

FLASH166

User's Manual



FORTH-SYSTEME GmbH

P.O. Box 11 03

Kueferstrasse 8

☎ +49 (7667) 908-0 • Fax +49 (7667) 908-200 • e-mail: sales@fsforth.de

• D-79200 Breisach, Germany

• D-79206 Breisach, Germany

© Copyright 1993 - 2000:

FS FORTH-SYSTEME GmbH
Postfach 1103, D-79200 Breisach a. Rh., Deutschland

Release of Document: March 24, 2000
Author: Dieter Fögele
Filename: FLSH166b.doc
Program Version: 3.01

All rights reserved. No part of this document may be copied or reproduced in any form or by any means without the prior written consent of FS FORTH-SYSTEME GmbH.

Contents

1. General.....	5
1.1. System Requirements	6
1.2. Connecting your PC to the target system	7
1.3. First Example.....	8
1.4. Trouble Shooting	10
1.5. Error Messages	11
1.6. Initialization file FLASH166.INI	15
1.7. Supported flash devices	17
2. FLASH166 Parameter Description.....	22
2.1. Program a Flash Device	25
2.2. Verify a Flash Device with file.....	28
2.3. Read a Flash Device to file.....	30
2.4. Download to Target Memory.....	32
2.5. Upload from Target to Host	34
2.6. Dump Target Memory	35
2.7. Toggle CPU pins.....	37
2.8. List of supported Flash Devices.....	38

1. General

This very compact program uses the bootstrap mode of the Infineon 80C166 and the STMicroelectronics ST10 microcontroller family

- to program data into external flash memory
- to verify and read the contents of flash memory
- to load data to static memory
- to upload data from target to host
- to make a memory dump
- to test CPU signals

All functions are realized with a very small monitor program running in the internal memory of the microcontroller. Initially developed to support the FS FORTH-SYSTEME modules with the Infineon 80C166 and the C167CR, you can use this tool for your own design with one of the Infineon or STMicroelectronics 16 Bit microcontrollers.

Currently the following CPUs are supported:

- C161RI/PI
- C165
- 80C166
- C167
- C167CR
- C167SR
- ST10F163
- ST10F168
- ST10F169

Many different Flash types are supported (see chapter 1.7 "Supported flash devices"). For Flash programming, no external SRAM is required in the target.

For latest documentation please refer to the file README.TXT on the distribution disk.

1.1. System Requirements

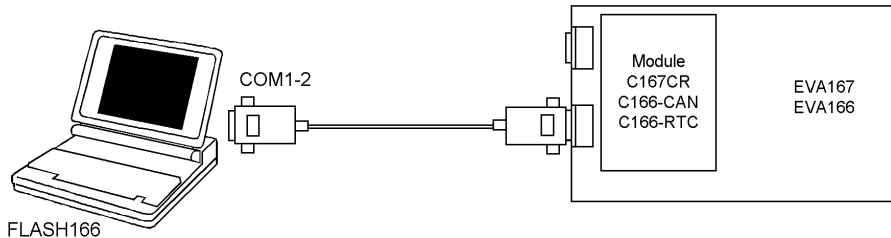
To successfully run this tool the following requirements must be met:

- MSDOS, WIN3.x, WIN9x or WinNT
- 205 kByte of free DOS memory
- no extended memory
- RS232 interface with interrupt
- on targets with the signal BHE#/WRH# (C167, C165, ...) it is assumed, that WRH# is used to select the high byte of two byte organized memories (for the download function only)

1.2. Connecting your PC to the target system

The serial interface ASC0 of the target must be connected to a standard serial port of a PC (COM1 to COM4). The COM ports must be able to generate hardware interrupts, IRQ4 for COM1 or IRQ3 for COM2.

Only the signals TxD and RxD are used. The serial channel uses 8 data bits, one stop bit and no parity bit.



Before you start FLASH166, enable the bootstrap mode (Jumper or switch) and push the reset button of the target system (and release it!).

The utility is started with the general format:

```
FLASH166 /function [filename] [/option_1] ... [/option_n].
```

After invocation the overlay file FLASH166.OVL is used. This file must reside in the same directory as the main program FLASH166.EXE. Otherwise an error message will occur.

Note that the function must be the first argument followed (if needed) by the filename.

If you want to cancel execution of JTAG386, press CTRL-Break-Key.

On any error message the program aborts with an MSDOS error level of one.

1.3. First Example

In the following simple example it is assumed that the COM1 of your PC is connected to the serial interface ASC0 of a C167CR. Two Flash-EPROMs Am29F010 are connected to CS0# of the CPU, bootstrap mode is enabled and target power is on.

Typing

```
FLASH166 /P MYAPP.BIN
```

at the DOS prompt results in the following output:

```
FLASH166 --- Utility for 80C166, C16x and ST10 using bootstrap  
Copyright © FS FORTH-SYSTEME GmbH, Breisach  
Version 3.01 of mm/dd/yyyy
```

- (1) Loading bootstrap code (32 Bytes)
- (2) Loading Target Monitor (316 Bytes)
- (3) Target Monitor located to 00FA40H
- (4) Infineon C167CR
- (5) CPU clock = 19.916.800 MHz
- (6) Configuration loaded from file FLASH166.INI
- (7) Target: Module C167CR, FS FORTH-SYSTEME

- (8) Looking for a known flash device. Please wait..
- (9) Dual AMD 29F010 detected
- (10) Loading flash algorithm (138 Bytes)
- (11) Erasing Flash-EPROM Block #:0 1
Programming File MYAPP.BIN (45578 Bytes)
45578 Bytes programmed
- (12) programming ok

- Erase Time : 1.3 sec
- Programming Time : 13.8 sec

- (1) After receiving the bootstrap loader's acknowledge byte (=0xC5 for a C167CR) the bootstrap code (32 Bytes) is loaded to memory address range 0FA40H to 0FA5F.
- (2) In a second step the target monitor with a length of 284 bytes is loaded to address range 0FA60 to 0FB7BH.
- (3) The current monitor base address is displayed.
- (4) Target CPU is the Infineon C167CR (first byte received from target is 0C5H).
- (5) The target's CPU clock is calculated. Please note that the result may have an error of up to $\pm 5\%$.
- (6) The initialization file FLASH166.INI was found in the current directory.
- (7) The target identification line of the initialization file is printed here.
- (8) Looking for a known Flash Device. Under WinNT this procedure can take several seconds, if the part is at the list of the known parts (see chapter 1.7 "Supported flash devices"). Use the option /DEVICE= to speed up.
- (9) Two Flash-EPROMs AMD 29F010 selected with chip select CS0 could be found.
- (10) After detecting the targets Flash type the appropriate flash programming algorithm is loaded to the target memory at address 0FB6EH (a small part of the target monitor is overwritten).
- (11) In this example two blocks must be erased.
- (12) After programming the file to Flash-EPROM a checksum is calculated.

1.4. Trouble Shooting

If you use SMARTDRV with the write back option enabled, bytes may be lost in case of uploading memory from target to host. Please configure SMARTDRV to write through.

If you work in a DOS-Box in the windows environment there may be problems if you use baud rates higher than 19200 baud. Please specify a lower baud rate or use this utility from a normal DOS prompt.

If you want to use this program in a DOS-Box with baud rates higher than 19200 baud, you must use a serial interface with a 16 Byte FIFO (such as a serial device 16550) and you must configure the windows driver to generate an interrupt with only one or a few bytes in the FIFO. Otherwise bytes may be lost and FLASH166 terminates with an error message.

If there are problems with autodetection of the flash devices use the /DEVICE= option.

Please do not use Flash-EPROMs with any protected sector.

Don't mix different types of flash devices on the module C167CR and other targets with two 8 bit devices.

The program does not handle Intel-HEX files. Please use a standard converting utility to generate a binary image.

1.5. Error Messages

- ***Baud rate out of range:***
Use a standard PC baud rate: 9600, 19200, 38400, 57600, 115200.
- ***Can't open x:\yyy\zzz\FLASH166.OVL***
The overlay file FLASH166.OVL must be in the same directory as FLASH166.EXE.
- ***Checksum error in file transfer***
After transferring of a file from target to host or from host to target a checksum is calculated.
- ***Checksum error in target monitor***
The target monitor wasn't send correctly.
- ***COMx not installed***
The COM port specified with /COMx cannot be found. Please check the COM port or specify a installed COM port. Check your BIOS setup.
- ***Configuration file XYZ not found***
The file specified with the option /INI= wasn't found.
- ***Disk full***
Writing a output file was aborted as a result of missing disk space.
- ***Error creating file:***
The output file could not be opened. Please check free disk space or write protection.
- ***Error on transfer of bootstrap code***
After sending of 32 bytes bootstrap code, the target must respond with an acknowledge byte ('V').
- ***Error on transfer of target monitor***
After loading the target monitor, target must respond with an acknowledge byte (0xAA).

- **Failed to load Flash program algorithm**
The Flash program algorithm wasn't send correctly.
- **illegal echo:**
There is a communication error between target and host.
- **illegal Flash Type**
The name specified with the option /DEVICE= must be one of the list of chapter 1.7
- **illegal function:**
The first parameter must be a valid function. See chapter 2 for a list of supported functions.
- **illegal option:**
See chapter 2 for a list of supported options.
- **illegal number:**
The specified number couldn't be interpret as a valid number. Check the relevant number base.
- **illegal Pin Type:**
The name specified with the option /PIN= must be one of the list of chapter 1.6.
- **Input file not found:**
The specified file could not be found.
- **IRQ# out of range:**
The specified IRQ is out of range. Allowed values are 3..15d.
- **missing filename**
Most functions need a filename as second parameter.
- **missing option /LENGTH=**
Some functions need the option /LENGTH= to be defined.

- **missing option /PIN=**
Some functions need the option /PIN= to be defined.
- **Number of wait states out of range:**
The value of the option /WAIT= is below zero or above 15d.
- **Option /COM-BASE= out of range:**
The value of the option /COM-BASE= is below zero or above 0FFFFh.
- **Specify only one of these options:**
Some options are exclusive (i.e. /8BIT and /16BIT). Don't mix them.
- **Specify option /IRQ= if using /COM-BASE=**
If you select the serial interface with option /COM-BASE= you additionally must specify the used IRQ.
- **Specify option /IRQ= if using COM3 or COM4**
If using COM3 or COM4 the IRQ# must be specified.
- **There is no UART at I/O-Address hhhh**
There is no UART at the specified address. Please check the serial interface settings in the BIOS setup of your PC.
- **Timeout! Boot mode not active**
There was no target response after sending a NUL-char to the target. Please reenter the bootstrap mode (Set Bootstrap jumper/switch and push the reset button of the target). Check the serial interface to ASC0 of the target. Check the used serial interface and the IRQ setting on the PC.
Check the existence of another DOS-Box, which reclaims the serial port.
- **Timeout programming**
After sending a block to target, no acknowledge byte was received.
- **Value of option /FILE-OFFSET= out of range**
The value of the option /FILE-OFFSET= points behind end of file.

- ***Value of option /OFFSET= out of range***
The value of the option /OFFSET= points behind the end of the Flash-EPROM.
- ***Wrong Identifier (xxxx)***
No valid flash identifier found. Check the specified chip select signal and the bus width. Try with the option /DEVICE=.

1.6. Initialization file FLASH166.INI

This file is used to set the CPU signals for input/output. In case of output signal an additional parameter is used to set the default level to high or low. This file can be used to adapt your own design with a member of the 80C166 family. The Target-Entry is used to identify your design which is displayed with most commands.

When the FLASH166 software is started it scans the current directory for an existing initialization file named FLASH166.INI. If no entry is found the port pins remain in their reset state. You may also specify the initialization file with the option /INI= . If the specified file isn't found, the program aborts with an error message.

The following pins can be set a defined value:

P0.0	P0.1	P0.2	P0.3	P0.4	P0.5
P0.6	P0.7	P0.8	P0.9	P0.10	P0.11
P0.12	P0.13	P0.14	P0.15	P0L.0	P0L.1
P0L.2	P0L.3	P0L.4	P0L.5	P0L.6	P0L.7
P0H.0	P0H.1	P0H.2	P0H.3	P0H.4	P0H.5
P0H.6	P0H.7	P1.0	P1.1	P1.2	P1.3
P1.4	P1.5	P1.6	P1.7	P1.8	P1.9
P1.10	P1.11	P1.12	P1.13	P1.14	P1.15
P1L.0	P1L.1	P1L.2	P1L.3	P1L.4	P1L.5
P1L.6	P1L.7	P1H.0	P1H.1	P1H.2	P1H.3
P1H.4	P1H.5	P1H.6	P1H.7	P2.0	P2.1
P2.2	P2.3	P2.4	P2.5	P2.6	P2.7
P2.8	P2.9	P2.10	P2.11	P2.12	P2.13
P2.14	P2.15	P3.0	P3.1	P3.2	P3.3
P3.4	P3.5	P3.6	P3.7	P3.8	P3.9
P3.10	P3.11	P3.12	P3.13	P3.14	P3.15
P4.0	P4.1	P4.2	P4.3	P4.4	P4.5
P4.6	P4.7	P6.0	P6.1	P6.2	P6.3
P6.4	P6.5	P6.6	P6.7	P7.0	P7.1
P7.2	P7.3	P7.4	P7.5	P7.6	P7.7
P8.0	P8.1	P8.2	P8.3	P8.4	P8.5
P8.6	P8.7				

The CPU pins can also be used with the /BLINK function in chapter 2.7 to test the signals on your design.

Example of FLASH166.INI:

```
// Description file for a specific target
Target: Module FS C167CR, FS FORTH-SYSTEME
// Port pins may be switched to a defined level.
// All undefined pins are left in their previous state,
// which means they are configured as input after reset.
```

```
P2.1 Out,Hi
P2.7 Out,Hi
P2.0 Out,Lo
```

The initialization file consists of a view lines with the pin name followed by the direction (for output signals default level added) and the target description line. Undefined signals are left in their previous state, which means they are configured as input after reset.

Changes to the structure of the file could result in errors. Remarks can be added by using //.

1.7. Supported flash devices

The following types could be used with the /DEVICE= option:

AM29F010	: AMD 29F010	
AM29F010*2	: Dual AMD 29F010	
* AM29LV010	: AMD 29LV010	(3.3V)
* AM29LV010*2	: Dual AMD 29LV010	(3.3V)
* AM29F100BW	: AMD 29F100B	(word mode)
* AM29F100BB	: AMD 29F100B	(byte mode)
* AM29F100TW	: AMD 29F100T	(word mode)
* AM29F100TB	: AMD 29F100T	(byte mode)
* AM29F200BW	: AMD 29F200B	(word mode)
* AM29F200BB	: AMD 29F200B	(byte mode)
* AM29F200TW	: AMD 29F200T	(word mode)
* AM29F200TB	: AMD 29F200T	(byte mode)
* AM29LV200BW	: AMD 29F200B	(3.3V, word mode)
* AM29LV200BB	: AMD 29F200B	(3.3V, byte mode)
* AM29LV200TW	: AMD 29F200T	(3.3V, word mode)
* AM29LV200TB	: AMD 29F200T	(3.3V, byte mode)
AM29F040	: AMD 29F040	
AM29F040*2	: Dual AMD 29F040	
* AM29F400BW	: AMD 29F400B	(word mode)
* AM29F400BB	: AMD 29F400B	(byte mode)
AM29F400TW	: AMD 29F400T	(word mode)
AM29F400TB	: AMD 29F400T	(byte mode)
* AM29LV400BW	: AMD 29LV400B	(3.3V, word mode)
* AM29LV400BB	: AMD 29LV400B	(3.3V, byte mode)
* AM29LV400TW	: AMD 29LV400T	(3.3V, word mode)
* AM29LV400TB	: AMD 29LV400T	(3.3V, byte mode)
* AM29LV004B	: AMD 29LV004B	(3.3V)
* AM29LV004T	: AMD 29LV004T	(3.3V)
AM29F080	: AMD 29F080	
* AM29F080*2	: Dual AMD 29F080	
* AM29LV081	: AMD 29LV081	(3.3V)
* AM29LV081*2	: Dual 29LV081	(3.3V)
* AM29F800BW	: AMD 29F800B	(word mode)
* AM29F800BB	: AMD 29F800B	(byte mode)
AM29F800TW	: AMD 29F800T	(word mode)
* AM29F800TB	: AMD 29F800T	(byte mode)
* AM29LV800BW	: AMD 29LV800B	(3.3V, word mode)
* AM29LV800BB	: AMD 29LV800B	(3.3V, byte mode)

AM29LV800TW	: AMD 29LV800T	(3.3V, word mode)
* AM29LV800TB	: AMD 29LV800T	(3.3V, byte mode)
* AM29LV008T	: AMD 29LV008T	(3.3V)
* AM29LV008B	: AMD 29LV008B	(3.3V)
AM29F016	: AMD 29F016	
AM29F016*2	: Dual AMD 29F016	
* AM29F017	: AMD 29F017	
AM29F017*2	: Dual AMD 29F017	
AM29LV017*2	: Dual AMD 29LV017	(3.3V)
* AM29F160BW	: AMD 29F160B	(word mode)
* AM29F160BB	: AMD 29F160B	(byte mode)
* AM29F160TW	: AMD 29F160T	(word mode)
* AM29F160TB	: AMD 29F160T	(byte mode)
* AM29F160BB*2	: Dual AMD 29F160B	(byte mode)
* AM29F160TB*2	: Dual AMD 29F160T	(byte mode)
AM29LV160BW	: AMD 29LV160B	(3.3V, word mode)
* AM29LV160BB	: AMD 29LV160B	(3.3V, byte mode)
* AM29LV160TW	: AMD 29LV160T	(3.3V, word mode)
* AM29LV160TB	: AMD 29LV160T	(3.3V, byte mode)
AM29LV160BB*2	: Dual AMD 29LV160B	(3.3V, byte mode)
* AM29LV160TB*2	: Dual AMD 29LV160T	(3.3V, byte mode)
AM29F032	: AMD 29F032	
AM29F032*2	: Dual AMD 29F032	
* AM29LV033	: AMD 29LV033	(3.3V)
* AM29LV033*2	: Dual AMD 29F033	(3.3V)
* AM29LV640	: AMD 29LV160	(3.3V, word mode)
* I28F001T	: Intel 28F001T	
* I28F001T*2	: Dual Intel 28F001T	
* I28F001B	: Intel 28F001B	
* I28F200BW	: Intel 28F200B	(word mode)
* I28F200BB	: Intel 28F200B	(byte mode)
* I28F200TW	: Intel 28F200T	(word mode)
* I28F200TB	: Intel 28F200T	(byte mode)
* I28F002B	: Intel 28F002B	
* I28F002T	: Intel 28F002T	
* I28F400BW	: Intel 28F400B	(word mode)
* I28F400BB	: Intel 28F400B	(byte mode)
I28F400TW	: Intel 28F400T	(word mode)
I28F400TB	: Intel 28F400T	(byte mode)
* I28F004B	: Intel 28F004B	
* I28F004T	: Intel 28F004T	
* I28F400B3B	: Intel 28F400B3B	Boot Block Smart3 Bottom

* I28F400B3T	: Intel 28F400B3T	Boot Block Smart3 Top
* I28F004S5	: Intel 28F004	FlashFile Smart3/5
* I28F800BW	: Intel 28F800B	(word mode)
* I28F800BB	: Intel 28F800B	(byte mode)
I28F800TW	: Intel 28F800T	(word mode)
* I28F800TB	: Intel 28F800T	(byte mode)
* I28F800B3B	: Intel 28F800B3B	Boot Block Smart3 Bottom
* I28F800B3T	: Intel 28F800B3T	Boot Block Smart3 Top
* I28F008B	: Intel 28F008B	
* I28F008T	: Intel 28F008T	
I28F008SA	: Intel 28F008	FlashFile 12V
I28F008SA*2	: Dual Intel 28F008	FlashFile 12V
I28F008S5	: Intel 28F008	FlashFile Smart3/5
I28F008S5*2	: Dual Intel 28F008	FlashFile Smart3/5
I28F016W	: Intel 28F016	FlashFile (word mode)
* I28F016B	: Intel 28F016	FlashFile (byte mode)
* I28F016S5	: Intel 28F016	FlashFile Smart3/5
* I28F160B3B	: Intel 28F160	Boot Block Smart3 Bottom
* I28F160B3T	: Intel 28F160	Boot Block Smart3 Top
* I28F160S5W	: Intel 28F160	FlashFile Smart5 (word mode)
* I28F160S5B	: Intel 28F160	FlashFile Smart5 (byte mode)
I28F320S5W	: Intel 28F320	FlashFile Smart3/5 (word mode)
* I28F320S5B	: Intel 28F320	FlashFile Smart3/5 (byte mode)
* I28F320J5W	: Intel 28F320	StrataFlash (word mode)
* I28F320J5B	: Intel 28F320	StrataFlash (byte mode)
* I28F320J3W	: Intel 28F320	StrataFlash (3.3V, word mode)
* I28F320J3B	: Intel 28F320	StrataFlash (3.3V, byte mode)
* I28F640J5W	: Intel 28F640	StrataFlash (word mode)
* I28F640J5B	: Intel 28F640	StrataFlash (byte mode)
* I28F640J3W	: Intel 28F640	StrataFlash (3.3V, word mode)
* I28F640J3B	: Intel 28F640	StrataFlash (3.3V, byte mode)
* I28F128J3W	: Intel 28F128	StrataFlash (3.3V, word mode)
* I28F128J3B	: Intel 28F128	StrataFlash (3.3V, byte mode)
* MBM29F200BW	: Fujitsu 29F200B	(word mode)
* MBM29F200BB	: Fujitsu 29F200B	(byte mode)
* MBM29F200TW	: Fujitsu 29F200T	(word mode)
* MBM29F200TB	: Fujitsu 29F200T	(byte mode)
* MBM29LV200BW	: Fujitsu 29LV200B	(3.3V, word mode)
* MBM29LV200BB	: Fujitsu 29LV200B	(3.3V, byte mode)
* MBM29LV200TW	: Fujitsu 29LV200T	(3.3V, word mode)
* MBM29LV200TB	: Fujitsu 29LV200T	(3.3V, byte mode)
* MBM29F002B	: Fujitsu 29F002B	

* MBM29F002SB	: Fujitsu 29F002SB	(TSOP40)
* MBM29LV002B	: Fujitsu 29LV002B	(3.3V)
* MBM29F002T	: Fujitsu 29F002T	
* MBM29F002ST	: Fujitsu 29F002ST	(TSOP40)
* MBM29LV002T	: Fujitsu 29LV002T	(3.3V)
* MBM29F040	: Fujitsu 29F040	
* MBM29F400BW	: Fujitsu 29F400B	(word mode)
* MBM29F400BB	: Fujitsu 29F400B	(byte mode)
* MBM29F400TW	: Fujitsu 29F400T	(word mode)
* MBM29F400TB	: Fujitsu 29F400T	(byte mode)
* MBM29LV400BW	: Fujitsu 29LV400B	(3.3V, word mode)
* MBM29LV400BB	: Fujitsu 29LV400B	(3.3V, byte mode)
* MBM29LV400TW	: Fujitsu 29LV400T	(3.3V, word mode)
* MBM29LV400TB	: Fujitsu 29LV400T	(3.3V, byte mode)
* MBM29LV004B	: Fujitsu 29LV004B	(3.3V)
* MBM29LV004T	: Fujitsu 29LV004T	(3.3V)
* MBM29F080	: Fujitsu 29F080	
* MBM29LV080	: Fujitsu 29LV080	(3.3V)
* MBM29F800BW	: Fujitsu 29F800B	(word mode)
* MBM29F800BB	: Fujitsu 29F800B	(byte mode)
* MBM29F800TW	: Fujitsu 29F800T	(word mode)
* MBM29F800TB	: Fujitsu 29F800T	(byte mode)
* MBM29LV800BW	: Fujitsu 29LV800B	(3.3V, word mode)
* MBM29LV800BB	: Fujitsu 29LV800B	(3.3V, byte mode)
* MBM29LV800TW	: Fujitsu 29LV800T	(3.3V, word mode)
* MBM29LV800TB	: Fujitsu 29LV800T	(3.3V, byte mode)
* MBM29LV008B	: Fujitsu 29LV008B	(3.3V)
* MBM29LV008T	: Fujitsu 29LV008T	(3.3V)
* MBM29F016	: Fujitsu 29F016	
* MBM29F016*2	: Dual Fujitsu 29F016	
* MBM29F017	: Fujitsu 29F017	
* MBM29F017*2	: Dual Fujitsu 29F017	
* MBM29F160BW	: Fujitsu 29LV160B	(word mode)
* MBM29F160BB	: Fujitsu 29LV160B	(byte mode)
* MBM29F160TW	: Fujitsu 29LV160T	(word mode)
* MBM29F160TB	: Fujitsu 29LV160T	(byte mode)
* MBM29F160BB*2	: Dual Fujitsu 29LV160B	(byte mode)
* MBM29F160TB*2	: Dual Fujitsu 29LV160T	(byte mode)
* MBM29LV160BW	: Fujitsu 29LV160B	(3.3V, word mode)
* MBM29LV160BB	: Fujitsu 29LV160B	(3.3V, byte mode)
* MBM29LV160TW	: Fujitsu 29LV160T	(3.3V, word mode)
* MBM29LV160TB	: Fujitsu 29LV160T	(3.3V, byte mode)
MBM29LV160BB*2	: Dual Fujitsu 29LV160B	(3.3V, byte mode)

* MBM29LV160TB*2	: Dual Fujitsu 29LV160T	(3.3V, byte mode)
MBM29F032	: Fujitsu 29F032	
MBM29F032*2	: Dual Fujitsu 29F032*2	
* M29F100BW	: ST 29F100B	(word mode)
* M29F100BB	: ST 29F100B	(byte mode)
* M29F100TW	: ST 29F100T	(word mode)
* M29F100TB	: ST 29F100T	(byte mode)
M29F200BW	: ST 29F200B	(word mode)
* M29F200BB	: ST 29F200B	(byte mode)
* M29F200TW	: ST 29F200T	(word mode)
* M29F200TB	: ST 29F200T	(byte mode)
* M29F040	: ST 29F040	
* M29F040*2	: Dual ST 29F040	
* M29W040	: ST 29W040	(3.3V)
* M29W040*2	: Dual ST 29W040	(3.3V)
* M29F400BW	: ST 29F400B	(word mode)
* M29F400BB	: ST 29F400B	(byte mode)
* M29F400TW	: ST 29F400T	(word mode)
* M29F400TB	: ST 29F400T	(byte mode)
* M29W400BW	: ST 29W400B	(3.3V, word mode)
* M29W400BB	: ST 29W400B	(3.3V, byte mode)
* M29W400TW	: ST 29W400T	(3.3V, word mode)
* M29W400TB	: ST 29W400T	(3.3V, byte mode)
* M29W004B	: ST 29W004B	(3.3V)
* M29W004T	: ST 29W004T	(3.3V)
* M29F800BW	: ST 29F800B	(word mode)
* M29F800BB	: ST 29F800B	(byte mode)
* M29F800TW	: ST 29F800T	(word mode)
* M29F800TB	: ST 29F800T	(byte mode)

The flash devices signed with '*' are not yet tested.

2. FLASH166 Parameter Description

When you start FLASH166.EXE without any parameters the following help screen with all possible functions and options is displayed:

```
FLASH166 --- Utility for 80C166, C16x and ST10 using bootstrap
Copyright © FS FORTH-SYSTEME GmbH, Breisach
Version 3.01 of mm/dd/yyyy
```

Programming of Flash-EPROMs and debugging of the hardware on targets with Infineon 80C166 and C16x and STMicroelectronics ST10.

Usage: FLASH166 /function [filename] [/option_1] ... [/option_n]

Supported functions:

```
/P      : Program a Flash Device
/R      : Read a Flash Device to file
/V      : Verify a Flash Device with file
/D      : Download to target memory
/U      : Upload from target to host
/DUMP   : Make a target dump
/BLINK  : Toggle a CPU pin
/PIN?   : Test a CPU pin
/LIST   : Print a list of supported Flash devices
```

Supported Options:

```
/CS0      /CS1      /CS2      /CS3      /CS4
/NOCS     /PAUSE    /P         /TOP      /16BIT
/8BIT     /NOMAN    /NOMUX    /MUX      /WRCS
/RDCS     /NOERASE  /ERASEALL /EINIT    /NOEINIT
/COM1     /COM2     /COM3     /COM4     /COM-BASE=
/WAIT=    /LENGTH=  /L=       /FILE-OFFSET=
/FO=      /OFFSET=  /O=       /DELAY=   /BAUD=
/IRQ=     /DEVICE-BASE=
/EXE=     /W_EXE    /W_EXE=   /DEVICE=  /PIN=
/OUT=     /INI=
```

The following options are valid for all functions:

`/8BIT /16BIT`

Specifies the bus interface to the target device. If this parameter is not specified, the standard hardware configuration is used. These parameters are ignored on the FS FORTH-SYSTEME modules 80C166-CAN and 80C166-RTC.

`/BAUD=x` with x=9600, 19200, 38400, 57600

A baud rate may be specified. If the target runs with a 10 MHz CPU clock, the maximum baud rate is 38400. With a CPU clock of 18.436MHz, you can also specify `BAUD=115200`.

Default: `/BAUD=38400`

`/COM1 /COM2 /COM3 /COM4`

A serial port may be specified where the target is connected.

Default: `/COM1`

`/COM-BASE=hhh1`

Instead of using the options `/COM1 ... /COM4` to select the serial interface you can use the option `/COM-BASE=`.

`/EXE=hhhhh`

Before terminating FLASH166 the target branches from the monitor loop to the specified address. This option may be used to execute a program after loading it to target memory.

`/INI=file`

An initialization file may be specified. By default the current directory is searched for the file FLASH166.INI. If this file is not found and no initialization file is specified in the command line, the port pins of the CPU are not initialized. (see also chapter 1.6 "Initialization file FLASH166.INI")

`/IRQ=dd2`

An IRQ number for the used serial port can be specified. Allowed values are 3..15d. If using the option `/COM-BASE=` or `/COM3` or `/COM4` specifying the `IRQ#` is mandatory.

¹hhhhh=number base is HEX

²dd=number base is decimal

Default: /IRQ=4 for COM1
 /IRQ=3 for COM2

/MUX /NOMUX

Specifies the bus interface to the target device. If this parameter is not specified, the standard hardware configuration is used. This parameters are ignored on the modules 80C166-CAN and 80C166-RTC.

/PAUSE

With the option **/PAUSE** you can force the program to stop after each screen. Please do not use this option if you redirect the output to a file.

Abbreviation /P

/RDCS /WRCS

Specifies the kind of chip select signal. This options does not apply to the 80C166. See the C16x user's manual for more information.

/WAIT=dd

Specifies the number of wait states. Allowed values are 0..15.

Default: /WAIT=15

/WATCH=

With the option **/WATCH=** a pin can be specified, which is toggled twice per second, while the program is active. This pin may be the trigger of a watchdog.

/W_EXE=hhhhh

Same as **/EXE=** but before branching to the specified address, the target waits for the reception of a char on the serial interface.

2.1. Program a Flash Device

Usage: FLASH166 /P filename [optionlist]

The specified file is programmed to the flash memory. Finally a checksum is calculated. If the checksum fails, the contents of the flash memory is written to a file with the extension DMP.

The type of flash device is normally detected by the software. When autodetect fails you should use the /DEVICE= option to set the right flash device. The known devices are shown in chapter 1.7 "Supported flash devices". Don't mix different types of flash devices on targets with two 8 bit devices.

If programming overlaps with the chip internal components (80C166: FA00..FFFF, C16x:E000..FFFF) this region is skipped.

Options:

/DEVICE=devicename

The device is detected automatically by switching to autoselect mode. In case of trouble you should select the device by using this parameter to avoid autodetection. Under WinNT the autodetection can take some seconds. Specifying the option /DEVICE= results in faster operation.

/NOMAN

If you use a flash device which is identical to one of the supported parts, but is from a different manufacturer, with this option you can suppress the comparison of the manufacturer identification code. We recommend to use this option always together with the /DEVICE= option to avoid failures in autodetection.

/DEVICE-BASE=hhhhh³

By default a device start address of 0h for a 80C166 or 400000h (=4MByte) for all other members of the microcontroller family is used for accesses to the flash device. This is done, to minimize conflicts with the chip internal components (SFRs, CAN-Controller, CPU core).

Default: /DEVICE-BASE=40000= (or 0)

Abbreviation: /DB=

³hhhhh=number base is HEX

/OFFSET=hhhhh

The programming starts at an offset of hhhhh relative to the start address of the flash device. If the offset is negative, the offset specifies an address relative to the end of the flash device. See also option **/TOP**

Default: **/OFFSET=0**

Abbreviation: **/O=**

/TOP

If the option **/TOP** is specified the option **/OFFSET=** specifies the address where the programming ends (plus one) instead of the starting address.

/FILE-OFFSET=hhhhh

If **FILE-OFFSET** is specified, the first hhhhh bytes of the file are skipped and not programmed to target.

Default: **/FILE-OFFSET=0**

Abbreviation: **/FO=**

/LENGTH=hhhhh

The number of programmed bytes may be limited to **LENGTH**. If no **LENGTH** is specified the whole file is programmed.

Default: **/LENGTH=1000000 (16 Mbytes)**

Abbreviation: **/L=**

/NOERASE

This option prevents the flash device from being erased.

/ERASEALL

Erase the whole flash device. If this option isn't set only the blocks are erased where new data is written to.

/CS0 /CS1 /CS2 /CS3 /CS4

This options may be used to specify a chip select signal to the flash device. If more than one chip selects are specified an error message occurs. This option does not apply to the 80C166 and is ignored.

Default: **/CS0 for C16x**

Example: FLASH166 /P ROMDOS.ROM /FO=60000 /O=60000 /COM2

This example may be used to program 128 Kbytes of the file ROMDOS.ROM (with a length of 80000H bytes) to flash memory.

2.2. Verify a Flash Device with file

Usage: FLASH166 /V filename [optionlist]

The target's flash device is verified with the contents of the specified file. If the verify fails, the contents of the flash memory is written to a file with the extension DMP.

The type of flash device is normally detected by the software. When autodetect fails you should use the /DEVICE= option to set the right flash device. The known devices are shown in chapter 1.7 "Supported flash devices". Don't mix different type of flash devices on the module C167CR and other targets with two 8 bit devices.

If the verify function overlaps with the chip internal components (80C166: FA00..FFFF, C16x: E000..FFFF) this region is skipped.

Options:

/DEVICE=devicename

See function /P (Chapter 2.1)

/NOMAN

See function /P (Chapter 2.1)

/DEVICE-BASE=hhhhhh

See function /P (Chapter 2.1)

/OFFSET=hhhhhh

See function /P (Chapter 2.1)

/TOP

See function /P (Chapter 2.1)

/FILE-OFFSET=hhhhhh

See function /P (Chapter 2.1)

/LENGTH=hhhhhh

See function /P (Chapter 2.1)

/CS0 /CS1 /CS2 /CS3 /CS4
See function /P (Chapter 2.1)

Example: FLASH166 /V ROMDOS.ROM /FO=60000 /O=60000 /COM=2

This example may be used to verify 128 Kbytes of the file ROMDOS.ROM (with a length of 80000H bytes) with the target's flash memory.

2.3. Read a Flash Device to file

Usage: FLASH166 /R filename [optionlist]

The contents of the target's flash device is written to the specified file. Finally a checksum is calculated. If the checksum fails an error message is printed.

The type of flash device is normally detected by the software. When autodetect fails you should use the /DEVICE= option to set the right flash device. The known devices are shown in chapter 1.7 "Supported flash devices". Don't mix different types of flash devices on the module C167CR and other targets with two 8 bit devices.

If the read function overlaps with the chip internal components (80C166: FA00..FFFF, C16x:E000..FFFF) zeros are written in this region.

Options:

/DEVICE=devicename

See function /P (Chapter 2.1)

/NOMAN

See function /P (Chapter 2.1)

/DEVICE-BASE=hhhhhh

See function /P (Chapter 2.1)

/OFFSET=hhhhhh

See function /P (Chapter 2.1)

/TOP

See function /P (Chapter 2.1)

/LENGTH=hhhhhh

The number of read bytes may be limited to LENGTH. If no LENGTH is specified the whole flash device is read (if no offset is specified).

/CS0 /CS1 /CS2 /CS3 /CS4

See function /P (Chapter 2.1)

Example: FLASH166 /R ROMDOS.ROM /FO=60000 /O=60000 /COM=2

This example may be used to read 128 Kbytes of the target's flash device (with a length of 80000H bytes) to the file ROMDOS.ROM.

2.4. Download to Target Memory

Usage: FLASH166 /D filename [optionlist]

The specified file is loaded to target memory. Finally a checksum is calculated. If the checksum fails, the contents of the target memory is written to a file with the extension DMP.

If the download overlaps with the chip internal components (80C166: FA00..FFFF, C16x:E000..FFFF) this region is skipped.

Options:

/OFFSET=hhhhh⁴

Download starts at an offset of hhhhhh.

Default: /OFFSET=0

Abbreviation: /O=

/TOP

If the option /TOP is specified the option /OFFSET= specifies the address where the download ends (plus one) instead of the starting address.

/FILE-OFFSET=hhhhh

If FILE-OFFSET is specified, the first hhhhhh bytes of the file are skipped and not loaded to target.

Default: /FILE-OFFSET=0

Abbreviation: /FO=

/LENGTH=hhhhh

The number of loaded bytes may be limited to LENGTH. If no LENGTH is specified the whole file is programmed.

Default: /LENGTH=1000000 (16 Mbytes)

Abbreviation: /L=

/CS0 /CS1 /CS2 /CS3 /CS4

This options may be used to specify a chip select signal to the memory. If more than one chip selects are specified an error message occurs. This option does

⁴hhhhh=number base is HEX

not apply to the 80C166 and is ignored. The default chip select is CS1, because the static memory on the FS FORTH-SYSTEME module C167CR is connected to CS1.

Default: /CS1 for C16x

Example: FLASH166 /D HELLO.BIN

This example loads the binary file HELLO.BIN to target memory.

2.5. Upload from Target to Host

Usage: FLASH166 /U filename /L=hhhhhh [optionlist]

Target data is written to the specified file. Finally a checksum is calculated. If the checksum fails, an error message is displayed.

Options:

/OFFSET=hhhhhh

See function /D (Chapter 2.4)

/TOP

See function /D (Chapter 2.4)

/LENGTH=hhhhhh

The number of uploaded bytes must be specified. If no length is specified an error message occurs.

Abbreviation: /L=

/CS0 /CS1 /CS2 /CS3 /CS4

See function /D (Chapter 2.4)

Example: FLASH166 /U HELLO /O=-10000 /L=10000

This example writes the last 64 kByte of the target's memory to file HELLO.

2.6. Dump Target Memory

Usage: FLASH166 /DUMP [optionlist]

A Hex-Dump of the target memory is printed on the standard output.

Options:

/OFFSET=hhhhh⁵

The memory dump starts at an offset of hhhhh.

Default: /OFFSET=0

Abbreviation: /O=

/TOP

See function /D (Chapter 2.4)

/LENGTH=hhhhh

The number of displayed bytes may be specified.

Default: /LENGTH=100

Abbreviation: /L=

/CS0 /CS1 /CS2 /CS3 /CS4

See function /D (Chapter 2.4)

Default: /CS1 for C16x

⁵hhhhh=number base is HEX

Example1: FLASH166 /DUMP /CS3 /8BIT /l=10

This example makes a memory dump of the RTC on the FS FORTH-SYSTEME module C167CR.

Example2: FLASH166 /DUMP /O=EF00 >CAN.DMP

This example makes a memory dump of the CAN controller of a C167CR and redirects the output to the file CAN.DMP.

2.7. Toggle CPU pins

Usage: FLASH166 /BLINK /PIN=pinname [optionlist]

This command allows to test the hardware by blinking with LEDs or toggling CPU signals. Faster signals can be generated by setting the delay option to zero. This can be a very helpful feature to watch signals on oscilloscope. The function is terminated by pressing any key.

Options:

/PIN=pinname

CPU pin to toggle. Most pins of the list in chapter 1.6 "Initialization file FLASH166.INI" can be used. If no pin is specified an error message occurs. If you type /PIN= without any pin declaration a list of the CPU pins is displayed. Please note that not all pins are available on all CPU types. Usage of some pins may be restricted, i.e. address signals.

/DELAY=dddddd⁶

Time to wait to next change of signal. This option can be adjusted to get optimum signals for measures with the oscilloscope.

Default: /DELAY=10000

Example: FLASH166 /BLINK /PIN=P2.0 /DELAY=0

This example toggles the Port P2.0 which can be followed with the use of an oscilloscope.

⁶dddddd=number base is decimal

2.8. List of supported Flash Devices

Usage: FLASH166 /LIST

This command lists all supported flash devices to screen if not redirected to file or device. Flash devices signed with '*' are not yet tested.