

THE POSSIBILITIES ARE INFINITE

FUJITSU



How to Program Fujitsu's Flash, OTP and Mask ROM Microcontrollers

Overview	1
Programming Possibilities	1
Programming Methods.....	2
Fujitsu Embedded Burn in ROM (Serial programming).....	2
EPROM /Parallel Programming	3
User application software (Bootloader)	4
Flash Field Programming Via CAN interface	4
Mask ROM programming.....	5
General procedure for parallel and serial programming.....	5
Appendix A	7
Appendix B	9
PC Front-end (Utility software) Communication with Kernel	9
Asynchronous Mode	9
Synchronous.....	10
Appendix C	11

How to Program Fujitsu's Flash, OTP and Mask ROM Microcontrollers

► Overview

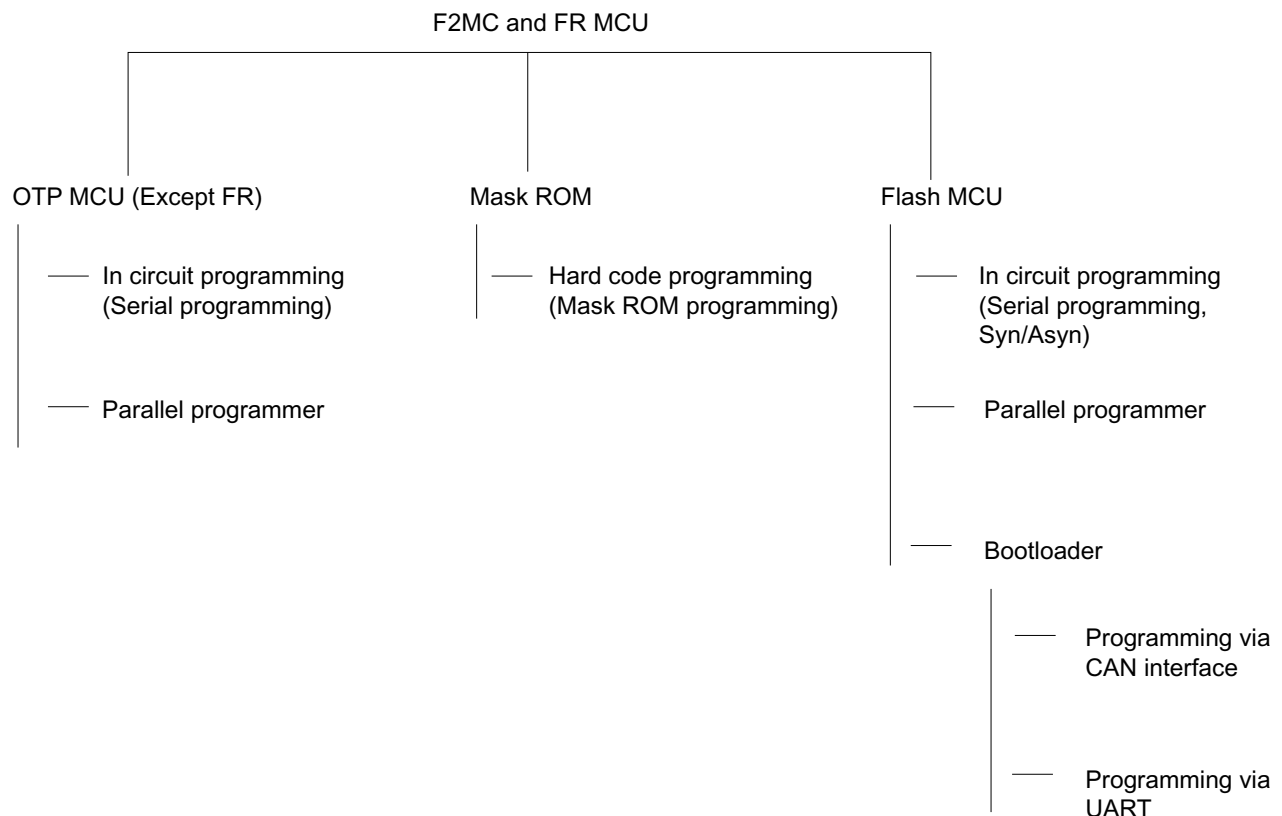
Fujitsu offers impressive range of 8/16/32-bit Microcontrollers. The series includes devices offering memory options such as Flash, mask ROM, and OTP (One-Time-Programmable) memory.

This application note is intended for engineers involved in the development of products using the Fujitsu F2MC and FR series Microcontrollers. It describes the various programming methods and if needed, setting and things to be required on the target board to program the Microcontroller. It also covers the brief information of

serial communication protocol and talks about the typical steps required for the programming.

Programming Possibilities

The chart below shows the different possible programming methods for each type of MCU.



How to Program Fujitsu's Flash, OTP and Mask ROM Micro-

Programming Methods

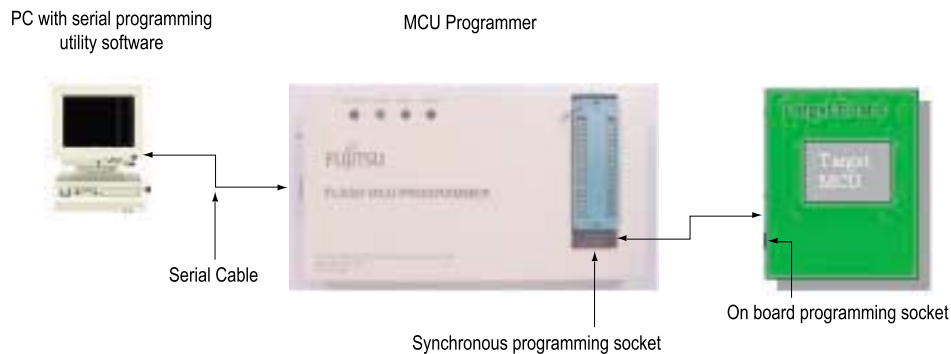
Fujitsu Embedded Burn in ROM (Serial programming)

Serial programming can be used only for the Flash Microcontroller (Except 90F523PFV and 90F594PF) and OTP part (Currently MB89P935A, MB89P580, MB89F969A 8-bit MCU) whose internal memory is based on Flash memory technology. Flash MCU contains a burn in ROM (In ROM bootloader) program (Hidden to the user) that supports a proprietary protocol to allow user to download program in to the on chip flash memory. A special mode pin setting of the MCU enables the hidden burn in ROM program. On the PC side, special software is used that communicates to the MCU via UART. Please refer to the application note "BI-ROM serial programming" for more detail information.

Serial programming could be done in two ways:

1. Synchronous Serial Programming
2. Asynchronous Serial Programming

Synchronous Serial Programming: Synchronous serial Programming can be performed by using special MCU Programmer, which supports synchronous programming. The serial programmer converts asyn data from PC into the syn data. By setting the mode pins on the target board to the syn configuration, MCU will be ready for the syn mode of programming. The target board HW should support the serial protocol. The code download speed is about 500kbps. The typical Hardware set up for synchronous serial programming is as shown in Figure 1.



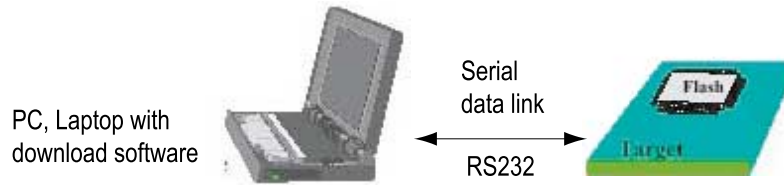
Note: Please see the appendix A for the device supported by Fujitsu synchronous MCU programmer called "Fujitsu Flash Kit", and the required setting of the UART interface and other port pins setting on the target board. No RS232 interface is required on the target board for this method of programming. Only a header pin out compatible with the SYN programmer should be provided on the target board for the communication between MCU and programmer. Also see Appendix B to understand the communication protocol between kernel software (A program which has capability to program flash. This should be downloaded first into the MCU RAM with the help of BI-ROM program and PC front-end software.

Figure 1. Hardware setup for synchronous serial programming

Application Note

Asynchronous Serial Programming: In the asynchronous programming mode, the MCU is able to communicate with PC through standard RS232 interface. So, if the RS232 line interface is provided on the user target board and the target board MCU supports the asynchronous programming mode, no additional HW is needed for programming the MCU- the only thing you have to

do is to connect your board to the PC through serial cable. The only disadvantage of this mode is the limited programming speed. The maximum communication speed is 19200bps. Please also refer to the application note “Flash programming” for detail information. The typical setup for Asynchronous serial programming is shown in Figure 2.



Note: To program the MCU it is necessary to change the operating mode of the Microcontroller to the asynchronous serial-programming mode. Therefore the mode pins MD0, MD1, MD2 (MD0 and MD1 only, in case of 8bit OTP MCU) and the port pins are used. Refer appendix A for the required setting of UART interface and Microcontroller pin number used by the asynchronous software. Also refer appendix B to understand more detail about ASYN serial communication protocol.

Figure 2. Hardware setup for asynchronous serial programming

EPROM /Parallel Programming

In Parallel programming mode, device is programmed through the EPROM programmer via 40/48 pins ZIF DIP socket. Programming speed is even faster than synchronous mode. Both F2MC and FR family Flash/OTP devices are supported by this

mode. However, suitable adapter has to be ordered separately for a particular device. The hardware set up for this is shown in Figure 3.

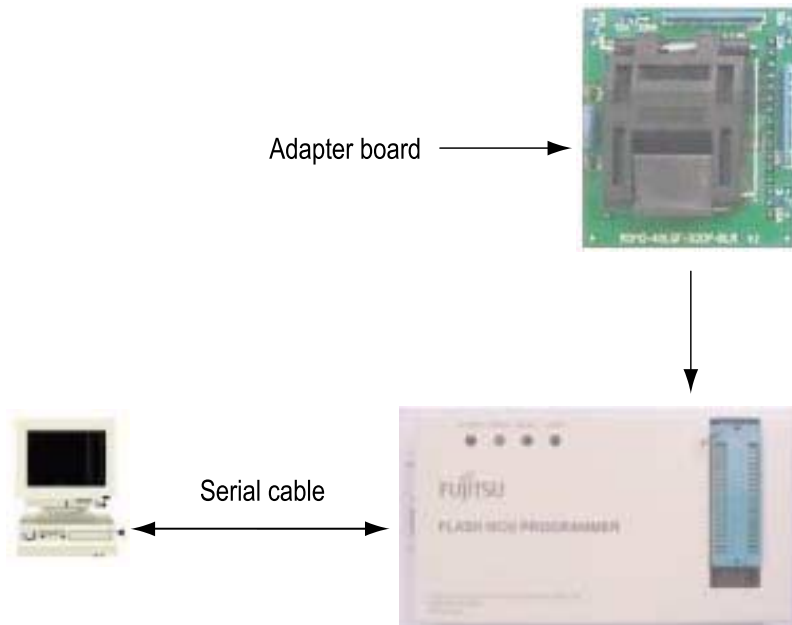


Figure 3. Hardware setup for parallel programming

How to Program Fujitsu's Flash, OTP and Mask ROM Micro-

Currently Fujitsu Microcontrollers are supported by following manufacturers of EPROM programmer:

- Minato
- Hi-lo system
- Data I/O
- Conitec Datensysteme
- BP Microsystems
- Stag
- EEtools, Inc

User application software (Bootloader)

In this method user application software called “Bootloader” is used to download the program in to the Flash memory. The bootloader can be programmed in to the Flash memory via serial or parallel programming method mentioned above. This bootloader resides on the Flash at hardwired reset vector location and can be started by special command via RS232 or CAN, USB or any other interface. After receiving the command the bootloader program gets copied in to the RAM and software runs itself in RAM. After that user application software can be loaded via CAN, serial or any other general-purpose interface (GPIO).

At the front end of the PC a utility such as SK WIZARD can be used to send command and data through serial interface.

Flash Field Programming Via CAN interface

This programming method is suitable for the device, which has CAN and Flash memory. This method is most suitable for field programming where MCU is assembled on the board and there is no serial interface for using the serial programming.

This method is based on master and slave configuration. The hardware setup for using the CAN interface for programming the flash is shown in Figure 4. There will be one master board and several slave boards connected on the CAN bus. Master board has capability to communicate serially through PC. Master board has Master bootloader running on its MCU, similarly Slave board has slave bootloader running on its MCU. The bootloader should be downloaded into the MCU either by parallel or serial programming method. Each slave board has one unique node ID. To update the software on particular node, connect the master board to the PC via serial interface and start the PC utility software. After connection has been established between master board and utility software, it will ask for which node you want to program and after this it asks for the file to be load on to the MCU. Now the software will get updated on to the slave node. For more detail please refer to the application Note “Flash programming via CAN”.

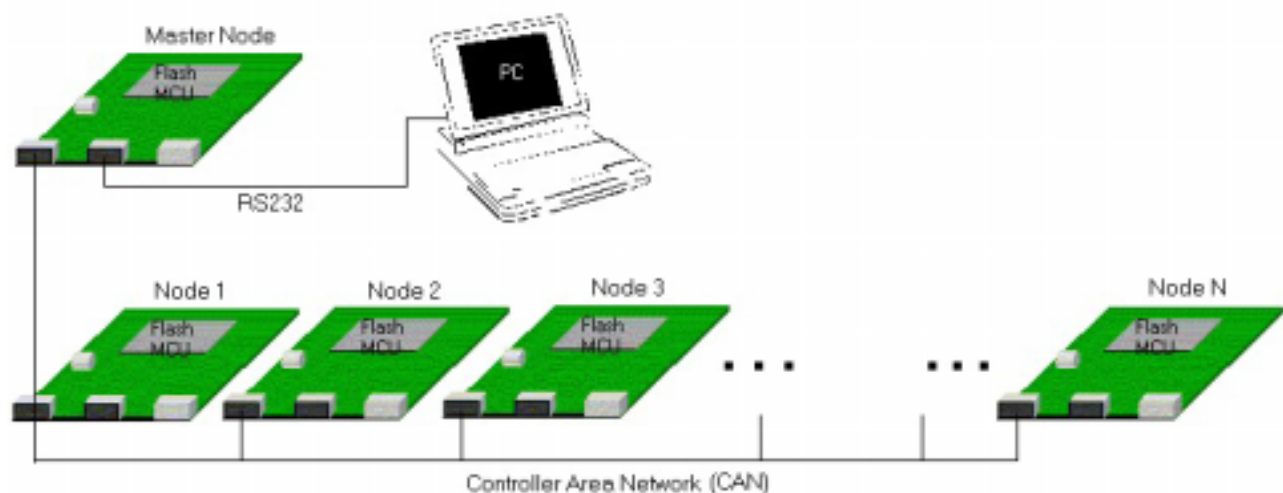


Figure 4. Hardware setup for programming the flash using CAN interface

Mask ROM programming

This programming is suitable only for the Mask ROM part. Generally customer who anticipate using 50,000 or more of a particular Fujitsu Microelectronics MCU are welcome to develop a mask ROM version of the chosen MCU, if one is available. An NRE is typically involved for the mask ROM procedure. The customer can order varying sizes of ES engineering samples. The customer can also do a RISK order to speed up delivery of product.

In this method user will build, test and debug the software prior to placing order for the Mask ROM part. When the software is ready user will place the order along with the hex code (exe file) of his program.

The steps below illustrate the process flow.

WEEK 1: Customer fills out Data Release Form, and submits code. Official part number is created. Customer reviews returned code, and signs Final Validation document. Customer issues Purchase Orders for NRE, ES build, and (if desired) volume production of Risk build

WEEK 5: ES build is delivered to customer. Customer verifies that the device works properly, and issues new PO for special Expedited Volume Production of verified device

WEEK 12: Volume production of Risk build is delivered to customer, 12 weeks after PO.

WEEK 13-15: Expedited Volume Production of Verified device is delivered, 8-10 weeks after PO.

AFTER WEEK 15: Volume production of verified device can be ordered at standard factory lead-time of 12 weeks.

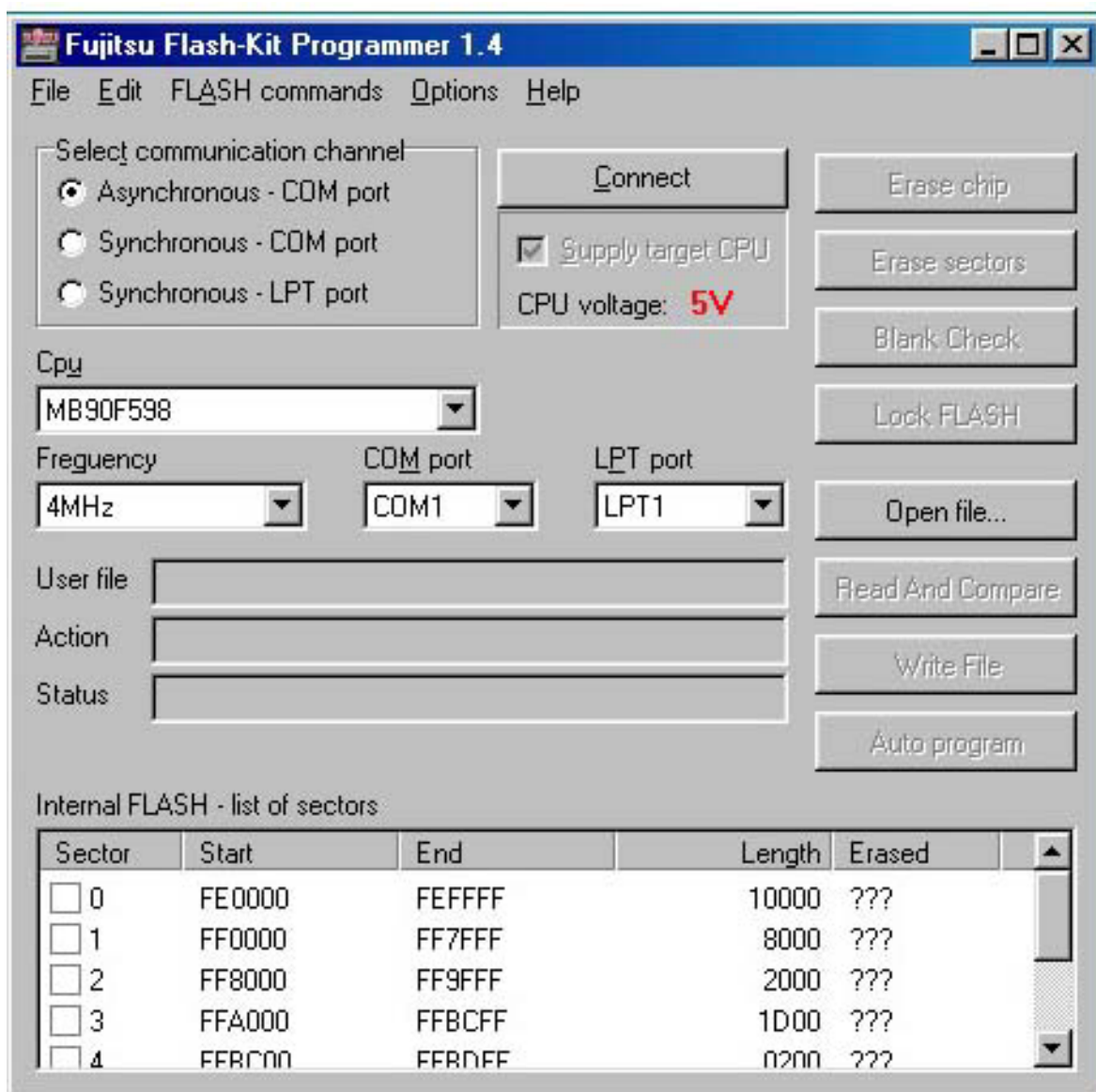
General procedure for parallel and serial programming:

- Setup the hardware suitable for the programming method used as described previously.
- Build the project file on softune workbench, this will make the executable file either in bin or hex format depends on the converter setting chosen on the softune workbench. (Softune workbench generates output data format either in Motorola or Intel hex format. You can find the exe file in the “abs” directory of the project file.)

- Download the PC side utility software (from Fujitsu or from programmer manufacturer) . Start the software. In case of serial programming first download the kernel program by pressing the download or connect button on the utility software. Choose the exe (.mhx or bin) file of the application program you want to load in to the MCU.
- Now either use the single commands such as <Erase>, <Blank check>, <write and verify> to perform one operation at a time or you can perform all these function in a single step by clicking the ‘Auto program ‘button.

A typical screen capture of the PC side utility software is shown in Figure 5.

How to Program Fujitsu's Flash, OTP and Mask ROM Micro-



Note: Some utility provided by Fujitsu only supports the async serial programming.

Figure 5. Screen capture of the Flash Programming Utility Software

Appendix A

In the following table, you can find the list of all the supported MCUs from the F2MC and FR family and the pin setting for synchronous and asynchronous programming:

F2MC-8L Family

CPU Type	UART used	SI Pin No.	SO Pin No.	SCK*** Pin No.	Mode Pins used	Mode Pin No.	Logic Level at Mode pins	Port Pins used	Port Pin No.	Logic Level at port pins
MB89P935	UART0	17	18	19	MOD 1,0	6,5	1,1	P37	11	0*,1**
								P40	22	1
								P41	23	1
								P42	24	0
								P43	25	1

Note: * Asynchronous programming

** Synchronous programming

*** SCK signal is used only in case of synchronous programming. No clock signal is used for asynchronous programming

F2MC-16LX Family

CPU Type	UART used	SIN Pin No.	SOT Pin No.	SCK*** Pin No.	Mode pins MD 2,1,0		Port pins P00,P01			
					Pin No	Logic level	P00 Pin No	P00 Logic level	P01Pin No.	P01 Logic level
MB90F428PF	UART1	88	89	90	51,50,49	1,1,0	85	0	86	0*,1**
MB90F497PFM	UART1	60	62	61	21,20,18	1,1,0	25	0	26	0*,1**
MB90F523PFV*	UART1			13	87,88,89	1,1,0	95	0	96	0,1**
MB90F543PF	UART1	21	24	22	51,50,49	1,1,0	85	0	86	0*,1**
MB90F548PF	UART1	21	24	22	51,50,49	1,1,0	85	0	86	0*,1**
MB90F549PF	UART1	21	24	22	51,50,49	1,1,0	85	0	86	0*,1**
MB90F553PF	UART0	20	19	18	51,50,49	1,1,0	85	0	86	0*,1**
MB90F562PFM	UART1	14	15	60	21,20,18	1,1,0	25	0	26	0*,1**
MB90F568PFM	UART1	14	15	60	21,20,18	1,1,0	25	0	26	0*,1**
MB90F574PFV	UART0	9	10	11	87,88,89	1,1,0	95	0	96	0*,1**
MB90F583PF	UART0	18	19	20	51,50,49	1,1,0	85	0	86	0*,1**
MB90F591PF	UART0	16	14	25	51,50,49	1,1,0	85	0	86	0*,1**
MB90F594APF	UART0	16	14	25	51,50,49	1,1,0	85	0	86	0*,1**
MB90F598PF	UART1	21	24	22	51,50,49	1,1,0	85	0	86	0*,1**

Note: * Asynchronous programming

** Synchronous programming

*** SCK signal is used only in case of synchronous programming. No clock signal is used for asynchronous programming

How to Program Fujitsu's Flash, OTP and Mask ROM Micro-

FR Family

CPU Type	UART used	SI pin	SO pin	SCK-pin***	Mode pins MD2,1,0		Port pins P20,P21				Other pins Used		Comment
					Pin no.	Logic level	P20 Pin No.	Logic level	P21 Pin no.	Logic level	Pin no	Logic level	
MB91F109	UART0	79	80	81	21,20,19	1,1,0	28	0	29	0*,1**	-	-	
MB91F361	UART0	152	153	-	113,112,111	0,0,0	-	-	-	-	46 (P93)	1	Syn not supported

Note: * Asynchronous programming

** Synchronous programming

*** SCK signal is used only in case of synchronous programming. No clock signal is used for asynchronous programming

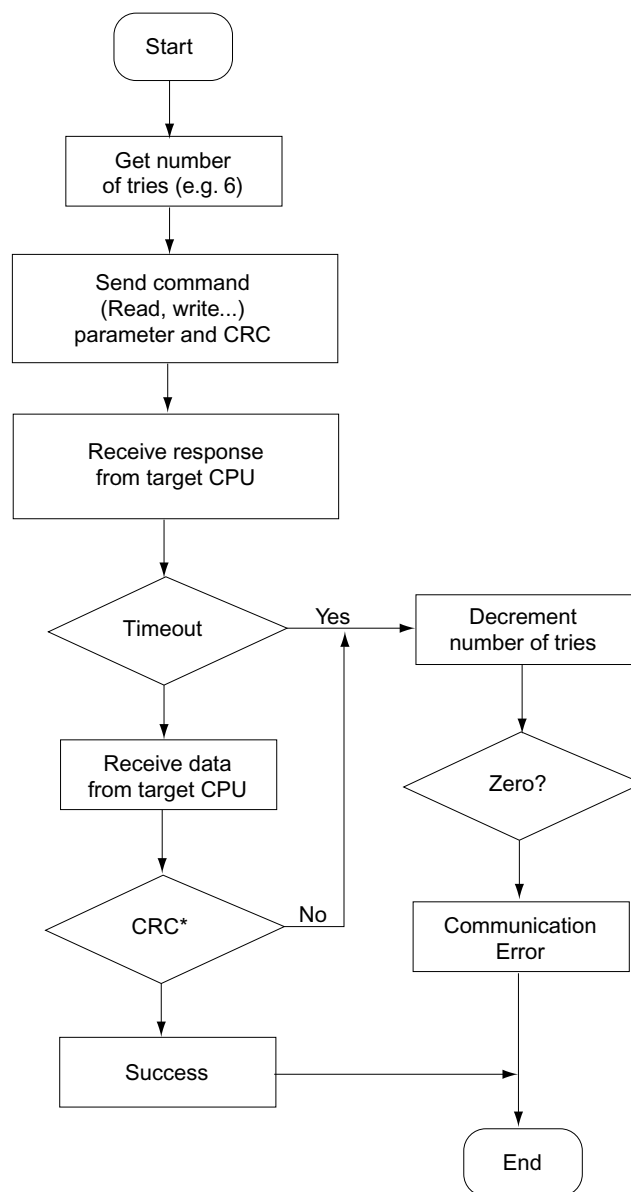
Appendix B

PC Front-end (Utility software) Communication with Kernel

The flowchart below shows the communication between Utility software and the kernel program (program download to the CPU RAM and has capability to program Flash) to download the actual target application program via serial communication method.

Asynchronous Mode

If the asynchronous communication is chosen in the serial programmer utility software, the communication protocol looks like the one shown in the flow chart below.

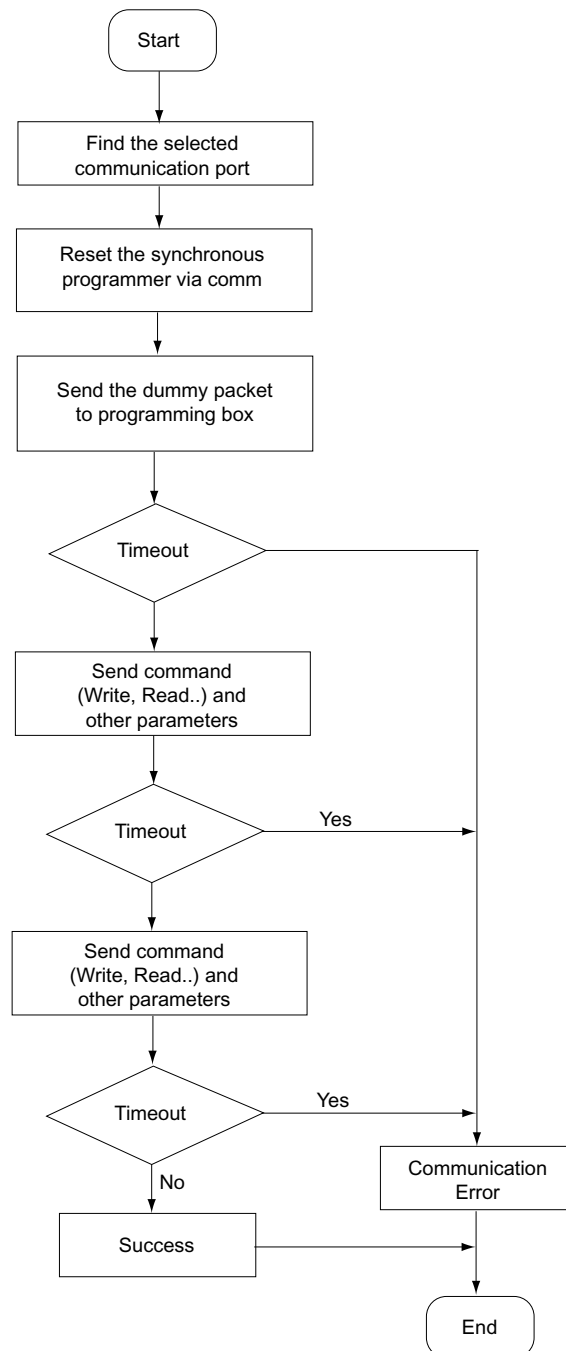


Note: * In some utility CRC checksum algorithm is not implemented so only the timeout is the decision making factor.

How to Program Fujitsu's Flash, OTP and Mask ROM Micro-

Synchronous

When the Synchronous Comm is selected in the utility software, the situation is little bit complicated, because there is a programmer box in between the PC and target CPU. The Flowchart below shows the actual communication.



Appendix C

List of Third Party Parallel Programmers

Manufacturer Name	Recommended EPROM Programmer	Contact Details
Minato Electronics Inc	Model 1890A , Model 1893, Model 1930, Model 1931	224-0026 4105 Minamiyamada-cho, Tsuzuki-ku, Yokohama, Japan Tel:+81 45-592-5549
Ando Electric Co. Ltd.	AF9704, AF9705, AF9708, AF9709, AF9723	2265 Wetwood Blvd #588, Los Angeles, CA 90064 Tel: (310)204-5141 Fax: (310)815-9773
Data I/O Corporation	OPTIMA, Gang programmer OCTAL and QUAD	10525 Willows Road N.E. P.O. Box 97046 Redmond WA 98073-9746 Tel: (425) 881-6444
Hi-LO system	ALL-11	Tribal Microsystems, Inc. 44388 S. Grimmer Blvd Fremont, CA 94538-6385 Tel: +1(510) 623 8859
Electronic Engineering Tools, Inc.	TopMax Universal programmer	550 Weddell Drive, Suite 5 Sunnyvale, CA 94089 Tel: (408) 734-8184 www.eetools.com
BP Microsystems	BP-1400, BP-1600 BP-2000 , BP-2100, BP-2200, BP-2500, BP-2510, BP-2600, BP-3100, BP-3500, BP-3510, BP-3600, BP-4100, BP-4500, BP-4510, BP-4600	1000 N Post Oak Rd. Suite 225 Houston, TX 77055-7237 USA Tel: +1(713) 688-4600 www.bpmicro.com
STAG Programmers Ltd.	Eclipse , P-803 , P-801	Barry Technical Sales Inc. 632 S. George Street Mt. Prospect, IL 60056 Tel: 847-870-9705
Conitec Daten system		Conitec Corporation 1951 4th Ave, Ste 301 San Diego, CA 92101 Tel +1 (619) 702-4420

FUJITSU MICROELECTRONICS AMERICA, INC.

Corporate Headquarters

1250 East Arques Avenue, Sunnyvale, California 94088-3470

Tel: (800) 866-8608 Fax: (408) 737-5999 E-Mail: inquiry@fma.fujitsu.com Internet: <http://www.fma.fujitsu.com>

© 2001 Fujitsu Microelectronics, Inc.;
All rights reserved. All company and
product names are trademarks or
registered trademarks of their
respective owners.

EC-AN-20873-07/2001