Please note that Cypress is an Infineon Technologies Company.
The document following this cover page is marked as “Cypress” document as this is the company that originally developed the product. Please note that Infineon will continue to offer the product to new and existing customers as part of the Infineon product portfolio.

Continuity of document content
The fact that Infineon offers the following product as part of the Infineon product portfolio does not lead to any changes to this document. Future revisions will occur when appropriate, and any changes will be set out on the document history page.

Continuity of ordering part numbers
Infineon continues to support existing part numbers. Please continue to use the ordering part numbers listed in the datasheet for ordering.



THIS SPEC IS OBSOLETE

Spec No: 002-05238

Spec Title: AN205238 - FM3 Microcontroller MB9A310 Series, FSS MB9AF314L EV-Board

Replaced by: None

AN205238

FM3 Microcontroller MB9A310 Series, FSS MB9AF314L EV-Board

This application note describes about the FSS MB9AF314L EV-Board (PN: FSSDC-9A314-EVB) which provides an economical and simple means for study usage for MB9A310 series MCU.

OBSOLETE

Contents

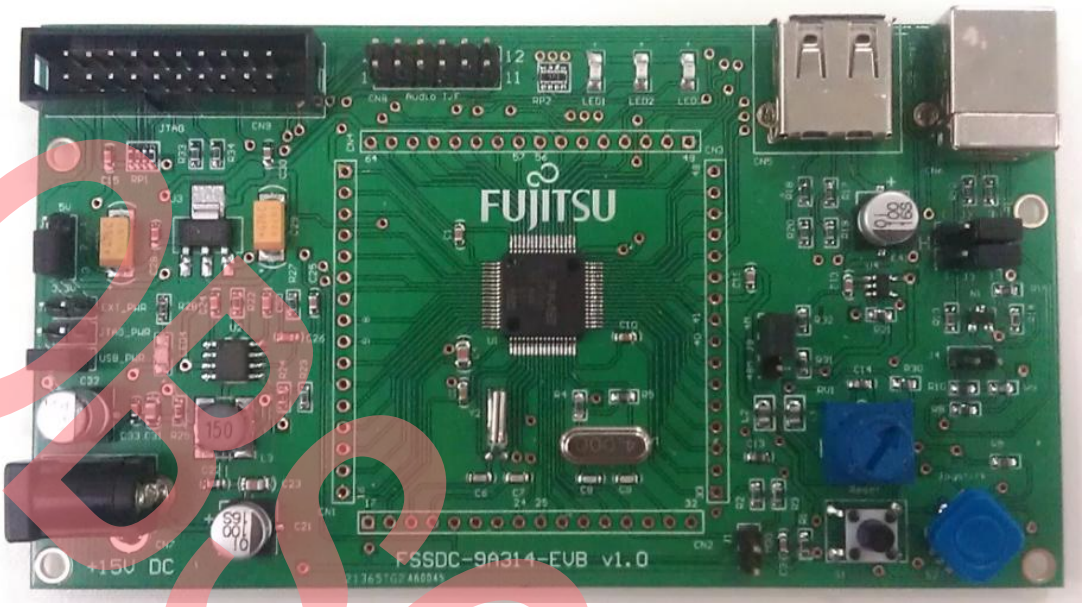
1 Introduction....1
1.1 Product Overview....1
1.2 MB9A310 Series MCU....2
1.3 Board Features....3
1.4 Getting Started....3
2 Hardware Setting....4
2.1 Main Features....4
2.2 Jumpers Overview....5
2.3 Setting for USB On-Board Programming....5
2.4 Setting for Debug Tool....6
3 Flash On-Board Programming....6
3.1 On-Board Programming via UART....6
3.2 On-Board Programming via USB....8
4 Sample Code....10
4.1 Blinky....11
4.2 USB Function....11
4.3 USB Host (Catch Mouse)....13
4.4 USB Host (Access to U-Disk)....14
5 Debug Tool and IDE....15
5.1 Debug with J-Link in IAR EWARM Workbench....16
5.2 Debug with U-Link ME in Keil uVision4....22
6 Materials Download....28
A Appendix....29
Links to Microcontrollers....29
Links to third party sites....29
Document History....30

1 Introduction

1.1 Product Overview

FSS MB9AF314L EV-Board (PN: FSSDC-9A314-EVB) provides an economical and simple means for study usage for MB9A310 series MCU. The board compatible with both 3.3 and 5V system contains some external resources (LED, Key, USB Device, USB Host interface…) to demonstrate MCU periphery function. It also provides standard 20 pin JTAG interface, which is both compatible with IAR and Keil debug tool. In addition, it allows On-board programming with USB mode.

Figure 1. FSSDC-9A314-EVB Overview



1.2 MB9A310 Series MCU

MB9A310 series MCU is 32-bit general purpose MCU of FM3 family that features the industry's leading-edge ARM Cortex-M3TM CPU and integrates Cypress's highly reliable and high-speed secure embedded flash technology. This MCU can operate at up to 40MHz CPU frequency and work at a wide voltage range (2.7V-5.5V), which can be both compatible with 3.3V and 5V system.

It includes a host of robust peripheral features, including motor control timers (MFT), base timer (can be configured to PWM, PPG, Reload, PWC timer), ADCs, on-chip memory (up to 256K Flash, up to 32K SRAM) and a wide range of communication interfaces (USB, I2C, SIO, LIN, CAN).

The size of on-chip memory can be configured according to different part number and the package is available in LQFP and BGA, shown in following table.

Table 1. Product List

Product	Flash	SRAM	Package
MB9AF316M/N	512kB	32kB	M: LQFP-80 N: LQFP-100
MB9AF315M/N	384kB	32kB	M: LQFP-80 N: LQFP-100
MB9AF314L/M/N	256kB	32kB	L: LQFP-64 M: LQFP-80 N: LQFP-100
MB9AF312L/M/N	128kB	16kB	L: LQFP-64 M: LQFP-80 N: LQFP-100
MB9AF311L/M/N	64kB	16Kb	L: LQFP-64 M: LQFP-80 N: LQFP-100

OBSOLETE

1.3 Board Features

FSS MB9AF314L EV-Board includes following features:

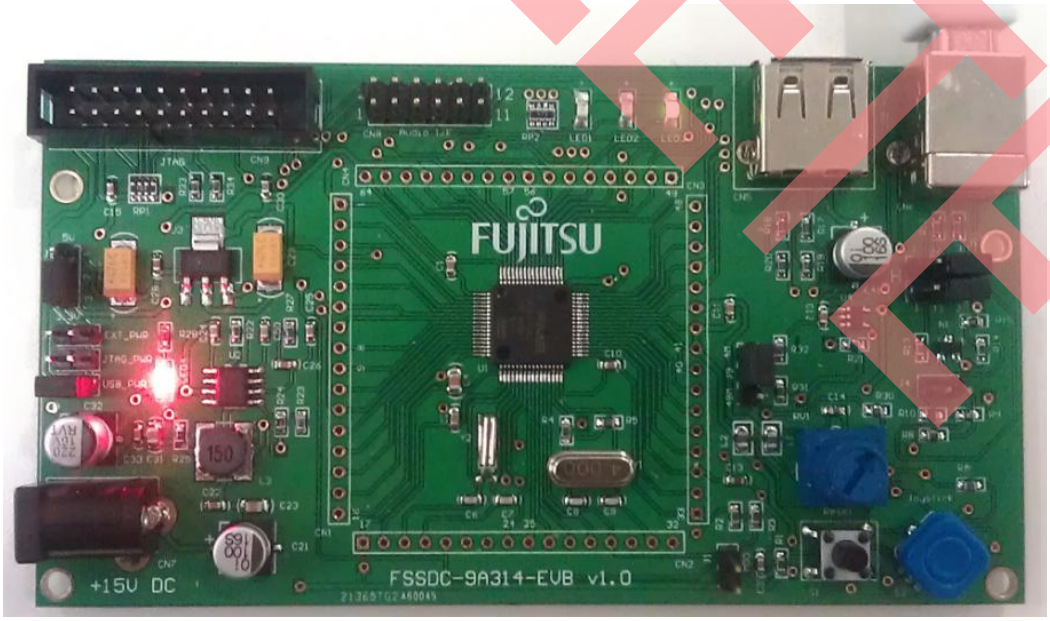
- Microcontroller MB9AF314L
- 1x USB-Host (Type-A connector)
- 1x USB-Device (Type-B connector)
- Standard JTAG Interface on a 20 pin-header
- 3x LED
- 1x Joystick
- 1x 'Reset'-button
- 1x potentiometer regulating input voltage to AD channel
- All 64 pins routed to test pads
- Audio Interface
- On-board 5V and 3V voltage regulators, 'Power'-LED
- Power supply via USB-Device, JTAG or external with 15V power connector

1.4 Getting Started

The microcontroller on the FSSDC-9A314-EVB is already programmed with a test program. Please follow the test process as below first.

1. Make sure J7 is short, and connect the FSSDC-9A314-EVB with PC via USB Function interface (Type-B Connector)
2. Watch that the LED3 blinks for a short time, then stays at "ON" state; LED1 and LED2 plays as horse-race lamp.
3. Turning the potentiometer clockwise will make the LED1 and LED2 blink fast, and turning the potentiometer anti-clockwise will make the LED1 and LED2 blink slowly.

Figure 2. FSSDC-9A314-EVB Test



OBSOLETE

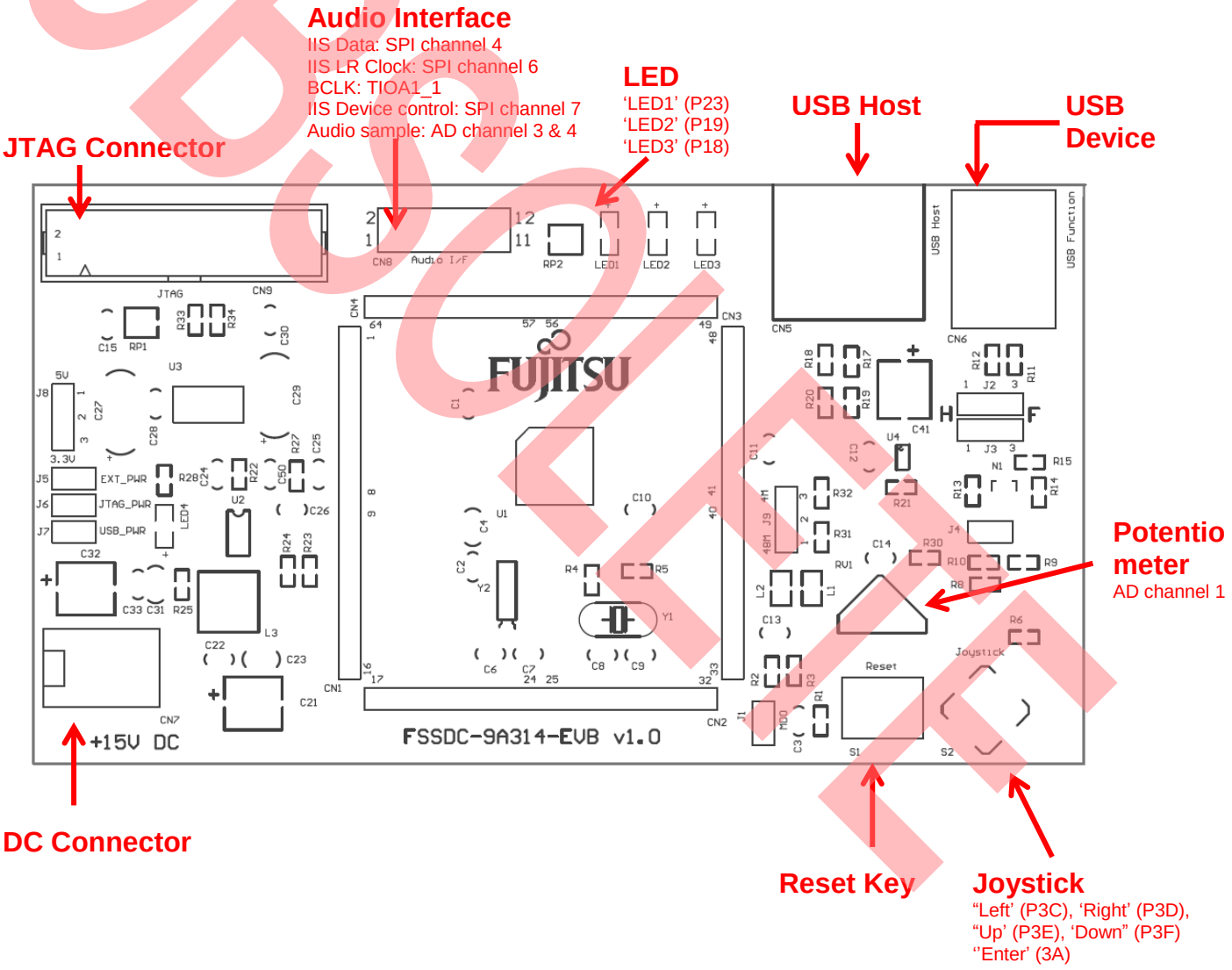
4. Actually now, the FSSDC-9A314-EVB can be used as a simple USB mouse, press the up/down/left/right key of Joystick to move the cursor on the screen, and the enter key of Joystick acts as “Left Click” of mouse.

If the test result is normal as described, congratulation! You finished the first test on the FSSDC-9A314-EVB, and now you will get more details about FSSDC-9A314-EVB at following chapters. You will learn about:

- The hardware setting
- How to program the Flash
- How to start with IAR-Embedded-Workbench and Keil µVision

2 Hardware Setting

2.1 Main Features



OBSOLETE

2.2 Jumpers Overview

Table 2. Jumpers Overview

Jumper Name	Function	Setting
J5-J7	Power selection	Short J5: select external power Short J6: select JTAG power Short J7: select USB power
J8*2	Voltage selection	Short 1,2: 5V Short 2,3: 3.3V
J2 J3	USB Host/Function selection	Short 1,2: USB Host Short 2,3: USB Function
J1	Mode setting	Short: Programming mode Open: Normal mode
J9	Oscillator Selection	Short 1,2: when 48M oscillator is used Short 2,3: when 4M oscillator is used
J4	Voltage division	Short: when 5V power is used Open: when 3.3V power is used

Attention:

1. Never short J5-J7 at the same time! When Keil U-Link ME is used, JTAG power cannot be used.
2. When Keil U-link is used, only 3.3V can be selected, never select 5V power!

2.3 Setting for USB On-Board Programming

Table 3. Setting for USB On-Board Programming

Jumper Number	Function	Setting
CN6	USB function connector	Connect with PC via USB cable
J2 J3	Host/Function selection	Short 2,3
J1	Mode setting	Short
J9	Oscillator Selection	Short 2,3
J4	Voltage division	Short: when 5V power is used Open: when 3.3V power is used

OBSOLETE

2.4 Setting for Debug Tool

- Use IAR J-Link

Table 4. Setting for J-Link

Part Number	Function	Setting
CN9	JTAG connctor	Connect with J-Link
J5-J7	Power selection	Short J5: select external power Short J6: select JTAG power Short J7: select USB power
J8	Voltage selection	Short 1,2: 5V Short 2,3: 3.3V
J1	Mode setting	Open

- Use Keil U-Link ME

Table 5. Setting for U-Link

Part Number	Function	Setting
CN9	JTAG connctor	Connect with U-Link ME
J5-J7	Power selection	Short J5: select external power Short J7: select USB power
J8	Voltage selection	Short 2,3
J1	Mode setting	Open

3 Flash On-Board Programming

3.1 On-Board Programming via UART

- First check the hardware setting as introduced by section 2.3.
  - First check the hardware setting as introduced by section 2.3.
  - Install the UART programmer: PCWFM3_V01L01. (It can be downloaded on the web)

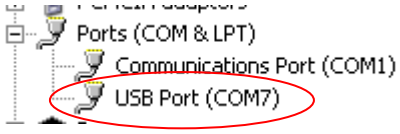
Figure 3. USB Device Sign

  - 

OBSOLETE

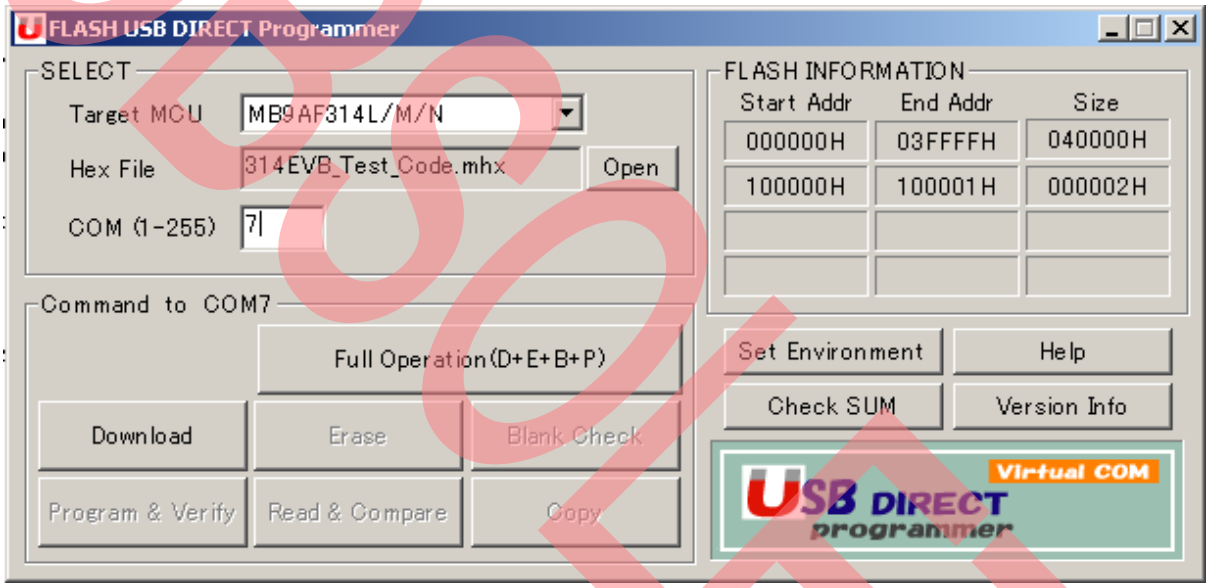
- Check the COM port for this USB port in the device manager.

Figure 4. : COM Port in Device Manager



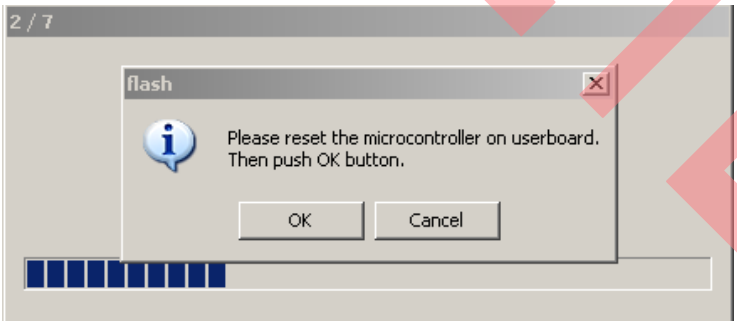
- Install the USB programmer: usbdirect-v01l04.zip. (It can be downloaded on the web)
- Open it, set the parameter as shown in following figure, and select Hex file.

Figure 5. USB Programmer Overview



- Click Full Operation.

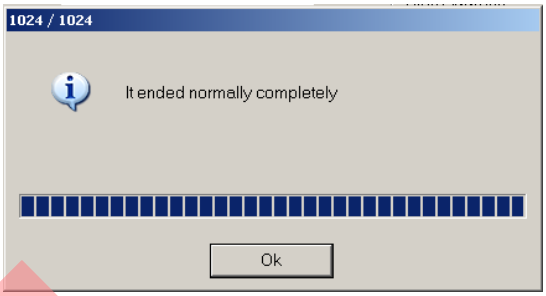
Figure 6. Press Reset Button



- Press reset key in the board, and the programming will start.

OBSOLETE

Figure 7. Start Programming



3.2 On-Board Programming via USB

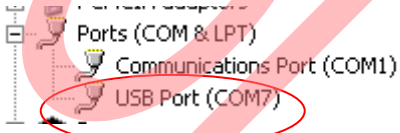
- First check the hardware setting as introduced by section 2.4.
- After connect with PC via USB cable, the EVB can be identified as a USB device.

Figure 8. USB Device Sign



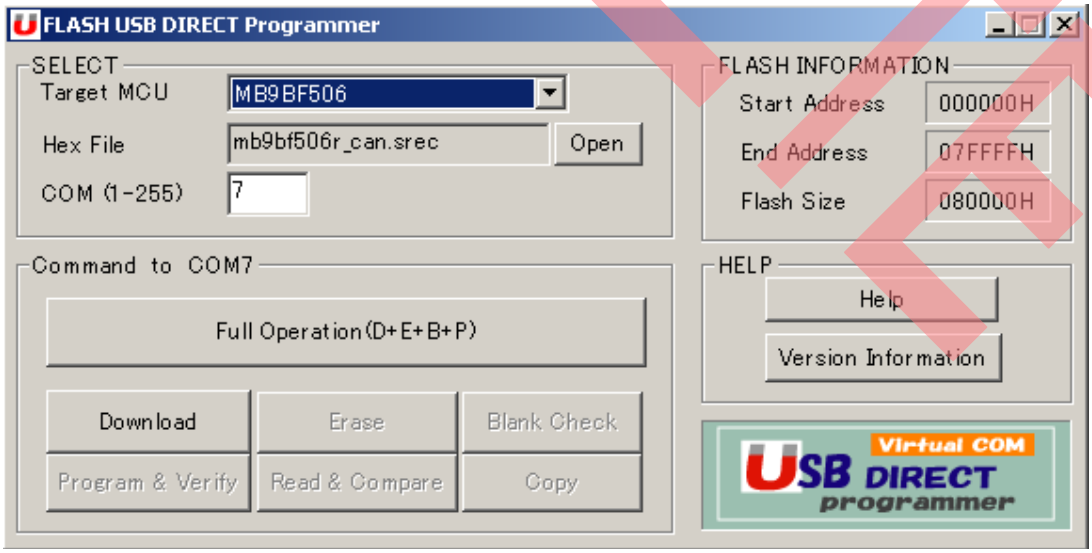
- Check the COM port for this USB port in the device manager.

Figure 9. COM Port in Device Manager



- Install the USB programmer: USBDW_V01L03. (It can be downloaded on the web)
- Open it, set the parameter as shown in following figure, and select Hex file.

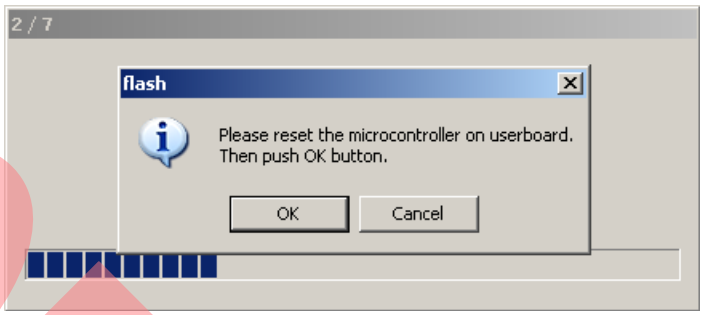
Figure 10. USB Programmer Overview



- Click Full Operation.

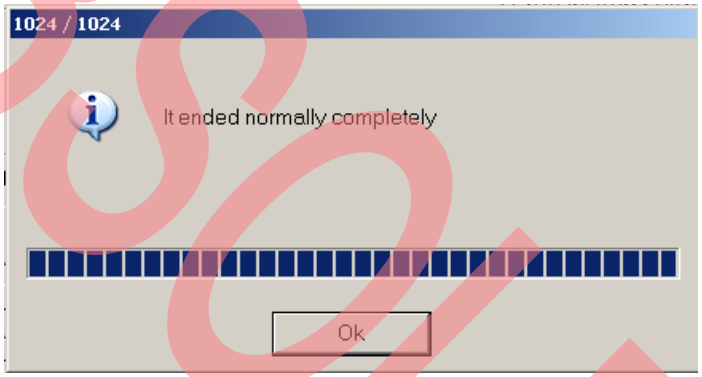
OBSOLETE

Figure 11. Press Reset Button



- Press reset key on the board. Programming will start.

Figure 12. Start Programming



OBSOLETE

4 Sample Code

The sample codes for FSSDC-9A314-EVB board are listed as following table.

Table 6. Sample Code

Project Name	Description
Blinky	Turn the potentiometer to change the LED blinking frequency.
usb_function_mouse	Implement the USB mouse to demonstrate USB function
usb_host_catch_mouse	Identify a USB mouse and print the mouse position on debug window
usb_host_msc	mo USB Host mass storage class function, and implement FatFS in the sample code, us can create/write/read/delete files on the U-disk.

Note:

1. It provides both IAR and Keil project for these sample code, IAR project is developed in EWARM Embedded Workbench V6.21, and Keil project is developed in Keil uVision 4.21.

2. If user use other version to open these projects, compiling error may occur, in this case, please check following setting.

- IAR IDE
  - MCU type
  - Pre-included file
  - ICF file
  - Flash loader
- Keil IDE
  - MCU type
  - Pre-included file
  - ROM & RAM memory address

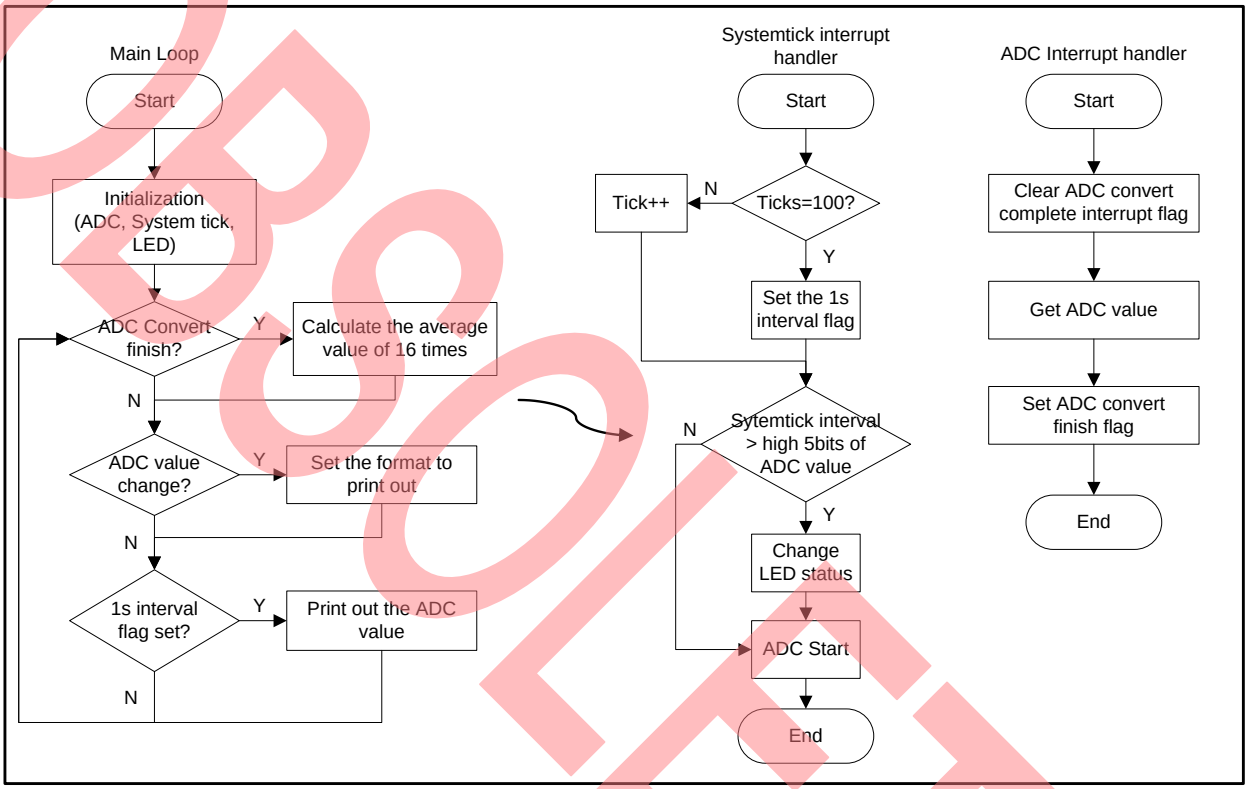
OBSOLETE

4.1 Blinky

- Hardware Setting
  None

- Flowchart

Figure 13. Blinky Flowchart



- Usage

1. Turn the potentiometer (RV1) clockwise and anti-clockwise
2. Watch the LED blinking frequency

4.2 USB Function

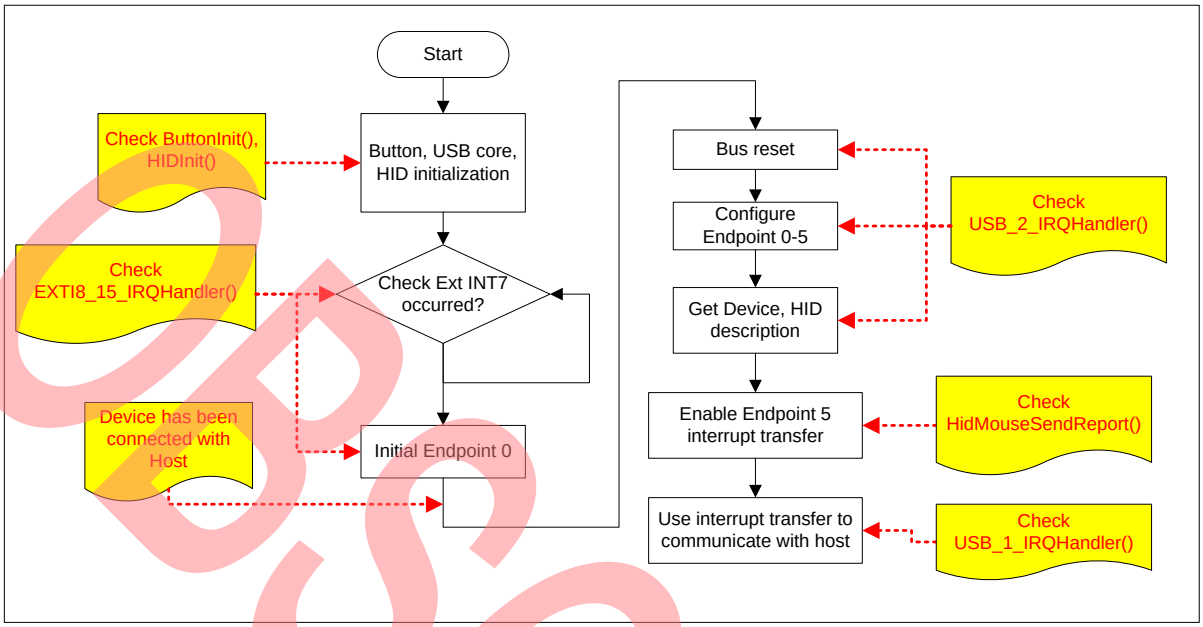
- Hardware Setting
  - Check if 2,3 of J2 and J3 short
  - Check J4 (Short: 5V, Open: 3.3V)
  - Connect with PC via USB cable

- Flowchart

The following flowchart illuminates the procedure to implement a USB mouse, it is not a certain flow of a function, but provides a clue to study and understand the sample code.

OBSOLETE

Figure 14. USB Function Sample Code Flowchart



OBSOLETE

- Usage

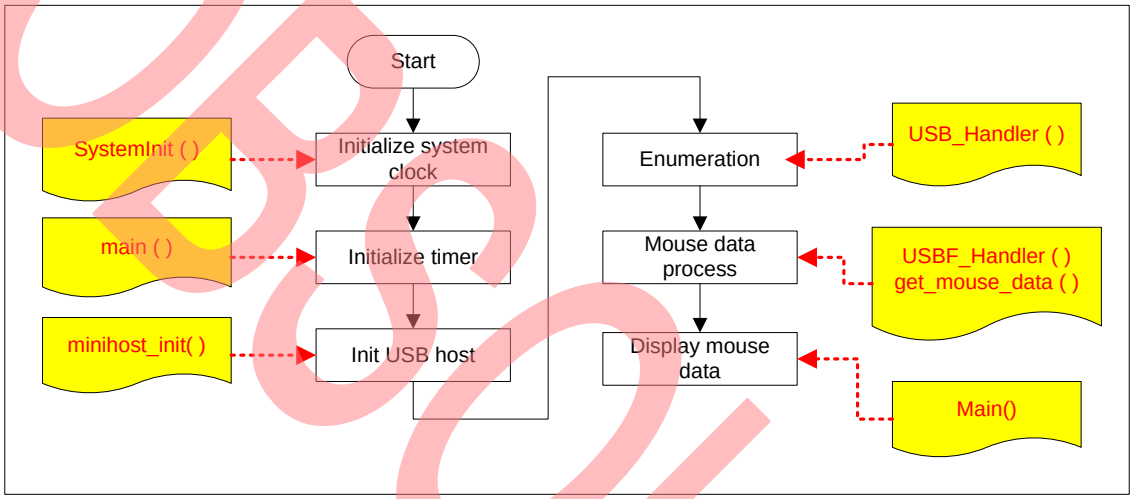
1. Make the hardware setting.
2. Watch the mouse moving when pressing “Up”, “Down”, “Left”, “Right”, “Enter” keys of Joystick.

4.3 USB Host (Catch Mouse)

- Hardware Setting
  - Check if 1,2 of J2 and J3 short
  - Connect with a USB mouse
- Flowchart

The following flowchart illuminates the procedure to implement USB host function, which can catch the position of a USB mouse, it is not a certain flow of a function, but provides a clue to study and understand the sample code.

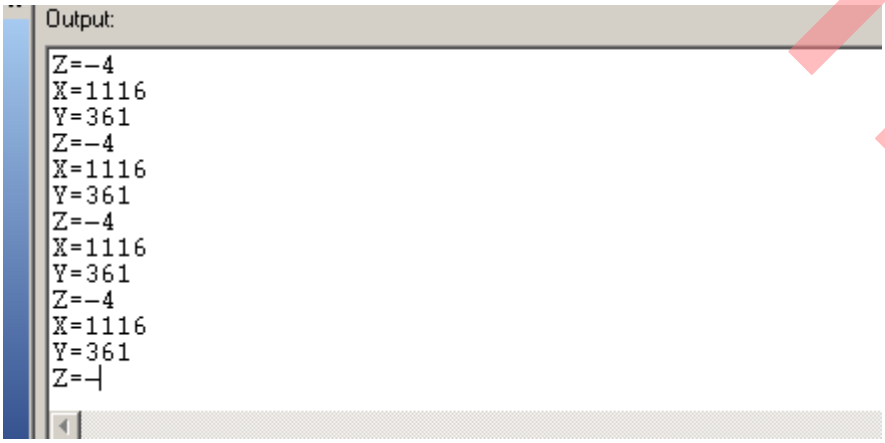
Figure 15. USB Host Catch Mouse Flowchart



- Usage

1. Make the hardware setting
2. Enable definition “Debug” in the code
3. Run the code.
4. Move the USB mouse, and watch the mouse position in the terminal IO window as following figure. (Debug viewer window in Keil IDE)

Figure 16. USB Mouse Position Display



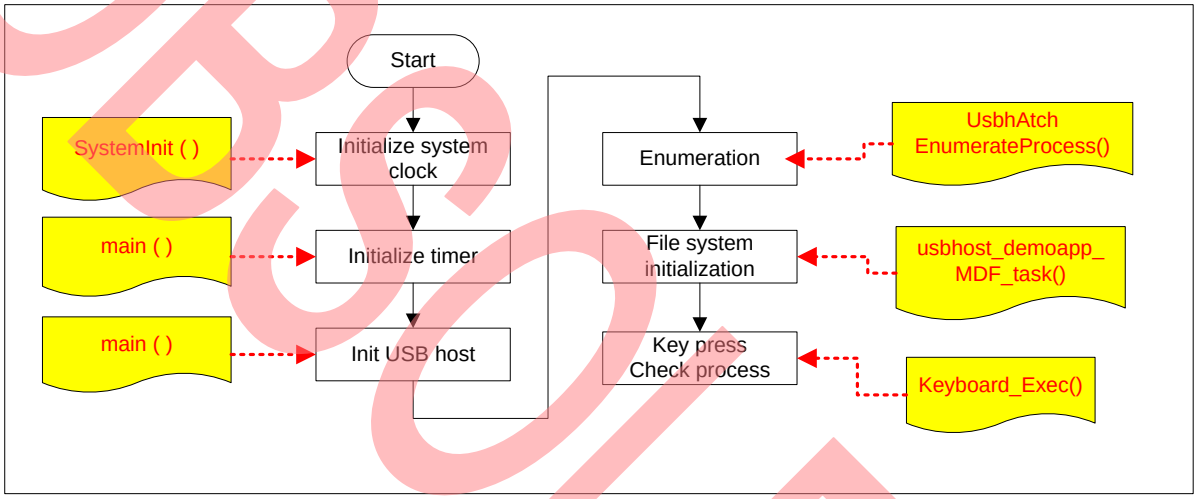
OBSOLETE

4.4 USB Host (Access to U-Disk)

- Hardware Setting
  - Check if 1,2 of J2 and J3 short
  - Connect with a U-Disk
- Flowchart

The following flowchart illuminates the procedure to implement USB Host Mass Storage Class function, which can identify the U-Disk and make file operation on the U-Disk, it is not a certain flow of a function, but provides a clue to study and understand the sample code.

Figure 17. U-Disk Access Demo Flowchart



- Usage

1. Run the code
2. Plug-in the U-Disk
3. Press “Left key” of Joystick
4. Plug-out the U-Disk and check if a file is created (Path: ..\F1\F2\) on PC.
5. Plug-in the U-Disk again
6. Press “Right key” of Joystick
7. Plug-out the U-Disk and check if the file is deleted

OBSOLETE

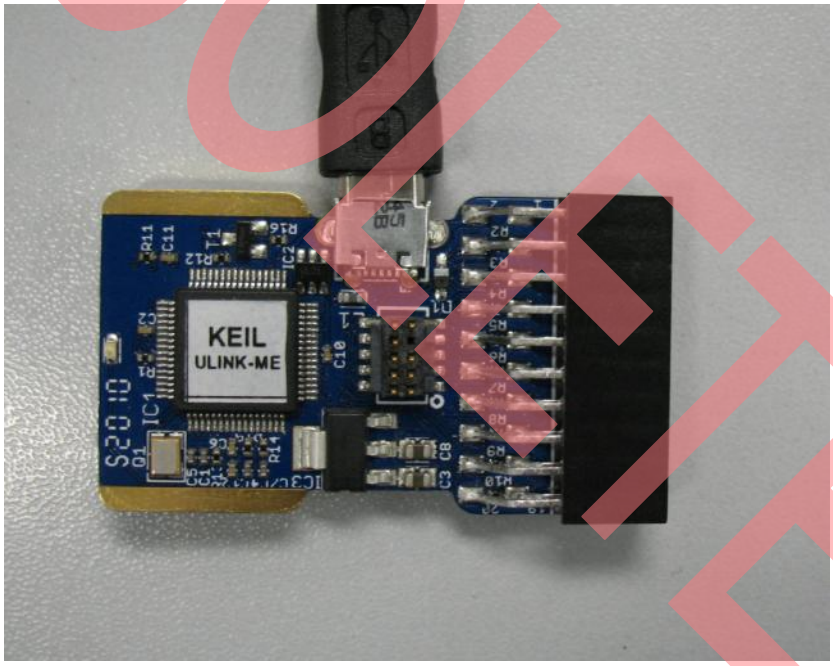
5 Debug Tool and IDE

FSS MB9AF314L EV-Board supports both Keil U-Link-ME and IAR J-Link for debug shown as following.

Figure 18. J-Link Overview



Figure 19. U-Link Overview



The U-Link-me should be used with Keil uVision 4 which can be downloaded freely from following web.

https://www.keil.com/update/sw/RVMDK/4.21

The J-Link should be used with IAR Embedded Workbench which can be downloaded freely from following web.

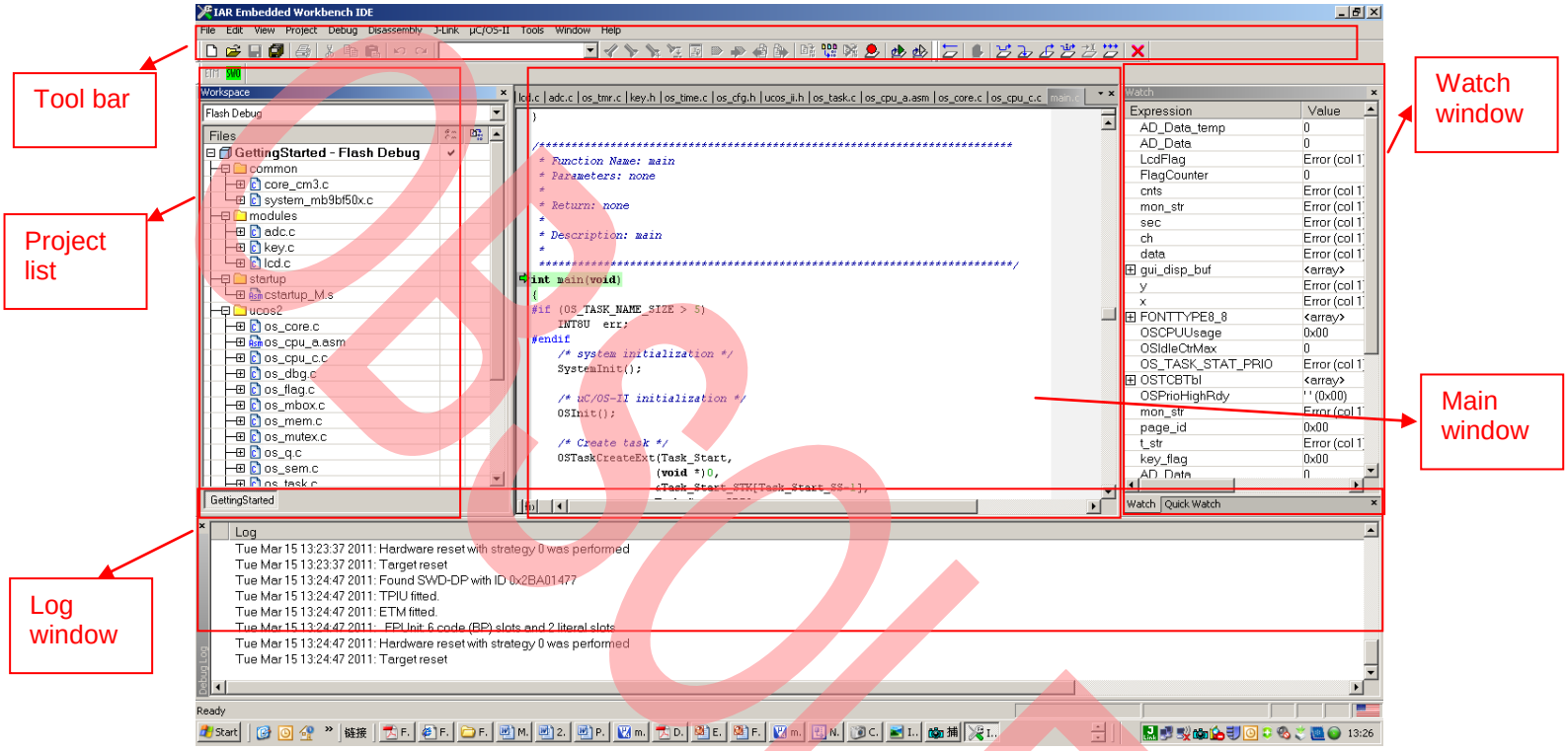
http://www.iar.com/website1/1.0.1.0/68/1/

OBSOLETE

5.1 Debug with J-Link in IAR EWARM Workbench

The sample code can be debugged in IAR EWARM Workbench with J-Link. The following figure shows basic debug window.

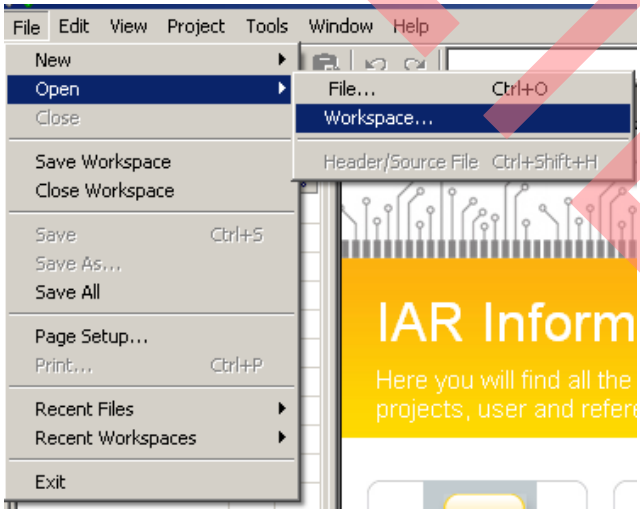
Figure 20. IAR IDE Overview



- Run an Existed Project

1. Open a project by clicking “File | Open | Workspace”

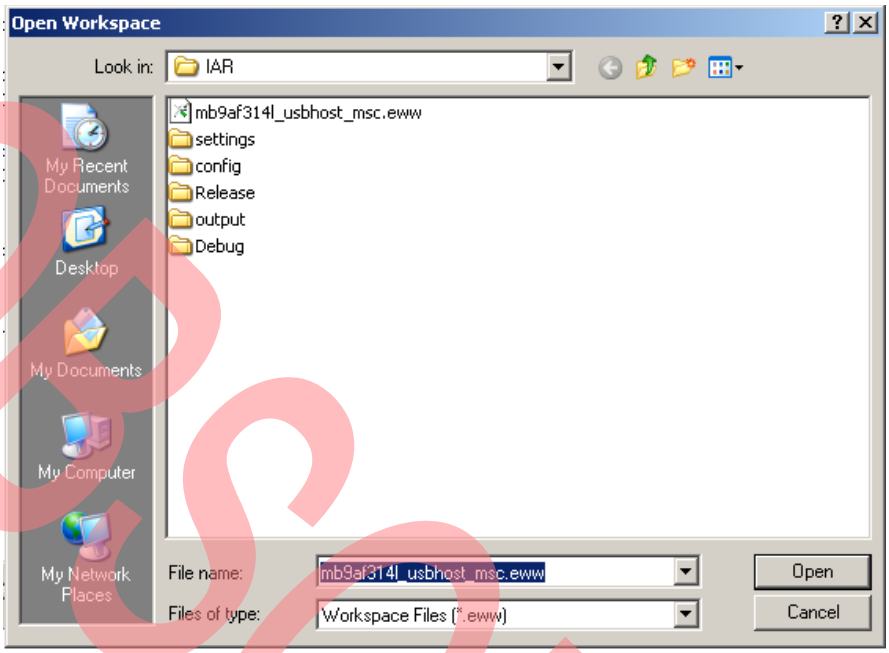
Figure 21. Open a Project



OBSOLETE

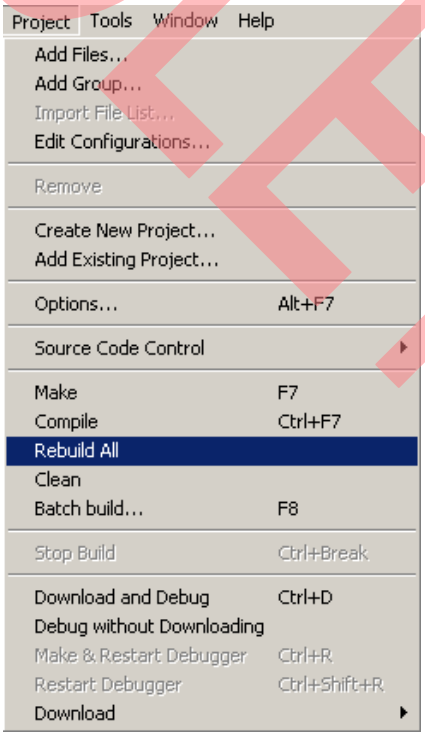
2. Select a project (eww file)

Figure 22. Select a Project



3. Click “Project | Rebuild All”

Figure 23. Rebuild All



OBSOLETE

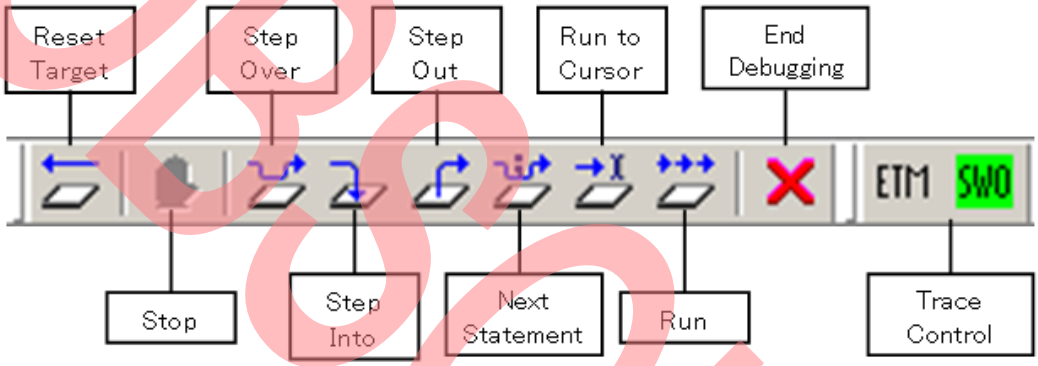
4. Click “Download and Debug”

Figure 24. Click Download and Debug



5. Use following tool bar to debug

Figure 25. Debug Tool Bar

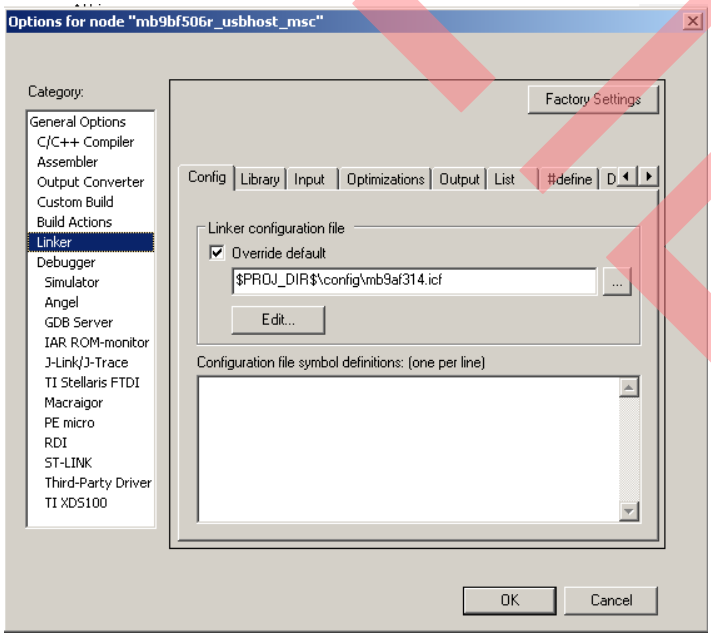


The sample codes support both Flash and RAM debug in IAR EWARM Workbench, if Flash debug is used, the code is programmed into MB9AF314L Flash. If RAM debug is used, the code only runs in RAM area, and after power off, the code will not be stored, but the RAM debug will be faster than Flash debug.

- Setting for Flash Debug

1. Check the configuration file path ($PROJ_DIR$\config\mb9af314.icf) in Linker table.

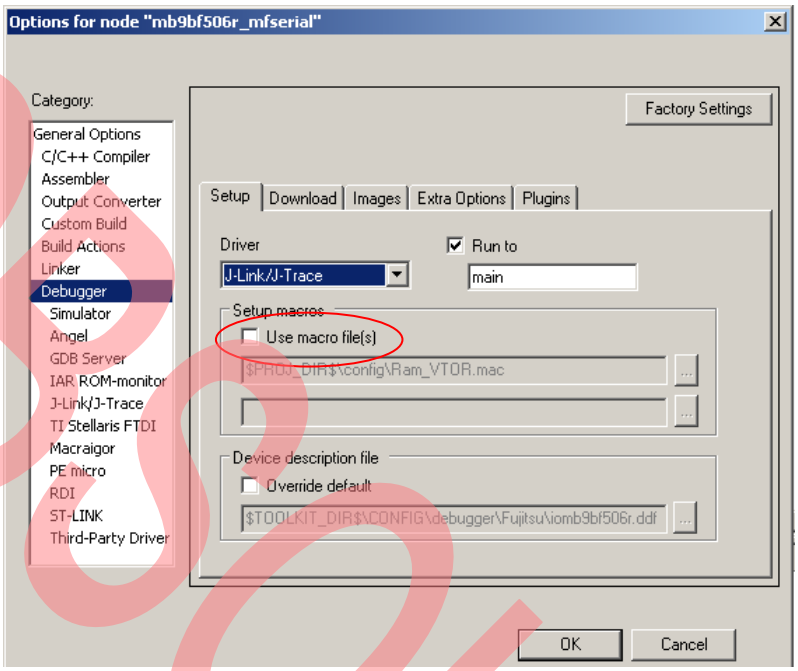
Figure 26. ICF File for Flash Debug



OBSOLETE

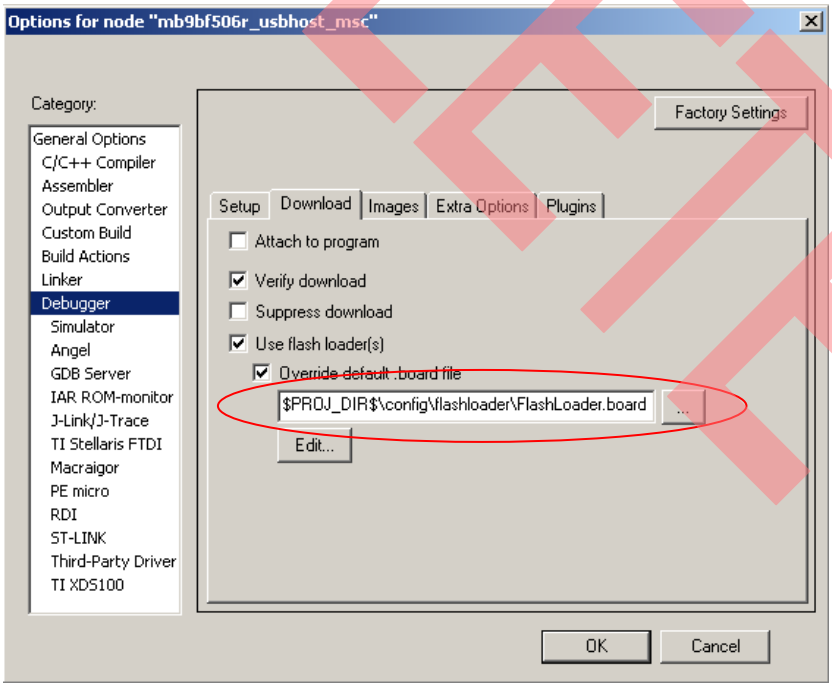
2. Don’t select “Use macro files” in “Debugger|Setup” table.

Figure 27. Macro File Disabled



3. Set Flash loader file path ($PROJ_DIR$\config\flashloader\FlashLoader.board) in “Debug|Download” table.

Figure 28. Flash Load File Path

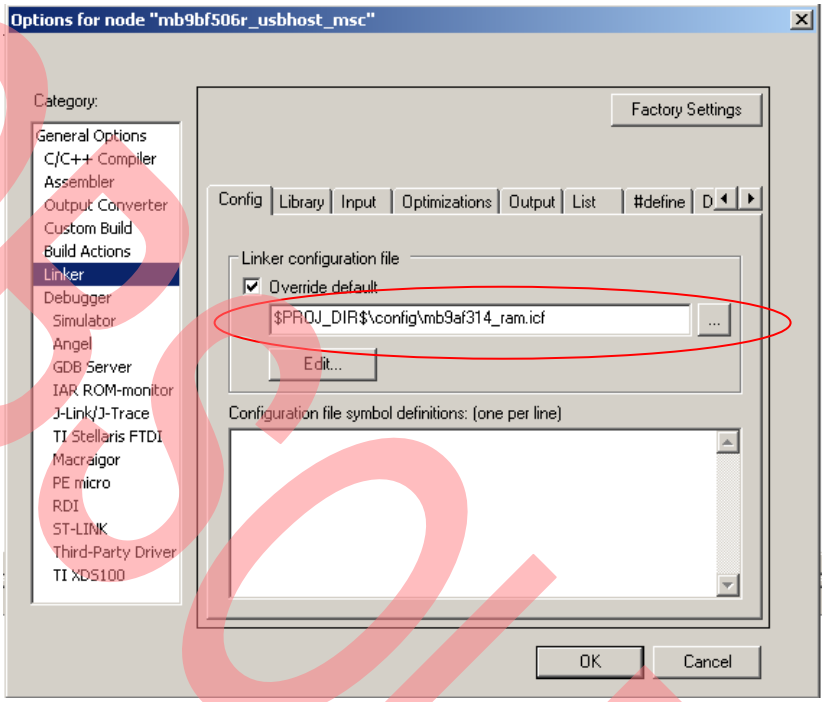


OBSOLETE

- Setting for RAM Debug

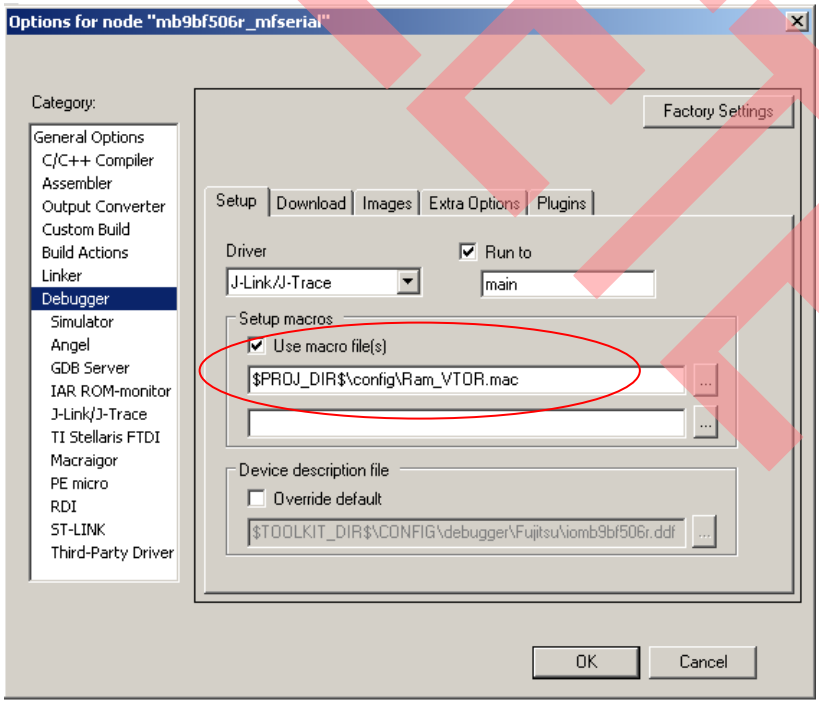
1. Check the configuration file path ($PROJ_DIR$\config\mb9bf314_ram.icf) in Linker table.

Figure 29. ICF File for RAM Debug



2. Select “Use macro files” in “Debugger|Setup” table.

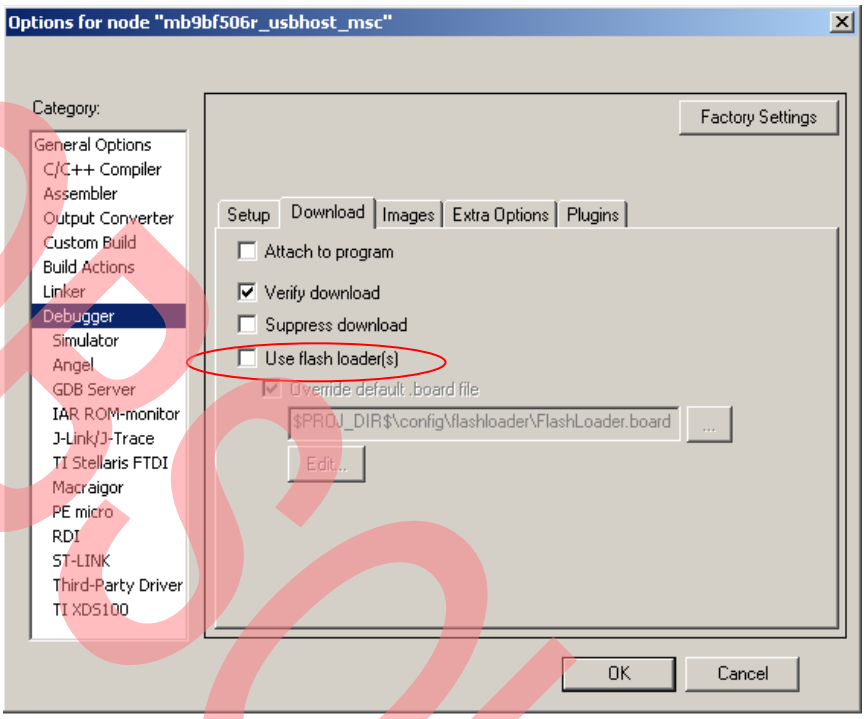
Figure 30. Macro File Enabled



OBSOLETE

3. Don’t use Flash loader file.

Figure 31. Flash Loader File Disabled

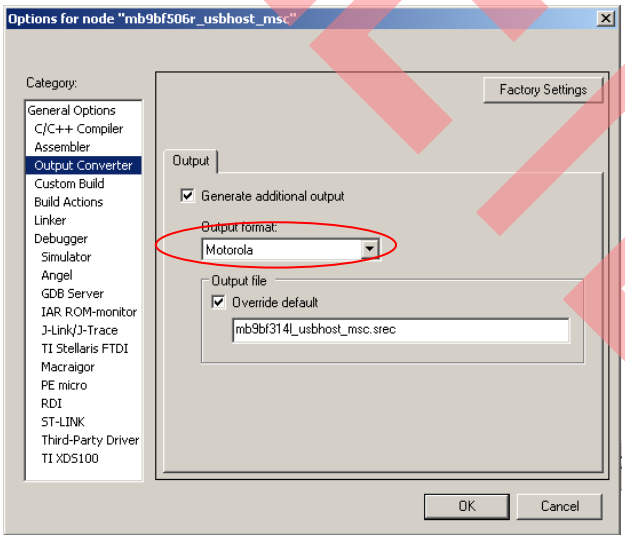


If user need to program the hex file into Flash via USB programmer, a hex file need to be produced first.

- How to Make a HEX File

1. Use Flash debug
2. Select “Generate additional output” in “Output Converter” table.

Figure 32. Hex File Genration in IAR IDE



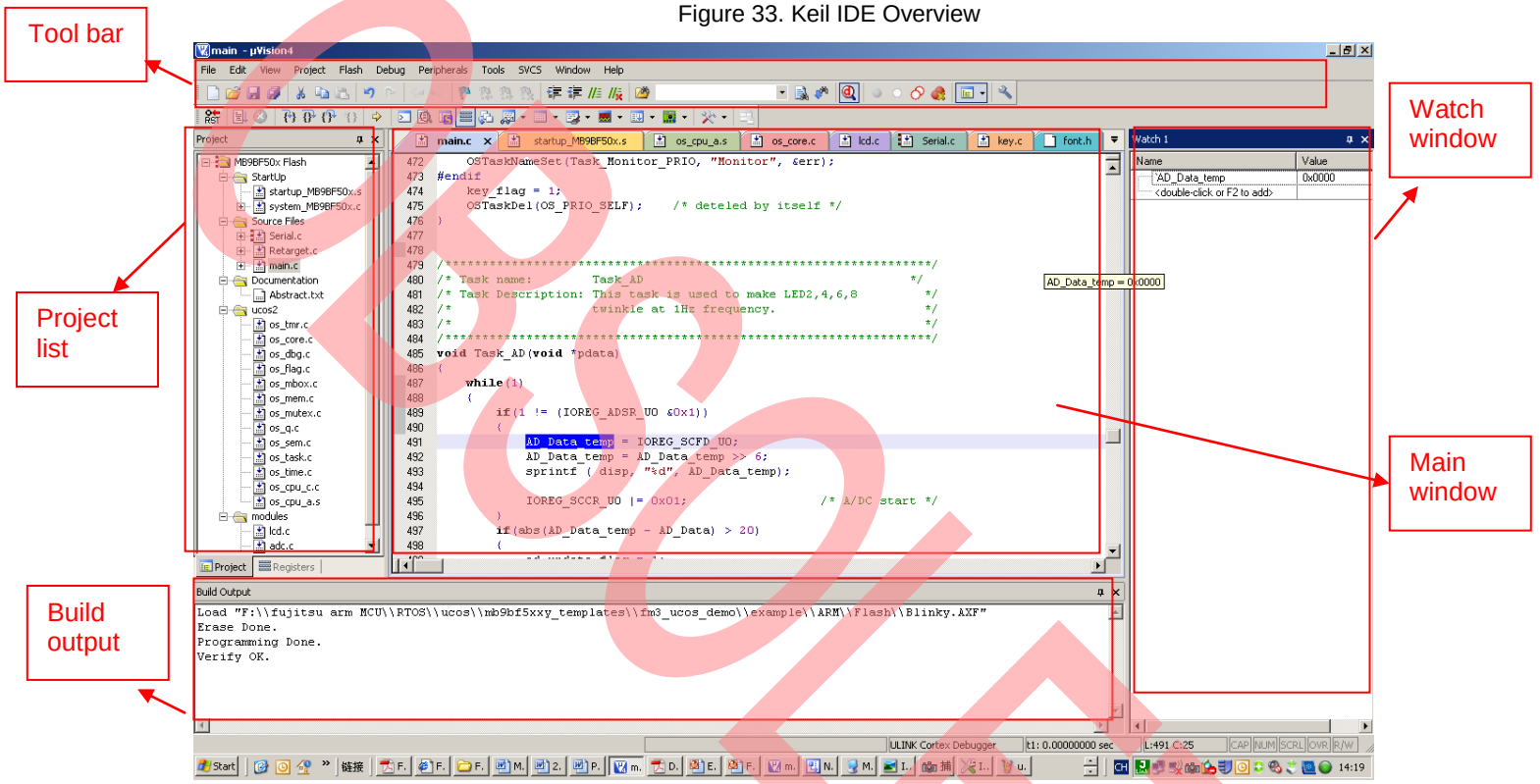
3) User can find the generated file in path (../Debug/Exe)

OBSOLETE

5.2 Debug with U-Link ME in Keil uVision4

The sample code can also be debugged in Keil µVersion4 with U-Link. The following figure shows basic debug window.

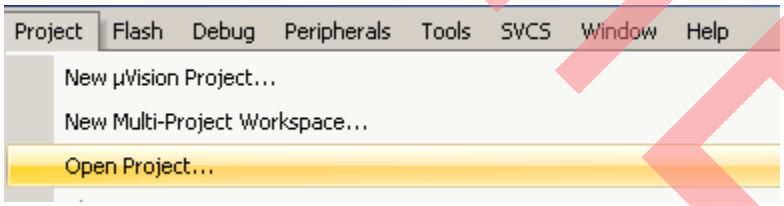
Figure 33. Keil IDE Overview



- Run an Existed Project

1. Open a project by clicking “Project | Open Project…”

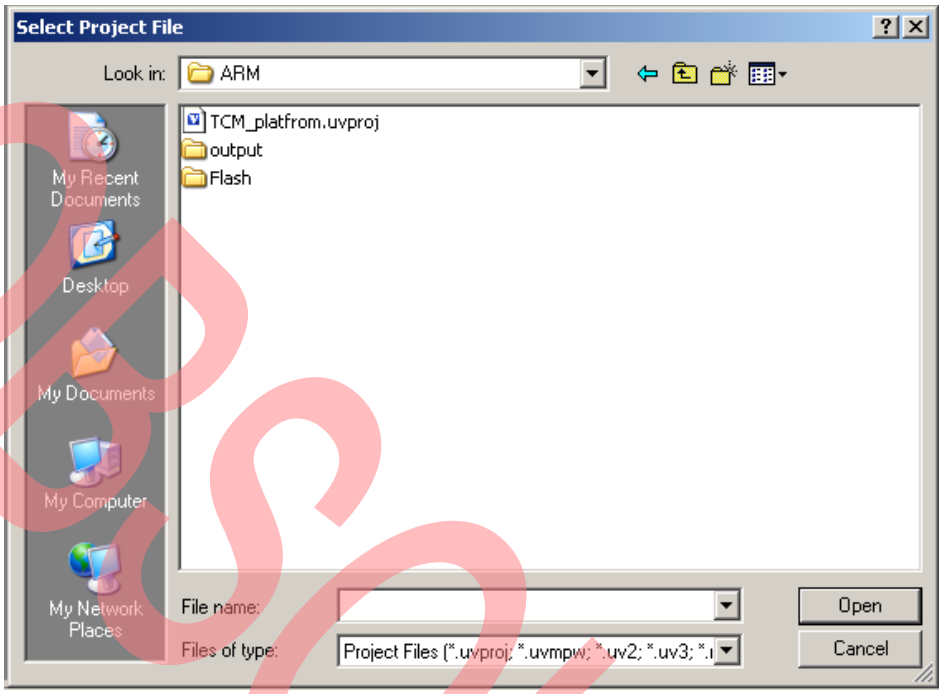
Figure 34. Open a Project



OBSOLETE

2. Select a project (uvproj file)

Figure 35 . Select a Project



3. Rebuild all

Figure 36. Rebuild All Files



4. Start debug

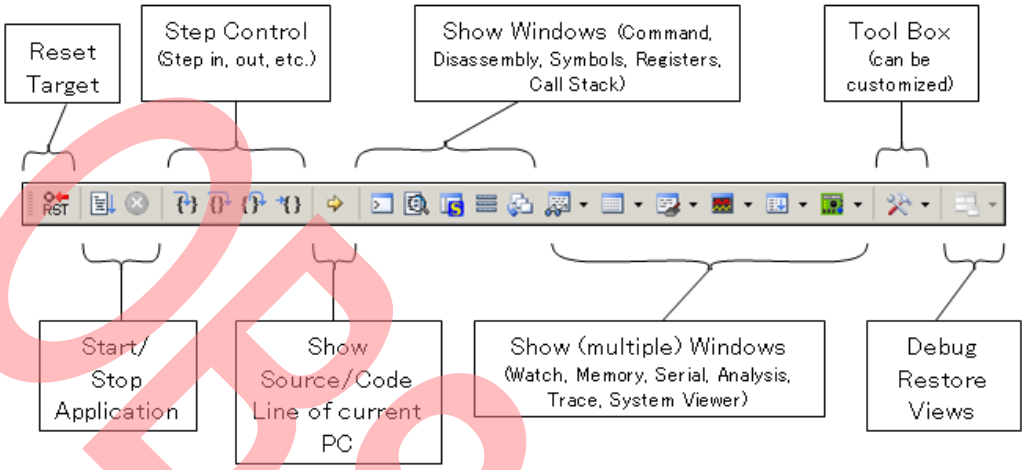
Figure 37. Start Debug



OBSOLETE

5. Use following tool bar to debug

Figure 38. Debug Tool Bar

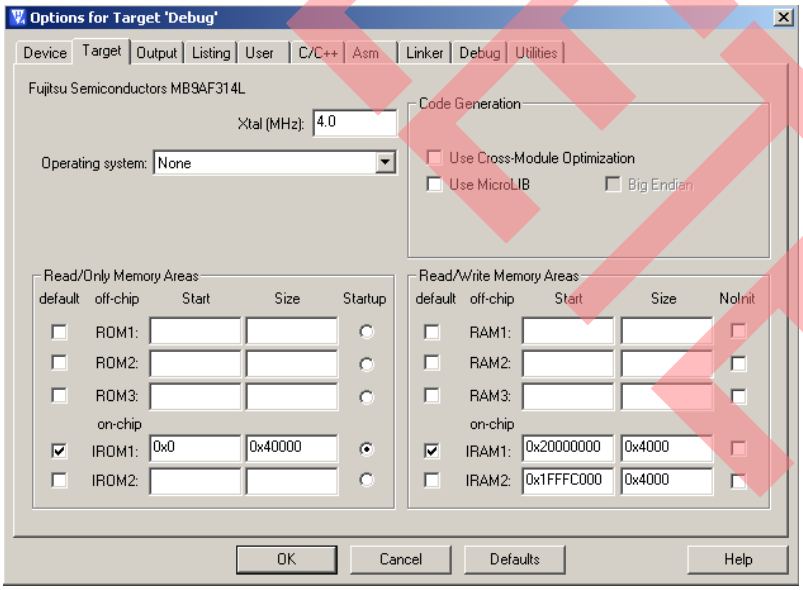


The sample codes support both Flash and RAM debug in Keil uVison 4, if Flash debug is used, the code is programmed into MB9AF314L Flash. If RAM debug is used, the code only runs in RAM area, and after power off, the code will not be stored, but the RAM debug will be faster than Flash debug.

- Setting for Flash Debug

1. Set ROM address in Flash area. (0x00000000-0x0003FFFF)

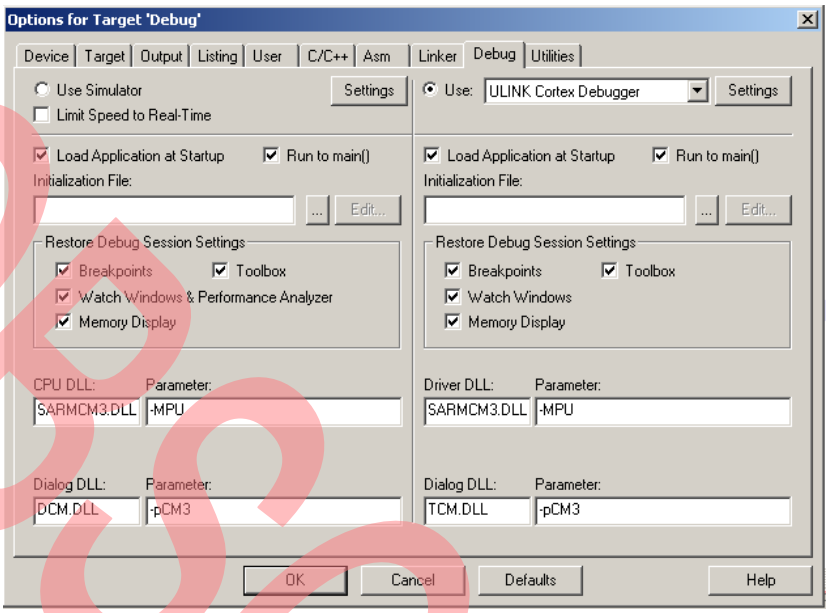
Figure 39. ROM Address Setting for Flash Debug



OBSOLETE

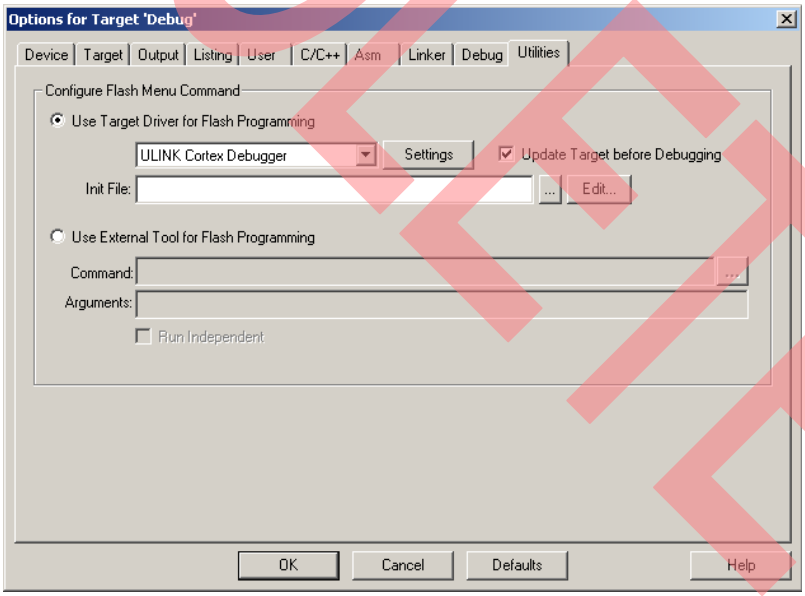
2. Don’t use initialization file.

Figure 40. Don’t use initialization File



3. Check “Update Target before Debugging” checkbox

Figure 41. Select Update Target Before Debugging

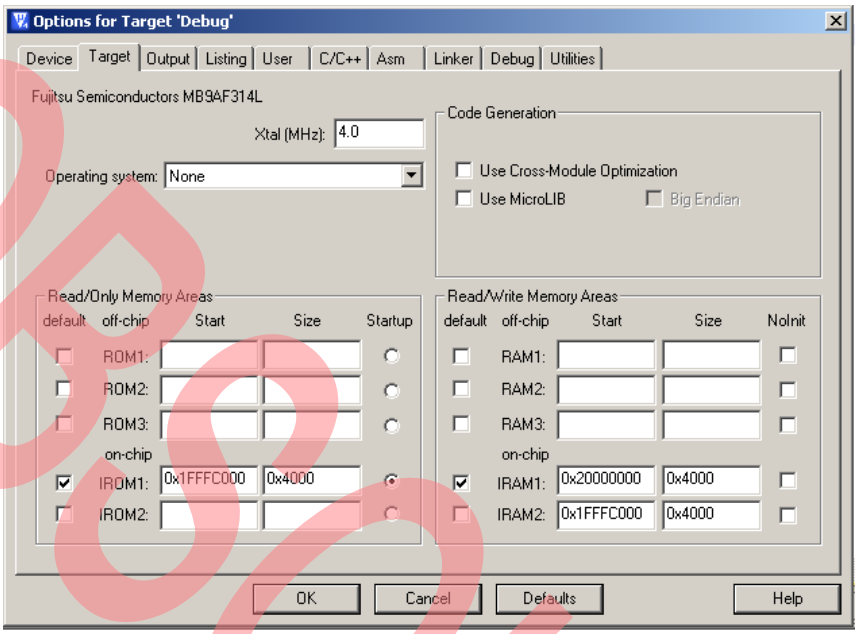


OBSOLETE

Setting for RAM Debug

1. Set ROM address in Code SRAM area. (0x1FFFC000-0x1FFFFFFF)

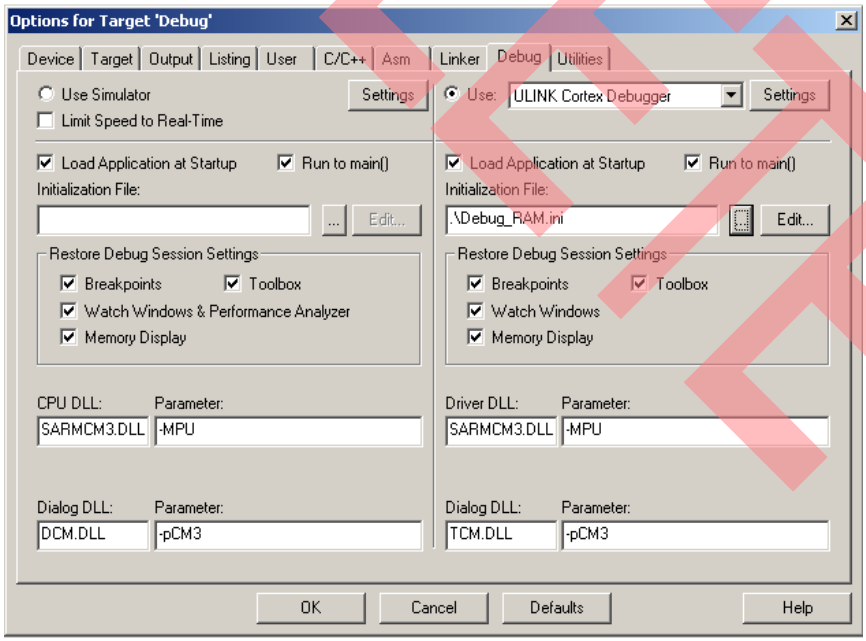
Figure 42. Set RAM Address for RAM Debug



2. Set initialization file path. (..\Debug_RAM.ini)

n

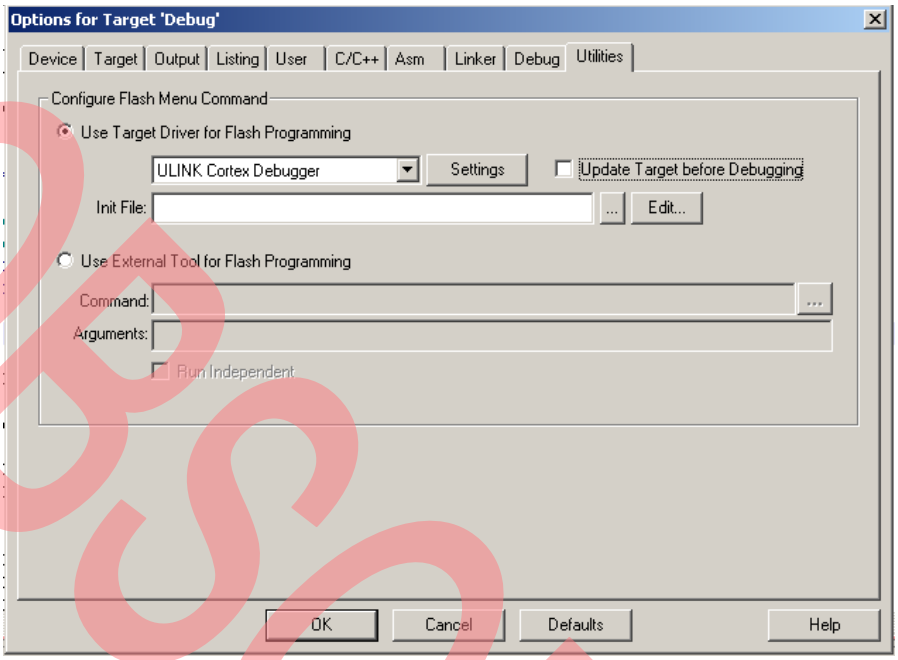
Figure 43. Select Initialization File



OBSOLETE

3) Don't Check "Update Target before Debugging" checkbox

Figure 44. Select Update Target Before Debugging



If user need to program the hex file into Flash via USB programmer, a hex file need to be produced first.

- How to Make a HEX File

1. Check "Create HEX File" checkbox (This file is Intel Format HEX)

Figure 45. HEX File Generation in Keil IDE

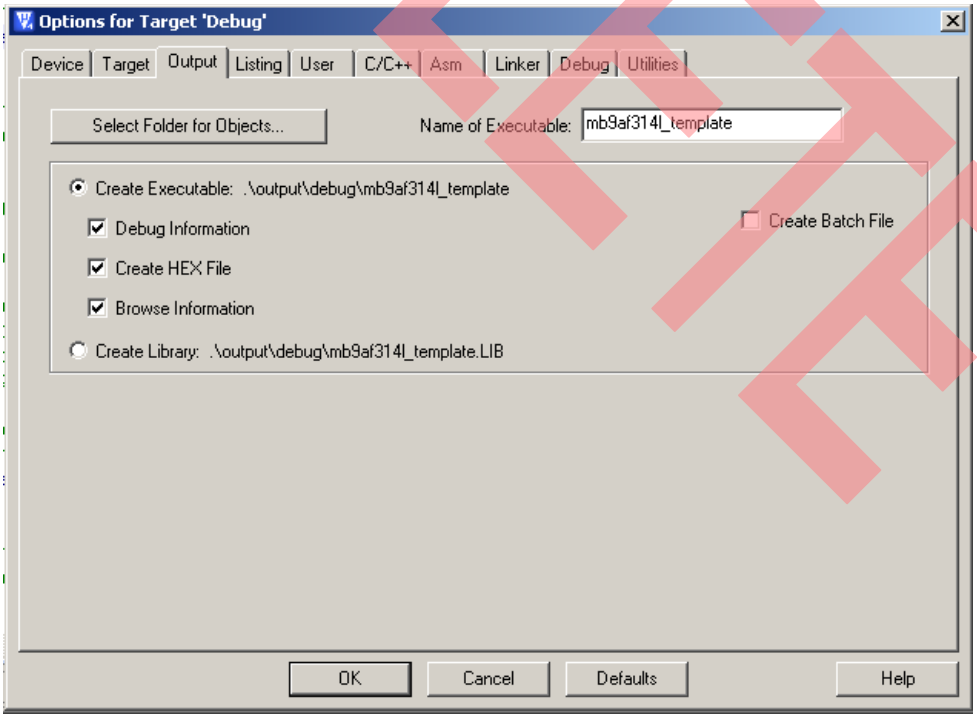


Figure 5-28: HEX File Generation in Keil IDE

OBSOLETE

2) User can find the generated file in path “..\output\debug\”

6 Materials Download

The following materials can be downloaded from below web.

http://www.spansion.com/Products/microcontrollers/32-bit-ARM-Core/fm3/Pages/fm3-app-mb9a310.aspx

- Software
  - Cypress Flash MCU Programmer
  - Cypress Flash USB DIRECT Programmer

- Document
  - FSSDC-9A314-EVB User Manual
  - FSSDC-9A314-EVB Schematic
  - MB9A310 Series Datasheet
  - MB9A310 Series Peripheral Manual
  - MB9A310 Series Flash Programming Manual

- Sample code
  - Blinky
  - usb_func_mouse
  - usb_host_catch_mouse
  - usb_host_msc

- IDE Study Material
  - IAR IDE study material
  - Keil IDE study material

OBSOLETE

A Appendix

Links to Microcontrollers

- http://www.spansion.com/Products/microcontrollers/32-bit-ARM-Core/fm3/Pages/fm3-app-mb9a310.aspx
- FM3 32 Bit Microcontrollers www.cypress.com/cypress-microcontrollers

Links to third party sites

- Cypress plug-in update site: www.spansion.com/Products/microcontrollers/32-bit-ARM-Core/fm3/Pages/overview_32fm3.aspx

OBSOLETE

Document History

Document Title: AN205238 - FM3 Microcontroller MB9A310 Series, FSS MB9AF314L EV-Board

Document Number: 002-05238

Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	—	HUAL	04/20/2012	Initial release.
*A	5272846	HUAL	07/01//2016	Migrated Spansion Application Note from MCU-AN-510042-E-10 to Cypress format. Link to Hardware and Firmware doesn't exist and this AN to be Obsolete.

OBSOLETE

Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturer's representatives, and distributors. To find the office closest to you, visit us at Cypress Locations.

Products

ARM® Cortex® Microcontrollers	cypress.com/arm
Automotive	cypress.com/automotive
Clocks & Buffers	cypress.com/clocks
Interface	cypress.com/interface
Lighting & Power Control	cypress.com/powerpsoc
Memory	cypress.com/memory
PSoC	cypress.com/psoc
Touch Sensing	cypress.com/touch
USB Controllers	cypress.com/usb
Wireless/RF	cypress.com/wireless

PSoC® Solutions

PSoC 1 | PSoC 3 | PSoC 4 | PSoC 5LP

Cypress Developer Community

Forums | Projects | Videos | Blogs | Training | Components

Technical Support

cypress.com/support

OBSOLETE

PSoC is a registered trademark and PSoC Creator is a trademark of Cypress Semiconductor Corporation. All other trademarks or registered trademarks referenced herein are the property of their respective owners.

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709

Phone : 408-943-2600
Fax : 408-943-4730
Website : www.cypress.com

© Cypress Semiconductor Corporation, 20012-2016. This document is the property of Cypress Semiconductor Corporation and its subsidiaries, including Spansion LLC ("Cypress"). This document, including any software or firmware included or referenced in this document ("Software"), is owned by Cypress under the intellectual property laws and treaties of the United States and other countries worldwide. Cypress reserves all rights under such laws and treaties and does not, except as specifically stated in this paragraph, grant any license under its patents, copyrights, trademarks, or other intellectual property rights. If the Software is not accompanied by a license agreement and you do not otherwise have a written agreement with Cypress governing the use of the Software, then Cypress hereby grants you a personal, non-exclusive, nontransferable license (without the right to sublicense) (1) under its copyright rights in the Software (a) for Software provided in source code form, to modify and reproduce the Software solely for use with Cypress hardware products, only internally within your organization, and (b) to distribute the Software in binary code form externally to end users (either directly or indirectly through resellers and distributors), solely for use on Cypress hardware product units, and (2) under those claims of Cypress's patents that are infringed by the Software (as provided by Cypress, unmodified) to make, use, distribute, and import the Software solely for use with Cypress hardware products. Any other use, reproduction, modification, translation, or compilation of the Software is prohibited.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS DOCUMENT OR ANY SOFTWARE OR ACCOMPANYING HARDWARE, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. To the extent permitted by applicable law, Cypress reserves the right to make changes to this document without further notice. Cypress does not assume any liability arising out of the application or use of any product or circuit described in this document. Any information provided in this document, including any sample design information or programming code, is provided only for reference purposes. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. Cypress products are not designed, intended, or authorized for use as critical components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or system could cause personal injury, death, or property damage ("Unintended Uses"). A critical component is any component of a device or system whose failure to perform can be reasonably expected to cause the failure of the device or system, or to affect its safety or effectiveness. Cypress is not liable, in whole or in part, and you shall and hereby do release Cypress from any claim, damage, or other liability arising from or related to all Unintended Uses of Cypress products. You shall indemnify and hold Cypress harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of Cypress products. Cypress, the Cypress logo, Spansion, the Spansion logo, and combinations thereof, PSoC, CapSense, EZ-USB, F-RAM, and Traveo are trademarks or registered trademarks of Cypress in the United States and other countries. For a more complete list of Cypress trademarks, visit cypress.com. Other names and brands may be claimed as property of their respective owners.