RX FAMILY
HARDWARE MANUAL GUIDE
(PERIPHERAL FUNCTIONS)

2025/01 REV 1.0
RENESAS ELECTRONICS

R01TU0444EJ0100 rev1.0
Jan. 2025

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.

(Note 1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.

(Note 2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

(Rev. 5.0-1 October 2020)

PURPOSE OF THIS DOCUMENT

- This manual is an easy-to-understand summary of each functional chapter of the hardware manual.
- The purpose of this document is to provide much more deeper understanding of the functions by using it with the hardware manual.
- For more detailed usage instructions, please refer to the application notes provided on each page.
- This document does not cover all the information described in the manual. For detailed information such as precautions for each function, please check the hardware manual for the product you want to use.

LIST OF PERIPHERAL FUNCTIONS

This document explains the following peripheral functions. The referenced function name is in parentheses.
The peripheral functions described in this material will continue to be added.

Click here for the Hardware Manual Guide Electrical Characteristics

I/O PORT

This chapter is created with reference to RX66T. However, it can be used as a reference for all RX family products.

LIST OF PORT FUNCTIONS

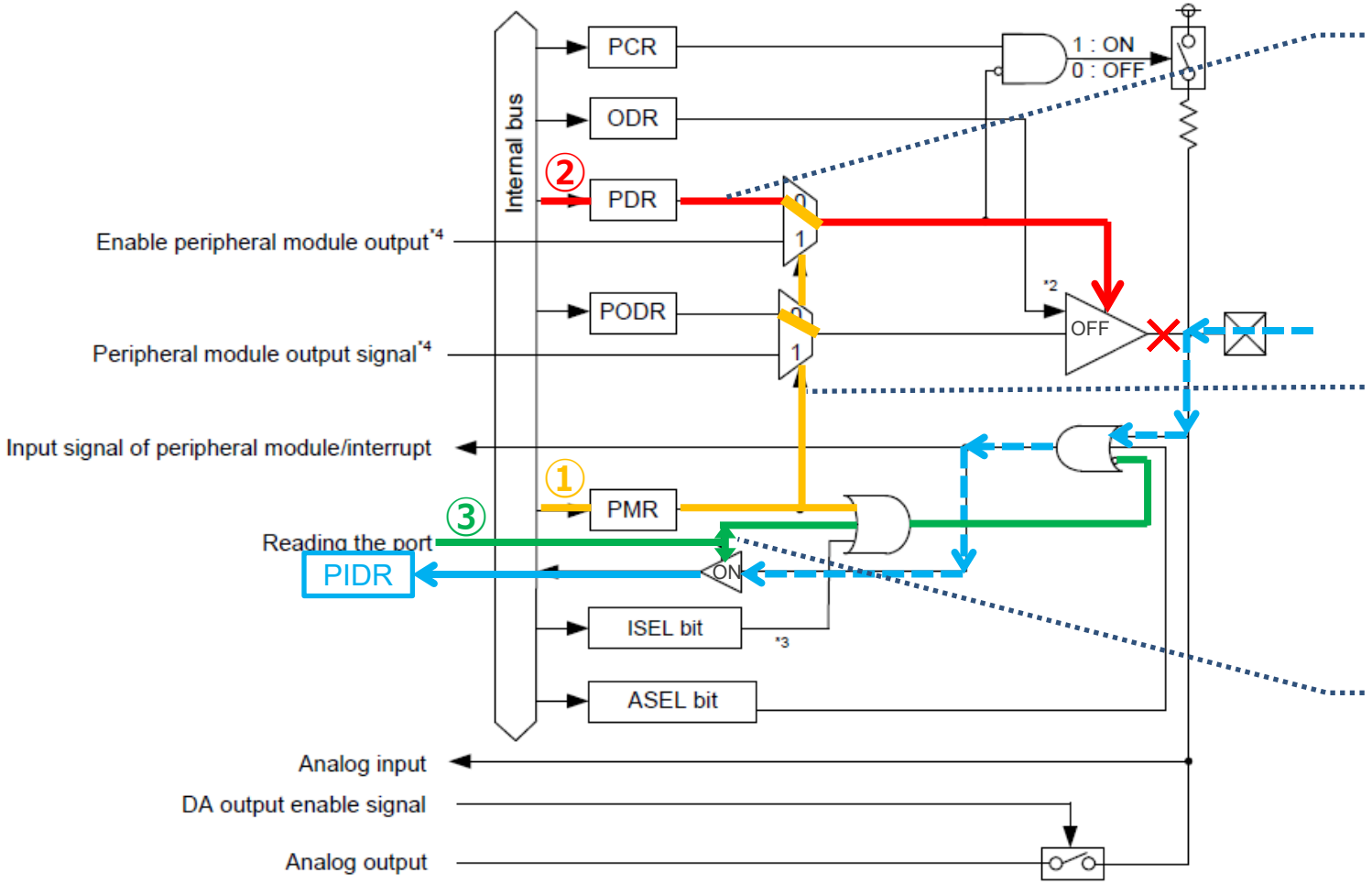
Port	Pin	Input Pull-up	Open-Drain Output	Driving Ability Switching	5-V Tolerant
PORT0	P00, P01	✓	✓	Normal drive/high drive	—
PORT1	P10, P11	✓	✓	Normal drive/high drive	—
PORT2	P20 to P24, P27	✓	✓	Normal drive/high drive	—
PORT3	P30 to P33	✓	✓	Normal drive/high drive	—
	P36, P37	✓	✓	Fixed to normal output	—
PORT4	P40 to P47	✓	✓	Fixed to normal output	—
PORT5	P50 to P55	✓	✓	Fixed to normal output	—
PORT6	P60 to P65	✓	✓	Fixed to normal output	—
PORT7	P70	✓	✓	Normal drive/high drive	—
	P71 to P76	✓	✓	Normal drive/high drive/ large current output	—
PORT8	P80, P82	✓	✓	Normal drive/high drive	—
	P81	✓	✓	Normal drive/high drive/ large current output	—
PORT9	P90 to P95	✓	✓	Normal drive/high drive/ large current output	—
	P96	✓	✓	Normal drive/high drive	—
PORTA	PA0 to PA5	✓	✓	Normal drive/high drive	—
PORTB	PB0, PB3, PB4, PB6, PB7	✓	✓	Normal drive/high drive	—
	PB1, PB2	✓	✓	Fixed to normal output	✓
	PB5	✓	✓	Normal drive/high drive/ large current output	—
PORTD	PD0 to PD2, PD4 to PD7	✓	✓	Normal drive/high drive	—
	PD3	✓	✓	Normal drive/high drive/ large current output	—
PORTE	PE0, PE1, PE3 to PE5	✓	✓	Normal drive/high drive	—
	PE2	—	—	—	—
PORTN	PN6*1	✓	✓	Normal drive/high drive	—
	PN7*2	✓	✓	Normal drive/high drive	—

Note 1. This pin is initially set to input and is pulled up.
Note 2. This pin is initially set to input and is pulled down.

The following port functions are also valid for other signals (such as serial and other peripheral functions) that share pins with general-purpose I/O ports.

- Input pull-up function
- Open-drain output function
- Drive ability switching function
- 5V tolerant setting

I/O PORT OPERATION - GENERAL I/O INPUT PORT -

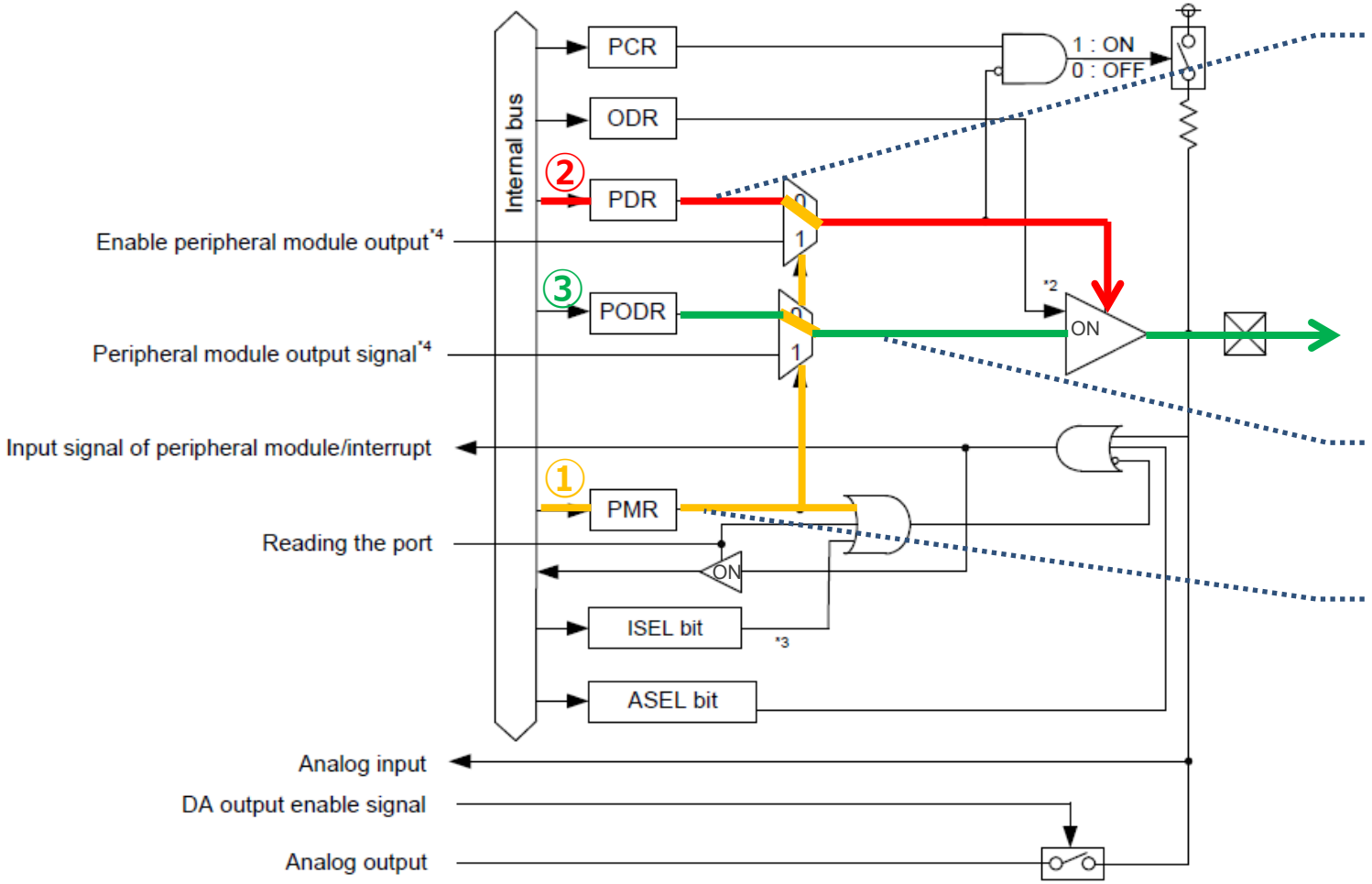


② The output buffer is turned off by setting the PDR register to the “Input” port (PORTn.PDR.Bm = “0”).

① The port is switched to a general I/O pin by setting the PMR register (PORTn.PMR.Bm = “0”).

③ When the port(PORTn.PIDR) is read, the input buffer is turned on and the High/Low of the target port is set to PIDR.

I/O PORT OPERATION - GENERAL I/O OUTPUT PORT -



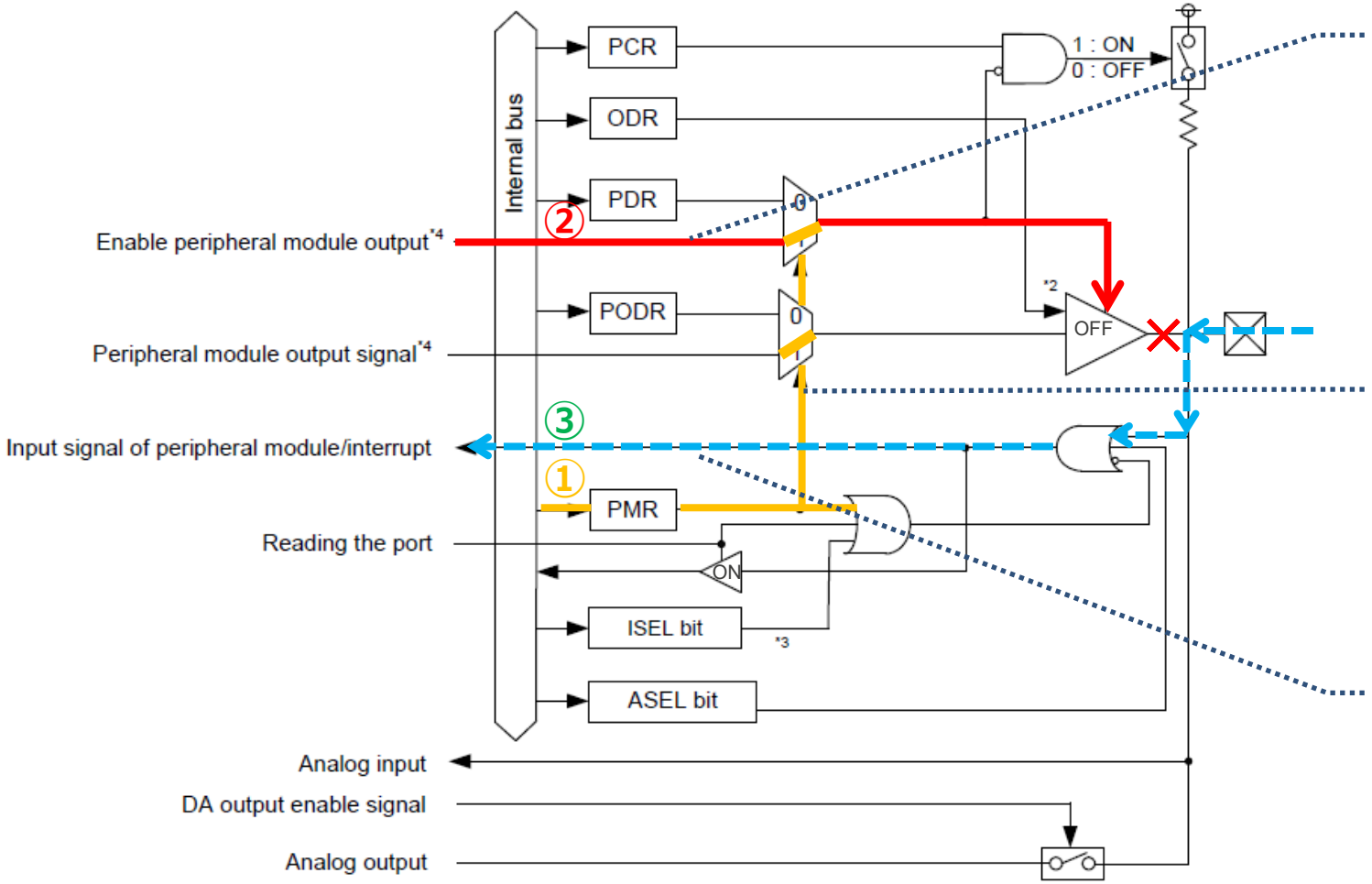
② The output buffer is turned on by setting the PDR register to the “Output” port (PORTn.PDR.Bm = “0”).

③ The value (“1” : High, “0” : Low) of the port output data register (PORTn.PODR) is output from the target port.

① The pin is switched to a general I/O pin by setting the PMR register (PORTn.PMR.Bm = “0”).

I/O PORT OPERATION - PERIPHERAL FUNCTION INPUT -

- When the pin function of the port is set to "Use the pin as peripheral function" and the peripheral function to be used is the **input** operation.



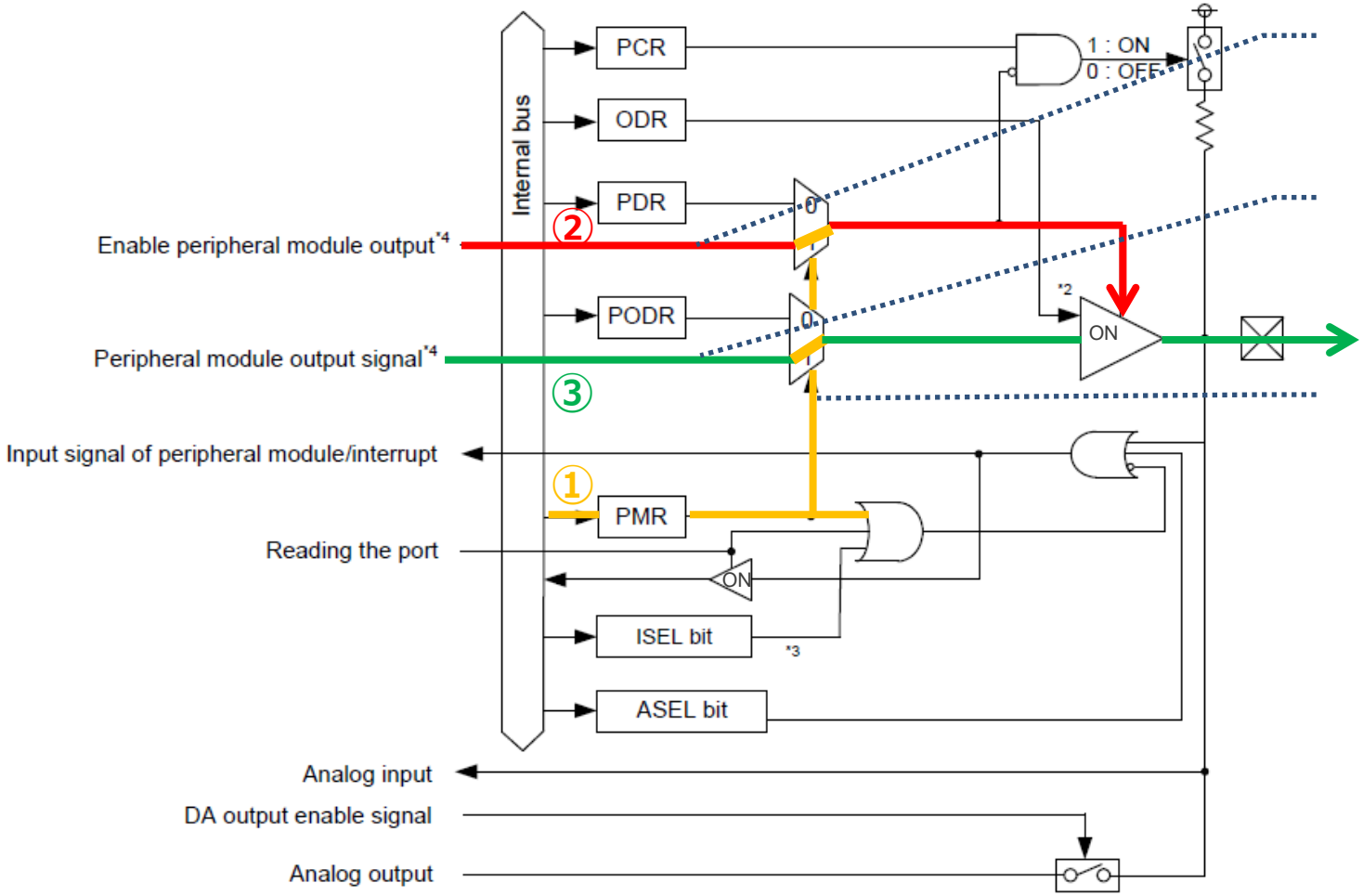
② The buffer is turned off by the output disable operation of peripheral functions.

① The pin is switched to a peripheral functions by setting the PMR register (PORTn.PMR.Bm = “1”).

③ Setting the pin function control register (PmnPFS.PSEL) to a peripheral function, High/Low of the target port is input to the peripheral function.

I/O PORT OPERATION - PERIPHERAL FUNCTION OUTPUT -

- When the pin function of the port is set to "Use the pin as peripheral function" and the peripheral function to be used is the **output** operation.

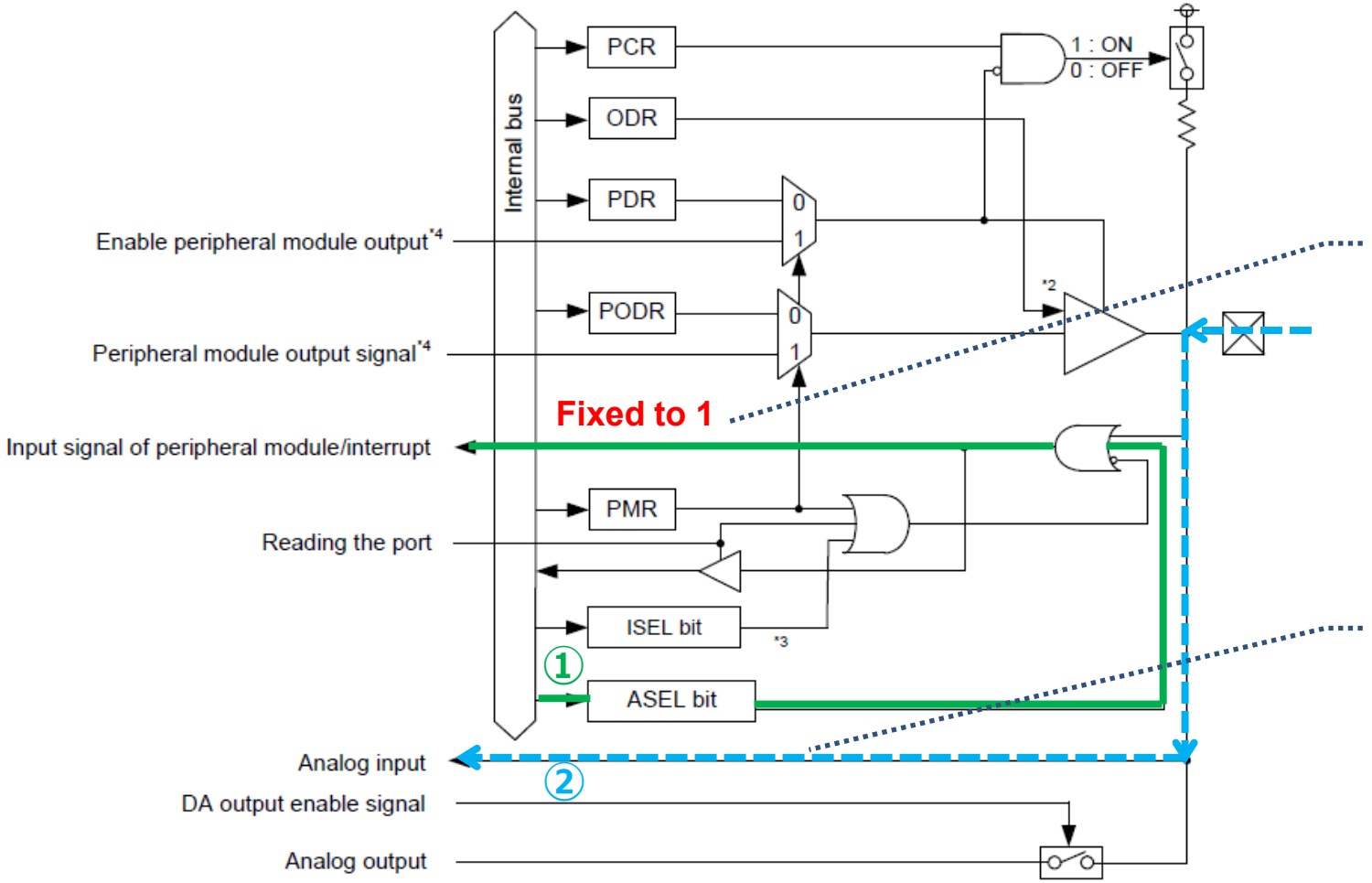


② The buffer is turned on by the output enable operation of the peripheral function.

③ Setting the pin function control register (PmnPFS.PSEL) to a peripheral function enables the port output operation from the peripheral function to output High/Low to the target port.

① By setting the PMR register to the peripheral function port (PORTn.PMR.Bm = “1”), the pin control is set to the peripheral function.

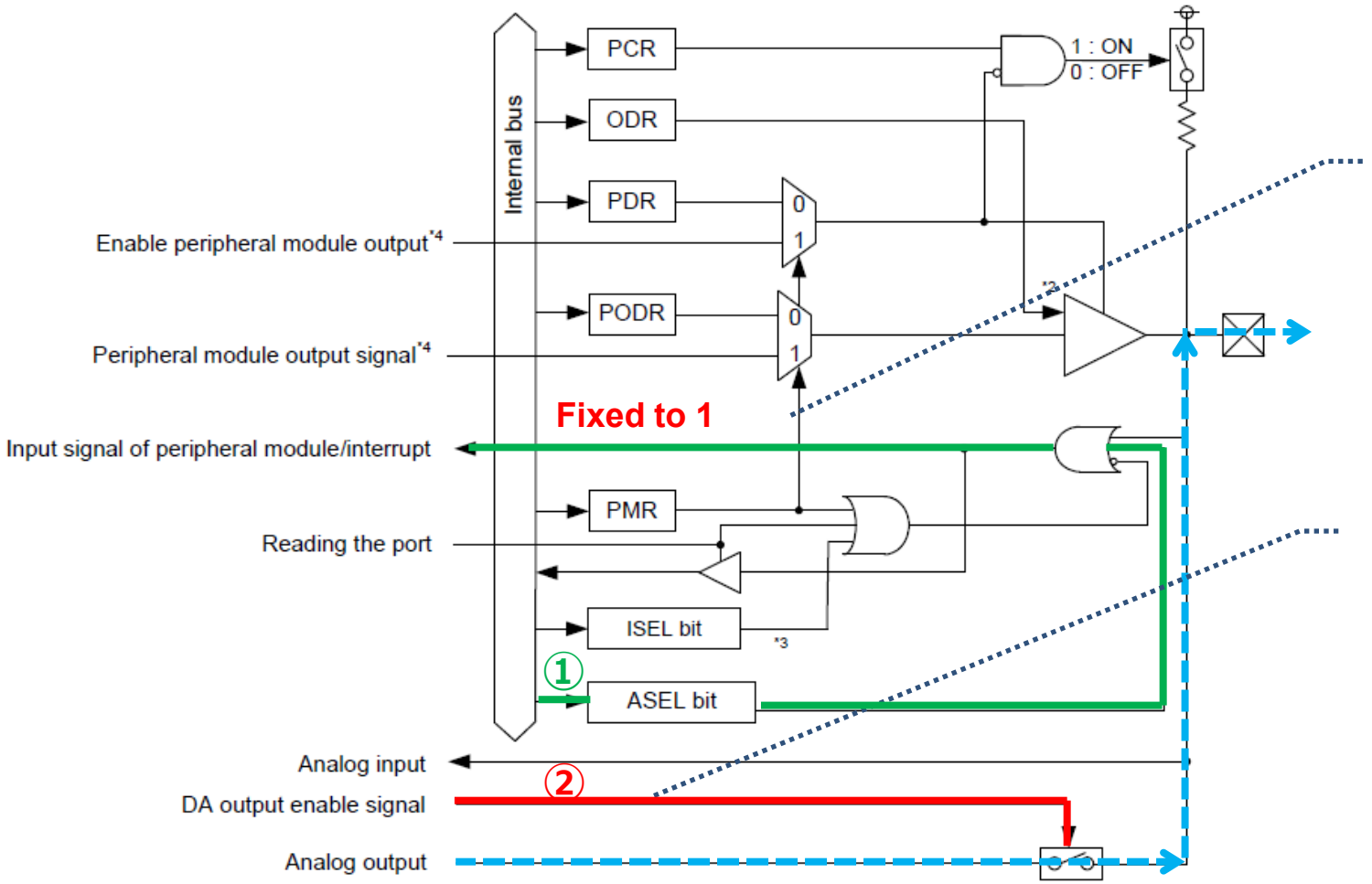
I/O PORT OPERATION - ANALOG FUNCTION INPUT -



① Setting this pin to analog port in the pin function control register (PmnPFS.ASEL) fixes the internal digital input signal to “1”.

② Analog input is enabled by starting an analog input operation (e.g., A/D sampling operation).

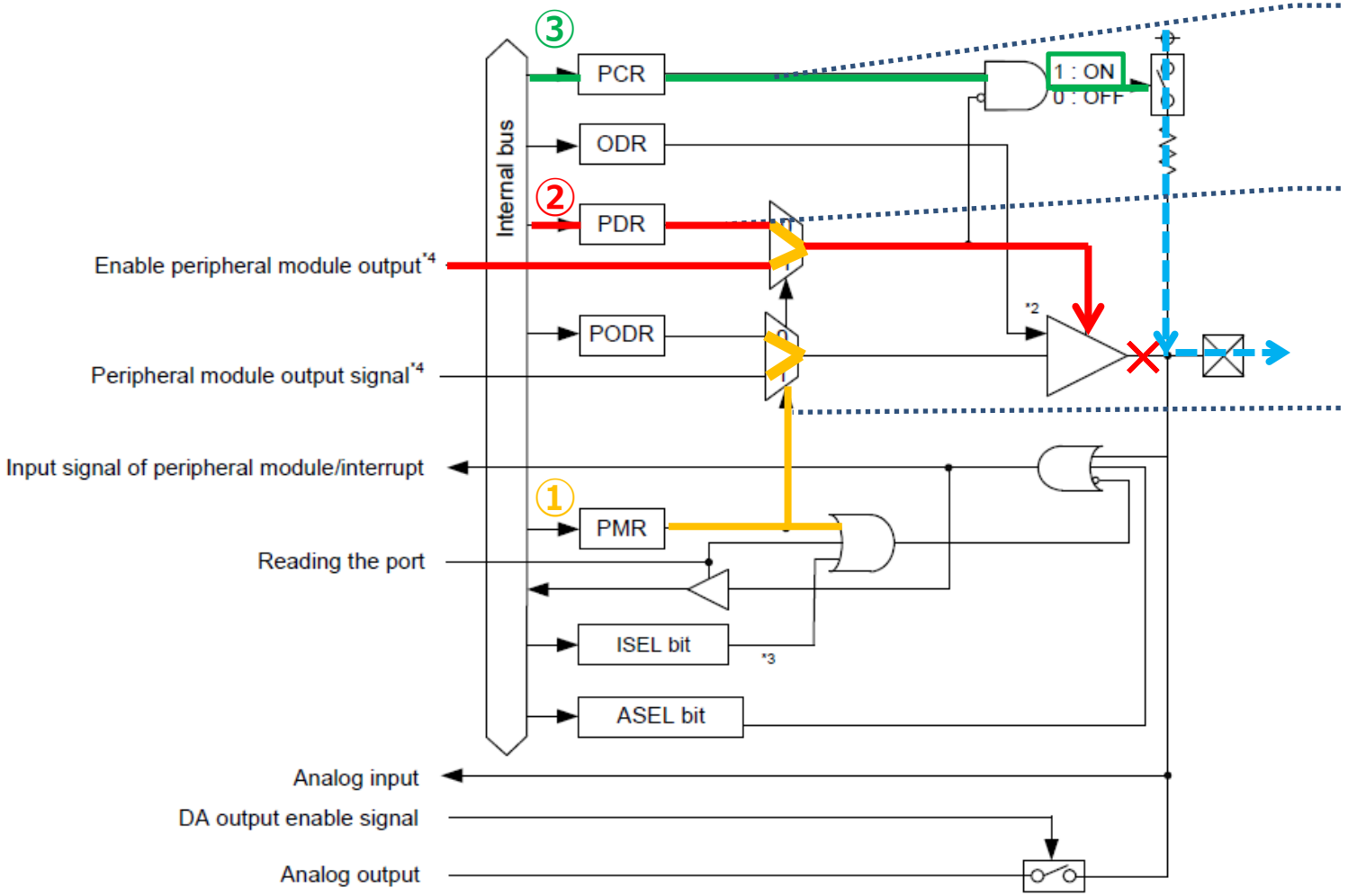
I/O PORT OPERATION - ANALOG FUNCTION OUTPUT -



① Setting this pin to analog port in the pin function control register (PmnPFS.ASEL) fixes the internal digital input signal to “1”.

② Setting the D/A output destination select register (DADSELR.OUTDAn) to “1” enables the D/A signal to be output from the target port.

I/O PORT OPERATION - PULL-UP -

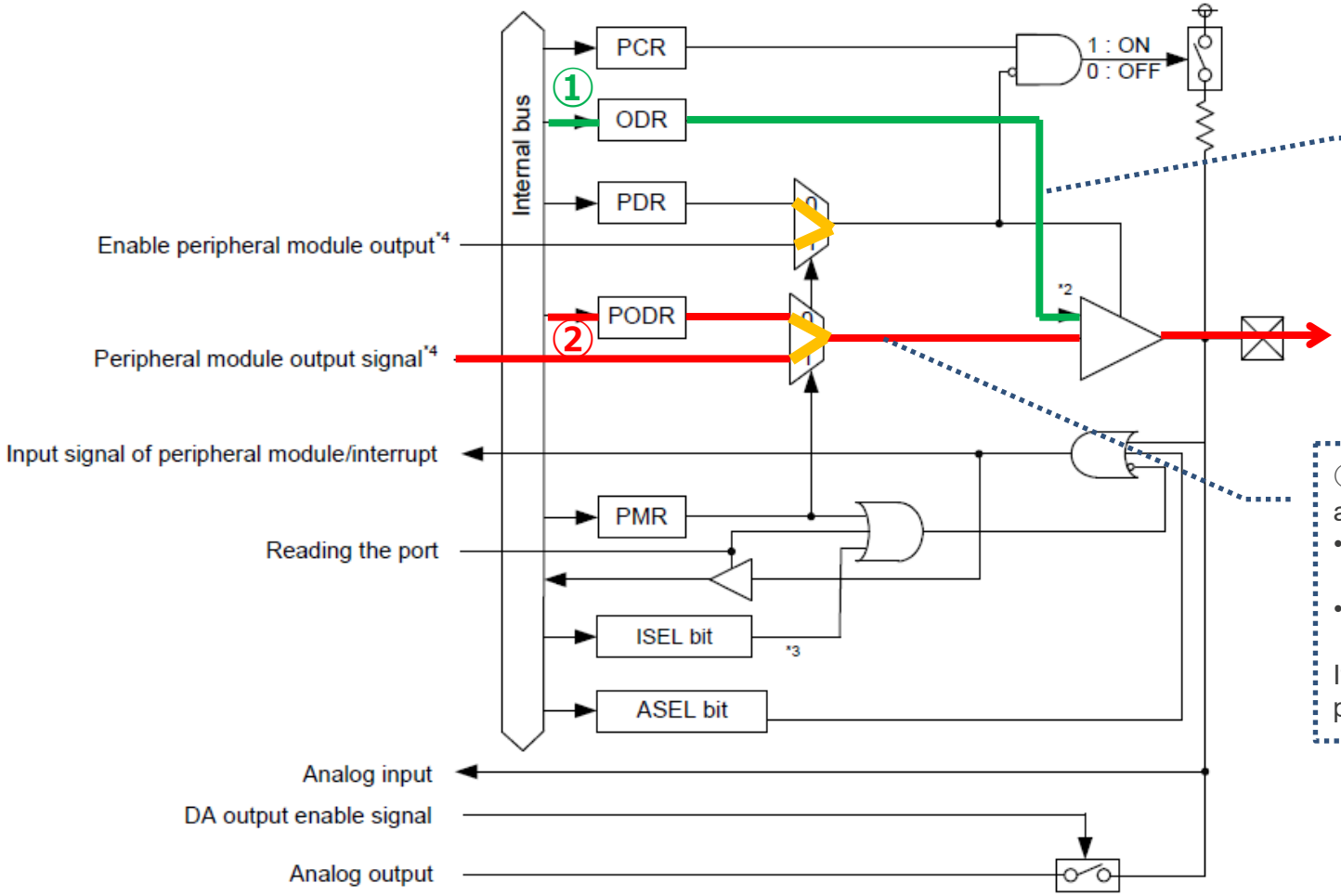


③ After setting ① and ②, pull-up is turned ON by setting the pull-up control register (PCR) to “1” (pull-up control should be turned ON after set to the input port).

② The output buffer is turned off when set to the input port by the PDR register (PORTn.PDR.Bm = “0”) or in the input operation of a peripheral function.

① Setting the PMR register to a general I/O pin or peripheral function (PORTn.PMR.Bm = “0” or “1”).

I/O PORT OPERATION - N-CHANNEL OPEN-DRAIN -



① By setting the Open-Drain Control Register (PORTn.ODRm), the pin function is set to the NMOS open-drain output.

② In N-channel open drain operation, the port output are as follows.

- High output :
  N-MOS turns OFF and the pin output becomes Hi-Z.
- Low output :
  N-MOS turns ON and the pin output becomes Low.

It can be used for both general-purpose output and peripheral function operation.

NOTICE FOR I/O PORT SETTINGS

- Common
  - When PORTm.PIDR is read, the pin status is read regardless of PORTm.PDR and PORTm.PMR registers.
  - Pull-up enables the input pull-up resistor of the pin corresponding to the bit with the PORTm.PCR register set to “1” when the pin is in the input state. The pull-up resistor is disabled during a reset.
  - The input-pull-up function, open-drain output function, drive capability switching function, and 5V tolerant setting are also valid for other signals that share a port with the general-purpose I/O port.

- Interrupt pins
  - The ISEL bit is set when used as an IRQ input pin (external pin interrupt). It can also be used in combination with peripheral functions. However, it is prohibited to enable IRQn of the same number by two or more pins.

- Analog port
  - When setting the pins as analog pins with ASEL bit, set the relevant bit of the port mode register (PORTm.PMR) and the relevant bit of the port direction register (PORTm.PDR) to “0” to set the relevant pins as general-purpose inputs, set PmnPFS.ASEL bit to “1”. The pin status cannot be read at this time.

- Peripheral function port
  - The PmnPFS.PSEL[5:0] bit should be changed with the PMR.Bn bit set to “0”.

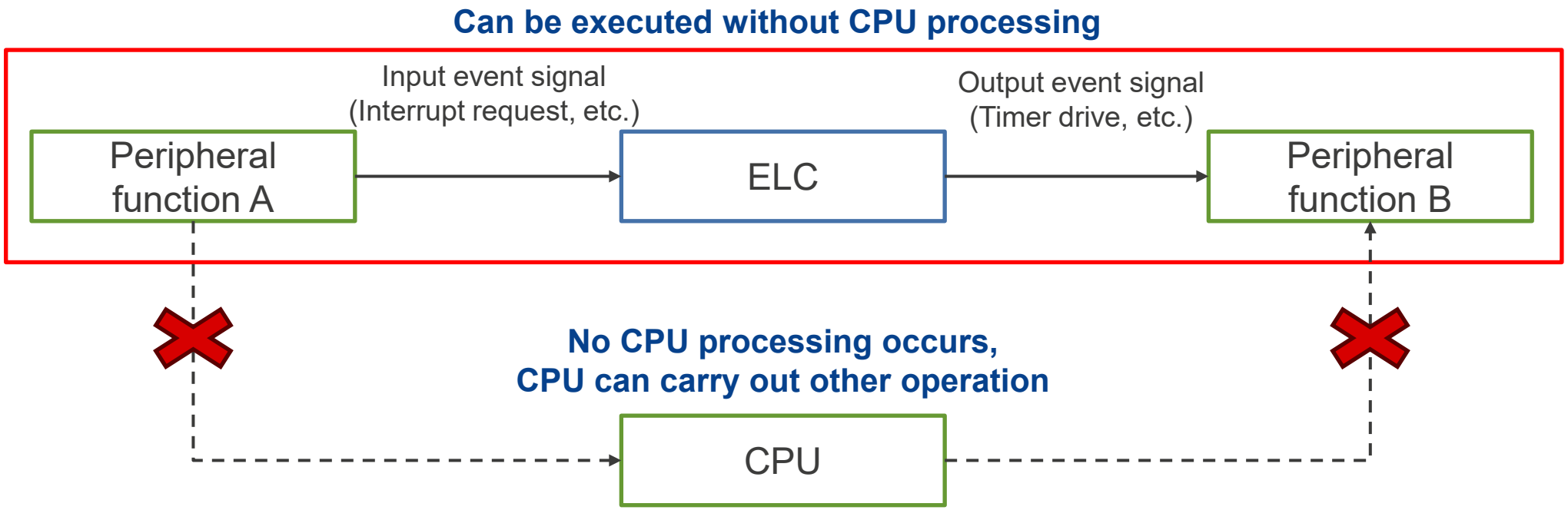
- Other
  - For ports to which RIIC and RI3C are assigned, set the PCR.Bn bit to “0”.
    (Pull-up is automatically turned off for peripheral function outputs other than RIIC and RI3C.)

EVENT LINK CONTROLLER(ELC)

This chapter is written with reference to RX66T, however, can be used as a reference for all products.

ELC OVERVIEW

Event Link Controller (ELC) is a function that triggers an event signal from a peripheral function to activate another peripheral function without involving the CPU. It is possible to trigger various other functions in conjunction with an event from one module.



COMPARING ELC AND CPU INTERRUPTS

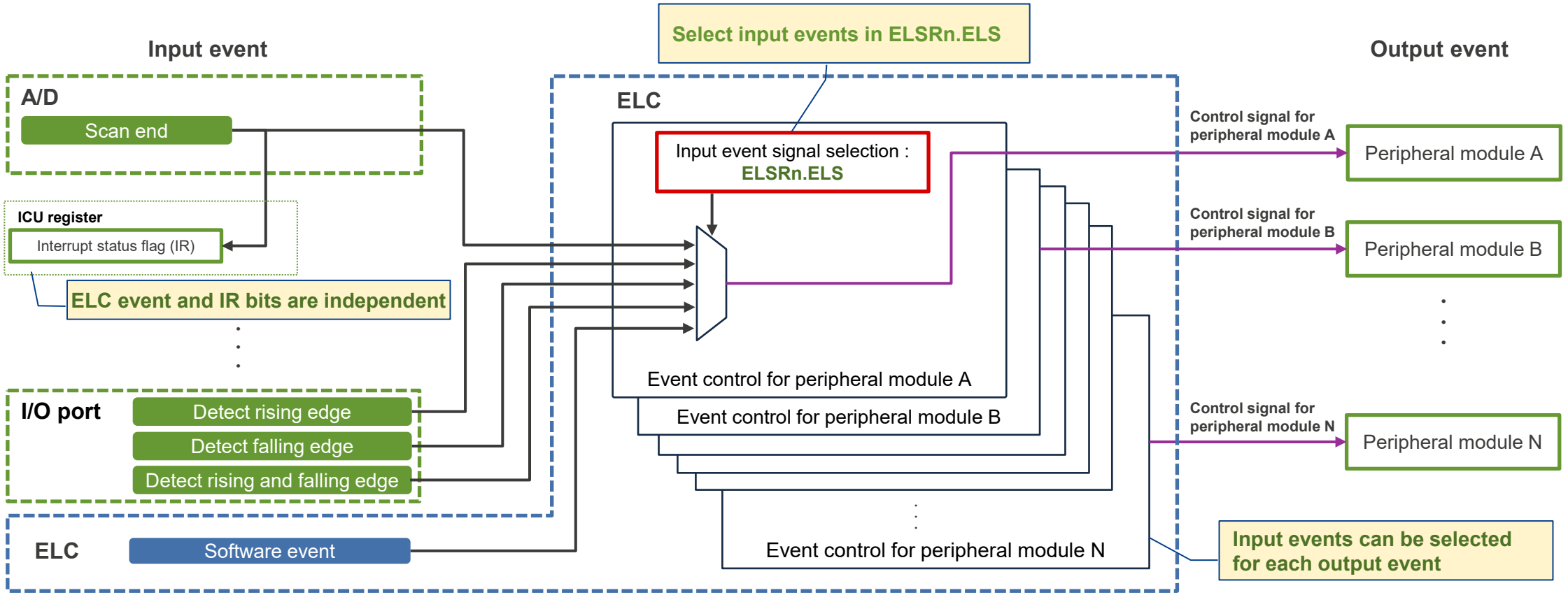
The following compares the case where the event link controller is used with the case where CPU interrupt is used.

Blue : Benefit

Item	ELC	CPU interrupt
CPU usage	**No**	Yes
Number of simultaneous output events for one input event	**Multiple**	Multiple (However, executed one by one depending on the instruction)
Number of output events executed when multiple input events occur	**Multiple output events can be executed in parallel**	Sequentially executed, one by one, in order of priority

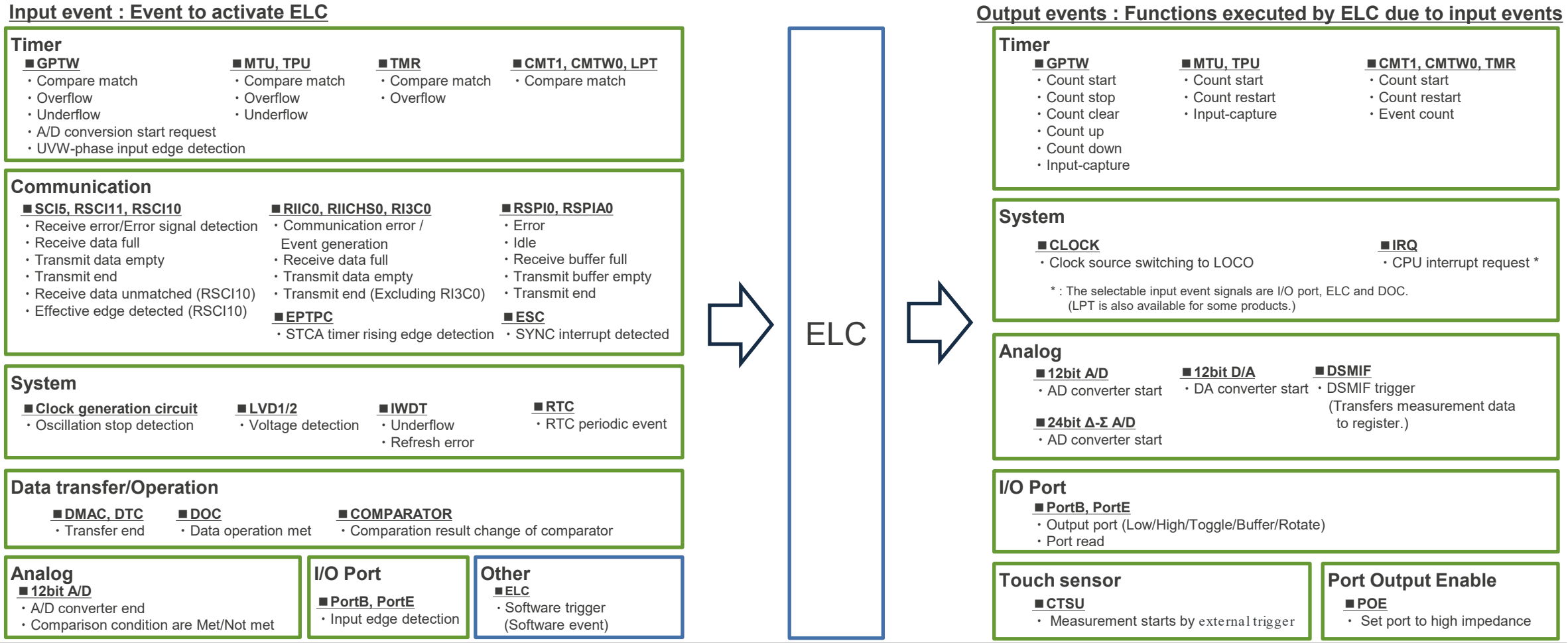
HOW TO CONNECT ELC INPUT AND OUTPUT EVENT

The figure below shows the structure of the Event Link Controller.
ELC is controlled by one control register (ELSRn) for each output event. Multiple input events can be set for one output event.
Input events and interrupt status flags (IR bits) are independent and have no effect on CPU interrupts.



SUMMARY OF ELC INPUT AND OUTPUT EVENTS

Summary of input and output events (The following is an example. Please refer to the “ELC” chapter of each RX product for details).



RELATED REGISTER WHEN USING ELC

ELC is controlled by the ELCR and ELSRn registers. When using the events in the blue frame, the other registers of ELC also need to be set. When using the functions in the red frame, the register settings within each function are also required (Please refer to the next page.).

Input event : Event to activate ELC

Timer

- **GPTW**
  - Compare match
  - Overflow
  - Underflow
  - A/D conversion start request
  - UVW-phase input edge detected
- **MTU, TPU**
  - Compare match
  - Overflow
  - Underflow
- **TMR**
  - Compare match
  - Overflow
- **CMT1, CMTW0, LPT**
  - Compare match

Communication

- **SCI5, RSCI11, RSCI10**
  - Receive error/Error signal detection
  - Receive data full
  - Transmit data empty
  - Transmit end
  - Receive data unmatched (RSCI10)
  - Effective edge detected (RSCI10)
- **RIIC0, RIICHS0, RI3C0**
  - Communication error / Event generation
  - Receive data full
  - Transmit data empty
  - Transmit end (Excluding RI3C0)
- **RSPI0, RSPIA0**
  - Error
  - Idle
  - Receive buffer full
  - Transmit buffer empty
  - Transmit end
- **EPTPC**
  - STCA timer rising edge detection
- **ESC**
  - SYNC interrupt detected

System

- **Clock generation circuit**
  - Oscillation stop detection
- **LVD1/2**
  - Voltage detection
- **IWDT**
  - Underflow
  - Refresh error
- **RTC**
  - RTC periodic event

Data transfer/Operation

- **DMAC, DTC**
  - Transfer end
- **DOC**
  - Data operation met
- **COMPARATOR**
  - Comparation result change of comparator

Analog

- **12bit A/D**
  - A/D converter end
  - Comparison condition are Met/Not met

I/O Port

- **PortB, PortE**
  - Input edge detection

Other

- **ELC**
  - Software trigger (Software event)

ELC Register

- **ELCR** : ELC Enable/Disable setting
- **ELSRn** : Input event settings
- **ELOPn** : Timer output event setting
- **PGR/PDFBn/PGCn/PELm** : I/O port-related I/O event settings
- **ELSEGR** : Software Event Settings

Output events : Functions executed by ELC due to input events

Timer

- **GPTW**
  - Count start
  - Count stop
  - Count clear
  - Count up
  - Count down
  - Input-capture
- **MTU, TPU**
  - Count start
  - Count restart
  - Input-capture
- **CMT1, CMTW0, TMR**
  - Count start
  - Count restart
  - Event count

System

- **CLOCK**
  - Clock source switching to LOCO
- **IRQ**
  - CPU interrupt request *

* : The selectable input event signals are I/O port, ELC and DOC. (LPT is also available for some products.)

Analog

- **12bit A/D**
  - AD converter start
- **24bit Δ-Σ A/D**
  - AD converter start
- **12bit D/A**
  - DA converter start
- **DSMIF**
  - DSMIF trigger (Transfers measurement data to register.)

I/O Port

- **PortB, PortE**
  - Output port (Low/High/Toggle/Buffer/Rotate)
  - Port read

Touch sensor

- **CTSU**
  - Measurement starts by external trigger

Port Output Enable

- **POE**
  - Set port to high impedance

RELATED REGISTER WHEN USING ELC

The following figure shows the registers that need to be set on the function side other than ELC.

Input event

Timer

■ **GPTW**

・A/D converter start request
GTINTAD.ADTRnUEN :
Set the conditions of event generation (Up count compare match)
GTINTAD.ADTRnDEN :
Set the conditions of event generation (Down count compare match)

Communication

■ **ESC**

・SYNCn signal
ESCICR.SYNCnC : Set the edge condition (Up count/Down count)

Output event

Timer

■ **GPTW**

・Count start
GTSSR : Count start factor setting
・Count stop
GTPSR : Count stop factor setting
・Count clear
GTCSR : Count clear factor setting
・Count up
GTUPSR : Count up factor setting
・Count down
GTDNSR : Count down factor setting
・Capture operation
GTICnSR : Input capture factor setting

System

■ **Clock**

・Clock source switching LOCO (Low-speed on-chip oscillator)
RSTCKCR.RSTCKEN : Set bit to 0 when using this event

Analog

■ **12bit A/D**

・A/D converter start
ADSTRGR.TRSAn/**ADGCTRGR.TRSC** :
A/D conversion start trigger selection

■ **24bit Δ-Σ A/D**

・A/D converter start
MR.TRGMD :
A/D conversion start trigger selection

Touch sensor

■ **CTSU**

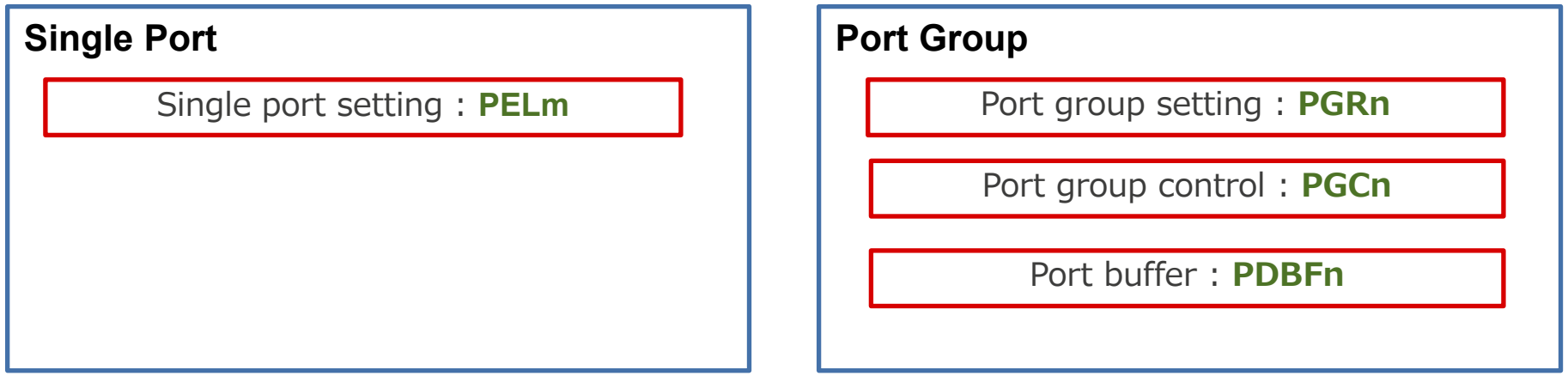
・Measurement starts by external trigger
CTSUCR0.CTSUCAP : Measurement start trigger selection

HOW TO SET ELC
I/O PORT : SINGLE PORT AND PORT GROUP

There are two types of input/output events using I/O port : “Single Port" and “Port Group".

- Single Port : This function connects I/O port pins and events on a one-to-one basis.
- Port Group: This is a function to connect multiple ports of an I/O port as a single group to an event.

Note that the registers set by "Single Port" and "Port Group" differ as follows.



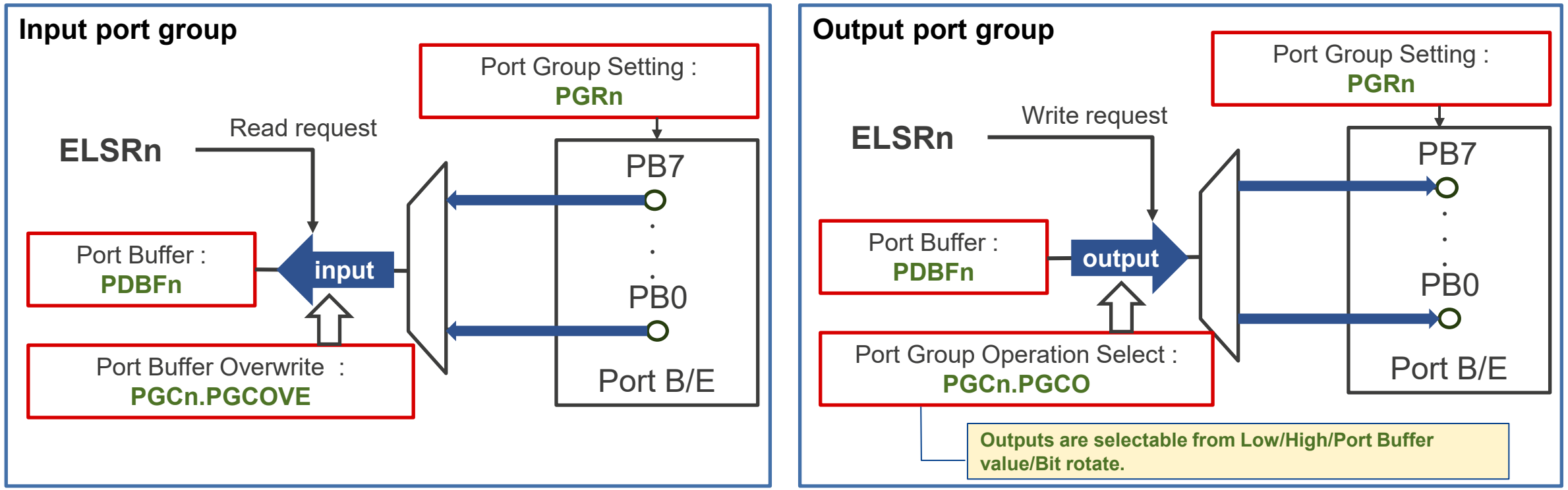
HOW TO SET ELC

I/O PORT : PORT GROUP OUTPUT EVENTS

There are two types of output events for a port group :

- Input port group : When an event is input, the port status(High/Low) of the port selected by PGRn is transferred to the port buffer (PDBFn).
  (Select whether to store only once by PGCn.PGCOVE or overwrite it and store it many times.)
- Output port group : Output from the pin selected by PGRn as specified by PGCn.PGCO when an event is input.
  (Use PGCn.PGCO to select the output mode.)

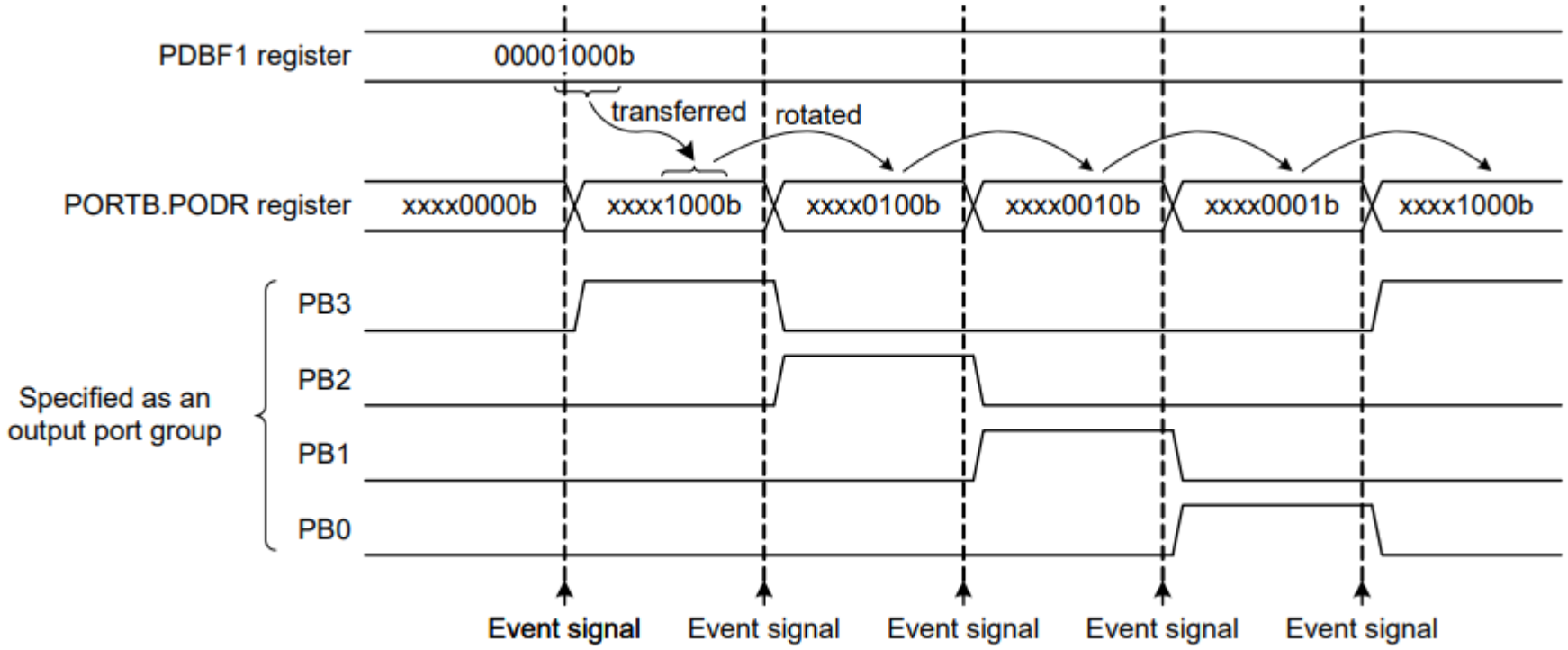
: Registers that need to be set



HOW TO SET ELC

SUPPLEMENTAL INFORMATION) WHAT IS BIT ROTATE OUTPUT?

- When an output event is used in a port group, the value set in PDBFn register is rotated to MSB→LSB for each input event, and the value is output from each port. For example, if PDBFn register is set to 00001000b as shown below, High output-port can be switched for each event-input.



SUPPLEMENTARY INFORMATION & NOTES FOR ELC SETTINGS

■ Supplementary information

- ELC can operate even when CPU is stopped, such as in sleep mode.
  However, please note that it will not operate in a mode where the clock supply to the ELC and the target peripheral module is stopped, as described in the "Usage Notes" section of the ELC chapter.
- Input events used for ELC can also be used as transferring source for DMAC and DTC.

■ Precautions

- Be sure to read the precautions for using ELC in the section "Usage Notes" in ELC chapter.
- Be sure to read the "Usage Notes" for each peripheral modules to be used.
- When ICU interrupts are used as output events, there are limitations on the input events that can be used.
  For details, refer to the hardware manual.

USE CASE

ELC can be used for a variety of purposes, the following is the example :

① : Execute peripheral function without CPU operation

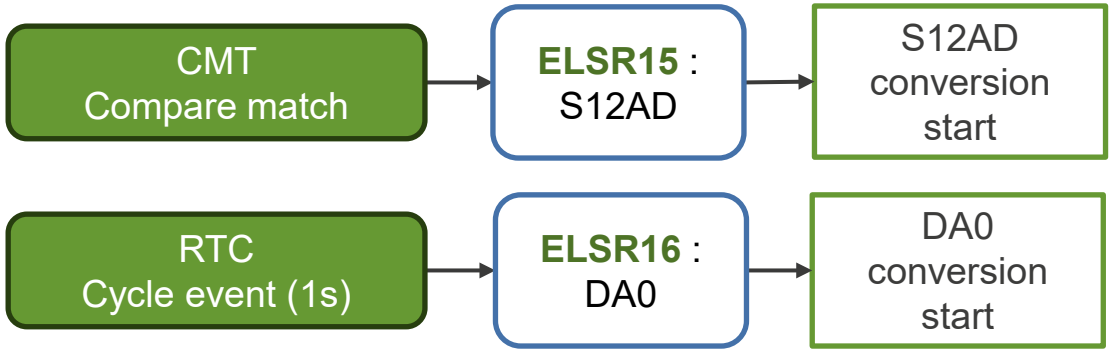
② : Execute several different peripherals with one event signal

③ : Execute sequentially of peripherals by connecting them together

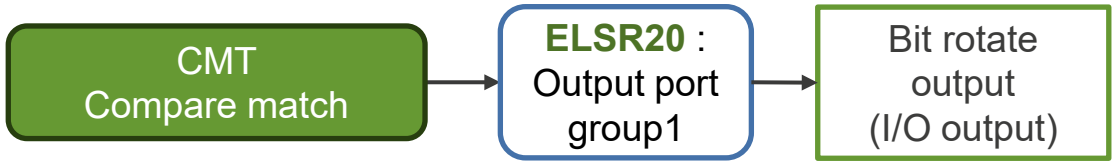
This chapter is written with reference to RX72M, however, can be used as a reference for all products.

USE CASE ① :
EXECUTE PERIPHERAL FUNCTION WITHOUT CPU OPERATION

- The following shows an example in which a peripheral module is executed at a fixed period without CPU operation by utilizing ELC.



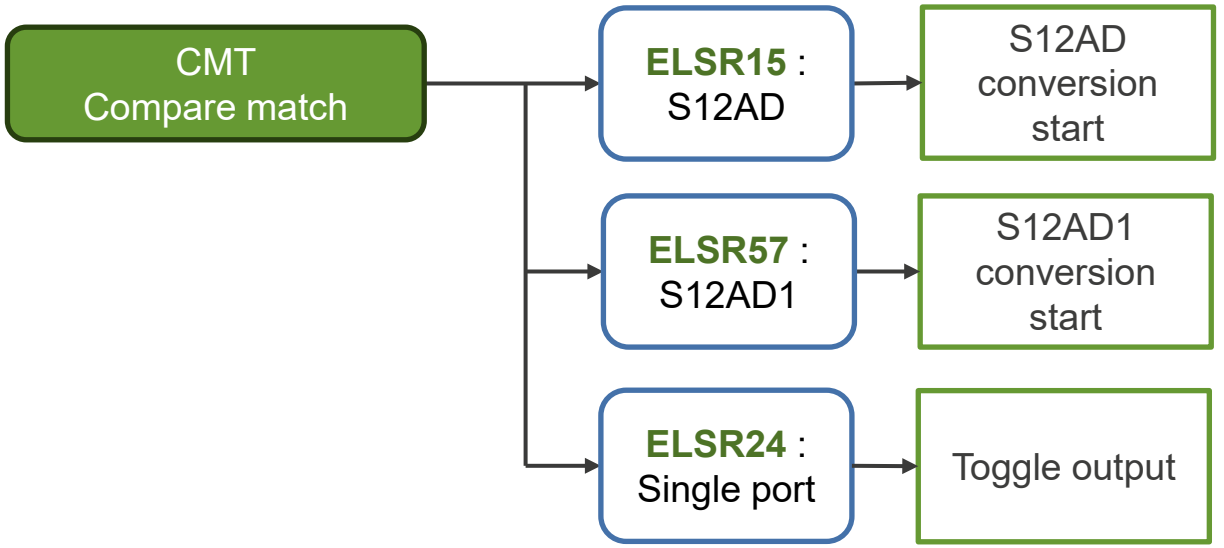
Example 1 : Starting 12 bits A/D or D/A converter by compare match timer or RTC cycle event



Example 2 : Bit Rotate Output at Period
(The value set in PDBF register is rotation from MSB to LSB.)

USE CASE ② :
EXECUTE SEVERAL DIFFERENT PERIPHERALS WITH ONE EVENT SIGNAL

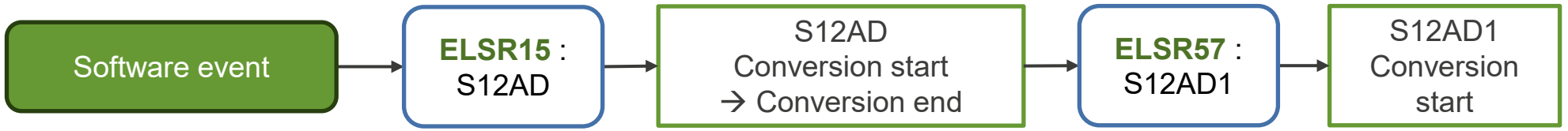
- The following shows how to use ELC to activate several peripheral modules with a single input-event signal.



e.g. : Simultaneously operating S12AD, S12AD1 and I/O outputting (toggling) at CMT compare match

USE CASE ③ :
EXECUTE SEQUENTIALLY OF PERIPHERALS BY CONNECTING THEM TOGETHER

- The following shows an example of starting several peripheral modules in succession with a single input-event signal by utilizing ELC.



e.g. : Conversion of S12AD is started by software event, conversion of S12AD1 is started by S12AD conversion end as input event.

ADVANCED EXAMPLE

Detailed examples are introduced on the following.

① : Operate SCI data transfer and re-set after timeout without using the CPU

② : Key-scanning without CPU operation

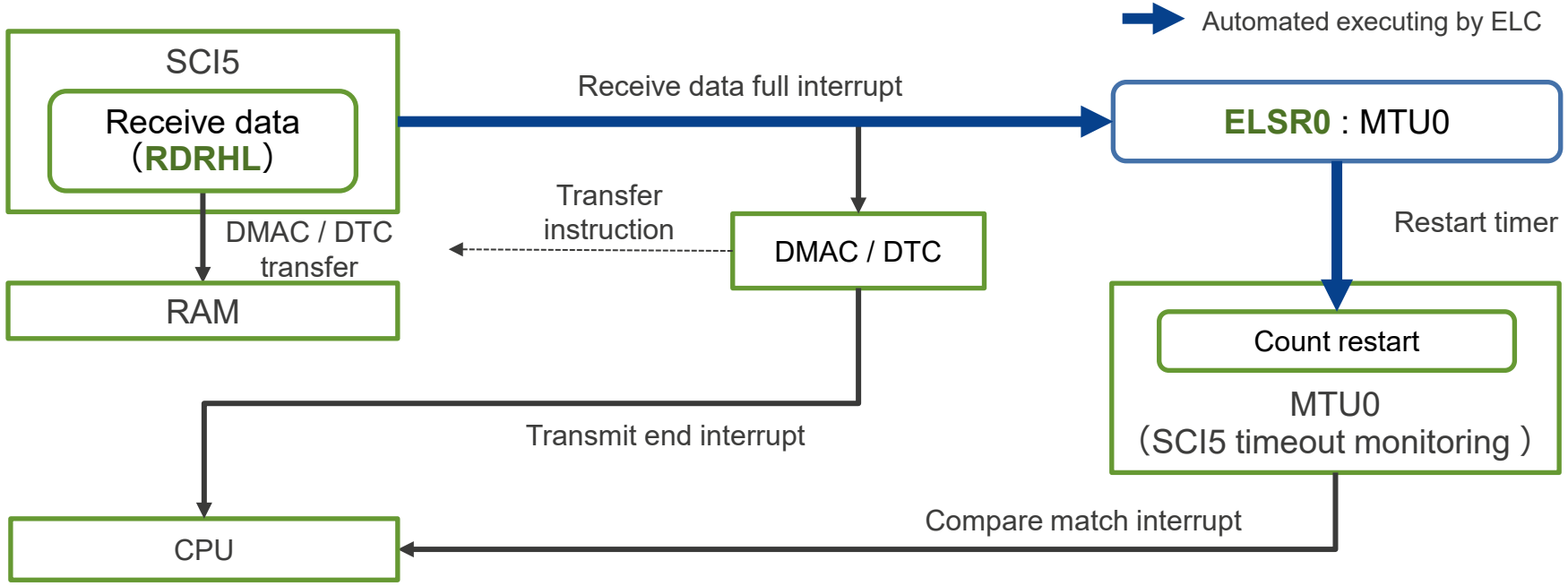
③ : Periodically AD converting & save the conversion result in RAM without CPU operation

This chapter is written with reference to RX72M, however, can be used as a reference for all products.

ADVANCED EXAMPLE ① :

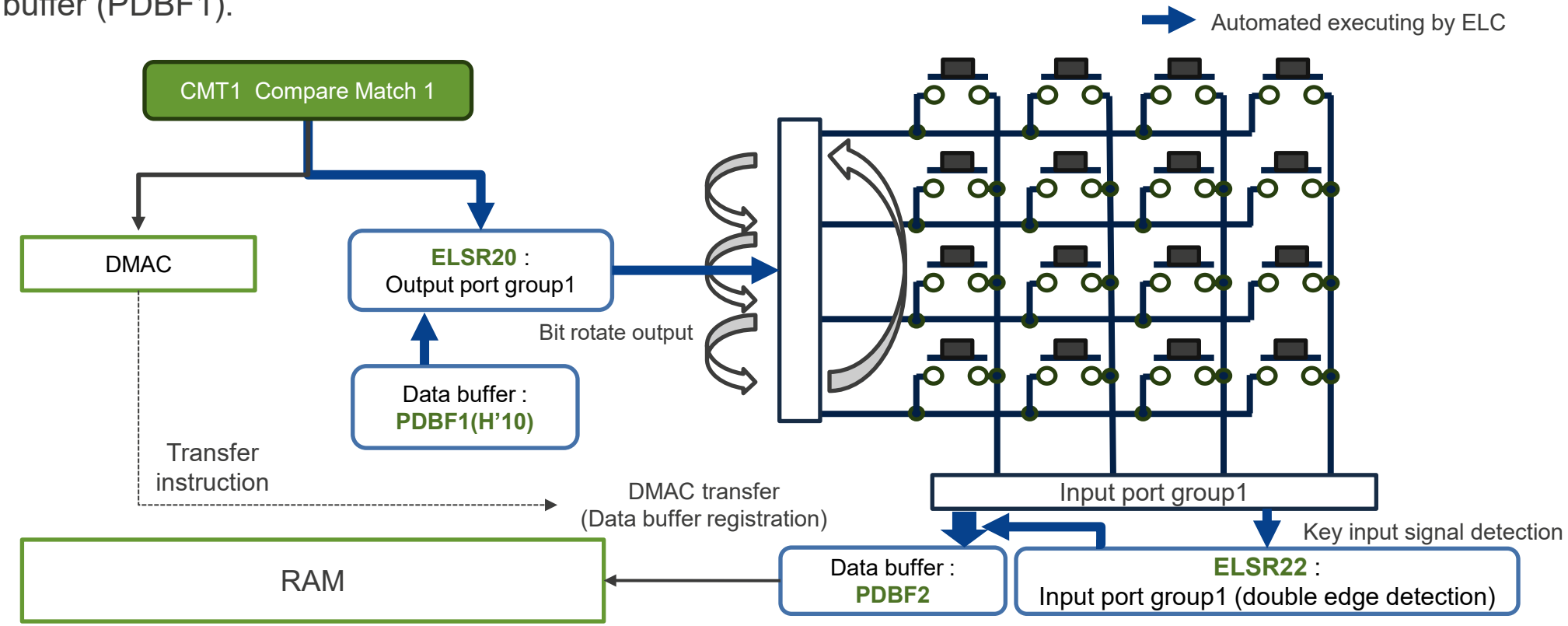
PERFORM SCI DATA TRANSFER AND RESET AFTER TIMEOUT WITHOUT USING THE CPU

When SCI reception is required at regular intervals, timeout detection is automatically performed using MTU after SCI reception until the next reception is completed. There is no operation in CPU in this example.
SCI receive data full interrupt is used as ELC input event, and ELC restarts MTU count. MTU is set to perform a compare match according to the reception interval. If “Restart” is not in time, CPU is notified that a compare match interrupt occurred and timed out. In addition, the received data is transferred to RAM by DMAC/DTC at the same time with the received data full interrupt as the transfer factor.



ADVANCED EXAMPLE ② :
KEY-SCANNING WITHOUT CPU OPERATION

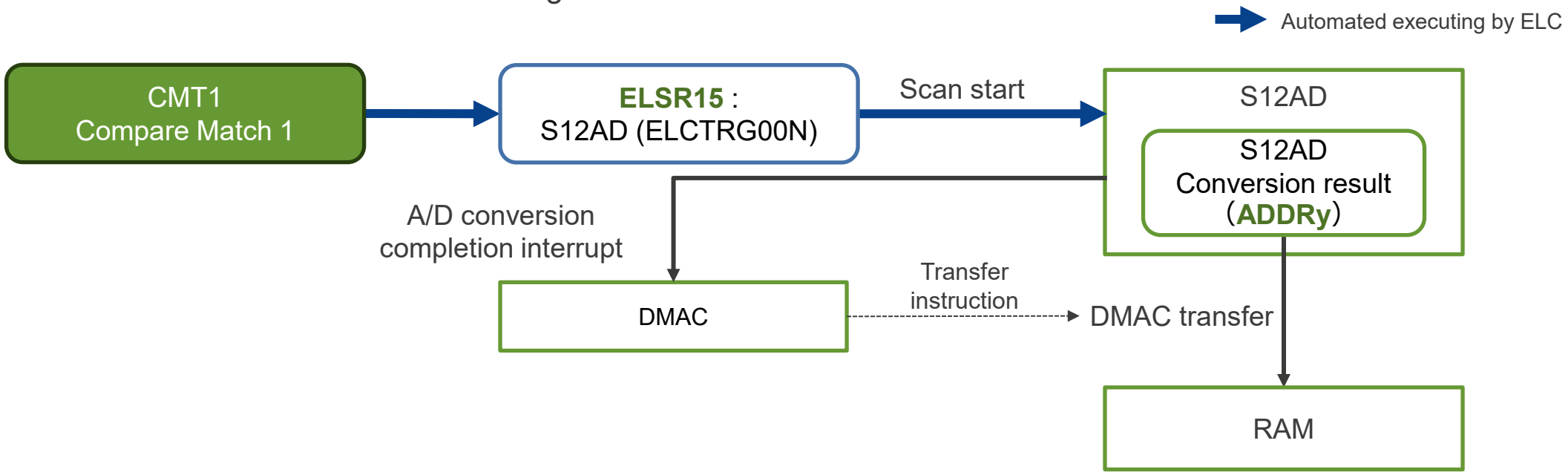
This is an example of implementing the key scan function. When an input-event CMT compare match occurs, ELC is activated, the output port group is rotated, and the rows are enabled in order. In addition, DMAC transfers data from the data buffer (PDBF2) to RAM using this input event as the transfer factor. In the operation of the column side, the input port group pins change by pressing the button, and an input event occurs. This activates ELC and stores the values of the input port group pins in the data buffer (PDBF1).



ADVANCED EXAMPLE ③ :

PERIODICALLY AD CONVERTING & SAVE THE CONVERSION RESULT IN RAM WITHOUT CPU OPERATION

By using a combination of ELC and DMAC, it is possible to periodically perform from A/D converting process to change data storing without going through CPU. When a compare match occurs on CMT, ELC starts A/D conversion. Also, A/D conversion result is saved to RAM by DMAC using A/D conversion end interrupt as the transfer factor. By setting DMAC to the free-run mode, you do not need to set the transfer count again.



LIST OF ELC APPLICATION NOTES

Please refer to the following for more detailed ELC usage.

- RX200 Series Using the ELC on the RX210 R01AN1099
- RX Family Tamper Detection Method Utilizing Existing Peripheral Functions R01AN7654

MTU

TIMER FUNCTION COMPARISON

- The following is a comparison of each timer function in RX family.

* Varies depending on the product

Operation mode	GPT/GPTW	MTU3	TPU	TMR	CMT	CMTW
Bits size	16/32	16	16	8	16	32
Maximum number of channels*	10	9	6	4	4	2
Maximum number of PWM outputs*	20	14	15	4	-	2
Maximum operating clock	Equivalent to ICLK	Equivalent to ICLK	PCLK	PCLK	PCLK	PCLK
PWM output	✓	✓	✓	✓	-	✓
Complementary PWM output	✓	✓	-	-	-	-
Positive-phase/negative-phase independently compare operation	✓	-	-	-	-	-
Input capture	✓	✓	✓	-	-	✓
Phase counting mode	✓	✓	✓	-	-	-
Output compare	✓	✓	✓	✓	-	✓
External clock count	✓	✓	✓	✓	-	-
Free-running operation	✓	✓	✓	✓	✓	✓
Compare match operation	✓	✓	✓	✓	✓	✓
High resolution output control	✓	-	-	-	-	-

MTU FUNCTION LIST

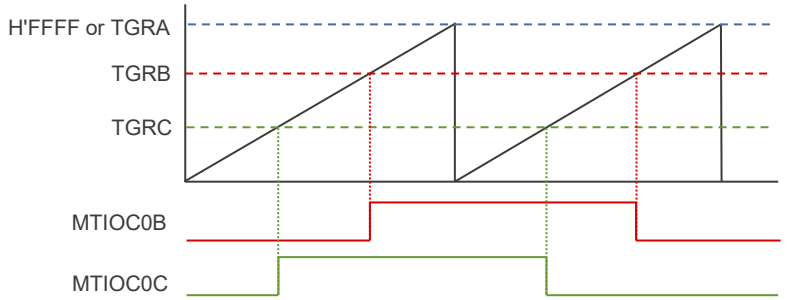
- The following table lists the operation modes held by MTU, registers supporting buffer operation, and optional operations.

Operation mode	Registers for buffer operation (MTU0,3,4,6,7,9)	Optional operation	Channel
Normal mode Output compare Input capture	TGR (Output compare/ Input capture)	Synchronous operation Noise Filter A/D conversion start request / A/D conversion Delaying	MTU0, 1, 2, 3, 4, 6, 7, 9 * A/D conversion start request is only available for MTU4, 7
PWM 1/2 mode Output compare Input capture	TGR (Output compare/ Input capture)	Synchronous operation Noise Filter A/D conversion start request / A/D conversion Delaying	MTU0, 1, 2, 3, 4, 6, 7, 9 * A/D conversion start request is only available for MTU4, 7
Reset-Synchronized PWM Mode	TGR (Output compare) TOCR2 (Output control)	Synchronous operation Noise Filter A/D conversion start request / A/D conversion Delaying AC synchronous motor drive mode	MTU3, 4, 6, 7 * A/D conversion start request is only available for MTU4, 7 * AC synchronous motor drive mode is only available for MTU3, 4
Complementary PWM Mode	TGR (Output compare/ Input capture) *Support for double buffer TOCR2 (Output control) TCDR (Cycle register)	Synchronous operation Noise Filter A/D conversion start request / A/D conversion Delaying AC synchronous motor drive mode Interrupt skipping	MTU3, 4, 6, 7 * A/D conversion start request is only available for MTU4, 7 * AC synchronous motor drive mode is only available for MTU3, 4
Phase Counting Mode	TGR (Output compare/ Input capture)	Noise Filter	MTU1, 2
Phase Counting Mode (Cascade Connection)	TGR (Output compare/ Input capture)	Noise Filter	MTU1, 2

OUTPUT WAVEFORM
NORMAL MODE, PWM MODE 1/2

Normal mode

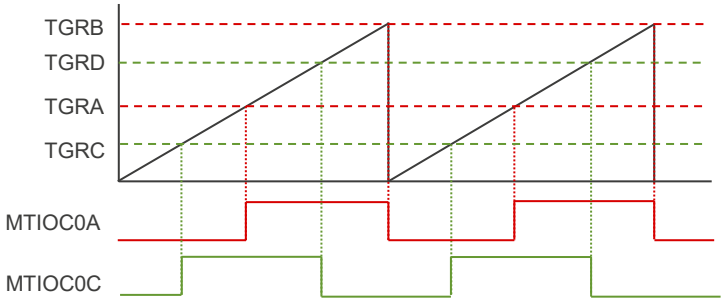
- TCNT clear condition : H'FFFF or TGR register
- PWM output waveform : Output waveforms corresponding to TGR
- Duty : 50% duty waveform only



Example : MTU0 normal mode operating (TGRB,TBRC are toggle operating)

PWM mode 1

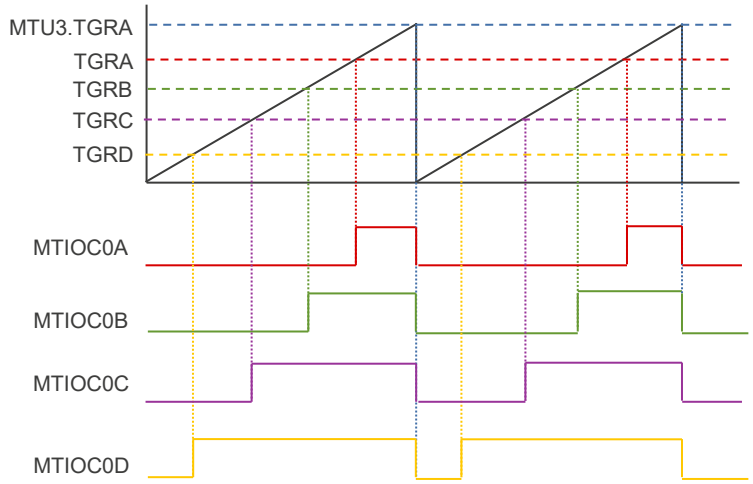
- TCNT clear condition : Any TGR register
- PWM output waveform : Outputs waveform by combination of TGRA&TGRB,TGRC&TGRD
- Duty : It is available PWM waveform output from 0%~100%



Example : MTU0 PWM mode 1 operating (Counter clear by TGRB)

PWM mode 2

- TCNT clear condition : Any TGR register or TGR register of other mode
- PWM output waveform : Only condition for clearing is TCNT counter clearing, the cycle is set by each TGR register.
- Duty : Available PWM waveform output from 0%~100%



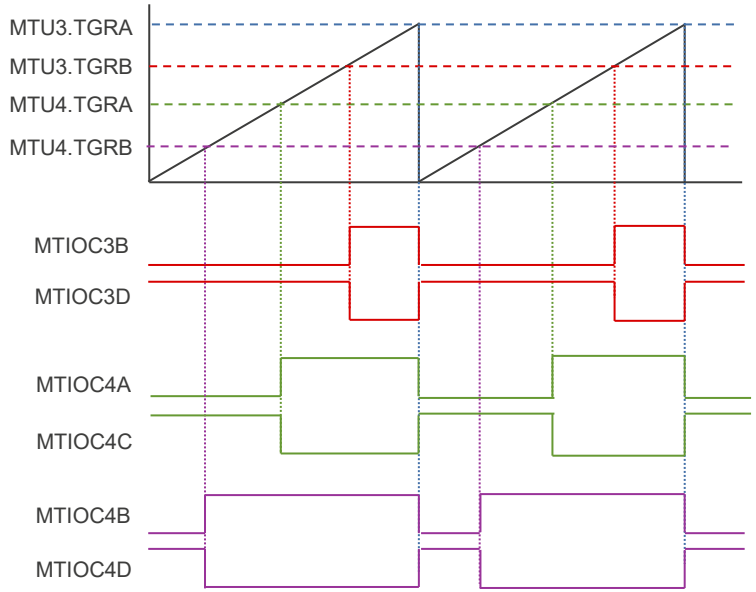
Example : MTU0 PWM mode 2 operating (TCNT count clear condition is MTU3.TGRA at operating mode 1)

OUTPUT WAVEFORM

RESET-SYNCHRONIZED PWM MODE, COMPLEMENTARY PWM MODE

Reset-synchronized PWM mode

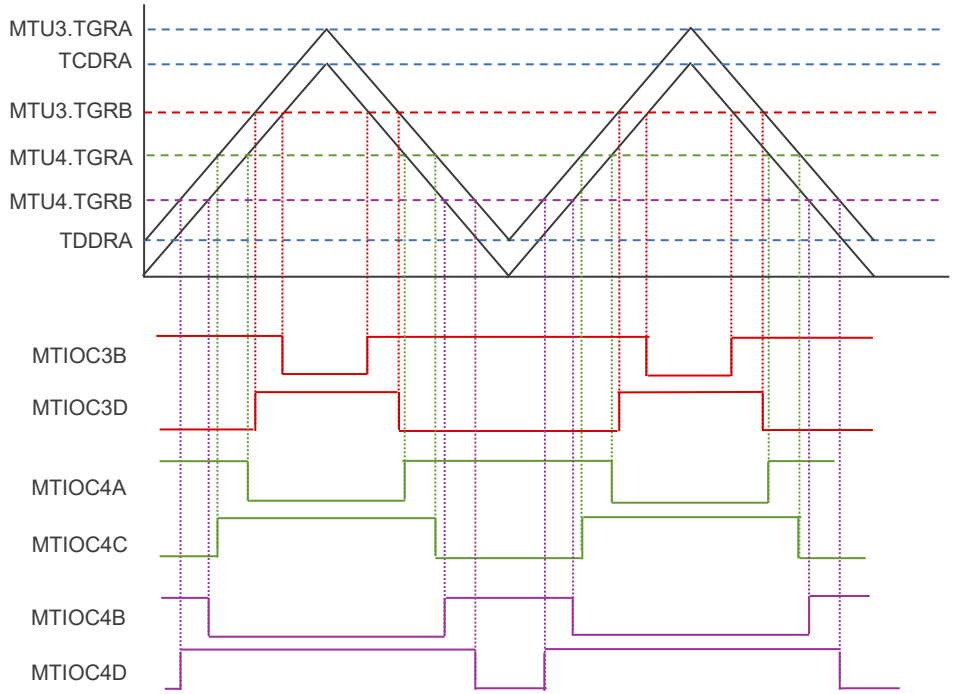
- TCNT clear condition : MTU3(6).TGRA register
- PWM output waveform : Toggle output (Positive and Negative) by compare-match of MTU3(6).TGRB/MTU4(7).TGRA/MTU4(7).TGRB and counter clear
- Duty : Available PWM waveform output between 0%~100%
- Dead time : Without



Example : MTU3, MTU4 Reset-synchronized PWM mode operating

Complementary PWM mode

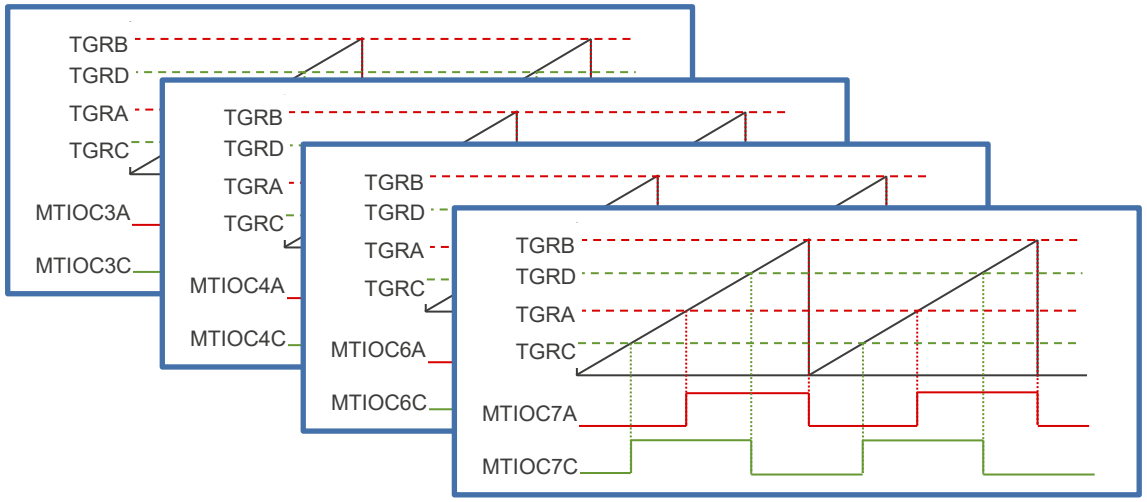
- TCNT clear condition : MTU3(6).TGRA register
- PWM output waveform : Toggle output (Positive and Negative) by compare-match of MTU3(6).TGRB/MTU4(7).TGRA/MTU4(7).TGRB.
- Duty : Available PWM waveform output between 0%~100%
- Dead-time : Retains the dead-time of value set in TDDRA



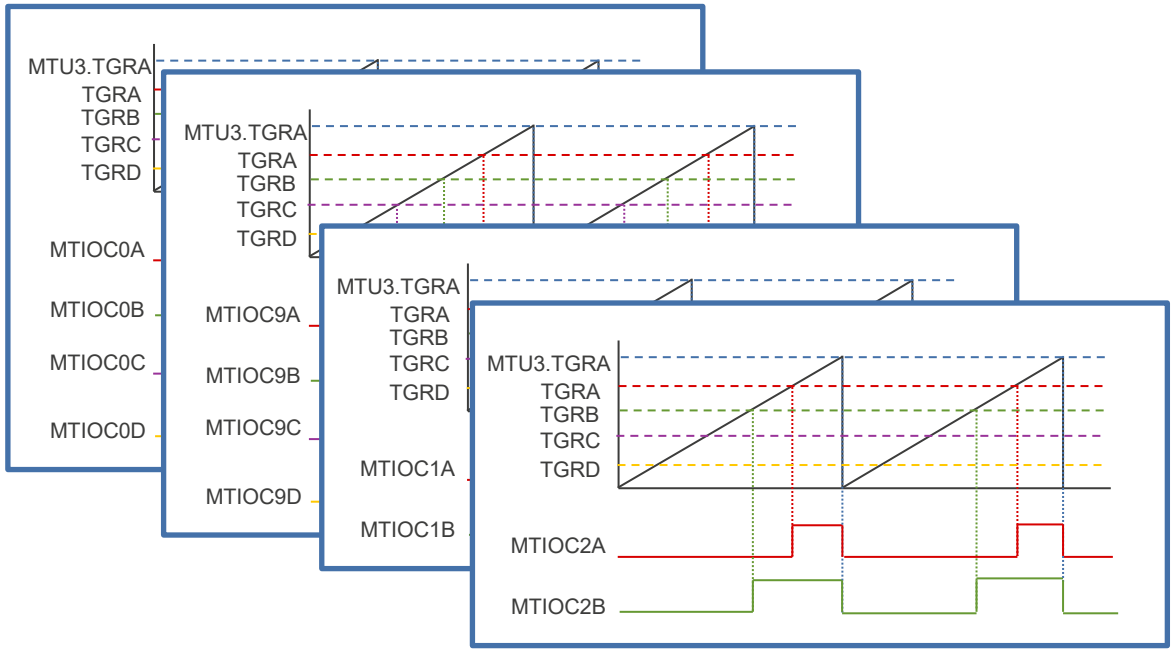
Example : MTU3, MTU4 complementary PWM mode operating

MAXIMUM NUMBER OF PWM OUTPUTS

The maximum number of PWM outputs is up to 14* in PWM mode 1 and up to 12* in PWM mode 2.
When PWM Mode 1 and PWM Mode 2 are mixed, up to 20* outputs are possible.
* Varies depending on the product

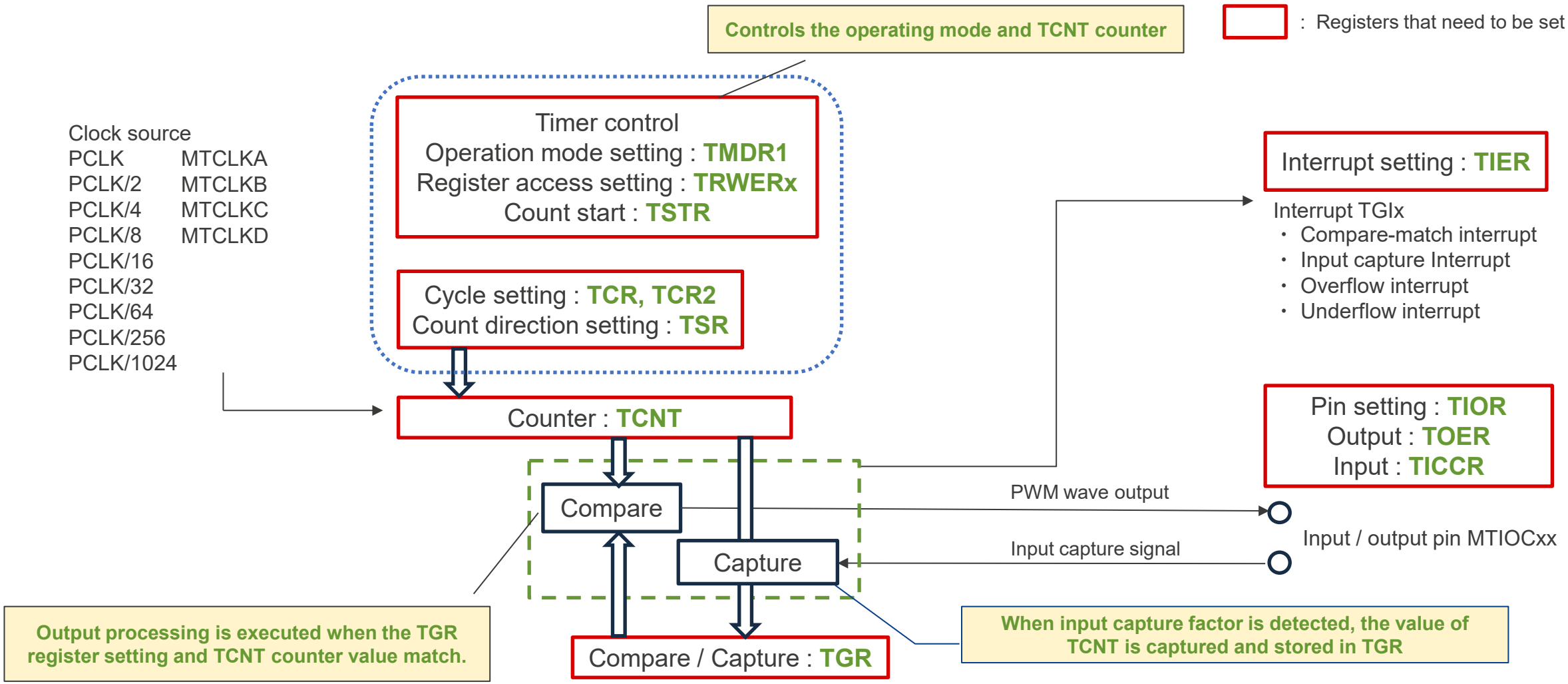


PWM mode 1 (MTU3, MTU4, MTU6, MTU7) : Total 8 outputs

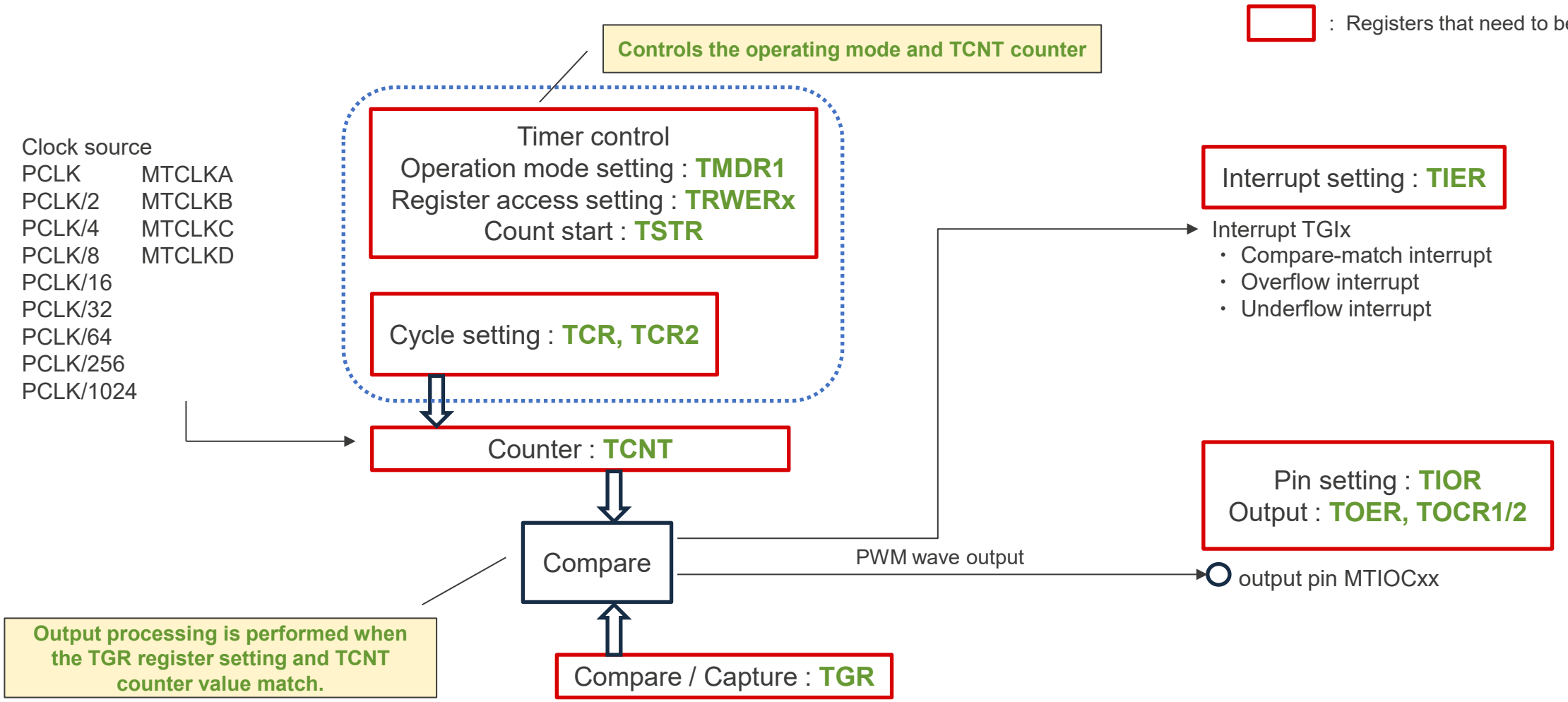


PWM mode 2 (MTU0~MTU2, MTU9) : Total 12 outputs
***TCNT count clearing is MTUn.TGRA of mode 1 operating**
n = 3, 4, 6, 7

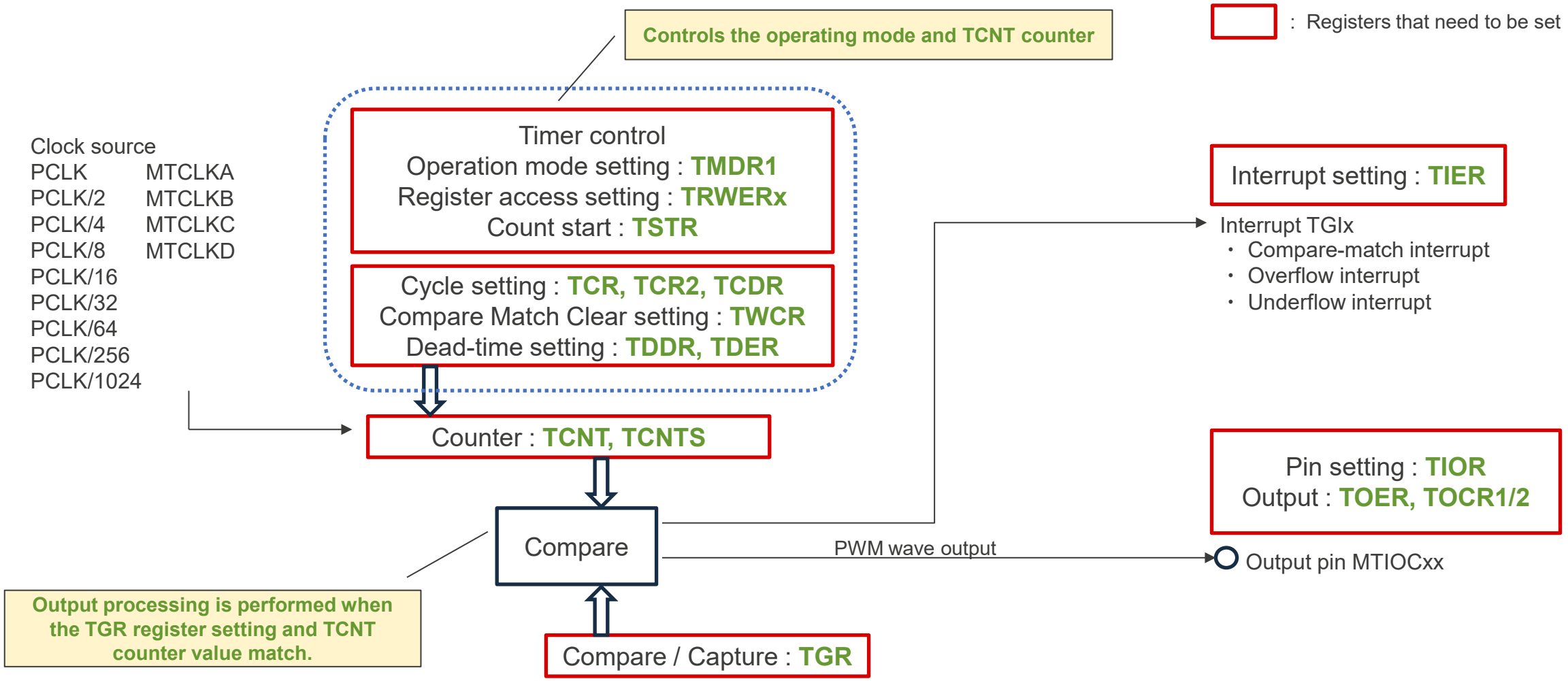
NORMAL MODE, PWM MODE 1/2 RELATED REGISTERS



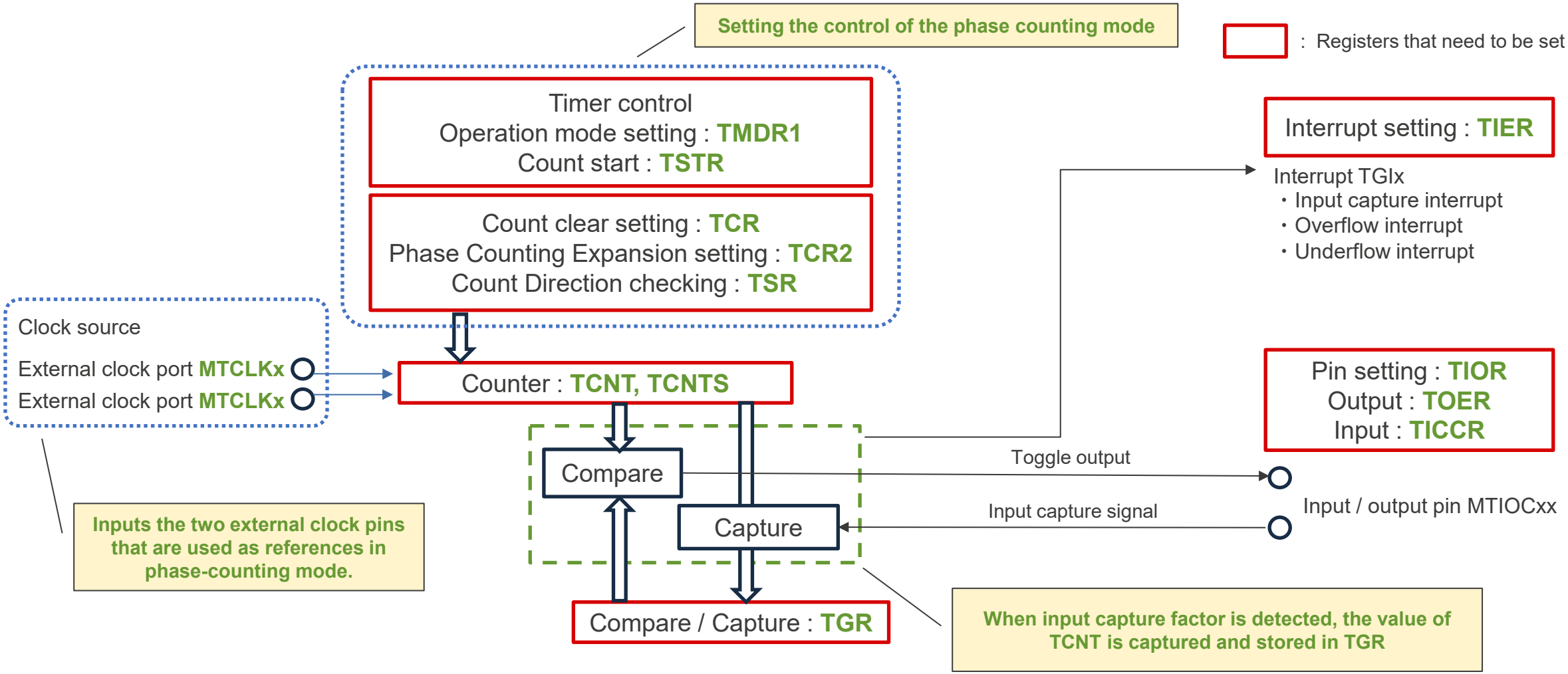
RESET-SYNCHRONIZED PWM MODE RELATED REGISTERS



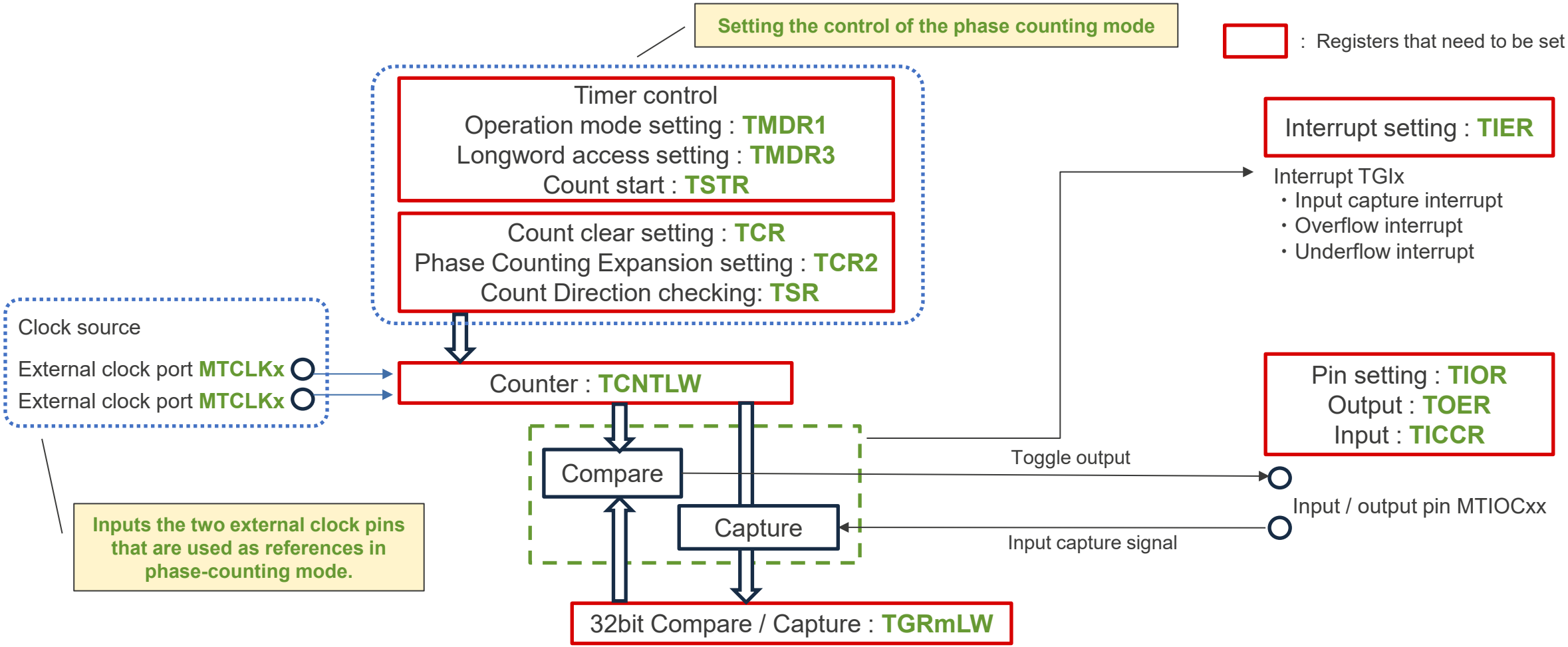
COMPLEMENTARY PWM MODE RELATED REGISTERS



PHASE COUNTING MODE RELATED REGISTERS



PHASE COUNTING MODE (CASCADE OPERATION) RELATED REGISTERS



* TCR, TCR2, TMDR and TIOR registers are enabled for MTU1 only.

BUFFER OPERATION RELATED REGISTERS (1/2)

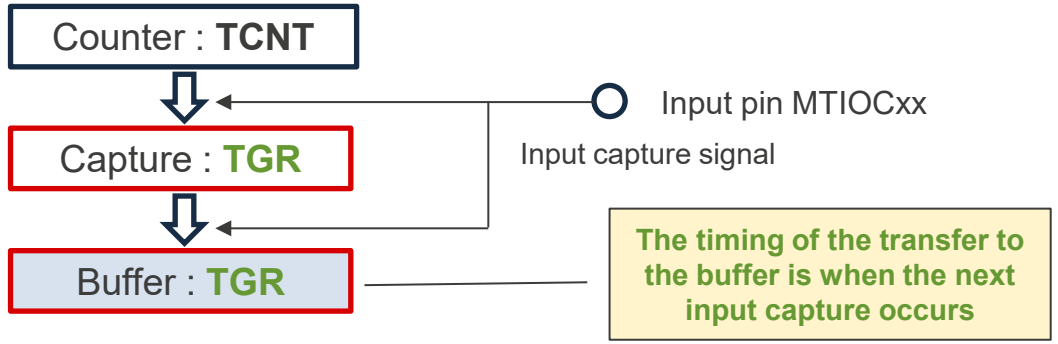
: Registers that need to be set

: Buffer registers

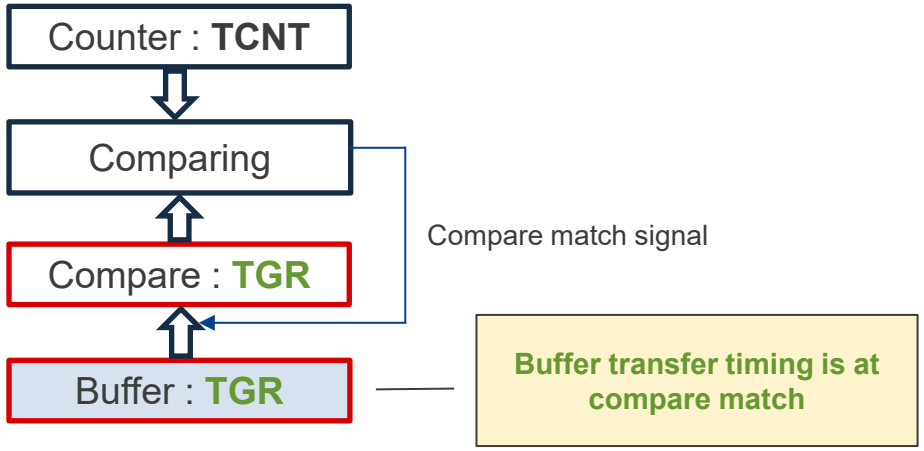
■ Common settings for normal mode, PWM mode 1/2 and phase counting mode

Timer control
Buffer setting : **TMDR1**

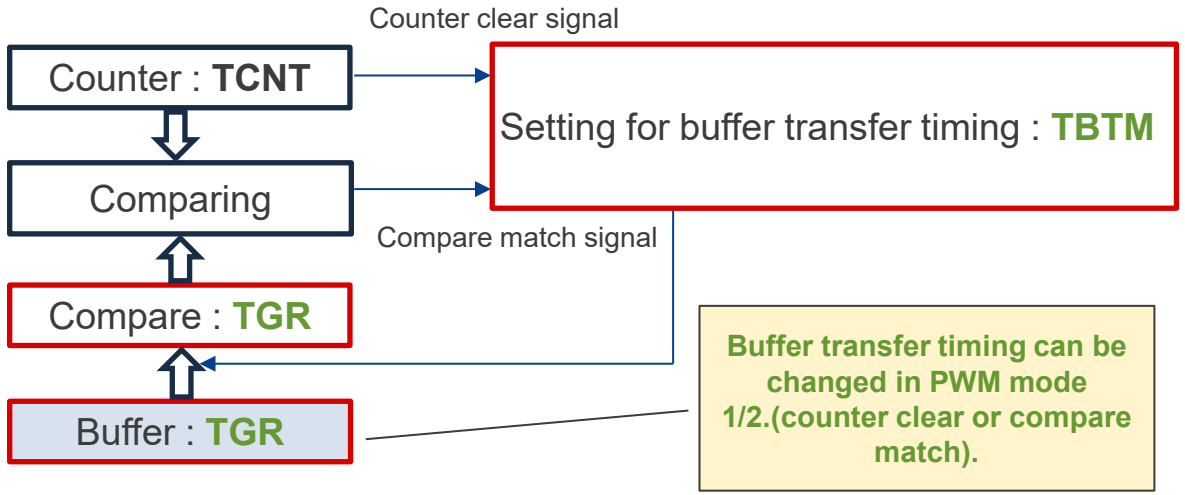
■ Input capture buffer operation (Normal mode, PWM mode 1/2, phase counting mode)



■ Output compare buffer operation (normal mode, phase counting mode)



■ Buffered operation of output compare (PWM mode 1/2)

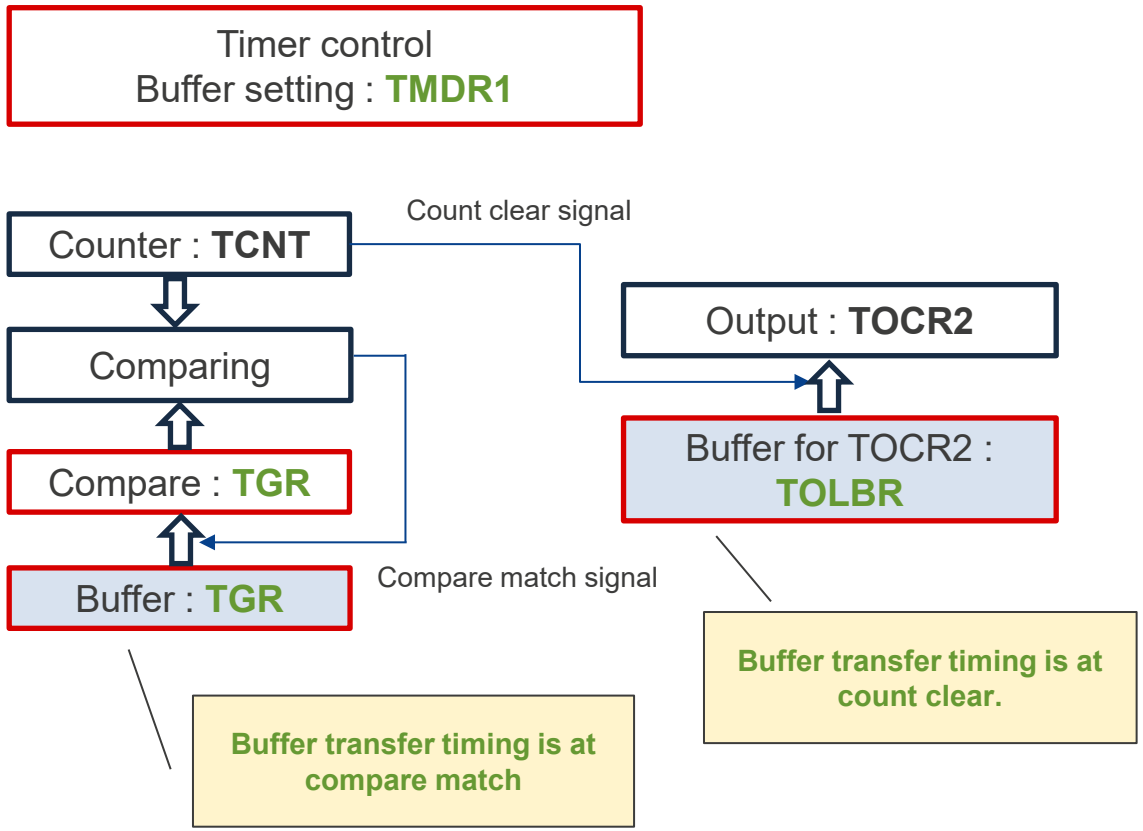


BUFFER OPERATION RELATED REGISTERS (2/2)

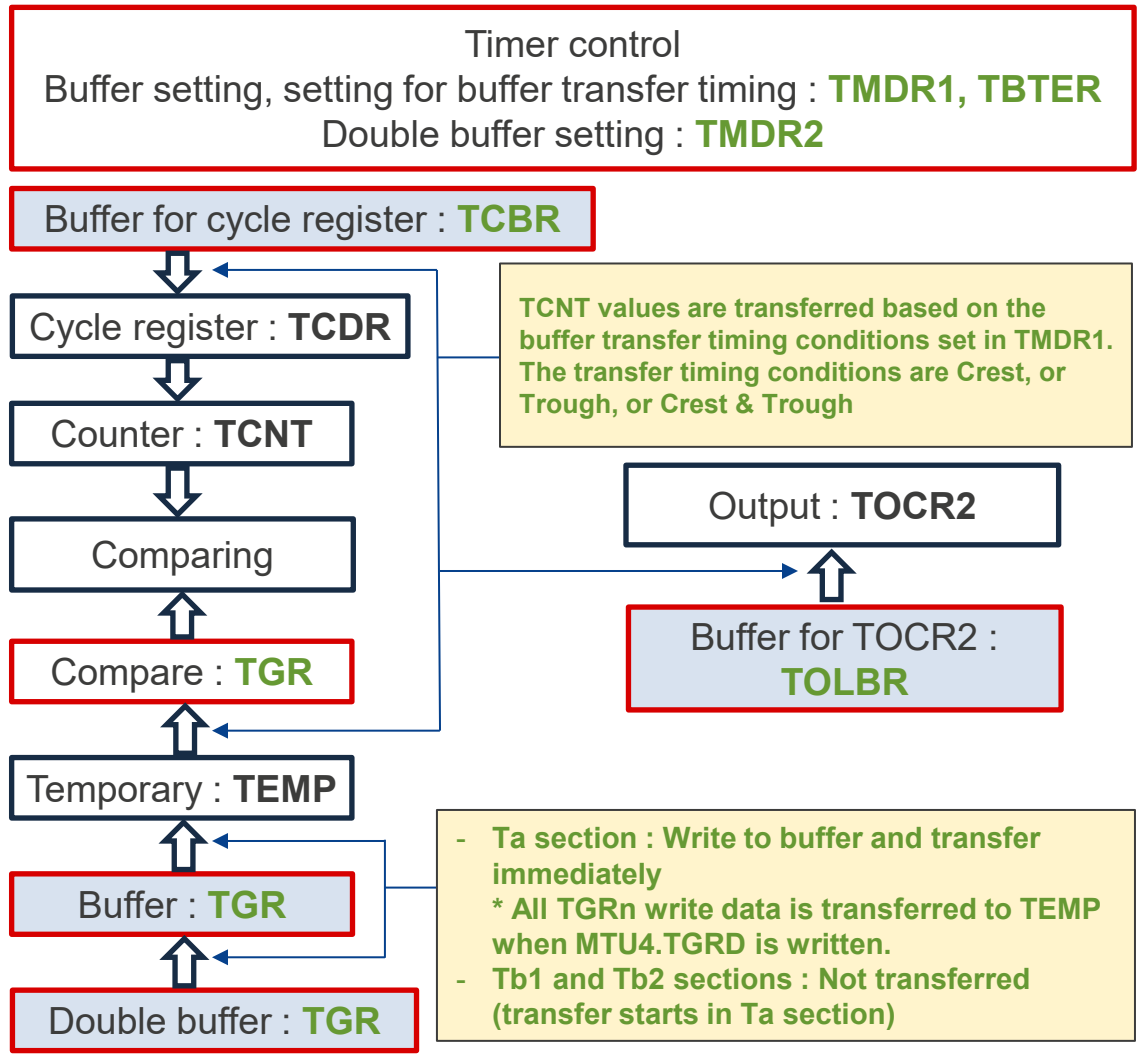
: Registers that need to be set

: Buffer registers

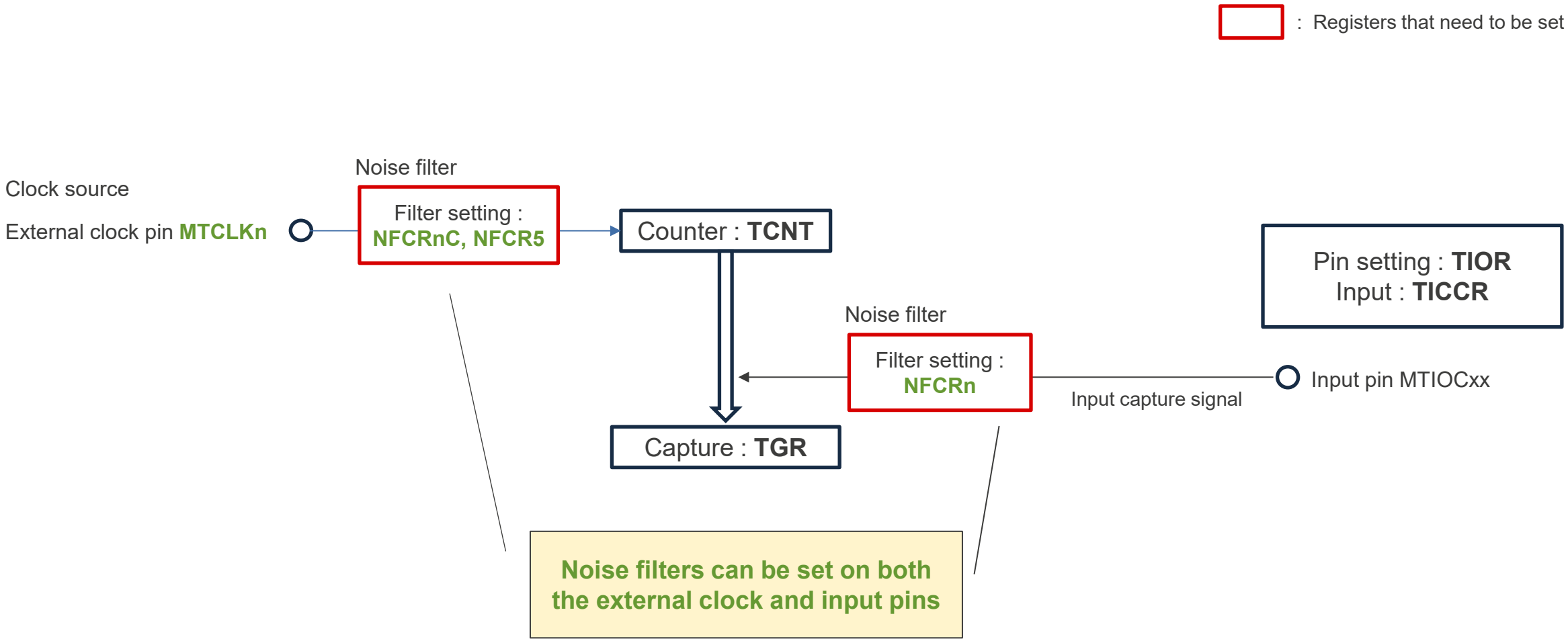
■ Buffer operation in reset synchronous PWM mode



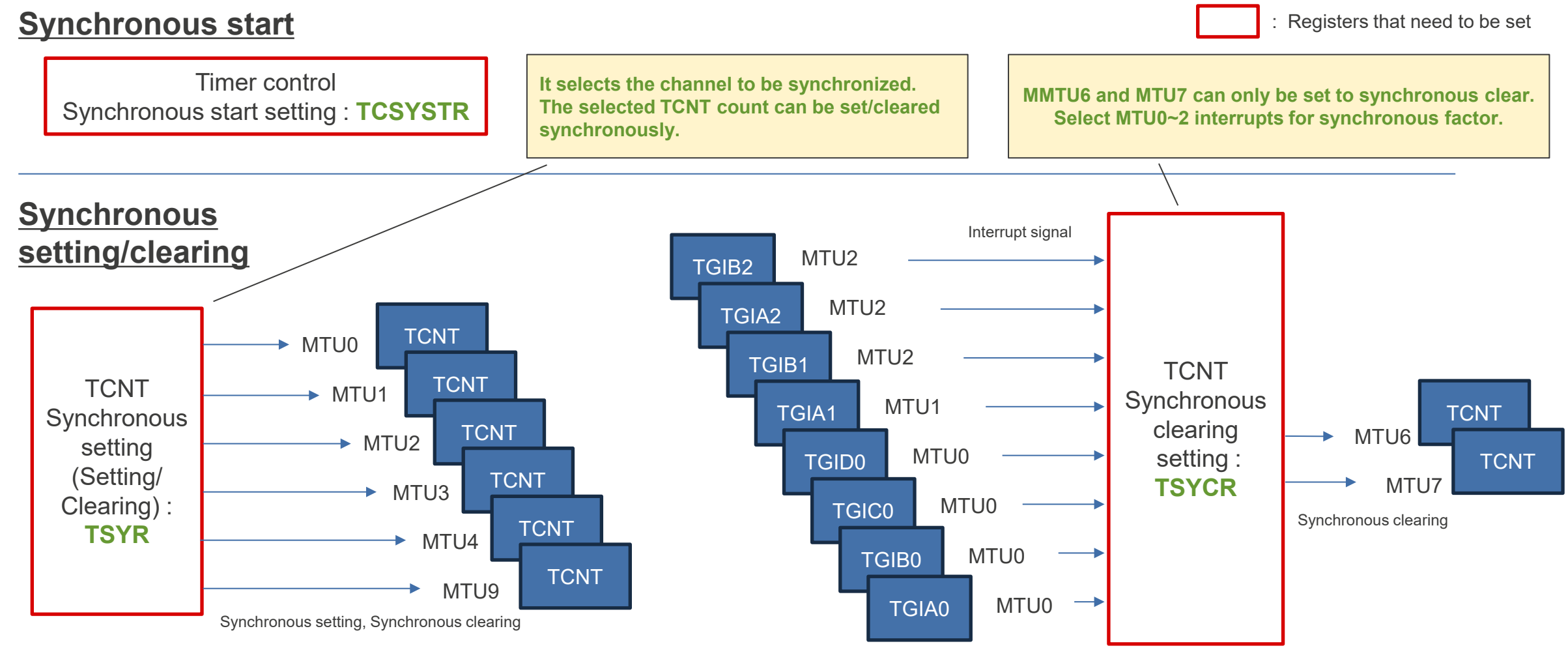
■ Buffer operation in complementary PWM mode



NOISE FILTER RELATED REGISTERS (COMMON TO ALL MODES)



SYNCHRONOUS OPERATION RELATED REGISTERS
(NORMAL MODE, PWM MODE 1/2, RESET SYNCHRONOUS PWM MODE)



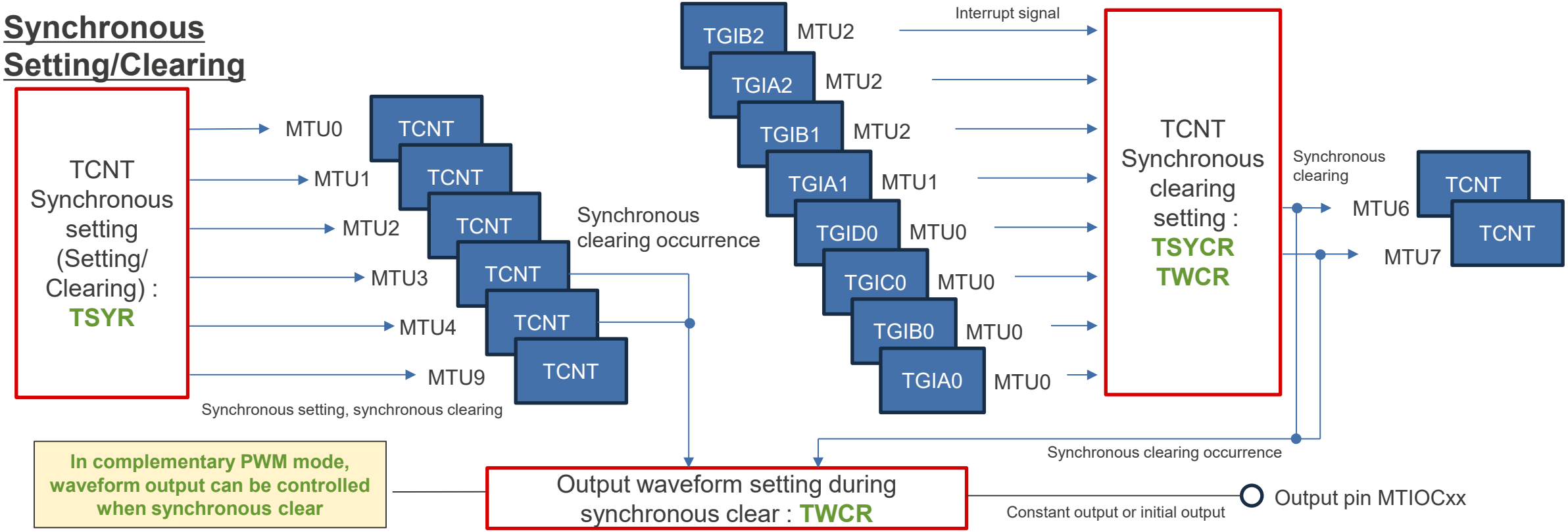
SYNCHRONOUS OPERATION RELATED REGISTERS
(COMPLEMENTARY PWM MODE)

Synchronous start

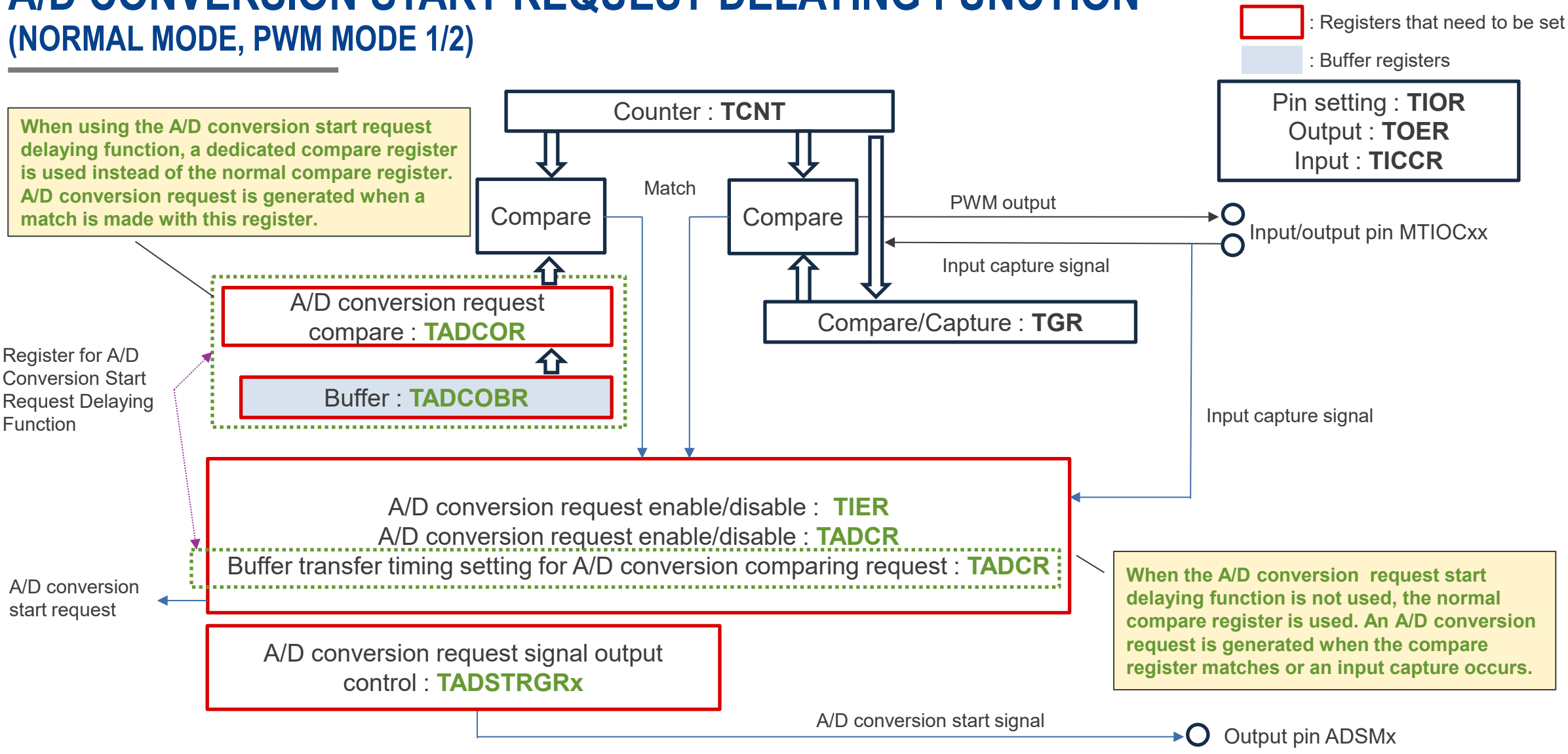
: Registers that need to be set

Timer control
Synchronous start setting : **TCSYSTR**

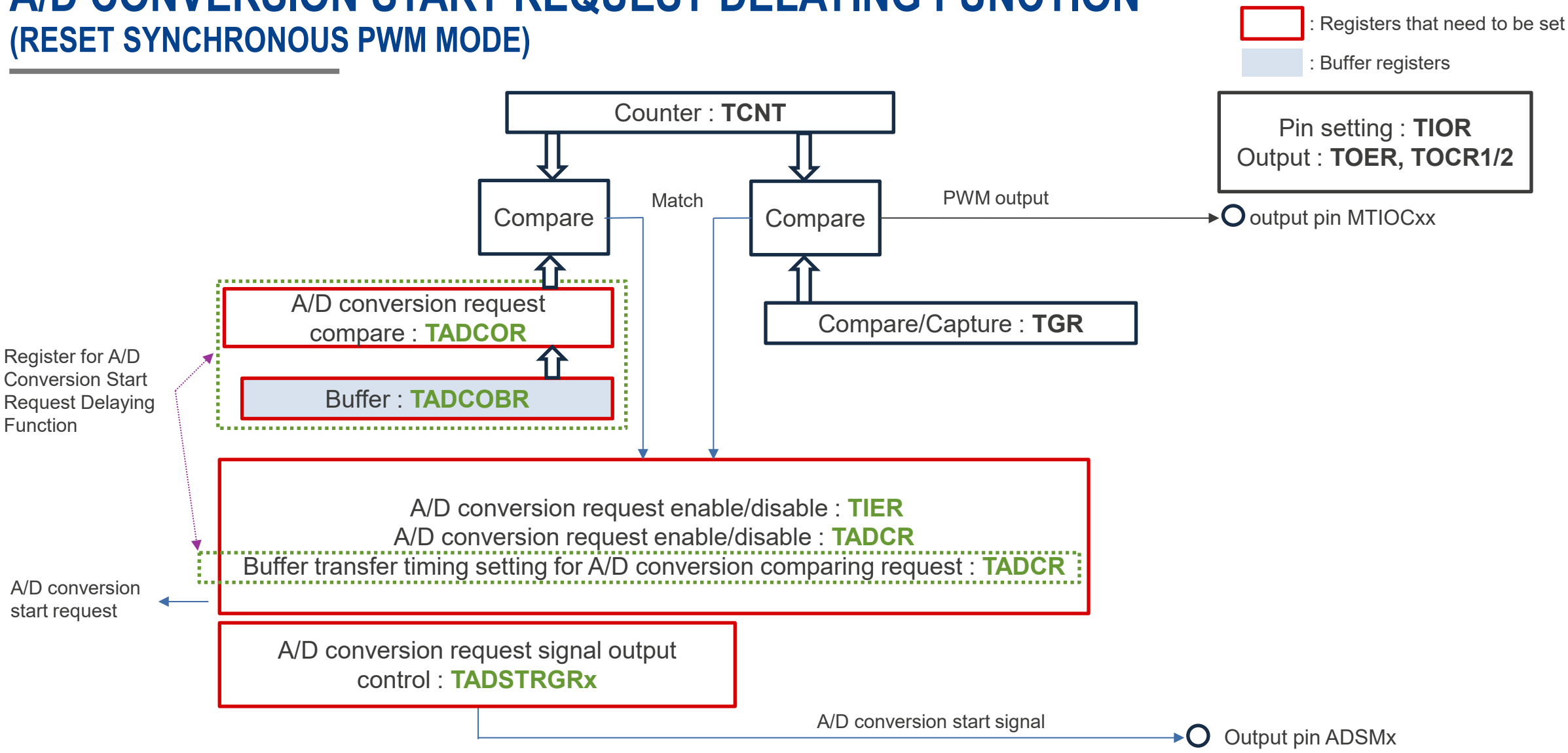
Synchronous Setting/Clearing



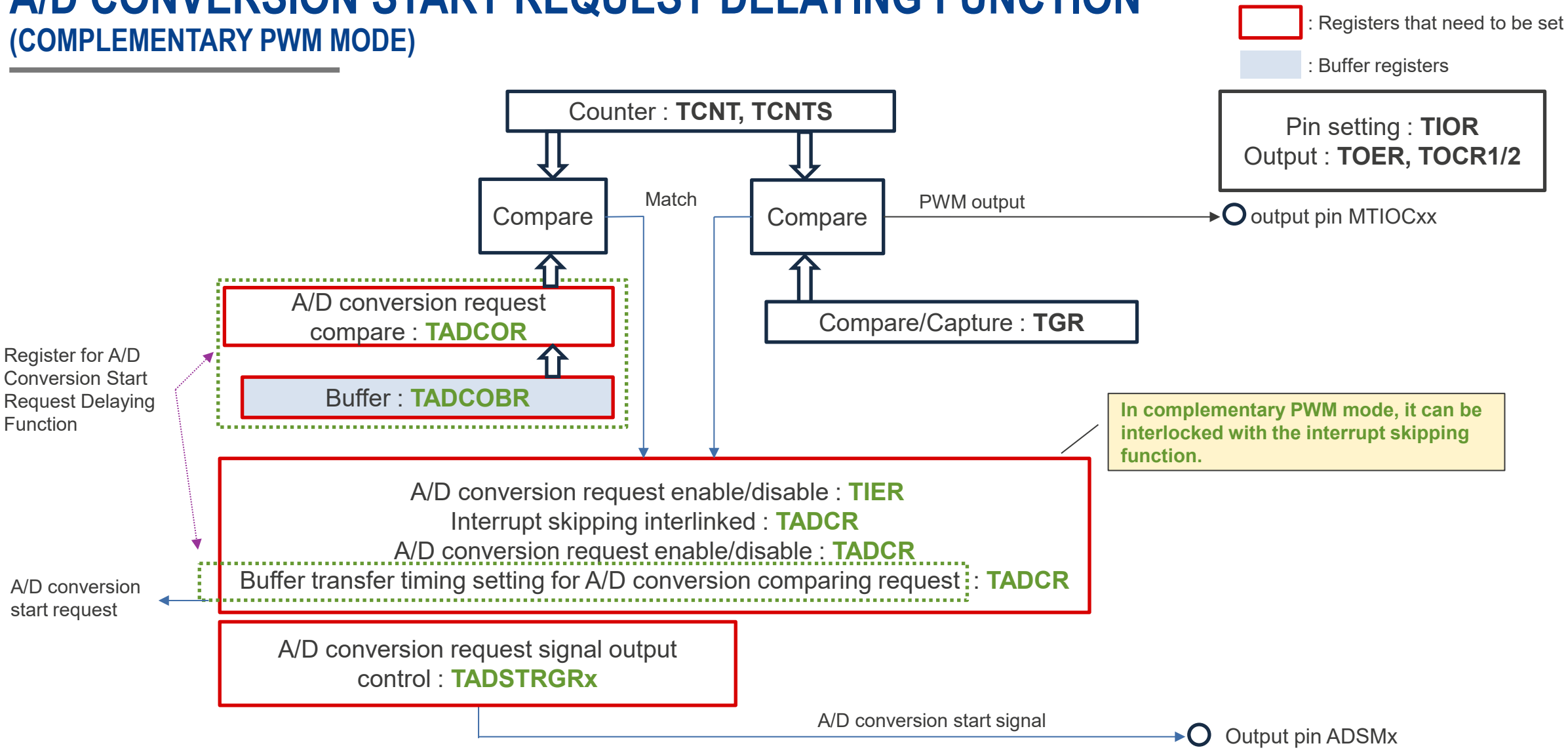
A/D CONVERSION START REQUEST, A/D CONVERSION START REQUEST DELAYING FUNCTION (NORMAL MODE, PWM MODE 1/2)



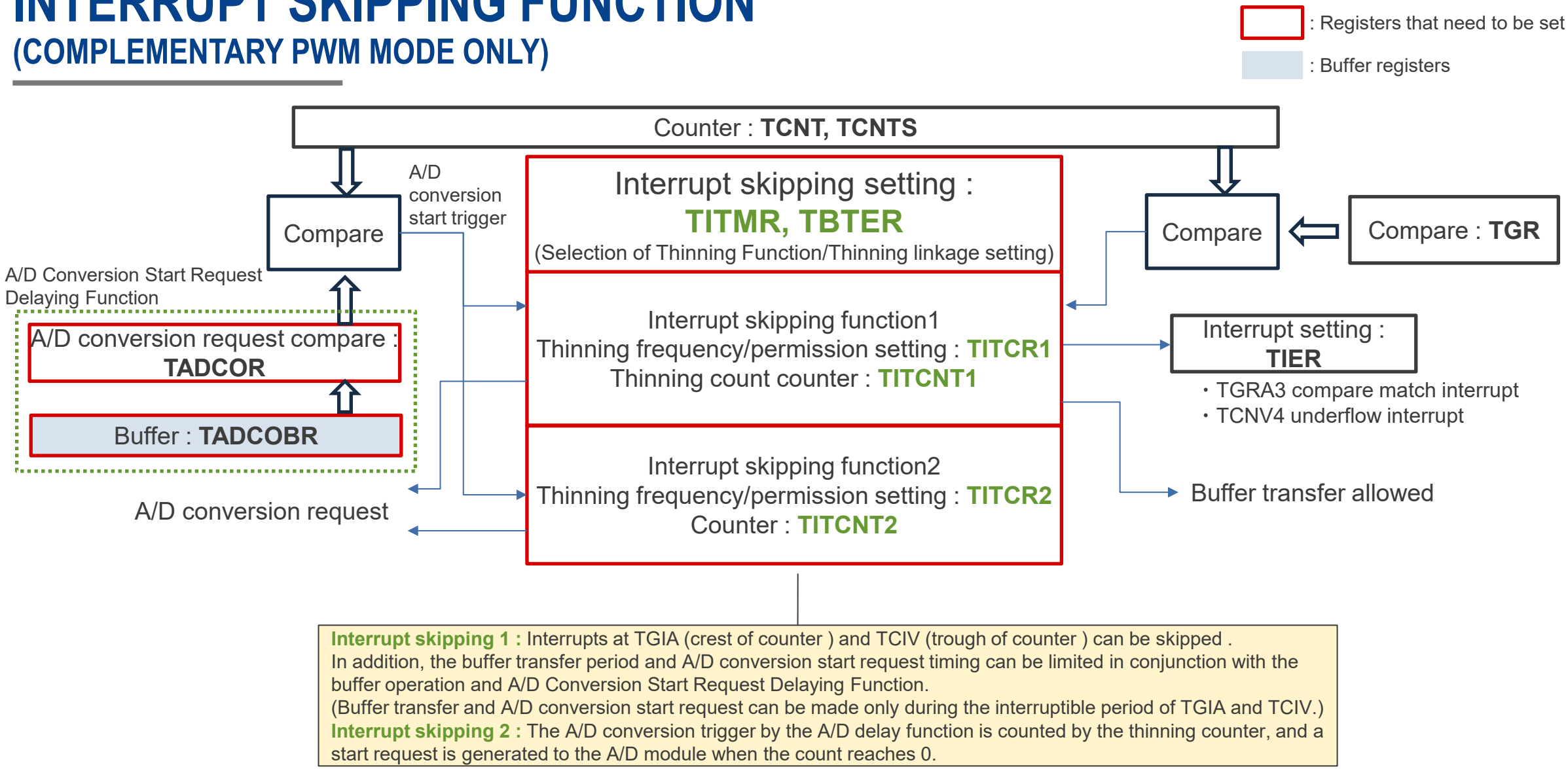
A/D CONVERSION START REQUEST, A/D CONVERSION START REQUEST DELAYING FUNCTION (RESET SYNCHRONOUS PWM MODE)



A/D CONVERSION START REQUEST, A/D CONVERSION START REQUEST DELAYING FUNCTION (COMPLEMENTARY PWM MODE)



INTERRUPT SKIPPING FUNCTION
(COMPLEMENTARY PWM MODE ONLY)



LIST OF APPLICATION NOTES USING MTU

Please refer to the following for more detailed MTU usage.

- RX Family PWM Output Methods Using MTU3/GPTW R01AN5995
- RX Family Example of Operation Near PWM 0% and 100% Duty Cycles Using MTU3 or GPTW R01AN6539
- A/D conversion start request delayed function using MTU3/GPTW R01AN6643
- RX Family Pulse Period Measurement Using MTU2/MTU3 R01AN6644
- RX Family Pulse Width Measurement Using MTU2/MTU3 R01AN6748
- RX Family Usage Examples for Phase Counting Modes Using MTU3/GPTW R01AN6387
- RX Family Synchronous Operation Using MTU3/GPTW R01AN6282

POE3

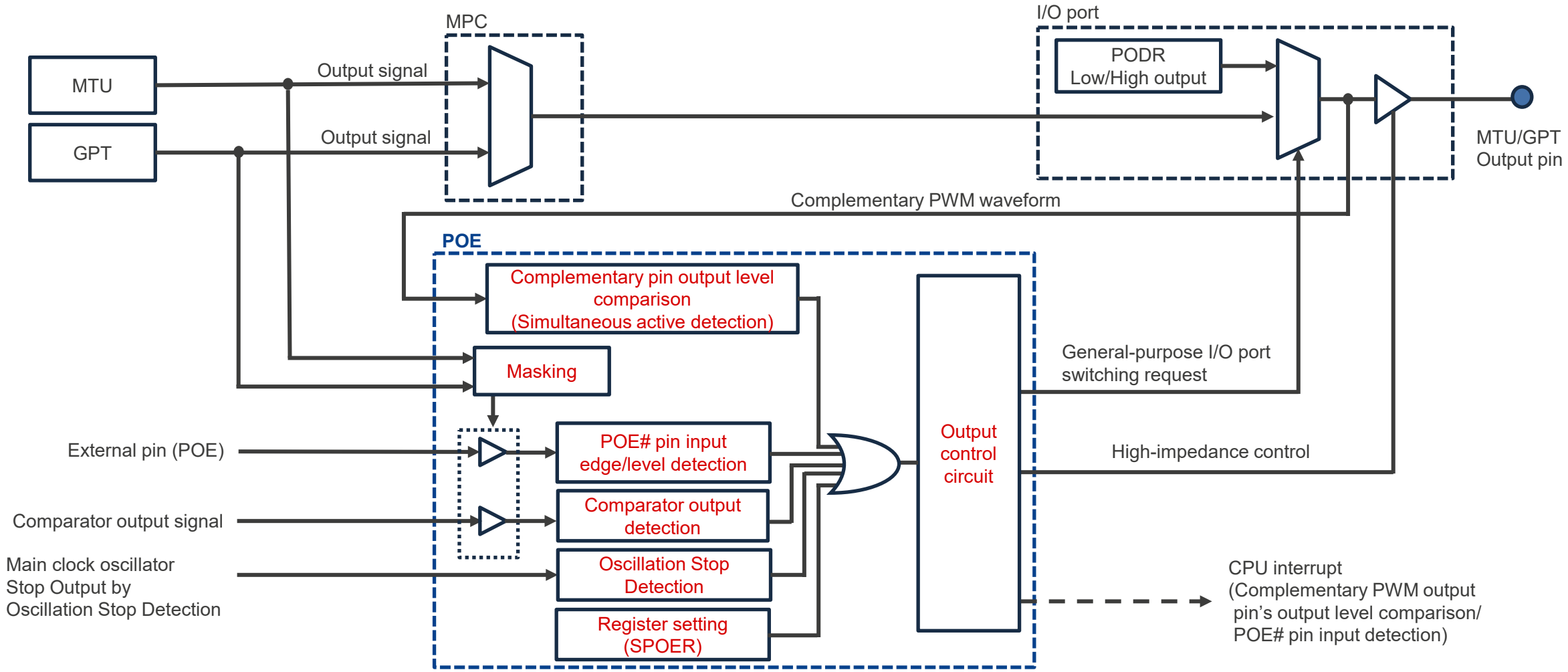
PORT OUTPUT ENABLE(POE) FUNCTION/SPECIFICATIONS

- POE function automatically switches target pins to high-impedance or GPIO ports and disables PWM output upon detection of an output disable condition.
  The following is POE function specification list. (This may vary depending on the product, please refer to the hardware manual for details.)

POE specification		
Status of target pin for Output disable when output disable condition is detected		Switch to high-impedance or general-purpose I/O port
Conditions for disabling the output	Condition 1 : Output-Level Compare of complementary PWM output pin	Positive and reverse phase output levels can be monitored, and a short circuit can be set as a stop condition (refer to P.60 for target terminals)
	Condition 2 : Change of POE# pin input	Trigger input (falling edge or low detection) on POE0, 4, 8~12# pins can be set as stop condition *Rising edge or High detection is also possible by the logic inversion function of the input signal
	Condition 3 : Comparator output detection	Output signal from comparator function can be set as stop condition
	Condition 4 : Oscillation Stop Detection (Main Clock)	Oscillation stop detection of main clock can be set as stop condition
	Condition 5 : Register setting (SPOER)	Stop condition can be generated by software
Target pins for switching to disabling of signal output		Refer to P.60
Target pins for switching to disabling of signal output for each output disable condition	When comparing output levels (output pins short-circuited) of complementary PWM output pins	When the output level comparison result is the MTU pin: All target pins of MTU When the output level comparison result is the GPT pin: All the target pins of GPT
	When POE# pin input is changed	Select from all MTU and GPT target pins
	When comparator output is detected	Select from all MTU and GPT target pins
	When oscillation (Main Clock) stop is detected	Select from all MTU and GPT target pins
	When register setting (SPOER)	Select from all MTU and GPT target pins
Masking of output conditions (POE# pin input change, comparator output detection)		The conditions of “POE# input terminal change” and “comparator output detection” can be masked depending on the MTU/GPT output terminal state. Refer to P.60 for target pins.
Interrupt factor		In case POE# pin is input In case comparing output levels of complementary pin

POE DETECT FUNCTION SYSTEM BLOCK DIAGRAM

- The figure below shows the overall block diagram of the POE detect function system.



OVERVIEW OF MASKING, OUTPUT LEVEL COMPARISON, AND PINS TARGETED FOR STOP CONTROL

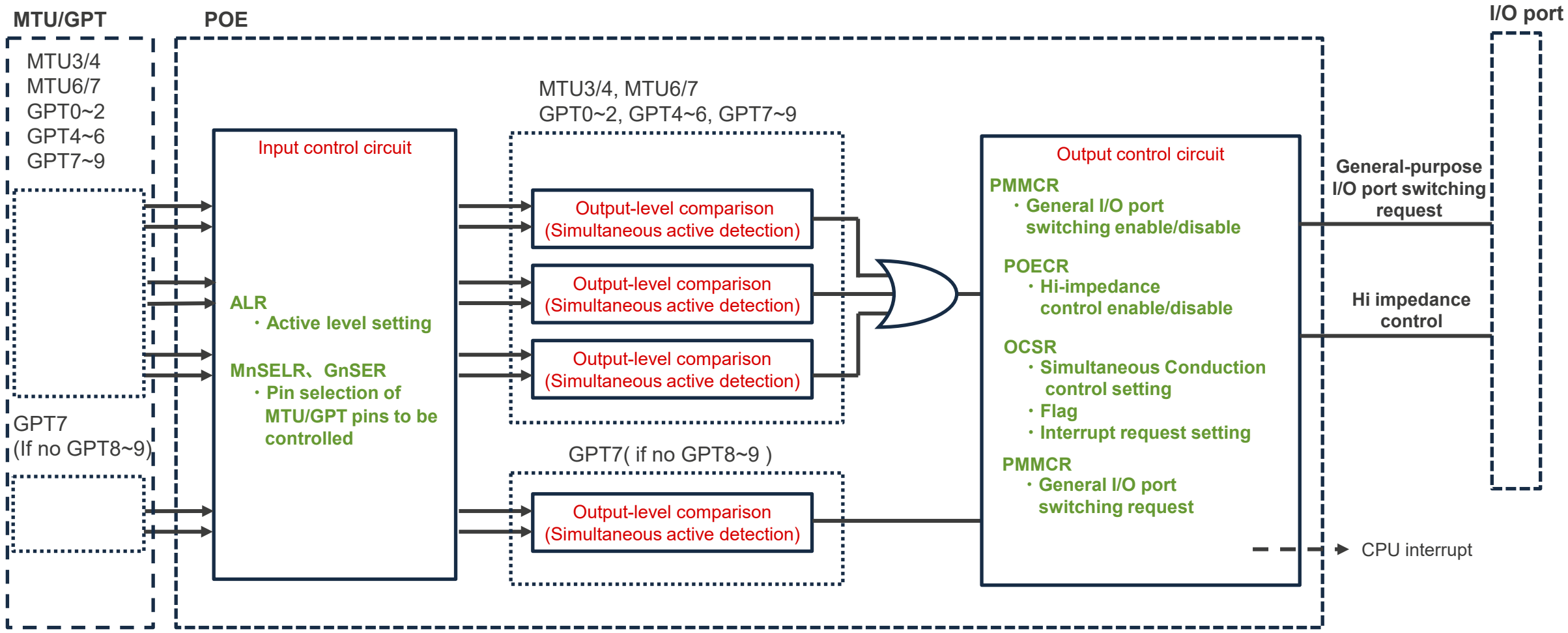
MTU function pins		Masking pins	Output level comparison pins	Stop control target pins
MTU0	MTIOC0A	✓		✓
	MTIOC0B	✓		✓
	MTIOC0C	✓		✓
	MTIOC0D	✓		✓
MTU1	MTIOC1A	✓		
	MTIOC1B	✓		
MTU2	MTIOC2A	✓		
	MTIOC2B	✓		
MTU3	MTIOC3A	✓		
	MTIOC3B	✓	✓	✓
	MTIOC3C	✓		
	MTIOC3D	✓	✓	✓
MTU4	MTIOC4A	✓	✓	✓
	MTIOC4B	✓	✓	✓
	MTIOC4C	✓	✓	✓
	MTIOC4D	✓	✓	✓
MTU6	MTIOC6A	✓		
	MTIOC6B	✓	✓	✓
	MTIOC6C	✓		
	MTIOC6D	✓	✓	✓
MTU7	MTIOC7A	✓	✓	✓
	MTIOC7B	✓	✓	✓
	MTIOC7C	✓	✓	✓
	MTIOC7D	✓	✓	✓

MTU function pins		Masking pins	Output level comparison pins	Stop control target pins
MTU9	MTIOC9A	✓		✓
	MTIOC9B	✓		✓
	MTIOC9C	✓		✓
	MTIOC9D	✓		✓

MTU function pins		Masking pins	Output level comparison pins	Stop control target pins
GPTW0	GTIOC0A	✓	✓	✓
	GTIOC0B	✓	✓	✓
GPTW1	GTIOC1A	✓	✓	✓
	GTIOC1B	✓	✓	✓
GPTW2	GTIOC2A	✓	✓	✓
	GTIOC2B	✓	✓	✓
GPTW3	GTIOC3A	✓		✓
	GTIOC3B	✓		✓
GPTW4	GTIOC4A	✓	✓	✓
	GTIOC4B	✓	✓	✓
GPTW5	GTIOC5A	✓	✓	✓
	GTIOC5B	✓	✓	✓
GPTW6	GTIOC6A	✓	✓	✓
	GTIOC6B	✓	✓	✓
GPTW7	GTIOC7A	✓	✓	✓
	GTIOC7B	✓	✓	✓
GPTW8	GTIOC8A	✓	✓	✓
	GTIOC8B	✓	✓	✓
GPTW9	GTIOC9A	✓	✓	✓
	GTIOC9B	✓	✓	✓

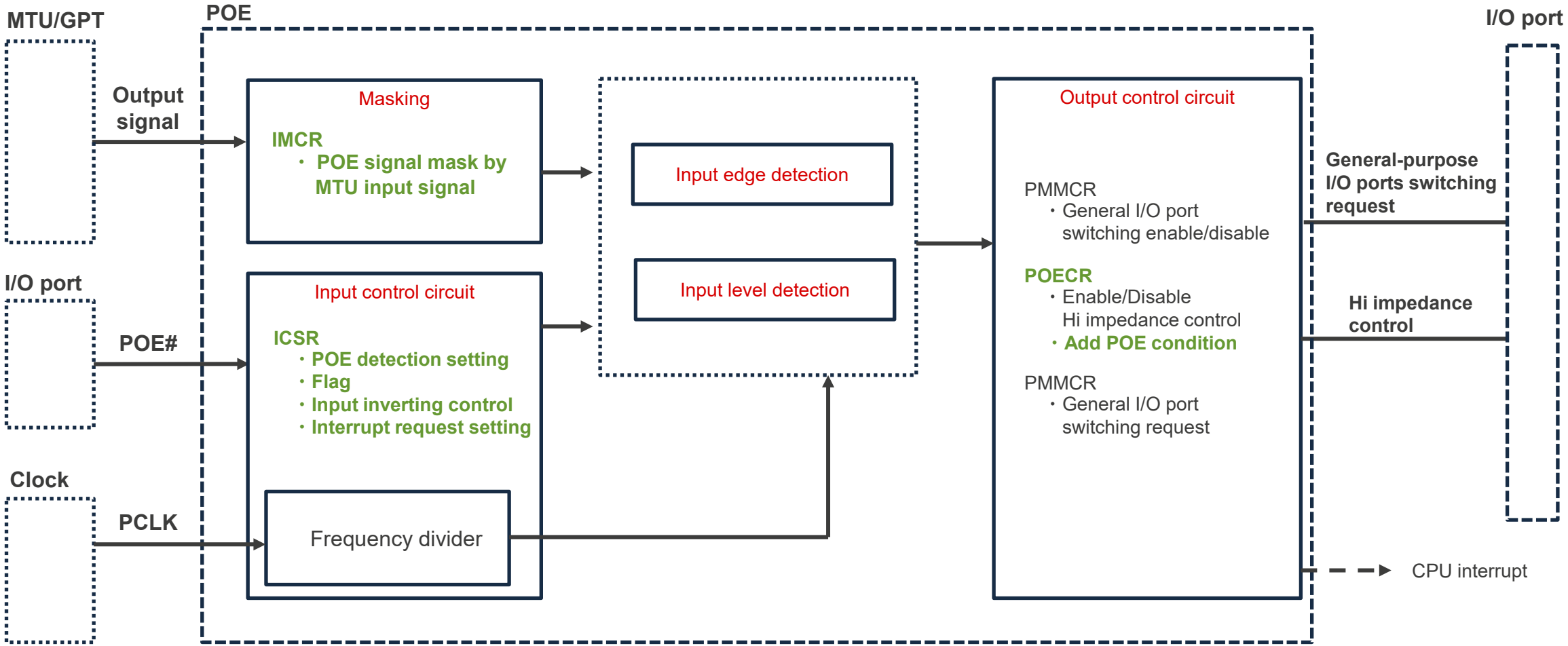
IN CASE OF USING THE OUTPUT LEVEL COMPARISON OF THE COMPLEMENTARY PWM OUTPUT PIN AS THE STOP CONDITION

- The control block and registers that need to be set when the output level comparison of the complementary PWM output pins is used as the stop condition are shown below.



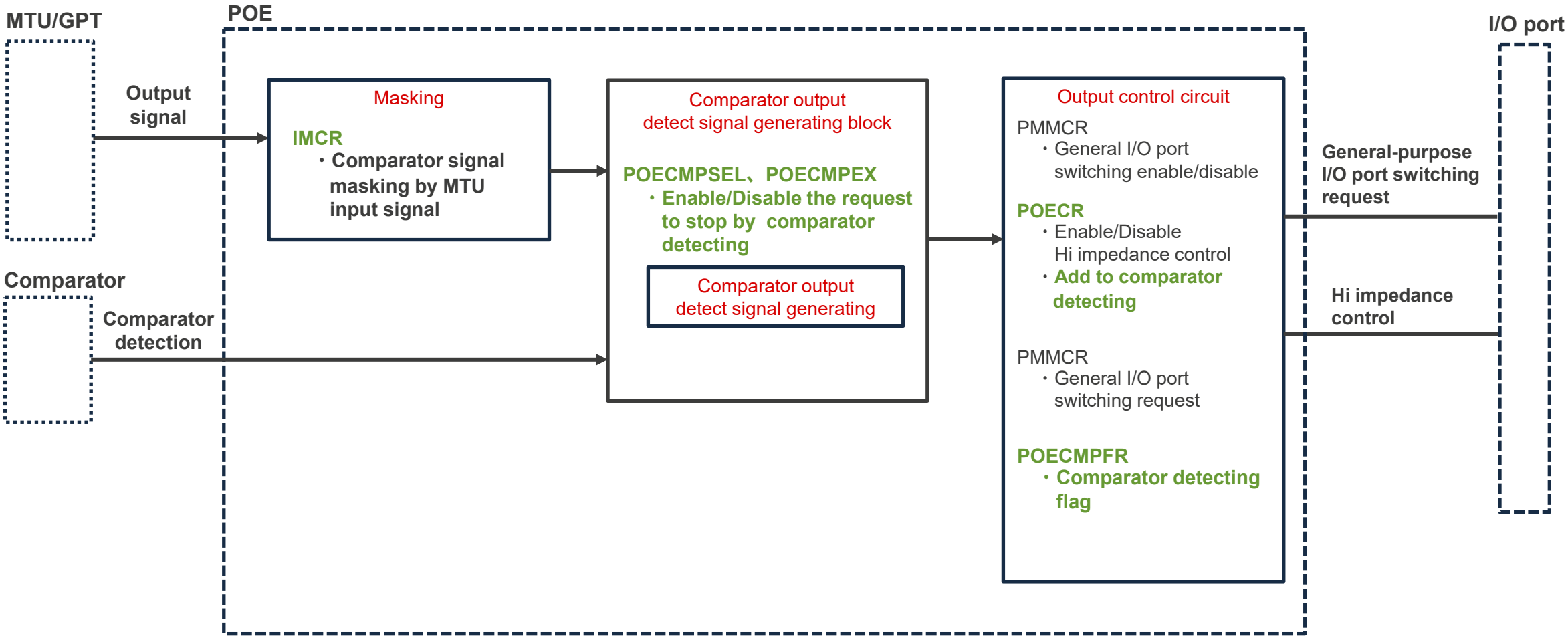
IN CASE OF USING THE POE# PIN INPUT AS THE STOP CONDITION

- The control block and registers that need to be set when the POE# pin input is used as the stop condition are shown below.



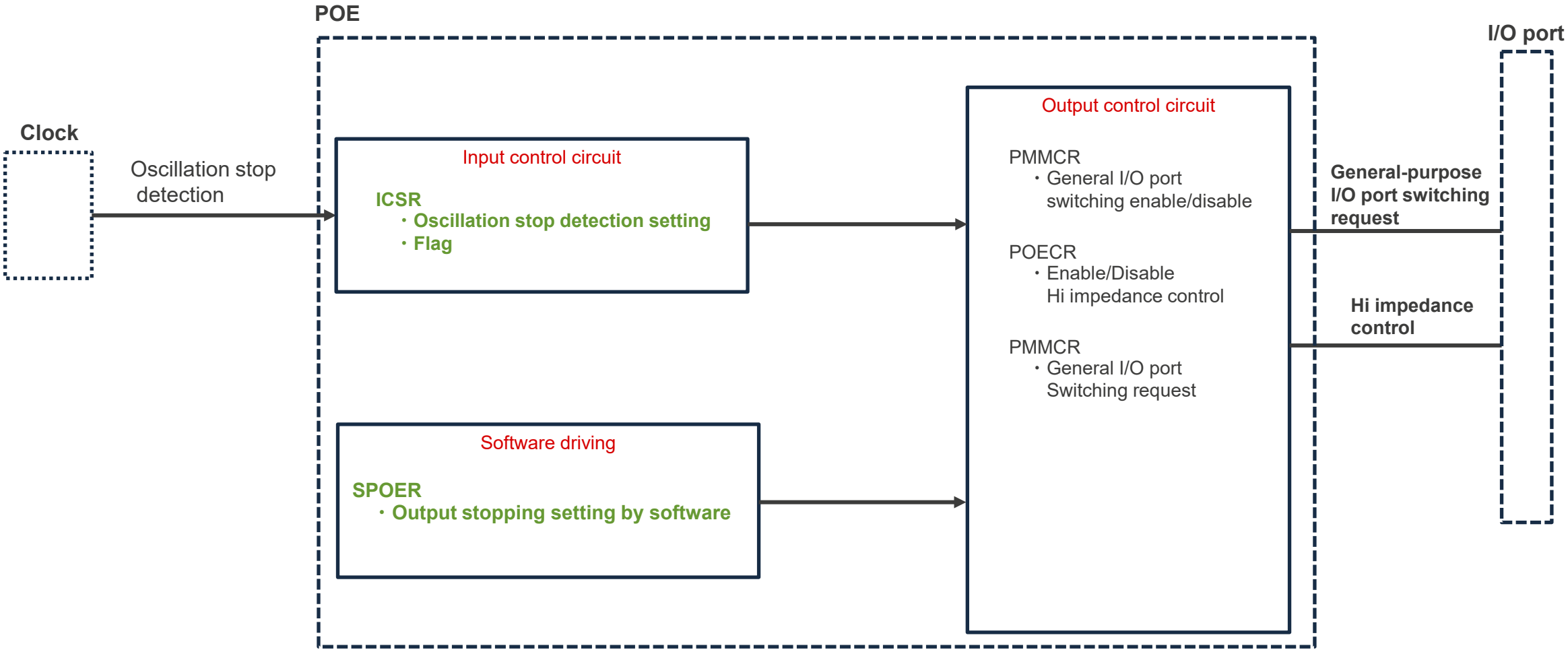
IN CASE OF USING THE COMPARATOR OUTPUT SIGNAL AS THE STOP CONDITION

- The control block and registers that need to be set when the comparator output signal is used as the stop condition are shown below.



IN CASE OF OSCILLATION STOP DETECTION SIGNAL AND CONTROL BY SOFTWARE

- The control block and registers that need to be set when the oscillation stop detection signal or software is used as the stop condition are shown below.



CONDITIONS FOR RELEASE OF POE OUTPUT

■ **The pins with output stopped by input level detection are released by one of the follows.**

- Reset to initial state
- Clear ICSRn.POExF flag

* When set to Low sampling by ICSRn.POExM[3:0] bits, only after a High is input from the POEn# pin and a High is detected, writing '0' to the flag is invalid and the flag is not set to '0'.

■ **The pins with output stopped due to output short-circuit detection are released by one of the follows.**

- Reset to initial state
- Set OCSRn.OSFn flag to "0"

* Note that just writing 0 to a flag is ignored (the flag is not set to 0); the flags can be cleared by writing 0 to it only after setting the pin to the inactive level. In the MTU, the inactive level (initial output level) can be output by stopping the count operation. In the GPTW, the inactive level can be output in accordance with the procedure described in 'Pin Initialization Due to Error during Operation'.

■ **The pins with output stopped by comparator output detection are released by one of the follows**

- Reset to initial state
- Set POECMPFR.CjFLAG flag(j = 0 ～ 5) to "0"

* When setting the POECMPFR.CjFLAG flag to "0", make sure that the analog input signal that was detected by the comparator output has returned to the appropriate value after confirming by performing an A/D conversion, etc. Note that if the flag is cleared without confirming that the analog input signal has returned to the appropriate value and the analog input signal remains higher than the reference voltage when the comparator is inverting output, or remains lower than the reference voltage when the comparator is inverting output, the POECMPFR.CjFLAG flag described above will not become "1" again.

■ **The pins with output stopped by oscillation stop detection are released by one of the follows**

- Reset to initial state
- Set SYSTEM.OSTDSR.OSTDF flag to "0" , then after that set ICSR6.OSTSTF flag to "0"

NOTES

- Depending on the product, the MTU3, 4, 6, and 7 pins are set to high impedance after resetting, or the output condition using some POE pins is enabled. Some products also enable the high-impedance state even when the MTU3, 4, 6, and 7 pins are not used (when the corresponding pins are used as other modules). If this is a problem, the corresponding POECR2 and MnSER registers must be cleared after reset. Please refer to the hardware manual for details.

LIST OF APPLICATION NOTES USING POE

Please refer to the following for more detailed POE usage.

- RX24U Group Disabling and Restoring PWM Output Using POE3A and MTU3d R01AN4140
- RX63T Group POE PWM Output Cutoff by Comparator Detection R01AN1404

WATCHDOG TIMER/ INDEPENDENT WATCHDOG TIMER (WDT/IWDT)

THE FUNCTIONAL COMPARISON OF WDT AND IWDT

WDT
IWDT

The functional comparison between WDT and IWDT is as follows.

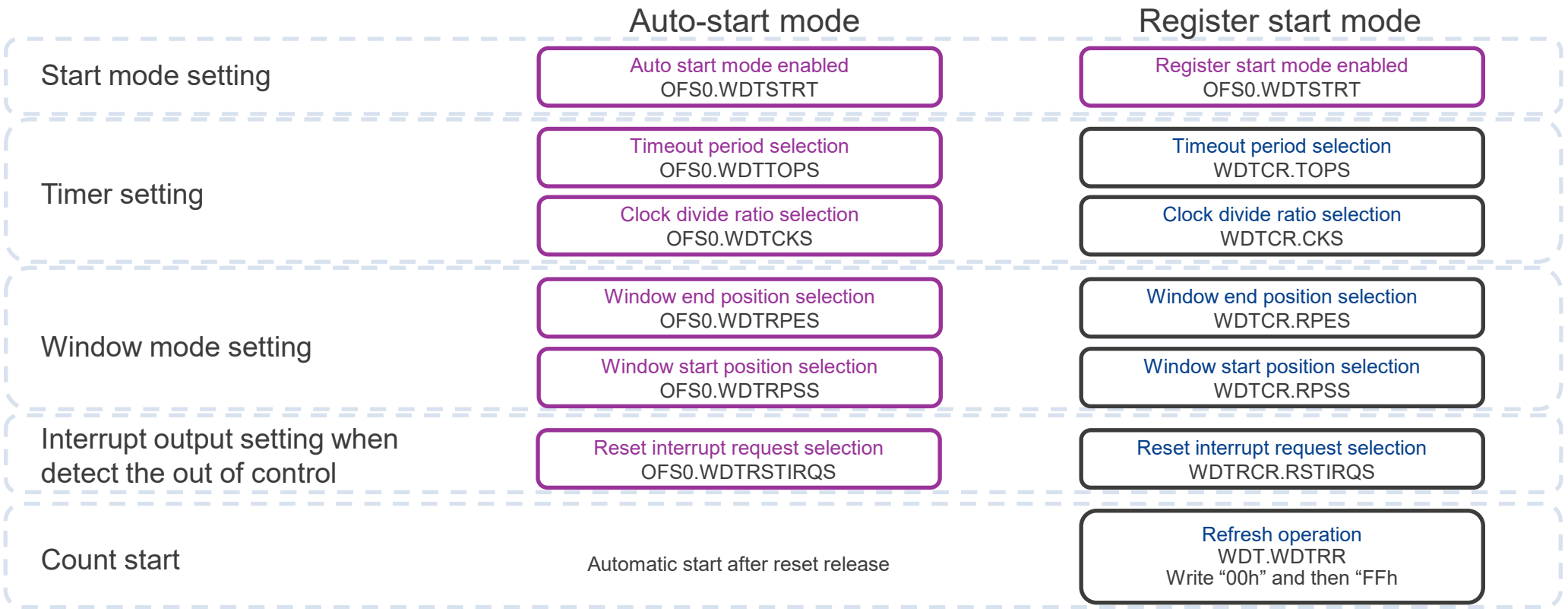
Blue text : Functional difference

Item	WDT	IWDT
Count source	Peripheral module clock	**IWDT-dedicated clock (IWDTCLK)**
Clock divide ratio	4/64/128/512/2048/8192	**1/16/32/64/128/256**
Conditions for starting the counter	・Auto-start mode: Counting automatically starts after a reset is released ・Register start mode: Counting is started by refresh operation	・Auto-start mode: Counting automatically starts after a reset is released ・Register start mode: Counting is started by refresh operation
Conditions for stopping the counter	・Reset ・In low power consumption states ・A counter underflows or a refresh error occurs (only in register start mode)	・Reset **・In low power consumption states(Depends on register setting)** ・A counter underflows or a refresh error occurs (only in register start mode)
Window function	Yes	Yes
Reset output sources	・Down-counter underflows ・Refreshing outside the refresh-permitted period (refresh error)	・Down-counter underflows ・Refreshing outside the refresh-permitted period (refresh error)
Non-maskable interrupt/ interrupt sources	・Down-counter underflows ・Refreshing outside the refresh-permitted period (refresh error)	・Down-counter underflows ・Refreshing outside the refresh-permitted period (refresh error)
Reading the counter value	Enable	Enable
Event link function	No	**Yes**
Output signal (Internal signal)	・Reset output ・Interrupt request output	・Reset output ・Interrupt request output **・Sleep mode count stop control output**
Operations during low power consumption mode	Stop	**Run / Stop (Selectable)**

WDT SETTINGS LIST

The watchdog timer has two modes of operation: auto start mode and register start mode.
Each mode is operated by setting registers in the Option Setting Memory (OFS) or WDT.
The following figure shows the registers that need to be set in each operation mode.

: OFS setting
: WDT setting



IWDT SETTINGS LIST

IWDT

IWDT has two modes of operation: auto start mode and register start mode.
Each mode is operated by setting registers in the Option Setting Memory (OFS) or IWDT.
The following figure shows the registers that need to be set in each operation mode.

: OFS setting
: WDT setting

	Auto-start mode	Register start mode
Start mode setting	Auto start mode enabled OFS0.IWDTSTRT	Register start mode enabled OFS0.IWDTSTRT
Timer setting	Timeout period selection OFS0.IWDTTOPS[1:0]	Timeout period selection IWDTCR.TOPS[1:0]
	Clock divide ratio selection OFS0.IWDTCKS[3:0]	Clock divide ratio selection IWDTCR.CKS[3:0]
Window mode setting	Window end position selection OFS0.IWDTRPES[1:0]	Window end position selection IWDTCR.RPES[1:0]
	Window start position selection OFS0.IWDTRPSS[1:0]	Window start position selection IWDTCR.RPSS[1:0]
Interrupt output setting when detect the out of control	Reset interrupt request selection OFS0.IWDTRSTIRQS	Reset interrupt request select IWDTRCR.RSTIRQS
Count stop setting in sleep mode	Sleep Mode Count Stop Control OFS0.IWDTSLCSTP	Sleep Mode Count Stop Control IWDTCSTPR.SLCSTP
Count start	Automatic start after reset release	Refresh Operation IWDT.IWDTRR Write “00h” and then “FFh

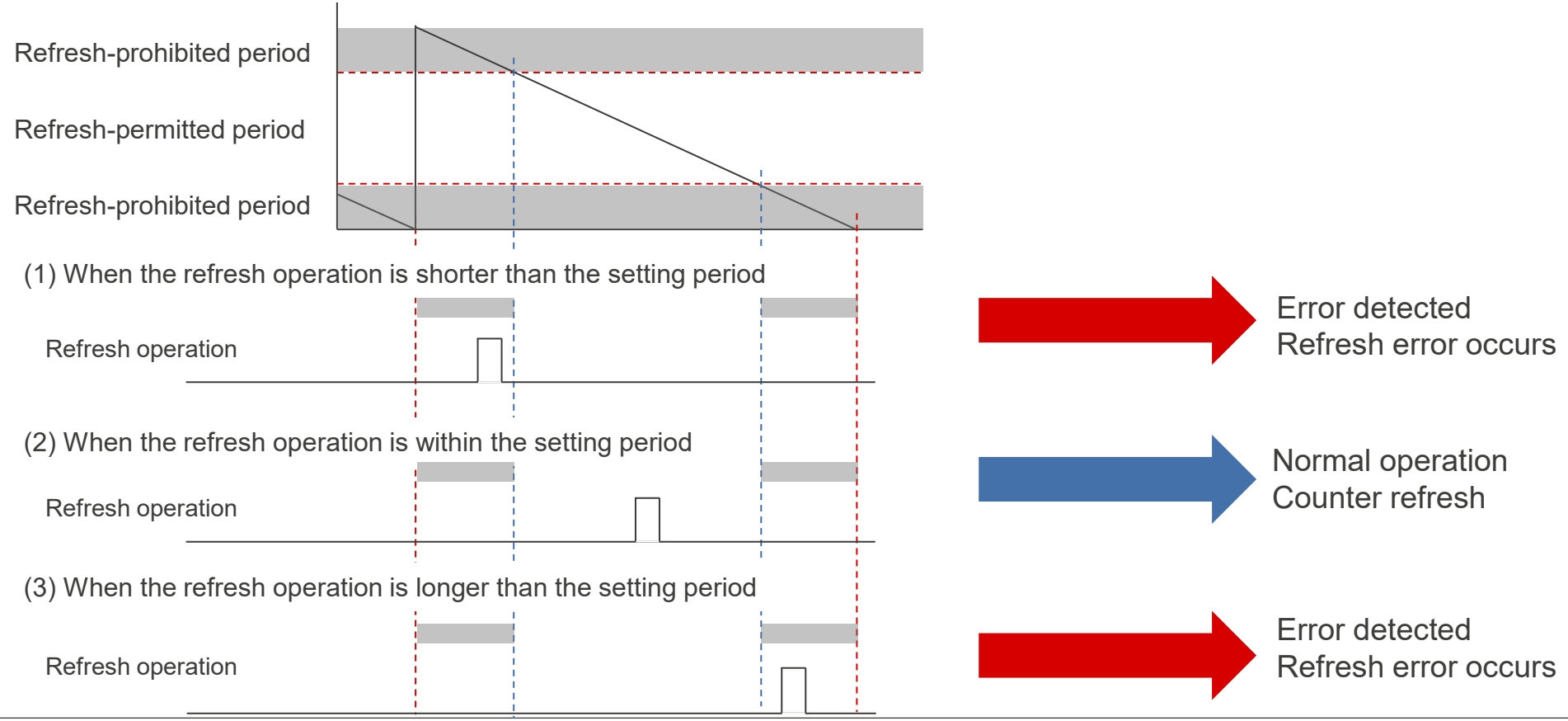
FUNCTION EXPLANATION : WINDOW MODE

WDT
IWDT

In window mode, set the permitted and prohibited periods for refresh operations.

If a refresh operation is executed during the prohibited period, a refresh error occurs and the system is judged to be out of control.

In addition to CPU running out of control, this function can also detect deviations in the processing cycle, making it suitable for applications with high periodicity and reliability requirements.



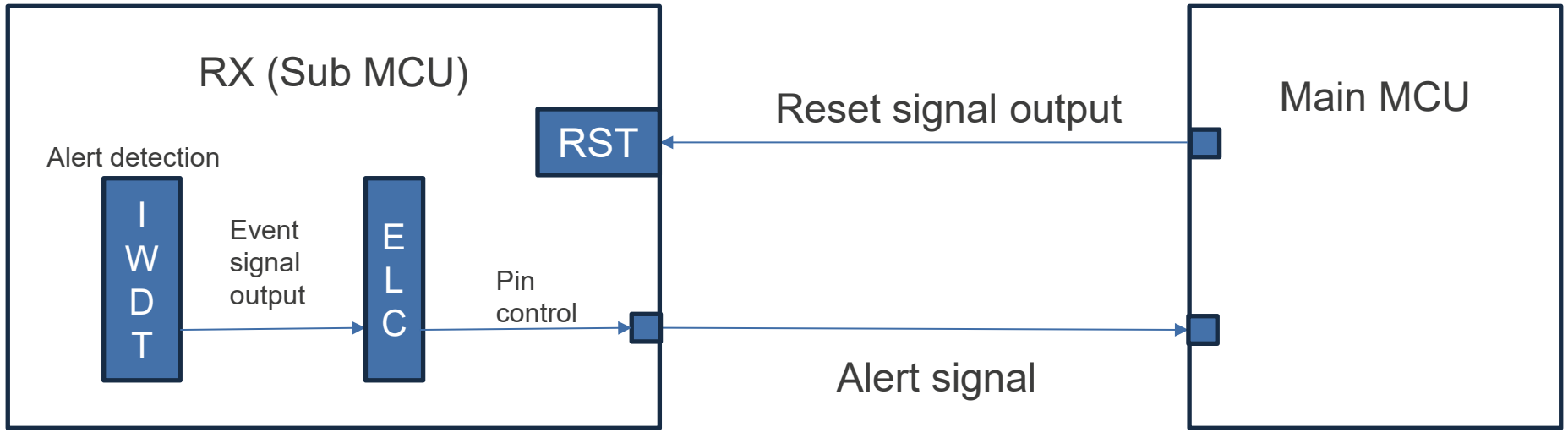
USE CASE :

OUTPUT ERROR ALERT TO EXTERNAL MAIN MCU USING EVENT LINK CONTROLLER (ELC)

IWDT

IWDT can output signals to the ELC when a refresh error or underflow occurs and can output external signals from I/O ports using the ELC without going through the CPU.

The figure below shows how to detect an alert in IWDT, send an alert notification to the external main MCU, and execute a hardware reset from the main MCU. The main MCU can check the status of the sub-MCUs and monitor the number of alerts.



*When using ELC, non-maskable interrupt request or interrupt request output must be enabled (IWDTRCR.RSTIRQS = 0).

NOTE

WDT
IWDT

- When WDT or IWDT is operated, it cannot be stopped except under the count stop condition. The stop conditions are as follows.
  - Reset
  - Low power consumption state
  - In case underflow or refresh error occurs (only in register start mode)

- When debugging on the emulator, the WDT and IWDT counts stop during the break. The stopped counts are restarted during program execution.

REVISION HISTORY

Revision	Date	Page	Contents
1.00	2025/1	-	First edition, issued