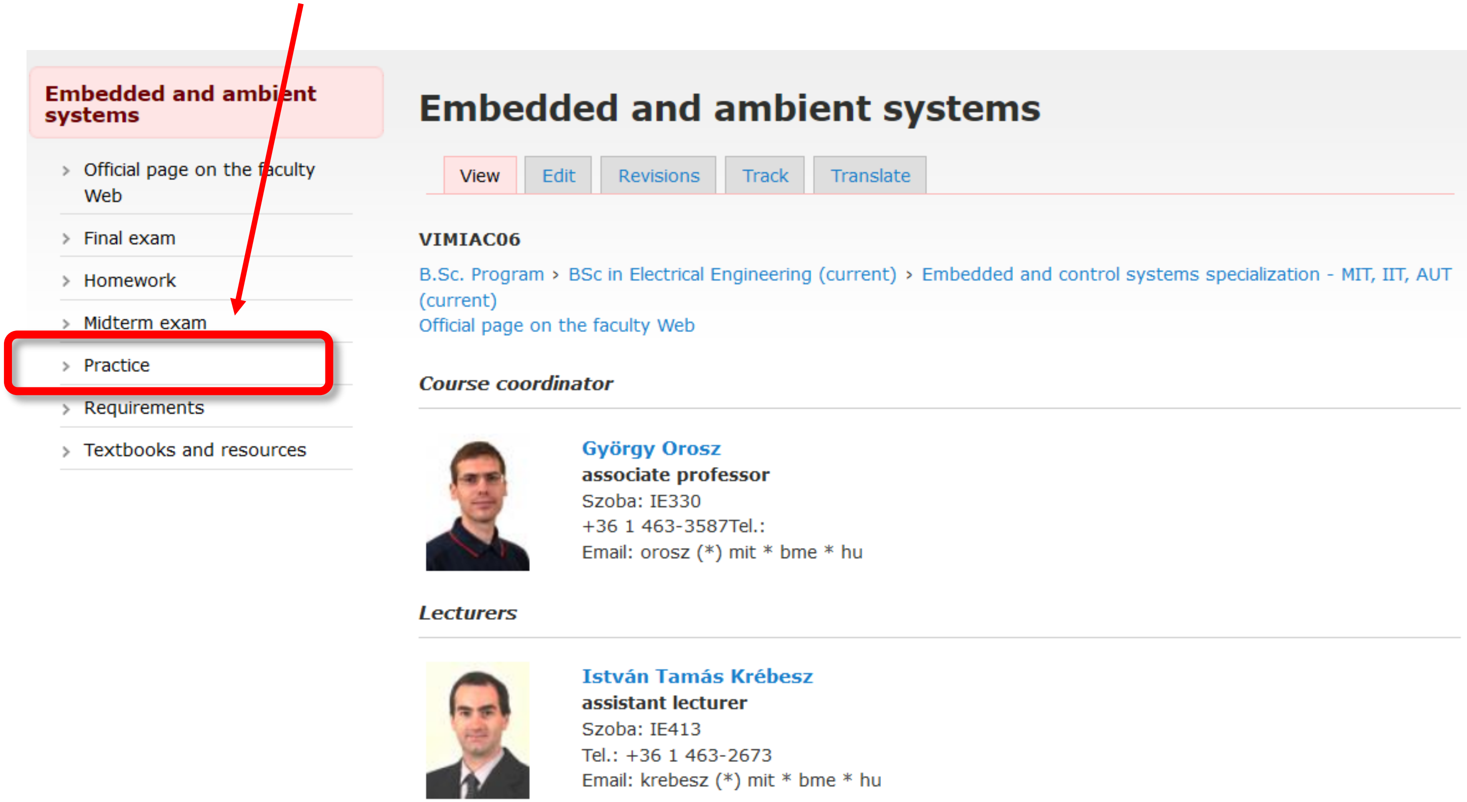
Embedded and ambient systems 2020.09.09.

Practice 1

Méréstechnika és Információs Rendszerek Tanszék

Preliminary

- Check the web site of the course: www.mit.bme.hu/eng/oktatas/targyak/VIMIAC06
- See menu on the left



Embedded and ambient systems

VIMIAC06

B.Sc. Program › BSc in Electrical Engineering (current) › Embedded and control systems specialization - MIT, IIT, AUT (current)

Official page on the faculty Web

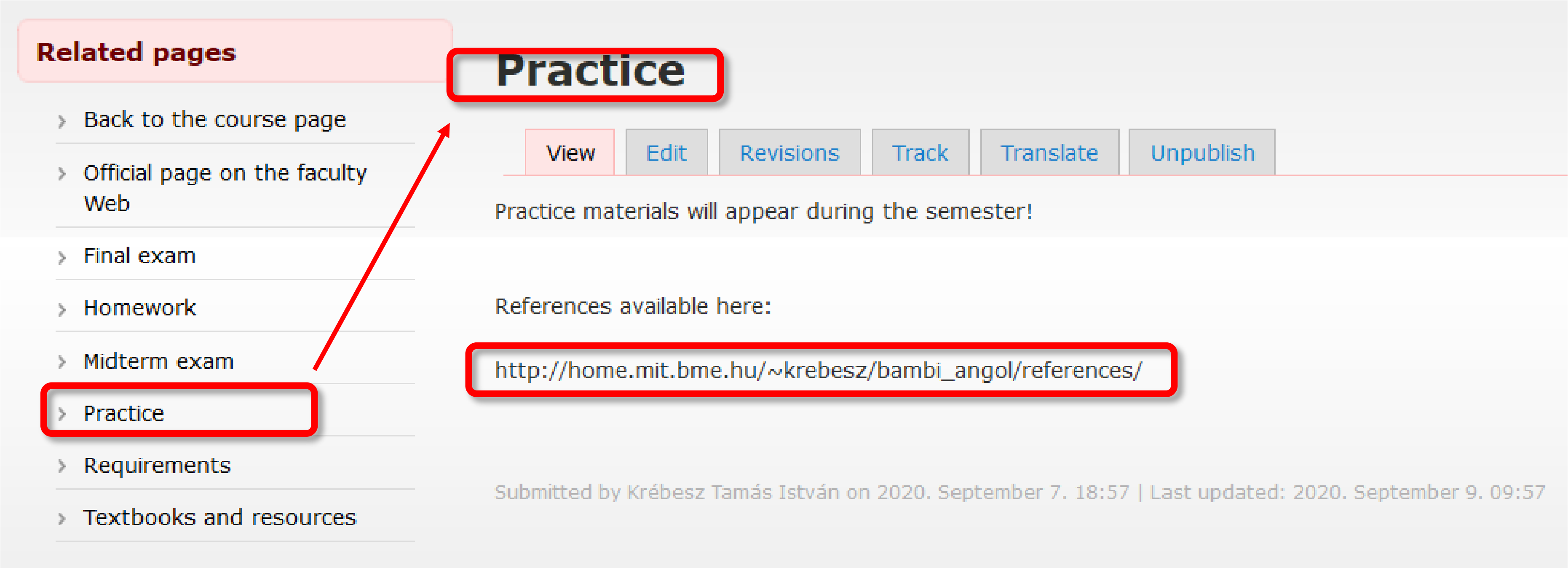
Course coordinator

György Orosz
associate professor
Szoba: IE330
+36 1 463-3587Tel.:
Email: orosz (*) mit * bme * hu

Lecturers

István Tamás Krébesz
assistant lecturer
Szoba: IE413
Tel.: +36 1 463-2673
Email: krebesz (*) mit * bme * hu

Preliminary



Related pages

- Back to the course page
- Official page on the faculty Web
- Final exam
- Homework
- Midterm exam
- Practice
- Requirements
- Textbooks and resources

Practice

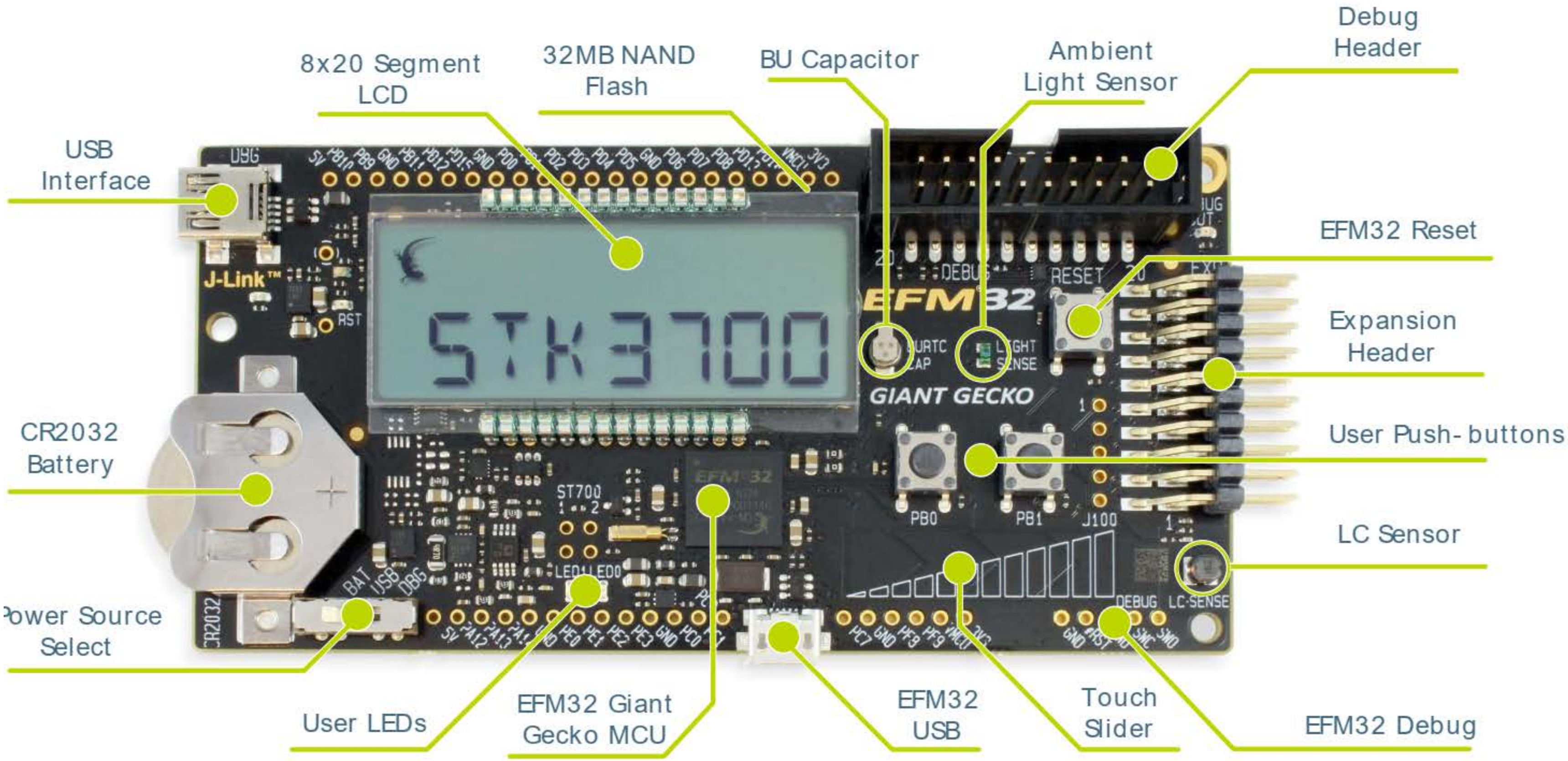
Practice materials will appear during the semester!

References available here:

http://home.mit.bme.hu/~krebesz/bambi_angol/references/

Submitted by Krébesz Tamás István on 2020. September 7. 18:57 | Last updated: 2020. September 9. 09:57

1) Development board: EFM32GG-STK3700

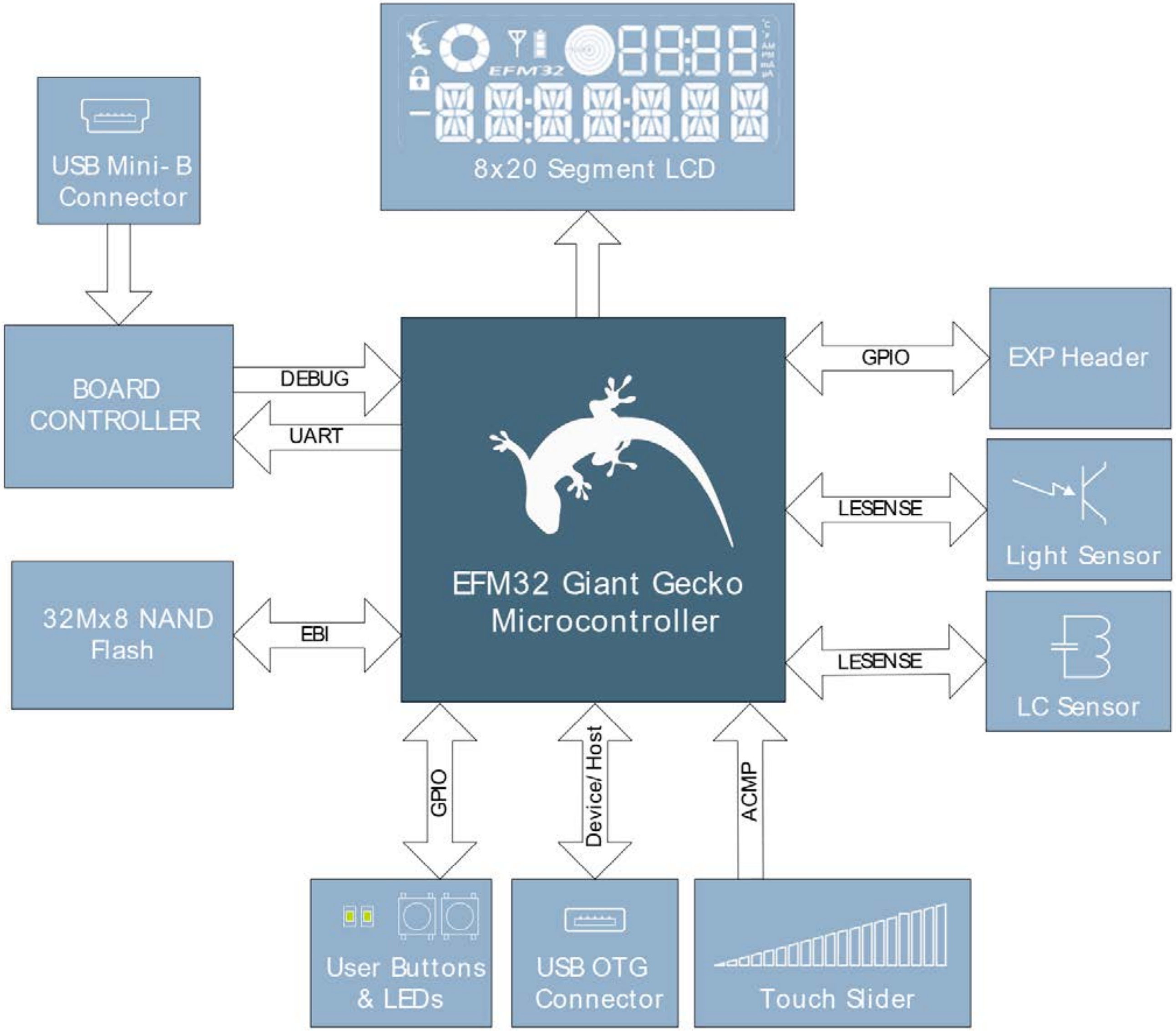


- https://www.silabs.com/development-tools/mcu/32-bit/efm32gg-starter-kit

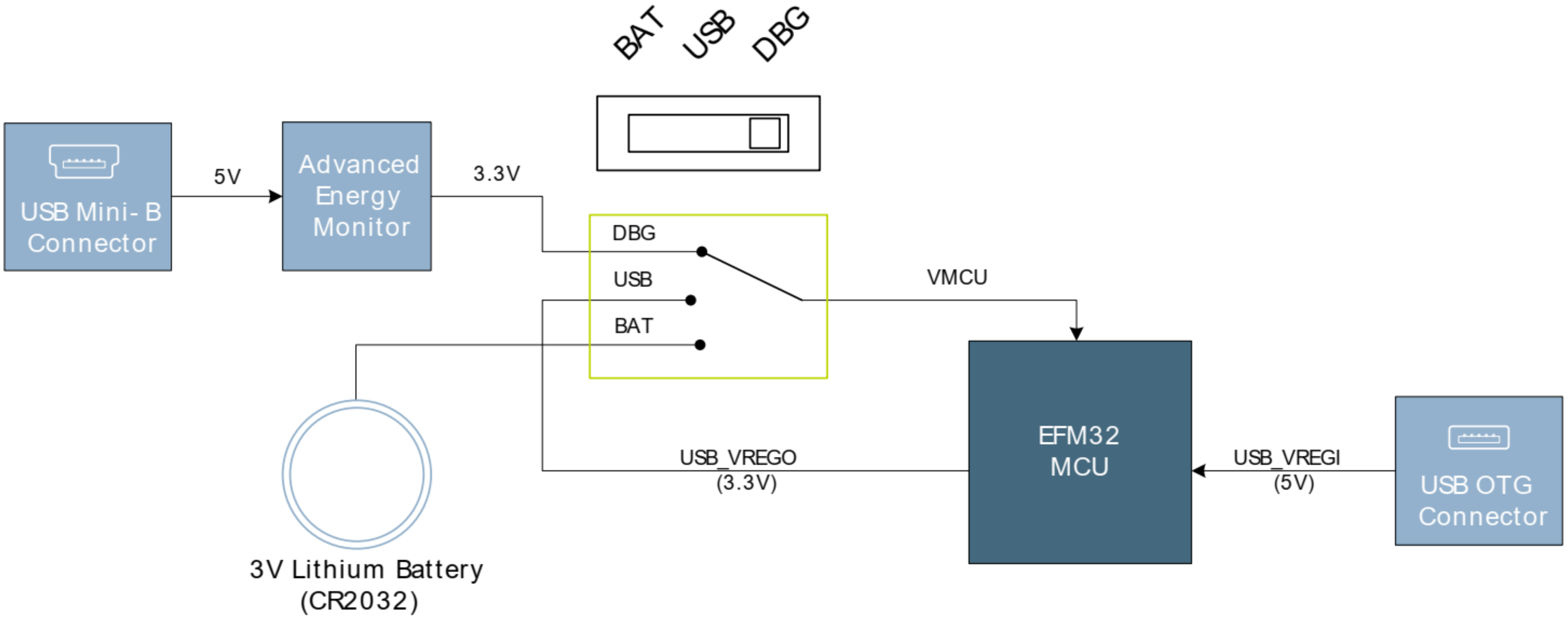
1.1) Main features

- EFM32GG990F1024 MCU with 1 MB Flash and 128 KB RAM.
- Advanced Energy Monitoring system for precise current tracking.
- Integrated Segger J-Link USB debugger/emulator with debug out functionality.
- 160 segment Energy Micro LCD.
- 20 pin expansion header.
- Breakout pads for easy access to I/O pins.
- Power sources include USB and CR2032 battery.
- 2 user buttons, 2 user LEDs and a touch slider.
- Ambient Light Sensor and Inductive-capacitive metal sensor.
- EFM32 OPAMP footprint.
- 32 MB NAND Flash.
- USB Micro-AB (OTG) connector.
- 0.03F Super Capacitor for backup power domain.
- Crystals for LFXO and HFXO: 32.768kHz and 48.000MHz.

1.2) Block diagram

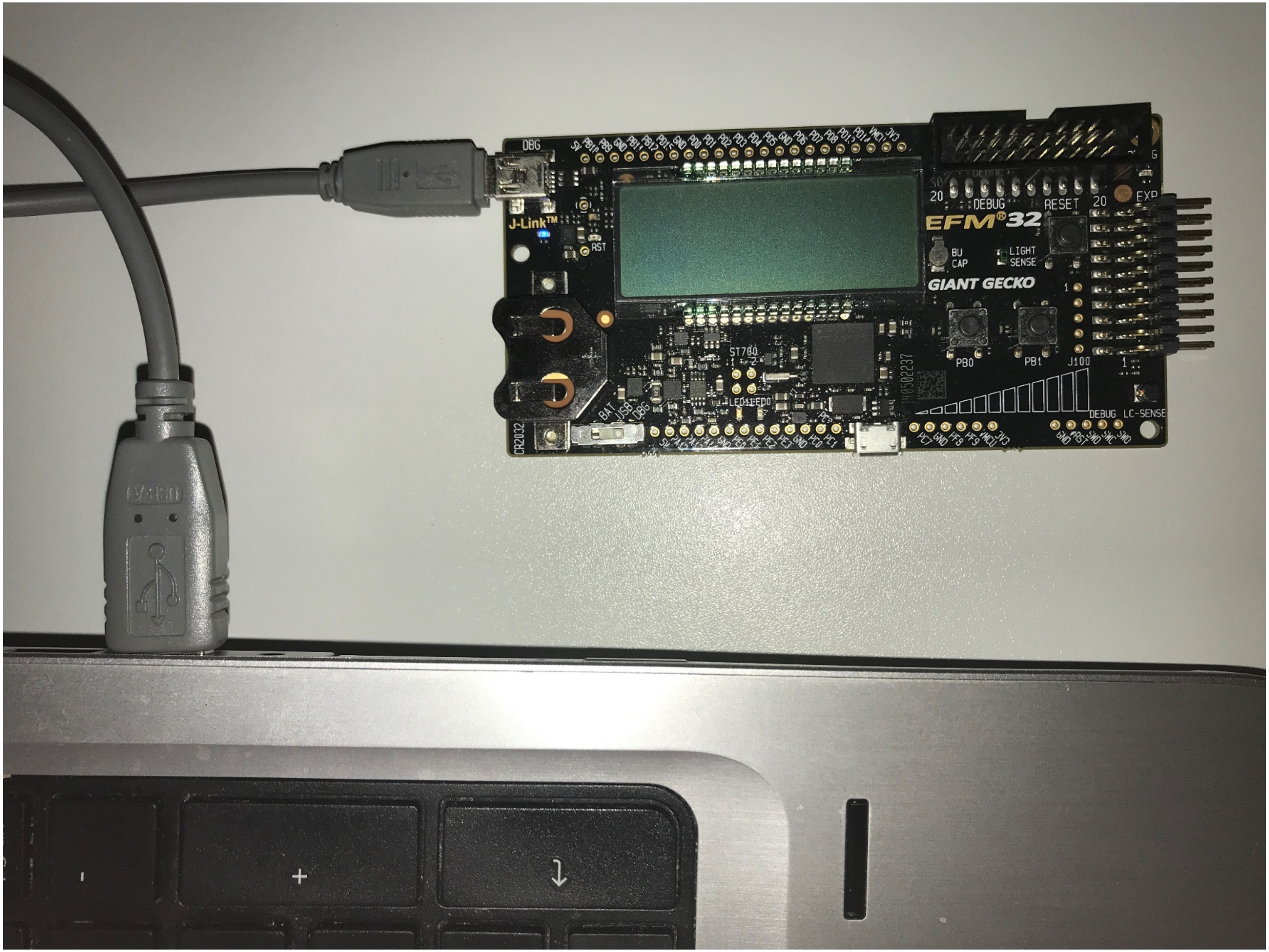


1.3) Power supply

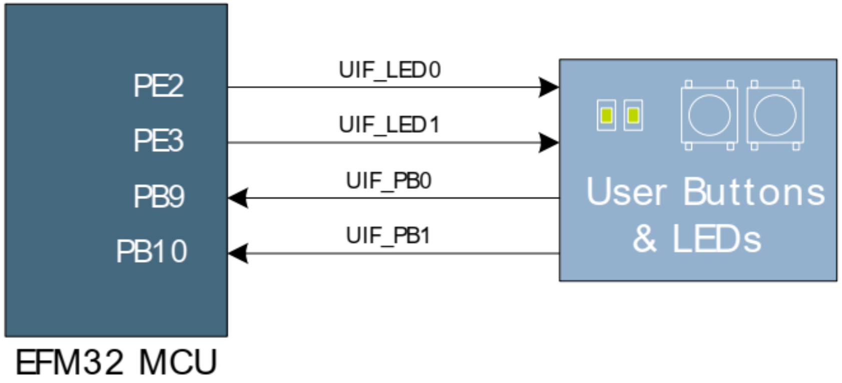


- DBG: via on-board debugger energy monitor can be used (use this)
- BAT: use CR2032 battery
- USB: MCU integrated voltage regulator is used

1.3) Power supply and proper connection



1.4) Peripherals-Buttons/LEDs

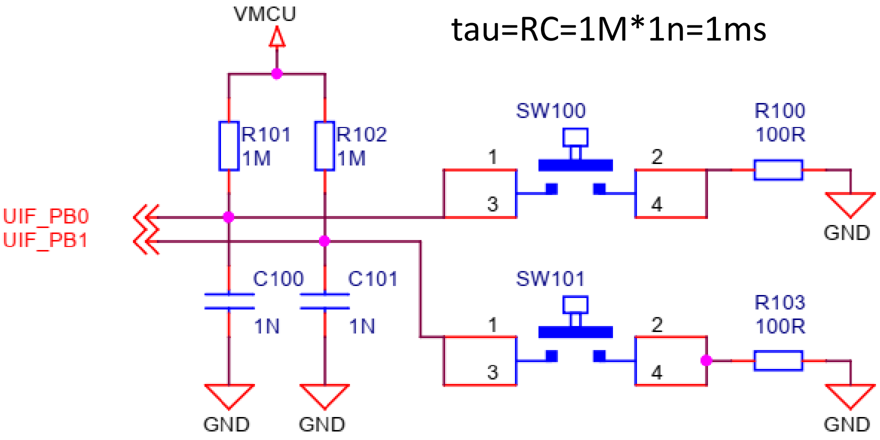


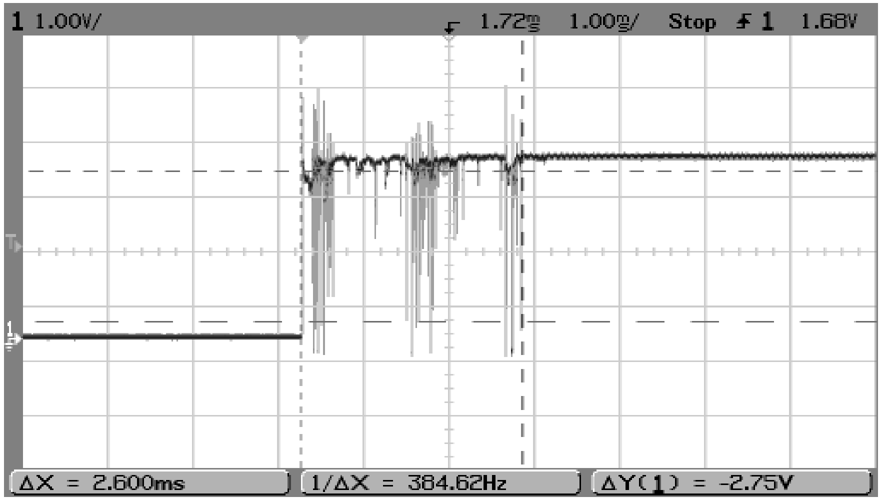
PB0=push button nr. 0

PB9=9th bit of port B

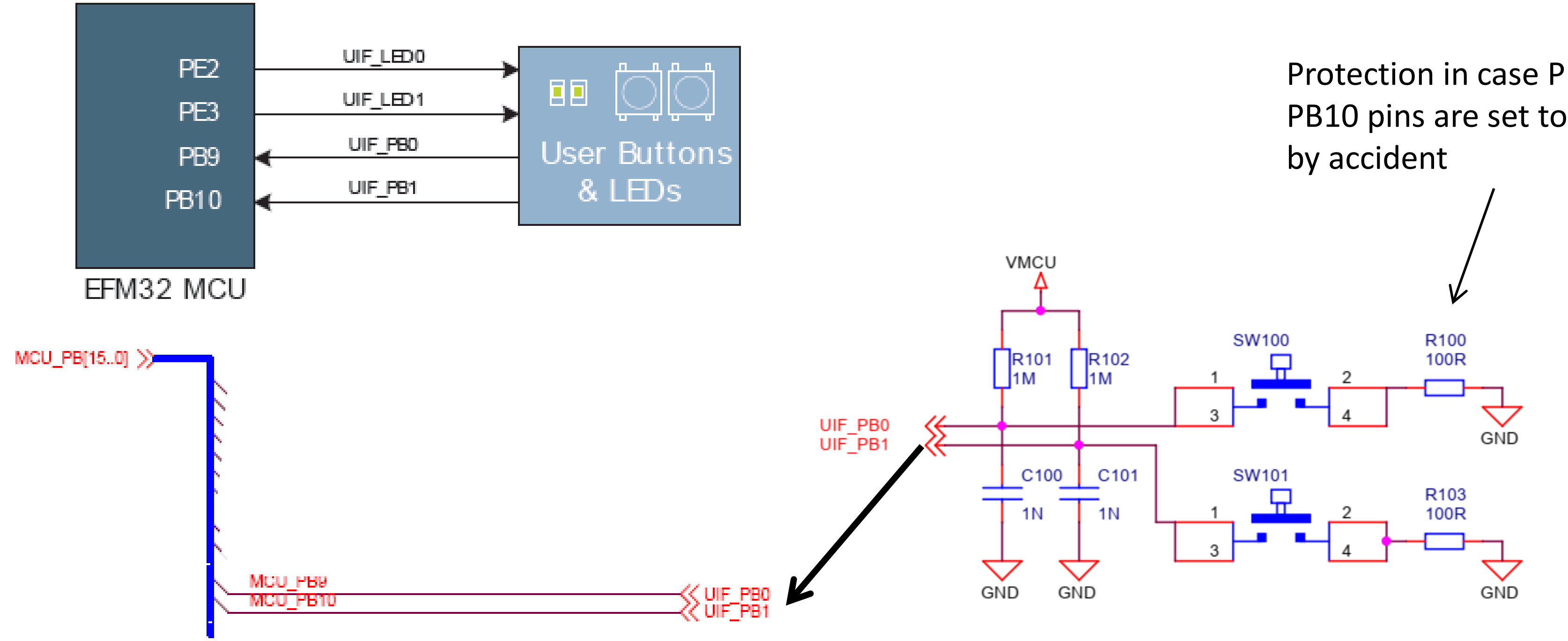
PE3=3rd bit of port E

- Push buttons are debounced by RC filter to avoid:


tau=RC=1M*1n=1ms



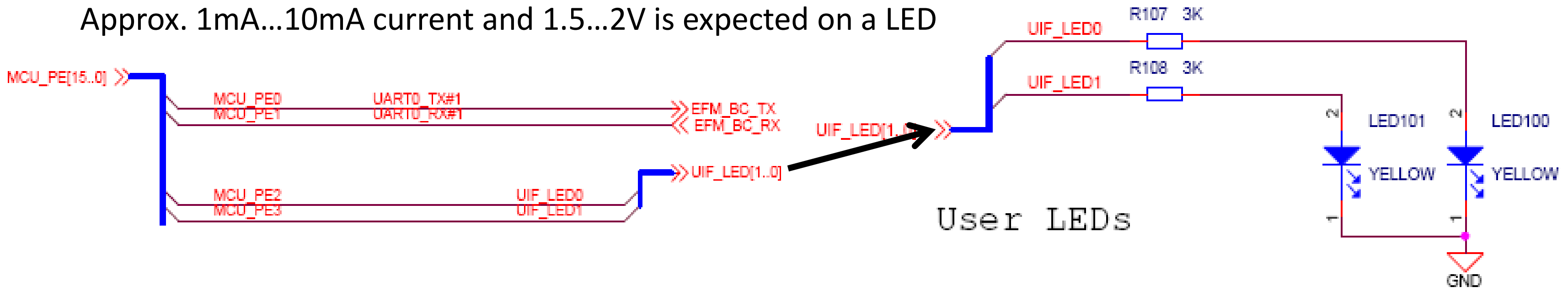
1.4) Peripherals-Buttons/LEDs



Protection in case PB9 or PB10 pins are set to output by accident

LED shunt resistor: (3.3V-2V)/3kΩ≈0.4mA

Approx. 1mA…10mA current and 1.5…2V is expected on a LED



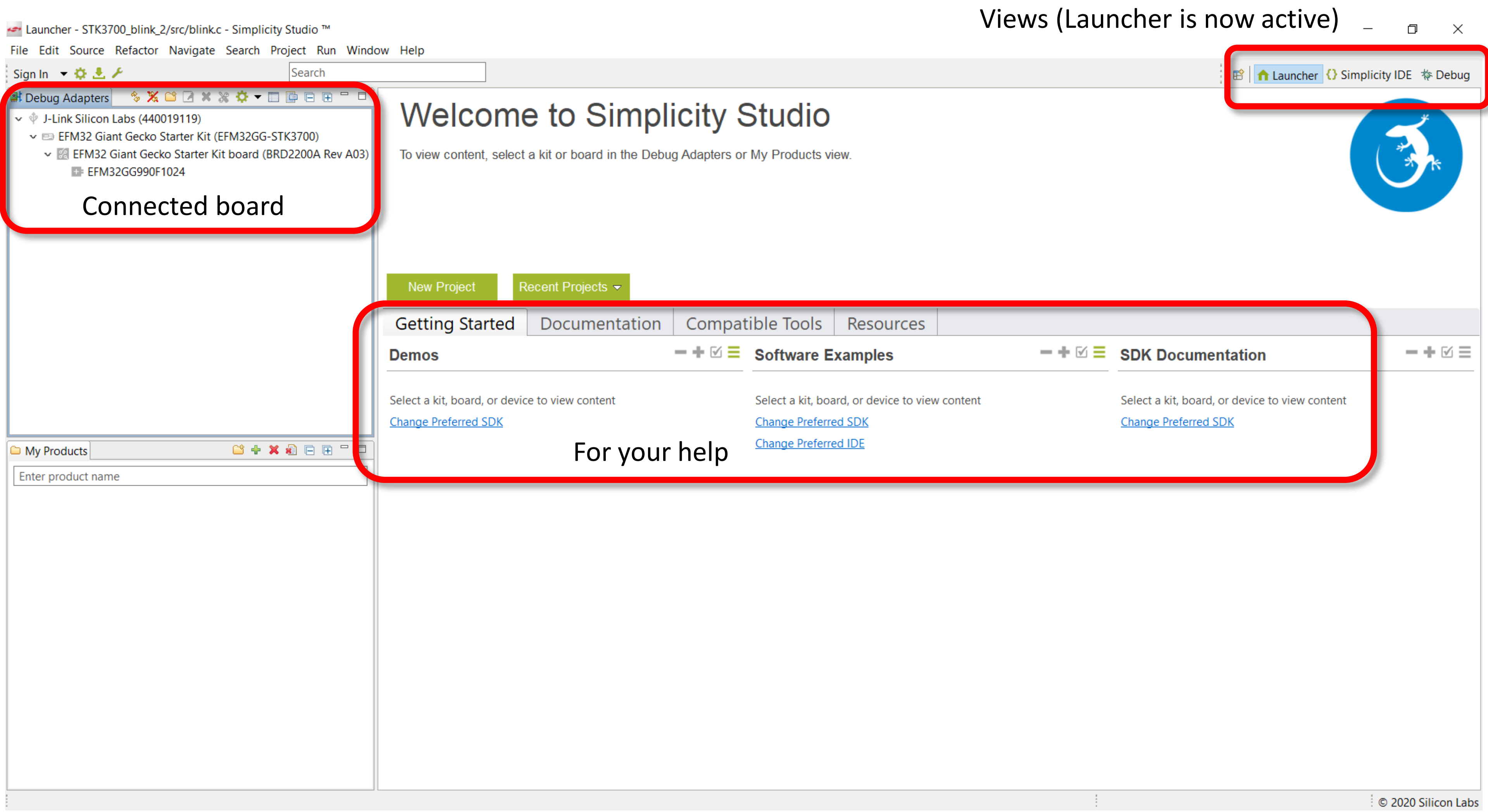
1.5) Board Controller

- Responsible for controlling board level tasks like debugger and Advanced Energy Monitor
- Interface is provided between the EFM32 and the board controller in the form of a UART connection
  - Set the EFM_BC_EN (PF7) line high
  - Use the lines
    EFM_BC_TX (PE0)
    and
    EFM_BC_RX (PE1)
- Board Support Package (bsp) is to be installed

2) Integrated Development Environment

- Integrated development environment (IDE): Simplicity Studio 4
- www.silabs.com/products/development-tools/software/simplicity-studio

2.1) Getting started with IDE-Launcher



- J-Link Silicon Labs (440019119)
  - EFM32 Giant Gecko Starter Kit (EFM32GG-STK3700)
    - EFM32 Giant Gecko Starter Kit board (BRD2200A Rev A03)
      - EFM32GG990F1024

Welcome to Simplicity Studio

To view content, select a kit or board in the Debug Adapters or My Products view.

Select a kit, board, or device to view content

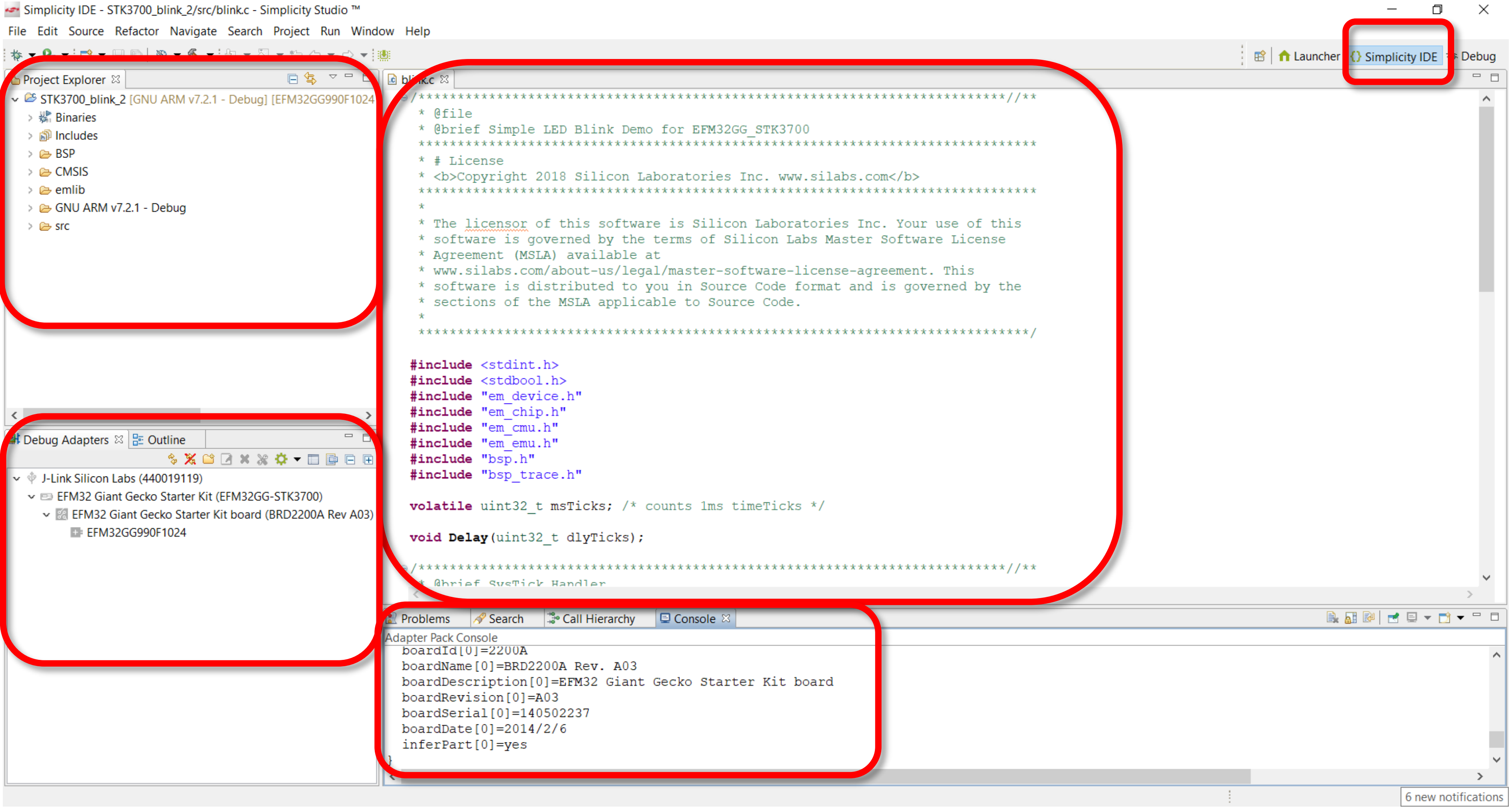
Views (Launcher is now active)

Connected board

For your help

2.2) Getting started with IDE-Simplicity IDE

View (Simplicity IDE active)


Simplicity IDE - STK3700_blink_2/src/blink.c - Simplicity Studio ™
File Edit Source Refactor Navigate Search Project Run Window Help
Launcher
Simplicity IDE
Debug
Project Explorer
STK3700_blink_2 [GNU ARM v7.2.1 - Debug] [EFM32GG990F1024
Binaries
Includes
BSP
CMSIS
emlib
GNU ARM v7.2.1 - Debug
src
Debug Adapters
Outline
J-Link Silicon Labs (440019119)
EFM32 Giant Gecko Starter Kit (EFM32GG-STK3700)
EFM32 Giant Gecko Starter Kit board (BRD2200A Rev A03)
EFM32GG990F1024
blink.c
Problems
Search
Call Hierarchy
Console
Adapter Pack Console
6 new notifications

```
/***************************************************************************//**
 * @file
 * @brief Simple LED Blink Demo for EFM32GG_STK3700
 *******************************************************************************
 * # License
 * <b>Copyright 2018 Silicon Laboratories Inc. www.silabs.com</b>
 *******************************************************************************
 *
 * The licensor of this software is Silicon Laboratories Inc. Your use of this
 * software is governed by the terms of Silicon Labs Master Software License
 * Agreement (MSLA) available at
 * www.silabs.com/about-us/legal/master-software-license-agreement. This
 * software is distributed to you in Source Code format and is governed by the
 * sections of the MSLA applicable to Source Code.
 *
 ******************************************************************************/

#include <stdint.h>
#include <stdbool.h>
#include "em_device.h"
#include "em_chip.h"
#include "em_cmu.h"
#include "em_emu.h"
#include "bsp.h"
#include "bsp_trace.h"

volatile uint32_t msTicks; /* counts 1ms timeTicks */

void Delay(uint32_t dlyTicks);

/***************************************************************************//**
 * @brief SysTick_Handler
```

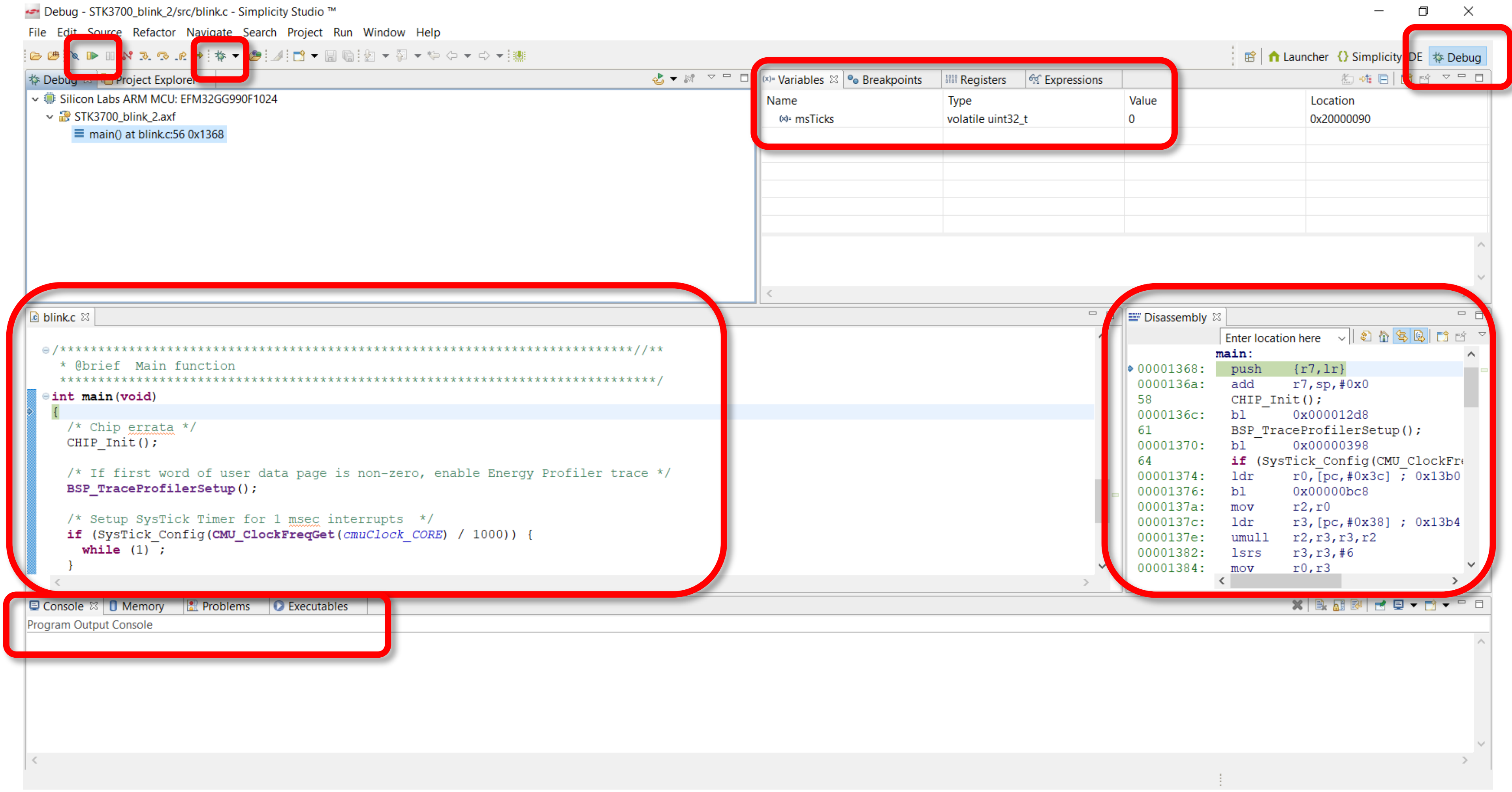
```
  boardId[0]=2200A
  boardName[0]=BRD2200A Rev. A03
  boardDescription[0]=EFM32 Giant Gecko Starter Kit board
  boardRevision[0]=A03
  boardSerial[0]=140502237
  boardDate[0]=2014/2/6
  inferPart[0]=yes
}
```

2.3) Getting started with IDE-Debug

Run

Debug deploy and run

View (Debug active)



```
Silicon Labs ARM MCU: EFM32GG990F1024
  STK3700_blink_2.axf
    main() at blink.c:56 0x1368
```

Name	Type	Value	Location
msTicks	volatile uint32_t	0	0x20000090

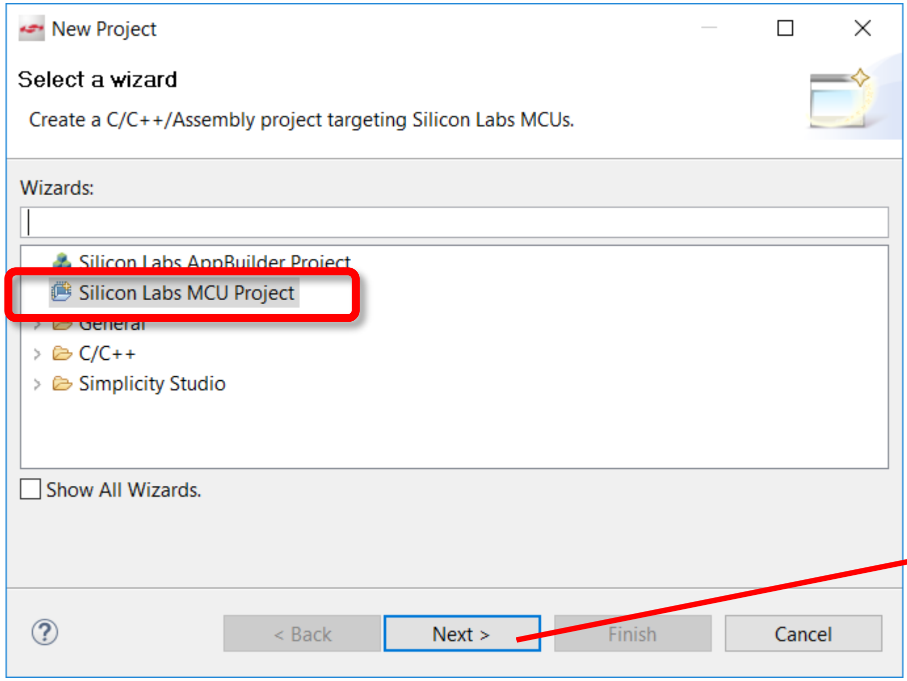
```c
/*************************************************************************//**
 * @brief  Main function
 *****************************************************************************/
int main(void)
{
  /* Chip errata */
  CHIP_Init();

  /* If first word of user data page is non-zero, enable Energy Profiler trace */
  BSP_TraceProfilerSetup();

  /* Setup SysTick Timer for 1 msec interrupts  */
  if (SysTick_Config(CMU_ClockFreqGet(cmuClock_CORE) / 1000)) {
    while (1) ;
  }
```

```
          main:
00001368:   push     {r7,lr}
0000136a:   add      r7,sp,#0x0
58          CHIP_Init();
0000136c:   bl       0x000012d8
61          BSP_TraceProfilerSetup();
00001370:   bl       0x00000398
64          if (SysTick_Config(CMU_ClockFr
00001374:   ldr      r0,[pc,#0x3c] ; 0x13b0
00001376:   bl       0x00000bc8
0000137a:   mov      r2,r0
0000137c:   ldr      r3,[pc,#0x38] ; 0x13b4
0000137e:   umull    r2,r3,r3,r2
00001382:   lsrs     r3,r3,#6
00001384:   mov      r0,r3
```

Program Output Console

3) Start a new project

- File->New->Project:



New Project

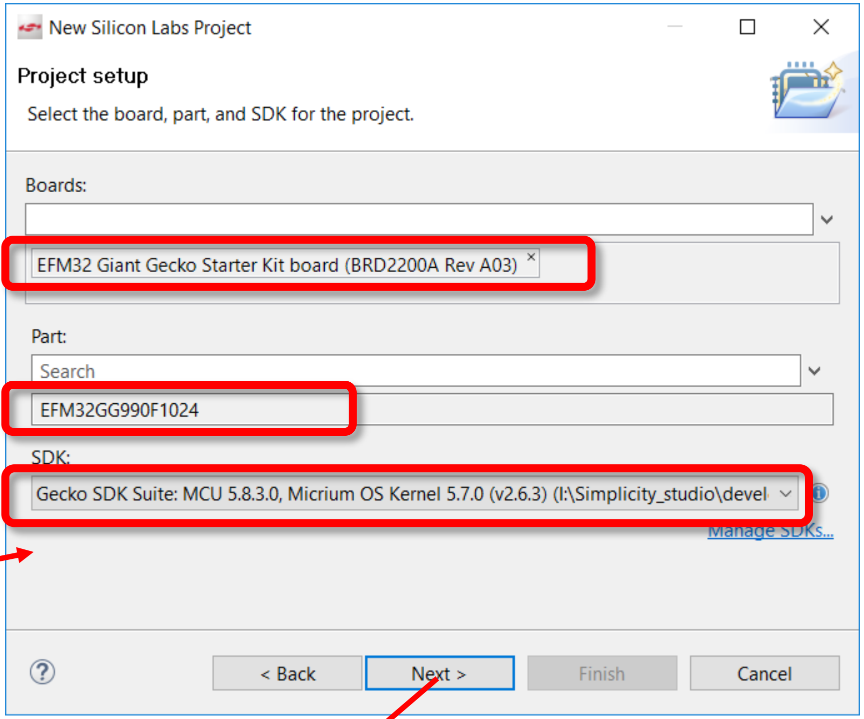
Select a wizard

Create a C/C++/Assembly project targeting Silicon Labs MCUs.

Wizards:

- Silicon Labs AppBuilder Project
- Silicon Labs MCU Project
- General
- C/C++
- Simplicity Studio

□ Show All Wizards.



New Silicon Labs Project

Project setup

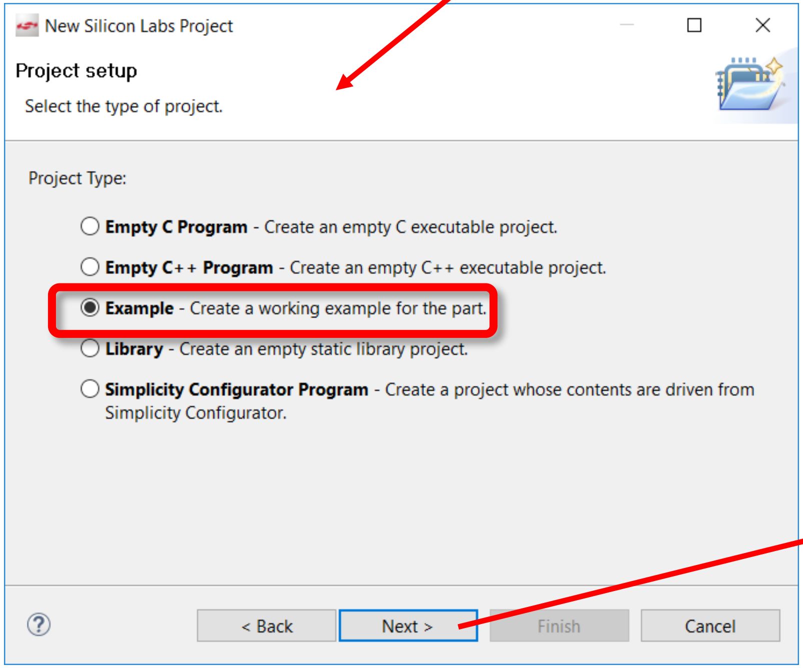
Select the board, part, and SDK for the project.

Boards: EFM32 Giant Gecko Starter Kit board (BRD2200A Rev A03)

Part: EFM32GG990F1024

SDK: Gecko SDK Suite: MCU 5.8.3.0, Micrium OS Kernel 5.7.0 (v2.6.3) (I:\Simplicity_studio\devel

3) Start a new project



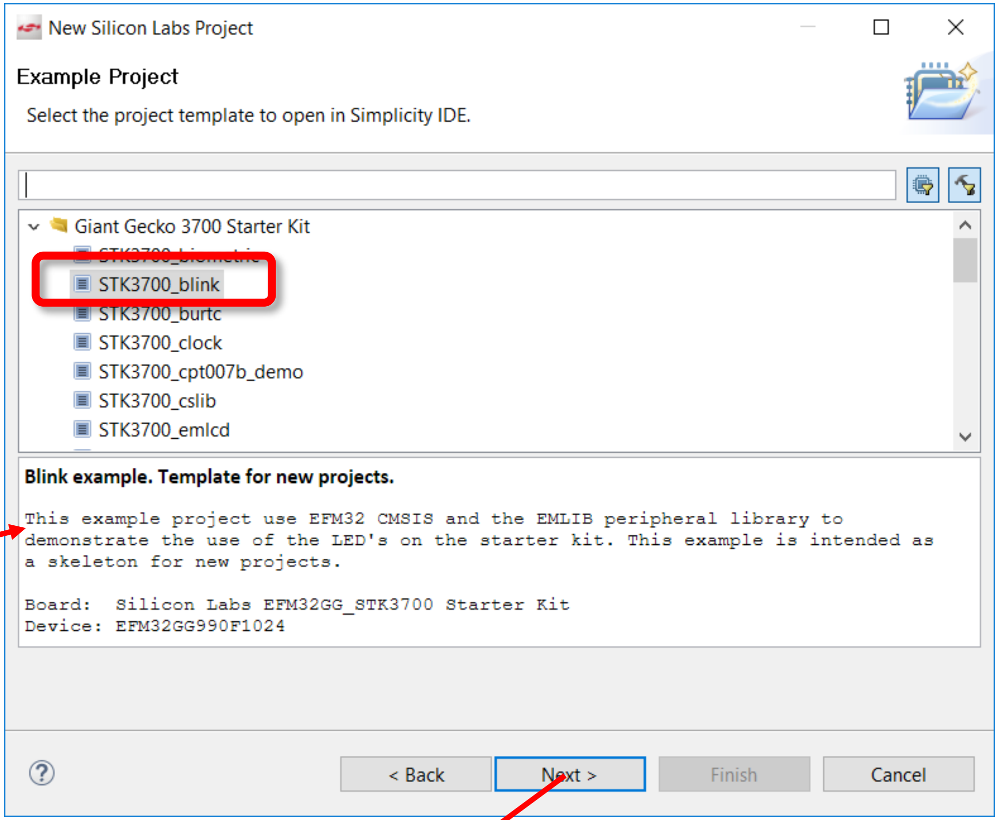
New Silicon Labs Project

Project setup

Select the type of project.

Project Type:

- ○ **Empty C Program** - Create an empty C executable project.
- ○ **Empty C++ Program** - Create an empty C++ executable project.
- ● **Example** - Create a working example for the part.
- ○ **Library** - Create an empty static library project.
- ○ **Simplicity Configurator Program** - Create a project whose contents are driven from Simplicity Configurator.



New Silicon Labs Project

Example Project

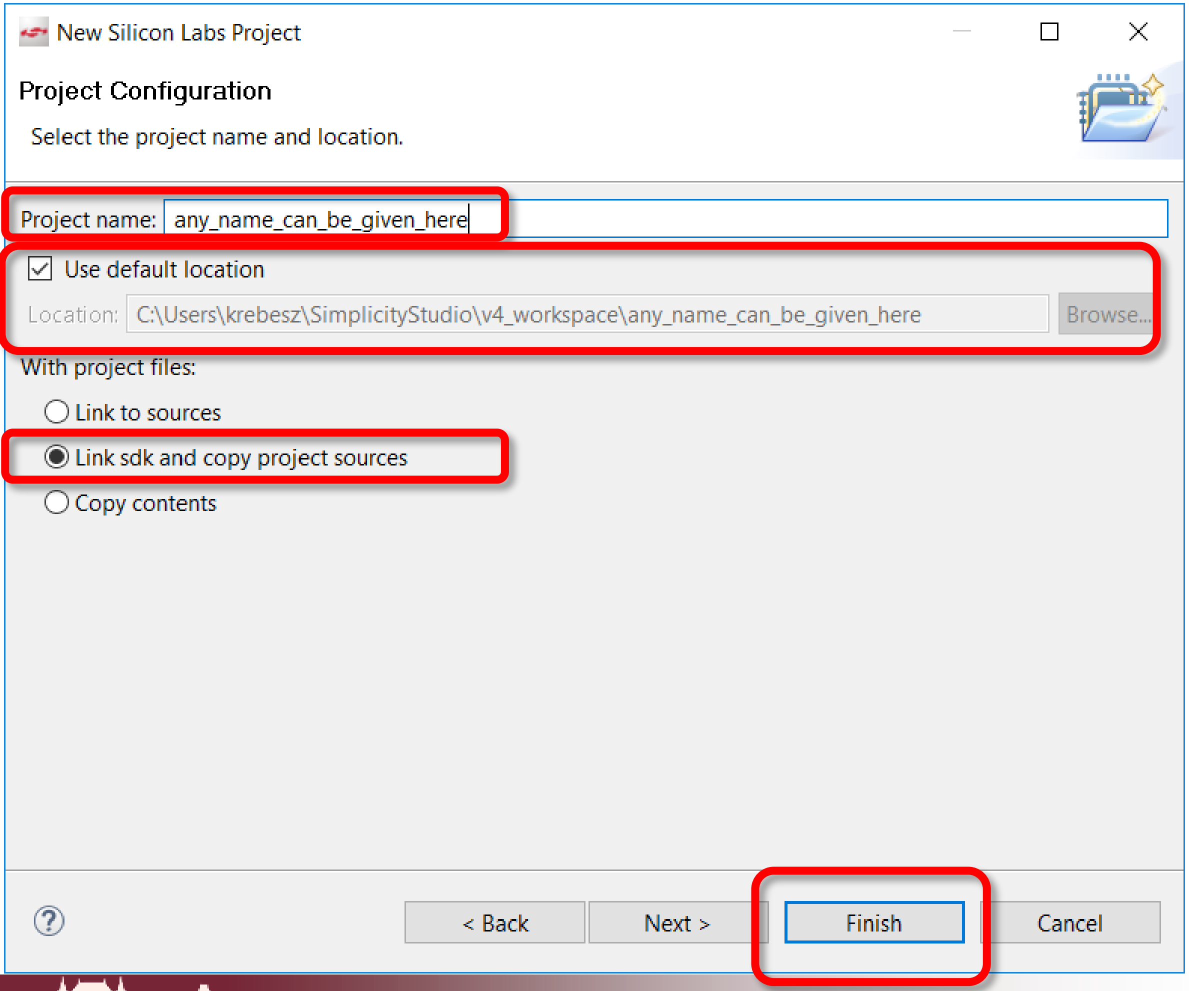
Select the project template to open in Simplicity IDE.

- Giant Gecko 3700 Starter Kit
  - STK3700_biometric
  - STK3700_blink
  - STK3700_burtc
  - STK3700_clock
  - STK3700_cpt007b_demo
  - STK3700_cslib
  - STK3700_emlcd

Blink example. Template for new projects.

```
This example project use EFM32 CMSIS and the EMLIB peripheral library to
demonstrate the use of the LED's on the starter kit. This example is intended as
a skeleton for new projects.

Board:  Silicon Labs EFM32GG_STK3700 Starter Kit
Device: EFM32GG990F1024
```

3) Start a new project



New Silicon Labs Project

Project Configuration

Select the project name and location.

Project name: any_name_can_be_given_here

☑ Use default location

Location: C:\Users\krebesz\SimplicityStudio\v4_workspace\any_name_can_be_given_here

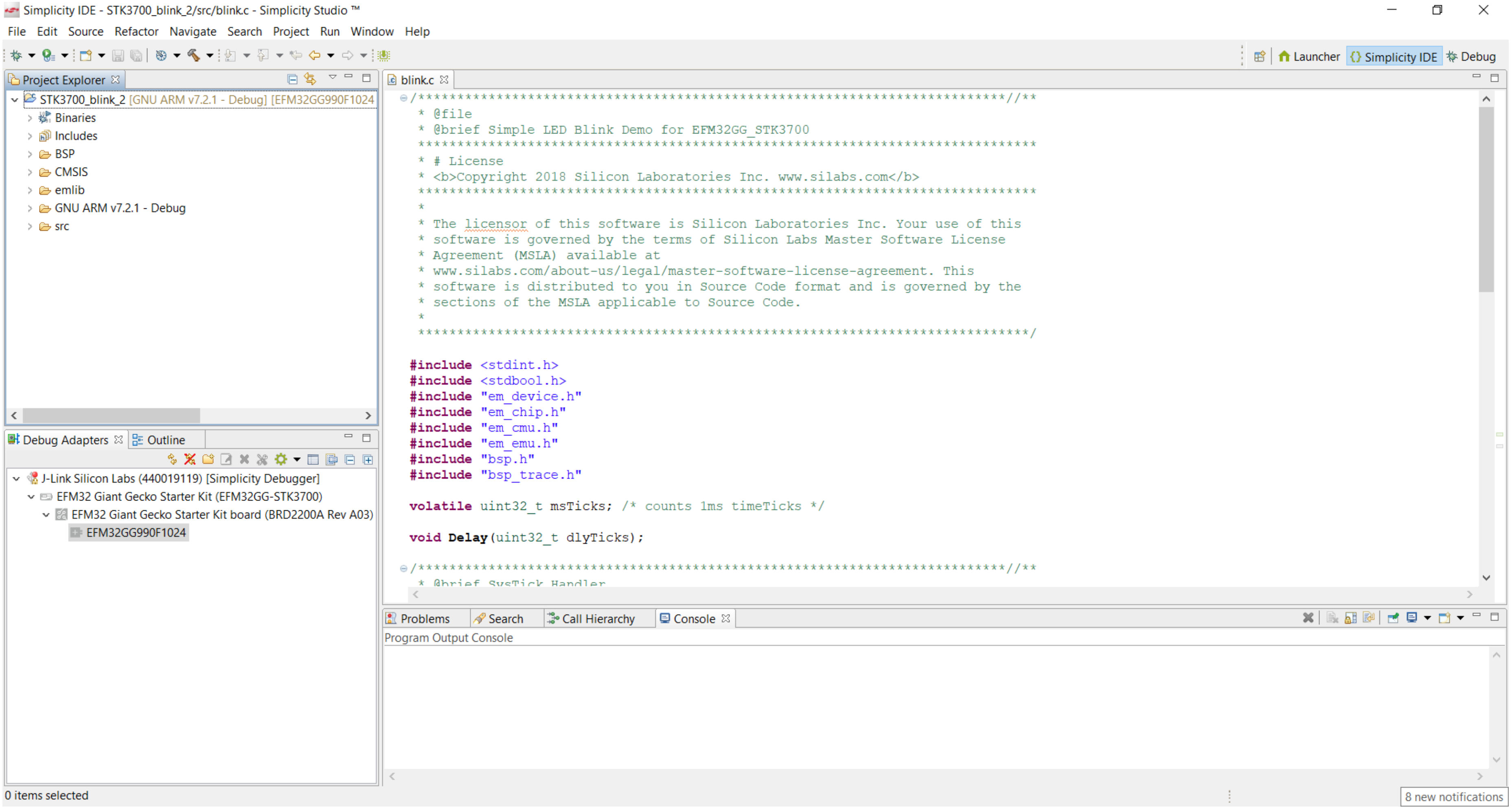
With project files:

○ Link to sources

● Link sdk and copy project sources

○ Copy contents

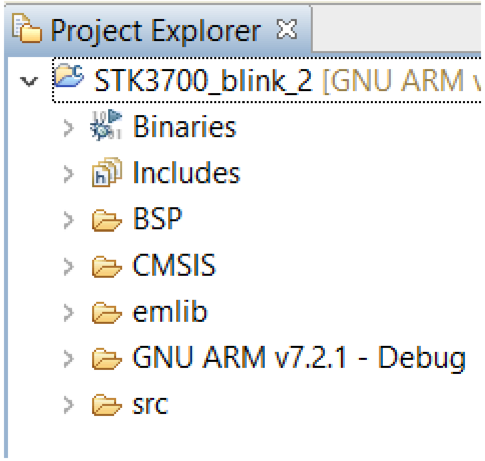
4) Example project created



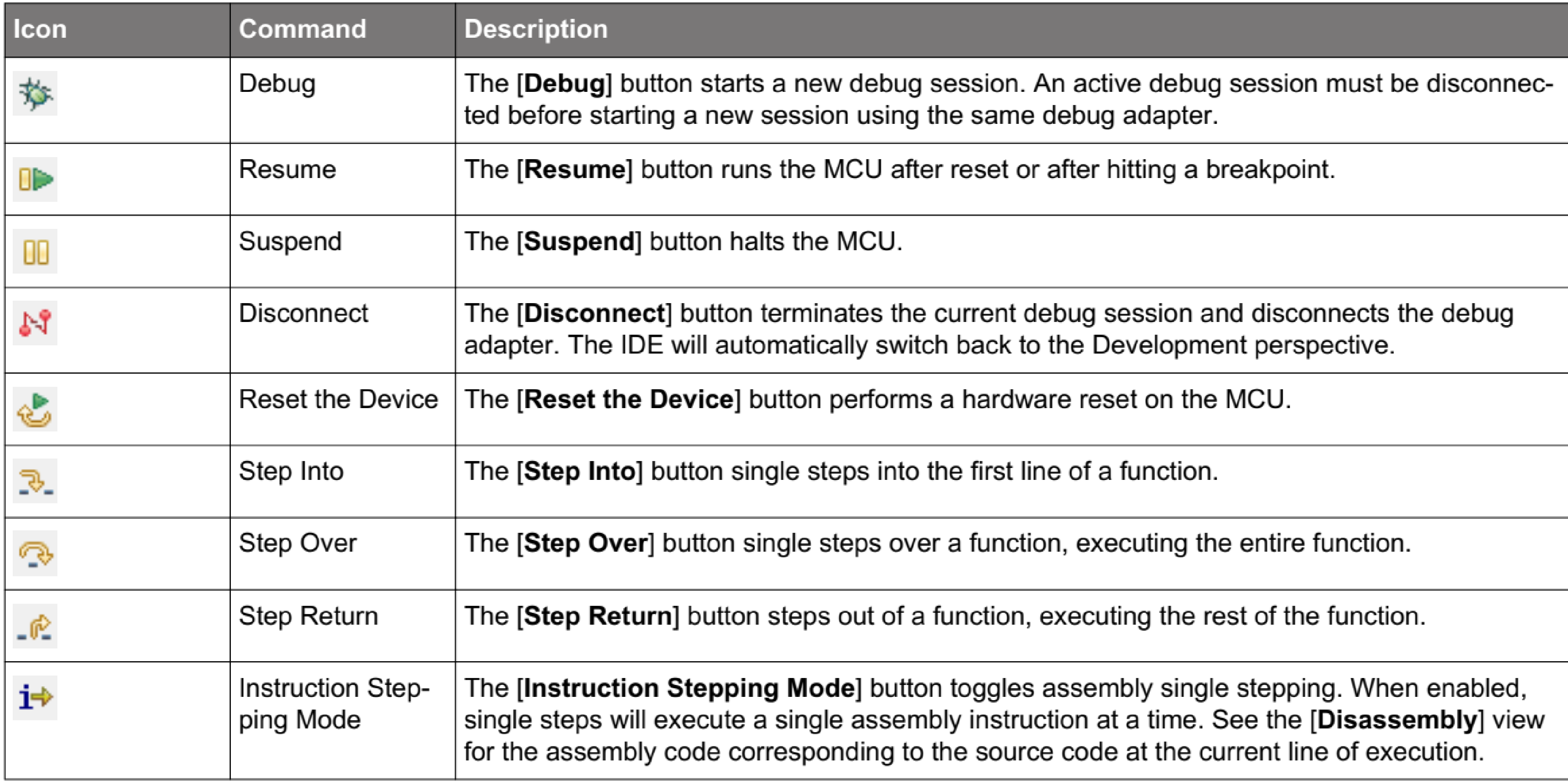
```
/***************************************************************************//**
 * @file
 * @brief Simple LED Blink Demo for EFM32GG_STK3700
 *******************************************************************************
 * # License
 * <b>Copyright 2018 Silicon Laboratories Inc. www.silabs.com</b>
 *******************************************************************************
 *
 * The licensor of this software is Silicon Laboratories Inc. Your use of this
 * software is governed by the terms of Silicon Labs Master Software License
 * Agreement (MSLA) available at
 * www.silabs.com/about-us/legal/master-software-license-agreement. This
 * software is distributed to you in Source Code format and is governed by the
 * sections of the MSLA applicable to Source Code.
 *
 ******************************************************************************/

#include <stdint.h>
#include <stdbool.h>
#include "em_device.h"
#include "em_chip.h"
#include "em_cmu.h"
#include "em_emu.h"
#include "bsp.h"
#include "bsp_trace.h"

volatile uint32_t msTicks; /* counts 1ms timeTicks */

void Delay(uint32_t dlyTicks);

/***************************************************************************//**
 * @brief SysTick_Handler
```

4.1) Project Explorer

- Binaries: “raw” files (hex, bin)
- Includes: header files (function defs)
- BSP: board support package
- CMSIS: core management
- emlib: manages the whole uC
- GNU… : compiled SW components
- src: source files

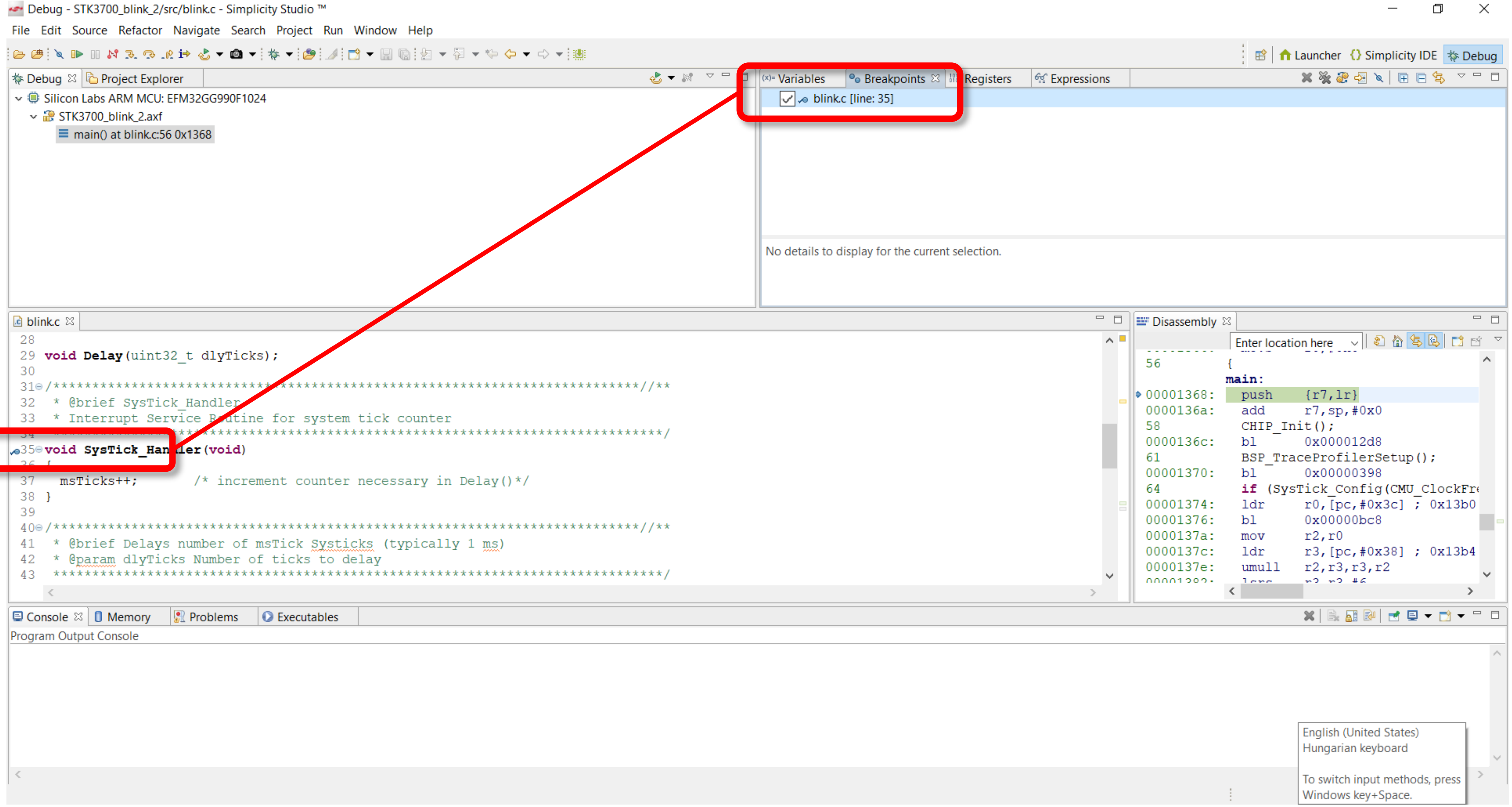


4.2) Debug mode



Icon	Command	Description
	Debug	The [**Debug**] button starts a new debug session. An active debug session must be disconnected before starting a new session using the same debug adapter.
	Resume	The [**Resume**] button runs the MCU after reset or after hitting a breakpoint.
	Suspend	The [**Suspend**] button halts the MCU.
	Disconnect	The [**Disconnect**] button terminates the current debug session and disconnects the debug adapter. The IDE will automatically switch back to the Development perspective.
	Reset the Device	The [**Reset the Device**] button performs a hardware reset on the MCU.
	Step Into	The [**Step Into**] button single steps into the first line of a function.
	Step Over	The [**Step Over**] button single steps over a function, executing the entire function.
	Step Return	The [**Step Return**] button steps out of a function, executing the rest of the function.
i→	Instruction Stepping Mode	The [**Instruction Stepping Mode**] button toggles assembly single stepping. When enabled, single steps will execute a single assembly instruction at a time. See the [**Disassembly**] view for the assembly code corresponding to the source code at the current line of execution.

4.2.1) Breakpoints



Debug - STK3700_blink_2/src/blink.c - Simplicity Studio ™

Silicon Labs ARM MCU: EFM32GG990F1024
STK3700_blink_2.axf
main() at blink.c:56 0x1368

☑ blink.c [line: 35]

No details to display for the current selection.

```
28
29 void Delay(uint32_t dlyTicks);
30
31 /***********************************************************************//**
32  * @brief SysTick_Handler
33  * Interrupt Service Routine for system tick counter
34  *****************************************************************************/
35 void SysTick_Handler(void)
36 {
37   msTicks++;       /* increment counter necessary in Delay()*/
38 }
39
40 /***********************************************************************//**
41  * @brief Delays number of msTick Systicks (typically 1 ms)
42  * @param dlyTicks Number of ticks to delay
43  *****************************************************************************/
```

```
56          {
           main:
00001368:   push    {r7,lr}
0000136a:   add     r7,sp,#0x0
58          CHIP_Init();
0000136c:   bl      0x000012d8
61          BSP_TraceProfilerSetup();
00001370:   bl      0x00000398
64          if (SysTick_Config(CMU_ClockFre
00001374:   ldr     r0,[pc,#0x3c] ; 0x13b0
00001376:   bl      0x00000bc8
0000137a:   mov     r2,r0
0000137c:   ldr     r3,[pc,#0x38] ; 0x13b4
0000137e:   umull   r2,r3,r3,r2
```

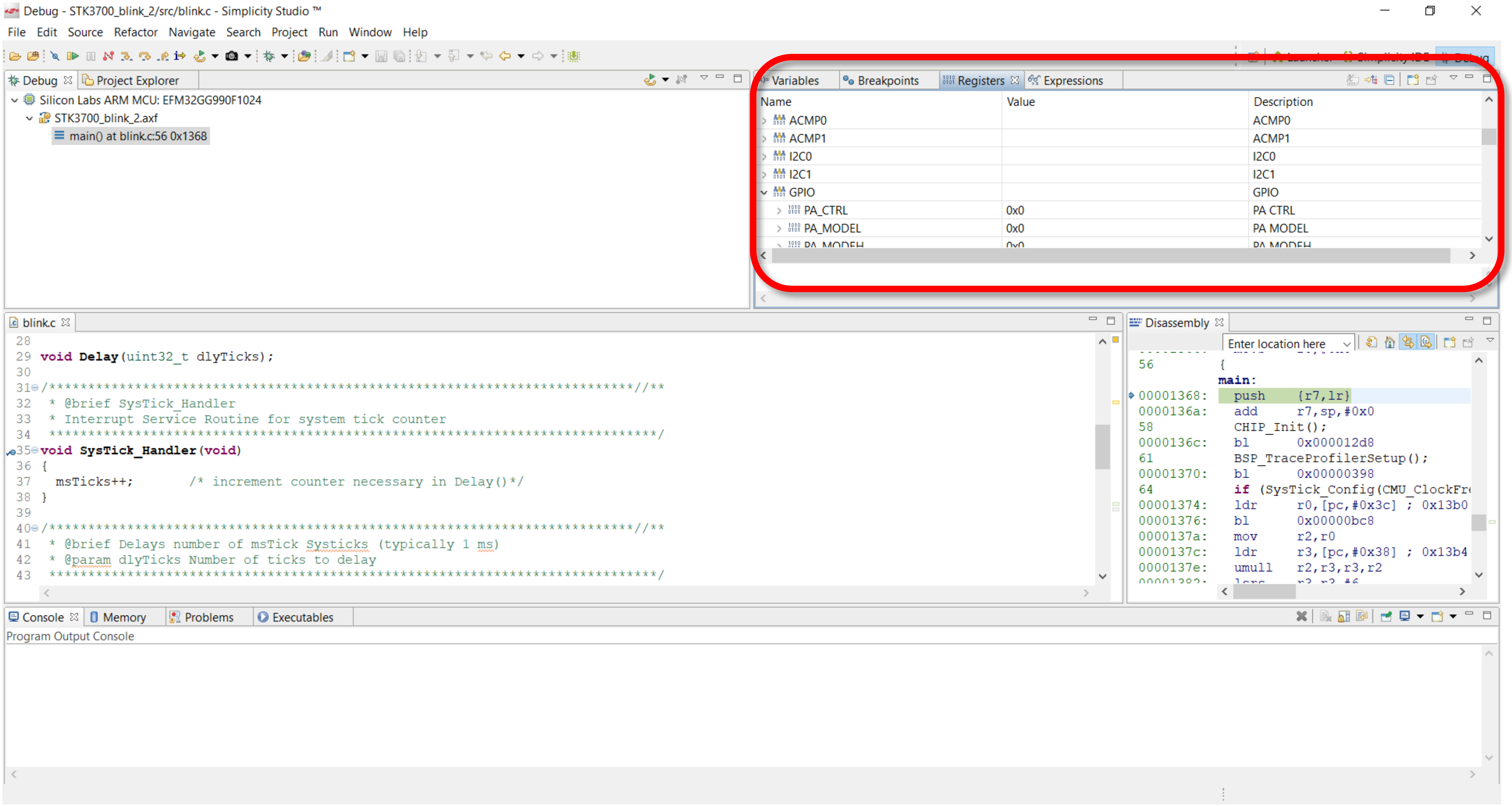
Program Output Console

English (United States)
Hungarian keyboard

To switch input methods, press Windows key+Space.

- Right click on the line to be able to add Breakpoint

4.2.2) Register values



Silicon Labs ARM MCU: EFM32GG990F1024
STK3700_blink_2.axf
main() at blink.c:56 0x1368

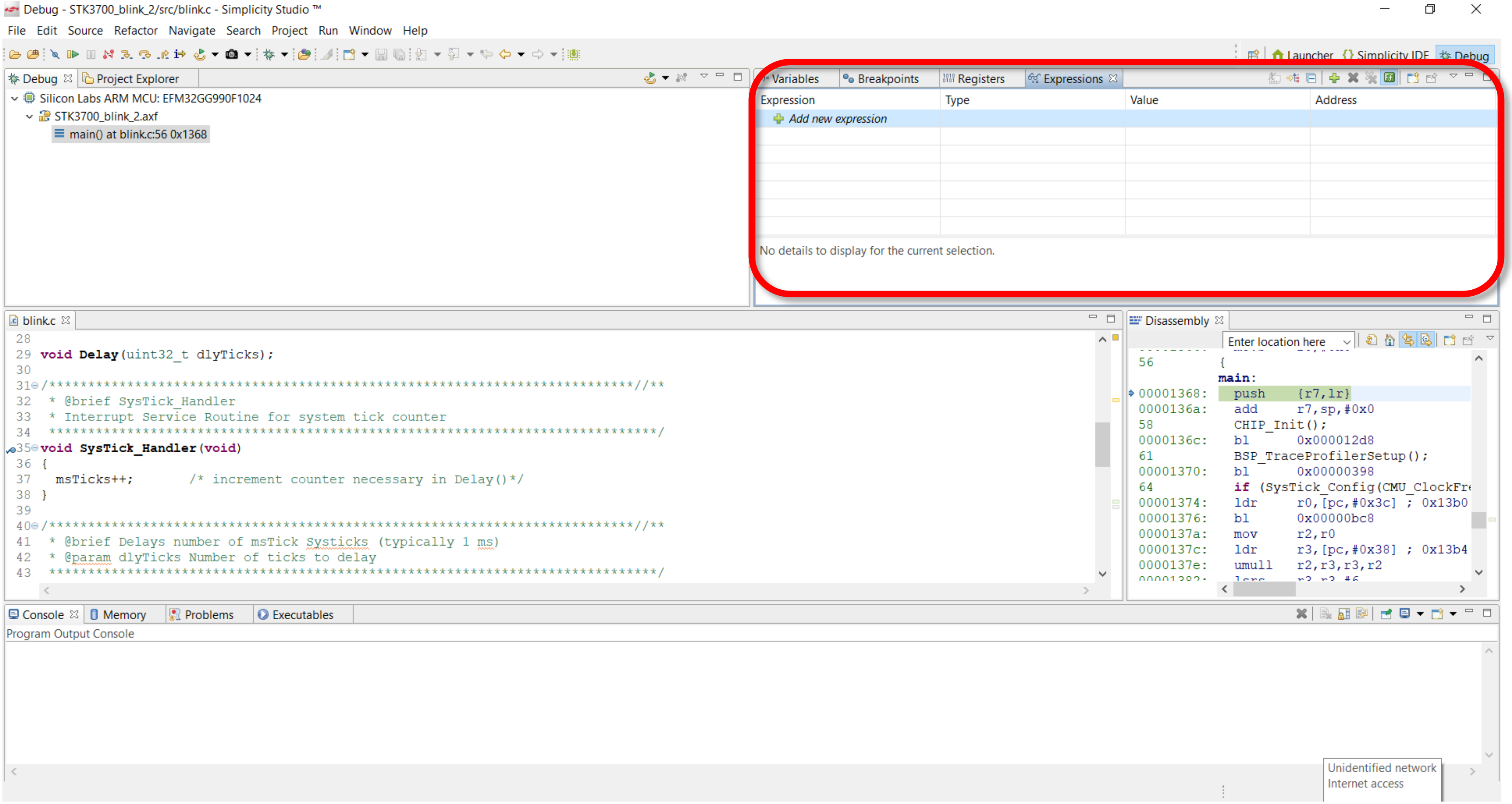
Name	Value	Description
ACMP0		ACMP0
ACMP1		ACMP1
I2C0		I2C0
I2C1		I2C1
GPIO		GPIO
PA_CTRL	0x0	PA CTRL
PA_MODEL	0x0	PA MODEL
PA_MODEH	0x0	PA MODEH

```
void Delay(uint32_t dlyTicks);

/**************************************************************************//**
 * @brief SysTick_Handler
 * Interrupt Service Routine for system tick counter
 *****************************************************************************/
void SysTick_Handler(void)
{
  msTicks++;       /* increment counter necessary in Delay()*/
}

/**************************************************************************//**
 * @brief Delays number of msTick Systicks (typically 1 ms)
 * @param dlyTicks Number of ticks to delay
 *****************************************************************************/
```

```
56          {
          main:
00001368:   push      {r7,lr}
0000136a:   add       r7,sp,#0x0
58          CHIP_Init();
0000136c:   bl        0x000012d8
61          BSP_TraceProfilerSetup();
00001370:   bl        0x00000398
64          if (SysTick_Config(CMU_ClockFr
00001374:   ldr       r0,[pc,#0x3c]  ; 0x13b0
00001376:   bl        0x00000bc8
0000137a:   mov       r2,r0
0000137c:   ldr       r3,[pc,#0x38]  ; 0x13b4
0000137e:   umull     r2,r3,r3,r2
```

- Register content can be manipulated

4.2.2) Expressions



Expression	Type	Value	Address
Add new expression			

No details to display for the current selection.

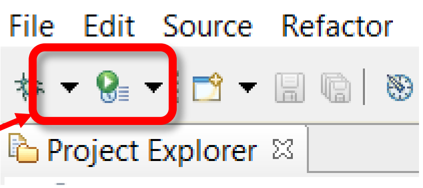
```
void Delay(uint32_t dlyTicks);

/**************************************************************************//**
 * @brief SysTick_Handler
 * Interrupt Service Routine for system tick counter
 *****************************************************************************/
void SysTick_Handler(void)
{
  msTicks++;       /* increment counter necessary in Delay()*/
}

/**************************************************************************//**
 * @brief Delays number of msTick Systicks (typically 1 ms)
 * @param dlyTicks Number of ticks to delay
 *****************************************************************************/
```

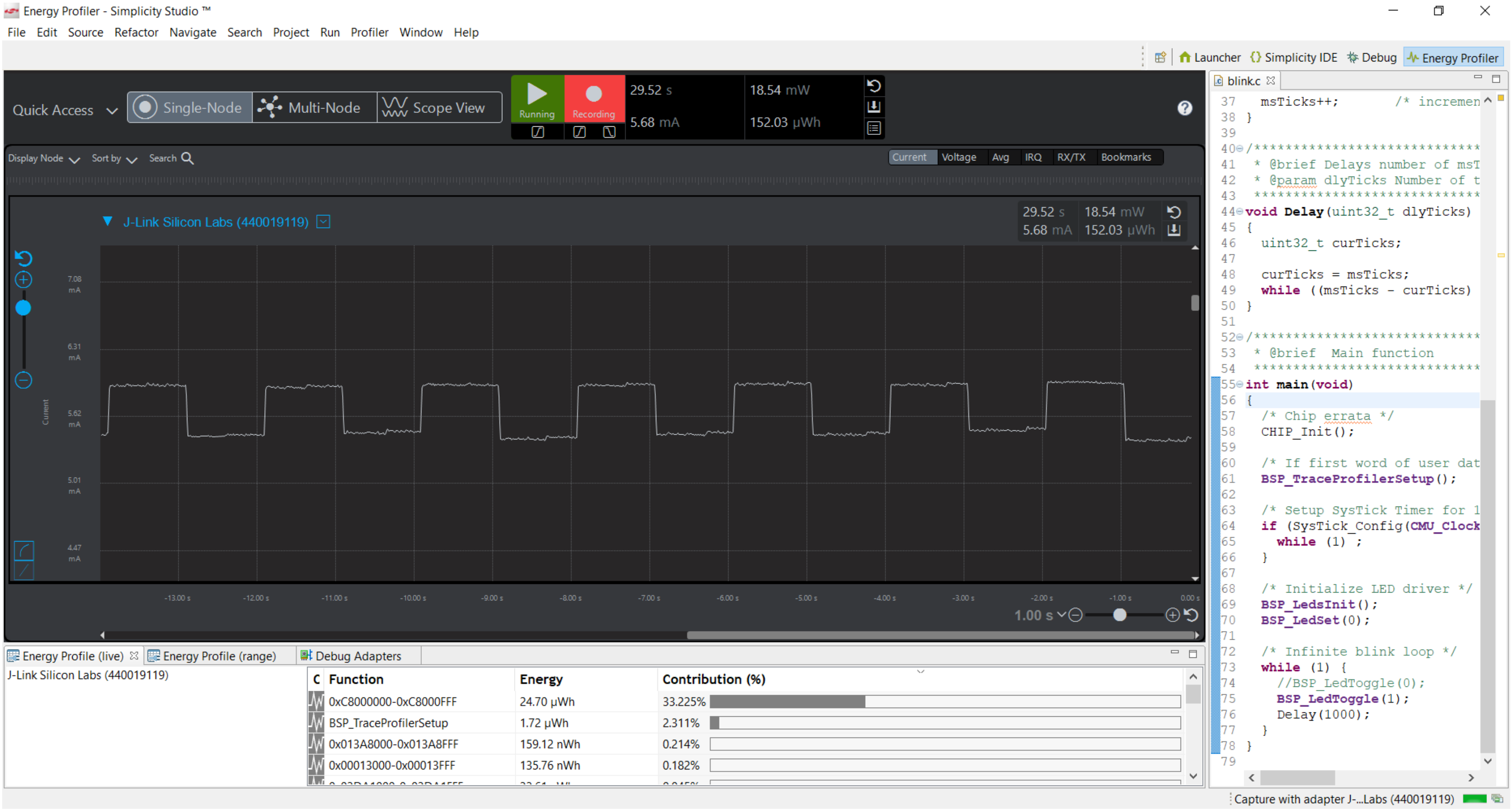
```
56          {
            main:
00001368:   push      {r7,lr}
0000136a:   add       r7,sp,#0x0
58          CHIP_Init();
0000136c:   bl        0x000012d8
61          BSP_TraceProfilerSetup();
00001370:   bl        0x00000398
64          if (SysTick_Config(CMU_ClockFre
00001374:   ldr       r0,[pc,#0x3c] ; 0x13b0
00001376:   bl        0x00000bc8
0000137a:   mov       r2,r0
0000137c:   ldr       r3,[pc,#0x38] ; 0x13b4
0000137e:   umull     r2,r3,r3,r2
```

- Expressions can be entered, e.g.: variable1+variable2

5) Energy profiler

- Disable one LED (use e.g. comment //)
- Switch IDE mode and choose this icon





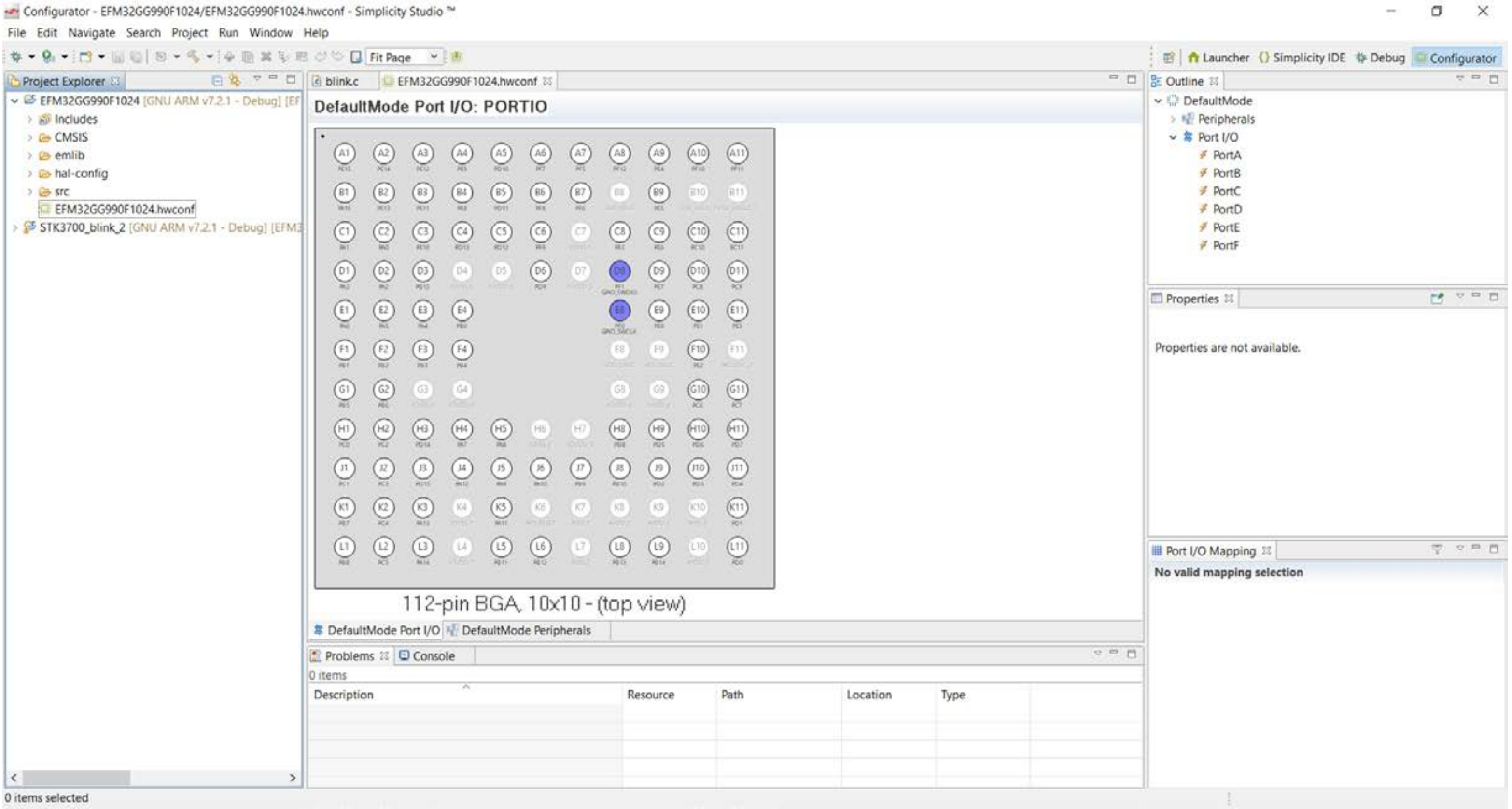
J-Link Silicon Labs (440019119)

C	Function	Energy	Contribution (%)
	0xC8000000-0xC8000FFF	24.70 µWh	33.225%
	BSP_TraceProfilerSetup	1.72 µWh	2.311%
	0x013A8000-0x013A8FFF	159.12 nWh	0.214%
	0x00013000-0x00013FFF	135.76 nWh	0.182%

```c
  msTicks++;        /* incremen
}

/*****************************
 * @brief Delays number of msT
 * @param dlyTicks Number of t
 *****************************
void Delay(uint32_t dlyTicks)
{
  uint32_t curTicks;

  curTicks = msTicks;
  while ((msTicks - curTicks)
}

/*****************************
 * @brief  Main function
 *****************************
int main(void)
{
  /* Chip errata */
  CHIP_Init();

  /* If first word of user dat
  BSP_TraceProfilerSetup();

  /* Setup SysTick Timer for 1
  if (SysTick_Config(CMU_Clock
    while (1) ;
  }

  /* Initialize LED driver */
  BSP_LedsInit();
  BSP_LedSet(0);

  /* Infinite blink loop */
  while (1) {
    //BSP_LedToggle(0);
    BSP_LedToggle(1);
    Delay(1000);
  }
}
```

6) HW configurator

- Project is created by selecting configurator mode
- Simplifies peripheral initialization by presenting peripherals in a graphical user interface



Configurator - EFM32GG990F1024/EFM32GG990F1024.hwconf - Simplicity Studio ™

DefaultMode Port I/O: PORTIO

112-pin BGA, 10x10 - (top view)

Properties are not available.

No valid mapping selection

0 items

0 items selected

7) Code development and manipulation

- Some useful hints
  - Code completion by Content Assist
    - type the first few letters of a function and press [**Ctrl+Space**]
      - display a list of functions that match
      - works for include files as well
  - Symbol expansion
    - stay over a function and information will pop-up
  - Open declaration
    - stay over a variable and press [**F3**]
      - Redirects where it was declared

7.1) Code development - #include

- Use a header file in your program by including it with the C preprocessing directive **#include**
- Two forms exist:
  - #include <file>
    Used for system header files. It searches for a file named 'file' in a standard list of system directories.
  - #include "file"
    Used for header files of your own program. It searches for a file named 'file' in the directory containing the current file.

7.2) Code explanation

- void
  - represents the absence of type
  - specifies that no value is available
- volatile
  - indicate that a value can change and the compiler should be prevented to perform optimization on it (which may lead to change the value into a constant)
- CHIP_Init();
  - HW errors are corrected in SW