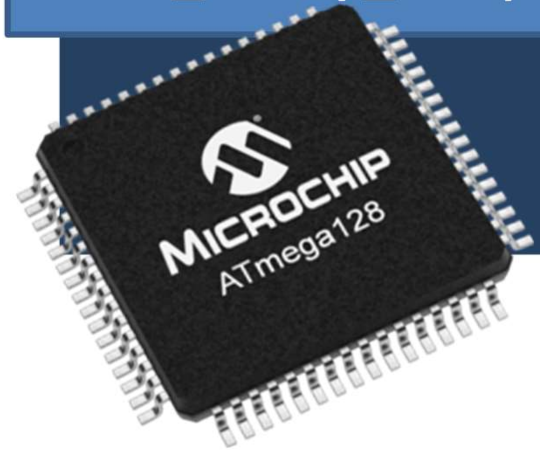
2023년 IoT기반 스마트 솔루션 개발자 양성과정



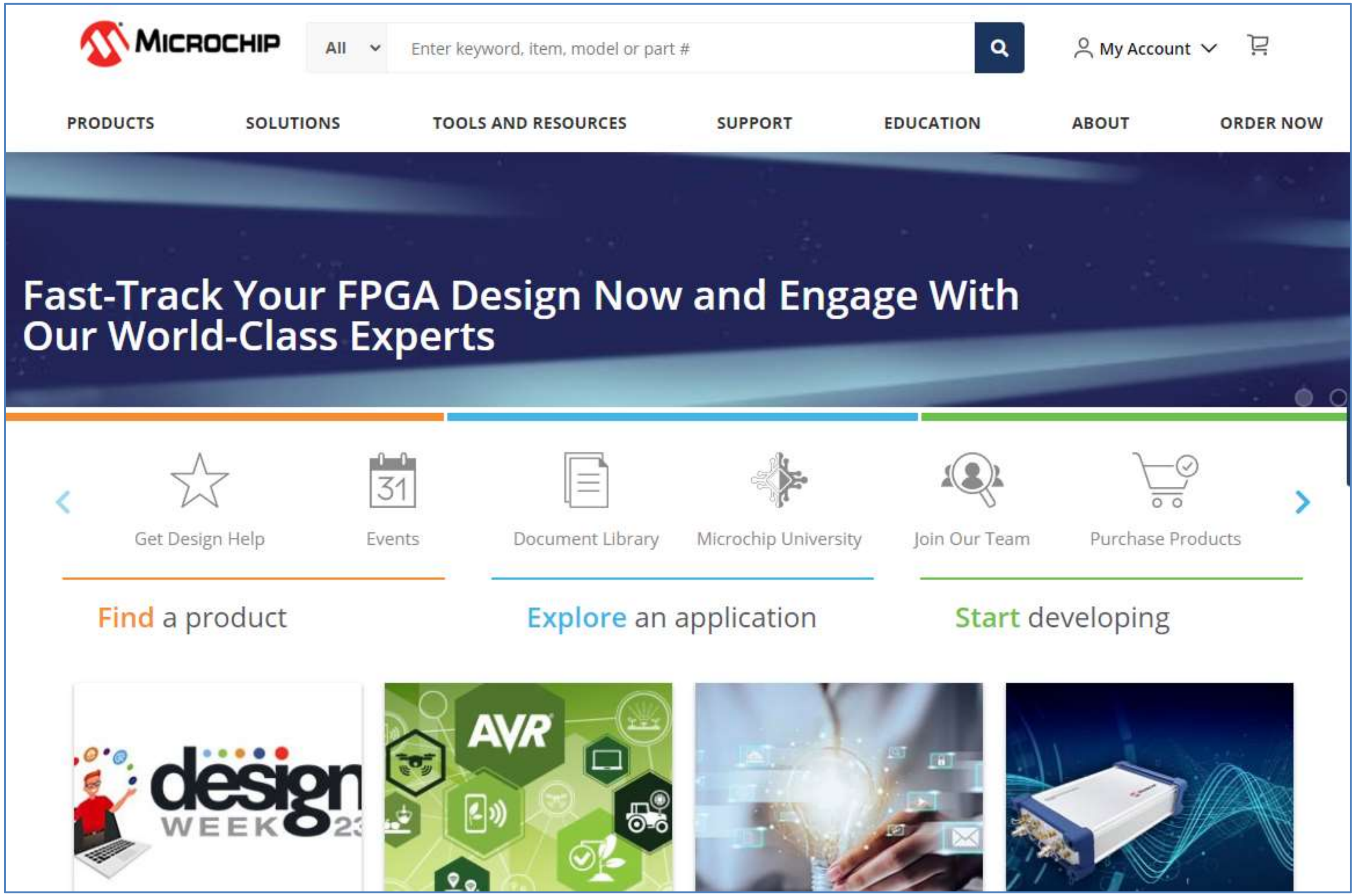
Embedded Application

4-ATmel Studio 7

담당 교수 : 윤 종 이
010-9577-1696
ojo1696@naver.com
https://cafe.naver.com/yoons2023

충북대학교 공동훈련센터

www.microchip.com



MICROCHIP

All

Enter keyword, item, model or part #

My Account

PRODUCTS SOLUTIONS TOOLS AND RESOURCES SUPPORT EDUCATION ABOUT ORDER NOW

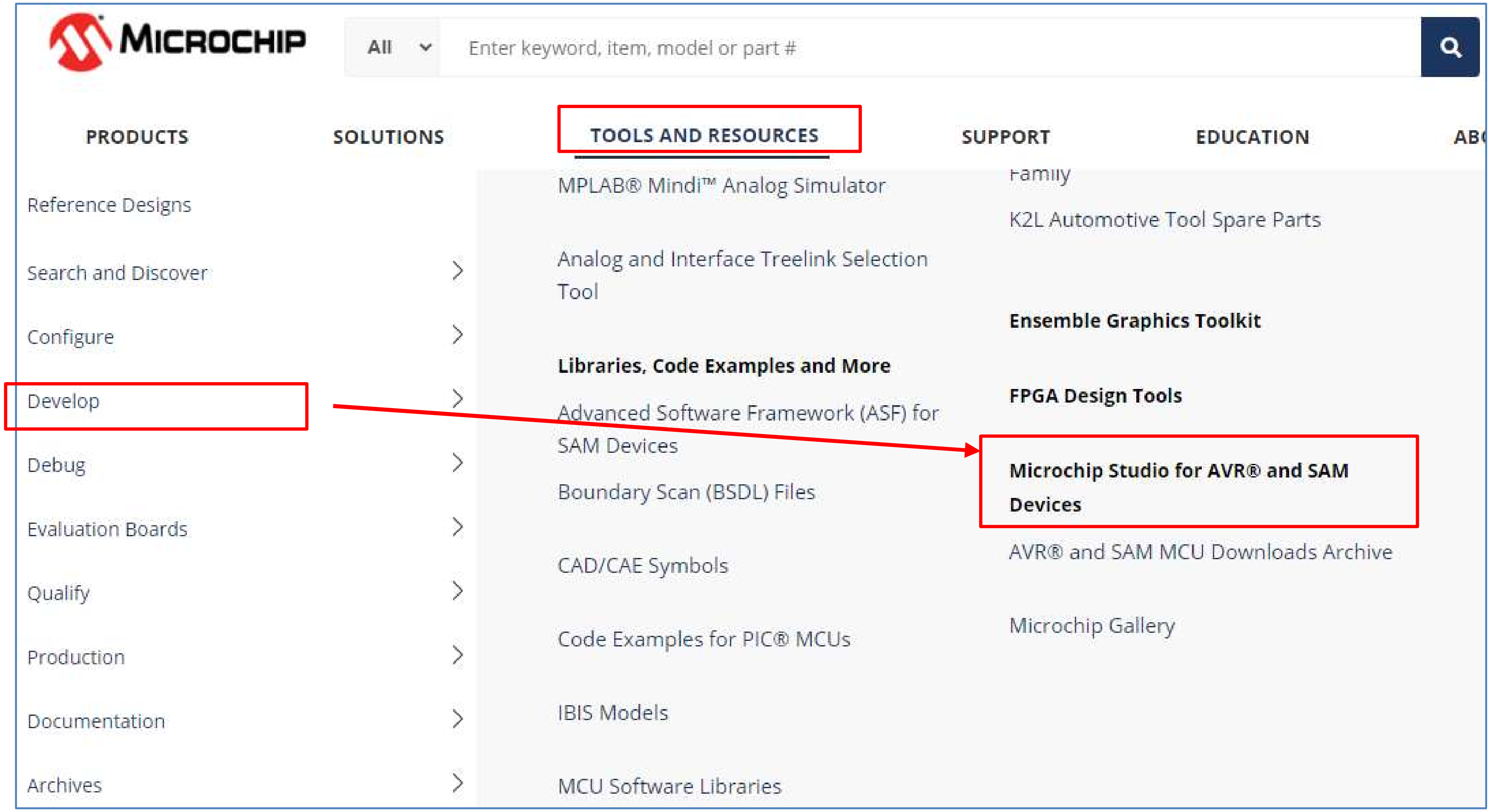
Fast-Track Your FPGA Design Now and Engage With Our World-Class Experts

Get Design Help · Events · Document Library · Microchip University · Join Our Team · Purchase Products

Find a product · Explore an application · Start developing

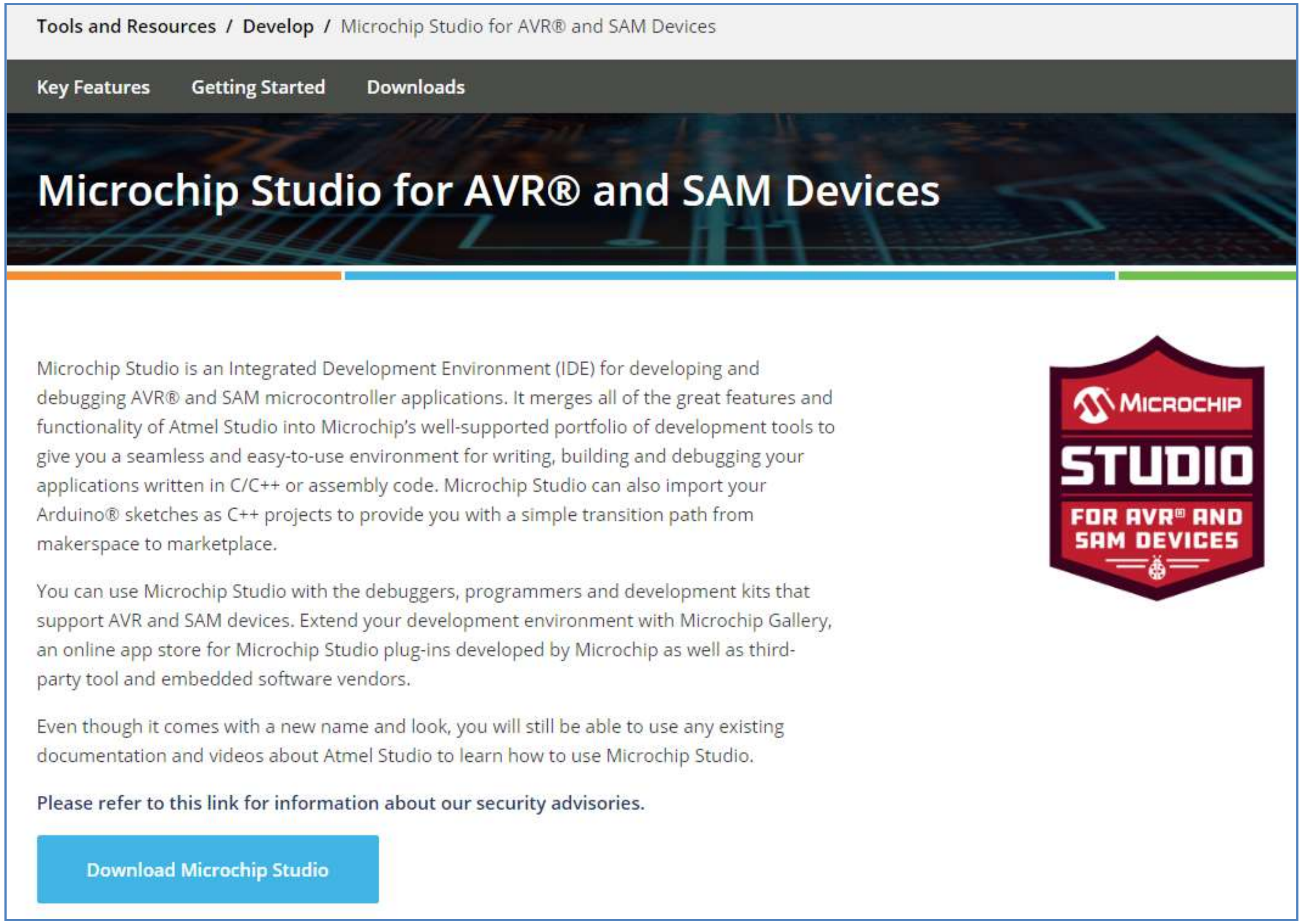
충북대학교 공동훈련센터

Microchip Studio for AVR



MICROCHIP
All
Enter keyword, item, model or part #
PRODUCTS
SOLUTIONS
TOOLS AND RESOURCES
SUPPORT
EDUCATION
AB
Reference Designs
Search and Discover
Configure
Develop
Debug
Evaluation Boards
Qualify
Production
Documentation
Archives
MPLAB® Mindi™ Analog Simulator
Analog and Interface Treelink Selection Tool
Libraries, Code Examples and More
Advanced Software Framework (ASF) for SAM Devices
Boundary Scan (BSDL) Files
CAD/CAE Symbols
Code Examples for PIC® MCUs
IBIS Models
MCU Software Libraries
Family
K2L Automotive Tool Spare Parts
Ensemble Graphics Toolkit
FPGA Design Tools
Microchip Studio for AVR® and SAM Devices
AVR® and SAM MCU Downloads Archive
Microchip Gallery

Microchip Studio



Tools and Resources / Develop / Microchip Studio for AVR® and SAM Devices

Microchip Studio for AVR® and SAM Devices

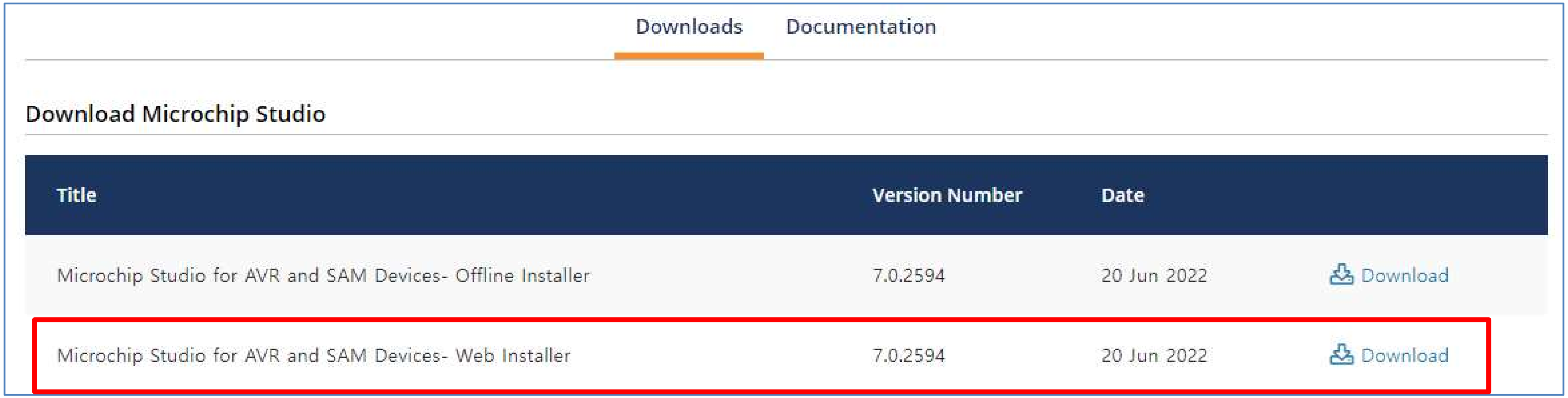
Microchip Studio is an Integrated Development Environment (IDE) for developing and debugging AVR® and SAM microcontroller applications. It merges all of the great features and functionality of Atmel Studio into Microchip's well-supported portfolio of development tools to give you a seamless and easy-to-use environment for writing, building and debugging your applications written in C/C++ or assembly code. Microchip Studio can also import your Arduino® sketches as C++ projects to provide you with a simple transition path from makerspace to marketplace.

You can use Microchip Studio with the debuggers, programmers and development kits that support AVR and SAM devices. Extend your development environment with Microchip Gallery, an online app store for Microchip Studio plug-ins developed by Microchip as well as third-party tool and embedded software vendors.

Even though it comes with a new name and look, you will still be able to use any existing documentation and videos about Atmel Studio to learn how to use Microchip Studio.

Please refer to this link for information about our security advisories.

Atmel Studio 7 Down Load

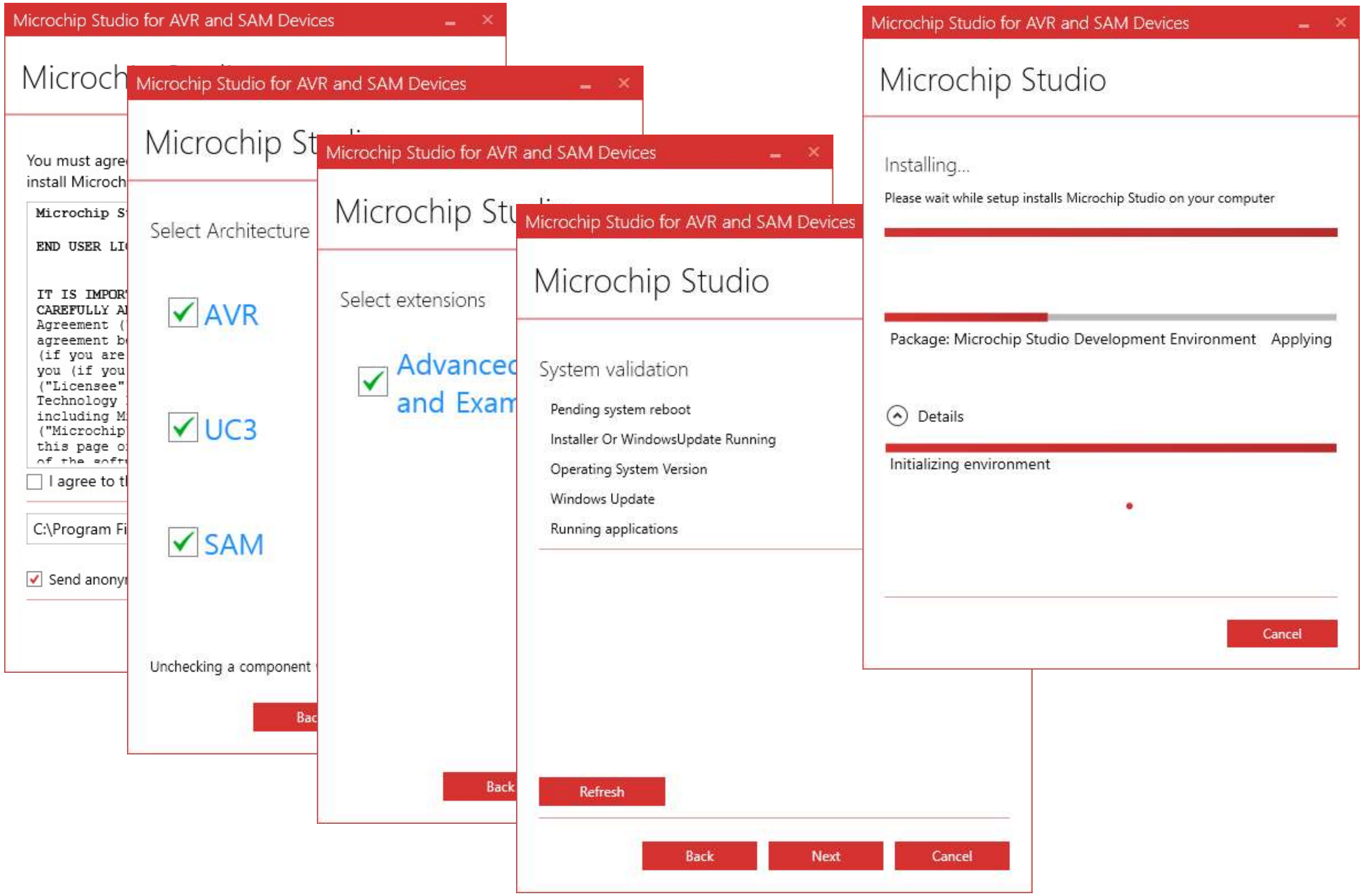


Downloads Documentation

Download Microchip Studio

Title	Version Number	Date	
Microchip Studio for AVR and SAM Devices- Offline Installer	7.0.2594	20 Jun 2022	Download
Microchip Studio for AVR and SAM Devices- Web Installer	7.0.2594	20 Jun 2022	Download

setup install



Microchip Studio for AVR and SAM Devices

Microch

You must agre
install Microch

```
Microchip S
END USER LI
IT IS IMPOR
CAREFULLY A
Agreement (
agreement b
(if you are
you (if you
("Licensee"
Technology
including M
("Microchip
this page o
of the soft
```

□ I agree to t

C:\Program Fi

☑ Send anony

Microchip Studio for AVR and SAM Devices

Microchip St

Select Architecture

☑ AVR

☑ UC3

☑ SAM

Unchecking a component

Microchip Studio for AVR and SAM Devices

Microchip Stu

Select extensions

☑ Advanced
and Exam

Microchip Studio for AVR and SAM Devices

Microchip Studio

System validation

Pending system reboot
Installer Or WindowsUpdate Running
Operating System Version
Windows Update
Running applications

Microchip Studio for AVR and SAM Devices

Microchip Studio

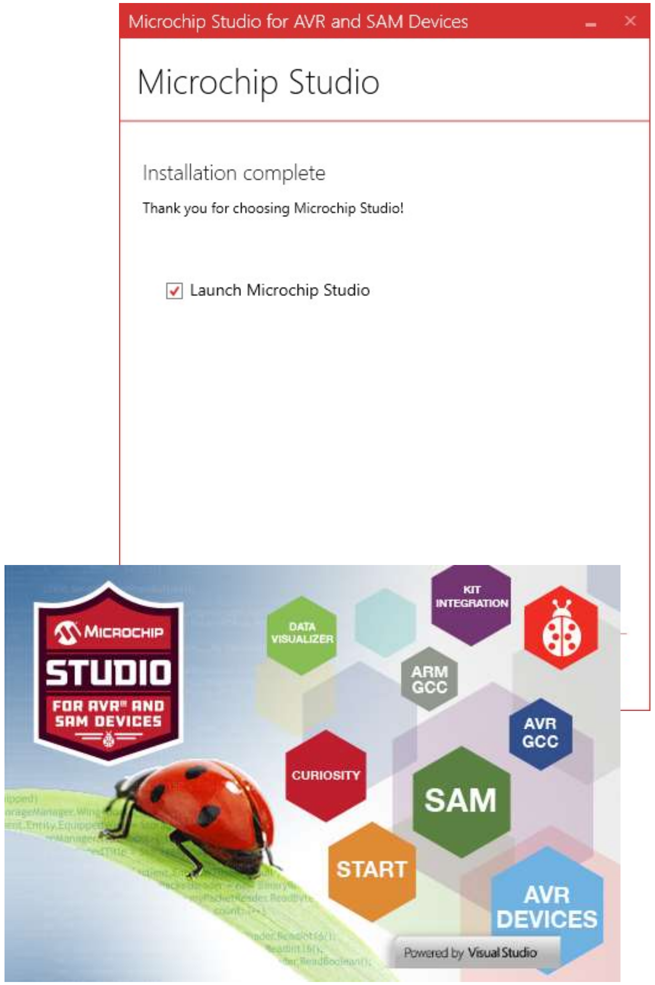
Installing...

Please wait while setup installs Microchip Studio on your computer

Package: Microchip Studio Development Environment Applying

Details

Initializing environment



Microchip Studio for AVR and SAM Devices

Microchip Studio

Installation complete

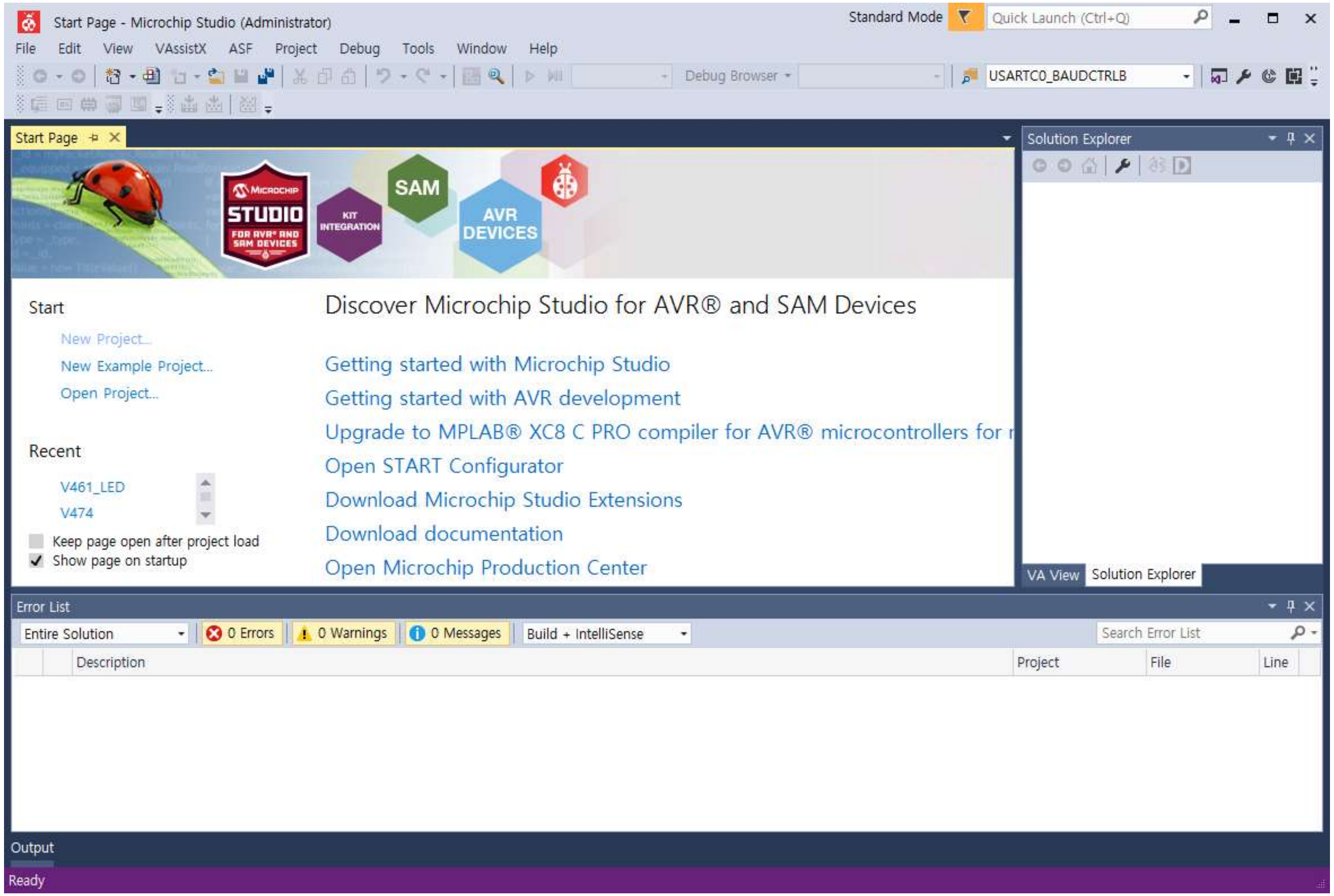
Thank you for choosing Microchip Studio!

☑ Launch Microchip Studio

System Requirements

- **Supported Operating Systems**
  - Windows 7 Service Pack 1 or higher
  - Windows Server 2008 R2 Service Pack 1 or higher
  - Windows 8/8.1
  - Windows Server 2012 and Windows Server 2012 R2
  - Windows 10
- **Supported Architectures**
  - 32-bit (x86)
  - 64-bit (x64)
- **Hardware Requirements**
  - A computer that has a 1.6 GHz or faster processor
  - RAM
    - 1 GB RAM for x86
    - 2 GB RAM for x64
    - An additional 512 MB RAM if running in a Virtual Machine
  - 6 GB available hard disk space

Start Page



Start Page - Microchip Studio (Administrator)
Standard Mode
Quick Launch (Ctrl+Q)
File Edit View VAssistX ASF Project Debug Tools Window Help
Debug Browser
USARTC0_BAUDCTRLB
Start Page
STUDIO FOR AVR® AND SAM DEVICES
KIT INTEGRATION
SAM
AVR DEVICES

Start

New Project...

New Example Project...

Open Project...

Recent

V461_LED

V474

□ Keep page open after project load

☑ Show page on startup

Discover Microchip Studio for AVR® and SAM Devices

Getting started with Microchip Studio

Getting started with AVR development

Upgrade to MPLAB® XC8 C PRO compiler for AVR® microcontrollers for r

Open START Configurator

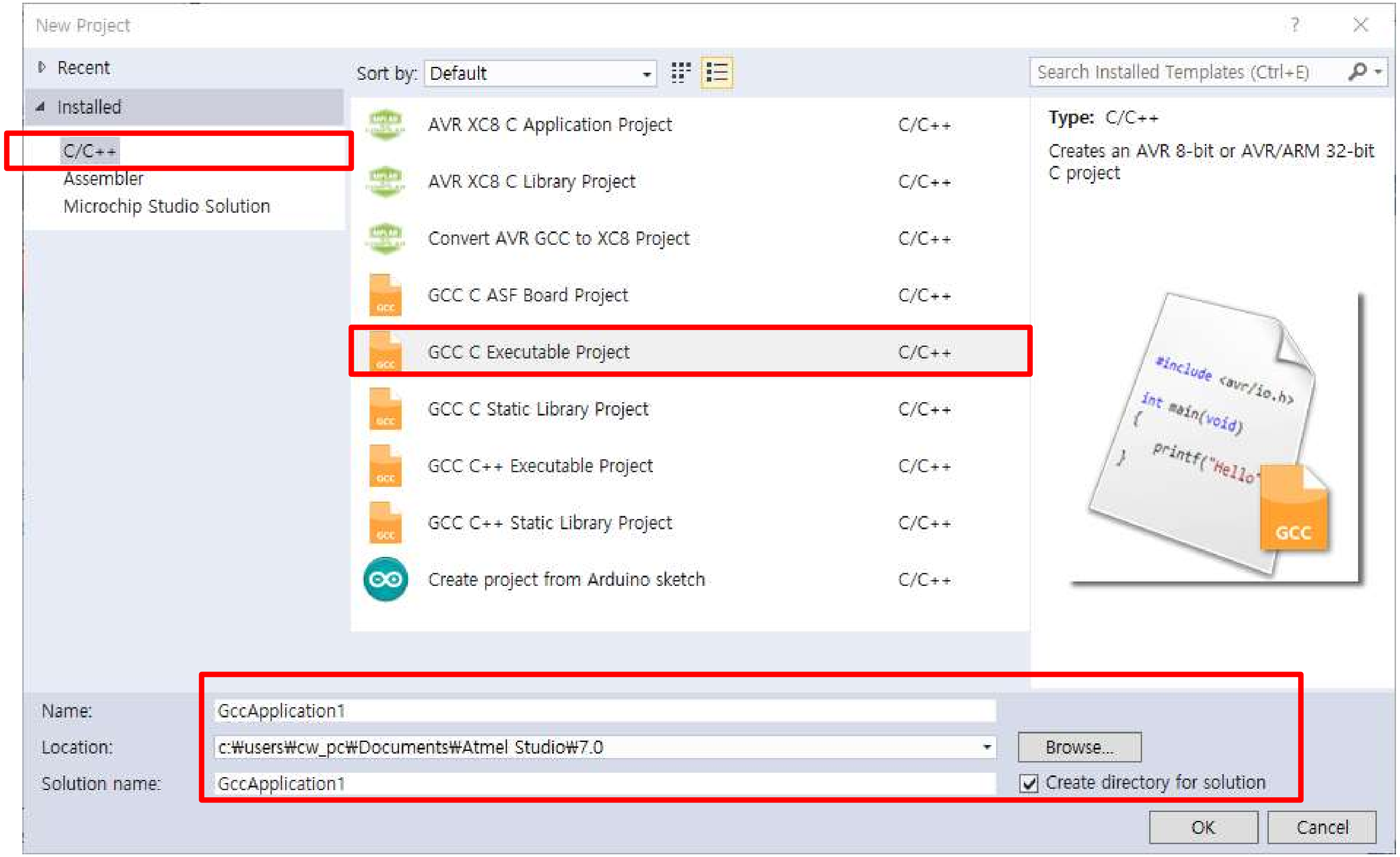
Download Microchip Studio Extensions

Download documentation

Open Microchip Production Center

충북대학교 공동훈련센터

New Project



New Project

?
Recent
Installed
C/C++
Assembler
Microchip Studio Solution
Sort by: Default
Search Installed Templates (Ctrl+E)

AVR XC8 C Application Project	C/C++
AVR XC8 C Library Project	C/C++
Convert AVR GCC to XC8 Project	C/C++
GCC C ASF Board Project	C/C++
GCC C Executable Project	C/C++
GCC C Static Library Project	C/C++
GCC C++ Executable Project	C/C++
GCC C++ Static Library Project	C/C++
Create project from Arduino sketch	C/C++

Type: C/C++

Creates an AVR 8-bit or AVR/ARM 32-bit C project

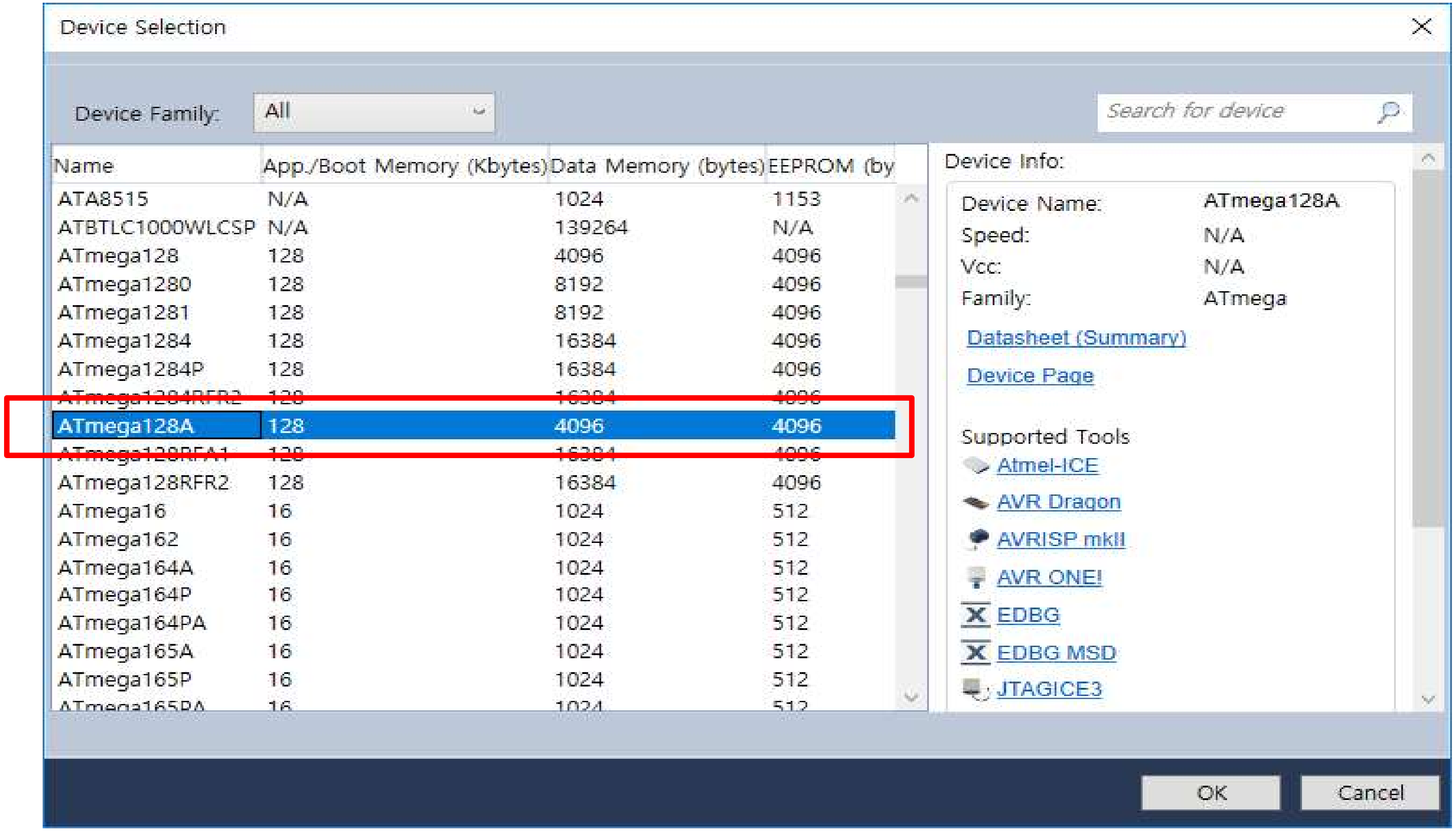
Name: GccApplication1

Location: c:₩users₩cw_pc₩Documents₩Atmel Studio₩7.0

Solution name: GccApplication1

☑ Create directory for solution

Device Selection



Device Selection

Device Family: All

Search for device

Name	App./Boot Memory (Kbytes)	Data Memory (bytes)	EEPROM (by
ATA8515	N/A	1024	1153
ATBTLC1000WLCSP	N/A	139264	N/A
ATmega128	128	4096	4096
ATmega1280	128	8192	4096
ATmega1281	128	8192	4096
ATmega1284	128	16384	4096
ATmega1284P	128	16384	4096
ATmega1284RFR2	128	16384	4096
ATmega128A	128	4096	4096
ATmega128RFA1	128	16384	4096
ATmega128RFR2	128	16384	4096
ATmega16	16	1024	512
ATmega162	16	1024	512
ATmega164A	16	1024	512
ATmega164P	16	1024	512
ATmega164PA	16	1024	512
ATmega165A	16	1024	512
ATmega165P	16	1024	512
ATmega165PA	16	1024	512

Device Info:

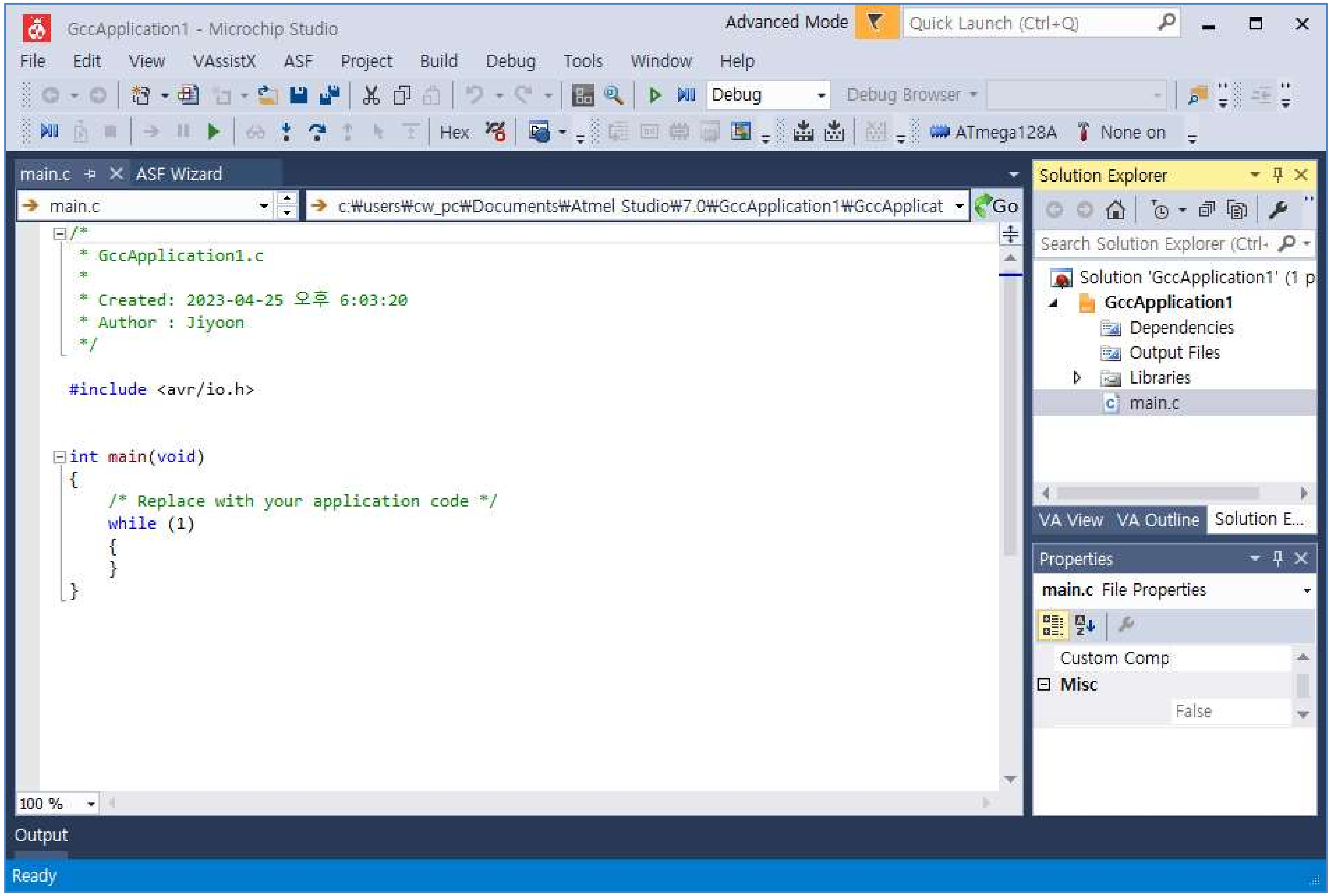
Device Name: ATmega128A
Speed: N/A
Vcc: N/A
Family: ATmega

Datasheet (Summary)

Device Page

Supported Tools
- Atmel-ICE
- AVR Dragon
- AVRISP mkII
- AVR ONE!
- EDBG
- EDBG MSD
- JTAGICE3

main.c



```
/*
 * GccApplication1.c
 *
 * Created: 2023-04-25 오후 6:03:20
 * Author : Jiyoon
 */

#include <avr/io.h>


int main(void)
{
    /* Replace with your application code */
    while (1)
    {
    }
}
```

IDE Window

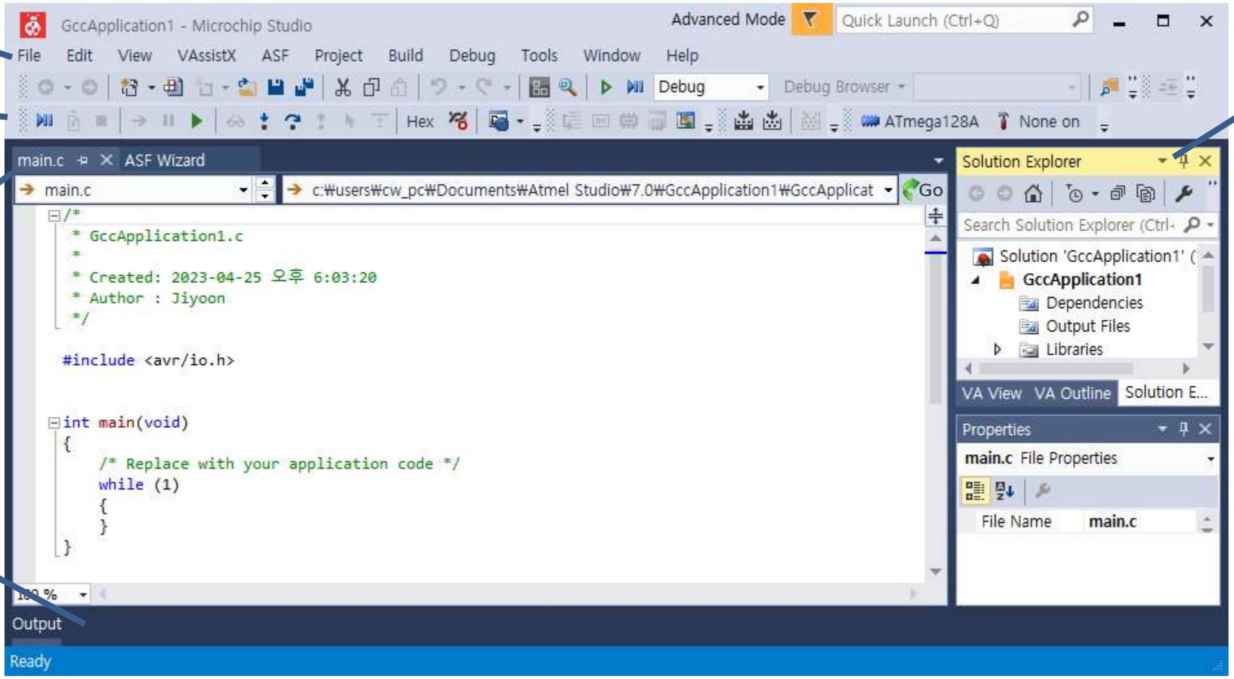
Menu

ToolBar

소스코드 탭

출력창

솔루션탐색기

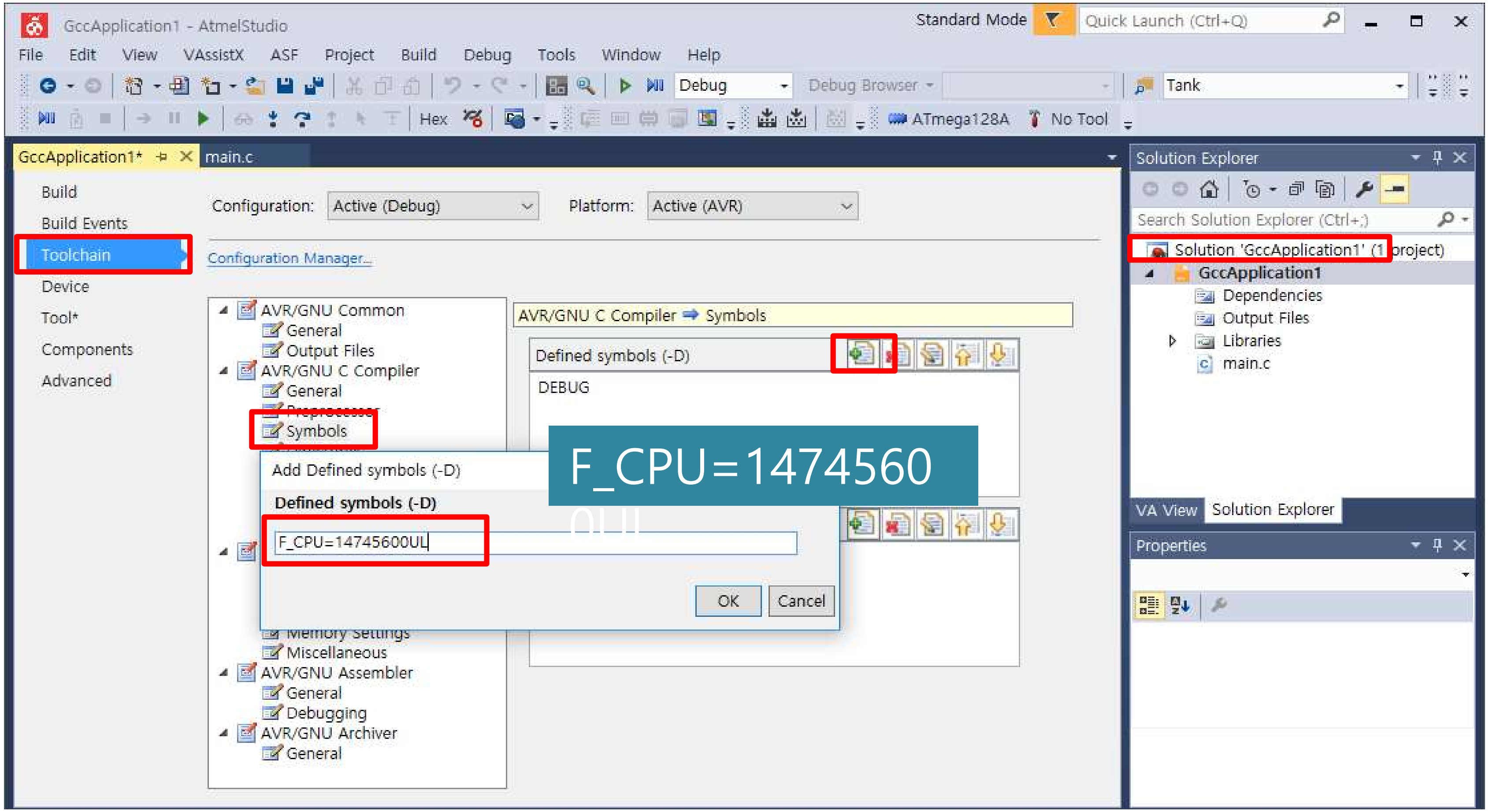


```
/*
 * GccApplication1.c
 *
 * Created: 2023-04-25 오후 6:03:20
 * Author : Jiyoon
 */

#include <avr/io.h>


int main(void)
{
    /* Replace with your application code */
    while (1)
    {
    }
}
```

Add Defined Symbols



GccApplication1 - AtmelStudio

Configuration: Active (Debug)

Platform: Active (AVR)

AVR/GNU C Compiler ➡ Symbols

Defined symbols (-D)

DEBUG

Add Defined symbols (-D)

Defined symbols (-D)

F_CPU=14745600UL

F_CPU=14745600UL

Program Coding

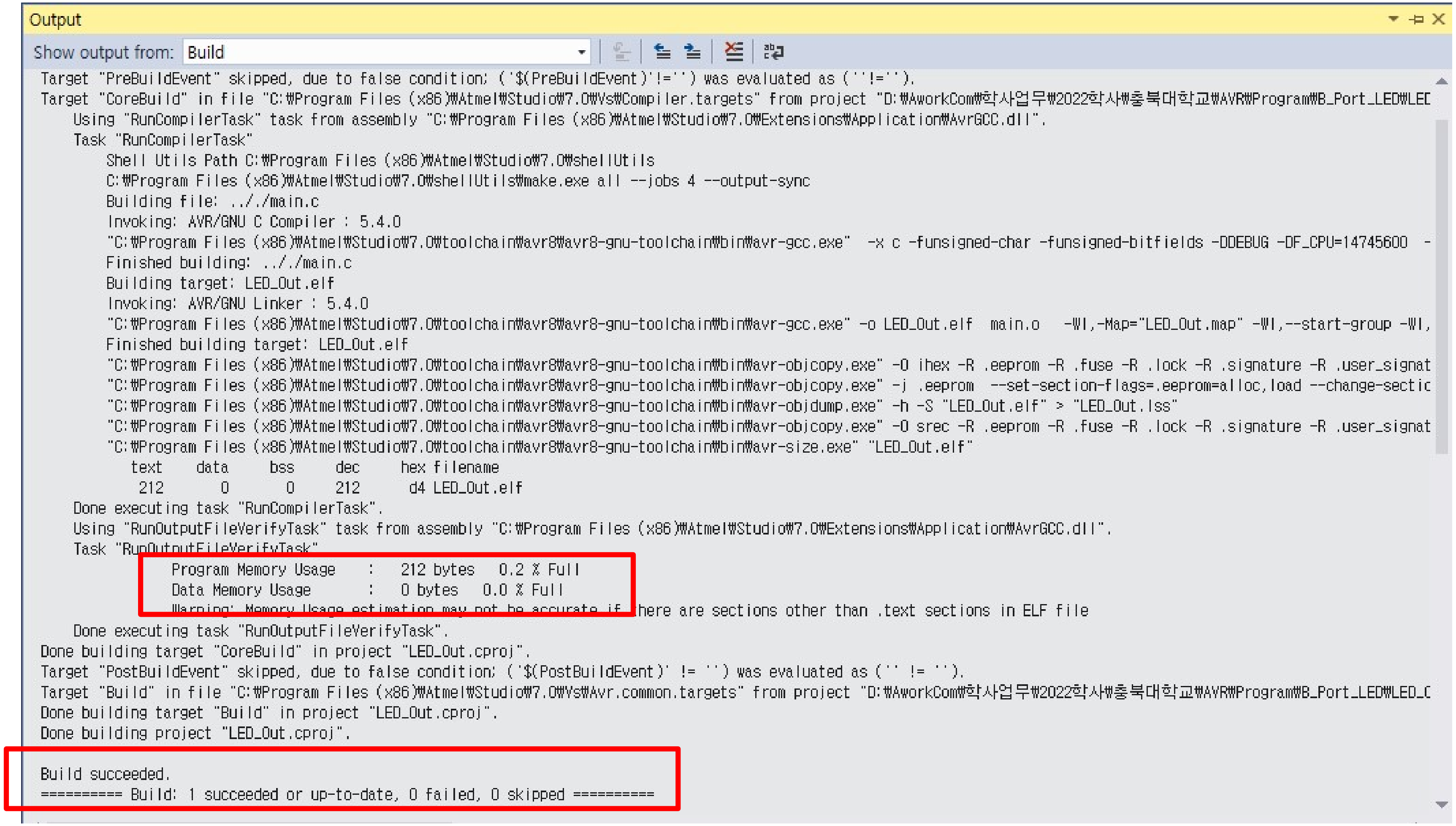
```
#include <avr/io.h>
#include <util/delay.h>

int main(void)
{
    DDRB=0xff;

    while (1)
    {
        PORTB=0x66;
        _delay_ms(500);

        PORTB=0x99;
        _delay_ms(500);
    }
}
```

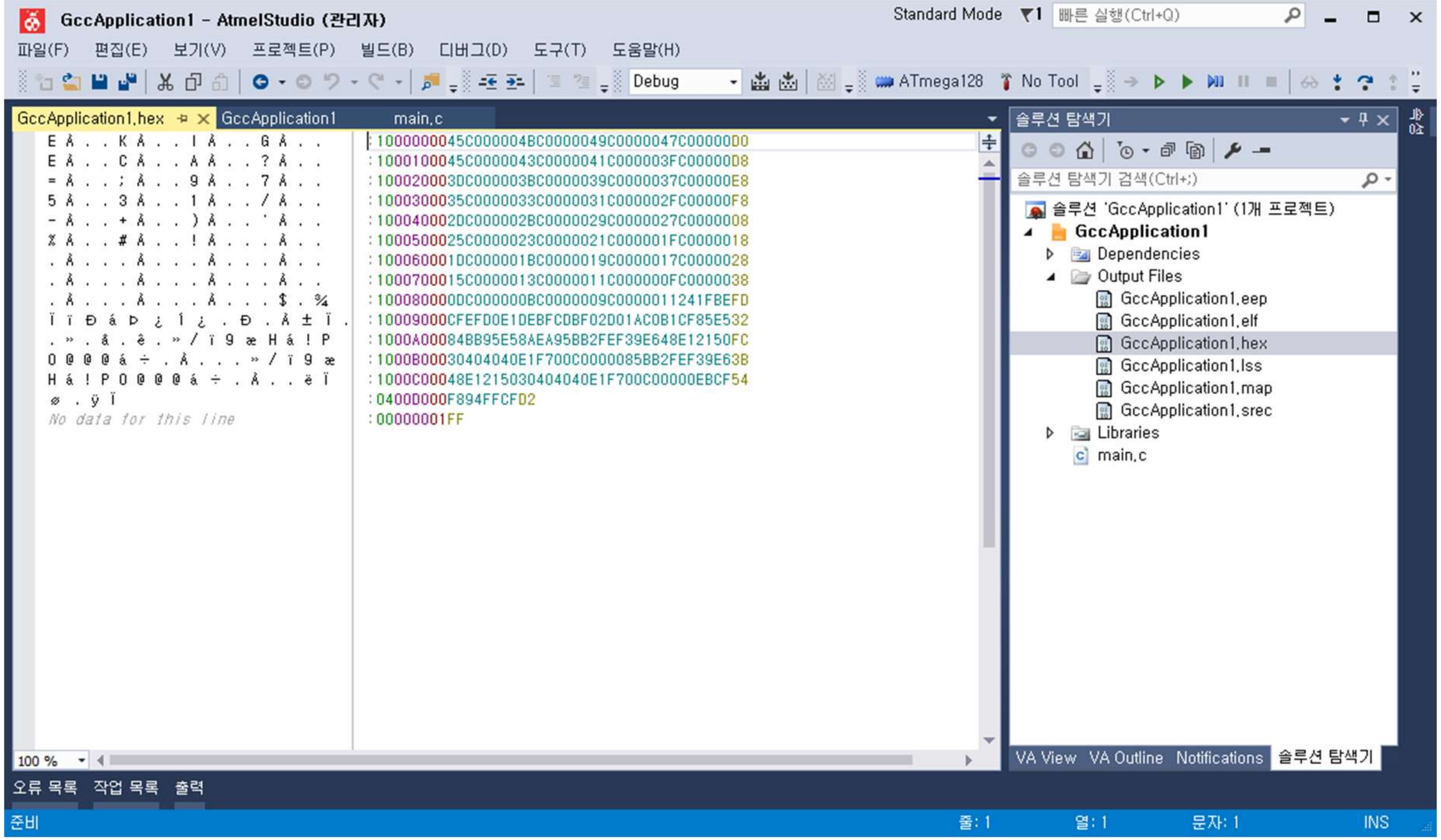
Build



```
Target "PreBuildEvent" skipped, due to false condition; ('$(PreBuildEvent)'!='') was evaluated as (''!='').
Target "CoreBuild" in file "C:₩Program Files (x86)₩Atmel₩Studio₩7.0₩Vs₩Compiler.targets" from project "D:₩AworkCom₩학사업무₩2022학사₩충북대학교₩AVR₩Program₩B_Port_LED₩LED
    Using "RunCompilerTask" task from assembly "C:₩Program Files (x86)₩Atmel₩Studio₩7.0₩Extensions₩Application₩AvrGCC.dll".
    Task "RunCompilerTask"
        Shell Utils Path C:₩Program Files (x86)₩Atmel₩Studio₩7.0₩shellUtils
        C:₩Program Files (x86)₩Atmel₩Studio₩7.0₩shellUtils₩make.exe all --jobs 4 --output-sync
        Building file: .././main.c
        Invoking: AVR/GNU C Compiler : 5.4.0
        "C:₩Program Files (x86)₩Atmel₩Studio₩7.0₩toolchain₩avr8₩avr8-gnu-toolchain₩bin₩avr-gcc.exe"  -x c -funsigned-char -funsigned-bitfields -DDEBUG -DF_CPU=14745600  -
        Finished building: .././main.c
        Building target: LED_Out.elf
        Invoking: AVR/GNU Linker : 5.4.0
        "C:₩Program Files (x86)₩Atmel₩Studio₩7.0₩toolchain₩avr8₩avr8-gnu-toolchain₩bin₩avr-gcc.exe" -o LED_Out.elf  main.o   -Wl,-Map="LED_Out.map" -Wl,--start-group -Wl,
        Finished building target: LED_Out.elf
        "C:₩Program Files (x86)₩Atmel₩Studio₩7.0₩toolchain₩avr8₩avr8-gnu-toolchain₩bin₩avr-objcopy.exe" -O ihex -R .eeprom -R .fuse -R .lock -R .signature -R .user_signat
        "C:₩Program Files (x86)₩Atmel₩Studio₩7.0₩toolchain₩avr8₩avr8-gnu-toolchain₩bin₩avr-objcopy.exe" -j .eeprom  --set-section-flags=.eeprom=alloc,load --change-sectic
        "C:₩Program Files (x86)₩Atmel₩Studio₩7.0₩toolchain₩avr8₩avr8-gnu-toolchain₩bin₩avr-objdump.exe" -h -S "LED_Out.elf" > "LED_Out.lss"
        "C:₩Program Files (x86)₩Atmel₩Studio₩7.0₩toolchain₩avr8₩avr8-gnu-toolchain₩bin₩avr-objcopy.exe" -O srec -R .eeprom -R .fuse -R .lock -R .signature -R .user_signat
        "C:₩Program Files (x86)₩Atmel₩Studio₩7.0₩toolchain₩avr8₩avr8-gnu-toolchain₩bin₩avr-size.exe" "LED_Out.elf"
           text    data     bss     dec     hex filename
            212       0       0     212      d4 LED_Out.elf
    Done executing task "RunCompilerTask".
    Using "RunOutputFileVerifyTask" task from assembly "C:₩Program Files (x86)₩Atmel₩Studio₩7.0₩Extensions₩Application₩AvrGCC.dll".
    Task "RunOutputFileVerifyTask"
                Program Memory Usage    :   212 bytes   0.2 % Full
                Data Memory Usage       :   0 bytes   0.0 % Full
                Warning: Memory Usage estimation may not be accurate if there are sections other than .text sections in ELF file
    Done executing task "RunOutputFileVerifyTask".
Done building target "CoreBuild" in project "LED_Out.cproj".
Target "PostBuildEvent" skipped, due to false condition; ('$(PostBuildEvent)' != '') was evaluated as ('' != '').
Target "Build" in file "C:₩Program Files (x86)₩Atmel₩Studio₩7.0₩Vs₩Avr.common.targets" from project "D:₩AworkCom₩학사업무₩2022학사₩충북대학교₩AVR₩Program₩B_Port_LED₩LED_C
Done building target "Build" in project "LED_Out.cproj".
Done building project "LED_Out.cproj".

Build succeeded.
========== Build: 1 succeeded or up-to-date, 0 failed, 0 skipped ==========
```

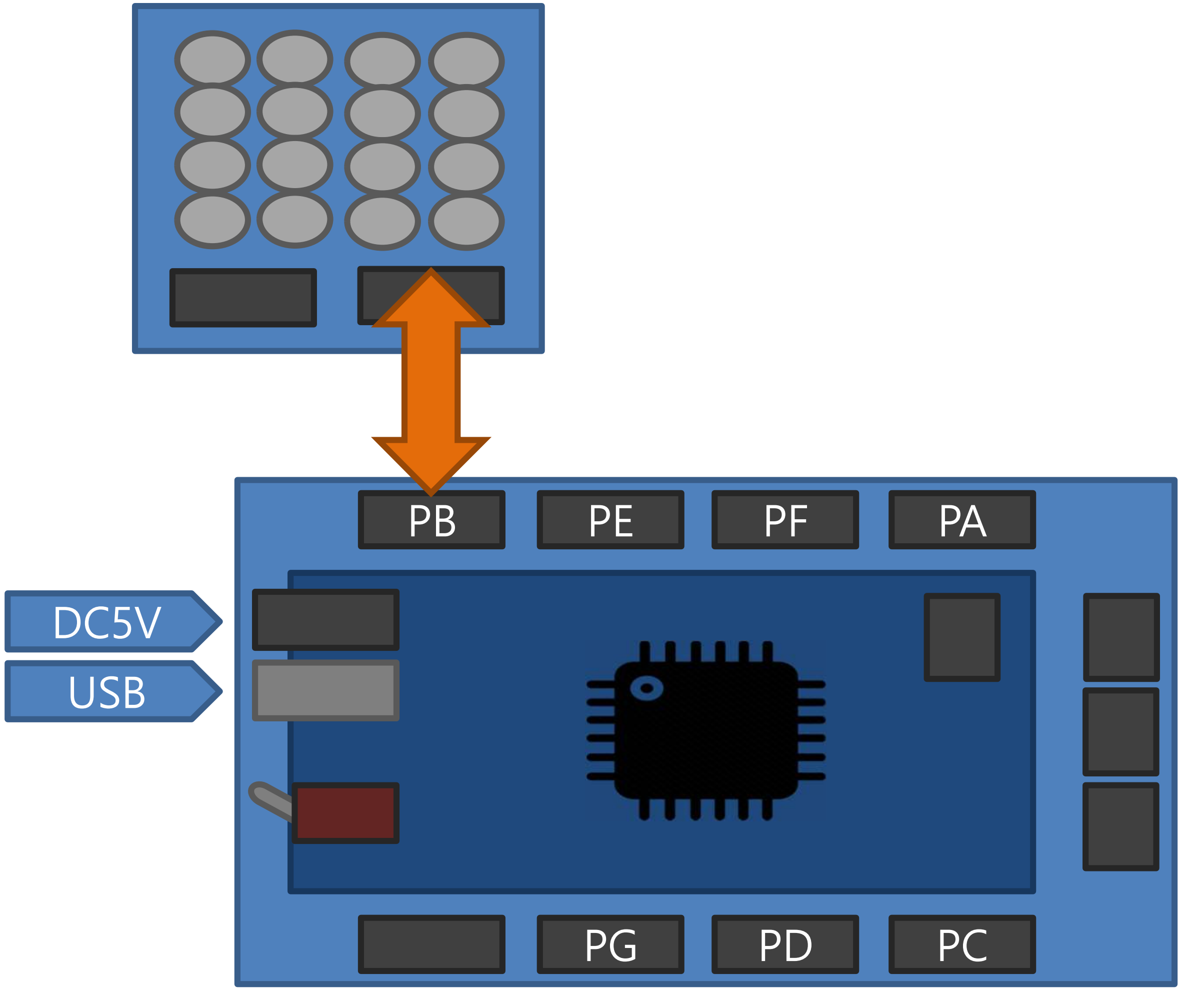
Output Files



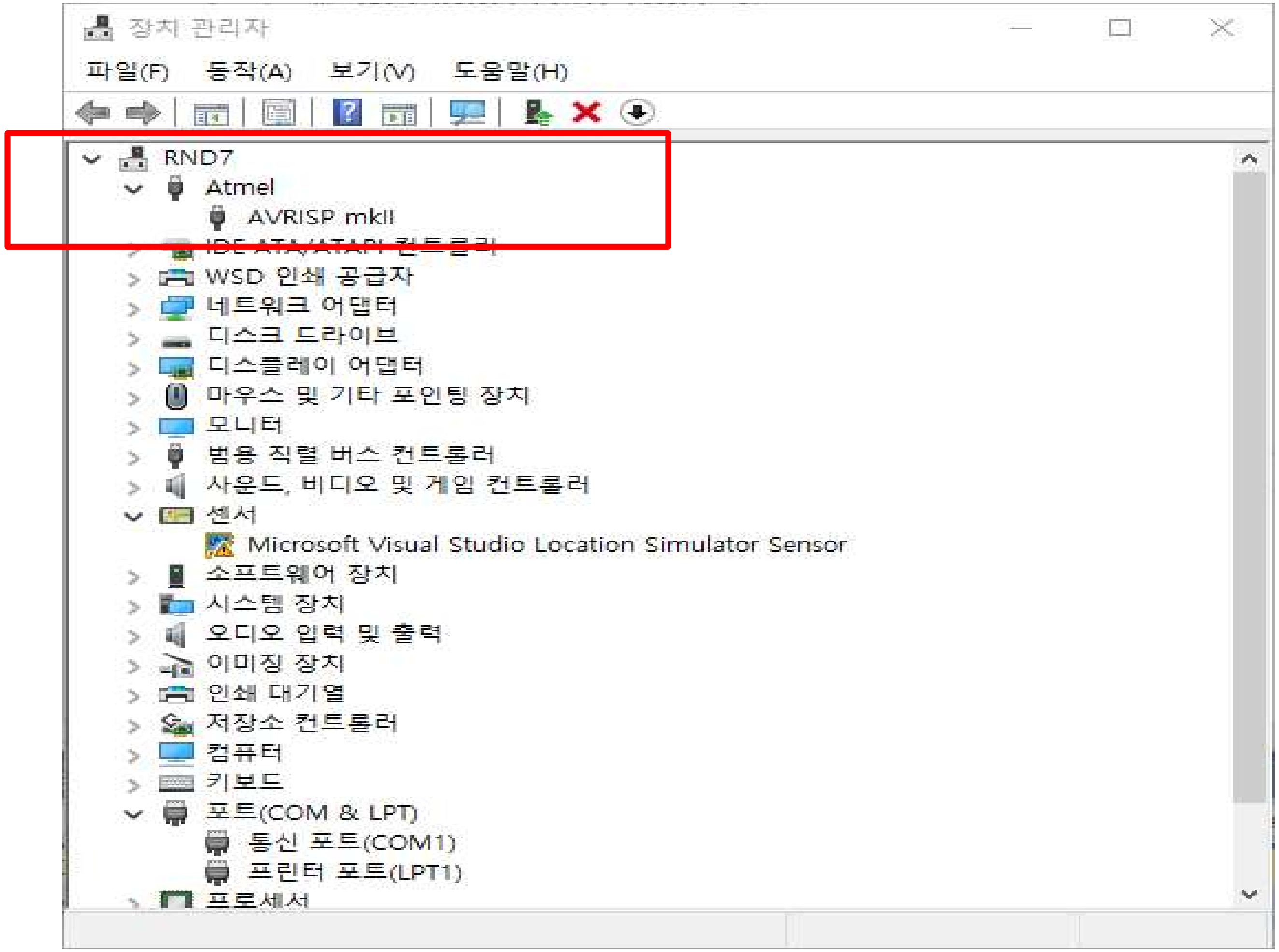
```
E À . . K À . . I À . . G À . .        :1000000045C000004BC0000049C0000047C00000D0
E À . . C À . . A À . . ? À . .        :1000100045C0000043C0000041C000003FC00000D8
= À . . ; À . . 9 À . . 7 À . .        :100020003DC000003BC0000039C0000037C00000E8
5 À . . 3 À . . 1 À . . / À . .        :1000300035C0000033C0000031C000002FC00000F8
- À . . + À . . ) À . . ' À . .        :100040002DC000002BC0000029C0000027C0000008
% À . . # À . . ! À . . . À . .        :1000500025C0000023C0000021C000001FC0000018
. À . . . À . . . À . . . À . .        :100060001DC000001BC0000019C0000017C0000028
. À . . . À . . . À . . . À . .        :1000700015C0000013C0000011C000000FC0000038
. À . . . À . . . À . . . $ . ¾        :100080000DC000000BC0000009C0000011241FBEFD
Ï ï Ð á Þ ¿ Í ¿ . Ð . À ± Ï .          :10009000CFEFD0E1DEBFCDBF02D01AC0B1CF85E532
. » . å . ê . » / ï 9 æ H á ! P        :1000A00084BB95E58AEA95BB2FEF39E648E12150FC
0 @ @ @ á ÷ . À . . . » / ï 9 æ        :1000B00030404040E1F700C0000085BB2FEF39E63B
H á ! P 0 @ @ @ á ÷ . À . . ë Ï        :1000C00048E1215030404040E1F700C00000EBCF54
ø . ÿ Ï                                :04000D000F894FFCFD2
No data for this line                  :00000001FF
```

충북대학교 공동훈련센터

Wiring



장치관리자



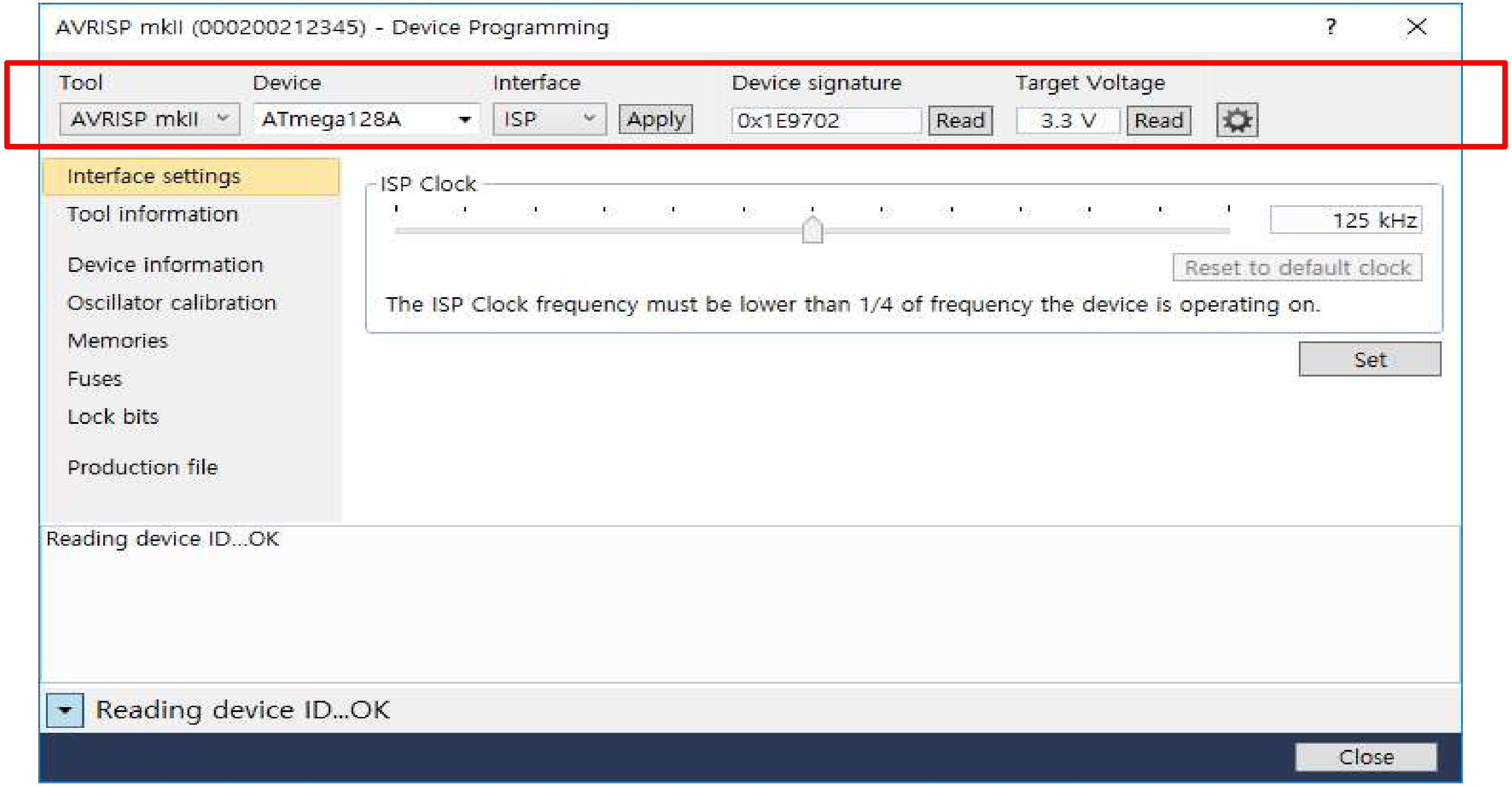
장치 관리자

파일(F) 동작(A) 보기(V) 도움말(H)

- RND7
  - Atmel
    - AVRISP mkII
  - IDE ATA/ATAPI 컨트롤러
  - WSD 인쇄 공급자
  - 네트워크 어댑터
  - 디스크 드라이브
  - 디스플레이 어댑터
  - 마우스 및 기타 포인팅 장치
  - 모니터
  - 범용 직렬 버스 컨트롤러
  - 사운드, 비디오 및 게임 컨트롤러
  - 센서
    - Microsoft Visual Studio Location Simulator Sensor
  - 소프트웨어 장치
  - 시스템 장치
  - 오디오 입력 및 출력
  - 이미징 장치
  - 인쇄 대기열
  - 저장소 컨트롤러
  - 컴퓨터
  - 키보드
  - 포트(COM & LPT)
    - 통신 포트(COM1)
    - 프린터 포트(LPT1)
  - 프로세서

Device programming

- [Tool]-[Device programming]



AVRISP mkII (000200212345) - Device Programming

Tool: AVRISP mkII

Device: ATmega128A

Interface: ISP

Device signature: 0x1E9702

Target Voltage: 3.3 V

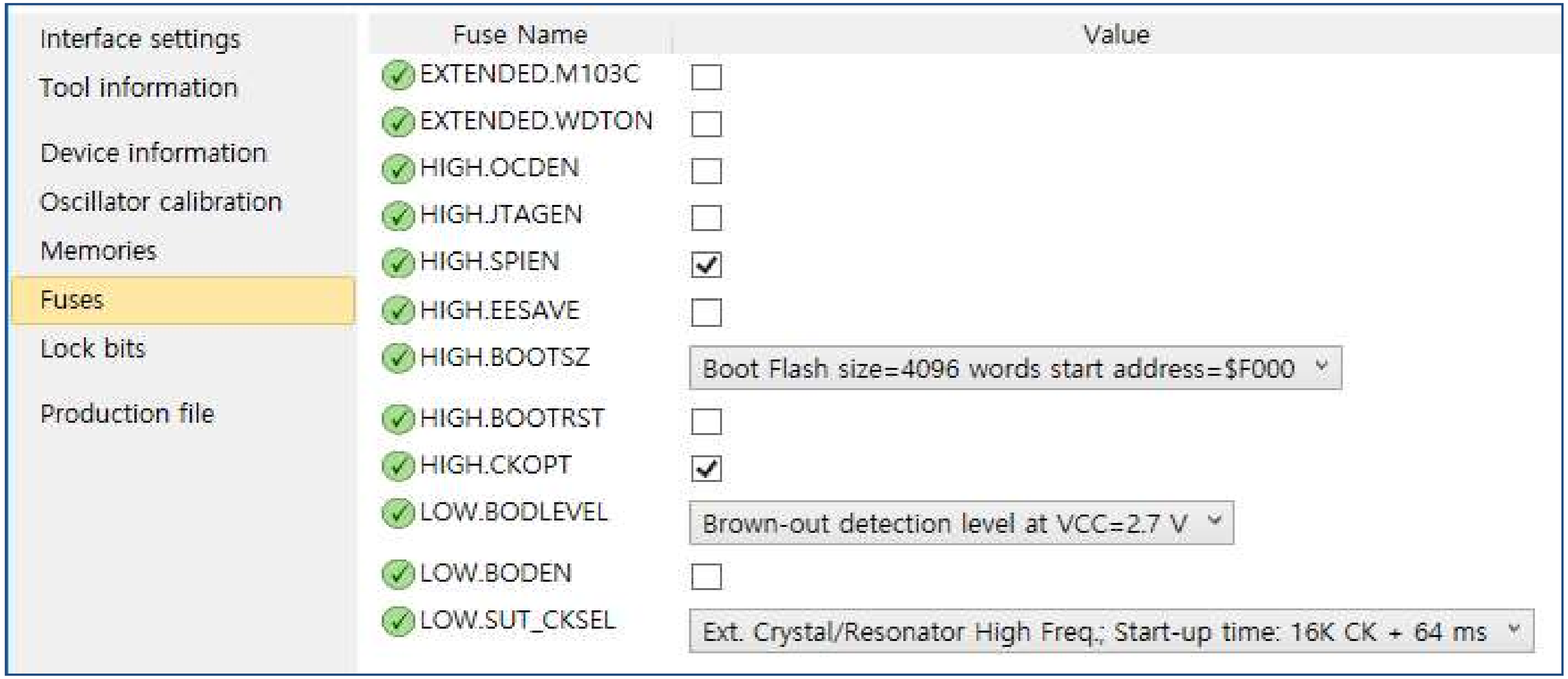
125 kHz

The ISP Clock frequency must be lower than 1/4 of frequency the device is operating on.

Reading device ID...OK

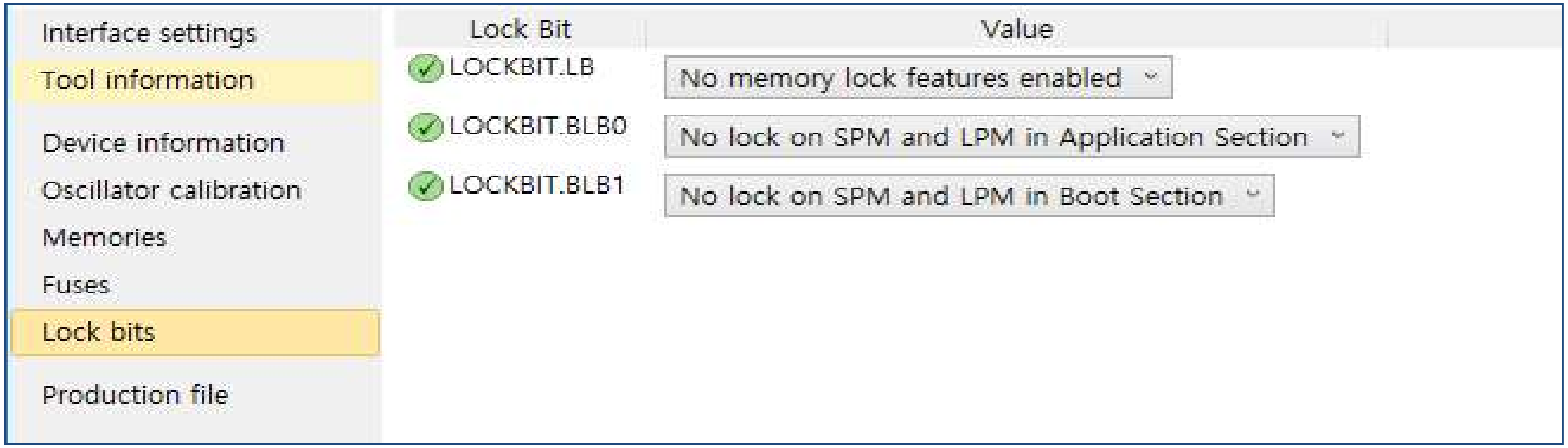
Reading device ID...OK

Fuses



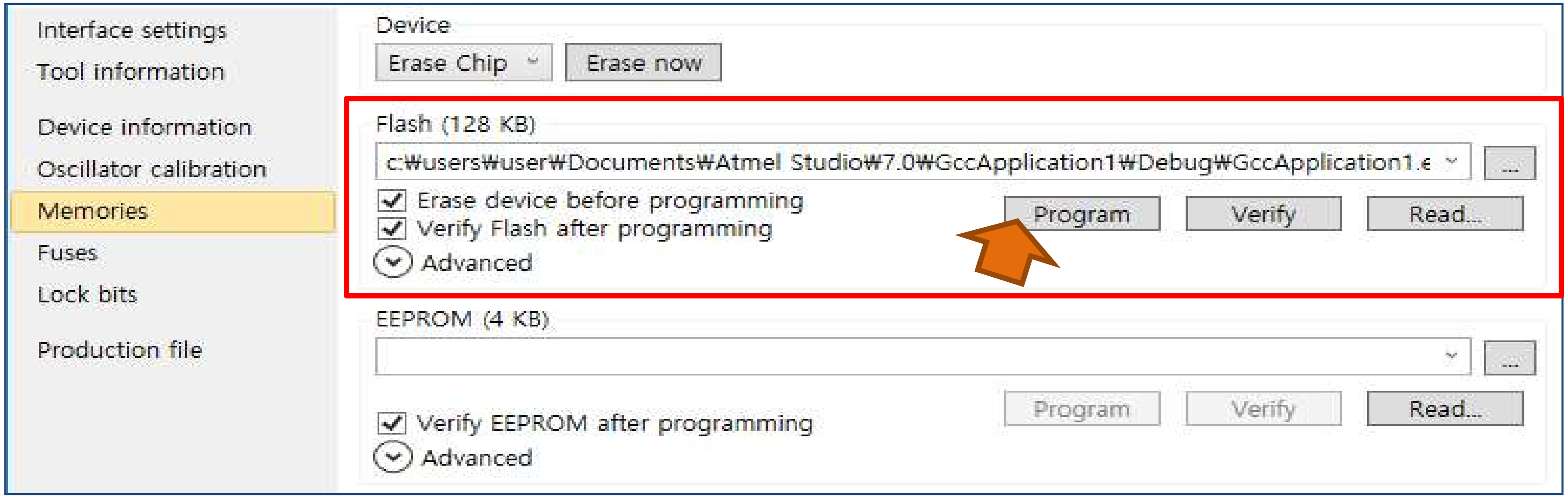
Fuse Name	Value
EXTENDED.M103C	□
EXTENDED.WDTON	□
HIGH.OCDEN	□
HIGH.JTAGEN	□
HIGH.SPIEN	☑
HIGH.EESAVE	□
HIGH.BOOTSZ	Boot Flash size=4096 words start address=$F000
HIGH.BOOTRST	□
HIGH.CKOPT	☑
LOW.BODLEVEL	Brown-out detection level at VCC=2.7 V
LOW.BODEN	□
LOW.SUT_CKSEL	Ext. Crystal/Resonator High Freq.; Start-up time: 16K CK + 64 ms

Lock bits



Lock Bit	Value
✓ LOCKBIT.LB	No memory lock features enabled
✓ LOCKBIT.BLB0	No lock on SPM and LPM in Application Section
✓ LOCKBIT.BLB1	No lock on SPM and LPM in Boot Section

Memories



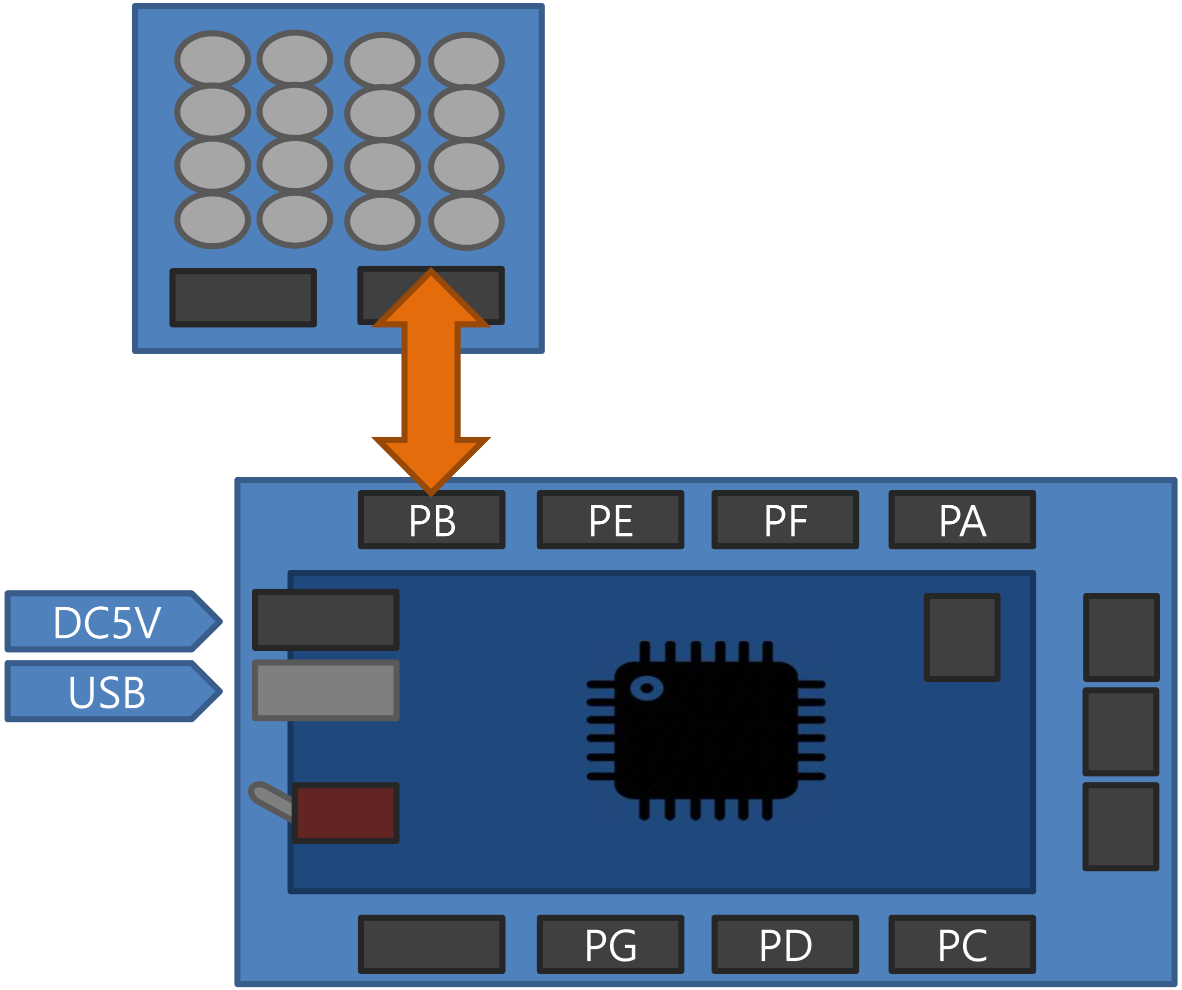
Interface settings
Tool information
Device information
Oscillator calibration
Memories
Fuses
Lock bits
Production file

Device
Erase Chip
Erase now

Flash (128 KB)
c:₩users₩user₩Documents₩Atmel Studio₩7.0₩GccApplication1₩Debug₩GccApplication1.e
☑ Erase device before programming
☑ Verify Flash after programming
Advanced
Program Verify Read...

EEPROM (4 KB)
☑ Verify EEPROM after programming
Advanced
Program Verify Read...

Run


PB
PE
PF
PA
DC5V
USB
PG
PD
PC