NEC

Customer Notification

78K0/Fx2

8-Bit Single-Chip Microcontrollers

Operating Precautions

78F0881, 78F0882, 78F0883
78F0884, 78F0885, 78F0886
78F0887, 78F0888, 78F0889, 78F0890
78F0891, 78F0892, 78F0893

Document No. U18390EE3V0IF00 (3rd edition)
Date Published June 2008

DISCLAIMER

The related documents in this customer notification may include preliminary versions. However, preliminary versions may not have been marked as such.

The information in this customer notification is current as of its date of publication. The information is subject to change without notice. For actual design-in, refer to the latest publications of NEC's data sheets or data books, etc., for the most up-to-date specifications of PRODUCT(S). Not all PRODUCT(S) and/or types are available in every country. Please check with an NEC sales representative for availability and additional information.

No part of this customer notification may be copied or reproduced in any form or by any means without prior written consent of NEC. NEC assumes no responsibility for any errors that may appear in this customer notification. NEC does not assume any liability for infringement of patents, copyrights or other intellectual property rights of third parties by or arising from the use of PRODUCT(S) listed in this customer notification or any other liability arising from the use of such PRODUCT(S).

No license, express, implied or otherwise, is granted under any patents, copyrights or other intellectual property rights of NEC or others. Descriptions of circuits, software and other related information in this customer notification are provided for illustrative purposes of PRODUCT(S) operation and/or application examples only. The incorporation of these circuits, software and information in the design of customer's equipment shall be done under the full responsibility of customer. NEC assumes no responsibility for any losses incurred by customers or third parties arising from the use of these circuits, software and information.

While wherever feasible, NEC endeavors to enhance the quality, reliability and safe operation of PRODUCT(S) the customer agrees and acknowledges that the possibility of defects and/or erroneous thereof cannot be eliminated entirely. To minimize risks of damage to property or injury (including death) to persons arising from defects and/or errors in PRODUCT(S) the customer must incorporate sufficient safety measures in their design, such as redundancy, fire-containment and anti-failure features.

The customer agrees to indemnify NEC against and hold NEC harmless from any and all consequences of any and all claims, suits, actions or demands asserted against NEC made by a third party for damages caused by one or more of the items listed in the enclosed table of content of this customer notification for PRODUCT(S) supplied after the date of publication.

PRODUCT(S) are classified into the following three quality grades: "Standard", "Special" and "Specific". The "Specific" quality grade applies only to PRODUCT(S) developed based on a customer-designated "quality assurance program" for a specific application. The recommended applications of PRODUCT(S) depend on its quality grade, as indicated below. Customers must check the quality grade of each PRODUCT(S) before using it in a particular application.

- "Standard": Computers, office equipment, communications equipment, test and measurement equipment, audio and visual equipment, home electronic appliances, machine tools, personal electronic equipment and industrial robots
- "Special": Transportation equipment (automobiles, trains, ships, etc.), traffic control systems, anti-disaster systems, anti-crime systems, safety equipment and medical equipment (not specifically designed for life support).
- "Specific": Aircraft, aerospace equipment, submersible repeaters, nuclear reactor control systems, life support systems and medical equipment for life support etc.

The quality grade of PRODUCT(S) is "Standard" unless otherwise expressly specified in NEC data sheets or data books, etc. If customers wish to use PRODUCT(S) in applications not intended by NEC, they must contact NEC sales representative in advance to determine NEC's willingness to support a given application.

If the supplied goods/information are subject to Japanese, German, European and/or North American export controls, the customer shall comply with the relevant export control regulations in the event that the goods are exported and/or re-exported. If deliveries are exported without payment of duty at the request of the customer, the customer accepts liability for any subsequent customs administration claims with respect to NEC.

Notes: (1) "**NEC**" as used in this statement means NEC Electronics Corporation and also includes its direct or indirect owned or controlled subsidiaries.

(2) "**PRODUCT(S)**" means any product developed or manufactured by or for NEC (as defined above).

Table of Contents

(A) Related Products

List of related products:

78K0/FC2:
μPD78F0881(A), μPD78F0882(A), μPD78F0883(A)
μPD78F0881(A2), μPD78F0882(A2), μPD78F0883(A2)

μPD78F0884(A), μPD78F0885(A), μPD78F0886(A)
μPD78F0884(A2), μPD78F0885(A2), μPD78F0886(A2)

78K0/FE2:
μPD78F0887(A), μPD78F0888(A), μPD78F0889(A), μPD78F0890(A)
μPD78F0887(A2), μPD78F0888(A2), μPD78F0889(A2), μPD78F0890(A2)

78K0/FF2:
μPD78F0891(A), μPD78F0892(A), μPD78F0893(A)
μPD78F0891(A2), μPD78F0892(A2), μPD78F0893(A2)

(B) Table of Operating Precautions

No.	Outline		μPD78F0881, μPD78F0882, μPD78F0883, μPD78F0884, μPD78F0885, μPD78F0886, μPD78F0887, μPD78F0888, μPD78F0889, μPD78F0890, μPD78F0891, μPD78F0892, μPD78F0893			
		Rev.	ES	CS, MP		
		Rank[Note]	I	K		
1	AFCAN Sleep Mode Wakeup (Specification Change)		×	×		
2	Low-Voltage Detector Reset function (Direction of Use)		×	×		
3	Low-Voltage Detector Interrupt function (Direction of Use)		×	×		
4	Clock generator STOP instruction execution (Direction of Use)		×	×		
5	Flash memory Flash memory programming (Direction of Use)		×	×		

✓: Not applicable
× applicable

Note: The rank is indicated by the letter appearing at the 5th position from the left in the lot number, marked on each product.

(C) Description of Operating Precautions

No. 1	AFCAN Sleep Mode Wakeup (Specification Change)

1. Description

When the AFCAN macro is set into SLEEP mode, it can be waken up by CAN bus activity. This waking up is asynchronous to the operation of the macro and the CPU. By configuration setting, a WAKEUP interrupt can be generated by the AFCAN macro on the wakeup event.
While the interrupt is generated asynchronously, the AFCAN macro may need another dominant edge on the CAN bus, or software clearing of the SLEEP mode, in order to restart its synchronous operation.
During the time, after the interrupt already has been indicated, and before the CAN macro has restarted its synchronous operation, the registers of the AFCAN macro will not operate, because the AFCAN macro still remains in SLEEP mode. This time we will refer to as “wakeup dead time” in the following context.
To resolve from the wakeup dead time, software and/or hardware measures are required.

2. Exclusions

This Operating Precaution is **only applicable** to applications, which are fulfilling at least one of the following three conditions:

- SLEEP Mode of AFCAN is used **and** the possibility to wake up AFCAN by CAN-Bus events is given (see remark 1 below).
- During SLEEP mode of the AFCAN macro, a CAN-Bus wakeup condition occurs, while the AFCAN macro is supplied with clock (see remark 2 below) **and**
  - after waking up from SLEEP mode of the AFCAN macro, the application software does not wait until the SLEEP mode is released by polling the CnCTRL (PSMODE) register, before continuing operation with the AFCAN macro (see remark 3 below) and
  - the CPU can reach instructions, where AFCAN registers are accessed while the AFCAN macro is still in SLEEP mode, due to the missing waiting condition.
- During SLEEP mode of the AFCAN macro, a CAN-Bus wakeup condition occurs, while the AFCAN macro is supplied with clock (see remark 2 below) **and**
  1. after waking up from SLEEP mode of the AFCAN macro, the CAN Bus Transceiver generates a **long-lasting or permanent dominant level** to the CRXD input of the AFCAN macro, instead of the propagated CAN-Bus level.

Remarks:

1. If the CAN-Bus Transceiver does not propagate the CAN-Bus signal, while the AFCAN macro is in SLEEP mode, and also does not forward a wakeup signal to CRXD, **this Operating Precaution is not applicable**.
2. The clock supply to the AFCAN macro can be stopped, depending on the features of the device, and the system design of the application. If the clock supply to the AFCAN macro is stopped, while a wakeup condition occurs, **this Operating Precaution is not applicable.**
3. The maximum waiting time for this loop can be up to 10 bits of the CAN-Bus Baudrate. Waiting while retrying to clear CnINTS (Bit 5) can be used alternatively.

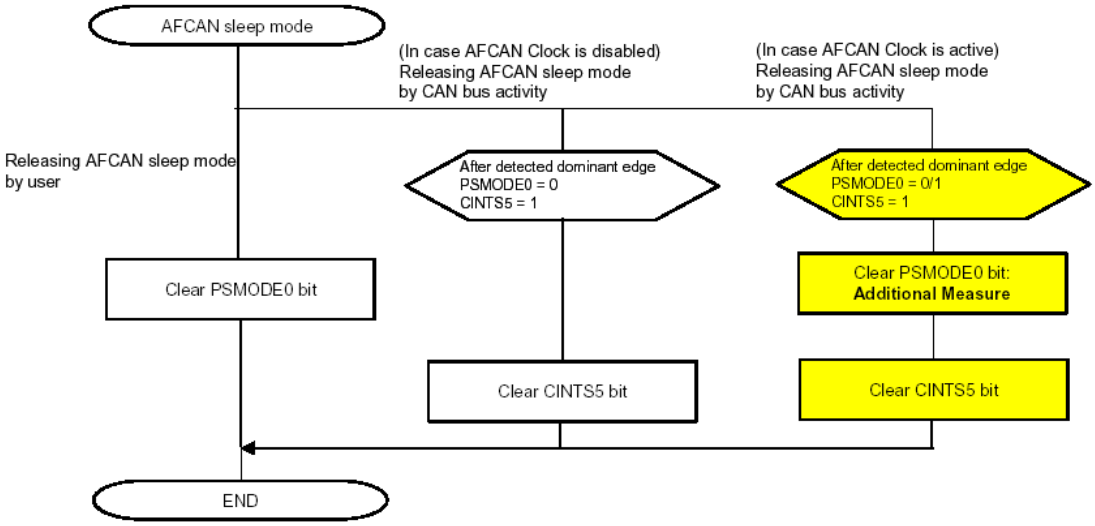
All other applications are not affected by this Operating Precaution.

3. Application Dependency

3.1 Overview

The following flowchart illustrates, how and whether additional measures have to be taken in software, to avoid the wakeup dead time.

Figure 1-1: Additional Measures in case AFCAN clock is active when waking up



3.2 Not affected Applications

3.2.1 Applications not using SLEEP mode

If SLEEP mode is not used, **this Operating Precaution is not applicable.**

3.2.2 Applications waking up from SLEEP mode by User Request only

If there is no condition, when SLEEP mode can be left by CAN-Bus activity, but only on User Request (by clearing the PSMODE flag by software), **this Operating Precaution is not applicable**.

3.2.3 Applications using a CPU Power Save Mode

If the clock to the AFCAN macro is disabled, while it is waken up from SLEEP mode, **this Operating Precaution is not applicable**.
This means, if the user selects a power save mode of the target device, which switches off the clock of the AFCAN macro, immediately after it had been set into SLEEP mode, like the CPU STOP mode, the precaution needs not to be considered.
This is associated with the software improvement hints below.

3.3 Affected Applications

3.3.1 Applications not waiting until SLEEP mode is left

If bus transceivers are used in conjunction with AFCAN, which will propagate the CAN bus signal to AFCAN permanently (not switched off or not in power saving modes), or, if bus transceivers are used in conjunction with AFCAN, which will propagate the unmodified CAN-Bus signal when waking up from a power save mode, the wakeup dead time lasts from the first recessive-to-dominant edge of the CAN-Bus signal, which generates the wake-up, until the next recessive-to-dominant edge of the CAN-Bus signal.

The worst case (maximum length) of the wakeup dead time, is given by the CAN bus speed and the rule of the CAN bus about the frequency of recessive-to-dominant edges. Given by the stuffing rule, at least every 10 bits, a recessive-to-dominant edge must occur.

If during the wakeup dead time, the CPU waits until the SLEEP mode is indicated to be cleared (either by polling the PSMODE flag, or by retrying to clear CnINTS[5]), **this Operating Precaution is not applicable**. In this case, the improvement hint according to 4.2.2 is followed implicitly.

If during the wakeup dead time, the CPU does not perform any access to the AFCAN macro in any case, **this Operating Precaution is not applicable**.

3.3.2 Applications using Bus Transceivers generating long-lasting dominant CAN-Bus Signals

If bus transceivers are used in conjunction with AFCAN, which *generate a permanent or long-lasting dominant level* when waking up from a power save mode, the Operating Precaution must be considered in any case.

In this case, the wakeup dead time lasts from the first recessive-to-dominant edge of the CAN bus signal, which generates the wake-up, until the next recessive-to-dominant edge of the CAN bus signal, depending on the behaviour of the CAN bus transceiver.

If no further dominant edge on the CAN bus occurs (in case of some CAN transceivers, which only provide one single edge on waking up), the time until SLEEP mode is left may become endless. Therefore, the waking up procedure of AFCAN regarding software, must be adjusted according to 4.1.1.

4. Software Improvement Hints

4.1 Recommended WAKEUP Handling by Software

4.1.1 Clearing the SLEEP Mode by Software

Within the WAKEUP interrupt routine, before accessing any other register or area of AFCAN, the SLEEP mode can be canceled by software, followed by a clearance of the WAKEUP interrupt flag.

Doing so, the AFCAN macro will start its synchronous operation right after these accesses. In the following C-code example, replace the objects in “<>” brackets by the hardware locations within your implementation. Use the appropriate access types, as described in the User’s Manual.

```
WAKEUP INTERRUPT VECTOR -->
                                <CnCTRL_PSMODE> = 0;      /* Clear SLEEP Mode */
                                <CnINTS_CINTS5> = 1;      /* Clear INTS5 */
                                ...
                                /* following other parts of interrupt routine */
                                ...
```

Remark: Clearing INTS5 is required to get another WAKEUP interrupt anyway, by specification.

4.2 Other WAKEUP Handling Hints

4.2.1 Switching off the Clock Supply to AFCAN, while in SLEEP Mode

If the clock supply to the AFCAN macro is stopped, while it is in SLEEP mode, the synchronization of the WAKEUP works without any restriction. To achieve this, the documentation of clock controlling unit of the target device should be consulted. Usually this is performed by setting the STOP mode of the CPU of the target device.
However, the user has to consider, that there must not be any WAKEUP condition (dominant level on the CAN-Bus), while the software is executing between setting SLEEP mode and stopping the AFCAN clock.

4.2.2 Using a Waiting Loop within the WAKEUP interrupt routine

Within the WAKEUP interrupt routine, create a waiting loop, which tests the capability of clearing the WAKEUP interrupt flag within AFCAN, by checking the actual power save mode.
In the following C-code example, replace the objects in "<>" brackets by the hardware locations within your implementation. Use the appropriate access types, as described in the User's Manual.

```
do
{
        AFCAN_SleepStatus = <CnCTRL_PSMODE>
        if( AFCAN_SleepStatus != 0 )
        {
                /* macro is still in SLEEP mode (waiting for latency time) */
                <CnINTS_CINTS5> = 1;          /* repeated trying to clear CINTS5 */
        }
} while( AFCAN_SleepStatus != 0 );
```

This improvement hint **cannot be applied**, if a CAN-Bus-Transceiver is attached to AFCAN, which generates a permanent or long-lasting dominant level to the FCRXDn receive input pin, if a wakeup condition occurs. Missing another dominant edge on the bus, the synchronisation will not happen, and the loop could run endlessly.

4.2.3 Using INIT Mode instead of SLEEP Mode

In this case, the waking up by CAN-bus activity must be performed via another free external interrupt. The CAN receive signal must be distributed on the FCRXDn pin, and to another external interrupt pin in parallel.
Using this external interrupt, the AFCAN macro can be restored into the previous operation mode. This implementation will not use the SLEEP mode of AFCAN at all, and use the INIT mode instead.

No. 2	Low-Voltage Detector Reset function (Direction of Use)

Details

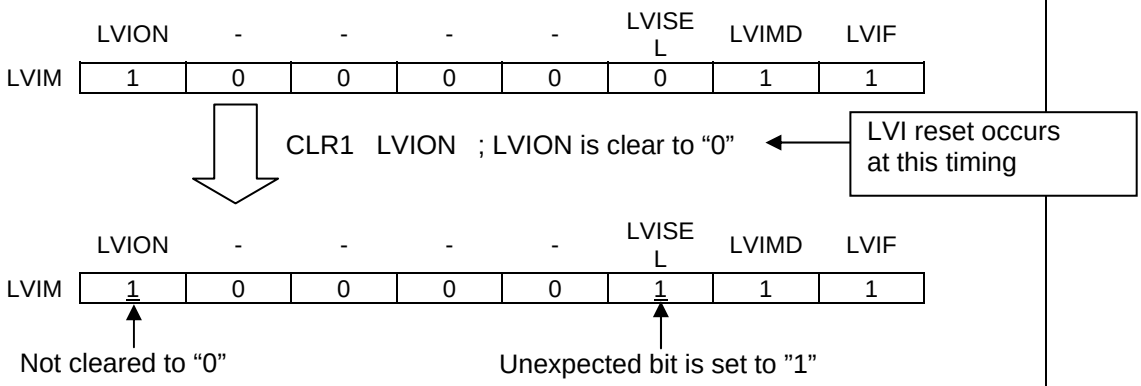
If a reset is generated from the Low Voltage Detection (LVI) function at the same time that the LVIM or LVIS register is being written, then one or more bits of the register being written can be changed.

To give an example, a change from “0” to “1” would happen.

- LVI will be not stopped.
- Low voltage detect pin will be changed to “EXLVI” pin.
- LVI detect voltage will be changed to the lower level than the level which is set by LVIS.

Figure 2-1 shows the example in case this phenomenon occurs at the timing that LVION flag is operated.

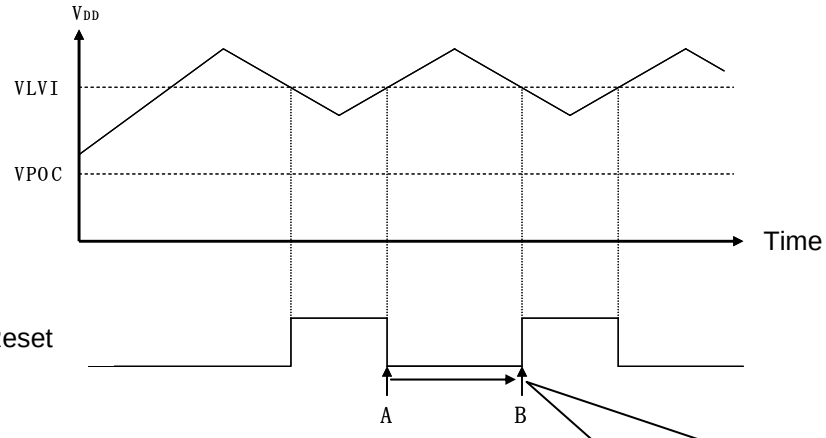
Figure 2-1. Example of phenomenon which occurs if LVION flag is cleared to ”0”



If a Reset is generated from any other source besides LVI, this phenomenon does not occur.

When VDD falls below V_{LVI} a Reset from the LVI is generated. If data is written to the LVIM or LVIS register at the same time that the reset signal is generated, then incorrect data can be written to the target register

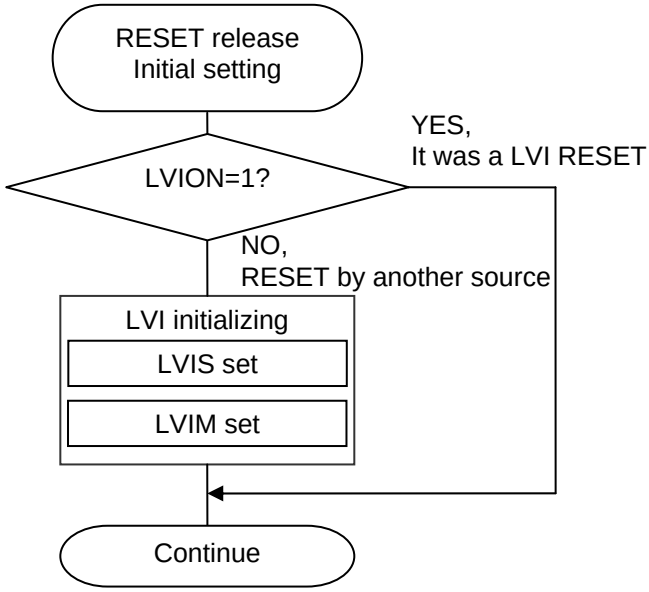
Figure 2-2 Example of Phenomenon



Workaround

1. If the LVI is to be configured for Reset mode, do not write to the LVIM register or the LVIS register once the LVI has been configured.
2. Before setting the LVIM/LVIS register, please confirm that bit-7 of the LVIM register (LVION) is “0”. If LVION is “1”, do not execute the write instruction for LVIM/LVIS. By this workaround, LVIM/LVIS will be set when an external reset or a POC reset or a Watchdog timer reset occurred, but not be set when a LVI reset occurred.

Both LVIRF and WDTRF may become “1” because the Reset control flag register (RESF) will not be cleared by a LVI reset or a Watchdog timer reset. When LVIRF bit is used for this judgment, it will be possible to judge if a LVI reset occurred or not, but additional instructions are needed to judge whether LVIM/LVIS is cleared by WDT reset or not. Therefore, please use LVION flag for the workaround.



No. 3	Low-Voltage Detector Interrupt function (Direction of Use)

Details

In case the Low Voltage Detector (LVI) is used to generate an interrupt when the voltage drops below the detection voltage (VDD or EXLVI) an unexpected LVI interrupt could be generated (LVIIF flag will be set), if the following conditions are true:

1. The voltage drops below the detection voltage (selected by LVIS or EXLVI) and
2. The LVIIM.7 (LVION) is cleared to “0”.

Workaround

To prevent an unexpected LVI interrupt (INTLVI), please mask the LVIMK flag (LVIMK = 1) before setting LVION = 0 and clear the LVIIF flag afterwards.

No. 4	Clock generator STOP instruction execution (Direction of Use)

Details

If the microcontroller is using the internal high-speed oscillator (fRH) as the CPU clock, then when a STOP instruction is executed, under certain circumstances the device will enter a state of undefined operation and it will not exit from STOP mode by interrupt. There are certain timing conditions between the execution of the STOP instruction and the internal high speed oscillator which cause this situation and they are listed below.

The instruction is executed 888-889 clock cycles after the internal high speed oscillator is enabled (RSTOP set to 0).
The instruction is executed 888-889 clock cycles after the device exits from STOP mode, and the CPU was operating from the high speed internal oscillator prior to entering STOP mode and continues to operate from the same clock source when it exits STOP mode.
The instruction is executed 408-701 clock cycles after the device finishes initializing after a non-POC (power up) Reset; i.e. WDT, LVI, external reset.

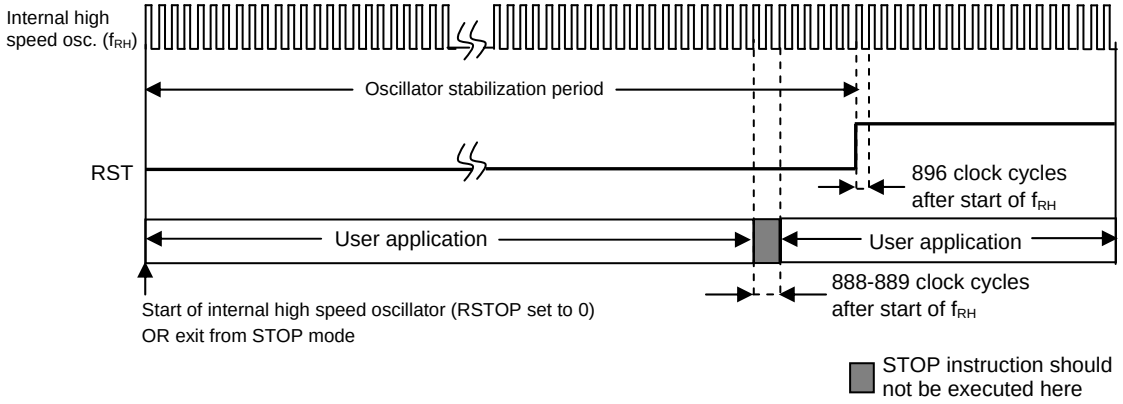


Fig. 4－1 Operation of internal high-speed oscillator after initial starting or after exit from STOP mode

NOTE: Figure is not drawn to scale.

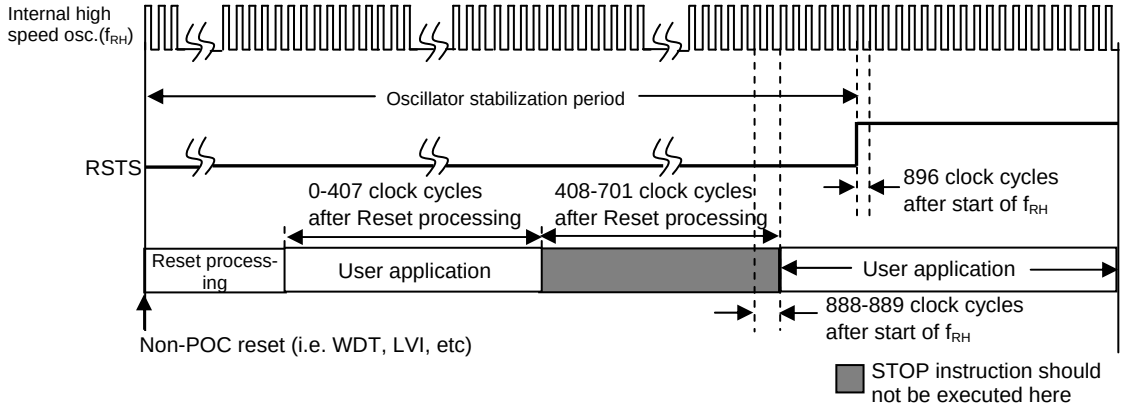


Fig. 4－2 Reset due to factors other than POC

NOTE: Figure is not drawn to scale

When one of the described situations occurs, the device abnormally enters into STOP mode. As a result, the following conditions occur-

- The device will not exit STOP mode when an interrupt occurs
- The internal high speed oscillator does not stop. Hence the current consumption is greater than what is specified for STOP mode; approximately 150-400uA.

If the watchdog timer is enabled (WDTON=1) and it is configured to operate in STOP mode (Option Byte->LSROSC=1), then it can Reset the device once the counter overflows and release the device from STOP mode.

Workaround

This condition can be avoided using a software modification. There are two different modifications which can be utilized.

1) Postpone the execution of the STOP instruction until a check is performed on the RSTS bit and it is verified that it is 1.
2) A delay is implemented in the software such that it can be insured that the STOP instruction will not be executed until after the previously mentioned conditions have passed.

It is suggested to use modification (1) if the STOP instruction will be executed during interrupt processing.

Following is a table of application conditions and which modifications will work for each.

Application condition	Modification	
	(1)	(2)
STOP instruction will be executed shortly after the internal high speed oscillator begins operation	✓	✓
STOP instruction will be executed shortly after exiting STOP mode	✓	✓
STOP instruction will be executed shortly after a non-POC reset	✓	✓
The time from which the internal high speed oscillator begins to when the STOP instruction will be executed can not be defined	✓	✗

✓: modification will work

✗ : modification will not work

No. 5	Flash memory Flash memory programming (Direction of Use)

Details

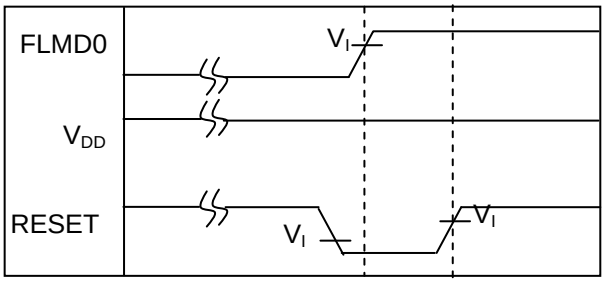
There is an issue with the device not entering into flash programming mode. There are two situations in which the microcontroller may not enter into the programming mode and they are listed below.

- When the Run After Disconnect function is used during programming with the flash memory programmer (PG-FP5) or MINICUBE2.
- When the programming environment has been configured based on the Application Note (Programmer) (U17739EJ2V0AN00) and when entering into flash programming mode during user program operation (Figure 5-1 (1))

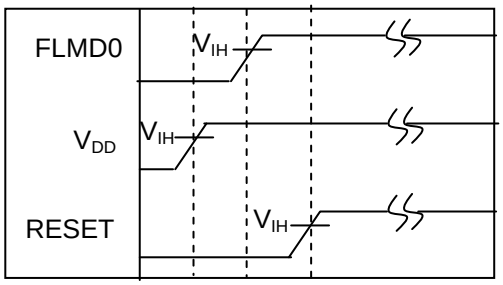
Even if either of the above situations exists, the device may still correctly enter into programming mode and the flash can be successfully programmed. If the verify indicates correct programming, then the flash programming worked correctly.

This issue is not applicable for the circumstances listed below.

- When entering into the programming mode at the same time as power-on
- When performing self programming
- When performing EEPROM emulation



① Entering into flash programming mode during user program operation



② Entering into flash programming mode at the same time as power-on

Figure 5-1 Methods of entering flash memory programming mode

If the duration of the reset signal applied to the external RESET pin is shorter than 1,950 ms when entering the flash programming mode during user program operation without dropping the power supply voltage to the level of the POC detection voltage (VPOC = 1.59 V ±0.15 V), the following phenomenon may occur.

- A POC reset occurs upon reset release and the flash memory programming mode is not entered normally. Consequently, the user program is executed without performing programming processing.

Workaround

Secure a reset period for 1,950 ms or longer by controlling the external RESET pin when entering the flash memory programming mode.

For those configuring the programming environment based on the Application Note (Programmer) (U17739EJ2V0AN00). Please apply and use the workaround.

For 3rd party programmers please contact directly the manufacturer of your programmer.

(D) Valid Specification

Item	Date published	Document No.	Document Title
1	March 2007	U17555EJ4V0UD00 or later	User's Manual 78K0/FC2
2	March 2007	U17554EJ4V0UD00 or later	User's Manual 78K0/FE2
3	March 2007	U17553EJ4V0UD00 or later	User's Manual 78K0/FF2

(E) Revision History

Item	Date published	Document No.	Comment
1	Sep 8, 2005	U18390EE1V0IF00	1st Release
2	January 17, 2008	U18390EE2V0IF00	1st Update Items 2 and 3 added
3	June 27, 2008	U18390EE3V0IF00	2nd Update - Items 4 and 5 added - List of related products added (Chapter A) - Included the Rank codes in the table (Chapter B) - Modified the text for item 2 in (Chapter C) - Renumber the figures to be in line with the item number (Chapter C)