RL78/F23, F24

Standby Function

R01AN6627EJ0100
Rev.1.00
2022.09.30

Introduction

This application note describes the standby function (HALT mode, STOP mode and SNOOZE mode) of the RL78/F23 and the RL78/F24 microcontrollers by providing examples for setting each of the modes. For the clocks and each peripheral function described in this document, refer to the applicable User's Manual: Hardware.

Contents

1. HALT Mode 2
1.1 Procedure for HALT Mode Setting 4
1.2 Notes on HALT Mode 5
2. STOP Mode 6
2.1 Procedure for STOP Mode Setting 8
2.2 Notes on STOP Mode 9
3. SNOOZE Mode 10
3.1 Procedure for SNOOZE Mode Setting 12
3.1.1 Example of LIN/UART Module (RLIN3) Setting 12
3.1.2 Example of DTC Setting 15
3.1.3 Example of SNOOZE Status Output 17
3.2 Notes on SNOOZE Mode 20
4. References 21
Revision History 22

1. HALT Mode

In HALT mode, power consumption is reduced by stopping the supply of the operation clock to the CPU. The CPU transitions to the HALT mode when the HALT instruction is executed. Even after the HALT instruction is executed, the state of each clock remains unchanged from the previous state. Table 1-1 shows the clock states in HALT mode.

The HALT mode is released when a source for the enabled interrupt (the value of the interrupt mask flag is 0) is generated. Figure 1-1 illustrates transition/release timing of HALT mode.

The processing after release of HALT mode varies depending on whether the interrupt request acknowledgement is enabled (EI) or disabled (DI). When the MCU transitions to HALT mode while the interrupt request acknowledgement is enabled (EI), the vectored-interrupt service will start after release of HALT mode. Meanwhile, when the MCU transitions to HALT mode while the interrupt request acknowledgement is disabled (DI), the instruction at the next address after the HALT instruction is executed after release of HALT mode. Table 1-2 lists the peripheral function interrupts that can be used to release HALT mode.

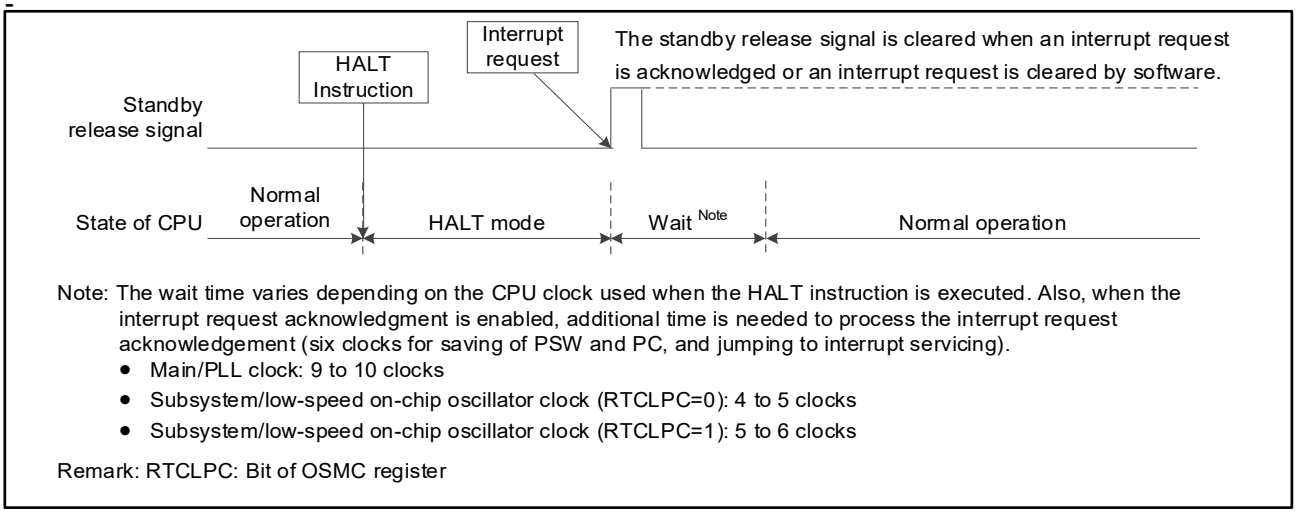
Also, when the Flash memory CRC operation function (high-speed CRC) is completed, HALT mode will be released. Regarding the Flash memory CRC operation function, refer to the application note “Safety Function (R01AN6624)”.

Table 1-1. Clock States in HALT Mode

Clock	Before transition to HALT mode	In HALT mode	After HALT mode released
Clock supply to CPU	Not stopped	Stops	Not Stopped
High-speed system clock	Oscillating or stops	Remains unchanged from the previous state	Remains unchanged from the previous state
High-speed on-chip oscillator clock	Oscillating or stops	Remains unchanged from the previous state	Remains unchanged from the previous state
PLL clock	Oscillating or stops	Remains unchanged from the previous state	Remains unchanged from the previous state
Low-speed on-chip oscillator clock	Oscillating or stops	Remains unchanged from the previous state	Remains unchanged from the previous state
Subsystem clock	Oscillating or stops	Remains unchanged from the previous state	Remains unchanged from the previous state
Watchdog timer-dedicated low-speed on-chip oscillator	Oscillating or stops	Remains unchanged from the previous state or stops Note	State before transition to HALT mode

Note: This varies depending on the settings of the WDTON bit and the WDSTBYON bit in the user option bytes (000C0H/040C0H).
When WDTON=1 and WDSTBYON=1: Continues oscillating.
When WDTON=1 and WDSTBYON=0: Stops oscillating. The counter is cleared after HALT mode is released and the clock oscillation starts.

-



Note: The wait time varies depending on the CPU clock used when the HALT instruction is executed. Also, when the interrupt request acknowledgment is enabled, additional time is needed to process the interrupt request acknowledgement (six clocks for saving of PSW and PC, and jumping to interrupt servicing).
- Main/PLL clock: 9 to 10 clocks
- Subsystem/low-speed on-chip oscillator clock (RTCLPC=0): 4 to 5 clocks
- Subsystem/low-speed on-chip oscillator clock (RTCLPC=1): 5 to 6 clocks

Remark: RTCLPC: Bit of OSMC register

Figure 1-1. Transition/Release Timing of HALT Mode

Table 1-2. Peripheral Function used to Release HALT Mode

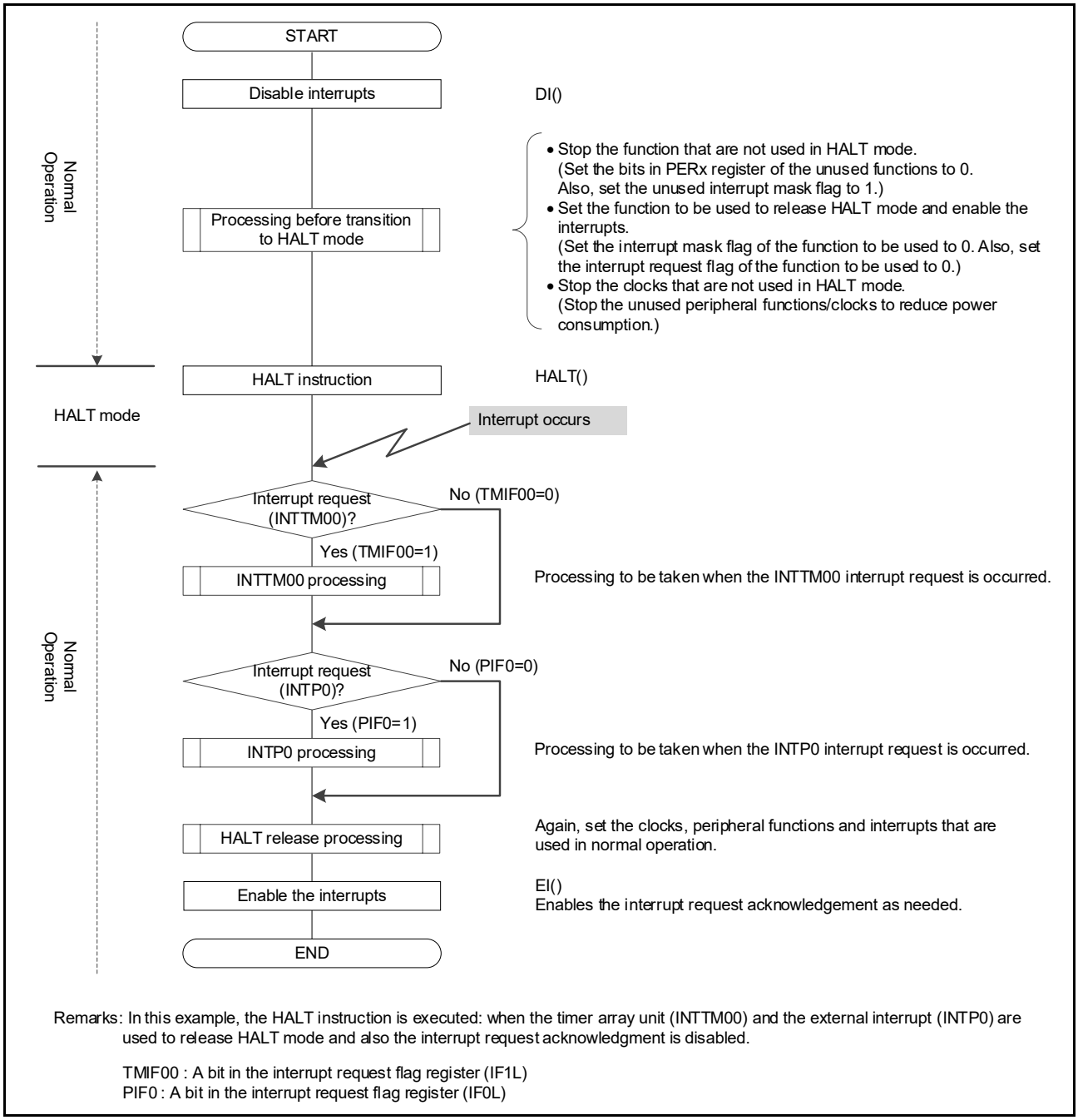
CPU's operation clock when the HALT instruction is executed	Peripheral function interrupts used to release HALT mode Note 1	
Main system clock (High-speed system clock or high-speed on-chip oscillator clock)	• Timer array unit • Real-time clock • Clock monitor • Timer RJ • Timer RDe • A/D converter • Comparator • Serial array unit • Serial interface (IICA)	• LIN/UART module (RLIN3) • CANFD interface • DTC • Voltage detection function • External interrupt • Key interrupt • Watchdog timer Note 2
Subsystem/Low-speed on-chip oscillator clock	• Timer array unit • Real-time clock • Timer RJ • Timer RDe • Serial array unit	• DTC • Voltage detection function • External interrupt • Key interrupt • Watchdog timer Note 2

Notes: 1. The peripheral function interrupts vary depending on the product used. For details, refer to the User's Manual: Hardware of the product used.

2. This varies depending on the settings of the WDTON bit and the WDSTBYON bit in the user option byte (000C0H/040C0H).
   When WDTON=1 and WDSTBYON=1: Continues oscillating.
   When WDTON=1 and WDSTBYON=0: Stops oscillating. The counter is cleared after HALT mode is released and the clock oscillation starts.

1.1 Procedure for HALT Mode Setting

Figure 1-2 shows an example for HALT mode setting.



Remarks: In this example, the HALT instruction is executed: when the timer array unit (INTTM00) and the external interrupt (INTP0) are used to release HALT mode and also the interrupt request acknowledgment is disabled.

TMIF00 : A bit in the interrupt request flag register (IF1L)
PIF0 : A bit in the interrupt request flag register (IF0L)

Figure 1-2. Example for HALT Mode Setting

1.2 Notes on HALT Mode

- When the value of the interrupt mask flag is 0 (interrupt servicing is enabled) and also when the value of the interrupt request flag is 1 (an interrupt request signal is generated), HALT mode will be released immediately even when the HALT instruction is executed.
- The processing after the release of HALT mode varies depending on whether the interrupt request acknowledgement is enabled (EI) or disabled (DI). When the MCU transitions to HALT mode while the interrupt request acknowledgement is enabled (EI), the vectored-interrupt servicing will start when an interrupt request (interrupt servicing is enabled) is generated while the value of the interrupt mask flag is 0. When the MCU transitions to HALT mode when the interrupt request acknowledgement is disabled (DI), the instruction at the next address after the HALT instruction will be executed when an interrupt request is generated.
- HALT mode is released when a reset signal is generated.
- Whether to make the watchdog timer-dedicated low-speed on-chip oscillator continue oscillating or stop oscillating in HALT mode is selected by setting the user option byte (000C0H/040C0H).

2. STOP Mode

In STOP mode, power consumption is reduced by stopping the main system clock (high-speed system clock, highspeed on-chip oscillator clock). The CPU transitions to STOP mode when the STOP instruction is executed. Table 2-1 shows the clock states in STOP mode.

STOP mode is released when a source for the enabled interrupt (the mask flag of the interrupt enabled is set to 0) is generated. Figure 2-1 illustrates transition/release timing of STOP mode.

The processing after the release of STOP mode varies depending on whether the interrupt request acknowledgement is enabled (EI) or disabled (DI). When the MCU transitions to STOP mode while the interrupt request acknowledgement is enabled (EI), the vectored-interrupt service will start after release of STOP mode. When the MCU transitions to STOP mode while the interrupt request acknowledgement is disabled (DI), the instruction at the next address after the STOP instruction is executed after release of STOP mode. Table 2-2 shows the peripheral function interrupts that can be used to release STOP mode.

Table 2-1. Clock States in STOP Mode

Clock	Before transition to STOP mode Note 1	In STOP mode	After STOP mode released
Clock supply to CPU	Not stopped	Stops	Not Stopped
High-speed system clock	Oscillating or stops	Stops	State before transition to STOP mode
High-speed on-chip oscillator clock	Oscillating or stops	Stops	State before transition to STOP mode
PLL clock Note 2	Stops	Stops	Stops
Low-speed on-chip oscillator clock	Oscillating or stops	Remains unchanged from the previous state	Remains unchanged from the previous state
Subsystem clock	Oscillating or stops	Remains unchanged from the previous state	Remains unchanged from the previous state
Watchdog timer-dedicated low-speed on-chip oscillator	Oscillating or stops	Remains unchanged from the previous state or stops Note 3	State before transition to STOP mode

Notes: 1. Be sure to execute the transition to STOP mode when the CPU clock is the main system clock (high-speed system clock or high-speed on-chip oscillator clock).
2. To make the CPU transition to STOP mode, set the PLLON bit to 0 (PLL stopped) beforehand.
3. This varies depending on the settings of the WDTON bit and the WDSTBYON bit in the user option byte (000C0H/040C0H).
When WDTON=1 and WDSTBYON=1: Continues oscillating.
When WDTON=1 and WDSTBYON=0: Stops oscillating. The counter is cleared after STOP mode is released and the clock oscillation starts.

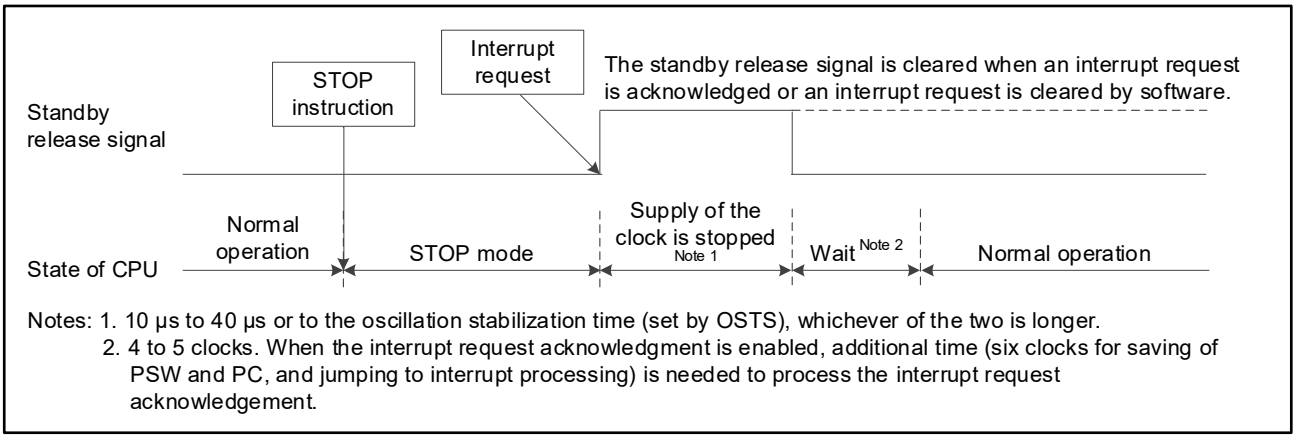


Figure 2-1. Transition/Release Timing of STOP Mode

Table 2-2. Peripheral Function Interrupts used to Release STOP Mode

CPU's operation clock when STOP instruction is executed	Peripheral function interrupts used to release STOP mode Note 1
Main system clock (High-speed system clock or high-speed on-chip oscillator clock)	• Real-time clock Note 2 • Timer RJ Note 2 • Timer RDe Note 2 • Comparator • Serial interface (IICA) Note 3 • Voltage detection function • External interrupt • Key interrupt • Watchdog timer Note 4

Notes: 1. The peripheral function interrupts vary depending on the product used. For details, refer to the User's Manual: Hardware of the product used.

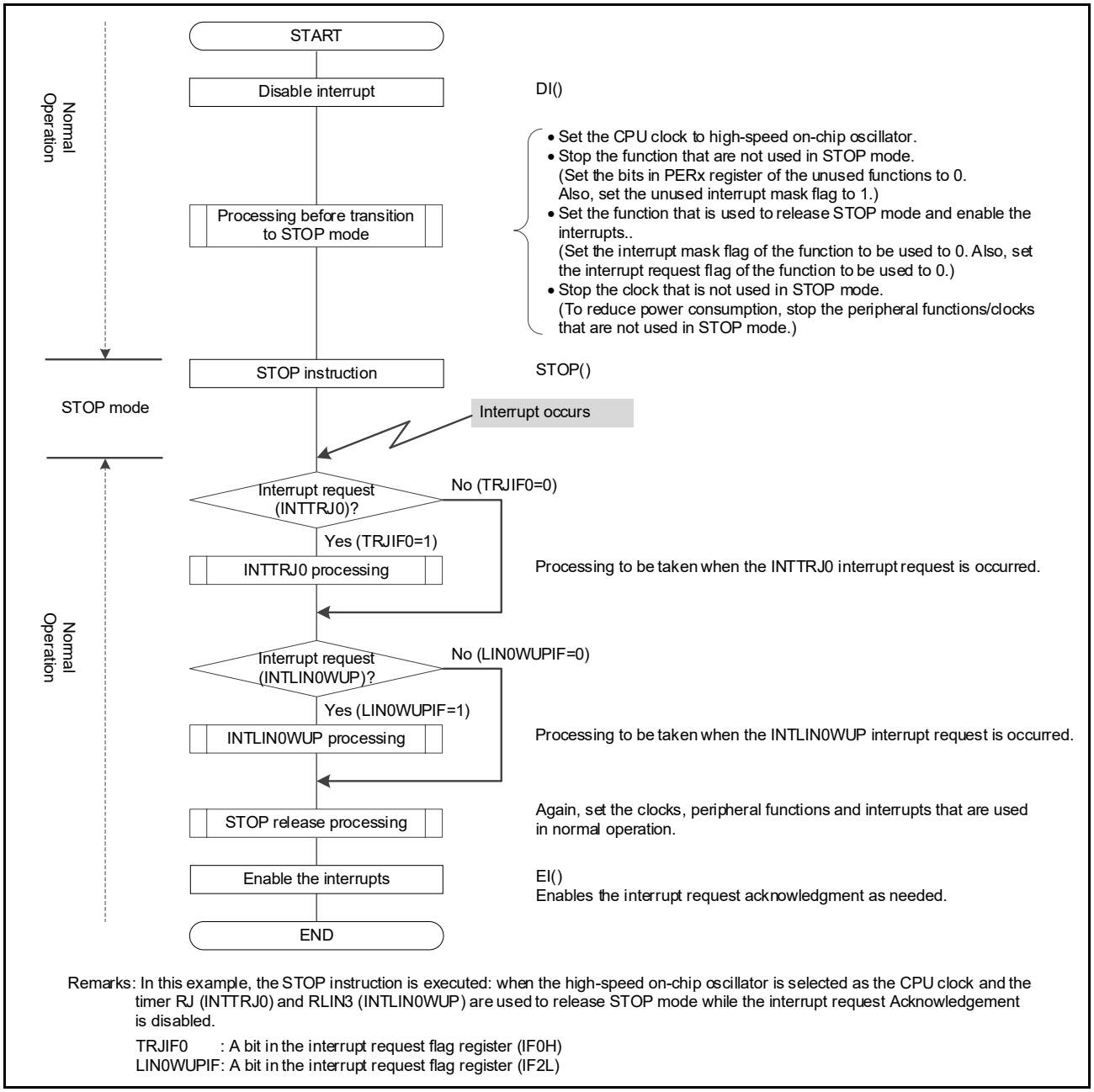
2. This can be used when the subsystem clock that continues oscillating in STOP mode is selected as the operation clock.

3. This can be used when an extension code from the master device or a local address has been received in STOP mode.

4. This varies depending on the settings of the WDTON bit and the WDSTBYON bit in the user option byte (000C0H/040C0H).
   When WDTON=1 and WDSTBYON=1: Continues oscillating.
   When WDTON=1 and WDSTBYON=0: Stops oscillating. The counter is cleared after STOP mode is released and the clock oscillation starts.

2.1 Procedure for STOP Mode Setting

Figure 2-2 shows an example for STOP mode setting.



Remarks: In this example, the STOP instruction is executed: when the high-speed on-chip oscillator is selected as the CPU clock and the timer RJ (INTTRJ0) and RLIN3 (INTLIN0WUP) are used to release STOP mode while the interrupt request Acknowledgement is disabled.

TRJIF0 : A bit in the interrupt request flag register (IF0H)
LIN0WUPIF: A bit in the interrupt request flag register (IF2L)

Figure 2-2. Example for STOP Mode Setting

2.2 Notes on STOP Mode

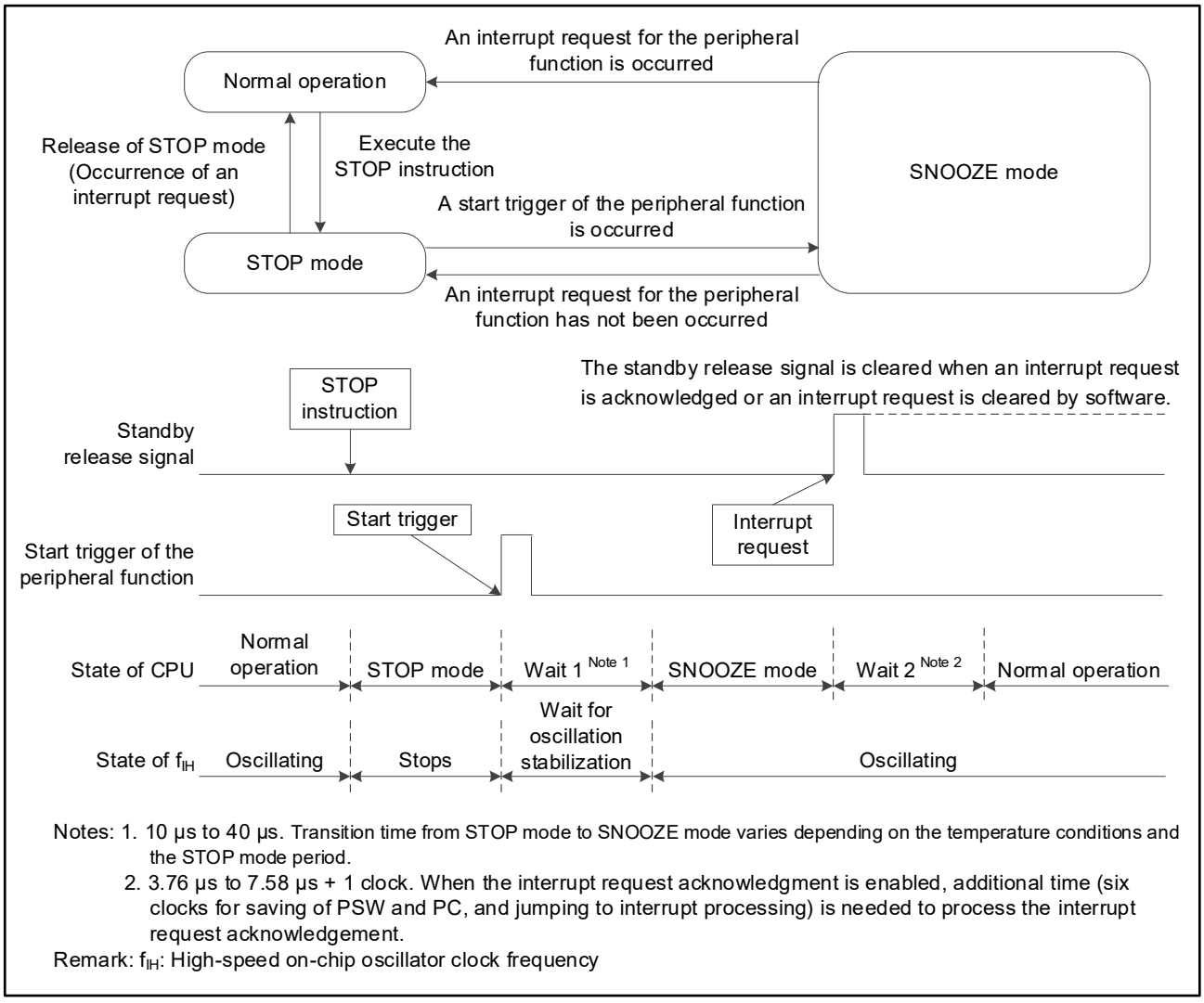
- The CPU can transition to STOP mode only when the CPU clock is the main system clock. Therefore, do not make the CPU transition to STOP mode when the CPU operates on the PLL clock, or the subsystem/low-speed on-chip oscillator clock.
- To make the CPU transition to STOP mode when the CPU clock is the high-speed system clock (X1 oscillator), the settings of the OSTS register need to be completed before executing the STOP instruction.
- Before executing the STOP instruction, be sure to stop the operation of the peripheral hardware (excluding the function(s) operating in SNOOZE mode) running on the main system clock (high-speed system clock or highspeed on-chip oscillator clock).
- When the value of the interrupt mask flag is 0 (interrupt servicing is enabled) and also when the value of the interrupt request flag is 1 (an interrupt request signal is generated), STOP mode will be released immediately even when the STOP instruction is executed.
- The processing after release of STOP mode varies depending on whether the interrupt request acknowledgement is enabled (EI) or disabled (DI). When the MCU transitions to STOP mode while the interrupt request acknowledgement is enabled (EI), the vectored-interrupt servicing will start when an interrupt request is generated while the value of the interrupt mask flag is 0 (interrupt servicing is enabled). When the MCU transitions to STOP mode while the interrupt request acknowledgement is disabled (DI), the instruction at the next address after the STOP instruction is executed when an interrupt request is generated.
- STOP mode is released when a reset signal is generated.
- Whether to make the watchdog timer-dedicated low-speed on-chip oscillator continue oscillating or stop oscillating in STOP mode is selected by setting the user option byte (000C0H/040C0H).

3. SNOOZE Mode

In SNOOZE mode, data reception by the LIN/UART module (RLIN3) and memory transfer by the DTC function are performed while the CPU operation is stopped. When a start trigger for the peripheral function is generated in STOP mode, the high-speed on-chip oscillator starts oscillating and the CPU transitions to SNOOZE mode.

SNOOZE mode is released when an interrupt request for a peripheral function operating in SNOOZE mode is generated. Otherwise, if the interrupt request of the peripheral function is not generated, the MCU returns to STOP mode. Figure 3-1 illustrates transition/release timing of SNOOZE mode. Table 3-1 shows the clock states in SNOOZE mode.

Also, the RL78/F23 and RL78/F24 MCUs are provided with a function to output the SNOOZE status at transition to SNOOZE mode and when the SNOOZE mode is released. The settings of each function (data reception, DTC function and SNOOZE status output) in SNOOZE mode are described through examples in “3.1 Procedure for SNOOZE Mode Setting”.



Notes: 1. 10 µs to 40 µs. Transition time from STOP mode to SNOOZE mode varies depending on the temperature conditions and the STOP mode period.

2. 3.76 µs to 7.58 µs + 1 clock. When the interrupt request acknowledgment is enabled, additional time (six clocks for saving of PSW and PC, and jumping to interrupt processing) is needed to process the interrupt request acknowledgement.

Remark: f_{IH}: High-speed on-chip oscillator clock frequency

Figure 3-1. SNOOZE Mode Transition/Release Timing

Table 3-1. Clock States in SNOOZE Mode

Clock	Normal operation mode (before transition to STOP mode)	In STOP mode Note 1	In SNOOZE mode	Normal operation mode (after release of SNOOZE mode)
Clock supply to CPU	Not stopped	Stops	Stops	Not stopped
High-speed system clock	Oscillating or stops	Stops	Stops	Stops
High-speed on-chip oscillator clock	Oscillating	Stops	Oscillating	Oscillating
PLL clock Note 2	Stops	Stops	Stops	Stops
Low-speed on-chip oscillator clock	Oscillating or stops	Remains unchanged from the previous state	Remains unchanged from the previous state	Remains unchanged from the previous state
Subsystem clock	Oscillating or stops	Remains unchanged from the previous state	Remains unchanged from the previous state	Remains unchanged from the previous state
Watchdog timer-dedicated low-speed on-chip oscillator clock	Oscillating or stops	Remains unchanged from the previous state or stops Note 3	Remains unchanged from the previous state or stops Note 3	State before transition to STOP mode

Notes: 1. Be sure to execute the transition to STOP mode when the CPU clock is the main system clock (high-speed system clock or high-speed on-chip oscillator clock).
2. To make the CPU transition to STOP mode, set the PLLON bit to 0 (PLL stopped) beforehand.
3. This varies depending on the settings of the WDTON bit and the WDSTBYON bit in the user option byte (000C0H/040C0H).
When WDTON=1 and WDSTBYON=1: Continues oscillating.
When WDTON=1 and WDSTBYON=0: Stops oscillating. The counter is cleared after transition to normal operation mode, and the clock oscillation starts.

The following peripheral functions can be used in SNOOZE mode.

- Real-time clock Note 1
- Timer RJ Note 1
- Timer RDe Note 1
- Clock output/buzzer output Note 1
- Comparator
- Serial interface (IICA) Note 2
- LIN/UART module (UART mode of RLIN3) Note 3
- Voltage detection function
- External interrupt
- Key interrupt
- Watchdog timer Note 4
- DTC

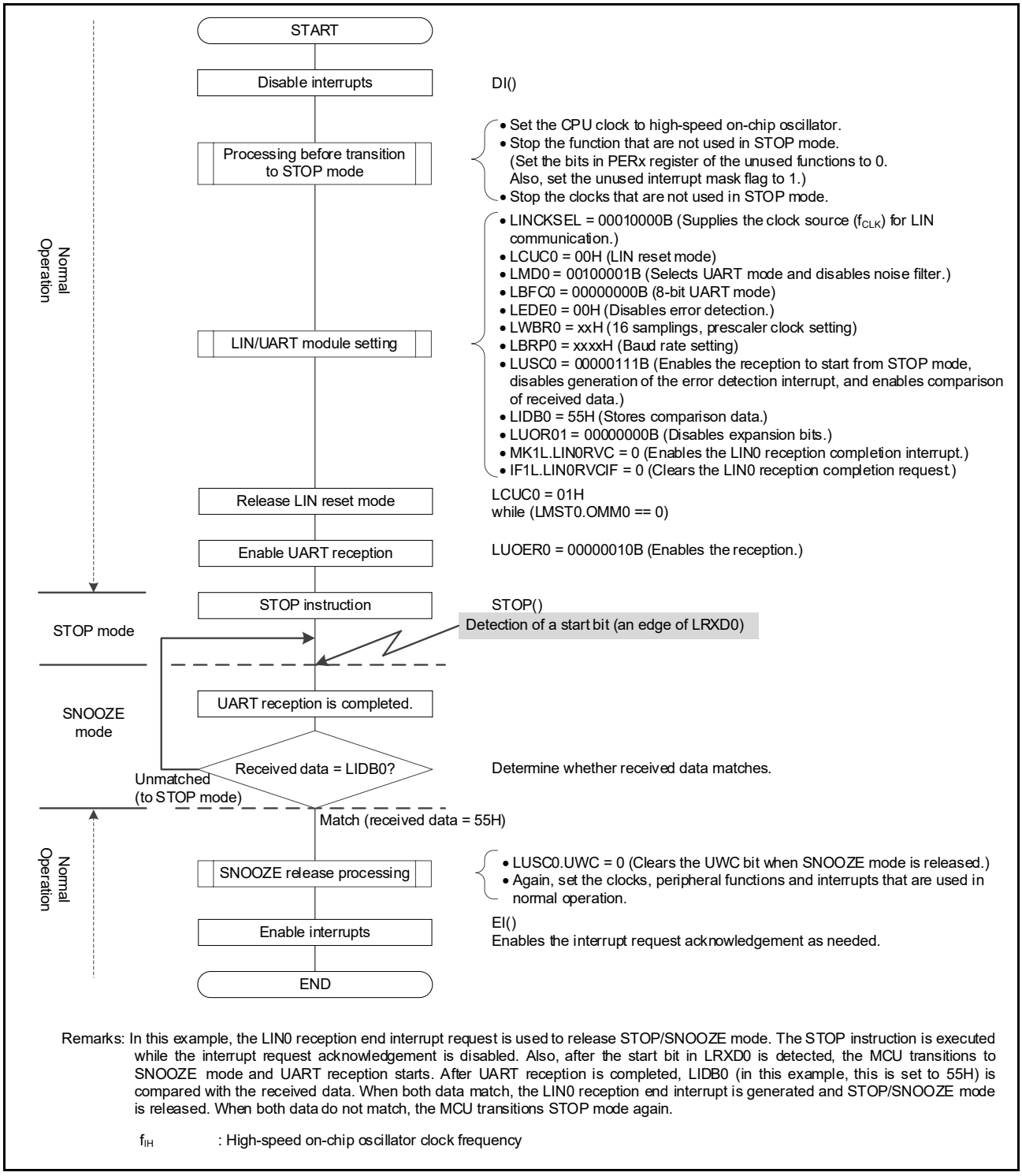
Notes: 1. These functions can be used when the subsystem clock that continues oscillating in STOP mode is selected as the operation clock.
2. This can be used when an extension code from the master device or a local address has been received in STOP mode.
3. Operates on the high-speed on-chip oscillator as the clock source in SNOOZE mode. This can be used as a factor for recovering from SNOOZE mode when the interrupt conditions are satisfied.
4. This varies depending on the settings of the WDTON bit and the WDSTBYON bit in the user option byte (000C0H/040C0H).
When WDTON=1 and WDSTBYON=1: Continues oscillating.
When WDTON=1 and WDSTBYON=0: Stops oscillating. The counter is cleared after the CPU returns to the normal operation mode.

3.1 Procedure for SNOOZE Mode Setting

3.1.1 Example of LIN/UART Module (RLIN3) Setting

Figure 3-2 is an example of setting the LIN/UART module (UART mode) to be used in SNOOZE mode. Figure 3-3 is a timing chart.

In this example below, the LIN/UART module (RLIN3) starts UART reception when the UART function of the LIN/UART module detects an edge (start bit) of LRXD0. When the data reception is completed, the data received by UART is compared with the data that has been set (the data is set in the LIDB0 register before the MCU transitions to STOP/SNOOZE mode). When the data matches, an interrupt is generated and SNOOZE mode is released.



Remarks: In this example, the LIN0 reception end interrupt request is used to release STOP/SNOOZE mode. The STOP instruction is executed while the interrupt request acknowledgement is disabled. Also, after the start bit in LRXD0 is detected, the MCU transitions to SNOOZE mode and UART reception starts. After UART reception is completed, LIDB0 (in this example, this is set to 55H) is compared with the received data. When both data match, the LIN0 reception end interrupt is generated and STOP/SNOOZE mode is released. When both data do not match, the MCU transitions STOP mode again.

f_{IH} : High-speed on-chip oscillator clock frequency

Figure 3-2. Example for SNOOZE Mode Setting (LIN/UART)

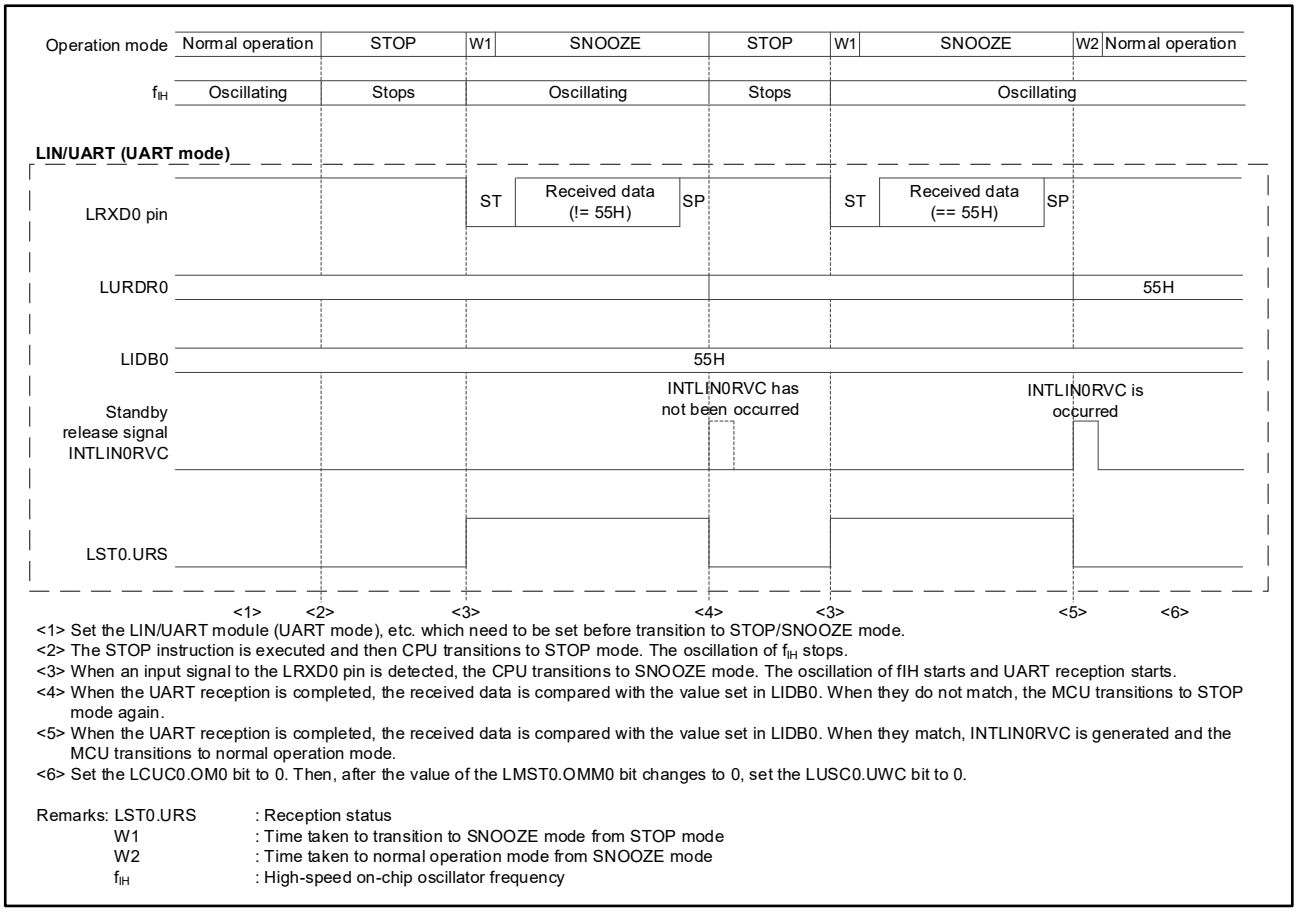


Figure 3-3. Timing Chart of SNOOZE Mode (LIN/UART)

Table 3-2 is an example of settings for communication speed that are available in SNOOZE mode.

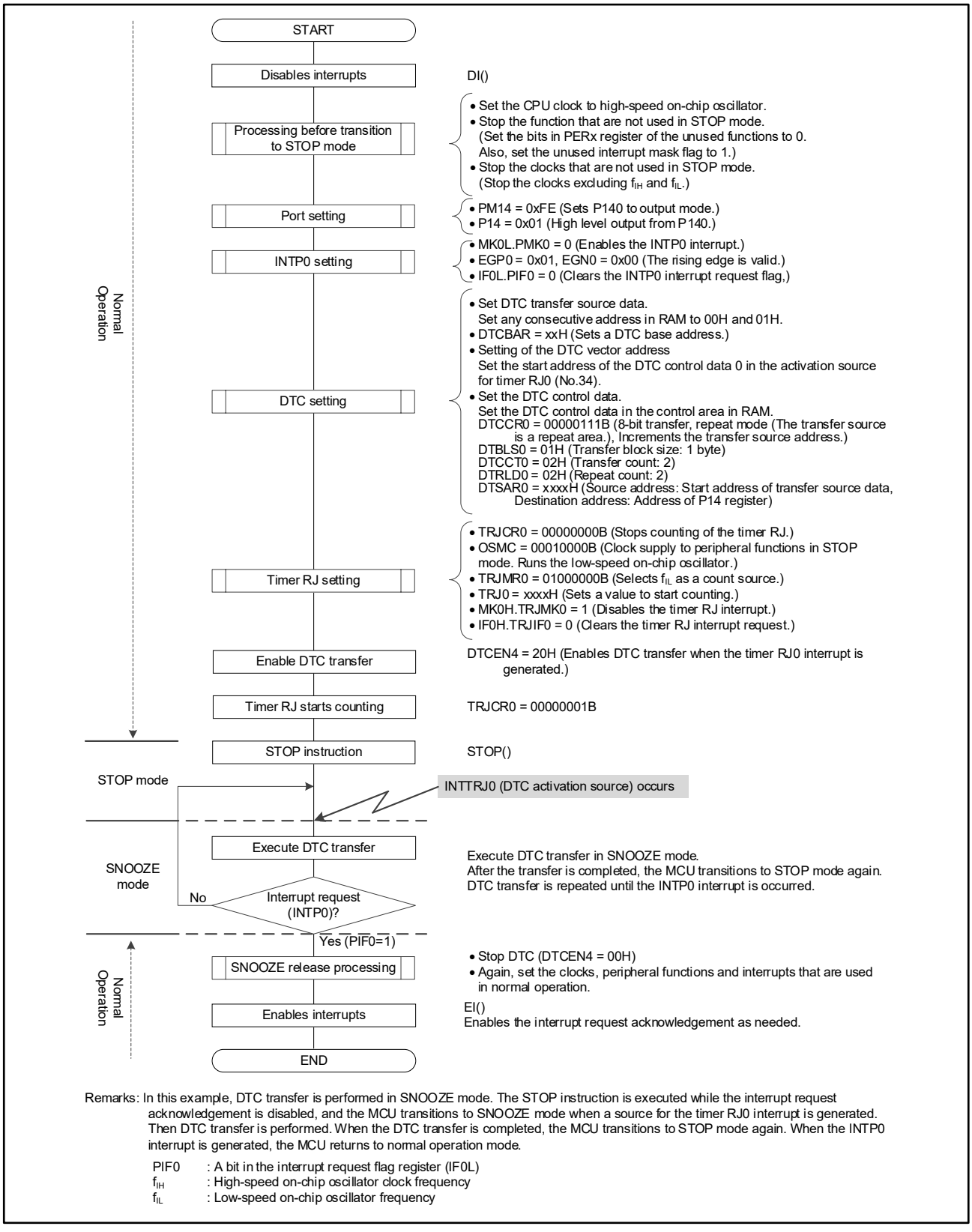
Table 3-2. Setting Example of Communication Speed

Format	Baud rate	LIN clock source	LWBRn		LBRPn	Maximum Allowable Value	Minimum Allowable Value
			LPRS[2:0]	NSPB[3:0]			
1ST-8DATA-1PRY-1SP	1200bps	40 MHz ± 2.0%	001B	0000B	1038	2.66%	-2.53%
	2400bps				517	2.50%	-2.36%
	4800bps				256	1.97%	-2.23%
	9600bps				126	1.30%	-1.58%
	1200bps	32 MHz ± 2.0%	001B	0000B	830	2.64%	-2.55%
	2400bps				413	2.40%	-2.46%
	4800bps				205	2.16%	-2.04%
	9600bps				101	1.68%	-1.19%
	1200bps	40 MHz ± 2.2%	001B	0000B	1038	2.47%	-2.32%
	2400bps				517	2.31%	-2.15%
	4800bps				256	1.78%	-2.03%
	9600bps				126	1.11%	-1.37%
	1200bps	32 MHz ± 2.2%	001B	0000B	830	2.45%	-2.34%
	2400bps				413	2.21%	-2.25%
	4800bps				205	1.97%	-1.83%
	9600bps				101	1.49%	-0.98%
1ST-8DATA-0PRY-1SP	1200bps	40 MHz ± 2.0%	001B	0000B	1037	3.08%	-3.08%
	2400bps				516	2.83%	-2.96%
	4800bps				256	2.52%	-2.54%
	9600bps				126	1.90%	-1.69%
	1200bps	32 MHz ± 2.0%	001B	0000B	830	3.15%	-3.00%
	2400bps				413	2.93%	-2.86%
	4800bps				205	2.71%	-2.34%
	9600bps				100	1.33%	-2.28%
	1200bps	40 MHz ± 2.2%	001B	0000B	1037	2.89%	-2.87%
	2400bps				516	2.64%	-2.76%
	4800bps				256	2.33%	-2.33%
	9600bps				126	1.71%	-1.49%
	1200bps	32 MHz ± 2.2%	001B	0000B	830	2.96%	-2.79%
	2400bps				413	2.74%	-2.66%
	4800bps				205	2.52%	-2.14%
	9600bps				100	1.15%	-2.08%

3.1.2 Example of DTC Setting

Figure 3-4 is an example of setting the DTC transfer to be used in SNOOZE mode. Figure 3-5 is a timing chart.

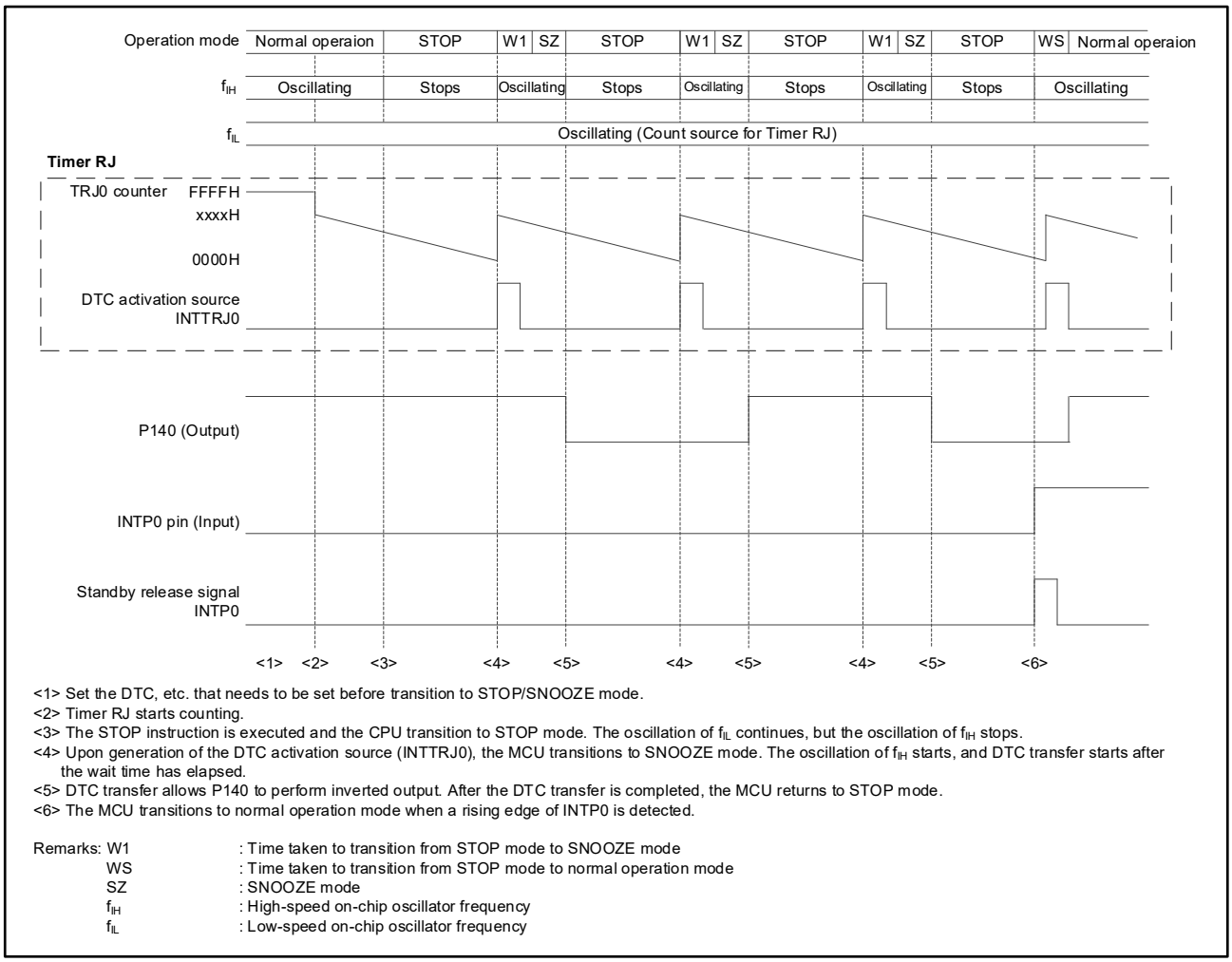
In this example below, DTC transfer is performed by using the timer RJ0 as a DTC activation source, which allows P140 to perform inverted output. The MCU transitions to normal operation mode from SNOOZE mode when the INTP0 interrupt is generated.



Remarks: In this example, DTC transfer is performed in SNOOZE mode. The STOP instruction is executed while the interrupt request acknowledgement is disabled, and the MCU transitions to SNOOZE mode when a source for the timer RJ0 interrupt is generated. Then DTC transfer is performed. When the DTC transfer is completed, the MCU transitions to STOP mode again. When the INTP0 interrupt is generated, the MCU returns to normal operation mode.

PIF0 : A bit in the interrupt request flag register (IF0L)
f_{IH} : High-speed on-chip oscillator clock frequency
f_{IL} : Low-speed on-chip oscillator frequency

Figure 3-4. Example for SNOOZE Mode Setting (DTC)



<1> Set the DTC, etc. that needs to be set before transition to STOP/SNOOZE mode.
<2> Timer RJ starts counting.
<3> The STOP instruction is executed and the CPU transition to STOP mode. The oscillation of f_{IL} continues, but the oscillation of f_{IH} stops.
<4> Upon generation of the DTC activation source (INTTRJ0), the MCU transitions to SNOOZE mode. The oscillation of f_{IH} starts, and DTC transfer starts after the wait time has elapsed.
<5> DTC transfer allows P140 to perform inverted output. After the DTC transfer is completed, the MCU returns to STOP mode.
<6> The MCU transitions to normal operation mode when a rising edge of INTP0 is detected.

Remarks: W1 : Time taken to transition from STOP mode to SNOOZE mode
WS : Time taken to transition from STOP mode to normal operation mode
SZ : SNOOZE mode
f_{IH} : High-speed on-chip oscillator frequency
f_{IL} : Low-speed on-chip oscillator frequency

Figure 3-5. Timing Chart of SNOOZE Mode (DTC)

3.1.3 Example of SNOOZE Status Output

The SNOOZE status output function outputs the state of SNOOZE mode to SNZOUTi (i=0 to 7) pin. Figure 3-6 shows a block diagram of SNOOZE status output, Figure 3-7 shows an example of setting the SNOOZE status output function in SNOOZE mode, and Figure 3-8 shows a timing chart.

The example below shows the SNOOZE mode status output combined with A/D converter.

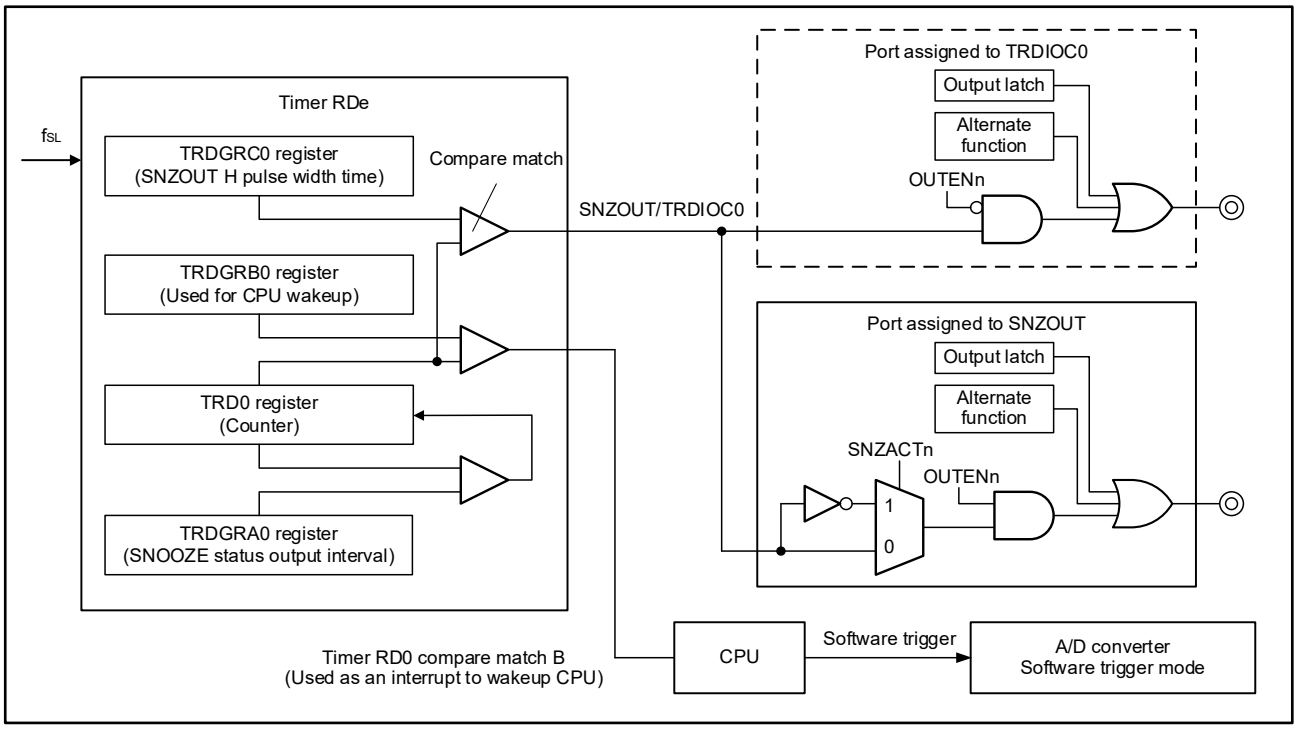
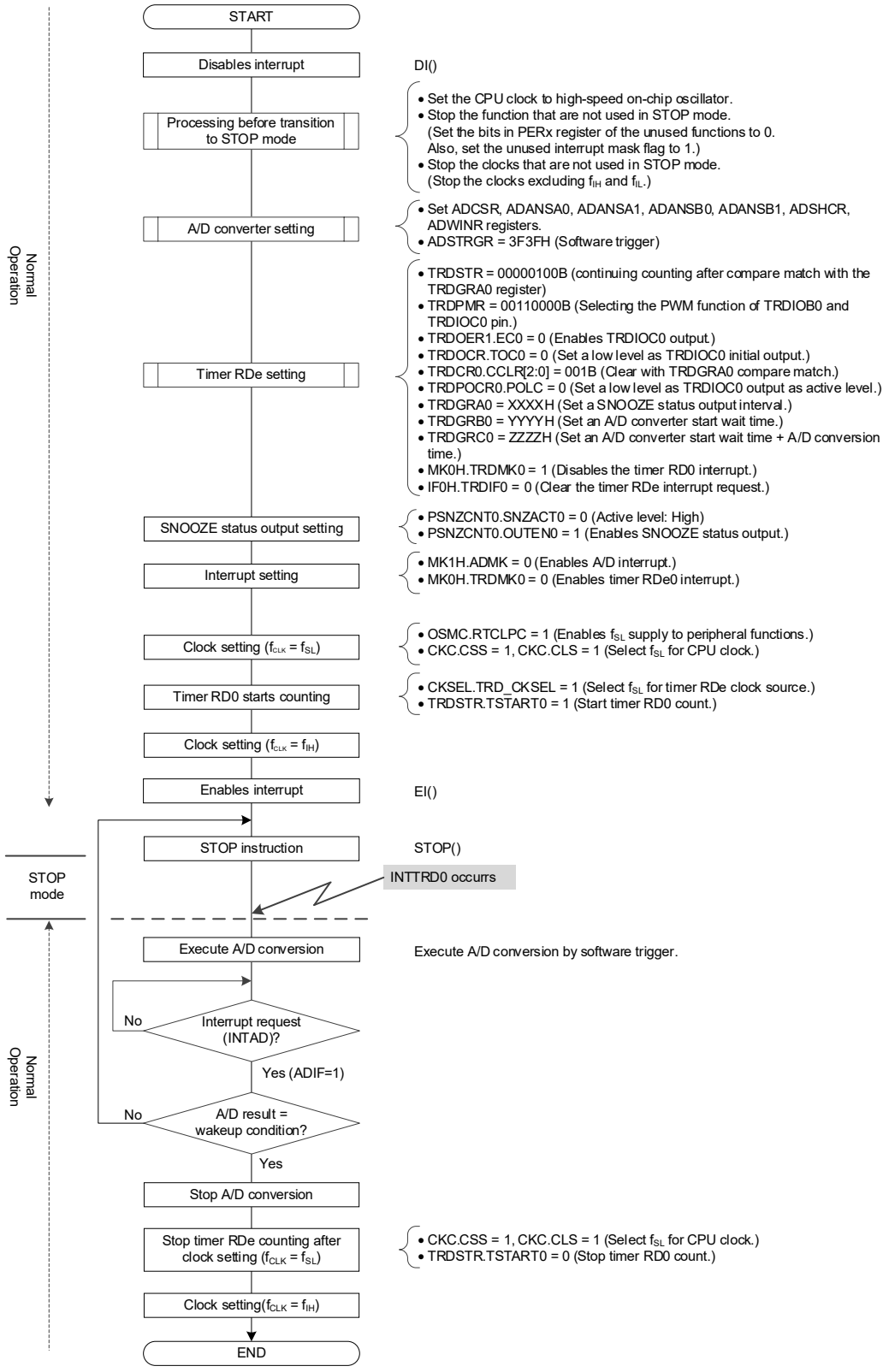


Figure 3-6. Block Diagram of SNOOZE Status Output



Remarks: This example is a setting for using the timer RDe interrupt request to release STOP mode and executing A/D conversion.
The A/D conversion result is determined by software, and the normal operation mode is entered.
Port-related registers are not set in this figure. Please set according to the conditions of use.

ADIF : A bit in the interrupt request flag register (IF1H)
f_{IH} : High-speed on-chip oscillator clock frequency
f_{SL} : Subsystem/low-speed on-chip oscillator select clock frequency

Figure 3-7. Example for SNOOZE Mode Setting (SNOOZE Status Output)

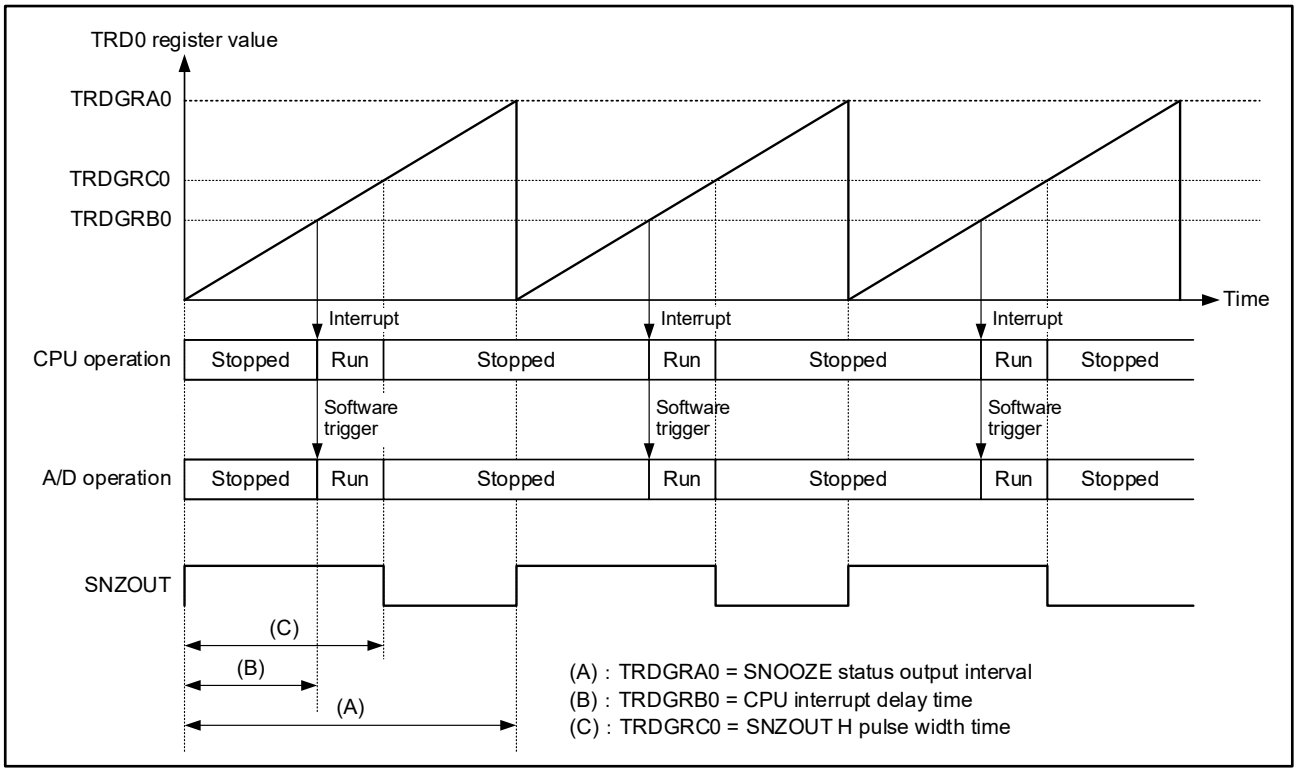


Figure 3-8. Timing Example of SNOOZE Status Output and A/D Conversion

3.2 Notes on SNOOZE Mode

- To use UART communication in SNOOZE mode, set the UWC bit in LUSCn register to 1 before executing the STOP instruction. To make the MCU transition to normal operation mode, set the value of the UWC bit to 0. In the following conditions, when the UWC bit is set to 1, data reception may not be performed properly (a framing error or parity error could occur):
  - After setting UWC to 1, data reception has started before the MCU transitions to STOP mode.
  - Data reception has started while another SNOOZE mode function is being executed.
  - After the MCU returns to normal operation mode from STOP mode, data reception has started before setting the UWC bit to 0.
- SNOOZE/STOP modes are released when a reset signal is generated.
- Whether to make the watchdog timer-dedicated low-speed on-chip oscillator continue oscillating or stop oscillating in SNOOZE mode is selected by setting the user option byte (000C0H/040C0H).
- The MCU transitions to SNOOZE mode from STOP mode. For the notes on STOP mode, refer to “2.2 Notes on STOP Mode”.

4. References

Documents referenced in this application note are shown below. When referring to these documents, make sure to obtain the latest version of each document from Renesas Electronics website.

- RL78/ F23, F24 User's Manual: Hardware Rev. 1.00
- RL78 Family User's Manual: Software Rev. 2.30

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	2022.09.30	-	First edition issued.

General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Precaution against Electrostatic Discharge (ESD)

   A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.
2. Processing at power-on

   The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.
3. Input of signal during power-off state

   Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.
4. Handling of unused pins

   Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.
5. Clock signals

   After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.
6. Voltage application waveform at input pin

   Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (Max.) and V_{IH} (Min.).
7. Prohibition of access to reserved addresses

   Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.
8. Differences between products

   Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
5. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
6. Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The intended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below.
   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.

   Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user’s manual or other Renesas Electronics document.
7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION (“Vulnerability Issues”). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
8. When using Renesas Electronics products, refer to the latest product information (data sheets, user’s manuals, application notes, “General Notes for Handling and Using Semiconductor Devices” in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
12. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
14. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.

(Note2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

(Rev.5.0-1 October 2020)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu, Koto-ku, Tokyo 135-0061, Japan

www.renesas.com

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

Contact information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit: www.renesas.com/contact/.