

# **FUJITSU FLASH MCU Programmer for FR Specifications**



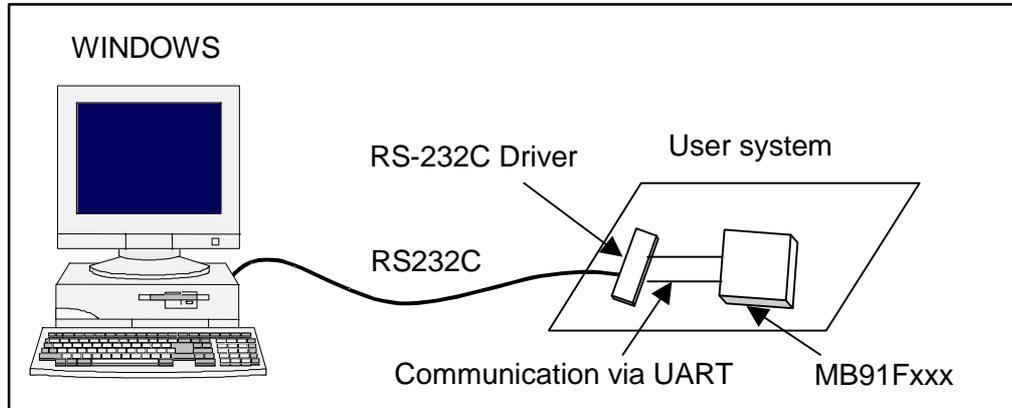
FUJITSU FLASH  
MCU Programmer for FR  
Specifications  
Version 1.7 20 June 2003  
Software version number: VO1L08  
©2002 FUJITSU LIMITED Printed in Japan

1. Circuit diagrams utilizing Fujitsu products are included as a mean of illustrating typical semiconductor applications. Complete information sufficient for construction proposes is not necessarily given.
2. The information contained in this document has been carefully checked and is believed to be reliable. However, Fujitsu assumes no responsibility for inaccuracies.
3. The information contained in this document does not convey any license under the copy right, patent right to trademarks claimed and owned by Fujitsu.
4. Fujitsu reserved the right to change products or specifications without notice.
5. No part of this publication may be copied or reproduced in any form or by any means, or transferred to any third party without prior written consent of Fujitsu.
6. The products described in this document are not intended for use in equipment requiring high reliability, such as marine relays and medical life-support systems. For such applications, contact your Fujitsu sales representative.
7. If the products and technologies described in this document are controlled by the Foreign Exchange and Foreign Trade Control Act established in Japan, their export is subject to prior approval based on the said act.

# CONTENTS

1. CONFIGURATION DIAGRAM.....	1
2. COMPATIBLE MICROCONTROLLERS .....	2
3. EXAMPLE OF CONNECTION FOR ON-BOARD REPROGRAMMING BY PROGRAMMER.....	3
4. PINS USED FOR ON-BOARD REPROGRAMMING.....	4
5. TIMING CHART FOR EACH PIN .....	8
6. INSTALLATION AND EXECUTION OF SOFTWARE .....	9
7. PROGRAMMER FUNCTIONS .....	10
7.1 Downloading .....	11
7.2 Erasing and Programming.....	13
7.3 Continuously writing mode .....	16
8. SECURITY FUNCTION OF MB91F154/155.....	17
9. SECURITY FUNCTION OF MB91F360 SERIES.....	20
10. STATUS OF OPERATION CHECK.....	22
11. OTHERS .....	23
12. CAUTIONS.....	26

## 1. CONFIGURATION DIAGRAM



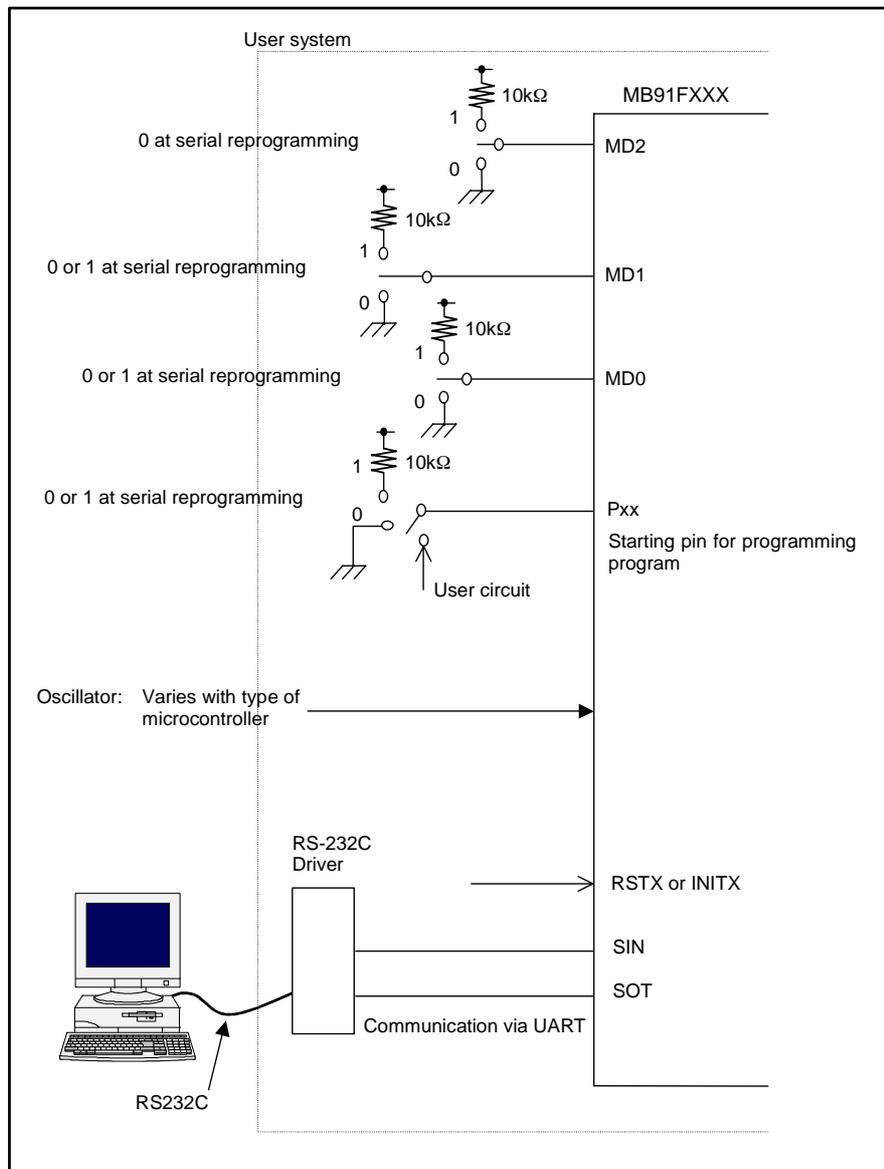
Using RS-232C cable connected to the personal computer (Windows PC), flash memory data in the microcontroller mounted in the user system can be reprogrammed. Note that the user system must have an RS-232C driver for communication with the microcontroller UART.

## 2. COMPATIBLE MICROCONTROLLERS

This software is compatible with the following microcontrollers.

MB91F005/A	MB91F109	MB91F120
MB91F127	MB91F128	MB91F133
MB91F154	MB91F155	MB91F158
MB91F233	MB91F264	MB91F353
MB91F355	MB91F362	MB91F365
MB91F366	MB91F367	MB91F368
MB91F369		

### 3. EXAMPLE OF CONNECTION FOR ON-BOARD REPROGRAMMING BY PROGRAMMER



The MD2, MD1 and MD0 pins, and P20 pin cannot be controlled by the PC and should be set in the user system. During serial reprogramming, when the RSTX or INITX pin is set from “Low” to “High” level after setting the MD2, MD1 and MD0 pins, and Pxx pin (programming program starting pin that varies with type of microcontroller, see **Chapter 4**), the microcontroller enters the serial reprogramming mode, enabling serial reprogramming from the PC.

After the reprogramming, control is shifted to the normally-used mode as for MD2, MD1 and MD0 pins and to the user circuit side as for Pxx pin. Then setting from “Low” to “High” level executes user program.

Note: When programming data to mass-produced products using the Yokogawa Digital Computer serial programmer some time in the future, it is best to generate the patterns for serial clock pins on the printed circuit board according to the connection example for serial programming described in the **hardware manual** for each microcontroller.

## 4. PINS USED FOR ON-BOARD REPROGRAMMING

### (1) Setting for MB91F005/F005A

Pin	Function	Supplementary Explanation
MD2, MD1, MD0	Mode pins	Should be controlled in flash memory reprogramming mode. When MD2 is set to "L" and MD1 and MD0 are set to "H", they enter the reprogramming mode.
P20	Starting pin for programming program	Set P20 to "L" in the flash reprogramming mode.
RSTX	Reset pin	Cancel reset after setting MD2, MD1, MD0 and P20 to the flash reprogramming mode.
SIN0	Serial data input pin	Use UART0.
SOT0	Serial data output pin	Use UART0.
	Power supply voltage	3.3-V and 5-V products

### (2) Setting for MB91F109

Pin	Function	Supplementary Explanation
MD2, MD1, MD0	Mode pins	Should be controlled in flash memory reprogramming mode. When MD2 and MD1 are set to "H" and MD0 is set to "L", they enter the reprogramming mode.
P20, P21	Starting pin for programming program	Set P20 and P21 to "L" in the flash reprogramming mode.
RSTX	Reset pin	Cancel reset after setting MD2, MD1, MD0, P20 and P21 to the flash reprogramming mode.
SIN0	Serial data input pin	Use UART0.
SOT0	Serial data output pin	Use UART0.
	Power supply voltage	3.3-V and 5-V products

### (3) Setting for MB91F120

Pin	Function	Supplementary Explanation
MD2, MD1, MD0	Mode pins	Should be controlled in flash memory reprogramming mode. When MD2 is set to "L" and MD1 and MD0 are set to "H", they enter the reprogramming mode.
P20	Starting pin for programming program	Set P20 to "L" in the flash reprogramming mode.
RSTX	Reset pin	Cancel reset after setting MD2, MD1, MD0 and P20 to the flash reprogramming mode.
SIN0	Serial data input pin	Use UART0.
SOT0	Serial data output pin	Use UART0.
	Power supply voltage	3.3-V and 5-V products

## (4) Setting for MB91F127/F128

Pin	Function	Supplementary Explanation
MD2, MD1, MD0	Mode pins	Should be controlled in flash memory reprogramming mode. When MD2 and MD1 are set to "H" and MD0 is set to "L", they enter the reprogramming mode.
P20, P21	Starting pin for programming program	Set P20 and P21 to "L" (asynchronous UART) in the flash reprogramming mode. (Set P20 to "L" and P21 to "H" in the CLK synchronous communication mode.)
P22, P23	Baud rate setting pins (baud rate of serial asynchronous communication with PC)	Set P22 and P23 to "L" at an original oscillation (input CLK) of 10 MHz. Set P22 to "H" and P23 to "L" at an original oscillation (input CLK) of 12.5 MHz. Set P22 to "L" and P23 to "H" at an original oscillation (input CLK) of 13.5 MHz. Set P22 and P23 to "H" at an original oscillation (input CLK) of 17 MHz.
RSTX	Reset pin	Cancel reset after setting MD2, MD1, MD0, P20, P21, P22 and P23 to the flash reprogramming mode.
SI2	Serial data input pin	Use UART2.
SO2	Serial data output pin	Use UART2.
	Power supply voltage	3.3-V product

## (5) Setting for MB91F133

Pin	Function	Supplementary Explanation
MD2, MD1, MD0	Mode pins	Should be controlled in flash memory reprogramming mode. When MD2 and MD1 are set to "H" and MD0 is set to "L", they enter the reprogramming mode.
P20, P21	Starting pin for programming program	Set P20 and P21 to "L" in the flash reprogramming mode.
RSTX	Reset pin	Cancel reset after setting MD2, MD1, MD0, P20 and P21 to the flash reprogramming mode.
SIN1	Serial data input pin	Use UART1.
SOT1	Serial data output pin	Use UART1.
	Power supply voltage	3.3-V and 5-V products

## (6) Setting for MB91F154/F155

Pin	Function	Supplementary Explanation
MD2, MD1, MD0	Mode pins	Should be controlled in flash memory reprogramming mode. When MD2 and MD1 are set to "H" and MD0 is set to "L", they enter the reprogramming mode.
PG3, PG4, PG5	Starting pin for programming program	Set PG3 and PG4 to "L" and PG5 to "H" in the flash reprogramming mode.
RSTX	Reset pin	Cancel reset after setting MD2, MD1, MD0, PG3, PG4 and PG5 to the flash reprogramming mode.
SIN1	Serial data input pin	Use UART1.
SOT1	Serial data output pin	Use UART1.
	Power supply voltage	3.3-V product

## (7) Setting for MB91F158

Pin	Function	Supplementary Explanation
MD2, MD1, MD0	Mode pins	Should be controlled in flash memory reprogramming mode. When MD2 and MD1 are set to "H" and MD0 is set to "L", they enter the flash reprogramming mode.
PG1, PG2, PG3	Starting pin for programming program	Set PG1 and PG2 to "L" and PG3 to "H" in the flash reprogramming mode.
RSTX	Reset pin	Cancel the reset after setting MD2, MD1, MD0, PG1, PG2 and PG3 to the flash reprogramming mode.
PI0	Serial data input pin	Use UART2.
PI1	Serial data output pin	Use UART2.
	Power supply voltage	3.3-V products

## (8) Setting for MB91F233

Pin	Function	Supplementary Explanation
MD2, MD1, MD0	Mode pins	Should be controlled in flash memory reprogramming mode. When MD2 is set to "H" and MD1 and MD0 are set to "L", they enter the reprogramming mode.
P10, P11	Starting pin for programming program	Set P10 and P11 to "L" in the flash reprogramming mode.
RSTX	Reset pin	Cancel reset after setting MD2, MD1, MD0, P10 and P11 to the flash reprogramming mode.
SIN0	Serial data input pin	Use UART0.
SOT0	Serial data output pin	Use UART0.
	Power supply voltage	3.3-V and 5-V products

## (9) Setting for MB91F264

Pin	Function	Supplementary Explanation
MD2, MD1, MD0	Mode pins	Should be controlled in flash memory reprogramming mode. When MD2 is set to "H" and MD1 and MD0 are set to "L", they enter the reprogramming mode.
P44, P45	Starting pin for programming program	Set P44 and P45 to "L" in the flash reprogramming mode.
RSTX	Reset pin	Cancel reset after setting MD2, MD1, MD0, P44 and P45 to the flash reprogramming mode.
SIN0	Serial data input pin	Use UART0.
SOT0	Serial data output pin	Use UART0.
	Power supply voltage	5-V products

## (10) Setting for MB91F353/F355

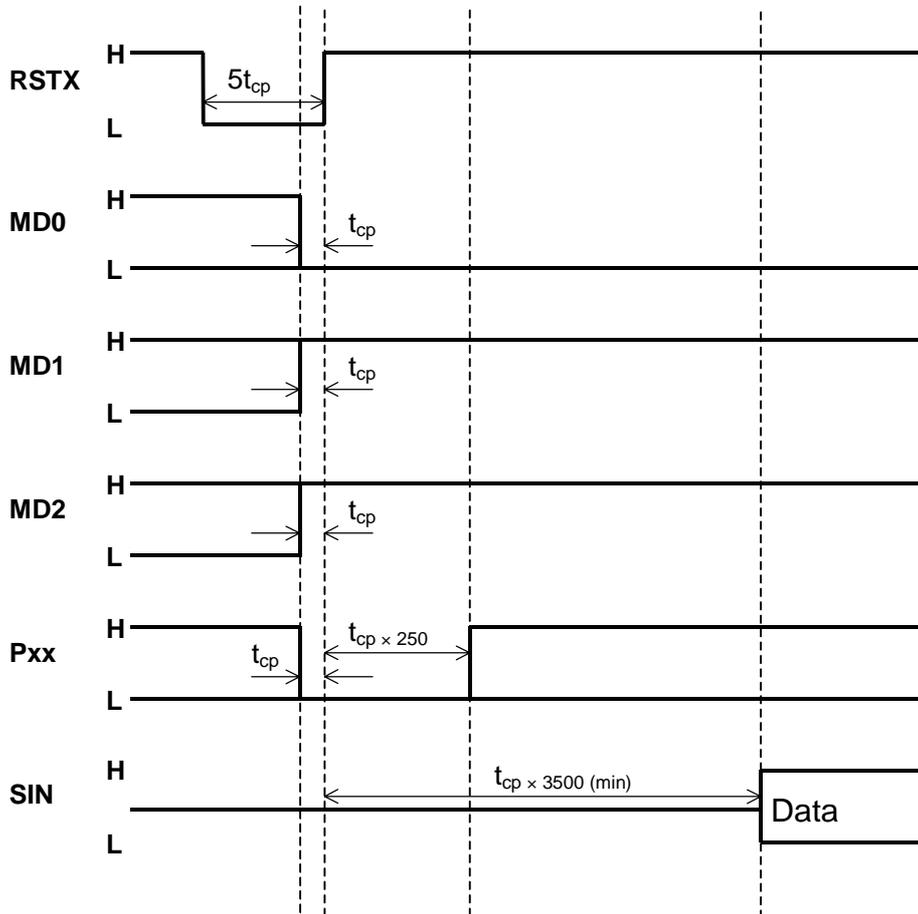
Pin	Function	Supplementary Explanation
MD2, MD1, MD0	Mode pins	Should be controlled in flash memory reprogramming mode. When MD2 is set to "H" and MD1 and MD0 are set to "L", they enter the reprogramming mode.
PN0, PN2	Starting pin for programming program	Set PN0 and PN2 to "L" in the flash reprogramming mode.
RSTX	Reset pin	Cancel reset after setting MD2, MD1, MD0, PN0 and PN2 to the flash reprogramming mode.
SIN3	Serial data input pin	Use UART3.
SOT3	Serial data output pin	Use UART3.
	Power supply voltage	3-V products

## (11) Setting for MB91F362/F365/F366/F367/F368/F369

Pin	Function	Supplementary Explanation
MD2, MD1, MD0	Mode pins	Internal ROM mode. Set MD2, MD1 and MD0 to "L".
	Starting pin for programming program	No setting
INITX	Reset pin	Cancel reset after setting MD2, MD1 and MD0.
SIN0	Serial data input pin	Use UART0.
SOT0	Serial data output pin	Use UART0.
	Power supply voltage	5-V product

### 5. TIMING CHART FOR EACH PIN

Input data to each pin of the microcontroller with the following timing on the basis of the input of the RSTX pin.



Minimum values of setup and hold times of each signal on rising edge of RSTX signal

Although the Pxx signal indicates a starting pin for programming program and the SIN signal a serial data input pin, the pin name varies with the product type of microcontroller.

---

## 6. INSTALLATION AND EXECUTION OF SOFTWARE

If the old software version is installed, uninstall it first before installation.

Starting the installer to operate as instructed will complete the installation. Note that the install might not be performed when a directory in a deep nest is specified as the install directory.

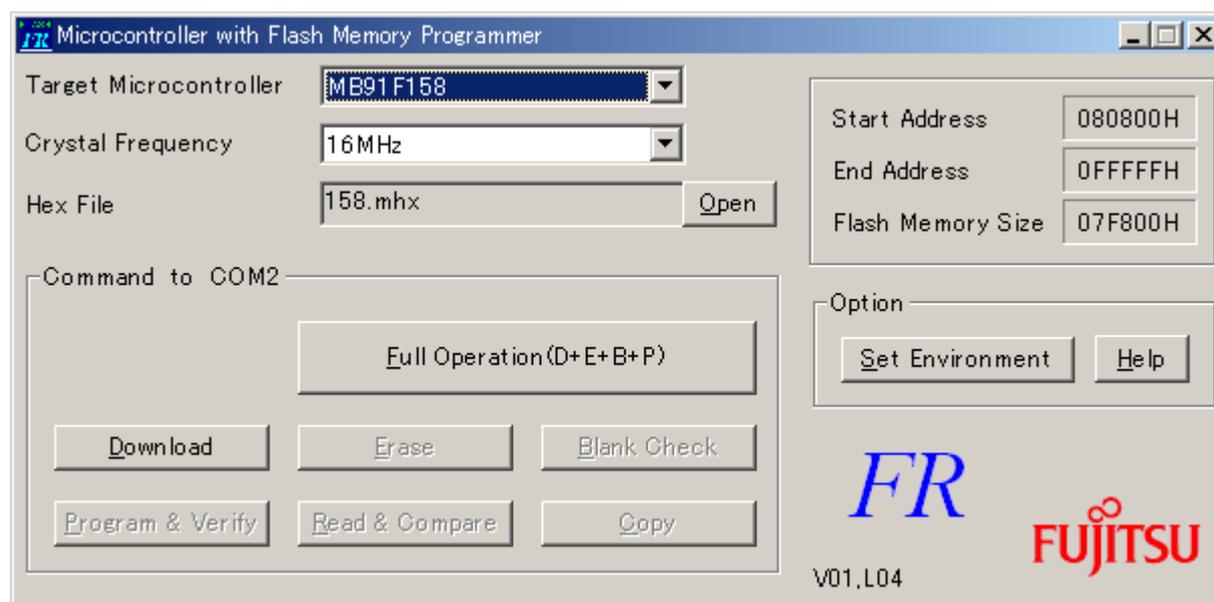
After installation, click the Windows **Start** button => **Program** => **FUJITSU FLASH MCU Programmer** => **FR** to start the programmer software.

## 7. PROGRAMMER FUNCTIONS

Erase, Blank Check, Program & Verify, Read & Compare, and Copy can be executed for flash memory integrated into the microcontroller.

- Main dialog box

Programmer software is started to open the dialog box as shown below.



- Overview of operating procedure

First, complete setting of the user system (microcontroller board) that data is programmed to (see **Chapter 3**). In starting or when setting has been changed, it is necessary to perform downloading (described later).

After downloading terminates normally, perform procedures such as Erase and Programming.

- Product types of microcontrollers supporting security function

The types of microcontrollers that support the security function have slightly different operating procedures from other types.

For the MB91F154/155, see **Chapter 8** first.

For the MB91F360 series, see **Chapter 9** first.

## 7.1 Downloading

This section describes the operating procedure for downloading and the operating state of the program.

- (a) Specify the type of microcontroller used in the user system in **Target Microcontroller** of the main dialog box.

The selectable types are:

MB91F005/A	MB91F109	MB91F120
MB91F127	MB91F128	MB91F133
MB91F154	MB91F155	MB91F158
MB91F233	MB91F264	MB91F353
MB91F355	MB91F362	MB91F365
MB91F366	MB91F367	MB91F368
MB91F369		

Note: To select the type of microcontroller, use the **Tab** key to move to **Target Microcontroller**, select with the cursor keys  $\uparrow$  and  $\downarrow$  and then press the **Enter** key, or click the  $\blacktriangledown$  button on **Target Microcontroller** for dragging.

- (b) Specify the frequency of the crystal oscillator input to the microcontroller in **Crystal Frequency** of the main dialog box.

The frequency of the crystal oscillator that can be specified for each type of microcontroller is limited as follows.

Product Type	Frequency of Crystal Oscillator (MHz)
MB91F127/F128	10, 12.5, 13.5, 17
MB91F133 MB91F158	16
MB91F154/F155	16.5
MB91F233	2, 4, 8, 16
MB91F264	2, 4
MB91F353/F355	12.5
MB91F362/F365/F366 MB91F367/F368/G369	4
Other than the above	12.5, 25

Notice: This program will not operate normally if the microcontroller uses a crystal oscillator frequency not listed in the above table.

(c) Select the COM port of the PC connected to the user system.

Click the **[Set Environment]** button in the main dialog box to open the setup window. When the **[COM PORT]** tab in the setup window is clicked, the specifying window is opened. Select any of the following COM ports.

COM1, COM2, COM3, COM4, COM5, COM6, COM7, COM8

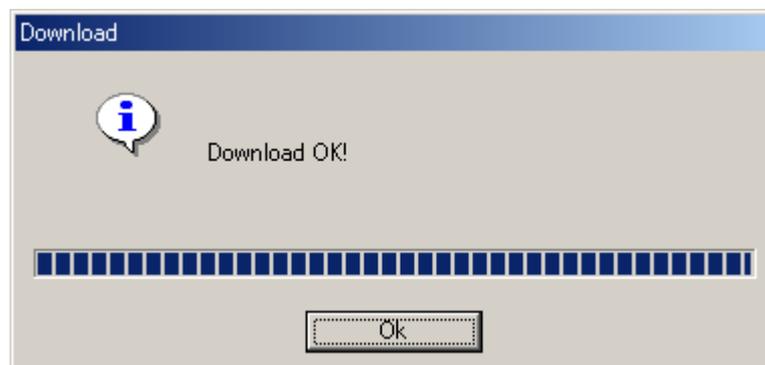
(d) Execution of downloading

Click the **[Download]** button.

If the following dialog window is opened, Input a reset signal to the microcontroller to start the program in the flash programming mode and then click the **[OK]** button



Downloading is performed to open the "Download" window. When downloading is completed normally, the following dialog window opens.



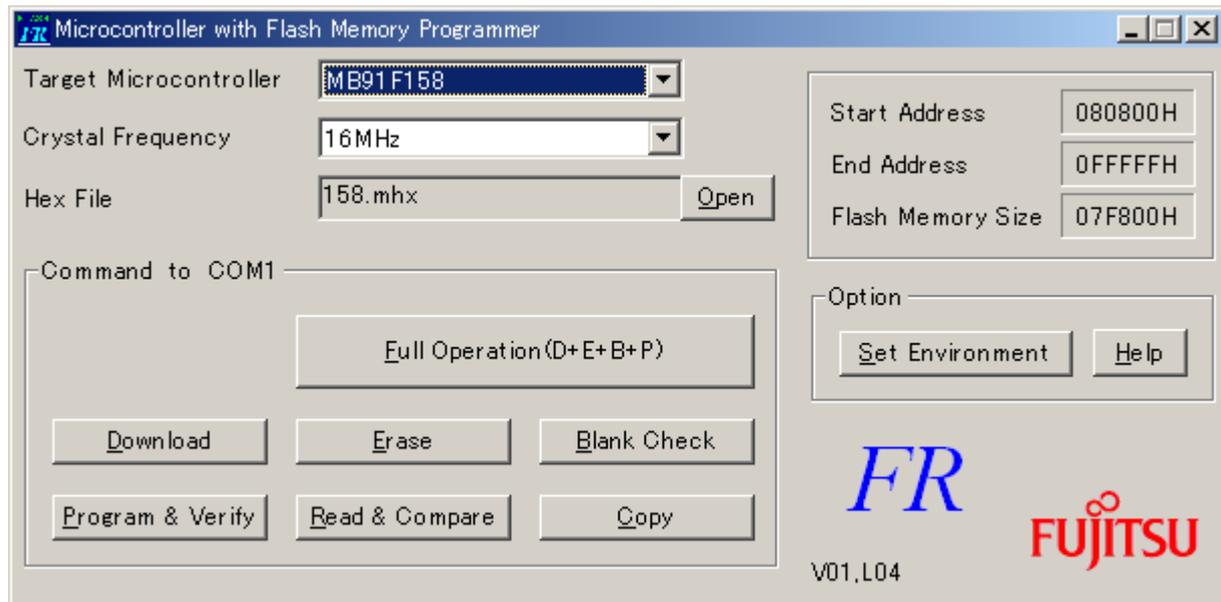
When the **[OK]** button is clicked to close the dialog window, the **[Erase]**, **[Blank Check]**, **[Program & Verify]**, **[Read & Compare]** and **[Copy]** buttons are enabled.

Note: Downloading can also be performed using the **Tab** key to move to the **[Download]** button and pressing the **Enter** key or pressing the **ALT** and **D** keys at the same time.

## 7.2 Erasing and Programming

This section explains how to specify **Hex File** and the processing and operation performed when the **[Erase]**, **[Blank Check]**, **[Program & Verify]**, **[Read & Compare]**, **[Copy]** and **[Full Operation (D+E+B+P)]** buttons are clicked.

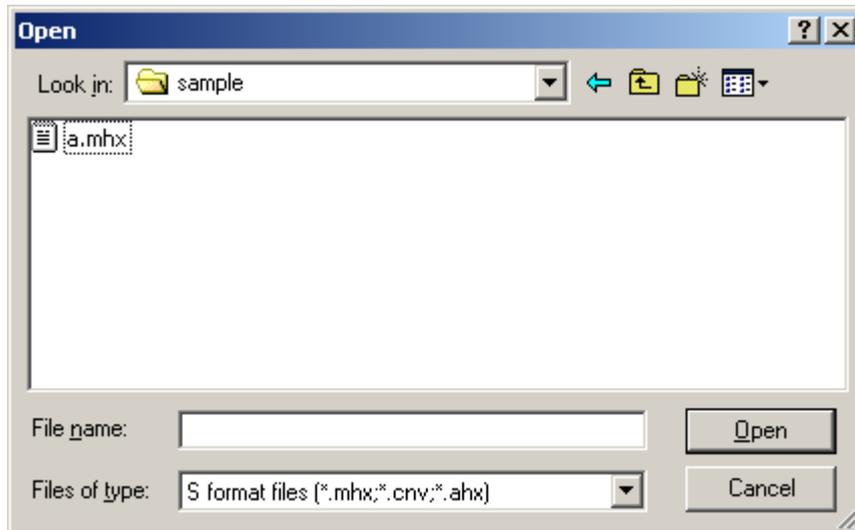
Each execution can also be performed by pressing the key corresponding to the underlined character in the button name while pressing the **ALT** key. (**Hex File** is a **O** character in **Open** button click the **ALT + O** keys).



(a) **Hex File:** Select the file to be programmed to flash memory

Specify the Motorola-S format file to be programmed to flash memory in the microcontroller. Drag and drop the file directly from Internet Explorer or click the **[Open]** button to open the file select window.

**Hex File** must be specified to execute **[Program & Verify]**, **[Read & Compare]** and **[Full Operation (D+E+B+P)]**.



Note: A network drive cannot be specified directly. It should be specified by allocating drive letters.

(b) **Erase:** Erase all flash memory areas

All flash memory must be in the erase state (0xff) when programming a new program to it. Click this button for erasing. The message window displays the message **Erasing** during execution and **Erase OK!** when erasing is completed normally.

(c) **Blank Check:** Check that all flash memory areas are blank

This button is clicked to check that all flash memory is in the erase state (0xff). The message window displays **Blank Checking** during execution and **Blank Check OK!** when execution is completed normally.

(d) **Program & Verify:** Program data to flash memory

This button is clicked to program the Motorola-S format file specified in **Hex File** to flash memory in the microcontroller concurrently with verification. The message window displays **Programming** during execution and **Program OK!** when programming is completed normally. This processing can be suspended by clicking the **[Cancel]** button during execution.

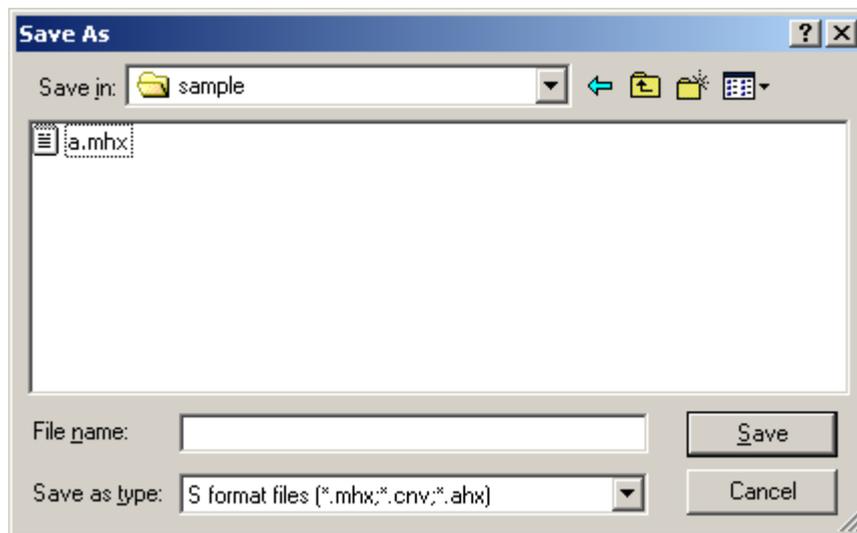
Note: At the start of programming, a temporary file with the same file name and **.BIN** extension is made in the same folder holding the specified Motorola-S format file. Keep in mind overwritten when the file of the same file name already exists. The temporary file can be erased manually at the completion of programming.

- (e) **Read & Compare:** Compare **Hex File** with data in flash memory in microcontroller

This button is clicked to compare data in the Motorola-S format file specified in **Hex File** with data in flash memory in the microcontroller. Like the **[Program & Verify]** processing, a temporary file is made before processing and data in flash memory in the microcontroller is read for comparison. The message window displays **Reading & Comparing** during execution and **Read & Compare OK!** when reading and comparing is completed normally. This processing can be suspended by clicking the **[Cancel]** button during execution.

- (f) **Copy:** Save data in flash memory in microcontroller to file

This button is clicked to read data from flash memory integrated into the microcontroller and save it as an Motorola-S format file. Specify the save destination folder, input a filename, and click the **[Save]** button to start processing. The message window displays **Copying** during execution and **Copy OK!** when execution is completed normally. This processing can be suspended by clicking the **[Cancel]** button during execution.



- (g) **Full Operation (D+E+B+P):** Automatic programming

This button is clicked to automatically perform **[Download]**, **[Erase]**, **[Blank Check]** and **[Program & Verify]** as a batch. The message window provides the following displays sequentially during execution and then **Full Operation OK!** when all are completed normally.

Operation	Display
Make temporary file	Full Operation – Making binary file
Download	Full Operation – Downloading
Erase	Full Operation – Erasing
Blank Check	Full Operation – Blank Checking
Program & Verify	Full Operation – Programming

### 7.3 Continuously writing mode

This mode is used to write the same program to two or more chips. Do not use the mode for purposes other than originally intended. Data written to chips may not be changed even if the write file is changed.

When the mode is enabled, the temporary file for write is not updated, reducing some writing time. If data in the temporary file does not coincide with that in the write file for some reason, no discriminations are made between the data. Consequently, invalid data may be written to chips. For safety writing to the products, proceed as follows:

1. After starting this software, perform FullOperation once in the normal mode and create or update the temporary file.
2. Select ContinuouslyWriteMode at the bottom of the pop-up menu displayed by a right click. This sets the Continuously writing mode to "enabled".
3. Click the button such as FullOperation and write data to two or more chips. In this case, the temporary file is not updated and the same data is written.
4. After the completion of the operation, terminate this program and set the Continuously writing mode to "disabled". The temporary file created in step (1) should be deleted manually.

## 8. SECURITY FUNCTION OF MB91F154/155

### (1) Outline

This type of microcontroller has a security function for protecting programs and data programmed to internal flash memory.

A cipher data string with a maximum length of 255 bytes is programmed to a specific flash memory area to perform verification. An operator who does not know the cipher written to the chip can only issue the **Erase** command to the chip, preventing program modification (corruption) and data reading. Note that the **Erase** command can be executed.

The security function does not work on a blank chip. A blank chip can be processed in the same manner as other chips having no security function.

### (2) Programming cipher data string to chip

The length of the cipher data string is 8 to 255 bytes consisting the "cipher length (1 byte) + cipher data (7 to 254 bytes)". Program the data strings successively to any area in the flash memory area from the Motorola-S format file in the same manner as normal programs. Or program the data strings with programs.

If all cipher data is "0xFF", verification is always successful.

The address range where the entire cipher data string can be programmed varies with the type of microcontroller.

Product Type	Address Range
MB91F154	0xA0000 to 0xA1FFE
MB91F155	0x80800 to 0x827FE

### (3) Accessing chip

To access a chip with set ciphers, it is necessary to make and set a cipher file. See item **(4)** for making and setting a cipher file.

#### a. **Erase**

This is a special command that is not verified. When it is executed, all data programmed to flash memory is erased.

#### b. **Full Operation**

This command is not verified. The old program is replaced by the new one even verification is unsuccessful.

When the chip is blank, verification is always successful and the “**Download, Blank Check, Program & Verify**” commands are executed sequentially.

Except when the blank chip is blank, if verification is successful, the “**Download, Blank Check, Erase, Blank Check, Program & Verify**” commands are executed sequentially.

If verification is unsuccessful, the “**Erase, Download, Blank Check, Program & Verify**” commands are executed sequentially.

#### c. **Download**

If verification is unsuccessful, downloading is not performed.

#### d. Other commands

These commands are not verified. They cannot be executed as long as the execution of the **Full Operation** or **Download** command to be verified is unsuccessful.

#### (4) Making and setting cipher file

This is explained using the example of setting for the MB91F155 chip to which a cipher data string (8 bytes of 07, 01, 02, 03, 04, 05, 06, and 07) is programmed from "0x080800".

Make a new text file, write "080800, 07, 01, 02, 03, 04, 05, 06, 07" to the file, and save the file under a correct name.

The beginning numeric data (for six characters) is the starting address where the cipher data string is programmed.

The following numeric data is a cipher data string which should be delimited with commas. All the numeric values are represented in hexadecimal format. This text file is called a "cipher file".

Find the folder containing the execution program of the PC writer and edit `chipdef.ini` stored in the same directory using Word Pad. Find the [MB91F155] entries.

Find the SecurityFile= string from the entries. Specify the full path of the cipher file directory in the string.

If the directory is `C:\FUJITSU FLASH MCU Programmer\FR\security.txt`, change it to `SecurityFile= C:\FUJITSU FLASH MCU Programmer\FR\security.txt`, and overwrite `chipdef.ini` for saving to complete the setting.

#### (5) Cautions

An verification check is always performed when the **Full Operation** or **Download** command is executed on the MB91F154 and MB91F155 chips. In this case, the cipher file is read and must be set as described in item (4). A dummy cipher file must be set even if verification is not required.

The dummy cipher file is provided under a name `security.txt` in the same folder containing the program.

A valid encrypted file is required for access to the chip to which encrypted data is written. When encrypted data is written to a blank chip or after deleting all data from the chip, the encrypted data part of the encrypted file is not referenced.

Therefore, any value can be set to data in the encrypted data part. It is however necessary to set the **address** and **encryption length** so as to fall properly into the area for each model explained in the above (2).

## 9. SECURITY FUNCTION OF MB91F360 SERIES

### (1) Outline

The MB91F360 series has the security function for protecting data in internal flash memory. When the chip starts from the power-on reset in the internal ROM mode, control jumps to the reset vector. The reset vector is fixed at the starting address of the boot ROM and cannot be changed; the boot program is started automatically.

The boot program references the security vector (#66, 0xFFEF4) to move to the sequence for control of programming to flash memory only when it is "0xFFFFFFFF". If the security vector is not "0xFFFFFFFF", the boot program jumps to the address written to the security vector and cannot control programming to flash memory. Therefore, write to the security vector after determining that data in flash memory is unchanged.

The boot program determines whether to start the boot loader or the user program. If nothing is input within 200 ms after a reset at input to the INITX pin, the boot program starts the user program from the starting address (0xf4000). For details, refer to the *Hardware Manual* for each chip. Therefore, note that the user program should be created to start from this starting address.

If "V" is received via a serial communication pin within 200 ms after input to the INITX pin, the boot loader is started. Since the flash memory programming program uses this boot loader, use the following procedure for programming to flash memory.

### (2) Operating procedure

1. Input a reset signal (INITX) to the chip.

It is important to disable the chip to transmit data serially. If the already-programmed program is started to transmit data serially, continue to input "Low" to the INITX pin while performing steps **2** and **3** and set the INITX pin to "High" at step **4**.

2. Click the **[Download]** or **[Full Operation]** button.
3. Wait until the following dialog box is opened.



4. Input a reset signal (INITX) to the chip.

At step **3**, the PC is ready to transmit “V” at fixed intervals. When a signal is input to the INITX pin, “V” is received within 200 ms after input to the INITX pin to start the boot loader.

5. If the dialog box display changes and the progress bar advances, downloading is successful. If successful, the processing is same as for other types of microcontrollers (**Chapter 7**). If the dialog box display does not change, refer to the following cautions and retry it.

(3) Cautions

1. For a chip with anything other than “0xFFFFFFFF” written to the security vector, flash memory cannot be checked, programmed, and erased by this programmer.
2. Data cannot be written to and erased from the vector area from “0xFFFF8” to “0xFFFF”. This programmer does not display errors caused by erasing, writing and comparing for this area.
3. If there is no reaction after a reset signal is input to the chip in step **4**, cancel and retry after checking whether:
  - (a) The chip has anything other than “0xFFFFFFFF” written to the security vector.
  - (b) The setting of the RS-232C port numbers is correct.
  - (c) The connection of serial communication pins and the setting of MD0, MD1 and MD2 pins are correct.

## 10. STATUS OF OPERATION CHECK

- Specifications for PC used for operation check

PC: FMV 6450TX2  
 CPU: Pentium 450 MHz  
 OS: Japanese and English version of Windows 98 SE, Windows Me,  
 Windows NT4.0 SP6, Windows 2000 SP2, Windows XP  
 Memory: 192 MB

- Example of programming times

MB91F155 (when making temporary file)

Original Oscillation	Time to Program 510 KB of Data "Program&Verify"
16.5 MHz	About 57 s

MB91F154 (when making temporary file)

Original Oscillation	Time to Program 384 KB of Data "Program&Verify"
16.5 MHz	About 43 s

## 11. OTHERS

### (A) Setting of voice output

The setting of voice generated when an error occurs and processing is terminated normally can be changed.

Select the **[Sound]** tab in the setup window that opens when the **[Set Environment]** button is clicked.

- To output sound, put a check in the **Use sound** checkbox.
- Check the status of sound output. Select **ERROR** or **END** in the sound column.
- Select **Wave** or **Beep** as the type of sound to be output in **Sound type**.
- Set the voice file to be output in the **Wave** file column only when **Wave** is selected. When the **[Open]** button is clicked, the File Open window is opened. Select the **Wave** file to be output. The **[Play]** button is used to play the set **Wave** file. The **[Stop]** button is used to stop the **Wave** file.

### (B) Setting of tool hint display

The tool hint display can be “enabled” or “disabled”.

Select the **[Tool]** tab in the setup window that opens when the **[Set Environment]** button is clicked.

When a checkmark is put in the **Display tool hint** checkbox to move the mouse cursor over the contents such as buttons in the dialog window, simple help (the full path of a file for Hex File) is displayed.

## (C) Error messages

No.	Item	Description
No.001	Message	Download error *1
	Cause	Downloading failed
	Action	Return the folder and file configurations to the installation defaults.
No.003	Message	Timeout error
	Cause	The microcontroller does not respond. (Not changed to flash memory reprogramming mode)
	Action	Recheck the setting of pins used for reprogramming flash memory.
No.006	Message	COM port open error
	Cause	The COM port is disabled.
	Action	Enable the COM port.
No.007	Message	Download file open error
	Cause	<code>m_flash.xxx</code> not found
	Action	Return the folder and file configurations to the installation defaults.
No.008	Message	File size get error
	Cause	File access failed
	Action	Check whether the PC is unstable.
No.009	Message	COM port setting information get error
	Cause	The COM port is disabled.
	Action	Enable the COM port.
No.010	Message	COM port setting information change error
	Cause	The COM port is disabled.
	Action	Enable the COM port.
No.011	Message	Communication error
	Cause	The microcontroller returned a communication error.
	Action	Re-execute the command or replace the chip.
No.012	Message	Read error
	Cause	Data cannot be read from flash memory in the microcontroller.
	Action	Re-execute the command or replace the chip.
No.013	Message	Write error
	Cause	Data cannot be programmed to flash memory in the microcontroller.
	Action	Re-execute the command again or replace the chip.
No.015	Message	COM port write error
	Cause	The COM port is disabled.
	Action	Check the RS-232C cable connected to the COM port.

No.	Item	Description
No.016	Message	COM port read error
	Cause	The COM port is disabled.
	Action	Check the RS-232C cable connected to the COM port.
No.017	Message	File access error
	Cause	<code>m_flash.xxx</code> not read
	Action	Return the folder and file configurations to the installation defaults.
No.018	Message	Erase error *1
	Cause	Erasing failed
	Action	Return the folder and file configurations to the installation defaults.
No.101	Message	Set "hex file."
	Cause	"Hex file" not set
	Action	Set "hex file" in the dialog box.
No.102	Message	Batch command error
	Cause	An error occurred at batch command execution.
	Action	Return the folder and file configurations to the installation defaults.
No.103	Message	Invalid "hex file"
	Cause	The selected "hex file" is invalid.
	Action	Select the Motorola-S format file as the "hex file."
No.105	Message	Invalid key length (short)
	Cause	The minimum conditions for key length are not met.
	Action	Prepare a correct security file.
No.106	Message	Invalid key length (long)
	Cause	The maximum conditions for key length are not met.
	Action	Prepare a correct security file.
No.107	Message	Invalid security file
	Cause	The security file description is invalid.
	Action	Prepare a correct security file.
No.207	Message	Memory allocation error
	Cause	Unable to allocate memory for execution
	Action	Quit any running application and retry.
*2	Message	Please redo from download operation.

\*1: "MCU xxH" is displayed if the error cause is returned from the microcontroller at a download error.

"MCU xxH" means:

MCU 02H → SUM error at downloading

MCU 04H → Abnormal termination at downloading

\*2: This is an additional message. It is displayed as necessary after other messages are displayed.

## 12. CAUTIONS

No responsibility is taken about the problem which faced this software use.

The operation of this program is not assured on NEC PC98 series personal computers.

This software is not a thing aiming at the correspondence to mass-production writing.

When using this program, there are restrictions on frequencies that are input to the microcontroller as original oscillations. For details, see **(b)** of **Section 7.1**.