32-bit RISC Microcontroller
Reference Manual

12-bit Analog to Digital Converter (ADC-G2)

Revision 1.1

2024-11

Toshiba Electronic Devices & Storage Corporation

Contents

List of Figures

List of Tables

Preface

Related Document

Document name
Datasheet
Input/Output Ports
Exception
Clock Control and Operation Mode
Product Information
Advanced Programmable Motor Control Circuit
Programmable Motor Control Circuit Plus

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-PMD	Advanced Programmable Motor Control Circuit
PMD+	Programmable Motor Control Circuit Plus
TRGSEL	Trigger Selection Circuit

1. Outlines

Refer to the "Electrical Characteristics" section of the data sheet for the range of conversion clock, conversion time, and sampling time.

The number of analog inputs, conversion result registers, and program registers for general-purpose start-up factors varies by product. For details, refer to reference manual "Product Information".

Available PMD triggers and general-purpose triggers vary by product. For details, refer to reference manual "Product Information".

Function classification	Function	Operation explanation
AD conversion	Conversion resolution	12 bits
	Conversion time	- 0.6µs (SCLK: 40MHz) - 0.73µs (SCLK: 30MHz) - 1.0µs (SCLK: 20MHz)
	Sampling time	Selectable from two sampling times for each analog input
Conversion operation	Programmable	- Program registers to set conversion target, conversion sequence and interrupt generation - Program registers for PMD trigger factors and general-purpose factors
Start-up factor	PMD trigger	- 12 triggers - Select one of eight programs per trigger - Up to 4 consecutive conversions in one program - Conversion result storage register is selected in pairs of four
	General-purpose factor	- General-purpose trigger or software factors (single/continuous) - Program register for each conversion result register - Set start-up factors, conversion channels, and interrupt generation in program registers - Conversions specified for the same start-up factor in the program registers are executed in sequence
Status flags	Operation state	- AD operation flag - Program running flag for each trigger
	Conversion result	- AD conversion result storage flag - Overrun flag
Special control	AD monitor function	- 2 channels of big/small comparison function between conversion result and set value - Selectable conversion result storage register to be monitored - Use of two channels allows determination of whether two values are within range - Selectable number of detections - Selectable continuous count and accumulated count - Can be used as PMD protection function
Linked control	Interrupt	- End of conversion interrupt PMD trigger (2 interrupts). General-purpose factor (per factor) - Monitor function interrupt (2 interrupts).
	DMA request	- DMA can be started at the end of conversion of general-purpose factors (per factor)

Figure 1.1 shows the connection of peripheral functions linked to the ADC.

The PMD trigger is input from the "Programmable Motor Control Circuit Plus" or "Advanced Programmable Motor Control Circuit" (hereafter PMD) in synchronization with the motor drive timing.

General-purpose trigger is input from functions such as general-purpose timers.

AD monitoring functions and other interrupts can be used for OVV protection of PMD and for activation of timers.

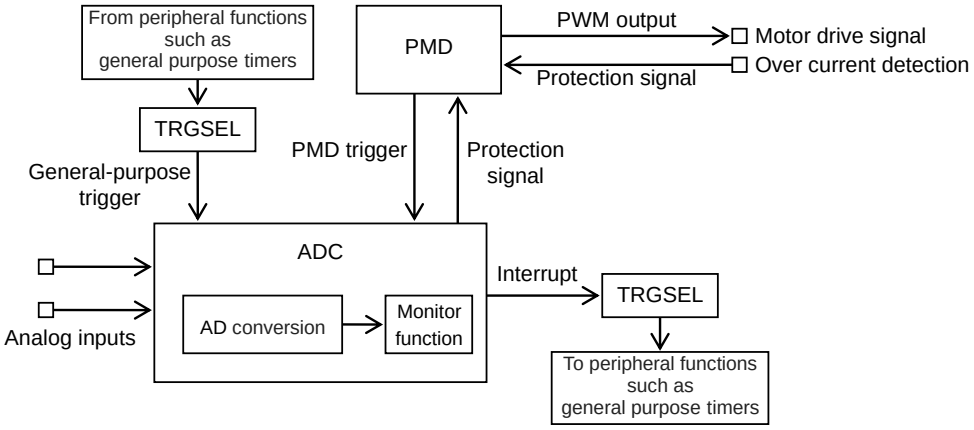


Figure 1.1 Related Figure of ADC Another Peripheral Function

2. Configuration

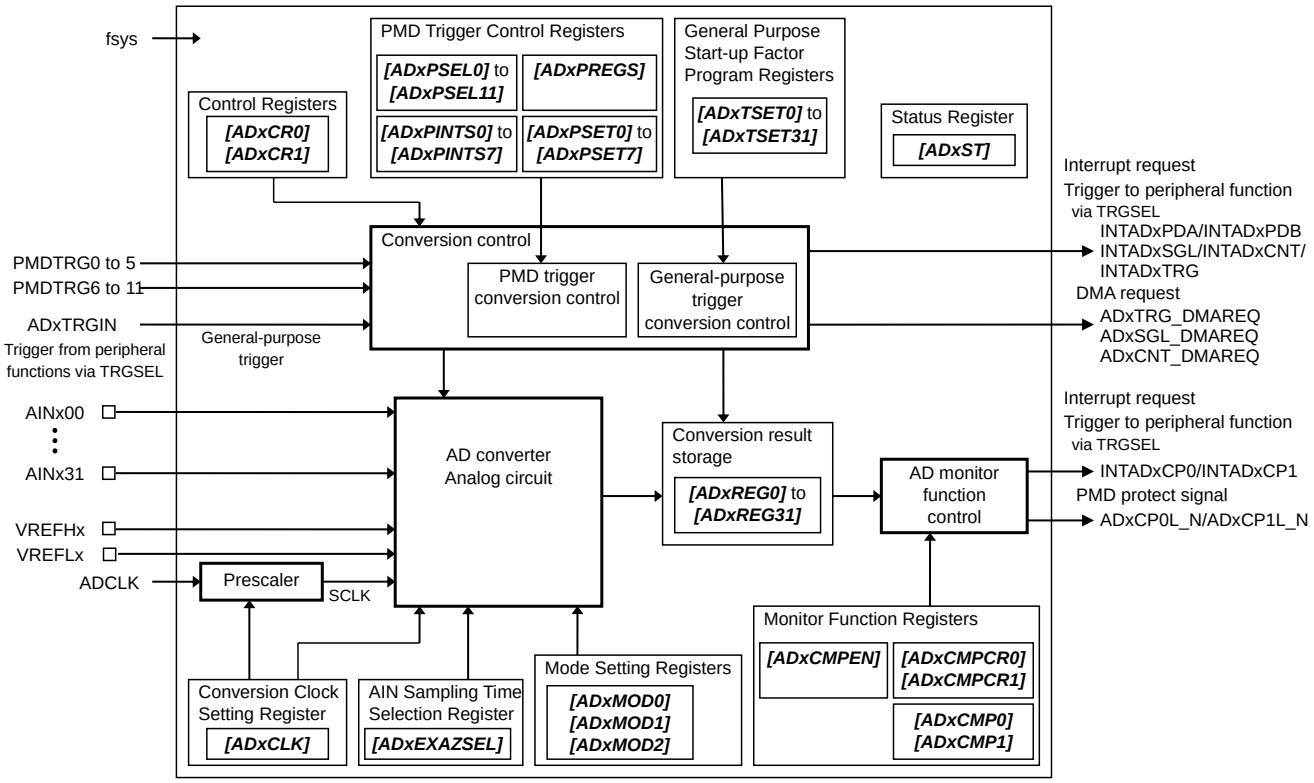


Figure 2.1 ADC Block Diagram

Table 2.1 List of Signals

No	Signal symbol	Signal name	I/O	Related reference manual
1	fsys	System clock	Input	Clock Control and Operation Mode
2	ADCLK	Conversion clock for ADC	Input	Clock Control and Operation Mode
3	AINx00 to AINx31	Analog input pins	Input	Datasheet, Input/Output Ports
4	VREFHx	Reference power pin for analog	Input	Datasheet
5	VREFLx	Reference GND pin for analog	Input	Datasheet
6	PMDTRG0 to 5	PMD triggers	Input	Product Information
7	PMDTRG6 to 11	Triggers from PMD/peripheral functions	Input	Product Information
8	ADxTRGIN	General-purpose trigger	Input	Product Information
9	ADxCP0L_N	Monitor function0 output for PMD protect function	Output	Product Information
10	ADxCP1L_N	Monitor function1 output for PMD protect function	Output	Product Information
11	INTADxPDA	PMD trigger interrupt A	Output	Exception, Product Information
12	INTADxPDB	PMD trigger interrupt B	Output	Exception, Product Information
13	INTADxTRG	General-purpose trigger interrupt	Output	Exception, Product Information
14	INTADxSGL	Single conversion interrupt	Output	Exception, Product Information
15	INTADxCNT	Continuous conversion interrupt	Output	Exception, Product Information
16	INTADxCP0	Monitor function0 interrupt	Output	Exception, Product Information
17	INTADxCP1	Monitor function1 interrupt	Output	Exception, Product Information
18	ADxTRG_DMAREQ	General-purpose trigger DMA request	Output	Product Information
19	ADxSGL_DMAREQ	Single conversion DMA request	Output	Product Information
20	ADxCNT_DMAREQ	Continuous conversion DMA request	Output	Product Information

3. Function and Operation

3.1. Clock Supply

When using ADC, set an applicable clock enable bit to "1" (clock supply) in Clock supply and stop register A for fsys (***[CGFSYSENA]***, ***[CGFSYSMENA]***), Clock supply and stop register B for fsys (***[CGFSYSENB]***, ***[CGFSYSMENB]***), Clock supply and stop register C for fsys (***[CGFSYSMENC]***), and Clock supply and stop register for fc (***[CGFCEN]***). Set the ADC conversion clock enable bit to "1" in Clock supply and stop register for ADC and Debug circuit (***[CGSPCLKEN]***).

Registers and bit positions vary depending on the product. Also, depending on the product, some registers may not exist. For details, refer to the Reference Manual "Clock Control and Operating Mode".

When stopping the clock supply, make sure that AD conversion is stopped. Also, when changing the operation mode to STOP1/STOP2 mode, make sure that AD conversion is stopped.

3.2. Initialization

To use the ADC, initially set ***[ADxMOD0]***<DACON> to "1". Then, wait for 3µs before executing the subsequent operation.

Set ***[ADxMOD2]*** and ***[ADxTRM]***. For setting values, refer to the Reference Manual "Product Information".

Set the SCLK frequency and sampling time with ***[ADxCLK]*** and ***[ADxMOD1]***. Refer to "Electrical Characteristics" in the datasheet for the available conditions.

3.3. Conversion Operation

3.3.1. Conversion Operation by General-purpose Start-up Factor

General-purpose trigger and software factors are available as general-purpose start-up factors.

Software factors include single and continuous conversions.

3.3.1.1. Operation

Conversion by general-purpose start-up factors operates according to the settings in the general purpose start-up factor program registers. The general purpose start-up factor program registers are prepared for each conversion result storage register.

When a start-up factor occurs, the analog inputs specified by the ***[ADxTSETn]***<AINSTn> in all general purpose start-up factor program registers with the same start-up factor are converted, starting with the number with the lowest n.

When the conversion is completed, the conversion result is stored in the corresponding conversion result storage register and an interrupt corresponding to the start-up factor is generated if ***[ADxTSETn]***<ENINTn> is enabled.

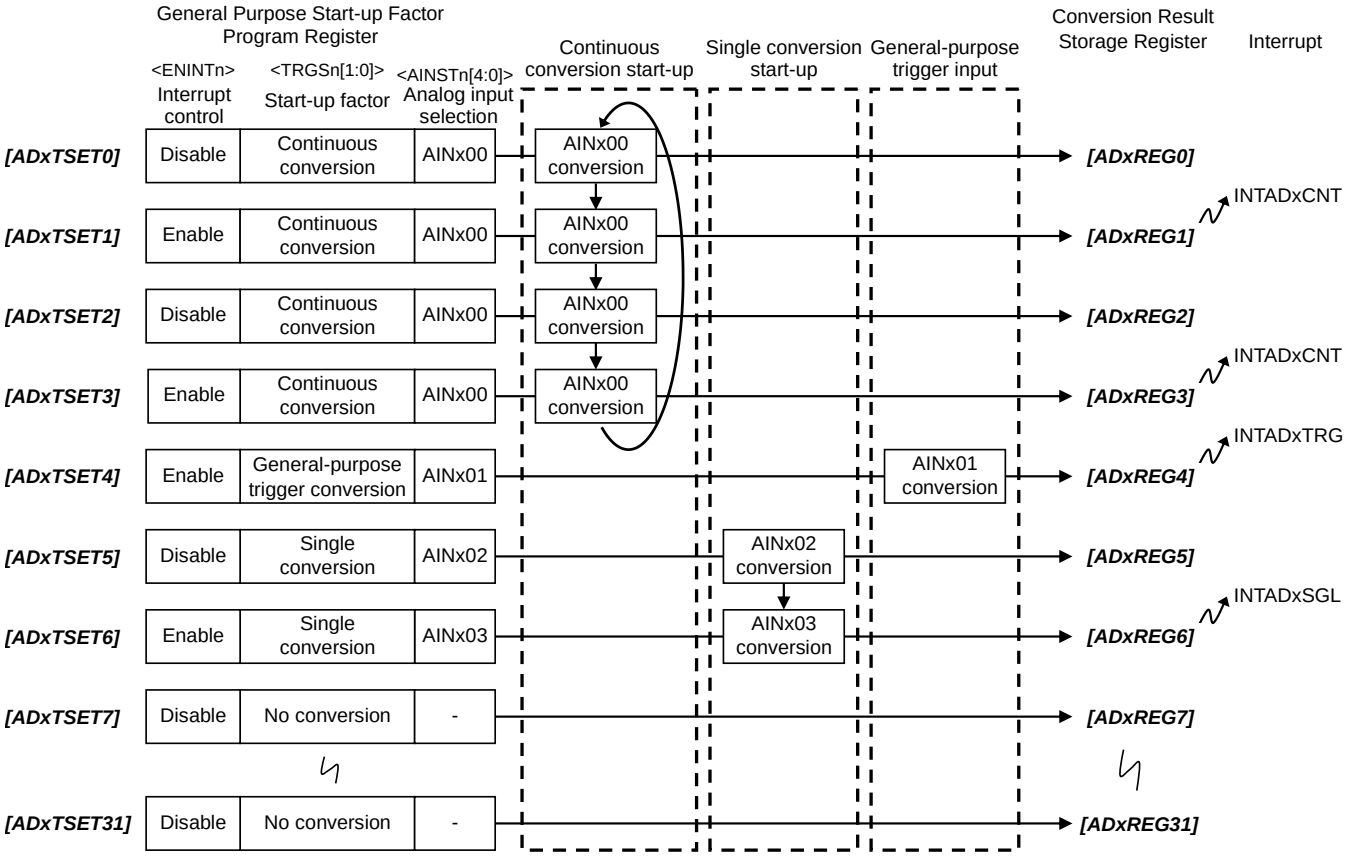


Figure 3.1 Conversion Operation of General-purpose Start-up Factors

The conversion operation for each start-up factor is as follows.

- Software factor
  - Single conversion
    When ***[ADxCR0]***<SGL> is set to "1", ADC starts up and performs one conversion operation according to the order of n.
  - Continuous conversion
    When ***[ADxCR0]***<CNT> is set to "1", ADC starts up and repeatedly performs conversion operations according to the order of n.
    Set "0" to [ADxCR0]<CNT> to terminate.

- General-purpose triggers
  When a general-purpose trigger is input, ADC starts up and performs one conversion operation according to the order of n.

A DMA request can be generated for each general purpose start-up factor. When ***[ADxCR1]***<CNTDMEN>, <SGLDMEN>, and <TRGDMEN> are "1" with interrupt generation enabled, a DMA request is generated at the same time as the interrupt generation.

Table 3.1 Start-up Factor and Interrupt/DMA Request

Start-up factor	Interrupt	DMA request
General-purpose trigger conversion	General-purpose trigger interrupt (INTADxTRG)	General-purpose trigger DMA request (ADxTRG_DMAREQ)
Single conversion	Single conversion interrupt (INTADxSGL)	Single conversion DMA request (ADxSGL_DMAREQ)
Continuous conversion	Continuous conversion interrupt (INTADxCNT)	Continuous conversion DMA request (ADxCNT_DMAREQ)

3.3.1.2. Operation Setup

Conversion by general-purpose start-up factors is set in the following registers.

- General-purpose start-up factor program register (***[ADxTSET0]*** to ***[ADxTSET31]***)
  The general-purpose start-up factor program register is prepared for each conversion result storage register.
  Set analog input selection to <AINSTn[4:0]>, start-up factor to <TRGSn[1:0]>, and interrupt control to <ENINTn> in ***[ADxTSETn]***.

- Control register0 (***[ADxCR0]***)
  To enable conversion, set ***[ADxCR0]***<ADEN> to "1".
  To start a single or continuous conversion of software factors, set "1" to ***[ADxCR0]***<SGL> or <CNT>. To stop continuous conversion, set "0" to <CNT>.

- Control register1 (***[ADxCR1]***)
  To enable conversion by a general-purpose trigger, set "1" to ***[ADxCR1]***<TRGEN>. With this setting, conversion starts when a general-purpose trigger is input.
  To enable DMA requests, set ***[ADxCR1]***<SGLDMEN>, <CNTDMEN>, and <TRGDMEN> to "1".

Follow the steps below to set the registers.

- Single conversion
  (1) Set analog input selection <AINSTn[4:0]> = arbitrary, start-up factor <TRGSn[1:0]> = "10", interrupt control <ENINTn> = arbitrary in ***[ADxTSETn]***.
  (2) For multiple conversions, set another ***[ADxTSETn]*** in the same way.
  (3) Set ***[ADxCR0]***<ADEN> to "1".
  (4) When ***[ADxCR0]***<SGL> is set to "1", the conversion starts and ***[ADxST]***<SNGF> is set to "1".
  (5) If interrupt is enabled (<ENINTn> = "1"), INTADxSGL occurs when the conversion is completed.
  (6) When all conversions set as start-up factor <TRGSn[1:0]> = "10" are completed, ***[ADxST]***<SNGF> is set to "0".

  To execute multiple single-conversion operation, repeat (4) through (6).

- Continuous conversion
  (1) Set analog input selection <AINSTn[4:0]> = arbitrary, start-up factor <TRGSn[1:0]> = "01", interrupt control <ENINTn> = arbitrary in ***[ADxTSETn]***.
  (2) To perform continuous conversion of multiple analog inputs, set another ***[ADxTSETn]*** in the same way.
  (3) Set ***[ADxCR0]***<ADEN> to "1".
  (4) When ***[ADxCR0]***<CNT> is set to "1", the conversion starts and ***[ADxST]***<CNTF> is set to "1".
  (5) If interrupt is enabled (<ENINTn> = "1"), INTADxCNT occurs when the conversion is completed.
  (6) When all conversions set as start-up factor <TRGSn[1:0]> = "01" are completed, the conversions are repeated from the beginning.
  (9) Setting "0" to ***[ADxCR0]***<CNT> terminates the conversion. When the conversion is completed, ***[ADxST]***<CNTF> is set to "0".

- General-purpose trigger conversion
  (1) Sets the general-purpose trigger (ADxTRGIN) to be used. (Note)
  (2) Set analog input selection <AINSTn[4:0]> = arbitrary, start-up factor <TRGSn[1:0]> = "11", interrupt control <ENINTn> = arbitrary in ***[ADxTSETn]***.
  (3) For multiple conversions, set another ***[ADxTSETn]*** in the same way.
  (4) Set ***[ADxCR1]***<TRGEN> to "1".
  (5) Set ***[ADxCR0]***<ADEN> to "1".
  (6) When the trigger is input, the conversion starts and ***[ADxST]***<TRGF> is set to "1".
  (7) If interrupt is enabled (<ENINTn> = "1"), INTADxTRG occurs when the conversion is completed.
  (8) When all conversions set as start-up factor <TRGSn[1:0]> = "11" are completed, ***[ADxST]***<TRGF> is set to "0".

  Repeat (6) to (8) for each general-purpose trigger input.

  Note: For available general-purpose triggers (ADxTRGIN), refer to the reference manual "Product Information".

3.3.2. Conversion Operation by PMD Trigger

3.3.2.1. Operation

The conversion is started by the PMDTRGn (n = 0 to 11). PMDTRGn are triggers from the PMD or the other peripheral function. (Note)

In the conversion operation by PMD trigger, the conversion is executed by setting the program for PMD trigger. Eight programs can be set, and one program is selected to be started for each PMDTRGn.

Up to four conversions can be performed in a single program. The destination of the conversion result can be selected in units of four conversion result storage registers.

Either INTADxPDA or INTADxPDB interrupt can be generated at program completion.

Note: Trigger connections vary by product, refer to the reference manual "Product Information".

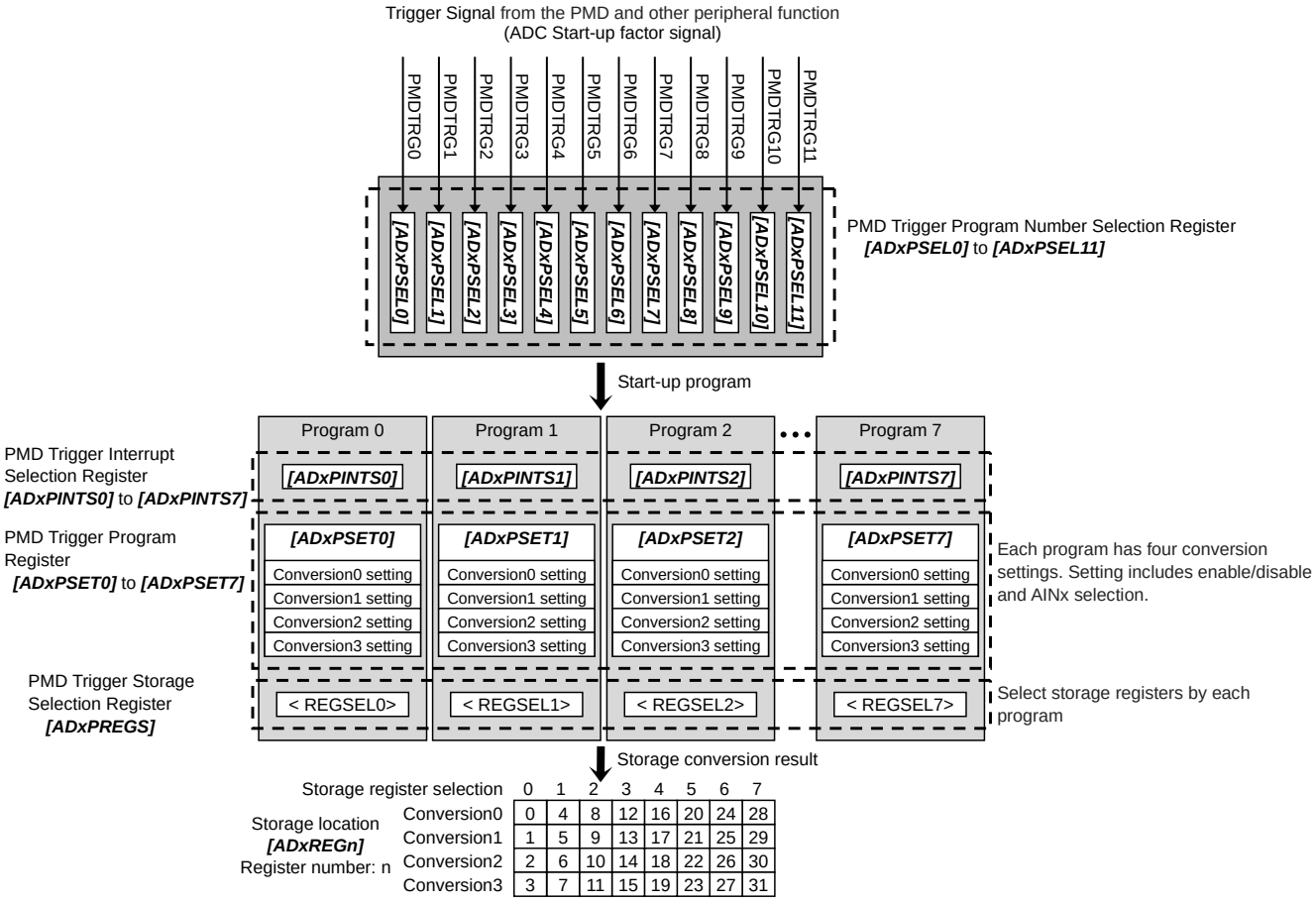


Figure 3.2 PMD Trigger Factor and Related Registers

3.3.2.2. Operation Setup

Conversion operation by PMD trigger is set by the following registers.

- PMD trigger program number selection register (***[ADxPSEL0]*** to ***[ADxPSEL11]***)
  Selects trigger enable/disable and the program number (0 to 7) to be activated for the 12 PMD triggers.

- PMD trigger program register (***[ADxPSET0]*** to ***[ADxPSET7]***)
  Up to 4 conversions can be set in one program. Enables/disables the first through fourth conversions and sets the analog input to be converted.

- PMD trigger interrupt selection register (***[ADxPINTS0]*** to ***[ADxPINTS7]***)
  An interrupt can be generated at the end of each program. Set the interrupt type (INTADxPDA, INTADxPDB) and enable/disable in the PMD trigger interrupt selection register.

- PMD trigger storage selection register (***[ADxPREGS]***)
  The destination of the conversion result for each program can be selected. The destination is selected from the group of the conversion result storage register 0 to 3, 4 to 7, 8 to 11, 12 to 15, 16 to 19, 20 to 23, 24 to 27, or 28 to 31.

- Control register0 (***[ADxCR0]***)
  To enable the conversion operation, set "1" to ***[ADxCR0]***<ADEN>.

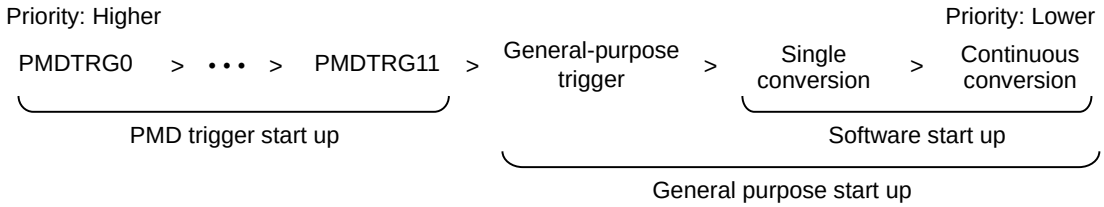
Follow the steps below to set the registers.

(1) Set trigger control <PENSm> = "1" and program number <PMDSm[2:0]> = arbitrary in ***[ADxPSELm]***.
(2) Set analog input selection <AINSPn[4:0]> = arbitrary, conversion control <ENSPn> = "1" in ***[ADxPSETn]***.
(3) Set interrupt selection <INTSELn[1:0]> = INTADxPDA or INTADxPDB in ***[ADxPINTSn]***.
(4) Set Register selection <REGSELn[2:0]> = arbitrary in ***[ADxPREGS]***.
(5) Set ***[ADxCR0]***<ADEN> to "1".
(6) When the PMD trigger is input, the conversion starts and ***[ADxST]***<PMDF> is set to "1".
(7) If interrupt is enabled, an interrupt (INTADxPDA or INTADxPDB) is generated at the end of the conversion program.

Repeat (6) and (7) for each PMD trigger input.

3.3.3. Priority of Start-up Factors

The start-up factors are prioritized as follows:



If multiple start-up factors occur simultaneously, the conversion of the highest priority start-up factor is executed and the other start-up factors are put on hold.

Once a conversion program with a PMD trigger factor has started, it will not be interrupted. Even if a higher priority PMD trigger occurs, it is held until the end of the running conversion program.

Conversions using general-purpose start-up factors (general-purpose trigger, single conversion, and continuous conversion) are suspended if a higher priority start-up factor occurs during execution, and the conversion of the higher priority start-up factor is executed. If a lower priority start-up factor occurs during execution, it is put on hold. If interrupted, resumes from the interrupted conversion when it becomes executable.

If the start-up factor for the conversion occurs again during the conversion, it is ignored.

The execution status can be checked at ***[ADxST]***<CNTF>, <SNGF>, <TRGF>, and <PMDF>. Single and continuous conversions can be started reliably by checking that the flag is "0" before performing the register operation.

Table 3.2 Priority of Start-up Factors

		Start-up factors occurring later			
		PMDTRGn (Note1)	**General purpose trigger**	**Software single conversion**	**Software continuous conversion**
Running start-up factors	PMDTRGm (Note)	Continue current factor	Continue current factor	Continue current factor	Continue current factor
	General-purpose trigger	Start later factor	Continue current factor	Continue current factor	Continue current factor
	Software single conversion	Start later factor	Start later factor	Continue current factor	Continue current factor
	Software continuous conversion	Start later factor	Start later factor	Start later factor	Continue current factor

Note: m, n = 0 to 11

3.3.4. Conversion Stop

Writing "0" to ***[ADxCR0]***<ADEN> immediately stops the conversion. If continuous conversion is enabled, write "0" to ***[ADxCR0]***<CNT> as well.

When the conversion stops completely, all bits in ***[ADxST]*** are set to "0". The values of registers other than ***[ADxST]*** are retained. Before enabling the next conversion, read the value in the conversion result storage registers to clear the flags.

When stopping ADCLK, make sure that ***[ADxST]***<ADBF> is "0".

3.4. AD Monitor Function

3.4.1. Operation

The AD monitor function generates an interrupt if the AD conversion result is bigger than or smaller than the set value. By using this function on two channels simultaneously, it is possible to detect whether the AD conversion result is within or outside the range of the two set values.

The following description is for ***[ADxCMPCR0]*** (Same for ***[ADxCMPCR1]***).

Sets the conversion result storage register to compare with ***[ADxCMPCR0]***<REGS0[4:0]>, the big/small determination with <ADBIG0>, the determination count condition with <CMPCND0>, and the determination count with <CMPCNT0[3:0]>.

When ***[ADxCMPEN]***<CMP0EN> is set to "1", the AD monitor function is enabled.

Whenever a conversion result is stored to the target conversion result storage register, a big or small determination is made. If the determination result matches the <ADBIG0> setting, the determination counter is increased.

There are two types of determination count conditions: continuous and accumulated method.

In the continuous method, the AD monitor function interrupt (INTADxCP0) and the protection signal to the PMD are generated when the status set in <ADBIG0> occurs continuously and reaches the number of times set in <CMPCNT0[3:0]>. If the number of determinations exceeds the <CMPCNT0[3:0]> setting, nothing occurs. The counter value is cleared when the determination is different from the state set in <ADBIG0>.

In the accumulation method, when the status set in <ADBIG0> accumulates and reaches the number of times set in <CMPCNT0[3:0]>, the AD monitor function interrupt (INTADxCP0) and the protection signal to the PMD are generated, and the counter is cleared. The counter value is retained even if the determination is different from the status set in <ADBIG0>.

If the value in the conversion result storage register specified in the ***[ADxCMPCR0]*** register is the same as the value in the conversion result comparison register, the determination counter is not increased and no AD monitor function interrupt (INTADxCP0) or trigger occurs.

Table 3.3 Monitor Function and Interrupt

Monitor function	Interrupt
Monitor function Setting Register0 (***[ADxCMPCR0]***)	Monitor function0 Interrupt (INTADxCP0)
Monitor function Setting Register1 (***[ADxCMPCR1]***)	Monitor function1 Interrupt (INTADxCP1)

When using the AD monitor function, the conversion result storage register is not normally read. Therefore, the overrun flag ***[ADxREGn]***<ADOVRFn> and the conversion result storage flag ***[ADxREGn]***<ADRFn> remain and have no meaning.

3.4.2. Operation Setup

(1) Determination by Continuous count

- Monitor function setting register0 (***[ADxCMPCR0]*** = 0x00000200)
  Conversion result storage register (Comparison target): ***[ADxREG0]***
  Magnitude determination: ***[ADxREG0]***<ADR0[11:0]> > ***[ADxCMP0]***<AD0CMP0[11:0]>
  (Bigger than the comparison register.)
  Determination count condition: Continuous count
  Magnitude determination count: 3 counts
- AD conversion result comparison register0 (***[ADxCMP0]***<AD0CMP0[11:0]> = 0x888)
- Monitor function enable register (***[ADxCMPEN]*** = 0x00000001)

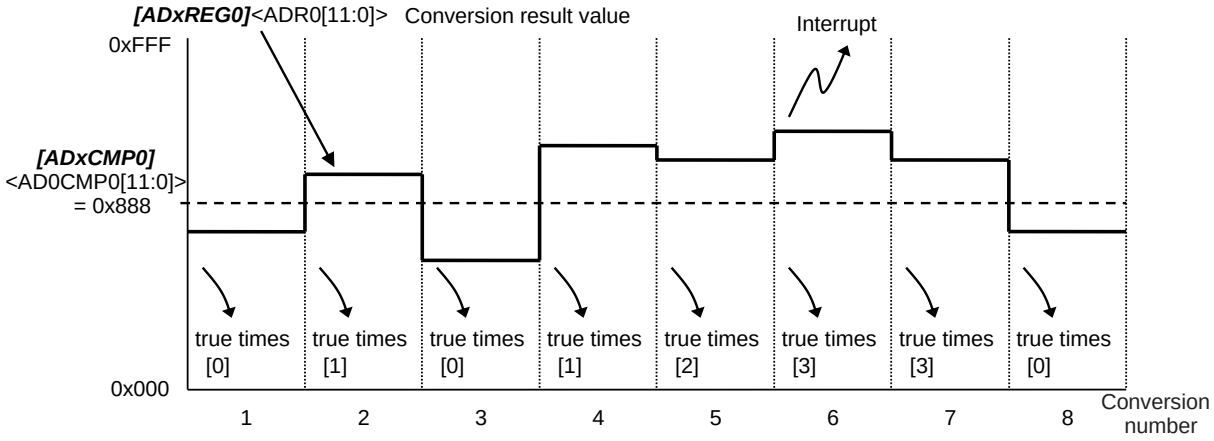


Figure 3.3 AD Monitor Function (Determination Count Condition: Continuous Count)

(2) Determination by Accumulated count

- Monitor function setting register0 (***[ADxCMPCR0]*** = 0x00000240)

  Conversion result storage register (Comparison target): ***[ADxREG0]***
  Magnitude determination: ***[ADxREG0]***<ADR0[11:0]> > ***[ADxCMP0]***<AD0CMP0[11:0]>
  (Bigger than the comparison register.)
  Determination count condition: Accumulated count
  Magnitude determination count: 3 counts
- AD conversion result comparison register0 (***[ADxCMP0]***<AD0CMP0[11:0]> = 0x888)
- Monitor function enable register (***[ADxCMPEN]*** = 0x00000001)

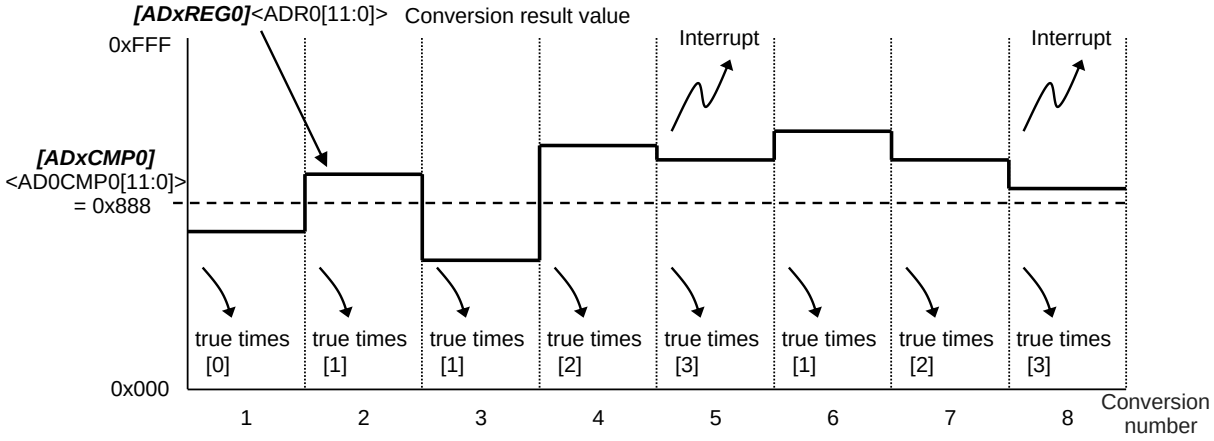


Figure 3.4 AD Monitor Function (Determination Count Condition: Accumulated Count)

3.5. Conversion Time

3.5.1. Calculation of Conversion Time

Refer to the "Electrical Characteristics" section of the data sheet for the range of conversion clock, conversion time, and sampling time.

The conversion time is calculated by the following formula.

Conversion time = Sampling time + Comparison time

The sampling time and The comparison time are calculated by the following formula.

Sampling time = SCLK period × m × n
Comparison time = SCLK period × 16

The value of "m" is set to ***[ADxCLK]***<EXAZ0> or <EXAZ1>.
The value of "n" is set to ***[ADxMOD1]***<MOD1>.

As an example of a case where the sampling time is 0.2μs at 40MHz (period: 25ns) SCLK, set ***[ADxCLK]***<EXAZ0> or <EXAZ1> to "0001" (m = 2) and ***[ADxMOD1]***<MOD1> to "0x00000000" (n = 4).
In this case, the sampling time and the comparison time is as follows.

Sampling time = 25ns × 2 ×4 = 0.2μs
Comparison time = 25ns × 16 = 0.4μs

And the conversion time is as follows.

Conversion time = 0.2μs + 0.4μs = 0.6μs

3.5.2. Sampling Time Selection

The value of "m" can be selected from ***[ADxCLK]***<EXAZ0> or <EXAZ1> for each conversion channel. Set ***[ADxEXAZSEL]*** to use either one.

3.6. Reduced Current Consumption

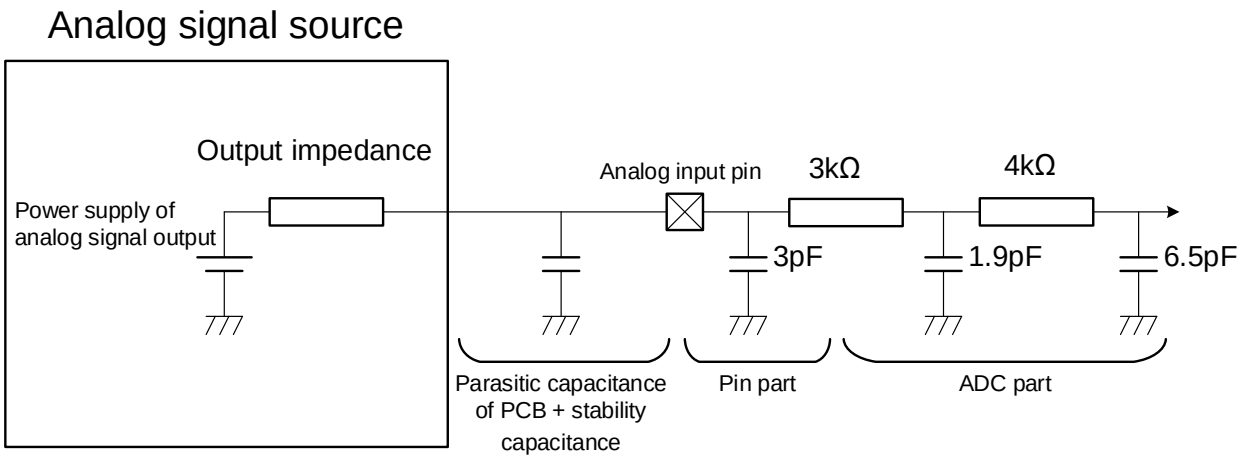
Setting "1" to the low-power mode selection bit ***[ADxMOD0]***<RCUT> reduces current consumption by shutting off the current between VREFHx and VREFLx except during conversion.

Note: When using multiple ADCs, do not select low-power mode.

3.7. Equivalent Circuit

Equivalent circuit of analog input pin is shown in Figure 3.5.

1. Condition 1
   - AVDD5 = 4.5 to 5.5V



2. Condition 2
   - AVDD5 = 2.7 to 4.5V

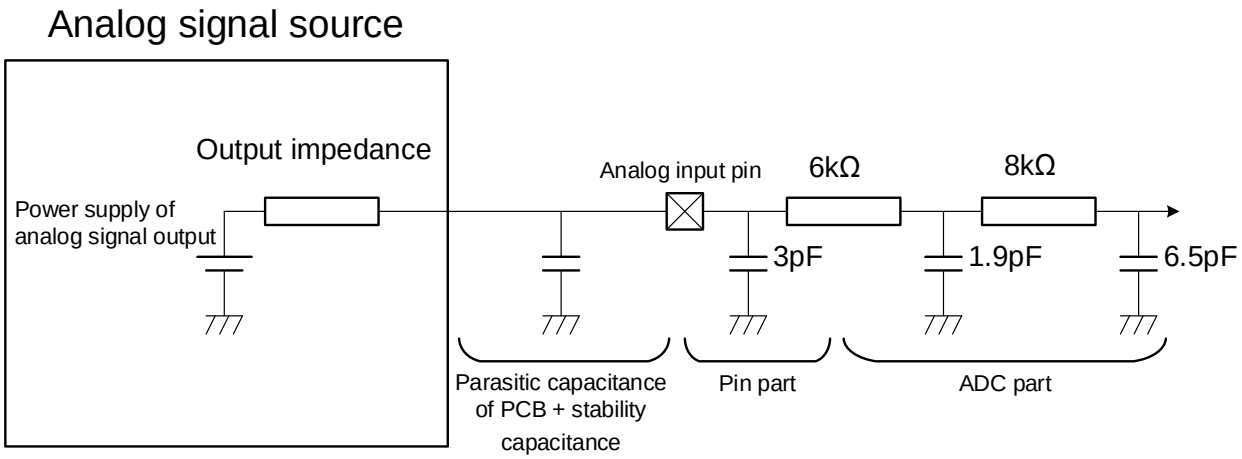


Figure 3.5 Equivalent Circuit of Analog Input Pin

4. Registers

4.1. List of Registers

The control registers and their addresses are shown as follows:

Peripheral function		Channel/unit	Base address		
			TYPE1	TYPE2	TYPE3
12-bit Analog to Digital Converter	ADC	Unit A	0x400B8800	0x400BA000	0x4005A000
		Unit B	0x400B8C00	0x400BA400	0x4005A400

Note: The Channel/Unit and Base address type are different by products. Please refer to the reference manual "Product Information" for the details.

Register name		Address (Base+)
Control Register 0	***[ADxCR0]***	0x0000
Control Register 1	***[ADxCR1]***	0x0004
Status Register	***[ADxST]***	0x0008
Conversion Clock Setting Register	***[ADxCLK]***	0x000C
Mode Setting Register 0	***[ADxMOD0]***	0x0010
Mode Setting Register 1	***[ADxMOD1]***	0x0014
Mode Setting Register 2	***[ADxMOD2]***	0x0018
Monitor function Enable Register	***[ADxCMPEN]***	0x0020
Monitor function Setting Register 0	***[ADxCMPCR0]***	0x0024
Monitor function Setting Register 1	***[ADxCMPCR1]***	0x0028
Conversion Result Comparison Register 0	***[ADxCMP0]***	0x002C
Conversion Result Comparison Register 1	***[ADxCMP1]***	0x0030
PMD Trigger Program Number Selection Register 0	***[ADxPSEL0]***	0x0040
PMD Trigger Program Number Selection Register 1	***[ADxPSEL1]***	0x0044
PMD Trigger Program Number Selection Register 2	***[ADxPSEL2]***	0x0048
PMD Trigger Program Number Selection Register 3	***[ADxPSEL3]***	0x004C
PMD Trigger Program Number Selection Register 4	***[ADxPSEL4]***	0x0050
PMD Trigger Program Number Selection Register 5	***[ADxPSEL5]***	0x0054
PMD Trigger Program Number Selection Register 6	***[ADxPSEL6]***	0x0058
PMD Trigger Program Number Selection Register 7	***[ADxPSEL7]***	0x005C
PMD Trigger Program Number Selection Register 8	***[ADxPSEL8]***	0x0060
PMD Trigger Program Number Selection Register 9	***[ADxPSEL9]***	0x0064
PMD Trigger Program Number Selection Register 10	***[ADxPSEL10]***	0x0068
PMD Trigger Program Number Selection Register 11	***[ADxPSEL11]***	0x006C
PMD Trigger Interrupt Selection Register 0	***[ADxPINTS0]***	0x0070
PMD Trigger Interrupt Selection Register 1	***[ADxPINTS1]***	0x0074
PMD Trigger Interrupt Selection Register 2	***[ADxPINTS2]***	0x0078
PMD Trigger Interrupt Selection Register 3	***[ADxPINTS3]***	0x007C
PMD Trigger Interrupt Selection Register 4	***[ADxPINTS4]***	0x0080
PMD Trigger Interrupt Selection Register 5	***[ADxPINTS5]***	0x0084
PMD Trigger Interrupt Selection Register 6	***[ADxPINTS6]***	0x0088
PMD Trigger Interrupt Selection Register 7	***[ADxPINTS7]***	0x008C

Register name		Address (Base+)
PMD Trigger Storage Selection Register	***[ADxPREGS]***	0x0090
Trimming Setting Register	***[ADxTRM]***	0x0094
AIN sampling time selection Register	***[ADxEXAZSEL]***	0x009C
PMD Trigger Program Register 0	***[ADxPSET0]***	0x00A0
PMD Trigger Program Register 1	***[ADxPSET1]***	0x00A4
PMD Trigger Program Register 2	***[ADxPSET2]***	0x00A8
PMD Trigger Program Register 3	***[ADxPSET3]***	0x00AC
PMD Trigger Program Register 4	***[ADxPSET4]***	0x00B0
PMD Trigger Program Register 5	***[ADxPSET5]***	0x00B4
PMD Trigger Program Register 6	***[ADxPSET6]***	0x00B8
PMD Trigger Program Register 7	***[ADxPSET7]***	0x00BC
General Purpose Start-up Factor Program Register 0	***[ADxTSET0]***	0x00C0
General Purpose Start-up Factor Program Register 1	***[ADxTSET1]***	0x00C4
General Purpose Start-up Factor Program Register 2	***[ADxTSET2]***	0x00C8
General Purpose Start-up Factor Program Register 3	***[ADxTSET3]***	0x00CC
General Purpose Start-up Factor Program Register 4	***[ADxTSET4]***	0x00D0
General Purpose Start-up Factor Program Register 5	***[ADxTSET5]***	0x00D4
General Purpose Start-up Factor Program Register 6	***[ADxTSET6]***	0x00D8
General Purpose Start-up Factor Program Register 7	***[ADxTSET7]***	0x00DC
General Purpose Start-up Factor Program Register 8	***[ADxTSET8]***	0x00E0
General Purpose Start-up Factor Program Register 9	***[ADxTSET9]***	0x00E4
General Purpose Start-up Factor Program Register 10	***[ADxTSET10]***	0x00E8
General Purpose Start-up Factor Program Register 11	***[ADxTSET11]***	0x00EC
General Purpose Start-up Factor Program Register 12	***[ADxTSET12]***	0x00F0
General Purpose Start-up Factor Program Register 13	***[ADxTSET13]***	0x00F4
General Purpose Start-up Factor Program Register 14	***[ADxTSET14]***	0x00F8
General Purpose Start-up Factor Program Register 15	***[ADxTSET15]***	0x00FC
General Purpose Start-up Factor Program Register 16	***[ADxTSET16]***	0x0100
General Purpose Start-up Factor Program Register 17	***[ADxTSET17]***	0x0104
General Purpose Start-up Factor Program Register 18	***[ADxTSET18]***	0x0108
General Purpose Start-up Factor Program Register 19	***[ADxTSET19]***	0x010C
General Purpose Start-up Factor Program Register 20	***[ADxTSET20]***	0x0110
General Purpose Start-up Factor Program Register 21	***[ADxTSET21]***	0x0114
General Purpose Start-up Factor Program Register 22	***[ADxTSET22]***	0x0118
General Purpose Start-up Factor Program Register 23	***[ADxTSET23]***	0x011C
General Purpose Start-up Factor Program Register 24	***[ADxTSET24]***	0x0120
General Purpose Start-up Factor Program Register 25	***[ADxTSET25]***	0x0124
General Purpose Start-up Factor Program Register 26	***[ADxTSET26]***	0x0128
General Purpose Start-up Factor Program Register 27	***[ADxTSET27]***	0x012C
General Purpose Start-up Factor Program Register 28	***[ADxTSET28]***	0x0130
General Purpose Start-up Factor Program Register 29	***[ADxTSET29]***	0x0134
General Purpose Start-up Factor Program Register 30	***[ADxTSET30]***	0x0138
General Purpose Start-up Factor Program Register 31	***[ADxTSET31]***	0x013C
Conversion Result Storage Register 0	***[ADxREG0]***	0x0140
Conversion Result Storage Register 1	***[ADxREG1]***	0x0144
Conversion Result Storage Register 2	***[ADxREG2]***	0x0148
Conversion Result Storage Register 3	***[ADxREG3]***	0x014C

Register name		Address (Base+)
Conversion Result Storage Register 4	***[ADxREG4]***	0x0150
Conversion Result Storage Register 5	***[ADxREG5]***	0x0154
Conversion Result Storage Register 6	***[ADxREG6]***	0x0158
Conversion Result Storage Register 7	***[ADxREG7]***	0x015C
Conversion Result Storage Register 8	***[ADxREG8]***	0x0160
Conversion Result Storage Register 9	***[ADxREG9]***	0x0164
Conversion Result Storage Register 10	***[ADxREG10]***	0x0168
Conversion Result Storage Register 11	***[ADxREG11]***	0x016C
Conversion Result Storage Register 12	***[ADxREG12]***	0x0170
Conversion Result Storage Register 13	***[ADxREG13]***	0x0174
Conversion Result Storage Register 14	***[ADxREG14]***	0x0178
Conversion Result Storage Register 15	***[ADxREG15]***	0x017C
Conversion Result Storage Register 16	***[ADxREG16]***	0x0180
Conversion Result Storage Register 17	***[ADxREG17]***	0x0184
Conversion Result Storage Register 18	***[ADxREG18]***	0x0188
Conversion Result Storage Register 19	***[ADxREG19]***	0x018C
Conversion Result Storage Register 20	***[ADxREG20]***	0x0190
Conversion Result Storage Register 21	***[ADxREG21]***	0x0194
Conversion Result Storage Register 22	***[ADxREG22]***	0x0198
Conversion Result Storage Register 23	***[ADxREG23]***	0x019C
Conversion Result Storage Register 24	***[ADxREG24]***	0x01A0
Conversion Result Storage Register 25	***[ADxREG25]***	0x01A4
Conversion Result Storage Register 26	***[ADxREG26]***	0x01A8
Conversion Result Storage Register 27	***[ADxREG27]***	0x01AC
Conversion Result Storage Register 28	***[ADxREG28]***	0x01B0
Conversion Result Storage Register 29	***[ADxREG29]***	0x01B4
Conversion Result Storage Register 30	***[ADxREG30]***	0x01B8
Conversion Result Storage Register 31	***[ADxREG31]***	0x01BC

4.2. Details of Registers

4.2.1. *[ADxCR0]* (Control Register 0)

Bit	Bit symbol	After reset	Type	Function
31:8	-	0	R	Read as "0".
7	ADEN	0	R/W	ADC control. 0: Disabled 1: Enabled Set this bit to "1" to enable conversion. Set this bit to "0" to terminate the conversion.
6:2	-	0	R	Read as "0".
1	SGL	0	W	Single conversion control 0: Don't care. 1: Conversion start When set to "1", a single conversion is started. The value when read is "0".
0	CNT	0	R/W	Continuous conversion control 0: Disabled 1: Enabled While set this bit to "1", continuous conversion is repeatedly performed. Set "1" when ***[ADxST]***<CNTF> is "0" (continuous conversion is not running).

4.2.2. *[ADxCR1]* (Control Register 1)

Bit	Bit symbol	After reset	Type	Function
31:7	-	0	R	Read as "0".
6	CNTDMEN	0	R/W	Continuous conversion DMA request control 0: Disabled 1: Enabled
5	SGLDMEN	0	R/W	Single conversion DMA request control 0: Disabled 1: Enabled
4	TRGDMEN	0	R/W	General-purpose trigger DMA request control 0: Disabled 1: Enabled
3:1	-	0	R	Read as "0".
0	TRGEN	0	R/W	General-purpose trigger start-up control 0: Disabled 1: Enabled

Note: This register must be set with ***[ADxCR0]***<ADEN> = 0.

4.2.3. *[ADxST]* (Status Register)

Bit	Bit symbol	After reset	Type	Function
31:8	-	0	R	Read as "0".
7	ADBF	0	R	ADCLK stop confirmation flag 0: Can stop ADCLK 1: Cannot stop ADCLK When stopping ADCLK, make sure this bit is "0".
6:4	-	0	R	Read as "0".
3	CNTF	0	R	Continuous conversion execution flag 0: Stop 1: Executing This bit is set to "1" when the request is accepted and is set to "0" when the last conversion result is stored.
2	SNGF	0	R	Single conversion execution flag 0: Stop 1: Executing (Note) This bit is set to "1" when the request is accepted and is set to "0" when the last conversion result is stored.
1	TRGF	0	R	General-purpose trigger conversion execution flag 0: Stop 1: Executing This bit is set to "1" when the request is accepted and is set to "0" when the last conversion result is stored.
0	PMDF	0	R	PMD trigger conversion execution flag 0: Stop 1: Executing This bit is set to "1" when the request is accepted and is set to "0" when the last conversion result is stored.

Note: If <SNGF> is read immediately after setting ***[ADxCR0]***<SGL> = 1, "0" may be read. In this case, read <SNGF> again to confirm that it is "1".

4.2.4. *[ADxCLK]* (Conversion Clock Setting Register)

Bit	Bit symbol	After reset	Type	Function
31:12	-	0	R	Read as "0".
11:8	EXAZ1[3:0]	0000	R/W	m value 0000: 1 0110: 32 0001: 2 0111: 64 0010: 3 1000: 128 0011: 4 1001: 256 0100: 8 1010: 512 0101: 16 1011: 1024 Others: Reserved
7	-	0	R	Read as "0".
6:3	EXAZ0[3:0]	0000	R/W	m value 0000: 1 0110: 32 0001: 2 0111: 64 0010: 3 1000: 128 0011: 4 1001: 256 0100: 8 1010: 512 0101: 16 1011: 1024 Others: Reserved
2:0	VADCLK[2:0]	000	R/W	AD prescaler output (SCLK) selection 000: ADCLK / 2 001: ADCLK / 4 010: ADCLK / 8 011: ADCLK / 16 100: ADCLK / 3 101: ADCLK / 5 110: ADCLK / 6 111: ADCLK / 10

Note: This register must be set with ***[ADxCR0]***<ADEN> = 0.

4.2.5. *[ADxMOD0]* (Mode Setting Register 0)

Bit	Bit symbol	After reset	Type	Function
31:2	-	0	R	Read as "0".
1	RCUT	1	R/W	Low-power consumption mode selection 0: Normal operation 1: Low-power consumption mode operation Setting this bit to "1" shuts off the current between VREFH and VREFL except during conversion.
0	DACON	0	R/W	DAC control 0: OFF 1: ON When using ADC, set <DACON> to "1" first.

Note: This register must be set while ***[ADxCR0]***<ADEN> = 0.

4.2.6. *[ADxMOD1]* (Mode Setting Register 1)

Bit	Bit symbol	After reset	Type	Function
31:0	MOD1[31:0]	0x00004000	R/W	n value 0x00000000: 4 0x00001000: 5 0x00002000: 6 0x00003000: 7 0x00004000: 8 0x00005000: 9 0x00006000: 10 0x00007000: 11 0x00008000: 12 0x00009000: 13 0x0000A000: 14 0x0000B000: 15 0x0000C000: 16 0x0000D000: 17 0x0000E000: 18 0x0000F000: 19 Others: Reserved

Note: This register must be set with ***[ADxCR0]***<ADEN> = 0.

4.2.7. *[ADxMOD2]* (Mode Setting Register 2)

Bit	Bit symbol	After reset	Type	Function
31:0	MOD2[31:0]	0x00000000	R/W	The setting value of this register varies depending on the product. For the setting value, refer to the reference manual "Product Information".

Note: This register must be set with ***[ADxCR0]***<ADEN> = 0.

4.2.8. *[ADxCMPEN]* (Monitor Function Enable Register)

Bit	Bit symbol	After reset	Type	Function
31:2	-	0	R	Read as "0"
1	CMP1EN	0	R/W	AD monitor function1 operation 0: Disabled 1: Enabled
0	CMP0EN	0	R/W	AD monitor function0 operation 0: Disabled 1: Enabled

4.2.9. *[ADxCMPCR0]* (Monitor Function Setting Register 0)

Bit	Bit symbol	After reset	Type	Function
31:12	-	0	R	Read as "0".
11:8	CMPCNT0[3:0]	0000	R/W	Comparison count 0000: 1 1000: 9 0001: 2 1001: 10 0010: 3 1010: 11 0011: 4 1011: 12 0100: 5 1100: 13 0101: 6 1101: 14 0110: 7 1110: 15 0111: 8 1111: 16
7	-	0	R	Read as "0".
6	CMPCND0	0	R/W	Determination condition 0: Continuous count method 1: Accumulated count method
5	ADBIG0	0	R/W	Big/small determination setting 0: Conversion result specified by <REGS0[4:0]> > ***[ADxCMP0]*** 1: Conversion result specified by <REGS0[4:0]> < ***[ADxCMP0]***
4:0	REGS0[4:0]	00000	R/W	Compared conversion result storage register 00000: ADxREG0 01011: ADxREG11 10110: ADxREG22 00001: ADxREG1 01100: ADxREG12 10111: ADxREG23 00010: ADxREG2 01101: ADxREG13 11000: ADxREG24 00011: ADxREG3 01110: ADxREG14 11001: ADxREG25 00100: ADxREG4 01111: ADxREG15 11010: ADxREG26 00101: ADxREG5 10000: ADxREG16 11011: ADxREG27 00110: ADxREG6 10001: ADxREG17 11100: ADxREG28 00111: ADxREG7 10010: ADxREG18 11101: ADxREG29 01000: ADxREG8 10011: ADxREG19 11110: ADxREG30 01001: ADxREG9 10100: ADxREG20 11111: ADxREG31 01010: ADxREG10 10101: ADxREG21

Note: This register must be set with ***[ADxCMPEN]***<CMP0EN> = 0.

4.2.10. *[ADxCMPCR1]* (Monitor Function Setting Register 1)

Bit	Bit symbol	After reset	Type	Function
31:12	-	0	R	Read as "0".
11:8	CMPCNT1[3:0]	0000	R/W	Comparison count 0000: 1 1000: 9 0001: 2 1001: 10 0010: 3 1010: 11 0011: 4 1011: 12 0100: 5 1100: 13 0101: 6 1101: 14 0110: 7 1110: 15 0111: 8 1111: 16
7	-	0	R	Read as "0".
6	CMPCND1	0	R/W	Determination condition 0: Continuous count method 1: Accumulated count method
5	ADBIG1	0	R/W	Big/small determination setting 0: Conversion result specified by <REGS1[4:0]> > ***[ADxCMP1]*** 1: Conversion result specified by <REGS1[4:0]> < ***[ADxCMP1]***
4:0	REGS1[4:0]	00000	R/W	Compared conversion result storage register 00000: ADxREG0 01011: ADxREG11 10110: ADxREG22 00001: ADxREG1 01100: ADxREG12 10111: ADxREG23 00010: ADxREG2 01101: ADxREG13 11000: ADxREG24 00011: ADxREG3 01110: ADxREG14 11001: ADxREG25 00100: ADxREG4 01111: ADxREG15 11010: ADxREG26 00101: ADxREG5 10000: ADxREG16 11011: ADxREG27 00110: ADxREG6 10001: ADxREG17 11100: ADxREG28 00111: ADxREG7 10010: ADxREG18 11101: ADxREG29 01000: ADxREG8 10011: ADxREG19 11110: ADxREG30 01001: ADxREG9 10100: ADxREG20 11111: ADxREG31 01010: ADxREG10 10101: ADxREG21

Note: This register must be set with ***[ADxCMPEN]***<CMP1EN> = 0.

4.2.11. *[ADxCMP0]* (Conversion Result Comparison Register 0)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:4	AD0CMP0[11:0]	0x000	R/W	AD conversion result comparison value storage The value compared with the AD conversion result is set.
3:0	-	0	R	Read as "0"

Note: This register must be set with ***[ADxCMPEN]***<CMP0EN> = 0.

4.2.12. *[ADxCMP1]* (Conversion Result Comparison Register 1)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:4	AD0CMP1[11:0]	0x000	R/W	AD conversion result comparison value storage The value compared with the AD conversion result is set.
3:0	-	0	R	Read as "0".

Note: This register must be set with ***[ADxCMPEN]***<CMP1EN> = 0.

4.2.13. *[ADxEXAZSEL]* (AIN Sampling Time Selection Register)

Bit	Bit symbol	After reset	Type	Function
31:0	EXAZSEL[31:0]	0x00000000	R/W	Selection of m value 0: ***[ADxCLK]***<EXAZ0[3:0]> is used 1: ***[ADxCLK]***<EXAZ1[3:0]> is used Selects whether <EXAZ0[3:0]> or <EXAZ1[3:0]> is used for each analog input. Each bit corresponds to each analog input. EXAZSEL[31]: m value selection for AINx24 EXAZSEL[30]: m value selection for AINx23 : : EXAZSEL[0]: m value selection for AINx00

Note: This register must be set with ***[ADxCR0]***<ADEN> = 0.

4.2.14. *[ADxTRM]* (Trimming Setting Register)

Bit	Bit symbol	After reset	Type	Function
31:0	TRM[31:0]	0x00000000	R/W	The setting value of this register varies depending on the product. For the setting value, refer to the reference manual "Product Information".

Note: This register must be set with ***[ADxCR0]***<ADEN> = 0.

4.2.15. PMD Trigger Control Registers

4.2.15.1. *[ADxPSEL0]* (PMD Trigger Program Number Selection Register 0)

This is an example of ***[ADxPSEL0]***. The same configuration is used for ***[ADxPSEL1]*** to ***[ADxPSEL11]***.

Bit	Bit symbol	After reset	Type	Function
31:8	-	0	R	Read as "0".
7	PENS0	0	R/W	PMDTRG0 trigger control 0: Disabled 1: Enabled
6:3	-	0	R	Read as "0".
2:0	PMDS0[2:0]	000	R/W	Program number selection 000: Program 0 001: Program 1 010: Program 2 011: Program 3 100: Program 4 101: Program 5 110: Program 6 111: Program 7

Note: This register must be set with ***[ADxCR0]***<ADEN> = 0.

4.2.15.2. *[ADxPINTS0]* (PMD Trigger Interrupt Selection Register 0)

This is an example of ***[ADxPINTS0]***. The same configuration is used for ***[ADxPINTS1]*** to ***[ADxPINTS7]***.

Bit	Bit symbol	After reset	Type	Function
31:2	-	0	R	Read as "0"
1:0	INTSEL0[1:0]	00	R/W	Interrupt selection 00: No interrupt 01: INTADxPDA 10: INTADxPDB 11: Reserved Select the interrupt to be activated in program 0.

Note: This register must be set while ***[ADxCR0]***<ADEN> = 0.

4.2.15.3. *[ADxPREGS]* (PMD Trigger Storage Selection Register)

Bit	Bit symbol	After reset	Type	Function
31	-	0	R	Read as "0".
30:28	REGSEL7[2:0]	000	R/W	Program 7 conversion result storage register selection 000: ADxREG0 to 3 100: ADxREG16 to 19 001: ADxREG4 to 7 101: ADxREG20 to 23 010: ADxREG8 to 11 110: ADxREG24 to 27 011: ADxREG12 to 15 111: ADxREG28 to 31
27	-	0	R	Read as "0".
26:24	REGSEL6[2:0]	000	R/W	Program 6 conversion result storage register selection 000: ADxREG0 to 3 100: ADxREG16 to 19 001: ADxREG4 to 7 101: ADxREG20 to 23 010: ADxREG8 to 11 110: ADxREG24 to 27 011: ADxREG12 to 15 111: ADxREG28 to 31
23	-	0	R	Read as "0".
22:20	REGSEL5[2:0]	000	R/W	Program 5 conversion result storage register selection 000: ADxREG0 to 3 100: ADxREG16 to 19 001: ADxREG4 to 7 101: ADxREG20 to 23 010: ADxREG8 to 11 110: ADxREG24 to 27 011: ADxREG12 to 15 111: ADxREG28 to 31
19	-	0	R	Read as "0".
18:16	REGSEL4[2:0]	000	R/W	Program 4 conversion result storage register selection 000: ADxREG0 to 3 100: ADxREG16 to 19 001: ADxREG4 to 7 101: ADxREG20 to 23 010: ADxREG8 to 11 110: ADxREG24 to 27 011: ADxREG12 to 15 111: ADxREG28 to 31
15	-	0	R	Read as "0".
14:12	REGSEL3[2:0]	000	R/W	Program 3 conversion result storage register selection 000: ADxREG0 to 3 100: ADxREG16 to 19 001: ADxREG4 to 7 101: ADxREG20 to 23 010: ADxREG8 to 11 110: ADxREG24 to 27 011: ADxREG12 to 15 111: ADxREG28 to 31
11	-	0	R	Read as "0".
10:8	REGSEL2[2:0]	000	R/W	Program 2 conversion result storage register selection 000: ADxREG0 to 3 100: ADxREG16 to 19 001: ADxREG4 to 7 101: ADxREG20 to 23 010: ADxREG8 to 11 110: ADxREG24 to 27 011: ADxREG12 to 15 111: ADxREG28 to 31
7	-	0	R	Read as "0".
6:4	REGSEL1[2:0]	000	R/W	Program 1 conversion result storage register selection 000: ADxREG0 to 3 100: ADxREG16 to 19 001: ADxREG4 to 7 101: ADxREG20 to 23 010: ADxREG8 to 11 110: ADxREG24 to 27 011: ADxREG12 to 15 111: ADxREG28 to 31
3	-	0	R	Read as "0".
2:0	REGSEL0[2:0]	000	R/W	Program 0 conversion result storage register selection 000: ADxREG0 to 3 100: ADxREG16 to 19 001: ADxREG4 to 7 101: ADxREG20 to 23 010: ADxREG8 to 11 110: ADxREG24 to 27 011: ADxREG12 to 15 111: ADxREG28 to 31

Note: This register must be set with ***[ADxCR0]***<ADEN> = 0.

4.2.15.4. *[ADxPSET0]* (PMD Trigger Program Register 0)

This is an example of ***[ADxPSET0]***. The same configuration is used for ***[ADxPSET1]*** to ***[ADxPSET7]***.

Bit	Bit symbol	After reset	Type	Function
31	ENSP03	0	R/W	Conversion 3 setting: Conversion control 0: Disabled 1: Enabled
30:29	-	00	R/W	Write as "00".
28:24	AINSP03[4:0]	00000	R/W	Conversion 3 setting: AIN selection 00000: AINx00 01000: AINx08 10000: AINx16 11000:AINx24 00001: AINx01 01001: AINx09 10001: AINx17 11001:AINx25 00010: AINx02 01010: AINx10 10010: AINx18 11010:AINx26 00011: AINx03 01011: AINx11 10011: AINx19 11011:AINx27 00100: AINx04 01100: AINx12 10100: AINx20 11100:AINx28 00101: AINx05 01101: AINx13 10101: AINx21 11101:AINx29 00110: AINx06 01110: AINx14 10110: AINx22 11110:AINx30 00111: AINx07 01111: AINx15 10111: AINx23 11111:AINx31
23	ENSP02	0	R/W	Conversion 2 setting: Conversion control 0: Disabled 1: Enabled
22:21	-	00	R/W	Write as "00".
20:16	AINSP02[4:0]	00000	R/W	Conversion 2 setting: AIN selection 00000: AINx00 01000: AINx08 10000: AINx16 11000:AINx24 00001: AINx01 01001: AINx09 10001: AINx17 11001:AINx25 00010: AINx02 01010: AINx10 10010: AINx18 11010:AINx26 00011: AINx03 01011: AINx11 10011: AINx19 11011:AINx27 00100: AINx04 01100: AINx12 10100: AINx20 11100:AINx28 00101: AINx05 01101: AINx13 10101: AINx21 11101:AINx29 00110: AINx06 01110: AINx14 10110: AINx22 11110:AINx30 00111: AINx07 01111: AINx15 10111: AINx23 11111:AINx31
15	ENSP01	0	R/W	Conversion 1 setting: Conversion control 0: Disabled 1: Enabled
14:13	-	00	R/W	Write as "00"
12:8	AINSP01[4:0]	00000	R/W	Conversion 1 setting: AIN selection 00000: AINx00 01000: AINx08 10000: AINx16 11000:AINx24 00001: AINx01 01001: AINx09 10001: AINx17 11001:AINx25 00010: AINx02 01010: AINx10 10010: AINx18 11010:AINx26 00011: AINx03 01011: AINx11 10011: AINx19 11011:AINx27 00100: AINx04 01100: AINx12 10100: AINx20 11100:AINx28 00101: AINx05 01101: AINx13 10101: AINx21 11101:AINx29 00110: AINx06 01110: AINx14 10110: AINx22 11110:AINx30 00111: AINx07 01111: AINx15 10111: AINx23 11111:AINx31
7	ENSP00	0	R/W	Conversion 0 setting: Conversion control 0: Disabled 1: Enabled
6:5	-	00	R/W	Write as "00"
4:0	AINSP00[4:0]	00000	R/W	Conversion 0 setting: AIN selection 00000: AINx00 01000: AINx08 10000: AINx16 11000:AINx24 00001: AINx01 01001: AINx09 10001: AINx17 11001:AINx25 00010: AINx02 01010: AINx10 10010: AINx18 11010:AINx26 00011: AINx03 01011: AINx11 10011: AINx19 11011:AINx27 00100: AINx04 01100: AINx12 10100: AINx20 11100:AINx28 00101: AINx05 01101: AINx13 10101: AINx21 11101:AINx29 00110: AINx06 01110: AINx14 10110: AINx22 11110:AINx30 00111: AINx07 01111: AINx15 10111: AINx23 11111:AINx31

Note: This register must be set with ***[ADxCR0]***<ADEN> = 0.

4.2.16. *[ADxTSET0]* (General Purpose Start-up Factor Program Register 0)

This is an example of ***[ADxTSET0]***. The same configuration is used for ***[ADxTSET1]*** to ***[ADxTSET31]***.

Bit	Bit symbol	After reset	Type	Function
31:8	-	0	R	Read as "0".
7	ENINT0	0	R/W	Interrupt control 0: Disabled 1: Enabled
6:5	TRGS0[1:0]	00	R/W	Start-up factor 00: No conversion 01: Continuous conversion 10: Single conversion 11: General-purpose trigger conversion
4:0	AINST0[4:0]	00000	R/W	Analog input selection 00000: AINx00 01000: AINx08 10000: AINx16 11000:AINx24 00001: AINx01 01001: AINx09 10001: AINx17 11001:AINx25 00010: AINx02 01010: AINx10 10010: AINx18 11010:AINx26 00011: AINx03 01011: AINx11 10011: AINx19 11011:AINx27 00100: AINx04 01100: AINx12 10100: AINx20 11100:AINx28 00101: AINx05 01101: AINx13 10101: AINx21 11101:AINx29 00110: AINx06 01110: AINx14 10110: AINx22 11110:AINx30 00111: AINx07 01111: AINx15 10111: AINx23 11111:AINx31

4.2.17. *[ADxREG0]* (Conversion Result Storage Register 0)

This is an example of ***[ADxREG0]***. The same configuration is used for ***[ADxREG1]*** to ***[ADxREG31]***.

Bit	Bit symbol	After reset	Type	Function
31:30	-	0	R	Read as "0".
29	ADOVR_M0	0	R	Mirror bit of overrun flag <ADOVRF0>
28	ADRF_M0	0	R	Mirror bit of AD conversion result storage flag <ADRF0>
27:16	ADR_M0[11:0]	0x000	R	Mirror field of AD conversion result <ADR0[11:0]>. The AD conversion result is stored in the lower 12 bits of the upper half word of the ***[ADxREG0]***.
15:4	ADR0[11:0]	0x000	R	AD conversion result The AD conversion result is stored in the upper 12 bits of the lower half-word of the ***[ADxREG0]***.
3:2	-	0	R	Read as "0".
1	ADOVRF0	0	R	Overrun flag 0: Not occurred. 1: Occurred. This flag is set to "1" when an AD conversion result is overwritten before reading the ***[ADxREG0]***. This flag is cleared to "0" when it is read.
0	ADRF0	0	R	AD conversion result storage flag 0: Conversion result not stored 1: Conversion result stored This flag is set to "1" when an AD conversion result is stored. This flag is cleared to "0" when it is read.

5. Example of Usage

5.1. Setting Example of Single Conversion

The single conversion is started by the software. Multiple conversions are possible.

In the following setting example, the conversion results of the two analog inputs (AINx02, AINx03) are stored in two conversion result storage registers (***[ADxREG4]***, ***[ADxREG5]***) and a single conversion interrupt INTADxSGL is generated at the end of the second conversion.
ADCLK is divided by 2, m is set to "2", and n is set to "4".

- Initial setting
  - ***[ADxMOD0]*** = 0x00000001
    DAC control: <DACON> = 1 (ON)
    Low-power mode selection: <RCUT> = 0 (Normal operation)
  - ***[ADxCLK]*** = 0x00000000
    AD prescaler output selection: <VADCLK[2:0]> = 000 (ADCLK / 2)
    m value 0: <EXAZ0[3:0]> = 0001 (m = 2)
  - ***[ADxMOD1]*** = 0x00004000
    n value: <MOD1[31:0]> = 0x00000000 (n = 4)
  - ***[ADxEXAZSEL]*** = 0x00000000
    Sampling time selection: <EXAZSEL[31:0]> = 0x00000000 (<EXAZ0[3:0]> is used.)
  - ***[ADxMOD2]*** = Specified value
    [ADxTRM] = Specified value

    Note: The setting value varies depending on the product. For the setting value, refer to the reference manual "Product Information".

- Conversion setting
  - ***[ADxTSET4]*** = 0x00000042
    Analog input selection: <AINST4[4:0]> = 00010 (AINx02)
    Start-up factor: <TRGS4[1:0]> = 10 (Single conversion)
    Interrupt control: < ENINT4> = 0 (Disabled)
  - ***[ADxTSET5]*** = 0x000000C3
    Analog input selection: <AINST5[4:0]> = 00011 (AINx03)
    Start-up factor: <TRGS5[1:0]> = 10 (Single conversion)
    Interrupt control: < ENINT5> = 1 (Enabled)

- Conversion start setting
  - ***[ADxCR1]*** = 0x00000000
    Disable DMA request
  - ***[ADxCR0]*** = 0x00000082
    Continuous conversion control: <CNT> = 0 (Disabled)

Single conversion control:	<SGL> = 1 (Conversion start)
ADC control:	<ADEN> = 1 (Enabled)

5.2. Example of Motor Control with PMD Trigger

5.2.1. 3-shunt Method

The following diagram shows the configuration when PMD channel 0 and ADC unit A are used.

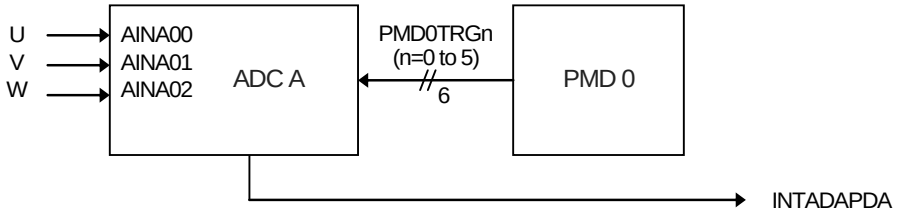


Figure 5.1 3-shunt Example

Table 5.1 shows an example of ADC settings in this case.

Table 5.1 ADC Setting in 3 Shunt

Program	0	1	2	3	4	5
Reg0	U	V	W	V	W	U
Reg1	V	W	U	U	V	W
Interrupt	INTADAPDA	INTADAPDA	INTADAPDA	INTADAPDA	INTADAPDA	INTADAPDA

Program numbers 0 to 5 are assigned to the six trigger inputs PMD0TRG0 to 5 by ***[ADAPSEL0]*** to ***[ADAPSEL5]***. Reg0 and Reg1 in the table represent ***[ADAPSETn]***[7:0] and ***[ADAPSETn]***[15:8] (n: Program number), respectively. U, V, and W in the table represent the motor phase. Select the analog input that will provide the respective phases.

When the PMD triggers are input, AD conversion is performed in the order of Reg0 and Reg1, and the conversion results are stored in the conversion result storage register. When all conversions are completed, an INTADAPDA interrupt is output.

5.2.2. 1-shunt Method

The following diagram shows the configuration when PMD channel 0 and ADC unit A are used.

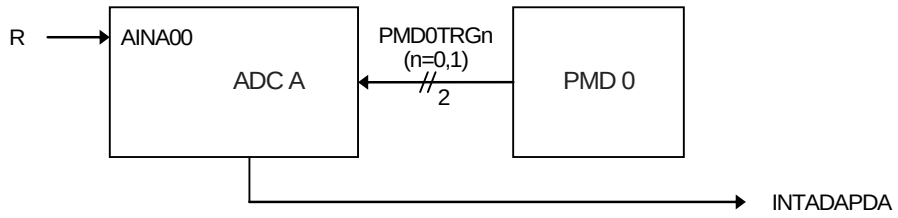


Figure 5.2 1-shunt Example

Table 5.2 shows an example of ADC settings in this case.

Table 5.2 ADC Setting in 1 Shunt

Trigger	PMD0	PMD0
	0	1
Program	0	1
Reg0	R	-
Reg1	-	R
Interrupt	-	INTADAPDA

Program numbers 0 and 1 are assigned to the two trigger signals PMD0TRG0 and PMD0TRG1.

Reg0 and Reg1 in the table represent ***[ADAPSETn]***[7:0] and ***[ADAPSETn]***[15:8] (n: Program number), respectively. R in the table represent the resistor. Select the corresponding analog input.

When the PMD triggers are input, AD conversion is performed and the conversion result is stored in the conversion result storage registers 0 and 1. Conversions are executed in the order of program 0, 1, and an INTADAPDA interrupt is output when completed.

6. Precaution

- The following factors may have an impact on the accuracy of the conversion results.
  - Voltage fluctuations of the power supply
  - Level fluctuations of pins near the analog input pin to be converted

- To obtain the desired conversion result, the conversion must be performed with the pin voltage and internal capacitors stable. Particular attention should be paid when the output impedance of the analog signal source is high or when conversions are being performed continuously.
  Adjust the time for stabilization according to usage conditions by:
  - Extend the sampling time.
  - Extend the time between the end of a conversion and the start of the next conversion.

7. Revision History

Table 7.1 Revision History

Revision	Date	Description
1.0	2024-07-22	- First release
1.1	2024-11-29	- 4.2.16. [ADxTSET0] (General Purpose Start-up Factor Program Register 0) Deleted note

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.**