32-bit RISC Microcontroller

TX Family
TMPM471F10FG

Reference Manual
Clock Control and Operation Mode
(CG-TMPM471F10FG)

Revision 1.1

2025-12

Toshiba Electronic Devices & Storage Corporation

Contents

List of Figures

List of Tables

Preface

Related Documents

Document name
Arm® documentation set for the Arm Cortex®-M4
Exception
Oscillation frequency detector
Voltage detection circuit
Clock selective watchdog timer
Flash memory
Datasheet

Conventions

- Numeric formats follow the rules as shown below:
  Hexadecimal: 0xABC
  Decimal: 123 or 0d123 - Only when it needs to be explicitly shown that they are decimal numbers.
  Binary: 0b111 - It is possible to omit the "0b" when the number of bits can be distinctly understood from a sentence.
- "_N" is added to the end of signal names to indicate low active signals.
- It is called "assert" that a signal moves to its active level, "deassert" to its inactive level.
- When two or more signal names are referred, they are described like as [m:n].
  Example: S[3:0] shows four signal names S3, S2, S1 and S0 together.
- The characters surrounded by ***[ ]*** defines the register.
  Example: ***[ABCD]***
- "N" substitutes suffix number of two or more same kind of registers, fields, and bit names.
  Example: ***[XYZ1]***, ***[XYZ2]***, ***[XYZ3]*** → ***[XYZn]***
- "x" substitutes suffix number or character of units and channels in the register list.
- In case of unit, "x" means A, B, and C, ...
  Example: ***[ADACR0]***, ***[ADBCR0]***, ***[ADCCR0]*** → ***[ADxCR0]***
- In case of channel, "x" means 0, 1, and 2, ...
  Example: ***[T32A0RUNA]***, ***[T32A1RUNA]***, ***[T32A2RUNA]*** → ***[T32AxRUNA]***
- The bit range of a register is written like as [m: n].
  Example: Bit[3: 0] expresses the range of bit 3 to 0.
- The configuration value of a register is expressed by either the hexadecimal number or the binary number.
  Example: ***[ABCD]***<EFG> = 0x01 (hexadecimal), ***[XYZn]***<VW> = 1 (binary)
- Word and byte represent the following bit length.
  Byte: 8 bits
  Half word: 16 bits
  Word: 32 bits
  Double word: 64 bits
- Properties of each bit in a register are expressed as follows:
  R: Read only
  W: Write only
  R/W: Read and write are possible.
- Unless otherwise specified, register access supports only word access.
- The register defined as "Reserved" must not be rewritten. Moreover, do not use the read value.
- The value read from the bit having default value of "-" is unknown.
- When a register containing both of writable bits and read-only bits is written, read-only bits should be written with their default value, In the cases that default is "-", follow the definition of each register.
- Reserved bits of the write-only register should be written with their default value. In the cases that default is "-", follow the definition of each register.
- Do not use read-modified-write processing to the register of a definition which is different by writing and read out.

Terms and Abbreviations

Some of abbreviations used in this document are as follows:

ADC	Analog to Digital Converter
A-ENC32	Advanced Encoder input Circuit (32-bit)
A-PMD	Advanced Programmable Motor Control Circuit
CG	Clock Control and Operation Mode
CRC	Cyclic Redundancy Check
D-Bus	DCode Memory Interface
DMAC	Direct Memory Access Controller
DNF	Digital Noise Filter
EHOSC	External High-speed Oscillator
EI2C	I^2C Interface Version A
fsys	Frequency of SYSTEM Clock
IA(INTIF)	Interrupt Control Register A
IB(INTIF)	Interrupt Control Register B
I-Bus	ICode Memory Interface
IHOSC	Internal High-speed Oscillator
IMN	Interrupt Monitor
INT	Interrupt
LVD	Voltage Detection Circuit
NMI	Non-maskable Interrupt
OFD	Oscillation Frequency Detector
POR	Power-on Reset Circuit
PORF	Power-on Reset Circuit for FLASH and Debug
RAMP	RAM Parity
RLM	Low-speed Oscillation/Power Supply Control/Reset
S-Bus	System Interface
SIWDT	Clock Selective Watchdog Timer
TRGSEL	Trigger Selection Circuit
TRM	Trimming Circuit
TSPI	Serial Peripheral Interface
T32A	32-bit Timer Event Counter
UART	Asynchronous Serial Communication Circuit

1. Clock Control and Operation Mode

1.1. Outlines

The clock/mode control block can select a clock gear and prescaler clock, and set the warming-up of oscillator and so on.

Furthermore, it has Normal mode and a low-power consumption mode in order to reduce power consumption using mode transition.

Functions related to a clock are as follows.

- System clock control
- Prescaler clock control

1.2. Clock Control

1.2.1. Clock Type

This section shows a list of clocks:

EHCLKIN	: The high-speed clock input from the external
fosc	: A clock generated in the internal oscillation circuit or input from the X1 and X2 pins after being selected by ***[CGOSCCR]***<OSCSEL>.
f_{PLL}	: A clock multiplied with PLL0
fc	: A clock selected by ***[CGPLL0SEL]***<PLL0SEL> (High-speed clock)
fsysh	: A high-speed system clock selected by ***[CGSYSCR]***<GEAR[2:0]>
fsysm	: A middle-speed system clock selected by ***[CGSYSCR]***<GEAR[2:0]><MCKSEL[1:0]>
ΦT0h	: A high-speed clock selected by ***[CGSYSCR]***<PRCK[3:0]> (High-speed prescaler clock)
ΦT0m	: A middle-speed clock selected by ***[CGSYSCR]***<PRCK[3:0]> <MCKSEL[1:0]> (Middle-speed prescaler clock)
f_{IHOSC1}	: A clock generated with the internal high-speed oscillator1
f_{IHOSC2}	: A clock generated with the internal high-speed oscillator2
ADCLK	: A conversion clock for ADC
TRCLKIN	: A clock for tracing facilities of a debugging circuit (Trace or SWV)

Note: The high-speed system clock and the middle-speed system clock are collectively called System clock (fsys). And the high-speed prescaler clock and the middle-speed prescaler clock are collectively called Prescaler clock (ΦT0).

1.2.2. Initial Value by Reset Action

A clock setup is initialized to the following states by a reset action.

External high-speed oscillator	: Stop
Internal high-speed oscillator1	: Oscillation
Internal high-speed oscillator2	: Stop
PLL (multiplying circuit)	: Stop
Gear clock	: fc (no frequency dividing)

1.2.3. Clock System Diagram

The figure below shows a clock system diagram.

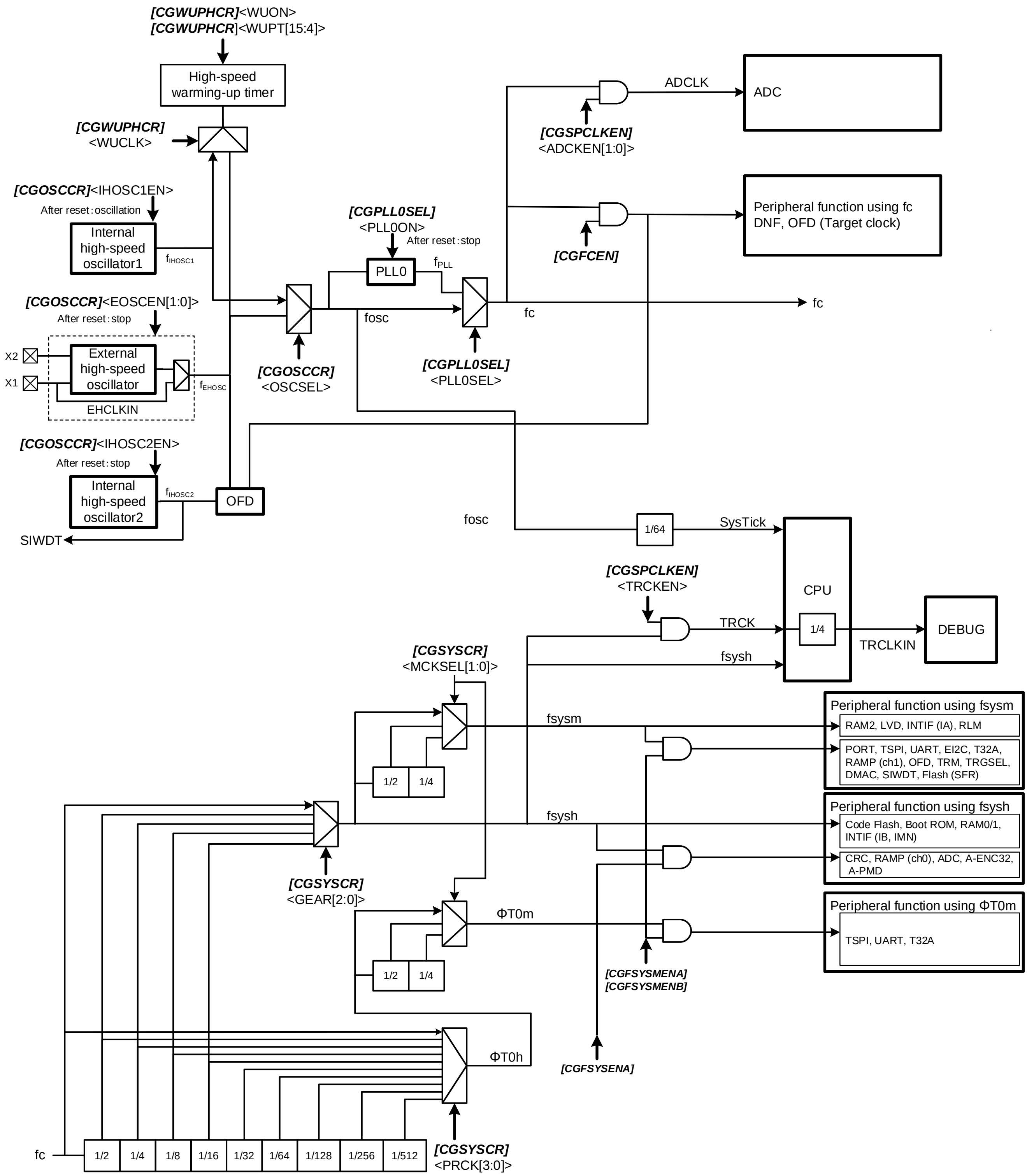

Figure 1.1 Clock System Diagram

1.2.4. Warming-up Function

A warming-up function starts the warming-up timer for high-speed oscillator automatically to secure the oscillation stable time when the STOP1 mode is released.

It is also available as a count-up timer which uses the warming-up timer for high-speed oscillator to secure the stability of an external oscillator or an internal oscillator.

This chapter explains the setting method to the register for warming-up timers, and the case where it is used as a count-up timer. The detailed explanation at the time of STOP1 mode release, refer to "1.3.3.2. Warming-up at Release of Low-power Consumption Mode".

1.2.4.1. Warming-up Timer for High-speed Oscillation

A 16-bit up-timer is built in as a warming-up timer only for a high-speed oscillation. Also when setting it before changes to the STOP1 mode, calculate with the following formula and set ***[CGWUPHCR]***<WUPT[15:4]> to the upper 12 bits of the result. Lower 4bits are ignored.

<Formula>

(Using external high-speed oscillator)

Warming-up timer setting value (16 bits) = (warming-up time (s) / clock period (s)) - 16

(Example) When 5 ms of warming-up time is set up with 10 MHz (100 ns of clock periods) of oscillators

Warming-up timer setting value (16 bits) = (5ms / 100ns) - 16
= 50000 - 16
= 49984
= 0xC340

Since upper 12 bits are used, set the register as follows.

[CGWUPHCR]<WUPT[15:4]> = 0xC34

(Using internal high-speed oscillator1)

Warming-up timer setting value (16 bits) = (warming-up time (s) – 63.3 (μs)) / clock period (s) - 41

(Example) When 163.4 μs of warming-up time is set up with 10 MHz (100 ns of clock periods) of oscillators

Warming-up timer setting value (16 bits) = ((163.4μs – 63.3μs) / 100ns) - 41
= (100.1μs – 100ns) - 41
= 960
= 0x03C0

Since upper 12 bits are used, set the register as follows.

[CGWUPHCR]<WUPT[15:4]> = 0x03C

The setting range is 0x03C ≤ <WUPT[15:4]> ≤ 0xFFF, warming-up time is set from 163.4 μs to 6.6194 ms.

1.2.4.2. Directions for Warming-up Timer

The directions for a warming-up function are explained.

(1) Selection of a clock

In a high-speed oscillation, the clock classification (internal oscillation/external oscillation) counted with a warming-up timer is selected by ***[CGWUPHCR]***<WUCLK>.

(2) Calculation of warming-up timer setting value

The warming-up time can set any value to the timer for a high-speed oscillation. Please compute and set up from the formula.

(3) The start of warming-up, and a termination Confirmation

When software (command) performs the start of warming-up, starting warming-up count is carried out by setting ***[CGWUPHCR]***<WUON> to "1". Termination is confirmed with ***[CGWUPHCR]***<WUEF> that becomes from "1" to "0". "1" shows that it is warming-up and "0" shows termination. After a counting end, a timer is reset and returns to an initial state.

It is not forced to terminate, although "0" is written to ***[CGWUPHCR]***<WUON> during timer operation. Writing "0" is disregarded.

Note: Since it is operating with the oscillating clock, a warming-up timer includes an error, when Oscillation frequency has fluctuation. Therefore, It serves an approximate time.

1.2.5. Clock Multiplying Circuit (PLL) for fsys

The clock multiplying circuit outputs the f_{PLL} clock (maximum 160MHz) multiplied by the optimum condition for the frequency (6 MHz to 24 MHz) of the output clock fosc of the high-speed oscillator.

So, it is possible to make input frequency to an oscillator low and to make an internal clock high-speed by this circuit.

1.2.5.1. PLL Setup after Reset Release

The PLL is disabled after reset release.

In order to use the PLL, set ***[CGPLL0SEL]***<PLL0SET[23:0]> to a multiplication value while ***[CGPLL0SEL]***<PLL0ON> is "0". Then wait until approximately 100 µs has elapsed as a PLL initial stabilization time, and set <PLL0ON> to "1" to start PLL operation. After that, to use f_{PLL} clock which is multiplied fosc, wait until approximately 400 µs has elapsed as a lock up time. Then set ***[CGPLL0SEL]***<PLL0SEL> to "1".

Note that a time is required until PLL operation becomes stable using the warming-up function, etc.

1.2.5.2. Formula and Example of Setting of PLL Multiplication Value

The details of the items of ***[CGPLL0SEL]***<PLL0SET[23:0]> which set up a PLL multiplication value are shown below.

Table 1.1 Details of *[CGPLL0SEL]*<PLL0SET[23:0]> Setup

Items of PLL0SET	Function		
[23:17]	Correction value setup	The quotient of fosc/450000 (integer). For details, refer to Table 1.2.	
[16:14]	fosc setup	000: 6 ≤ fosc ≤ 7 001: 7 < fosc ≤ 8 010: 8 < fosc ≤ 10 011: 10 < fosc ≤ 12 (unit: MHz)	100: 12 < fosc ≤15 101: 15 < fosc ≤ 19 110: 19 < fosc ≤ 24 111: Reserved
[13:12]	Dividing setup	00: Reserved 01: Dividing by 2 (×1/2) 10: Dividing by 4 (×1/4) 11: Dividing by 8 (×1/8)	
[11:8]	Fraction part multiplication setup	0000: 0.0000 0001: 0.0625 0010: 0.1250 0011: 0.1875 0100: 0.2500 0101: 0.3125 0110: 0.3750 0111: 0.4375	1000: 0.5000 1001: 0.5625 1010: 0.6250 1011: 0.6875 1100: 0.7500 1101: 0.8125 1110: 0.8750 1111: 0.9375
[7:0]	Integer part multiplication setup	0x00: 0 0x01: 1 0x02: 2 : 0xFD: 253 0xFE: 254 0xFF: 255	

Note: A multiplication value is the total of <PLL0SET[7:0]> (integer part) and <PLL0SET[11:8]> (fraction part).

f_{PLL} is denoted by the following formulas.

f_{PLL} = fosc × (***[CGPLL0SEL]***<PLL0SET[7:0]> + ***[CGPLL0SEL]***<PLL0SET[11:8]>) × (***[CGPLL0SEL]***<PLL0SET[13:12]>)

Note1: The absolute value of frequency accuracy is not guaranteed.

Note2: There is no Linearity in the frequency by the Fraction part Multiplication setup.

Note3: $f_{PLL} \leq$ (Maximum Operating Frequency)

Table 1.2 PLL Correction (Example)

fosc (MHz)	<PLL0SET[23:17]> (decimal, integral value)
6.00	14
8.00	18
10.00	23
12.00	27
24.00	54

The PLL correction value can be calculated below.

fosc = 10.0 MHz, 10.0/0.45 = 22.22 → 23; A fraction part is rounded up.

The main examples of a setting of ***[CGPLL0SEL]***<PLL0SET[23:0]> are shown below.

It multiplies by PLL, and dividing is carried out and the target Clock frequency (f_{PLL}) is generated for input frequency (fosc).

A dividing value is chosen from 1/2, 1/4, and 1/8.
Moreover, set up the frequency after multiplication in the following ranges.
200 MHz $\leq$ (fosc $\times$ Multiplication value) $\leq$ 400 MHz

Table 1.3 PLL0SET Setting Value (Example: f_{PLL} = 160MHz)

fosc (MHz)	Multiplication value	Dividing value	f_{PLL} (MHz)	<PLL0SET[23:0]>
6.00	53.3125	2	159.94	0x1C1535
8.00	40.0000	2	160	0x245028
10.00	32.0000	2	160	0x2E9020
12.00	26.6250	2	159.75	0x36DA1A
24.00	13.3125	2	159.75	0x6D950D

Table 1.4 PLL0SET Setting Value (Example: f_{PLL} = 120MHz)

fosc (MHz)	Multiplication value	Dividing value	f_{PLL} (MHz)	<PLL0SET[23:0]>
6.00	40.0000	2	120	0x1C1028
8.00	30.0000	2	120	0x24501E
10.00	24.0000	2	120	0x2E9018
12.00	20.0000	2	120	0x36D014
24.00	10.0000	2	120	0x6D900A

1.2.5.3. Change of PLL Multiplication Value under Operation

To change the setting of a PLL multiplication during PLL multiplication clock operation, set ***[CGPLL0SEL]***<PLL0SEL> to "0" that does not use a PLL multiplication clock. And ***[CGPLL0SEL]***<PLL0ST> = 0 is read to confirm that a multiplication clock setting is not used, then, ***[CGPLL0SEL]***<PLL0ON> is set to "0", and PLL is stopped.
Then, the multiplication value of ***[CGPLL0SEL]***<PLL0SET[23:0]> is changed, as reset time of PLL, after about 100 μs has elapsed, ***[CGPLL0SEL]***<PLL0ON> is set to "1", and operation of PLL is started.
Then, ***[CGPLL0SEL]***<PLL0SEL> is set to "1" after lock-up time (about 400μs) has elapsed.
Finally, ***[CGPLL0SEL]***<PLL0ST> is read and it checks having changed.

1.2.5.4. PLL Operation Start/Stop/Switching Procedure

(1) fc setup (PLL stop >>> PLL start)

As an fc setup, the example of switching procedure from the PLL stop state to the PLL operation state is as follows.

<< State before switching >>	
[CGPLL0SEL]<PLL0ON> = 0	Stops the PLL operation for fsys.
[CGPLL0SEL]<PLL0SEL> = 0	Selects the setting of the PLL for fsys to "PLL is unused (fosc)".
[CGPLL0SEL]<PLL0ST> = 0	Indicates the status of the PLL for fsys to "PLL is unused (fosc)".
[CGSYSCR]<MCKSEL> = 00	Ratios of (high-speed system clock vs middle-speed system clock) and (high-speed prescaler clock vs middle-speed system clock) are 1:1.

<< Example of switching procedure >>		
1	***[CGSYSCR]***<MCKSEL[1:0]> = 01 or 1*	Ratios of (high-speed system clock vs middle-speed system clock) and (high-speed prescaler clock vs high-speed system clock) are changed.
2	***[CGSYSCR]*** <MCKSELGST> <MCKSELPST> is read	Wait until they become the values set at Step 1.
3	***[CGPLL0SEL]***<PLL0SET[23:0]> = 0xX	A PLL multiplication value setup is chosen.
4	Wait 100 µs or more.	Latency time after a multiplication setup
5	***[CGPLL0SEL]***<PLL0ON> = 1	PLL operation for fsys is carried out to an oscillation.
6	Wait 400 µs or more.	PLL output clock stable latency time
7	***[CGPLL0SEL]***<PLL0SEL> = 1	PLL selection for fsys is carried out to PLL use (f_{PLL}).
8	***[CGPLL0SEL]***<PLL0ST> is read	It waits until the PLL selection status for fsys becomes PLL use (f_{PLL}) (=1).

Note1: 1 and 2 are executed when the ratio of the system clock should be changed.

Note2: 3 to 6 are unnecessary when the state before switching is ***[CGPLL0SEL]***<PLL0ON> = 1.
When changing from the state where the PLL output clock is stable, it can be changed to the PLL operation state by execution of only 7 and 8.

(2) fc setup (conduct PLL >>> PLL stop)

As an fc setup, the example of switching procedure from the PLL operation state to a PLL stop state is as follows.

<< State before switching >>	
[CGPLL0SEL]<PLL0ON> = 1	Sets the PLL for fsys to oscillate.
[CGPLL0SEL]<PLL0SEL> = 1	Selects the PLL for fsys to "PLL is used (f_{PLL})".
[CGPLL0SEL]<PLL0ST> = 1	Indicates the status of the PLL for fsys to "PLL is used (f_{PLL})".

<< Example of switching sequence >>		
1	***[CGPLL0SEL]***<PLL0SEL> = 0	Selects the PLL for fsys to "PLL is unused (fosc)".
2	***[CGPLL0SEL]***<PLL0ST> is read	Waits until the status of the PLL for fsys becomes "PLL is unused (fosc) (=0)".
3	***[CGPLL0SEL]***<PLL0ON> = 0	Sets the PLL operation for fsys to stop.

1.2.6. System Clock

An internal high-speed oscillation clock or external high-speed oscillation clock (connected oscillator or clock input) can be used as a source of system clock.

The system clock consists of "High-speed system clock (fsysh) (maximum 160MHz)" for high-speed operation and "Middle-speed system clock (fsysm) (maximum 80MHz)" which is generated by dividing high-speed system clock. Middle-speed system clock is used by peripheral function to save power dissipation without degrading CPU performance. The clock domains of the peripheral function can be checked in Table 1.5.

High-speed system clock can be generated by dividing fc using ***[CGSYSCR]***<GEAR[2:0]> (Clock gear). And middle-speed system clock is generated by dividing the high-speed system clock using ***[CGSYSCR]*** <MCKSEL[1:0]>. Although a setting can be changed during operation, after register writing before the clock actually changes, a time interval shown in Table 1.6 is required. The completion of the clock change should be checked by ***[CGSYSCR]***<GEARST[2:0]> <MCKSELGST[1:0]>.

Table 1.5 Clock Domains of CPU and Peripherals

Clock domain	Block
High-speed system clock	CPU, Code Flash, Boot ROM, RAM0/1, CG, INTIF (IB, IMN), CRC, RAMP (ch0), ADC, A-PMD, A-ENC32
Middle-speed system clock	DMAC, SIWDT, UART, TSPI, I2C, EI2C, T32A, PORT, INTIF (IA), DNF, LVD, TRM, Flash (SFR), OFD, RAMP (ch1), RLM, TRGSEL, RAM2

Table 1.6 Time Interval for Changing System Clock

System clock	High-speed (fsysh)	Middle-speed (fsysm)
fsys	16 fc cycles at maximum	16 fc cycles at maximum
fsys/2	-	32 fc cycles at maximum
fsys/4	-	64 fc cycles at maximum

Note1: The clock gear and the system clock should not be changed while the peripheral function such as the timer/counter is operating.

Note2: An access between high-speed system clock domain and middle-speed system clock domain cannot be done when the system clock is changing.

The table below shows the example of operating frequency by the clock gear ratio (1/1 to 1/16) to the frequency fc set up with oscillation frequency, a PLL multiplication value, etc.

Table 1.7 Example of Operating Frequency (fc = 160MHz)

External oscillation (MHz)	External clock input (MHz)	Built-in oscillation IHOSC1 (MHz)	PLL multiplication value (after dividing)	Maximum frequency (fc) (MHz)	Operating frequency (MHz) by clock gear ratio PLL = ON					Operating frequency (MHz) by clock gear ratio PLL = OFF				
					1/1	1/2	1/4	1/8	1/16	1/1	1/2	1/4	1/8	1/16
6	6	-	26.66	159.94	159.94	79.97	39.99	20	10	6	3	1.5	-	-
8	8	-	20	160	160	80	40	20	10	8	4	2	1	-
10	10	10	16	160	160	80	40	20	10	10	5	2.5	1.25	-
12	12	-	13.33	159.75	159.75	79.88	39.94	19.97	9.98	12	6	3	1.5	-
24	24	-	6.66	159.75	159.75	79.88	39.94	19.97	9.98	24	12	6	3	1.5

Table 1.8 Example of Operating Frequency (fc = 120MHz)

External oscillation (MHz)	External clock input (MHz)	Built-in oscillation IHOSC1 (MHz)	PLL multiplication value (after dividing)	Maximum frequency (fc) (MHz)	Operating frequency (MHz) by clock gear ratio PLL = ON					Operating frequency (MHz) by clock gear ratio PLL = OFF				
					1/1	1/2	1/4	1/8	1/16	1/1	1/2	1/4	1/8	1/16
6	6	-	20	120	120	60	30	15	7.5	6	3	1.5	-	-
8	8	-	15	120	120	60	30	15	7.5	8	4	2	1	-
10	10	10	12	120	120	60	30	15	7.5	10	5	2.5	1.25	-
12	12	-	10	120	120	60	30	15	7.5	12	6	3	1.5	-
24	24	-	5	120	120	60	30	15	7.5	24	12	6	3	1.5

Table 1.9 Operating Frequency Examples of High-speed and Middle-speed System Clocks

High-speed system clock fsysh (MHz)	Middle-speed system clock fsysm (MHz)		
	1/1	1/2	1/4
160	-	80	40
120	-	60	30
80	80	40	20

Note: The maximum frequency of middle-speed system clock is 80 MHz.

1.2.6.1. Setting Method of System Clock

(1) fosc setup (Internal oscillation >>> External oscillation)

As a fosc setup, the example of switching procedure to the external high-speed oscillator (EHOSC) from an internal high-speed oscillator1 (IHOSC1) is shown below.

<< State before switching >>	
[CGOSCCR]<IHOSC1EN> = 1	An internal high-speed oscillator1 oscillates.
[CGOSCCR]<OSCSEL> = 0	The high-speed oscillation selection for fosc is an internal high-speed oscillator1 (IHOSC1).
[CGOSCCR]<OSCF> = 0	The high-speed oscillation selection status for fosc is an internal high-speed oscillator1 (IHOSC1).
An oscillator is connected to X1 / X2 pin.	Do not connect any devices except a resonator.

<< Example of switching procedure >>		
1	***[PHPDN]***<bit[1:0]> = 00 ***[PHIE]***<bit[1:0]> = 00	Disable the pull-down resistors of X1 and X2 pins. Disable input control of X1 and X2 pins.
2	***[CGOSCCR]***<EOSCEN[1:0]> = 01	It is an external high-speed oscillator (EHOSC) about selection of an external oscillation of operation.
3	***[CGWUPHCR]***<WUCLK> = 1 ***[CGWUPHCR]***<WUPT[15:4]> = arbitrary value	It is the external high-speed oscillator (EHOSC) about high-speed oscillation warming-up clock selection. A warming-up timer setting value is set to oscillator stable time.
4	***[CGWUPHCR]***<WUON> = 1	High-speed oscillation warming-up is started.
5	***[CGWUPHCR]***<WUEF> is read.	It waits until it becomes the termination of high-speed oscillation warming-up (= 0).
6	***[CGOSCCR]***<OSCSEL> = 1	It is high-speed oscillation selection for fosc to the external high-speed oscillator (EHOSC).
7	***[CGOSCCR]***<OSCF> is read	It waits until the high-speed oscillation selection status for fosc becomes external high-speed oscillator (=1).
8	***[CGOSCCR]***<IHOSC1EN> = 0	An internal high-speed oscillator1 is suspended.

(2) fosc setup (Internal oscillation >>> External clock input)

As a fosc setup, the example of switching procedure to the external clock input (EHCLKIN) from an internal oscillator1 (IHOSC1) is shown below.

<< State before switching >>	
[CGOSCCR]<IHOSC1EN> = 1	An internal high-speed oscillator1 oscillates.
[CGOSCCR]<OSCSEL> = 0	The high-speed oscillation selection for fosc is an internal high-speed oscillator1 (IHOSC1).
[CGOSCCR]<OSCF> = 0	The high-speed oscillation selection status for fosc is an internal high-speed oscillator1 (IHOSC1).
Clock into to EHCLKIN	Input in the proper voltage range.

<< Example of switching procedure >>		
1	***[PHPDN]***<bit[0]> = 0 ***[PHIE]***<bit[0]> = 0/1	Disable the pull-down resistor of X1/EHCLKIN pin. The input control of X1/EHCLKIN pin is arbitrary.
2	***[CGOSCCR]***<EOSCEN[1:0]> = 10	Selection of an external high-speed oscillation of operation is carried out to an external clock input (EHCLKIN).
3	***[CGOSCCR]***<OSCSEL> = 1	It is high-speed oscillation selection for fosc to an external clock.
4	***[CGOSCCR]***<OSCF> is read	It waits until the high-speed oscillation selection status for fosc becomes external high-speed oscillator (= 1).
5	***[CGOSCCR]***<IHOSC1EN> = 0	An internal high-speed oscillator1 is suspended.

(3) fosc setup (External oscillation/External clock input >>> Internal oscillation)

As a fosc setup, the example of switching procedure to the internal high-speed oscillator1 (IHOSC1) from an external high-speed oscillator (EHOSC) Operation State or an external clock input (EHCLKIN) Operation State is shown below.

<< State before switching >>	
[CGOSCCR]<EOSCEN[1:0]> = 01 or 10	Selection of an external oscillator of operation is an external high-speed oscillator (EHOSC) or external clock input.
[CGOSCCR]<OSCSEL> = 1	The high-speed oscillation selection for fosc is the external high-speed oscillator (EHOSC).
[CGOSCCR]<OSCF> = 1	The high-speed oscillation selection status for fosc is the external high-speed oscillator (EHOSC).

<< Example of switching procedure >>		
1	***[CGWUPHCR]***<WUCLK> = 0	Set the warming-up clock selection to internal high-speed oscillator1 (IHOSC1).
2	***[CGWUPHCR]***<WUPT[15:4]> = 0x03C	Set the high-speed oscillation warming-up timer setting value of 163.4 μs (= 0x03C) or more.
3	***[CGOSCCR]***<IHOSC1EN> = 1	An internal high-speed oscillator1 is oscillated.
4	***[CGWUPHCR]***<WUON> = 1	Start the high-speed oscillation warming-up timer
5	***[CGWUPHCR]***<WUEF> is read	Wait until an warming-up timer status flag becomes ends (= 0).
6	***[CGOSCCR]***<OSCSEL> = 0	Set the high-speed oscillation selection for fosc to internal high-speed oscillator1 (IHOSC1).
7	***[CGOSCCR]***<OSCF> is read	It waits until the high-speed oscillation selection status for fosc becomes an internal high-speed oscillator1 (= 0).
8	***[CGOSCCR]***<EOSCEN[1:0]> = 00	Set the selection of an external high-speed oscillator operation to unused.

1.2.7. Clock Supply Setting Function

This MCU has the clock supply on/off function for the peripheral circuits. To reduce the power consumption, this MCU can stop supplying the clock to the peripheral functions that are not used.

Except some peripheral functions, clocks are not supplied after reset.

In order to supply the clock of the function to be used, set the bit of relevance of ***[CGFSYSENA]***, ***[CGFSYSMENA]***, ***[CGFSYSMENB]***, ***[CGFCEN]*** and ***[CGSPCLKEN]*** to "1".

For details, refer to "1.4 Explanation of Register".

1.2.8. Prescaler Clock

Each peripheral function has a prescaler circuit to divide the ΦT0 clock. The ΦT0 clock which is input into the prescaler circuit can be divided by the ***[CGSYSCR]***<PRCK[3:0]> to generate High-speed prescaler clock. And Middle-speed prescaler clock is generated by dividing High-speed prescaler clock using ***[CGSYSCR]*** <MCKSEL[1:0]>. For ΦT0 clock after reset, fc is chosen.

After register writing before a clock actually changes, a time interval shown in Table 1.10 is required.

To confirm the completion of the clock change, check the status of ***[CGSYSCR]***<PRCKST[3:0]> <MCKSELPST[1:0]>.

Table 1.10 Time Interval for Changing Prescaler Clocks

Prescaler clock	High-speed (ΦT0h)	Middle-speed (ΦT0m)
ΦT0	512 fc cycles at maximum	512 fc cycles at maximum
ΦT0/2	-	1024 fc cycles at maximum
ΦT0/4	-	2048 fc cycles at maximum

Note1: Do not change a prescaler clock during operation of peripheral functions, such as a timer counter.

Note2: An access between high-speed system clock domain and middle-speed system clock domain cannot be done when the prescaler clock is changing.

1.3. Operation Mode

There are NORMAL mode and a low-power consumption mode (IDLE, STOP1) in this product as an Operation mode, and it can reduce power consumption by performing mode changes according to directions for use.

1.3.1. Details of Operation Mode

1.3.1.1. Feature in Each Mode

The feature in NORMAL, low-power consumption modes is as follows.

- NORMAL mode

  CPU core and peripheral circuits operate with the high-speed oscillation clock. After reset release, the system operates in NORMAL mode.

- Low-power consumption mode

  The feature in low-power consumption modes is as follows.

  - IDLE mode

    It is the mode which CPU stops.

    The peripheral function should perform operation/stop by the register of each peripheral function, a clock supply setting function, etc.

    Note: Note that the CPU cannot clear the watchdog timer in IDLE mode.

  - STOP1 mode

    In this mode, all the internal circuits including the internal oscillator stop.

    If STOP1 mode is canceled, the internal high-speed oscillator1 (IHOSC1) will start oscillation, and the system will return to NORMAL mode.

    Please disable interrupt which is not used for STOP1 release before shifting to the STOP1 mode.

1.3.1.2. Transition to and Return from Low-power Consumption Mode

In order to shift to each low-power consumption mode, the IDLE/STOP1 mode is chosen by standby control register ***[CGSTBYCR]***<STBY[1:0]>, and a WFI (Wait For Interrupt) command is executed. When the transition to the low-power consumption mode has been done by WFI instruction, the return from the mode can be done by the reset or an interrupt generation. To return by interrupt, it is necessary to set up. Please refer to "Interrupts" chapter of the reference manual "Exception" for details.

Note1: This product does not support a return by events; therefore, do not make a transition to low-power consumption mode triggered by WFE (Wait For Event).

Note2: This product does not support low-power consumption mode by SLEEPDEEP of the Cortex-M4 processor with FPU core. Do not use the <SLEEPDEEP> bit of the system control register.

1.3.1.3. Selection of Low-power Consumption Mode

Low-power consumption mode selection is chosen by setup of ***[CGSTBYCR]***<STBY[1:0]>.

Following table shows the mode chosen from a setup of <STBY[1:0]>.

Table 1.11 Low-power Consumption Mode Selection

Mode	*[CGSTBYCR]*<STBY[1:0]>
IDLE	00
STOP1	01

Note: Do not use the settings other than the above.

1.3.1.4. Peripheral Function State in Low-power Consumption Mode

The following Table 1.12 shows the Operation State of the peripheral function (block) in each mode.
In addition, after reset release, it will be in the state where a clock is not supplied except for some blocks.
If needed, set up ***[CGFSYSENA]***, ***[CGFSYSMENA]***, ***[CGFSYSMENB]***, ***[CGFCEN]***, ***[CGSPCLKEN]*** and enable clock supply.

Table 1.12 Block Operation Status in Each Low-power Consumption Mode

Block		NORMAL	IDLE	STOP1
Processor core (Including Debug)		✓	-	-
DMAC		✓	✓	-
I/O port	Pin status	✓	✓	✓
	Register	✓	✓	-
ADC		✓	✓	-
UART		✓	✓	-
EI2C		✓	✓	-
TSPI		✓	✓	-
A-PMD		✓	✓	-
A-ENC32		✓	✓	-
T32A		✓	✓	-
TRGSEL		✓	✓	-
CRC		✓	✓	-
SIWDT		✓	✓ (Note1)	-
LVD		✓	✓	✓
OFD		✓	✓	-
TRM		✓	Unavailable	-
CG		✓	✓	✓
PLL		✓	✓	-
RAMP		✓	✓	-
External high-speed oscillator (EHOSC)		✓	✓	-
Internal high-speed oscillator1 (IHOSC1)		✓	✓	-
Internal high-speed oscillator2 (IHOSC2)		✓	✓	-
Code Flash		Access possible	Access possible (Note2)	Data hold
RAM				

✓: Operation is possible.

-: If it shifts to the object mode, the clock to peripheral circuits stop automatically.

Note1: Protect A mode only. In other cases, stop SIWDT before transiting to IDLE mode.

Note2: It becomes a data hold when peripheral functions (DMA etc.) except CPU which carry out data access (R/W) are not connected on the bus matrix.

1.3.2. Mode State Transition

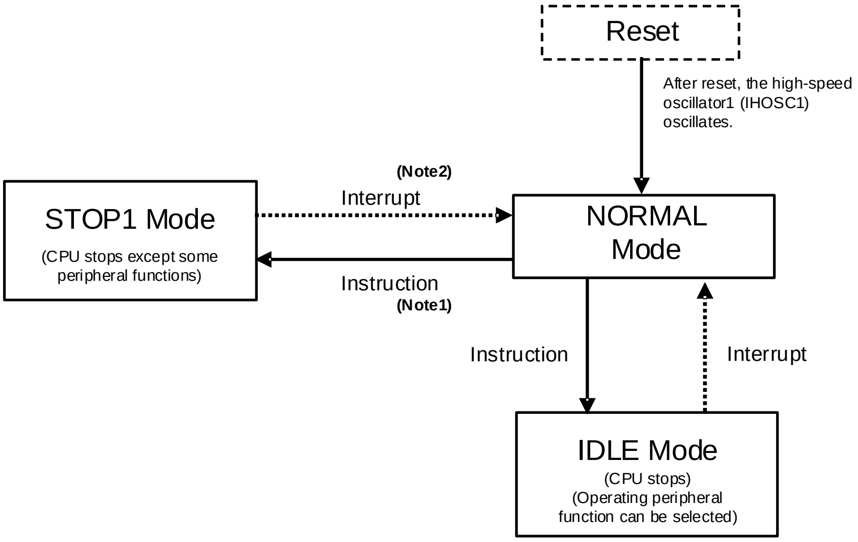

Figure 1.2 Mode State Transition

Note1: Warming-up is required at returning. A warming-up time must be set in the previous mode (NORMAL mode) before entering to STOP1 mode.

Note2: When the MCU returns from STOP1 mode, the MCU branches to the interrupt service routine triggered by interrupt events.

1.3.2.1. IDLE Mode Transition Flow

Set up the following procedure at switching to IDLE mode.

Because IDLE mode is released by an interrupt, set the interrupt before switching to IDLE mode. For the interrupts that can be used to release the IDLE mode, refer to "1.3.3.1. Release Source of Low-power Consumption Mode". Disable interrupts not used for release and interrupts that cannot be used.

Transition flow (from NORMAL mode)		
1	***[SIWDxEN]***<WDTE> = 0	Disable SIWDT.
2	***[SIWDxCR]***<WDCR[7:0]> = 0xB1	Disable SIWDT.
3	***[FCSR0]***<RDYBSY> is read.	It waits until Flash will be in a ready state (= 1).
4	***[CGSTBYCR]***<STBY[1:0]> = 00	Low-power consumption mode selection is set to IDLE.
5	***[CGSTBYCR]***<STBY[1:0]> is read.	Check the 4th line register writing (= 00).
6	WFI command execution	Switch to IDLE.

Note: When using the protected A mode of SIWDT, 1 and 2 step are not required.

1.3.2.2. STOP1 Mode Transition Flow

Set up the following procedure at switching to STOP1.

Because STOP1 mode is released by an interrupt, set the interrupt before switching to STOP1 mode. For the interrupts that can be used to release the STOP1 mode, refer to "1.3.3.1. Release Source of Low-power Consumption Mode". Disable interrupts not used for release and interrupts that cannot be used.

Transition flow (from NORMAL mode)		
1	***[SIWDxEN]***<WDTE> = 0	Disable SIWDT.
2	***[SIWDxCR]***<WDCR[7:0]> = 0xB1	Disable SIWDT.
3	***[FCSR0]***<RDYBSY> is read.	Wait until Flash becomes the ready state (= 1).
4	***[CGWUPHCR]***<WUEF> is read.	Wait until the high-speed oscillation warming-up ends (= 0).
5	***[CGWUPHCR]***<WUCLK> = 0	Set the warming-up clock selection to internal high-speed oscillator1 (IHOSC1).
	[CGWUPHCR]<WUPT[15:4]> = 0x03C	Set the high-speed oscillation warming-up timer setting value to 163.4 µs (= 0x03C) or more.
6	***[CGSTBYCR]***<STBY[1:0]> = 01	Low-power consumption mode selection is set to STOP1.
7	***[CGPLL0SEL]***<PLL0SEL> = 0	Set PLL of fsys to fosc (= PLL no USE)
8	***[CGPLL0SEL]***<PLL0ST> is read.	Wait until PLL status of fsys becomes off state (= 0).
9	***[CGPLL0SEL]***<PLL0ON> = 0	Stop PLL for fsys
10	***[CGOSCCR]***<IHOSC1EN> = 1	Enable the internal high-speed oscillator1.
11	***[CGWUPHCR]***<WUON> = 1	Start the high-speed oscillation warming-up timer
12	***[CGWUPHCR]***<WUEF> is read	Wait until an warming-up timer status flag becomes ends (= 0).
13	***[CGOSCCR]***<OSCSEL> = 0	Set the high-speed oscillation selection for fosc to internal high-speed oscillator1 (IHOSC1).
14	***[CGOSCCR]***<OSCF> is read.	Wait until the high-speed oscillation selection status for fosc becomes internal high-speed oscillator1 (IHOSC1) (= 0).
15	***[CGOSCCR]***<EOSCEN[1:0]> = 00	Selection of an external oscillator1 is set to "Unused".
16	***[CGOSCCR]***<IHOSC2EN> = 0	The internal high-speed oscillator2 (IHOSC2) is stopped.
17	***[CGOSCCR]***<EOSCEN[1:0]> is read.	The register writing of above 15th is checked (= 00).
18	***[CGOSCCR]***<IHOSC2F> is read.	Wait until the status of IHOSC2 becomes "0".
19	WFI command execution	Switch to STOP1.

1.3.3. Return from Low-power Consumption Mode

1.3.3.1. Release Source of Low-power Consumption Mode

Interrupt, Non-maskable interrupt, and reset can perform return from a low-power consumption mode. The standby release source which can be used is decided by a low-power consumption mode.

It shows the following table about details.

Table 1.13 Release Source List

Low-power Consumption mode			IDLE	STOP1
Release source	Interrupt	INT0 to INTF (Note)	✓	✓
		INTEMGx, INTOVVx, INTPWMx	✓	-
		INTENCx0, INTENCx1	✓	-
		INTADxPDA, INTADxPDB, INTADxCP0, INTADxCP1, INTADxTRG, INTADxSGL, INTADxCNT	✓	-
		INTSCxRX, INTSCxTX, INTSCxERR	✓	-
		INTI2CxTBE, INTI2CxRBF, INTI2CxST	✓	-
		INTT32AxAC, INTT32AxACCAP0, INTT32AxACCAP1, INTT32AxB, INTT32AxBCAP0, INTT32AxBCAP1	✓	-
		INTPARIx	✓	-
		INTDMAATC,INTDMAAERR	✓	-
		INTFLCRDY	✓	-
	SysTick interrupt		✓	-
	Non-maskable interrupt (INTWDT0)		✓	-
	Non-maskable interrupt (INTLVD)		✓	✓
	Reset (SIWDT)		✓	-
	Reset (LVD)		✓	✓
	Reset (OFD)		✓	-
	Reset (RESET_N pin)		✓	✓

✓: After release, the interrupt procedure will start.

-: It cannot be used for release.

Note: INT0 to INTF (External Interrupt) can select one of falling edge, rising edge and level. For details, please refer to the reference manual "Exception".

- Released by an interrupt request

  When interrupt cancels a low-power consumption mode, it is necessary to prepare so that interrupt may be detected by CPU. The interrupt used for release in STOP1 mode needs to set up CPU, and needs to set up detection by INTIF.

- Released by Non-maskable interrupt (NMI)

  The SIWDT interrupt (INTWDT0, Protect A mode only.) or the LVD interrupt (INTLVD) can perform release from the low-power consumption modes.

- Released by reset

  The reset can perform release from all the low-power consumption modes.

  When released by reset, the registers will be initialized in NORMAL mode after release. For details, refer to "3.2.6.1. Reset Factor and Reset Range".

- Released by SysTick interrupt

  SysTick interrupt is available only in IDLE mode.

Refer to "Interrupt" chapter of the reference manual "Exception" for details of interrupt.

1.3.3.2. Warming-up at Release of Low-power Consumption Mode

Warming-up may be required because of stability of an internal oscillator at the time of mode transition.

When the transition from STOP1 mode to NORMAL mode is done, the internal oscillation is selected automatically and the warming-up timer starts up. The Output of a system clock is started after warming-up time progress.

For this reason, before executing the command which move to the STOP1 mode, set up warming-up time by ***[CGWUPHCR]***<WUPT[15:4]>. For the setting method, refer to "1.2.4.1. Warming-up Timer for High-speed Oscillation".

The following table shows the necessity of a warming-up setup at the time of each Operation mode transition.

Table 1.14 Warming-up

Operation mode transition	Warming-up setup
NORMAL >>> IDLE	Not required.
NORMAL >>> STOP1	Not required.
IDLE >>> NORMAL	Not required.
STOP1 >>> NORMAL	Required.

1.3.4. Clock Operation by Mode Transition

The clock operation in case of mode transition is shown below.

1.3.4.1. NORMAL >>> IDLE >>> NORMAL Operation Mode Transition

CPU stops at IDLE mode. The clock supply to a peripheral function holds a setting state. Please perform operation/stop by the register of each peripheral function, a clock supply setting function, etc. if needed. Execution of warming-up operation is not performed at the time of the restart operation in NORMAL mode from IDLE state.

After the command (WFI) execution which switches to IDLE mode, a program counter will show the next point and will be in a CPU idle state. With a release source, it becomes a CPU reboot and, in the case of an enable interrupt state, the shift to next point of the transition command (WFI) will be done, after the interrupt processing by release source.

1.3.4.2. NORMAL >>> STOP1 >>> NORMAL Operation Mode Transition

When returning to NORMAL mode from the STOP1 mode, warming-up is started automatically.

Please set ***[CGWUPHCR]***<WUPT[15:4]> to warming-up time (163.4 μs or more) before moving to the STOP1 mode.

Note: When the RESET_N pin and LVD reset are the release factors, warming-up time is the same as the operation at the time of warm-reset, and is replaced by "internal processing time" and "CPU operation wait time". For details, refer to "3.2.2.1. Warm Reset by RESET_N Pin" and "3.2.2.2. Warm Reset by LVD".

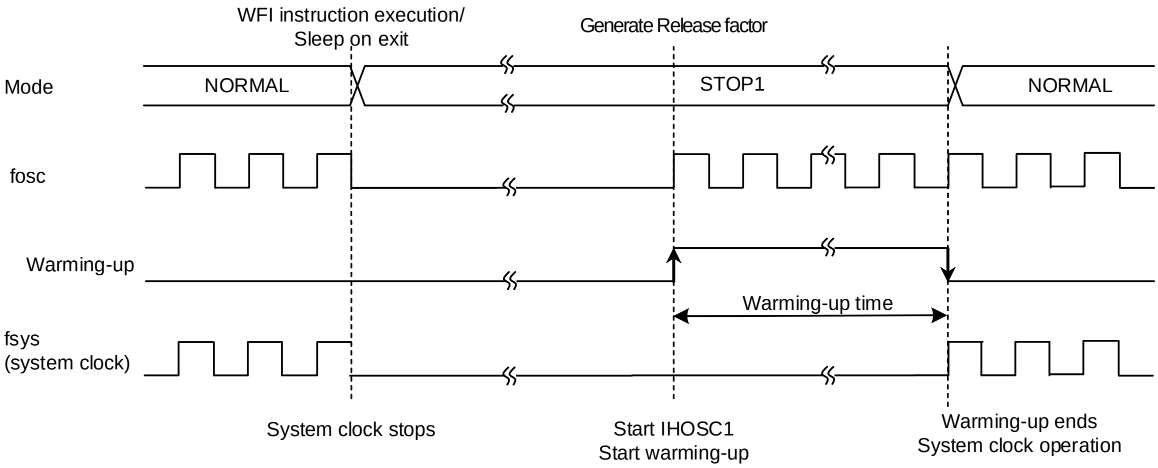

Figure 1.3 NORMAL >>> STOP1 >>> NORMAL Operation Mode Transition

1.4. Explanation of Register

1.4.1. Register List

The register related to CG and its address information are shown below.

Peripheral function		Channel/unit	Base address
Clock control and operation mode	CG	-	0x40083000

Register name		Address (Base+)
CG write protection register	***[CGPROTECT]***	0x0000
Oscillation control register	***[CGOSCCR]***	0x0004
System clock control register	***[CGSYSCR]***	0x0008
Standby control register	***[CGSTBYCR]***	0x000C
PLL selection register for fsys	***[CGPLL0SEL]***	0x0020
High-speed oscillation warming-up register	***[CGWUPHCR]***	0x0030
Supply and stop register A for fsysm	***[CGFSYSMENA]***	0x0048
Supply and stop register B for fsysm	***[CGFSYSMENB]***	0x004C
Supply and stop register A for fsysh	***[CGFSYSENA]***	0x0050
Clock supply and stop register for fc	***[CGFCEN]***	0x0058
Clock supply and stop register for ADC and Debug circuit	***[CGSPCLKEN]***	0x005C

1.4.2. Detail of Register

1.4.2.1. *[CGPROTECT]* (CG Write Protection Register)

Bit	Bit symbol	After reset	Type	Function
31:8	-	0	R	Read as “0”.
7:0	PROTECT[7:0]	0xC1	R/W	Control write-protection for the CG register (all registers included except this register) 0xC1: CG Registers are write-enabled. Other than 0xC1: Sets write protection (Protect enable)

1.4.2.2. *[CGOSCCR]* (Oscillation Control Register)

Bit	Bit symbol	After reset	Type	Function
31:20	-	0	R	Read as “0”.
19	IHOSC2F	0	R	Indicates the stability flag of internal oscillation for IHOSC2 0: Stopping or being in warming-up 1: Stable oscillation
18:17	-	0	R	Read as “0”.
16	IHOSC1F	1	R	Indicates the stability flag of internal oscillation for IHOSC1 (Note4) 0: Stopping or being in warming-up 1: Stable oscillation
15:13	-	0	R	Read as “0”.
12	-	0	R/W	Write as “0”.
11:10	-	0	R	Read as “0”.
9	OSCF	0	R	Indicates high-speed oscillator for fosc selection status. 0: Internal high-speed oscillator1 (IHOSC1) 1: External high-speed oscillator (EHOSC)
8	OSCSEL	0	R/W	Selects a high-speed oscillation for fosc. (Note1) 0: Internal high-speed oscillator1 (IHOSC1) 1: External high-speed oscillator (EHOSC)
7:4	-	0	R	Read as “0”.
3	IHOSC2EN	0	R/W	Enables the internal high-speed oscillator2 (IHOSC2) (Note 2) 0: Stop 1: Oscillation
2:1	EOSCEN[1:0]	00	R/W	Selects the operation of the external high-speed oscillator. (EHOSC) (Note3) 00: External high-speed oscillator is not used 01: Uses the external high-speed oscillator (EHOSC) 10: Uses the external clock (EHCLKIN) 11: Reserved
0	IHOSC1EN	1	R/W	Internal high-speed oscillator1 (IHOSC1) 0: Stop 1: Oscillation

Note1: When the setting is modified, confirm whether the written value has been reflected to the ***[CGOSCCR]*** <OSCF> bit before executing the next operation.

Note2: Setting cannot be changed, when it is ***[SIWDxOSCCR]***<OSCPRO> = 1 (Write protection of SIWDT is effective)

Note3: When using the oscillator connection, set this bit to "01" (external high-speed oscillator).

Note4: To wait stabilizing oscillation of an internal high-speed oscillator1 (IHOSC1), use a warming-up timer and confirm ***[CGWUPHCR]***<WUEF> instead of <IHOSC1F>.

1.4.2.3. *[CGSYSCR]* (System Clock Control Register)

Bit	Bit symbol	After reset	Type	Function
31:30	MCKSELPST[1:0]	00	R	Middle-speed prescaler clock (ΦT0m) selection status 00: <PRCK[3:0]> setting value (no division) 01: <PRCK[3:0]> setting value is divided by 2 10,11: <PRCK[3:0]> setting value is divided by 4
29:28	-	0	R	Read as “0”.
27:24	PRCKST[3:0]	0000	R	High-speed prescaler clock (ΦT0h) selection status 0000: fc 0100: fc/16 1000: fc/256 0001: fc/2 0101: fc/32 1001: fc/512 0010: fc/4 0110: fc/64 1010 to 1111: Reserved 0011: fc/8 0111: fc/128
23:22	MCKSELGST[1:0]	00	R	Middle-speed system clock (fsysm) selection status 00: <GEAR[2:0]> setting value (no division) 01: <GEAR[2:0]> setting value is divided by 2 10,11: <GEAR[2:0]> setting value is divided by 4
21:19	-	0	R	Read as “0”.
18:16	GEARST[2:0]	000	R	High-speed system clock (fsysh) gear selection status 000: fc 100: fc/16 001: fc/2 101 to 111: Reserved 010: fc/4 011: fc/8
15:12	-	0	R	Read as “0”.
11:8	PRCK[3:0]	0000	R/W	High-speed prescaler clock (ΦT0h) selection 0000: fc 0100: fc/16 1000: fc/256 0001: fc/2 0101: fc/32 1001: fc/512 0010: fc/4 0110: fc/64 1010 to 1111: Reserved 0011: fc/8 0111: fc/128 Selects a prescaler clock for the peripheral functions.
7:6	MCKSEL[1:0]	00	R/W	Middle-speed system clock (fsysm) and middle-speed prescaler clock (ΦT0m) selection 00: <GEAR[2:0]>, <PRCK[3:0]> setting values (no division) 01: <GEAR[2:0]>, <PRCK[3:0]> setting values are divided by 2. 10,11: <GEAR[2:0]>,<PRCK[3:0]> setting values are divided by 4. Maximum operating frequency of middle-speed system clock is 80MHz.
5:3	-	0	R	Read as “0”.
2:0	GEAR[2:0]	000	R/W	High-speed system clock (fsysh) gear selection 000: fc 100: fc/16 001: fc/2 101 to 111: Reserved 010: fc/4 011: fc/8

1.4.2.4. *[CGSTBYCR]* (Standby Control Register)

Bit	Bit symbol	After reset	Type	Function
31:2	-	0	R	Read as "0".
1:0	STBY[1:0]	00	R/W	Selects a low-power consumption mode. 00: IDLE 01: STOP1 10: Reserved 11: Reserved

1.4.2.5. *[CGPLL0SEL]* (PLL Selection Register for fsys)

Bit	Bit symbol	After reset	Type	Function
31:8	PLL0SET[23:0]	0x000000	R/W	PLL0 multiplication setup About a multiplication setup, refer to "1.2.5.2. Formula and Example of Setting of PLL Multiplication Value".
7:3	-	0	R	Read as "0".
2	PLL0ST	0	R	Indicates PLL for fsys selection status. 0: fosc 1: f_{PLL}
1	PLL0SEL	0	R/W	Selects clock selection for fsys 0: fosc 1: f_{PLL}
0	PLL0ON	0	R/W	Selects PLL operation for fsys 0: Stop 1: Oscillation

1.4.2.6. *[CGWUPHCR]* (High-speed Oscillation Warming-up Register)

Bit	Bit symbol	After reset	Type	Function
31:20	WUPT[15:4]	0x800	R/W	Sets the upper 12 bits of the 16 bits of calculation values of the warming-up timer. About a setup of a warming-up timer, refer to "1.2.4.1. Warming-up Timer for High-speed Oscillation".
19:16	WUPT[3:0]	0000	R	Sets the lower 4 bits of the 16 bits of calculation values of the warming-up timer. It is fixed to "0000".
15:9	-	0	R	Read as “0”.
8	WUCLK	0	R/W	Warming-up clock selection (Note1) 0: Internal high-speed oscillator1 (IHOSC1) 1: External high-speed oscillator (EHOSC)
7:2	-	0	R	Read as “0”.
1	WUEF	0	R	Indicates status of the warming-up timer. (Note2) 0: The end of warming-up 1: In warming-up operation
0	WUON	0	W	Control the warming-up timer. 0: Don't care 1: Warming-up operation start.

Note1: Use the internal oscillator for warming-up when the MCU returns from STOP1 mode. Do not use an external oscillator when the MCU returns from STOP1 mode.

Note2: Do not modify the registers during the warming-up (<WUEF> = 1). Set the registers when <WUEF> = 0.

1.4.2.7. *[CGFSYSMENA]* (Supply and Stop Register A for fsysm)

Bit	Bit symbol	After reset	Type	Function
31:19	-	0	R	Read as "0".
18	IPMENA18	0	R/W	Clock enable of PORT R 0: Clock stop 1: Clock supply
17	IPMENA17	0	R/W	Clock enable of PORT P 0: Clock stop 1: Clock supply
16	IPMENA16	0	R/W	Clock enable of PORT N 0: Clock stop 1: Clock supply
15	IPMENA15	0	R/W	Clock enable of PORT L 0: Clock stop 1: Clock supply
14	IPMENA14	0	R/W	Clock enable of PORT K 0: Clock stop 1: Clock supply
13	IPMENA13	0	R/W	Clock enable of PORT J 0: Clock stop 1: Clock supply

Bit	Bit symbol	After reset	Type	Function
12	IPMENA12	0	R/W	Clock enable of PORT H 0: Clock stop 1: Clock supply
11	IPMENA11	0	R/W	Clock enable of PORT G 0: Clock stop 1: Clock supply
10	IPMENA10	0	R/W	Clock enable of PORT F 0: Clock stop 1: Clock supply
9	IPMENA09	1	R/W	Clock enable of PORT E 0: Clock stop 1: Clock supply
8	IPMENA08	0	R/W	Clock enable of PORT D 0: Clock stop 1: Clock supply
7	IPMENA07	0	R/W	Clock enable of PORT C 0: Clock stop 1: Clock supply
6	IPMENA06	0	R/W	Clock enable of PORT B 0: Clock stop 1: Clock supply
5	IPMENA05	0	R/W	Clock enable of PORT A 0: Clock stop 1: Clock supply
4	IPMENA04	0	R/W	Clock enable of DMAC 0: Clock stop 1: Clock supply
3	IPMENA03	0	R/W	Clock enable of TRGSEL 0: Clock stop 1: Clock supply
2	IPMENA02	0	R/W	Clock enable of TRM 0: Clock stop 1: Clock supply
1	IPMENA01	0	R/W	Clock enable of OFD 0: Clock stop 1: Clock supply
0	IPMENA00	0	R/W	Clock enable of RAMP ch1 0: Clock stop 1: Clock supply

Note: Even if the initial value of the register is set to stop of the clock, the clock is supplied during the reset.

1.4.2.8. *[CGFSYSMENB]* (Supply and Stop Register B for fsysm)

Bit	Bit symbol	After reset	Type	Function
31	IPMENB31	1	R/W	Clock enable of SIWDT 0: Clock stop 1: Clock supply
30:29	-	0	R	Read as "0".
28	IPMENB28	1	R/W	Write as "1".
27:16	-	0	R	Read as “0”.
15	IPMENB15	0	R/W	Clock enable of T32A ch4 0: Clock stop 1: Clock supply
14	IPMENB14	0	R/W	Clock enable of T32A ch3 0: Clock stop 1: Clock supply
13	IPMENB13	0	R/W	Clock enable of T32A ch2 0: Clock stop 1: Clock supply
12	IPMENB12	0	R/W	Clock enable of T32A ch1 0: Clock stop 1: Clock supply
11	IPMENB11	1	R/W	Clock enable of T32A ch0 0: Clock stop 1: Clock supply
10	IPMENB10	0	R/W	Clock enable of EI2C ch1 0: Clock stop 1: Clock supply
9	IPMENB09	0	R/W	Clock enable of EI2C ch0 0: Clock stop 1: Clock supply
8	IPMENB08	0	R/W	Clock enable of UART ch4 0: Clock stop 1: Clock supply
7	IPMENB07	0	R/W	Clock enable of UART ch3 0: Clock stop 1: Clock supply
6	IPMENB06	0	R/W	Clock enable of UART ch2 0: Clock stop 1: Clock supply
5	IPMENB05	0	R/W	Clock enable of UART ch1 0: Clock stop 1: Clock supply
4	IPMENB04	1	R/W	Clock enable of UART ch0 0: Clock stop 1: Clock supply
3	IPMENB03	0	R/W	Clock enable of TSPI ch3 0: Clock stop 1: Clock supply

Bit	Bit symbol	After reset	Type	Function
2	IPMENB02	0	R/W	Clock enable of TSPI ch2 0: Clock stop 1: Clock supply
1	IPMENB01	0	R/W	Clock enable of TSPI ch1 0: Clock stop 1: Clock supply
0	IPMENB00	0	R/W	Clock enable of TSPI ch0 0: Clock stop 1: Clock supply

Note: Even if the initial value of the register is set to stop of the clock, the clock is supplied during the reset.

1.4.2.9. *[CGFSYSENA]* (Supply and Stop Register A for fsysh)

Bit	Bit symbol	After reset	Type	Function
31:8	-	0	R	Read as “0”.
7	IPENA07	0	R/W	Clock enable of A-PMD ch1 0: Clock stop 1: Clock supply
6	IPENA06	0	R/W	Clock enable of A-PMD ch0 0: Clock stop 1: Clock supply
5	IPENA05	0	R/W	Clock enable of A-ENC32 ch1 0: Clock stop 1: Clock supply
4	IPENA04	0	R/W	Clock enable of A-ENC32 ch0 0: Clock stop 1: Clock supply
3	IPENA03	0	R/W	Clock enable of ADC Unit B 0: Clock stop 1: Clock supply
2	IPENA02	0	R/W	Clock enable of ADC Unit A 0: Clock stop 1: Clock supply
1	IPENA01	0	R/W	Clock enable of RAMP ch0 0: Clock stop 1: Clock supply
0	IPENA00	0	R/W	Clock enable of CRC 0: Clock stop 1: Clock supply

Note: Even if the initial value of the register is set to stop of the clock, the clock is supplied during the reset.

1.4.2.10. *[CGFCEN]* (Clock Supply and Stop Register for fc)

Bit	Bit symbol	After reset	Type	Function
31:25	-	0	R	Read as “0”.
24	FCIPEN24	0	R/W	Clock enable of DNF (INT0 to F) 0: Clock stop 1: Clock supply
23:21	-	0	R	Read as “0”.
20	FCIPEN20	0	R/W	Clock enable of OFD detection target clock 1 (fc) 0: Clock stop 1: Clock supply
19:0	-	0	R	Read as “0”.

Note: Even if the initial value of the register is set to stop of the clock, the clock is supplied during the reset.

1.4.2.11. *[CGSPCLKEN]* (Clock Supply and Stop Register for ADC and Debug Circuit)

Bit	Bit symbol	After reset	Type	Function
31:20	-	0	R	Read as “0”.
19:18	-	0	R/W	Write as "0".
17	ADCKEN1	0	R/W	Enable the clock for ADC Unit B (Note2) 0: Clock stop 1: Clock supply
16	ADCKEN0	0	R/W	Enable the clock for ADC Unit A (Note2) 0: Clock stop 1: Clock supply
15:1	-	0	R	Read as “0”.
0	TRCKEN	0	R/W	Enable the Clock for the Trace function of Debug circuit (Trace or SWV). 0: Clock stop 1: Clock supply

Note1: Even if the initial value of the register is set to stop of the clock, the clock is supplied during the reset.

Note2: When setting this bit to "0" (clock stop), please make sure that AD conversion is stopped.

2. Memory Map

2.1. Outlines

The memory map for TMPM471F10FG is based on the Arm Cortex-M4 (with FPU) processor core memory map.

The internal ROM, internal RAM and special function registers (SFR) of TMPM471F10FG are mapped to the Code, SRAM and peripheral regions of the Cortex-M4 (with FPU) respectively. The special function register (SFR) means the control registers of all input/output ports and peripheral functions.

The CPU register region is the processor core’s internal register region.

For more information on each region, see the "Arm documentation set for the Arm Cortex-M4".

Note that access to regions indicated as "Fault" causes a bus fault if bus faults are enabled, or causes a hard fault if bus faults are disabled. Also, do not access the vendor-specific region.

2.1.1. TMPM471F10FG

- Code Flash: 1MB
- RAM: 64KB

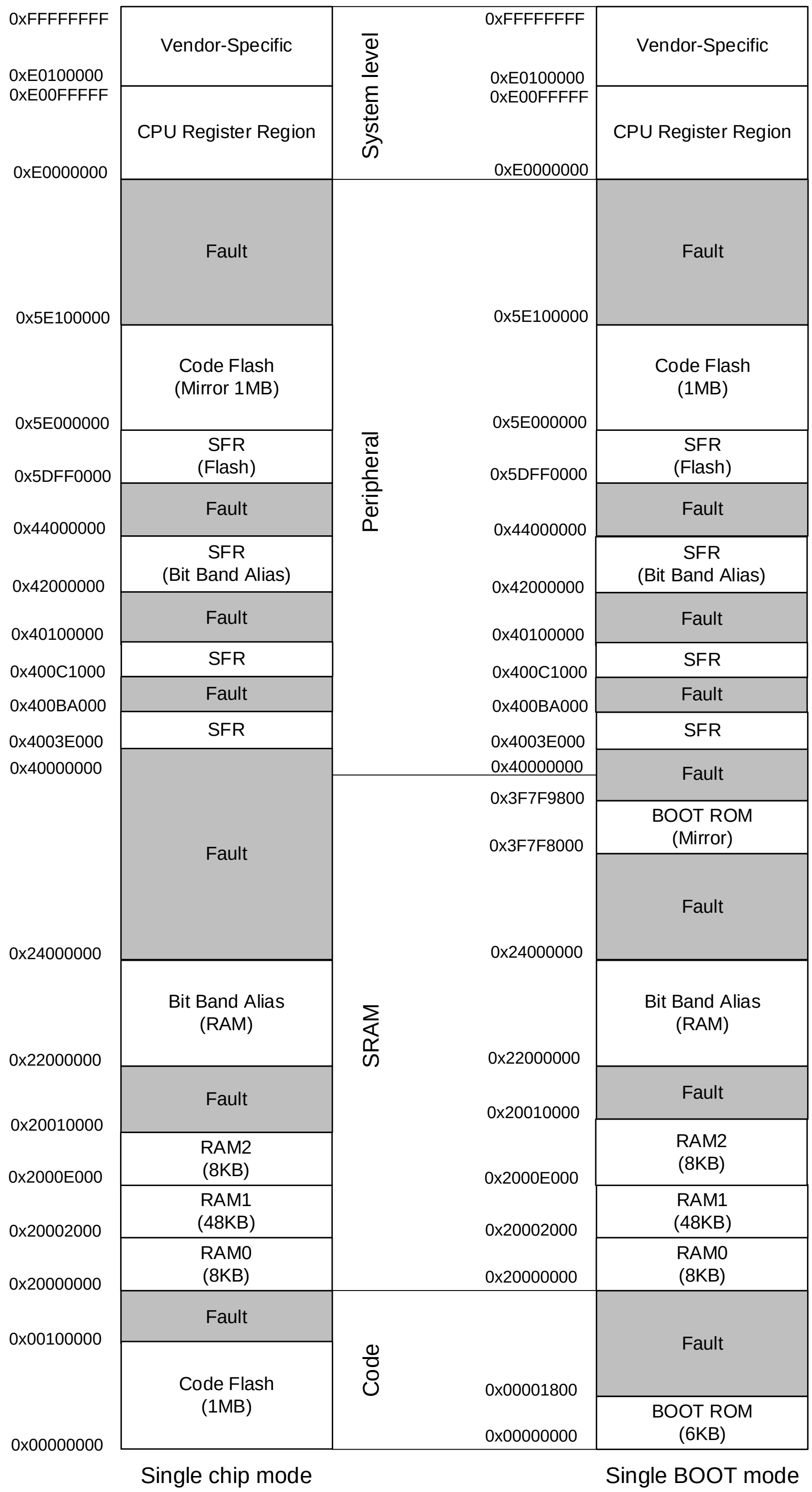

Figure 2.1 TMPM471F10FG

2.2. Bus Matrix

The Bus Matrix of TMPM471F10FG consists of a high-speed system clock domain and a middle-speed system clock domain.

In the high-speed system clock domain, a CPU core connected to the subordinate ports (S1 to S3) accesses peripheral functions connected to the manager ports (M0 to M8) via symbols (○) or (●) indicating the connection. (●) shows a connection to a mirror area.

In the middle-speed system clock domain, a DMA controller (DMAC) connected to the subordinate port (SS1) accesses peripheral functions connected to the manager ports (SM0 to SM6) via symbols (○) indicating the connection.

If multiple bus managers access the same manager port simultaneously, the bus manager connected to the subordinate port with the lowest number has priority.

2.2.1. Structure

2.2.1.1. Single Chip Mode

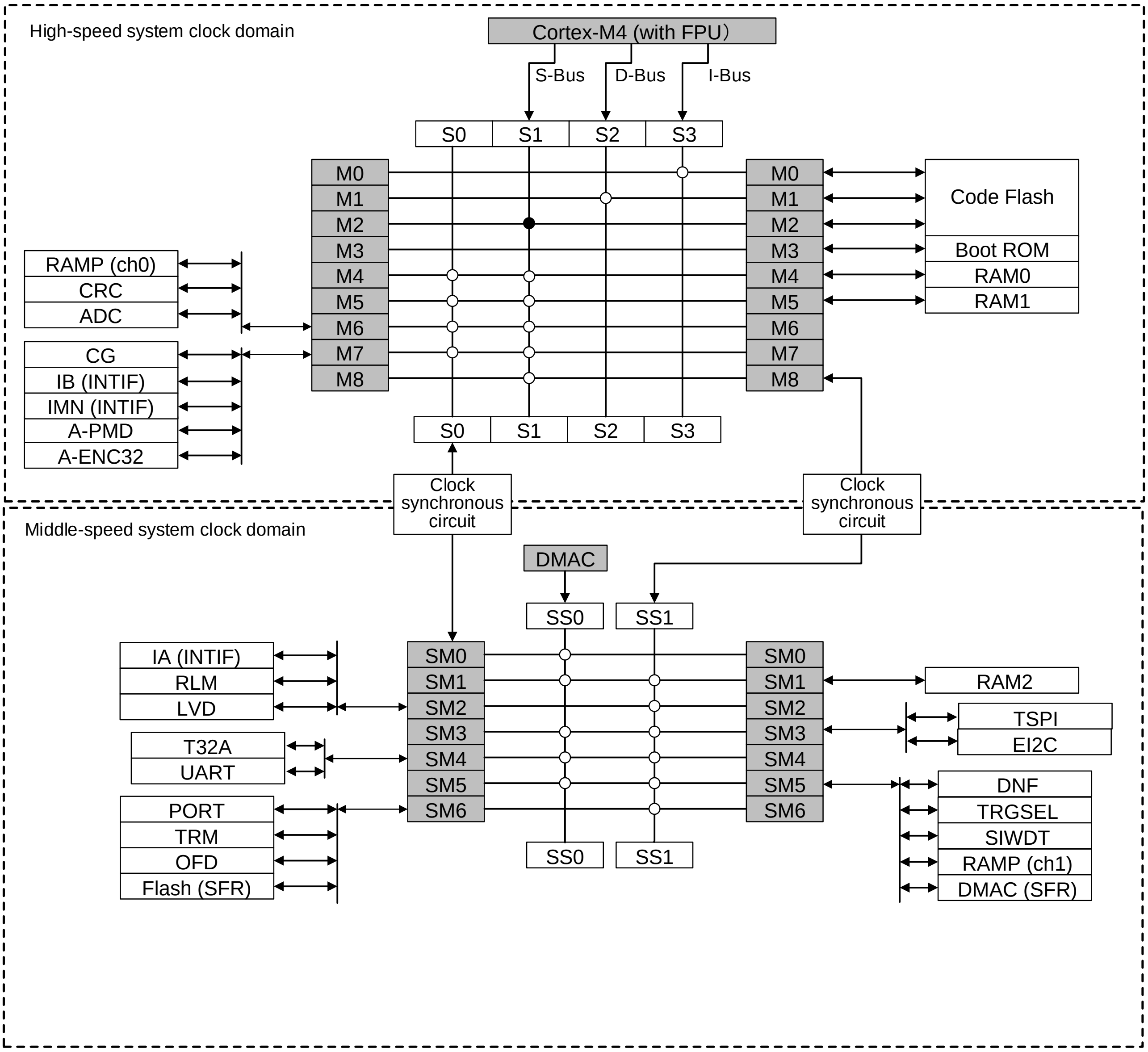

Figure 2.2 Single Chip Mode

2.2.1.2. Single Boot Mode

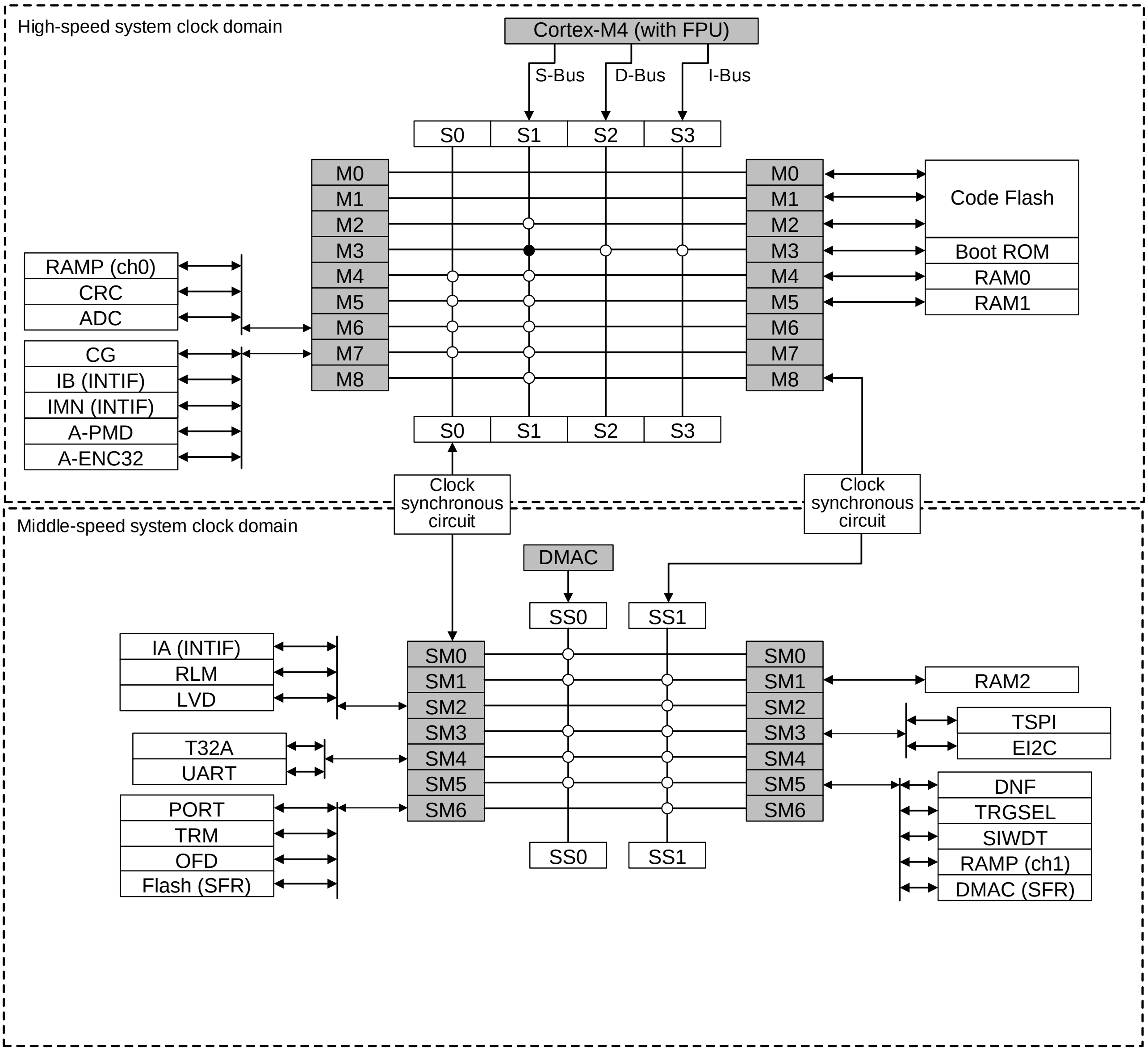

Figure 2.3 Single Boot Mode

2.2.2. Connection Table

2.2.2.1. Connection of Memory Related

(1) TMPM471F10FG

- Single chip mode

Table 2.1 Single Chip Mode

Start address	Subordinate		Sub manager	Main manager		
			DMAC	Core S-Bus	Core D-Bus	Core I-Bus
			SS0	S1	S2	S3
0x00000000	Code Flash	M0	Fault	-	-	✓
		M1	Fault	-	✓	-
0x20000000	RAM0	M4	✓	✓	-	-
0x20002000	RAM1	M5	✓	✓	-	-
0x2000E000	RAM2	SM1	✓	✓	-	-
0x5E000000	Code Flash (Mirror)	M2	Fault	✓	-	-

✓: Accessible, -: not accessible, Fault: Fault is caused

- Single boot mode

Table 2.2 Single Boot Mode

Start address	Subordinate		Sub manager	Main manager		
			DMAC	Core S-Bus	Core D-Bus	Core I-Bus
			SS0	S1	S2	S3
0x00000000	Boot ROM	M3	Fault	-	✓	✓
0x20000000	RAM0	M4	✓	✓	-	-
0x20002000	RAM1	M5	✓	✓	-	-
0x2000E000	RAM2	SM1	✓	✓	-	-
0x3F7F8000	Boot ROM (Mirror)	M3	Fault	✓	-	-
0x5E000000	Code Flash	M2	Fault	✓	-	-

✓: Accessible, -: not accessible, Fault: Fault is caused

2.2.2.2. Connection of Peripheral Function

Table 2.3 Connection of Peripheral Function

Start address	Subordinate		Sub manager	Main manager		
			DMAC	Core S-Bus	Core D-Bus	Core I-Bus
			SS0	S1	S2	S3
0x4003E000	IA (INTIF)	SM2	Fault	✓	-	-
0x4003E400	RLM		Fault	✓	-	-
0x4003EC00	LVD		Fault	✓	-	-
0x40043000	RAMP (ch0)	M6	✓	✓	-	-
0x40043100	CRC		✓	✓	-	-
0x4005A000	ADC		✓	✓	-	-
0x40083000	CG	M7	✓	✓	-	-
0x40083200	IB (INTIF)		✓	✓	-	-
0x40083300	IMN (INTIF)		✓	✓	-	-
0x40089000	A-PMD		✓	✓	-	-
0x4008A000	A-ENC32		✓	✓	-	-
0x400A0200	DNF	SM5	✓	✓	-	-
0x400A0400	TRGSEL		✓	✓	-	-
0x400A0600	SIWDT		✓	✓	-	-
0x400A3000	RAMP (ch1)		✓	✓	-	-
0x400A4000	DMAC (SFR)		✓	✓	-	-
0x400C1000	T32A	SM4	✓	✓	-	-
0x400CA000	TSPI	SM3	✓	✓	-	-
0x400CE000	UART	SM4	✓	✓	-	-
0x400D1000	EI2C	SM3	✓	✓	-	-
0x400E0000	PORT	SM6	Fault	✓	-	-
0x400E3100	TRM		Fault	✓	-	-
0x400E4000	OFD		Fault	✓	-	-
0x5DFF0000	Flash (SFR)		Fault	✓	-	-

✓: Accessible, -: not accessible, Fault: Fault is caused

3. Reset and Power Supply Control

3.1. Outlines

Function classification	Factor	Functional description
Cold reset (Reset by turning on a power supply)	Power on reset	Reset which occurs at the time of a power supply turning on or turning off.
	LVD reset	Reset which occurs below the set-up voltage
	Reset pin	Reset by a RESET_N pin
	PORF reset	Reset that occurs when power is turned on or off and flash memory and debug circuits are reset with priority.
Warm reset (Reset without turning on a power supply)	Internal reset	Reset by SIWDT, OFD, LVD, LOCKUP, and <SYSRESETREQ>
	Reset pin	Reset by a RESET_N pin
Single Boot starting	Reset pin	After reset is released, it starts from the internal boot ROM.

3.2. Description of Function and Operation

This chapter describes the power-on, power-off, and reset related items.

Note: Refer to "Electrical Characteristics" of a datasheet for the time and voltage of description of the symbol in a figure.

3.2.1. Cold Reset

When turn on a power supply, the stabilization time for the built-in regulator, the built-in flash memory, and the built-in high-speed oscillator is necessary. The TMPM471F10FG automatically inserts a wait time for the stabilization of these circuits.

When turning on the power, make sure that the slope of the power supply voltage rises to the right. If the power supply voltage drops and rises near POR and PORF detection voltage, it may not operate normally even if the power supply voltage rises to the guaranteed operating range thereafter.

3.2.1.1. Reset by Power On Reset Circuit (without Using RESET_N Pin)

After a supply voltage exceeds the release voltage of a power on reset (POR), internal reset is deasserted after "Internal initialization time" is elapsed. Please increase a supply voltage goes up into an operating voltage range before "Internal initialization time" is elapsed. The CPU operates after internal reset is released.

After a supply voltage exceeds the release voltage of a power on reset (POR), LVD continues to output reset signal until supply voltage exceeds the LVD release voltage.
And internal reset has priority during the time of "Internal initialization time". When rising time of a supply voltage beyond "Internal initialization time", please refer to "3.2.1.3. Continuation of Reset by LVD".

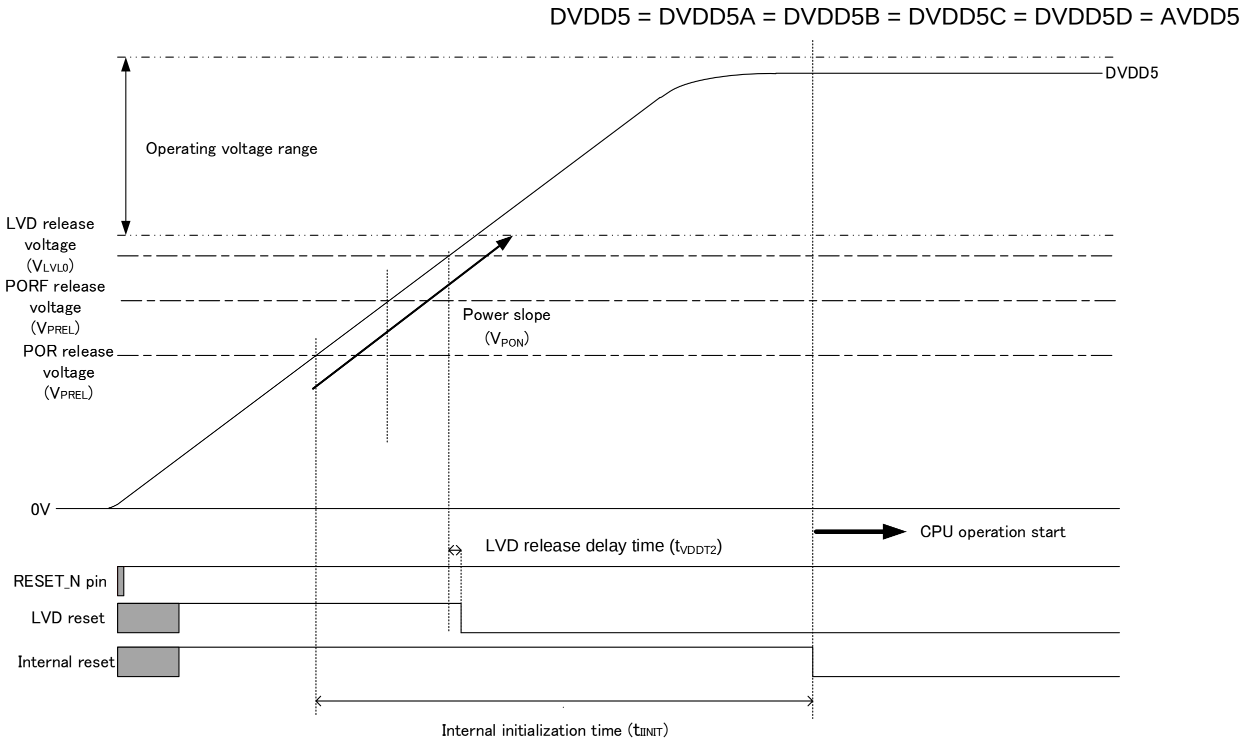

Figure 3.1 Reset Operation by Power On Reset Circuit

Note: When you use only a power on reset circuit without RESET_N pin, the RESET_N pin should input "High" level or opened.

3.2.1.2. Reset by RESET_N Pin

When turn on a power supply, it can control the timing of reset release by using RESET_N pin.

After a supply voltage exceeds the release voltage of a power on reset and even after "Internal initialization time" elapsed, if the RESET_N pin is "Low", internal reset continues.
After a supply voltage goes up into an operating voltage range, if a RESET_N pin becomes "High", Internal reset is deasserted after "CPU operation wait time" elapses.

DVDD5 = DVDD5A = DVDD5B = DVDD5C = DVDD5D = AVDD5

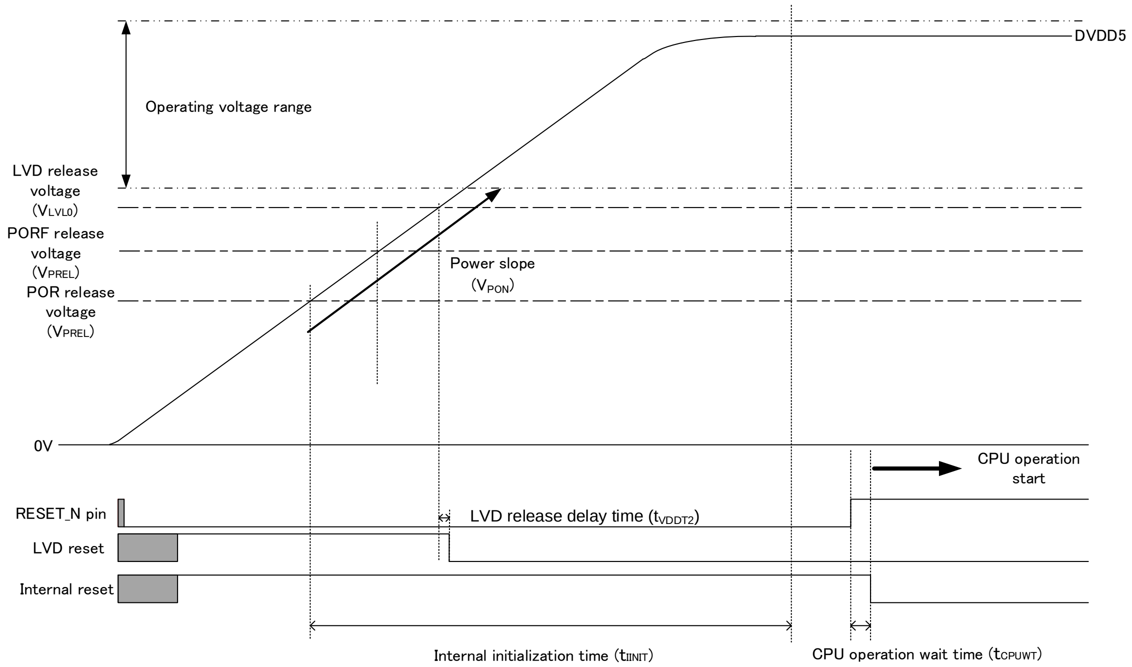

Figure 3.2 Reset Operation by RESET_N Pin (1)

In case of RESET_N pin input change from "Low" to "High" before "Internal initialization time" elapses, internal reset signal is released after "Internal initialization time" elapses.

Please goes up a supply voltage into an operating voltage range before "Internal initialization time" elapses. The CPU operates after internal reset release.

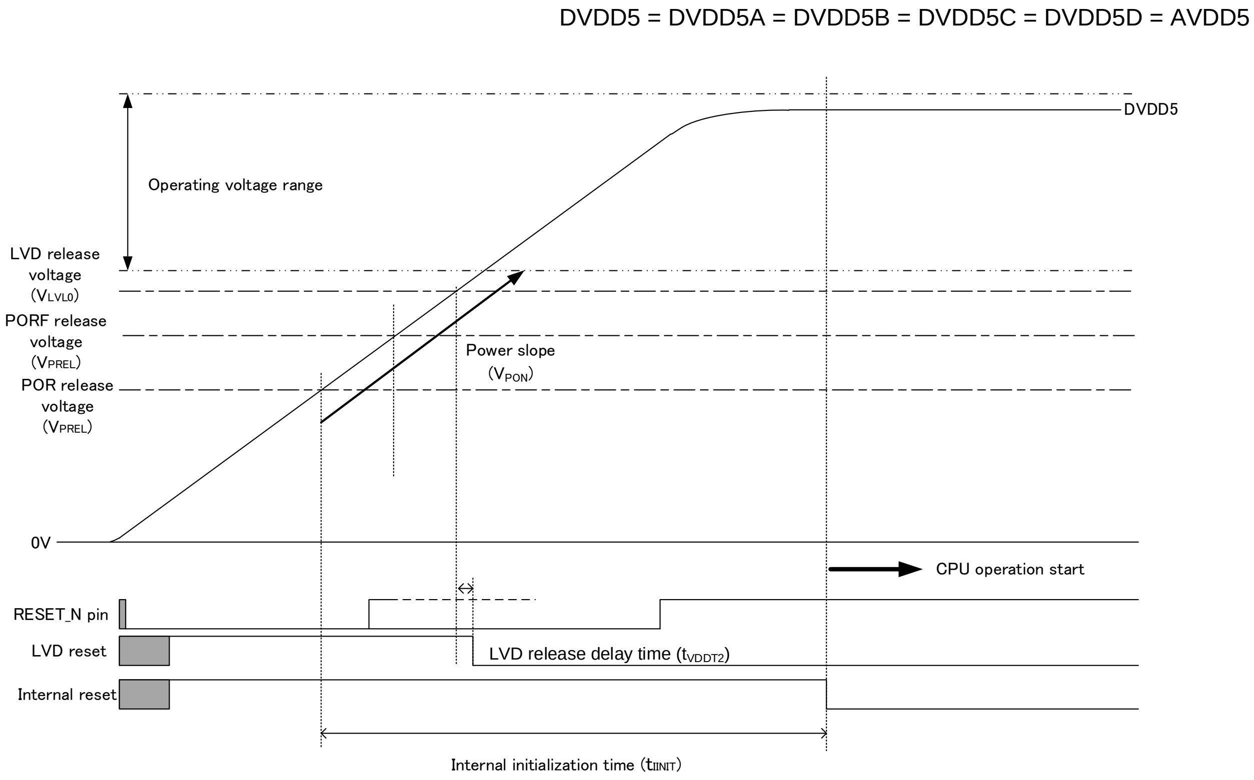

Figure 3.3　Reset Operation by RESET_N Pin (2)

3.2.1.3. Continuation of Reset by LVD

When the power supply voltage has not exceeded the LVD release voltage even after "Internal initialization time" elapsed, LVD generates the reset signal and the reset state continues. After the power supply voltage exceeds the LVD release voltage and "LVD detection release time" + "CPU operation wait time" elapses, the internal reset is released. And CPU starts operating. Refer to the reference manual "Voltage Detection Circuit" for details of LVD.

DVDD5 = DVDD5A = DVDD5B = DVDD5C = DVDD5D = AVDD5

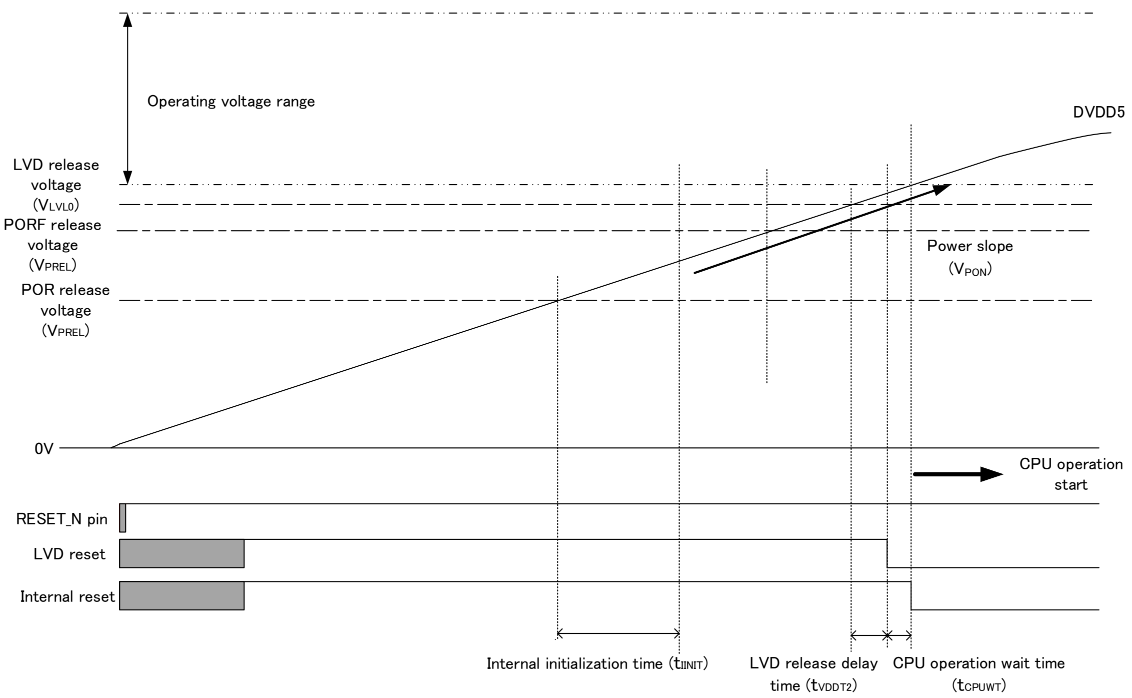

Figure 3.4 Reset Operation by LVD Reset

3.2.2. Warm Reset

3.2.2.1. Warm Reset by RESET_N Pin

When resetting with the RESET_N pin, set the RESET_N pin to "Low" for 17.2 µs or more while the power supply voltage is within the operating range.

When the "Low" period of a RESET_N pin is longer than "Internal processing time", after a RESET_N pin changes to "High", Internal reset is released after "CPU operation wait time" elapsed.

When the "Low" period of a RESET_N pin is shorter than "Internal processing time", after internal reset is extended and from a RESET_N pin changes to "Low", Internal reset is released after "Internal processing time" + "CPU operation wait time" has elapsed.

DVDD5 = DVDD5A = DVDD5B = DVDD5C = DVDD5D = AVDD5

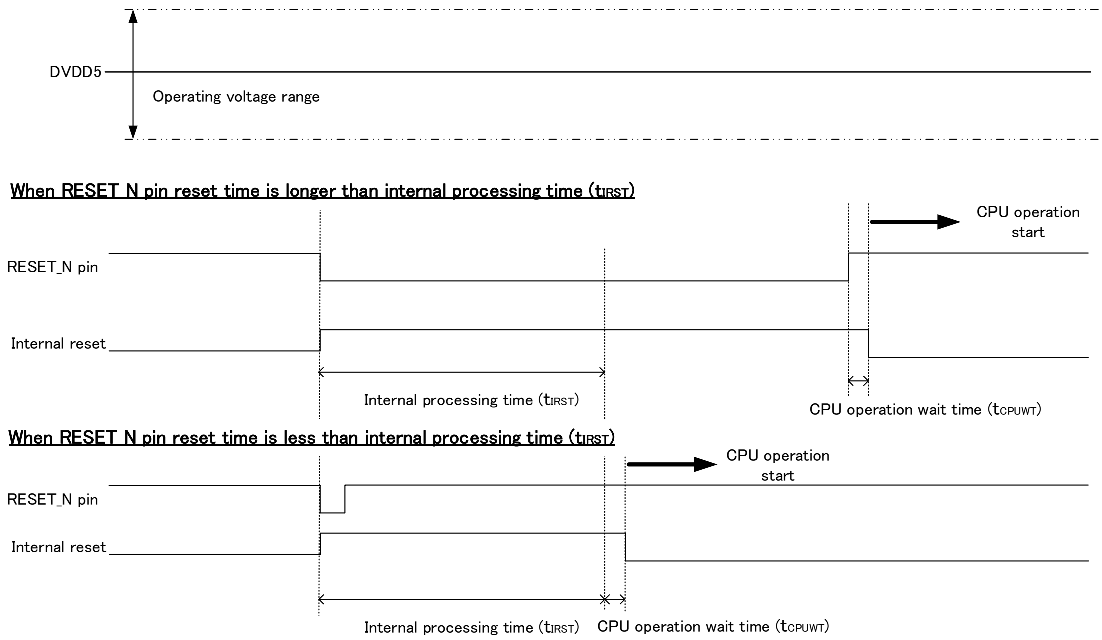

Figure 3.5 Warm Reset Operation

3.2.2.2. Warm Reset by LVD

LVD reset is performed correctly when the LVD reset voltage or less and the power supply voltage is within the operating voltage range. When the power supply voltage drop period is longer than the "internal processing time", internal reset is released after "internal processing time" has elapsed, LVD release voltage has been exceeded, and "LVD detection release time" + "CPU operation wait time" has elapsed. When the power supply voltage drop period is shorter than the "internal processing time", internal reset is released after "internal processing time" + "CPU operation wait time" has elapsed from LVD reset is detected.

3.2.2.3. Warm Reset by Other Internal Reset

In case of reset asserted by internal factors, such as SIWDT, OFD, LOCKUP, and <SYSRESETREQ>, Internal reset is released after "Internal processing time" + "CPU operation wait time" elapsed.

3.2.3. Starting in Single Boot Mode

Refer to the reference manual "Flash Memory" for details of "Single Boot Mode".

3.2.3.1. Starting by RESET_N Pin

When "Low" is inputted to a BOOT_N pin, if reset is released (a RESET_N pin "Low" to "High"), "Single Boot Mode" is started.

When turn on power supply, input "Low" to the RESET_N pin longer than "Internal initialization time" to reset. And release reset, after a supply voltage goes up into an operating voltage range.

DVDD5 = DVDD5A = DVDD5B = DVDD5C = DVDD5D = AVDD5

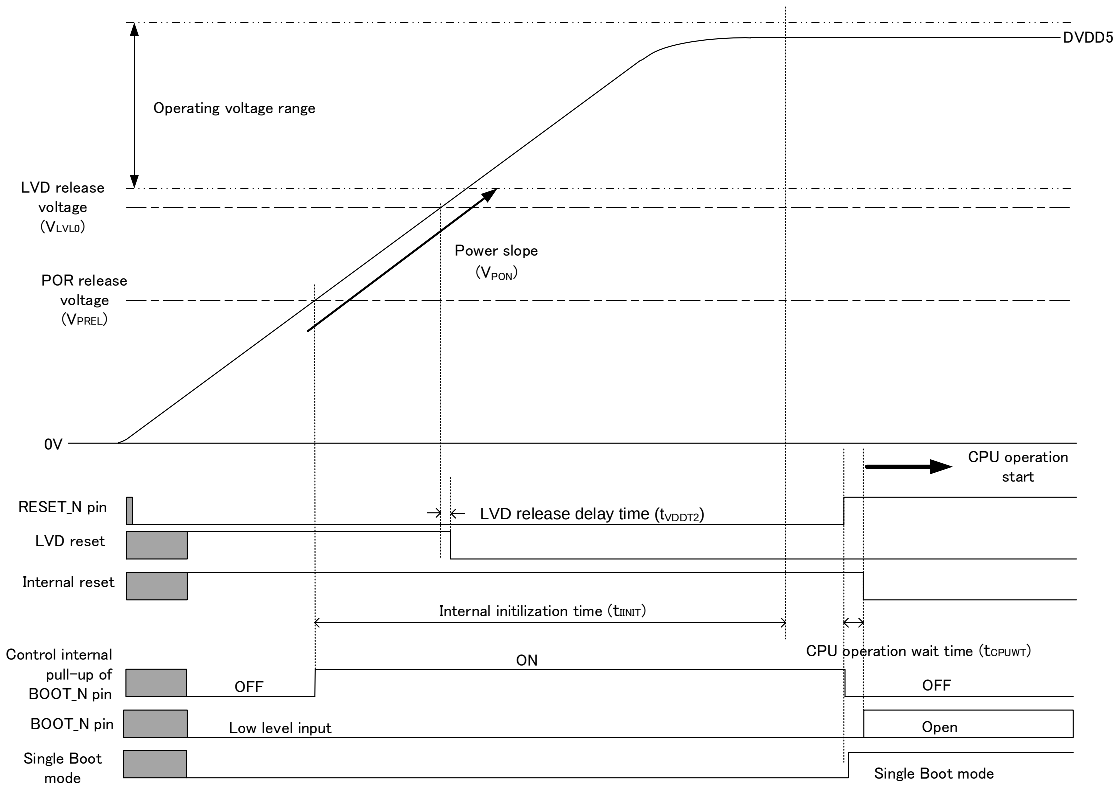

Figure 3.6 When Power Supply is On, Starting in Single Boot Mode by RESET_N Pin

3.2.3.2. Starting in Single Boot Mode when Power Supply is Stable

When the supply voltage is stable within an operating voltage range, input "Low" to RESET_N pin for reset longer than "Internal processing time", while "Low" is inputted to the BOOT_N pin. And release reset (RESET_N pin to "High").

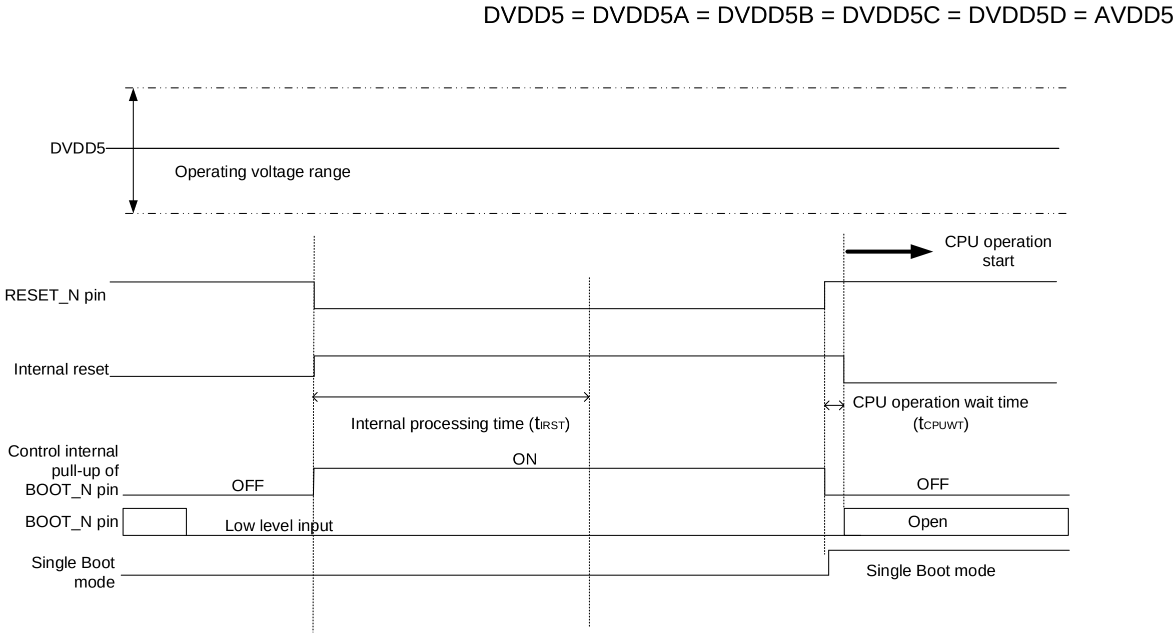

Figure 3.7 Starting in Single Boot Mode when Power Supply is Stable

3.2.4. Power On Reset Circuit

The power on reset circuit (POR) generates a reset signal when the power is turned on or turned off.

Note: The power on reset circuit may not operate correctly due to the fluctuation of the power supply. Equipment should be designed with full consideration of the electrical characteristics.

The power on reset circuit consists of a detection voltage generation circuit, a reference voltage generation circuit, and a comparator.

The supply voltage has referred to DVDD5 (= DVDD5A = DVDD5B = DVDD5C = DVDD5D).

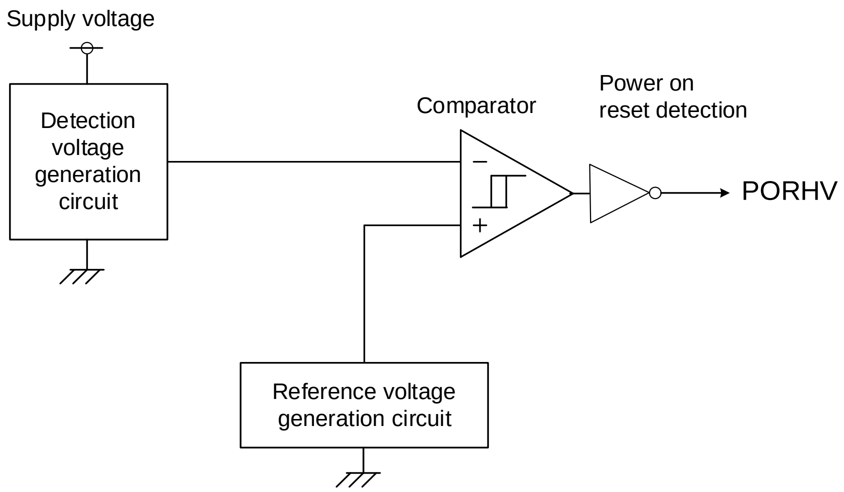

Figure 3.8 Power On Reset Circuit

3.2.4.1. Operation at Time of Power Supply

When turn on power supply, while the power supply voltage is lower than power on reset circuit release voltage (V_{PREL}), the power on reset detection signal is generated. Refer to "Figure 3.1 Reset Operation by Power On Reset Circuit" for details.

While the power on reset signal is generated, the reset is asserted to the CPU and the peripherals.

3.2.4.2. Operation at Time of Turn Off

When turn off power supply or when the power supply voltage is lower than power on reset detection voltage (V_{PDET}), the power on reset detection signal is generated.

While the power on reset signal is generated, the reset is asserted to the CPU and the peripherals.

3.2.5. Turning Off and Re-turning on Power Supply

When a power supply is turned off, a power supply voltage must be down gentler gradient than Max value of "Power gradient (V_{POFF})" specified in "Electrical Characteristics".

3.2.5.1. When Using External Reset Circuit or Internal LVD Reset Output

When the power supply is turned off and the power supply voltage drops below the operation guaranteed voltage, reset is performed with an external reset circuit or built-in LVD (when the voltage is less than the set voltage). After that, from the state where the reset is applied, please follow the same constraints as when turning on the power and turned on the power supply voltage.

3.2.5.2. When not Using External Reset Circuit and Internal LVD Reset Output

When the power supply is turned off and the power supply voltage drops below the operation guaranteed voltage, be sure to lower the power supply voltage below the power on reset detection voltage (VPDET) and hold it for 200 µs or more. After that, please follow the same constraints as when turning on the power and turned on the power supply voltage.

When the power supply voltage drops below the power on reset detection voltage (VPDET) and cannot be held for 200 µs or more, or when the same constraints as at power on cannot keep, the MCU may not operate properly.

3.2.6. After Reset Release

All of the control register of the Cortex-M4 processor with FPU and the peripheral function control register (SFR) are initialized by reset. But depend on the reset factor, initialized range is different.

Please refer to "Table 3.1 Reset Factor and Range Initialized" for the initialized range by each reset factor.

The reset factor when reset occurs can be checked by a reset flag register which are ***[RLMRSTFLG0]*** and ***[RLMRSTFLG1]***. For details of ***[RLMRSTFLG0]*** and ***[RLMRSTFLG1]***, please refer to the reference manual "Exception".

After reset is released, CPU starts operation by a clock of Internal High Speed Oscillator1 (IHOSC1). External clock and PLL multiple circuit should be set if necessary.

3.2.6.1. Reset Factor and Reset Range

Reset factors and the range initialized are shown in Table 3.1.

Table 3.1 Reset Factor and Range Initialized

Registers and peripheral function		Reset factors							
		Cold reset	Warm reset						
		POR (Note)	Reset pin	OFD reset	SIWDT reset	LVD reset	CPU <SYS RESET REQ> reset	CPU LOCKUP reset	PORF reset
	Reset signal name	PORHV	RESET_N	OFD RSTOUT	SIWDT RSTOUT	LVD RSTOUT	SYS RESET REQ	LOCKUP RESET REQ	PORF RESET
Reset flag	*[RLMRSTFLG0]* *[RLMRSTFLG1]*	✓	-	-	-	-	-	-	-
Interrupt control	*[IAIMCxx]* *[IANIC00]*	✓	✓	✓	✓	✓	✓	✓	✓
	[IBIMCxxx] *[IBNIC00]*	✓	✓	✓	✓	✓	✓	✓	✓
Flash	*[FCSBMR]*	✓	-	-	-	-	-	-	✓
PORT	All the registers	✓	✓	✓	✓	✓	✓	✓	✓
OFD		✓	✓	✓	✓	✓	✓	✓	✓
LVD		✓	✓	-	-	-	-	-	-
Debugging interface		✓	-	-	-	-	-	-	✓
Except above		✓	✓	✓	✓	✓	✓	✓	✓

✓: It is initialized.
-: It is not initialized.

Note: When reset is performed, the data of built-in RAM will not be guaranteed.

4. Revision History

Table 4.1 Revision History

Revision	Date	Description
1.0	2024-08-30	- First release
1.1	2025-12-15	- 1.2.5.2. Formula and Example of Setting of PLL Multiplication Value Changed table 1.3 and table 1.4

RESTRICTIONS ON PRODUCT USE

Toshiba Corporation and its subsidiaries and affiliates are collectively referred to as “TOSHIBA”.
Hardware, software and systems described in this document are collectively referred to as “Product”.

- TOSHIBA reserves the right to make changes to the information in this document and related Product without notice.
- This document and any information herein may not be reproduced without prior written permission from TOSHIBA. Even with TOSHIBA's written permission, reproduction is permissible only if reproduction is without alteration/omission.
- Though TOSHIBA works continually to improve Product's quality and reliability, Product can malfunction or fail. Customers are responsible for complying with safety standards and for providing adequate designs and safeguards for their hardware, software and systems which minimize risk and avoid situations in which a malfunction or failure of Product could cause loss of human life, bodily injury or damage to property, including data loss or corruption. Before customers use the Product, create designs including the Product, or incorporate the Product into their own applications, customers must also refer to and comply with (a) the latest versions of all relevant TOSHIBA information, including without limitation, this document, the specifications, the data sheets and application notes for Product and the precautions and conditions set forth in the "TOSHIBA Semiconductor Reliability Handbook" and (b) the instructions for the application with which the Product will be used with or for. Customers are solely responsible for all aspects of their own product design or applications, including but not limited to (a) determining the appropriateness of the use of this Product in such design or applications; (b) evaluating and determining the applicability of any information contained in this document, or in charts, diagrams, programs, algorithms, sample application circuits, or any other referenced documents; and (c) validating all operating parameters for such designs and applications. **TOSHIBA ASSUMES NO LIABILITY FOR CUSTOMERS' PRODUCT DESIGN OR APPLICATIONS.**
- **PRODUCT IS NEITHER INTENDED NOR WARRANTED FOR USE IN EQUIPMENTS OR SYSTEMS THAT REQUIRE EXTRAORDINARILY HIGH LEVELS OF QUALITY AND/OR RELIABILITY, AND/OR A MALFUNCTION OR FAILURE OF WHICH MAY CAUSE LOSS OF HUMAN LIFE, BODILY INJURY, SERIOUS PROPERTY DAMAGE AND/OR SERIOUS PUBLIC IMPACT ("UNINTENDED USE").** Except for specific applications as expressly stated in this document, Unintended Use includes, without limitation, equipment used in nuclear facilities, equipment used in the aerospace industry, lifesaving and/or life supporting medical equipment, equipment used for automobiles, trains, ships and other transportation, traffic signaling equipment, equipment used to control combustions or explosions, safety devices, elevators and escalators, and devices related to power plant. **IF YOU USE PRODUCT FOR UNINTENDED USE, TOSHIBA ASSUMES NO LIABILITY FOR PRODUCT.** For details, please contact your TOSHIBA sales representative or contact us via our website.
- Do not disassemble, analyze, reverse-engineer, alter, modify, translate or copy Product, whether in whole or in part.
- Product shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable laws or regulations.
- The information contained herein is presented only as guidance for Product use. No responsibility is assumed by TOSHIBA for any infringement of patents or any other intellectual property rights of third parties that may result from the use of Product. No license to any intellectual property right is granted by this document, whether express or implied, by estoppel or otherwise.
- **ABSENT A WRITTEN SIGNED AGREEMENT, EXCEPT AS PROVIDED IN THE RELEVANT TERMS AND CONDITIONS OF SALE FOR PRODUCT, AND TO THE MAXIMUM EXTENT ALLOWABLE BY LAW, TOSHIBA (1) ASSUMES NO LIABILITY WHATSOEVER, INCLUDING WITHOUT LIMITATION, INDIRECT, CONSEQUENTIAL, SPECIAL, OR INCIDENTAL DAMAGES OR LOSS, INCLUDING WITHOUT LIMITATION, LOSS OF PROFITS, LOSS OF OPPORTUNITIES, BUSINESS INTERRUPTION AND LOSS OF DATA, AND (2) DISCLAIMS ANY AND ALL EXPRESS OR IMPLIED WARRANTIES AND CONDITIONS RELATED TO SALE, USE OF PRODUCT, OR INFORMATION, INCLUDING WARRANTIES OR CONDITIONS OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, ACCURACY OF INFORMATION, OR NONINFRINGEMENT.**
- Do not use or otherwise make available Product or related software or technology for any military purposes, including without limitation, for the design, development, use, stockpiling or manufacturing of nuclear, chemical, or biological weapons or missile technology products (mass destruction weapons). Product and related software and technology may be controlled under the applicable export laws and regulations including, without limitation, the Japanese Foreign Exchange and Foreign Trade Law and the U.S. Export Administration Regulations. Export and re-export of Product or related software or technology are strictly prohibited except in compliance with all applicable export laws and regulations.
- Please contact your TOSHIBA sales representative for details as to environmental matters such as the RoHS compatibility of Product. Please use Product in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. **TOSHIBA ASSUMES NO LIABILITY FOR DAMAGES OR LOSSES OCCURRING AS A RESULT OF NONCOMPLIANCE WITH APPLICABLE LAWS AND REGULATIONS.**

Toshiba Electronic Devices & Storage Corporation
https://toshiba.semicon-storage.com/