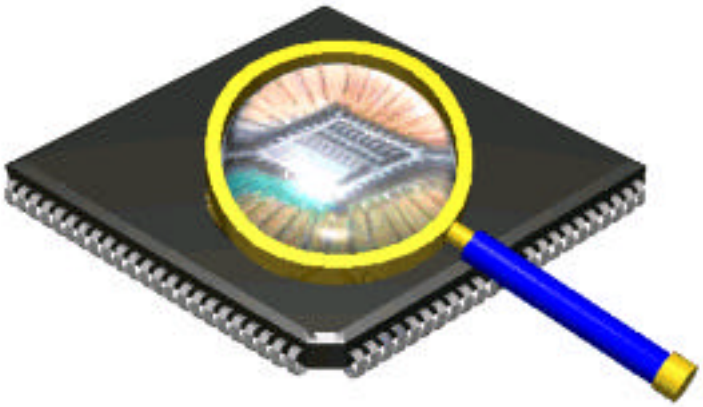


ACCEMIC MDE

Accemic MDE Std. for Fujitsu 16LX
User & reference manual

accelerate your microcontroller design

Schneewittchenstr. 2
81739 München
Germany

www.accemic.com

Disclaimer

All product names, specifications, prices and other information are subject to change without notice. While the information herein is assumed to be accurate,
Accemic GmbH & Co. KG (the manufacturer) assumes no responsibility for any errors or omissions. The manufacturer makes and you receive no warranties or conditions, express, implied, statutory or in any communications with you. The manufacturer specifically disclaims any implied warranty of merchantability or fitness for a particular purpose.
Should one of the above stipulations be or become invalid and/or unenforceable, the remaining stipulations shall stay in full effect.

Accemic MDE full version license terms

The following terms govern your use of the Accemic MDE software unless you have a separate written agreement with Accemic.

PLEASE READ THIS SOFTWARE LICENSE AGREEMENT ("LICENSE") CAREFULLY BEFORE USING THE SOFTWARE. BY USING THE SOFTWARE, YOU ARE AGREEING TO BE BOUND BY THE TERMS OF THIS LICENSE. IF YOU DO NOT AGREE TO THE TERMS OF THIS LICENSE, DO NOT USE THE ACCEMIC SOFTWARE AND (IF APPLICABLE) RETURN THE SOFTWARE TO THE PLACE WHERE YOU OBTAINED IT FOR A REFUND.

1. License.
The software, documentation and any fonts accompanying this License whether on disk, in read only memory, on any other media or in any other form (the “Accemic Software”) are licensed to you by Accemic GmbH & Co. KG or its local subsidiary, if any (“Accemic”). The Software is owned and copyrighted by Accemic or its third party suppliers. Your license confers no title or ownership in the Software and is not a sale of any rights in the Software. The licenses granted herein are limited to Accemic's intellectual property rights in the Accemic Software and do not include any other patents or intellectual property rights. You own the media on which the Accemic Software is recorded but Accemic and/or Accemic's licensor(s) retain title to the Accemic Software.

2. Permitted Uses and Restrictions.
Accemic grants you a license to Use one copy of the Software. "Use" means storing, loading, installing, executing or displaying the Software on a single computer at a time. **This License allows you to store, load, install or execute the Accemic MDE monitor kernel in a maximum of 100 target microcontroller at a time.** You may make one copy of the Accemic Software in machine-readable form for backup purposes only. The backup copy must include all copyright information contained on the original. Except as expressly permitted in this License, you may not decompile, reverse engineer, disassemble, modify, rent, lease, loan, sublicense, distribute or create derivative works based upon the Accemic Software in whole or part or transmit the Accemic Software over a network. In some jurisdictions, Accemic's consent may not be required for disassembly or decompilation. Upon request, you will provide Accemic with reasonably detailed information regarding any disassembly or decompilation. You may, however, transfer your rights under this License provided you transfer the related documentation, this License and a copy of the Accemic Software to a party who agrees to accept the terms of this License and destroy any other copies of the Accemic Software in your possession. **The Accemic software and all its deliverables are intended and must only be used in an EVALUATION LABORATORY ENVIRONMENT. Under no circumstances this version of Accemic MDE may be used in a product.** To the maximum extent permitted by applicable law, in no event shall the manufacturer and its suppliers be liable for any damages whatsoever (including but without limitation, consequential and/or indirect damages for personal injury, assets of substantial value, loss of profits, interruption of business operation, loss of information, or any other monetary or pecuniary loss) arising from the use of the product. Your rights under this License will terminate automatically without notice from Accemic if you fail to comply with any term(s) of this License.

3. Limited Warranty on Media (if applicable).
Accemic warrants the media on which the Accemic Software is recorded to be free from defects in materials and workmanship under normal use for a period of 1 year from the date of original retail purchase. Your exclusive remedy under this paragraph shall be, at Accemic’s option, a refund of the purchase price of the product containing the Accemic Software or replacement of the Accemic Software which is returned to Accemic or an Accemic authorized representative with a copy of the receipt. Some jurisdictions do not

allow limitations on how long an implied warranty lasts, so this limitation may not apply to you.

4. Disclaimer of Warranty on Accemic Software.
You expressly acknowledge and agree that use of the Accemic Software is at your sole risk. The Accemic Software is provided “AS IS” and without warranty of any kind and Accemic and Accemic's licensor(s) (for the purposes of provisions 4 and 5, Accemic and Accemic's licensor(s) shall be collectively referred to as "Accemic") EXPRESSLY DISCLAIM ALL WARRANTIES AND/OR CONDITIONS, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES AND/OR CONDITIONS OF MERCHANTABILITY OR SATISFACTORY QUALITY AND FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. ACCEMIC DOES NOT WARRANT THAT THE FUNCTIONS CONTAINED IN THE ACCEMIC SOFTWARE WILL MEET YOUR REQUIREMENTS, OR THAT THE OPERATION OF THE ACCEMIC SOFTWARE WILL BE UNINTERRUPTED OR ERROR-FREE, OR THAT DEFECTS IN THE ACCEMIC SOFTWARE WILL BE CORRECTED. FURTHERMORE, ACCEMIC DOES NOT WARRANT OR MAKE ANY REPRESENTATIONS REGARDING THE USE OR THE RESULTS OF THE USE OF THE ACCEMIC SOFTWARE OR RELATED DOCUMENTATION IN TERMS OF THEIR CORRECTNESS, ACCURACY, RELIABILITY, OR OTHERWISE. NO ORAL OR WRITTEN INFORMATION OR ADVICE GIVEN BY ACCEMIC OR AN ACCEMIC AUTHORIZED REPRESENTATIVE SHALL CREATE A WARRANTY OR IN ANY WAY INCREASE THE SCOPE OF THIS WARRANTY. SHOULD THE ACCEMIC SOFTWARE PROVE DEFECTIVE, YOU (AND NOT ACCEMIC OR AN ACCEMIC AUTHORIZED REPRESENTATIVE) ASSUME THE ENTIRE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO THE ABOVE EXCLUSION MAY NOT APPLY TO YOU. THE TERMS OF THIS DISCLAIMER DO NOT AFFECT OR PREJUDICE THE STATUTORY RIGHTS OF A CONSUMER ACQUIRING ACCEMIC PRODUCTS OTHERWISE THAN IN THE COURSE OF A BUSINESS, NEITHER DO THEY LIMIT OR EXCLUDE ANY LIABILITY FOR DEATH OR PERSONAL INJURY CAUSED BY ACCEMIC’S NEGLIGENCE.

5. Limitation of Liability.
UNDER NO CIRCUMSTANCES, INCLUDING NEGLIGENCE, SHALL ACCEMIC BE LIABLE FOR ANY INCIDENTAL, SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES ARISING OUT OF OR RELATING TO THIS LICENSE. SOME JURISDICTIONS DO NOT ALLOW THE LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES SO THIS LIMITATION MAY NOT APPLY TO YOU. In no event shall Accemic's total liability to you for all damages exceed the amount of four hundred Euro (400).

6. Export Law Assurances.
You may not use or otherwise export or reexport the Accemic Software except as authorized by Federal Republic of Germany law and the laws of the jurisdiction in which the Accemic Software was obtained.

7. High Risk Activities
The Software is not fault-tolerant and is not designed, manufactured or intended for use or resale as on-line control equipment in hazardous environments requiring fail-safe performance, such as in the operation of nuclear facilities, aircraft navigation or communication systems, air traffic control, direct life support machines, or weapons systems, in which the failure of the Software could lead directly to death, personal injury, or severe physical or environmental damage ("High Risk Activities"). Accemic and its suppliers specifically disclaim any express or implied warranty of fitness for High Risk Activities.

8. Controlling Law and Severability.
If there is a local subsidiary of Accemic in the country in which the Accemic Software License was obtained, then the local law in which the subsidiary sits shall govern this License. Otherwise, this License shall be governed by the laws of the Federal Republic of Germany. If for any reason a court of competent jurisdiction finds any provision, or portion thereof, to be unenforceable, the remainder of this License shall continue in full force and effect.

9. Complete Agreement.
This License constitutes the entire agreement between the parties with respect to the use of the Accemic Software licensed hereunder and supersedes all prior or contemporaneous understandings regarding such subject matter. No amendment to or modification of this License will be binding unless in writing and signed by Accemic. The parties hereto confirm that they have requested that this Agreement and all related documents be drafted in English.

In case of using a demo version of Accemic MDE, please read carefully the additional demo version license terms. The Accemic MDE demo version license terms contains additional restrictions of Use and Limitation of Liability.

Accemic MDE demo version license terms

The following terms govern your use of the Accemic MDE demo software unless you have a separate written agreement with Accemic.

PLEASE READ THIS SOFTWARE LICENSE AGREEMENT ("LICENSE") CAREFULLY BEFORE USING THE SOFTWARE. BY USING THE SOFTWARE, YOU ARE AGREEING TO BE BOUND BY THE TERMS OF THIS LICENSE. IF YOU DO NOT AGREE TO THE TERMS OF THIS LICENSE, DO NOT USE THE ACCEMIC SOFTWARE.

1. License.
The software, documentation and any fonts accompanying this License whether on disk, in read only memory, on any other media or in any other form (the “Accemic Software”) are licensed to you by Accemic GmbH & Co. KG, its suppliers or its local subsidiary, if any (for the purposes of all provisions, Accemic GmbH & Co. KG, its suppliers or its local subsidiary, if any, shall be collectively referred to as "Accemic"). The Software is owned and copyrighted by Accemic or its third party suppliers. Your license confers no title or ownership in the Software and is not a sale of any rights in the Software. The licenses granted herein are limited to Accemic's intellectual property rights in the Accemic Software and do not include any other patents or intellectual property rights. You own the media on which the Accemic Software is recorded but Accemic and/or Accemic's licensor(s) retain title to the Accemic Software.

2. Permitted Uses and Restrictions.
Accemic grants you a license to Use one copy of the Software. "Use" means storing, loading, installing, executing or displaying the Software on a single computer at a time. **This License allows you to store, load, install or execute the Accemic MDE monitor kernel in one target microcontroller at a time.** You may make one copy of the Accemic Software in machine-readable form for backup purposes only. The backup copy must include all copyright information contained on the original. Except as expressly permitted in this License, you may not decompile, reverse engineer, disassemble, modify, rent, lease, loan, sublicense, distribute or create derivative works based upon the Accemic Software in whole or part or transmit the Accemic Software over a network. In some jurisdictions, Accemic's consent may not be required for disassembly or decompilation. Upon request, you will provide Accemic with reasonably detailed information regarding any disassembly or decompilation. You may, however, transfer your rights under this License provided you transfer the related documentation, this License and a copy of the Accemic Software to a party who agrees to accept the terms of this License and destroy any other copies of the Accemic Software in your possession. **The Accemic software and all its deliverables are intended and must only be used in an EVALUATION LABORATORY ENVIRONMENT. Under no circumstances this version of Accemic MDE may be used in a product.** To the maximum extent permitted by applicable law, in no event shall the manufacturer and its suppliers be liable for any damages whatsoever (including but without limitation, consequential and/or indirect damages for personal injury, assets of substantial value, loss of profits, interruption of business operation, loss of information, or any other monetary or pecuniary loss) arising from the use of the product. Your rights under this License will terminate automatically without notice from Accemic if you fail to comply with any term(s) of this License.

3. Limited Warranty on Media.
Accemic **does not warrant** the media on which the Accemic Software is recorded to be free from defects in materials and workmanship.

4. Disclaimer of Warranty on Accemic Software.
You expressly acknowledge and agree that use of the Accemic Software is at your sole risk. The Accemic Software is provided “AS IS” and without warranty of any kind and

Accemic EXPRESSLY DISCLAIM ALL WARRANTIES AND/OR CONDITIONS, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES AND/OR CONDITIONS OF MERCHANTABILITY OR SATISFACTORY QUALITY AND FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. ACCEMIC DOES NOT WARRANT THAT THE FUNCTIONS CONTAINED IN THE ACCEMIC SOFTWARE WILL MEET YOUR REQUIREMENTS, OR THAT THE OPERATION OF THE ACCEMIC SOFTWARE WILL BE UNINTERRUPTED OR ERROR-FREE, OR THAT DEFECTS IN THE ACCEMIC SOFTWARE WILL BE CORRECTED. FURTHERMORE, ACCEMIC DOES NOT WARRANT OR MAKE ANY REPRESENTATIONS REGARDING THE USE OR THE RESULTS OF THE USE OF THE ACCEMIC SOFTWARE OR RELATED DOCUMENTATION IN TERMS OF THEIR CORRECTNESS, ACCURACY, RELIABILITY, OR OTHERWISE. NO ORAL OR WRITTEN INFORMATION OR ADVICE GIVEN BY ACCEMIC OR AN ACCEMIC AUTHORIZED REPRESENTATIVE SHALL CREATE A WARRANTY OR IN ANY WAY INCREASE THE SCOPE OF THIS WARRANTY. SHOULD THE ACCEMIC SOFTWARE PROVE DEFECTIVE, YOU (AND NOT ACCEMIC OR AN ACCEMIC AUTHORIZED REPRESENTATIVE) ASSUME THE ENTIRE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO THE ABOVE EXCLUSION MAY NOT APPLY TO YOU. THE TERMS OF THIS DISCLAIMER DO NOT AFFECT OR PREJUDICE THE STATUTORY RIGHTS OF A CONSUMER ACQUIRING ACCEMIC PRODUCTS OTHERWISE THAN IN THE COURSE OF A BUSINESS, NEITHER DO THEY LIMIT OR EXCLUDE ANY LIABILITY FOR DEATH OR PERSONAL INJURY CAUSED BY ACCEMIC'S NEGLIGENCE.

5. Limitation of Liability.
UNDER NO CIRCUMSTANCES, INCLUDING NEGLIGENCE, SHALL ACCEMIC BE LIABLE FOR ANY INCIDENTAL, SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES ARISING OUT OF OR RELATING TO THIS LICENSE. SOME JURISDICTIONS DO NOT ALLOW THE LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES SO THIS LIMITATION MAY NOT APPLY TO YOU. In no event shall Accemic's total liability to you for all damages exceed the amount of four hundred Euro (400).

6. Export Law Assurances.
You may not use or otherwise export or reexport the Accemic Software except as authorized by Federal Republic of Germany law and the laws of the jurisdiction in which the Accemic Software was obtained.

7. High Risk Activities
The Software is not fault-tolerant and is not designed, manufactured or intended for use or resale as on-line control equipment in hazardous environments requiring fail-safe performance, such as in the operation of nuclear facilities, aircraft navigation or communication systems, air traffic control, direct life support machines, or weapons systems, in which the failure of the Software could lead directly to death, personal injury, or severe physical or environmental damage ("High Risk Activities"). Accemic specifically disclaim any express or implied warranty of fitness for High Risk Activities.

8. Controlling Law and Severability.
If there is a local subsidiary of Accemic in the country in which the Accemic Software License was obtained, then the local law in which the subsidiary sits shall govern this License. Otherwise, this License shall be governed by the laws of the Federal Republic of Germany. If for any reason a court of competent jurisdiction finds any provision, or portion thereof, to be unenforceable, the remainder of this License shall continue in full force and effect.

9. Complete Agreement.
This License constitutes the entire agreement between the parties with respect to the use of the Accemic Software licensed hereunder and supersedes all prior or

contemporaneous understandings regarding such subject matter. No amendment to or modification of this License will be binding unless in writing and signed by Accemic. The parties hereto confirm that they have requested that this Agreement and all related documents be drafted in English.

Copyright notice

Trademarks

Contact / Registration

Please register the software via email. This way we can make sure you will receive updates or notifications of updates as soon as they become available.
For registration please provide the following information:

- Company name and address
- Your name and job title
- Your email address and telephone number

Please send this information to: register@accemic.com

Contact address

Accemic GmbH & Co. KG
Schneewittchenstr. 2
D-81739 Munich
Germany

info@accemic.com
www.accemic.com

Support

In case of getting problems with Accemic MDE, please check www.accemic.com for latest service packs of Accemic MDE. Be sure to read carefully this manual, particularly the “Using Accemic MDE” and the “Tips, Hints and Troubleshooting” section.

If there is no solution for a problem, please contact the sales agent where you purchased Accemic MDE.
They will be able to help you more quickly if you have the following information:

- Serial number
- Version number
- Your name, company name and address

Reporting Errors

Feels committed to provide you a very stable and helpful tool, we would be able to address any problems with Accemic MDE faster if you provide us with some details. Please send an email to support@accemic.com with

- detailed description of the malfunction
- detailed information about your environment (e.g. operating system, Accemic MDE software version, target CPU type, clock frequency)
- sample code for demonstration of the malfunction (if applicable)

We will try to fix the problem as soon as possible.

We hope that you find Accemic MDE helpful and easy to use. Please send us suggestions for improvement to comments@accemic.com.

Version of software, manual, availability

This manual describes the actual Accemic MDE version for the Fujitsu F16LX line of microcontrollers for use with the Fujitsu fcc907s compiler which is part of the FFMC-16 Family Softune Workbench.

Print date: 2002-01-25

About Accemic

The mission of Accemic is to help the customers to ACCElerate MICrocontroller designs. Accemic provides cost-effective, highly integrated, leading–edge software development tools for embedded development.
Accemic also provides consulting and training services in embedded development.

Table of Content

1 About this document

This guide describes how to install and use the Accemic MDE for the Fujitsu F16LX series of microcontrollers.

1.1 Assumptions

This guide assumes that you already have a solid knowledge of the following:

- The C-language
- The software-tools used to build your application (assembler, linker, compiler)
- The target MCU

For a complete description of Fujitsu Microcontrollers see the following references and the user's manual of the chip you are actually using:

- Fujitsu microcontroller CD
- FFMC-16 Family Softune Workbench, compiler and assembler manuals

1.2 How to use this manual

This Manual explains all the functions that Accemic MDE offers.
Before actually using Accemic MDE, you should read or at least glance over this manual in order to become familiar with the software.
First you should read the sections “Using Accemic MDE” and the “Tips, Hints and Troubleshooting” very carefully. Because the monitor kernel of Accemic MDE runs “parallel” with the application it is very important to understand the operating method of the monitor kernel.
Next you should follow the steps in the “Getting started” section to get familiar with Accemic MDE.
Now it should be easy to modify the demo project or to prepare your application for using it with Accemic MDE.

1.3 Typographic Conventions for Syntax

This manual uses the following typographic conventions for syntax:

Regular size Arial for normal text

`Regular size courier for AccemicMDE-functions mentioned in the text`

```
Reduced size courier new in a frame for program examples
```

Boldface Arial for very important sections

2 Introduction to Accemic MDE

Thank you for choosing Accemic MDE.

Accemic MDE (MDE = Monitor Debugging Environment) is a source-level and symbolic debugger for debugging embedded C applications. The debugger allows downloading of a program from the host PC and debugging it on the actual target system. A monitor kernel on the target system controls program execution and communicates with the host system via a serial port.
Accemic MDE combines powerful and extensive debugging features with an intuitive, easy-to-use visual interface.
Accemic MDE is designed to deliver functionality that will reduce the time spent for testing and debugging. The Accemic MDE is the first tool on the market permitting to debug Fujitsu 16LX microcontrollers in the single chip mode without emulator.
Accemic MDE offers features, which have always been required by microcontroller developers but were rarely to get on the market:

- put variables by drag & drop into the Watch Window
- figure large structures and arrays clearly as tree diagram
- understand the meaning of the control registers without looking up in the data book
- switch an IO port
- transmit messages and variables from current applications to the PC

This functions are usually done by one mouse-click.

Use of Accemic MDE is subject to the Accemic GmbH & Co. KG license terms. Using the software indicates your acceptance of these license terms. If you do not accept these license terms, you must return the software for a full refund.

2.1 Function range

Specially for the embedded system programmers Accemic has designed a number of productive debugging features:

Program download
Accemic MDE contains a fast flashing tool for programming the internal flash memory of the target. MHX-, ABS-, and Raw data can be used. A CRC-check guarantees error free programming. With 38400 Baud a flash rate of approx. 2.7 kByte/s is reached.

User interface
The display of the source code and the integrated disassembler contains useful functions for a debugger such as syntax highlighting, display of the current code line, display of possible break points, mixed mode representation etc.
An ergonomic interface with drag & drop functionality is provided.

Execution Functions
The behavior of the target MCU after a reset is programmable. The monitor kernel can be started directly, in order to load new programs and execute them under control. Beyond that there exists the "Run and Break"-mode. In this mode the user program starts automatically after reset, in case of needing debugging, it is possible to start the debugger with a special sequence of instructions.
Single stepping allows to watch a program being executed line by line. It is possible to step single lines of source or, step in or over procedure calls.
The “Call Function”-function offers the opportunity to call and test functions.

Breakpoints
The code breakpoints makes the program halt in critical places and observe values. A maximum of two breakpoints are supported, because the target provides only two ROM correction units, which are used for breakpoint functionality. Breakpoints can also be inserted while the application is running. Of course, breakpoints can be selected on all code addresses, even in RAM or ROM areas.

Processor status
Accemic MDE contains an extensive Processor Status Window for controlling IO-Ports and peripheral units of the target. Each of the control registers is accessible and provides the meaning of the control bits in plain text. So its very easy to control the peripheral units and to test different register settings.

Online watching
This feature manages to check the contents of memory and registers while the target program is running even without impairing the real-time capability of program execution. The update rate is adjustable. A highly optimized update manager guarantees a fast update process with minimal system load. The update manager updates only the variables visible on the screen at that very
moment, that it is possible to inspect large structures (e.g. char-arrays) very fast.

Transfer function
In addition to the online watching function some powerful messaging functions are implemented for linking into the application, that enable to send data and messages to the PC screen.

Watchdog timer update
A very helpful feature is the function `acc_WatchdogEnable()`. If this function is called, Accemic MDE will update the watchdog automatically.

Supported targets
At present the following targets are supported:
- MB90F443
- MB90F462
- MB90F497
- MB90F543, MB90F546, MB90F548, MB90F549
- MB90F553
- MB90F562, MB90F568
- MB90F574
- MB90F583
- MB90F591, MB90F594A
- MB90F598

The following 16LX microcontrollers with internal flash memory will follow soon:
- MB90F394
- MB90F474

Software protection
The Accemic MDE is equipped with a software copy protection. After first installation the full functionality is available for 10 days. Within this time a device-specific registration code must be requested by Accemic, with which the software is unlocked for permanent use. Transferring of a license from a full-licensed computer to another computer is supported.

2.2 Basic concepts

Accemic MDE has been designed to be as small and efficient as possible. The monitor uses the following target MCU resources:

- Internal Flash: Sector including 0xFFA000 (target depend) + 256 Byte
- Internal RAM: 128 Byte
- System stack 12Byte
- Bootloader-UART (target depend UART0 or UART1) with interrupts
- Both ROM-Correction units

Communication between PC and target MCU is made by the serial interface with up to 38400 Baud. Within the target MCU the PLL factor as well as the clock supply (internal or external) of the monitor UARTs are selectable. A reset of the target MCU can be released by the PC via the RTS or the DTR line.

There are two systems for controlling program execution: ROM-Correction and Bootloader-UART interrupts.
Accemic MDE uses the two ROM-Correction units for providing breakpoint functionality. If a match between the predefined ROM correction address and the actual program counter occurs, an INT9 will be requested. This non-maskable interrupt calls the monitor and sends a message to the PC.
For stopping the program execution the PC sends a command via UART to the target MCU and a receive interrupt occurs. This interrupt may have a smaller priority than other processes or the interrupts are globally disabled. Because of this it is not guaranteed that the command sent by the PC is regarded. For a proper function of the monitor it is very important to minimize the periods with global interrupt disabled or with a higher interrupt priority than the monitor interrupt.

3 Using Accemic MDE

This section describes the steps for getting an application running with Accemic MDE.

3.1 Installation

Accemic MDE is shipped on a CD-ROM or via web.
In order to install it, start the program “Install.exe” and follow the instructions.

In this manual, we will assume that the program is installed in the directory “C:\programs\AccemicMDE”.

After first installation of Accemic MDE the program will run in a full access mode for 10 days. In this time a registration must be performed for getting a system-specific unlock code. After 10 days without registration the program will switch to a demo mode with limited functionality. Even any manipulation of license files, system time etc. will switch the program into demo mode.

3.2 Getting started

Before using Accemic MDE with a target MCU, the monitor kernel must be flashed into the target. This procedure has to be done only once. After that the monitor kernel is permanently located inside the flash-memory.

Please keep in mind, that the Accemic MDE demolicense allows to store, load, install or execute the Accemic MDE monitor kernel in a maximum of one target microcontroller at a time. The demo version of Accemic MDE is limited to run only the Accemic MDE sample projects and to display in the Processor Status Window only a timer and the ADC.

One Accemic MDE full license allows to store, load, install or execute the Accemic MDE monitor kernel in a maximum of 100 target microcontroller at a time.

Accemic MDE is intended and must only be used in an EVALUATION LABORATORY ENVIRONMENT. Under no circumstances Accemic MDE may be used in a product. If you need more than 100 monitor kernels or if you want to use Accemic MDE in a product, please contact Accemic.

3.2.1 Compiling the sample project

Before downloading, the demo project must be compiled.
Accemic couldn't prepare precompiled sample files, because the *.abs file contains absolute path names.

1. Start FFMC-16 Family Softune Workbench and load the project c:\programs\AccemicMDE\Samples\90xxx\IOPort\abs\IOPort.abs. (xxx depends on the target MCU)

2. Build the project by clicking the "Build"-Button  or by performing the command Project - Build in the FFMC-16 Family Softune Workbench.

3.2.2 Loading the monitor kernel

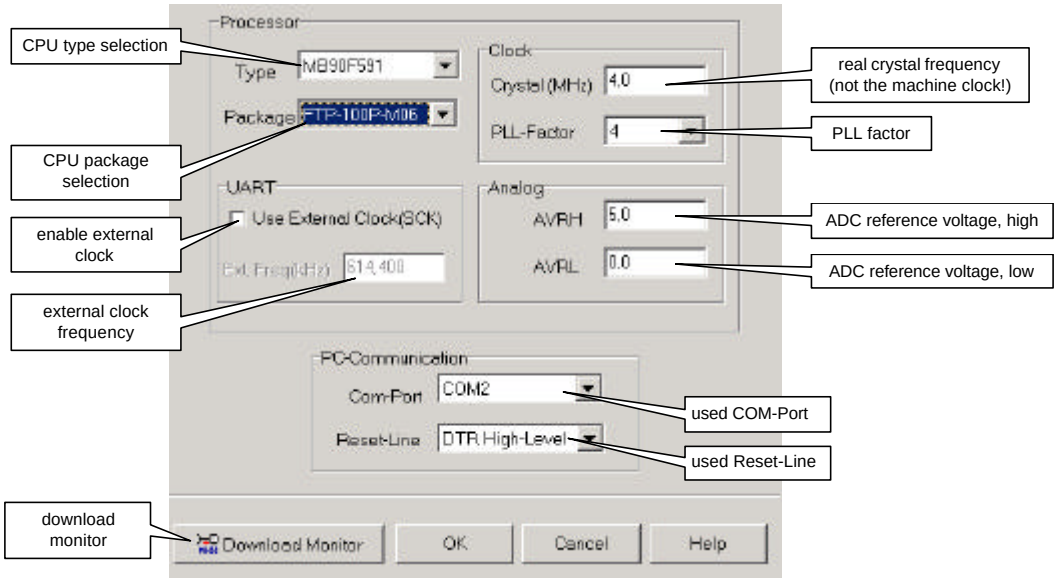
The following steps must be performed to flash the monitor kernel into the target.

1. Connect the serial interface of your PC to the Bootloader-UART of the target board. Here is a list of supported MCU Bootloader-UART's
   - MB90440 Family: UART1
   - MB90460 Family: UART0
   - MB90495 Family: UART1
   - MB90540/545 Family: UART1
   - MB90550 Family: UART0
   - MB90560 Family: UART1
   - MB90570 Family: UART0
   - MB90580 Family : UART0
   - MB90590 Family : UART0
   - MB90595 Family: UART1

2. To boot the flash device the Pins P00,P01,MD0, MD1 und MD2 must have the following states:
   P00 = Low
   P01 = Low
   MD2 = High
   MD1 = High
   MD0 = Low

3. Apply power to the target.

4. Start Accemic MDE, select the target MCU type and PC serial port connected to the target MCU in the System Preferences Window



5. Reset the target MCU if a reset line was not specified.

6. Pressing the “Download Monitor” button flashes the monitor kernel into the target MCU.

7. Put the target MCU in reset state if a reset line was not specified.

8. Switch the Pins P00,P01, MD0, MD1 und MD2 to the following states:
   MD2 = Low
   MD1 = High
   MD0 = High
   P00 = don’t care
   P01 = don’t care

9. Release the target MCU reset state, if a reset line was not specified.
   Now the target can be accessed through the monitor-kernel.

10. Press the “Connect”-Button  to start a debug session.

3.2.3 Loading the demo project

You received a ready to go demo project and it is a good idea to use it as a starting point of all your applications.

1. Start Select File - Load File… and select the c:\programs\AccemicMDE\Samples\90xxx\IOPort\abs\IOPort.abs.
   (*xxx* depends on the target – 495,540 or 590)
   The demo project will be downloaded.

2. Press F9 for starting the application.

3.3 How to prepare an application for debugging with Accemic MDE

With the following steps an application can be prepared for using with Accemic MDE.

1. Add the file “monitor.asm” to the project (located in the directory “C:\programs\AccemicMDE\include”).

2. If the target messaging functions will be used, include the file “monitor.h” into the application (located in the directory “C:\programs\AccemicMDE\include”).

3. Do not use functions that are changing the PLL Register, if you do not use an external clock for the Bootloader-UART. Otherwise a connection error will be received while a transfer is in progress. The clock will be initialized with the settings in the System Preferences Window by Accemic MDE.
   Note: The PLL register may be changed in the file “start.asm”.

```
; ------- Clock selection --------

#set  NOCLOCK        0         ; do not touch CKSCR register
#set  MAINCLOCK      1         ; select main clock (1/2 external)
#set  PLLx1          2         ; set PLL to x1 ext. clock/quartz
#set  PLLx2          3         ; set PLL to x2 ext. clock/quartz
#set  PLLx3          4         ; set PLL to x3 ext. clock/quartz
#set  PLLx4          5         ; set PLL to x4 ext. clock/quartz

#set  CLOCKSPEED     NOCLOCK   ; <<< set PLL ratio
#set  CLOCKWAIT      ON        ; <<< wait for stabilized PLL
```

Clock selection section in the file “start.asm”, be sure to set CLOCKSPEED to NOCLOCK

4. Do not use functions which are changing the ROM-Correction Registers

5. Do not use functions which are changing the Bootloader-UART interrupt level and the Bootloader-UART Registers. Be sure, that “vectors.c” don’t overwrite the Accemic MDE settings.

```
...
ICR13 = 7;      /*  IRQ37  IRQ38 */
/* ICR14 = 7;       IRQ39  IRQ40   UART1-Interrupts of monitor */
ICR15 = 7;      /*  IRQ41  IRQ42 */
...
```

Example for MB90F543: modification of “vectors.c”

Sometimes it is useful to set interrupt service routines to a higher priority than Accemic MDE communication interrupts. For doing this set the Bootloader-UART interrupt level to the requested level.
Keep in mind, that a communication error may occur, when the Bootloader-UART interrupt service routine will be disabled for a longer time.

```
...
ICR13 = 7;      /*  IRQ37  IRQ38 */
ICR14 = 5;      /*  IRQ39  IRQ40   UART1-Interrupts of monitor */
ICR15 = 7;      /*  IRQ41  IRQ42 */
...
```

Example for MB90F543: modification of “vectors.c” for changing Bootloader-UART interrupt level

6. Do not use or exclude the sections that are used by the kernel (see “monitor.asm” for detail) in the project linker preferences, especially the normal code entry-point address 0xFFA000.

7. It is recommended to disable all optimization options in the C-Compiler preferences. This options are controlled by Project – Setup Tool Options – C-Compiler – Category: Optimize in the FFMC-16 Family Softune Workbench environment.

8. Simple applications without interrupts (e.g. the Fujitsu sample project IOPort.prj) needs to add the instruction __EI().

```
void main(void)
{
  __EI() ;

  // your code
}
```

Add the __EI()-instruction to “main.c”

9. If the application program uses the Watchdog Timer, enable the automatic Watchdog Timer update function of Accemic MDE.
If the function `acc_WatchdogEnable()`was called, the monitor kernel of Accemic MDE automatically updates the Watchdog Timer in case of entering a breakpoint or receiving a stop command. Note, the watchdog timer can be stopped only by power-on, hardware standby or reset by watchdog timer himself. This means, before reloading a project with Watchdog Timer enabled, the target MCU must be reseted.

```
...
WDTC = 0x03;           // start watchdog timer
acc_WatchdogEnable();
...
```

Add the acc_WatchdogEnable() function after enabling Watchdog timer.

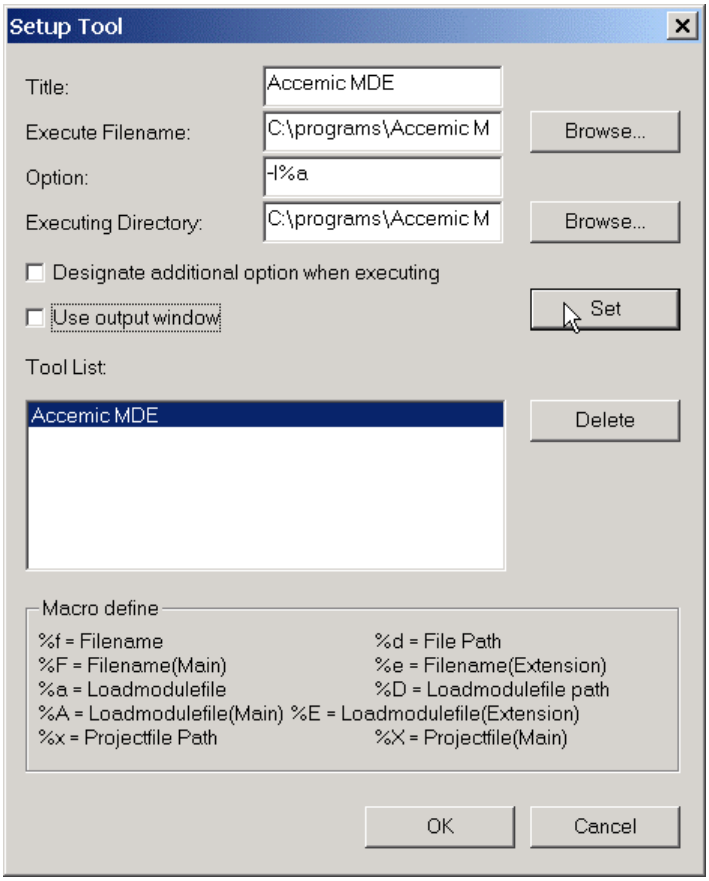
3.4 How to implement Accemic MDE into the FFMC-16 Family Softune Workbench environment

1. Open the FFMC-16 Family Softune Workbench environment and go to **Setup/Tools…** in the menu bar.

2. Enter Accemic MDE as tool name.

3. Press **Browse…** in the Executable Filename line and search for "MDE.exe". In the options line enter “–l%a” (l is a small “L”)

Now Accemic MDE can be started by selecting **Setup/Tool execution/Accemic MDE** in the menu bar.



Note:
FFMC-16 Family Softune Workbench does not support path names or file names which contain spaces (blanks).

3.5 How to update the watchdog timer by Accemic MDE

If the function `acc_WatchdogEnable()` is called, Accemic MDE updates the Watchdog Timer automatically, if Accemic MDE breaks the application program execution (e.g. entering a breakpoint, stop command, single stepping).

Prototype
```
void __far acc_WatchdogEnable(void)
```

Return value
none

defined in
"monitor.h"

```
...
WDTC = 0x03;           // start watchdog timer
acc_WatchdogEnable();
...
```
Add the acc_WatchdogEnable() function after enabling Watchdog timer.

The watchdog timer can be stopped only by power-on, hardware standby or reset by watchdog timer himself. This means, before reloading a project with Watchdog Timer enabled, the target MCU must be reseted.

Accemic MDE connects the target MCU ca. 50 times/second, even if no breakpoint was enabled. Each connection updates the watchdog timer. An application program malfunction can be hidden, because Accemic MDE updates the watchdog timer instead of the application program.

3.6 How to use the Bootloader UART by application

Sometimes the Bootloader UART is already used by application periphery and must be shared with the monitor kernel. The function `acc_KernelUART()` provides a simple mechanism to manage the dual use of the UART.

Originally Accemic MDE redirects the interrupt vectors of the Bootloader UART by modifying the interrupt vector table. With the function `acc_KernelUART()` the Bootloader UART interrupt vectors can be (re-) redirected to the application (on its original address).

Prototype

```
void __far (* __far acc_KernelUART)(char Disable)
```

Parameter	Meaning	
`Disable`	0	UART interrupt will be handled by monitor kernel (initial state)
	1	Redirection of the UART interrupt vectors to the original address, UART interrupts will be handled by application

Return value

none

defined in

“monitor.h”

Example

This example demonstrates controlling the usage of the Bootloader interrupt by an external port pin. If P00 == 1 the application interrupt vector will be called.
The application should be in automatic start mode

```
...
if (PDR0_P00)          // control usage of bootloader UART by P00
{ acc_KernelUART(1);   // Bootloader UART is used by application
}
...
```

4 Tips, Hints and Troubleshooting

This section contains some hints and tips about solving problems that may occur when using Accemic MDE.

4.1 Data monitoring

There are two possibilities for monitoring data while the application is running:

A fast and flexible way is enabling the Watch Button in the Browser Window, in the Memory Window or in the Watch Window.
Since the refreshing of the contents is independent from the data structures (sequential reading from memory areas) it may occur, that the displayed data are not correct.

Example:
An integer variable will be incremented by an event. Before reading the LSB of the variable the value is 0x10FF. The LSB of the value (0xFF) will be transmitted to the PC. Before the PC reads the MSB, the counter increments and the new counter value is 0x1100. Now the PC reads the LSB (0x11) and the PC displays a wrong value: 0x11FF.

This malfunction may occur only during online watching. If a breakpoint was entered or a "Stop" command has been sent, the system was stopped and the variables couldn't be changed during transfer to the PC. (Exception: routines with higher interrupt priority or registers of non-stopped peripheral units, e.g. timers can cause the described malfunction)

Alternatively the target messaging functions can be used for correct data transfer to the PC. The disadvantage is, that the required functions must be compiled into the source code.

4.2 Monitor interrupt priority

There are two types of monitor interrupts:
Entering a breakpoint or communication events on the monitor UART.

On entering a breakpoint, the ROM-correction unit will generate a non-maskable interrupt (INT9). This interrupt **also** occurs when all interrupts are disabled.

Note:
In single step mode the breakpoints are disabled, because the two ROM-correction units are needed for single stepping. A step over a function, which contains a breakpoint will **not** jump to the breakpoint in the function, because this breakpoint is temporarily disabled.
If the Bootloader-UART used by Accemic MDE receives a byte, the UART interrupt service routine is entered, if the interrupts are enabled and the UART interrupt level is higher then the interrupt level of the actual running process.
To prevent communication errors of the monitor kernel (buffer overflow) it is very important to

- minimize the times in which the application is in a higher interrupt level then the monitor UART
- minimize the periods in which the interrupts are disabled

4.3 Source code optimization

If the compiler uses the inline assembly functionality, some breakpoints may not be visible.

For initial testing of a system it is hardly recommended to avoid using the optimization function of FFMC-16 Family Softune Workbench C-compiler.

4.4 Programmers rules

Please keep in mind, that the monitor kernel of Accemic MDE runs “parallel” with the application.
Please notice the following rules to avoid disturb the monitor kernel:

- Don’t write the RAM area of the monitor-kernel (defined in the file “monitor.asm”)
- Don’t access the Bootloader-UART used by the monitor-kernel. This includes the Bootloader-UART register, the corresponding IO-Ports, Interrupt-Control-Register and interrupt vectors. Please check, if in the file “vectors.c” don’t change the Interrupt-Control-Register and interrupt vectors.
- If the Bootloader-UART isn’t clocked by external clock, the access to the PLL configuration registers is prohibited. Please check in the file “start.asm”, that the “CLOCKSPEED” is set to “NOCLOCK”.
- Don’t access the ROM Correction unit and the corresponding interrupt vectors.
- The application is uninterruptible by the monitor kernel, if the application interrupt level is higher or equal than the monitor kernel (which is the monitor-kernel UART interrupt level) or if the application has the interrupts disabled (e.g. __DI() instruction).

Only change the Mode-Pins (MD0, MD1, MD2, P00, P01), when the device is in reset state. Otherwise the device may enter undocumented states.

4.5 Troubleshooting Guide

I couldn't connect my target

Check

- Cable connection between PC and target (connections on 2-2, 3-3 , 5-5; good contacts, no broken lines, no X-modem cable used)?
- Charge pump capacitors of the RS232-Interface (for example MAX232) too small?
- Power supply okay?
- Wrong setting of Mode Pins (MD0, MD1, MD2, P00, P01)?
- Applying Reset after changing Mode pins?
- Target in Reset state (RST- and HST-Line)?
- HST-Line low or open?
- Connecting target with other flash utility (e.g. Softune\Utility\Flash510.exe) is possible?

Connection error while downloading

Check

- Cable connection between PC and target (good contacts, no broken lines)?
- Charge pump capacitors of the RS232-Interface (for example MAX232) too small?
- Power supply instable (keep in mind, that during flashing the target needs a higher current).
- For MB90F591 and the MB90F594A Accemic MDE must send the complete download stream before getting the first acknowledge. In case of using this MCUs please check the "I couldn't connect my target" hints.

Connection error after starting target application

Check

- Does the application program calls the __EI() function at the begin of "main.c"?
- Does the application program disable interrupts (__DI()) for a longer time (or forgot the following __EI() instruction)?
- Does the application program change the interrupt priority of the Bootloader-UART?
- Does the application program start the watchdog timer without calling the `acc_WatchdogEnable()` function?
- Does the application program calls the `acc_KernelUART()` function?

A breakpoint in an interrupt service routine will not be reached

Check

- Does the application program sets the interrupt level higher then 7 and are the interrupts enabled (__EI() instruction) - this is our most common newbe error ;-)

4.6 License Transfer

From time to time, there may be the need to move the full-licensed Accemic MDE software from one machine to another. Accemic MDE has a mechanism for transfer a license via a blank floppy disk.

If the PC (Transmit PC) has a full-licensed Accemic MDE version, this license can be exported to another PC (Receive PC). After transferring the license the Transmit-PC will run in demo mode, the Receive- PC will run with a full-licensed Accemic MDE.

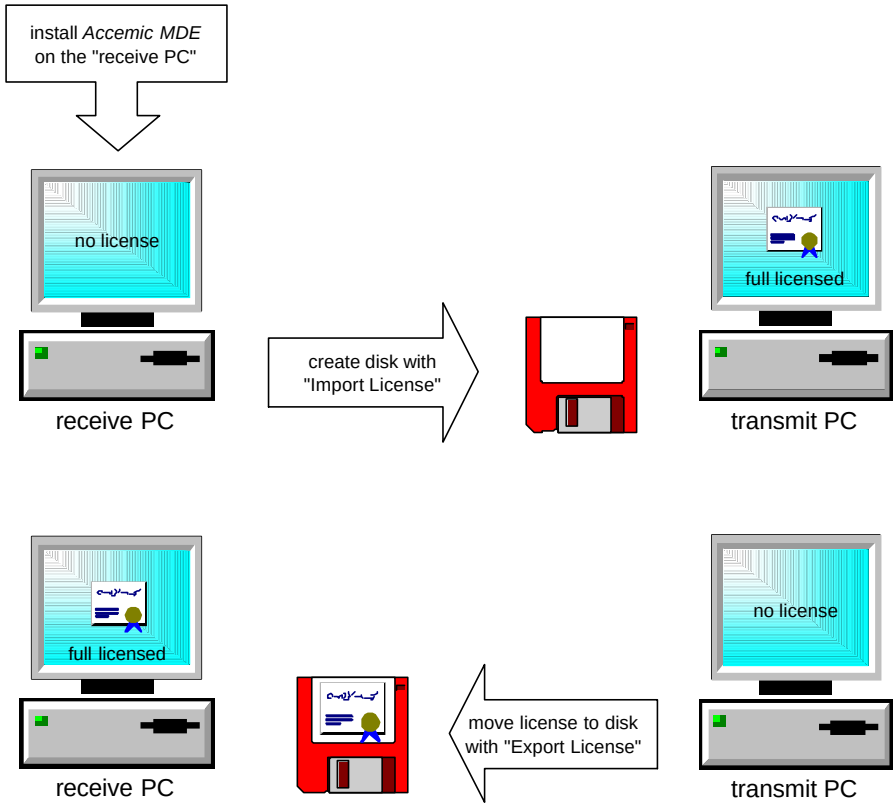
The License transfer feature of Accemic MDE is intended for occasional transfer of the license. Transferring the license requires close-together computers, because the floppy disk must be transferred from the non-licensed computer to the licensed computer and back to the non-licensed computer.

Tip
Before changing the hardware configuration on the full-licensed computer, please transfer the license to another computer, change the hardware and then transfer the license back.

Step 1: To transfer a license from the Transmit PC to the Receive-PC, first the Accemic MDE must be installed on the Receive-PC. This software will be unauthorized, so it will run in unregistered mode or in demo mode.
After installation select “Import License” and follow the instruction to put a formatted floppy disk in drive A: of the Receive-PC.

Step 2: Select “Export License” on the Transmit-PC and follow the instruction to put the floppy disk (created in Step 1) in drive A: of the Transmit-PC. During this step, the License File on the floppy is enabled and the License File on the Transmit PC is disabled.

Step 3: The last step is to simply move the floppy disk from the Transmit-PC to the Receive-PC for enabling Accemic MDE in the Receive-PC.



5 Menus

Accemic MDE provides an accelerator bar and short cuts which provide quick access to frequently used debugging commands.
Most of the operations available through the menus can also be performed via buttons in the buttons bar or via short cuts.

5.1 File Menu

The File Menu is used to select the application for debugging. The following selections are available from the File Menu:

Load 
Loads an application into the flash of the target. The following types can be used:

- ABS-Files: This file includes the code and all necessary debug information. The FFMC-16 Family Softune Workbench environment will automatically generate it.
- MHX-Files: This file type includes only code. No debug information is available.
- Unspecified file type (*.*): This file type includes only raw data. The data will be written to the specified destination address.

Reload  ***(Ctrl+R)***
Lists the last used applications. Select one for downloading into the flash of the target.

Exit (*Ctrl+X)*
Terminates the Accemic MDE debugging session.

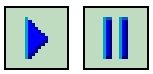
5.2 Start Menu

The Start Menu is used to control the execution of the application. It provides features for starting the application from a reset state, halting the application, resuming execution after a halt, and single-stepping. The following selections are available from the Start Menu:

Connect 
Connects Accemic MDE to the target. This function will be automatically done before loading a file.

Reset
This menu item will only appear if a reset line in the System Preferences Window is selected. The target is set to its initial state.

Restart  ***(Ctrl+R)***
The program counter is set to its initial value and program execution will begin at the address of your application's startup code. If no hardware reset of the target is done, this means no registers are modified except the program counter.
The application will then be stopped at the address of your application's startup code or at the main function, if you have enabled this option in the General Options Window.

Run / Stop  ***(F9)***
Starts and stops the current loaded application.

Step over  ***(F8)***
The type of step (i.e., source statements or machine instructions) is controlled by the current active window.

If the current active window is a Source Window and the stepped instruction represents a call to a function, target execution will be halted at the instruction immediately following the function call, and the Source Window viewing and execution positions will be moved accordingly.

If the current active window is the Disassembler Window, a Step Over on a 'jump to subroutine' instruction will halt target execution at the instruction immediately following the subroutine call.

Note: During single stepping mode, breakpoints are not available.
Explanation: Because the ROM-Correction unit is used by the stepping engine, the ROM-Correction unit is not available for regularly breakpoints. Because of that, the system will not stop if you step over a function having a breakpoint inside.

Step into  ***(F7)***
The type of step (i.e. source statements or machine instructions) is controlled by the current active window.

If the current active window is a *Source Window* and the stepped instruction represents a call to a function for which symbol information is available, target execution will be halted at the next appropriate instruction in the called function, and the *Source Window* viewing and execution positions will be moved accordingly.

If the current active window is the *Disassembler Window*, a Step Into on a 'jump to subroutine' instruction will halt target execution at the next appropriate instruction in the called subroutine.

Note: During single stepping mode, breakpoints are not available.
Explanation: Because the ROM-Correction unit is used by the stepping engine, the ROM-Correction unit is not available for regularly breakpoints. Because of that, the system will not stop if you step over a function having a breakpoint inside.

Return from Call  ***(F6)***
This feature is only available while stepping into a function/instruction call. The Target execution will be resumed from the current execution position until it returns from the current function.

Note: The 'Jump To…'-function is available in the *Source Window* and in the *Disassembler Window.*

5.3 View Menu

The View Menu is used to give detailed information about the target MCU and the loaded application. The following selections are available from the View Menu:

Processor  **(Ctrl+P)**

Processor  ***(Ctrl+P)***
Display of the Processor Status Window.

Memory  ***(Ctrl+M)***
Display of the Memory Window.

Register  ***(Ctrl+R)***

Stack  ***(Ctrl+S)***
Display of the Stack Window.

Disassembler  ***(Ctrl+D)***
Display of the Disassembler Window.

Breakpoints  ***(Ctrl+B)***
Display of the Breakpoint Window.

Vectormap ***(Ctrl+V)***
Display of the Vectormap Window.

Targetinfo ***(Ctrl+T)***
Display of the Targetinfo Window.

Register
Display of the Register Window.

Target Messages
Display of the Target Messages Window.

Browser
Display of the Browser Window, only available if an ABS-File is loaded.

Watch  ***(Ctrl+W)***
Display of the Watch Window, only available if an ABS-File is loaded.

Sectionmap
Display of the Sectionmap Window, only available if an ABS-File is loaded.

5.4 Tools Menu

The Tools Menu bar contains the following items:

Erase flash
Selecting this item will erase the target MCU flash-memory except the debug-kernel (see monitor.asm for details) and its used interrupt vectors.
For complete chip erase use the Softune tool \Softune\Utility\Flash510.exe or other flashing tools.

Automatic Start
This item specifies the start mode.
If “Automatic start” is checked, the application starts immediately after Reset (“Run and Break”). Of course, in this mode Accemic MDE can connect the target.
If “Automatic start” is not checked, the monitor kernel starts. The application will be stopped on “main” or on the code start address, dependent of the settings in the General Options Window.

The current automatic start mode is determinable in the Targetinfo Window or by the checked state of the Automatic Start item.
The automatic start mode is only settable while an application is still loaded into the device (see Targetinfo Window).

Call function
This feature allows calling functions independently from a running user application

5.5 Preferences Menu

The Preferences Menu bar contains the following items:

System
Display of the System Preferences Window

General
Display of the General Options Window

License
The License item contains the following subitems:

- **Registration**
  Display of the Registration Window for entering serial number and contact information.
- **Unlock**
  Display of the Unlock Window for entering the unlock code. This item is available, if the software is not in full licensed state.
- **Import License**
  Display of the Import License Dialog. This item is available, if the software is not in the full licensed state.
- **Export License**
  Display of the Export License Dialog. This item is available, if the software is in the full licensed state.

5.6 Window Menu

The following selections are available from the Window Menu:

Tile vertical 
Tiles the source windows vertical.

Tile horizontal 
Tiles the source windows horizontal.

Cascade 
Cascades all opened source windows.

Arrange
Arranges all minimized windows on the bottom.

Minimize all
Minimizes all windows.

5.7 Help Menu

The Help menu bar contains the following items:

Contents  **F1**
Opens the help file

Search
Search the help file

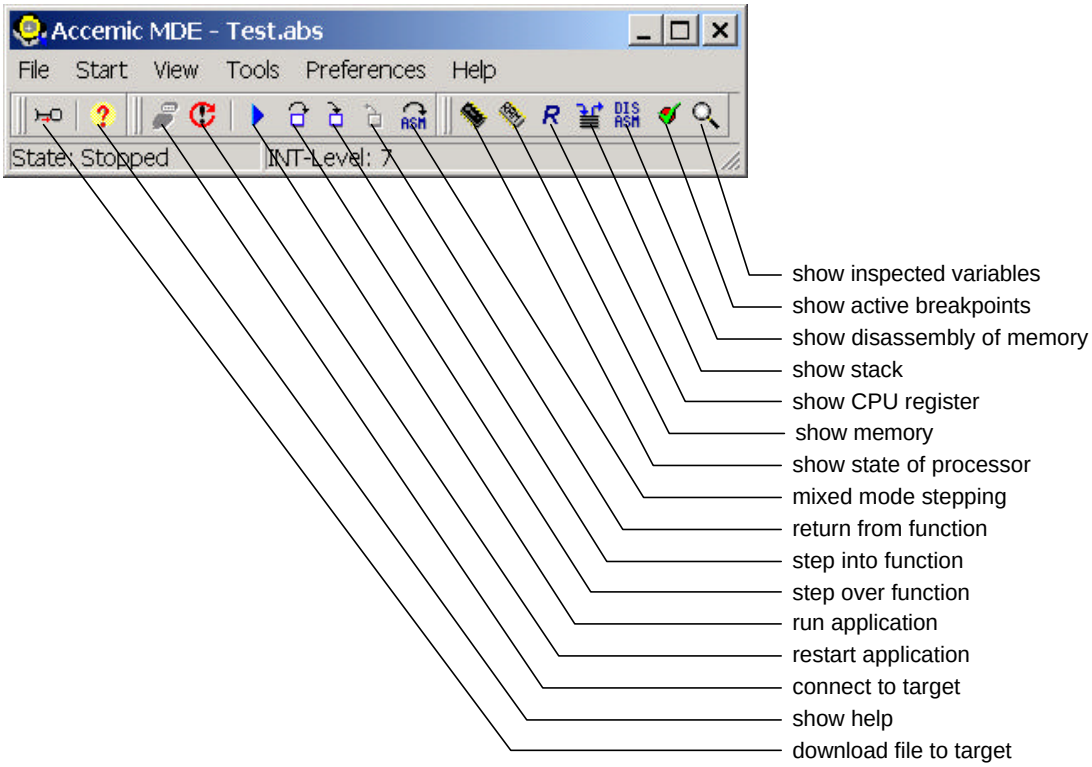
Accemic Homepage
Jumps to www.accemic.com

About
Shows information about Accemic MDE

6 Windows

6.1 Main Window

The ***Main Window*** is divided into different parts.
On the top there are the menu bar and the button bar. Each button has a link to a menu item.
The status bar on the bottom is divided into two parts. The left side shows hints about items under the mouse. The right side shows the state of the target connection.

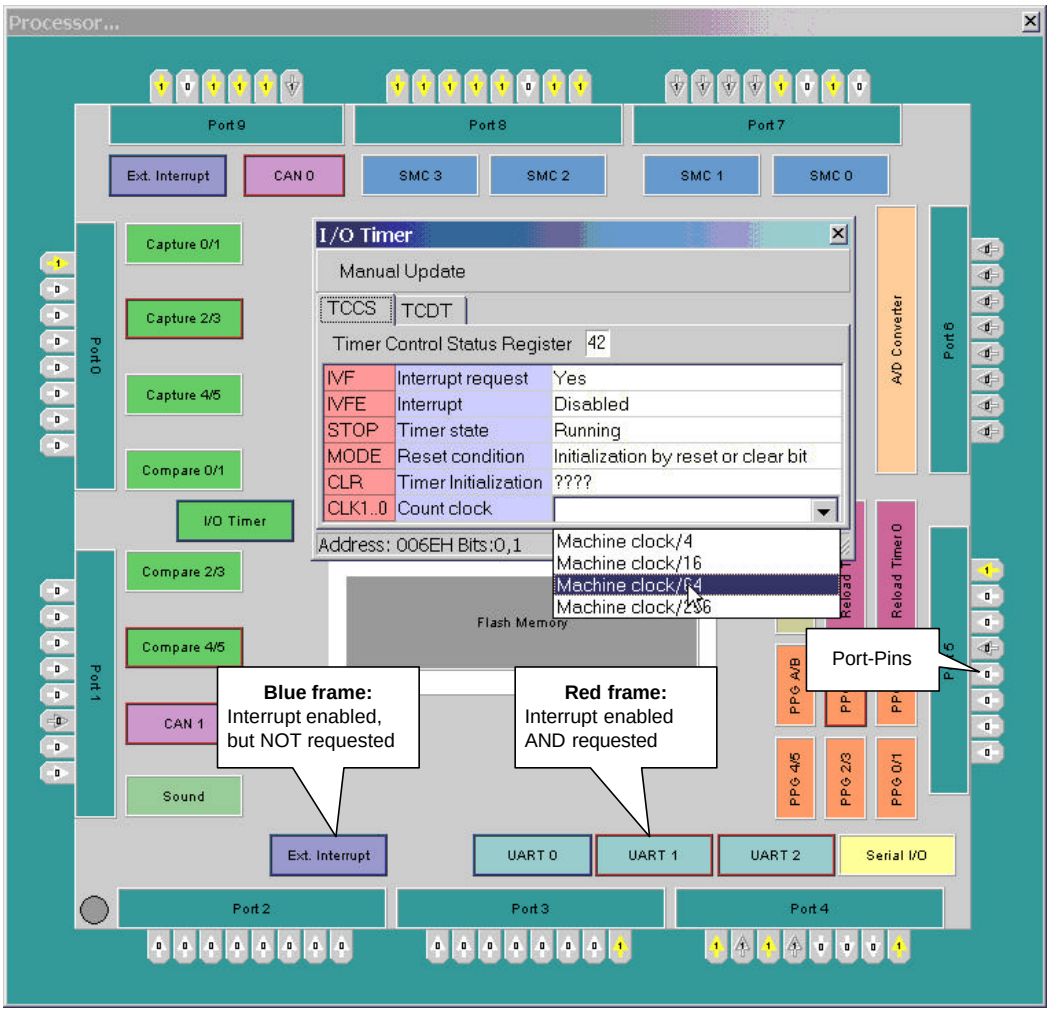


6.2 Processor Window

The Processor Window gives detailed information about the currently used device.

Core
In the middle of the screen the core and its modules are visible.
By clicking on a module an information window will appear, where specified informations about the module are displayed. In addition, the visible registers of the selected module can be edited by double-clicking on it.
The information windows can be closed by mouse right-clicking.

Port Pins
Around the core there are displayed the state and the direction of the port pins. Used pins are displayed light-grayed.
By moving the mouse over one of the pins a hint will be displayed, which port pin it is.
In case of using the pin by an internal module, additionally the name of the module will be displayed.
By clicking on a non-grayed pin, a popup-menu will appear for editing the direction and the state of the pin.



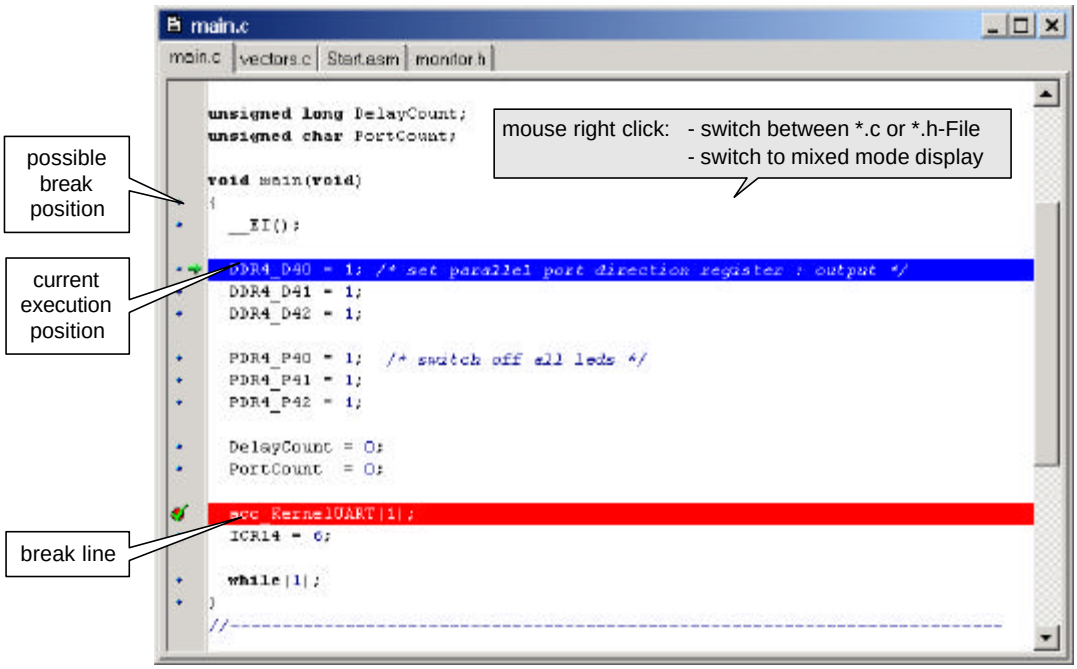
6.3 Source Window

The Source Window displays a source file(Header/C/ASM) .
A blue line indicates the current execution-position if the application is stopped. A red line indicates a breakpoint.

On the left side a blue circle for each line indicates a possible breakpoint.
By left-clicking on the blue circle a breakpoint will be enabled or disabled.
By right-clicking on a blue circle the application jumps to this location.

If the Source Window has the focus, stepping will be done with C-instructions.

Navigation in this window can be done with the scroll boxes, the mouse wheel or the keys Cursor Up/Down, Cursor Left/Right, Page Up/Down or Pos1/End.
A blue line indicates the current address-position if the application is stopped. A red line indicates a breakpoint.



By double clicking a variable this variable can be moved to the Watch Window or can be displayed in the Browser Window.

By clicking the right mouse button a Popup-Menu will be called. In this menu there are the following options (apart from various window calls):

Source/Header-File
If the current file is a C source code file (*.c), the header file (*.h) with the same name will be opened, if it exists.
If the current file is a header file (*.h), the C source code file (*.h) with the same name will be opened, if it exists.

Mixed Mode

The C source code will be displayed in mixed mode. If the assembler stepping button 

in the tool bar is pressed, the single stepping command will step with Assembler-instructions, otherwise with C-instructions.

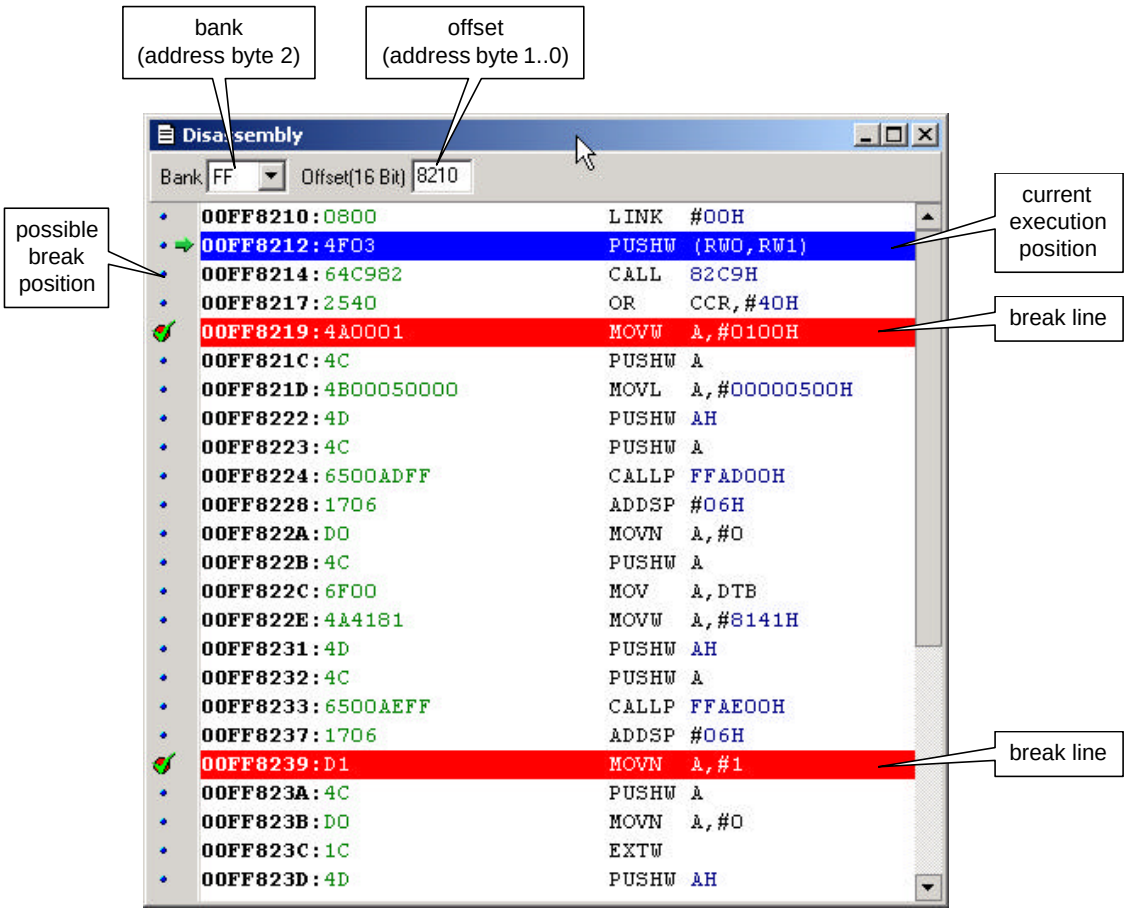
6.4 Disassembler Window

In the Disassembler Window the disassembled instructions are displayed. If the Disassembler Window is opened, the disassembler will automatically decode the current execution position. Another address can be selected with the 64K bank address and a 16 bit offset in the disassembler tool bar.
A blue line indicates the current execution-position if the application is stopped. A red line indicates a breakpoint.

On the left side a blue circle for each line indicates a possible breakpoint position.
By left-clicking on the blue circle a breakpoint will be enabled or disabled.
By right-clicking on a blue circle the application jumps to this location.

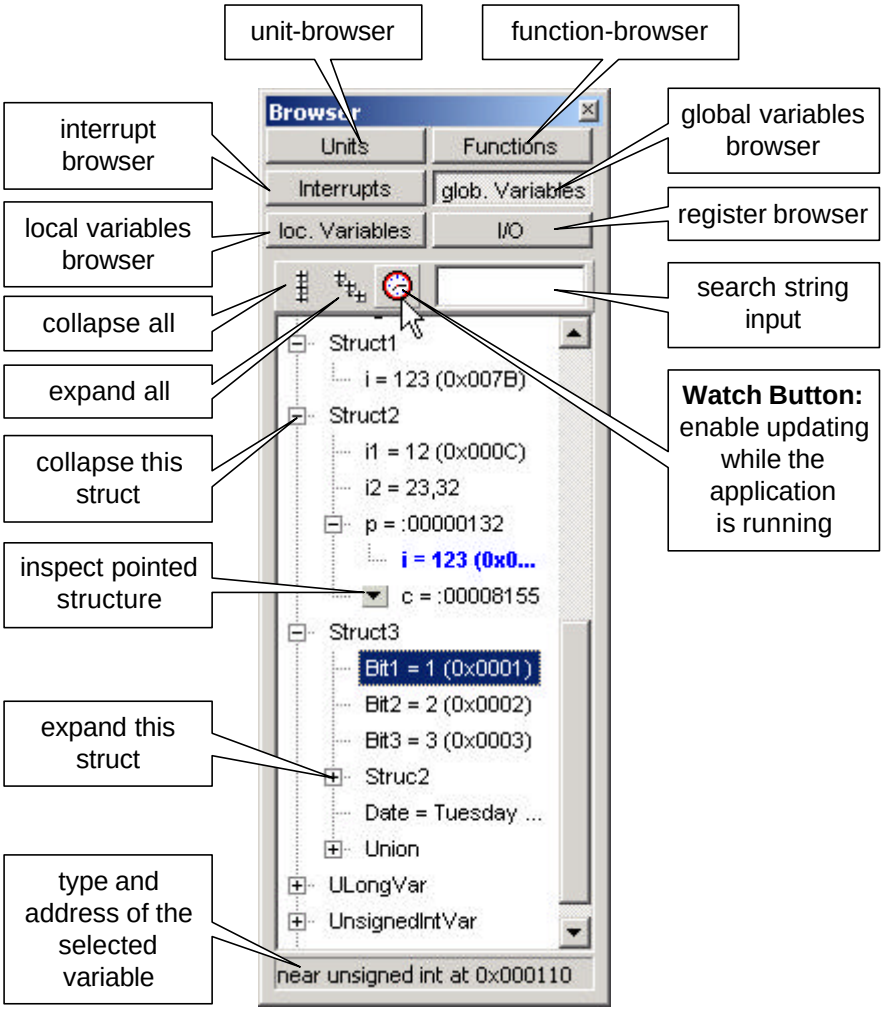
If the Disassembler Window has the focus, stepping will be done by instruction.

Navigation in this window can be done with the scroll boxes, the mouse wheel or the keys Cursor Up/Down, Cursor Left/Right, Page Up/Down or Pos1/End.



6.5 Browser Window

If an ABS-File is loaded, browsers are visible on the left side of the main window. There are browsers for units, functions, interrupts, global and local variables.



With the collapse/expand buttons all variables can be collapsed or expanded. By right clicking onto a variable the selected node only can be collapsed or expended.
The type and the address of the variable will appear in the bottom status line.

Accemic MDE supports the drag&drop ability for copying one value to another or to copy a variable to the Watch Window. It is also possible to add a variable to the Watch Window by right clicking onto the variable and choosing the “Add to Watch” item.

Note: For copying one value to another, the value of the variable and not the name must be selected.

Watch function
By enabling the Watch Button the expanded variables will be updated while the application is running. The update interval is adjustable.
Changed values are highlighted for easy identification.

Search function
Simple input the first characters of the requested function, variables or registers. All matching items in the actual browser window will be displayed.

Unit Browser
In this list box all units which are used by the application are displayed. By double-clicking on a unit, this unit will appear in the Main Window.

Interrupts & Function Browser
These browsers are displaying all functions and interrupts used by the application. By double-clicking on a function/interrupt the definition file will appear in the Main Window. By right-clicking a function name the Call Function feature of Accemic MDE can be called.

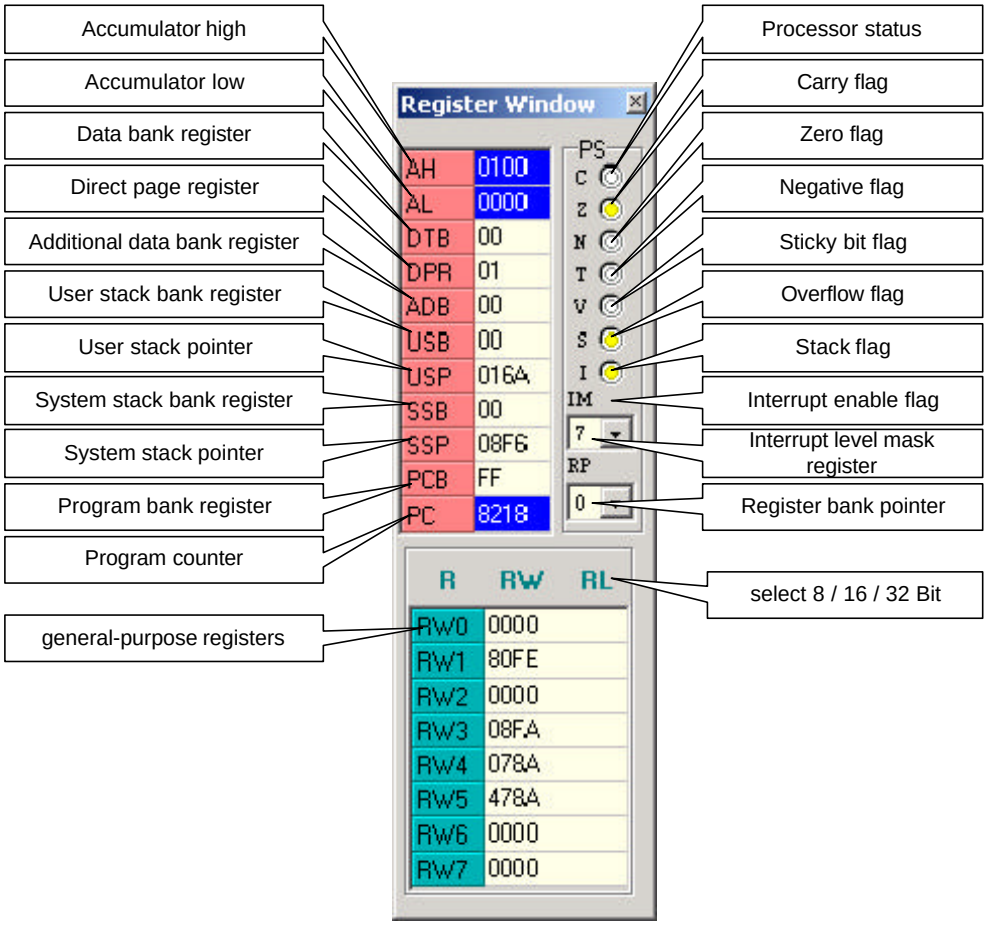
Global Variables Browser
In this tree all global variables of the application are displayed. By double-clicking onto a value the Change Value Window will appear.
Pointers, structures and arrays have a compact and expanded form. The expanded form shows all the fields of a pointer, struct or array. The structures are expandable within structures (nesting). Variable values that cannot be evaluated yet, will be displayed with '????'.
Evaluation of a pointer can be done by clicking on the inspect-item  .
For manual updating the variables contents simple press F3.

Local Variables Browser
The same options as within the global browser are supported, except the online watching option. Variables, which are stored in registers are marked with a red bitmap to indicate this type.
For manual updating the variables contents simple press F3.

Register Browser (I/O)
In this list box all registers of the processor are displayed. For fast access to the requested register the search function is very helpful.
For manual updating the register contents simple press F3.

6.6 Register Window

The Register Window is used to display and modify the values of the target's registers. The contents of the Register Window are updated after every halt in target execution. Registers, that have changed from the last stop, are highlighted blue. The value of a register can be changed by double-clicking on it. While an application is running, the values of the registers cannot be changed.



6.7 Memory Window

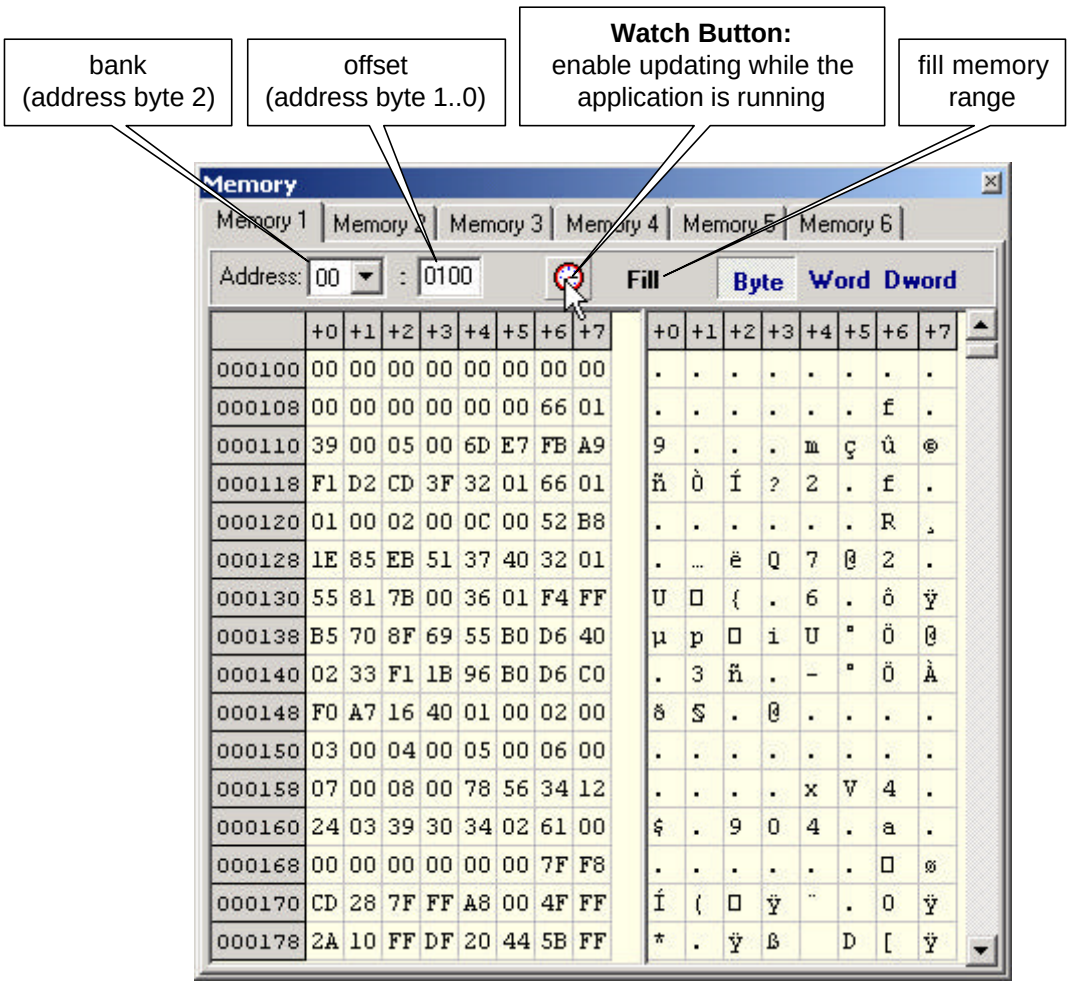
The Memory Window enables monitoring and modifying any memory location in the target, with complete control of size and format of the data.
The Memory Window contains six pages, each for one memory area.

The Memory Window consists of one address column and one or more data columns. On the right side the row contents are displayed in ASCII-Code. One row consists of eight bytes. The representation of the data row as byte, word or double-word is selectable by the same named buttons. The cell in the Memory Window can be changed by clicking on it. The addresses of the displayed target memory can be changed by selecting the 64K bank address and a 16 bit offset.

By enabling the Watch Button the memory contents will be updated during the running application. The update interval can be specified in the General Options Window.
For manual updating the memory contents simple press F3.

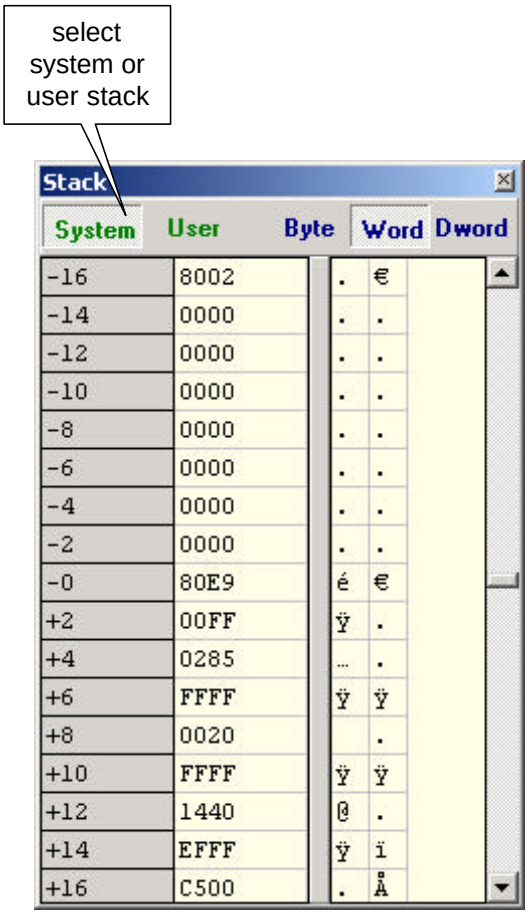
An memory area can be filled by clicking on the Fill Button. The start position, a range in bytes and a byte value is selectable in the Fill Memory Dialog.

All read-only memory areas (flash) are white backgrounded.



6.8 Stack Window

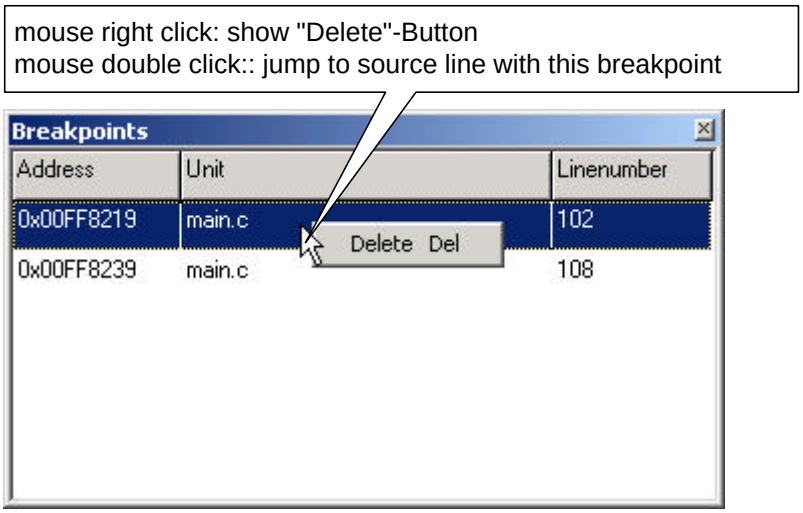
The Stack Window is used to display the application's call stack. The contents of the Stack Window are updated after every halt in target execution.
Two views of the Stack Window can be chosen: the system stack and the user stack. For a detailed description of the stack concept please refer to the FFMC-16 Family Softune Workbench C-Compiler Help.



6.9 Breakpoint Window

The Breakpoint Window gives information about the location of the current breakpoints. The address of the breakpoint will be displayed. If a breakpoint is located inside a Source Window, the unit name and the line number will also be displayed.
By double-clicking on a breakpoint row, the Disassembler Window or the Source Window (if available) will appear where the breakpoint is set.

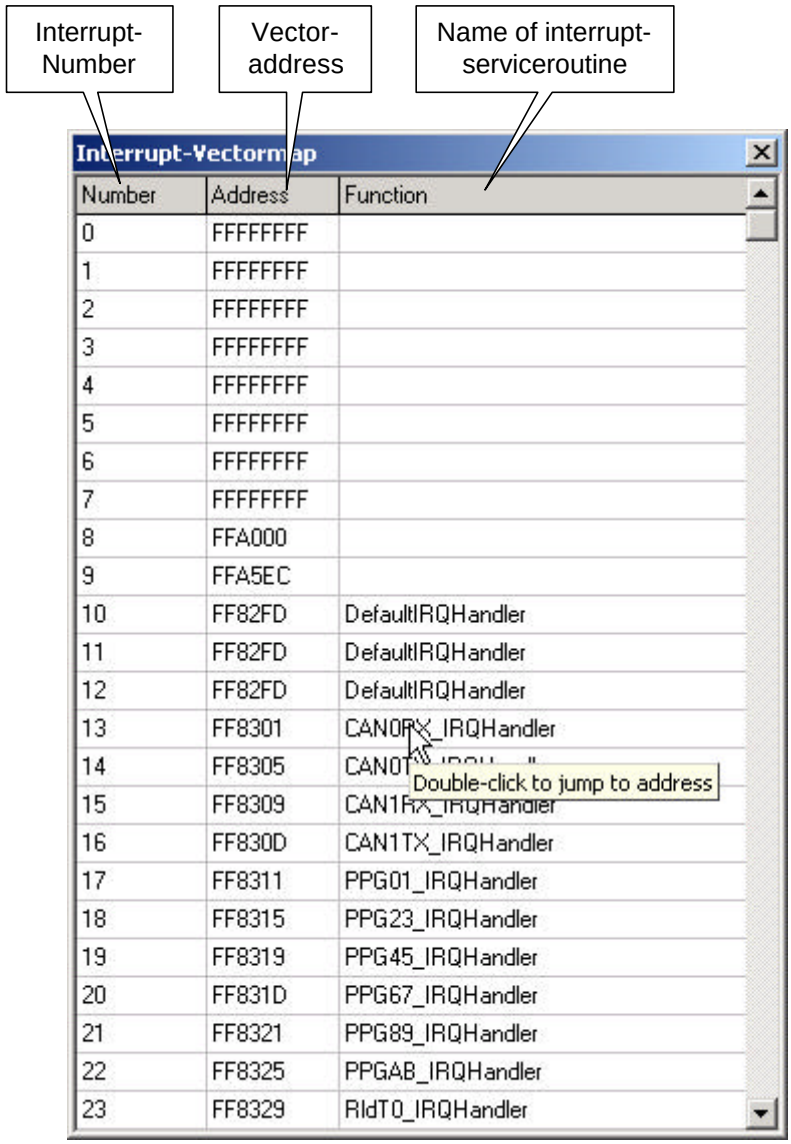
Note:
The maximum amount of Breakpoints is 2, because the device has only 2 ROM-correction units.
When attempting to set a third breakpoint in the Source Window or in the Disassembler Window, the Breakpoint Window will be displayed and it is possible to delete one breakpoint.



6.10 Vectormap Window

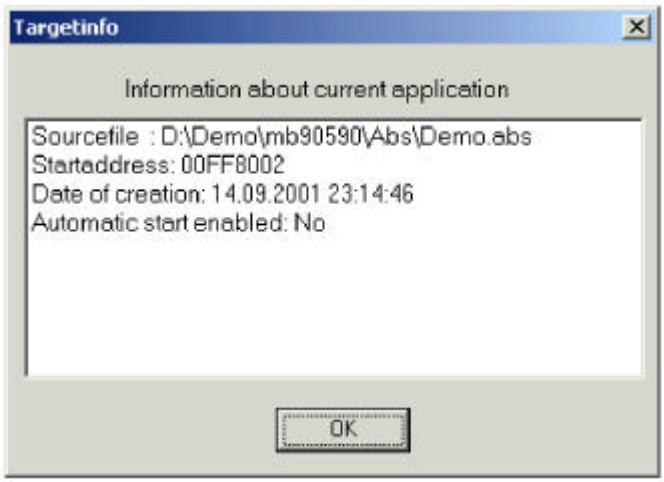
The Vectormap Window shows all interrupt vectors on the associated addresses. If an interrupt service routine is available, this function will be also displayed.

Double-clicking onto a vector will open the interrupt service routine either in the Disassembler Window or in the Source Window.



6.11 Targetinfo Window

This window displays information about the application name, the creation time and the automatic start mode, if an application was loaded into the device.



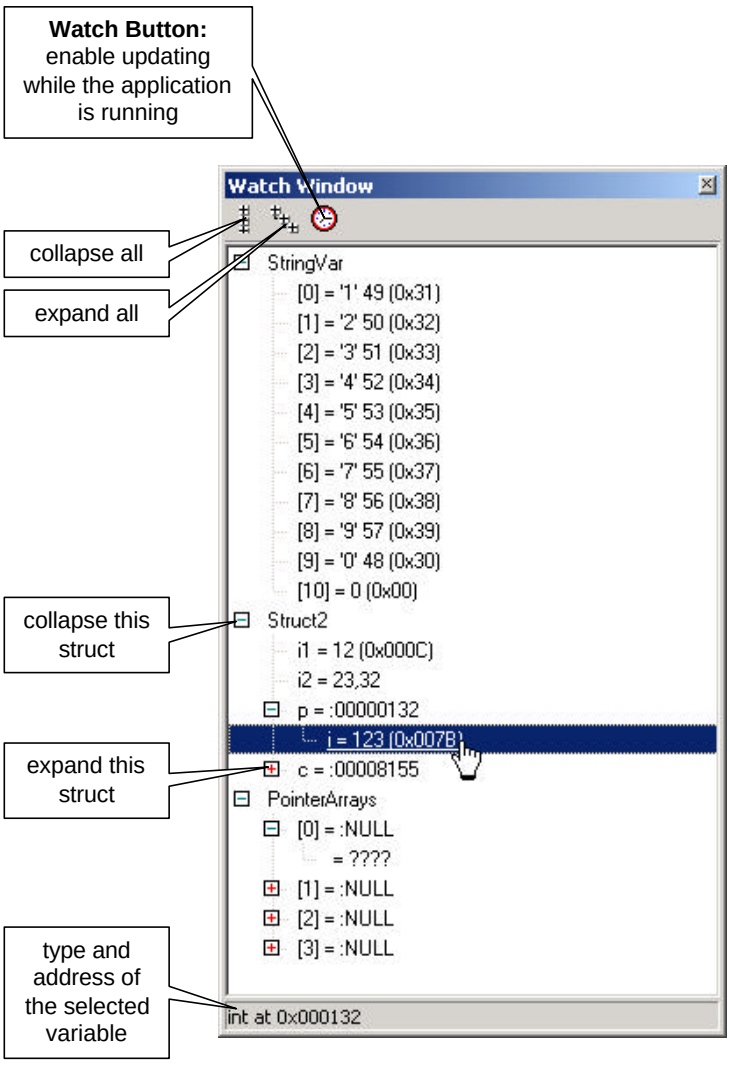
6.12 Watch Window

The Watch Window allows to show selected variables. Like in the Browser Window the variables will be updated every time the program stops. The variables will be updated while the application is running, if the Watch Button is enabled. The update interval is adjustable in the General Options Window.
For manual updating the variables contents simple press F3.

The drag&drop ability can be used for copying one value to another or to copy a variable to the Watch Window.

Note: For copying one value to another, the value of the variable and not the name must be selected.

Pointers, structures and arrays have a compact and expanded form. The expanded form shows all the fields of a pointer, struct or array. The structures are expandable within structures (nesting).
With the **collapse/expand buttons** all variables can be collapsed or expanded. By right clicking onto a variable the selected node only can be collapsed or expanded.
Simple input the first characters of the requested function, variables or registers. All matching items in the actual browser window will be displayed.
Evaluation of a pointer can be done by clicking on the inspect-item .



6.13 Target Message Window

Some powerful functions for transferring messages and data form the application to the PC screen are integrated in Accemic MDE.
So it is very easy to open a MessageBox on the PC screen and control the application program flow by the return value of this MessageBox. Even the continuous displaying and logging of data and messages is possible.
Please refer to the Accemic MDE sample files, here are some demonstrations of target messaging.

For an efficient management of memory resources the application program must reserve a buffer for transferring the messages. The buffer size depends on the largest message which should be send to the PC. The corresponding function is `acc_InitMonitorMsgBuffer.`
After this initialization two functions can be called.

- Displaying a simple Message with return value on the PC-screen (`acc_MessageBox` - Function)
- Displaying variables or text on the Target Message Window or in a file (`acc_printf` - Function). If the “Save to File” Button is selected, the data stream will be stored in a text file. With the “Close File”-Button the storage in a file can be stopped.

6.13.1 acc_InitMonitorMsgBuffer: Initialization of message buffer

Description
Before using the message functions (`acc_printf` and `acc_MessageBox`) the function `acc_InitMonitorMsgBuffer` must be called to initialize the Message Buffer.

Prototype

```
void __far acc_InitMonitorMsgBuffer ( __far char* MsgBuf,
                                      char BufferSize)
```

Parameter	Meaning
`MsgBuffer`	Pointer to buffer
`BufferSize`	Size of buffer in bytes. The requested size of the message buffer can be calculated with the following formula: BufferSize = 4 + maximal count of bytes to transfer, minimal 12

Return value
none

defined in
“monitor.h”

6.13.2 acc_MessageBox: Show a MessageBox on the PC screen

Description
With this function a MessageBox will be called on the PC screen. This function is similar to the "MessageBox"-function of the Win32-API.

Prototype
```
char __far acc_MessageBox (const __far char* Msg,
                           char Flags)
```

Parameter	Meaning
Msg	Pointer to the Message String, which should be transferred
Flags	Indicates what kind of button to show 0 Message box contains one push button: OK 1 Message box contains two push buttons: OK and Cancel 2 Message box contains three push buttons: Abort, Retry and Ignore 3 Message box contains three push buttons: Yes, No and Cancel 4 Message box contains two push buttons: Yes and No 5 Message box contains two push buttons: Retry and Cancel

Return value
-1 Error
0 PC not connected or transmission not possible
1 Button OK on the PC was pressed
2 Button Cancel on the PC was pressed
3 Button Abort on the PC was pressed
4 Button Retry on the PC was pressed
5 Button Ignore on the PC was pressed
6 Button Yes on the PC was pressed
7 Button No on the PC was pressed

defined in
"monitor.h"

6.13.3 acc_printf: Show a Text in the Target Message Box

Description
This function is used to show data and text in the Tqarget Message Box on the PC screen.

Prototype
```
char __far acc_printf ( char Location,
                        char Type,
                        const __far void* Data,
                        char Position,
                        char Wait)
```

Parameter	Meaning	
`Location`	ACC_TYPE_LISTBOX	Listbox
	ACC_TYPE_ROW_CAPTION	Row Caption
	ACC_TYPE_ROW_TEXT	Row Text
`Type`	ACC_TYPE_STRING	String
	ACC_TYPE_BOOL	bool, 1 byte
	ACC_TYPE_UCHAR	unsigned char, 1 byte
	ACC_TYPE_CHAR	signed char, 1 byte
	ACC_TYPE_USHORT	unsigned short, 2 bytes
	ACC_TYPE_SHORT	signed short, 2 bytes
	ACC_TYPE_ULONG	unsigned long, 4 bytes
	ACC_TYPE_LONG	signed long, 4 bytes
	ACC_TYPE_FLOAT	float, 4 bytes
	ACC_TYPE_DOUBLE	double, 8 bytes
`Data`	Pointer to the data, which should be transferred	
`Position`	Position of row (if type equal ACC_TYPE_ROW_CAPTION or ACC_TYPE_ROW_TEXT)	
`Wait`	ACC_NOT_WAIT	not wait for transmission
	ACC_WAIT	wait for transmission

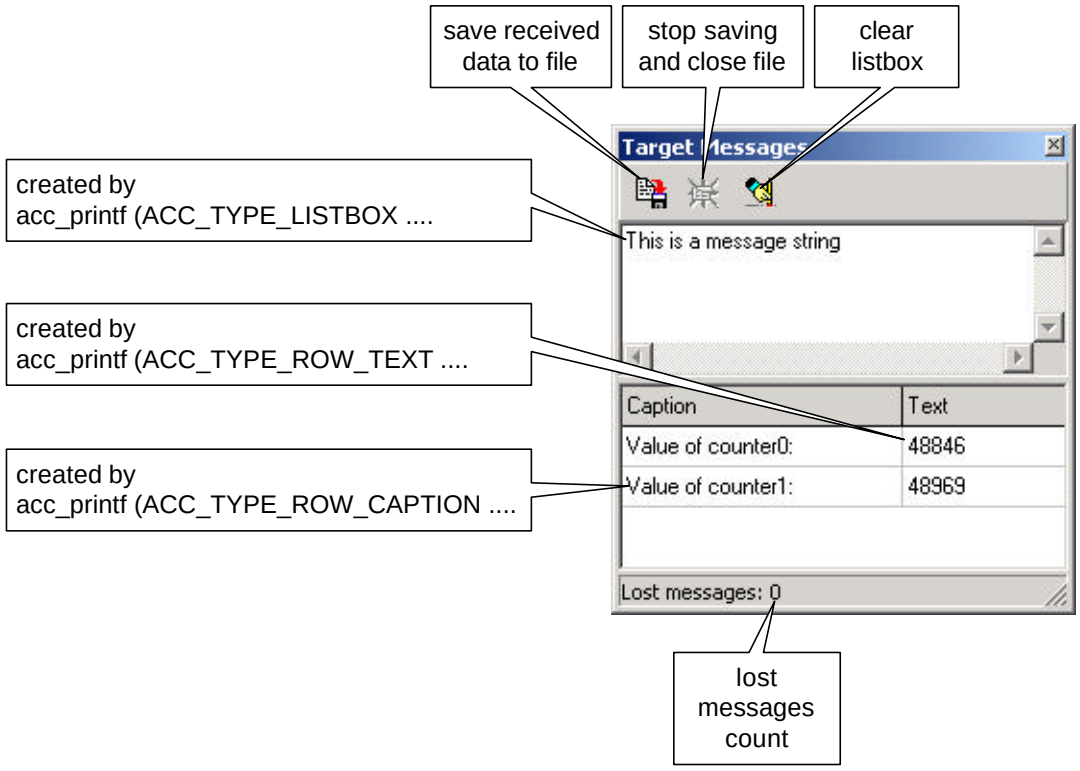
Return value
-1 Error
0 PC not connected or transmission not possible
1 transmitted

defined in
“monitor.h”

6.13.4 Target messaging example

The following code creates this target messaging window:

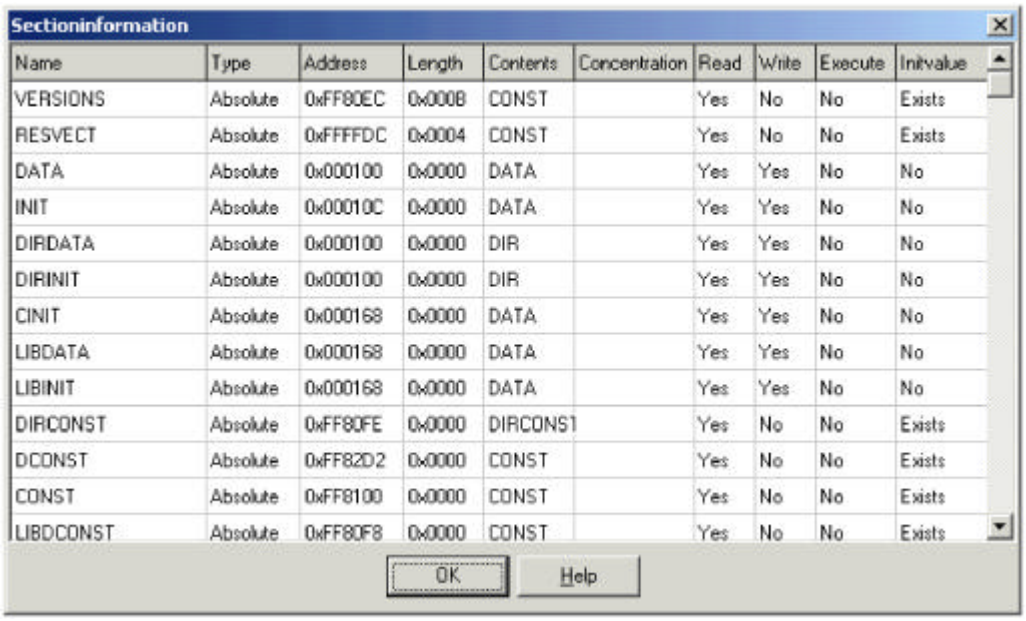
```
__far char Buffer[104];
unsigned short counter0 = 48846;
unsigned short counter1 = 48969;

// give Accemic MDE the pointer to the message buffer
acc_InitMonitorMsgBuffer(Buffer, 104);

// write a text in the ListBox
acc_printf (ACC_TYPE_LISTBOX, ACC_TYPE_STRING,
            (const __far char*)"This is a message string",
            0, ACC_WAIT);

// write row captions
acc_printf (ACC_TYPE_ROW_CAPTION, ACC_TYPE_STRING,
            (const __far char*)"Value of counter0:",
            0, ACC_WAIT);

acc_printf (ACC_TYPE_ROW_CAPTION, ACC_TYPE_STRING,
            (const __far char*)"Value of counter1:",
            1, ACC_WAIT);

// output variables
acc_printf (ACC_TYPE_ROW_TEXT, ACC_TYPE_USHORT, &counter0, 0, ACC_WAIT);
acc_printf (ACC_TYPE_ROW_TEXT, ACC_TYPE_USHORT, &counter1, 1, ACC_WAIT);
```



6.14 Sectionmap Window

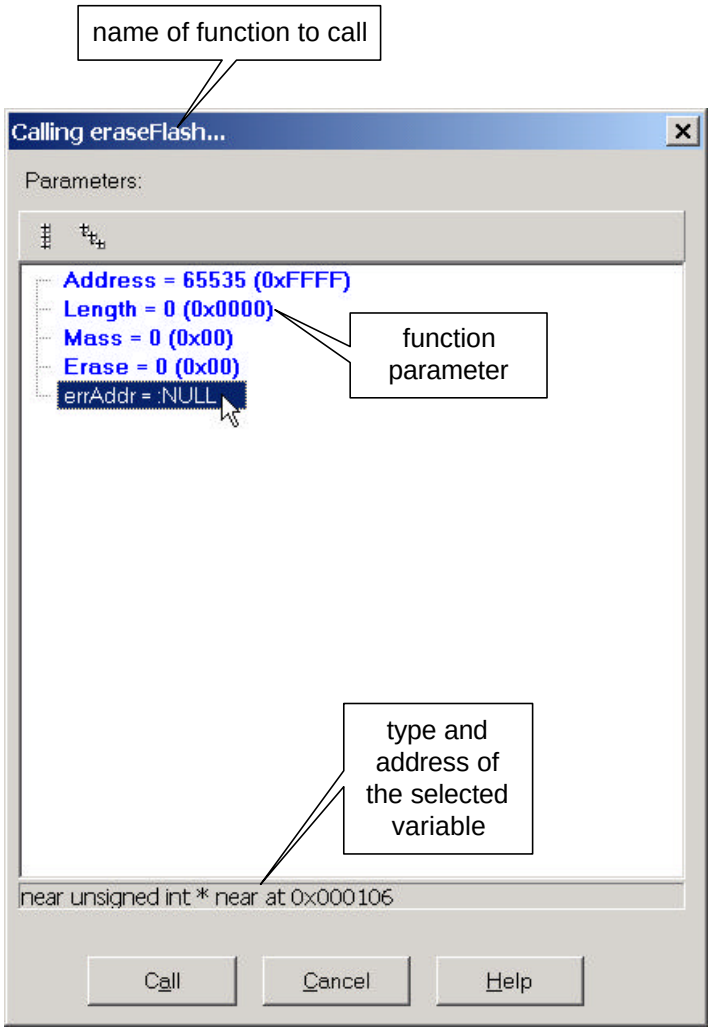
The Sectionmap Window shows detailed information about section allocation.

Please refer to the “Softune Linkage Kit documentation”, chapter “Section Allocation Detailed Information List File” for further information.



6.15 Call Function Window

This feature allows to call functions independently from a running application. First enter the name of the function to call in the edit-field. After pressing “return” the parameters of the function are displayed in the function parameter window. After initializing the function parameters, the function will be called by pressing the “Call”-Button.
The type and the address of the variable will appear in the bottom status line.



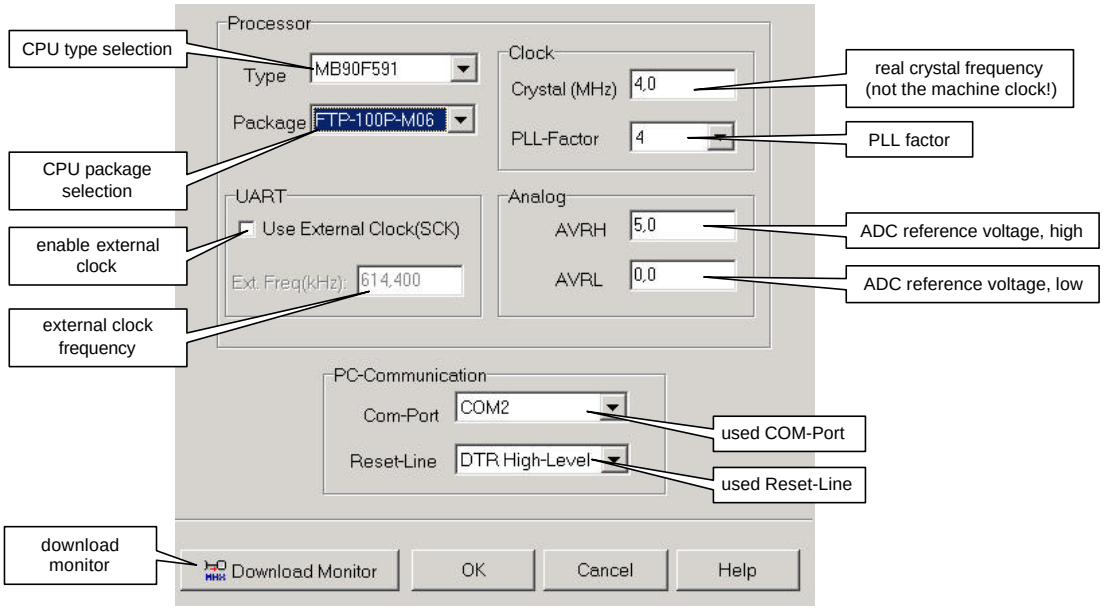
6.16 System Preferences Window

The System Preference Window allows to setup system configuration.
The target MCU type, the clock rate and the PLL factor are selectable.

UART
A external clock source for the Bootloader-UART can be selected by checking the “Use external clock”-Button. The external clock frequency must be given in kHz.
Accemic MDE checks, if this clock matches with a supported baudrate. If not, an error message will be displayed.

PC communication
Please select the used COM-Port. The selected COM-Port cannot be used by another application at the same time.
Accemic MDE supports automatic reset generation. If the target MCU reset line is connected (via RS232-Interface like MAX232) to an RS232-signal (RTS or DTR), the suitable Reset Line (RTS-Low, RTS-High, DTR-Low, DTR-High) can be selected. Accemic MDE will reset the system with this signal (e.g. after downloading the monitor) automatically.

Download Monitor
For downloading the monitor kernel press the “Download Monitor”-Button. A message window with the correct settings of the Mode pins (MD0: 0, MD1: 1, MD2: 1, P00: 0, P01: 0) will be displayed. After correct setting of the hardware press reset (if the automatic reset generation of Accemic MDE is not used). Accemic MDE will download the monitor-kernel into your target MCU. After downloading the mode must be changed to “single chip mode” (MD0: 1, MD1: 1, MD2: 0, P00 and P01: don’t care) and the system must be resetted. .
Please keep in mind, that a Accemic MDE license contains only 100 loaded monitor-kernels in a time. If you need more than 100 monitor kernels in a time, please contact Accemic.



6.17 General Options Window

The General Options Window is divided in two parts: general settings and selection of fonts and colors.

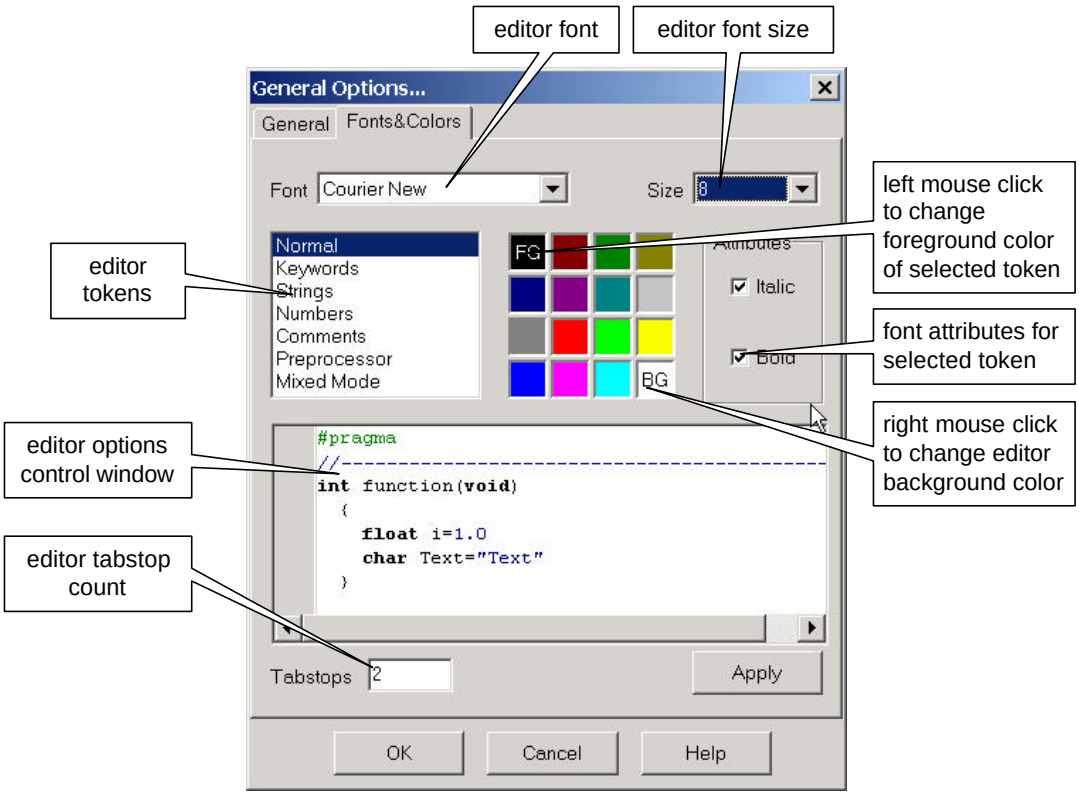
General…

The Online Interval specifies the update interval for online-watching (Global Variable Browser Window, Watch Window, Memory Window). Accemic MDE automatically computes the system workload. If the workload is too high (more than 75%), the interval will be updated automatically to a new value that fits to 75%.
If the "Jump-To-Main" check box is checked, Accemic MDE will automatically search for the "main" function in the application and will stop there after restarting or loading.
If the "Jump-To-Main" check box is not checked, the program will jump to the code start address after restarting or loading.

Fonts&Colors

In this section the font, font-size,font-color, text-attributes and the tab width for the Source Windows and the Disassembler Window are configurable by the user.

Left-click on the color sheet sets the foreground color for the specified highlighting type. Right-clicking on the color sheet sets the background color.

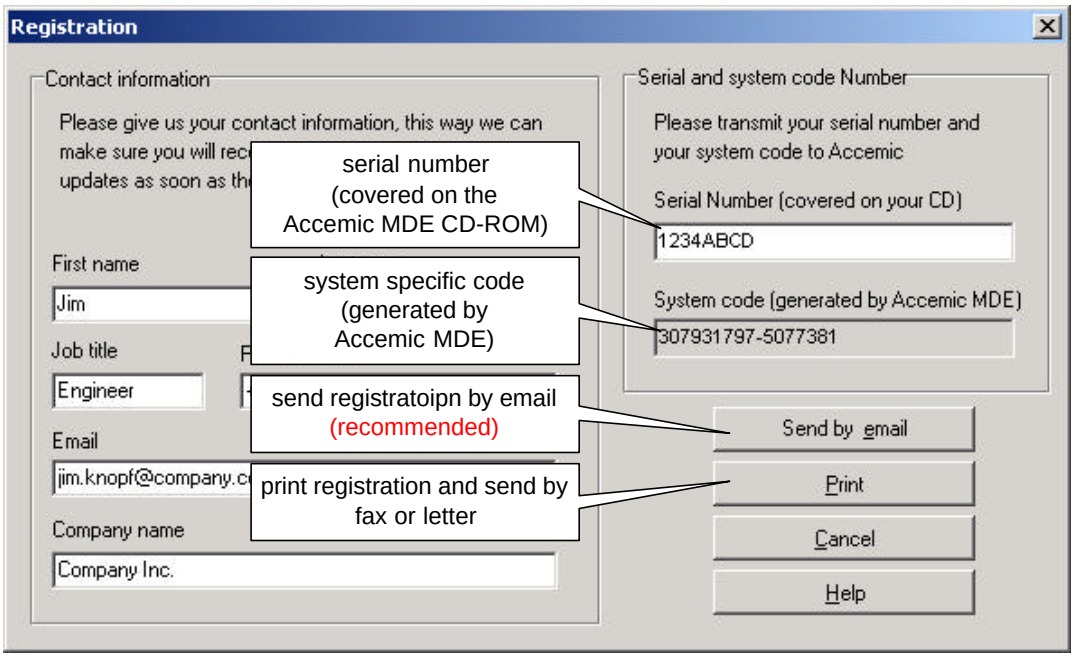


6.18 Registration Window/Unlock Window

Accemic MDE will work after first installation for 10 days without limitations. In this time the program license must be registered by Accemic. 10 days after the first installation without registration, the program will enter a demo mode with many limitations. Only the sample files are executed.

For receiving the registration code, the system code and the unique serial number must be transmitted to Accemic, either by pressing the E-Mail Button in the Registration Window or by sending a mail to register@accemic.com.

After receiving the authorization code from Accemic (it will last about one day), the code must be entered in the Unlock Window to get unlimited access to Accemic MDE.



Each registration request is protected with an unique system code.
Don't create a new registration request or make a license transfer while waiting for an unlock code from Accemic. In this case the requested unlock code from Accemic will be invalid. By changing hardware configuration or deleting / manipulating the license file(s), Accemic MDE will enter the demo mode. For getting the full access again a new registration is required.

Please keep in mind, that the Accemic MDE demo license allows to store, load, install or execute the Accemic MDE monitor kernel in a maximum of one target microcontroller at a time. One Accemic MDE full license allows to store, load, install or execute the Accemic MDE monitor kernel in a maximum of 100 target microcontroller at a time. Accemic MDE is intended and must only be used in an EVALUATION LABORATORY ENVIRONMENT. Under no circumstances Accemic MDE may be used in a product. If you need more than 100 monitor kernels or if you want to use Accemic MDE in a product, please contact Accemic.

Tip:
Before changing the hardware configuration of the licensed computer, the license can be transferred to an other computer. After changing the hardware the license can be transferred back.

7 Technical data

7.1 Supported operating systems

Accemic MDE is designed for Microsoft Windows 95, 98, NT4 and 2000.
The standard font size is required.

7.2 Supported development tools

This version of Accemic MDE has been developed for the Fujitsu compiler fcc907s, which is part of the FFMC-16 Family Softune Workbench.
Used versions of tools:

- Fcc907s V30L02 or later
- Fasm907s V30L04
- Flib907s V30L03

It works with a Fujitsu compiler only, since other C-Compiler's may use different calling conventions (incompatible object file formats) and are therefore not compatible.

7.3 Supported Target MCUs:

- MB90F443
- MB90F462
- MB90F497
- MB90F543, MB90F546, MB90F548, MB90F549
- MB90F553
- MB90F562, MB90F568
- MB90F574
- MB90F583
- MB90F591, MB90F594A
- MB90F598

7.4 Input File Formats

- ABS-Files: This file includes the code and all necessary debug information. The FFMC-16 Family Softune Workbench environment will automatically generate it.
- MHX-Files: This file type includes only code. No debug information is available.
- Unspecified file type (*.*): This file type includes only raw data. The data will be written to the specified destination address.

7.5 System resources

- Internal Flash: Sector including 0xFFA000 (size depends on target) + 256 Byte
- Internal RAM: 128 Byte
- System stack 12Byte
- Bootloader-UART (target depend UART0 or UART1) with interrupts
- Both ROM-Correction units

7.6 System Load

These absolute timings are based on a 16 MHz F16LX system.

- Target MCU receive interrupt: ca. 35µs
- Target MCU transmit interrupt: ca. 25µs

While online-watching large structures with high refresh rate, the maximal system load is smaller then 14%.

8 Index

G

H

I

L

M

O

P

R

S

T

U

V

W