TOSHIBA

TOSHIBA Original CMOS 8-Bit Microcontroller

TLCS-870/C Series

TMP86FM25FG

TOSHIBA CORPORATION

Semiconductor Company

Revision History

Date	Revision	
2008/3/6	1	First Release
2008/8/29	2	Contents Revised

Caution in Setting the UART Noise Rejection Time

When UART is used, settings of RXDNC are limited depending on the transfer clock specified by BRG. The combination "O" is available but please do not select the combination "–".

The transfer clock generated by timer/counter interrupt is calculated by the following equation :

Transfer clock [Hz] = Timer/counter source clock [Hz] ÷ TTREG set value

BRG setting	Transfer clock [Hz]	RXDNC setting			
		00 (No noise rejection)	01 (Reject pulses shorter than 31/fc[s] as noise)	10 (Reject pulses shorter than 63/fc[s] as noise)	11 (Reject pulses shorter than 127/fc[s] as noise)
000	fc/13	O	O	O	–
110 (When the transfer clock generated by timer/counter interrupt is the same as the right side column)	fc/8	O	–	–	–
	fc/16	O	O	–	–
	fc/32	O	O	O	–
The setting except the above		O	O	O	O

Document Change Notification

The purpose of this notification is to inform customers about the launch of the Pb-free version of the device. The introduction of a Pb-free replacement affects the datasheet. Please understand that this notification is intended as a temporary substitute for a revision of the datasheet.

Changes to the datasheet may include the following, though not all of them may apply to this particular device.

1. Part number

   Example: TMPxxxxxxF → TMPxxxxxxFG

   All references to the previous part number were left unchanged in body text. The new part number is indicated on the prelims pages (cover page and this notification).

2. Package code and package dimensions

   Example: LQFP100-P-1414-0.50C → LQFP100-P-1414-0.50F

   All references to the previous package code and package dimensions were left unchanged in body text. The new ones are indicated on the prelims pages.

3. Addition of notes on lead solderability

   Now that the device is Pb-free, notes on lead solderability have been added.

4. RESTRICTIONS ON PRODUCT USE

   The previous (obsolete) provision might be left unchanged on page 1 of body text. A new replacement is included on the next page.

5. Publication date of the datasheet

   The publication date at the lower right corner of the prelims pages applies to the new device.

1. Part number

2. Package code and dimensions

Previous Part Number (in Body Text)	Previous Package Code (in Body Text)	New Part Number	New Package Code	OTP
TMP86FM25F	P-QFP100-1420-0.65A	TMP86FM25FG	QFP100-P-1420-0.65Q	–

*: For the dimensions of the new package, see the attached Package Dimensions diagram.

3. Addition of notes on lead solderability

The following solderability test is conducted on the new device.

Lead solderability of Pb-free devices (with the G suffix)

Test	Test Conditions	Remark
Solderability	(1) Use of Lead (Pb) ·solder bath temperature = 230°C ·dipping time = 5 seconds ·the number of times = once ·use of R-type flux (2) Use of Lead (Pb)-Free ·solder bath temperature = 245°C ·dipping time = 5 seconds ·the number of times = once ·use of R-type flux	Leads with over 95% solder coverage till lead forming are acceptable.

4. RESTRICTIONS ON PRODUCT USE

The following replaces the "RESTRICTIONS ON PRODUCT USE" on page 1 of body text.

RESTRICTIONS ON PRODUCT USE 20070701-EN

- The information contained herein is subject to change without notice.
- TOSHIBA is continually working to improve the quality and reliability of its products. Nevertheless, semiconductor devices in general can malfunction or fail due to their inherent electrical sensitivity and vulnerability to physical stress. It is the responsibility of the buyer, when utilizing TOSHIBA products, to comply with the standards of safety in making a safe design for the entire system, and to avoid situations in which a malfunction or failure of such TOSHIBA products could cause loss of human life, bodily injury or damage to property.
  In developing your designs, please ensure that TOSHIBA products are used within specified operating ranges as set forth in the most recent TOSHIBA products specifications. Also, please keep in mind the precautions and conditions set forth in the "Handling Guide for Semiconductor Devices," or "TOSHIBA Semiconductor Reliability Handbook" etc.
- The TOSHIBA products listed in this document are intended for usage in general electronics applications (computer, personal equipment, office equipment, measuring equipment, industrial robotics, domestic appliances, etc.).These TOSHIBA products are neither intended nor warranted for usage in equipment that requires extraordinarily high quality and/or reliability or a malfunction or failure of which may cause loss of human life or bodily injury ("Unintended Usage"). Unintended Usage include atomic energy control instruments, airplane or spaceship instruments, transportation instruments, traffic signal instruments, combustion control instruments, medical instruments, all types of safety devices, etc.. Unintended Usage of TOSHIBA products listed in his document shall be made at the customer's own risk.
- The products described in this document shall not be used or embedded to any downstream products of which manufacture, use and/or sale are prohibited under any applicable laws and regulations.
- The information contained herein is presented only as a guide for the applications of our products. No responsibility is assumed by TOSHIBA for any infringements of patents or other rights of the third parties which may result from its use. No license is granted by implication or otherwise under any patents or other rights of TOSHIBA or the third parties.
- Please contact your sales representative for product-by-product details in this document regarding RoHS compatibility. Please use these products in this document in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances. Toshiba assumes no liability for damage or losses occurring as a result of noncompliance with applicable laws and regulations.
- For a discussion of how the reliability of microcontrollers can be predicted, please refer to Section 1.3 of the chapter entitled Quality and Reliability Assurance/Handling Precautions.

5. Publication date of the datasheet

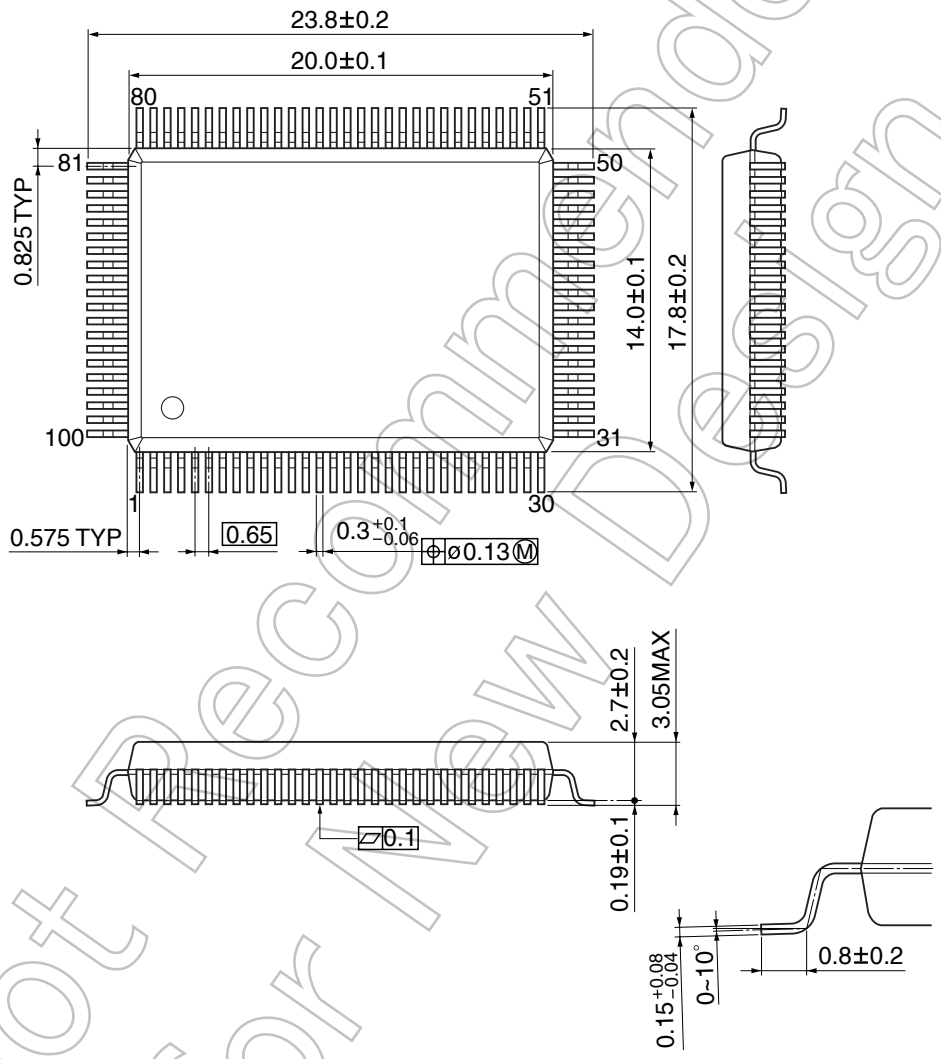
The publication date of this datasheet is printed at the lower right corner of this notification.

(Annex)

Package Dimensions

QFP100-P-1420-0.65Q

Unit: mm



Comparison table of TMP86CM25F/CS25F/PS25F/C925XB and TMP86CM25AF/FM25F Difference

	TMP86CM25F/ TMP86CS25F	TMP86PS25F	TMP86C925XB (Emulation chip)	TMP86FM25F	TMP86CM25AF
ROM	32 K (Mask ROM) 60 K (Mask ROM)	60 K (OTP)	–	32 K (Flash)	32 K (Mask ROM)
RAM	2 K		–	2 K	
I/O	42 pin		42 pin (MCU part)	42 pin	
External Interrupt	5 pin			5 pin	
AD Converter	8-bit AD converter × 8 ch			8-bit AD converter × 8 ch (Note 3)	
Timer Counter	18-bit timer × 1 ch 8-bit timer × 4 ch			18-bit timer × 1 ch 8-bit timer × 4 ch	
Serial Interface	8-bit SIO × 2 ch UART × 1 ch			8-bit SIO × 2 ch UART × 1 ch	
LCD	60 seg × 16 com			60 seg × 16 com (Note 4)	
Key-on Wakeup	4 ch			4 ch	
Operating Voltage in MCU Mode	1.8 to 5.5 V at 4.2 MHz 2.7 to 5.5 V at 8 MHz 4.5 to 5.5 V at 16 MHz		1.8 to 5.25 V at 4.2 MHz 2.7 to 5.25 V at 8 MHz 4.5 to 5.25 V at 16 MHz	1.8 to 3.6 V at 4.2 MHz (External clock) 1.8 to 3.6 V at 8 MHz (Resonator) 2.7 to 3.6 V at 16 MHz	
Operating Temperature in MCU Mode	−40 to 85℃		0 to 60°C	−40 to 85°C	
Writing to Flash Memory	–			2.7 to 3.6V at 16 MHz 25°C ± 5°C	–
Package	P-QFP100-1420-0.65A		FBGA272	P-QFP100-1420-0.65A	
CPU Wait (Note 1)	N/A			Available (Note 2)	

Note 1: The CPU wait is a CPU halt function for stabilizing of power supply of Flash memory. The CPU wait period is as follows. In the CPU wait period except RESET, CPU is halted but peripheral functions are not halted. Therefore, if the interrupt occurs during the CPU wait period, the interrupt latch is set. In this case, if the IMF has been set to "1", the interrupt service routine is executed after CPU wait period. For details refer to 1.1 "Flash Memory" in TMP86FM25F data sheet.

Condition	Wait Time	Halt/Operate	
		CPU	Peripherals
After reset release	2^{10}/fc[s]	Halt	Halt
Changing from STOP mode to NORMAL mode (at EEPCR<MNPWDW> = "1")	2^{10}/fc[s]	Halt	Operate
Changing from STOP mode to SLOW mode (at EEPCR<MNPWDW> = "1")	2^{3}/fs[s]	Halt	Operate
Changing from IDLE0/1/2 mode to NORMAL mode (at EEPCR<ATPWDW> = "0")	2^{10}/fc[s]	Halt	Operate
Changing from SLEEP0/1/2 mode to SLOW mode (at EEPCR<ATPWDW> = "0")	2^{3}/fs[s]	Halt	Operate

Note 2: Though the TMP86CM25AF does not have a Flash memory, the CPU wait function is inserted in TMP86CM25A to keep the compatibility with Flash product (TMP86FM25F).

Note 3: AD conversion time of TMP86CM25A/FM25 is different from that of TMP86CM25/CS25/PS25/C925. For details, refer to 2.12 "8-Bit AD Converter (ADC)".

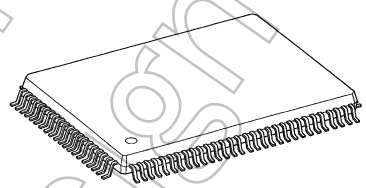
Note 4: The reference voltage of TMP86CM25A/FM25 is different from that of TMP86CM25/CS25/PS25/C925. For details, refer to "Electrical Characteristics".

CMOS 8-Bit Microcontroller
TMP86FM25F

The TMP86FM25 is a Flash type MCU which includes 32 Kbytes Flash memory. It is a pin compatible with a mask ROM product "A" version of the TMP86CM25A. Writing the program to built-in Flash memory, the TMP86FM25 operates as the same way as the TMP86CM25A. The TMP86FM25 has a 2 Kbytes BOOT ROM (Masked ROM) for programming to Flash memory.

Product No.	Flash Memory	BOOT ROM	RAM	Package
TMP86FM25F	32 K × 8 bits	2 K × 8 bits	2.0 K × 8 bits	P-QFP100-1420-0.65A

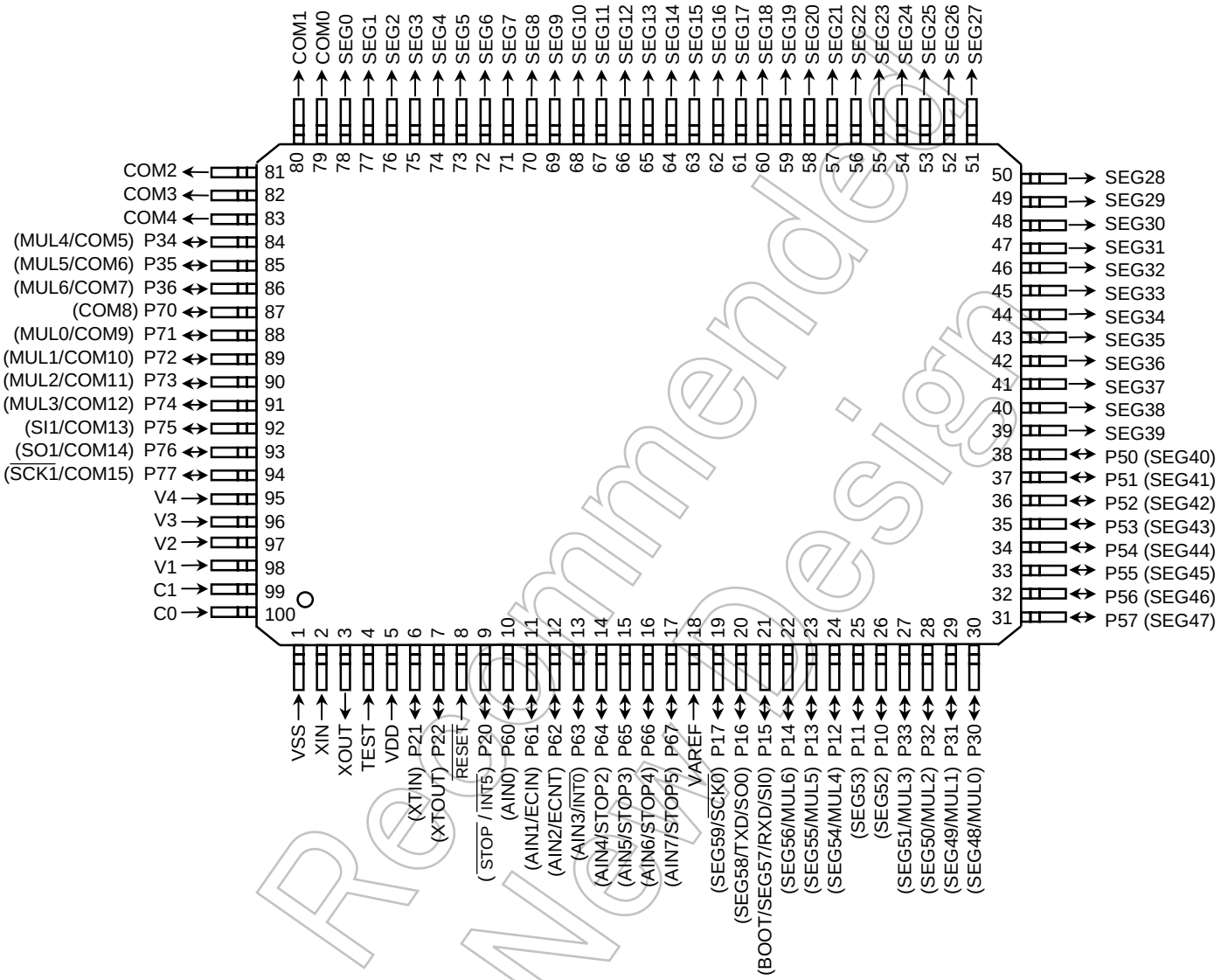
P-QFP100-1420-0.65A



TMP86FM25F

Pin Assignments (Top view)

P-QFP100-1420-0.65A

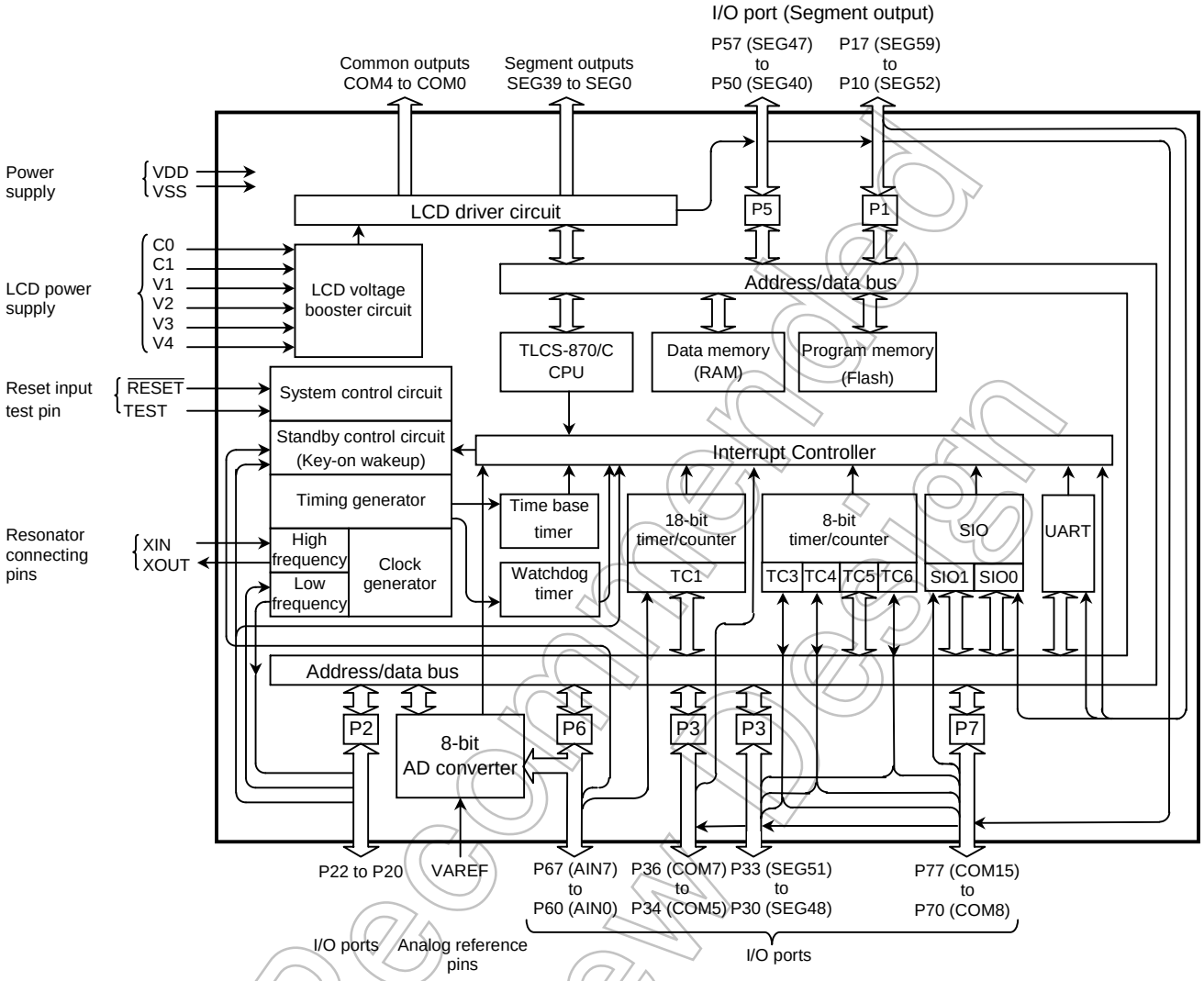


Note1: The masked ROM product (TMP86CM25AF/CM25F/CS25F), the OTP product (TMP86PS25F) and the emulation chip (TMP86C925XB) don't have a BOOT function in P15 pin.

Note2: Ports assigned as MUL6 to MUL0 can switch pin assignment by the multifunction register (MULSEL). For functions assigned to each pin, see the table below.

Pin Name	Function	Pin Assignment
MUL0	$\overline{\text{DVO}}$	P30 or P71
MUL1	$\overline{\text{PDO3}}$, $\overline{\text{PWM3}}$,TC3	P31 or P72
MUL2	$\overline{\text{PDO4}}$, $\overline{\text{PWM4}}$, $\overline{\text{PPG4}}$,TC4	P32 or P73
MUL3	$\overline{\text{PDO6}}$, $\overline{\text{PWM6}}$, $\overline{\text{PPG6}}$,TC6	P33 or P74
MUL4	INT1	P12 or P34
MUL5	INT2	P13 or P35
MUL6	INT3	P14 or P36

Block Diagram



Pin Funtions

The TMP86FM25 has MCU mode and serial PROM mode.

(1) MCU mode

In the MCU mode, the TMP86FM25 is a pin compatible with the TMP86CM25A (Make sure to fix the TEST pin to low level).

(2) Serial PROM mode

In the Serial PROM mode, programming to Flash memory is available by executing BOOT ROM.

In the serial PROM mode, TXD (P16) and RXD (P15) pins are used as a serial interface pin. Therefore, if the programming is executed on-board after mounting, these pins should be released from the other devices for communication in serial PROM mode.

1.1 FLASH Memory

1.1.1 Outline

The TMP86FM25 incorporates 32768 bytes of FLASH memory (Address 8000H to FFFFH). The writing to FLASH is controlled by FLASH control register (EEPCR), FLASH status register (EEPSR).

To write data to the FLASH, execute the Serial PROM mode. For details about the Serial PROM mode, refer to "2.1 Serial PROM Mode".

The FLASH memory of the TMP86FM25 features:

- The FLASH memory is constructed of 512 pages FLASH memory and one page size is 64 bytes (512 pages × 64 bytes = 32768 bytes).
- The TMP86FM25 incorporates a 64-byte temporary data buffer. The data written to FLASH memory is temporarily stored in this data buffer. After 64 bytes data have been written to the temporary data buffer, the writing to FLASH memory automatically starts by page writing (The 64 bytes data are written to specified page of FLASH simultaneously). At the same time, page-by-page erasing occurs automatically. So, it is unnecessary to erase individual pages in advance.
- The FLASH control circuit incorporates an oscillator dedicated to the FLASH. So FLASH writing time is independent of the system clock frequency (fc). In addition, because an FLASH control circuit controls writing time for each FLASH memory cell, the writing time varies in each page (Typically 4 ms per page).
- Controlling the power for the FLASH control circuit (Regulator and voltage step-up circuit) achieves low power consumption if the FLASH is not in use (Example: When the program is executed in RAM area).

1.1.2 Conditions for Accessing the FLASH Areas

The conditions for accessing the FLASH areas vary depending on each operation mode. The following tables shows FLASH are access conditions.

Table 1.1.1 FLASH Area Access Conditions

	Area	Operation Mode	
		MCU Mode (Note 1)	Serial PROM Mode (Note 2)
FLASH memory	8000H to FFFFH	Read/Fetch only	Write/Read/Fetch supported

Note 1: "MCU Mode" shows NORMAL1/2 and SLOW1/2 modes.

Note 2: "Serial PROM Mode" shows the FLASH controlling mode. For details, refer to 2.1 "Serial PROM Mode".

Note 3: "Fetch" means reading operation of FLASH data as an instruction by CPU.

1.1.3 Differences among Product Series

The specifications of the FLASH product (TMP86FM25) are different from TMP86CM25A (Masked ROM "A" version), TMP86C925XB (Emulation chip), TMP86CM25F/CS25F (Masked ROM) and TMP86PS25 (OTP) as listed below. See 1.2.2 "Control" for explanations about the control registers.

		FLASH Product (TMP86FM25)	Masked ROM "A" Version (TMP86CM25AF)	The Current Products TMP86C925XB (Emulation chip) TMP86CM25F/CS25F (Mask ROM) TMP86PS25F (OTP)
Rewriting the EEPCR register <EEPMD, EEPRS, MNPWDW>		It is possible to rewrite the EEPCR register only when the program execution area in use is RAM/BOOT-ROM.	Neither the EEPMD nor EEPRS itself does not function.	The FLASH function is not executed because the emulation chip and the MASK (except "A" version)/OTP products don't have EEPCR and EEPSR registers. Therefore, the software including the FLASH register can not be emulated by the emulation chip. If the software including the FLASH register is executed in the MASK (except "A" version)/OTP or the emulation chip, the software process differs from the FLASH product.
FLASH write time		Typically 4 ms (Independent of the system clock)	(Writing to an area that corresponds to the FLASH area causes nothing.)	
Executing a read instruction/fetch to the 8000H to FFFFH area when EEPSR<BFBUSY> = "1".		If EEPSR<BFBUSY> = "1", executing a read instruction/fetch to the FLASH area causes FFH to be read regardless of what the current ROM data is. Fetching FFH results in a software interrupt occurring.	Always masked ROM data is read.	
Executing a write instruction to the 8000H to FFFFH area when EEPCR<EEPMD> = "0011" EEPSR<EWUPEN> = "1" and EEPSR<BFBUSY> = "0"	MCU mode	The EEPSR<BFBUSY> stays at "0" (Write disabled).		
	Serial PROM mode	The EEPSR<BFBUSY> is set to "1" (Write enabled).	–	
CPU wait for Flash (Wait period for stabilizing of the power supply of Flash control circuit)		The wait period is inserted in the releasing from Reset, STOP mode (EEPCR<MNPWDW> = "1") and IDLE/SLEEP mode (EEPCR<ATPWDW> = "0"). Even if the FLASH register is not used for software, the wait period is inserted in Reset process.		The wait period is not inserted. Even if the FLASH register is not used for software, the Reset and STOP process differs from the FLASH product.
BOOT-ROM		2 Kbytes are included in the 3800H to 3FFFH area.	No BOOT-ROM is included. Executing a read/fetch to the 3800H to 3FFFH area causes "FFH" to be read. Fetching "FFH" results in a software interrupt occurring.	The current products don't have BOOT-ROM. Therefore, the serial PROM mode can not be emulated in the current products.
Operating voltage (VDD)		1.8 to 3.6 V (1 MHz to 4.2 MHz: External clock) 1.8 to 3.6 V (1 MHz to 8 MHz: Resonator) 2.7 to 3.6 V (1 MHz to 16 MHz)		1.8 to 5.5 V (1 MHz to 4.2 MHz) 2.7 to 5.5 V (1 MHz to 8 MHz) 4.5 to 5.5 V (1 MHz to 16 MHz) The maximum voltage of the TMP86C925XB is 5.25 V.

1.1.4 FLASH Memory Configuration

64 consecutive bytes in the FLASH area are treated as one group, which is defined as a page. The TMP86FM25 incorporates a one-page temporary data buffer. Writing data to FLASH is temporarily stored in this 64-byte data buffer. After 64 bytes data have been written to the temporary data buffer, these data are written to specified page of FLASH at a time. However, data can be read from any address byte by byte.

1.1.4.1 Page Configuration

The FLASH area has a page configuration of 64 bytes/page as shown below. The total number of bytes in it is 512 pages × 64 bytes (= 32768 bytes). The writeable area is 8000H to FFFFH in Serial PROM mode.

Note: The FLASH area (8000H to FFFFH) can be written only in the Serial PROM mode. For details of the Serial PROM mode, refer to 2.1 “Serial PROM Mode”.

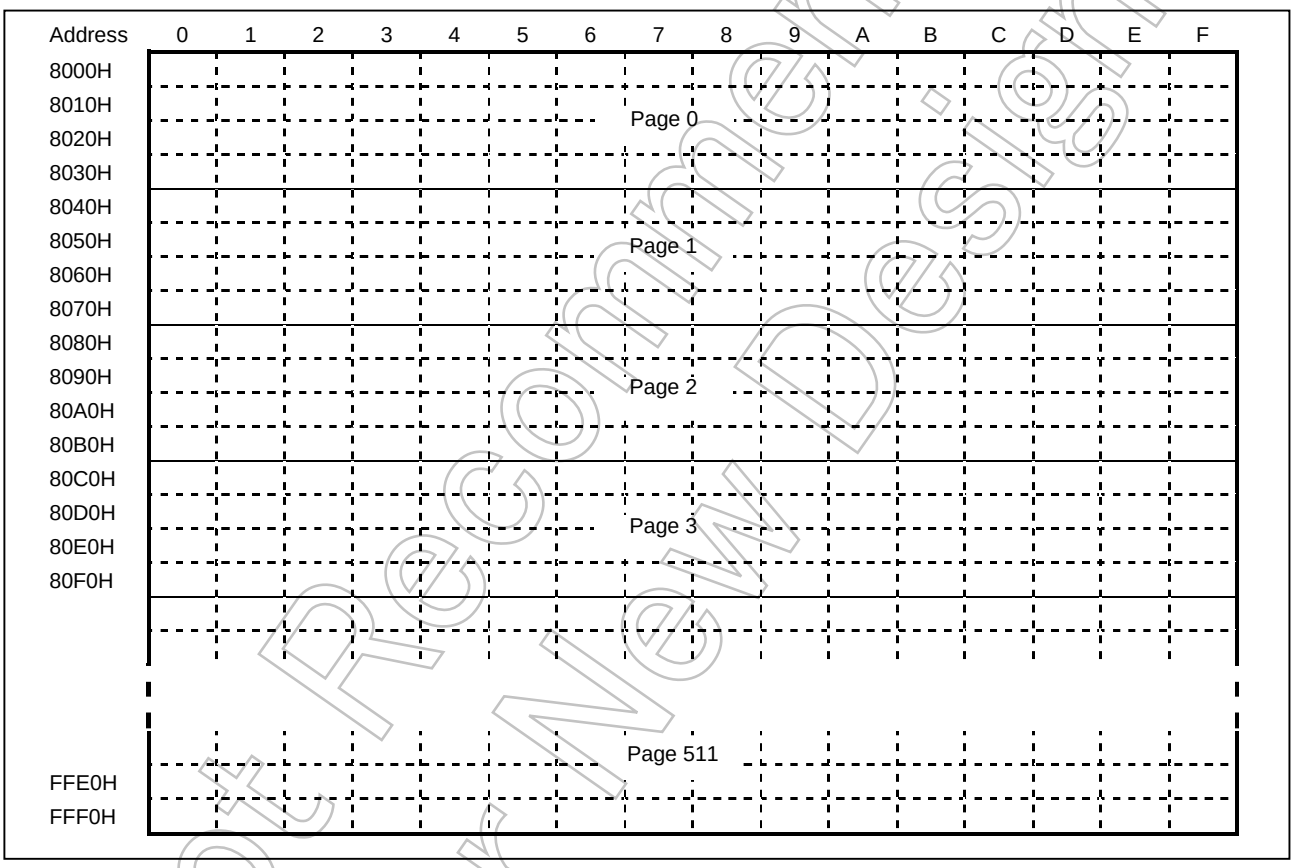


Figure 1.1.1 Page Configuration

1.2 FLASH Memory Control Circuit

1.2.1 Configuration

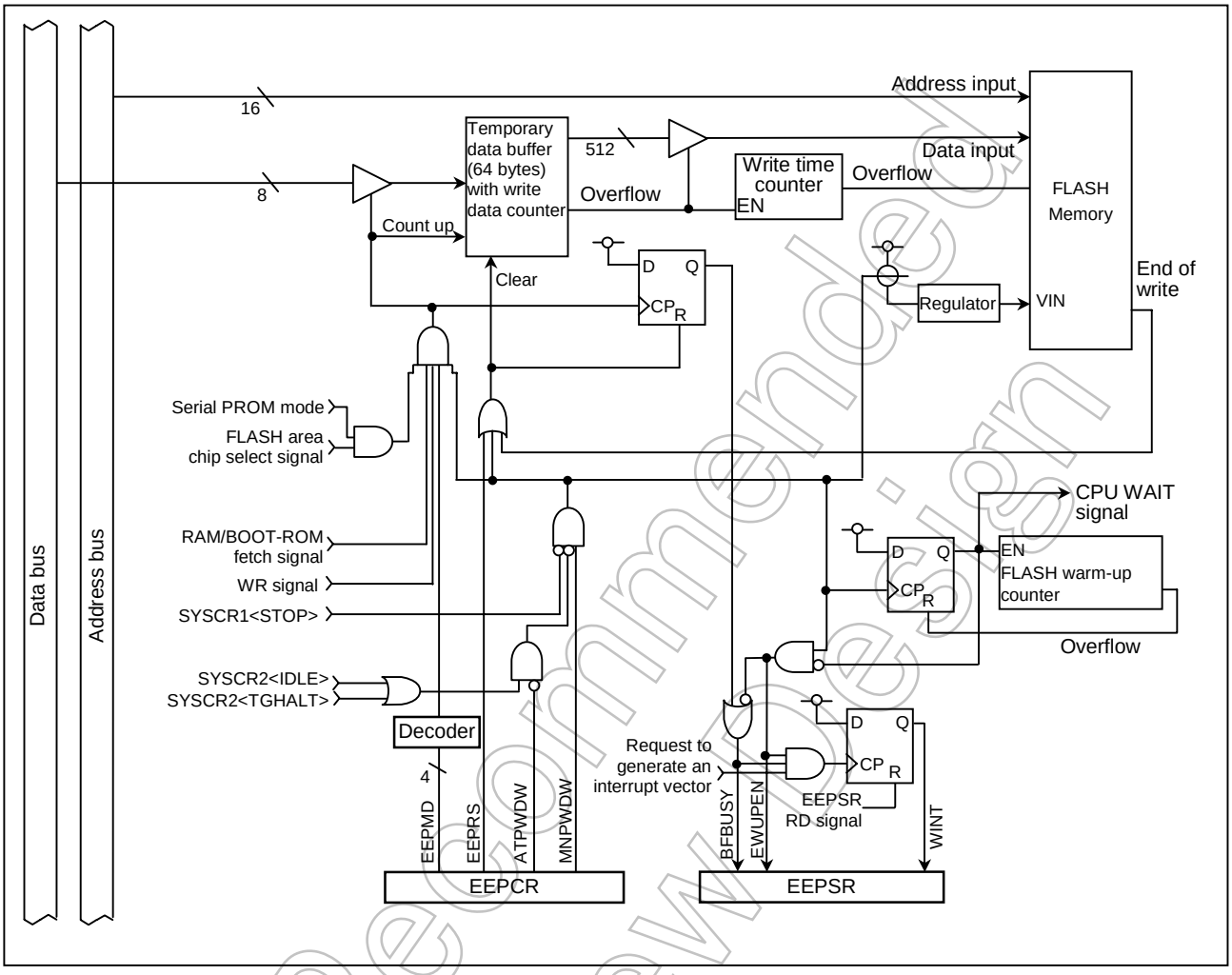


Figure 1.2.1 FLASH Memory Control

1.2.2 Control

The FLASH memory is controlled by FLASH control register (EEPCR) and FLASH status register (EEPSR). These registers are assigned to DBR.

Address	Read	Write
$0F90_H$	SIO0BR0 (SIO0 buffer 0)	
91	SIO0BR1 (SIO0 buffer 1)	
92	SIO0BR2 (SIO0 buffer 2)	
93	SIO0BR3 (SIO0 buffer 3)	
94	SIO0BR4 (SIO0 buffer 4)	
95	SIO0BR5 (SIO0 buffer 5)	
96	SIO0BR6 (SIO0 buffer 6)	
97	SIO0BR7 (SIO0 buffer 7)	
98	–	SIO0CR1 (SIO0 control register 1)
99	SIO0SR (SIO0 status register)	SIO0CR2 (SIO0 control register 2)
9A	–	STOPCR (Key-on wakeup control register)
9B	RDBUF (UART received data buffer)	TDBUF (UART transmit data buffer)
9C	Reserved	
:	:	
9F	Reserved	
A0	SIO1BR0 (SIO1 buffer 0)	
A1	SIO1BR1 (SIO1 buffer 1)	
A2	SIO1BR2 (SIO1 buffer 2)	
A3	SIO1BR3 (SIO1 buffer 3)	
A4	SIO1BR4 (SIO1 buffer 4)	
A5	SIO1BR5 (SIO1 buffer 5)	
A6	SIO1BR6 (SIO1 buffer 6)	
A7	SIO1BR7 (SIO1 buffer 7)	
A8	–	SIO1CR1 (SIO1 control register 1)
A9	SIO1SR (SIO1 status register)	SIO1CR2 (SIO1 control register 2)
AA	Reserved	
:	:	
BF	Reserved	
C0	MULSEL (Multiplexed function select register)	
C1	Reserved	
:	:	
DF	Reserved	
E0	***EEPCR (Flash memory control)***	
E1	***EEPSR(Flash memory status)***	–
E2	Reserved	
:	:	
FF	Reserved	

Note 1: Do not access reserved areas by the program.

Note 2: –: Cannot be accessed.

Note 3: Write-only registers and interrupt latches cannot use the read-modify-write instructions (Bit manipulation instructions such as SET, CLR, etc. and logical operation instructions such as AND, OR, etc.).

Figure 1.2.1 The Data Buffer Register (DBR) for TMP86FM25

FLASH Control Register

	7	6	5	4	3	2	1	0	
EEPCR ($0FE0_H$)	EEPMD					EEPRS	ATPWDW	MNPWDW	(Initial value: 1100 *011)

			Program Execution Area	
			RAM/ BOOT	FLASH
EEPMD	FLASH write enable control (Write protect).	1100: FLASH write disable. 0011: FLASH write enable. Other values: Reserved.	R/W	Read only
EEPRS	FLASH write forcible stop.	0: – 1: FLASH writing is forced to stop. (The write data counter is initialized.) * After writing “1” to EEPRS, it is automatically cleared to “0”.		
ATPWDW	Automatic power control for the FLASH control circuit in the IDLE0/1/2, SLEEP0/1/2 modes. (This bit is available only when MNPWDW is set to “1”.)	0: Automatic power shut down is executed in IDLE0/1/2 and SLEEP0/1/2 modes. 1: Automatic power shut down is not executed in IDLE0/1/2 and SLEEP0/1/2 modes. (The power is always supplied in these modes.)		R/W
MNPWDW	Software-based power control for the FLASH control circuit.	0: The power for the FLASH control circuit is turned off. 1: The power for the FLASH control circuit is turned on.		Read only

Note 1: The EEPMD, EEPRS, and MNPWDW can be rewritten only when a program fetch is taking place in the RAM or BOOT-ROM area. If an attempt is made to rewrite the EEPCR register when a program is being executed in the FLASH area, the EEPMD, EEPRS, and MNPWDW keep holding the previous data; they are not rewritten.

Note 2: To write to the FLASH, set the EEPMD with “0011B” in advance when a program fetch is taking place in the RAM area.

Note 3: To forcibly stop writing of FLASH, set the EEPRS to “1” when a program fetch is taking place in the RAM area.

Note 4: The ATPWDW functions only if the MNPWDW is “1”. If the MNPWDW is “0”, the power for the FLASH control circuit is kept turned off regardless of the setting of the ATPWDW.

Note 5: When a STOP mode is executed, the power for the FLASH control circuit is turned off regardless of the setting of the ATPWDW. If the MNPWDW is “0”, entering/exiting the STOP mode allows the power for the FLASH control circuit to be kept turned off.

Note 6: Executing a read instruction to the EEPCR register results in bit3 being read as undefined. Bit2 is always read as “0”.

Note 7: The following attention is necessary when the MNPWDW is set or cleared.

When the MNPWDW is changed from “1” to “0”	Clear the interrupt master enable flag (IMF) to “0” in advance to disable an interrupt. After that, do not set IMF to “1” during EEPSR<EWUPEN> = “0”. If a watchdog timer is in use, clear the binary counter for the watchdog timer just before MNPWDW is changed from “1” to “0”.
When the MNPWDW is changed from “0” to “1”	When write to or read from the Flash area, make sure that the EEPSR<EWUPEN> is “1” by software. Once the MNPWDW is rewritten from “0” to “1” by software, keep performing software-based polling until the EEPSR<EWUPEN> becomes “1”.

Note 8: In MCU mode, the EEPMD and EEPRS should be set to “1100B” and “0”.

Figure 1.2.2 FLASH Control Register

FLASH Status Register

	7	6	5	4	3	2	1	0	
EEPSR ($0FE1_H$)						WINT	EWUPEN	BFBUSY	(Initial value: **** *010)

WINT	Interrupt detection during a write to the FLASH		0: Not detected 1: Detected (Interrupt occurred) * WINT is automatically cleared to "0" when read instruction is executed to EEPSR.			Read only
EWUPEN	FLASH control circuit status monitor	Control circuit status	Operating (Power on)		Halt (Power off) or warm-up	
		FLASH status	Temporary data buffer empty	Writing	Disable	
			1	1	0	
BFBUSY	FLASH write busy flag		0	1	1	

Note 1: If a nonmaskable interrupt occurs during a write to the FLASH, the WINT is set to "1" and the writing is discontinued, and then warm-up period (CPU wait) for the control circuit of Flash memory is executed. (The write data counter is initialized.) If WINT = "1" is detected in the nonmaskable interrupt service routine, a write is not completed successfully. So, it is necessary to try a write again. The content of the page to which a write is taking place may be changed to an unexpected value depending on the timing when the WINT becomes "1".

Note 2: Even if a nonmaskable interrupt occurs during an FLASH warm-up , the CPU stays at a halt until the warm-up is finished.

Note 3: The WINT is automatically cleared to "0" when a read instruction is executed to the EEPSR register.

Note 4: When MNPWDW is changed from "0" to "1", EWUPEN becomes "1" after taking 2^{10}/fc [s] (if SYSCK = "0") or 2^3/fs [s] (if SYSCK = "1"). Before accessing the FLASH, make sure that the EWUPEN is "1" in the RAM area.

Note 5: If the BFBUSY is "1", executing a read instruction or fetch to the FLASH area causes FFH to be read. Fetching FFH results in a software interrupt occurring.

Note 6: In the TMP86CM25A, if the EWUPEN is "1", writing to the masked ROM area that corresponds to the FLASH area does not set the BFBUSY of the TMP86CM25A to "1".

Figure 1.2.3 FLASH Status Register

1.2.3 FLASH Write Enable Control (EEPCR<EEPMD>)

In the FLASH product, the control register can be used to disable a write to the FLASH (Write protect) in order to prevent a write to the FLASH from occurring by mistake because of a program error or microcontroller malfunction. To enable a write to the FLASH, set the EEPCR<EEPMD> with 0011B. To disable a write to the FLASH, set the EEPCR<EEPMD> with 1100B. A reset initializes the EEPCR<EEPMD> to 1100B to disable a write to the FLASH. Usually, set the EEPCR<EEPMD> with 1100B, except when it is necessary to write to the FLASH.

Note 1: The FLASH memory (8000H to FFFFH) can be written only in the serial PROM mode.

Note 2: The EEPCR<EEPMD> can be rewritten only when a program is being executed in the RAM area. Executing a write instruction to the EEPCR<EEPMD> in the FLASH area does not change its setting.

Note 3: In the TMP86CM25A, executing a write instruction to the EEPCR<EEPMD> changes its setting; however, the new setting does not take effect.

Note 4: This function can be used in serial PROM mode. In MCU mode, the EEPCR<EEPMD> should be always set to “1100B”.

1.2.4 FLASH Write Forcible Stop (EEPCR<EEPRS>)

To forcibly stop a write to the FLASH, set the EEPCR<EEPRS> to "1". Setting the EEPCR<EEPRS> to "1" initializes the write data counter of data buffer and forcibly stops a write, and then a warm-up period (CPU wait) for the control circuit of Flash memory is executed. After warm-up period, the EEPSR<BFBUSY> is cleared to "0". The warm-up period is 2^{10}/fc (SYSCK = "0") or 2^{3}/fs (SYSCK = "1"). After this, if writing to FLASH starts again, data is stored as the first byte of the temporary data buffer and sets the EEPSR<BFBUSY> to "1". Therefore, it is necessary to write 64 bytes data to the temporary data buffer.

After 1 to 63 bytes are saved to the temporary data buffer, if the EEPCR<EEPRS> is set to "1" the specified page of flash is not written. (It keeps previous data.)

Note 1: After 64 bytes are written to the temporary data buffer, the setting the EEPCR<EEPRS> to "1" may cause the writing the page of FLASH to an unexpected value.

Note 2: The EEPCR<EEPRS> can be rewritten only when a program is being executed in the RAM area. In the FLASH area, executing a write instruction to the EEPCR<EEPRS> does not affect its setting.

Note 3: During the warm-up period for Flash memory (CPU wait), the peripheral circuits continue operating, but the CPU stays at a halt until the warm-up is finished. Even if an interrupt latch is set to "1" by generating of interrupt request, an interrupt sequence doesn't start till the end of warm up. If interrupts occur during a warm-up period with IMF = "1", the interrupt sequence which depends on interrupt priority will start after warm-up period.

Note 4: When the EEPCR<EEPRS> is set to "1" with EEPSR<BFBUSY> = "0", a warm-up period is not executed.

Note 5: If executed a write or read instruction to the Flash area immediately after setting EEPCR<EEPRS>, insert one or more machine cycle instructions after setting EEPCR<EEPRS>.

Example: Reads the Flash memory data immediately after setting EEPCR<EEPRS> to "1".

```
LD      HL,8000H
LD      (EEPCR),3FH     ; Set EEPCR<EEPRS> to "1".
NOP                     ; NOP
                          (Do not execute read instruction immediately after setting
                          EEPCR<EEPRS>.)
LD      A,(HL)          ; Reads the data of address 8000H.
                          (Read instruction to the Flash memory.)
```

Note 6: This function can be used in serial PROM mode. In this mode, the EEPCR<EEPRS> should be always set to "0".

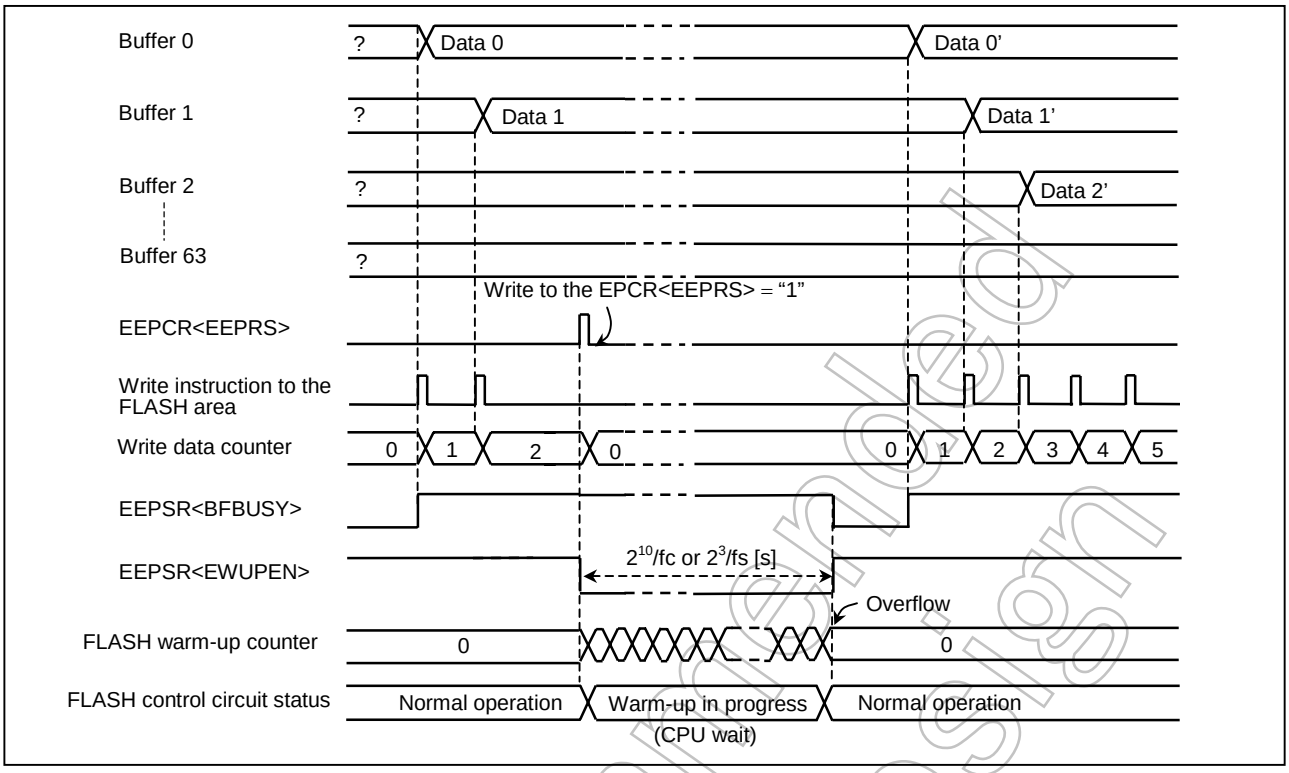


Figure 1.2.4 Write Data Counter Initialization and Write Forcible Stop

1.2.5 Power Control for the FLASH Control Circuit

For the FLASH product, it is possible to turn off the power for FLASH control circuit (such as a regulator) to suppress power consumption if the FLASH area is not accessed. For the TMP86CM25A, the register setting and the CPU wait functions behave in the same manner as for the FLASH product to maintain compatibility; however, power consumption is not suppressed.

The EEPCR<MNPWDW> and EEPCR<ATPWDW> are used to control the power for the FLASH control circuit. If the power for the FLASH control circuit is turned off according to the setting of these registers, starting to use the circuits again needs to allow warm-up time for the power supply.

Table 1.2.1 Power Supply Warm-up Time (CPU wait) for the FLASH Control Circuit

NORMAL1/2 IDLE0/1/2 Mode	SLOW1/2 SLEEP0/1/2 Mode	STOP Mode (when EEPCR<MNPWDW> = "1")	
		To Return to a NORMAL Mode	To Return to a SLOW Mode
2^{10}/fc [s] (64 μs at 16 MHz)	2^{3}/fs [s] (244 μs at 32.768 kHz)	STOP warm-up time + 2^{10}/fc [s]	STOP warm-up time + 2^{3}/fs [s]

1.2.5.1 Software-based Power Control for the FLASH Control Circuit (EEPCR<MNPWDW>)

The EEPCR<MNPWDW> is a software-based power control bit for the FLASH control circuit. When a program is being executed in the RAM area, setting this bit enables software-based power control. Clearing the EEPCR<MNPWDW> to "0" immediately turns off the power for the FLASH control circuit. Once the EEPCR<MNPWDW> is switched from "0" to "1", before attempting a read or fetch from the FLASH area, it is necessary to insert a warm-up period by software until the power supply is stabilized. In this case, because the CPU wait is not executed, any other instructions except accessing to Flash (write or read) are available. When MNPWDW is changed from "0" to "1", EWUPEN becomes "1" after taking 2^{10}/fc [s] (SYSCK = "0") or 2^{3}/fs [s] (SYSCK = "1"). Usually software-based polling should be performed until the EEPSR<EWUPEN> becomes "1". An example of setting is given below.

(1) Example of controlling the EEPCR<MNPWDW>

1. Transfer a program for controlling the EEPCR<MNPWDW> to the RAM area.
2. Release an address trap in the RAM area (set up the WDTCR1 and WDTCR2 registers).
3. Jump to the control program transferred to the RAM area.
4. Clear the interrupt master enable flag (IMF ← "0").
5. Clear the binary counter if the watchdog timer is in use.
6. To turn off the power for the FLASH control circuit, clear the EEPCR<MNPWDW> to "0".
7. Perform CPU processing as required.
8. To access the FLASH area again, set the EEPCR<MNPWDW> to "1".
9. Keep program polling until the EEPSR<EWUPEN> becomes "1".
   (Upon completion of an FLASH warm-up, the EEPSR<EWUPEN> is set to "1". It takes 2^{10}/fc (SYSCK = "0") or 2^{3}/fs (SYSCK = "1") until EWUPEN becomes "1".)

This procedure enables the FLASH area to be accessed.

If the EEPCR<MNPWDW> is “1”, entering a STOP mode forcibly turns off the power for the FLASH control circuit. When the STOP mode is released, a STOP mode oscillation warm-up is carried out, and then the CPU wait period (warm-up for stabilizing of FLASH power supply circuit) is automatically performed. If the EEPCR<MNPWDW> is “0”, entering/exiting the STOP mode keeps the power for the FLASH control circuit turned off.

Note 1: If the EEPSR<EWUPEN> is “0”, do not access (Fetch, read, or write) the FLASH area. Executing a read instruction or fetch to the FLASH area causes FFH to be read. Fetching FFH results in a software interrupt occurring. For the TMP86CM25A, however, masked ROM data is always read regardless of the state of the EEPSR<EWUPEN>.

Note 2: To clear the EEPCR<MNPWDW> to “0”, clear the interrupt master enable flag (IMF) to “0” in advance to disable an interrupt. After that, do not set IMF to “1” during EEPSR<EWUPEN> = “0”.

Note 3: If the EEPCR<MNPWDW> is “0”, generating a nonmaskable interrupt automatically rewrites the MNPWDW to “1” to warm-up the FLASH control circuit (CPU wait). That time, the peripheral circuits continue operating, but the CPU stays at a halt until the warm-up is finished.

Note 4: The EEPCR<MNPWDW> can be rewritten only when a program is being executed in the RAM area. In the FLASH area, executing a write instruction to the EEPCR<MNPWDW> does not affect its setting.

Note 5: If a watchdog timer is used as an interrupt request, clear the binary counter for the watchdog timer just before MNPWDW is changed from “1” to “0”.

Note 6: During the warm-up period with a software polling of EEPSR<EWUPEN>, if a nonmaskable interrupt occurs during an FLASH warm-up, the CPU stays at a halt until the warm-up is finished.

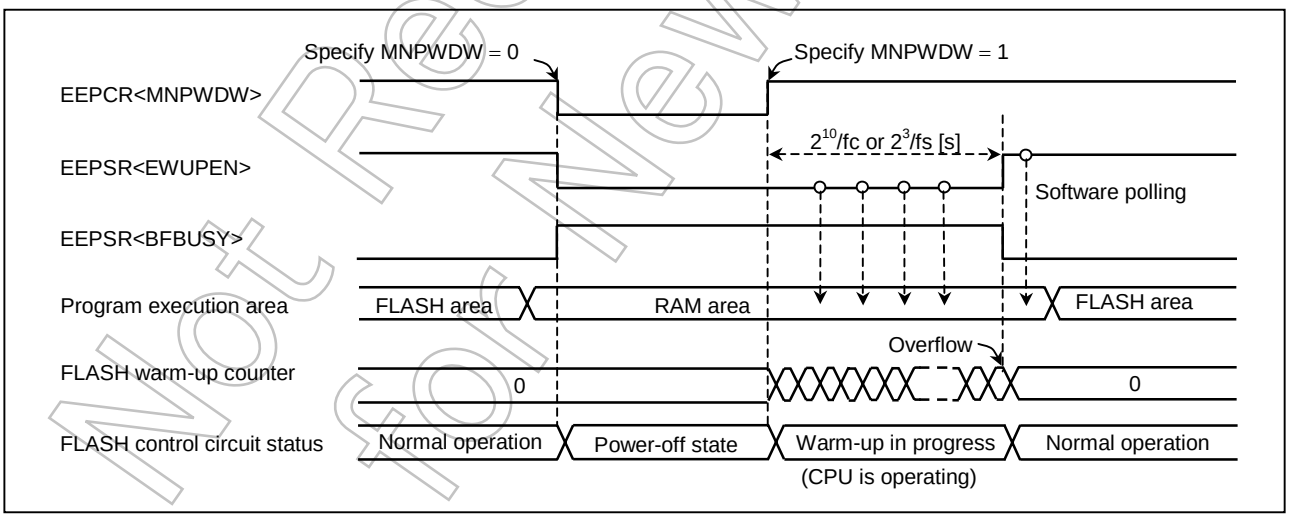


Figure 1.2.5 Software-based Power Control for the FLASH Control Circuit (EEPCR<MNPWDW>)

Example: Performing software-based power control for the FLASH control circuit

```
sRAMAREA:
            DI                          ;   Disable an interrupt (IMF ← “0”).
            LD      (WDTCR2),4Eh        ;   Clear the binary counter if the watchdog timer
                                        ;   is in use.
            CLR     (EEPCR).0           ;   Clear the EEPCR<MNPWDW> to “0”.
             ¦       ¦
sLOOP1:     SET     (EEPCR).0           ;   Set the EEPCR<MNPWDW> to “1”.
            TEST    (EEPSR).1           ;   Monitor the EEPSR<EWUPEN> register.
            JRS     T,sLOOP1            ;   Jump to sLOOP1 if EEPSR<EWUPEN> = “0”.
            JP      MAIN                ;   Jump to the FLASH area.
```

1.2.5.2 Automatic Power Control for the FLASH Control Circuit (EEPCR<ATPWDW>)

The EEPCR<ATPWDW> is an automatic power control bit for the FLASH control circuit. It is possible to suppress power consumption by automatically shutting down the power for the FLASH control circuit when an operation mode is changed to IDLE0/1/2 and SLEEP0/1/2 modes. This bit can be specified regardless of the area in which a program is being executed.

After the EEPCR<ATPWDW> is cleared to "0", entering an operation mode (IDLE0/1/2 or SLEEP0/1/2) where the CPU is at a halt automatically turns off the power for the FLASH control circuit. Once the operation mode is released, the warm-up time (CPU wait) is automatically counted to resume normal processing. The CPU wait period is either 2^{10}/fc (SYSCK = "0") or 2^3/fs (SYSCK = "1"). If the EEPCR<ATPWDW> is "1", releasing the operation mode does not cause the CPU wait.

If EEPCR<MNPWDW> = "1", executing a STOP mode forcibly turns off the power for the FLASH control circuit regardless of the setting of the EEPCR<ATPWDW>. When the STOP mode is released, a STOP mode oscillation warm-up is carried out, and then an FLASH control circuit warm-up (CPU wait) is automatically performed. If the EEPCR<MNPWDW> is "0", entering/exiting a STOP mode allows the power for the FLASH control circuit to be kept turned off.

Note 1: The EEPCR<ATPWDW> functions only if the EEPCR<MNPWDW> is "1". If the EEPCR<MNPWDW> is "0", the power for the FLASH control circuit is kept turned off when an operation mode is executed or released.

Note 2: During an FLASH warm-up (CPU wait), the peripheral circuits continue operating, but the CPU stays at a halt. Even if an interrupt latch is set under this condition, no interrupt process occurs until the CPU wait is completed. If the IMF is "1" when the interrupt latch is set, interrupt process takes place according to the interrupt priority after the CPU has started operating.

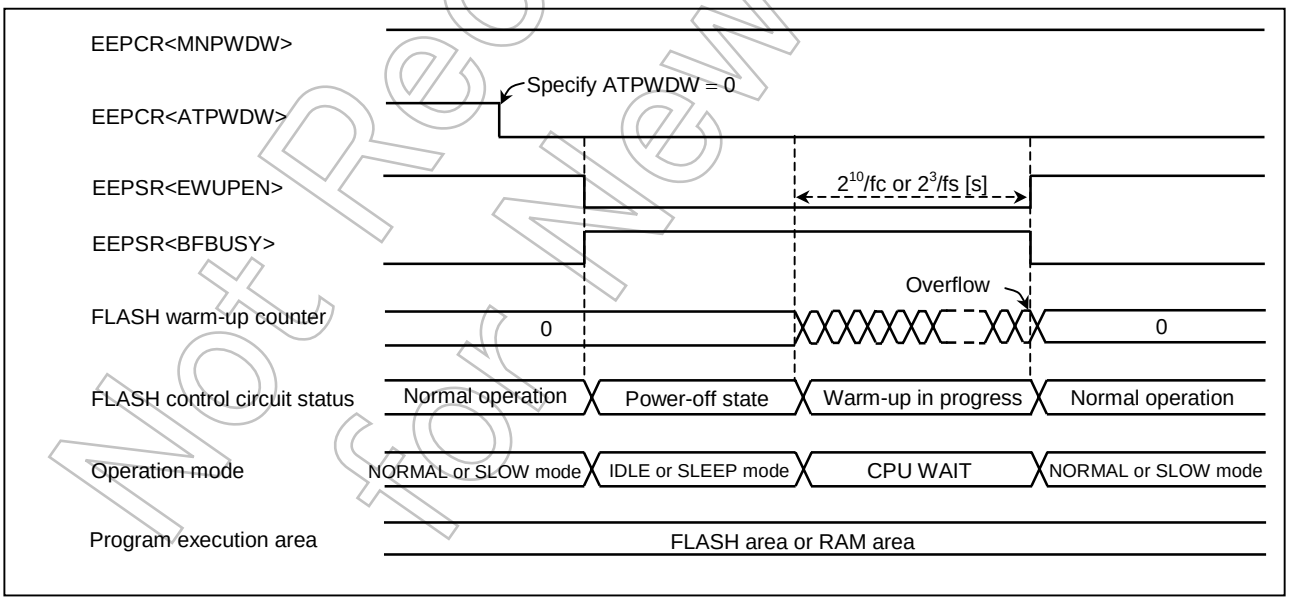


Figure 1.2.6 Automatic Power Control for the FLASH Control Circuit (EEPCR<ATPWDW>)

1.2.6 Accessing to the FLASH Memory

During the writing to the FLASH area, neither a read nor fetch can be performed for the 8000H to FFFFH area. Therefore, to write the FLASH area, the program should be executed in the BOOTROM or RAM area. Basically, to write the FLASH area, the program can be executed in BOOTROM area by using the FLASH writing mode of the Serial PROM mode, but it can be also executed any user program in RAM area by using the RAM loader mode of the Serial PROM mode.

Explanation here is made of only the method of FLASH programming in RAM area. For detail about each operation mode of the Serial PROM mode, refer to 2.1 “Serial PROM Mode”.

Although the writing to FLASH is executed on page-by-page, the reading from FLASH is executed on byte-by-byte.

If a nonmaskable interrupt occurs during a write to the FLASH (EEPSR<BFBUSY> = “1”), the WINT is set to “1” and the writing is discontinued, and then the warm-up period for control circuit of Flash memory is executed (The write data counter is also initialized). If WINT = “1” is detected in the nonmaskable interrupt service routine, a write is not completed successfully. So, it is necessary to try a write again. The warm-up period is 2^{10}/fc (SYSCK = “0”) or 2^{3}/fs (SYSCK = “1”). After 1 to 63 bytes are saved to the temporary data buffer, if an interrupt generates, the specified page of FLASH is not written. (It keeps previous data.)

Note 1: Writing to the FLASH area is enabled only in serial PROM mode. For details of serial PROM mode, refer to 2.1 “Serial PROM Mode”.

Note 2: After 64 bytes are written to the temporary data buffer, the generating of an interrupt may cause the writing the page of FLASH to an unexpected value.

Note 3: During the warm-up period for Flash memory (CPU wait), the peripheral circuits continue operating, but the CPU stays at a halt until the warm-up is finished. Even if an interrupt latch is set to “1” by generating of interrupt request, an interrupt sequence doesn’t start till the end of warm-up. If interrupts occur during a warm-up period with IMF = “1”, the interrupt sequence which depends on interrupt priority will start after warm-up period.

Note 4: When write the data to Flash memory from RAM area, disable all the non-maskable interrupt by clearing interrupt master enable flag (IMF) to “0” beforehand.

1.2.6.1 FLASH Writing Program in the RAM Area

To develop the program in RAM, the write control program should be loaded from external device by using RAM loader mode in Serial PROM mode. Given below is an example of writing the control program in the RAM area.

(1) Example of writing program in the RAM area

1. Monitor the EEPSR<EWUPEN>. If it is “0”, set the EEPCR<MNPWDW> to “1”, and then start and keep polling until the EEPSR<EWUPEN> becomes “1”.
2. Clear the interrupt master enable flag (IMF ← “0”).
3. Set the EEPCR with “3BH” (to enable a write to the FLASH).
4. Execute a write instruction for 64 bytes to the FLASH area.
5. Start and keep polling by software until the EEPSR<BFBUSY> becomes “0”. (Upon completion of an erase and write to the FLASH cells, the EEPSR<BFBUSY> is set to “1”. For the FLASH product, the required write time is typically 4 ms. For the emulation chip, it is the value specified in the EEPEVA register.)
6. Set the EEPCR with “CBH” (to disable a write to the FLASH).

Note: See (2), “Method of specifying an address for a write to the FLASH”, for a description about the FLASH address to be specified at step 4 above.

(2) Method of specifying an address for a write to the FLASH

The FLASH page to be written is specified by the 10 high-order bits of the address of the first byte data. The first byte data is stored at the first address of the temporary data buffer. If the data to be written is, for example, 8040H, page 1 is selected, and the data is stored at the first address of the temporary data buffer. Even if the 6 low-order bits of the specified address is not 000000B, the first byte data is always stored at the first address of the data buffer.

Any address can be specified as the second and subsequent address within FLASH area (8000H to FFFFH). The write data bytes are stored in the temporary data buffer in the sequence they are written, regardless of what address is specified. Usually, the address that is the same as the first byte is specified for the second and subsequent address. A 16-bit transfer instruction (LDW) can also be used for writing to the temporary data buffer.

Example: Data bytes 00H to 3FH are written to page 1.
(Figure 1.2.9 shows the example of data buffer and pages.)

```
        DI                          ;  Disable an interrupt (IMF ← "0").
        LD      C,00H
        LD      HL,EEPCR            ;  Specify the EEPCR register address.
        LD      IX,8040H            ;  Specify a write address.
        LD      (HL),3BH            ;  Specify the EEPCR.
sLOOP1:
        LD      (IX),C              ;  Store data to the temporary data buffer.
                                       (A write page is selected when the first
                                       byte is written.)
        INC     C                   ;  C = C + 1.
        CMP     C,40H               ;  Jump to sLOOP1 if C is not 40H.
        JR      NZ,sLOOP1
sLOOP2:
        TEST    (EEPSR).0
        JRS     F,sLOOP2            ;  Jump to sLOOP2 if EEPSR<BFBUSY> =
                                       "1".
        LD      (HL),0CBH           ;  Specify the EEPCR.
```

Note: If the BFBUSY is "1", executing a read instruction or fetch to the FLASH area causes "FFH" to be read. Fetching "FFH" results in a software interrupt occurring.

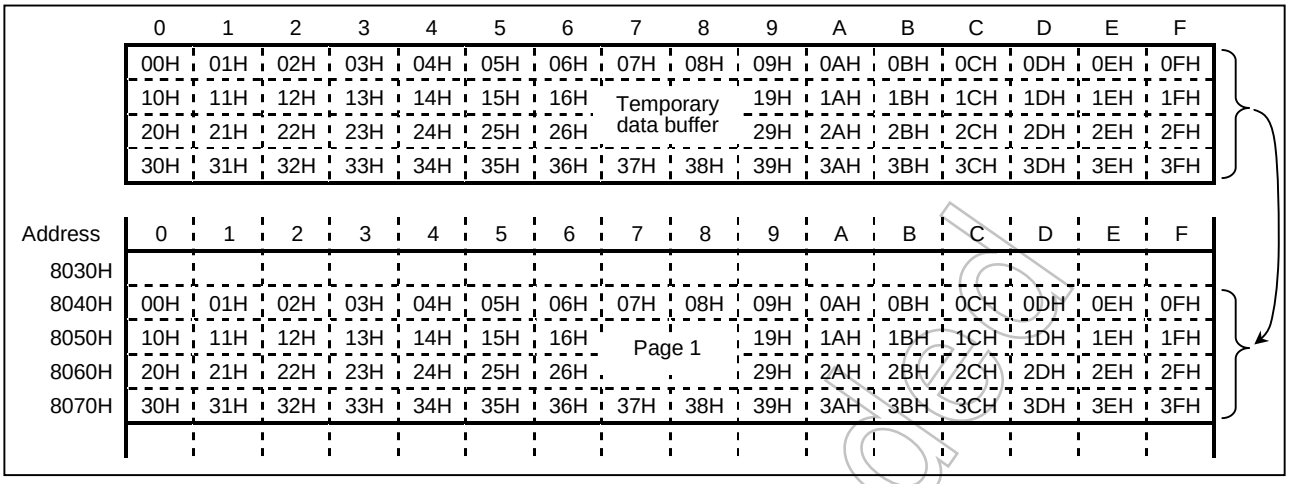


Figure 1.2.7 Data Buffer and Write Page (Example)

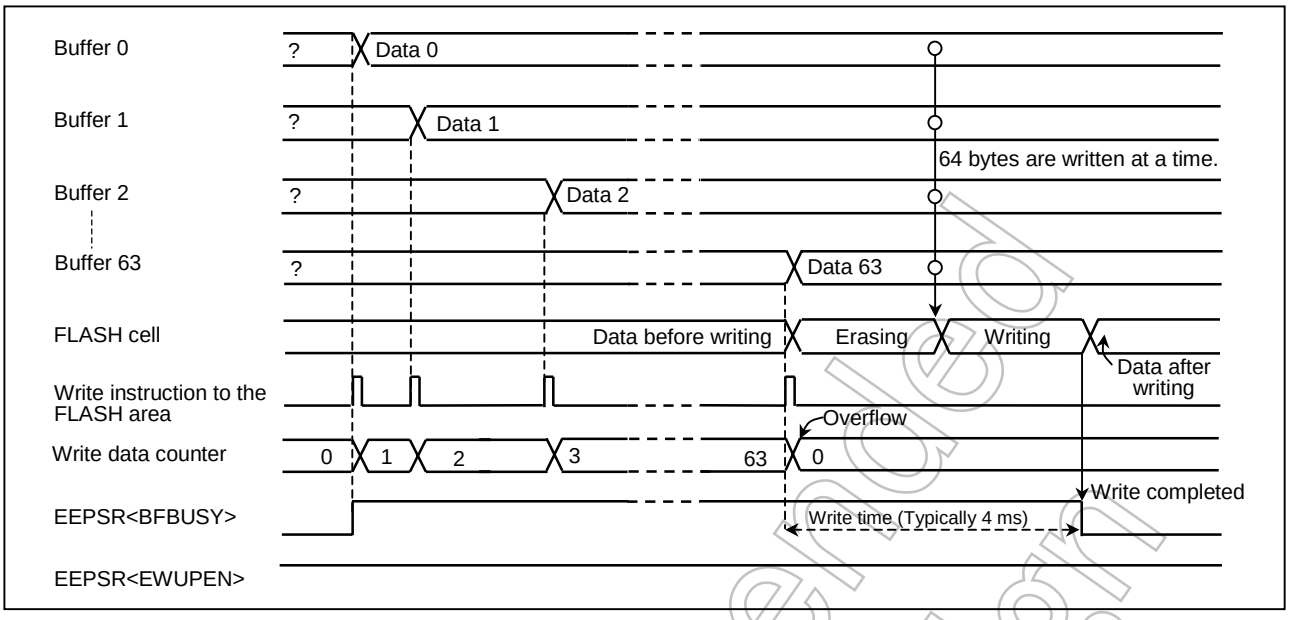


Figure 1.2.8 Write to the FLASH Area

2.1 Serial PROM Mode

2.1.1 Outline

The TMP86FM25 has a 2-Kbyte BOOT-ROM for programming to FLASH memory. This BOOT-ROM is a mask ROM that contains a program to write the FLASH memory on-board. The BOOT-ROM is available in a serial PROM mode and it is controlled by P11 pin, BOOT (P15) pin, TEST pin and $\overline{\text{RESET}}$ pin, and is communicated via TXD (P16) and RXD (P15) pins. There are four operation modes in a serial PROM mode: FLASH writing mode, RAM loader mode, FLASH memory SUM output mode and Product discrimination code output mode. Operating area of serial PROM mode differs from that of MCU mode. The operating area of serial PROM mode shows in Table 2.1.1.

Table 2.1.1 Operating Area of Serial PROM Mode

Parameter	Min	Max	Unit
Operating voltage	2.7	3.6	V
High frequency (Note)	2	16	MHz
Temperature	25 ± 5		°C

Note: Even though included in above operating area, part of frequency can not be supported in serial PROM mode. For details, refer to Table 2.1.6.

2.1.2 Memory Mapping

The BOOT-ROM is mapped in address 3800_H to $3FFF_H$. The Figure 2.1.1 shows a memory mapping.

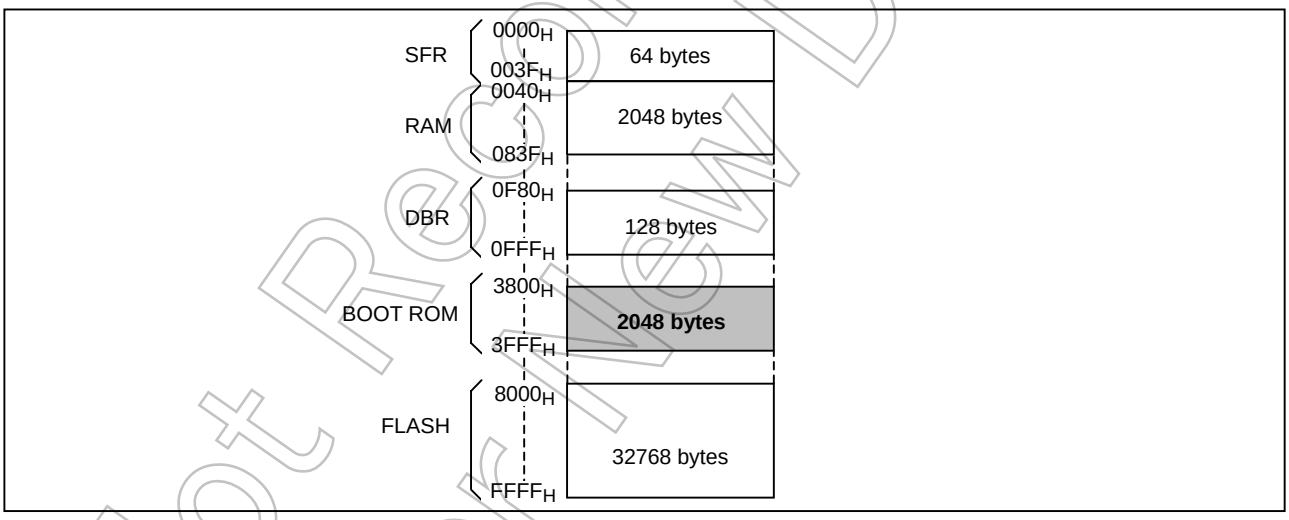


Figure 2.1.1 Memory Address Maps

2.1.3 Serial PROM Mode Setting

2.1.3.1 Serial PROM Mode Control Pins

To execute on-board programming, start the TMP86FM25 in serial PROM mode. Setting of a serial PROM mode is shown in Table 2.1.2.

Table 2.1.2 Serial PROM Mode Setting

Pin	Setting
BOOT/RXD pin (P15)	High
P11 pin	Low
$\overline{\text{RESET}}$, TESTpin	↗

2.1.3.2 Pin Function

In the serial PROM mode, TXD (P16) and RXD (P15) pins are used as a serial interface pin.

Table 2.1.3 Pin Function in the Serial PROM Mode

Pin Name (Serial PROM mode)	Input/ Output	Function		Pin Name (MCU mode)
TXD	Output	Serial data output	(Note 1)	P16
RXD/BOOT	Input	Serial PROM mode control/Serial data input		P15
$\overline{\text{RESET}}$	Input	Serial PROM mode control		$\overline{\text{RESET}}$
TEST	Input	Serial PROM mode control		TEST
P11	Input	Serial PROM mode control (Fix to “L” level)		P11
VDD	Power supply	2.7 V to 3.6 V		
VSS		0 V		
VAREF		Open or equal with VDD		
P10, P12 to P14, P17	I/O	Placed in High-Z state during serial PROM mode.		
P20 to P22				
P30 to P36				
P50 to P57				
P60 to P67				
P70 to P77				
SEG39 to SEG0	Output	Open		
COM4 to COM0				
C0, C1, V4 to V1	LCD voltage booster pin			
XIN	Input	Resonator connecting pins for high-frequency clock. For inputting external clock, XIN is used and XOUT is opened.		(Note 2)
XOUT	Output			

Note 1: When the device is used as on-board writing and other parts are already mounted in place, be careful no to affect these communication control pins.

Note 2: Operating area of high frequency in serial PROM mode is from 2 MHz to 16 MHz.

To set a serial PROM mode, connect device pins as shown in Figure 2.1.2.

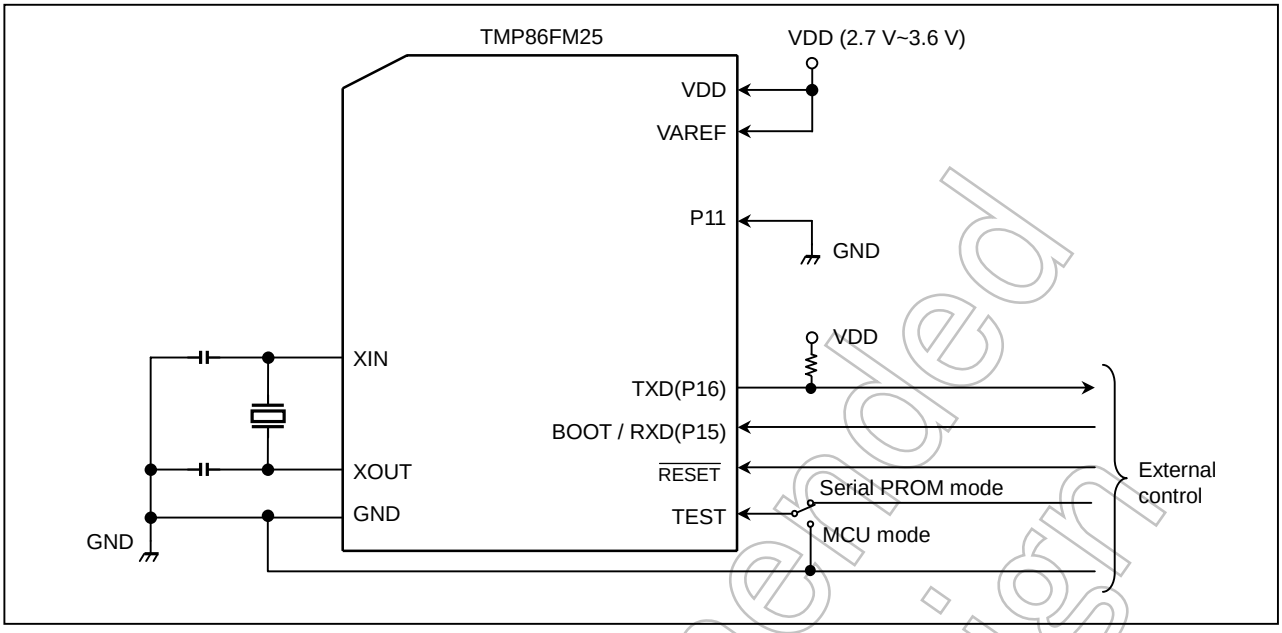


Figure 2.1.2 Serial PROM Mode Port Setting

2.1.3.3 Activating Serial PROM Mode

The following is a procedure of setting of serial PROM mode. Figure 2.1.3 shows a serial PROM mode timing.

(1) Turn on the power to the VDD pin.
(2) Set the P11 pin, TEST pin and $\overline{\text{RESET}}$ pin to low level.
(3) Set the BOOT/RXD pin (P15) to high level.
(4) Wait until the power supply and clock sufficiently stabilize.
(5) Set the TEST pin from low level to high level.
(6) Release the $\overline{\text{RESET}}$. (Set to high level)
(7) Input a matching data (5AH) to RXD pin after waiting for setup sequence.

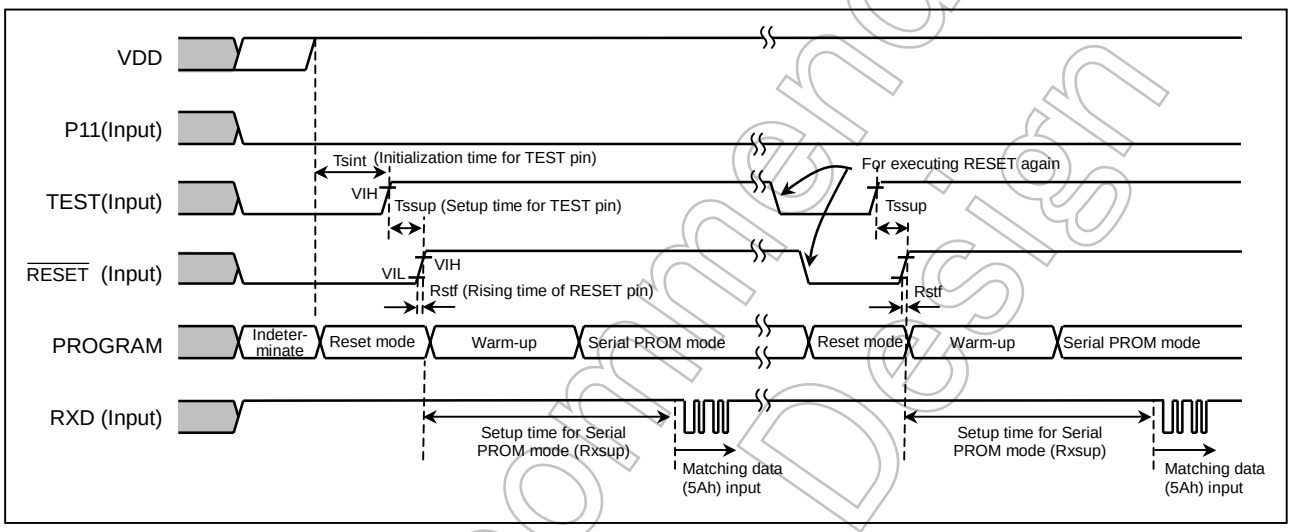


Figure 2.1.3 Serial PROM Mode Timing

Table 2.1.4 Serial PROM Mode Timing characteristics

Parameter		Symbol	The Number of Clock (fc)	Required Minimum Time	
				at fc = 2 MHz	at fc = 16 MHz
Setup time for TEST pin	Rstf > 512 / fc [s]	Tssup	-	1 ms	
	Rstf < 512 / fc [s]		-	0 *Note1	
Initialization time for TEST pin		Tsint	-	1ms	
Time from reset release until acceptance of start bit of RXD pin		RXsup	110000	55 ms	6.9 ms

Note 1: If Rstf is shorter than 512 / fc[s] due to using CMOS-type reset IC or Logic IC, the TEST pin can input the same pulse as the $\overline{\text{RESET}}$ pin input. (TEST pin can be directly connected to the $\overline{\text{RESET}}$ pin.) However, drive the pins carefully not to affect the pin's input level, as the TEST pin and the $\overline{\text{RESET}}$ pin have pull-down resistor and pull-up resistor built-in.

Note 2: fc; High-frequency clock

2.1.3.4 Examples of On-board writing

Figure 2.1.4 shows examples of On-board writing.

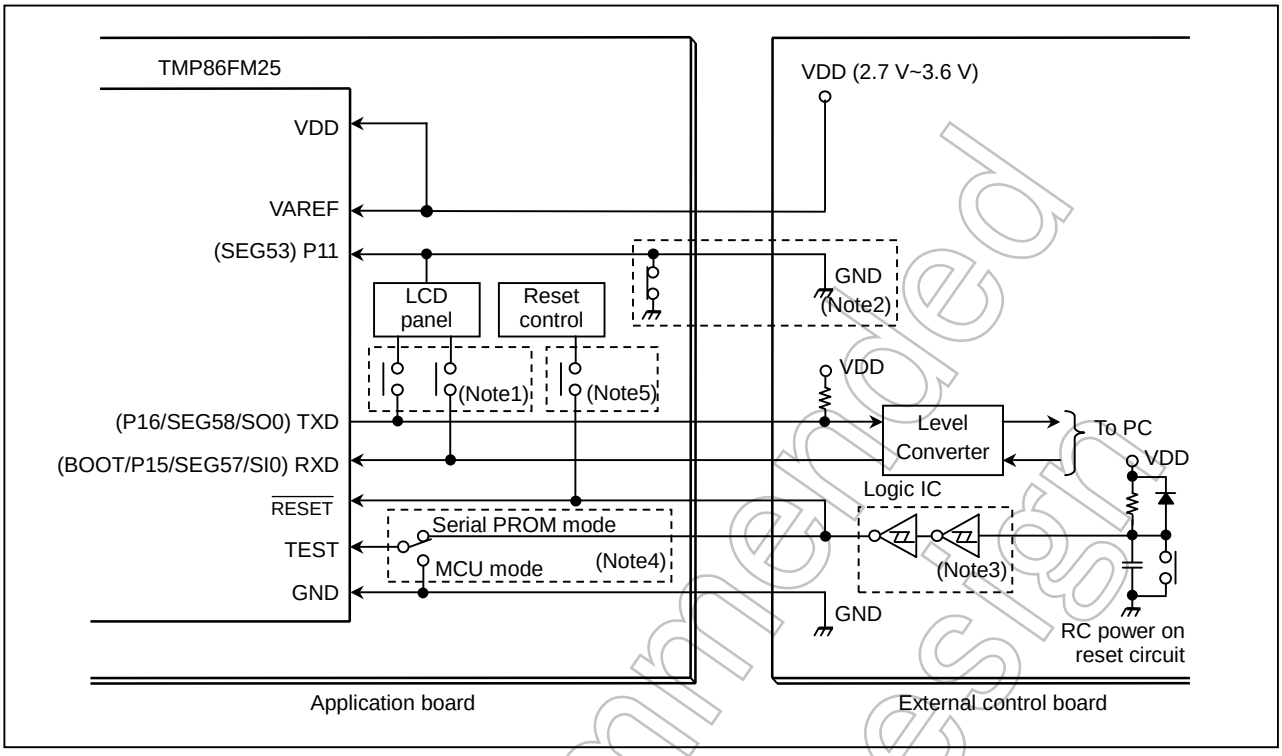


Figure 2.1.4 Examples of Onboard writing

Note 1: If capacity for LCD panel and other devices on the application board affect UART communication in Serial PROM mode, disconnect these pins by using a jumper or a switch.

Note 2: Set the P11 pin to GND. There are two ways. Set P11 pin to GND on the external board, or set it to GND by setting a jumper on the application board.

Note 3: If input signal has analog delay due to the use of such circuit as RC power on reset circuit, connect both TEST pin and $\overline{\text{RESET}}$ pin to logic ICs (Schmitt input IC such as TC74HC14). In this case, control the pin capacity to require the condition Rstf<512/fc[s].

Note4: In MCU mode, the TEST pin can be disconnected as it has a pull-down resistor built-in. However, we recommend connecting it to GND level to avoid noise influence.

Note5: If the RESET control circuit on the application board affects the Serial PROM mode to start, disconnect it by using a jumper, etc.

2.1.4 Interface Specifications for UART

The following shows the UART communication format used in serial PROM mode.

Before on-board programming can be executed, the communication format on the external controller side must also be set up in the same way as for this product.

Note that although the default baud rate is 9600 bps, it can be changed to other values as shown in Table 2.1.5. The Table 2.1.6 shows an operating frequency and baud rate in serial PROM mode. Except frequency which is not described in Table 2.1.6 can not use in serial PROM mode.

Baud rate (Default): 9600 bps
Data length: 8 bits
Parity addition: None
Stop bit length: 1 bit

Table 2.1.5 Baud Rate Modification Data

Baud rate modification data	04H	05H	06H	07H	0AH	18H	28H
Baud rate (bps)	76800	62500	57600	38400	31250	19200	9600

Table 2.1.6 Operating Frequency and Baud Rate in Serial PROM Mode

(Note 3)	Reference Baud Rate (bps)		76800		62500		57600		38400		31250		19200		9600	
	Baud Rate Modification Data		04H		05H		06H		07H		0AH		18H		28H	
	Reference Frequency (MHz)	Area (MHz)	Baud Rate (bps)	(%)	(bps)	(%)	(bps)	(%)	(bps)	(%)	(bps)	(%)	(bps)	(%)	(bps)	(%)
1	2	1.91~2.10	–	–	–	–	–	–	–	–	–	–	–	–	9615	+0.16
2	4	3.82~4.19	–	–	–	–	–	–	–	–	31250	0.00	19231	+0.16	9615	+0.16
	4.19	3.82~4.19	–	–	–	–	–	–	–	–	32734	+4.75	20144	+4.92	10072	+4.92
3	4.9152	4.70~5.16	–	–	–	–	–	–	38400	0.00	–	–	19200	0.00	9600	0.00
	5	4.70~5.16	–	–	–	–	–	–	39063	+1.73	–	–	19531	+1.73	9766	+1.73
4	6	5.87~6.45	–	–	–	–	–	–	-	–	–	–	–	–	9375	–2.34
	6.144	5.87~6.45	–	–	–	–	–	–	-	–	–	–	–	–	9600	0.00
5	7.3728	7.05~7.74	–	–		–	57600	0.00	-	–	–	–	19200	0.00	9600	0.00
6	8	7.64~8.39	–	–	62500	0.00	-	–	38462	+0.16	31250	0.00	19231	+0.16	9615	+0.16
7	9.8304	9.40~10.32	76800	0.00	–	–	-	–	38400	0.00	–	–	19200	0.00	9600	0.00
	10	9.40~10.32	78125	+1.73	–	–	-	–	39063	+1.73	–	–	19531	+1.73	9766	+1.73
8	12	11.75~12.90	–	–	–	–	57692	+0.16	-	–	31250	0.00	18750	–2.34	9375	–2.34
	12.288	11.75~12.90	–	–	–	–	59077	+2.56	-	–	32000	+2.40	19200	0.00	9600	0.00
	12.5	11.75~12.90	–	–	60096	–3.85	60096	+4.33	-	–	30048	–3.85	19531	+1.73	9766	+1.73
9	14.7456	14.10~15.48	–	–	–	–	57600	0.00	38400	0.00	–	–	19200	0.00	9600	0.00
10	16	15.27~16.77	76923	+0.16	62500	0.00	-	–	38462	+0.16	31250	0.00	19231	+0.16	9615	+0.16

Note 1: “Reference Frequency” and “Area” show the high-frequency area supported in serial PROM mode. Except the above frequency can not be supported in serial PROM mode even though the high frequency is included in area from 2 MHz to 16 MHz.

Note 2: The total error of frequency must be kept within +/–3% so that the auto-detection of frequency is executed correctly.

Note 3: An external controller should transmit a matching data repeatedly till the TMP86FM25 transmit an echo back data. Above number indicates a transmission number of times of matching data till transmission of echo back data.

2.1.5 Command

There are five commands in serial PROM mode. After reset release, the TMP86FM25 waits a matching data (5AH).

Table 2.1.7 Command in Serial PROM Mode

Command Data	Operation Mode	Remarks
5AH	Setup	Matching data. Always start with this command after reset release.
30H	FLASH memory writing	Writing to area from 8000H to FFFFH is enable.
60H	RAM loader	Writing to area from 0050H to 082FH is enable.
90H	FLASH memory SUM output	The checksum of entire FLASH area (from 8000H to FFFFH) is output in order of the upper byte and the lower byte.
C0H	Product discrimination code output	Product discrimination code, that is expressed by 13 bytes data, is output.

2.1.6 Operation Mode

There are four operating modes in serial PROM mode: FLASH memory writing mode, RAM loader mode, FLASH memory SUM output mode and Product discrimination code output mode. For details about these modes, refer to (1) FLASH memory writing mode through (4) Product discrimination code output mode.

(1) FLASH memory writing mode

The data are written to the specified FLASH memory addresses. The controller should send the write data in the Intel Hex format (Binary). For details of writing data format, refer to 2.1.7 “FLASH Memory Writing Data Format”.

If no errors are encountered till the end record, the SUM of 32 Kbytes of FLASH memory is calculated and the result is returned to the controller.

To execute the FLASH memory writing mode, the TMP86FM25 checks the passwords except a blank product. If the passwords did not match, the program is not executed.

(2) RAM loader mode

The RAM loader transfers the data into the internal RAM that has been sent from the controller in Intel Hex format. When the transfer has terminated normally, the RAM loader calculates the SUM and sends the result to the controller before it starts executing the user program. After sending of SUM, the program jumps to the start address of RAM in which the first transferred data has been written. This RAM loader function provides the user's own way to control on-board programming.

To execute the RAM loader mode, the TMP86FM25 checks the passwords except a blank product. If the passwords did not match, the program is not executed.

(3) FLASH memory SUM output mode

The SUM of 32 Kbytes of FLASH memory is calculated and the result is returned to the controller.

The BOOT ROM does not support the reading function of the FLASH memory. Instead, it has this SUM command to use. By reading the SUM, it is possible to manage Revisions of application programs.

(4) Product discrimination code output mode

The product discrimination code is output as a 13-byte data, that includes the start address and the end address of ROM (In case of TMP86FM25, the start address is 8000H and the end address is FFFFH). Therefore, the controller can recognize the device information by using this function.

2.1.6.1 FLASH Writing Mode (Operation command: 30H)

Table 2.1.8 shows FLASH memory writing mode process.

Table 2.1.8 FLASH Writing Mode Process

	Number of Bytes Transferred	Transfer Data from External Controller to TMP86FM25	Baud Rate	Transfer Data from TMP86FM25 to External Controller
BOOT ROM	1st byte 2nd byte	Matching data (5AH) –	9600 bps 9600 bps	– (Baud rate auto set) OK: Echo back data (5AH) Error: Nothing transmitted
	3rd byte 4th byte	Baud rate modification data (See Table 2.1.5) –	9600 bps 9600 bps	– OK: Echo back data Error: A1H × 3, A3H × 3 , 62H × 3 (Note 1)
	5th byte 6th byte	Operation command data (30H) –	Changed new baud rate Changed new baud rate	– OK: Echo back data (30H) Error: A1H × 3, A3H × 3 , 63H × 3 (Note 1)
	7th byte 8th byte	Address 15H to 08H in which to store Password count (Note 4)	Changed new baud rate Changed new baud rate	– OK: Nothing transmitted Error: Nothing transmitted
	9th byte 10th byte	Address 07H to 00H in which to store Password count (Note 4)	Changed new baud rate Changed new baud rate	– OK: Nothing transmitted Error: Nothing transmitted
	11th byte 12th byte	Address 15H to 08H in which to start Password comparison (Note 4)	Changed new baud rate Changed new baud rate	– OK: Nothing transmitted Error: Nothing transmitted
	13th byte 14th byte	Address 07H to 00H in which to start Password comparison (Note 4)	Changed new baud rate Changed new baud rate	– OK: Nothing transmitted Error: Nothing transmitted
	15th byte : m'th byte	Password string (Note 5) –	Changed new baud rate Changed new baud rate	– OK: Nothing transmitted Error: Nothing transmitted
	m'th + 1 byte : n'th – 2 byte	Intel Hex format (Binary) (Note 2)	Changed new baud rate	–
	n'th – 1 byte	–	Changed new baud rate	OK: SUM (High) (Note 3) Error: Nothing transmitted
	n'th byte	–	Changed new baud rate	OK: SUM (Low) (Note 3) Error: Nothing transmitted
	n'th + 1 byte	(Wait for the next operation) (Command data)	Changed new baud rate	–

Note 1: "xxH × 3" denotes that operation stops after sending 3 bytes of xxH. For details, refer to 2.1.8 "Error Code".

Note 2: Refer to 2.1.10 "Intel Hex Format (Binary)".

Note 3: Refer to 2.1.9 "Checksum (SUM)".

Note 4: Refer to 2.1.11 "Passwords".

Note 5: If all data of addresses from FFE0H to FFFFH are "00H" or "FFH", the passwords comparison is not executed because the device is considered as blank product. However, it is necessary to specify the password count storage addresses and the password comparison start address even though it is a blank product. If a password error occurs, the UART function of TMP86FM25 stops without returning error code to the controller. Therefore, when a password error occurs, the TMP86FM25 should be reset by $\overline{\text{RESET}}$ pin input.

Description of FLASH memory writing mode

1. The receive data in the 1st byte is the matching data. When the boot program starts in serial PROM mode, TMP86FM25 (Mentioned as “device” hereafter) waits for the matching data (5AH) to receive. Upon receiving the matching data, it automatically adjusts the UART's initial baud rate to 9,600bps.
2. When the device has received the matching data, the device transmits the data “5AH” as an echo back to the controller. If the device can not receive the matching data, the device does not transmit the echo back data and waits for the matching data again with changing baud rate. Therefore, the controller should send the matching data continuously until the device transmits the echo back data. An external controller should transmit a matching data repeatedly till the device transmit an echo back data. The transmission number of times of matching data varies by the frequency of device. For details, refer to Table 2.1.6.
3. The receive data in the 3rd byte is the baud rate modification data. The seven kinds of baud rate modification data shown in Table 2.1.5 are available. Even if baud rate changing is no need, be sure to send the initial baud rate data (28H: 9,600 bps).
4. When the 3rd byte data is one of the baud rate modification data corresponding to the device's operating frequency, the device sends the echo back data which is the same as received baud rate modification data. Then the baud rate is changed. If the 3rd byte data does not correspond to the baud rate modification data, the device stops UART function after sending 3 bytes of baud rate modification error code: (62H). The changing of baud rate is executed after transmitting the echo back data.
5. The receive data in the 5th byte is the command data (30H) to write the FLASH memory.
6. When the 5th byte is one of the operation command data shown in Table 2.1.7, the device sends the echo back data which is the same as received operation command data (in this case, 30H). If the 5th byte data does not correspond to the operation command data, the device stops UART function after sending 3 bytes of operation command error code: (63H).
7. The 7th byte is used as an upper bit (Bit15 to bit8) of the password count storage address. When the receiving is executed correctly (No error), the device does not send any data. If the receiving error or password error occur, the device does not send any data and stops UART function.
8. The 9th byte is used as a lower bit (Bit7 to bit0) of the password count storage address. When the receiving is executed correctly (No error), the device does not send any data. If the receiving error or password error occur, the device does not send any data and stops UART function.
9. The 11th byte is used as an upper bit (Bit15 to bit8) of the password comparison start address. When the receiving is executed correctly (No error), the device does not send any data. If the receiving error or password error occur, the device does not send any data and stops UART function.
10. The 13th byte is used as a lower bit (Bit7 to bit0) of the password comparison start address. When the receiving is executed correctly (No error), the device does not send any data. If the receiving error or password error occur, the device does not send any data and stops UART function.

11. The 15th through the m'th bytes are the password data. The number of passwords is the data (N) indicated by the password count storage address. The password data are compared for N entries beginning with the password comparison start address. The controller should send N bytes of password data to the device. If the passwords do not match, the device stops UART function without returning error code to the controller. If the data of addresses from FFE0H to FFFFH are all "FFH" or "00H", the comparison of passwords is not executed because the device is considered as a blank product.
12. The receive data in the m'th + 1 through n'th – 2 byte are received as binary data in Intel Hex format. No received data are echoed back to the controller. The data which is not the start mark (3AH for ":") in Intel Hex format is ignored and does not send an error code to the controller until the device receives the start mark. After receiving the start mark, the device receives the data record, that consists of length of data, address, record type, writing data and checksum. After receiving the checksum of data record, the device waits the start mark data (3AH) again. The data of data record is temporarily stored to RAM and then, is written to specified FLASH memory by page (64 bytes) writing. For details of an organization of FLASH, refer to 2.1.7 "FLASH Memory Writing Data Format". Since after receiving an end record, the device starts to calculate the SUM, the controller should wait the SUM after sending the end record. If receive error or Intel Hex format error occurs, the device stops UART function without returning error code to the controller.
13. The n'th – 1 and the n'th bytes are the SUM value that is sent to the controller in order of the upper byte and the lower byte. For details on how to calculate the SUM, refer to 2.1.9 "Checksum (SUM)". The SUM calculation is performed after detecting the end record, but the calculation is not executed when receive error or Intel Hex format error has occurred. The time required to calculate the SUM of the 32 Kbytes of FLASH memory area is approximately 100 ms at fc = 16 MHz. After the SUM calculation, the device sends the SUM data to the controller. After sending the end record, the controller can judge that the transmission has been terminated correctly by receiving the checksum.
14. After sending the SUM, the device waits for the next operation command data.

2.1.6.2 RAM Loader Mode (Operation command: 60H)

Table 2.1.9 shows RAM loader mode process.

Table 2.1.9 RAM Loader Mode Process

	Number of Bytes Transferred	Transfer Data from External CONTROLLER to TMP86FM25	Baud Rate	Transfer Data from TMP86FM25 to External Controller
BOOT ROM	1st byte 2nd byte	Matching data (5AH) –	9600 bps 9600 bps	– (Baud rate auto set) OK: Echo back data (5AH) Error: Nothing transmitted
	3rd byte 4th byte	Baud rate modification data (See Table 2.1.5) –	9600 bps 9600 bps	– OK: Echo back data Error: A1H × 3, A3H × 3, 62H × 3 (Note 1)
	5th byte 6th byte	Operation command data (60H) –	Changed new baud rate Changed new baud rate	– OK: Echo back data (60H) Error: A1H × 3, A3H × 3, 63H × 3 (Note 1)
	7th byte 8th byte	Address 15H to 08H in which to store Password count (Note 4)	Changed new baud rate Changed new baud rate	– OK: Nothing transmitted Error: Nothing transmitted
	9th byte 10th byte	Address 07H to 00H in which to store Password count (Note 4)	Changed new baud rate Changed new baud rate	– OK: Nothing transmitted Error: Nothing transmitted
	11th byte 12th byte	Address 15H to 08H in which to start Password comparison (Note 4)	Changed new baud rate Changed new baud rate	– OK: Nothing transmitted Error: Nothing transmitted
	13th byte 14th byte	Address 07H to 00H in which to start Password comparison (Note 4)	Changed new baud rate Changed new baud rate	– OK: Nothing transmitted Error: Nothing transmitted
	15th byte : m'th byte	Password string (Note 5) –	Changed new baud rate Changed new baud rate	– OK: Nothing transmitted Error: Nothing transmitted
	m'th + 1 byte : n'th – 2 byte	Intel Hex format (Binary) (Note 2)	Changed new baud rate	–
	n'th – 1 byte	–	Changed new baud rate	OK: SUM (High) (Note 3) Error: Nothing transmitted
	n'th byte	–	Changed new baud rate	OK: SUM (Low) (Note 3) Error: Nothing transmitted
RAM	–	The program jumps to the start address of RAM in which the first transferred data has been written.		

Note 1: "xxH × 3" denotes that operation stops after sending 3 bytes of xxH. For details, refer to 2.1.8 "Error Code".

Note 2: Refer to 2.1.10 "Intel Hex Format (Binary)".

Note 3: Refer to 2.1.9 "Checksum (SUM)".

Note 4: Refer to 2.1.11 "Passwords".

Note 5: If all data of addresses from FFE0H to FFFFH are "00H" or "FFH", the passwords comparison is not executed because the device is considered as blank product. However, it is necessary to specify the password count storage addresses and the password comparison start address even though it is a blank product. If a password error occurs, the UART function of TMP86FM25 stops without returning error code to the controller. Therefore, when a password error occurs, the TMP86FM25 should be reset by $\overline{\text{RESET}}$ pin input.

Note 6: Do not send only end record after transferring of password string. If the TMP86FM25 receives the end record only after reception of password string, it does not operate correctly.

Note 7: When the FLASH power supply is turned off in user's program by setting EEPCR<MNPWDW>, be sure to disable the watchdog timer (WDT) or to clear the binary counter of WDT immediately before.

Description of RAM loader mode

1. The process of the 1st byte through the 4th byte are the same as FLASH memory writing mode.
2. The receive data in the 5th byte is the RAM loader command data (60H) to write the user's program to RAM.
3. When the 5th byte is one of the operation command data shown in Table 2.1.7, the device sends the echo back data which is the same as received operation command data (in this case, 60H). If the 5th byte data does not correspond to the operation command data, the device stops UART function after sending 3 bytes of operation command error code: (63H).
4. The process of the 7th byte through the m'th byte are the same as FLASH memory writing mode.
5. The receive data in the m'th + 1 through n'th – 2 bytes are received as binary data in Intel Hex format. No received data are echoed back to the controller.
   The data which is not the start mark (3AH for ":") in Intel Hex format is ignored and does not send an error code to the controller until the device receives the start mark. After receiving the start mark, the device receives the data record, that consists of length of data, address, record type, writing data and checksum. After receiving the checksum of data record, the device waits the start mark data (3AH) again. The data of data record is written to specified RAM by the receiving data. Since after receiving an end record, the device starts to calculate the SUM, the controller should wait the SUM after sending the end record. If receive error or Intel Hex format error occurs, the UART function of TMP86FM25 stops without returning error code to the controller.
6. The n'th – 1 and the n'th bytes are the SUM value that is sent to the controller in order of the upper byte and the lower byte. For details on how to calculate the SUM, refer to 2.1.9 "Checksum (SUM)". The SUM calculation is performed after detecting the end record, but the calculation is not executed when receive error or Intel Hex format error has occurred.
   The SUM is calculated by the data written to RAM, but the length of data, address, record type and checksum in Intel Hex format are not included in SUM.
7. The boot program jumps to the first address that is received as data in Intel Hex format after sending the SUM to the controller.

2.1.6.3 FLASH Memory SUM Output Mode (Operation command: 90H)

Table 2.1.10 shows FLASH memory SUM output mode process.

Table 2.1.10 FLASH Memory SUM Output Process

	Number of Bytes Transferred	Transfer Data from External Controller to TMP86FM25	Baud Rate	Transfer Data from TMP86FM25 to External Controller
BOOT ROM	1st byte 2nd byte	Matching data (5AH) –	9600 bps 9600 bps	– (Baud rate auto set) OK: Echo back data (5AH) Error: Nothing transmitted
	3rd byte 4th byte	Baud rate modification data (See Table 2.1.5) –	9600 bps 9600 bps	– OK: Echo back data Error: A1H × 3, A3H × 3, 62H × 3 (Note 1)
	5th byte 6th byte	Operation command data (90H) –	Changed new baud rate Changed new baud rate	– OK: Echo back data (90H) Error: A1H × 3, A3H × 3, 63H × 3 (Note 1)
	7th byte	–	Changed new baud rate	OK: SUM (High) (Note 2) Error: Nothing transmitted
	8th byte	–	Changed new baud rate	OK: SUM (Low) (Note 2) Error: Nothing transmitted
	9th byte	(Wait for the next operation) (Command data)	Changed new baud rate	–

Note 1: "xxH × 3" denotes that operation stops after sending 3 bytes of xxH. For details, refer to 2.1.8 "Error Code".

Note 2: Refer to 2.1.9 "Checksum (SUM)"

Description of FLASH memory SUM output mode

1. The process of the 1st byte through the 4th byte are the same as FLASH memory writing mode.
2. The receive data in the 5th byte is the FLASH memory SUM command data (90H) to calculate the entire FLASH memory.
3. When the 5th byte is one of the operation command data shown in Table 2.1.7, the device sends the echo back data which is the same as received operation command data (in this case, 90H). If the 5th byte data does not correspond to the operation command data, the device stops UART function after sending 3 bytes of operation command error code: (63H).
4. The 7th and the 8th bytes are the SUM value that is sent to the controller in order of the upper byte and the lower byte. For details on how to calculate the SUM, refer to 2.1.9 "Checksum (SUM)".
5. After sending the SUM, the device waits for the next operation command data.

2.1.6.4 Product Discrimination Code Output Mode (Operation command: C0H)

Table 2.1.11 shows product discrimination code output mode process.

Table 2.1.11 Product Discrimination Code Output Process

	Number of Bytes Transferred	Transfer Data from External Controller to TMP86FM25	Baud Rate	Transfer Data from TMP86FM25 to External Controller	
BOOT ROM	1st byte 2nd byte	Matching data (5AH) –	9600 bps 9600 bps	– (Baud rate auto set) OK: Echo back data (5AH) Error: Nothing transmitted	
	3rd byte 4th byte	Baud rate modification data (See Table 2.1.5) –	9600 bps 9600 bps	– OK: Echo back data Error: A1H × 3, A3H × 3, 62H × 3 (Note 1)	
	5th byte 6th byte	Operation command data (C0H) –	Changed new baud rate Changed new baud rate	– OK: Echo back data (C0H) Error: A1H × 3, A3H × 3, 63H × 3 (Note 1)	
	7th byte		Changed new baud rate	3AH	Start mark
	8th byte		Changed new baud rate	0AH	The number of transfer data (from 9th to 18th byte)
	9th byte		Changed new baud rate	02H	Length of address (2 bytes)
	10th byte		Changed new baud rate	00H	Reserved data
	11th byte		Changed new baud rate	00H	Reserved data
	12th byte		Changed new baud rate	00H	Reserved data
	13th byte		Changed new baud rate	00H	Reserved data
	14th byte		Changed new baud rate	01H	The number of ROM block (1 block)
	15th byte		Changed new baud rate	80H	First address of ROM
	16th byte		Changed new baud rate	00H	
	17th byte		Changed new baud rate	FFH	End address of ROM
	18th byte		Changed new baud rate	FFH	
	19th byte		Changed new baud rate	7FH	Checksum of transferred data (from 9th to 18th byte)
	20th byte	(Wait for the next operation) (command data)	Changed new baud rate	–	

Note: "xxH × 3" denotes that operation stops after sending 3 bytes of xxH. For details, refer to 2.1.8 "Error Code".

Description of product discrimination code output mode

1. The process of the 1st byte through the 4th byte are the same as FLASH memory writing mode.
2. The receive data in the 5th byte is the product discrimination code output command data (C0H).
3. When the 5th byte is one of the operation command data shown in Table 2.1.7, the device sends the echo back data which is the same as received operation command data (in this case, C0H). If the 5th byte data does not correspond to the operation command data, the device stops UART function after sending 3 bytes of operation command error code: (63H).
4. The 7th and the 19th bytes are the product discrimination code. For details, refer to 2.1.12 "Product Discrimination Code".
5. After sending the SUM, the device waits for the next operation command data.

2.1.7 FLASH Memory Writing Data Format

FLASH area of TMP86FM25 consists of 512 pages and one page size is 64 bytes.

Writing to FLASH is executed by page writing. Therefore, it is necessary to send 64 bytes data (for one page) even though only a few bytes data are written. Figure 2.1.5 shows an organization of FLASH area. When the controller sends the writing data to the device, be sure to keep the format described below.

1. The address of data after receiving the FLASH writing command should be the first address of page. For example, in case of page 2, the first address should be 8080H.
2. If the last data's address of data record is not end address of page, the address of the next data record should be the address + 1. For example, if the last data's address is 802FH (Page 0), the address of the next data record should be 8030H (Page 0).

   Example:

   :1080200020212223242526272829 2A2B2C2D2E2FD8 ' 8020H to 802FH data

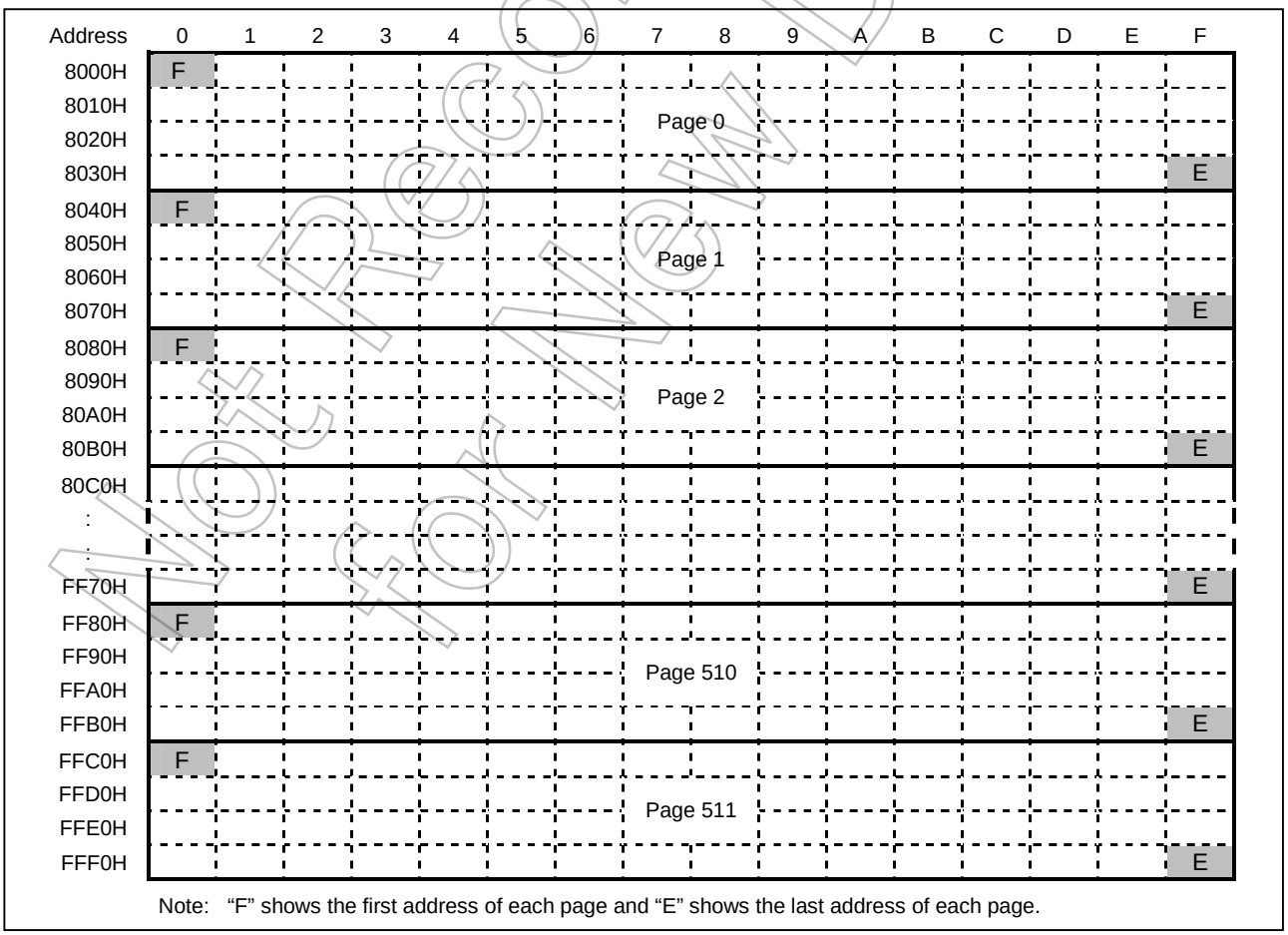
   :1080300030313233343536373839 3A3B3C3D3E3FC8 ' 8030H to 803FH data
3. The last data's address of data record immediately before sending the end record should be the last address of page. For example, in case of page 1, the last data's address of data record should be 807FH.

   Example:

   :1080700030313233343536373839 3A3B3C3D3E3F88 ' 8070H to 807FH data

   :00000001FF ' End record

Note: Do not write only the addresses from FFE0H to FFFFH when all data of FLASH memory are the same data. If these area are only written, the next operation can not be executed because of password error.



Note: "F" shows the first address of each page and "E" shows the last address of each page.

Figure 2.1.5 Organization of FLASH Area

2.1.8 Error Code

When the device detects an error, the error codes are sent to the controller.

Table 2.1.12 Error Code

Transmit Data	Meaning of Transmit Data
62H, 62H, 62H	Baud rate modification error occurred.
63H, 63H, 63H	Operating command error occurred.
A1H, A1H, A1H	Framing error in received data occurred.
A3H, A3H, A3H	Overrun error in received data occurred.

Note: If password error occurs, the TMP86FM25 doesn't send error codes.

2.1.9 Checksum (SUM)

(1) Calculation method

SUM consists of byte + byte... + byte, the checksum of which is returned in word as the result.

Namely, data is read out in byte and checksum of which is calculated, with the result returned in word.

Example:

If the data to be calculated consists of the four bytes shown to the left, SUM of the data is

A1H + B2H + C3H + D4H = 02EAH
SUM (HIGH) = 02H
SUM (LOW) = EAH

The SUM returned when executing the FLASH memory write command, RAM loader command, or FLASH memory SUM command is calculated in the manner shown above.

(2) Calculation data

The data from which SUM is calculated are listed in Table 2.1.13 below.

Table 2.1.13 Checksum Calculation Data

Operating Mode	Calculation Data	Remarks
FLASH memory writing mode	Data in the entire area (32 Kbytes) of FLASH memory	Even when written to part of the FLASH area, data in the entire memory area (32 Kbytes) is calculated.
FLASH memory SUM output mode		The length of data, address, record type and checksum in Intel Hex format are not included in SUM.
RAM loader mode	Data written to RAM	The length of data, address, record type and checksum in Intel Hex format are not included in SUM.
Product Discrimination Code Output mode	Checksum of transferred data (from 9th to 18th byte)	For details, refer to 2.1.12 "Product Discrimination Code".

2.1.10 Intel Hex Format (Binary)

1. After receiving the checksum of a record, the device waits for the start mark data (3AH for ":") of the next record. Therefore, the device ignores the data, which does not match the start mark data after receiving the checksum of a record.
2. Make sure that once the controller program has finished sending the checksum of the end record, it does not send anything and waits for two bytes of data to be received (Upper and lower bytes of checksum). This is because after receiving the checksum of the end record, the boot program calculates the checksum and returns the calculated checksum in two bytes to the controller.
3. If a receive error or Intel Hex format error occurs, the UART function of TMP86FM25 stops without returning error code to the controller. In the following cases, an Intel Hex format error occurs:
   - When the record type is not 00H, 01H, or 02H
   - When a SUM error occurred
   - When the data length of an extended record (Type = 02H) is not 02H
   - When the address of an extended record (Type = 02H) is larger than 1000H and after that, receives the data record
   - When the data length of the end record (Type = 01H) is not 00H

2.1.11 Passwords

The eight or more bytes consecutive data in flash memory area can be used as password. In password check, TMP86FM25 compares these data with data which are transmitted from the external controller. The area in which passwords can be specified is located at addresses 8000H to FF9FH. The area from FFA0H to FFFFH can not be specified as passwords area. The device compares the stored passwords with the passwords, which are received from the controller. If all data of addresses from FFE0H to FFFFH are "00H" or "FFH", the passwords comparison is not executed because the device is considered as blank product. It is necessary to specify the password count storage addresses and the password comparison start address even though it is a blank product. Table 2.1.14 shows the password setting in the blank product and non blank product.

Table 2.1.14 Password setting in the Blank Product and Non Blank Product

Password	Blank Product (Note 1)	Non Blank Product
PNSA (Password count storage addresses)	8000H ≤ PNSA ≤ FF9FH	8000H ≤ PNSA ≤ FF9FH
PCSA (Password comparison start address)	8000H ≤ PCSA ≤ FF9FH	8000H ≤ PCSA ≤ FFA0 − N
N (Password count)	*	8 ≤ N
Setting of password	No need	Need (Note 2)

Note 1: When all data of addresses from FFE0H to FFFFH area are “00H” or “FFH”, the device is judged as blank product.

Note 2: The same three or more bytes consecutive data can not be used as password. When the password includes the same consecutive data (Three or more bytes), the password error occurs. If the password error occurred, the UART function of device stops without returning error code.

Note 3: *: Don’t care.

Note 4: When the password doesn't match the above condition, the password error occurs. If the password error occurred, the UART function of device stops without returning error code.

Note 5: In case of the blank product, the device receives Intel Hex Format immediately after receiving PCSA without receiving password strings. In this time, because the device ignores the data except the start mark data (3AH for “:”) as Intel Hex Format data, even if external controller transmitted dummy password strings, process operates correctly. However, if the dummy password strings contain data “3AH”, the device detects it as start mark data mistakenly, and device stops process without returning error code. Therefore, if these process becomes issue, the external controller should not transmit the dummy password strings.

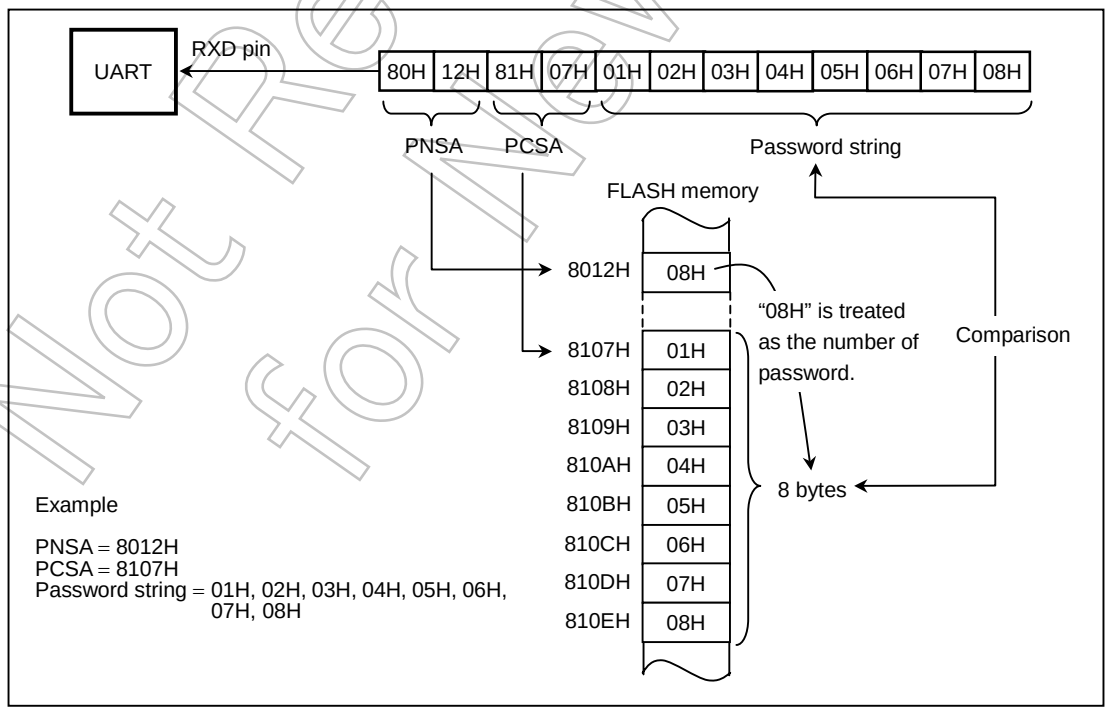


Figure 2.1.6 Example of password compare

2.1.11.1 Confirmation method of the blank product and non blank product

The external controller can confirm whether the device is the blank product or not, by transmission of data described below.

(1) Executes FLASH memory writing mode or RAM loader mode.

(2) Transmits the PNSA and PCSA.

(3) Transmits the end record.

(4) In case of the blank product, the device sends checksum of flash memory. In case of the non blank product, the device doesn't send checksum of flash memory but the UART function stops without sending any data.

The external controller can confirm the blank product and non blank product by receiving checksum.

Note: When the UART function stops in non blank product, the TMP86FM25 should be reset by pin reset input for restarting the serial PROM mode.

2.1.11.2 Password String

A string of passwords in the received data are compared with the data in the FLASH memory. In the following cases, a password error occurs:

- When the received data does not match the data in the FLASH memory

2.1.11.3 Handling of Password Error

If a password error occurs, the UART function of TMP86FM25 stops without returning error code to the controller. Therefore, when a password error occurs, the TMP86FM25 should be reset by $\overline{\text{RESET}}$ pin input.

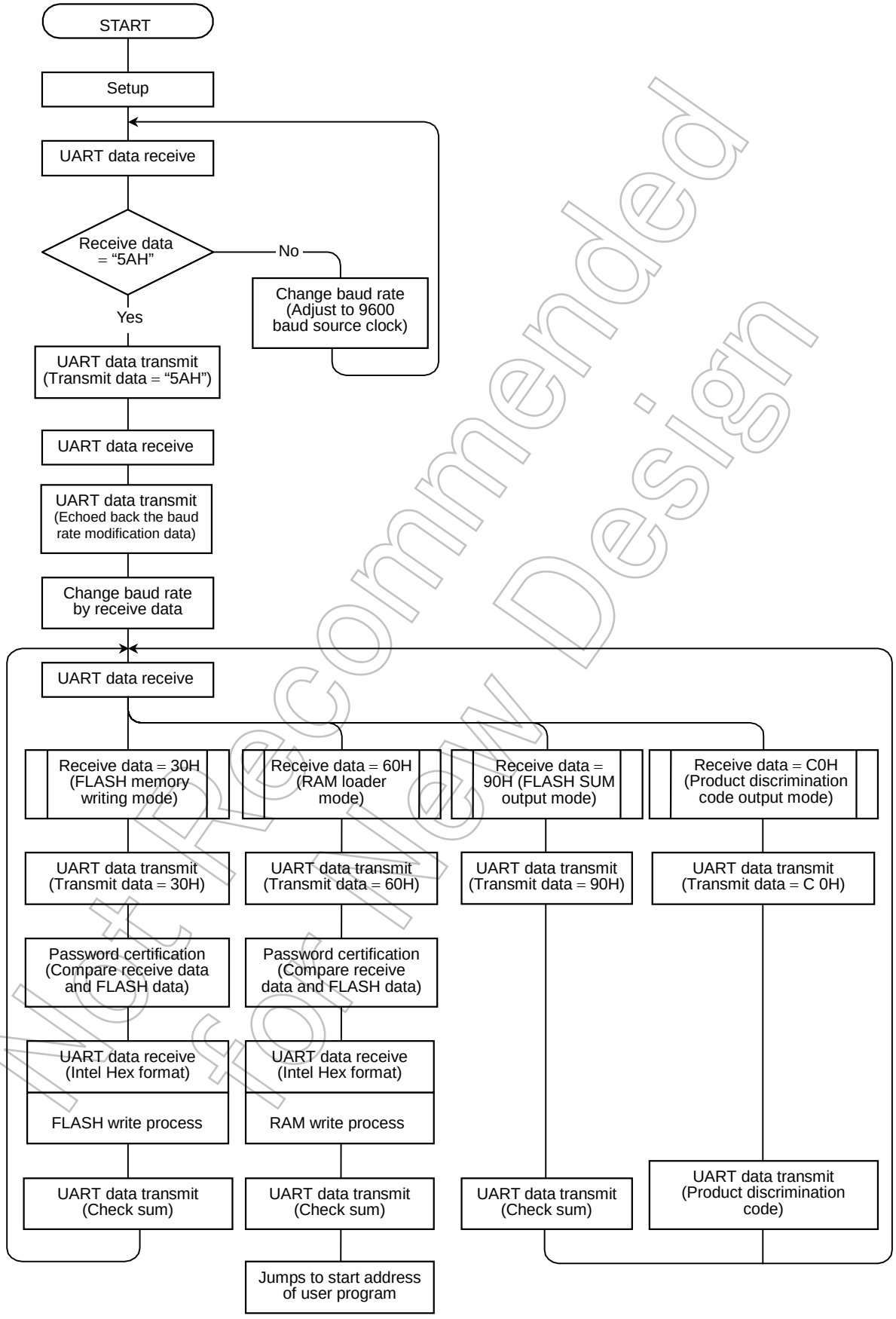
2.1.12 Product Discrimination Code

The product discrimination code is a 13-byte data, that includes the start address and the end address of ROM. Table 2.1.15 shows the product discrimination code format.

Table 2.1.15 Product Discrimination Code Format

Data	The Meaning of Data	In Case of TMP86FM25
1st	Start mark (3AH)	3AH
2nd	The number of transfer data (from 3rd to 12th byte)	0AH
3rd	Length of address	02H
4th	Reserved data	00H
5th	Reserved data	00H
6th	Reserved data	00H
7th	Reserved data	00H
8th	The number of ROM block	01H
9th	The upper byte of the first address of ROM	80H
10th	The lower byte of the first address of ROM	00H
11th	The upper byte of the end address of ROM	FFH
12th	The lower byte of the end address of ROM	FFH
13th	Checksum of transferred data (from 3rd to 12th byte)	7FH

2.1.13 Flowchart



Electrical Characteristics

Absolute Maximum Ratings (V_{SS} = 0 V)

Parameter	Symbol	Pins	Rating	Unit
Supply voltage	V_{DD}		−0.3 to 4.0	V
Input voltage	V_{IN}		−0.3 to V_{DD} + 0.3	
Output voltage	V_{OUT1}	Except V4 pin	−0.3 to V_{DD} + 0.3	
	V_{OUT2}	V4 pin	-0.3 to 4.0	
Output current (Per 1 pin)	I_{OUT1}	P6 port	−1.8	mA
	I_{OUT2}	P1, P2, P34 to P36, P5, P6, P7 ports	3.2	
	I_{OUT3}	P30 to P33 port	30	
Output current (Total)	ΣI_{OUT1}	P6 port	−30	
	ΣI_{OUT2}	P1, P2, P34 to P36, P5, P6, P7 ports	60	
	ΣI_{OUT3}	P30 to P33 port	80	
Power dissipation [Topr = 85°C]	PD		350	mW
Soldering temperature (Time)	Tsld		260 (10 s)	°C
Storage temperature	Tstg		−55 to 125	
Operating temperature	Topr		−40 to 85	

Note: The absolute maximum ratings are rated values which must not be exceeded during operation, even for an instant. Any one of the ratings must not be exceeded. If any absolute maximum rating is exceeded, a device may break down or its performance may be degraded, causing it to catch fire or explode resulting in injury to the user. Thus, when designing products which include this device, ensure that no absolute maximum rating value will ever be exceeded.

Recommended Operating Condition-1 (MCU mode) (VSS = 0 V, Topr = −40 to 85°C)

Parameter	Symbol	Pins	Condition		Min	Max	Unit
Supply voltage	V_{DD}		fc = 16 MHz	NORMAL1, 2 mode	2.7	3.6	V
				IDLE0, 1, 2 mode			
			fc = 4.2 MHz (in case of external clock)	NORMAL1, 2 mode	1.8		
				IDLE0, 1, 2 mode			
			fc = 8 MHz (in case of connecting a resonator)	NORMAL1, 2 mode	1.8		
				IDLE0, 1, 2 mode			
			fs = 32.768 kHz	SLOW1, 2 mode	1.8		
				SLEEP0, 1, 2 mode			
			STOP mode				
Input high level	V_{IH1}	Except hysteresis input	$V_{DD} \geq 2.7$ V		$V_{DD} \times 0.70$	V_{DD}	
	V_{IH2}	Hysteresis input			$V_{DD} \times 0.75$		
	V_{IH3}		$V_{DD} < 2.7$ V		$V_{DD} \times 0.90$		
Input low level	V_{IL1}	Except hysteresis input	$V_{DD} \geq 2.7$ V		0	$V_{DD} \times 0.30$	
	V_{IL2}	Hysteresis input				$V_{DD} \times 0.25$	
	V_{IL3}		$V_{DD} < 2.7$ V			$V_{DD} \times 0.10$	
Clock frequency (in case of external clock)	fc	XIN, XOUT	V_{DD} = 1.8 to 3.6 V		1.0	4.2	MHz
			V_{DD} = 2.7 to 3.6 V			16.0	
	fs	XTIN, XTOUT	V_{DD} = 1.8 to 3.6 V		30.0	34.0	kHz
Clock frequency (in case of connecting a resonator)	fc	XIN, XOUT	V_{DD} = 1.8 to 3.6 V		1.0	8.0	MHz
			V_{DD} = 2.7 to 3.6 V			16.0	
	fs	XTIN, XTOUT	V_{DD} = 1.8 to 3.6 V		30.0	34.0	kHz
LCD reference voltage	$V2_{IN}$	V2	LCDCTL1<REFV> = "1" VDD < V4 (Note 2)		1.650	1.800	V
	$V3_{IN}$	V3			2.250	2.700	
	$V4_{IN}$	V4			3.000	3.600	
	$V4_{IN}$	V4 (Note 3)	LCDCTL1<REFV> = "0"		3.000	VDD	
Capacity for LCD booster circuit	C_{LCD}				0.1	0.47	μF

Note 1: The recommended operating conditions for a device are operating conditions under which it can be guaranteed that the device will operate as specified. If the device is used under operating conditions other than the recommended operating conditions (Supply voltage, operating temperature range, specified AC/DC values etc.), malfunction may occur. Thus, when designing products which include this device, ensure that the recommended operating conditions for the device are always adhered to.

Note 2: When LCDCTL1<REFV> is set to "1", always keep the condintion of VDD < V4.

Note 3: When LCDCTL1<REFV> is cleared to "0", always supply the reference voltage from V4 pin.

Recommended Operating Condition-2 (Serial PROM mode) (VSS = 0 V, Topr = 25°C ± 5°C)

Parameter	Symbol	Pins	Condition	Min	Max	Unit
Supply voltage	V_{DD}		2 MHz ≤ fc ≤16 MHz	2.7	3.6	V
Clock frequency	fc	XIN, XOUT	V_{DD} = 2.7 to 3.6 V	2.0	16.0	MHz

Note: The operating temperature area of serial PROM mode is 25°C ± 5°C and the operating area of high frequency of serial PROM mode is different from MCU mode.

DC Characteristics (VSS = 0 V, Topr = –40 to 85°C)

Parameter	Symbol	Pins	Condition	Min	Typ.	Max	Unit
Hysteresis voltage	V_{HS}	Hysteresis input	V_{DD} = 3.3 V	–	0.4	–	V
Input current	I_{IN1}	TEST	V_{DD} = 3.6 V, V_{IN} = 0 V	–	–	–5	μA
	I_{IN2}	Sink open drain, Tri-state	V_{DD} = 3.6 V, V_{IN} = 3.6 V/0 V	–	–	±5	
	I_{IN3}	$\overline{RESET}$, $\overline{STOP}$	V_{DD} = 3.6 V, V_{IN} = 3.6 V	–	–	+5	
Input resistance	R_{IN1}	TEST pull down	V_{DD} = 3.6 V, V_{IN} = 3.6 V	–	70	–	kΩ
	R_{IN2}	$\overline{RESET}$ pull up	V_{DD} = 3.6 V, V_{IN} = 0 V	100	220	450	
High-frequency feedback resistor	R_{FB}	XOUT	V_{DD} = 3.6 V	–	1.2	–	MΩ
Low-frequency feedback resistor	R_{FBT}	XTOUT	V_{DD} = 3.6 V	–	14	–	
Output leakage current	I_{LO}	Sink open drain, Tri-state	V_{DD} = 3.6 V V_{OUT} = 3.4V / 0.2 V	–	–	±10	μA
Output high voltage	V_{OH}	C-MOS, Tri-state	V_{DD} = 3.6 V, I_{OH} = –0.6 mA	3.2	–	–	V
Output low voltage	V_{OL}	Except XOUT, P30 to P33 port	V_{DD} = 3.6 V, I_{OL} = 0.9 mA	–	–	0.4	
Output low current	I_{OL}	P30 to P33 ports	V_{DD} = 3.6 V, V_{OL} = 1.0 V	–	6	–	mA
Supply current in NORMAL1, 2 mode	I_{DD}	Fetch area: Flash area	V_{DD} = 3.6 V V_{IN} = 3.4V/0.2V fc = 16 MHz fs = 32.768 kHz	MNP = "1"	–	6.0	7.2	mA
		Fetch area: RAM area		MNP = "0"	–	3.9	4.8	
Supply current in IDLE0, 1, 2 mode				MNP·ATP = "1"	–	3.3	4.3	
				MNP·ATP = "0"	–	2.5	3.0	
Supply current in SLOW1 mode		Fetch area: Flash area	V_{DD} = 3.6 V V_{IN} = 3.4V/0.2V fs = 32.768 kHz	MNP = "1"	–	850	1200	μA
		Fetch area: RAM area		MNP = "0"	–	10	21	
Supply current in SLEEP1 mode				MNP·ATP = "1"	–	850	1200	
				MNP·ATP = "0"	–	7	17	
Supply current in SLEEP0 mode				MNP·ATP = "1"	–	850	1200	
				MNP·ATP = "0"	–	6	16	
Supply current in STOP mode			V_{DD} = 3.6 V V_{IN} = 3.4 V/0.2 V		–	0.5	10	

Note1: Typical values show those at Topr = 25°C, V_{DD} = 3.3 V

Note2: Input current (I_{IN1}, I_{IN2}): The current through pull-up or pull-down resistor is not included.

Note3: I_{DD} does not include I_{REF} current.

Note4: The supply currents of SLOW2 and SLEEP2 modes are equivalent to IDLE0, IDLE1, IDLE2.

Note5: MNP (MNPWDW) shows bit0 in EEPCR register and ATP(ATPWDW) shows bit1 in EEPCR register.

Note6: "Fetch" means reading operation of FLASH data as an instruction by CPU.

AD Conversion Characteristics (VSS = 0.0 V, 2.7 V ≤ VDD ≤ 3.6 V, Topr = –40 to 85°C)

Parameter	Symbol	Condition	Min	Typ.	Max	Unit
Analog reference voltage	V_{AREF}		V_{DD} – 1.0	–	V_{DD}	V
Analog reference voltage range (Note 4)	ΔV_{AREF}		2.5	–	–	
Analog input voltage	V_{AIN}		V_{SS}	–	V_{AREF}	
Power supply current of analog reference voltage	I_{REF}	$V_{DD} = V_{AREF} = 3.6$ V $V_{SS} = 0.0$ V	–	0.4	–	mA
Non linearity error		$V_{DD} = 2.7$ V $V_{SS} = 0.0$ V $V_{AREF} = 2.7$ V	–	–	±1	LSB
Zero point error			–	–	±1	
Full scale error			–	–	±1	
Total error			–	–	±2	

(V_{SS} = 0.0 V, 2.0 V ≤ V_{DD} < 2.7 V, Topr = –40 to 85°C)

Parameter	Symbol	Condition	Min	Typ.	Max	Unit
Analog reference voltage	V_{AREF}		V_{DD} – 0.6	–	V_{DD}	V
Analog reference voltage range (Note 4)	ΔV_{AREF}		2.0	–	–	
Analog input voltage	V_{AIN}		V_{SS}	–	V_{AREF}	
Power supply current of analog reference voltage	I_{REF}	$V_{DD} = V_{AREF} = 2.0$V $V_{SS} = 0.0$ V	–	0.22	–	mA
Non linearity error		$V_{DD} = 2.0$ V $V_{SS} = 0.0$ V $V_{AREF} = 2.0$ V	–	–	±1	LSB
Zero point error			–	–	±1	
Full scale error			–	–	±1	
Total error			–	–	±2	

(V_{SS} = 0.0 V, 1.8 V ≤ V_{DD} < 2.0 V, Topr = –10 to 85°C) (Note 5)

Parameter	Symbol	Condition	Min	Typ.	Max	Unit
Analog reference voltage	V_{AREF}		V_{DD} – 0.1	–	V_{DD}	V
Analog reference voltage range (Note 4)	ΔV_{AREF}		1.8	–	–	
Analog input voltage	V_{AIN}		V_{SS}	–	V_{AREF}	
Power supply current of analog reference voltage	I_{REF}	$V_{DD} = V_{AREF} = 1.8$V $V_{SS} = 0.0$ V	–	0.2	–	mA
Non linearity error		$V_{DD} = 1.8$ V $V_{SS} = 0.0$ V $V_{AREF} = 1.8$ V	–	–	±2	LSB
Zero point error			–	–	±2	
Full scale error			–	–	±2	
Total error			–	–	±4	

Note 1: The total error includes all errors except a quantization error, and is defined as a maximum deviation from the ideal conversion line.

Note 2: Conversion time is different in recommended value by power supply voltage.

Note 3: Please use input voltage to AIN input Pin in limit of $V_{AREF} - V_{SS}$.
When voltage of range outside is input, conversion value becomes unsettled and gives affect to other channel conversion value.

Note 4: Analog Reference Voltage Range: $\Delta V_{AREF} = V_{AREF} - V_{SS}$

Note 5: When AD is used with V_{DD} < 2.0 V, the guaranteed temperature range varies with the operating voltage.

AC Characteristics (VSS = 0 V, VDD = 2.7 to 3.6 V, Topr = –40 to 85°C)

Parameter	Symbol	Condition	Min	Typ.	Max	Unit
Machine cycle time	tcy	NORMAL1, 2 mode	0.25	–	4	μs
		IDLE1, 2 mode				
		SLOW1, 2 mode	117.6	–	133.3	
		SLEEP1, 2 mode				
High level clock pulse width	twcH	For external clock operation (XIN input) fc = 16 MHz	–	31.25	–	ns
Low level clock pulse width	twcL					
High level clock pulse width	twcH	For external clock operation (XTIN input) fs = 32.768 kHz	–	15.26	–	μs
Low level clock pulse width	twcL					

(V_{SS} = 0 V, V_{DD} = 1.8 to 3.6 V, Topr = –40 to 85°C)

Parameter	Symbol	Condition	Min	Typ.	Max	Unit
Machine cycle time	tcy	NORMAL1, 2 mode	0.5	–	4	μs
		IDLE1, 2 mode				
		SLOW1, 2 mode	117.6	–	133.3	
		SLEEP1, 2 mode				
High level clock pulse width	twcH	For external clock operation (XIN input) fc = 4.2 MHz	–	119.04	–	ns
Low level clock pulse width	twcL					
High level clock pulse width	twcH	For external clock operation (XTIN input) fs = 32.768 kHz	–	15.26	–	μs
Low level clock pulse width	twcL					

Timer Counter 1 input (ECIN) Characteristics (V_{SS} = 0 V, Topr = –40 to 85°C)

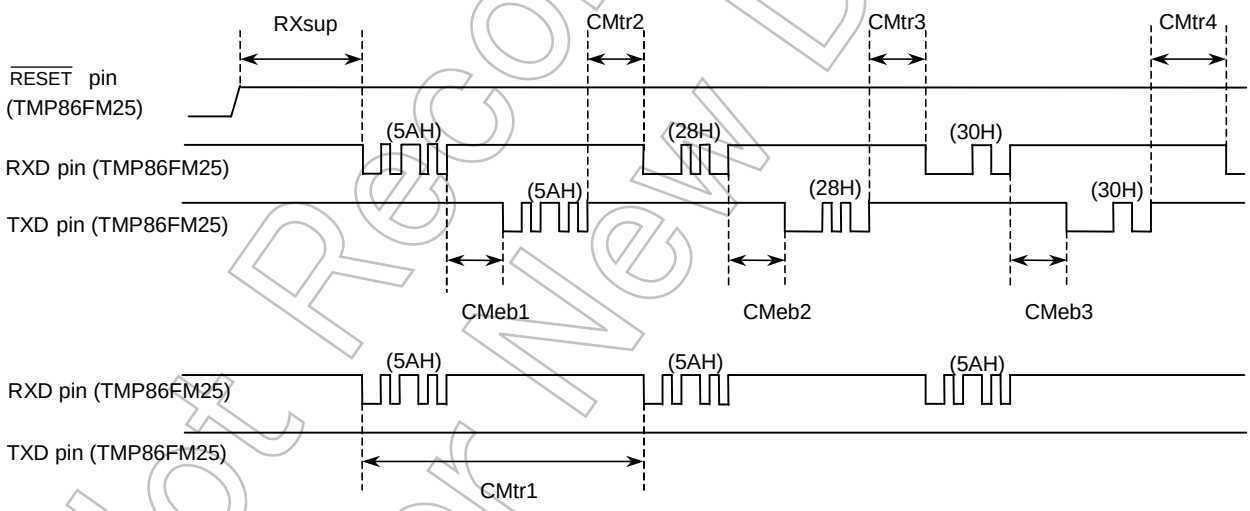
Parameter	Symbol	Condition	Min	Typ.	Max	Unit
TC1 input (ECIN input)	t_{TC1}	Frequency measurement mode V_{DD} = 2.7 to 3.6 V	–	–	0.5.	MHz
		Frequency measurement mode V_{DD} = 1.8 to 2.7 V	–	–	0.25	

UART Timing-1 (VDD = 2.7 V to 3.6 V, fc = 2 MHz to 16 MHz, Ta = 25°C)

Parameter	Symbol	The Number of Clock (fc)	Required Minimum Time	
			At fc = 2 MHz	At fc = 16 MHz
Time from the reception of a matching data until the output of an echo back	CMeb1	Approx. 600	300 μs	37.5 μs
Time from the reception of a Baud Rate Modification Data until the output of an echo back	CMeb2	Approx. 700	350 μs	43.7 μs
Time from the reception of an operation command until the output of an echo back	CMeb3	Approx. 600	300 μs	37.5 μs
Calculation time of checksum	CKsm	Approx. 1573000	786.5 ms	98.3 ms

UART Timing-2 (VDD = 2.7 V to 3.6 V, fc = 2 MHz to 16 MHz,Ta = 25°C)

Parameter	Symbol	The Number of Clock (fc)	Required Minimum Time	
			At fc = 2 MHz	At fc = 16 MHz
Time from reset release until acceptance of start bit of RXD pin	RXsup	83850	41.9 ms	5.3 ms
Time between a matching data and the next matching data	CMtr1	28500	14.3 ms	1.8 ms
Time from the echo back of matching data until the acceptance of baud rate modification data	CMtr2	600	300 μs	37.5 μs
Time from the output of echo back of baud rate modification data until the acceptance of an operation command	CMtr3	750	375 μs	46.9 μs
Time from the output of echo back of operation command until the acceptance of Password count storage addresses	CMtr4	950	475 μs	59.4 μs



Flash Characteristics (V_{SS} = 0 V)

Parameter	Condition	Min	Typ.	Max	Unit
Number of guaranteed writes (Page writing) to Flash memory in serial PROM mode	V_{DD} = 2.7 to 3.6 V, 2 MHz ≤ fc ≤ 16 MHz (Topr = 25°C ± 5°C)	–	–	10^5	Times

Package Dimensions

P-QFP100-1420-0.65A

Unit: mm