NEC

Application Note

V850

32-Bit Single-Chip Microcontrollers

Clock / Standby Functionality

V850E Series

V850ES Series

Document No. U18910EE1V0AN00
Date Published 30 th August 2007

Printed in Germany

[MEMO]

- **The information in this document is current as of April, 2007. The information is subject to change without notice. For actual design-in, refer to the latest publications of NEC Electronics data sheets or data books, etc., for the most up-to-date specifications of NEC Electronics products. Not all products and/or types are available in every country. Please check with an NEC Electronics sales representative for availability and additional information.**
- No part of this document may be copied or reproduced in any form or by any means without the prior written consent of NEC Electronics. NEC Electronics assumes no responsibility for any errors that may appear in this document.
- NEC Electronics 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 NEC Electronics products listed in this document or any other liability arising from the use of such products. No license, express, implied or otherwise, is granted under any patents, copyrights or other intellectual property rights of NEC Electronics or others.
- Descriptions of circuits, software and other related information in this document are provided for illustrative purposes in semiconductor product operation and application examples. The incorporation of these circuits, software and information in the design of a customer's equipment shall be done under the full responsibility of the customer. NEC Electronics assumes no responsibility for any losses incurred by customers or third parties arising from the use of these circuits, software and information.
- While NEC Electronics endeavors to enhance the quality, reliability and safety of NEC Electronics products, customers agree and acknowledge that the possibility of defects thereof cannot be eliminated entirely. To minimize risks of damage to property or injury (including death) to persons arising from defects in NEC Electronics products, customers must incorporate sufficient safety measures in their design, such as redundancy, fire-containment and anti-failure features.
- NEC Electronics products are classified into the following three quality grades: "Standard", "Special" and "Specific".
  The "Specific" quality grade applies only to NEC Electronics products developed based on a customer-designated "quality assurance program" for a specific application. The recommended applications of an NEC Electronics product depend on its quality grade, as indicated below. Customers must check the quality grade of each NEC Electronics product 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 NEC Electronics products is "Standard" unless otherwise expressly specified in NEC Electronics data sheets or data books, etc. If customers wish to use NEC Electronics products in applications not intended by NEC Electronics, they must contact an NEC Electronics sales representative in advance to determine NEC Electronics' willingness to support a given application.

(Note)
(1) "NEC Electronics" as used in this statement means NEC Electronics Corporation and also includes its majority-owned subsidiaries.
(2) "NEC Electronics products" means any product developed or manufactured by or for NEC Electronics (as defined above).

M8E 02. 11-1

Regional Information

Some information contained in this document may vary from country to country. Before using any NEC product in your application, please contact the NEC office in your country to obtain a list of authorized representatives and distributors. They will verify:

- Device availability
- Ordering information
- Product release schedule
- Availability of related technical literature
- Development environment specifications (for example, specifications for third-party tools and components, host computers, power plugs, AC supply voltages, and so forth)
- Network requirements

In addition, trademarks, registered trademarks, export restrictions, and other legal issues may also vary from country to country.

NEC Electronics America Inc.
Santa Clara, California
Tel: 408-588-6000
800-366-9782
Fax: 408-588-6130
800-729-9288

NEC Electronics (Europe) GmbH
Duesseldorf, Germany
Tel: 0211-65 030
Fax: 0211-65 03 1327

Sucursal en España
Madrid, Spain
Tel: 091- 504 27 87
Fax: 091- 504 28 60

Succursale Française
Vélizy-Villacoublay, France
Tel: 01-30-67 58 00
Fax: 01-30-67 58 99

Filiale Italiana
Milano, Italy
Tel: 02-66 75 41
Fax: 02-66 75 42 99

Branch The Netherlands
Eindhoven, The Netherlands
Tel: 040-244 58 45
Fax: 040-244 45 80

Branch Sweden
Taeby, Sweden
Tel: 08-63 80 820
Fax: 08-63 80 388

United Kingdom Branch
Milton Keynes, UK
Tel: 01908-691-133
Fax: 01908-670-290

NEC Electronics Hong Kong Ltd.
Hong Kong
Tel: 2886-9318
Fax: 2886-9022/9044

NEC Electronics Hong Kong Ltd.
Seoul Branch
Seoul, Korea
Tel: 02-528-0303
Fax: 02-528-4411

NEC Electronics Singapore Pte. Ltd.
Singapore
Tel: 65-6253-8311
Fax: 65-6250-3583

NEC Electronics Taiwan Ltd.
Taipei, Taiwan
Tel: 02-2719-2377
Fax: 02-2719-5951

Introduction

Target Readers

This application note is intended for users who understand the functions of the V850E / V850ES and will use this product to design application systems.

Purpose

The purpose of this application note is to help users to understand the functionality of the clock generator of V850E / V850ES and the benefits of standby modes in principal.
This application note describes the clock generator of V850ES/FG3 (μPD70F337x) as example for whole V850E / V850ES – cores.
The clock generators of the V850E / V850ES – cores differ from each other, but the principal is the same.
The handling and usage shown in this document are for reference only. Correct operation is not guaranteed if these samples are implemented as they are described here.
The user has to adapt the usage and handling to his application specific needs.

Organization

This manual consists of the following main sections.

- Features of the V850ES/Fx3 clock generator
- Details of the different clock sources
- Clock operation state transition
- Details of the different standby modes
- Current consumption in different standby modes

How to Read This Manual

It is assumed that the reader of this manual has general knowledge in the fields of electrical engineering, logic circuits, and microcontrollers.

- To gain a general understanding of functions:

→ Read this manual in the order of the **CONTENTS**. The mark “<R>” shows major revised points. The revised points can be easily searched by copying an “<R>” in the PDF file and specifying it in the “Find what:” field.

- To learn more about the 78K0/Kx2’s hardware functions:

→ See the user’s manual of each 78K0 product.

Conventions

Data significance: Higher digits on the left and lower digits on the right
Active low representation: xxx (overscore over pin or signal name)
Note: Footnote for item marked with **Note** in the text
Caution: Information requiring particular attention
Remark: Supplementary information
Numeral representation: Binary.....xxxx or xxxxB
Decimal xxxx
Hexadecimal .....xxxxH

Table of Contents

List of Figures

List of Tables

CHAPTER 1 FEATURES OF THE V850ES/FX3 CLOCK GENERATOR

1.1 The clock generator offers four clock sources

Table 1-1 Clock generators clock sources

Clock Source	Description
Main oscillator	A built-in oscillator that requires an external crystal or resonator with a frequency between 4 MHz and 16 MHz
Sub oscillator	A built-in oscillator that requires an external crystal (32.768 kHz) or an external RC resonator (40 kHz)
Internal low-speed oscillator	An internal oscillator without external components and a nominal frequency of 240 kHz
Internal high-speed oscillator	An internal oscillator without external components and a nominal frequency of 8 MHz

1.2 Operating and standby - modes operated by the clock generator

There are different modes settable within the clock generator which allows fastest speed operation up to sleep – operation, which influences the performance and power consumption of the controller. Within some of these modes, additional speed optimisation via PLL/SSCG and processor clock selection is possible. These settings are not discussed within this application note. In principal: as faster the CPU is clocked, as more power is consumed.

The following operation / standby modes can be set:

Table 1-2 Operation and standby modes

Mode	Description
Operating	The CPU is executing the instructions, clocked by mainclock or internal high-speed oscillator. The peripherals may operate.
HALT	Only the operating clock of the CPU is stopped (no instruction execution).
IDLE 1	All the internal operations of the chip except the oscillator, PLL/SSCG and flash memory are stopped.
IDLE 2	All the internal operations of the chip except the oscillator are stopped.
STOP	All the internal operations of the chip except the subclock oscillator are stopped.
Subclock operation	The subclock is used as the CPU system clock. Either internal low-speed oscillator or the sub oscillator is clocking the CPU.
Sub - IDLE	All the internal operations of the chip except the oscillator, PLL/SSCG and flash memory are stopped during subclock operation mode.

1.3 Clock sources and standby modes in V850ES series

Table 1-3 Different clock sources and standby modes within V850 ES series

V850ES Line	**V850ES S - Line**		**V850ES F - Line**		**V850ES K – Line**		
V850ES Sub - Line	V850ES / Sx2	V850ES / Sx3	V850ES / Fx2	V850ES / Fx3	V850ES / Kx1+	V850ES / Kx1H	V850ES / Kx2
	μPD70F3260 - μPD70F3288	μPD70F3333 - μPD70F3368	μPD70F3230 - μPD70F3239	μPD70F3370 - μPD70F3385	μPD70F3302 - μPD70F3318	μPD70F3207 - μPD70F3218	μPD70F3726 - μPD70F3734
V850ES Sub - Line	V850ES / Jx2	V850ES / Jx3	V850ES / Hx2	V850ES / Hx3			
	μPD70F3715 - μPD70F3724	μPD70F3739 - μPD70F3746	μPD70F3700 - μPD70F3712	μPD70F3747 - μPD70F3759			
Mainclock X1, X2	2.5 – 10 MHz	2.5 – 10 MHz	4 – 5 MHz	4 – 16 MHz	2 – 10 MHz	2 – 10 MHz	2 – 10 MHz
Max. internal clock	20 MHz	32 MHz	20 MHz	32 MHz (≤ 256 k Flash) 48 MHz (≥ 384 k Flash)	20 MHz	20 MHz	20 MHz
Subclock XT1, XT2	32 kHz crystal	32 kHz crystal	32 kHz crystal or 20 kHz R/C	32 kHz crystal or 40 kHz R/C	32 kHz crystal	32 kHz crystal	32 kHz crystal
Internal low speed oscillator	200 kHz typ.	220 kHz typ.	200 kHz typ.	240 kHz typ.	240 kHz typ.	240 kHz typ.	240 kHz typ.
Internal high speed oscillator	No	No	No	8 MHz typ.	No	No	No
HALT instruction	Yes	Yes	Yes	Yes	Yes	Yes	Yes
IDLE 1 mode	Yes	Yes	Yes	Yes	Yes	Yes	Yes
IDLE 2 mode	Yes	Yes	Yes	Yes	No	No	No
STOP mode	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Subclock operation	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Subclock IDLE	Yes	Yes	Yes	Yes	Yes	Yes	Yes

1.4 Fundamental description of the clock generator

A simplified block diagram of the clock generator of V850ES/Fx3 is shown in Figure 1-1. This figure shall illustrate the clock sources, clock pathes and setting of standby modes on a glance.

Figure 1-1. Simplified block diagram of the clock generator of V850ES/Fx3

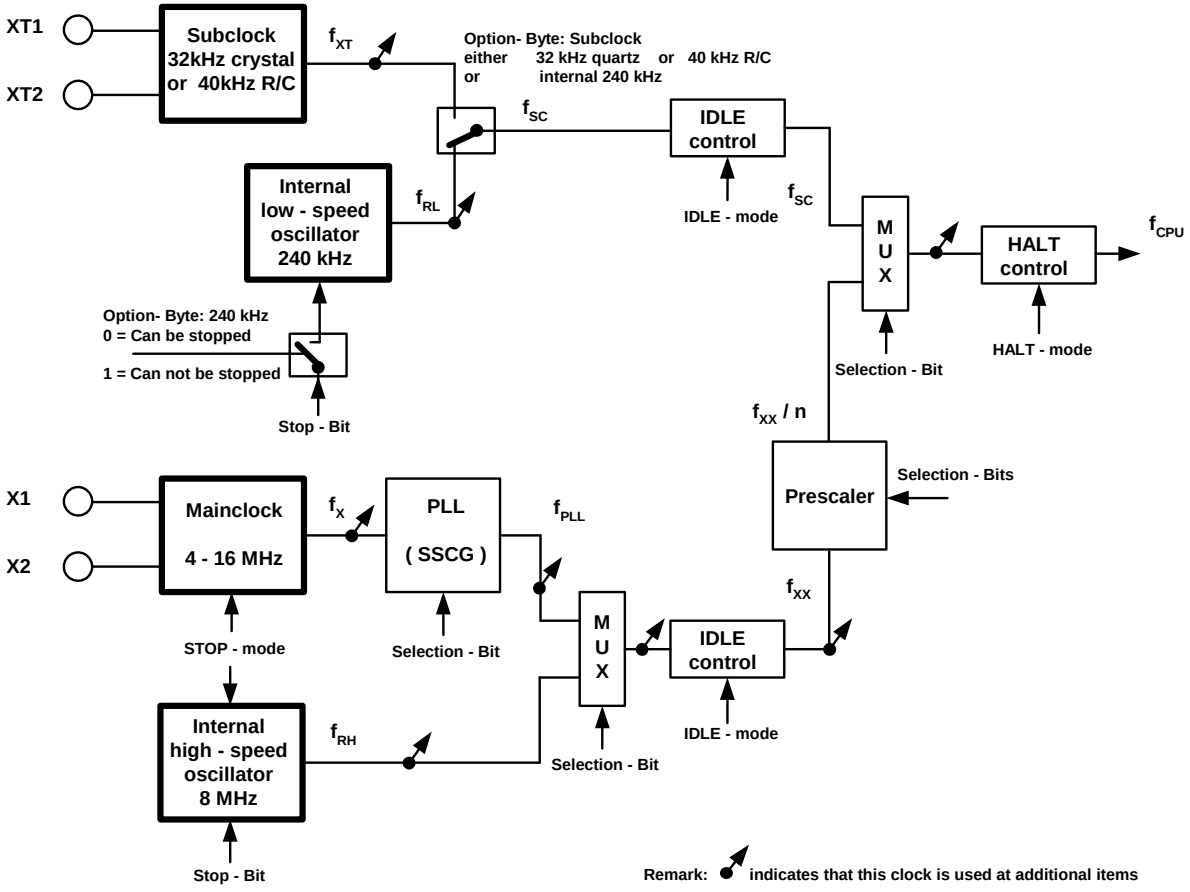


Figure 1-1 shows that all four clock sources can be selected as clock source for CPU instruction execution. After reset release, the internal high-speed oscillator (8 MHz) is clock source for the CPU per default. Using this clock, the CPU can select an other clock as its own source. Selection or clock enabling / disabling is done by instructions.

Nevertheless, switching from one clock to an other one has to be done carefully and in an prescribed manner.

Be careful when stopping a clock source by a Stop – Bit. When doing so, the clock to be stopped must not be the actual CPU clock source. Otherwise, the CPU has switched off its own clock and functionality is given after reset only.

CHAPTER 2 DETAILS OF THE DIFFERENT CLOCK SOURCES

It depends on the applications requirements which clocks are used and which are not necessary. It is even possible to run the application without any external clock source.

2.1 External clock sources

2.1.1 Main oscillator at pin X1 / X2

As mainclock at the pins X1 / X2, a quartz or ceramic resonator has to be soldered externally. In V850ES/Fx3, the frequency has to be in the range of 4 MHz up to 16 MHz. A quartz or ceramic resonator has a small frequency tolerance, so the mainclock should be used during operation mode to achieve exact timings in time relevant designs.
A quartz is more precise in frequency tolerance than a ceramic resonator. A quartz needs a relative long time (up to several milliseconds) to get stable. A ceramic resonator is cheaper and needs less time to get stable than a quartz.
During that time when the oscillator is not stable, this undefined clock must not be used as CPU clock source, otherwise the CPU might fail. An “oscillation stabilisation time” has to be elapsed before the quartz or ceramic resonator can be used as CPU clock. A separate counter is implemented to supervise that time. This counter (Oscillation Stabilisation Time Select register = OSTS) can be set to the individual requirements for the selected quartz or ceramic resonator when STOP – or IDLE - mode is released. After reset release, the OSTS - time is fixed.

Usually, the mainclock is used as clock source for the controller. Due to this is an external component, it is jeopardised by external influence. It might be destroyed by mechanical reason. In this case, the application will be out of order. Depending on the kind of application, this might destroy the surroundings.
Therefore, a clock monitor is implemented in the controller. If the mainclock fails, an internal reset is generated. The CPU starts clocked by the internal high-speed oscillator (8 MHz). The software can detect that clock monitor has triggered the reset and the software can react on such malfunction accordingly.

2.1.2 Subsystem oscillator at pin XT1 / XT2

At pin XT1 / XT2 either an external crystal (32.768 kHz) or an external R/C resonator can be connected. The selection is done via an option byte. That means either a crystal or an R/C oscillator is used in that specific application. It is not possible, to use both in the same application.
The subsystem clock can be used internally for timer clock source as well for CPU clock supply. There is no possibility to stop the subsystem clock source by software. When it is implemented, this clock oscillates all time.

Remark: An option byte is a byte in the flash area (pre-defined address) where fixed selections for the processor can be made. This bit setting can not be changed by software.
The task of the option byte is adequate to the mask option setting of a ROM – device.

2.1.2.1 Crystal resonator as subsystem clock

At the pins XT1 / XT2, a 32.768 kHz crystal resonator can be soldered externally. This clock can be used for driving a specific watch timer. When exact 32.768 kHz is used, it is easy to build up a realtime clock.

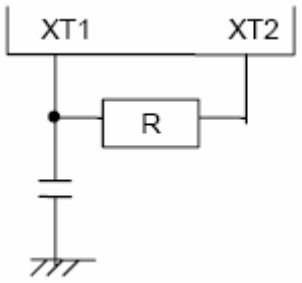
When a realtime clock is build up, there is no intention to stop this clock. So, there is no bit to stop that clock source. Due to its nature, a crystal resonator needs long time (up to several seconds) to get stable. So, after reset release this stabilisation time has to be elapsed up to the clock is stable and precise enough for realtime clock functionality.

2.1.2.2 R/C oscillator as subsystem clock

At the pins XT1 / XT2, a resistor and a capacitor can be soldered externally to build up an R/C oscillator of 40 kHz typical. The R/C oscillator needs shorter time (up to 0.1 second) to get stable than an crystal resonator.

On the other hand, R/C oscillators tolerance is very big. It is not possible to stop the R/C clock source by software, when implemented.

Figure 2-1. Connection of an external R/C oscillator



2.2 Internal clock sources

The internal oscillators have no connectionto any pin. Therefore they can not be influenced from externally. They have a short oscillation stabilisation time but a big frequency tolerance compared to a crystal resonator. Due to these oscillators are none sensitive for external influence, these clocks are mainly used for security and supervision function.

The internal oscillators can also be used as clock source for CPU and some peripherals.

Be careful to use these internal clocks as clock source for exact timings or for time relevant software routines. The frequency varies in a brought range when Vdd changes or temperature drifts. If you apply, be sure that the timing is applicable for your needs over the whole tolerance range.

2.2.1 Internal low-speed oscillator (240 kHz typ.)

The internal low-speed oscillator is intended to be used as supervision clock for the mainclock within the clock monitor.
Furthermore, low-speed oscillator can be used e.g. for cyclic wake up of the CPU in low power modes.

2.2.2 Internal high-speed oscillator (8 MHz typ.)

After reset release, the CPU starts up clocked by the internal high-speed oscillator. Even an external quartz is connected as mainclock, but destroyed, the CPU can operate with the high-speed oscillator. If applicable, when a big frequency tolerance is allowed, the whole system can operate with internal oscillators only.

2.3 Selection of the CPU´s subsystem – clock

For V850ES/Fx3 CPU´s subclock operation mode, one of three different clock sources is selectable:

- Internal low-speed oscillator (240 kHz typ.)
- Crystal, externally connected to XT1 / XT2 (32.768 kHz)
- R/C - oscillator, externally connected to XT1 / XT2 (40 kHz typ.)

The selection which one of these three clocks is used has to be done via an option byte. So, depending on the setting of this option byte, only one of these three clocks is fixed as CPU´s subsystem clock in a specific application.

During program flow, switching between Internal low-speed oscillator and that clock, connected to XT1/XT2 is not possible. Nevertheless, for the peripherals, different clock sources can be used.

CHAPTER 3 CLOCK OPERATION STATE TRANSITION

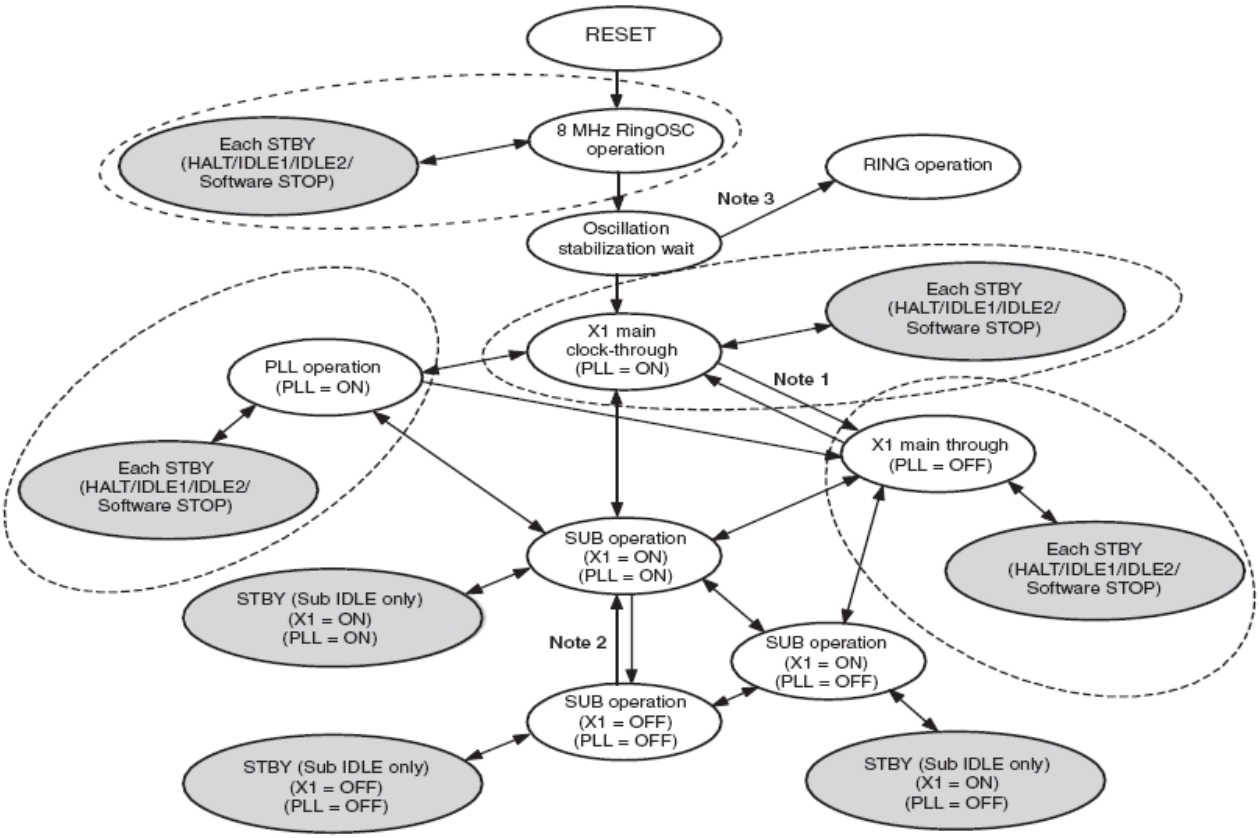
3.1 Transition between the clock sources as CPU´s clock source

Figure 3-1 shows the possibilities for the transition between the several clock sources. After reset release, CPU is clocked by the internal high-speed oscillator. Mainclock has to be enabled first before it is oscillating, ready to be used as CPU´s clock source.Transition is performed via setting of the appropriate bits in the clock generator control registers.

Furthermore, Figure 3-1 shows the different standby – modes which can be performed when the CPU is clocked by the different clock sources.

Keep in mind, that this figure shows only that settings which are allowed. Transitions which are not shown are undefined. E.g. after reset, direct transition from internal high-speed oscillator to internal low-speed oscillator is not possible.

Figure 3-1. Clock operation state transition



Note 1. PLL lockup time is required (LOCK bit of LOCKR register = 0).
2. Oscillation stabilization time must be secured by program.
Every transition from stand-by to PLL operation (PLL = ON)
Every transition from stand-by to X1 main clock-through (PLL = ON)
3. If the Watchdog Timer overflows (resets) while the oscillation stabilization time is being counted, the CPU starts clock operation with the 240 kHz internal low-speed oscillator f_{RL}.

CHAPTER 4 DETAILS OF THE DIFFERENT STANDBY MODES

4.1 Before entering standby mode

There are several power saving standby modes available in V850ES/Fx3 series. Before entering any standby mode, the controller has to be prepared for proper wake up from standby mode.

Before entering standby mode, within the power save control register (PSC), the stand-by mode release signals have to be enabled. Three different types of signals can wake up the controller:

- Stand-by mode release by the watchdog timer interrupt (INTWDT2) signal
- Stand-by mode release by none maskable interrupt (NMI) pin input
- Stand-by mode release by any maskable interrupt request signal

In general, target of the standby modes is to reduce the power consumption as much as possible. When a clock source is switched off in standby mode, power consumption will decrease additionally. On the other hand, when a clock source is switched off, user has to take some time for oscillation stabilisation into account. It depends on the application which power down mode is possible and which one not and if a clock should be switched off or not. If the necessary reaction time is too slow, oscillator has to keep operating.

4.2 Standby modes

When the CPU operates with internal high-speed oscillator or mainclock, each one of the standby – modes HALT, IDLE 1, IDLE 2 and STOP is possible. When the CPU operates with subsystem – clock, restricted standby – modecan be assigned. For details, please refer to Figure 3-1.

4.2.1 HALT

The HALT – mode is entered by executing the HALT instruction. In this mode, only the clock for the CPU is stopped, so program execution is stopped.
The clock oscillators continue operating, supply to the other on-chip peripheral functions continues, the contents of the internal RAM before the HALT mode was set are retained.
The on-chip peripheral functions that are not dependent upon the instruction processing of the CPU continue operating.

HALT – mode can be released by a non-maskable interrupt request signal (NMI pin input or INTWDT2 signal), unmasked external interrupt request signal, unmasked internal interrupt request of a peripheral function that can operate in the HALT mode, or reset signal. Due to the clock is still operating during release, program execution continues just after release signal occurs.

The HALT - mode can reduce the average current consumption of the system if it is used with the normal operation mode for intermittent operation.

HALT – instruction is allowed when CPU is operating with mainclock as well with subclock. So, power consumption can be reduced additionally by stopping the instruction execution.

4.2.2 IDLE 1

The IDLE1 mode is set when the PSM1 and PSM0 bits of the PSMR register are cleared to “00” and the STP bit of the PSC register is set to 1 in the normal operation mode.

In the IDLE1 mode clock supply to the CPU and the other on-chip peripheral functions is stopped. The main oscillator, PLL/SSCG, and flash memory continue operating.

Program execution is stopped, and the contents of the internal RAM before the IDLE1 mode was set are retained. The CPU and other on-chip peripheral functions stop operating. However, the on-chip peripheral functions that can operate on the subclock or external clock continue operating.

The IDLE1 mode is released by a non-maskable interrupt request signal or an unmasked maskable interrupt request signal or reset signal..

The IDLE1 mode can reduce current consumption more than the HALT mode because the operations of the on-chip peripheral functions are stopped.

4.2.3 IDLE 2

The IDLE2 mode is set when the PSM1 and PSM0 bits of the PSMR register are set to “10” and the STP bit of the PSC register is set to 1 in the normal operation mode.

In the IDLE2 mode, the main clock oscillator continues operating, but clock supply to the CPU, PLL/SSCG, flash memory, and the other on-chip peripheral functions is stopped.

Program execution is stopped and the contents of the internal RAM before the IDLE2 mode was set are retained. Not only the CPU but also the other on-chip peripheral functions stop operating. However, the on-chip peripheral functions that can operate on the subclock or external clock continue operating.

The IDLE2 mode is released by a non-maskable interrupt request signal (NMI pin input or INTWDT2 signal), unmasked external interrupt request signal or reset signal.

The IDLE2 mode can reduce current consumption more than the IDLE1 mode because the operations of the on-chip peripheral functions and flash memory are stopped. Because the PLL/SSCG and flash memory are stopped, however, setup times for the PLL/SSCG and flash memory must be maintained after the IDLE2 mode is released.

4.2.4 STOP

The STOP mode is set when the PSM1 and PSM0 bits of the PSMR register are set to “01” or “11”, and the STP bit of the PSC register is set to 1 in the normal operation mode.

In the STOP mode, the main clock oscillator stops operating but the subclock oscillator continues operating. The mainclock supply to the CPU and the other on-chip peripherals is stopped.

Program execution is stopped, and the contents of the internal RAM before the STOP mode was set are retained. Not only the CPU but also the other on-chip peripheral functions stop operating. However, the on-chip peripherals which can operate on the subclock or external clock continue operating.

The STOP mode is released by a non-maskable interrupt request signal (NMI pin input or INTWDT2 signal), unmasked external interrupt request signal, unmasked internal interrupt request signal of a peripheral function that can operate in the STOP mode, or reset signal.

The STOP mode can reduce current consumption more than the IDLE2 mode because the operation of the main clock oscillator is stopped. When the subclock oscillator, internal oscillator and external clock are not used, the current consumption can be substantially reduced down to leakage current flow.

4.2.5 Subclock operation

The subclock operation mode is set when the CK3 bit of the PCC register is set to 1 in the normal operation mode.

When the subclock operation mode is set, the CPU system clock is changed from the main system clock to the subclock. Subclock can be either internal low-speed oscillator f_{RL} or sub oscillator clock f_{XT}. The selection is made by the SUBCLK bit of the option byte.

Check that the CPU system clock has been changed by using the CLS bit of the PCC register. When the MCK bit of the PCC register is set to 1, the operation of the main clock oscillator is stopped. Consequently, the entire system operates on the subclock.

The subclock operation mode is released by clearing the CK3 bit to 0 or by a reset signal.

In the subclock operation mode, the subclock is used as the CPU system clock, so that the current consumption can be reduced from that in the normal operation mode. Of course, the program execution performance is much slower. In addition, a current consumption close to that in the STOP mode can be achieved by stopping the operation of the main clock oscillator.

HALT – instruction is allowed in each mode, also in subclock operation mode. So, power consumption can be reduced additionally by stopping the instruction execution.

4.2.6 Sub – IDLE

The sub-IDLE mode is set when the PSM1 and PSM0 bits of the PSMR register are set to “10” and the STP bit of the PSC register is set to 1 while the processor is in the subclock operation mode.

In the sub-IDLE mode, the clock supply to the CPU, flash memory, and the other on-chip peripheral is stopped, but clock oscillator continues operating.

Program execution is stopped, and the contents of the internal RAM before the sub-IDLE mode was set is retained. Not only the CPU but also the other on-chip peripheral functions stop operating. However, the on-chip peripheral functions that can operate on the subclock continue operating.

The sub-IDLE mode is released by a non-maskable interrupt request signal (NMI pin input or INTWDT2 signal), unmasked external interrupt request signal, unmasked internal interrupt request of a peripheral function that can operate in the sub-IDLE mode, or reset signal.

The sub-IDLE mode can reduce current consumption more than the subclock operation mode because the operations of the CPU, flash memory, and other on-chip peripheral functions are stopped.

If the sub-IDLE mode is set after the main clock is stopped, a current consumption close to that in the STOP mode can be achieved.

4.3 How to enter the standby modes IDLE1, IDLE2 or STOP

The selection of one of the standby modes IDLE1, IDLE2 or STOP is done in three steps.

At first, the selection of the mode is done within the power save mode control register (PSMR):

Table 4-1 Pre – selection of standby mode in register PSMR

Bit position	Bit name	Function
1 to 0	PSM[1:0]	Specification of operation in software stand-by mode:

PSM1	PSM0	Power save mode
0	0	IDLE1 mode
0	1	STOP mode
1	0	IDLE2 mode or sub-IDLE mode[a]
1	1	STOP mode

Note: The PSM0 and PSM1 bits take effect after PSC.STP = 1.

a) Sub-IDLE mode is entered if the processor is in subclock mode (clocked by f_{XT} or f_{RL}).

The second step is to write into register PRCMD:

Due to the power save control register (PSC) is a write protected register, directly before writing into PSC, a write instruction into the write protection control register PRCMD has to be accessed. The value, written into PRCMD is meaningless.

The third step is to set bit STP = 1 in the power save control register (PSC) with the next instruction, following PRCMD writing, that the previously selected standby mode takes effect.

Remark: Insert five or more NOP instructions after the store instruction that manipulates the PSC register.

4.3.1 Example how to enter IDLE 2 – mode

The following example shows how to initialize and enter a IDLE1, IDLE2 or STOP power save mode.

First the desired power save mode is specified
(for IDLE2 mode in this example, means PSMR.PSM[1:0] = 10_B).

The PSC register is a write-protected register, and the PRCMD register is the corresponding write-enable register. PRCMD has to be written immediately before writing to PSC.

In this example, maskable interrupts are permitted to leave the power save mode.

```
// xxIC.xxMK = 0          // mask all none wake-up interrupts
// xxIC.xxMK = 1          // unmask all wake-up interrupts


di                        // disable vectored interrupts
mov     0x02,r10
st.b    r10,PSMR[r0]      // PSMR.PSM[1:0] = 10B: IDLE2 mode
mov     0x62,r10
st.b    r10,PRCMD[r0]     // enable writing into PSC
st.b    r10,PSC[r0]       // wake up by maskable interrupts ( INTM = 0 )
                          // and enter power save mode      ( STP = 1 )
nop
nop                       // append 5 or more NOP´s for pipeline clearance
nop                       //    after wake up
nop
nop
                          // after wake-up
// xxIC.xxIF = 0          // discard all unwanted pending interrupts


ei                        // enable vectored interrupts
```

CHAPTER 5 CURRENT CONSUMPTION IN DIFFERENT MODES

As described in the chapter before, the microcontroller has different power consumption in its different operating / standby modes. Table 5-1 shall give a rough feeling of the amount of power consumption resp. possibility of power saving within the different modes.

This is an example from the data sheet of V850ES/FG3 (µPD70F337x). The power consumption varies over the product families. When you design your application, please refer to the actual data sheet for your target device.

Table 5-1 Example for power consumption of V850ES/FG3 in several modes

VDD = EVDD = BVDD = 3.3 to 5.5V, AVREF0 = 3.3 to 5.5V, VSS = EVSS = BVSS = AVSS = 0V

Mode	Symbol	Condition			TYP.	Unit
Operating mode	IDD1	All peripherals running	PLL: OFF 4MHz ≤ f_{XX} ≤ 16MHz	fxx=16MHz fx=16MHz	22	mA
		All peripherals stopped			19	
HALT mode	IDD2	All peripherals running			13	
		All peripherals stopped			9	
IDLE 1 mode	IDD3	Peripheral (TAA, UARTD) running			3.3	
		All peripherals stopped			1.6	
IDLE 2 mode	IDD4	IDLE 2 = all peripherals stopped			0.8	
Subclock operating mode	IDD5	POC is working. 240kHz internal-OSC is working. 8MHz internal-high speed OSC is stopped.	Crystal resonator (f_{XT} = 32.768 kHz)		80	µA
			Internal low-speed oscillator (f_{RL} = 240 kHz typ.)		220	
Sub - IDLE mode	IDD6	POC is working. 240kHz internal-OSC is working. 8MHz internal-OSC is stopped.	Crystal resonator (f_{XT} = 32.768 kHz)		20	
			Internal low-speed oscillator (f_{RL}=240kHz typ.) (Sub-OSC stopped)		25	
STOP mode	IDD7	POC stop	Internal low-speed oscillator (f_{RL} = 240 kHz typ.) stop		7.5	
			Internal low-speed oscillator (f_{RL} = 240 kHz typ.) operating		15.5	

Caution: The values above are arbitrary sample values under special conditions, only for comparison.

fx: MainOSC clock, fed into X1 / X2 – pins

fXX: Main system clock, fed into CPU and peripherals. Source is fx – clock, multipled by a PLL

fXT: SubOSC clock fed into XT1 / XT2 – pins

fRL: 240 kHz internal low-speed oscillator clock

POC: Power On Circuit, a peripheral which resets the controller when Vdd is below a certain voltage level