



## FLASHly means ...

### SIMPLEST FLASH-PROGRAMMER TOOL

FLASHly will erase, program and verify Microcontrollers as simple as possible:

- ✓ Freeware for standard version
- ✓ Execution without installation
- ✓ Command line controlled
- ✓ Exit code and Result Message
- ✓ DTR/RTS control for reset-/ mode-handling
- ✓ COM-port release
- ✓ USB/RS232-converter support
- ✓ Can be integrated easily into Softune
- ✓ Can be called by Windows 'SentTo' menu
- ✓ Log file

### FREEWARE

FLASHly is freeware for standard version.

For registered users:

- ✓ Disclaimer window will be skipped
- ✓ Email notification for new releases

### EXECUTION WITHOUT INSTALLATION

No installation is required. Save FLASHly to any location on your harddisk.

### DISCLAIMER

This tool is allowed ONLY for evaluation purposes.

This tool is NOT allowed for mass production!  
USE THIS TOOL AT YOUR OWN RISK!!!

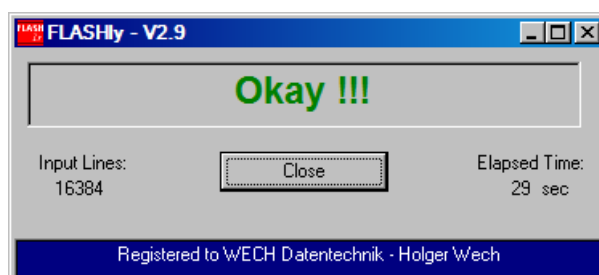
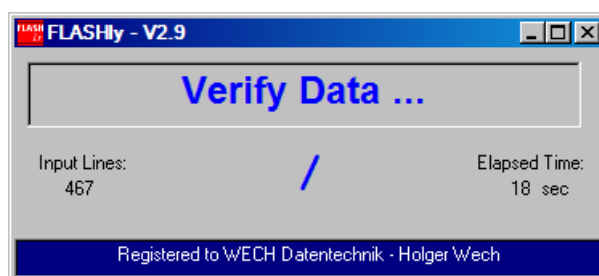
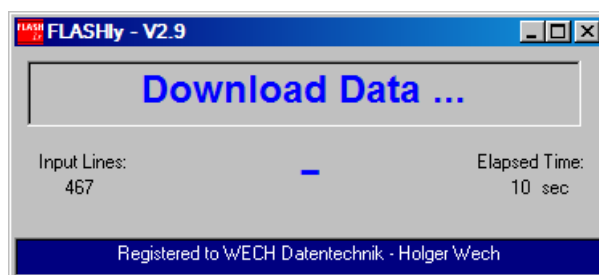
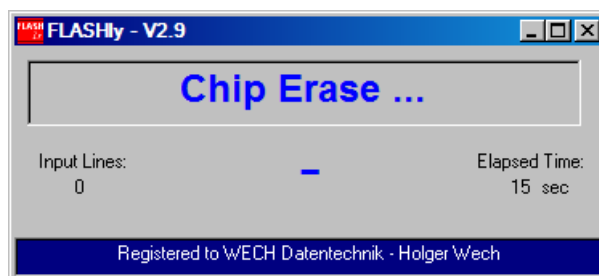
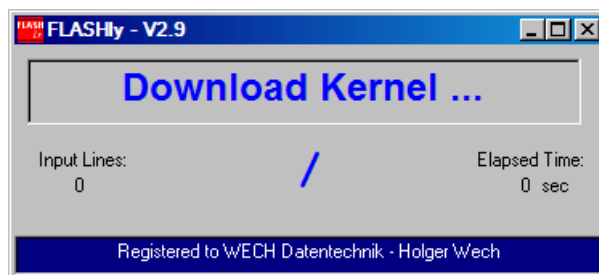
No Responsibility or Liability will be accepted for any Error or Damage.

### CONTACT

For questions, bug report or any other reason please contact [FLASHly@gmx.de](mailto:FLASHly@gmx.de)

### DOWNLOAD

Please download the latest version from [www.holgerium.de/elektronik](http://www.holgerium.de/elektronik)



## COMMAND LINE CONTROLLED

All functions are invoked by command line options only:

Some commands are available only for dedicated microcontrollers!

- c:n** Selects the COM-port (n=1,2,3... all available COM-ports)
- d:ms** Insert delay after reset before connection (e.g.: -d:300)
- m:line** Mode-selection while programming (line=RTS+, RTS-, DTR+, DTR-)
- r:line** Reset-control while reset (line=RTS+, RTS-, DTR+, DTR-)
- debug** Allows to check the correct levels when using options -m:line or -r:line
- h** Open help-menu
- M:micro** Selects microcontroller type/series, e.g.: -M:96340
- M:HEXDWL** Selects Fujitsu's user-bootloader  
 Select COM-port by parameter -c:n, e.g.: -c:1  
 Define user baudrate by parameter -b:baudrate, e.g.: -b:38400  
 Define download file by parameter -P:file  
 Sorry, no other FLASHly commands are supported for user-bootloader mode 'hexdwl'
- cpu partnumber** Selects microcontroller type/series, e.g.: -cpu MB96F348HSB  
 alternatively to -M for compatibility reasons with SWB parameters
- M:filename** Extracts MCU-type from filename defined by -P:file
- Q:crystal** Selects external crystal frequency (default is 4MHz)
- E** Erase microcontroller before programming (Chip Erase)  
 Caution: This command will erase all flash contents, incl. security bit, trimming data (see -TR), etc.
- E:address** Erase defined sector before programming (Sector Erase)  
 Multiple erasing possible, e.g.: -E:FD0000 -E:FE0000 -E:FF0000  
 Only available for FM, 16FX and MB90340, MB90350 Microcontroller  
 In order to simplify the handling, no sector validation related to the MCU is done.
- P:file** Programs MHX- or Binary- File to FLASH-microcontroller  
 In order to simplify the handling, neither address validation of the downloaded software  
 neither a check of the correct address space for the microcontroller is done.  
 A binary file is indicated by file extension \*.bin and requires additional parameter -bin:address
- skipFF** Blocksequences of 0xFF within the programming file will be skipped,  
 assuming that the chip was erased successfully before programming.  
 Note: This option might increase programming speed, but flash area must be erased before
- V** Verifies FLASH with the programmed file given by -P option
- V:file** Verifies FLASH with the given file (both -V options valid only for 16FX MCU)
- X:adr1,adr2,chk** Calculates checksum from address *adr1* to *adr2* and compares with *chk*  
 Several areas possible: e.g.: -X:DF0000,DF7FFF,F80000,FFFFFF,<chk>  
 Note: In case no chk is given, an error occurs with showing the calculated checksum only
- bin:address** Defines the start address in case a binary file (\*.bin) is given  
 by parameter -P:file.bin and/or -V:file.bin
- nolog** No logfile generation
- newlog** Overwrites old logfile
- msgok** Close application automatically after successful finish
- msgerr** Close application automatically after erroneous finish
- errorlevel** Returns the specific error code (0..99) instead of simple 0:okay, 1:failed

Newer microcontrollers (e.g. FM3) have a built-in and trimmed RC-oscillator. Be careful when erasing the Flash or changing the trimming-data:

- TR -TW** Keeps existing trimming-data. (Command available only for FM3)  
This command should be combined with the erase command (-E) in order to avoid destroying the pre-programmed trimming-data.
- TR** Reads the actual trimming-data and writes the value to the logfile  
(Command available only for FM3)
- TW: *value*** Writes a new value to the trimming-data area. (Command available only for FM3)  
Caution: Please avoid to use this command, all FMcontrollers are already trimmed by the fab.
- TC** Calculates the trimming data needed for the built in high-speed RC-oscillator. The result is written to the logfile.  
(Command available only for MB9BF510N/R/S/T)
- TC -TW** Calculates the trimming data needed for the built in high-speed RC-oscillator and writes the value to the trimming-data area.  
(Command available only for MB9BF510N/R/S/T)
- TR -TC -TW** Reads existing trimming-data and writes it to the logfile, calculates new value and writes it to the trimming-data area.  
(Command available only for MB9BF510N/R/S/T)

## Examples:

```
FLASHly.exe -M:MB90360 -Q:4 -E -P:demo.mhx -c:1 -r:DTR+ -m:RTS-
```

```
FLASHly.exe -c:1 -M:filename -E -P:90360_demo.mhx
```

If **FLASHly** is used within the Fujitsu Softune Workbench, then the parameters %D%A can be used to handle the project filename:

```
FLASHly.exe -c:1 -M:filename -E -P:%D%A.MHX
```

## EXIT CODE

When **FLASHly** is started within a batch file then an exit code (errorlevel) is returned:

(Note: This feature may not work with all Windows versions)

**FLASHly** will return exit code 0 after successful finish.

**FLASHly** will return exit code 1 after erroneous finish.

For a more specific exit code see also `-errorlevel` parameter description.

### Example: start.bat

```
echo off
call FLASHly.exe -M:MB90360 -Q:4 -E -P:demo.mhx -c:1 -msgerr -msgok
if errorlevel 1 goto error
goto ende
:error
echo FLASHly terminated with error
:ende
echo finished
```

## RESULT MESSAGE BOX

By default a message box displays the result of the performed action (Okay / Error).

The message box 'Okay' can be skipped by using the parameter '`-msgok`'.

The message box 'Error' can be skipped by using the parameter '`-msgerr`'.

## DTR/RTS CONTROL FOR RESET-/ MODE-HANDLING

COM-port lines DTR and RTS can be used for automatic reset and mode-selection.

## COM-PORT RELEASE

When **FLASHly** finishes the COM-port is closed automatically.

## USB/RS232-CONVERTER COMPATIBLE

Most USB/RS232-converter will work with **FLASHly**.

With some USB/RS232-converters faster programming times can be reached by use of Alias Baud Rates (See appendix).

## LOG FILE

Each execution of **FLASHly** is logged in textfile '`FLASHly.log`'.

Start- and endtime, parameters and result will be saved.

The log file generation is disabled by the parameter '`-nolog`'.

The parameter '`-newlog`' overwrites the old log file.

**ERRORLEVEL:**

If the parameter `-errorlevel` is given then the specific exit code will be returned:

(Note: This feature may not work with all Windows versions)

```
FLASHly.exe -M:MB90340 -Q:4 -E -P:demo.mhx -c:1 -errorlevel
```

0:No Error	20:Program Timeout	40:Checksum Error
1:Help	21:Program Checksum Error	41:Trim Read Error
2:Disclaimer	22:Read Error 1	42:Trim Read Timeout
3:COM port not exists	23:Read Timeout 1	43:Trim Command not supported
4:COM open denied	24:Read Error 2	44:Trim Write Error
5:Wrong Parameter	25:Read Timeout 2	45:Trim Write Timeout
6:Wrong MCU	26:File Error	46:Trim Write Checksum
7:Wrong MCU parameter	27:Invalid Data Record	47:Invalid Read Adress
8:Kernel too big	28:Break by user	48:Invalid Check Adress
9:MCU connection failed	29:Main Unlock Error	49:RX Overflow
10:Kernel download failed	30:Sat Unlock Error	50:FLASHly Closed by User
11:Kernel not started	31:Unknown Error	51:Log File Error
12:Kernel PLL not active	32:Verify-File not found	52:Download1 Error
13:Erase Error 1	33:Verify not supported	53:Download2 Error
14:Erase Timeout 1	34:Verify failed	54:Missing -bin Parameter
15:Erase Error 2	35:Verify Error 1	55:Reset Request
16:Erase Timeout 2	36:Verify Timeout 1	56:Hardware Debug Break by User
17:Prog-File not found	37:Verify Error 2	
18:MHX file out of range	38:Verify Checksum Error	
19:Program Error	39:Checksum Cmd Error	

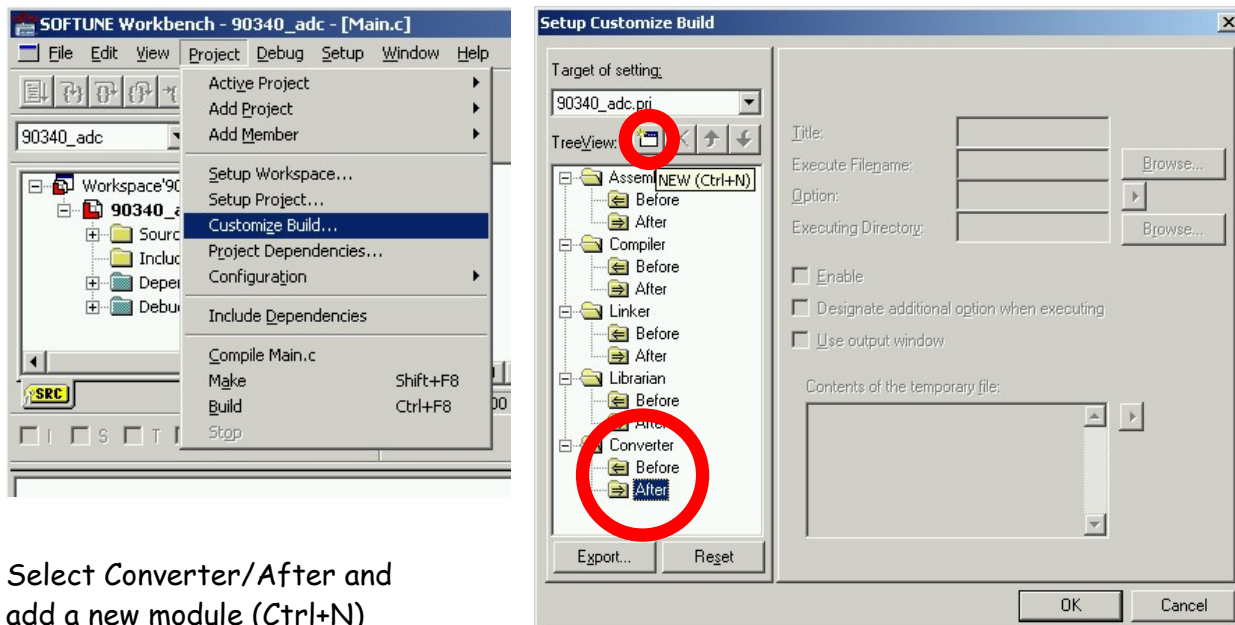
E.g. in case a reset must be asserted while programming flash secured devices, the error code 55 will indicate, that the microcontroller must be reset before starting again **FLASHly**.

```
echo off
call FLASHly.exe -M:MB90340 -Q:4 -E -P:demo.mhx -c:1 -errorlevel
echo %errorlevel%
if errorlevel 56 goto error
if errorlevel 55 goto RESET
if errorlevel 1 goto error
goto ende
:RESET
echo Please reset system and start FLASHly again without Erase!
pause
call FLASHly.exe -M:MB90340 -Q:4 -E -P:demo.mhx -c:1
echo %errorlevel%
pause
goto ende
:ERROR
echo FLASHly terminated with error
goto ende
:ende
echo finished
pause
```

## INTEGRATION TO SOFTUNE WORKBENCH

FLASHly can be added to the FUJITSU development software SOFTUNE Workbench. After successful compilation and building of your software project the microcontroller is programmed automatically.

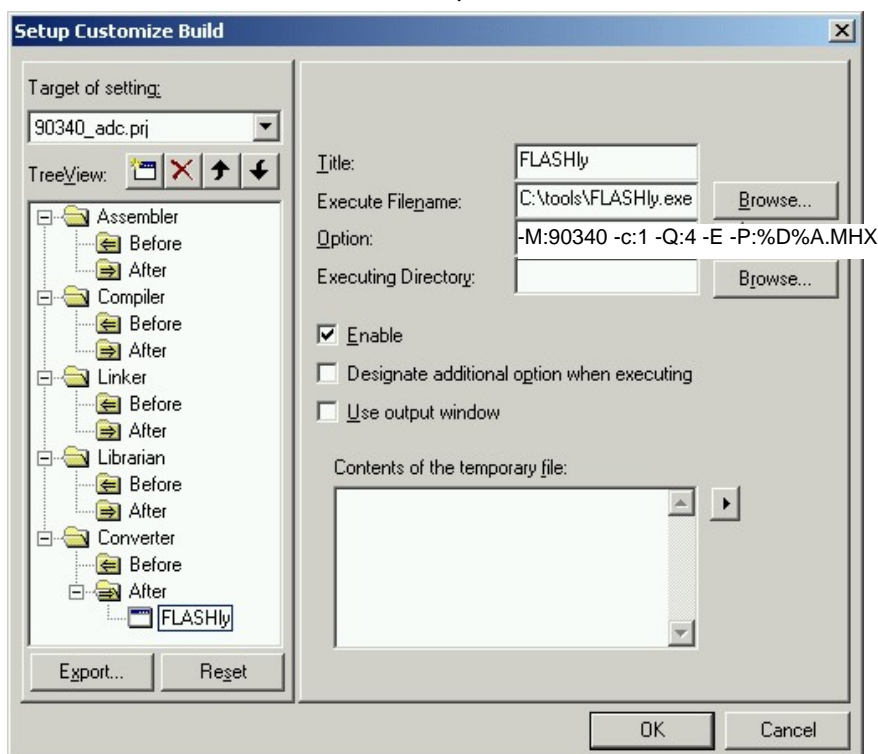
Open the 'Customize Build...'-Setup (Project / Customize Build...)



Select Converter/After and add a new module (Ctrl+N)

Add title 'FLASHly' and browse to 'FLASHly.exe' for executable filename.

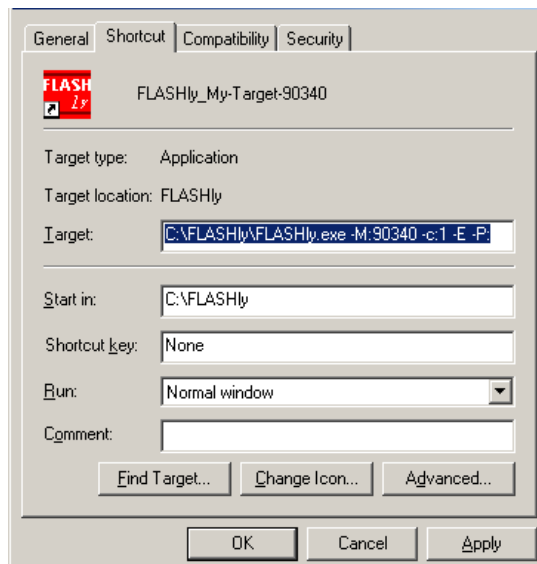
Define command-line parameters as options. The SOFTUNE workbench macro %D%A.MHX is used to address the MHX-file for command-line parameter '-P:%D%A.MHX'



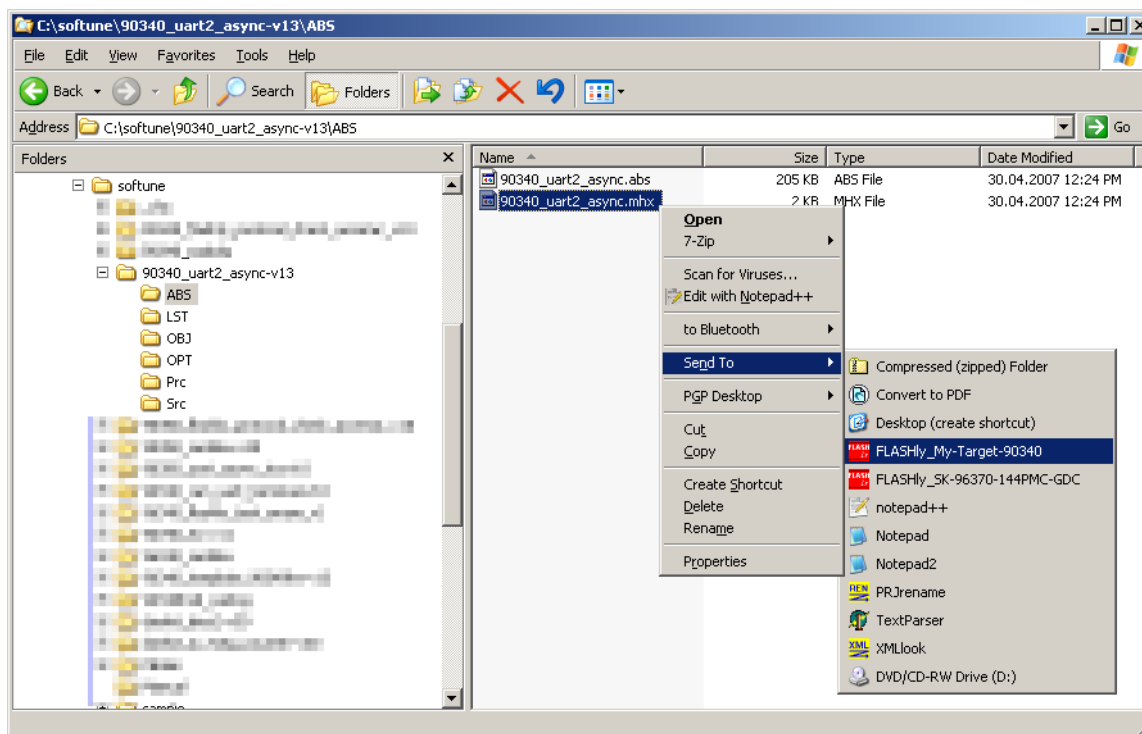
## INTEGRATION TO WINDOWS 'SendTo' MENU

FLASHly can be called by the „Send To“ menu from the file explorer:

1. Copy a shortcut of **FLASHly** to the Windows "SendTo" folder
2. Rename the shortcut to an adequate name referring to your application, e.g.: **FLASHly\_My-Target-90340**
3. Open the properties of the shortcut
4. Add all target specific parameter, like **COM-Port**, **MCU type**, **Erase** e.g.: **-c:1 -M:90340 -E**
5. Finally place parameter **'-P:'**
6. Apply these settings and press **OK**



- Within the file explorer make right mouse click to any **MHX-file**
- Select **'Sent To'** and choose your defined shortcut



The name of the **MHX-file** will be added to the predefined parameters and **FLASHly** will start immediately programming the target application.

## USING ALIAS BAUD RATES

Standard COM-ports allow Baud rates up to 115kBaud. Some USB/RS232-CONVERTERS allow 'Alias Baud rates', which means lower Baud rates, can be replaced by higher ones. The USB/RS232 windows driver has to be modified accordingly. Details can be found here:

<http://www.ftdichip.com/Documents/AppNotes.htm>

AN232B-05 : Configuring FT232R, FT2232 and FT232BM Baud Rates

FLASHly V2.0 or newer supports alias Baud rates by two parameters:

- baud:rate** 'Official Baud rate' that will be replaced by alias Baud rate
- bgr:divisor** 'Baud rate Generator Reload' value for Alias Baud rate  
(Note: The divisor must fit to the requirements (hardware manual, datasheet) of the microcontroller.)

Example: USB/RS232-CONVERTER patches 300Baud to 500kBaud

(Note: The USB/RS232 windows driver has to be modified accordingly)

MCU is running with 4MHz

'Baud rate Generator Reload' = MCU-Frequency / Alias Baud rate - 1

'Baud rate Generator Reload' = 4.000.000 / 500.000 - 1 = 7

```
FLASHly.exe -c:1 -M:MB96340 -Q:4MHz -E -P:test.mhx -baud:300 -bgr:7
```

FLASHly V2.0 or newer supports alias Baud rates for dedicated microcontroller only.

Please see the `_ReadMe.txt` of your FLASHly version.