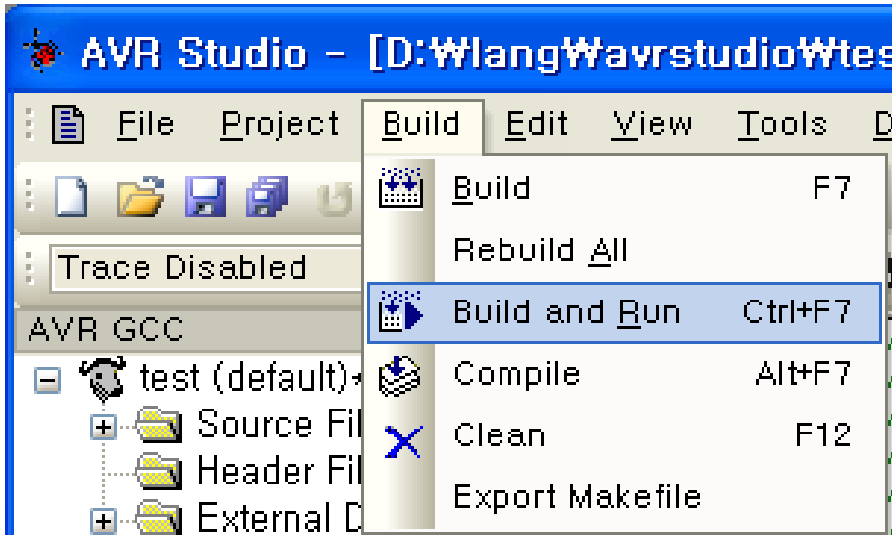
1. AVR Studio를 시작 합니다.



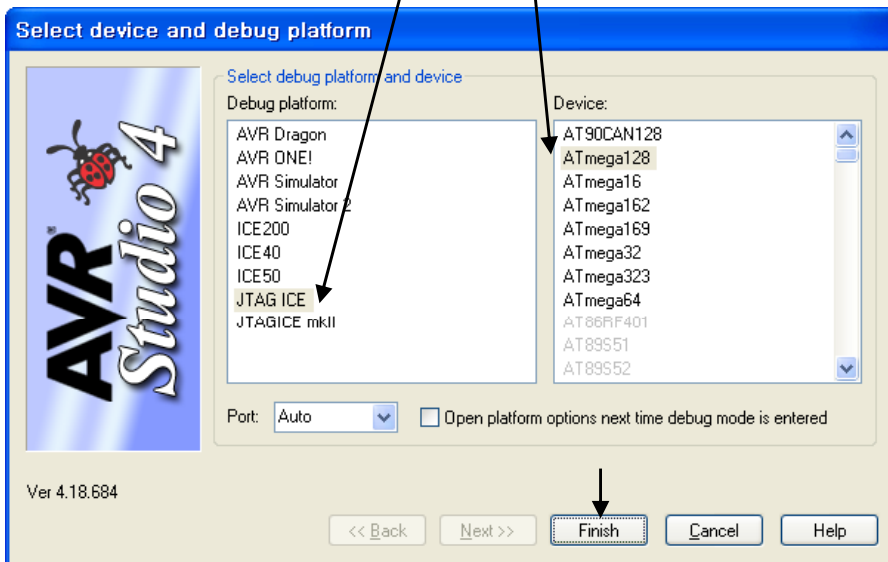
2. Open 버튼을 누른후 기존의 test Project를 Open 합니다.



2. Build and Run을 실행 합니다.



\* 장치요구시 아래 내용 설정(장치명, 사용 CPU)



### 3. Compile 한후 Program이 로딩 된후 디버그 커저가 나타 납니다.

The screenshot shows the AVR Studio IDE with the following components:

- Processor Window:** Lists various registers and their values, such as Program Counter (0x00005F), Stack Pointer (0x10FD), and SREG.
- Code Editor:** Displays a C program for ATmega128. The code includes headers for AVR Studio and defines the main function. A yellow arrow points to the start of the `main` function.
- Build Window:** Shows the compilation output, including the linker script `(.text + .data + .bootloader)` and the data section size `Data: 0 bytes (0.0% Full)`.
- I/O View Window:** Lists hardware components like AD\_CONVERTER, ANALOG\_COMPARATOR, BOOT\_LOAD, CPU, EEPROM, EXTERNAL\_INTERRUPT, JTAG, MISC, PORTA, PORTB, PORTC, PORTD, PORTE, PORTF, and PORTG.

```
사용 CPU : ATmega128-16M
날 짜 : 2006
버 션 : Ver1.0
작 성 자 : www.cuplaza.com
컴파일러 : AVR Studio

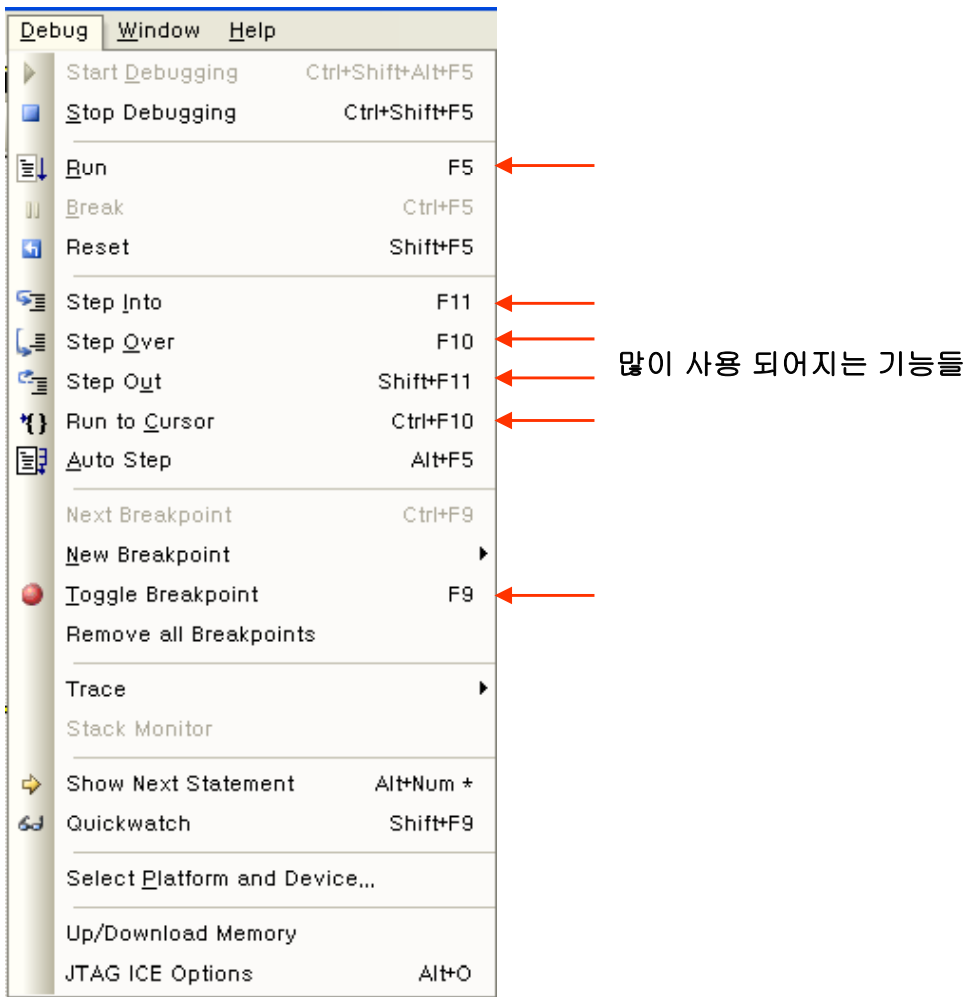
===== 기본 헤더 =====
#include <stdint.h>
#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>

#include "avr/io.h"
#include "avr/interrupt.h"

===== 메인 프로그램 =====
[인수] void
[참고] PORT B를 출력으로 설정하고 펄스를 출력한다.

int main(void)
// PortB 설정
PORTB=0xff; // 포트B 출력 1
DDRB=0xff; // 포트B 출력 설정
while(1){
    PORTB=0x00; // 포트B 출력 0
    PORTB=0xff; // 포트B 출력 1
}
```

4. Debug창의 기능들을 사용 하여 디버그 합니다.



5. 변수를 디버그할때는 해당 변수를 더블 클릭후 오른쪽 마우스를 누른후 메뉴에서 Add Watch 를 선택 합니다.

