Laboratorio di Architetture e Programmazione dei Sistemi Elettronici Industriali

Prof. Luca Benini <luca.benini@unibo.it>

Simone Benatti <simone.benatti@unibo.it>

Filippo Casamassima<filippo.casamassima@unibo.it>

Course organization

- Hands-on session Lab1 **Thursday 4.30pm-7pm**
- Prof. Benini **Friday 9am-11pm room 5.5**
- Lab is available also **Tuesday 3pm-5pm**
- **Exam** Prof. Benini
  - Homeworks (weekly)
  - Final project
  - Final discussion (homeworks + final project)

Course organization

- Is it possible to use either its own **notebook** or **PCs** available in the lab
- For **questions** (homeworks, final project, etc…):
  - http://groups.google.com/group/labarchunibo
- Every question / answer is forwarded to **everyone** who is subscribed to the group

Groups and development boards

- Each **group** is composed by 2 people
- Each group is provided with a **development board**
- Please, **register** your group and your development kit:
  - http://goo.gl/KAGx0
- Boards are **returned** during the final discussion

What you need

- PC with **Windows** (XP / Vista / 7 / 8 /…)
- Development **board**
- **USB** cable
- **Keil uVision IDE** for **ARM**

Keil uVision IDE for ARM

Download and install MDK-ARM

- Download the product from:

  http://www.keil.com/arm/mdk.asp
- Run the downloaded executable
- Follow the instructions displayed by the SETUP program
- Install ST-Link Driver and update ST-Link Firmware

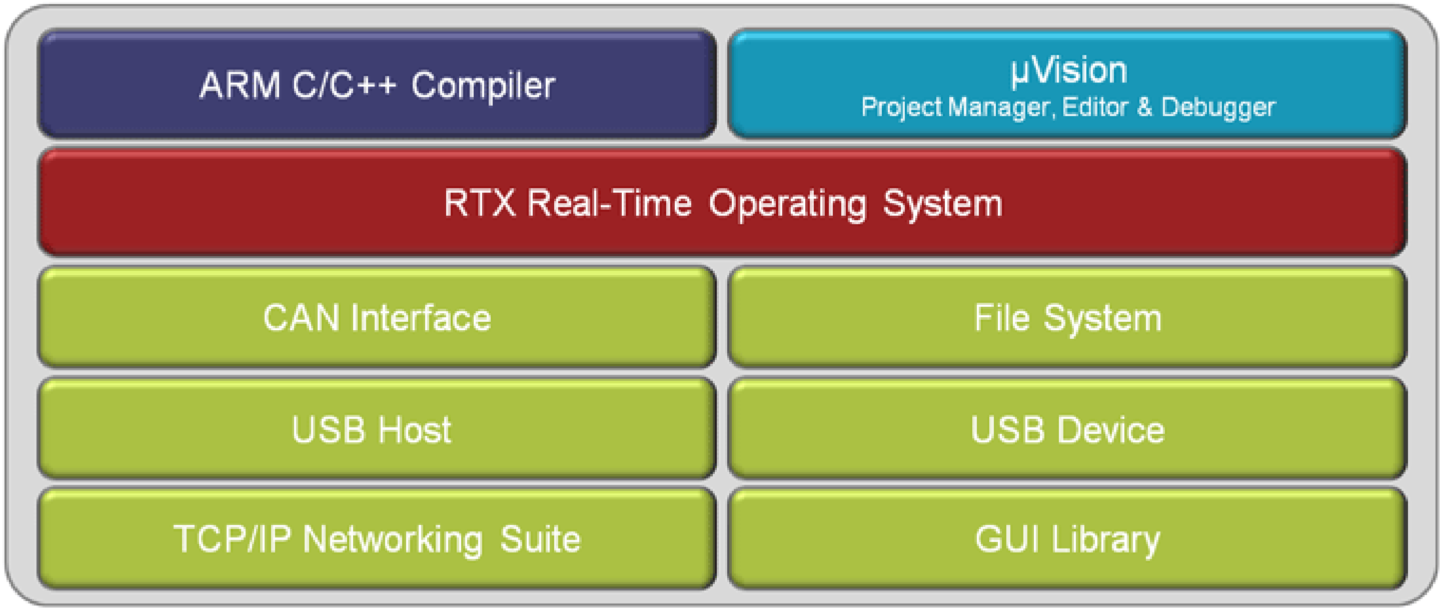
Outline

- Keil Introduction
- Sample program
- Project Options
- Creation of New Project
- Compilation
- Code download
- Debug

Keil MDK-ARM

- The MDK-ARM is a complete software development environment for Cortex™-M, Cortex-R4, ARM7™ and ARM9™ processor-based devices.

- MDK-ARM is specifically designed for microcontroller applications, it is easy to learn and use, yet powerful enough for the most demanding embedded applications.



Your First Porject

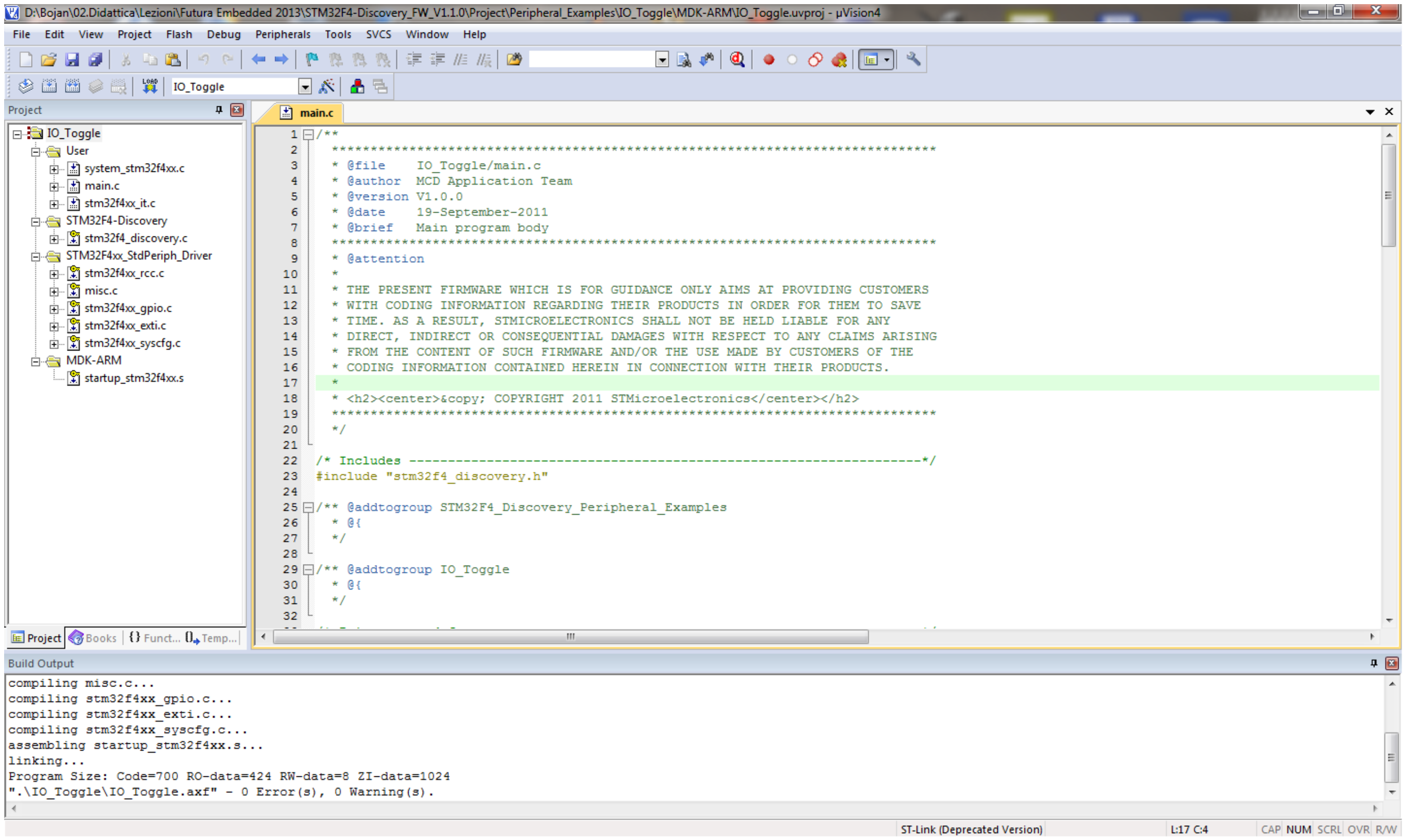
- Copy files from: http://.... (Usb Key)
- Unpack files
- navigate to directory:

..\stm32vldiscovery_Examples and template\Project\Template\MDK-ARM

- Open file: DISCOVER.uvproj
- Keil uVision4 will open

uVision IDE

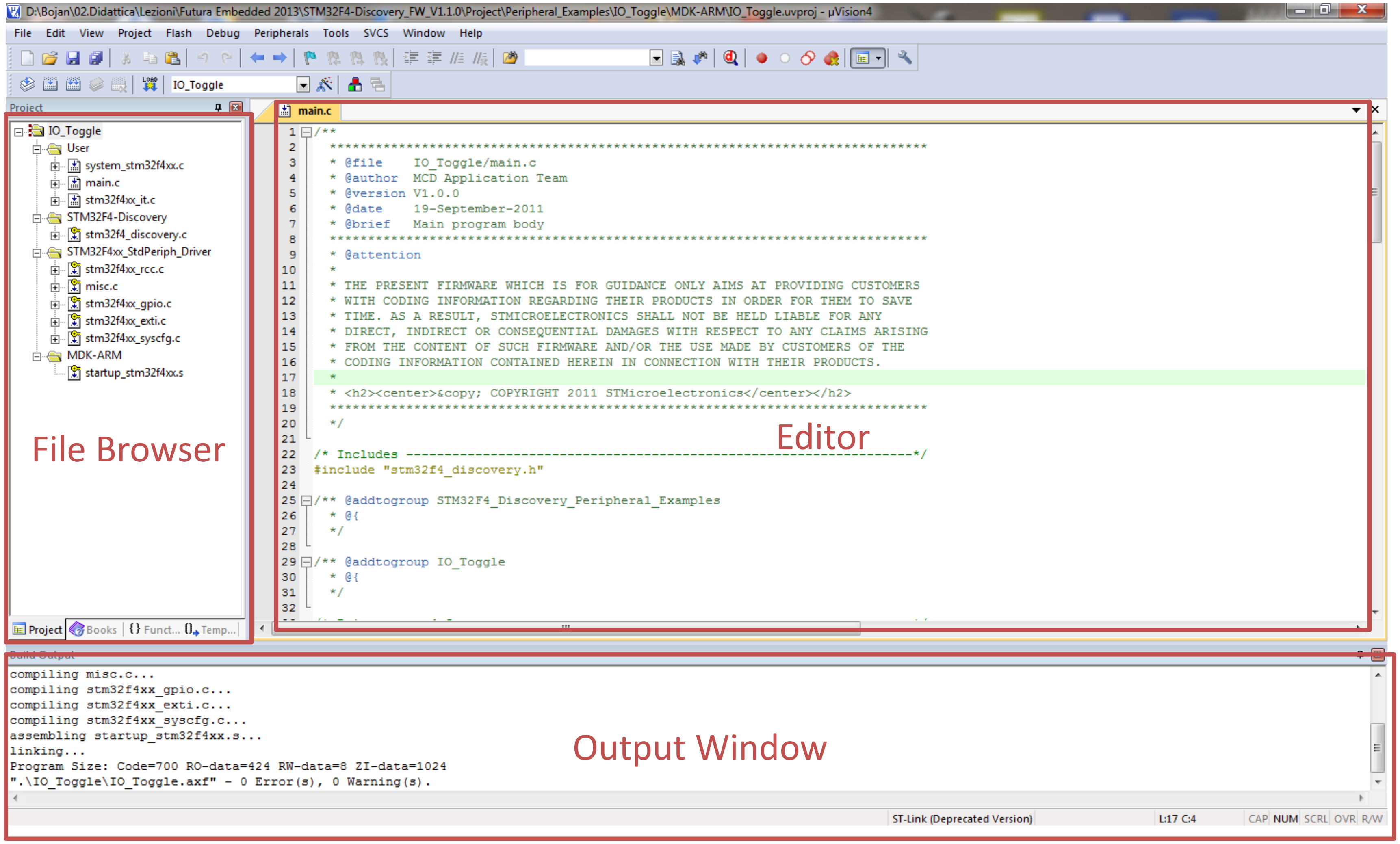
- Integrated Development Environment with MDK-ARM software environment and compiler
- Manual at: http://www.keil.com/product/brochures/uv4.pdf


D:\Bojan\02.Didattica\Lezioni\Futura Embedded 2013\STM32F4-Discovery_FW_V1.1.0\Project\Peripheral_Examples\IO_Toggle\MDK-ARM\IO_Toggle.uvproj - µVision4
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
IO_Toggle
Project
IO_Toggle
User
system_stm32f4xx.c
main.c
stm32f4xx_it.c
STM32F4-Discovery
stm32f4_discovery.c
STM32F4xx_StdPeriph_Driver
stm32f4xx_rcc.c
misc.c
stm32f4xx_gpio.c
stm32f4xx_exti.c
stm32f4xx_syscfg.c
MDK-ARM
startup_stm32f4xx.s
Project Books Funct... Temp...
main.c
Build Output
ST-Link (Deprecated Version)
L:17 C:4
CAP NUM SCRL OVR R/W

```c
/**
  ******************************************************************************
  * @file    IO_Toggle/main.c
  * @author  MCD Application Team
  * @version V1.0.0
  * @date    19-September-2011
  * @brief   Main program body
  ******************************************************************************
  * @attention
  *
  * THE PRESENT FIRMWARE WHICH IS FOR GUIDANCE ONLY AIMS AT PROVIDING CUSTOMERS
  * WITH CODING INFORMATION REGARDING THEIR PRODUCTS IN ORDER FOR THEM TO SAVE
  * TIME. AS A RESULT, STMICROELECTRONICS SHALL NOT BE HELD LIABLE FOR ANY
  * DIRECT, INDIRECT OR CONSEQUENTIAL DAMAGES WITH RESPECT TO ANY CLAIMS ARISING
  * FROM THE CONTENT OF SUCH FIRMWARE AND/OR THE USE MADE BY CUSTOMERS OF THE
  * CODING INFORMATION CONTAINED HEREIN IN CONNECTION WITH THEIR PRODUCTS.
  *
  * <h2><center>&copy; COPYRIGHT 2011 STMicroelectronics</center></h2>
  ******************************************************************************
  */

/* Includes ------------------------------------------------------------------*/
#include "stm32f4_discovery.h"

/** @addtogroup STM32F4_Discovery_Peripheral_Examples
  * @{
  */

/** @addtogroup IO_Toggle
  * @{
  */

```

```
compiling misc.c...
compiling stm32f4xx_gpio.c...
compiling stm32f4xx_exti.c...
compiling stm32f4xx_syscfg.c...
assembling startup_stm32f4xx.s...
linking...
Program Size: Code=700 RO-data=424 RW-data=8 ZI-data=1024
".\IO_Toggle\IO_Toggle.axf" - 0 Error(s), 0 Warning(s).
```

uVision IDE


D:\Bojan\02.Didattica\Lezioni\Futura Embedded 2013\STM32F4-Discovery_FW_V1.1.0\Project\Peripheral_Examples\IO_Toggle\MDK-ARM\IO_Toggle.uvproj - µVision4
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
IO_Toggle
Project
main.c
IO_Toggle
User
system_stm32f4xx.c
main.c
stm32f4xx_it.c
STM32F4-Discovery
stm32f4_discovery.c
STM32F4xx_StdPeriph_Driver
stm32f4xx_rcc.c
misc.c
stm32f4xx_gpio.c
stm32f4xx_exti.c
stm32f4xx_syscfg.c
MDK-ARM
startup_stm32f4xx.s
Project Books {} Funct... 0→ Temp...
File Browser
Editor
Build Output
Output Window
ST-Link (Deprecated Version)
L:17 C:4
CAP NUM SCRL OVR R/W

```c
/**
  ******************************************************************************
  * @file    IO_Toggle/main.c
  * @author  MCD Application Team
  * @version V1.0.0
  * @date    19-September-2011
  * @brief   Main program body
  ******************************************************************************
  * @attention
  *
  * THE PRESENT FIRMWARE WHICH IS FOR GUIDANCE ONLY AIMS AT PROVIDING CUSTOMERS
  * WITH CODING INFORMATION REGARDING THEIR PRODUCTS IN ORDER FOR THEM TO SAVE
  * TIME. AS A RESULT, STMICROELECTRONICS SHALL NOT BE HELD LIABLE FOR ANY
  * DIRECT, INDIRECT OR CONSEQUENTIAL DAMAGES WITH RESPECT TO ANY CLAIMS ARISING
  * FROM THE CONTENT OF SUCH FIRMWARE AND/OR THE USE MADE BY CUSTOMERS OF THE
  * CODING INFORMATION CONTAINED HEREIN IN CONNECTION WITH THEIR PRODUCTS.
  *
  * <h2><center>&copy; COPYRIGHT 2011 STMicroelectronics</center></h2>
  ******************************************************************************
  */

/* Includes ------------------------------------------------------------------*/
#include "stm32f4_discovery.h"

/** @addtogroup STM32F4_Discovery_Peripheral_Examples
  * @{
  */

/** @addtogroup IO_Toggle
  * @{
  */

```

```
compiling misc.c...
compiling stm32f4xx_gpio.c...
compiling stm32f4xx_exti.c...
compiling stm32f4xx_syscfg.c...
assembling startup_stm32f4xx.s...
linking...
Program Size: Code=700 RO-data=424 RW-data=8 ZI-data=1024
".\IO_Toggle\IO_Toggle.axf" - 0 Error(s), 0 Warning(s).
```

uVision IDE – Project Files

Project files can be organized in groups for an easy menagment. This organization is independent of the actual files organization on the disk.

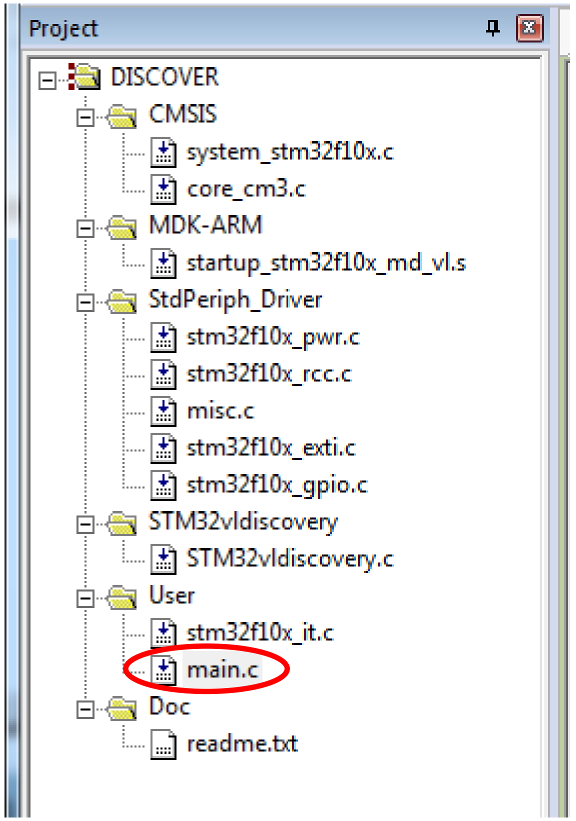
The code is usually organized in subgroups following the abstraction layers and libraries. In this example:

Core library CMSIS and MDK-ARM (startup and system)

ST StdPeriph_Driver library (on-chip devices init and use)

Discovery Library (on-board devices init and use)

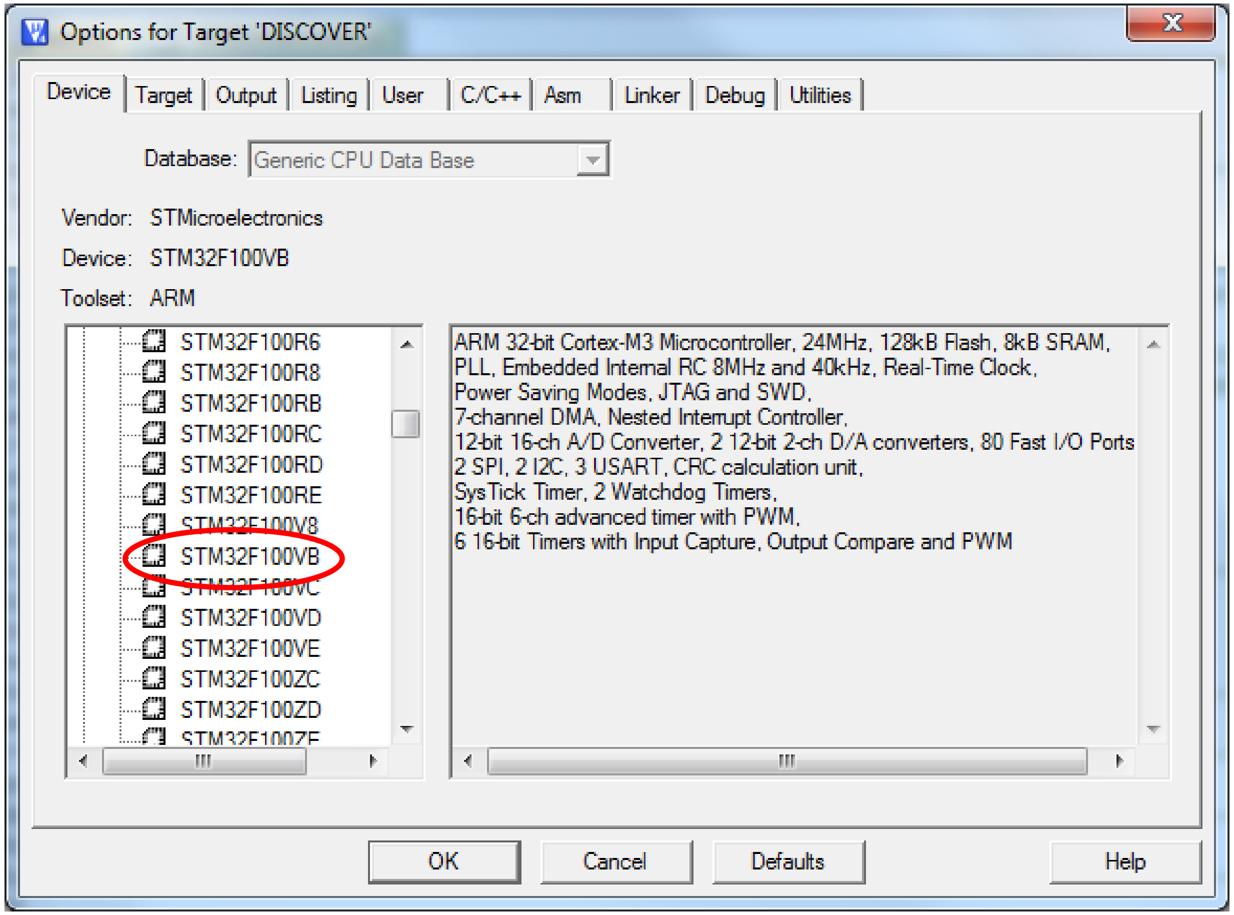
User code (main and all the application code)



```
Project
DISCOVER
  CMSIS
    system_stm32f10x.c
    core_cm3.c
  MDK-ARM
    startup_stm32f10x_md_vl.s
  StdPeriph_Driver
    stm32f10x_pwr.c
    stm32f10x_rcc.c
    misc.c
    stm32f10x_exti.c
    stm32f10x_gpio.c
  STM32vldiscovery
    STM32vldiscovery.c
  User
    stm32f10x_it.c
    main.c
  Doc
    readme.txt
```

Project Options

- Choose the correct target



Options for Target 'DISCOVER'

Database: Generic CPU Data Base

Vendor: STMicroelectronics

Device: STM32F100VB

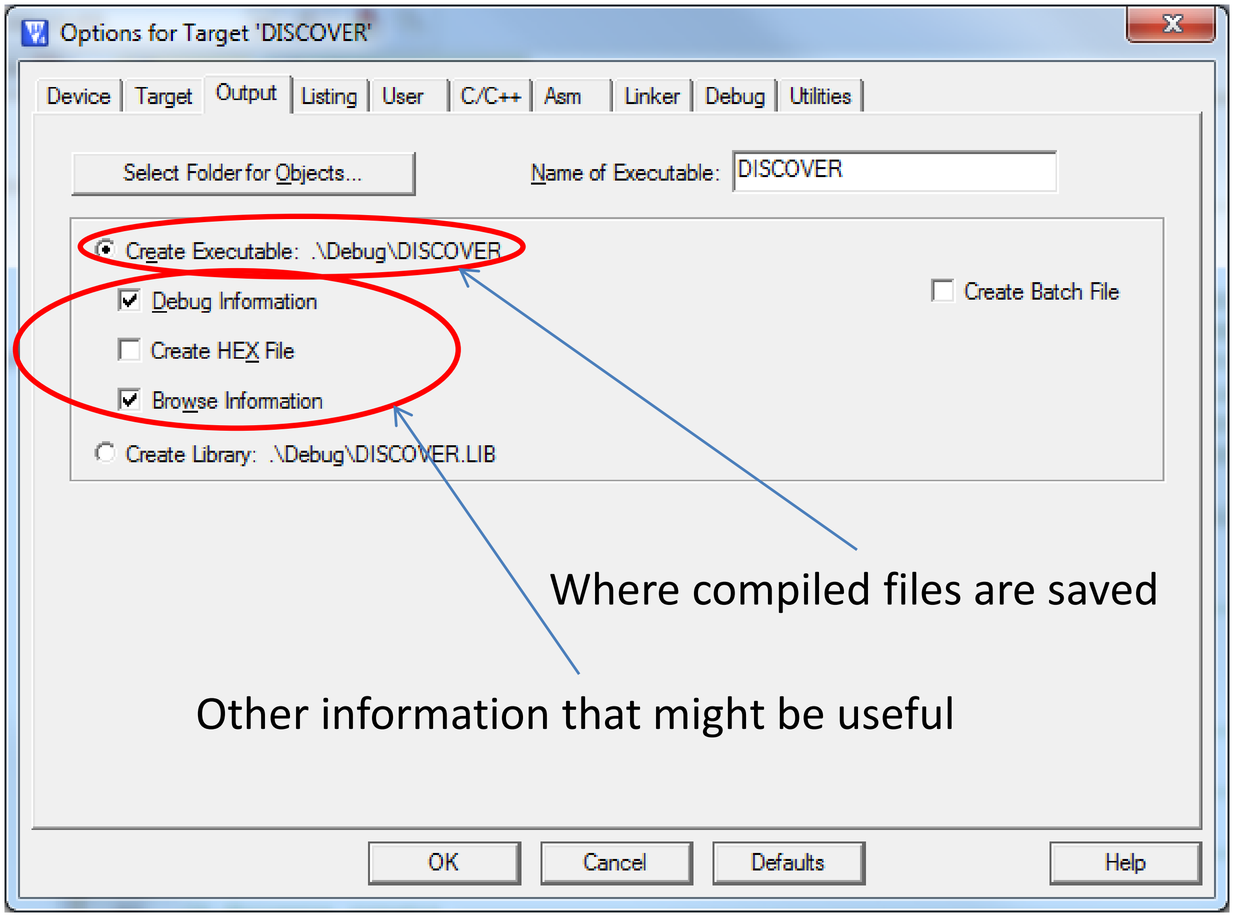
Toolset: ARM

STM32F100R6
STM32F100R8
STM32F100RB
STM32F100RC
STM32F100RD
STM32F100RE
STM32F100V8
[✓ STM32F100VB]
STM32F100VC
STM32F100VD
STM32F100VE
STM32F100ZC
STM32F100ZD
STM32F100ZE

ARM 32-bit Cortex-M3 Microcontroller, 24MHz, 128kB Flash, 8kB SRAM,
PLL, Embedded Internal RC 8MHz and 40kHz, Real-Time Clock,
Power Saving Modes, JTAG and SWD,
7-channel DMA, Nested Interrupt Controller,
12-bit 16-ch A/D Converter, 2 12-bit 2-ch D/A converters, 80 Fast I/O Ports
2 SPI, 2 I2C, 3 USART, CRC calculation unit,
SysTick Timer, 2 Watchdog Timers,
16-bit 6-ch advanced timer with PWM,
6 16-bit Timers with Input Capture, Output Compare and PWM

Project Options

- Choose the compilation output



Options for Target 'DISCOVER'

Device | Target | Output | Listing | User | C/C++ | Asm | Linker | Debug | Utilities

Select Folder for Objects...

Name of Executable: DISCOVER

(●) Create Executable: .\Debug\DISCOVER

[✓] Debug Information

[ ] Create HEX File

[✓] Browse Information

( ) Create Library: .\Debug\DISCOVER.LIB

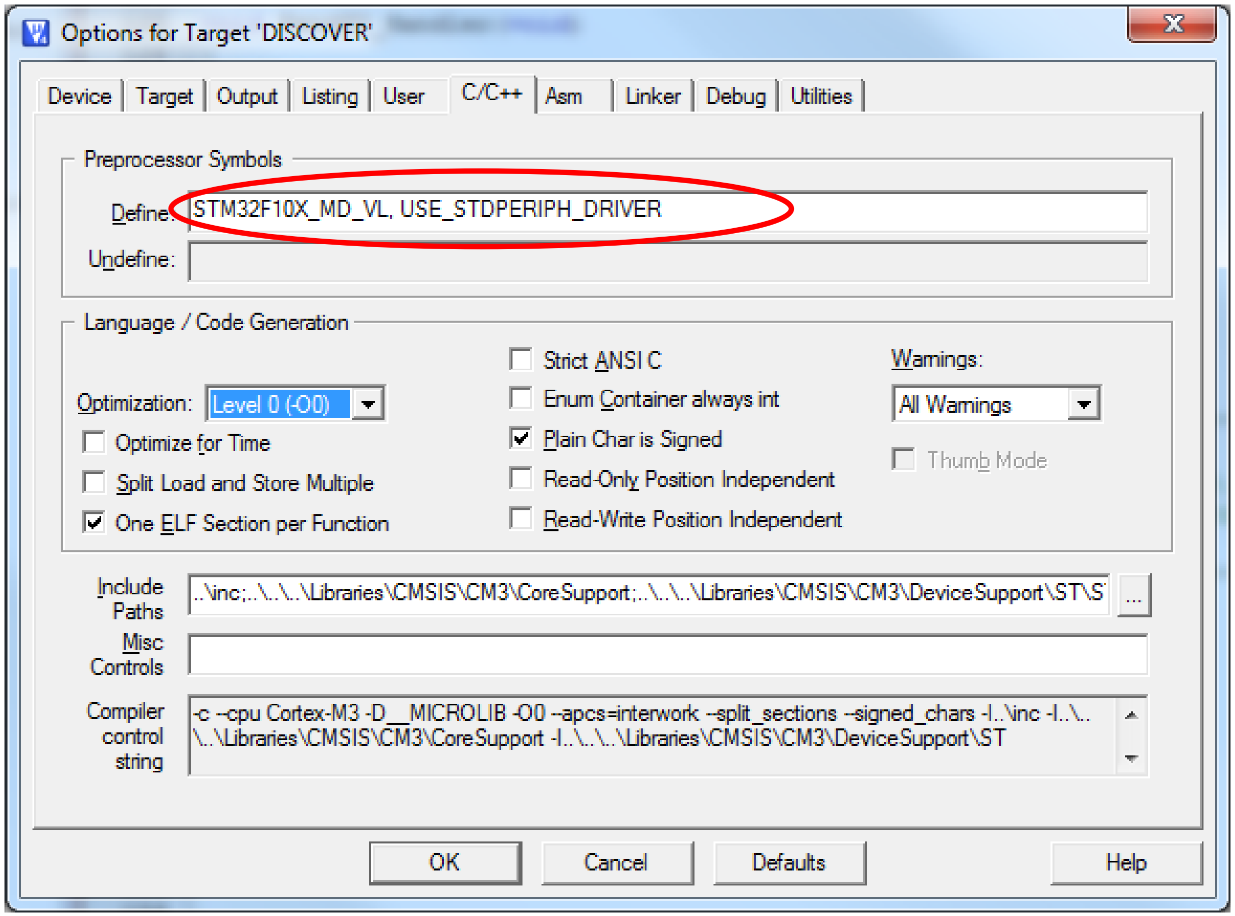
[ ] Create Batch File

Where compiled files are saved

Other information that might be useful

Project Options

- Global Defines



Options for Target 'DISCOVER'

Preprocessor Symbols

Define: STM32F10X_MD_VL, USE_STDPERIPH_DRIVER

Undefine:

Language / Code Generation

Optimization: Level 0 (-O0)

[ ] Optimize for Time
[ ] Split Load and Store Multiple
[✓] One ELF Section per Function
[ ] Strict ANSI C
[ ] Enum Container always int
[✓] Plain Char is Signed
[ ] Read-Only Position Independent
[ ] Read-Write Position Independent

Warnings: All Warnings

[ ] Thumb Mode

Include Paths: ..\inc;..\..\..\Libraries\CMSIS\CM3\CoreSupport;..\..\..\Libraries\CMSIS\CM3\DeviceSupport\ST\S

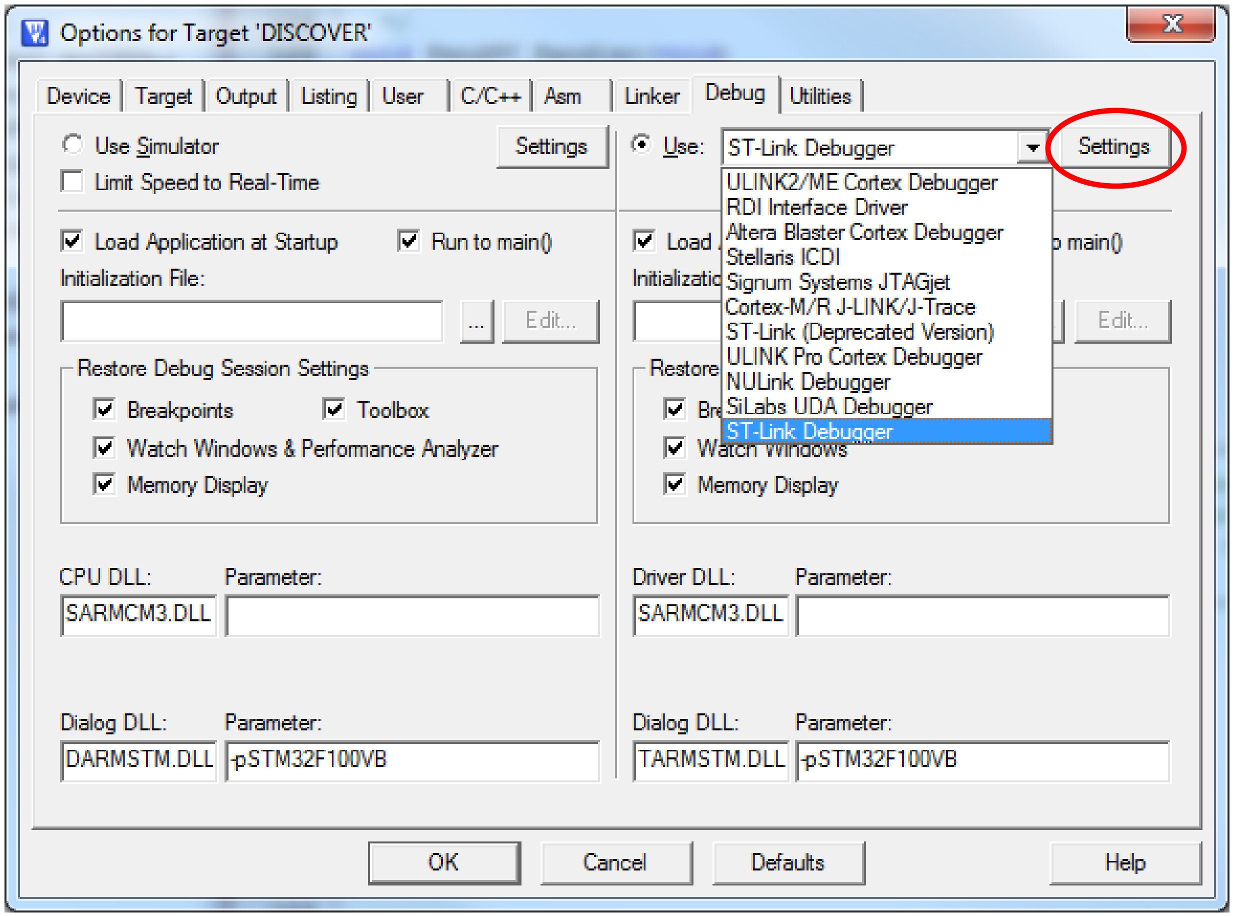
Misc Controls:

Compiler control string:

```
-c --cpu Cortex-M3 -D__MICROLIB -O0 --apcs=interwork --split_sections --signed_chars -I..\inc -I..\..\
..\Libraries\CMSIS\CM3\CoreSupport -I..\..\..\Libraries\CMSIS\CM3\DeviceSupport\ST
```

Project Options

- Debug Settings



Options for Target 'DISCOVER'

Device | Target | Output | Listing | User | C/C++ | Asm | Linker | Debug | Utilities

○ Use Simulator　Settings

☐ Limit Speed to Real-Time

☑ Load Application at Startup　☑ Run to main()

Initialization File:

Restore Debug Session Settings

☑ Breakpoints　☑ Toolbox

☑ Watch Windows & Performance Analyzer

☑ Memory Display

CPU DLL: SARMCM3.DLL　Parameter:

Dialog DLL: DARMSTM.DLL　Parameter: -pSTM32F100VB

● Use: ST-Link Debugger　Settings

- ULINK2/ME Cortex Debugger
- RDI Interface Driver
- Altera Blaster Cortex Debugger
- Stellaris ICDI
- Signum Systems JTAGjet
- Cortex-M/R J-LINK/J-Trace
- ST-Link (Deprecated Version)
- ULINK Pro Cortex Debugger
- NULink Debugger
- SiLabs UDA Debugger
- ST-Link Debugger

☑ Load　☑ Run to main()

☑ Breakpoints

☑ Watch Windows

☑ Memory Display

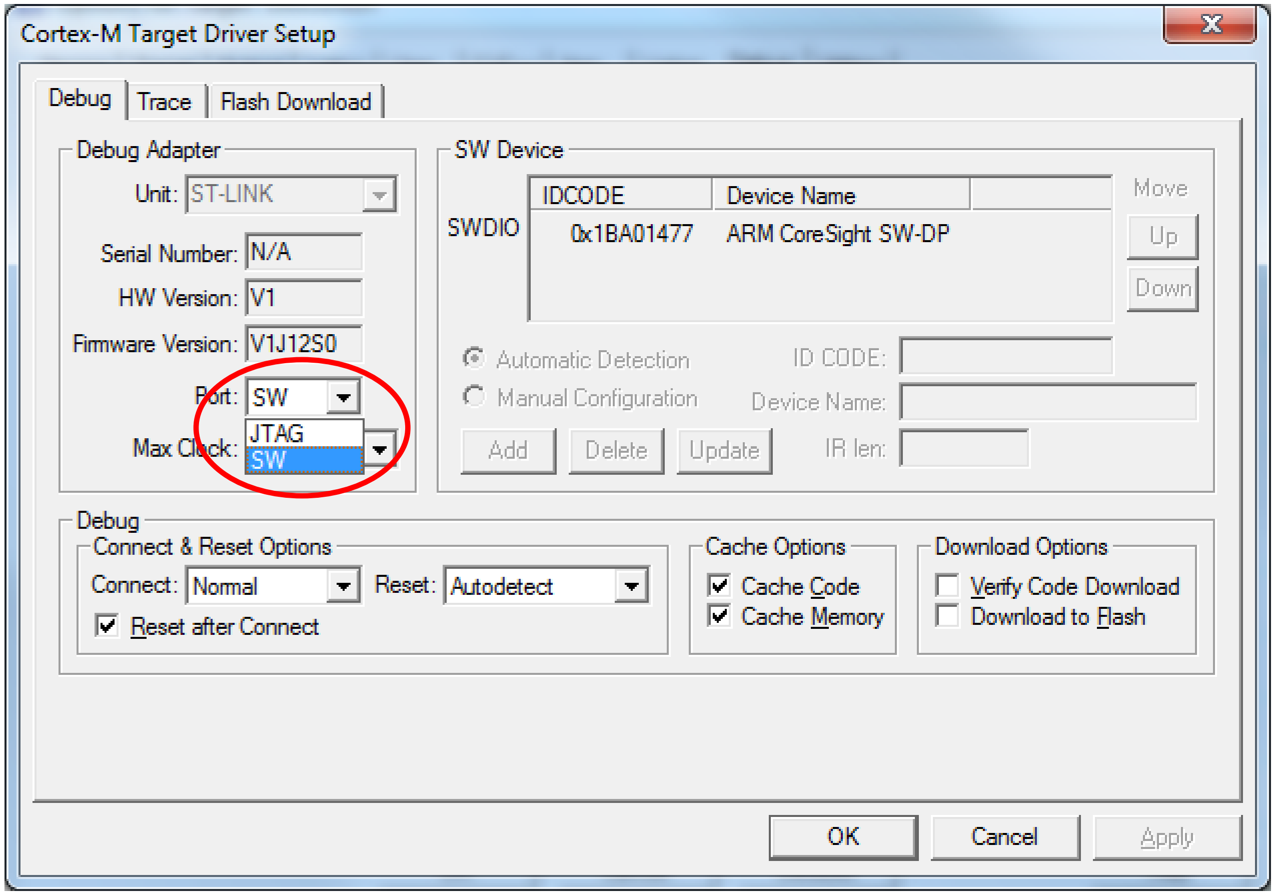
Driver DLL: SARMCM3.DLL　Parameter:

Dialog DLL: TARMSTM.DLL　Parameter: -pSTM32F100VB

OK　Cancel　Defaults　Help

Project Options

- ST-LINK configuration



Cortex-M Target Driver Setup

Debug | Trace | Flash Download

Debug Adapter

Unit: ST-LINK

Serial Number: N/A

HW Version: V1

Firmware Version: V1J12S0

Port: SW

JTAG

SW

Max Clock:

SW Device

	IDCODE	Device Name
SWDIO	0x1BA01477	ARM CoreSight SW-DP

Automatic Detection

Manual Configuration

ID CODE:

Device Name:

IR len:

Debug

Connect & Reset Options

Connect: Normal

Reset: Autodetect

[✓] Reset after Connect

Cache Options

[✓] Cache Code

[✓] Cache Memory

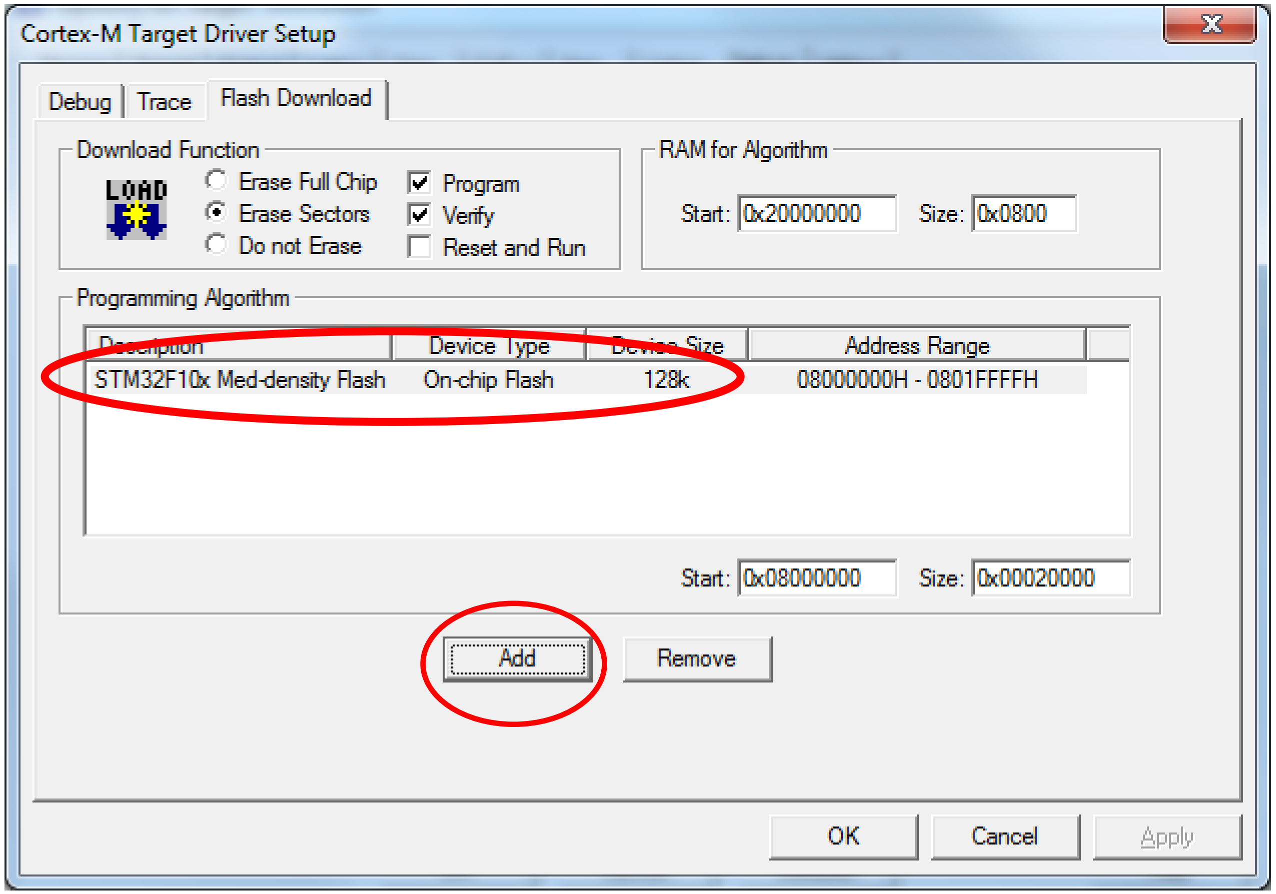
Download Options

[ ] Verify Code Download

[ ] Download to Flash

Project Options

- ST-LINK configuration



Cortex-M Target Driver Setup

Debug | Trace | Flash Download

Download Function

( ) Erase Full Chip
(●) Erase Sectors
( ) Do not Erase

[✓] Program
[✓] Verify
[ ] Reset and Run

RAM for Algorithm

Start: 0x20000000 Size: 0x0800

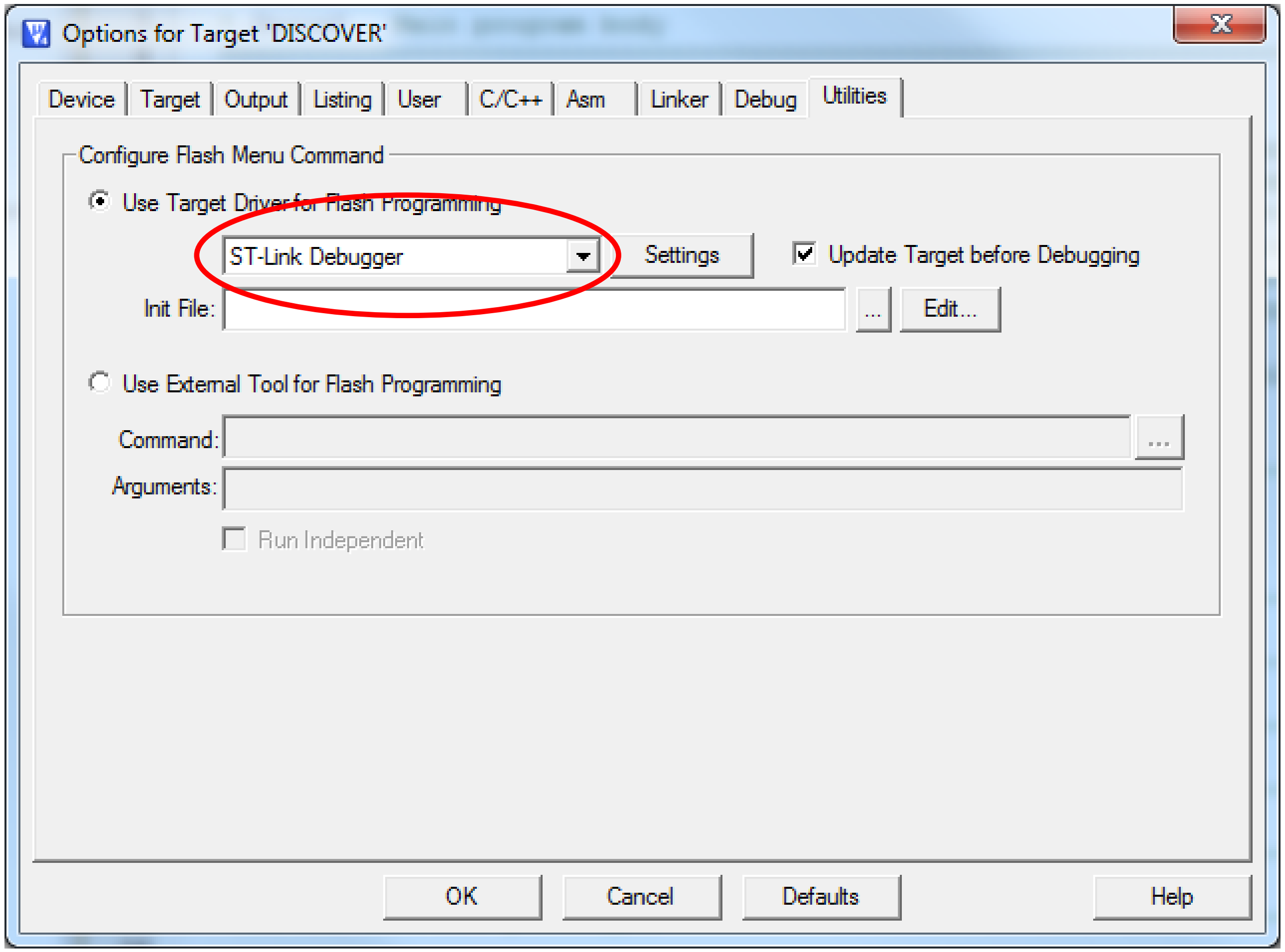
Programming Algorithm

Description	Device Type	Device Size	Address Range
STM32F10x Med-density Flash	On-chip Flash	128k	08000000H - 0801FFFFH

Start: 0x08000000 Size: 0x00020000

Project Options

- Target programming options



Options for Target 'DISCOVER'

Device | Target | Output | Listing | User | C/C++ | Asm | Linker | Debug | Utilities

Configure Flash Menu Command

(●) Use Target Driver for Flash Programming

ST-Link Debugger | Settings | [✓] Update Target before Debugging

Init File: | ... | Edit...

( ) Use External Tool for Flash Programming

Command: | ...

Arguments:

[ ] Run Independent

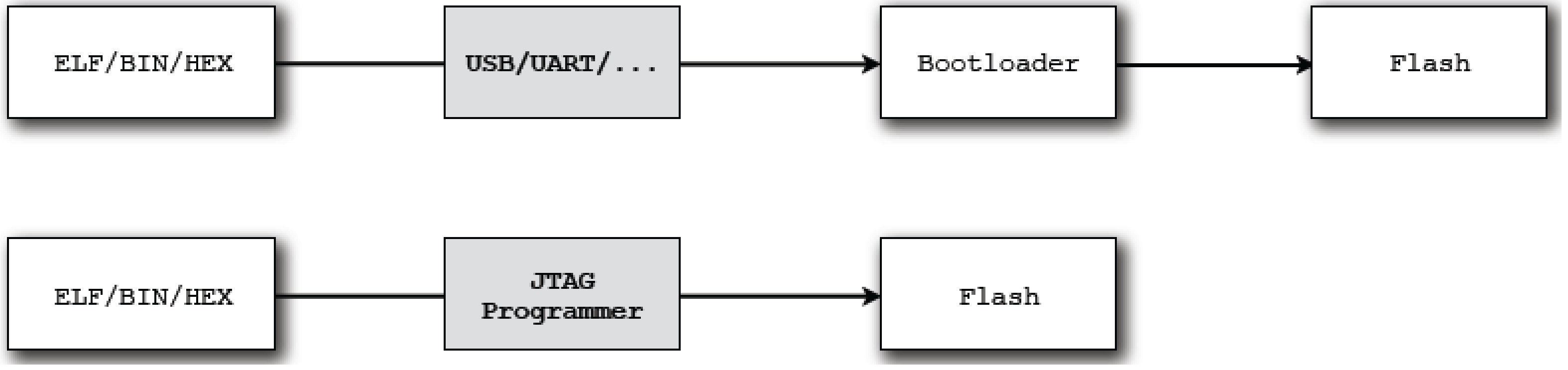
OK | Cancel | Defaults | Help

JTAG

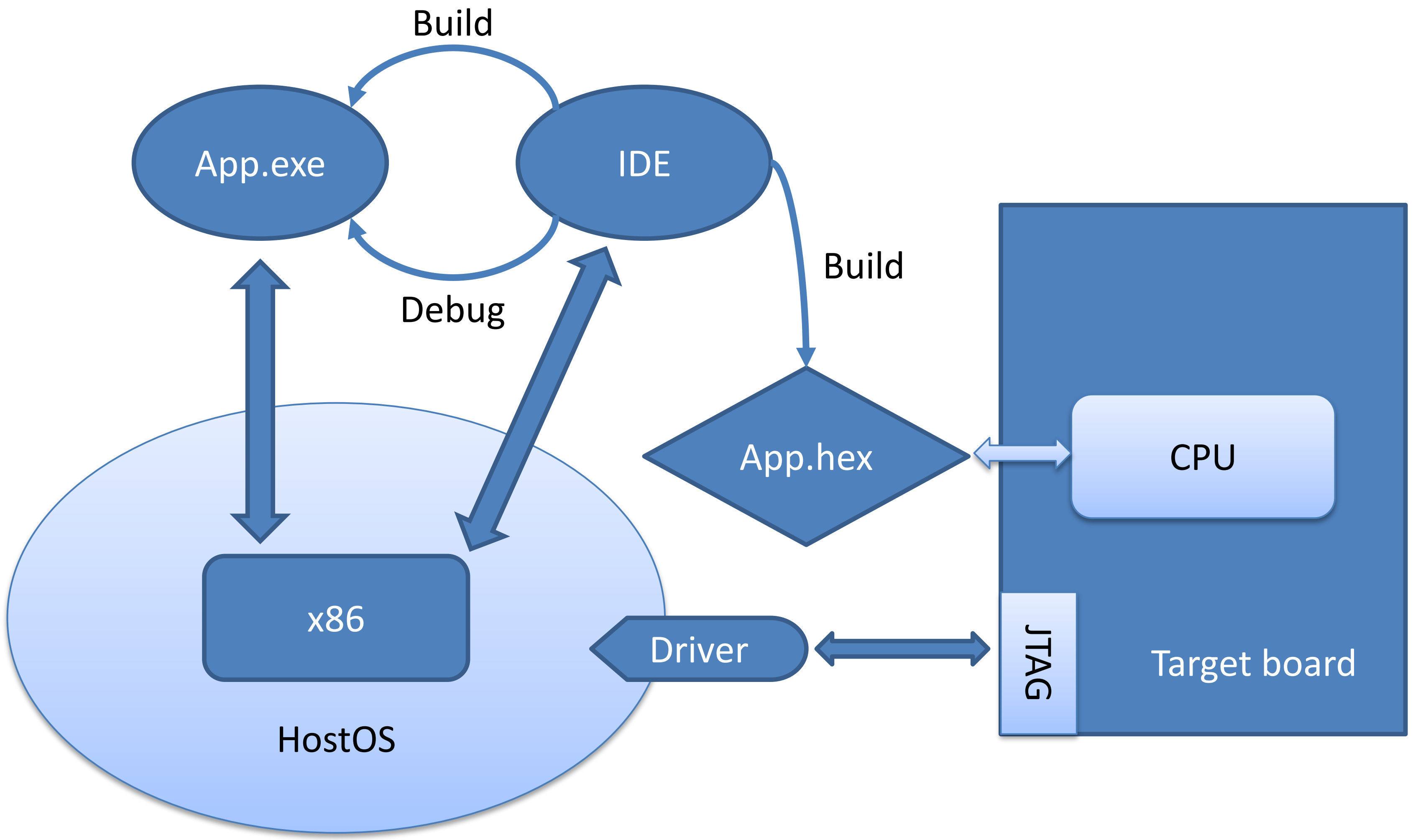
- **Integrated Debug Circuitry / On-Chip Debug:** every chip shipped contains the debug functionality. A serial communication channel is used to connect the debug circuitry to a host debugger
- Besides debugging, another application of JTAG is allowing **device programmer** to transfer data into iternal memory

JTAG for programming

- To program a device we have two alternatives:
  - Using a USB / UART / … connection in **bootloader mode**
  - Using JTAG and programmer **to write flash memory**

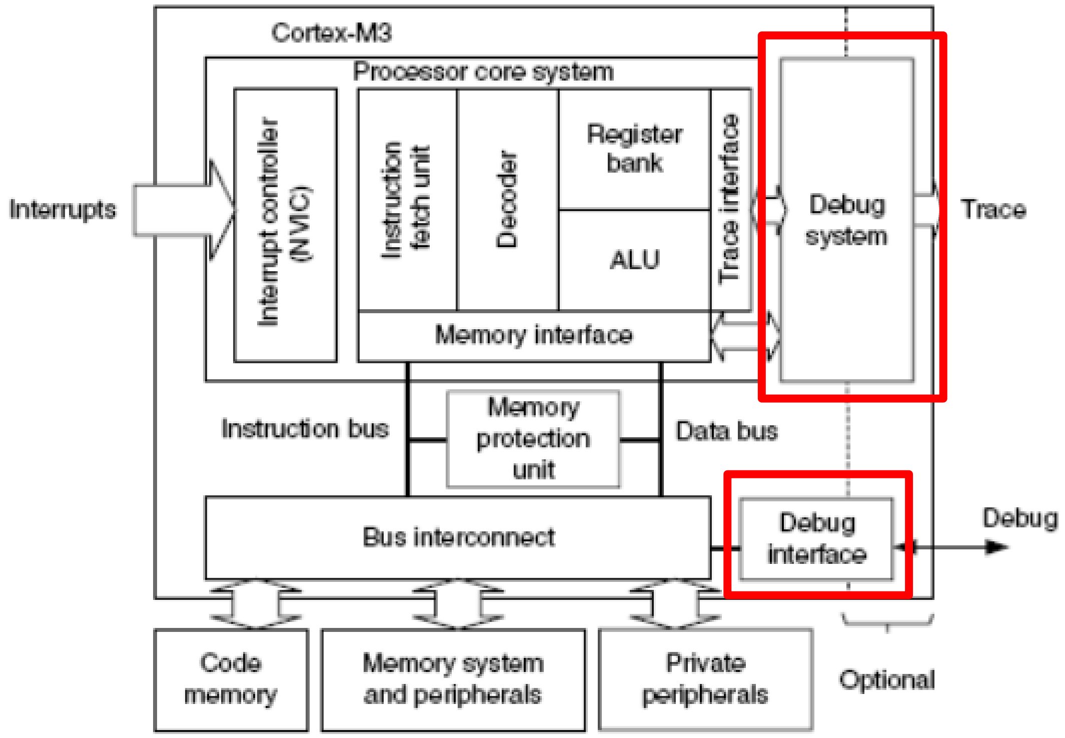


JTAG for debugging

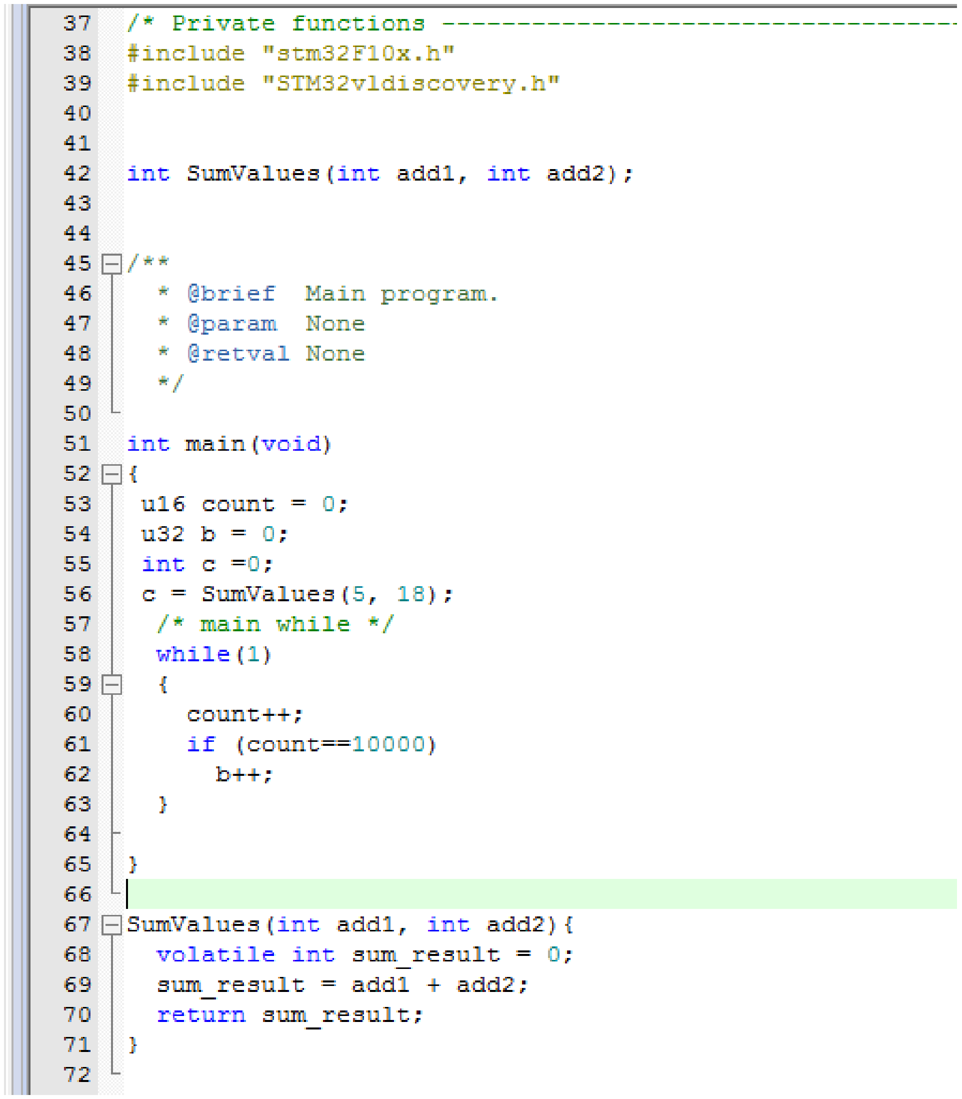


JTAG for debugging

The HW debug support block is **within** the cortex-M3 core. The debug interface has access to the register bank, ALU, memory, etc …



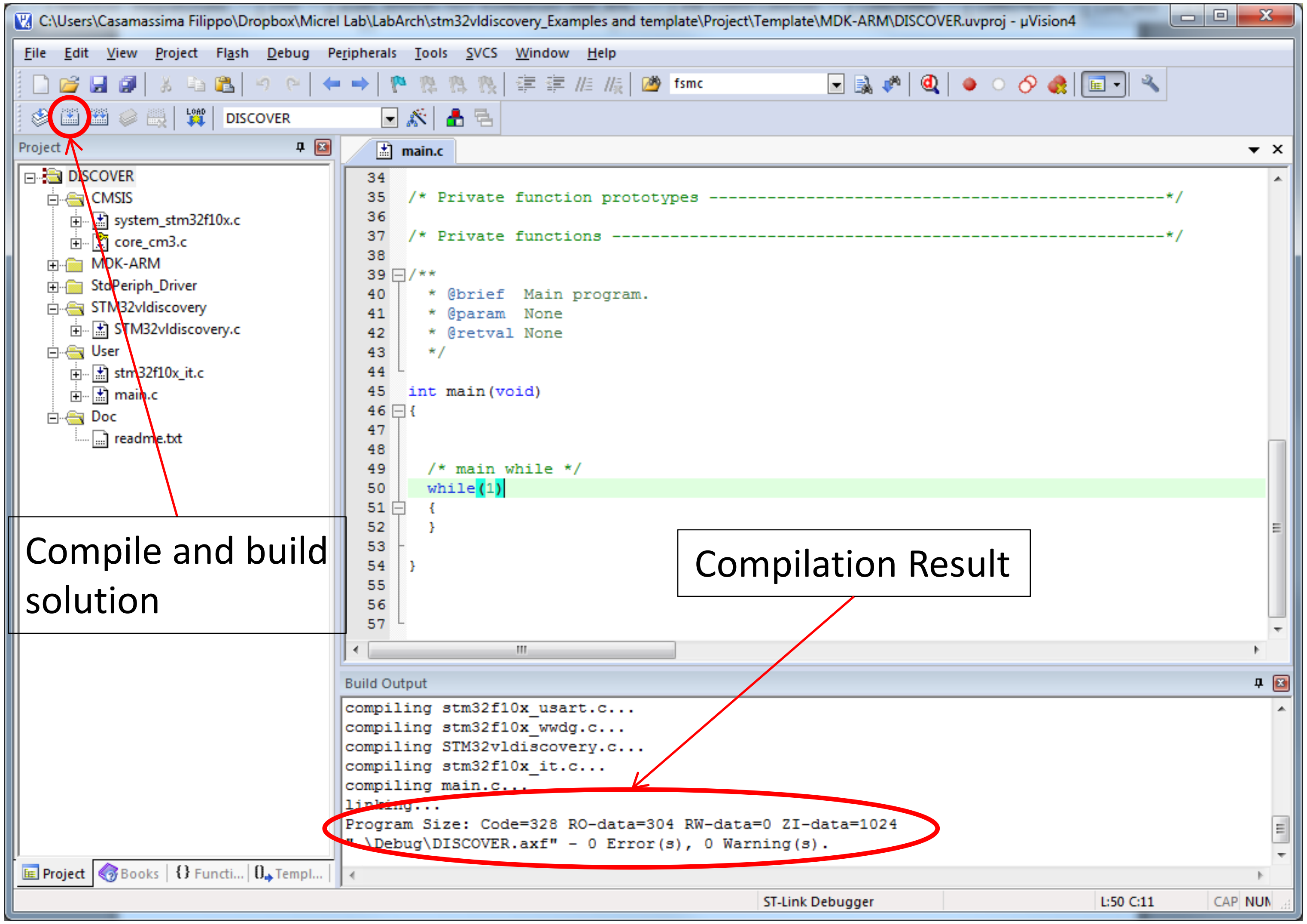
Write Your Code!



```
/* Private functions ----------------------------------
#include "stm32F10x.h"
#include "STM32vldiscovery.h"


int SumValues(int add1, int add2);


/**
  * @brief  Main program.
  * @param  None
  * @retval None
  */

int main(void)
{
 u16 count = 0;
 u32 b = 0;
 int c =0;
 c = SumValues(5, 18);
  /* main while */
  while(1)
  {
    count++;
    if (count==10000)
      b++;
  }

}

SumValues(int add1, int add2){
  volatile int sum_result = 0;
  sum_result = add1 + add2;
  return sum_result;
}

```

- In main.c write a simple function
- no printf, no getch

Code Compilation



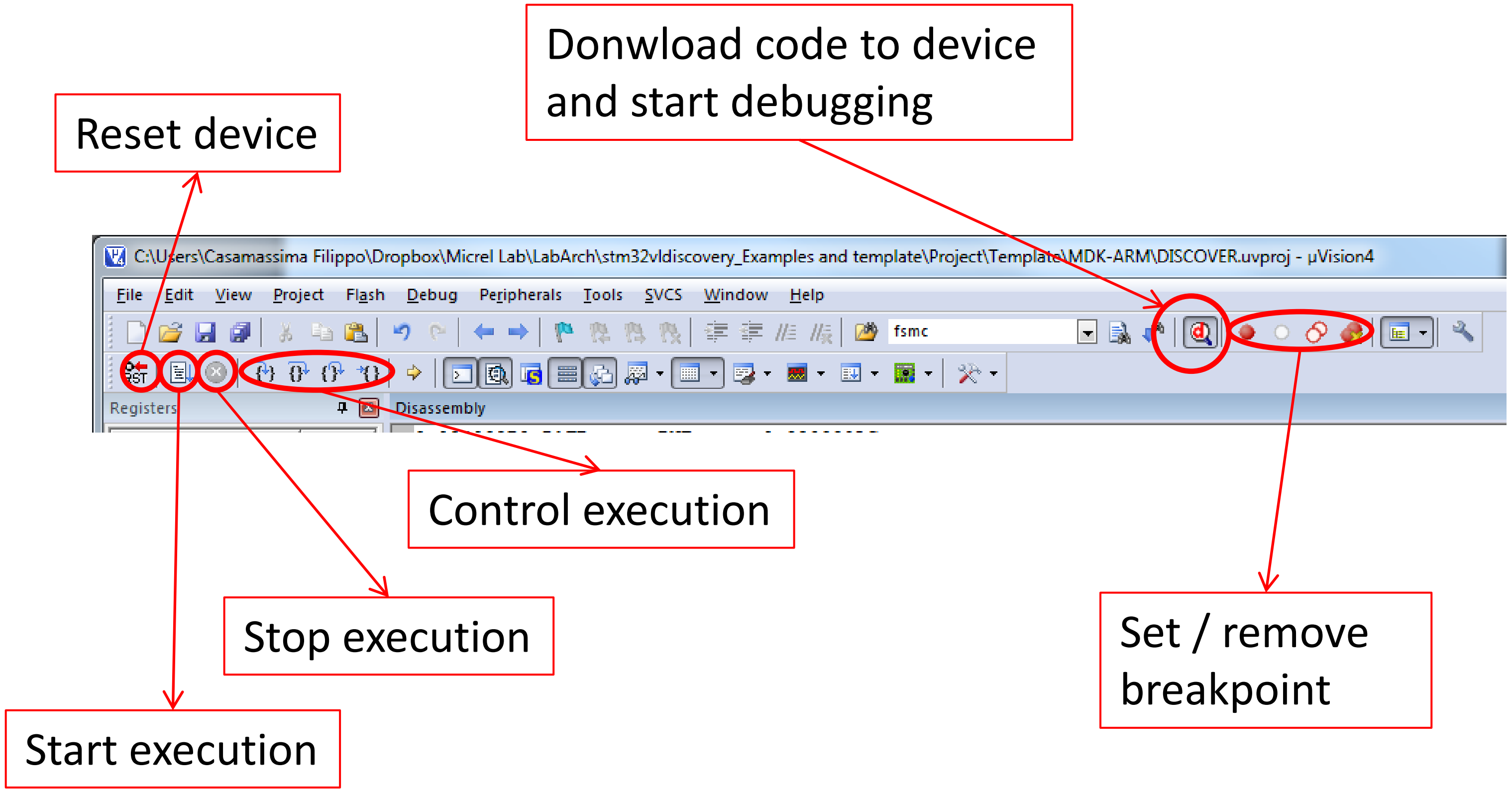
Compile and build solution

Compilation Result

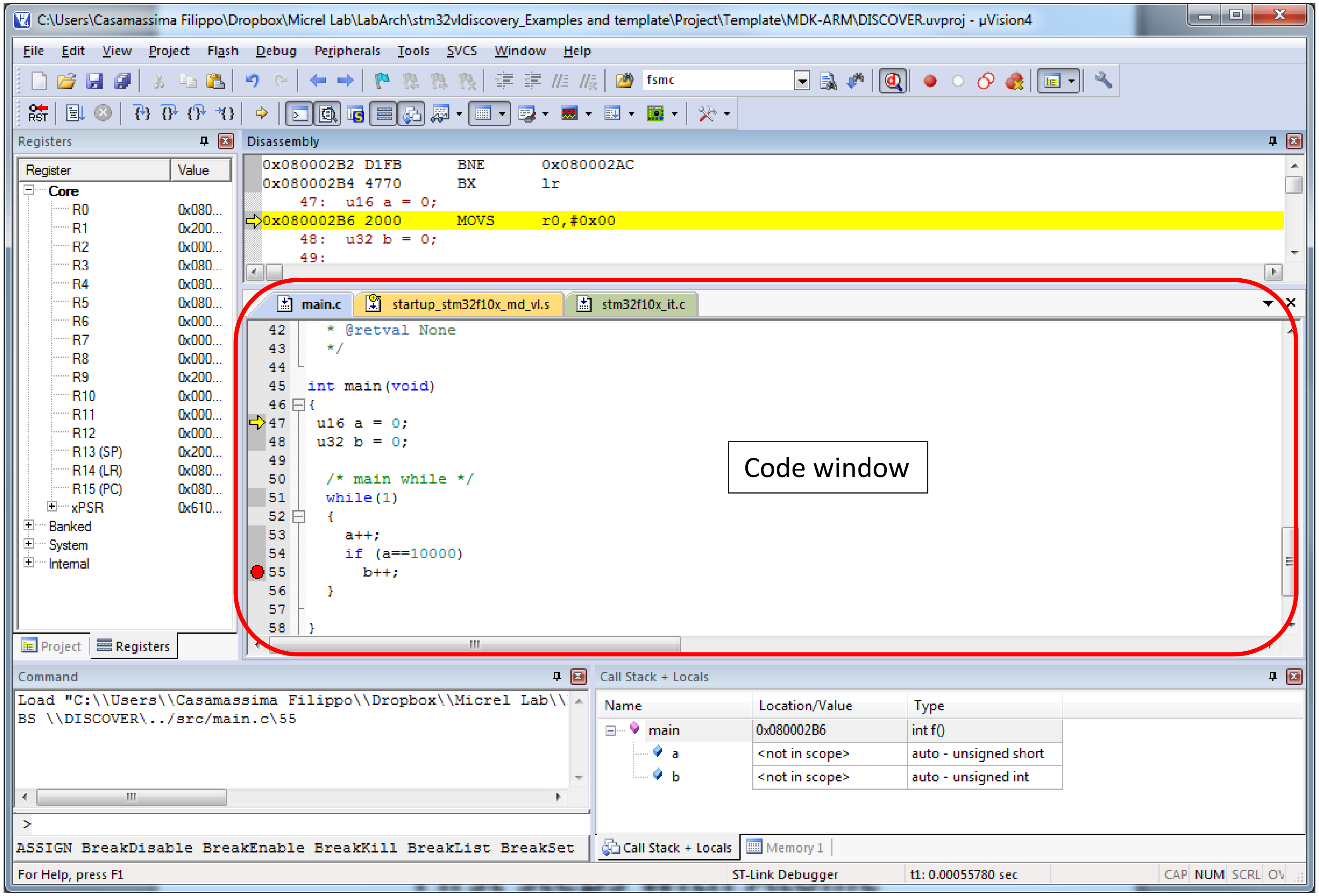
```
/* Private function prototypes -----------------------------------------------*/

/* Private functions ---------------------------------------------------------*/

/**
  * @brief  Main program.
  * @param  None
  * @retval None
  */

int main(void)
{


  /* main while */
  while(1)
  {
  }

}
```

```
compiling stm32f10x_usart.c...
compiling stm32f10x_wwdg.c...
compiling STM32vldiscovery.c...
compiling stm32f10x_it.c...
compiling main.c...
linking...
Program Size: Code=328 RO-data=304 RW-data=0 ZI-data=1024
".\Debug\DISCOVER.axf" - 0 Error(s), 0 Warning(s).
```

Code Debugging


Donwload code to device and start debugging
Reset device
Control execution
Stop execution
Start execution
Set / remove breakpoint
C:\Users\Casamassima Filippo\Dropbox\Micrel Lab\LabArch\stm32vldiscovery_Examples and template\Project\Template\MDK-ARM\DISCOVER.uvproj - µVision4
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
fsmc
Registers
Disassembly

Code Debugging



C:\Users\Casamassima Filippo\Dropbox\Micrel Lab\LabArch\stm32vldiscovery_Examples and template\Project\Template\MDK-ARM\DISCOVER.uvproj - µVision4
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
fsmc
Registers
Disassembly
main.c
startup_stm32f10x_md_vl.s
stm32f10x_it.c
Code window
Project
Registers
Command
Call Stack + Locals
Memory 1
For Help, press F1
ST-Link Debugger
t1: 0.00055780 sec
CAP NUM SCRL OVR

Register	Value
Core	
R0	0x080...
R1	0x200...
R2	0x000...
R3	0x080...
R4	0x080...
R5	0x080...
R6	0x000...
R7	0x000...
R8	0x000...
R9	0x200...
R10	0x000...
R11	0x000...
R12	0x000...
R13 (SP)	0x200...
R14 (LR)	0x080...
R15 (PC)	0x080...
xPSR	0x610...
Banked	
System	
Internal	

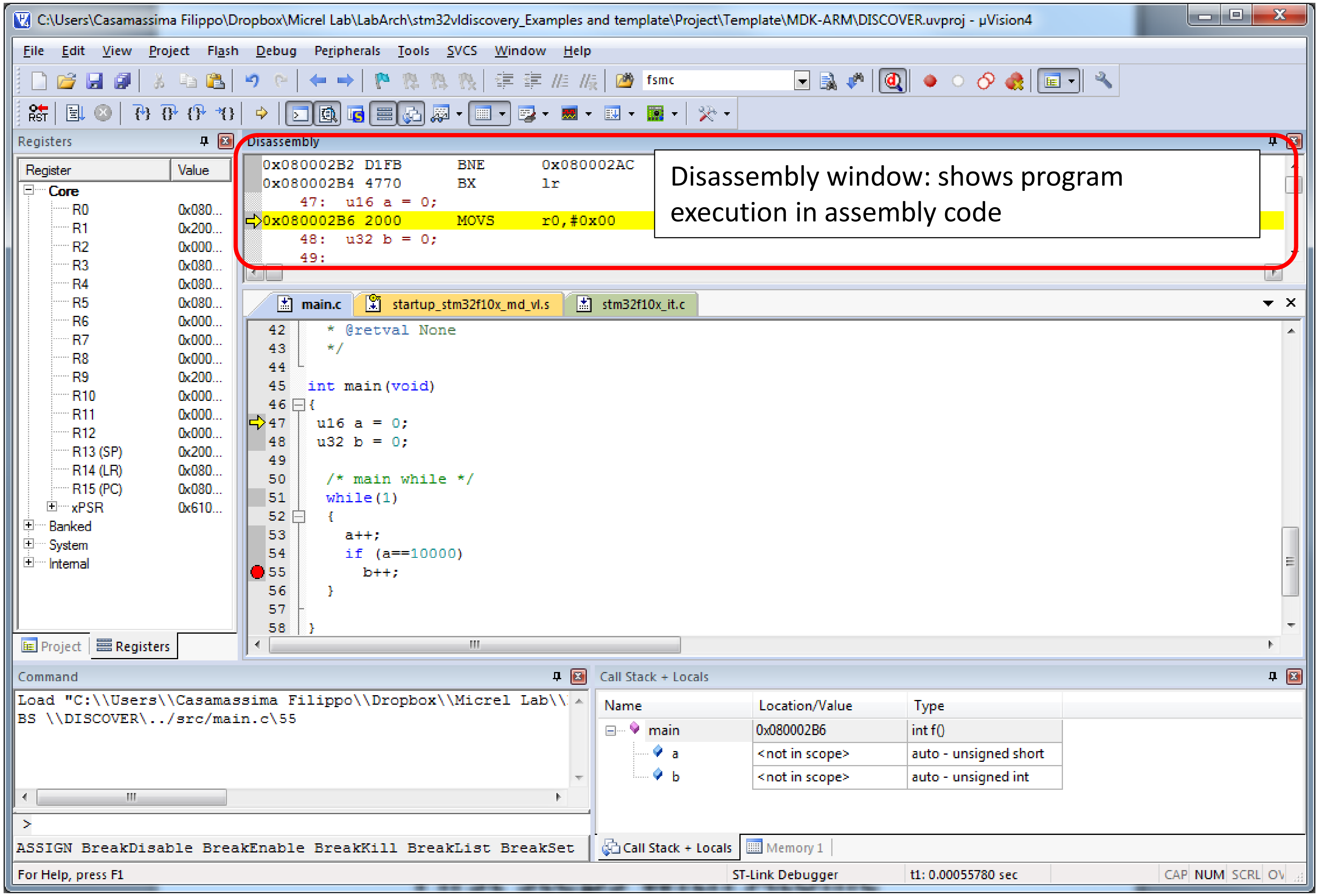
```
0x080002B2 D1FB      BNE      0x080002AC
0x080002B4 4770      BX       lr
    47:   u16 a = 0;
0x080002B6 2000      MOVS     r0,#0x00
    48:   u32 b = 0;
    49:
```

```
42     * @retval None
43     */
44
45   int main(void)
46   {
47    u16 a = 0;
48    u32 b = 0;
49
50     /* main while */
51     while(1)
52     {
53       a++;
54       if (a==10000)
55         b++;
56     }
57
58   }
```

```
Load "C:\\Users\\Casamassima Filippo\\Dropbox\\Micrel Lab\\
BS \\DISCOVER\../src/main.c\55

>
ASSIGN BreakDisable BreakEnable BreakKill BreakList BreakSet
```

Name	Location/Value	Type
main	0x080002B6	int f()
a	<not in scope>	auto - unsigned short
b	<not in scope>	auto - unsigned int

Code Debugging



C:\Users\Casamassima Filippo\Dropbox\Micrel Lab\LabArch\stm32vldiscovery_Examples and template\Project\Template\MDK-ARM\DISCOVER.uvproj - µVision4

File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
fsmc
Registers
Disassembly
main.c
startup_stm32f10x_md_vl.s
stm32f10x_it.c
Project
Registers
Command
Call Stack + Locals
Memory 1

Disassembly window: shows program execution in assembly code

```
0x080002B2 D1FB      BNE      0x080002AC
0x080002B4 4770      BX       lr
    47:   u16 a = 0;
0x080002B6 2000      MOVS     r0,#0x00
    48:   u32 b = 0;
    49:
```

Register	Value
Core	
R0	0x080...
R1	0x200...
R2	0x000...
R3	0x080...
R4	0x080...
R5	0x080...
R6	0x000...
R7	0x000...
R8	0x000...
R9	0x200...
R10	0x000...
R11	0x000...
R12	0x000...
R13 (SP)	0x200...
R14 (LR)	0x080...
R15 (PC)	0x080...
xPSR	0x610...
Banked	
System	
Internal	

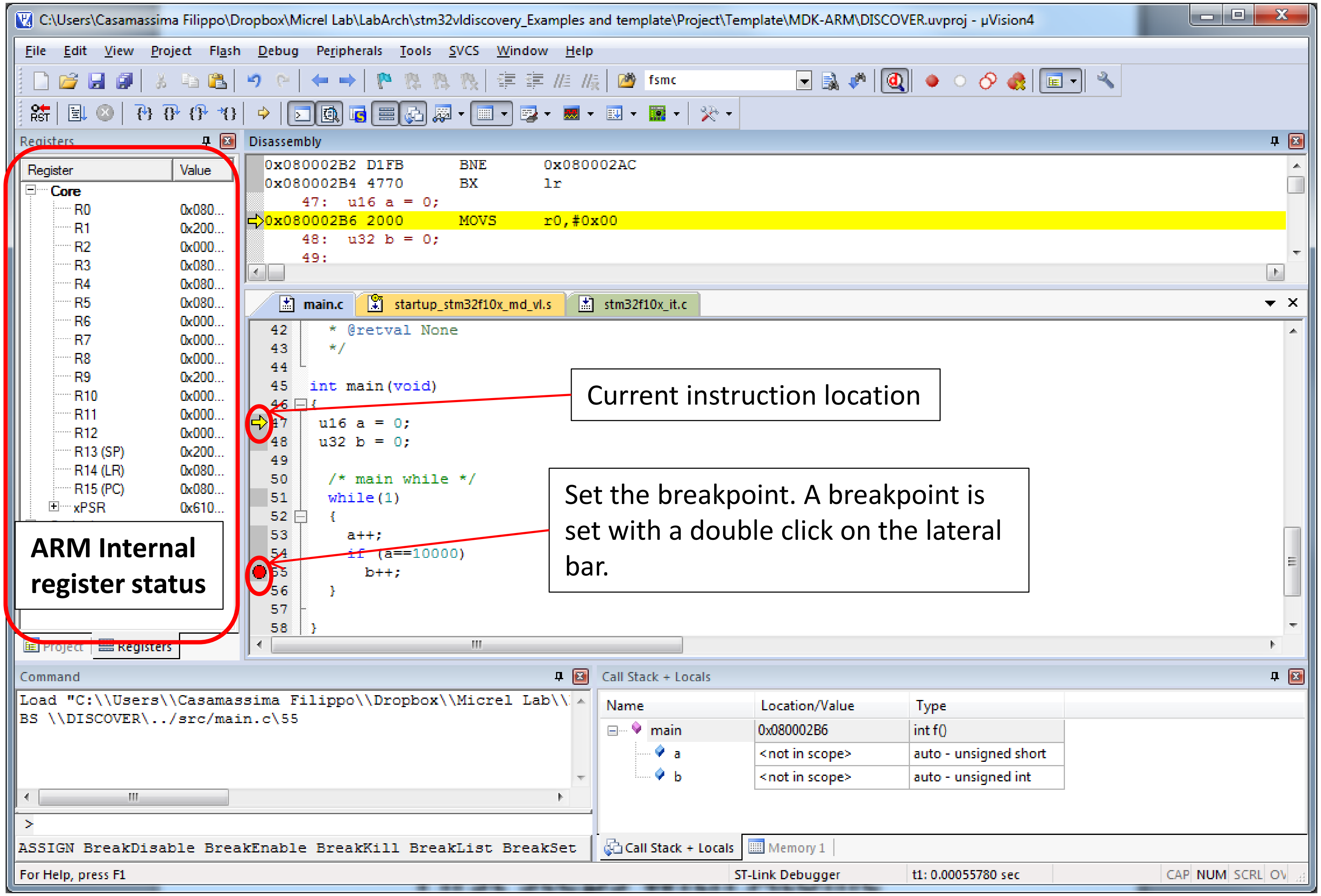
```
  * @retval None
  */

int main(void)
{
 u16 a = 0;
 u32 b = 0;

  /* main while */
  while(1)
  {
    a++;
    if (a==10000)
      b++;
  }

}
```

```
Load "C:\\Users\\Casamassima Filippo\\Dropbox\\Micrel Lab\\
BS \\DISCOVER\../src/main.c\55

>
ASSIGN BreakDisable BreakEnable BreakKill BreakList BreakSet
```

Name	Location/Value	Type
main	0x080002B6	int f()
a	<not in scope>	auto - unsigned short
b	<not in scope>	auto - unsigned int

For Help, press F1 ST-Link Debugger t1: 0.00055780 sec CAP NUM SCRL OVR

Code Debugging



C:\Users\Casamassima Filippo\Dropbox\Micrel Lab\LabArch\stm32vldiscovery_Examples and template\Project\Template\MDK-ARM\DISCOVER.uvproj - µVision4

File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
fsmc
Registers
Disassembly
Project
Registers
main.c
startup_stm32f10x_md_vl.s
stm32f10x_it.c
Command
Call Stack + Locals
Memory 1

Register	Value
Core	
R0	0x080...
R1	0x200...
R2	0x000...
R3	0x080...
R4	0x080...
R5	0x080...
R6	0x000...
R7	0x000...
R8	0x000...
R9	0x200...
R10	0x000...
R11	0x000...
R12	0x000...
R13 (SP)	0x200...
R14 (LR)	0x080...
R15 (PC)	0x080...
xPSR	0x610...

```
0x080002B2 D1FB      BNE      0x080002AC
0x080002B4 4770      BX       lr
    47:   u16 a = 0;
0x080002B6 2000      MOVS     r0,#0x00
    48:   u32 b = 0;
    49:
```

```
   * @retval None
   */

int main(void)
{
  u16 a = 0;
  u32 b = 0;

   /* main while */
   while(1)
   {
     a++;
     if (a==10000)
       b++;
   }

}
```

Current instruction location

Set the breakpoint. A breakpoint is set with a double click on the lateral bar.

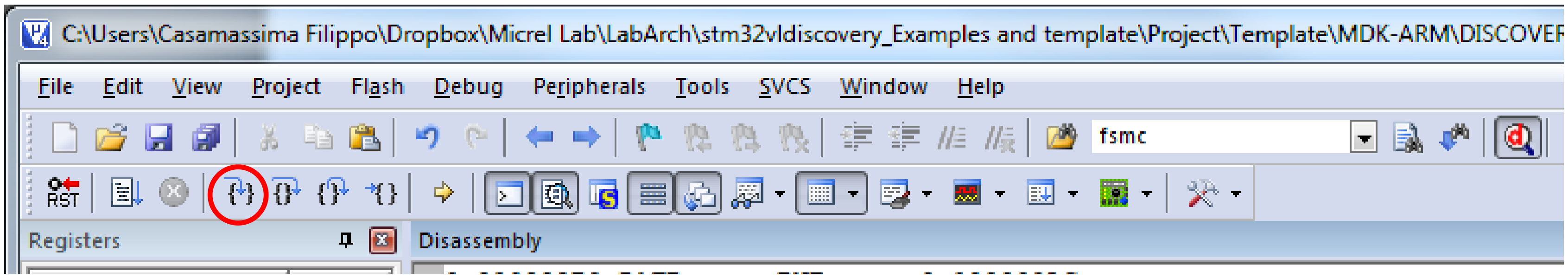
ARM Internal register status

```
Load "C:\\Users\\Casamassima Filippo\\Dropbox\\Micrel Lab\\
BS \\DISCOVER\\../src/main.c\55

>
ASSIGN BreakDisable BreakEnable BreakKill BreakList BreakSet
```

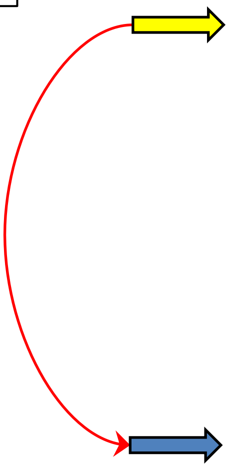
Name	Location/Value	Type
main	0x080002B6	int f()
a	<not in scope>	auto - unsigned short
b	<not in scope>	auto - unsigned int

For Help, press F1   ST-Link Debugger   t1: 0.00055780 sec   CAP NUM SCRL OV

Code Debugging

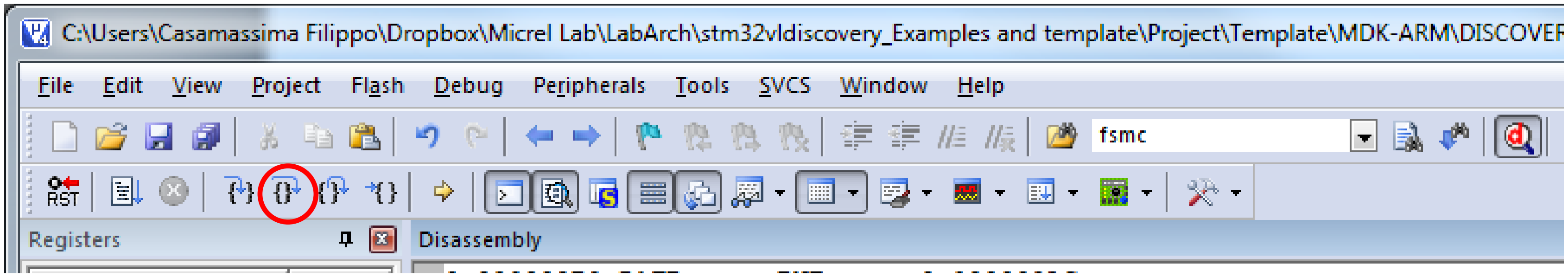

C:\Users\Casamassima Filippo\Dropbox\Micrel Lab\LabArch\stm32vldiscovery_Examples and template\Project\Template\MDK-ARM\DISCOVER
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
fsmc
RST
Registers
Disassembly

Step: executes next instruction

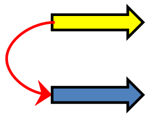


```
int main(void)
{
 u16 count = 0;
 u32 b = 0;
 int c =0;
 c = SumValues(5, 18);
  /* main while */
  while(1)
  {
      count++;
      if (count==10000)
      b++;
  }

}

SumValues(int add1, int add2){
           volatile int sum_result = 0;
           sum_result = add1 + add2;
           return sum_result;
}
```

Code Debugging

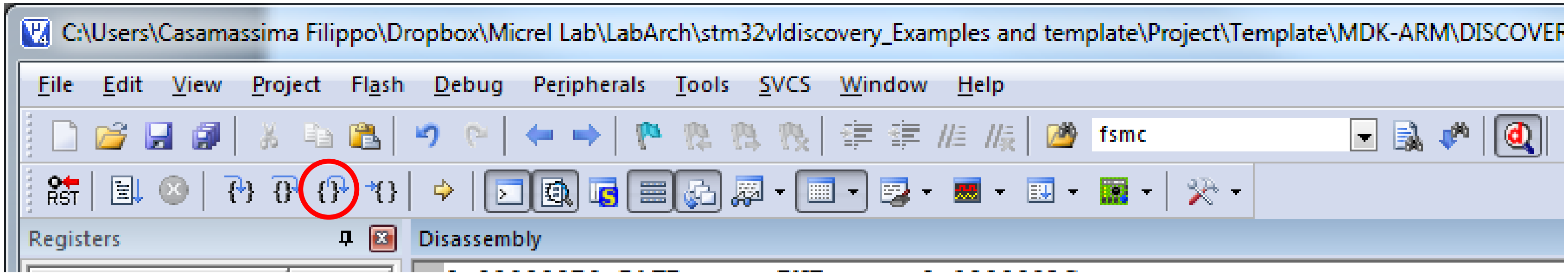

C:\Users\Casamassima Filippo\Dropbox\Micrel Lab\LabArch\stm32vldiscovery_Examples and template\Project\Template\MDK-ARM\DISCOVEF

Step Over: executes next line

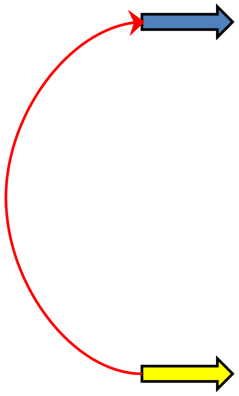


```
int main(void)
{
 u16 count = 0;
 u32 b = 0;
 int c =0;
 c = SumValues(5, 18);
  /* main while */
  while(1)
  {
      count++;
      if (count==10000)
      b++;
  }

}

SumValues(int add1, int add2){
           volatile int sum_result = 0;
           sum_result = add1 + add2;
           return sum_result;
}
```

Code Debugging

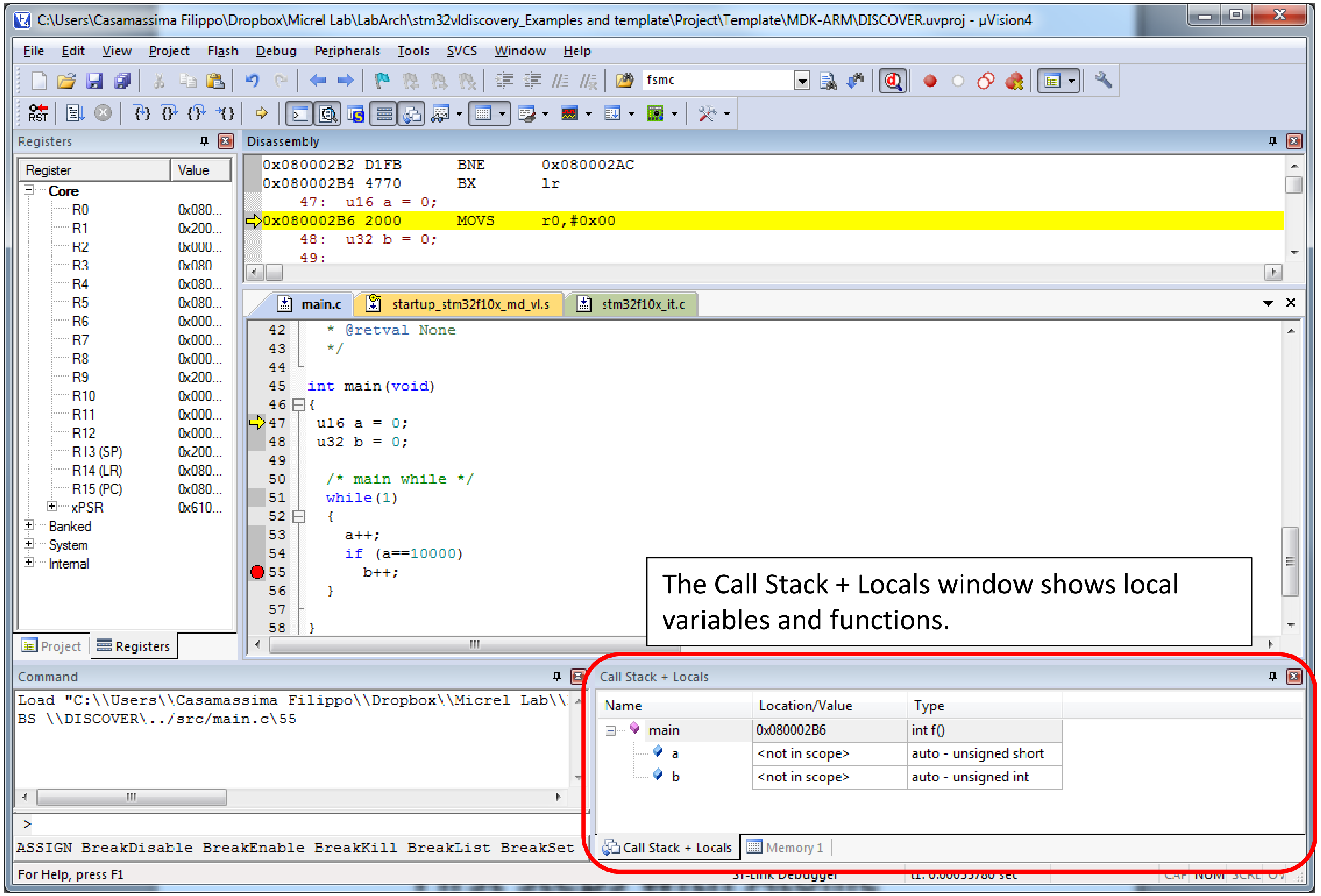

C:\Users\Casamassima Filippo\Dropbox\Micrel Lab\LabArch\stm32vldiscovery_Examples and template\Project\Template\MDK-ARM\DISCOVEF

Step Out: Exit from the current function



```
int main(void)
{
 u16 count = 0;
 u32 b = 0;
 int c =0;
 c = SumValues(5, 18);
  /* main while */
  while(1)
  {
      count++;
      if (count==10000)
      b++;
  }

}

SumValues(int add1, int add2){
           volatile int sum_result = 0;
           sum_result = add1 + add2;
           return sum_result;
}
```

Code Debugging



C:\Users\Casamassima Filippo\Dropbox\Micrel Lab\LabArch\stm32vldiscovery_Examples and template\Project\Template\MDK-ARM\DISCOVER.uvproj - µVision4

File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
fsmc
Registers
Disassembly

Register	Value
Core	
R0	0x080...
R1	0x200...
R2	0x000...
R3	0x080...
R4	0x080...
R5	0x080...
R6	0x000...
R7	0x000...
R8	0x000...
R9	0x200...
R10	0x000...
R11	0x000...
R12	0x000...
R13 (SP)	0x200...
R14 (LR)	0x080...
R15 (PC)	0x080...
xPSR	0x610...
Banked	
System	
Internal	

```
0x080002B2 D1FB      BNE      0x080002AC
0x080002B4 4770      BX       lr
    47:  u16 a = 0;
0x080002B6 2000      MOVS     r0,#0x00
    48:  u32 b = 0;
    49:
```

```
42    * @retval None
43    */
44
45  int main(void)
46  {
47   u16 a = 0;
48   u32 b = 0;
49
50    /* main while */
51    while(1)
52    {
53      a++;
54      if (a==10000)
55        b++;
56    }
57
58  }
```

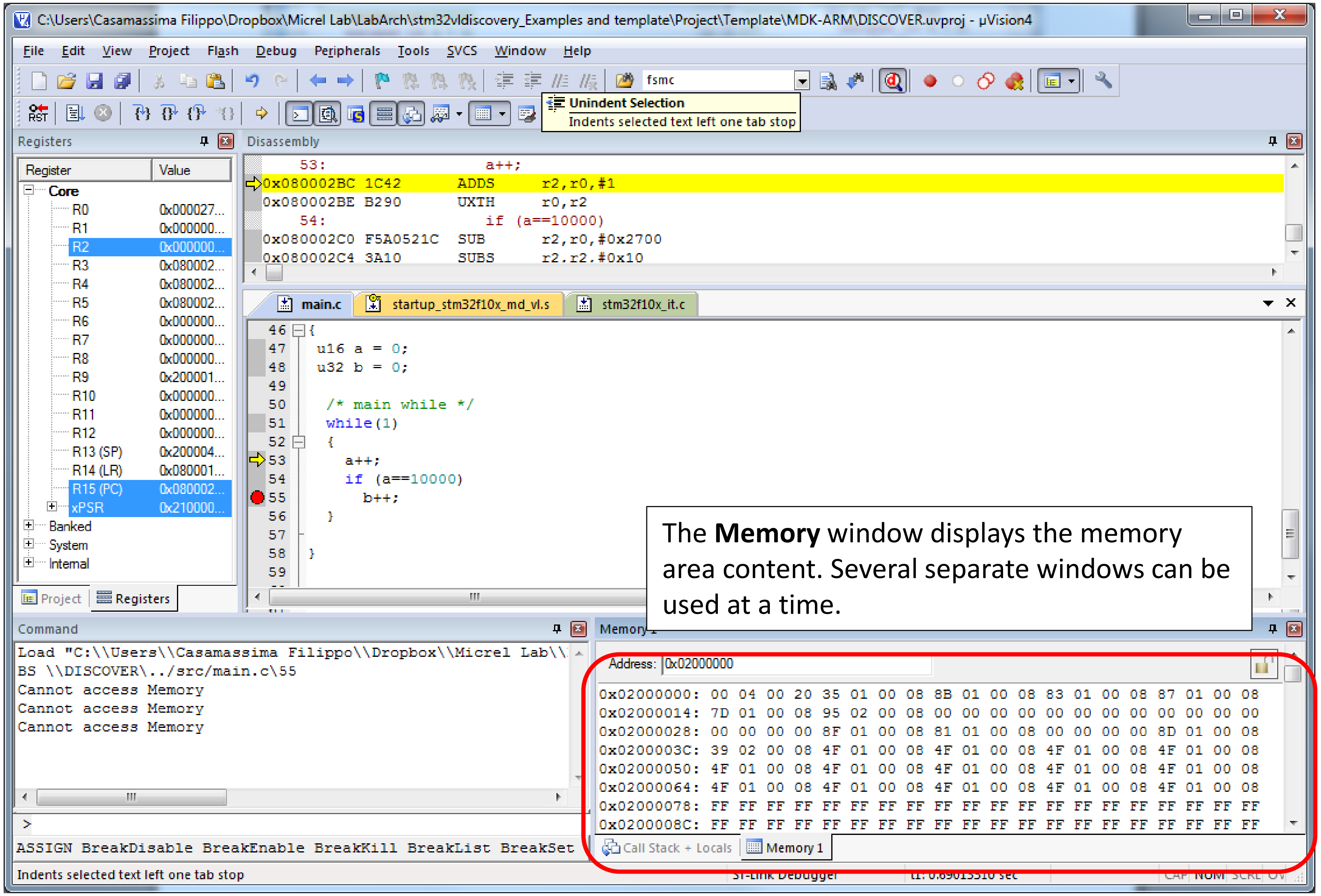
The Call Stack + Locals window shows local variables and functions.

```
Load "C:\\Users\\Casamassima Filippo\\Dropbox\\Micrel Lab\\
BS \\DISCOVER\../src/main.c\55
```

```
>
ASSIGN BreakDisable BreakEnable BreakKill BreakList BreakSet
```

Name	Location/Value	Type
main	0x080002B6	int f()
a	<not in scope>	auto - unsigned short
b	<not in scope>	auto - unsigned int

Code Debugging



C:\Users\Casamassima Filippo\Dropbox\Micrel Lab\LabArch\stm32vldiscovery_Examples and template\Project\Template\MDK-ARM\DISCOVER.uvproj - µVision4

File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
fsmc
Unindent Selection
Indents selected text left one tab stop
Registers
Disassembly
Project
Registers
main.c
startup_stm32f10x_md_vl.s
stm32f10x_it.c
Command
Memory 1
Call Stack + Locals

Register	Value
Core	
R0	0x000027...
R1	0x000000...
R2	0x000000...
R3	0x080002...
R4	0x080002...
R5	0x080002...
R6	0x000000...
R7	0x000000...
R8	0x000000...
R9	0x200001...
R10	0x000000...
R11	0x000000...
R12	0x000000...
R13 (SP)	0x200004...
R14 (LR)	0x080001...
R15 (PC)	0x080002...
xPSR	0x210000...
Banked	
System	
Internal	

```
    53:                 a++;
0x080002BC 1C42      ADDS      r2,r0,#1
0x080002BE B290      UXTH      r0,r2
    54:                 if  (a==10000)
0x080002C0 F5A0521C  SUB       r2,r0,#0x2700
0x080002C4 3A10      SUBS      r2,r2,#0x10
```

```
{
  u16 a = 0;
  u32 b = 0;

   /* main while */
   while(1)
   {
     a++;
     if (a==10000)
       b++;
   }

}

```

The **Memory** window displays the memory area content. Several separate windows can be used at a time.

```
Load "C:\\Users\\Casamassima Filippo\\Dropbox\\Micrel Lab\\
BS \\DISCOVER\../src/main.c\55
Cannot access Memory
Cannot access Memory
Cannot access Memory
```

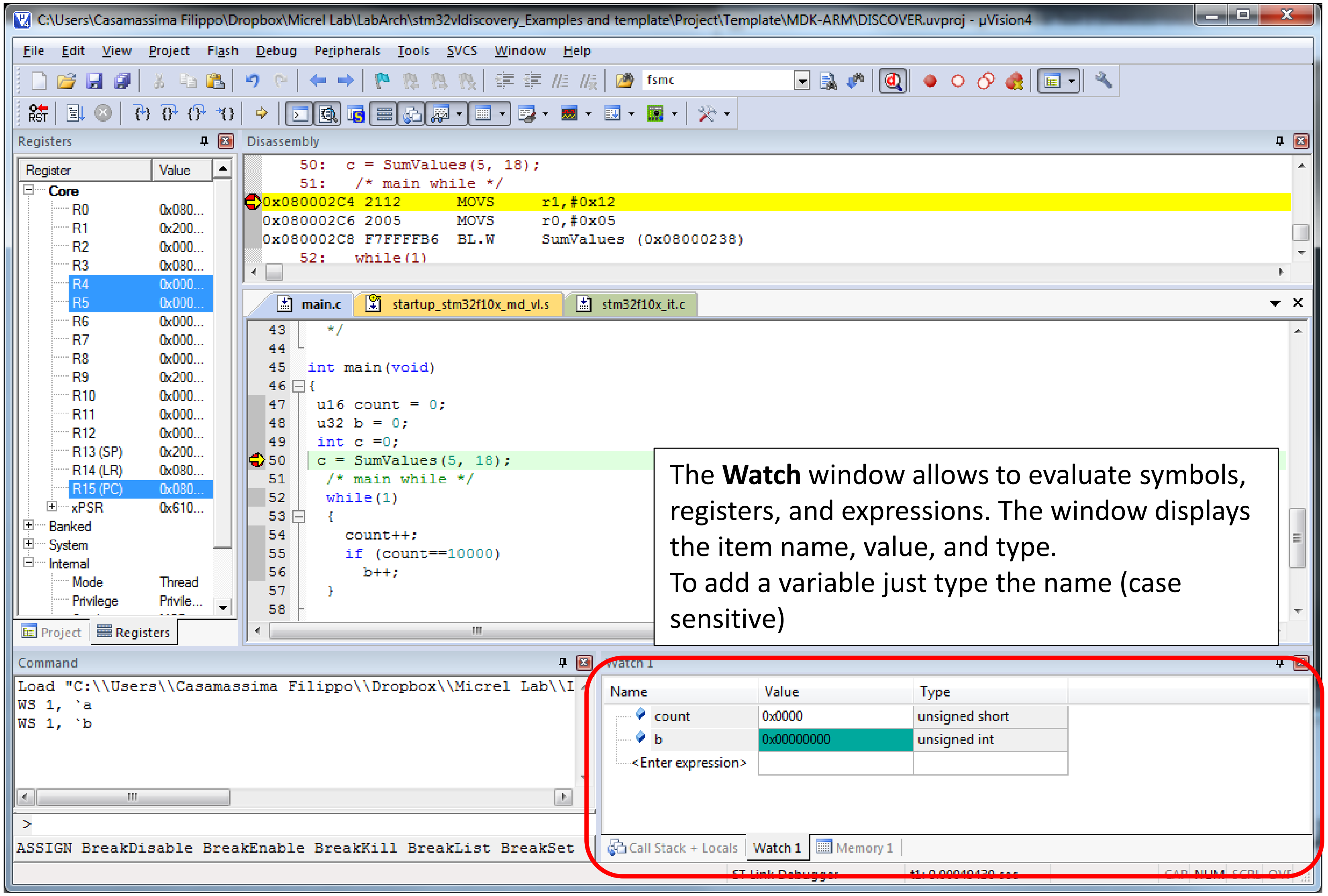
```
>
ASSIGN BreakDisable BreakEnable BreakKill BreakList BreakSet
```

Address: 0x02000000

```
0x02000000: 00 04 00 20 35 01 00 08 8B 01 00 08 83 01 00 08 87 01 00 08
0x02000014: 7D 01 00 08 95 02 00 08 00 00 00 00 00 00 00 00 00 00 00 00
0x02000028: 00 00 00 00 8F 01 00 08 81 01 00 08 00 00 00 00 8D 01 00 08
0x0200003C: 39 02 00 08 4F 01 00 08 4F 01 00 08 4F 01 00 08 4F 01 00 08
0x02000050: 4F 01 00 08 4F 01 00 08 4F 01 00 08 4F 01 00 08 4F 01 00 08
0x02000064: 4F 01 00 08 4F 01 00 08 4F 01 00 08 4F 01 00 08 4F 01 00 08
0x02000078: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
0x0200008C: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
```

Indents selected text left one tab stop   ST-Link Debugger   t1: 0.69015310 sec   CAP NUM SCRL OVR

Code Debugging


C:\Users\Casamassima Filippo\Dropbox\Micrel Lab\LabArch\stm32vldiscovery_Examples and template\Project\Template\MDK-ARM\DISCOVER.uvproj - µVision4
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
fsmc
Registers
Disassembly
main.c
startup_stm32f10x_md_vl.s
stm32f10x_it.c
Project
Registers
Command
Watch 1
Call Stack + Locals
Watch 1
Memory 1
ST-Link Debugger
CAP NUM SCRL OVR

Register	Value
Core	
R0	0x080...
R1	0x200...
R2	0x000...
R3	0x080...
R4	0x000...
R5	0x000...
R6	0x000...
R7	0x000...
R8	0x000...
R9	0x200...
R10	0x000...
R11	0x000...
R12	0x000...
R13 (SP)	0x200...
R14 (LR)	0x080...
R15 (PC)	0x080...
xPSR	0x610...
Banked	
System	
Internal	
Mode	Thread
Privilege	Privile...

```
    50:   c = SumValues(5, 18);
    51:   /* main while */
0x080002C4 2112       MOVS      r1,#0x12
0x080002C6 2005       MOVS      r0,#0x05
0x080002C8 F7FFFFB6   BL.W      SumValues (0x08000238)
    52:   while(1)
```

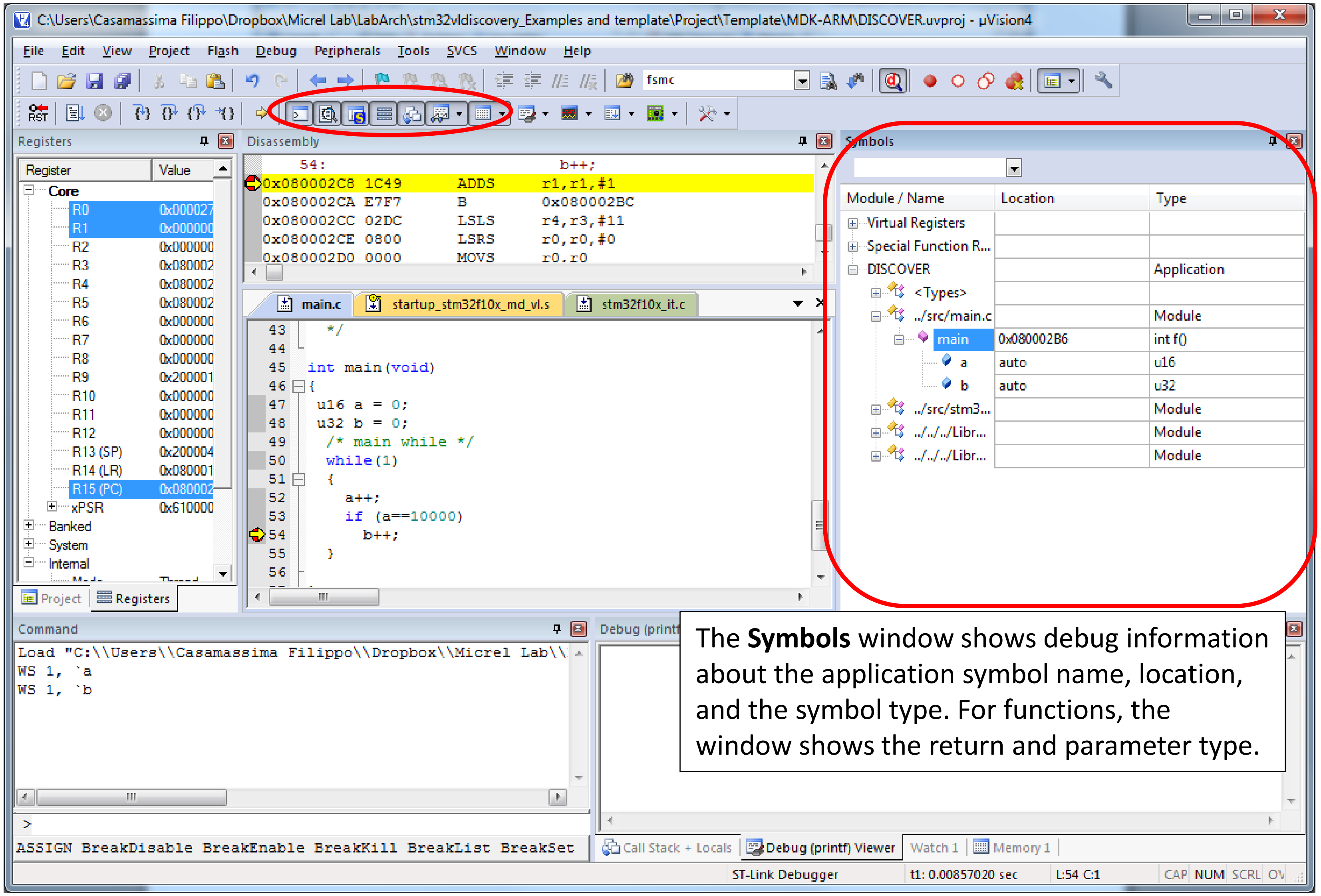
```
  */

int main(void)
{
 u16 count = 0;
 u32 b = 0;
 int c =0;
 c = SumValues(5, 18);
  /* main while */
  while(1)
  {
    count++;
    if (count==10000)
      b++;
  }
```

```
Load "C:\\Users\\Casamassima Filippo\\Dropbox\\Micrel Lab\\I
WS 1, `a
WS 1, `b

>
ASSIGN BreakDisable BreakEnable BreakKill BreakList BreakSet
```

Name	Value	Type
count	0x0000	unsigned short
b	0x00000000	unsigned int
<Enter expression>		

The **Watch** window allows to evaluate symbols, registers, and expressions. The window displays the item name, value, and type.
To add a variable just type the name (case sensitive)

Code Debugging



C:\Users\Casamassima Filippo\Dropbox\Micrel Lab\LabArch\stm32vldiscovery_Examples and template\Project\Template\MDK-ARM\DISCOVER.uvproj - µVision4

Register	Value
Core	
R0	0x000027
R1	0x000000
R2	0x000000
R3	0x080002
R4	0x080002
R5	0x080002
R6	0x000000
R7	0x000000
R8	0x000000
R9	0x200001
R10	0x000000
R11	0x000000
R12	0x000000
R13 (SP)	0x200004
R14 (LR)	0x080001
R15 (PC)	0x080002
xPSR	0x610000
Banked	
System	
Internal	

```
        54:                    b++;
0x080002C8 1C49      ADDS      r1,r1,#1
0x080002CA E7F7      B         0x080002BC
0x080002CC 02DC      LSLS      r4,r3,#11
0x080002CE 0800      LSRS      r0,r0,#0
0x080002D0 0000      MOVS      r0,r0
```

```
   */

int main(void)
{
  u16 a = 0;
  u32 b = 0;
   /* main while */
   while(1)
   {
     a++;
     if (a==10000)
       b++;
   }
```

Module / Name	Location	Type
Virtual Registers		
Special Function R...		
DISCOVER		Application
<Types>		
../src/main.c		Module
main	0x080002B6	int f()
a	auto	u16
b	auto	u32
../src/stm3...		Module
../../../Libr...		Module
../../../Libr...		Module

```
Load "C:\\Users\\Casamassima Filippo\\Dropbox\\Micrel Lab\\
WS 1, `a
WS 1, `b
>
ASSIGN BreakDisable BreakEnable BreakKill BreakList BreakSet
```

ST-Link Debugger   t1: 0.00857020 sec   L:54 C:1   CAP NUM SCRL OVR

The **Symbols** window shows debug information about the application symbol name, location, and the symbol type. For functions, the window shows the return and parameter type.

Questions?