32-bit RISC Microcontroller

TXZ Family

Reference Manual
12-bit Analog to Digital Converter (ADC-C)

Revision 3.0

2019-06

TOSHIBA ELECTRONIC DEVICES & STORAGE CORPORATION

Contents

List of Figures

List of Tables

Preface

Related document

Document name
Exception
Clock Control and Operation Mode
Product Information

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 bit 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.

The Flash memory uses the Super Flash® technology under license from Silicon Storage Technology, Inc.
Super Flash® is registered trademark of Silicon Storage Technology, Inc.

All other company names, product names, and service names mentioned herein may be trademarks of their respective companies.

Terms and Abbreviation

Some of abbreviations used in this document are as follows:

ADC	Analog to Digital Converter
TRGSEL	Trigger Selection circuit

1. Outlines

The 12-bit analog to digital converter (ADC) can convert multiple analog inputs(AINx0 to AINxn) to digital in each unit. The function list is shown as follows.

Function classification	Function	Operation explanation
AD conversion	Conversion resolution	12bits
	Conversion time	AVDD3 =2.7 to 3.6[V]: 1.0 to 5.0 [μs] Minimum conversion time condition: 1.0 [μs] at ADCLK=60[MHz]
	Store conversion result	24 conversion result storage registers.
Start conversion	Highest start-up factor	- Highest priority trigger - Highest priority conversion operation (software) Start-up factor program (Note1) can perform up to 24 AD conversions at each start-up.
	General purpose start-up factor	- General purpose trigger conversion - Software (Continuous conversion operation, Single conversion operation) Start-up factor program (Note1) can perform up to 24 AD conversions at each start-up.
Conversion status	Status flags	- Flag showing that the AD conversion is executing. - Flag showing that the program is executing (for each trigger). - Conversion result storage flag (for each conversion result storage register). - Conversion result overrun flag (for each conversion result storage register).
Interrupt	-	- Highest priority program AD conversion completion (INTADxHP) - General purpose trigger program completion (INTADxTRG) - Software single conversion program completion (INTADxSGL) - Software continuous conversion program completion (INTADxCNT) - Monitor function interrupt (INTADxCP0, INTADxCP1, INTADxCP2, INTADxCP3) (Note2)
Monitor conversion result	AD monitor function	- Each ADC unit has 4 channels of monitor function. - Selectable conversion result storage register to be monitored. - Selectable detection method: Whether the target register value is larger or smaller than the comparison register. - Selectable number of detections. - Selectable continuous count and accumulated count.

Note1: Conversion program can specify conversion channel (analog input) and presence / absence of interrupt. There are multiple programs. Each is started with the start-up factor / trigger.

Note2: For the channel of the AD monitor function, refer to “Product Information” in the reference manual.

Figure 1.1 shows the connection relationships with the peripheral functions that are linked with the ADC. AD conversion can be started from general trigger, highest priority trigger, etc.

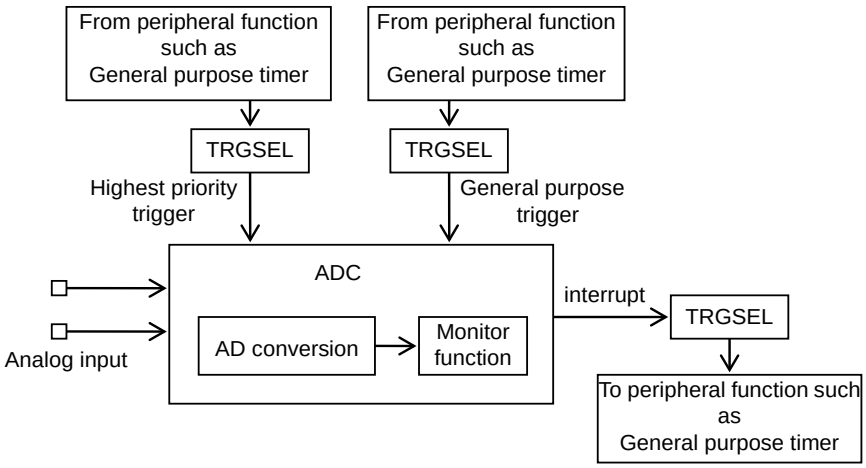


Figure 1.1 Related figure of ADC and another peripheral function

2. Block Diagram

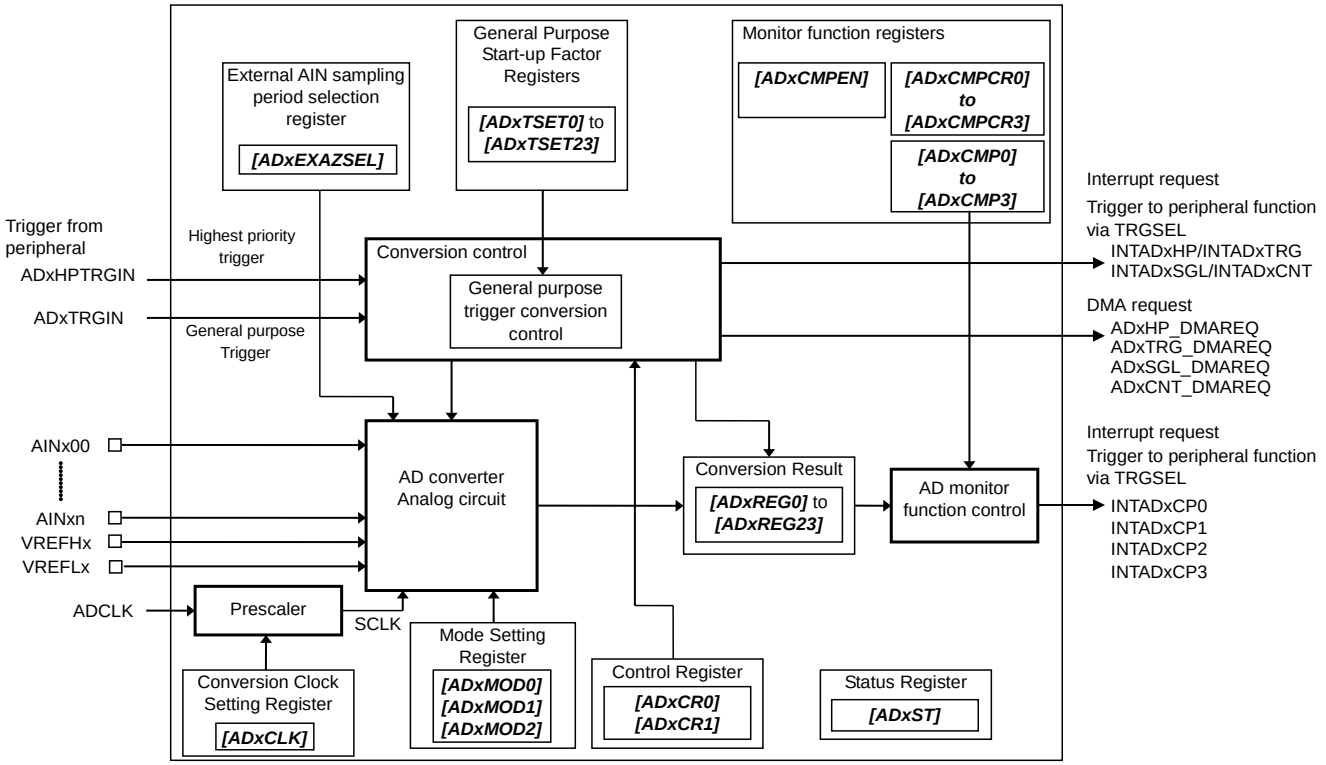


Figure 2.1 ADC block diagram

Table 2.1 List of Signals

No	Signal name		I/O	Related Reference manual
1	ADCLK	Conversion clock for ADC	Input	Clock Control and Operation Mode
2	AINx00 to AINxn	Analog input pin	Input	Product Information
3	VREFHx	Reference power pin for analog	Input	Product Information
4	VREFLx	Reference GND pin for analog	Input	Product Information
5	ADxHPTRGIN	Highest priority trigger input	Input	Product Information
6	ADxTRGIN	General purpose trigger	Input	Product Information
7	INTADxHP	Highest priority conversion completion interrupt	Output	Exception, Product Information
8	INTADxTRG	General purpose trigger interrupt	Output	Exception, Product Information
9	INTADxSGL	Single conversion interrupt	Output	Exception, Product Information
10	INTADxCNT	Continuous conversion interrupt	Output	Exception, Product Information
11	INTADxCP0	Monitor function interrupt 0	Output	Exception, Product Information
12	INTADxCP1	Monitor function interrupt 1	Output	Exception, Product Information
13	INTADxCP2	Monitor function interrupt 2	Output	Exception, Product Information
14	INTADxCP3	Monitor function interrupt 3	Output	Exception, Product Information
15	ADxHP_DMAREQ	Highest priority DMA request	Output	Product Information
16	ADxTRG_DMAREQ	General purpose trigger DMA request	Output	Product Information
17	ADxSGL_DMAREQ	Single conversion DMA request	Output	Product Information
18	ADxCNT_DMAREQ	Continuous conversion DMA request	Output	Product Information

3. Function and Operation

The ADC is triggered to start the conversion by the software start-up (Software trigger) or the trigger signal from a timer, and others.

3.1. Clock Supply

When using ADC, set an applicable clock enable bit to "1" (clock supply) in fsys supply stop register A (***[CGFSYSENA]***, ***[CGFSYSMENA]***), fsys supply stop register B (***[CGFSYSENB]***, ***[CGFSYSMENB***]), and fc supply stop register (***[CGFCEN]***). Set the AD converter conversion clock enable bit to “1” in Clock supply and stop register for ADC and TRACE (***[CGSPCLKEN]***).
An applicable register and the bit position vary according to a product. Therefore, the register may not exist with the product. Please refer to "Clock Control and Operation Mode" of the reference manual for the details.

When the clock is stopped, it should be checked that AD conversion is not executed. And when the mode transits to STOP mode or STOP1 mode, it should be checked that AD conversion is not executed, too.

3.2. Conversion Operation

The conversion is started by the highest priority start-up factor and the general purpose start-up factor.
The highest priority start-up factor is the factor to start up the highest priority conversion. It has two conversion start-up operations (the highest priority trigger and the highest conversion start-up). And the general purpose start-up factor has three start-up operations (a general purpose trigger, a single conversion start-up, and continuous conversion start-up).

Table 3.1 Start-up factor

Highest start-up factor	General purpose start-up factor
(a) Highest priority conversion - Highest priority trigger -Highest priority conversion operation (software)	(b) General purpose trigger conversion (c) Single conversion operation (software) (d) Continuous conversion operation (software)

Note: (a) to (d) start-up factors are shown in Figure 3.1

3.2.1. Operation

When the conversion is triggered by start-up factor, the conversion executes according to the setting in start-up factor program register which is prepared for each conversion result register.

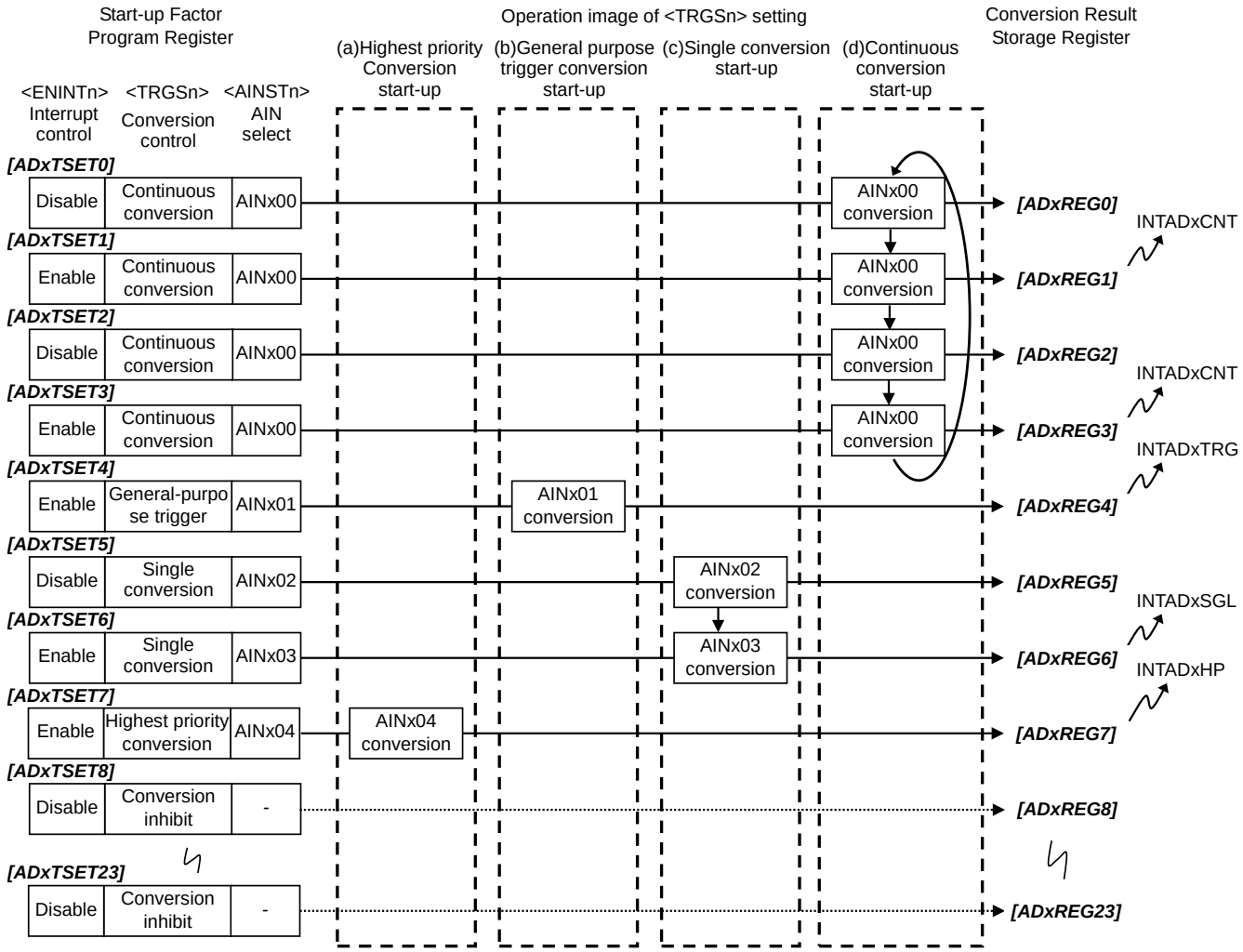


Figure 3.1 Start-up factor and operation (Example)

The start-up factor select (Conversion control), the AIN select, and the interrupt enable or disable (Interrupt control) are programmed to the start-up factor program register. When the start-up factor occurs, the specified conversions are executed from the smallest number of the register.

(a) Highest priority conversion

The highest priority conversion is started by the following two factors.
The highest priority trigger conversion:
When the highest priority trigger is input, the conversion which is specified to the highest priority is executed once.
The highest conversion start-up (software):
The conversion which is specified to the highest priority is executed once, too.

(b) General purpose trigger conversion

For general purpose trigger conversion, when a general purpose trigger is input, the conversion specified for general purpose trigger conversion is executed once.

(c) Single conversion operation (software)

Single conversion executes the conversion specified for single conversion once.

(d) Continuous conversion operation (software)

Continuous conversion operation iteratively executes the conversion specified for continuous conversion.

When the interrupt is enabled (***[ADxTSETn]***<ENINTn>=1), the interrupt is generated at the conversion completion for any trigger causes (General purpose trigger, Single conversion, Continuous conversion). The interrupt request of each start-up factor(INTADxHP, INTADxTRG, INTADxSGL, INTADxCNT) is different from others.

The start-up factor can generate a DMA request per cause. When a DMA request is enabled (***[ADxCR1]***<HPDMEN>, <CNTDMEN>,<SGLDMEN>,<TRGDMEN>=1), a DMA request and an interrupt request are generated simultaneously.

Table 3.2 Factor and interrupt / DMA request

Factor	Interrupt	DMA request
Highest priority conversion	Highest priority program AD conversion completion (INTADxHP)	Highest priority conversion DMA request (ADxHP_DMAREQ)
General purpose trigger conversion	General purpose trigger program completion (INTADxTRG)	General purpose trigger DMA request (ADxTRG_DMAREQ)
Single conversion operation (software)	Software single conversion program completion (INTADxSGL)	single conversion DMA request (ADxSGL_DMAREQ)
Continuous conversion operation (software)	Software continuous conversion program completion (INTADxCNT)	continuous conversion DMA request (ADxCNT_DMAREQ)

3.2.2. Control Register

- Start-up factor program register (***[ADxTSET0]*** to ***[ADxTSET23]***)

  Start-up factor program register is prepared for each conversion result storage register. The AIN select <AINSTn>, the conversion control <TRGSn>, and the interrupt control <ENINTn> are set to ***[ADxTSETn]***.

- Mode register0 (***[ADxMOD0]***)

  When using the ADC, set "1" to ***[ADxMOD0]***<DACON>. And the interval of 3[μs] are necessary for the stabilization.

- Control register0 (***[ADxCR0]***)

  When the AD conversion can be started, after setting, ***[ADxCR0]***<ADEN> should be set to "1".
  The software single conversion or the software continuous conversion is enabled by setting ***[ADxCR0]*** <SGL> or <CNT> to "1", respectively. When the continuous conversion should be stopped, <CNT> is set to "0".
  Also, when starting the highest priority conversion by software, write “1” to ***[ADxCR0]***<HPSGL>.

- Control register1 (***[ADxCR1]***)

  [ADxCR1]<TRGEN> enables the trigger, and then the program start-up is done by the general purpose trigger. The conversion starts when a trigger is received.
  [ADxCR1]<SGLDMEN><CNTDMEN><TRGDMEN> are set to “1” to enable the DMA request generation.

  To enable the program to be activated by the highest priority trigger, write “1” to ***[ADxCR1]***<HPTRGEN>. In this state, conversion is started when a trigger is input. To enable generation of DMA request, write “1” to ***[ADxCR1]***<HPDMEN>.

  Note: ***[ADxCR1]*** register must be set while ***[ADxCR0]***<ADEN> =0.

3.2.3. Conversion start procedure

To start the conversion operation, set the register in the following procedure.

- Highest priority trigger conversion
  (1) Set interrupt to use INTADxHP.
  (2) Set "1" to ***[ADxMOD0]*** <DACON>.
  (3) Wait at least 3[μs].
  (4) Set "1" to ***[ADxCR1]***<HPTRGEN>.
  (5) Sets what trigger to use for the highest priority trigger (ADxHPTRGIN). (Note)
  (6) Set ***[ADxTSETn]***. AIN selection <AINSTn> = arbitrary, conversion control <TRGSn> = 1xx, interrupt control <ENINTn> = 1.
  (7) To activate the highest priority trigger using multiple channels, change the AIN selection and set (6) again.
  (8) Set "1" to ***[ADxCR0]***<ADEN>.
  (9) When you input a trigger, conversion starts.
  (10) When conversion is complete, INTADxHP will be generated. Read ***[ADxREGn]*** in the interrupt service routine.
  (11) Repeat steps (9) to (10).

  Note: For details of the signal connected to the highest priority trigger (ADxHPTRGIN), refer to the reference manual "Product Information".

- Highest priority conversion
  (1) Set interrupt to use INTADxHP.
  (2) Set "1" to ***[ADxMOD0]***<DACON>.
  (3) Wait at least 3[μs].
  (4) Set ***[ADxTSETn]***. AIN selection <AINSTn> = arbitrary, conversion control <TRGSn> = 1xx, interrupt control <ENINTn> = 1.
  (5) To perform the highest priority conversion using multiple channels, change the AIN selection and set (4) again.
  (6) Set "1" to ***[ADxCR0]***<ADEN>.
  (7) Set "1" to ***[ADxCR0]***<HPSGL>, starts the conversion.
  (8) When conversion is complete, INTADxHP will be generated. Read ***[ADxREGn]*** in the interrupt service routine.
  (9) Repeat steps (7) to (8).

- General purpose trigger conversion
  (1) Set interrupt to use INTADxTRG.
  (2) Set "1" to ***[ADxMOD0]***<DACON>.
  (3) Wait at least 3[μs].
  (4) Set "1" to ***[ADxCR1]***<TRGEN>.
  (5) Sets what trigger to use for the general purpose trigger (ADxTRGIN). (Note)
  (6) Set ***[ADxTSETn]***. AIN selection <AINSTn> = arbitrary, conversion control <TRGSn> = 011, interrupt control <ENINTn> = 1.
  (7) To activate the general purpose trigger using multiple channels, change the AIN selection and set (6) again.
  (8) Set "1" to ***[ADxCR0]***<ADEN>.
  (9) When you input a trigger, conversion starts.
  (10) When conversion is complete, INTADxTRG will be generated. Read ***[ADxREGn]*** in the interrupt service routine.
  (11) Repeat steps (9) to (10).

  Note: For details of the signal connected to the general purpose trigger (ADxTRGIN), refer to the reference manual "Product Information".

- Single conversion operation
  (1) Set interrupt to use INTADxSGL.
  (2) Set "1" to ***[ADxMOD0]***<DACON>.
  (3) Wait at least 3[μs].
  (4) Set ***[ADxTSETn]***. AIN selection <AINSTn> = arbitrary, conversion control <TRGSn> = 010, interrupt control <ENINTn> = 1.
  (5) To perform the Single conversion using multiple channels, change the AIN selection and set again.
  (6) Set "1" to ***[ADxCR0]***<ADEN>.
  (7) Set "1" to ***[ADxCR0]***<SGL>, starts the conversion.
  (8) When conversion is complete, INTADxSGL will be generated. Read ***[ADxREGn]*** in the interrupt service routine.
  (9) Repeat steps (7) to (8).

- Continuous conversion
  (1) Set interrupt to use INTADxCNT.
  (2) Set "1" to ***[ADxMOD0]***<DACON>.
  (3) Wait at least 3[μs].
  (4) Set ***[ADxTSETn]***. AIN selection <AINSTn> = arbitrary, conversion control <TRGSn> = 001, interrupt control <ENINTn> = 1.
  (5) To perform the continuous conversion using multiple channels, change the AIN selection and set again.
  (6) Set "1" to ***[ADxCR0]***<ADEN>.
  (7) Set "1" to ***[ADxCR0]***<CNT>, starts the conversion.
  (8) When conversion is complete, INTADxCNT will be generated. Read ***[ADxREGn]*** in the interrupt service routine.
  (9) Repeat steps (8).

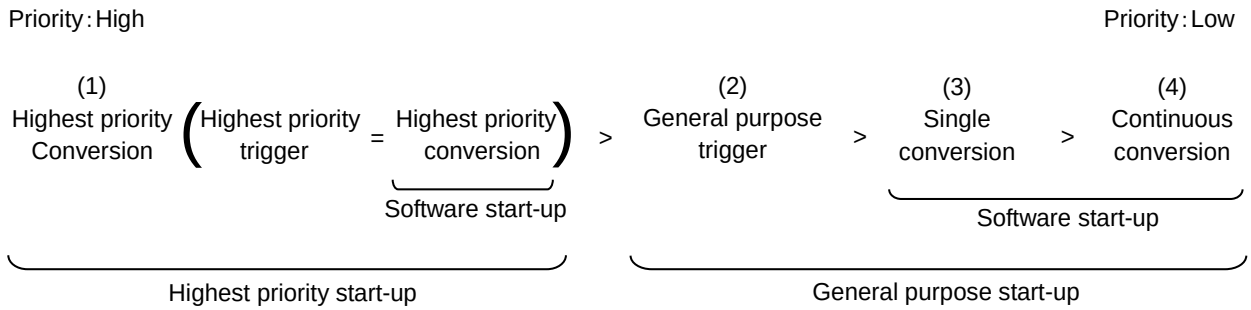
3.3. Conversion Stop

When ***[ADxCR0]***<ADEN> is set to “0”, the conversion stops immediately. If the continuous conversion is enabled, ***[ADxCR0]***<CNT> should be also set to “0”.
When the conversion stops completely, all bits in ***[ADxST]*** become “0”. The registers other than ***[ADxST]*** keep their data, as well as the conversion result registers. Before the next conversion is enabled, the conversion result registers should be read to clear the corresponding flags.

Before ADCLK is stopped, ***[ADxST]***<ADBF> =0 should be confirmed.

3.4. Start-up Priority

The start-up factors are prioritized as follows.



Once a Highest priority conversion program starts to execute, it is never suspended.
In general purpose trigger, single conversion, and continuous conversion programs, when a high priority factor is generated, execution of the current program is interrupted and a program with a higher priority is executed. When a low priority factor occurs, it waits for execution.

The programs of suspended general purpose trigger, single conversion, continuous conversion re-starts from suspended conversion when they become executable.

When the start-up factor is generated again during execution of the program of the same start-up factor, the factor is ignored. The status of the program can be checked by ***[ADxST]***<CNTF><SNGF><TRGF><HPF>. For the software start-up factors, it should be confirmed whether the corresponding flags are "0". Then, the start-up is certainly executed.

Table 3.3 Operation when the start-up factor occurs during the conversion

		Later start-up factor			
		Highest priority conversion	**General purpose trigger**	**Software Single conversion**	**Software Continuous conversion**
Current start-up factor during conversion	Highest priority conversion	Continue current factor (Note2)	Continue current factor (Note1)	Continue current factor (Note1)	Continue current factor (Note1)
	General purpose trigger	Start later factor (Note3)	Continue current factor (Note2)	Continue current factor (Note1)	Continue current factor (Note1)
	Software Single conversion	Start later factor (Note3)	Start later factor (Note3)	Continue current factor (Note2)	Continue current factor (Note1)
	Software Continuous conversion	Start later factor (Note3)	Start later factor (Note3)	Start later factor (Note3)	Continue current factor (Note2)

Note1: The later factor is performed after the current factor is completed.

Note2: The later factor is ignored.

Note3: The current factor is suspended. And then, the current factor re-start after the later factor is completed.

3.5. AD Monitor Function

The AD monitor function generates an interrupt if the AD conversion result is larger than the set value or smaller. It is possible to detect whether the AD conversion result is within the range of four set values or to detect whether the AD conversion result is out of the range by using this function simultaneously in four channels.

When ***[ADxCMPEN]***<CMP0EN>,<CMP1EN>,<CMP2EN> and <CMP2EN> is set to "1", the corresponding AD monitor function is enabled. The four monitor functions can be enabled simultaneously.

The following description is for ***[ADxCMPCR0]*** (The same for ***[ADxCMPCR1]***, ***[ADxCMPCR2]*** and ***[ADxCMPCR3]***).

[ADxCMPCR0]<REGS0[4:0]> sets the conversion result storage register which value should be compared. <ADBIG0> sets the determination condition (larger or smaller). <CMPCND0> sets the determination count condition. And <CMPCNT0[3:0]> sets the determination count value.

Whenever a conversion result is stored to the target conversion result storage register, the result is compared (larger or smaller). If the comparison result is the same as the <ADBIG0> setting, the determination counter increments.

The determination count condition is either the continuous count or the accumulated count.

The continuous count condition is as follows: when the status set in <ADBIG0> continues the count times set in <CMPCNT0[3:0]>, the AD monitor function interrupt (INTADxCP0). When it continues exceeding the set-up count number, nothing occurs. If the status is different from the <ADBIG0> status, the counter is cleared.

The accumulated count condition is as follows: when the count of the status set in <ADBIG0> is accumulated and the accumulated value reaches the value set in <CMPCNT0[3:0]>, the AD monitor function interrupt (INTADxCP0), and the counter is cleared. Even when the status is different from the status set in <ADBIG0>, the counter value is maintained. When the value in the conversion result storage register specified by the ***[ADxCMPCR0]*** register is equal to the value in the conversion result comparison register, the counter does not increment and the AD monitor function interrupt and the trigger are not generated.

Table 3.4 Monitor function and interrupt

Monitor function	Interrupt
Monitor function Setting Register0 (***[ADxCMPCR0]***)	Monitor function 0 Interrupt (INTADxCP0)
Monitor function Setting Register1 ([***ADxCMPCR1***])	Monitor function 1 Interrupt (INTADxCP1)
Monitor function Setting Register2 ([***ADxCMPCR2***])	Monitor function 2 Interrupt (INTADxCP2)
Monitor function Setting Register3 ([***ADxCMPCR3***])	Monitor function 3 Interrupt (INTADxCP3)

Note: For the handling of the interrupts, refer to "Product Information" in Reference manual.

When the AD monitor function is used, the overrun flag ***[ADxREGn]***<ADOVRFn> and the conversion result storage flag ***[ADxREGn]***<ADRFn> are set because the storage register is not read by the software. So, when the AD monitor function is executing, the flags of the corresponding conversion result storage registers should not be used.

Note: The monitor function registers must be set while ***[ADxCR0]***<ADEN> =0.

(1) Determination by Continuous count

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

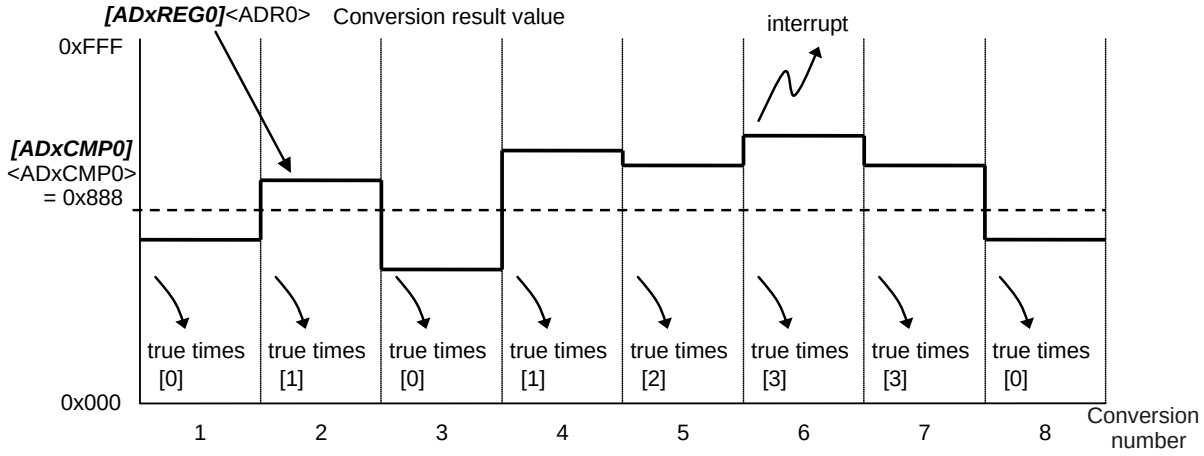


Figure 3.2 AD monitor function (Determination condition: Continuous count)

(2) Determination by Accumulated count

- Monitor function setting register (***[ADxCMPCR0]*** =0x00000240)
  Conversion result storage register (Comparison target): ***[ADxREG0]***
  Magnitude determination: ***[ADxREG0]***<ADR0> > ***[ADxCMP0]*** (Larger than the comparison register.)
  Determination count condition: Accumulated count
  Magnitude determination count: 3 counts
- AD conversion result comparison register (***[ADxCMP0]***<ADxCMP0> =0x888)
- Monitor function enable register (***[ADxCMPEN]*** =0x00000001)

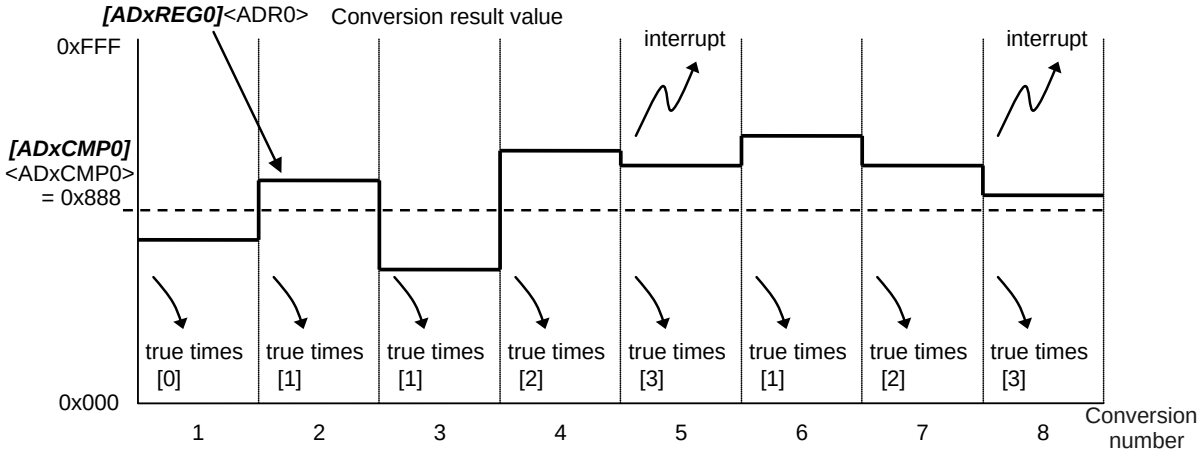


Figure 3.3 AD monitor function (Determination condition: Accumulated count)

3.6. Analog Reference Voltage

Analog reference pins VREFHx and VREFLx in the ADC unit are connected to a High level and a Low level, respectively. When ***[ADxMOD0]***<RCUT> is set to "1", the switch between VREFHx and VREFLx is turned on only during the conversion to reduce the power consumption.

The VREFHx and VREFLx pins are shared with the AVDD3 and AVSS pins depending on the product. For details, refer to "Product Information" in the reference manual.

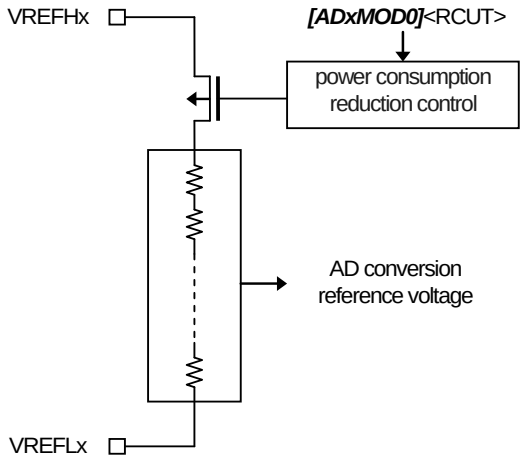


Figure 3.4 Configuration of Analog reference voltage

3.7. Conversion Time

3.7.1. Conversion timing

Conversion time is shown Figure 3.5.

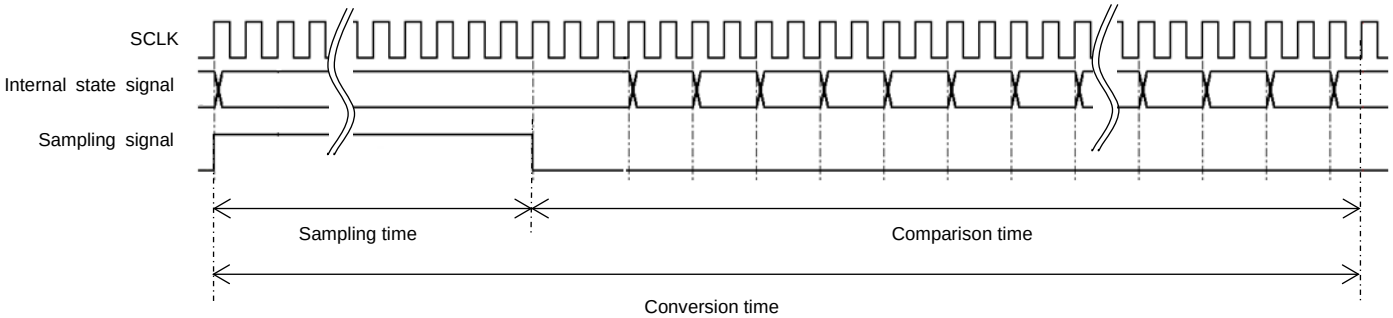


Figure 3.5 Example of Conversion time

3.7.2. Sampling time

The sampling time is set with ***[ADxCLK]***<EXAZ0> or <EXAZ1>,<VADCLK>.
Two types of sampling time can be set, and sampling time can be selected for each AIN channel.

Sampling time = ***[ADxCLK]***<EZAZ0> or <EXAZ1> × n
= SCLK period × m × n
(m: <EXAZ0> or <EZAZ1>, n: SCLK period condition value)

The SCLK period condition value (n) has a different value depending on the SCLK period.
The SCLK period condition value (n) is shown in the table below.

Table 3.5 SCLK period condition value (n)

SCLK period	SCLK period condition value (n)
SCLK ≤ 40MHz	10
40MHz < SCLK ≤ 50MHz	14
50MHz < SCLK ≤ 60MHz	16
60MHz < SCLK ≤ 80MHz	20

An example of sampling time setting is shown in the table below.

Table 3.6 Example of setting of sampling time (Unit: μs)

SCLK (MHz)	***[ADxCLK]***<EXAZ0> or <EXAZ1>			
	0000	0001	0010	0011
40	0.25	0.50	0.75	-
50	0.28	0.56	0.84	1.12
60	0.27	0.53	0.80	1.07
80	0.25	0.50	0.75	1.00

-: Cannot be set

3.7.3. Selection of sampling time

Select the sampling time set by ***[ADxCLK]*** <EXAZ0> or <EXAZ1> for each AIN channel with the external AIN sampling period selection register (***[ADxEXAZSEL]***).

3.7.4. Setting of Conversion time

The conversion time can be obtained by the following formula.

Conversion time = Sampling time + Comparison time

(1) SCLK ≤ 60MHz

Conversion time = Sampling time + (SCLK period × 44) [μs]

Note: Refer to Table 3.6 for sampling time.

(2) SCLK > 60MHz

Conversion time = Sampling time + (SCLK period × 72) [μs]

Note: Refer to Table 3.6 for sampling time.

Example of conversion time is shown below.

Table 3.7 Example of setting of conversion time (Unit: μs)

SCLK (MHz)	***[ADxCLK]***<EXAZ0> or <EXAZ1>			
	0000	0001	0010	0011
40	1.35	2.70	4.05	-
50	1.16	2.32	3.48	4.64
60	1.00	2.00	3.00	4.00
80	1.15	2.30	3.45	4.60

-: Cannot be set

4. Registers

4.1. List of Registers

The control registers and their addresses are shown as follows.

Function		Channel/Unit	Base address	
			TYPE 1	TYPE 2
12-bit Analog to Digital Converter	ADC	Unit A	0x40050000	0x400BA000
		Unit B	0x40051000	0x400BA400
		Unit C	0x40052000	0x400BA800
		Unit D	0x40053000	0x400BAC00

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

Register Name		Address (Base+)
Control Register0	***[ADxCR0]***	0x0000
Control Register1	***[ADxCR1]***	0x0004
Status Register	***[ADxST]***	0x0008
Conversion Clock Setting Register	***[ADxCLK]***	0x000C
Mode Setting Register0	***[ADxMOD0]***	0x0010
Mode Setting Register1	***[ADxMOD1]***	0x0014
Mode Setting Register2	***[ADxMOD2]***	0x0018
Monitor function Enable Register	***[ADxCMPEN]***	0x0020
Monitor function Setting Register0	***[ADxCMPCR0]***	0x0024
Monitor function Setting Register1	***[ADxCMPCR1]***	0x0028
Conversion Result Comparison Register0	***[ADxCMP0]***	0x002C
Conversion Result Comparison Register1	***[ADxCMP1]***	0x0030
Monitor function Setting Register2	***[ADxCMPCR2]***	0x0034
Monitor function Setting Register3	***[ADxCMPCR3]***	0x0038
Conversion Result Comparison Register2	***[ADxCMP2]***	0x003C
Conversion Result Comparison Register3	***[ADxCMP3]***	0x0040
External AIN sampling period selection Register	***[ADxEXAZSEL]***	0x00BC
Start-up Factor Program Register0	***[ADxTSET0]***	0x00C0
Start-up Factor Program Register1	***[ADxTSET1]***	0x00C4
Start-up Factor Program Register2	***[ADxTSET2]***	0x00C8
Start-up Factor Program Register3	***[ADxTSET3]***	0x00CC
Start-up Factor Program Register4	***[ADxTSET4]***	0x00D0
Start-up Factor Program Register5	***[ADxTSET5]***	0x00D4
Start-up Factor Program Register6	***[ADxTSET6]***	0x00D8
Start-up Factor Program Register7	***[ADxTSET7]***	0x00DC
Start-up Factor Program Register8	***[ADxTSET8]***	0x00E0

Register Name		Address (Base+)
Start-up Factor Program Register9	***[ADxTSET9]***	0x00E4
Start-up Factor Program Register10	***[ADxTSET10]***	0x00E8
Start-up Factor Program Register11	***[ADxTSET11]***	0x00EC
Start-up Factor Program Register12	***[ADxTSET12]***	0x00F0
Start-up Factor Program Register13	***[ADxTSET13]***	0x00F4
Start-up Factor Program Register14	***[ADxTSET14]***	0x00F8
Start-up Factor Program Register15	***[ADxTSET15]***	0x00FC
Start-up Factor Program Register16	***[ADxTSET16]***	0x0100
Start-up Factor Program Register17	***[ADxTSET17]***	0x0104
Start-up Factor Program Register18	***[ADxTSET18]***	0x0108
Start-up Factor Program Register19	***[ADxTSET19]***	0x010C
Start-up Factor Program Register20	***[ADxTSET20]***	0x0110
Start-up Factor Program Register21	***[ADxTSET21]***	0x0114
Start-up Factor Program Register22	***[ADxTSET22]***	0x0118
Start-up Factor Program Register23	***[ADxTSET23]***	0x011C
Conversion Result Storage Register0	***[ADxREG0]***	0x0140
Conversion Result Storage Register1	***[ADxREG1]***	0x0144
Conversion Result Storage Register2	***[ADxREG2]***	0x0148
Conversion Result Storage Register3	***[ADxREG3]***	0x014C
Conversion Result Storage Register4	***[ADxREG4]***	0x0150
Conversion Result Storage Register5	***[ADxREG5]***	0x0154
Conversion Result Storage Register6	***[ADxREG6]***	0x0158
Conversion Result Storage Register7	***[ADxREG7]***	0x015C
Conversion Result Storage Register8	***[ADxREG8]***	0x0160
Conversion Result Storage Register9	***[ADxREG9]***	0x0164
Conversion Result Storage Register10	***[ADxREG10]***	0x0168
Conversion Result Storage Register11	***[ADxREG11]***	0x016C
Conversion Result Storage Register12	***[ADxREG12]***	0x0170
Conversion Result Storage Register13	***[ADxREG13]***	0x0174
Conversion Result Storage Register14	***[ADxREG14]***	0x0178
Conversion Result Storage Register15	***[ADxREG15]***	0x017C
Conversion Result Storage Register16	***[ADxREG16]***	0x0180
Conversion Result Storage Register17	***[ADxREG17]***	0x0184
Conversion Result Storage Register18	***[ADxREG18]***	0x0188
Conversion Result Storage Register19	***[ADxREG19]***	0x018C
Conversion Result Storage Register20	***[ADxREG20]***	0x0190
Conversion Result Storage Register21	***[ADxREG21]***	0x0194
Conversion Result Storage Register22	***[ADxREG22]***	0x0198
Conversion Result Storage Register23	***[ADxREG23]***	0x019C

4.2. Details of Registers

4.2.1. *[ADxCR0]* (Control Register0)

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. When “1” is set, the conversion is enabled. When “0” is set, the conversion stops.
6:3	-	0	R	Read as “0”
2	HPSGL	0	W	Highest priority conversion control 0: Don’t care 1: Conversion start. When “1” is set, the highest priority conversion program starts to execute. If this bit is read, “0” is returned.
1	SGL	0	W	Single conversion control 0: Don’t care 1: Conversion start. When “1” is set, the single conversion program starts to execute. If this bit is read, “0” is returned.
0	CNT	0	R/W	Continuous conversion control 0: Disabled. 1: Enabled. When “1” is set, the continuous conversion starts to execute. This bit should be set to “1” when ***[ADxST]***<CNTF> is “0” (a continuous conversion program does not execute).

4.2.2. *[ADxCR1]* (Control Register1)

Bit	Bit Symbol	After Reset	Type	Function
31:8	-	0	R	Read as “0”
7	HPDMEN	0	R/W	Highest priority conversion DMA request control 0: Disabled 1: Enabled
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
3:2	-	0	R	Read as “0”
1	HPTRGEN	0	R/W	Highest priority trigger conversion start-up control 0: Disabled 1: Enabled
0	TRGEN	0	R/W	General purpose trigger start-up control 0: Disabled 1: Enabled

Note: This register must be set while ***[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	AD operation flag 0: Stop (ADCLK can be stopped) 1: Executing (ADCLK cannot be stopped) Before ADCLK is stopped, this bit should be confirmed to be “0”.
6:4	-	0	R	Read as “0”
3	CNTF	0	R	Continuous conversion program flag 0: Stop 1: Executing When the request is received, this bit becomes “1”. When the last conversion result is stored, this bit becomes “0”.
2	SNGF	0	R	Single conversion program flag 0: Stop 1: Executing When the request is received, this bit becomes “1”. When the last conversion result is stored, this bit becomes “0”.
1	TRGF	0	R	General purpose trigger program flag 0: Stop 1: Executing When the request is received, this bit becomes “1”. When the last conversion result is stored, this bit becomes “0”.
0	HPF	0	R	Highest priority conversion program flag 0: Stop 1: Executing When the request is received, this bit becomes “1”. When the last conversion result is stored, this bit becomes “0”.

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	External AIN sampling interval (Note2)(Note3) 0000: (1 / SCLK) x n 0001: (1 / SCLK) x 2n 0010: (1 / SCLK) x 3n 0011: (1 / SCLK) x 4n (Note4) Others: Reserved.
7	-	0	R	Read as “0”
6:3	EXAZ0[3:0]	0000	R/W	External AIN sampling interval (Note2)(Note3) 0000: (1 / SCLK) x n 0001: (1 / SCLK) x 2n 0010: (1 / SCLK) x 3n 0011: (1 / SCLK) x 4n (Note4) Others: Reserved.
2:0	VADCLK[2:0]	000	R/W	AD prescaler output (SCLK) selection 000: ADCLK/1 001: ADCLK/2 010: ADCLK/4 011: ADCLK/8 1xx: ADCLK/16 This bit should be set so that SCLK is 80[MHz].

Note1: This register must be set while ***[ADxCR0]***<ADEN> =0.

Note2: Select external AIN sampling period with ***[ADxEXAZSEL]***.

Note3: The value of n depends on the condition of SCLK as follows.

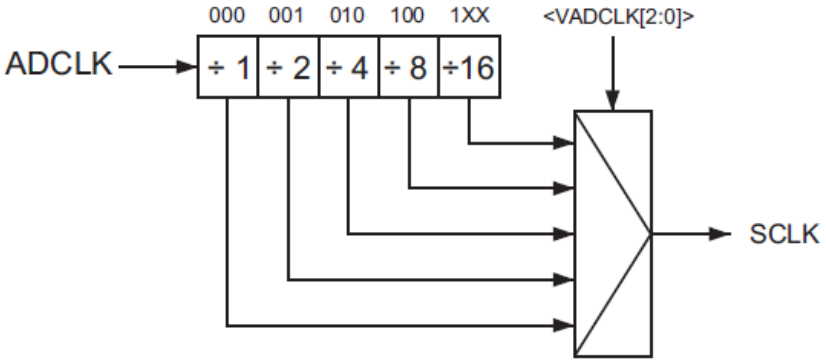
SCLK ≤ 40 MHz, n = 10.

40 MHz < SCLK ≤ 50 MHz, n = 14.

50 MHz < SCLK ≤ 60 MHz, n = 16.

60 MHz < SCLK ≤ 80 MHz, n = 20.

Note4: When Use SCK ≤ 40MHz, “0011” cannot be set.



4.2.5. *[ADxMOD0]* (Mode Setting Register0)

Bit	Bit Symbol	After Reset	Type	Function
31:2	-	0	R	Read as “0”
1	RCUT	1	R/W	Low power mode selection 0: Normal operation 1: Low power operation (Energized between VREFHx and VREFLx only during the conversion)
0	DACON	0	R/W	DAC control (Note2) 0: OFF 1: ON When the ADC is used, <DACON> should be set to “1”.

Note1: This register must be set while ***[ADxCR0]***<ADEN> =0.

Note2: After ***[ADxMOD0]***<DACON> is set to “1”, the interval of 3[μs] are necessary for the stabilization.

4.2.6. *[ADxMOD1]* (Mode Setting Register1)

Bit	Bit Symbol	After Reset	Type	Function
31:0	MOD1[31:0]	0x00004000	R/W	Set the applicable value according to SCLK conditions. 0x00001000: (SCLK ≤ 40MHz) 0x00003000: (40MHz < SCLK ≤ 50MHz) 0x00004000: (50MHz < SCLK ≤ 60MHz) 0x00106011: (60MHz < SCLK ≤ 80MHz) Others: Reserved.

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

4.2.7. *[ADxMOD2]* (Mode Setting Register2)

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 “Product Information” of the reference manual.

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

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

Bit	Bit Symbol	After Reset	Type	Function
31:4	-	0	R	Read as “0”
3	CMP3EN	0	R/W	AD monitor function3 0: Disabled. 1: Enabled.
2	CMP2EN	0	R/W	AD monitor function2 0: Disabled. 1: Enabled.
1	CMP1EN	0	R/W	AD monitor function1 0: Disabled. 1: Enabled.
0	CMP0EN	0	R/W	AD monitor function0 0: Disabled. 1: Enabled.

4.2.9. *[ADxCMPCR0]* (Monitor function Setting Register0)

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 1: Accumulated count
5	ADBIG0	0	R/W	Magnitude determination setting 0: Conversion result specified by <REGS0> > ***[ADxCMP0]*** (Larger than the comparison register) 1: Conversion result specified by <REGS0> < ***[ADxCMP0]*** (Smaller than the comparison register)
4:0	REGS0[4:0]	00000	R/W	Compared conversion result storage register 00000: ADxREG0 01000: ADxREG8 10000: ADxREG16 00001: ADxREG1 01001: ADxREG9 10001: ADxREG17 00010: ADxREG2 01010: ADxREG10 10010: ADxREG18 00011: ADxREG3 01011: ADxREG11 10011: ADxREG19 00100: ADxREG4 01100: ADxREG12 10100: ADxREG20 00101: ADxREG5 01101: ADxREG13 10101: ADxREG21 00110: ADxREG6 01110: ADxREG14 10110: ADxREG22 00111: ADxREG7 01111: ADxREG15 10111: ADxREG23 11000 to 11111: Inhibited setting

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

4.2.10. *[ADxCMPCR1]* (Monitor function Setting Register1)

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 1: Accumulated count
5	ADBIG1	0	R/W	Magnitude determination setting 0: Conversion result specified by <REGS1> > ***[ADxCMP1]*** (Larger than the comparison register) 1: Conversion result specified by <REGS1> < ***[ADxCMP1***] (Smaller than the comparison register)
4:0	REGS1[4:0]	00000	R/W	Compared conversion result storage register 00000: ADxREG0 01000: ADxREG8 10000: ADxREG16 00001: ADxREG1 01001: ADxREG9 10001: ADxREG17 00010: ADxREG2 01010: ADxREG10 10010: ADxREG18 00011: ADxREG3 01011: ADxREG11 10011: ADxREG19 00100: ADxREG4 01100: ADxREG12 10100: ADxREG20 00101: ADxREG5 01101: ADxREG13 10101: ADxREG21 00110: ADxREG6 01110: ADxREG14 10110: ADxREG22 00111: ADxREG7 01111: ADxREG15 10111: ADxREG23 11000 to 11111: Inhibited setting

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

4.2.11. *[ADxCMPCR2]* (Monitor function Setting Register2)

Bit	Bit Symbol	After Reset	Type	Function
31:12	-	0	R	Read as “0”
11:8	CMPCNT2[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	CMPCND2	0	R/W	Determination condition 0: Continuous count 1: Accumulated count
5	ADBIG2	0	R/W	Magnitude determination setting 0: Conversion result specified by <REGS2> > ***[ADxCMP2]*** (Larger than the comparison register) 1: Conversion result specified by <REGS2> < ***[ADxCMP2]*** (Smaller than the comparison register)
4:0	REGS2[4:0]	00000	R/W	Compared conversion result storage register 00000: ADxREG0 01000: ADxREG8 10000: ADxREG16 00001: ADxREG1 01001: ADxREG9 10001: ADxREG17 00010: ADxREG2 01010: ADxREG10 10010: ADxREG18 00011: ADxREG3 01011: ADxREG11 10011: ADxREG19 00100: ADxREG4 01100: ADxREG12 10100: ADxREG20 00101: ADxREG5 01101: ADxREG13 10101: ADxREG21 00110: ADxREG6 01110: ADxREG14 10110: ADxREG22 00111: ADxREG7 01111: ADxREG15 10111: ADxREG23 11000 to 11111: Inhibited setting

Note: This register must be set while ***[ADxCMPEN]***<CMP2EN> =0.

4.2.12. *[ADxCMPCR3]* (Monitor function Setting Register3)

Bit	Bit Symbol	After Reset	Type	Function
31:12	-	0	R	Read as “0”
11:8	CMPCNT3[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	CMPCND3	0	R/W	Determination condition 0: Continuous count 1: Accumulated count
5	ADBIG3	0	R/W	Magnitude determination setting 0: Conversion result specified by <REGS3> > ***[ADxCMP3]*** (Larger than the comparison register) 1: Conversion result specified by <REGS3> < ***[ADxCMP3]*** (Smaller than the comparison register)
4:0	REGS3[4:0]	00000	R/W	Compared conversion result storage register 00000: ADxREG0 01000: ADxREG8 10000: ADxREG16 00001: ADxREG1 01001: ADxREG9 10001: ADxREG17 00010: ADxREG2 01010: ADxREG10 10010: ADxREG18 00011: ADxREG3 01011: ADxREG11 10011: ADxREG19 00100: ADxREG4 01100: ADxREG12 10100: ADxREG20 00101: ADxREG5 01101: ADxREG13 10101: ADxREG21 00110: ADxREG6 01110: ADxREG14 10110: ADxREG22 00111: ADxREG7 01111: ADxREG15 10111: ADxREG23 11000 to 11111: Inhibited setting

Note: This register must be set while ***[ADxCMPEN]***<CMP3EN> =0.

4.2.13. *[ADxCMP0]* (Conversion Result Comparison Register0)

Bit	Bit Symbol	After Reset	Type	Function
31:16	-	0	R	Read as “0”
15:4	ADCMP0[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 while ***[ADxCMPEN]***<CMP0EN> =0.

4.2.14. *[ADxCMP1]* (Conversion Result Comparison Register1)

Bit	Bit Symbol	After Reset	Type	Function
31:16	-	0	R	Read as “0”
15:4	ADCMP1[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 while ***[ADxCMPEN]***<CMP1EN> =0.

4.2.15. *[ADxCMP2]* (Conversion Result Comparison Register2)

Bit	Bit Symbol	After Reset	Type	Function
31:16	-	0	R	Read as “0”
15:4	ADCMP2[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 while ***[ADxCMPEN]***<CMP2EN> =0.

4.2.16. *[ADxCMP3]* (Conversion Result Comparison Register3)

Bit	Bit Symbol	After Reset	Type	Function
31:16	-	0	R	Read as “0”
15:4	ADCMP3[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 while ***[ADxCMPEN]***<CMP3EN> =0.

4.2.17. *[ADxEXAZSEL]* (External AIN sampling period selection Register)

Bit	Bit Symbol	After Reset	Type	Function
31:0	EXAZSEL[31:0]	0x00000000	R/W	Selection of external AIN sampling period (EXAZ) 0: ***[ADxCLK]***<EXAZ0[3:0]> setting value 1: ***[ADxCLK]***<EXAZ1[3:0]> setting value Please select which one of EXAZ0 or EXAZ1 is used for each AIN channels. Bit represents the AIN channel. (Refer to the following.) Bit[23]: Setting value selection at AD conversion of AINx23. Bit[22]: Setting value selection at AD conversion of AINx22. . . Bit[0]: Setting value selection at AD conversion of AINx00.

4.2.18. *[ADxTSET0]* (Start-up Factor Program Register0)

The following is an example of ***[ADxTSET0]***. ***[ADxTSET1]*** to ***[ADxTSET23]*** have the same configuration.

Bit	Bit Symbol	After Reset	Type	Function
31:11	-	0	R	Read as “0”
10:8	TRGS0[2:0]	000	R/W	Conversion Result Storage Register0 setting: Conversion control 000: Disable 001: Continuous conversion 010: Single conversion 011: General purpose trigger conversion. 1xx: Highest priority conversion
7	ENINT0	0	R/W	Conversion Result Storage Register0 setting: Interrupt control 0: Disabled 1: Enabled.
6:5	-	0	R	Read as “0”
4:0	AINST0[4:0]	00000	R/W	Conversion Result Storage Register0 setting: AIN selection (Note2) 00000: AINx00 01000: AINx08 10000: AINx16 00001: AINx01 01001: AINx09 10001: AINx17 00010: AINx02 01010: AINx10 10010: AINx18 00011: AINx03 01011: AINx11 10011: AINx19 00100: AINx04 01100: AINx12 10100: AINx20 00101: AINx05 01101: AINx13 10101: AINx21 00110: AINx06 01110: AINx14 10110: AINx22 00111: AINx07 01111: AINx15 10111: AINx23 11000 to 11111: Inhibited setting

Note1: This register must be set while ***[ADxCR0]***<ADEN> =0.
Note2: The AIN which the product does not have is inhibited to be set (Refer to “Product Information” of the reference manual).

4.2.19. *[ADxREG0]* (Conversion Result Storage Register0)

The following is an example of ***[ADxREG0]***. ***[ADxREG1]*** to ***[ADxREG23]*** have the same configuration.

Bit	Bit Symbol	After Reset	Type	Function
31:30	-	0	R	Read as “0”
29	ADOVRF_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 bit of AD conversion result <ADR0>. The AD conversion result is read from the lower 12 bits in the upper half word of ***[ADxREG0]*** register.
15:4	ADR0[11:0]	0x000	R	AD conversion result is stored. The AD conversion result is read from the upper 12 bits in the lower half word of ***[ADxREG0]*** register.
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 the ***[ADxREG0]*** register is read. This flag is cleared to “0” when it is read.
0	ADRF0	0	R	AD conversion result storage flag 0: No conversion results are stored 1: A conversion result is stored. This flag is set to “1” when an AD conversion value is stored. This flag is cleared to “0” when it is read.

5. Usage example

5.1. Single conversion

The single conversion is started by software and enable more than one conversion.

In the following setting example, the conversion results of the two analog inputs (AINx02, AINx03) are saved in two result storage registers (***[ADxREG4]***, ***[ADxREG5]***), and a single conversion interrupt INTADxSGL is generated at the end of two conversion.

- Initial setting
  - ***[ADxCLK]*** =0x00000000
    AD prescaler output: <VADCLK[2:0]> =000
    (AD prescaler output: 60MHz (ADCLK / 1 @ADCLK:60MHz))
    External AIN sampling period: <EXAZ0[3:0]> =0000 and <EXAZ1[3:0]> =0001
    (Sampling period: EXAZ0: 266.6ns, EXAZ1: 533.3ns)
  - ***[ADxEXAZSEL]*** =0x00000008
    External AIN sampling period selection (AINx02: EXAZ0, AINx03: EXAZ1)
  - ***[ADxMOD0]*** =0x00000001
    DAC ON: <DACON> =1
    Normal operation: <RCUT> =0
    [ADxMOD1] =0x00004000
    (Conversion time AINx02: 1.0 [μs], AINx03: 1.26[μs])
  - ***[ADxMOD2]*** =0x00000000

  Note: The settings depend on a product. For the setting values, refer to “Product Information” in the Reference manual.

- Conversion program setting
  - ***[ADxTSET4]*** =0x00000202
    Single conversion: <TRGS4> =010
    AINx02: <AINST4> =00010
    Disable interrupt output: <ENINT4> =0
  - ***[ADxTSET5]*** =0x00000283
    Single conversion: <TRGS5> =010
    AINx03: <AINST5> =00011
    Enable interrupt output: <ENINT5> =1

- Conversion start setting
  - ***[ADxCR1]*** =0x00000000
    Disable Single conversion DMA request: <SGLDMEN> =0
  - ***[ADxCR0]*** =0x00000082
    Enable ADC: <ADEN> =1
    Disable continuous conversion: <CNT> =0
    Enable single conversion: <SGL> =1 ; conversion start

5.2. Highest priority conversion

The conversion with the highest priority conversion can be started with the highest priority trigger.

In the following, save the conversion result of analog input (AINx04) in the conversion result storage register (***[ADxREG6]***) with the highest priority trigger (ADxHPTRGIN). This is an example of setting to generate the highest priority conversion interrupt INTADxHP at the end of conversion.

- Initial setting
  - ***[ADxCLK]*** =0x00000000
    AD prescaler output: <VADCLK[2:0]> =000
    (AD prescaler output: 60MHz (ADCLK / 1 @ADCLK:60MHz)
    External AIN sampling period: <EXAZ1[3:0]> =0000
    (sampling period: EXAZ1: 266.6ns)
  - ***[ADxEXAZSEL]*** =0x00000010
    External AIN sampling period selection(AINx04: EXAZ1)
  - ***[ADxMOD0]*** =0x00000001
    DAC ON: <DACON> =1
    Normal operation: <RCUT> =0
  - ***[ADxMOD1]*** =0x00004000
    (Conversion time AINx04: 1.0 [μs])
    [ADxMOD2] =0x00000000

  Note: The settings depend on a product. For the setting values, refer to “Product Information” in the Reference manual.

- Conversion program setting
  - ***[ADxTSET6]*** =0x00000484
    Highest priority conversion: <TRGS6> =1xx
    AINx04: <AINST6> =00100
    Enable interrupt output: <ENINT6> =1

- Conversion start setting
  - ***[ADxCR1]*** =0x00000002
    Disable highest priority conversion DMA request: <HPDMEN> =0
    Enable highest priority trigger conversion control: <HPTRGEN> =1
  - ***[ADxCR0]*** =0x00000080
    Enable ADC: <ADEN> =1
    Disable highest priority trigger conversion operation: <HPSGL> =0

6. Precaution

- The AD conversion result may have some variation due to the fluctuation of the power supply and surrounding noises. The data of the output pins should not be changed during AD conversion to prevent from degrading the AD conversion accuracy. The AD conversion accuracy may degrade if the signal on the shared pin with the AD input/output changes or other output pin changes its output during the AD conversion. In the above case, the AD conversion result should be acquired with the mean value of multiple conversion results and other countermeasures.

- Measures should be taken to prevent digital noise from mixing into the analog power supply pins (AVDD3, AVSS) and the reference voltage pins (VREFHx, VREFLx) of the ADC.
  - Insert a bypass capacitor between AVDD3 and AVSS pins, the VREFHx and VREFLx pins. Place the capacitor as close to the terminal as possible.

7. Revision History

Table 7.1 Revision history

Revision	Date	Description
1.0	2018-01-10	First release
2.0	2018-04-04	1. Outlines Corrected: “Monitor conversion result” line “Selectable number of ----------” ->” - Selectable number of detections.” “ - Selectable continuous count and accumulated count.” 3.Function and Operation Added: section “3.1.Clock Supply” 3.1.Clock Supply Deleted: “Before AD conversion is started,--------- should be set to “1”.” 3.2.1.Operation Corrected: “general purpose start-up”->”start-up” Corrected: Explanation before Table 3.2 “When the interrupt is ----------” 3.2.2. Control Register Added: ・Mode register0 (***[ADxMOD0]***) Other 3.7. Conversion Time Added: section “3.7. Conversion Time” 4.2.4. ***[ADxCLK]*** (Conversion Clock Setting Register) Deleted: subsection “AD conversion time (tcov)” 6.Precaution Corrected: “AVSS3”->”AVSS”
3.0	2019-05-28	1. Outline Modified description of Conversion time 3.7.2 Sampling time Deleted value of ***[ADxCLK]***<EXAZn>=0101 and 0111 in Table 3.6 Corrected “1.00” to “-” in Table 3.6 3.7.4 Setting of Conversion time Deleted value of ***[ADxCLK]***<EXAZn>=0101 and 0111 in Table 3.7 Corrected “5.04” to “-” in Table 3.7 4.2.4 ***[ADxCLK]*** Corrected parameter of the 0101 and 0111 to “reserved” Added Note4

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.**