

**F<sup>2</sup>MC-8FX FAMILY**  
8-BIT MICROCONTROLLER  
**MB95200 SERIES**

---

**STARTER KIT MB2146-410 -01-E**  
**MB2146-420 -01-E**

SETUP GUIDE



## PREFACE

Thank you for purchasing the F2MC\*<sup>1</sup>-8FX Family Starter Kit: MB2146-410-01-E / MB2146-420-01-E \*<sup>2</sup>.

This product is a starter kit for F2MC-8FX MB95200 Series MCU, which comes with MB2146-08-E (F2MC-8FX Family MB95200 Series BGM Adapter)\*<sup>3</sup>, MB2146-410-E / MB2146-420-E (F2MC-8FX Family MB95200 Series Evaluation Board)\*<sup>4</sup> and F2MC-8L/8FX Family SOFTUNE Professional Pack Evaluation Version\*<sup>5</sup>.

This manual explains how to use the Starter Kit. Be sure to read this manual before using the product.

For mass production/evaluation MCUs for this product, consult with sales representatives or support representatives.

\*1: F2MC is the abbreviation of FUJITSU Flexible Microcontroller.

\*2: Referred below as the “Starter Kit”.

MB2146-410-01-E is the Starter Kit for MB95200H/210H MCU;

MB2146-420-01-E is the Starter Kit for MB95260H MCU.

\*3: Referred below as the “BGMA”.

\*4: Referred below as the “EV-Board”.

MB2146-410-E is the EV-Board for MB95200H/210H MCU;

MB2146-420-E is the EV-Board for MB95260H MCU.

\*5: Referred below as the “SOFTUNE”.

### ■ Handling and use

Handling and use of this product and notes regarding its safe use are described in the manuals for products bundled with the Starter Kit.

Follow the instructions in the manuals to use this product.

Keep this manual at hand so that you can refer to it anytime during use of this product.

### ■ Notice on this document

All information included in this document is current as of the date it is issued.

Such information is subject to change without any prior notice.

Please confirm the latest relevant information with the sales representatives.

■ **Caution of the products described in this document**

The following precautions apply to the product described in this manual.



Indicates a potentially hazardous situation which, if not avoided appropriately, could result in death or serious injury and/or a fault in the user's system.

<b>Electric shock, Damage</b>	Before performing any operation described in this manual, turn off all the power supplies to the system. Performing such an operation with the power on may cause an electric shock or device fault.
<b>Electric shock, Damage</b>	Once the product has been turned on, do not touch any metal part of it. Doing so may cause an electric shock or device fault.



Indicates the presence of a hazard that may cause a minor or moderate injury, damages to this product or devices connected to it, or may cause to loose software resources and other properties such as data, if the device is not used appropriately.

<b>Cuts, Damage</b>	Before moving the product, be sure to turn off all the power supplies and unplug the cables. Watch your step when carrying the product. Do not use the product in an unstable location such as a place exposed to strong vibration or a sloping surface. Doing so may cause the product to fall, resulting in an injury or fault.
<b>Cuts</b>	The product contains sharp edges that are left unavoidably exposed, such as jumper plugs. Handle the product with due care not to get injured with such pointed parts.
<b>Damage</b>	Do not place anything on the product or expose the product to physical shocks. Do not carry the product after the power has been turned on. Doing so may cause a malfunction due to overloading or shock.
<b>Damage</b>	Since the product contains many electronic components, keep it away from direct sunlight, high temperature, and high humidity to prevent condensation. Do not use or store the product where it is exposed to much dust or a strong magnetic or electric field for an extended period of time. Inappropriate operating or storage environments may cause a fault.
<b>Damage</b>	Use the product within the ranges given in the specifications. Operation over the specified ranges may cause a fault.
<b>Damage</b>	To prevent electrostatic breakdown, do not let your finger or other object come into contact with the metal parts of any of the connectors. Before handling the product, touch a metal object (such as a door knob) to discharge and any static electricity from your body.
<b>Damage</b>	Before turning the power on, in particular, be sure to finish making all the required connections. Furthermore, be sure to configure and use the product by following the instructions given in this document. Using the product incorrectly or inappropriately may cause a fault.
<b>Damage</b>	Always turn the power off before connecting or disconnecting any cables from the product. When unplugging a cable, unplug the cable by holding the connector part without pulling on the cable itself. Pulling the cable itself or bending it may expose or disconnect the cable core, resulting in a fault.
<b>Damage</b>	Because the structure of the MCU socket does not allow an evaluation MCU to be mounted in the incorrect orientation, be very careful of the orientation of the evaluation MCU when mounting it. Inserting the evaluation MCU in the wrong orientation may damage the MCU, causing the MCU to become faulty.
<b>Damage</b>	Because the product has no casing, it is recommended that it be stored in the original packaging. Transporting the product may cause a damage or fault. Therefore, keep the packaging materials and use them in case of for the re-shipment of the product.

- The contents of this document are subject to change without notice. Customers are advised to consult with sales representatives before ordering.
- The information, such as descriptions of function and application circuit examples, in this document are presented solely for the purpose of reference to show examples of operations and uses of FUJITSU MICROELECTRONICS semiconductor device; FUJITSU MICROELECTRONICS does not warrant proper operation of the device with respect to use based on such information. When you develop equipment incorporating the device based on such information, you must assume any responsibility arising out of such use of the information. FUJITSU MICROELECTRONICS assumes no liability for any damages whatsoever arising out of the use of the information.
- Any information in this document, including descriptions of function and schematic diagrams, shall not be construed as license of the use or exercise of any intellectual property right, such as patent right or copyright, or any other right of FUJITSU MICROELECTRONICS or any third party or does FUJITSU MICROELECTRONICS warrant non-infringement of any third-party's intellectual property right or other right by using such information. FUJITSU MICROELECTRONICS assumes no liability for any infringement of the intellectual property rights or other rights of third parties which would result from the use of information contained herein.
- The products described in this document are designed, developed and manufactured as contemplated for general use, including without limitation, ordinary industrial use, general office use, personal use, and household use, but are not designed, developed and manufactured as contemplated (1) for use accompanying fatal risks or dangers that, unless extremely high safety is secured, could have a serious effect to the public, and could lead directly to death, personal injury, severe physical damage or other loss (i.e., nuclear reaction control in nuclear facility, aircraft flight control, air traffic control, mass transport control, medical life support system, missile launch control in weapon system), or (2) for use requiring extremely high reliability (i.e., submersible repeater and artificial satellite). Please note that FUJITSU MICROELECTRONICS will not be liable against you and/or any third party for any claims or damages arising in connection with above-mentioned uses of the products.
- Any semiconductor devices have an inherent chance of failure. You must protect against injury, damage or loss from such failures by incorporating safety design measures into your facility and equipment such as redundancy, fire protection, and prevention of over-current levels and other abnormal operating conditions.
- Exportation/release of any products described in this document may require necessary procedures in accordance with the regulations of the Foreign Exchange and Foreign Trade Control Law of Japan and/or US export control laws.
- The company names and brand names herein are the trademarks or registered trademarks of their respective owners.

Copyright © 2009 FUJITSU MICROELECTRONICS LIMITED. All rights reserved

# Contents

<b>PREFACE</b> .....	<b>i</b>
■ Handling and use.....	i
■ Notice on this document.....	i
■ Caution of the products described in this document.....	ii
<b>Contents</b> .....	<b>iv</b>
<b>1 PRODUCT OVERVIEW</b> .....	<b>1</b>
1.1 Objective and Deliverable.....	1
1.2 System Block .....	1
1.3 Handling Precautions .....	1
1.4 Feature.....	2
1.5 Hardware Setup .....	2
<b>2 BGMA MANUAL</b> .....	<b>3</b>
2.1 BGMA Overview.....	3
2.2 Function List.....	3
2.3 IDC10 Interface Description .....	4
2.4 BGMA USB Configuration .....	4
2.5 LED Description .....	7
<b>3 EV-BOARD MANUAL</b> .....	<b>8</b>
3.1 EV-board Overview .....	8
3.2 Function List.....	9
3.3 EV-board Schematic .....	10
3.4 HW Module Description and Jumper settings.....	11
3.4.1 Power Module.....	11
3.4.2 BGMA Interface .....	13
3.4.3 Clock Settings.....	13
3.4.4 Buzzer Module.....	14
3.4.5 A/D Module.....	14
3.4.6 LED Module.....	14
3.4.7 Key Module.....	15
3.4.8 Lin-UART Module .....	15
<b>4 SAMPLE CODE MANUAL</b> .....	<b>16</b>
4.1 Topic List.....	16
4.2 Project Structure.....	18
4.3 Source Code File Description .....	19
4.3.1 Header Files .....	19

4.3.2	Startup.asm File.....	20
4.3.3	Vectors.c File.....	20
4.3.4	Main.c File .....	21
<b>5</b>	<b>DEVELOPMENT PLATFORM QUICK START .....</b>	<b>22</b>
5.1	Tools Setup Sequence .....	22
5.2	Open Project and Start Debug.....	22
5.3	Operation Precautions.....	22





# 1 Product Overview

This product is a set of Starter Kit of MB95200 series. It is composed of a BGMA (MB2146-08-E) and an EV-board (MB2146-410-E or MB2146-420-E). Combining the SOFTUNE Workbench on PC, the Starter kit enables the quick start of development before the user system is ready.

## 1.1 Objective and Deliverable

The Starter kit provides users a complete development platform. Before start using the Starter Kit, make sure that the following devices are placed in the package:

- ✓ BGMA (MB2146-08-E) 1PCS;
- ✓ EV-board (MB2146-410-E or MB2146-420-E) 1PCS;
- ✓ USB cable 1PCS
- ✓ CD-ROM (SOFTUNE, user manuals, sample code) 1PCS
- ✓ Hardcopy (China RoHS report, quick start guide) 2PCS

## 1.2 System Block

To setup a debugging system, connect a PC, a BGMA and an EV-board together as shown below:

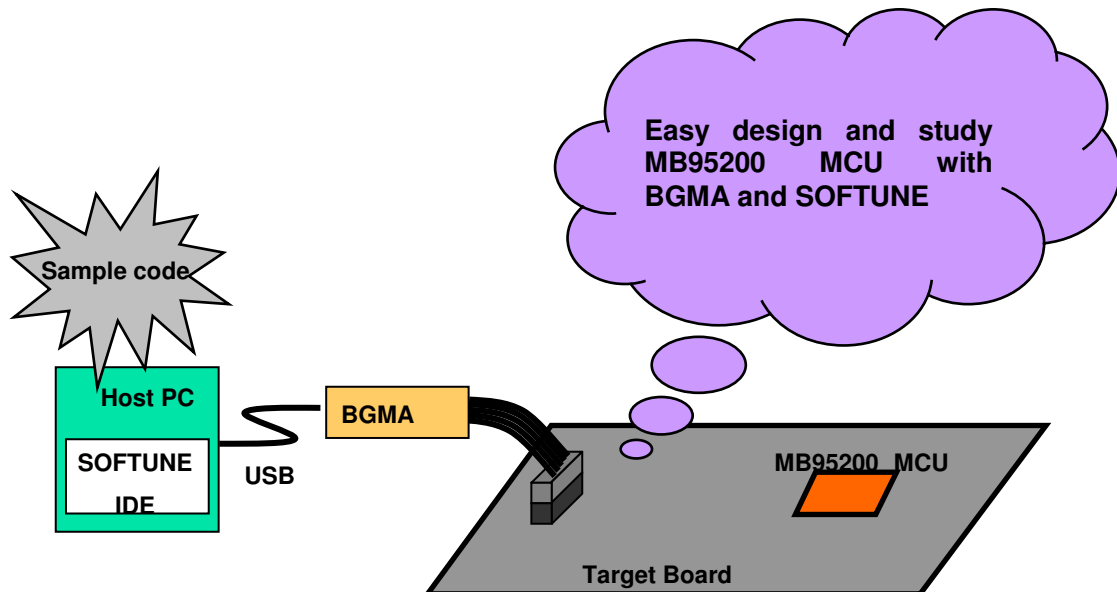


Figure 1.2-1 System Block

## 1.3 Handling Precautions

The Starter Kit can be used in connection with its bundled products. To ensure correct use of this product in a proper environment, observe the following guideline:

- Follow the instructions described in each manual for the bundled product to use this product.

## 1.4 Feature

The MB95200 Series starter kit is the best for a performance and functional evaluation, and a check of operation before including MB95200 Series MCU in a user's system.

Below, the feature of the BGM debugger for MB95200 Series MCU is shown.

- Microcomputer operation voltage. It corresponds to +2.7V to +5.5V.  
(The maximum and minimum of microcomputer operation voltage and frequency of operation differs with each MCU. refer to the documents (a data sheet, hardware manual, etc.) of each device relation for the operation voltage and frequency of MCU of operation.)
- Compact development environment, a light and small BGM Adapter.
- Since a monitor program is performed in exclusive memory space, it does not consume user memory space.
- Continuation execution, step execution and break correspondence.
- It connects with a host computer by the USB interface.

## 1.5 Hardware Setup

In the hardware setup procedure, you configure and connect the hardware products. This chapter includes the configuring and connecting procedure for each product in order. Check the contents and complete the hardware setup.

- Configuration of each product
  - Configuring EV-Board
- Connection of each product
  - Connecting BGMA and EV-board
  - Connecting EV-Board power supply

## 2 BGMA Manual

This chapter gives introduction how to setup BGMA.

### 2.1 BGMA Overview

Below is the close look of a MB95200 Series BGMA. The Part Number is MB2146-08-E. It provides a debug platform for the MB95200 Series MCU in a small size (55.7mm (W) X127mm (D) X30mm (H)).

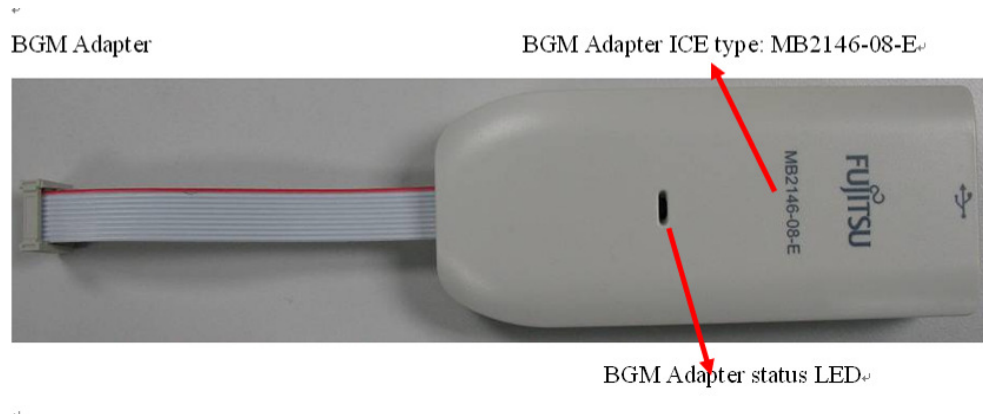


Figure 2.1-1 BGMA overview

### 2.2 Function List

ID	Function description	Remarks
1	Support MB95200 Series MCU	MCU MAX machine clock: 16.25 MHz MCU power voltage: 2.4 <sup>*1*</sup> 2 V ~ 5.5 <sup>*1</sup> V
2	Break pointer	256 software breakpoints
3	USB interface to PC/SOFTUNE	Compatible to USB protocol version 1.1
4	1-Line UART interface to the MB95200 Series MCU	The Baud rate is 62,500 bps
5	Support the MCU flash programming for engineering development	Provide high voltage for flash operation. <sup>*3</sup> The program and read speed is about 800 B/S.

\*1: The value varies depending on the operating frequency, the machine clock or the analog guaranteed range.

\*2: The value is 2.88 V when the low-voltage detection reset is used.

\*3: High voltage only for MB95200H/210H MCU

### 2.3 IDC10 Interface Description

Pin Number	Pin Name	Description
1	UVCC	Target MCU Vcc
2	GND	Target MCU Vss
3	RSTIN	Target MCU reset input
4	RSTOUT	Target MCU reset output
5	RSV	Reserved
6	RSV	Reserved
7	RSV	Reserved
8	DBG	Target MCU debug pin
9	RSV	Reserved
10	RSV	Reserved

### 2.4 BGMA USB Configuration

The BGMA is provided with a USB cable. Connect the BGMA to a PC with a USB cable. If the connection is right, the following window will pop up. Follow the instructions displayed, and then click “Next”,



Figure 2.4-1 Install BGMA in Windows (1)

Select “Install from a list or specific location (Advanced)”, then click “Next”,

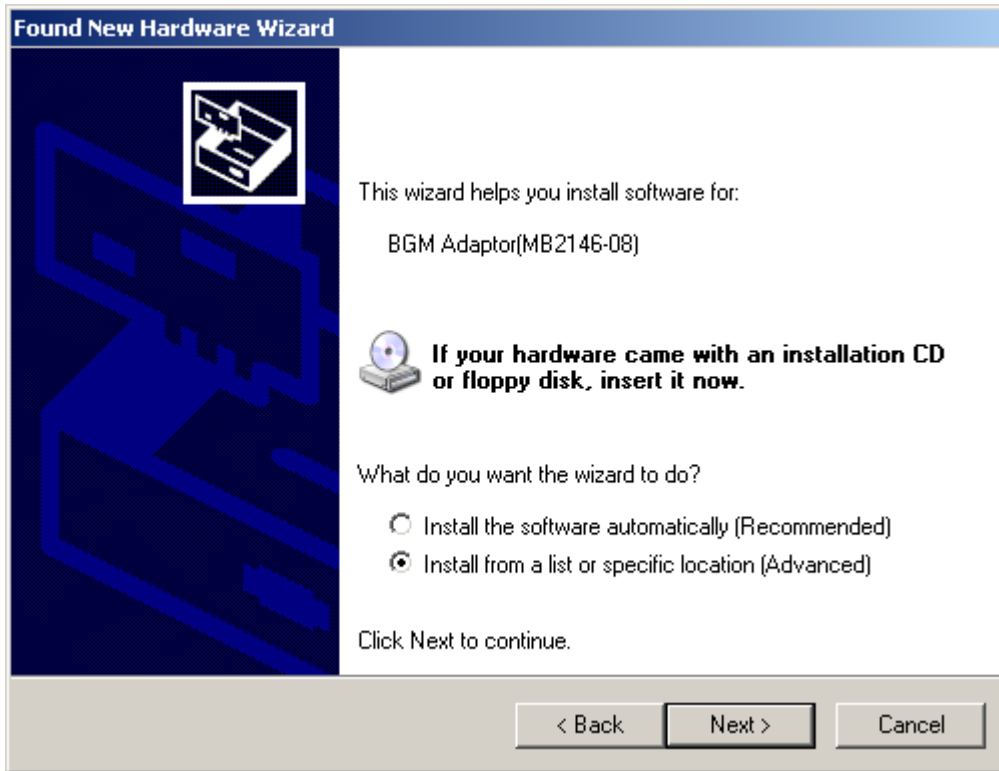


Figure 2.4-2 Install BGMA in Windows (2)

Select “...\Drivers” from the folder where SOFTUNE is installed, click “Next”,

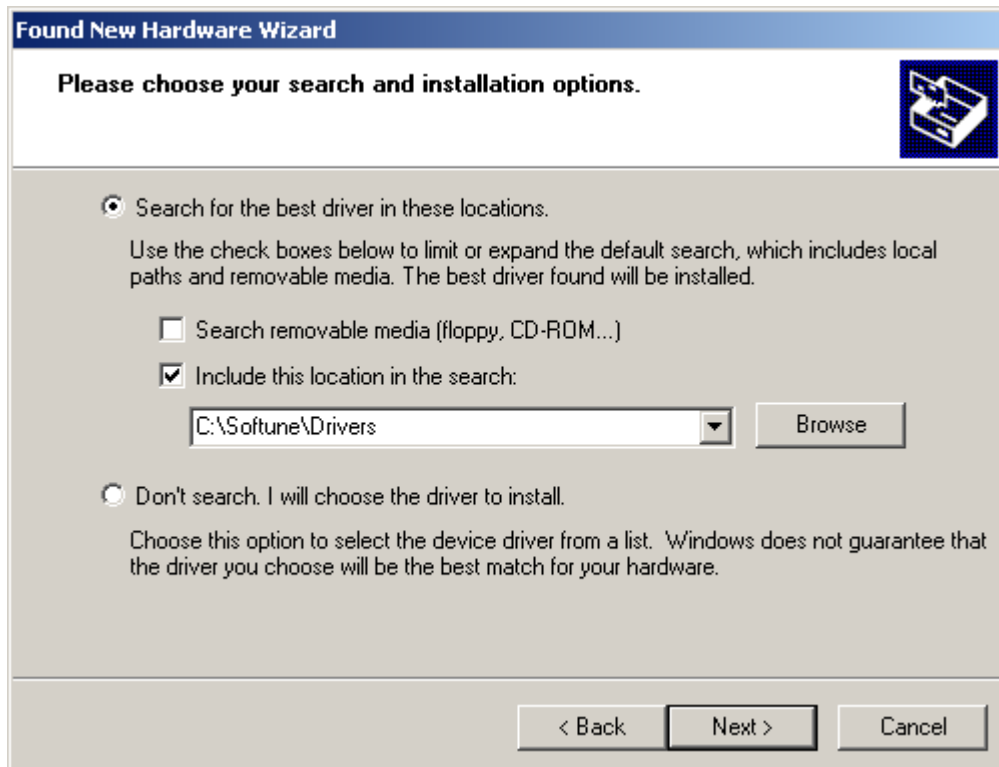


Figure 2.4-3 Install BGMA in Windows (3)

Select BGMA (MB2146-08) as displayed below, and then click “Next”,

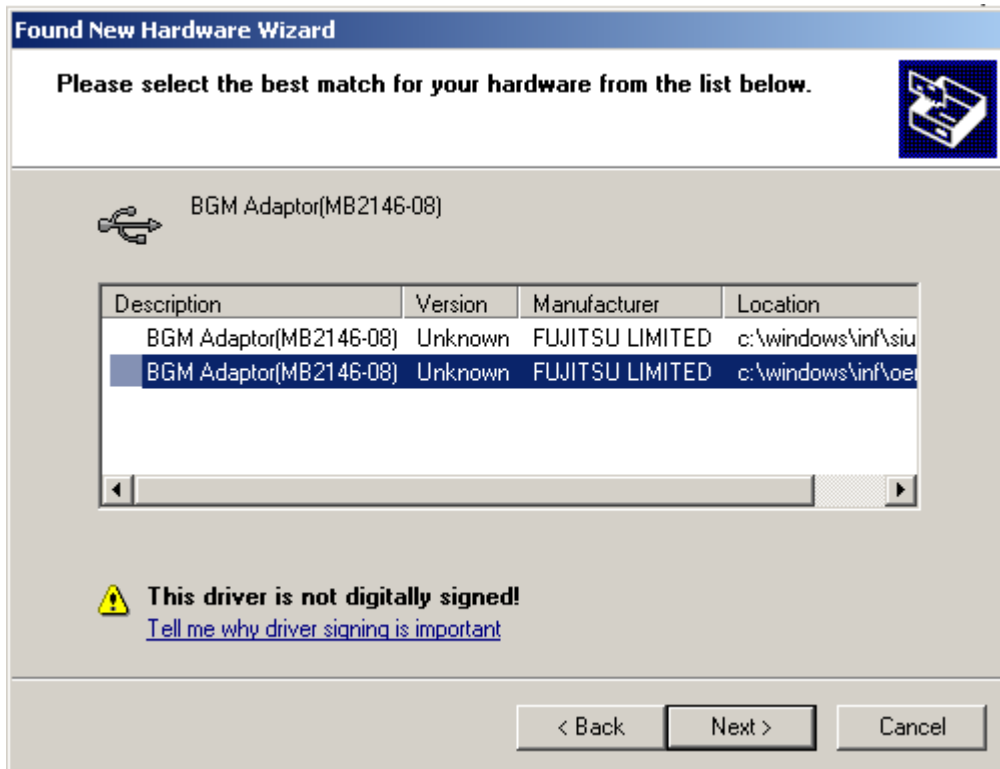


Figure 2.4-4 Install BGMA in Windows (4)

Windows will install the driver automatically. Click “Finish” after the driver has completed the installation normally. Then users can find the BGMA is recognized as MB2146-08 in Windows system.



Figure 2.4-5 BGMA is installed in Windows

## 2.5 LED Description

First, only plug USB cable to PC, check the Power LED on BGMA turns Green. Refer to Figure 2.5-1.



Figure 2.5-1 BGMA Power LED (1)

Second, plug IDC10 cable to the EV-board (target MCU board), then turn on EV-board. After that check Power LED on the BGMA turns Orange. Refer to Figure 2.5-2.

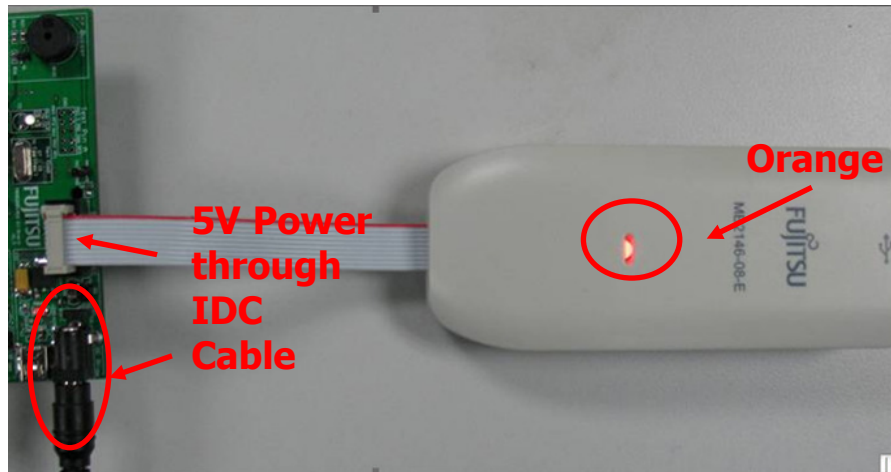


Figure 2.5-2 BGMA Power LED (2)

### 3 EV-board Manual

This chapter gives introduction how to setup EV-board.

#### 3.1 EV-board Overview

MB95200 Series EV-board is provided as a user-friendly introductory and evaluation platform for the MB95200 Family microcontroller.

MB95200H/210H MCU EV-board PN is MB2146-410-E;

MB95260H MCU EV-board PN is MB2146-420-E.

Figure 3.1-1 below is a close look of EV-board.

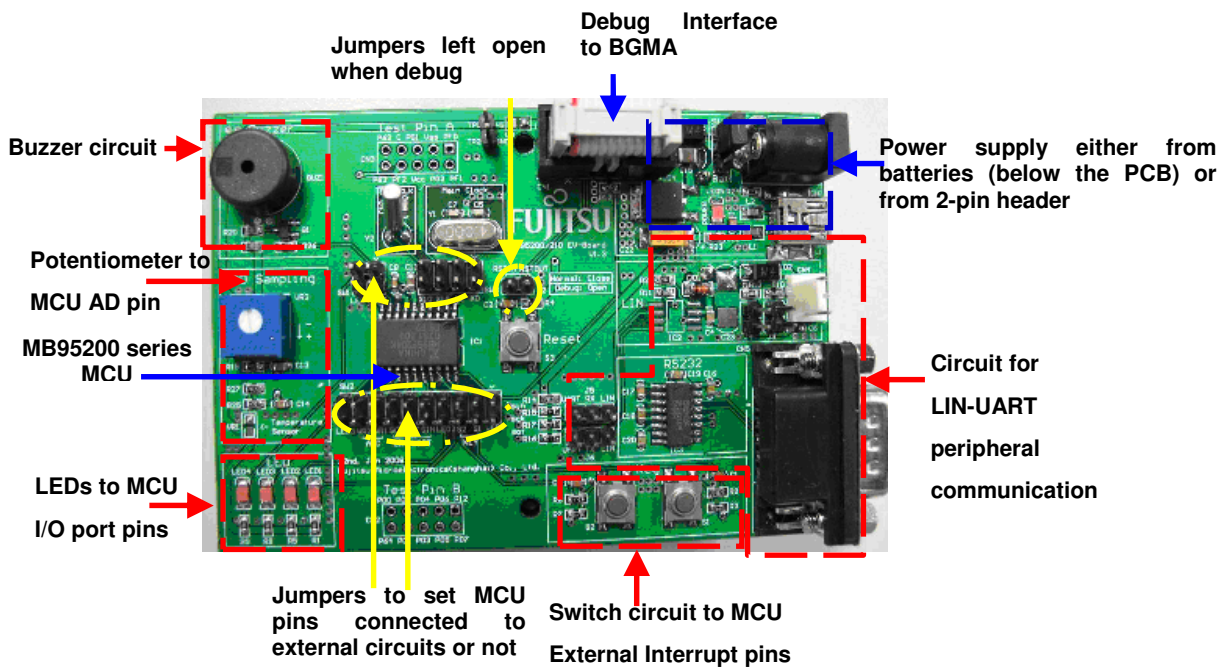


Figure 3.1-1 EV-board Overview



### 3.2 Function List

The EV-board consists of a board and a sample firmware. The board provides a useful platform for using the MCU and its peripherals. It is a useful development platform together with a BGMA (PN: MB2146-08-E) and a SOFTUNE. It features the following functions,

- ✓ Clock and sub-clock
- ✓ USB 5V power IF, external 5V power IF and the battery
- ✓ Reset circuit and reset key
- ✓ Provide IDC10 debug interface
- ✓ LEDs for general use, LED1 indicates DBG pin work status
- ✓ Provide one buzzer to demonstrate timer output
- ✓ 2 keys for general use
- ✓ Potentiometer and temperature sensor
- ✓ RS-232 level converter and DB9 interface (MAX232)
- ✓ LIN circuit reserved
- ✓ SIO 4 pins reserved
- ✓ All MCU ports are easy accessible through test pins
- ✓ Optional Starter kit to support each available MCU packages

### 3.3 EV-board Schematic

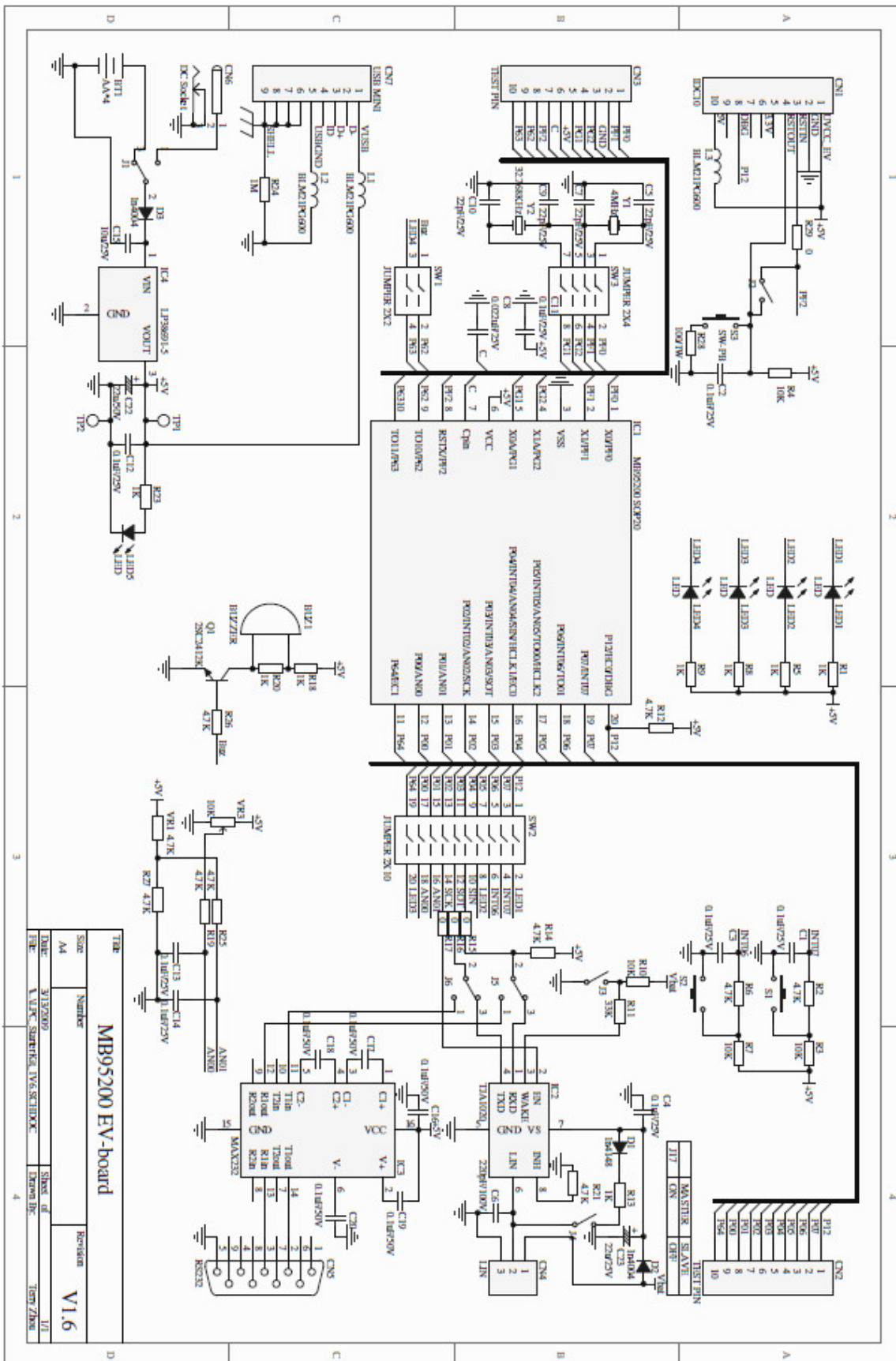


Figure 3.3-1 EV-board Schematic

### 3.4 HW Module Description and Jumper settings

#### 3.4.1 Power Module

EV-board has 4 kinds of power supply for user to choose. Please read below instructions before using.

✓ DC Adaptor: 9V DC:

Output voltage: 9V

Connection: Connector (CN6)

✓ Battery:

QTY: 4PCS;

Model: AA;

Nominal voltage: 1.5V.

Connection: Socket for Battery

✓ External Power Supply:

There are 2 test points (TP1&2) on EV-board which can supply power to target board (EV-board).

Power on method: short L3 on EV-board; then connect anode of DC power to TP1 (MCU Vcc) and cathode to TP2 (MCU GND).

✓ USB cable:

The Mini-B USB receptacle is only for supplying power to EV board by PC USB port. The current and power of this method is limited in 100mA/0.5W.

Please make sure to supply stable power via the Mini-B USB receptacle while operation.

If any of the power supplies is connected to the EV-board correctly, power LED (LED5) on the EV-board will be on. Refer to Figure 3.4.1-1.

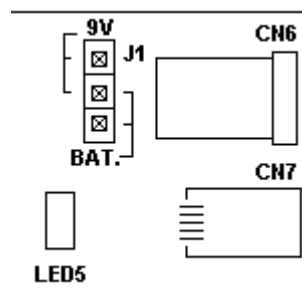


Figure 3.4.1-1 Power Module

The following two power supplies are recommended. Please follow the settings below,

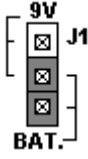
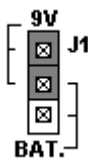
Power supply	Header name	Settings
4 AA batteries from BT1 on the back of the EV-board.	J1: BAT.	
9V DC from CN6	J1: 9V.	

Table 3.4.1-1 Power Supply Selection

### 3.4.2 BGMA Interface

To start the debug using a BGMA, users shall connect IDC10 socket from the BGMA to CN1 on an EV-board, and J2 shall be open in a debug mode. In a normal (free-run) mode, J2 shall be closed to enable reset key S3. Refer to Figure 3.4.2-1.

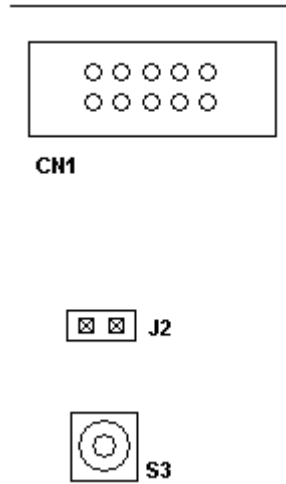


Figure 3.4.2-1 Debug Interface

MCU Mode	Header name	Settings
Debug mode	J2	J2
Normal mode	J2	J2

Table 3.4.2-1

### 3.4.3 Clock Settings

The MB95200 series MCU uses an internal main RC as a clock source by default. Users can select on-board crystal as a main clock and a sub-clock. Follow the settings below:

Clock	Header name	Settings
Main clock	SW3: X1, X0	SW3
Sub-clock	SW3: X1A, X0A	SW3

Table 3.4.3-1

### 3.4.4 Buzzer Module

A buzzer module is provided to demonstrate an 8/16 composite timer output (a continuous mode). To enable buzzer module, follow the table below:

Modules	Header name	Settings
Buzzer: BUZ1	SW1: BUZ.	

Table 3.4.4-1

### 3.4.5 A/D Module

VR1 and VR3 are to demonstrate a MCU A/D converter usage. Select VR1 or VR3 by the following table.

VR1 is connected to MCU A/D channel 1, and VR3 is connected to MCU A/D channel 0.

Modules	Header name	Settings
A/D: VR1	SW2: VR1	
A/D: VR3	SW2: VR3	

Table 3.4.5-1

### 3.4.6 LED Module

There are four LEDs on EV-board to demonstrate the I/O function.

Enable LED2, LED3 and LED4 in the following table,

Modules	Header name	Settings
LED*: LED2, LED3, LED4	SW2: LED2, LED3	
	SW1: LED4	

Table 3.4.6-1

\*: LED module has four LEDs in total, but P12 (LED1) is not active in a debug mode using BGMA.

### 3.4.7 Key Module

Key S1 and key S2 are provided to demonstrate an external interrupt function. Enable these two keys by the following table.

Key S1 is connected to external Int7, and key S2 is connected to external Int6.

Modules	Header name	Settings
Key: S1, S2	SW2: S1, S2	

Table 3.4.7-1

### 3.4.8 Lin-UART Module

Lin-UART module can be configured as a LIN or an UART module. Enable each module by the following table.

The UART module features an RS232 transceiver and a standard DB9 interface with PC.

Lin module enables an easy setup of Master/Slave communication by a Lin bus.

Modules	Header name	Settings
UART	SW2: SOT, SIN	
	J5: UART J6: UART	
LIN*	SW2: SCK, SOT, SIN	
	J5: LIN J6: LIN	

Table 3.4.8-1

\*: LIN module is optional; contact your local distributor for the EV-board with a LIN module.

## 4 Sample Code Manual

### 4.1 Topic List

The following sample codes are provided with MB95200 Series Starter Kit,

- ✓ IO\_LED project

In this example, the 3 LEDs will be on in the following sequence:

...->LED2->LED3->LED4...

- ✓ A/D\_Potentiometer project

In this example, the 3 LEDs will display “on” or “off” according to the arrow direction of VR3.

- ✓ Timer\_Buzzer project (Continuous timer)

In this example, Buzzer will be on once by pressing Key S1; Buzzer will be on twice by pressing Key S2.

- ✓ ExInt\_Key Project

Press Key S1, the 3 LEDs will be on in the following sequence:

...->LED4->LED3->LED2...

Press Key S2, the 3 LEDs will be on in the following sequence:

...->LED2->LED3->LED4...

- ✓ UART project

In this example, the MCU works in an asynchronous mode.

(9,600bps, 1 stop bit, no parity)

After a reset, the MCU will send "Welcome to the LIN-UART (asynchronous mode) of MB95200series (8FX)" to RS232 transceiver.

Then the MCU feedbacks any bytes it received.

- ✓ SIO project

This example uses two EV-boards to demonstrate the SIO (synchronous mode) function. (9600bps)

One MCU keeps sending data while the other keeps receiving. One LED is toggled to indicate the communication is correct.

- ✓ LIN slave

In this example, the MCU is running in a Lin Slave mode. After a reset, the MCU is waiting for data from the Master. If the data is received correct, LED4 starts twinkling.



## ✓ LIN master

In this example, the MCU is running in a Lin Master mode. After a reset, the MCU starts sending data to a Slave. If the Master gets correct data from the Slave, the MCU restarts sending. While sending data to the Slave, LED2 keeps twinkling.

## 4.2 Project Structure

The Sample code is organized by the following structure in each project. Here take IO\_LED project for example shown in Figure 4.2-1.

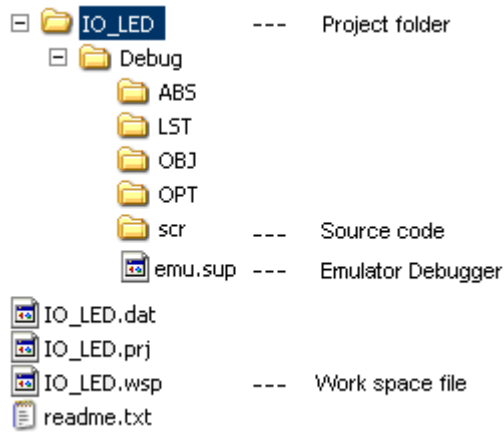
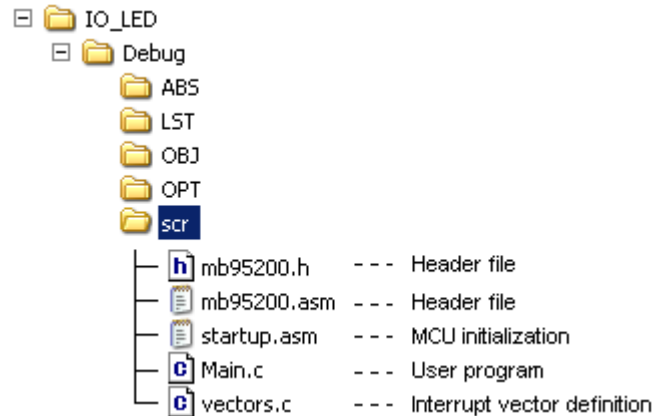


Figure 4.2-1 I/O\_LED Project Structure

### 4.3 Source Code File Description

**Five** files are available in each sample code source code folder shown below,



**Figure 4.3-1 Source Code Files**

#### 4.3.1 Header Files

The MB95200.h and the MB95200.asm are header files, including MB95200 Series MCU I/O registers definition;

Here take PDR0 for example. In MB95200.h, PDR0 is defined as below.

```

/* REGISTER BIT STRUCTURES */
typedef union{ /* Port0 */
    IO_BYTE  byte;
    struct{
        IO_BYTE _P07:1;
        IO_BYTE _P06:1;
        IO_BYTE _P05:1;
        IO_BYTE _P04:1;
        IO_BYTE _P03:1;
        IO_BYTE _P02:1;
        IO_BYTE _P01:1;
        IO_BYTE _P00:1;
    } bit;
} PDR0STR;

.....
__IO_EXTERN __io PDR0STR _pdr0; /* Port0 */
#define PDR0 _pdr0.byte
#define PDR0_P00 _pdr0.bit._P00
#define PDR0_P01 _pdr0.bit._P01
#define PDR0_P02 _pdr0.bit._P02

```

```
#define PDR0_P03 _pdr0.bit._P03
#define PDR0_P04 _pdr0.bit._P04
#define PDR0_P05 _pdr0.bit._P05
#define PDR0_P06 _pdr0.bit._P06
#define PDR0_P07 _pdr0.bit._P07
```

In MB95200.asm, PDR0 address 0x0000 is allocated to PDR0 as below.

```
__pdr0 .res.b 1 ;000000 /* Port0 */
PDR0 .equ 0x0000
```

#### 4.3.2 Startup.asm File

The Startup.asm is the MB95200 Series MCU initialization file including stack settings, register bank settings and watchdog settings etc;

#### 4.3.3 Vectors.c File.

The Vectors.c contains the MB95200 Series MCU Interrupt vector definition.

User can pre-set all interrupt control registers in function InitIrqLevels(). It can be used to set all interrupt priorities in static applications. For example, to set the external interrupt ch.0 to level 0, change the following code:

```
ILR0 = 0xFF; // IRQ0: external interrupt ch.0 | ch.4
            // IRQ1: external interrupt ch.1 | ch.5
            // IRQ2: external interrupt ch.2 | ch.6
            // IRQ3: external interrupt ch.3 | ch.7
```

To

```
ILR0 = 0xFC; // IRQ0: external interrupt ch.0 | ch.4
            // IRQ1: external interrupt ch.1 | ch.5
            // IRQ2: external interrupt ch.2 | ch.6
            // IRQ3: external interrupt ch.3 | ch.7
```

And declare the interrupt function as in Vectors.c below,

```
.....
__interrupt void external_int00 (void);
.....
#pragma intvect external_int00 0 // IRQ0: external interrupt ch0 | ch4
```

Then user can write his own interrupt sub-routine in Main.C shown as below.

```
/*----- INTERRUPT SERVICE ROUTINE -----*/
__interrupt void external_int00(void)
{
    //User code
}
```

#### 4.3.4 Main.c File

Main.c contains the user code.

## 5 Development Platform Quick Start

### 5.1 Tools Setup Sequence

Start the debugging system in the following sequence:

- ✓ Connect a BGMA to the PC using a USB cable, confirm the LED on the BGMA is Green;
- ✓ Connect an EV-board to BGMA IDC10 socket;
- ✓ Select the EV-board power supply and turn on the EV-board, confirm the LED on the BGMA is Orange and the Power LED on the EV-board is on.

### 5.2 Open Project and Start Debug

Users can start a debug from a PC software SOFTUNE workbench in the following sequence. Here take IO\_LED project for example.

- ✓ Start the SOFTUNE from “Startup Menu>Programs> SOFTUNE V3> FFMC-8L Family SOFTUNE Workbench” in Windows;
- ✓ Click “Open workspace” from “File” Menu in SOFTUNE;
- ✓ Select “IO\_LED.wsp” in “Open Space” window;
- ✓ Click “Start debug” from “Debug” Menu.

If the entire procedure goes right, a debug will start normally.

### 5.3 Operation Precautions

- ✓ All pins of MB95200 Series MCU are connected to Testing Pin on the EV-board. If the user wants to test separate pin performance, do disconnect the header of related peripheral modules.
- ✓ Note that J2 shall be open while debugging using the BGMA. User reset from S3 (reset key) is invalid when J2 is open. To use reset key S3 in a normal mode, J2 should be closed.
- ✓ It's recommended that only one power supply is used as a power module input at a time.

---

**MCU-UM-500002-E-16**

---

**FUJITSU MICROELECTRONICS LIMITED • SUPPORT SYSTEM**

**F2MC-8FX Family MB95200 Series**

**Starter Kit**

**MB2146-410-01-E / MB2146-420-01-E**

**SETUP GUIDE**

---

**Mar. 2009 the sixth edition**

**Published FUJITSU MICROELECTRONICS LIMITED**

**Edited Strategic Business Development Dept.**

---

---

The Fujitsu logo is centered on the page. It consists of the word "FUJITSU" in a bold, red, serif font. The letter "J" is stylized with a small circle above it, which is connected to the top of the letter. The logo is positioned between two horizontal lines that span the width of the page.

**FUJITSU**

---