



THIS SPEC IS OBSOLETE

Spec No: 002-09873

Spec Title: Starter kit User Guide SK-FM3-48PMC-  
MB9BF524K SK-FM3-48PMC-9BF524K-JL

Replaced by: None



# Starter kit User Guide

SK-FM3-48PMC-MB9BF524K

SK-FM3-48PMC-9BF524K-JL

Hardware V1.1 / Documentation V1.1

# Warranty and Disclaimer

▪The use of the deliverables (deliverables shall include, but not limited to, software, application examples, hardware, target boards, evaluation boards, starter kits, schematics, engineering drawings, etc.) is subject to the terms and conditions of Spansion LLC and its affiliates ("SPANSION") as set out below and in (i) the terms and conditions of the License Agreement and/or the Evaluation Agreement and/or any other agreement under which deliverables have been delivered, (ii) the technical descriptions and (iii) all accompanying written materials.

▪1. Please note that the deliverables are intended for and must only be used for test applications in an evaluation laboratory environment.

▪2. The software deliverables are provided on an as-is basis without charge and are subject to alterations. It is the user's obligation to fully test the software in its environment and to verify its functionality, qualification and compliance with component specifications.

▪3. Regarding hardware deliverables, the following limited warranty shall apply:

▪Except as otherwise provided in the following paragraphs, for a period of one (1) year from date of shipment to customer ("Warranty Period"), SPANSION warrants the hardware deliverables to be free of defects in material and workmanship, and (ii) conform to SPANSION applicable data sheet specifications (available at [www.spansion.com](http://www.spansion.com) or upon request).

▪This warranty does not extend beyond the first purchaser of the deliverables. The liability of SPANSION under this warranty is limited, at SPANSION's option, solely to repair the defective deliverable, replacement deliverable, or to make an appropriate credit adjustment or refund in an amount not to exceed the original purchase price actually paid for the deliverable returned to SPANSION. SPANSION's warranty obligations are conditioned upon the following: (a) SPANSION is promptly notified in writing by customer during the applicable warranty period of any defect or nonconformance; (b) customer obtains authorization from SPANSION to return the defective deliverable; (c) the defective deliverable is returned to SPANSION in accordance with SPANSION'S shipping instructions set forth below; and (d) SPANSION'S examination of such deliverable discloses to its satisfaction that any defect or nonconformance actually existed and was not caused by improper use or misuse, failure to follow data sheet specifications for the deliverable, abuse, negligence, improper installation, accident, loss or damage in transit, or unauthorized repair or alteration by a person other than SPANSION. SPANSION shall ship such defective deliverable to SPANSION via SPANSION'S carrier, collect. Risk of loss will transfer to SPANSION when the defective deliverable is provided to SPANSION'S carrier. If customer fails to adhere to these warranty returns guidelines, customer shall assume all risk of loss and shall pay for all freight to SPANSION'S specified location. This warranty shall not apply to deliverables that have been repaired or altered, except those which have been repaired or altered by SPANSION. The aforementioned provisions do not extend the original warranty period of any deliverable that has either been repaired or replaced by Seller.

▪THESE WARRANTIES ARE IN LIEU OF ALL OTHER WARRANTIES, EXPRESSED, IMPLIED OR STATUTORY, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, AND ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, SPECIFICATION OR AGREEMENT. SPANSION NEITHER ASSUMES NOR AUTHORIZES ANY OTHER PERSON TO ASSUME FOR IT ANY OTHER LIABILITIES. THE FOREGOING CONSTITUTES CUSTOMER'S SOLE AND EXCLUSIVE REMEDY FOR THE FURNISHING OF DEFECTIVE OR NONCONFORMING DELIVERABLES.

▪4. The following limitation of liability shall apply for all deliverables

▪EXCEPT FOR DAMAGES FOR BODILY INJURY OR DEATH, SPANSION SHALL NOT BE LIABLE FOR ANY INDIRECT, INCIDENTAL, SPECIAL, RELIANCE, OR CONSEQUENTIAL DAMAGES, AND/OR PUNITIVE, OR EXEMPLARY DAMAGES, WHETHER ANY SUCH DAMAGES ARE BASED ON CONTRACT, TORT OR ANY OTHER LEGAL THEORY, AND WHETHER OR NOT SPANSION HAS BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES, AND NOTWITHSTANDING ANY FAILURE OF ESSENTIAL PURPOSE OF ANY LIMITED REMEDY.

▪REGARDLESS OF THE BASIS ON WHICH CUSTOMER IS ENTITLED TO CLAIM DAMAGES FROM SPANSION (INCLUDING FUNDAMENTAL BREACH, NEGLIGENCE, MISREPRESENTATION, OR OTHER CONTRACT OR TORT CLAIM), SPANSION ENTIRE LIABILITY IN ANY CALENDAR YEAR, REGARDLESS OF THE NUMBER OF CLAIMS, SHALL NOT EXCEED FIFTY PERCENT (50%) OF THE TOTAL AMOUNT PAID BY CUSTOMER TO SPANSION FOR THE DELIVERABLES SOLD IN SUCH CALENDAR YEAR.

▪5. Should one of the above stipulations be or become invalid and/or unenforceable, the remaining stipulations shall stay in full effect.

▪6. The contents of this document are subject to change by SPANSION without a prior notice, thus contact SPANSION about the latest one.

▪This board and its deliverables must only be used for test applications in an evaluation laboratory environment.

## Information about this PDF document

- For your convenience this user guide includes external links to simplify installing of drivers, software utilities, and quick jump to documentation.
- Some PDF viewer do not allow access to external content because of security reasons.
- A viewer called “PDF XChange” is provided in the software package of this starter kit. It’s use is free of charge and no additional installation is required.
- Launching “start.bat” opens this user guide in the PDF XChange viewer.
- Please ensure you have copied the complete software package related to this starter kit in order to use and run the link examples given on the next pages.
- Please contact the [Spansion Support](#) in case of any question.

# Overview

## ■ Introduction

- [About the SK-FM3-48PMC-MB9BF524K](#)
- [SK-FM3-48PMC-MB9BF524K content](#)
- [SK-FM3-48PMC-9BF524K-JL content](#)
- [Test it](#)
- [The Hardware](#)
- [The Software](#)

## ■ Try yourself

- [Software examples](#)
- [Program download](#)
- [IAR-Embedded Workbench](#)
- [KEIL  \$\mu\$ Vision](#)
- [Solutions](#)

## ■ [Workshops](#), [Contacts](#) & [More](#)



## ■ [Additional documents](#)

- [Schematic](#)
- [Factsheet](#)
- [Data sheet MB9B520M Series](#)
- [Peripheral Manual](#) and [Errata](#)
- [Peripheral Manual \(Timer Part\)](#) and [E](#)
- [Peripheral Manual \(Analog Part\)](#) and [E](#)
- [Peripheral Manual \(Communication F](#)
- [Cortex M3 Technical Reference Manu](#)
- [Flash Programming Manual](#)

Please visit [www.spansion.com](http://www.spansion.com) to find la  
of the above mentioned documents.

## About

The SK-FM3-48PMC-MB9BF524K is available in two versions:

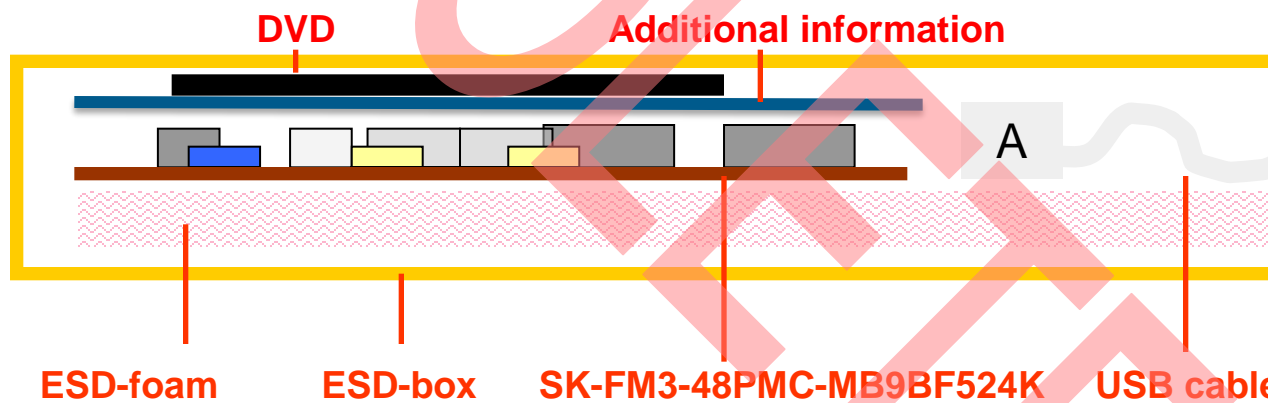
- The SK-FM3-48PMC-MB9BF524K includes a low-cost evaluation board based on the Spansion FM3 microcontroller MB9B520M Series
- SK-FM3-48PMC-9BF524K-JL includes a low-cost evaluation board based on the Spansion FM3 microcontroller MB9B520M Series and a JTAG adapter J-Link
- The MB9B520M Series includes the following features:
  - Up to 288 KByte Dual Operation Flash Memory
  - Up to 64 KByte RAM
  - Up to 2 CAN controller 2.0A/B
  - Up to 8 LIN-USART-I<sup>2</sup>C interfaces
  - USB-Host/-Device interface
  - Timers (ICUs, OCUs, PPGs, others)
  - Two 12 Bit ADCs, up to 26 channels
  - External interrupts

## Features

- Features of the SK-FM3-48PMC-MB9BF524K board:
  - Microcontroller MB9BF524K
  - 1x UART-Transceiver (SUB-D9 connector)
  - 1x USB to serial converter (Type-B connector)
  - 1x High-speed CAN-Transceiver (SUB-D9 connector)
  - 1x USB-Host (Type-A connector)
  - 1x USB-Device (Type-B connector)
  - 1x LED-Display (7-Segment)
  - 2x 'User'-button
  - 1x 'Reset'-button, 'Reset'-LED
  - JTAG-Interface on a 20 pin-header
  - FMtouch connector interface for [software touch solutions](#)
  - TSC-Interface to connect for example the Spansion SK-TSC-1127S-SB
  - All 48 pins routed to pin-header
  - On-board 5V and 3V voltage regulators, 'Power'-LED
  - Power supply via USB (UART'B'), USB-Device, JTAG or external with a 8V to 12V connector

## SK-FM3-48PMC-MB9BF524K content

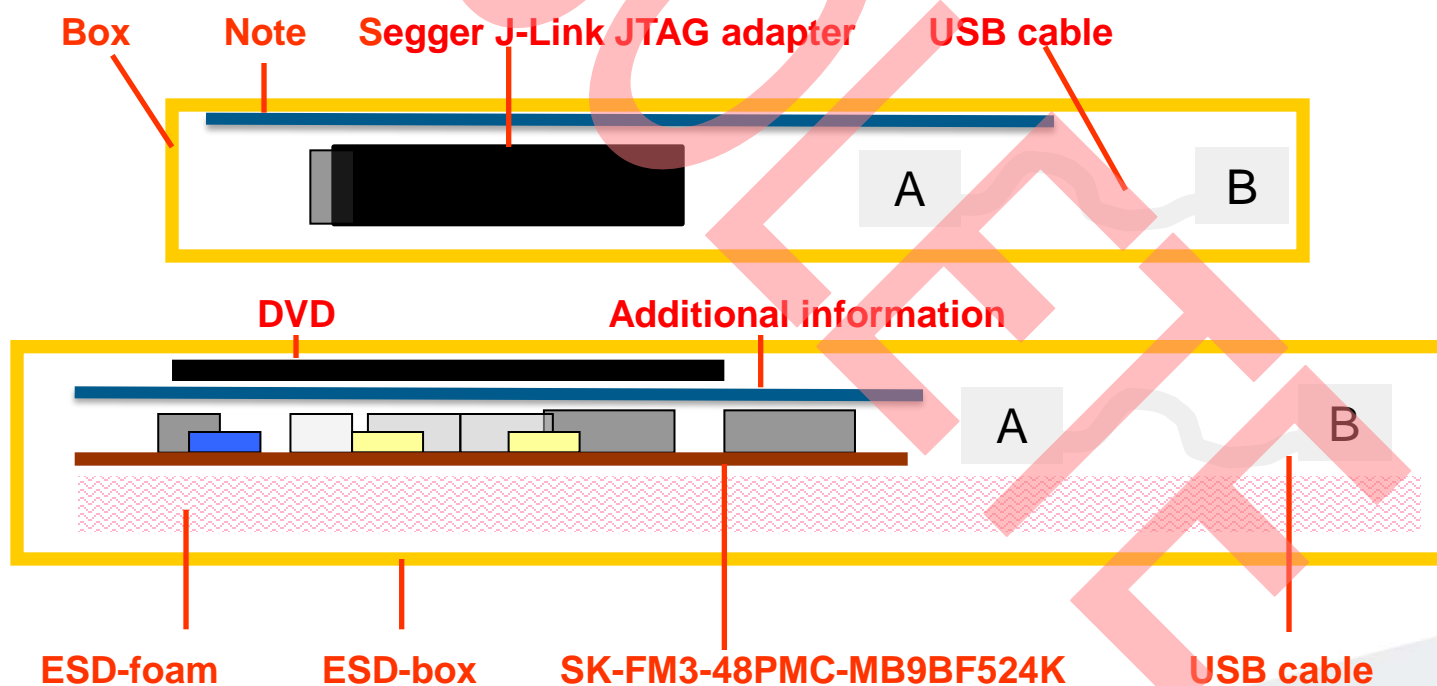
- The SK-FM3-48PMC-MB9BF524K contains
  - SK-FM3-48PMC-MB9BF524K evaluation board with MB9BF524K
  - USB cable
  - DVD: Documentation, USB driver, Software examples, Programme





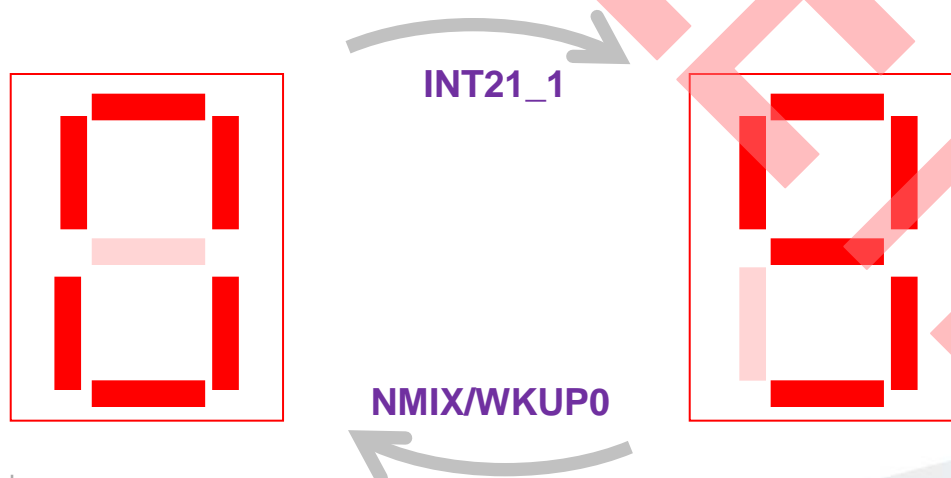
## SK-FM3-48PMC-9BF524K-JL content

- The SK-FM3-48PMC-9BF524K-JL contains
  - SK-FM3-48PMC-MB9BF524K evaluation board with MB9BF524K
  - USB cable
  - DVD: Documentation, USB driver, Software examples, Programme
  - Segger J-Link JTAG adapter incl. USB cable



## Test it

- The microcontroller on the SK-FM3-48PMC-MB9BF524K is already preprogrammed with a simple application.
  - Connect the SK-FM3-48PMC-MB9BF524K via USB (X5) with the PC
  - [Install the USB driver from the DVD](#)
  - Press the ‚Reset‘- Button
  - The SK-FM3-48PMC-MB9BF524K will automatically start counting
  - The count direction can be changed by pressing the key buttons



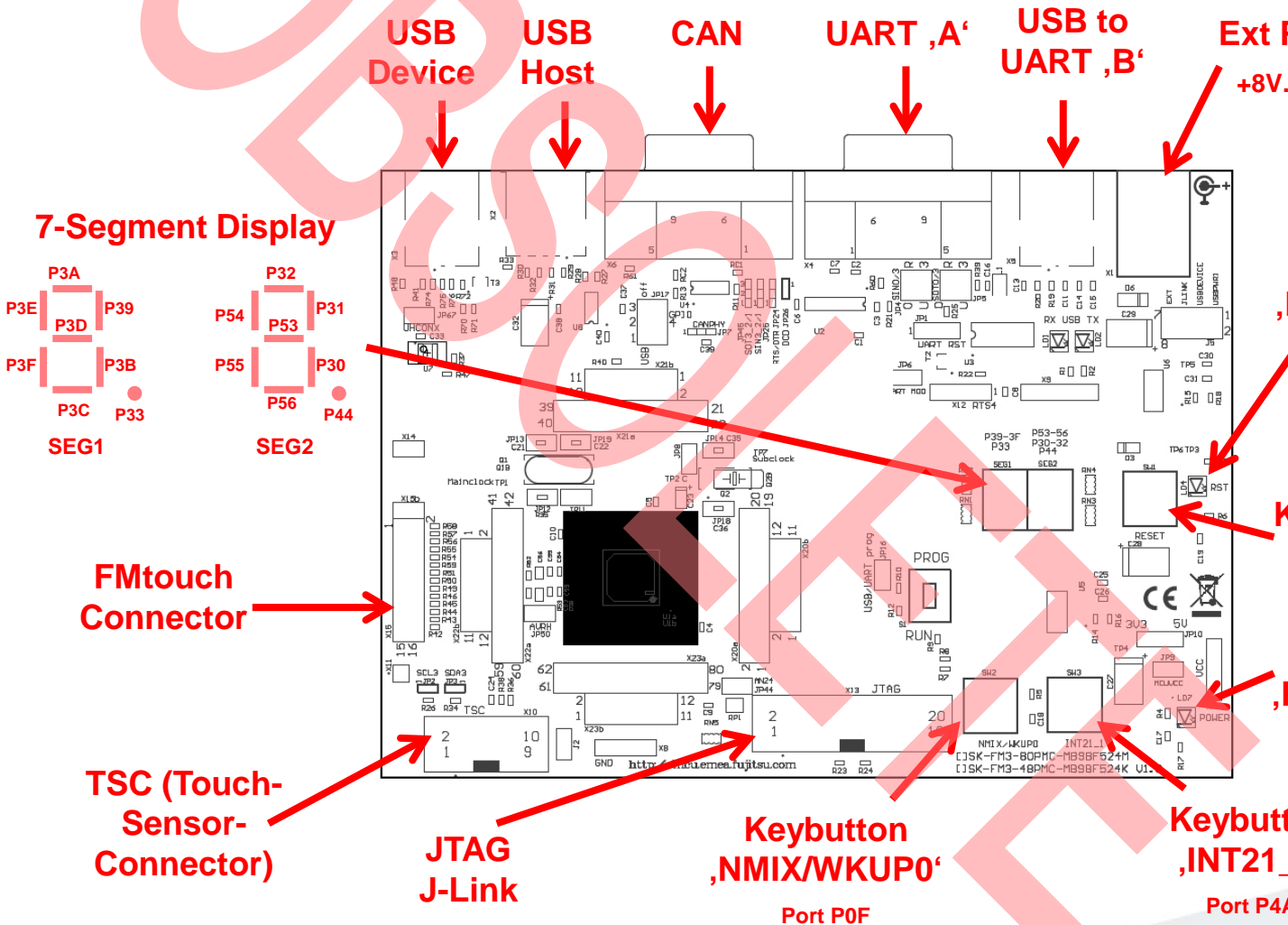
## Test it

**Congratulations!**

- You finished successfully the first test
- Now you will get more details about the SK-FM3-48PMC-MB9BF524K board
- You will learn more about
  - The on-board features
  - How to program the Flash
  - How to start with IAR-Embedded-Workbench and KEIL  $\mu$ Vision

# The Hardware

- Main features



# The Hardware

## ■ The jumpers

### JP1: UART-Reset

1-2: DTR-Signal of the UART connector is connected to the MCU reset-pin.

2-3: DTR-Signal of the USB connector is connected to the MCU reset-pin.

Some terminal-programs, e.g. Spansion's Skwizard, allow to reset the evaluation board by using the DTR-Signal.

### JP6: MD0 selection

Close this jumper to control the MD0 level by the RTS signal of the USB interface

### S1: Mode selection

PROG: Program-mode

RUN: Run-mode

### JP10: 5V / 3.3V

1-2: 5V supply is used

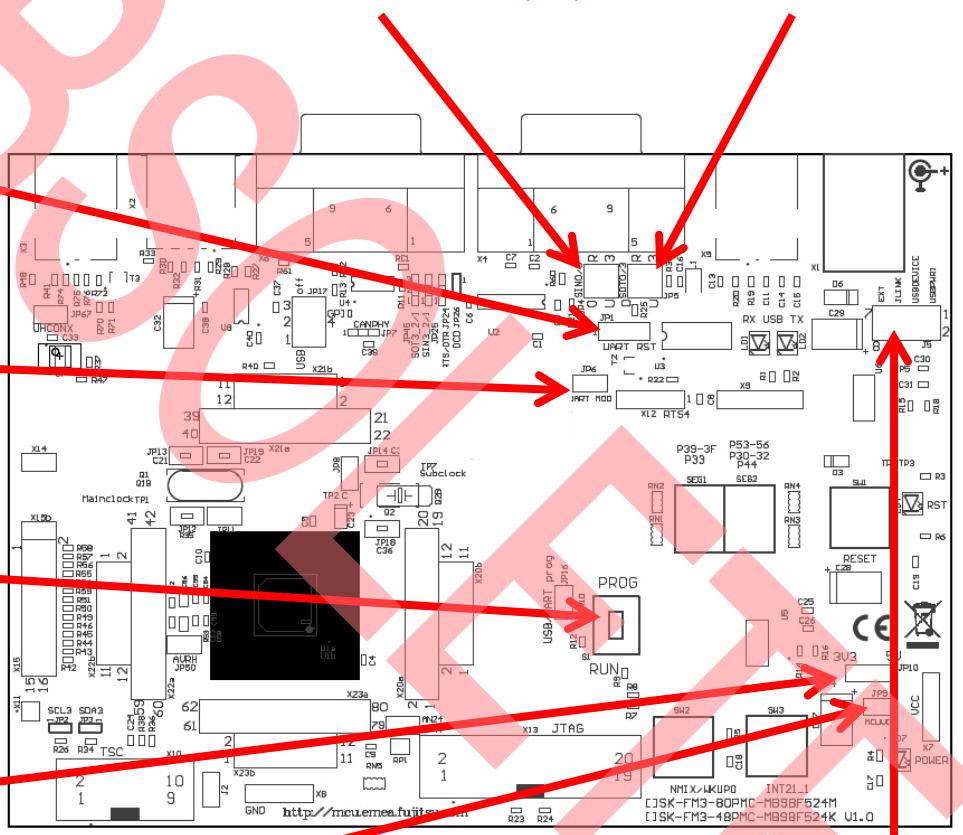
2-3: 3.3V supply is used

### JP4: UART RX select

R-0: UART0=UART'A' / U-4: UART4=UART'B' (USB) R-0: UART0=UART'A' / U-4: UART4=UART

R-3: UART3=UART'A' / U-0: UART0=UART'B' (USB) R-3: UART3=UART'A' / U-0: UART0=UART

### JP5: UART TX select



### JP9: MCU Vcc

This jumper can be used to measure the current consumption of the MCU

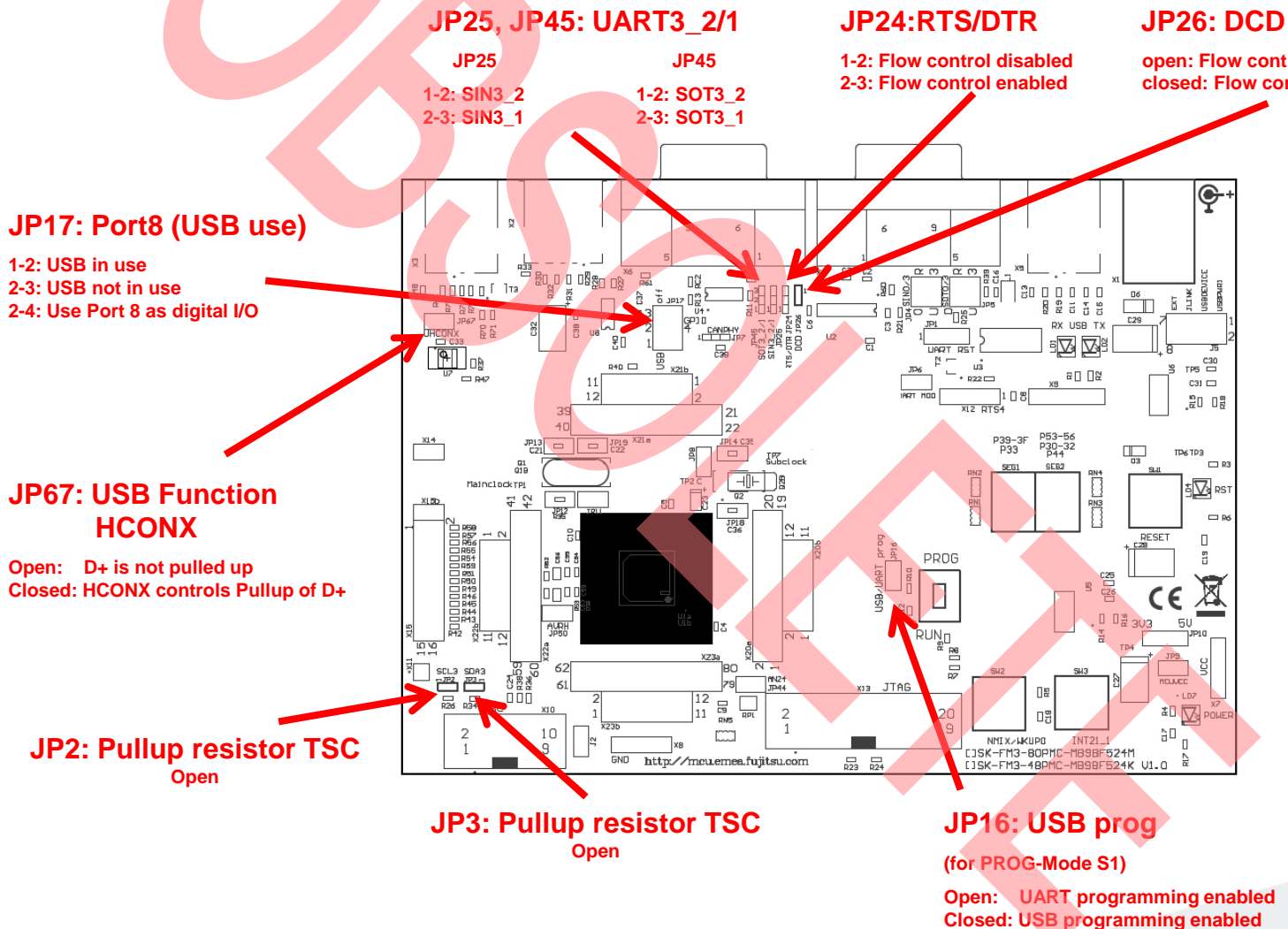
### J5: Power Supply

1-2: USB (UART ,B') supply 3-4: USB Device

5-6: JLINK supply 7-8: External s

# The Hardware

## ■ The jumpers

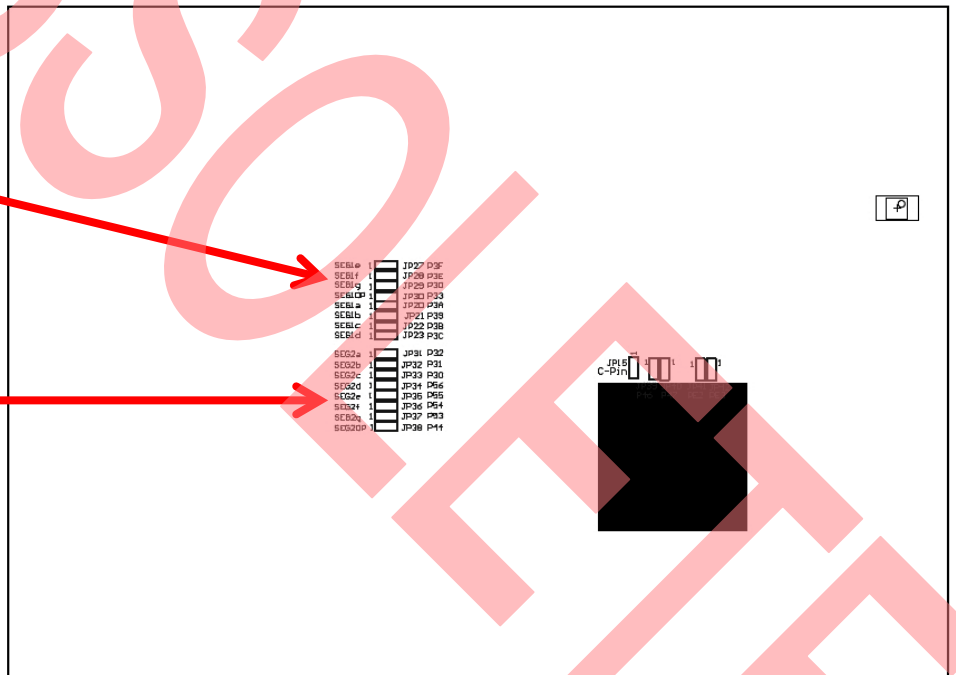


# The Hardware

- The jumpers(back)

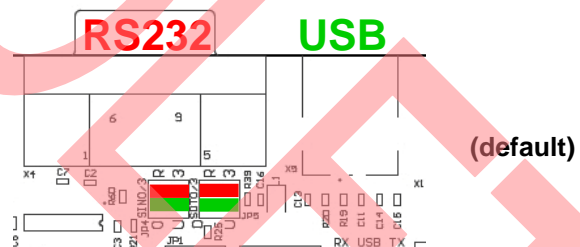
**JP20-JP23, JP27-JP30: SEG1**  
Closed: SEG1a- SEG1DP active

**JP31-JP38: SEG2**  
Closed: SEG2a- SEG2DP active

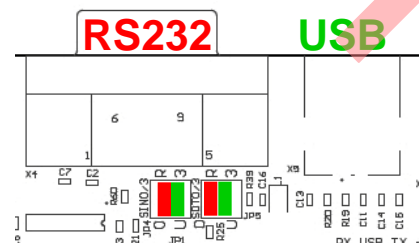


## The Hardware

- JP4, JP5 : UART selection
  - UART0 and UART3 of the microcontroller can be used together with typical RS232 SUB-D9 connector and a serial/USB converter
  - The jumpers JP4 and JP5 routes the channel to the connector
  - UART0 = USB-connector (X5), UART3 = Sub-D9 (X4) (default)
    - ◆ Setting of Jumper JP4 and JP5: U-0 / R-3



- UART0 = Sub-D9 (X4), UART3 = USB-connector (X5)
  - ◆ Setting of Jumper JP4 and JP5: U-3 / R-0

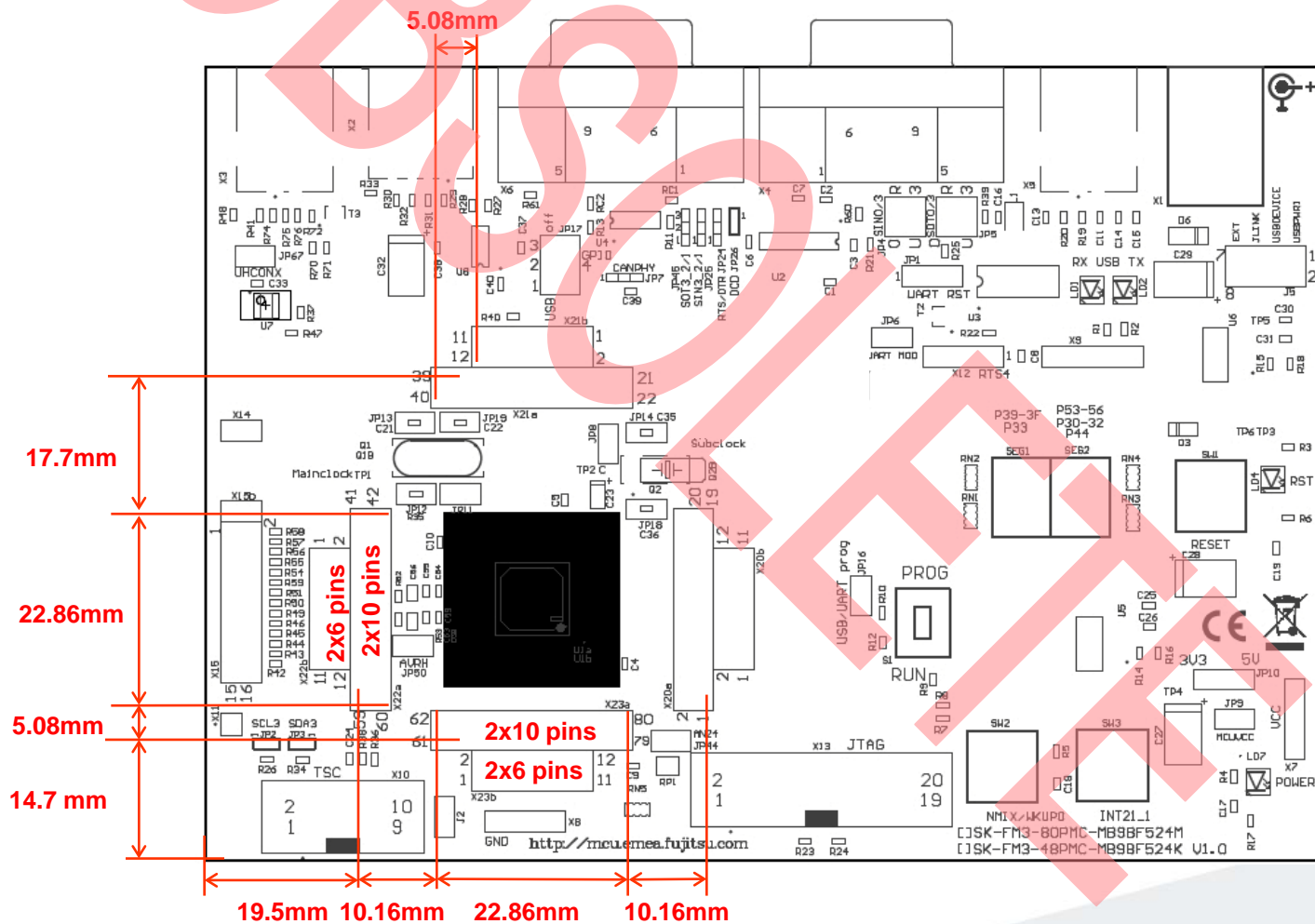




# The Hardware

## ■ Extension headers X20-X23

- Standard 0.1" / 2.54mm grid for use with prototype boards



## The Hardware

- The microcontroller pins

Board Function	Pin Function	MB9BF5
MCUVCC	VCC	1
(SW-Touch_4 , Connector X15)/(TSC_7, Connector X10)/ UART3_1	P50/INT00_0 /AIN0_2/SIN3_1/AN22	2
(SW-Touch_3, Connector X15)/(TSC _2, Connector X10)/ UART3_1	INT01_0/BIN0_2/SOT3_1/AN23	3
(SW-Touch_2, Connector X15)/ (TSC _4, Connector X10)	P52/INT02_0/ZIN0_2/SCK3_1/AN24	4
SEG1b	P39/DTT10X_0/INT06_0/ADTG_2	5
SEG1a	P3A/RTO00_0/TIOA0_1/INT07_0/SUBOUT_2/RTCCO_2	6
SEG1c	P3B/RTO01_0/TIOA1_1	7
SEG1d	P3C/RTO02_0 /TIOA2_1/INT18_2	8
SEG1g	P3D/RTO03_0/TIOA3_1	9
SEG1f	P3E/RTO04_0/TIOA4_1/INT19_2	10

## The Hardware

- The microcontroller pins

Board Function	Pin Function	MB9BF5
SEG1e	P3F/RTO05_0/TIOA5_1	11
GND	VSS	12
C-Pin	C	13
MCUVCC	VCC	14
(32.768KHz Crystal)	P46/X0A	15
(32.768KHz Crystal)	P47/X1A	16
Key button- ,Reset'	INITX	17
	P49/TIOB0_0/INT20_1/DA0_0/SOT3_2/AIN0_1	18
Key button ,INT'	P4A/TIOB1_0/INT21_1/DA1_0/SCK3_2/BIN0_1	19
GND	MD1/PE0	20

## The Hardware

- The microcontroller pins

Board Function	Pin Function	MB9BF5
Mode-Switch ,S1'	MD0	21
4MHz Crystal	X0/PE2	22
4MHz Crystal	X1/PE3	23
GND	VSS	24
USB Switch Device/Host	P10/AN00	25
CAN RX	P11/AN01/SIN1_1/INT02_1/RX1_2/FRCK0_2/WKUP1	26
CAN TX	P12/AN02/SOT1_1/TX1_2/IC00_2	27
GND	AVSS	28
USB Current limitation	AN04/INT03_1/IC02_2/SIN0_1	29
USB Host Power enable	P15/AN05/IC03_2/SOT0_1/INT14_0	30

## The Hardware

- The microcontroller pins

Board Function	Pin Function	MB9BF5
AVCC	AVCC	31
AVRH	AVRH	32
GND	AVRL	33
(SW-Touch _13, Connector X15)	SCK0_0/TIOA7_1/AN12	34
UART0 TX	P22/SOT0_0/TIOB7_1/ZIN1_1/AN13	35
UART0 RX	P21/SIN0_0/INT06_1/WKUP2/BIN1_1/AN14	36
(JTAG TRSTX, Connector X13)	P00/TRSTX	37
(JTAG TCK, Connector X13)	P01/TCK/SWCLK	38
(JTAG TDI, Connector X13)	P02/TDI	39
(JTAG TMS, Connector X13)	P03/TMS/SWDIO	40

## The Hardware

- The microcontroller pins

Board Function	Pin Function	MB9BF5
(JTAG TDO, Connector X13)	P04/TDO/SWO	41
NMIX/ WKUP	P0F/NMIX/SUBOUT_0/CROUT_1/RTCCO_0/WKUP0/AN18	42
USB UHCONX	P61/SOT5_0/TIOB2_2/UHCONX/DTTI0X_2/AN20	43
Mode Switch ,S1'	P60/SIN5_0/TIOA2_2/INT15_1/WKUP3/IGTRG_1/AN21	44
USB-power supply	USBVCC	45
USB Data -	P80/UDM0/INT16_1	46
USB Data +	P81/UDP0/INT17_1	47
GND	VSS	48

## The Software

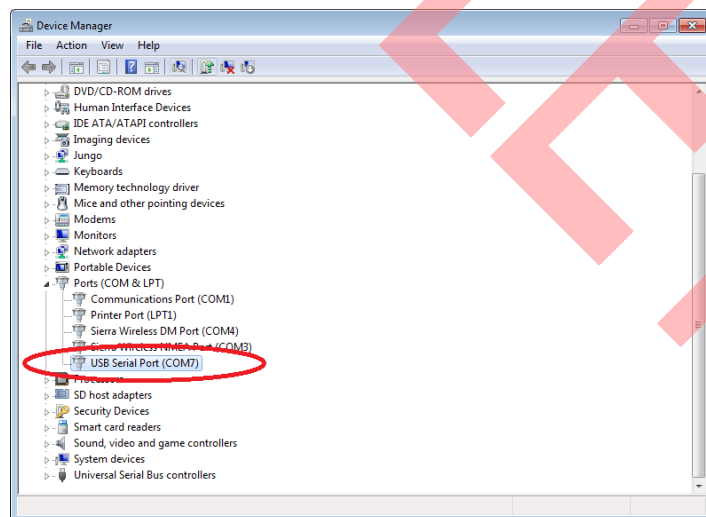
- The SK-FM3-48PMC-MB9BF524K DVD includes the following software:
  - MCU Flash programming tools
    - FLASH MCU Programmer for FM3
    - FLASH USB DIRECT Programmer
  - USB driver for on-board USB-to-RS232 converter
  - The terminal program ,Serial Port Viewer‘
  - The USB configuration tool ,USB Wizard‘
  - Software examples for the SK-FM3-48PMC-MB9BF524K
- Please check our dedicated microcontroller website:

[www.spansion.com](http://www.spansion.com)

- for updates of the Flash programmer tool, utilities and examples
- for data sheets, hardware manuals, application notes, etc.

## Installation of the USB-driver

- Install the USB driver from the [DVD](#) with administrator privileges
- Start the Device Manager of the Windows Control Panel
  - START -> Settings -> Control Panel
  - Control Panel -> System -> Hardware -> Device Manager
- Check 'Ports' for the assigned virtual COM-port number
  - USB Serial Port (e.g.: COM7)



- Ready!



## Tools and Software Examples

- **Serial Port Viewer**
  - Free of charge terminal program, [Start installation](#)
- **USB Wizard**
  - Free of charge USB configuration tool, [Start installation](#)
- Following examples are provided with SK-FM3-48PMC-MB9BF524K f Embedded Workbench V6 and KEIL  $\mu$ Vision4:
  - [mb9bf52xk\\_template](#) ,Empty' project as base for user applications
  - [mb9bf52xk\\_adc\\_dvm](#) Digital Voltage Meter based on the A/D-converter
  - [mb9bf52xk\\_can\\_uart\\_terminal](#) Simple CAN example (CAN0)
  - [mb9bf52xk\\_ioport\\_counter](#) Counts from 0 to 99 on the 7-segment Display
  - Further examples on [DVD](#) and on our website

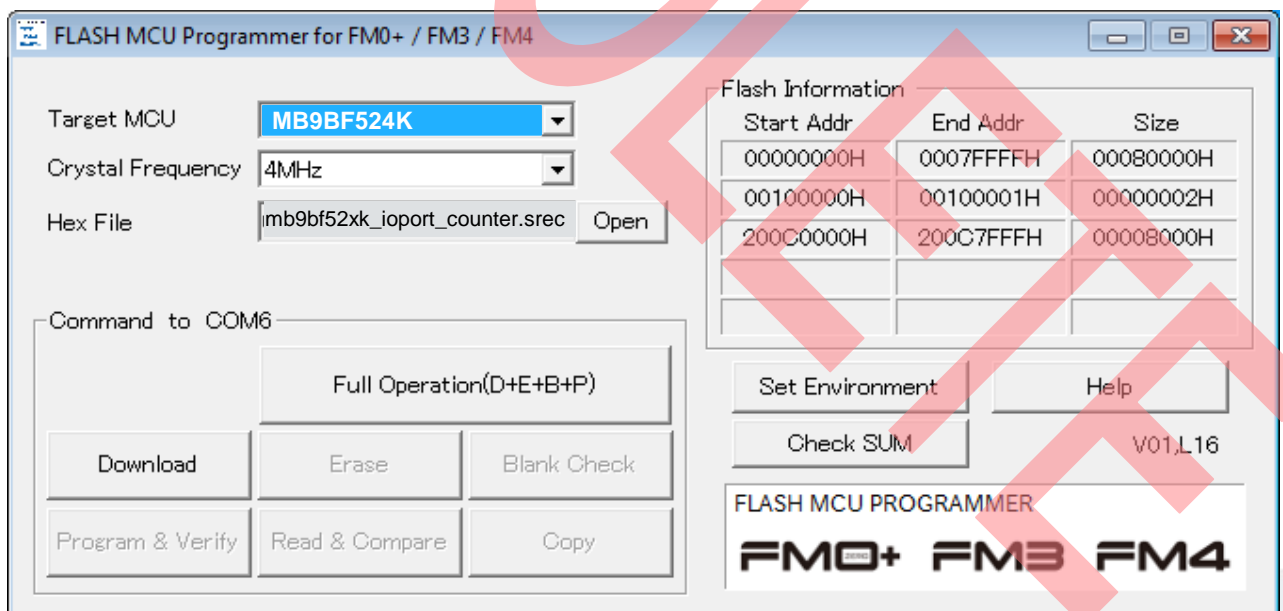
Note: **Please copy the examples to your local drive!**

## Flash Programming

- There are three options to program the flash:
  - UART Programming (X4, X5)
    - ◆ Check jumper JP16 is opened
    - ◆ Connect UART0 of the board to the USB-Port of the PC
      - via USB (JP4,JP5: U-0, R-4)
      - via RS232 (JP4,JP5: U-4, R-0)
    - ◆ Use the [FLASH MCU Programmer](#)
  - USB Programming (X3)
    - ◆ Check jumper JP16 is closed
    - ◆ Connect the board via USB-Device (X3) to the USB-Port of the PC
    - ◆ Use the [FLASH USB DIRECT Programmer](#)
  - JTAG
    - ◆ Use the JTAG-adapter supported by the development toolchain.

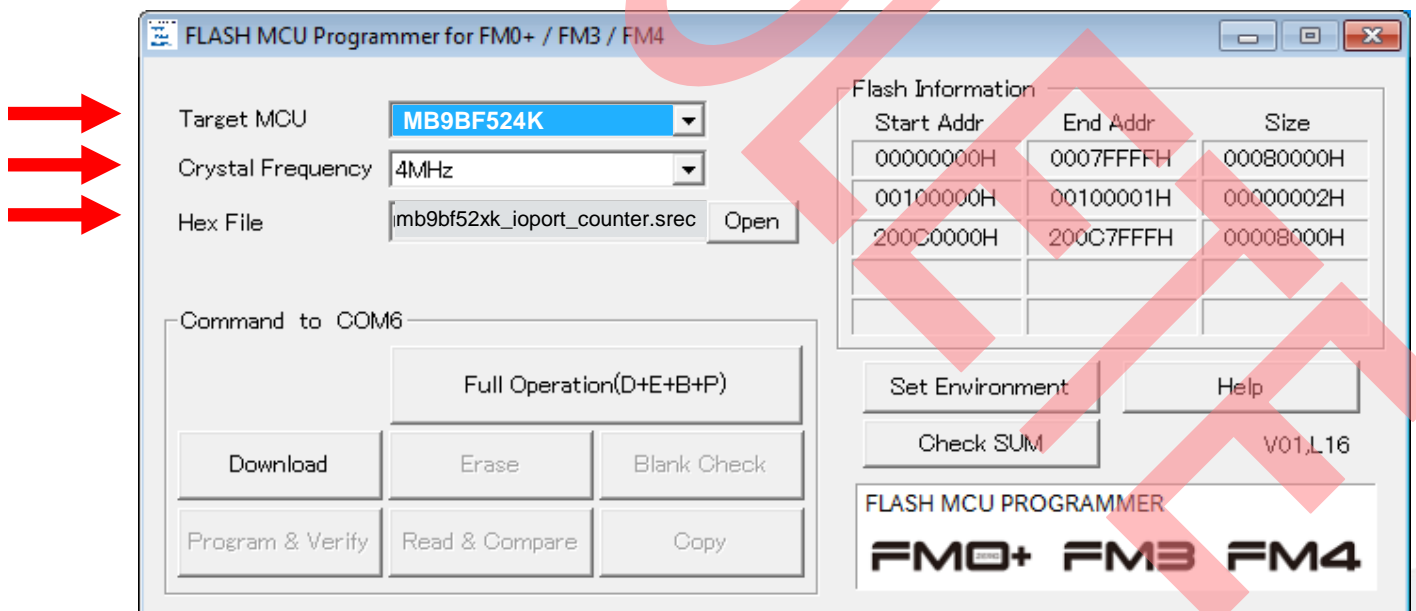
# FLASH MCU Programmer for UART Programming

- FLASH MCU Programmer
  - Free of charge, no registration required
  - Windows based programming tool for FM3 microcontroller
  - Uses PC serial port COMx (incl. virtual COM port: USB-to-RS232)
  - [Start installation](#)



## Program Download

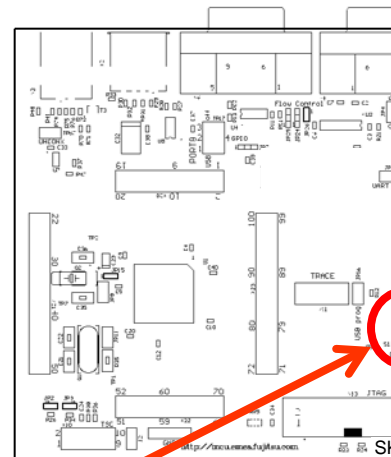
- Start the FLASH MCU Programmer
- Select the target microcontroller (MB9BF524K)
- Select the crystal frequency (4 MHz)
- Choose the software example from the example 'exe'-folder (e.g. Examples\mb9bf52xk\_ioport\_counter-v10\example\IAR\output\release\exe\mb9bf52xk\_ioport\_counter.srec)



# Program Download

- Connect to the PC
  - Connect UART0 with RS232 (X4) or with the USB interface X5
  - Select COM port (,Set Environment')
- Open JP16
- Set switch S1 to position ,PROG'
- Press ,Reset'
- Start ,Full Operation'

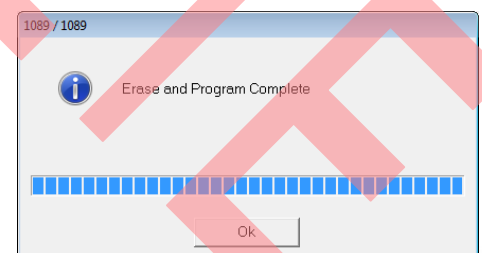
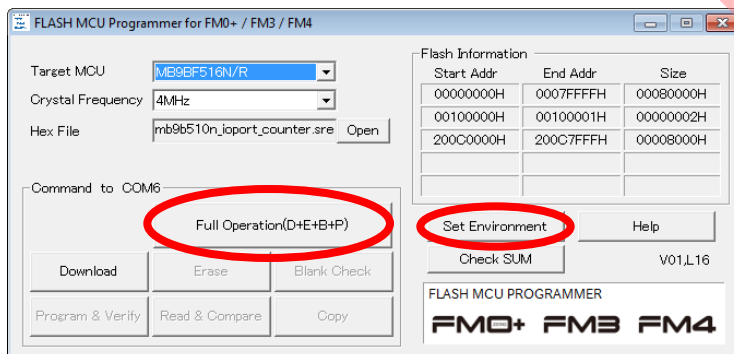
(see JP4, JP5 jump  
RS232



**S1: Mode selection**

**PROG: Set switch to position ,PROG' in order to select the program-mode**

**Keybutt**

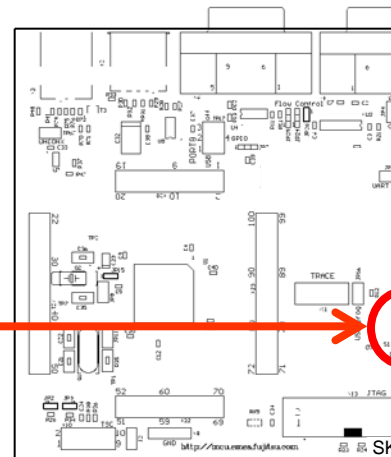


## Program Download

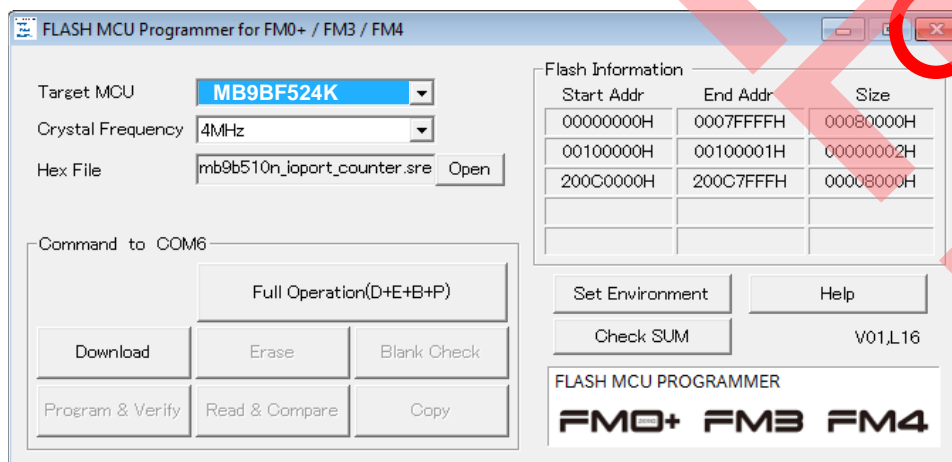
- Close the FLASH MCU Programmer
- Set switch S1 to position ,RUN‘
- Press ,Reset‘

**S1: Mode selection**

**RUN: Set switch to position ,RUN‘  
in order to select the run-mode**



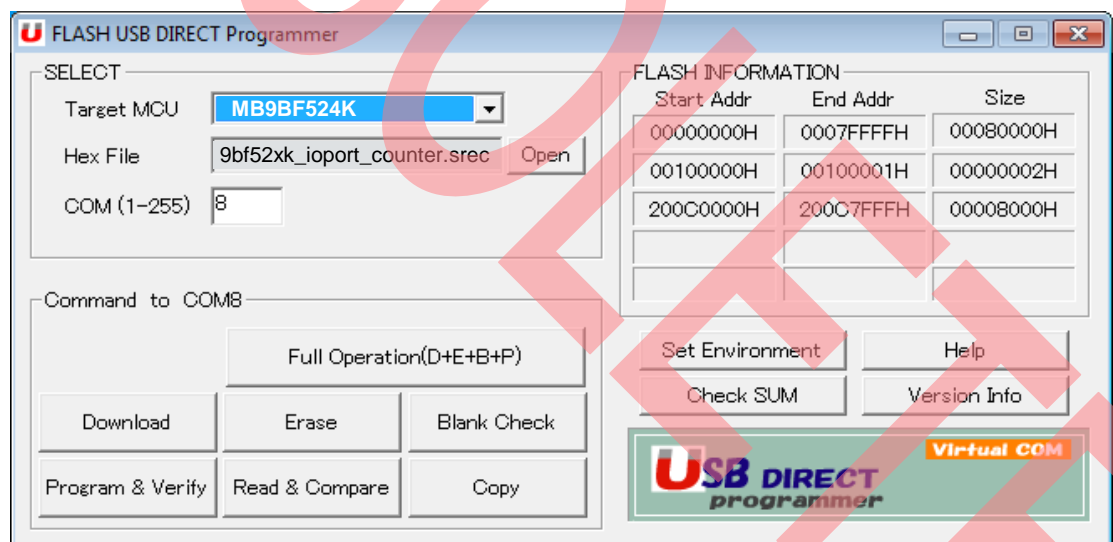
**Keybutt**



**Close the FLASH MC  
Programmer**

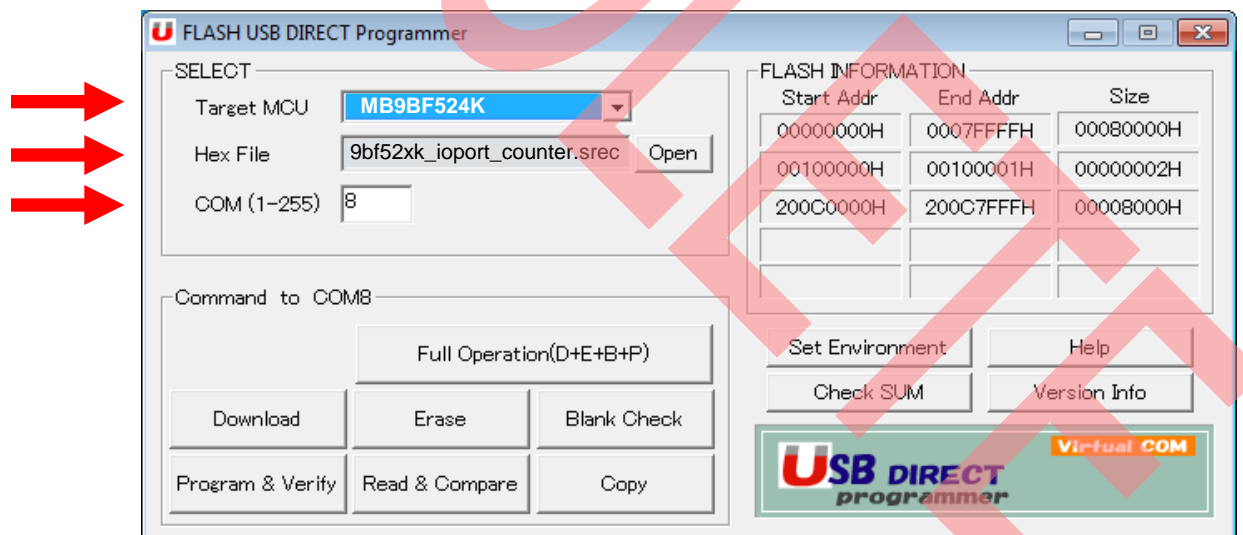
# FLASH USB DIRECT Programmer for USB Direct Programmer

- FLASH USB DIRECT Programmer
  - Windows based programming tool for FM3 microcontroller
  - Uses direct USB connection (via X3)
  - [Start installation](#)



## Program Download

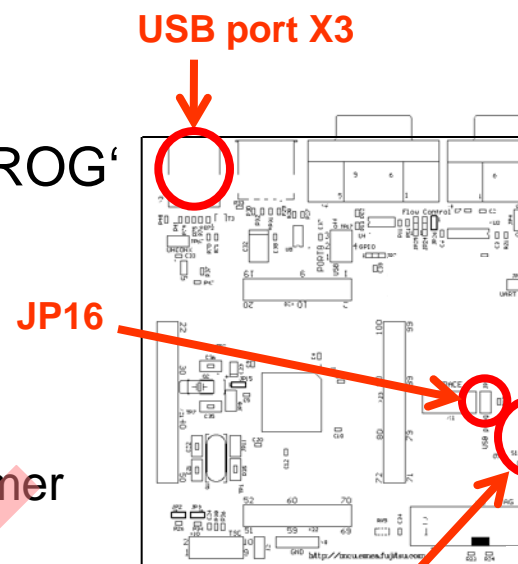
- Start the FLASH USB DIRECT Programmer
- Select the target microcontroller (MB9BF524K)
- Choose the software example from the example 'exe'-folder (e.g. Examples\mb9bf52xk\_ioport\_counter-v10\example\IAR\output\release\exe\mb9bfxxx\_ioport\_counter.srec)
- Select the COM port



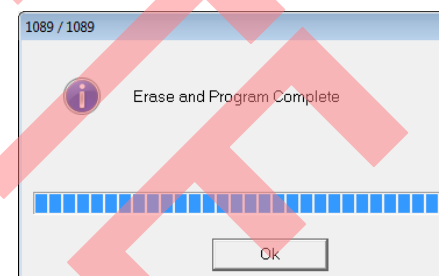
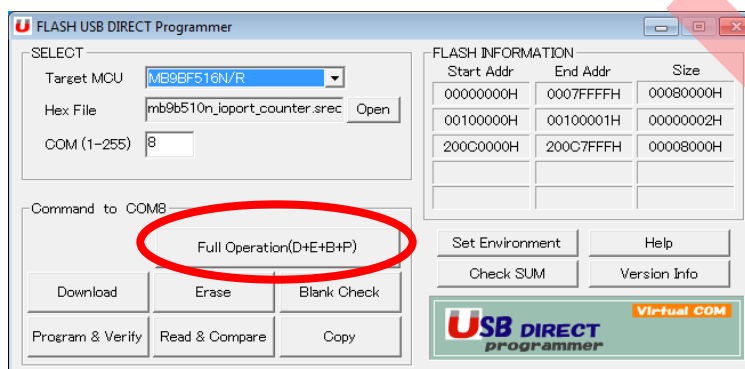


## Program Download

- Select the MCU power supply ([J5](#))
  - Close JP16, Set switch S1 to position 'PROG'
  - Connect USB port X3 with the PC
  - Install the USB driver
    - See subfolder 'driver' of installed programmer
    - E.g.: C:\Program Files (x86)\Spansion\...  
..FLASH USB DIRECT Programmer\driver
- Press 'Reset' and Start 'Full Operation'



**S1: Mode selection**  
**PROG: Set switch to position 'PROG' in order to select the program-mode**

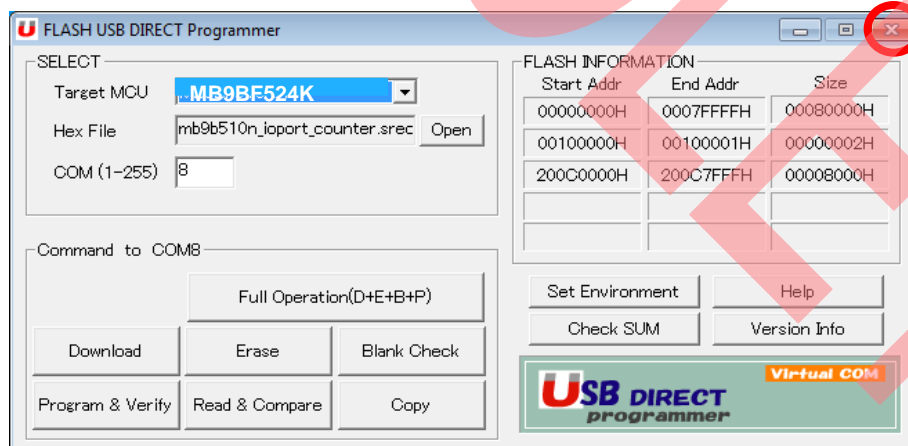
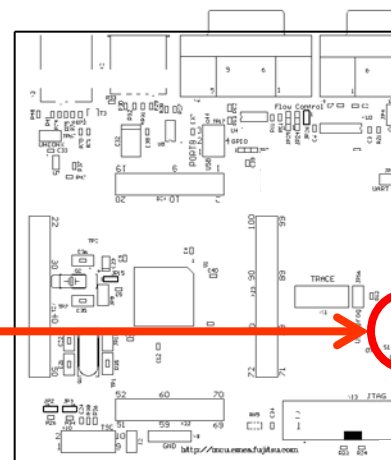


## Program Download

- Close the FLASH USB DIRECT Programmer
- Set switch S1 to position ,RUN‘
- Press ,Reset‘

S1: Mode selection

RUN: Set switch to position ,RUN‘  
in order to select the run-mode

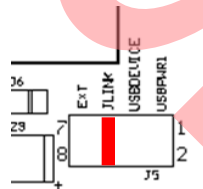


Keybu

Close the FLASH USB  
DIRECT Programmer

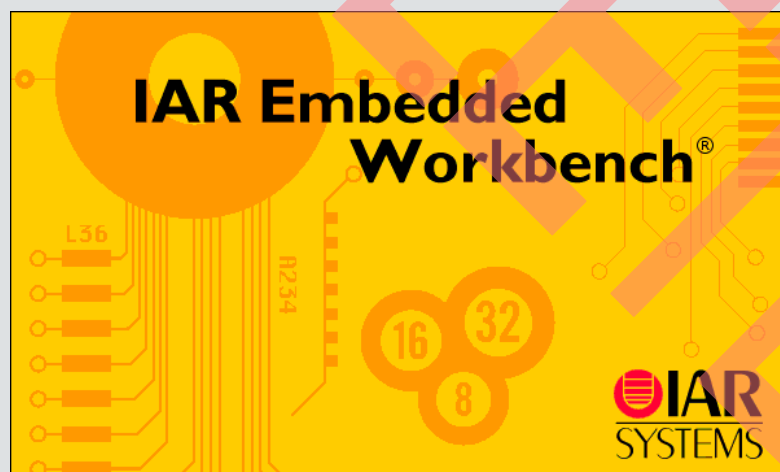
## Debugging via JTAG

- The MB9BF524K microcontroller offers a JTAG-Interface that is supported by SK-FM3-48PMC-MB9BF524K.
  - Debug your program with a JTAG-Adapter e.g. Segger J-Link
  - Connect the J-Link to the JTAG-Interface routed to the 20-Pin-Header X13 and to the USB-Port of your PC
  - Use IAR-Embedded Workbench to debug your program
  - If the JTAG-Adapter allows powering the target, then jumper J5 can be set as follows:



# IAR Embedded Workbench

- Installation
- Getting Started
- Open Project
- Build Project
- Debug Project



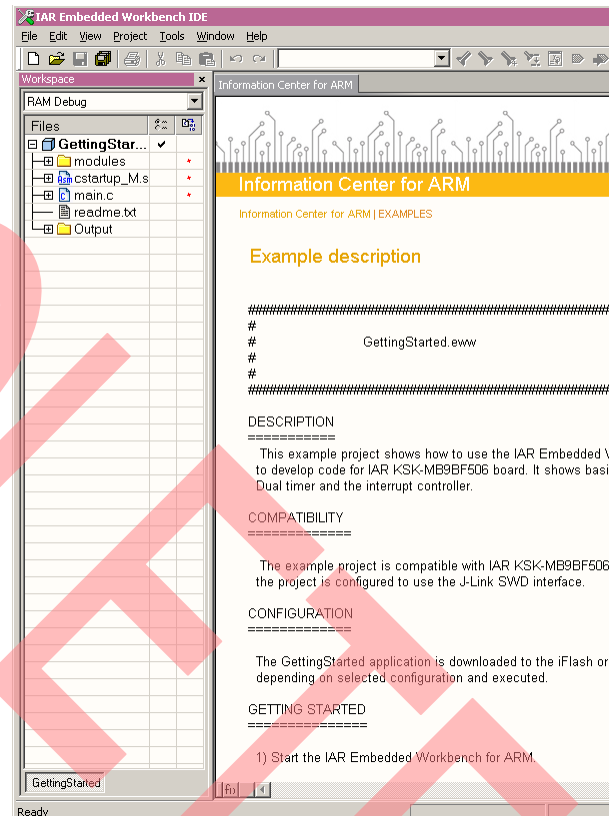
## IAR Workbench Getting Started

- Install EWARM from IAR-CD or download latest version from IAR Website
  - EWARM size-limited (32k) or time-limited (full) Evaluation Version
    - ◆ <http://supp.iar.com/Download/SW/?item=EWARM-EVAL>
- Start EWARM Workbench
- Choose File → Open → Workspace
  - e.g.: <drive:>\<board>\Examples\mb9bf52xk\_ioport\_counter-v11\example\
  - Choose mb9b52xk\_ioport\_counter.eww



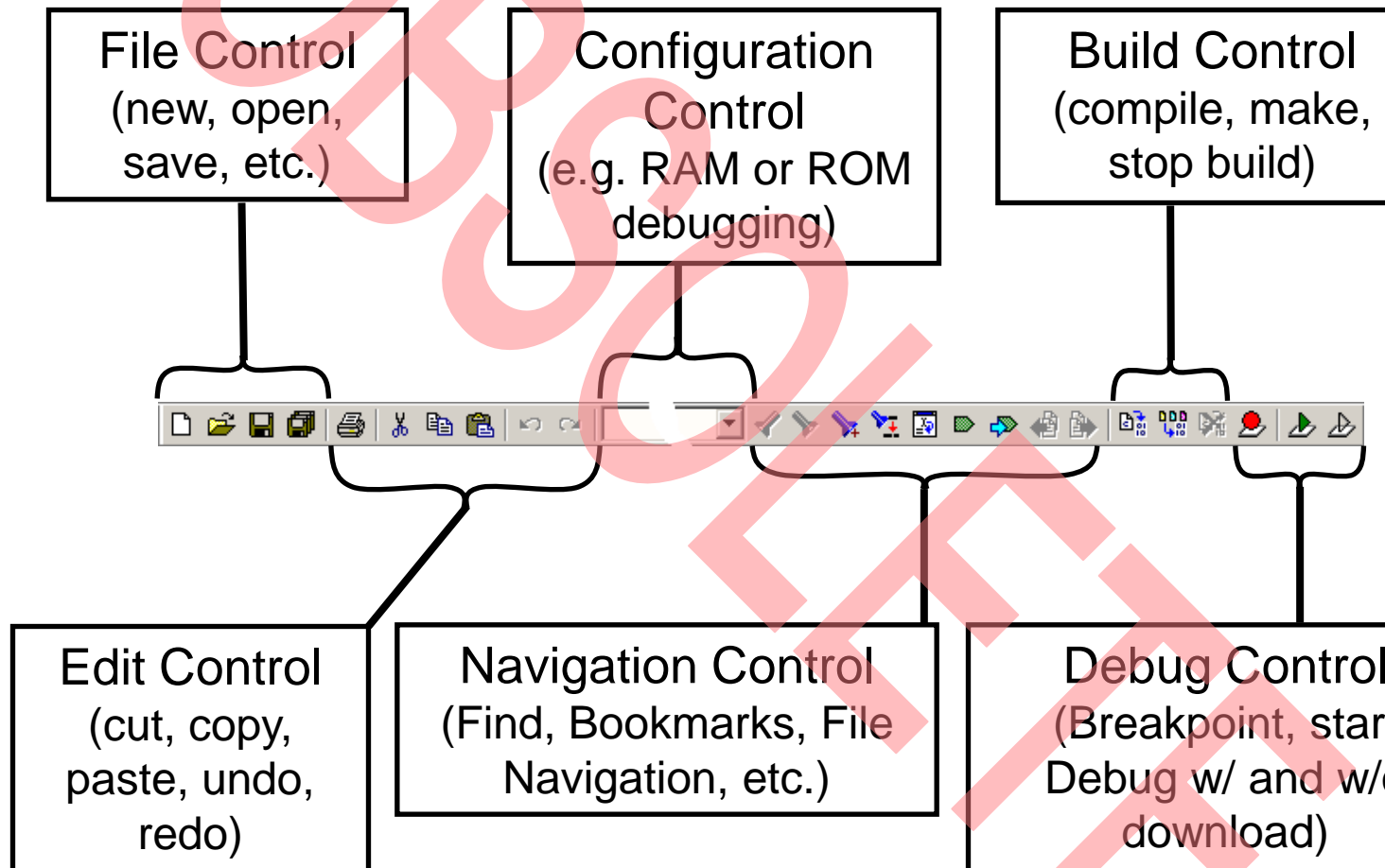
## IAR Workbench – Main Window

- IAR Workbench
  - Workspace on left side of Workbench window
    - ◆ If hidden then View → Workspace
  - Source files on right side of Workbench window as tabbed windows
  - Project open  
File → Open → Workspace → \*.eww
  - For new projects start with ,mb9bf52xk\_template'



# IAR Workbench – Menu Bar

- IAR Menu Bar



# IAR Workbench – Workspace

- IAR Workspace Window

Project Name

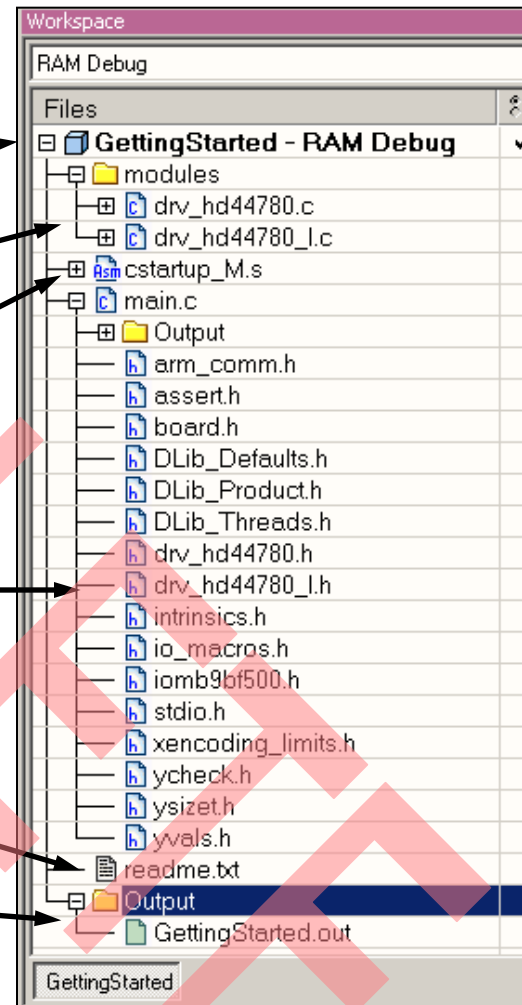
Sub Folder Modules

Main Modules

Module Includes

Project Description




Project Built Output

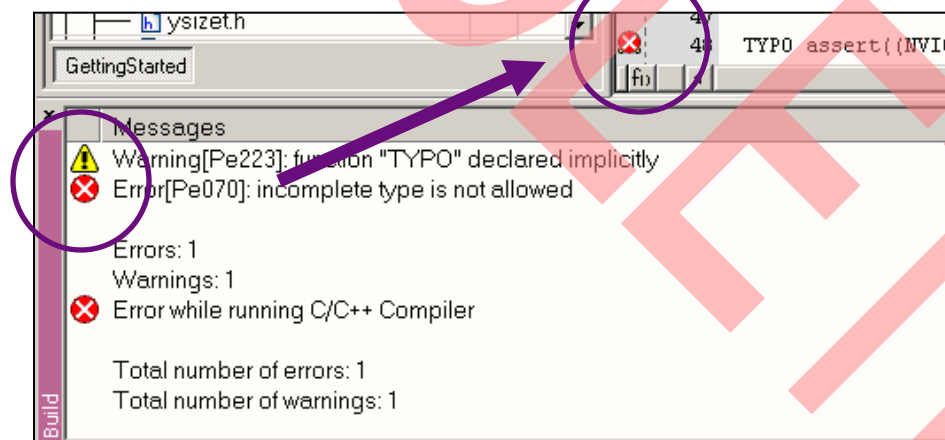
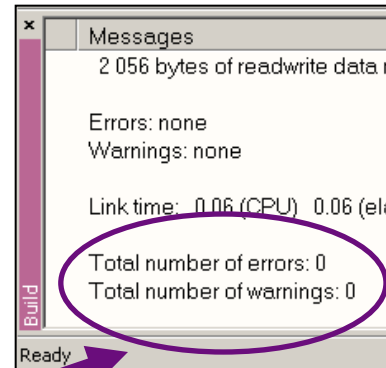





# IAR Workbench – Making Project

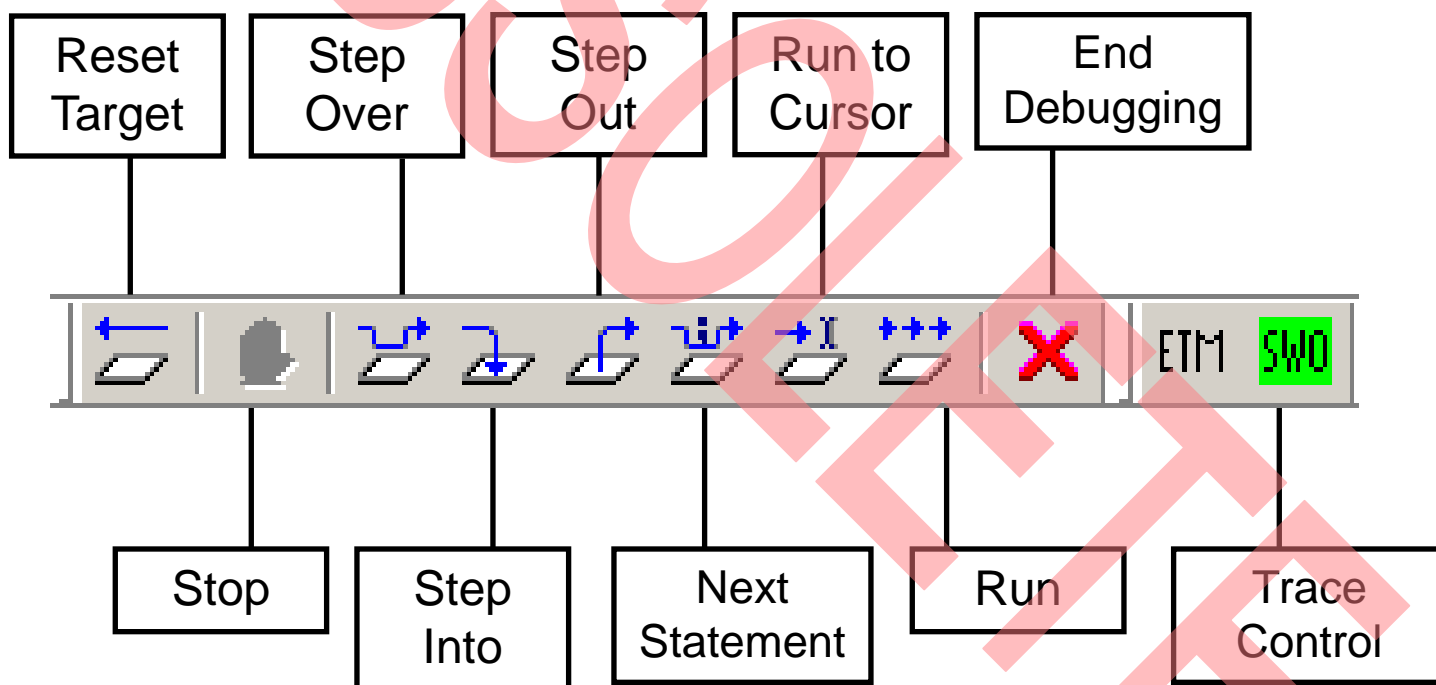
## ■ Making the Project

- Use Make-Icon ( , <F7> or Menu: Project→Make
- Check for no errors in Output window below
- Build errors are indicated by  or  In Output window and Source view



## IAR Workbench – Download to Target

- Download to Target and Start Debugging
  - Use  Icon, <Ctrl>-D, or Project→Download and Debug
  - A new menu bar will occur on successful connection to target



## IAR Workbench – Debug (1)

- Source Window

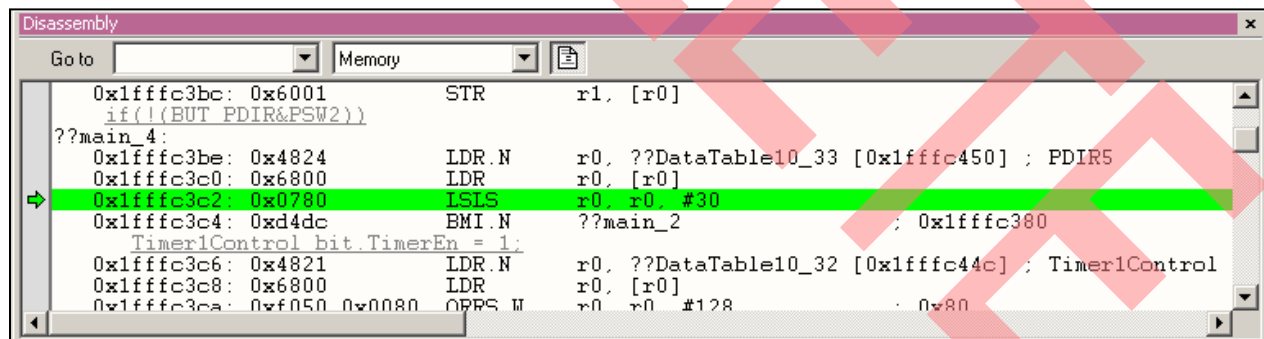
- The Source windows do not change contents but get additional info

- ◆ Current line (PC):
    - ◆ Halted on Breakpoint:
    - ◆ Halted on Data break (example):

```
165 CSW_TMR_bit.MOWT = 9;  
172 PSW_TMR_bit.POWT = 2;  
148 Timer1IntClr = 1;
```

- Disassembly Window

- Shows 'pure' disassembly view
  - Shows mixed mode view

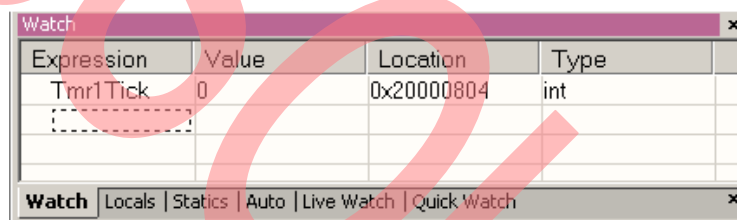


## IAR Workbench – Debug (2)

### ■ Watch Window

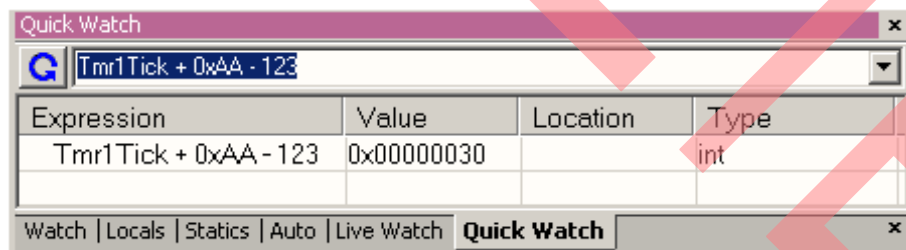
#### – Watch

- ◆ Expressions/Variables have to be added by user and are updated by Halt/Breakpoint




#### – Quick Watch

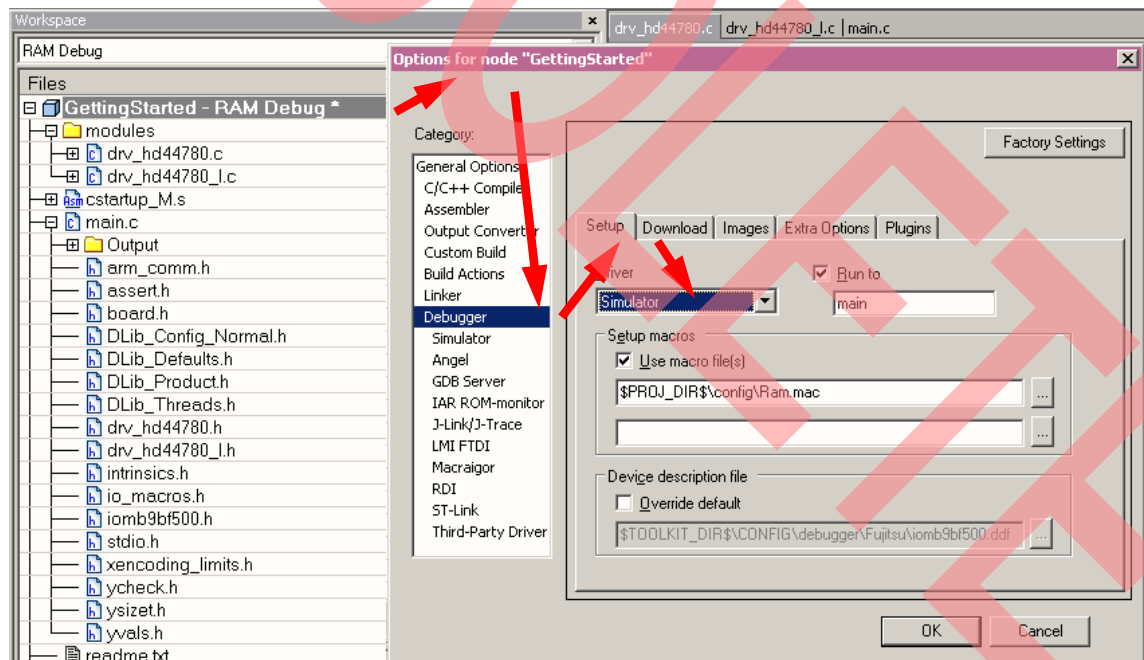
- ◆ The Quick watch allows the user to calculate and recalculate expressions even with variables



- ◆ The drop down menu memorizes the last typed contents

# IAR Workbench – Simulator

- Simulator
  - Mark Project File in Workspace
  - Choose Project → Options
  - Choose Simulator in Debugger Setup
  - Start Simulator with usual  Icon



# KEIL $\mu$ Vision

- Installation
- Getting Started
- Open Project
- Build Project
- Debug Project



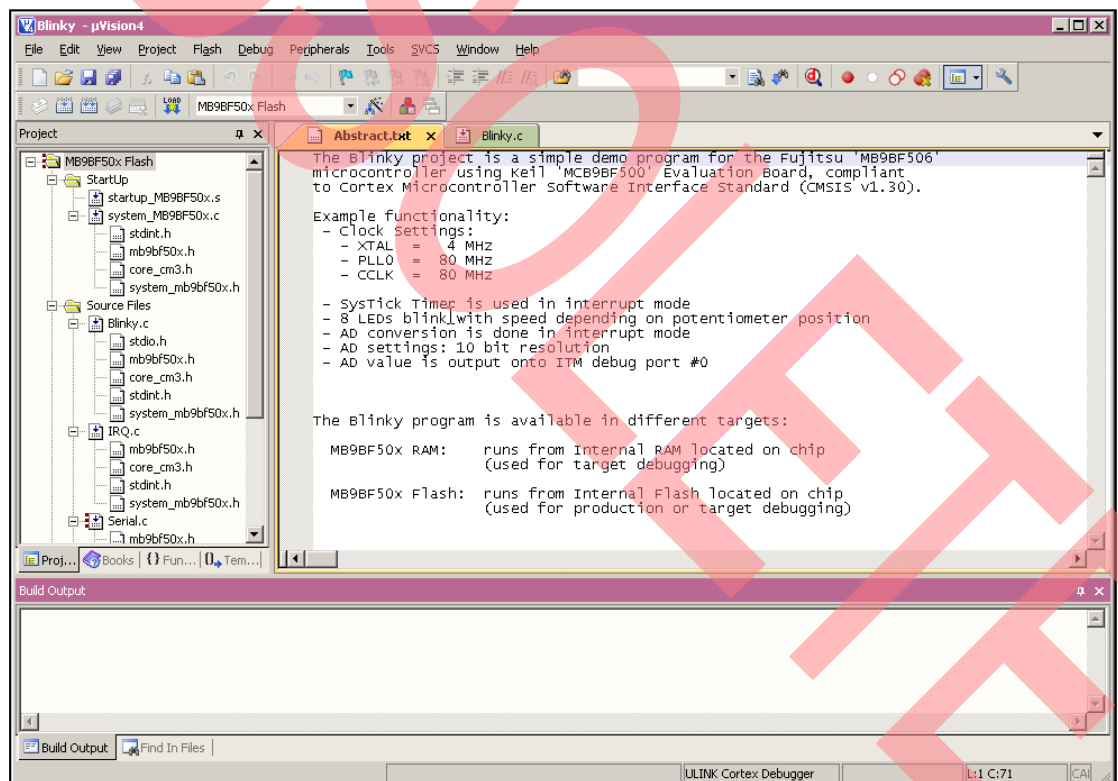
## KEIL $\mu$ Vision IDE and Debugger Getting Started

- Install  $\mu$ Vision from KEIL-CD or download latest version from KEIL Website
  - Evaluation Version
    - ◆ <https://www.keil.com/demo/eval/arm.htm>
    - ◆ Registration required
- Install ULINK-ME
  - Special installation is not needed, because ULINK-ME acts as a USB Human Interface Device (HID) and thus needs no extra USB driver
- Install ULINK Pro (optional)
  - ULINK Pro needs an own dedicated USB driver located in:  
<Installation Path>\KEIL\ARM\ULINK
- Start  $\mu$ Vision

# KEIL $\mu$ Vision – Getting Started

- Choose Menu: Project  $\rightarrow$  Open Project...

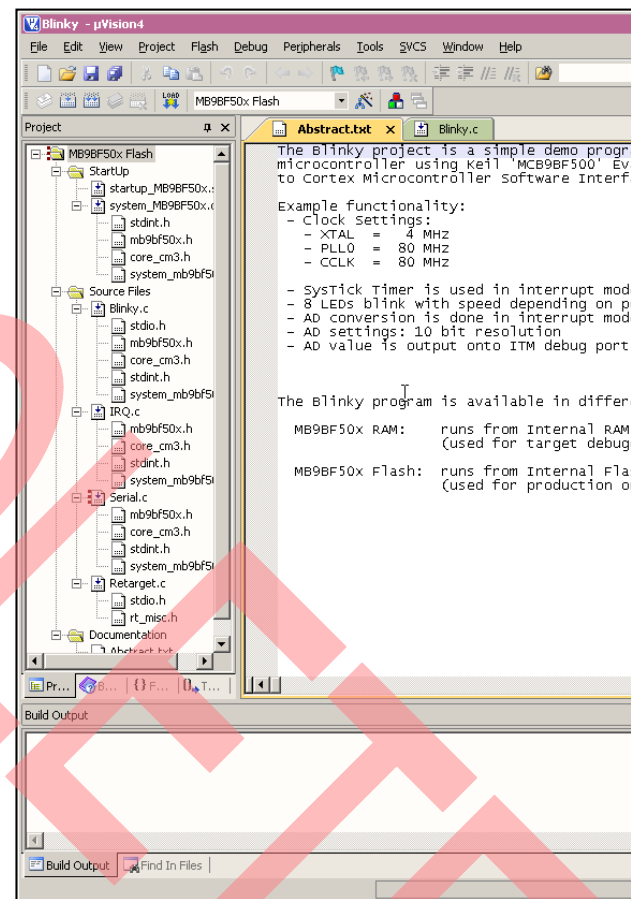
- Browse to: <drive:>\<board>\Examples\mb9bf52xk\_adc\_dvm-v11\example\
- Choose mb9b52xk\_adc\_dvm.uvproj





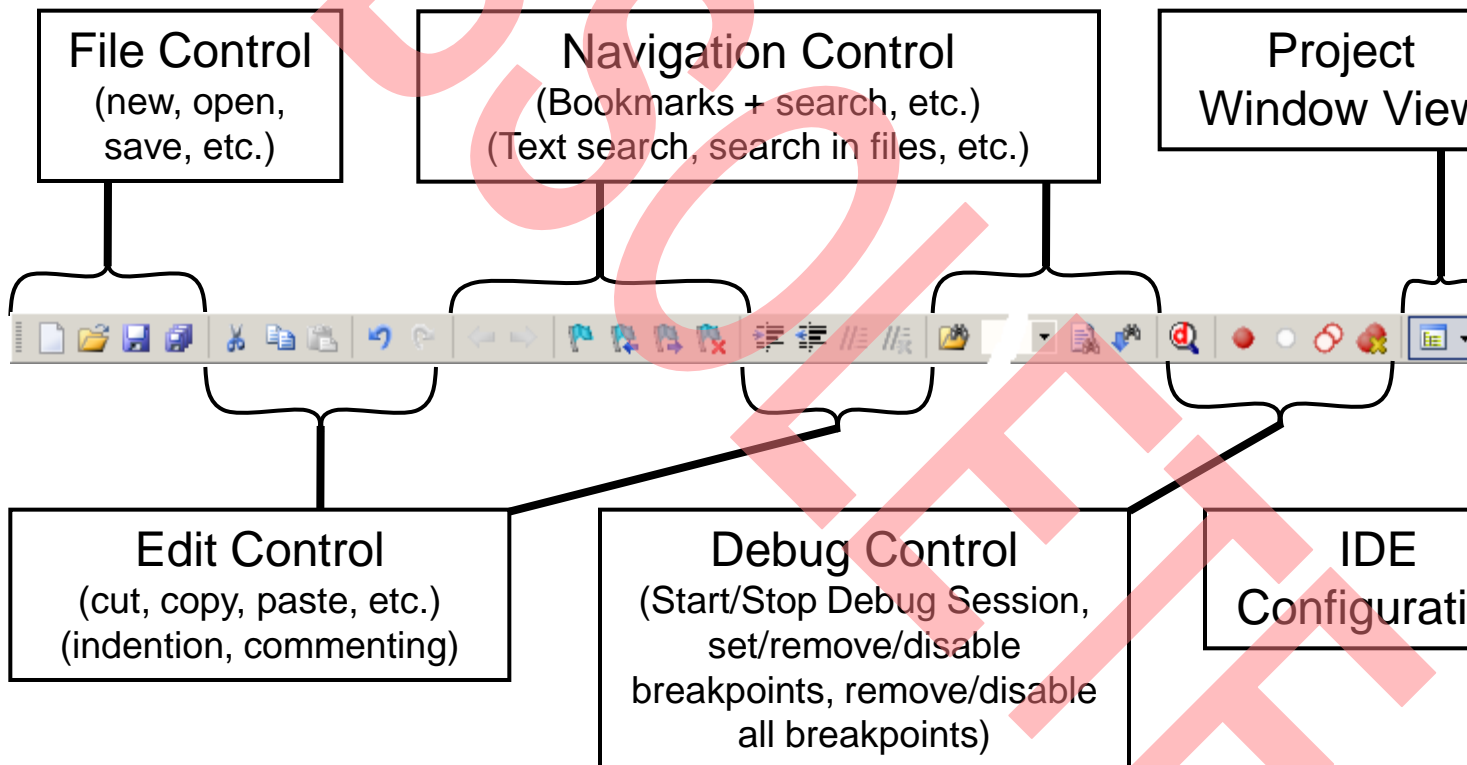
## KEIL $\mu$ Vision – Main Window

- KEIL  $\mu$ Vision
  - Project window on left side of IDE window
    - ◆ Choose:  
View→Project Window  
if hidden
  - Source files on right side of IDE window as tabbed windows
  - Output window on bottom side of IDE window



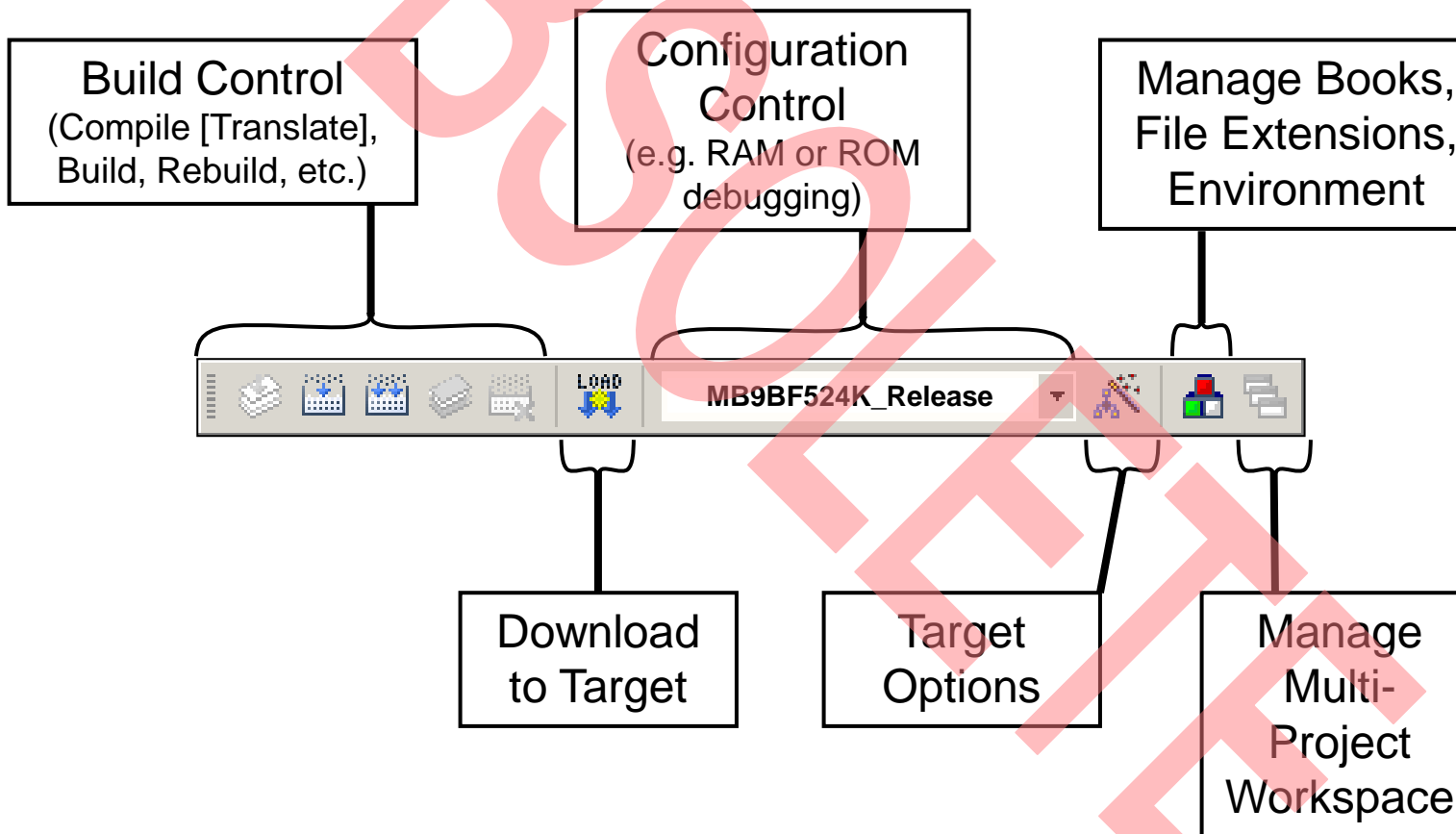
## KEIL $\mu$ Vision – Menu Bars (1)

- Menu Bar 1
  - Can be moved in bar window area or set floating



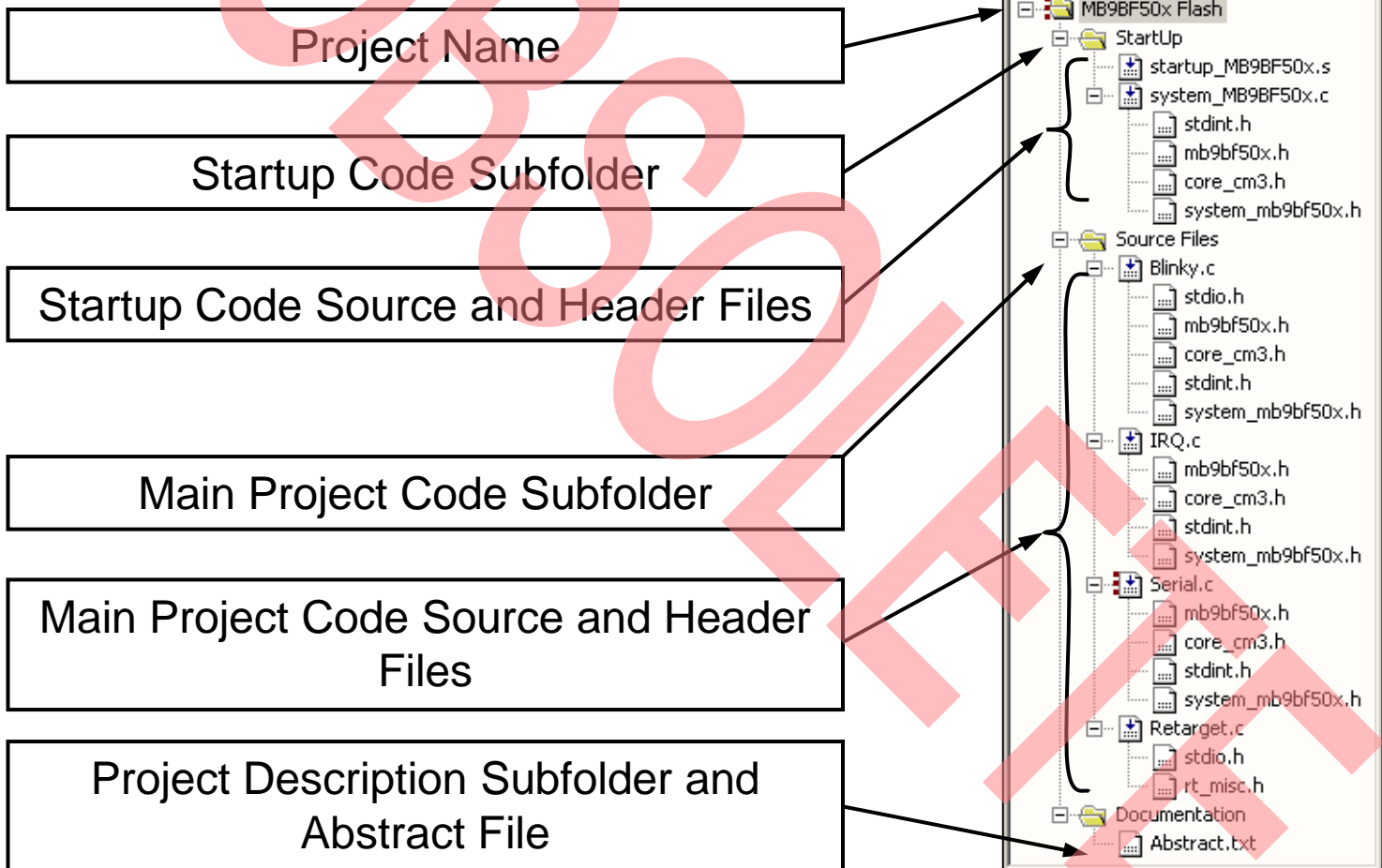
## KEIL $\mu$ Vision – Menu Bars (2)

- Menu Bar 2
  - Can be moved in bar window area or set floating




# KEIL $\mu$ Vision – Project Window

- $\mu$ Vision Project Window



## KEIL $\mu$ Vision – Making Project

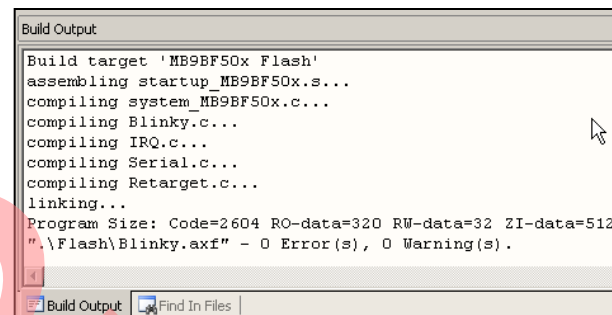
### ■ Making the Project

– Use Rebuild Icon  
(  ) or  
Project→Rebuild all target  
files

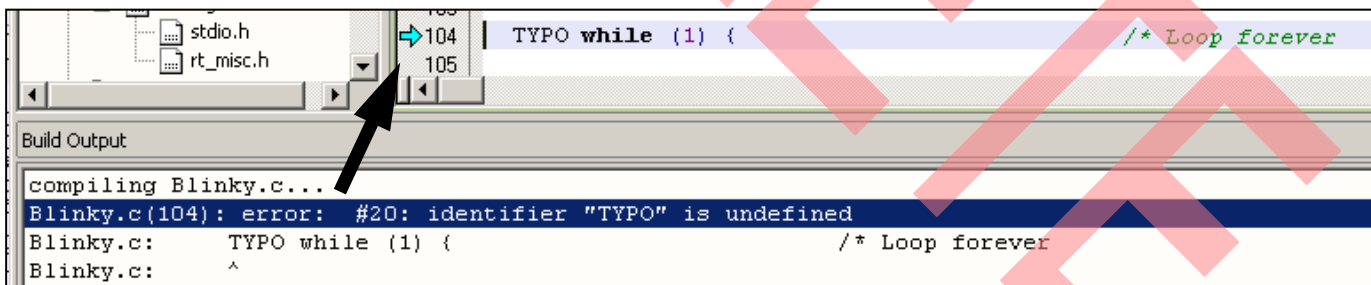
– Check for no errors in  
Output window below

– Build errors are shown in Output window.

- ◆ Can be double-clicked by showing the source line with a blue arrow





```
Build Output
Build target 'MB9BF50x Flash'
assembling startup_MB9BF50x.s...
compiling system_MB9BF50x.c...
compiling Blinky.c...
compiling IRQ.c...
compiling Serial.c...
compiling Retarget.c...
linking...
Program Size: Code=2604 RO-data=320 RW-data=32 ZI-data=512
".\Flash\Blinky.axf" - 0 Error(s), 0 Warning(s).
```



```
stdio.h
rt_misc.h
104 | TYPO while (1) { /* Loop forever
105 |
Build Output
compiling Blinky.c...
Blinky.c(104): error: #20: identifier "TYPO" is undefined
Blinky.c:      TYPO while (1) { /* Loop forever
Blinky.c:      ^
```

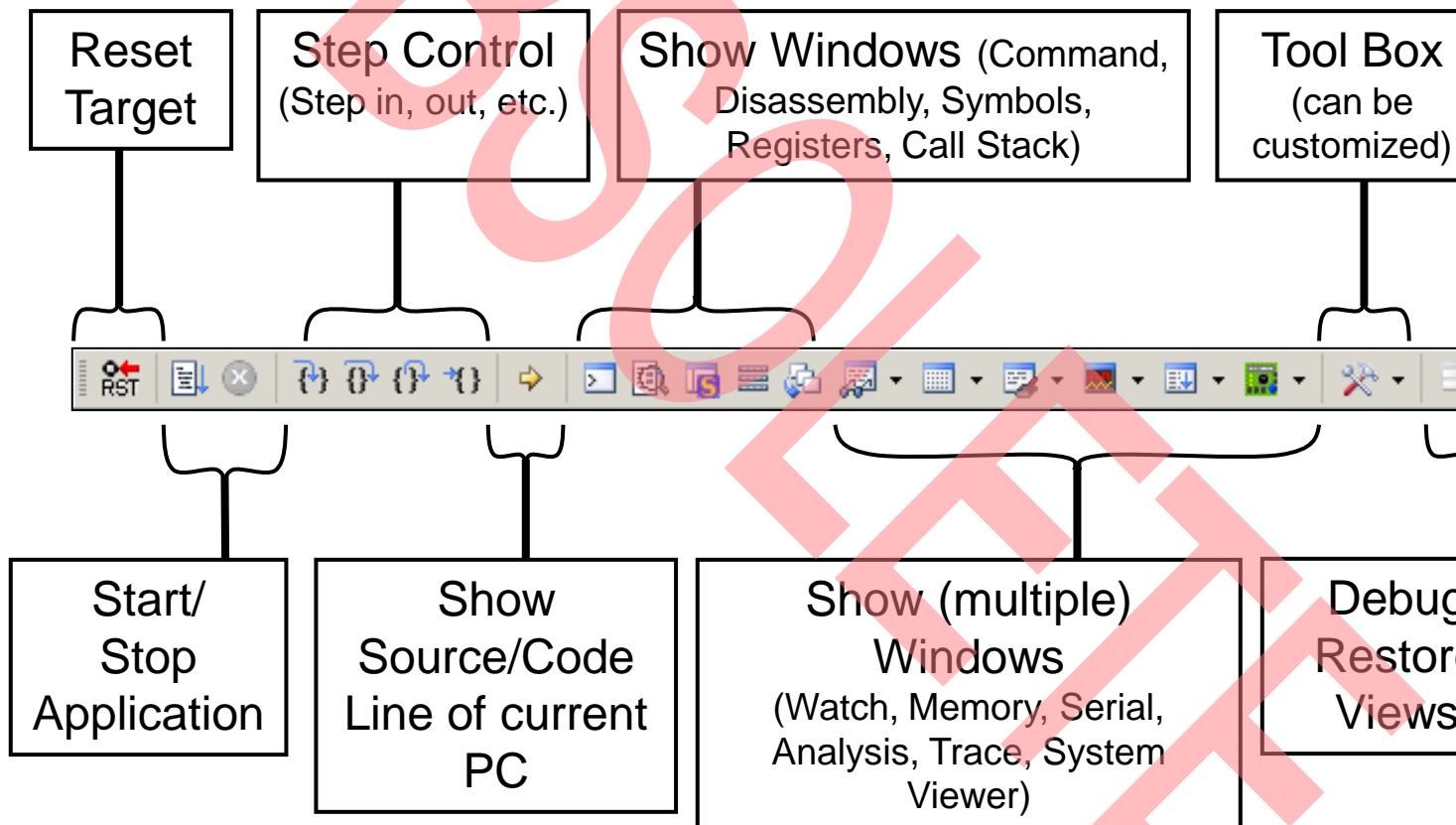
## KEIL $\mu$ Vision – Debug (1)

### ■ Start Debugging

- Download to target first, when MCU Flash does not contain the current application opened and built in the IDE
  - ◆ Use Download Icon () or Menu: Flash→Download
- Start Debug Session
  - ◆ Use Start/Stop Debug Icon () or Menu: Debug→Start/Stop Debug Session
- Ending Debug Session
  - ◆ Use same way as for starting debug session

## KEIL $\mu$ Vision – Debug (2)

- Debugging Icon Bar
  - During a Debug Session there will be visible a new icon bar



## KEIL $\mu$ Vision – Debug (3)

- Source View

- The Source windows do not change contents but get additional info

The diagram illustrates the source code view in KEIL  $\mu$ Vision. It shows a window titled 'Blinky.c' with the following code:

```
098 SysTick_Config(SystemCoreCl  
099  
100 LED_init();  
101 ADC_init();  
102 SER_init();  
103  
104 while (1) {  
105  
106     AD_value = AD_last;  
107     if (AD_value != AD_last)  
108         AD_value = AD_last;  
109  
110     if (AD_value != AD_print)  
111         AD_print = AD_value;
```

Annotations on the left side of the code window:

- Active Breakpoint:** A red square next to line 101.
- Disabled Breakpoint:** A white square next to line 104.
- Current Program Counter:** A yellow arrow pointing to line 106.
- Current Cursor Line of Source Code:** A cyan arrow pointing to line 108.
- Code Lines with compiled Instructions (dark grey):** Lines 106, 107, 108, and 109 are highlighted in dark grey.



## KEIL $\mu$ Vision – Debug (4)

- Disassembly View
  - Mixed mode is selectable and deselectable

Active Breakpoint

Disabled Breakpoint

Current Program Counter

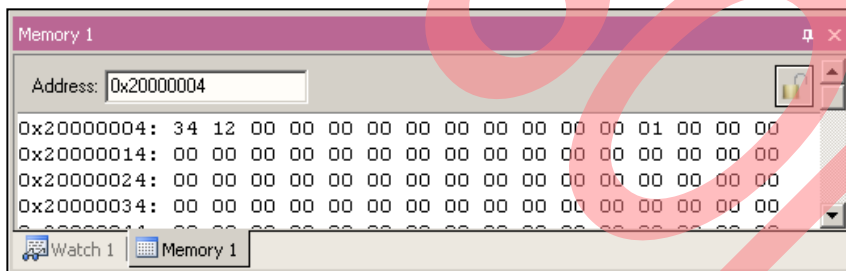
Current Cursor Line of Code highlighted in yellow background (■)

```
Disassembly
0x0000042A F7FFFA3 BL.W LE
101: ADC_init();
 0x0000042E F7FFF67 BL.W AD
102: SER_init();
103:
0x00000432 F000F8AE BL.W SE
104: while (1) {
105:
 0x00000436 E015 B Ox
106: AD_value = AD_last;
 0x00000438 4816 LDR r0
0x0000043A 8804 LDRH r4
107: if (AD_value != AD_1
```

## KEIL $\mu$ Vision – Debug (5)

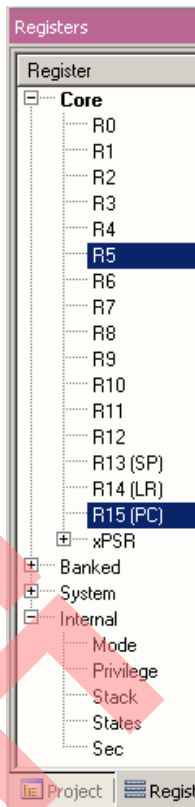
### ■ Memory Window

- Up to 4 Memory windows can be displayed in tabs
- Memory is updated during runtime
- Memory window tabs are shared with Watch windows



### ■ Register View

- Register view is a tab of the Project window
- Changes are highlighted in dark blue text background
- Register tree knots can be expanded

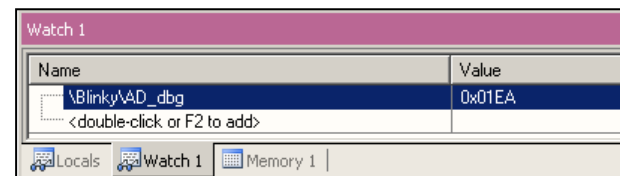


## KEIL $\mu$ Vision – Debug (6)

### ■ Variable Windows

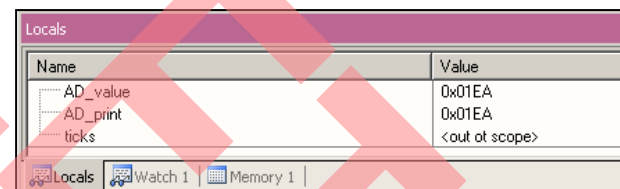
#### – Watch Windows

- ◆ Up to 2 Watch windows are sharing their tabs with e.g. Memory and views
- ◆ Updated during runtime
- ◆ Any changes are highlighted in dark blue text background color
- ◆ Displayed values can be changed by user during break



#### – Local View

- ◆ The local view shares the tab with e.g. Memory and Watch windows
- ◆ Any changes are highlighted in dark blue text background color
- ◆ Displayed values can be changed by user during break



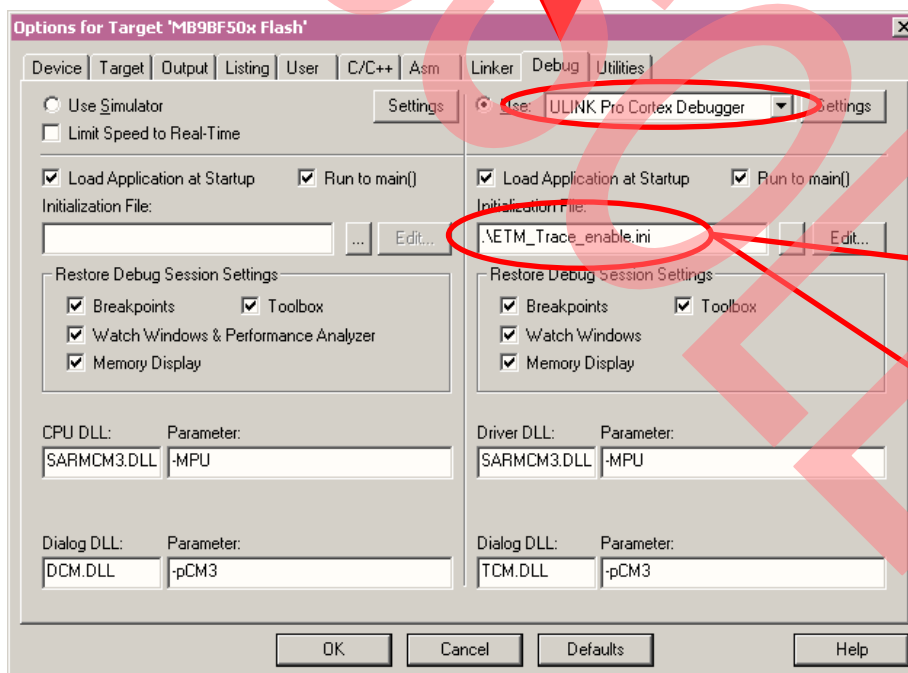
## KEIL $\mu$ Vision – Trace (ULINK ME)

- Trace via ITM
  - Simple Trace views via Instrumentation Trace Macro is supported by  $\mu$ LINK ME
    - ◆ Records
    - ◆ Exceptions
    - ◆ Counters

Type	Dly	Num	Address	Data	PC	Dly	Cycles	Time
ITM		0	41H				82975148	1.03718
ITM		0	44H				82975293	1.03719
ITM		0	20H			X	82988592	1.03735
ITM		0	76H			X	82988592	1.03735
ITM		0	61H			X	82988592	1.03735
ITM		0	6CH			X	82988592	1.03735
ITM		0	75H			X	82988592	1.03735
ITM		0	65H			X	82988592	1.03735
ITM		0	20H			X	82988592	1.03735
ITM		0	3DH			X	82988592	1.03735
ITM		0	20H			X	82988592	1.03735
ITM		0	30H			X	82988592	1.03735
ITM		0	78H			X	82988592	1.03735
ITM		0	30H				82993831	1.03742
ITM		0	31H			X	83001392	1.03751
ITM		0	45H			X	83001392	1.03751
ITM		0	42H			X	83001392	1.03751
ITM		0	0DH			X	83001392	1.03751
ITM		0	0AH			X	83001392	1.03751
ITM		0	0DH			X	83001392	1.03751

## KEIL $\mu$ Vision – Trace (ULINK Pro) (1)

- Trace via ETM
  - Check settings in menu:  
Flash→Configure Flash Tools... Tab:Debug



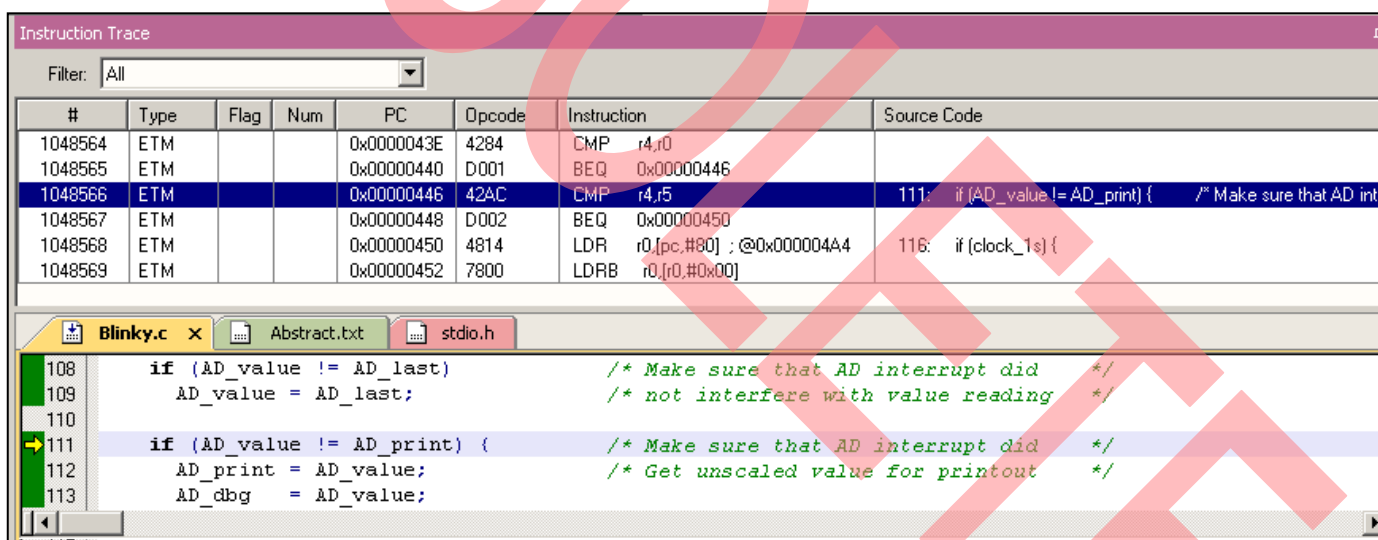
```
ETM_Trace_enable.ini - Notepad
File Edit Format View Help
_LDWORD(0x40033000, 0x000003FF);
_WBYTE(0x40033603, 0x03);
```

enables ETM pins

## KEIL $\mu$ Vision – Trace (ULINK Pro) (2)

### ■ Instruction Trace

- Real Time Trace recording
- Output can be filtered by several ETM and ITM events
- Trace buffer is held in PC memory and transferred to  $\mu$ Vision on break



The screenshot displays the 'Instruction Trace' window in KEIL  $\mu$ Vision. The window has a filter set to 'All'. Below the filter is a table with columns: #, Type, Flag, Num, PC, Opcode, Instruction, and Source Code. The table contains several rows of instruction data. Below the table, the source code for 'Blinky.c' is visible, with line 111 highlighted in blue, corresponding to the selected instruction in the trace table.

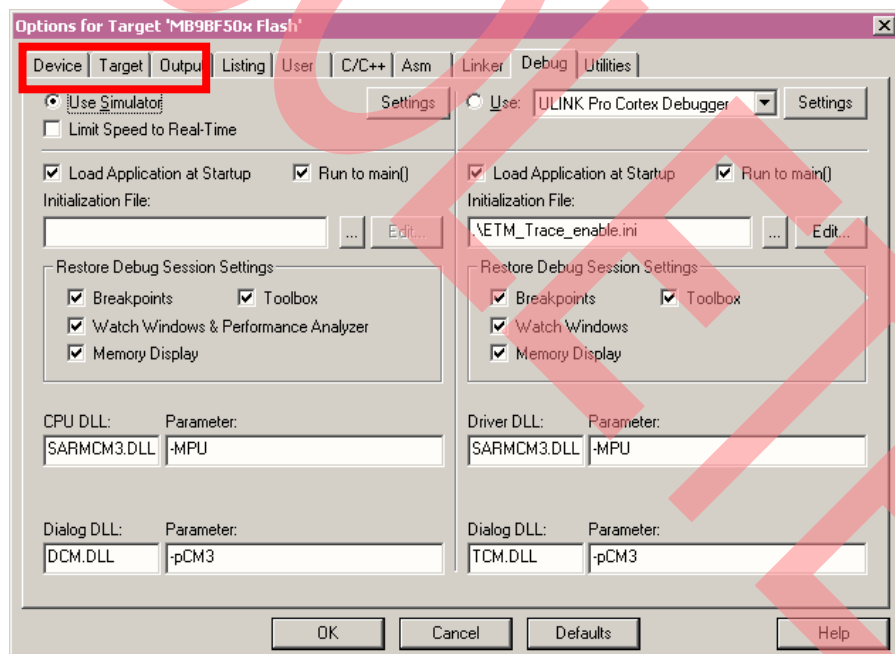
#	Type	Flag	Num	PC	Opcode	Instruction	Source Code
1048564	ETM			0x0000043E	4284	CMP r4,r0	
1048565	ETM			0x00000440	D001	BEQ 0x00000446	
1048566	ETM			0x00000446	42AC	CMP r4,r5	111: if (AD_value != AD_print) { /* Make sure that AD int
1048567	ETM			0x00000448	D002	BEQ 0x00000450	
1048568	ETM			0x00000450	4814	LDR r0,[pc,#80] ; @0x000004A4	116: if (clock_1s) {
1048569	ETM			0x00000452	7800	LDRB r0,[r0,#0x00]	

```
108     if (AD_value != AD_last)           /* Make sure that AD interrupt did */
109         AD_value = AD_last;           /* not interfere with value reading */
110
111     if (AD_value != AD_print) {        /* Make sure that AD interrupt did */
112         AD_print = AD_value;          /* Get unscaled value for printout */
113         AD_dbg   = AD_value;
```

## KEIL $\mu$ Vision – Simulator

### ■ Simulator

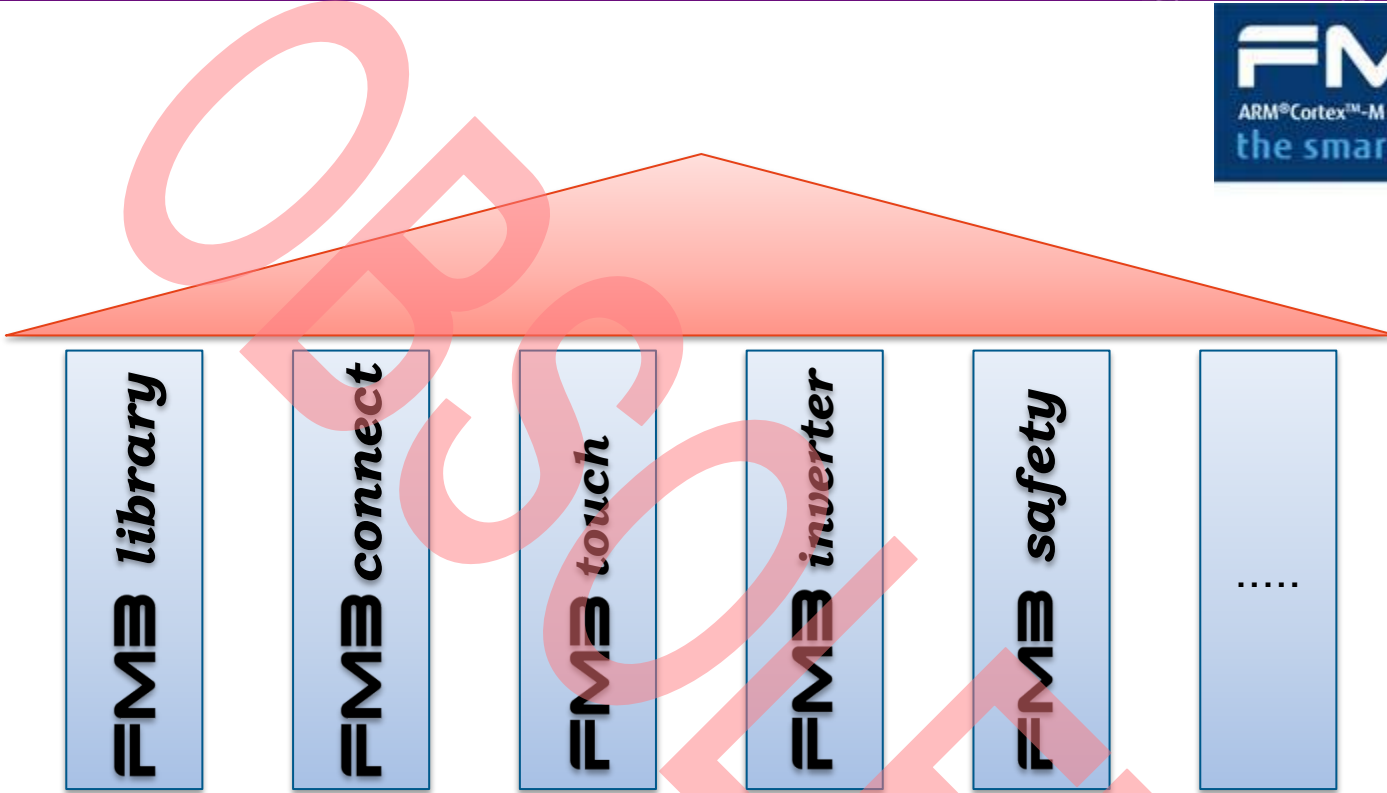
- The Core Simulator can be selected by the menu: [Flash] → [Configure Flash Tools...] → [Debug] and then choosing [Use Simulator]
- Look & feel is like using ULINK debugger
- Controlable also with \*.ini files



# Solutions

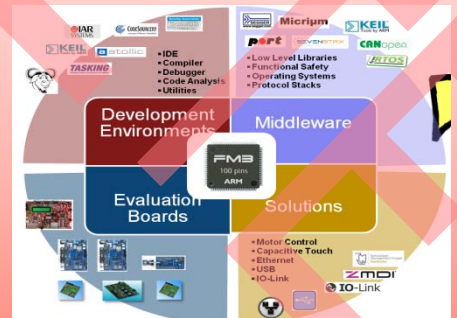
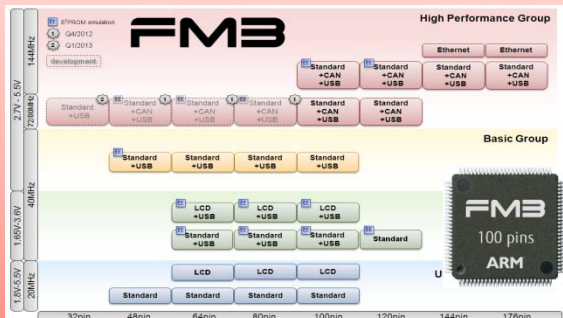


# Solutions



## Products

## Eco system



## Solutions

### FM3 library

- FM3 Low Level Library (L3)
  - CMSIS compliant header files
  - Driver collection to supports MCU peripherals
  - Hardware abstraction layer offers an API
  - Interrupt handling supported
  - Optimized memory use
    - ◆ For unused resources, no memory for library code is allocated
- Supported Modules
  - ADC (A/D-Converter), BT (Base Timer), CAN, CRC, CLK , CRTRIM (Clock Trimming), CSV (Clock Supervisor) , DAC (D/A-Converter), DMA, DSM (Deep Standby Modes), DT (Dual Timer), EXINT (External Interrupts), EXTIF (External Bus Interface), FLASH, GPIO, LVD (Low Voltage Detection), MFS (Multi Function Serial: UART, SPI, I2C, LIN), MFT (Multi Function Timer), QPRC (Quadrature Encoding), RESET (Reset Cause), RTC (Real Time Clock), USB (Host and Device) , WC (Watch Counter), WDG (Watchdog: SW, HW) and more.

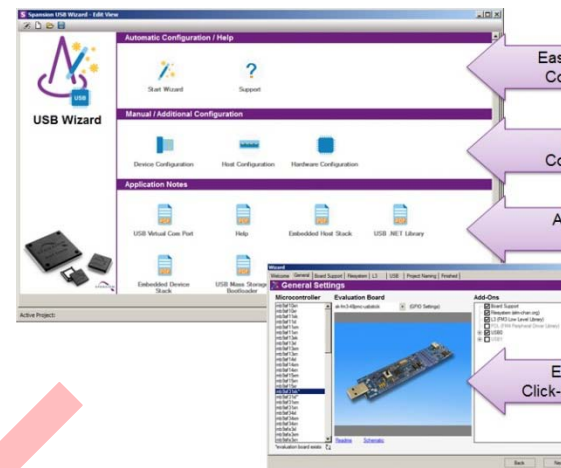


# Solutions

## FMB connect

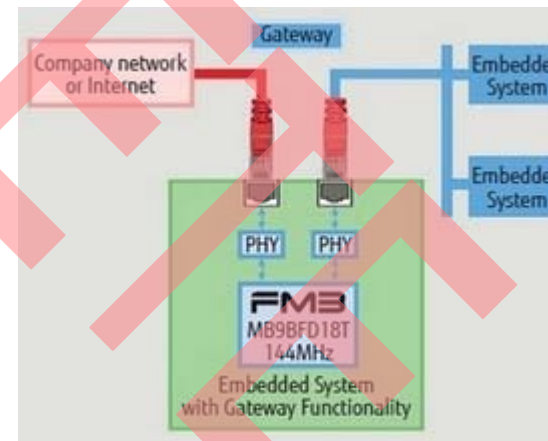
### FMconnect USB

- Up to two USB interfaces
  - ◆ Supports Host/Device/OTG
  - ◆ Control, interrupt, bulk, isochronous
- Free software examples
- Spansion USB Wizard (PC based GUI):
  - ◆ USB driver configuration
  - ◆ Easy creation of USB descriptors
  - ◆ Code injection in existing projects



### FMconnect Ethernet

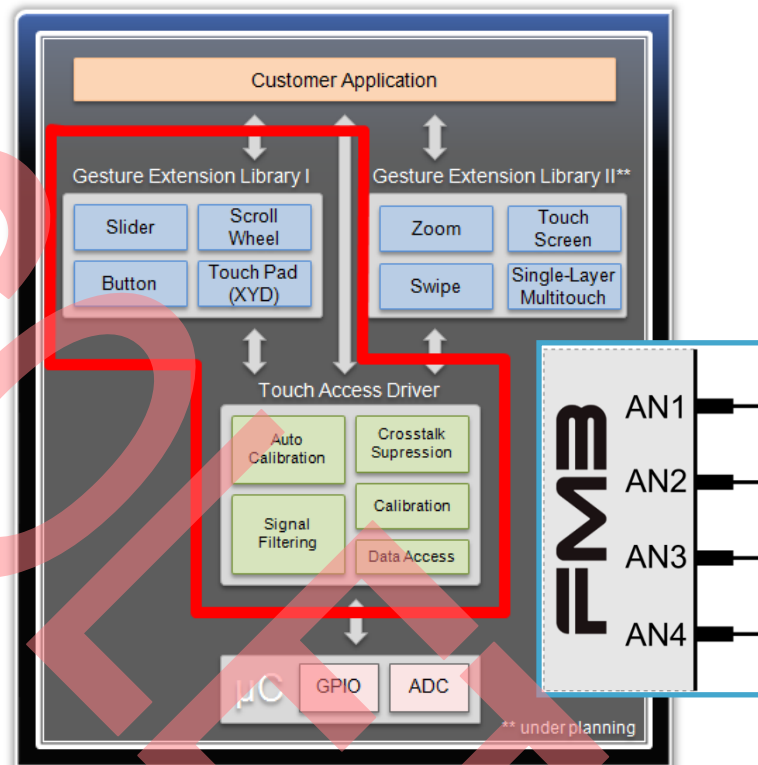
- One or two channels Ethernet MAC
- Dedicated Ethernet starter kit
- Free Ethernet software:
  - ◆ Low level driver available
  - ◆ TCP/IP stack available
  - ◆ Software examples, e.g.: web server
- Commercial products from partners



## Solutions

### FM3 touch

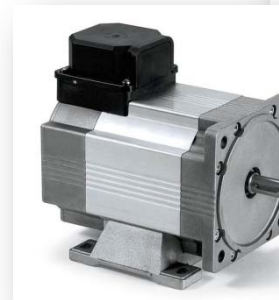
- Software FM3touch library
  - Works on all FM3 derivatives, user can freely choose best-fitting FM3 MCU and add touch functionality
  - No external components
  - Only one pin (ADC channel) per touch input
  - High sensitivity (<math><10\text{fF}</math>)
  - Low resource usage, no 'atomic' handling required
  - Flexible configuration and event system for easy integration
  - Configuration tool included
  - Free of charge (basic variant)



## Solutions

### FM3 inverter

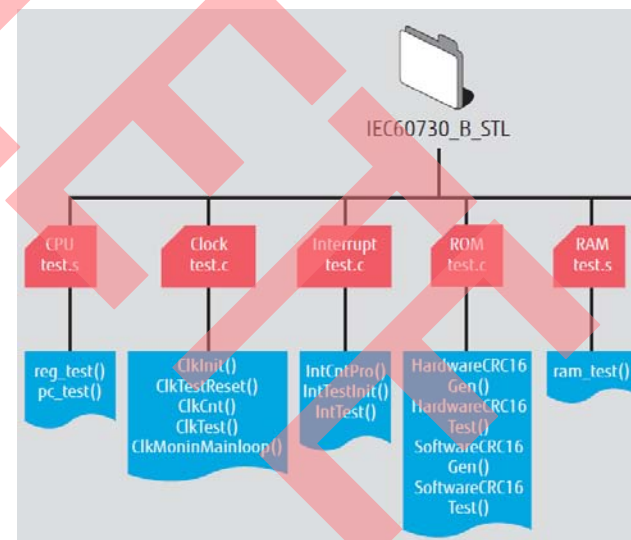
- FM3 inverter drive hardware features
  - Up to 3 ch flexible 3-phase motor timers, automatic dead time insertion
  - Up to 3 ch independent 12-bit 1Msps ADCs, up to 32 ADC inputs
  - Up to 3 ch ABZ quadrature decoder units
  - DTTI input for motor emergency stop
  - 3.3V and true 5V single supply guarantees robustness
- FM3 software motor control library
  - Support for BLDC, PMSM, IPM and ACIM
  - Field oriented control
  - Support for encoder or hall sensor feedback, or sensorless application



# Solutions

## FM3 safety

- FM3 functional safety hardware features
  - Two stage watchdog with independent clock source
  - Clock supervisor (clock failure and abnormal frequency detection)
  - On-Chip Low Voltage Detector
  - CRC hardware module
  - MPU (Memory Protection Unit)
  - DTTI input for motor emergency stop
- IEC60730 Class B
  - Self-Test Library available
  - CPU test
  - Clock test
  - Interrupt test
  - Memory test
  - I/O test
  - A/D converter test



Finally

OBVIOUSLY

# Workshops & Seminars

FM Seminar	Motor Control	USB Workshop	Ethernet Wo
Please register here: <a href="http://news.spansion.com/seminars">http://news.spansion.com/seminars</a>			
<ul style="list-style-type: none"> <li>• Overview FM family               <ul style="list-style-type: none"> <li>• Memory</li> <li>• Peripheral resources</li> <li>• Packages</li> </ul> </li> <li>• Processor architecture               <ul style="list-style-type: none"> <li>• Bus structure</li> <li>• Flash memory</li> <li>• Flash programming</li> </ul> </li> <li>• Peripheral resources               <ul style="list-style-type: none"> <li>• Clock distribution</li> <li>• Timer</li> <li>• Interfaces</li> <li>• FM features</li> </ul> </li> <li>• Development tool chains               <ul style="list-style-type: none"> <li>• IAR workbench / J-Link</li> <li>• KEIL <math>\mu</math>Vision / uLink</li> <li>• Starter Kits</li> </ul> </li> <li>• Practical exercises               <ul style="list-style-type: none"> <li>• Flash programming</li> <li>• Project setup/modification</li> <li>• Debugging</li> <li>• External interrupts</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• Introduction of Spansion MCU               <ul style="list-style-type: none"> <li>• Line-Up of microcontrollers with motion control features</li> <li>• Performance</li> </ul> </li> <li>• Introduction of motors types               <ul style="list-style-type: none"> <li>• ACIM</li> <li>• BLDC</li> <li>• PMSM</li> </ul> </li> <li>• Introduction of control types               <ul style="list-style-type: none"> <li>• Sinusoidal commutation</li> <li>• Field Orientated Control</li> <li>• Space Vector Modulation</li> </ul> </li> <li>• Peripherals of FM3/FM4 MCUs               <ul style="list-style-type: none"> <li>• Base Timer</li> <li>• Multifunction Timer</li> <li>• 12-bit A/D Converter</li> <li>• Quadrature Position and Revolution Counter</li> <li>• Interrupt Controller</li> </ul> </li> <li>• Hands-on exercise / SW-Example               <ul style="list-style-type: none"> <li>• BLDC motor with hall sensor</li> <li>• PMSM motor with field orientated control</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• Introduction of Spansion MCU               <ul style="list-style-type: none"> <li>• Line-op of USB MCUs</li> </ul> </li> <li>• USB vs. RS232               <ul style="list-style-type: none"> <li>• Historical Background</li> </ul> </li> <li>• Electrical Layer</li> <li>• USB Protocol               <ul style="list-style-type: none"> <li>• Enumeration Process (Descriptors &amp; USB Settings)</li> <li>• Transfer Types</li> <li>• Data Transfers</li> <li>• USB Class Concept</li> </ul> </li> <li>• Software Driver Concepts               <ul style="list-style-type: none"> <li>• USB Host</li> </ul> </li> <li>• USB Examples               <ul style="list-style-type: none"> <li>• Virtual COM Port</li> <li>• USB Descriptor Manager                   <ul style="list-style-type: none"> <li>• Create Template Classes</li> <li>• Create Descriptors</li> </ul> </li> </ul> </li> <li>• PC software based on LibUSB</li> <li>• Special Use Cases               <ul style="list-style-type: none"> <li>• e.g. boot loader</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• Introduction of Ethernet</li> <li>• Line-op of Ethernet</li> <li>• Fundamentals of Ethernet</li> <li>• Ethernet Microcontrollers</li> <li>• Hardware Design</li> <li>• Software Design</li> <li>• Communication</li> <li>• The Internet Protocol</li> <li>• Web technology in embedded systems</li> <li>• Developing Ethernet</li> <li>• Tools and methods</li> <li>• Practical hints and tricks on FM3 Ethernet</li> <li>• Hands-on training</li> </ul>



## Spansion Support

- Please check the following website, for any available updates

[www.spansion.com](http://www.spansion.com)

[www.spansion.com/starterkit](http://www.spansion.com/starterkit)

- Please contact your local support team for any technical questions

America: [Spansion.Solutions@Spansion.com](mailto:Spansion.Solutions@Spansion.com)

China: [mcu-ticket-cn@spansion.com](mailto:mcu-ticket-cn@spansion.com)

Europe: [mcu-ticket-de@spansion.com](mailto:mcu-ticket-de@spansion.com)

Japan: [mcu-ticket-jp@spansion.com](mailto:mcu-ticket-jp@spansion.com)

Other: <http://www.spansion.com/Support/SES/Pages/Ask-Spansion.aspx>

## Recycling

- Gültig für EU-Länder:
  - Gemäß der Europäischen WEEE-Richtlinie und deren Umsetzung in landesspezifische Gesetze nehmen wir dieses Gerät wieder zurück.
  - Zur Entsorgung schicken Sie das Gerät bitte an die folgende Adresse:
- Valid for European Union Countries:
  - According to the European WEEE-Directive and its implementation in national laws we take this device back.
  - For disposal please send the device to the following address:



**CCS Express GMBH**  
**c/o Spansion International Inc.**  
**Frankfurter Str. 83-107**  
**D-65479 Raunheim**  
**Germany**



- This board is compliant with China RoHS

The Spansion logo features a stylized purple wave above the word "SPANSION" in a bold, black, sans-serif font. A registered trademark symbol (®) is located to the upper right of the word. Below the word, there are two purple geometric shapes: a horizontal bar and a trapezoidal shape.

**SPANSION**<sup>®</sup>

[www.spansion.com](http://www.spansion.com)

Spansion<sup>®</sup>, the Spansion logo, MirrorBit<sup>®</sup>, MirrorBit<sup>®</sup> Eclipse<sup>™</sup> and combinations thereof are trademarks and registered trademarks of Spansion LLC in the United States and other countries. Other names used are for informational purposes only and may be trademarks of their respective owners.

This document is for informational purposes only and subject to change without notice. Spansion does not represent that it is complete, accurate or up-to-date; it is provided "AS IS." To the maximum extent permitted by law, Spansion disclaims any liability for loss or damages arising from use of or reliance on this document.

## X-ON Electronics

Largest Supplier of Electrical and Electronic Components

*Click to view similar products for [Development Boards & Kits - ARM category](#):*

*Click to view products by [Cypress manufacturer](#):*

Other Similar products are found below :

[SAFETI-HSK-RM48](#) [PICOHOBBITFL](#) [CC-ACC-MMK-2443](#) [EVALSPEAR320CPU](#) [TMDX570LS04HDK](#) [TXSD-SV70](#) [TXSD-SV71](#)  
[YGRPEACHNORMAL](#) [PICODWARFFL](#) [YR8A77450HA02BG](#) [3580](#) [VISIONSTK-6ULL V.2.0](#) [DEV-17717](#) [EAK00360](#)  
[YR0K77210B000BE](#) [RTK7EKA2L1S00001BE](#) [SLN-VIZN-IOT](#) [1410](#) [LPC-H2294](#) [CHIPKIT WI-FIRE](#) [QC-DB-H00003](#) [2125](#) [2126](#)  
[EVALSP1310CPU](#) [ATSAM4C32-EK](#) [ATSAM4CMP32-DB](#) [ATSAM4CMS32-DB](#) [ATSAM4CP16C-EK](#) [ATSAM4S-WPIR-RD](#)  
[ATSAMG55-XPRO](#) [2266](#) [ATSAM4CP16B-EK](#) [2390](#) [2419](#) [A20-OLINUXINO-LIME2](#) [2738](#) [2772](#) [WIZWIKI-W7500P](#) [SK-FM0-V48-](#)  
[S6E1A1](#) [3277](#) [STM32L4R9I-DISCO](#) [ATSAMC21N-XPRO](#) [ATSAMHA1G16A-XPRO](#) [EA-QSB-010](#) [EA-OEM-202](#) [ABX00012](#) [3241](#)  
[ATSAME54-XPRO](#) [32F072BDISCOVERY](#) [32F412GDISCOVERY](#)