Last Time

Digital I/O and the Atmel Mega8s

- DDRx
- PORTx
- PINx

Today

- Project 1
- Bion programming
- Serial I/O

Schedule

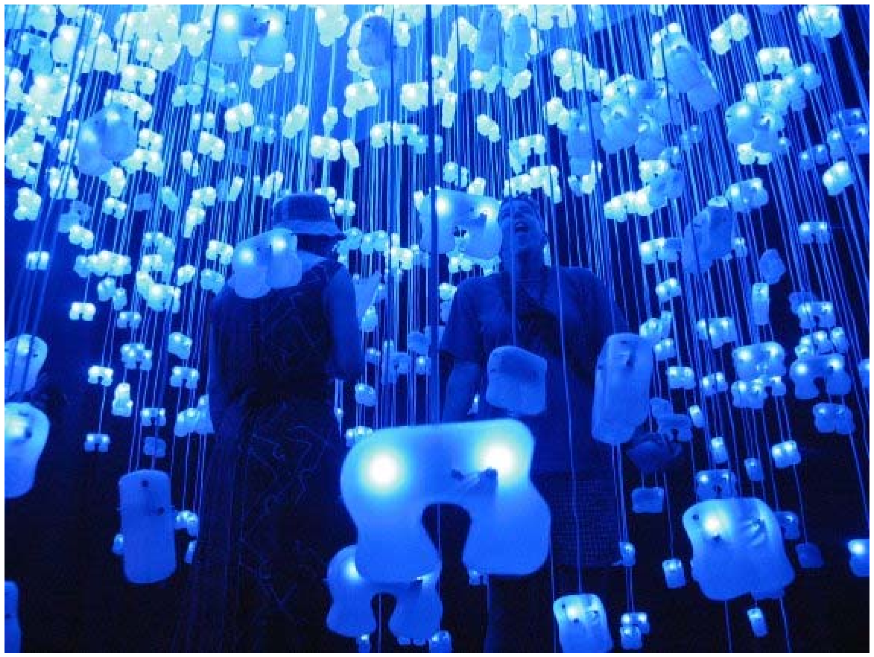
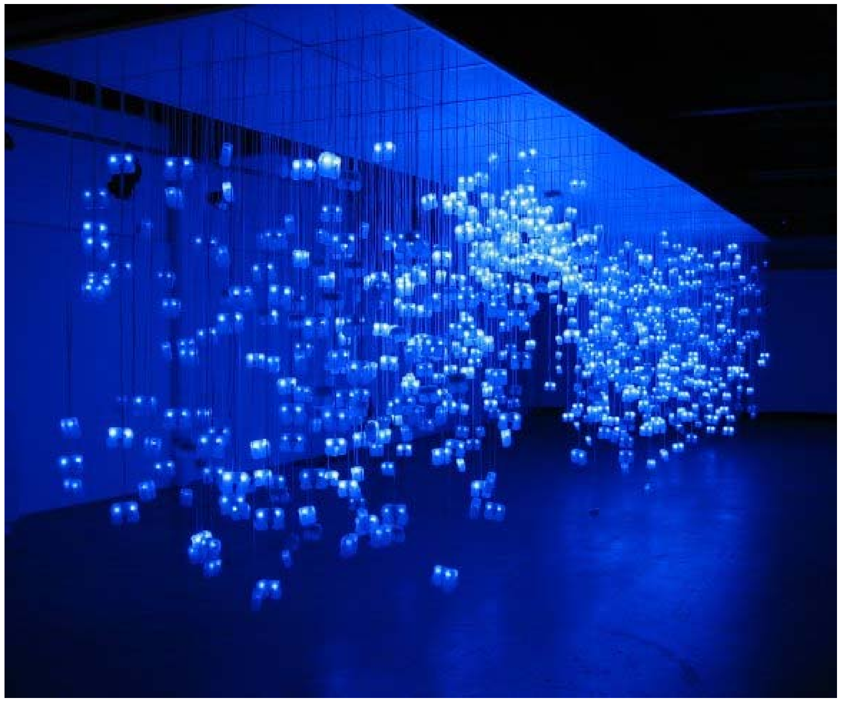
- Project 1: Due March 5th (1 week)
- HW 1 & 2: coming back early next week
- HW 3: out tonight. Due March 10th at the beginning of class
- March 10th: Midterm review
- March 12th: Midterm
  - See prior class web pages for exams and solution sets

Bion

Sensor network:

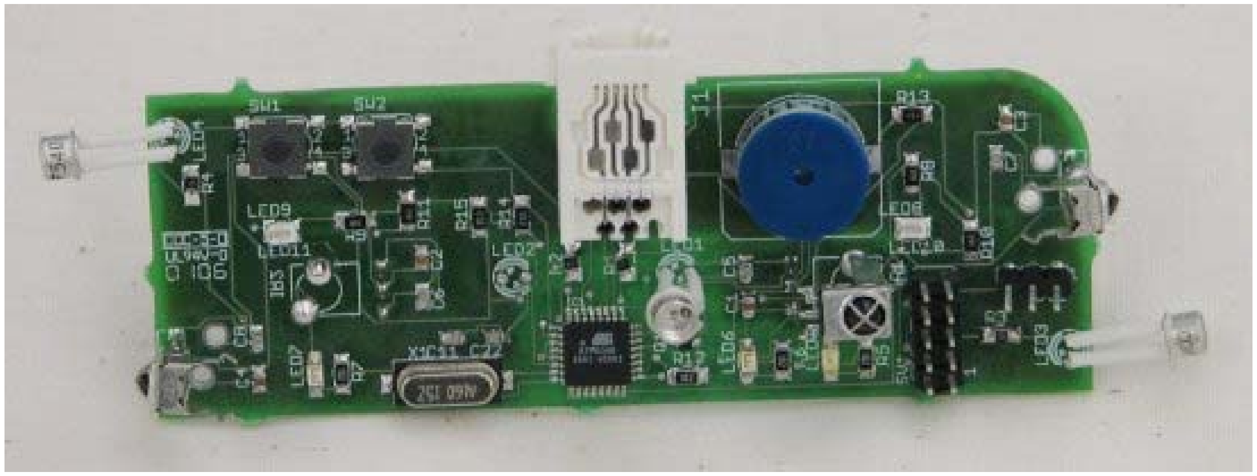
- 1000 sensor nodes
- 3 miles of telephone cable

Wilhelm Reich

Project 1: Digital I/O and Timing

- Control of LEDs and Speaker
  - Precise timing requires timer use
- Respond to button presses



Part 1

- Internal 4-bit (software) counter
- Counter state is reflected by the LEDs
  - Bit 0 (LSB): Blue
  - Bit 1: Green
  - Bit 2: Red
  - Bit 3: Yellow

Part 1

- Each button release:
  - Increment counter
  - Show the new state of the counter with the LEDs

Part 2

- Generate tone with the speaker
  - Different tone for each counter state (higher frequencies for higher values)
  - This tone should be produced continuously (no pauses)

- Speaker is controlled by a digital I/O line
  - So: in one of two states
  - Tones are produced by producing a “square wave” at a given frequency

Required Components

- Modular code
  - E.g., implement a separate function that translates the current counter value into the LED state

Project Administrivia

Due on March 5th

- Demonstrate to me, or Di Wang
- Documented code: hand-in on D2L
  - One copy per 2-person group
- Personal report: distribution of work
  - You will not receive a grade if this is not turned in

Bion Care

- Hold bions on the side of the board (don’t touch the components)
- Minimize the bending of the components
- Don’t let the bion come in contact with metal while it is powered on

- If things get hot: disconnect power immediately and ask for help

Getting Started

See: http://www.cs.ou.edu/~fagg/classes/general/atmel/

Summary:

- (perhaps) Install AVRstudio
- Install WinAVR
- Plug the programmer into your computer
- Plug the programmer into the bion
- Plug the power into the bion
- Create a program

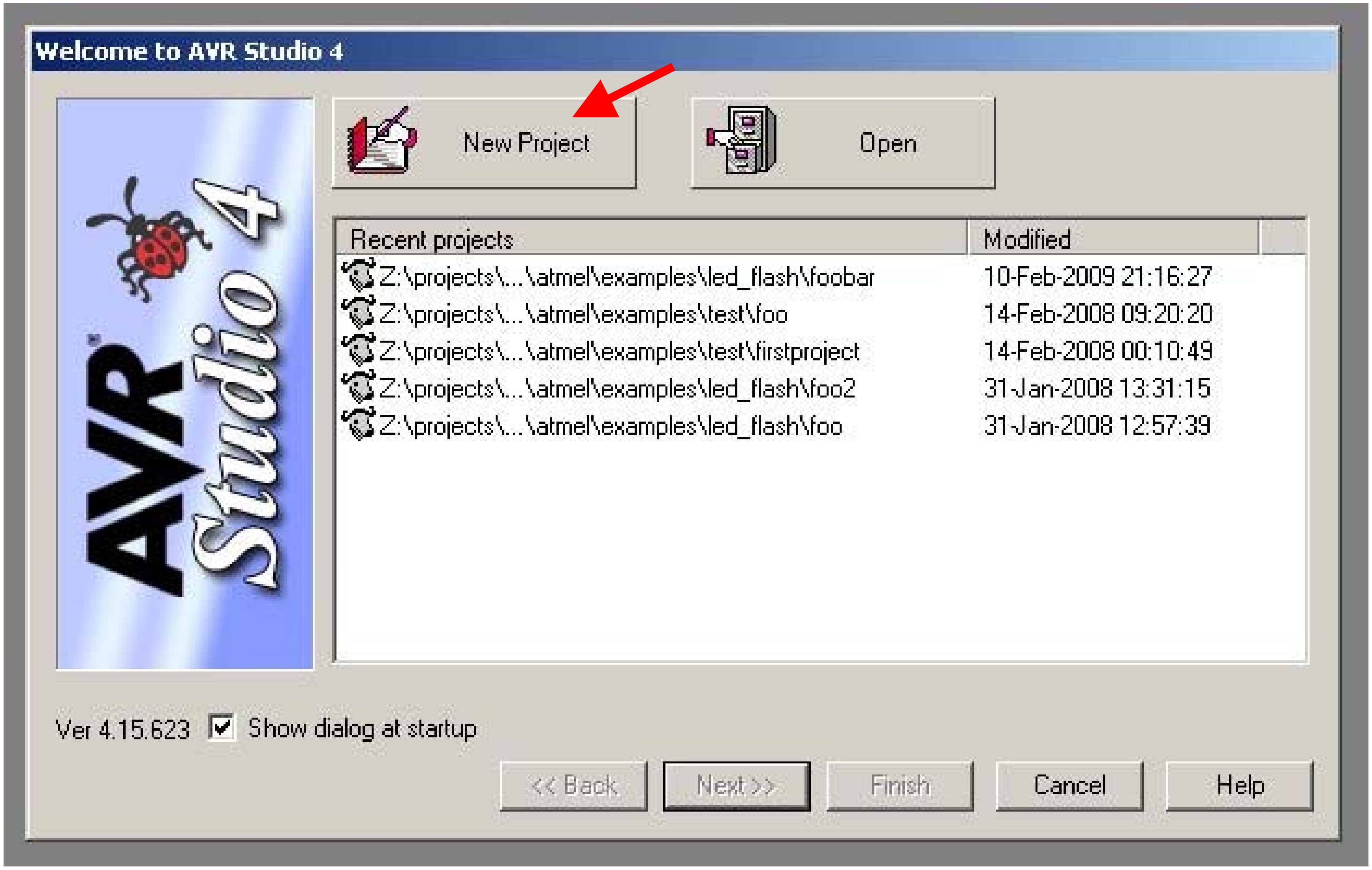
Downloads from Atmel HOWTO

- libou_atmega8.a
- oulib.h
- oulib_serial_buffered.h
- makefile (OSX and linux)

Compiling and Downloading (the easy way)

- Obtain a copy of the “makefile”
  - Modify the “TARGET” line for your program
- Type “make”
  - You should see no errors
- Type “make program”
  - This will download your code to the bion
  - Again, you should see no errors

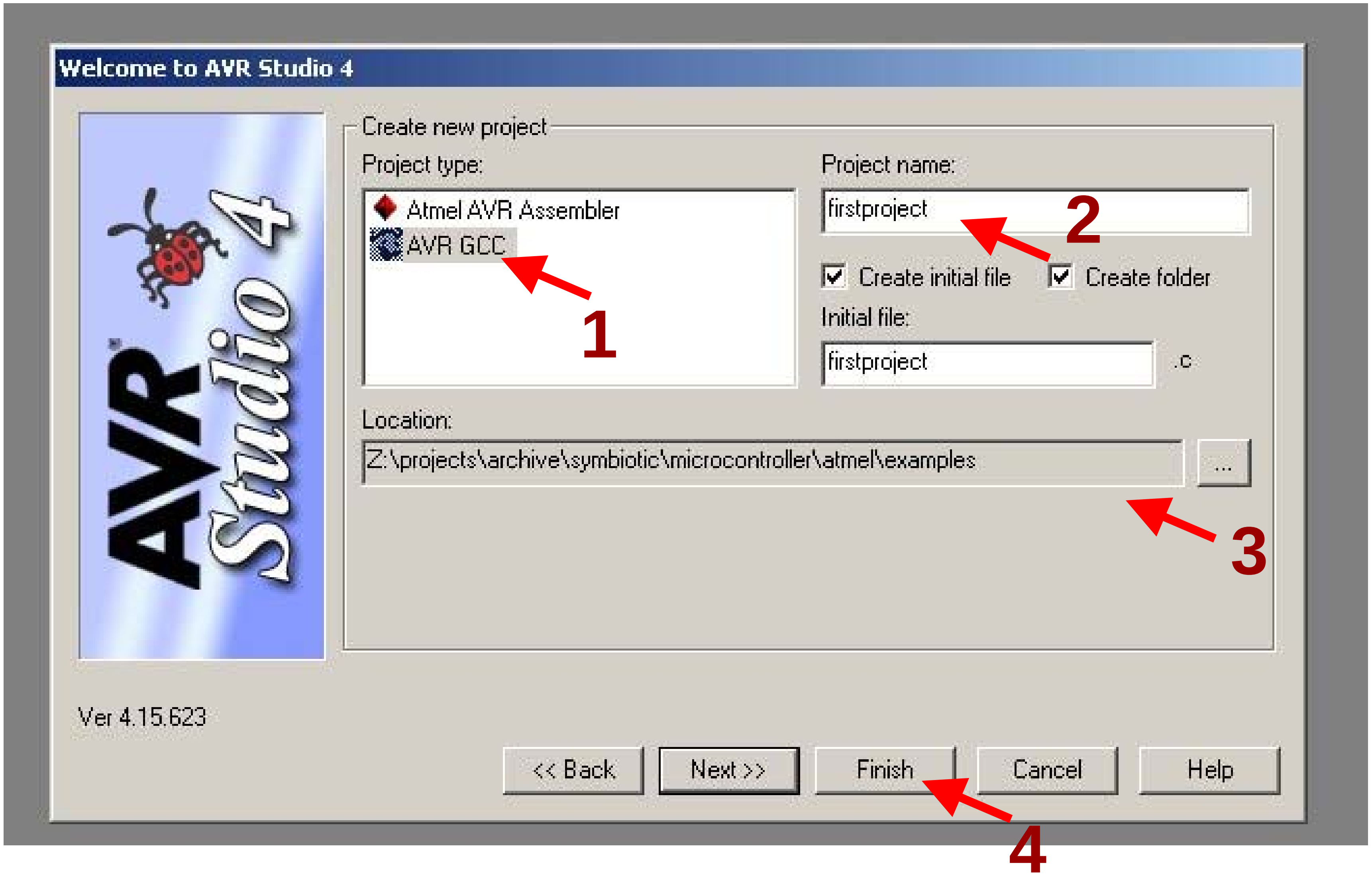
Getting Started



Recent projects	Modified
Z:\projects\...\atmel\examples\led_flash\foobar	10-Feb-2009 21:16:27
Z:\projects\...\atmel\examples\test\foo	14-Feb-2008 09:20:20
Z:\projects\...\atmel\examples\test\firstproject	14-Feb-2008 00:10:49
Z:\projects\...\atmel\examples\led_flash\foo2	31-Jan-2008 13:31:15
Z:\projects\...\atmel\examples\led_flash\foo	31-Jan-2008 12:57:39

Ver 4.15.623 [✓] Show dialog at startup

Project Menu: New Project



Welcome to AVR Studio 4

Create new project

Project type:
- Atmel AVR Assembler
- AVR GCC

Project name: firstproject

[✓] Create initial file [✓] Create folder

Initial file: firstproject .c

Location: Z:\projects\archive\symbiotic\microcontroller\atmel\examples

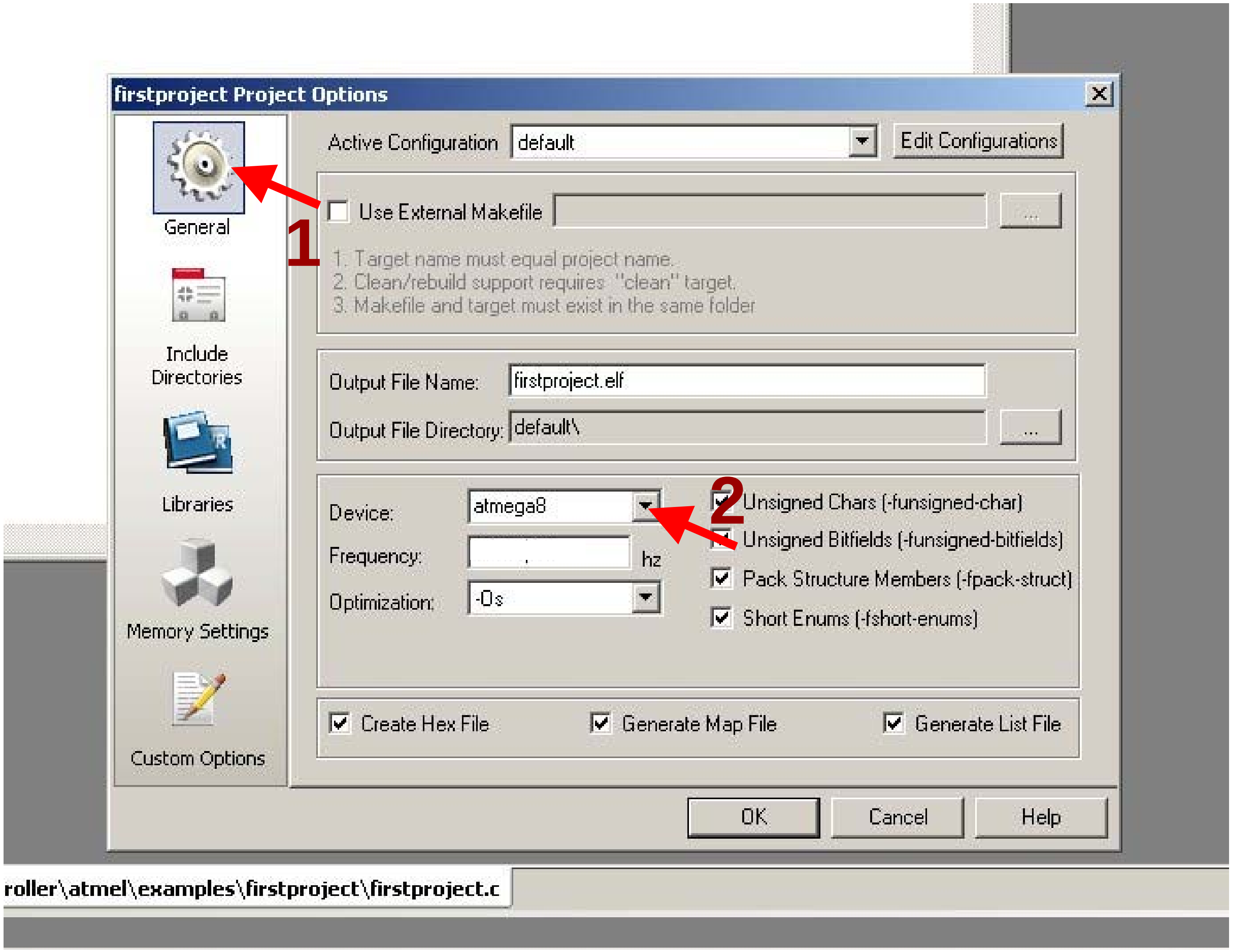
Ver 4.15.623

Back to the OS…

Copy the following to your “firstproject” folder:

- oulib.h
- libou_atmega8.a
- (useful later): oulib_serial_buffered.h

Project Menu: Configuration Options



Active Configuration: default

☐ Use External Makefile

1. Target name must equal project name.
2. Clean/rebuild support requires "clean" target.
3. Makefile and target must exist in the same folder

Output File Name: firstproject.elf

Output File Directory: default\

Device: atmega8

Frequency: hz

Optimization: -Os

☑ Unsigned Chars (-funsigned-char)

☑ Unsigned Bitfields (-funsigned-bitfields)

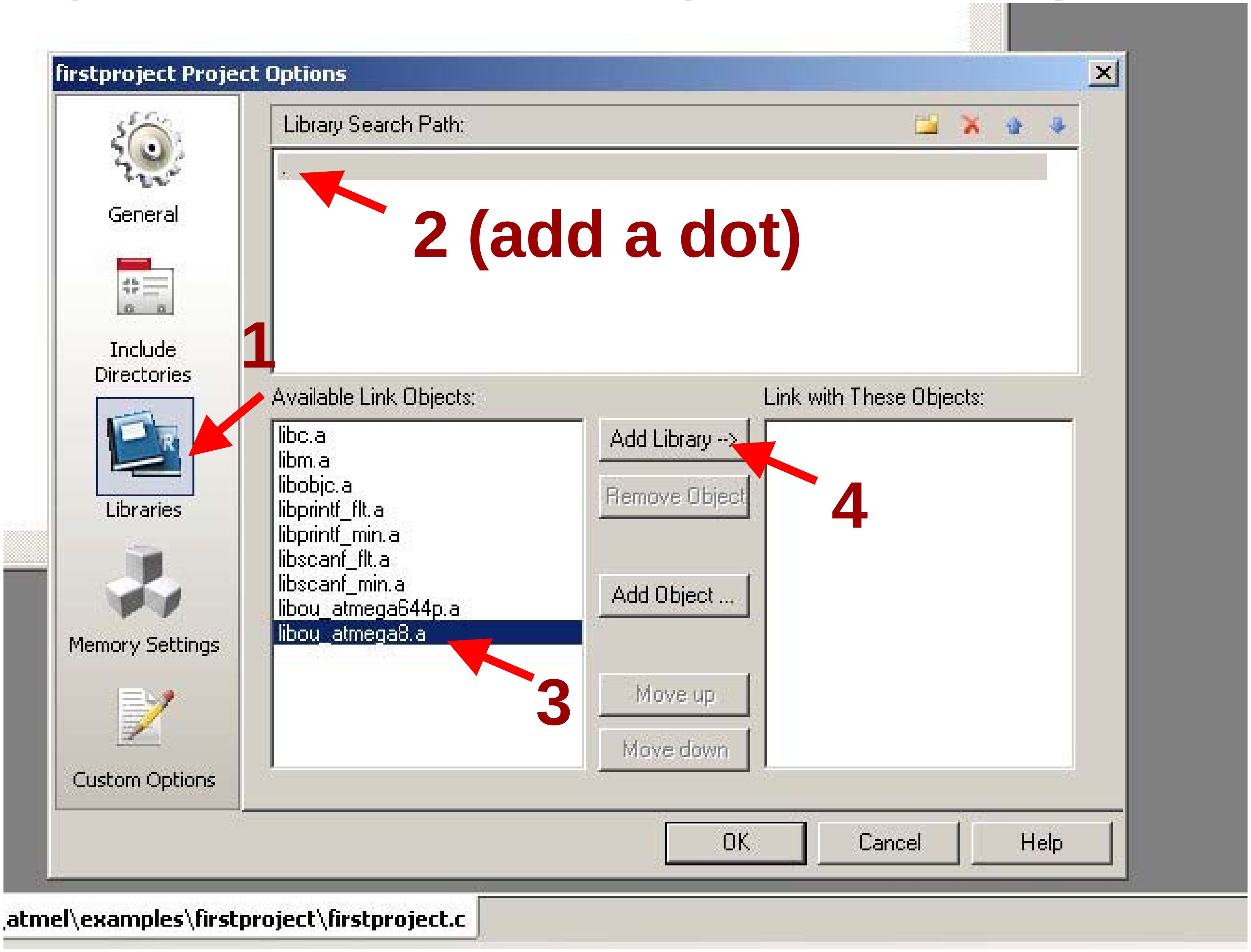
☑ Pack Structure Members (-fpack-struct)

☑ Short Enums (-fshort-enums)

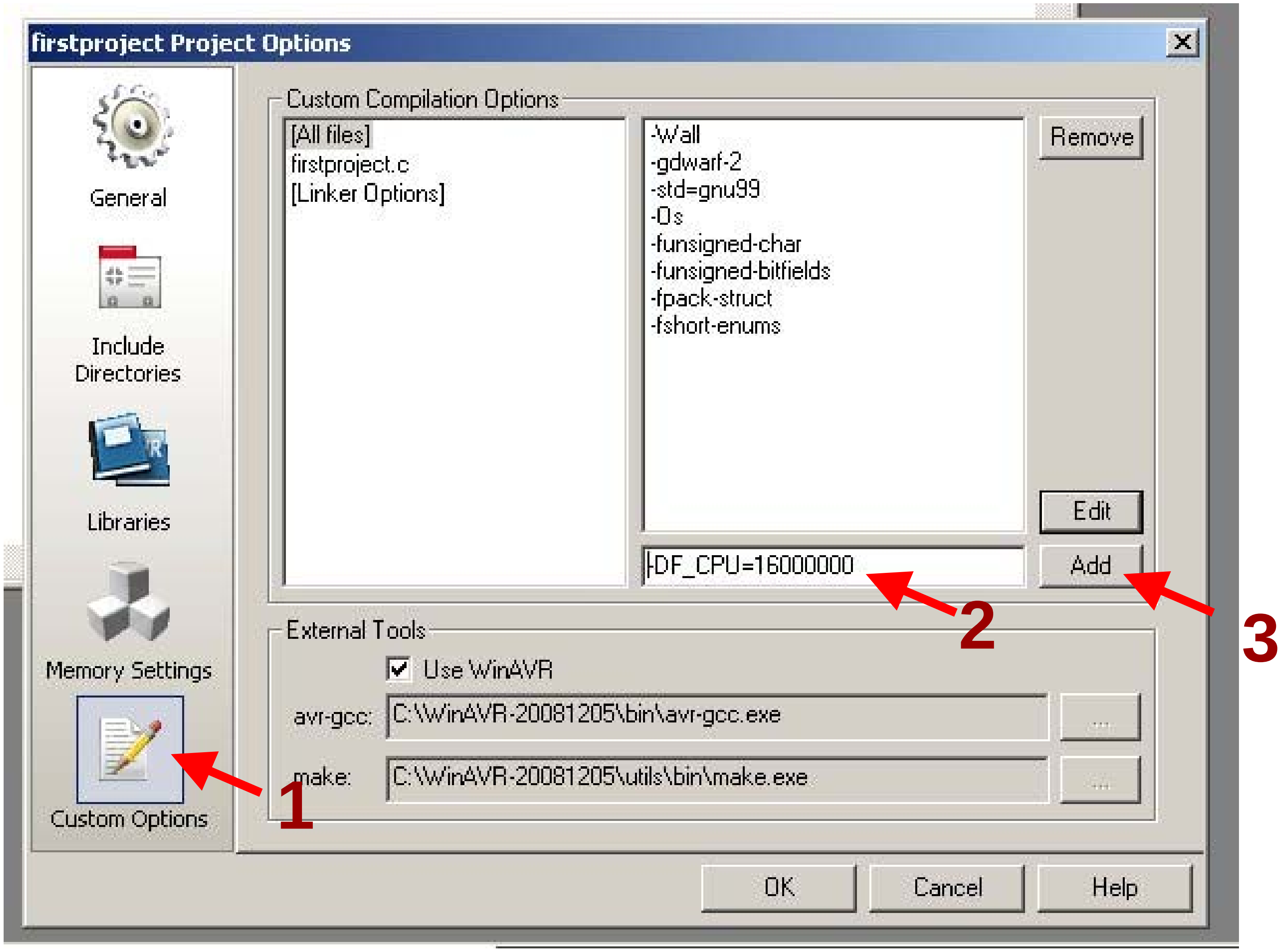
☑ Create Hex File ☑ Generate Map File ☑ Generate List File

roller\atmel\examples\firstproject\firstproject.c

Project Menu: Configuration Options



Project Menu: Configuration Options



firstproject Project Options

Custom Compilation Options

[All files]
firstproject.c
[Linker Options]

```
-Wall
-gdwarf-2
-std=gnu99
-Os
-funsigned-char
-funsigned-bitfields
-fpack-struct
-fshort-enums
```

-DF_CPU=16000000

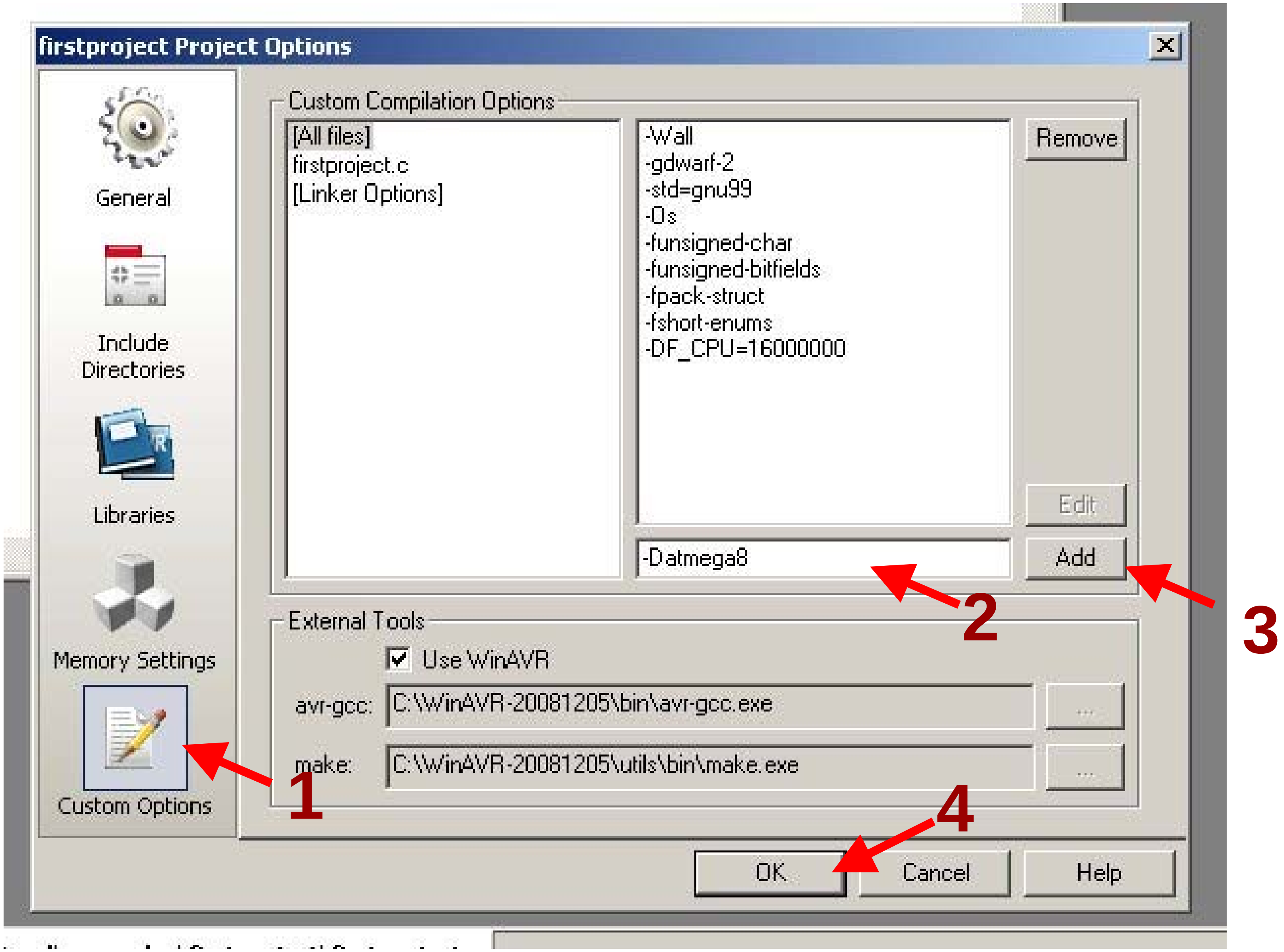
External Tools

[✓] Use WinAVR

avr-gcc: C:\WinAVR-20081205\bin\avr-gcc.exe

make: C:\WinAVR-20081205\utils\bin\make.exe

Project Menu: Configuration Options



firstproject Project Options

Custom Compilation Options

[All files]
firstproject.c
[Linker Options]

```
-Wall
-gdwarf-2
-std=gnu99
-Os
-funsigned-char
-funsigned-bitfields
-fpack-struct
-fshort-enums
-DF_CPU=16000000
```

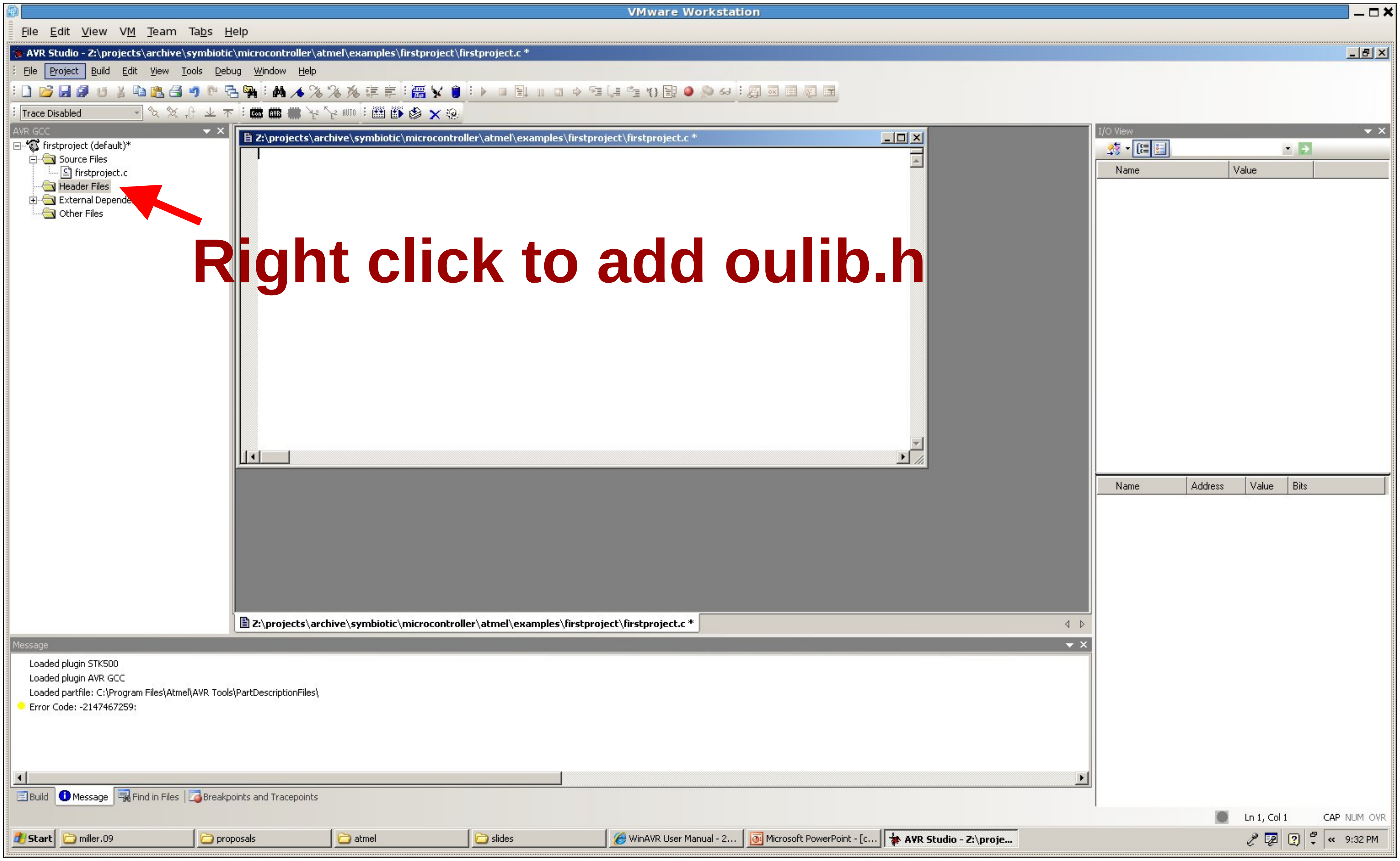
-Datmega8

External Tools

Use WinAVR [✓]

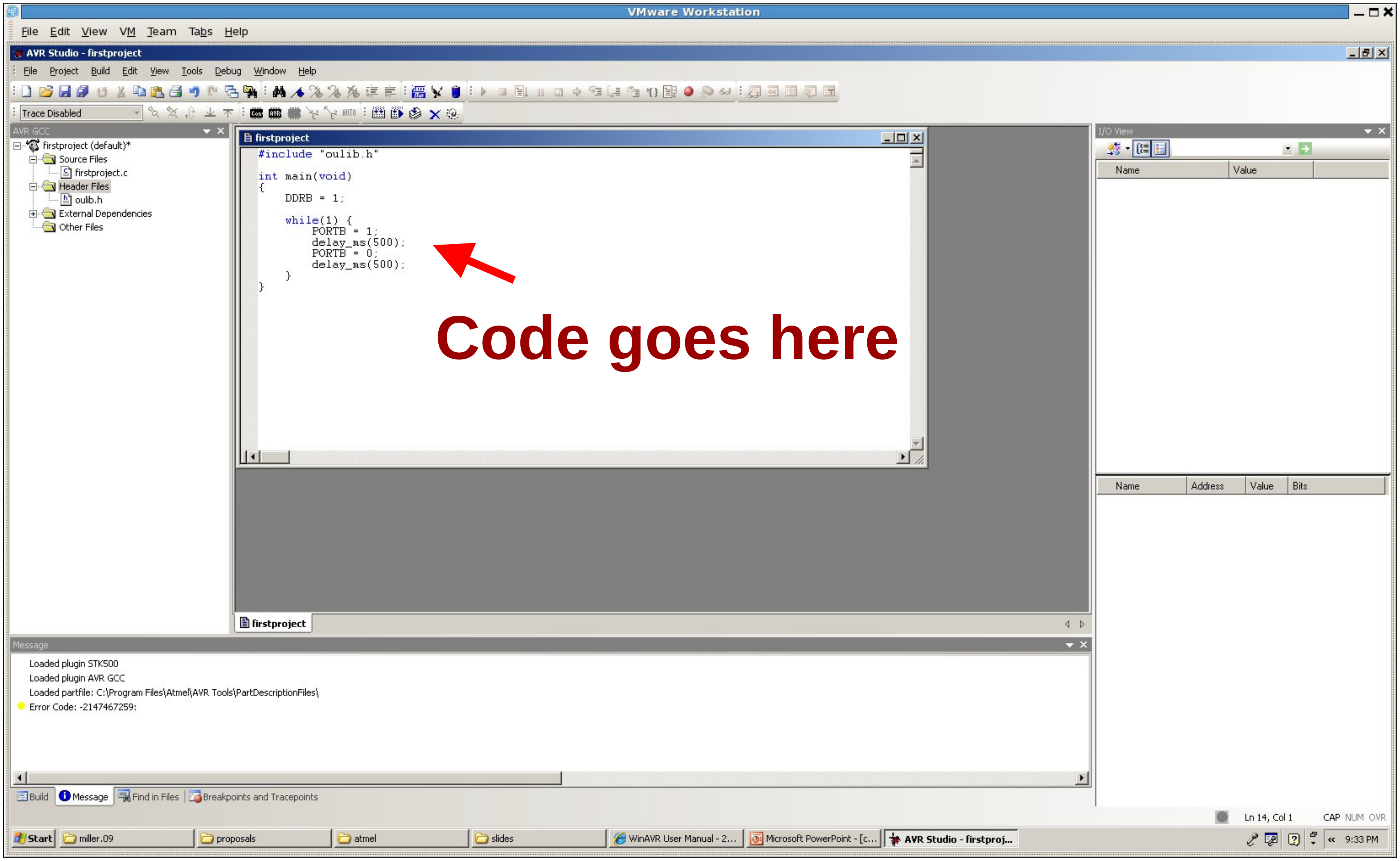
avr-gcc: C:\WinAVR-20081205\bin\avr-gcc.exe

make: C:\WinAVR-20081205\utils\bin\make.exe


VMware Workstation
File Edit View VM Team Tabs Help
AVR Studio - Z:\projects\archive\symbiotic\microcontroller\atmel\examples\firstproject\firstproject.c *
File Project Build Edit View Tools Debug Window Help
Trace Disabled
AVR GCC
firstproject (default)*
Source Files
firstproject.c
Header Files
External Depende
Other Files
Z:\projects\archive\symbiotic\microcontroller\atmel\examples\firstproject\firstproject.c *
I/O View
Name
Value
Address
Bits
Build
Message
Find in Files
Breakpoints and Tracepoints
Ln 1, Col 1
CAP NUM OVR
Start
miller.09
proposals
atmel
slides
WinAVR User Manual - 2...
Microsoft PowerPoint - [c...
AVR Studio - Z:\proje...
9:32 PM

Right click to add oulib.h

```
Loaded plugin STK500
Loaded plugin AVR GCC
Loaded partfile: C:\Program Files\Atmel\AVR Tools\PartDescriptionFiles\
Error Code: -2147467259:
```


VMware Workstation
File Edit View VM Team Tabs Help
AVR Studio - firstproject
File Project Build Edit View Tools Debug Window Help
Trace Disabled
AVR GCC
firstproject (default)*
Source Files
firstproject.c
Header Files
oulib.h
External Dependencies
Other Files
I/O View
Name Value
Name Address Value Bits
Message
Build Message Find in Files Breakpoints and Tracepoints
Ln 14, Col 1 CAP NUM OVR
Start miller.09 proposals atmel slides WinAVR User Manual - 2... Microsoft PowerPoint - [c... AVR Studio - firstproj...
9:33 PM

```
#include "oulib.h"

int main(void)
{
    DDRB = 1;

    while(1) {
        PORTB = 1;
        delay_ms(500);
        PORTB = 0;
        delay_ms(500);
    }
}
```

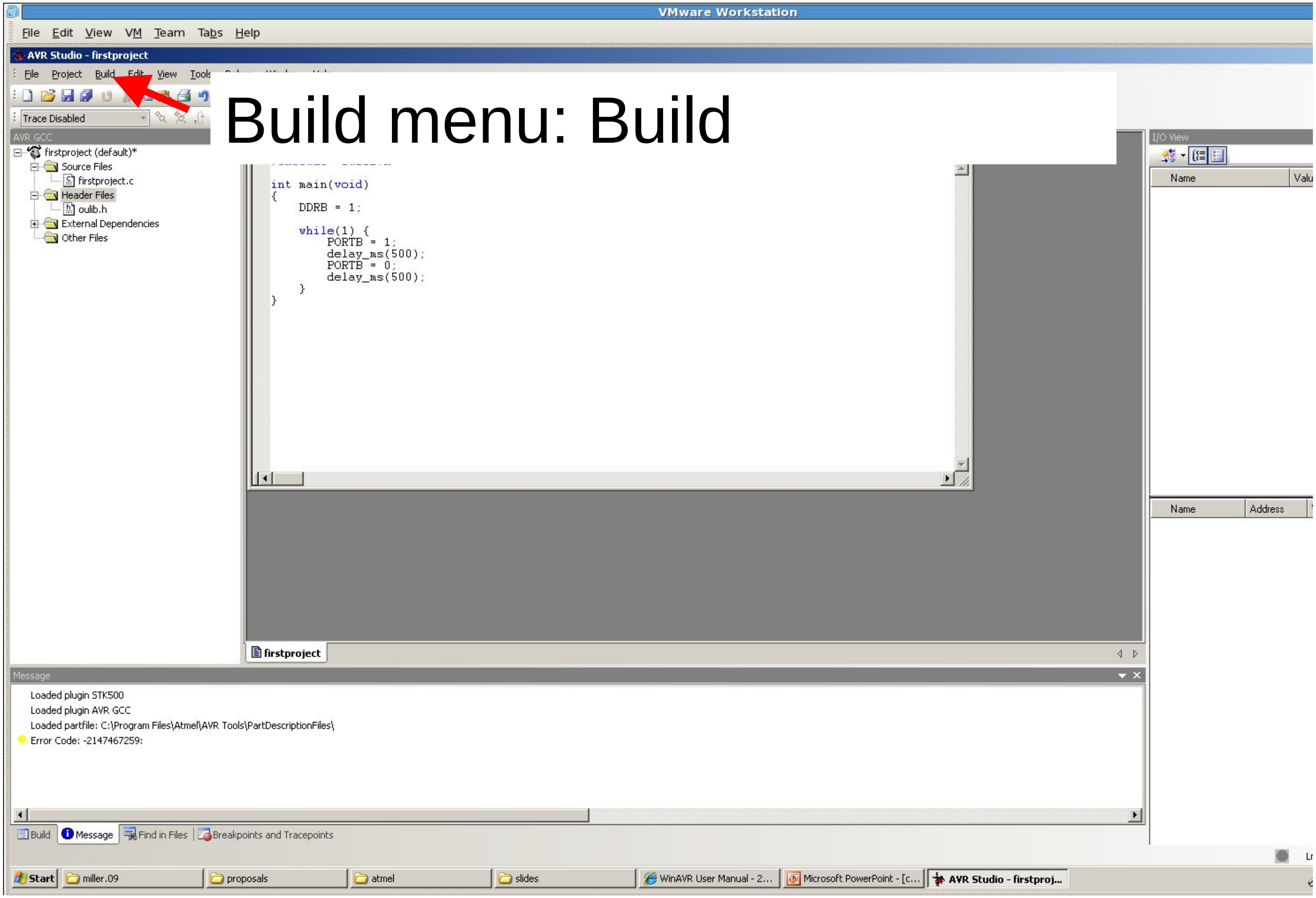
```
Loaded plugin STK500
Loaded plugin AVR GCC
Loaded partfile: C:\Program Files\Atmel\AVR Tools\PartDescriptionFiles\
Error Code: -2147467259:
```

Code goes here

Now for the code…

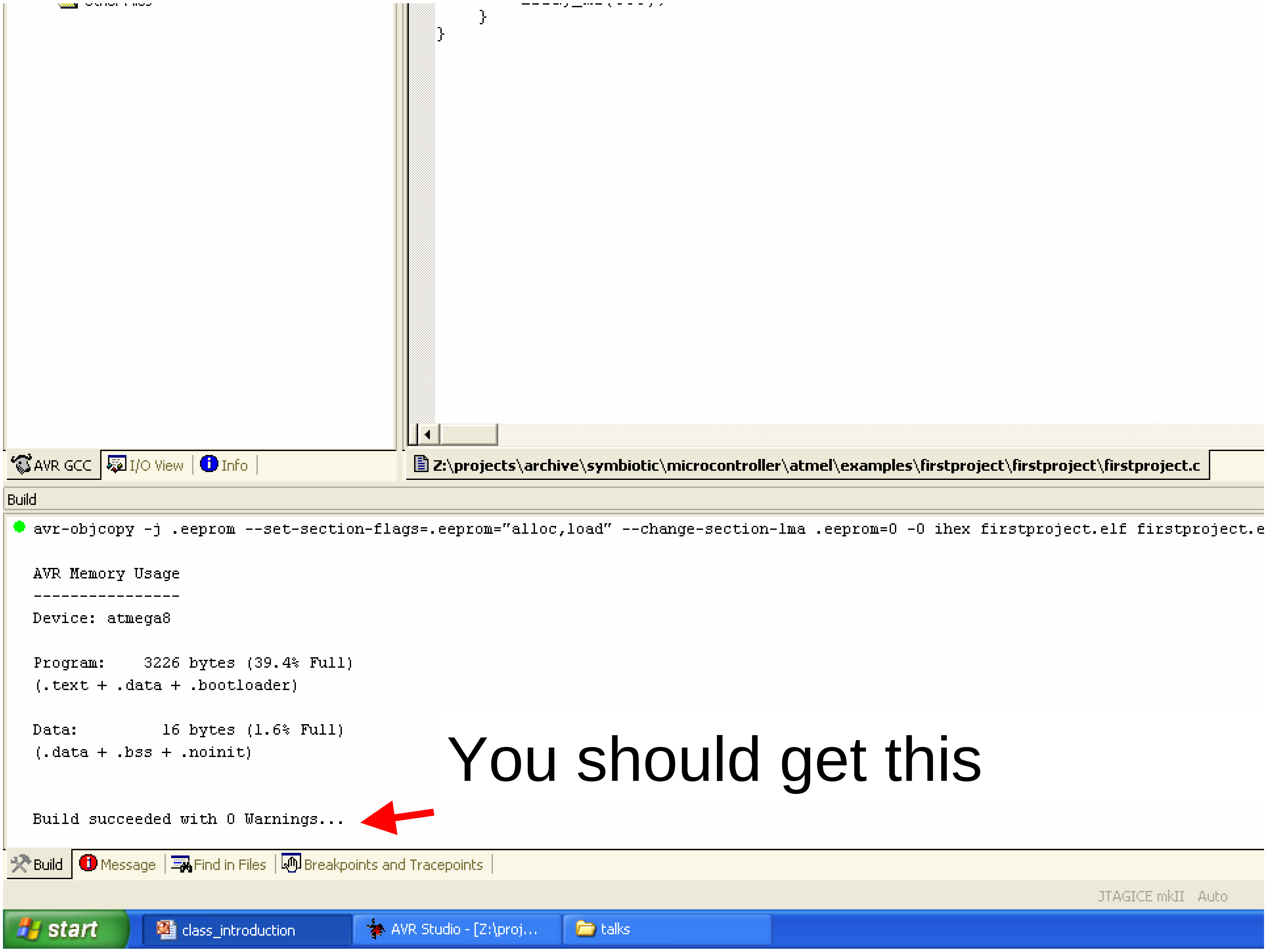
```
#include "oulib.h"

int main(void)
{
   DDRB = 1;

   while(1) {
       PORTB = 1;
       delay_ms(500);
       PORTB = 0;
       delay_ms(500);
   }
}
```

Build menu: Build



```
int main(void)
{
    DDRB = 1;

    while(1) {
        PORTB = 1;
        delay_ms(500);
        PORTB = 0;
        delay_ms(500);
    }
}
```

```
Loaded plugin STK500
Loaded plugin AVR GCC
Loaded partfile: C:\Program Files\Atmel\AVR Tools\PartDescriptionFiles\
Error Code: -2147467259:
```


AVR GCC
I/O View
Info
Z:\projects\archive\symbiotic\microcontroller\atmel\examples\firstproject\firstproject\firstproject.c
Build
Build
Message
Find in Files
Breakpoints and Tracepoints
JTAGICE mkII Auto
start
class_introduction
AVR Studio - [Z:\proj...
talks

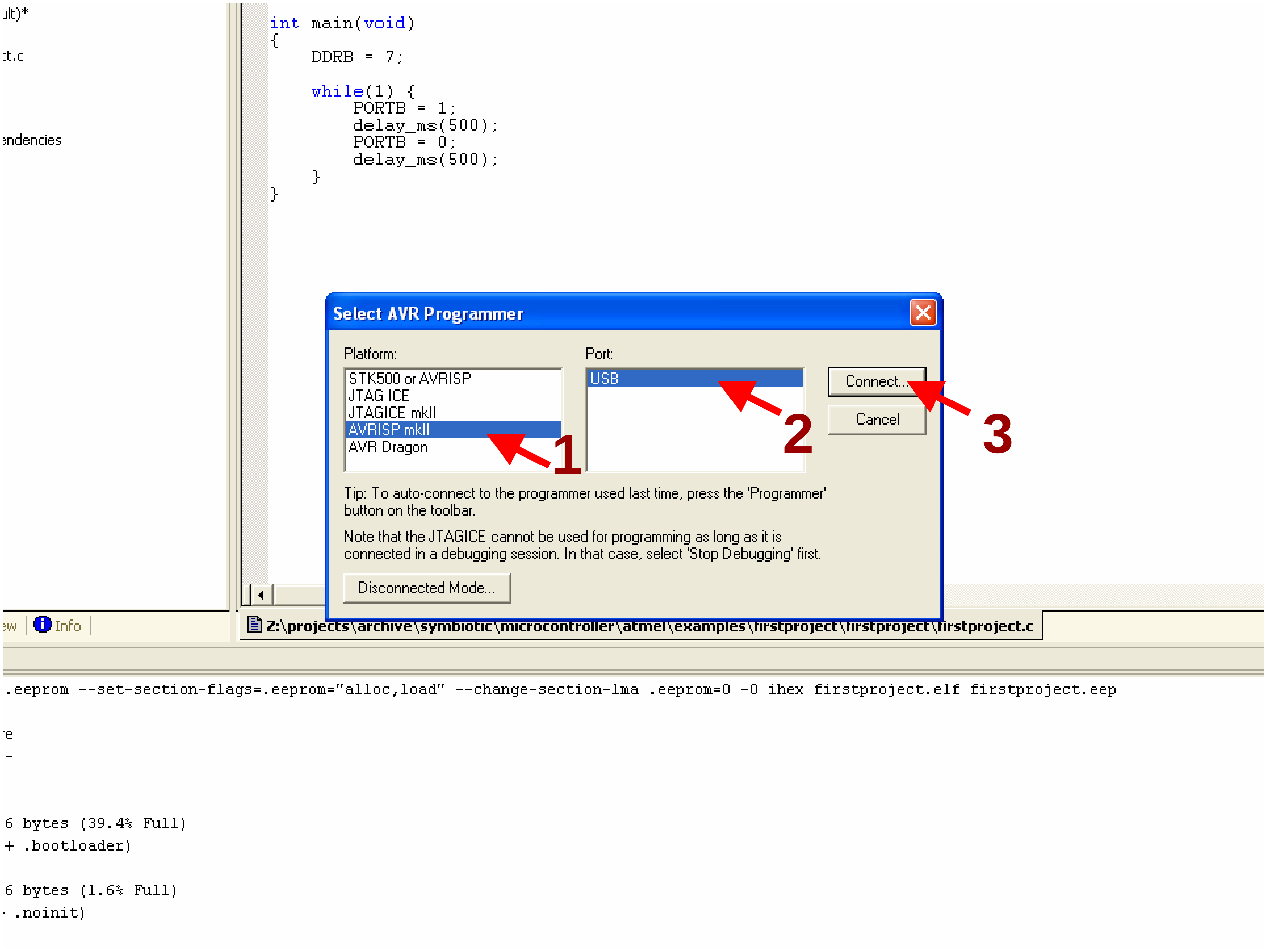
```
    }
}
```

```
avr-objcopy -j .eeprom --set-section-flags=.eeprom="alloc,load" --change-section-lma .eeprom=0 -O ihex firstproject.elf firstproject.e

AVR Memory Usage
----------------
Device: atmega8

Program:    3226 bytes (39.4% Full)
(.text + .data + .bootloader)

Data:         16 bytes (1.6% Full)
(.data + .bss + .noinit)


Build succeeded with 0 Warnings...
```

You should get this

Now We Are Ready…

- Plug the programmer into the bion (If it is not already)
- Power up the bion
- And download the program…
  - Tools Menu: AVR: Connect



```
int main(void)
{
    DDRB = 7;

    while(1) {
        PORTB = 1;
        delay_ms(500);
        PORTB = 0;
        delay_ms(500);
    }
}
```

Select AVR Programmer

Platform:	Port:
STK500 or AVRISP	[✓ USB]
JTAG ICE	
JTAGICE mkII	
[✓ AVRISP mkII]	
AVR Dragon	

Connect...

Cancel

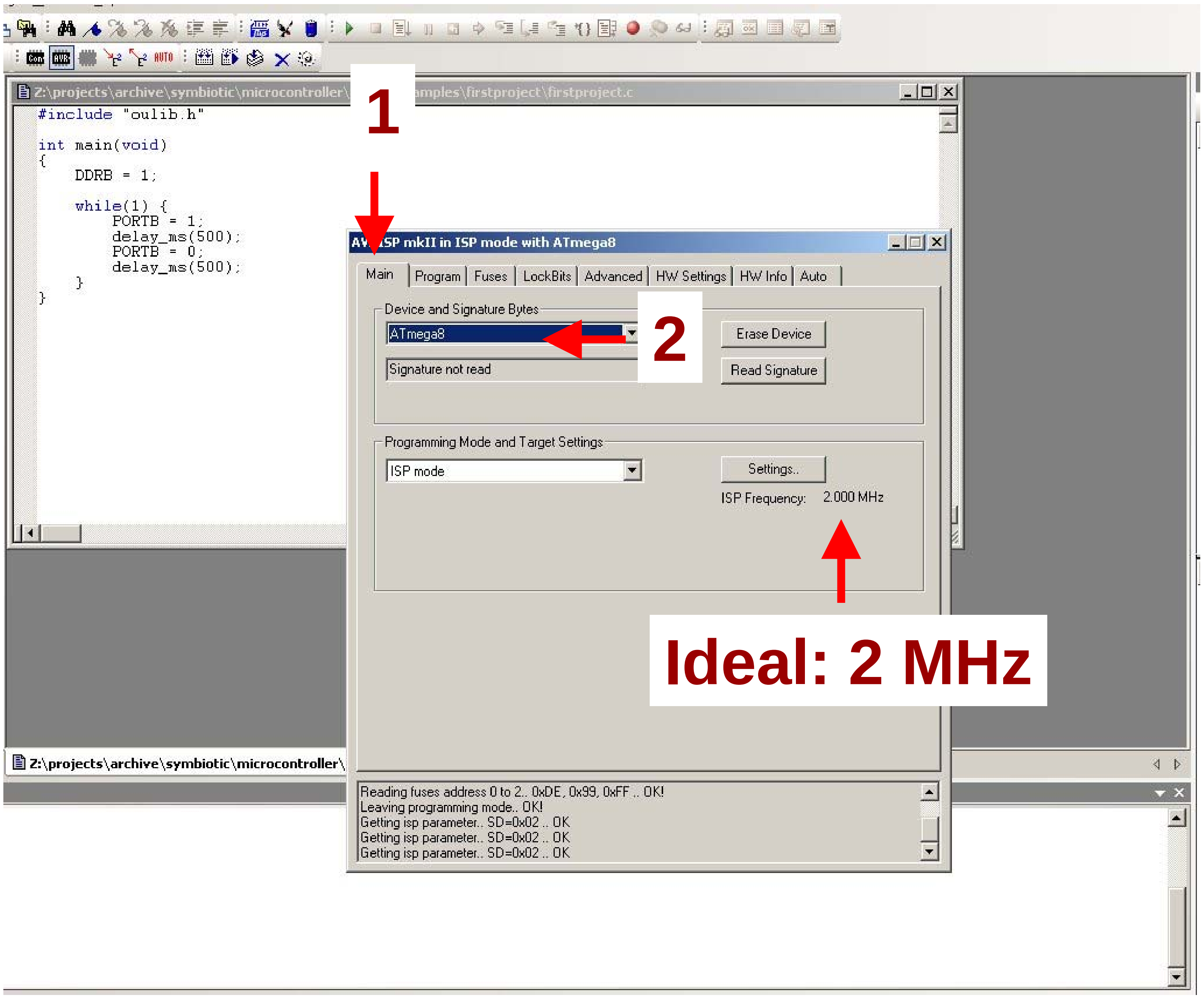
Tip: To auto-connect to the programmer used last time, press the 'Programmer' button on the toolbar.

Note that the JTAGICE cannot be used for programming as long as it is connected in a debugging session. In that case, select 'Stop Debugging' first.

Disconnected Mode...

Z:\projects\archive\symbiotic\microcontroller\atmel\examples\firstproject\firstproject\firstproject\firstproject.c

```
.eeprom --set-section-flags=.eeprom="alloc,load" --change-section-lma .eeprom=0 -O ihex firstproject.elf firstproject.eep

e
-


6 bytes (39.4% Full)
+ .bootloader)

6 bytes (1.6% Full)
· .noinit)
```


1
2
Z:\projects\archive\symbiotic\microcontroller\ ... amples\firstproject\firstproject.c
AVR ISP mkII in ISP mode with ATmega8
Main
Program
Fuses
LockBits
Advanced
HW Settings
HW Info
Auto
Device and Signature Bytes
Erase Device
Read Signature
Programming Mode and Target Settings
Settings..
Z:\projects\archive\symbiotic\microcontroller\

```
#include "oulib.h"

int main(void)
{
    DDRB = 1;

    while(1) {
        PORTB = 1;
        delay_ms(500);
        PORTB = 0;
        delay_ms(500);
    }
}
```

ATmega8

Signature not read

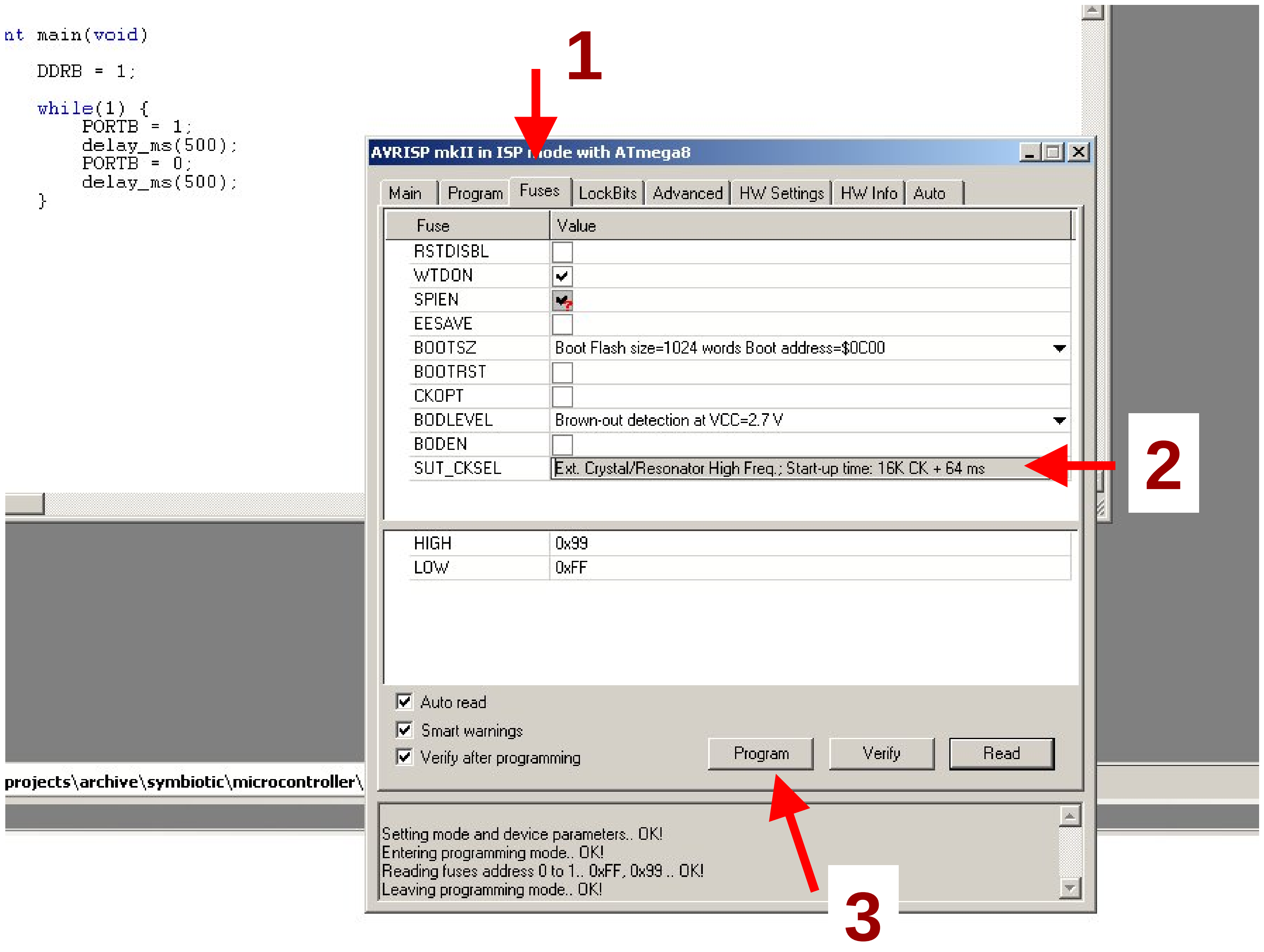
ISP mode

ISP Frequency: 2.000 MHz

```
Reading fuses address 0 to 2.. 0xDE, 0x99, 0xFF .. OK!
Leaving programming mode.. OK!
Getting isp parameter.. SD=0x02 .. OK
Getting isp parameter.. SD=0x02 .. OK
Getting isp parameter.. SD=0x02 .. OK
```

Ideal: 2 MHz

(should only need to do this once)



```
nt main(void)

   DDRB = 1;

   while(1) {
       PORTB = 1;
       delay_ms(500);
       PORTB = 0;
       delay_ms(500);
   }
```

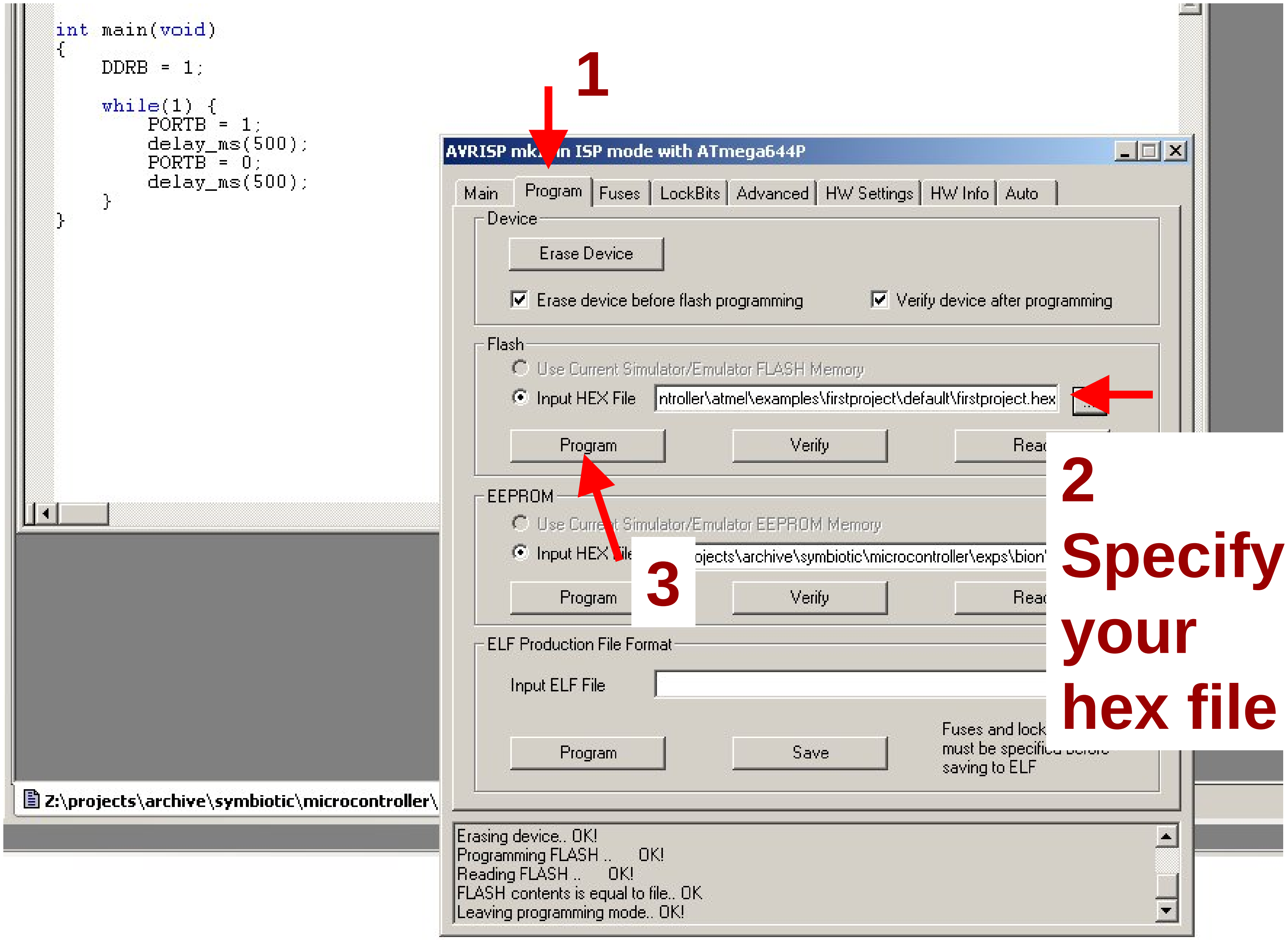
1
2
3
AVRISP mkII in ISP mode with ATmega8
Main
Program
Fuses
LockBits
Advanced
HW Settings
HW Info
Auto
Program
Verify
Read
projects\archive\symbiotic\microcontroller\

Fuse	Value
RSTDISBL	☐
WTDON	✓
SPIEN	✓
EESAVE	☐
BOOTSZ	Boot Flash size=1024 words Boot address=$0C00
BOOTRST	☐
CKOPT	☐
BODLEVEL	Brown-out detection at VCC=2.7 V
BODEN	☐
SUT_CKSEL	Ext. Crystal/Resonator High Freq.; Start-up time: 16K CK + 64 ms

HIGH	0x99
LOW	0xFF

☑ Auto read
☑ Smart warnings
☑ Verify after programming

```
Setting mode and device parameters.. OK!
Entering programming mode.. OK!
Reading fuses address 0 to 1.. 0xFF, 0x99 .. OK!
Leaving programming mode.. OK!
```



```
int main(void)
{
    DDRB = 1;

    while(1) {
        PORTB = 1;
        delay_ms(500);
        PORTB = 0;
        delay_ms(500);
    }
}
```

Z:\projects\archive\symbiotic\microcontroller\

AVRISP mk[illegible]in ISP mode with ATmega644P

Flash: Input HEX File: ntroller\atmel\examples\firstproject\default\firstproject.hex

EEPROM: Input HEX File: ojects\archive\symbiotic\microcontroller\exps\bion

[✓] Erase device before flash programming
[✓] Verify device after programming

```
Erasing device.. OK!
Programming FLASH ..      OK!
Reading FLASH ..      OK!
FLASH contents is equal to file.. OK
Leaving programming mode.. OK!
```

1
2
Specify your hex file
3
Main
Program
Fuses
LockBits
Advanced
HW Settings
HW Info
Auto
Device
Erase Device
Flash
Use Current Simulator/Emulator FLASH Memory
Program
Verify
Read
EEPROM
Use Current Simulator/Emulator EEPROM Memory
ELF Production File Format
Input ELF File
Save
Fuses and lock bits must be specified before saving to ELF

Flashing?

Your program will start executing as soon as the download is complete …

Your green Light Emitting Diode should be blinking at 1 Hertz (once per second)