Application Processor Lite *ApP Lite*

TZ1000 Series

Reference Manual

MCU IO

Revision 1.2

2018-02

TOSHIBA ELECTRONIC DEVICES & STORAGE CORPORATION

Table of Contents

List of Figures

List of Tables

Preface

This document provides the specification for the MCU IO designed for the TZ1000 Series.

Intended Audience

This document is intended for the following users.

Driver software developers.

System designers

Conventions in this document

- The numerical values are expressed as follows.
  Hexadecimal number: 0xABC
  Decimal number: 123 or 0d123 - Only when it needs to be explicitly shown that they are decimal numbers.
  Binary number: 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 [3:0].
  [3:0] shows four signal names 3, 2, 1, and 0 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]***, and ***[XYZ3]*** to ***[XYZn]***
- The bit range of a register is written like as [3:0].
  Example: Bit[3:0] expresses the range of bits 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-bit
  Half word: 16-bit
  Word: 32-bit
  Double word: 64-bit
- Unless otherwise specified, register access supports only word access.
- The register defined as reserved must not be rewritten. Moreover, don't use the read value.
- Properties of each bit in a register are expressed as follows.
  R: Read only
  W: Write only
  W1C: Write 1 Clear - The corresponding bit is cleared (=0) when "1" is written to this bit.
  W1S: Write 1 Set - The corresponding bit is set (=1) when "1" is written to this bit.
  R/W: Read and Write are possible.
  R/W0C: Read/Write 0 Clear
  R/W1C: Read/Write 1 Clear
  R/W1S: Read/Write 1 Set
  RS/WC: Read Set/Write Clear - Set after read operation, cleared after write operation.
- 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.

1. Overview

This document describes about the specification of the MCU IO in the TZ1000 Series, including the internal IO circuits which have no external pins in the TZ1000 Series (the IOs to connect the SPI Flash memory and the MEMS sensor in the package).

The outlines of the IO ports are as follows.

- 5 kinds of digital IO cells
  - Bi-directional IO: Multi-drive, a programmable pull-up or pull-down resistor, and Schmitt trigger input
  - Bi-directional IO: Multi-drive, a pull-up resistor, and Schmitt trigger input
  - 5 V tolerant input port
  - Schmitt trigger input port
  - Output port
- Control for IO reset, latch, and standby in a power save mode

2. Block Diagram

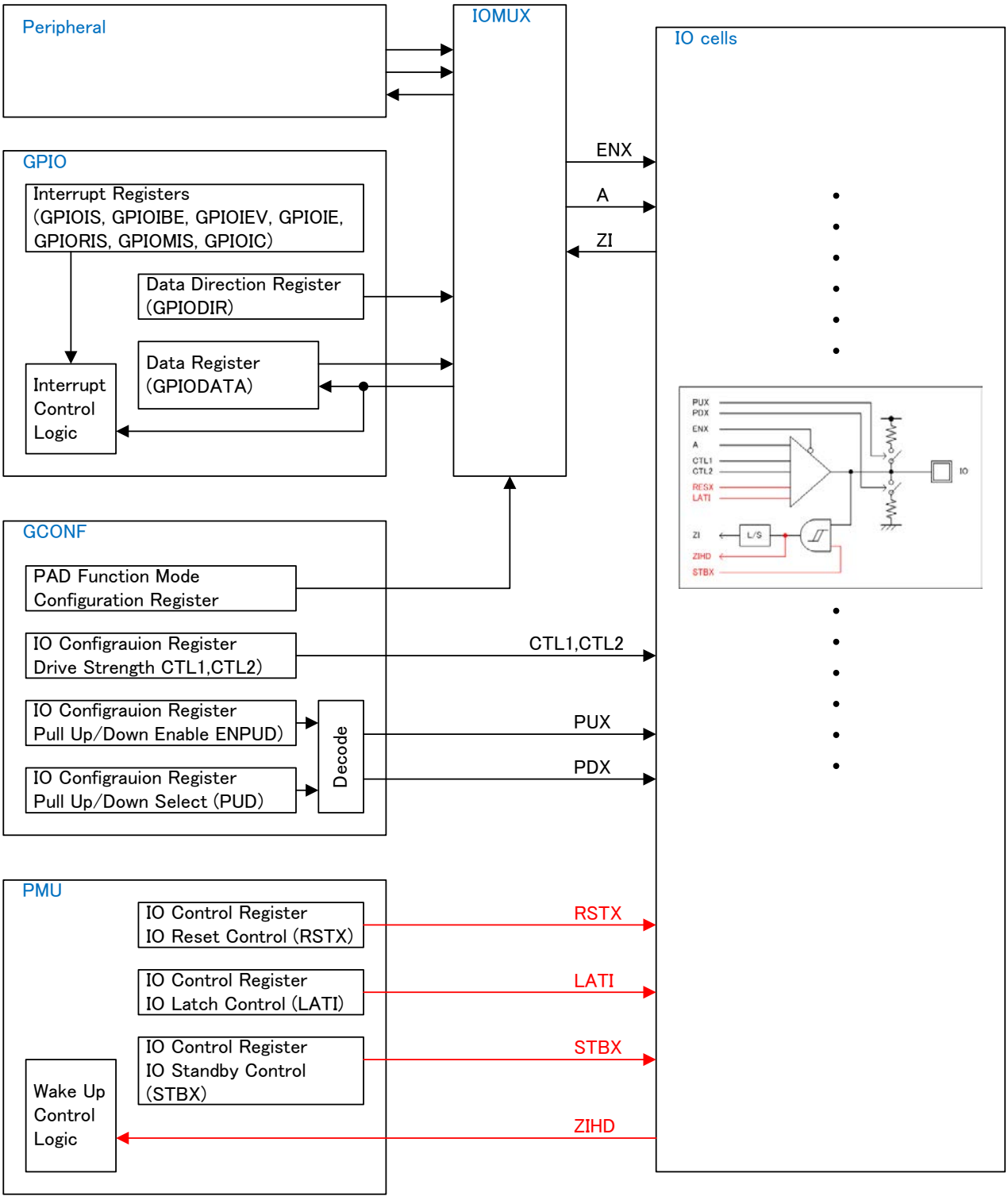


Figure 2.1 Blocks related to IO cells

3. Function

3.1. Digital IO Cell

This product supports 5 kinds of digital IO cells as follows.

- Bi-directional IO: Multi-drive, a programmable pull-up or pull-down resistor, and Schmitt trigger input
- Bi-directional IO: Multi-drive, a pull-up resistor, and Schmitt trigger input
- 5 V tolerant input port
- Schmitt trigger input port
- Output port

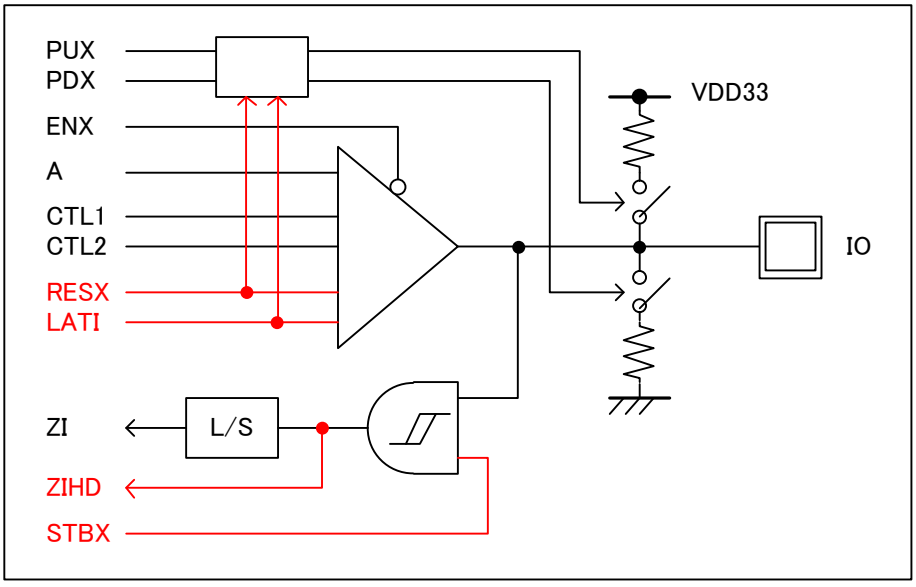
The detail of the IO cells is described as follows.

3.1.1. Bi-directional IO with Multi-drive, Programmable Pull-up or Pull-down Resistor, and Schmitt Trigger Input

- Cell name
  - BDSPZ
  - BDSPU
  - BDSPD

- Features
  These IO cells are bi-directional cells which support the selection of the current drivability (2/4/5/7 mA), and programmable pull-up or pull-down resistor. These IO cells are used for GPIO0 to 3 except GPIO0 PIN0, UART0 to 2, I2C0 and 1, SPIM0 to 3, SPIC, and other pins.
  - BDSPZ: In initial state, neither the pull-up nor pull-down resistor.
  - BDSPU: In initial state, with the pull-up resistor.
  - BDSPD: In initial state, with the pull-down resistor.

- IO Structure



- IO Signal Function

<table>
<tr><th>Node Name</th><th>Direction</th><th>Power Supply</th><th>Description</th></tr>
<tr><td>IO</td><td>In/Out</td><td>VDD33</td><td>External Pin</td></tr>
<tr><td>ENX</td><td>In</td><td>VDD12</td><td>Output Enable
When ENX = 0, Data A is driven to the IO pin through the output
When ENX = 1, the output is disabled.</td></tr>
<tr><td>A</td><td>In</td><td>VDD12</td><td>Output Data
When ENX = 0, Data A is driven to the IO pin through the output</td></tr>
<tr><td>PUX</td><td>In</td><td>VDD12</td><td>Pull-up Enable
When PUX = 0, the pull-up resistor is connected.</td></tr>
<tr><td>PDX</td><td>In</td><td>VDD12</td><td>Pull-down Enable
When PDX = 0, the pull-up resistor is connected.</td></tr>
<tr><td>CTL1
CTL2</td><td>In</td><td>VDD12</td><td>Drivability Setting
Selection for the output drivability
<table>
<tr><th>CTL2</th><th>CTL1</th><th>Drivability</th></tr>
<tr><td>0</td><td>0</td><td>2 mA (*1)</td></tr>
<tr><td>0</td><td>1</td><td>4 mA (*1)</td></tr>
<tr><td>1</td><td>0</td><td>6 mA (*1)</td></tr>
<tr><td>1</td><td>1</td><td>8 mA (*1)</td></tr>
</table></td></tr>
<tr><td>ZI</td><td>Out</td><td>VDD12</td><td>Input Data
When STBX = 1, the data on the IO pin is issued to ZI.</td></tr>
<tr><td>RESX</td><td>In</td><td>VDD33</td><td>IO Cell Reset
When RESX = 0, the IO cell output buffer is initialized. The initial state is as follows.
BDSPZ: No drive, and neither the pull-up nor pull-down resistor
BDSPU: No drive, and the pull-up resistor connected
BDSPD: No drive, and the pull-down resistor connected.</td></tr>
<tr><td>LATI</td><td>In</td><td>VDD33</td><td>IO Cell Status Protection
When LATI = 1, the output buffer status of the IO cell (the output buffer enable or disable, and the selection of the pull-up or pull-down resistor) is protected as the status just before LATI becomes 1. While LATI = 1, ENX, A, PUX, and PDX cannot change the IO cell status.
RESX is prioritized. When RESX = 0, the status is not protected.</td></tr>
<tr><td>STBX</td><td>In</td><td>VDD33</td><td>IO Cell Input Standby Control
Control for the input buffer. When STBX = 1, the data on the IO pin is issued to ZIHD and ZI.</td></tr>
<tr><td>ZIHD</td><td>Out</td><td>VDD33</td><td>Input Data (HV Signal)
When STBX = 1, the data on the IO pin is issued to ZIHD.</td></tr>
</table>

(*1) The each current value is different depending on VDD33 voltage. Refer to DC Characteristics of each product Technical Data sheet.

ENX and A receive the output signals from the GPIO and the peripheral blocks through the IOMUX multiplexer. The signal issued from ZI is transferred to the GPIO or the peripheral blocks through the IOMUX demultiplexer. For the IOMUX function, refer to "Data Sheet Summary 5.1 Setting Multiple function I/O."

PUX and PDX are connected to the pull-up or pull-down control signals of the IO configuration register in the GCONF. The IO configuration register sets the availability of either of pull-up or pull-down resistor, and also specifies the pull-up or pull-down one. Both PUX and PDX never become valid at the same time.

Power Supply of the pull-up resistor is VDD33.

GCONF Register		IO Cell Control		Pull-up or Pull-down
[IO_CFGn].*_ENPUD	*[IO_CFGn]*.*_PUD	PUX	PDX	
0	1	0	1	Pull-up available
0	0	1	0	Pull-down available
1	*	1	1	Neither Pull-up nor Pull-down available

CTL1 and CTL2 are connected the output buffer drivability control signals of the IO configuration register in the GCONF.

GCONF Register		IO Cell Control		Drivability
[IO_CFGn].*_CTL2	*[IO_CFGn]*.*_CTL1	CTL2	CTL1	
0	0	0	0	2 mA (*1)
0	1	0	1	4 mA (*1)
1	0	1	0	6 mA (*1)
1	1	1	1	8 mA (*1)

(*1) The each current value is different depending on VDD33 voltage. Refer to DC Characteristics of each product Technical Data sheet.

RESX and LATI are connected to the IO reset control bit and the IO setting protection control bit of the IO control register in the PMU, respectively. Two IO registers are implemented. Each register controls the PM domain and the PP1 domain, respectively. All IO cells in the domain are collectively controlled by the register.

STBX is connected to the input standby control bit of the IO control register in the PMU. STBX can be set per IO cell or per module, which is different from RESX and LATI.

The signals issued from ZIHD of some IO cells are transferred to the Wake Up control block in the PMU. They are used as the Wake Up signals which return a block from the Low Power mode such as the WAIT mode, the RETENTION mode or others.

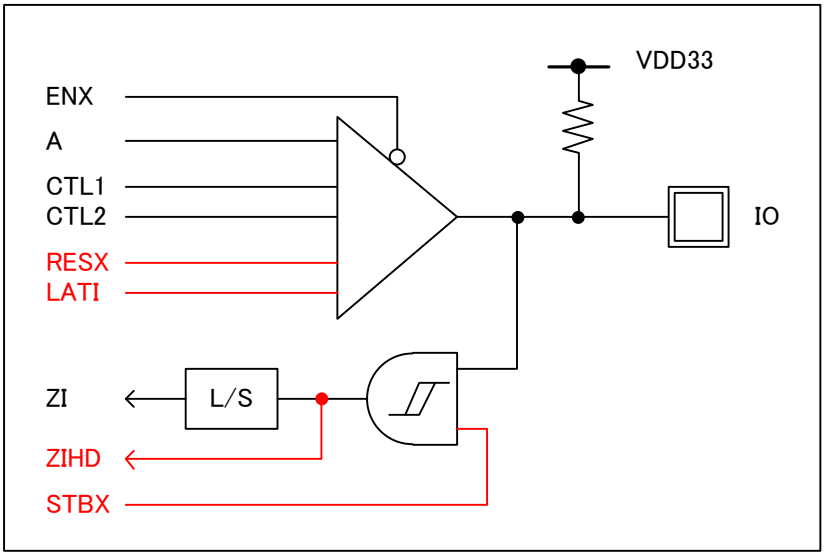
For the IO cells in the SPIC, RESX, LATI, and STBX are fixed to the invalid value. These signals cannot be controlled.

3.1.2. Bi-directional IO with Multi-drive, Pull-up Resistor, and Schmitt Trigger Input

- Cell name: BDSUI2C

- Features
  Bidirectional IO whose pull-up resistor is adjusted to the I2C interface.
  This IO cell is used for the MCU_I2C2_DATA and MCU_I2C2_CLK pins.

- IO Structure



- IO Signal Function

Node Name	Direction	Power Supply	Description
IO	In/Out	VDD33	External Pin
ENX	In	VDD12	Output Enable When ENX = 0, Data A is driven to the IO pin through the output When ENX = 1, the output is disabled.
A	In	VDD12	Output Data When ENX = 0, Data A is driven to the IO pin through the output
PUX	In	VDD12	Pull-up Enable When PUX = 0, the pull-up resistor is connected.
PDX	In	VDD12	Pull-down Enable When PDX = 0, the pull-up resistor is connected.
CTL1 CTL2	In	VDD12	Drivability Setting Selection for the output drivability CTL2 / CTL1 / Drivability 0 / 0 / 2 mA (*1) 0 / 1 / 4 mA (*1) 1 / 0 / 6 mA (*1) 1 / 1 / 8 mA (*1)
ZI	Out	VDD12	Input Data When STBX = 1, the data on the IO pin is issued to ZI.
RESX	In	VDD33	IO Cell Reset When RESX = 0, the IO cell output buffer is initialized. The initial state is No drive. (The pull-up resistor is always connected.)

LATI	In	VDD33	IO Cell Status Protection When LATI = 1, the output buffer status of the IO cell (the output buffer enable or disable, and the selection of the pull-up or pull-down resistor) is protected as the status just before LATI becomes 1. While LATI = 1, ENX, A, PUX, and PDX cannot change the IO cell status. RESX is prioritized. When RESX = 0, the status is not protected.
STBX	In	VDD33	IO Cell Input Standby Control Control for the input buffer. When STBX = 1, the data on the IO pin is issued to ZIHD and ZI.
ZIHD	Out	VDD33	Input Data (HV Signal) When STBX = 1, the data on the IO pin is issued to ZIHD.

(*1) The each current value is different depending on VDD33 voltage. Refer to DC Characteristics of each product Technical Data sheet.

ENX and A receive the output signals from the I2C2 through the IOMUX multiplexer. The signal issued from ZI is transferred to the I2C2 through the IOMUX demultiplexer. For the IOMUX function, refer to "Data Sheet Summary 5.1 Setting Multiple function I/O."

Pull-up resistor is not a programmable, is always connected. Power Supply of the pull-up resistor is VDD33.

CTL1 and CTL2 are connected the output buffer drivability control signals of the IO configuration register in the GCONF.

GCONF Register		IO Cell Control		Drivability
[IO_CFG4].I2C2_*_CTL2	*[IO_CFG4]*.I2C2_*_CTL1	CTL2	CTL1	
0	0	0	0	2 mA (*1)
0	1	0	1	4 mA (*1)
1	0	1	0	6 mA (*1)
1	1	1	1	8 mA (*1)

(*1) The each current value is different depending on VDD33 voltage. Refer to DC Characteristics of
each product Technical Data sheet.

RESX and LATI are connected to the IO reset control bit and the IO setting protection control bit of the IO control register in the PMU, respectively. Two IO registers are implemented. Each register controls the PM domain and the PP1 domain, respectively. All IO cells in the domain are collectively controlled by the register.

STBX is connected to the input standby control bit of the IO control register in the PMU. STBX can be set per IO cell or per module, which is different from RESX and LATI.

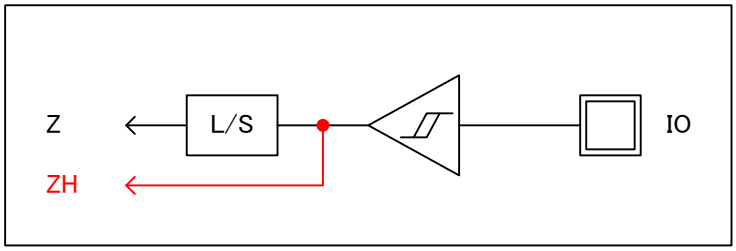
ZIHD is not used.

3.1.3. 5 V Tolerant Input Only IO

- Cell name: IS5

- Features
  5 V tolerant input only IO cell.
  This IO cell is used for the MCU_GPIO_0 pin. The pin is supposed to connect the VBUS signal in the USB component. The general purpose IO, GPIO0 PIN0, is only input because the IO is shared with the MCU_GPIO_0 pin.

- IO Structure



- IO Signal Function

Node Name	Direction	Power Supply	Description
IO	In/Out	VDD33	External Pin
Z	Out	VDD12	Input Data The data on the IO pin is issued to Z.
ZH	Out	VDD33	Input Data (HV Signal) The data on the IO pin is issued to ZH.

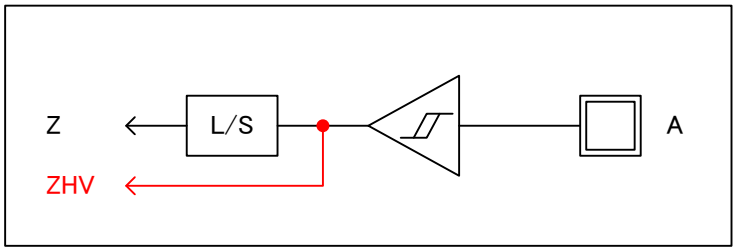
The signal issued from Z is transferred to the GPIO0 through the IOMUX demultiplexer. The signal issued from ZH is connected to the Wake Up control block in the PMU. It is used as the Wake Up signal which returns a block from the Low Power mode such as the WAIT mode, the RETENTION mode or others.

3.1.4. Schmitt Trigger Input Only IO

- Cell name: IS

- Features
  Input only IO cell.
  This IO cell is used for the MCU_TEST, MCU_SYS_RESET_N, and MCU_BOOTMODE0-4 pins.

- IO Structure



- IO Signal Function

Node Name	Direction	Power Supply	Description
A	In/Out	VDD33	External Pin
Z	Out	VDD12	Input Data The data on the IO pin is issued to Z.
ZHV	Out	VDD33	Input Data (HV Signal) The data on the IO pin is issued to ZHV.

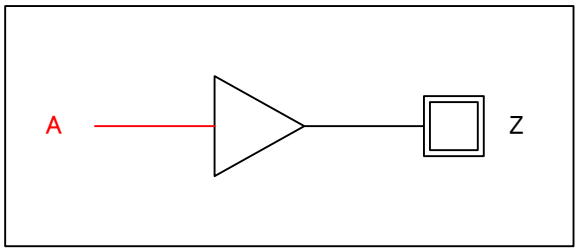
Z is not used. Only ZHV is used. The signal issued from ZHV is connected to the PMUHV which is the MCU_VDD33 domain in the MPU.

3.1.5. Output Only IO

- Cell name: O2

- Features
  Output only IO cell.
  This cell is used for the MCU_CLK32K_OUT pin.

- IO Structure



- IO Signal Function

Node Name	Direction	Power Supply	Description
Z	In/Out	VDD33	External Pin
A	In	VDD12	Output Data A is issued to the Z pin through the output buffer.

A is connected to 32 kHz clock in the PMU.

3.2. IO Configuration

PUX and PDX in the IO cell are connected to the pull-up or pull-down control signals of the IO configuration register in the GCONF. The IO configuration register sets the availability of either of pull-up or pull-down resistor (*_ENPUD), and also specifies the pull-up or pull-down one (*_PUD). These register outputs are decoded in the GCONF so that both PUX and PDX never become valid at the same time.

CTL1 and CTL2 are connected the output buffer drivability control signals of the IO configuration register in the GCONF.

RESX and LATI are connected to the IO reset control bit and the IO setting protection control bit of the IO control register in the PMU, respectively. Two IO registers are implemented. Each register controls the PM domain and the PP1 domain, respectively. All IO cells in the domain are collectively controlled by the register.

STBX is connected to the input standby control bit of the IO control register in the PMU. STBX can be set per IO cell or per module, which is different from RESX and LATI.

The signals issued from ZIHD and ZH of some IO cells are transferred to the Wake Up control block in the PMU. They are used as the Wake Up signals which return a block from the Low Power mode such as the WAIT mode, the RETENTION mode or others.

Table 3.1 IO configuration registers and IO cell control signals (Pull-up and Pull-down)

GCONF Register		IO Cell Control		Pull-up or Pull-down
*[IO_CFGn].*_ENPUD*	*[IO_CFGn].*_PUD*	PUX	PDX	
0	1	0	1	Pull-up available
0	0	1	0	Pull-down available
1	*	1	1	Neither Pull-up nor Pull-down available

Table 3.2 IO configuration registers and MCU chip pins (Pull-up and Pull-down)

Pin Name	Register Name	IO Cell Control	
		PUX	PDX
MCU_DBG_TCK	-	1	0
MCU_DBG_TDO		1	1
MCU_DBG_TDI		0	1
MCU_DBG_TMS		0	1
MCU_DBG_TRST_N		0	1
MCU_GPIO_0	*[IO_CFG0]*	—	—
MCU_GPIO_1		GPIO_1_ENPUD || !GPIO_1_PUD	GPIO_1_ENPUD || GPIO_1_PUD
MCU_GPIO_2		GPIO_2_ENPUD || !GPIO_2_PUD	GPIO_2_ENPUD || GPIO_2_PUD
MCU_GPIO_3		GPIO_3_ENPUD || !GPIO_3_PUD	GPIO_3_ENPUD || GPIO_3_PUD
MCU_GPIO_4		GPIO_4_ENPUD || !GPIO_4_PUD	GPIO_4_ENPUD || GPIO_4_PUD
MCU_GPIO_5		GPIO_5_ENPUD || !GPIO_5_PUD	GPIO_5_ENPUD || GPIO_5_PUD

Pin Name	Register Name	IO Cell Control	
		PUX	PDX
MCU_GPIO_6		GPIO_6_ENPUD || !GPIO_6_PUD	GPIO_6_ENPUD || GPIO_6_PUD
MCU_GPIO_7		GPIO_7_ENPUD || !GPIO_7_PUD	GPIO_7_ENPUD || GPIO_7_PUD
MCU_GPIO_8	*[IO_CFG1]*	GPIO_8_ENPUD || !GPIO_8_PUD	GPIO_8_ENPUD || GPIO_8_PUD
MCU_GPIO_9		GPIO_9_ENPUD || !GPIO_9_PUD	GPIO_9_ENPUD || GPIO_9_PUD
MCU_GPIO_10		GPIO_10_ENPUD || !GPIO_10_PUD	GPIO_10_ENPUD || GPIO_10_PUD
MCU_GPIO_11		GPIO_11_ENPUD || !GPIO_11_PUD	GPIO_11_ENPUD || GPIO_11_PUD
MCU_GPIO_12		GPIO_12_ENPUD || !GPIO_12_PUD	GPIO_12_ENPUD || GPIO_12_PUD
MCU_GPIO_13		GPIO_13_ENPUD || !GPIO_13_PUD	GPIO_13_ENPUD || GPIO_13_PUD
MCU_GPIO_14		GPIO_14_ENPUD || !GPIO_14_PUD	GPIO_14_ENPUD || GPIO_14_PUD
MCU_GPIO_15		GPIO_15_ENPUD || !GPIO_15_PUD	GPIO_15_ENPUD || GPIO_15_PUD
MCU_GPIO_24	*[IO_CFG3]*	GPIO_24_ENPUD || !GPIO_24_PUD	GPIO_24_ENPUD || GPIO_24_PUD
MCU_GPIO_25		GPIO_25_ENPUD || !GPIO_25_PUD	GPIO_25_ENPUD || GPIO_25_PUD
MCU_GPIO_26		GPIO_26_ENPUD || !GPIO_26_PUD	GPIO_26_ENPUD || GPIO_26_PUD
MCU_GPIO_27		GPIO_27_ENPUD || !GPIO_27_PUD	GPIO_27_ENPUD || GPIO_27_PUD
MCU_GPIO_28		GPIO_28_ENPUD || !GPIO_28_PUD	GPIO_28_ENPUD || GPIO_28_PUD
MCU_GPIO_29		GPIO_29_ENPUD || !GPIO_29_PUD	GPIO_29_ENPUD || GPIO_29_PUD
MCU_GPIO_30		GPIO_30_ENPUD || !GPIO_30_PUD	GPIO_30_ENPUD || GPIO_30_PUD
MCU_GPIO_31		GPIO_31_ENPUD || !GPIO_31_PUD	GPIO_31_ENPUD || GPIO_31_PUD
MCU_I2C0_DATA	*[IO_CFG4]*	I2C0_DATA_ENPUD || !I2C0_DATA_PUD	I2C0_DATA_ENPUD || I2C0_DATA_PUD
MCU_I2C0_CLK		I2C0_CLK_ENPUD || !I2C0_CLK_PUD	I2C0_CLK_ENPUD || I2C0_CLK_PUD
MCU_I2C1_DATA		I2C1_DATA_ENPUD || !I2C1_DATA_PUD	I2C1_DATA_ENPUD || I2C1_DATA_PUD
MCU_I2C1_CLK		I2C1_CLK_ENPUD || !I2C1_CLK_PUD	I2C1_CLK_ENPUD || I2C1_CLK_PUD
MCU_I2C2_DATA		—	—
MCU_I2C2_CLK		—	—
MCU_UA0_RXD	*[IO_CFG5]*	UA0_RXD_ENPUD || !UA0_RXD_PUD	UA0_RXD_ENPUD || UA0_RXD_PUD
MCU_UA0_TXD		UA0_TXD_ENPUD || !UA0_TXD_PUD	UA0_TXD_ENPUD || UA0_TXD_PUD
MCU_UA1_RXD		UA1_RXD_ENPUD || !UA1_RXD_PUD	UA1_RXD_ENPUD || UA1_RXD_PUD
MCU_UA1_TXD		UA1_TXD_ENPUD || !UA1_TXD_PUD	UA1_TXD_ENPUD || UA1_TXD_PUD
MCU_UA1_RTS_N		UA1_RTS_N_ENPUD || !UA1_RTS_N_PUD	UA1_RTS_N_ENPUD || UA1_RTS_N_PUD

Pin Name	Register Name	IO Cell Control	
		PUX	PDX
MCU_UA1_CTS_N		UA1_CTS_N_ENPUD || !UA1_CTS_N_PUD	UA1_CTS_N_ENPUD || UA1_CTS_N_PUD
MCU_UA2_RXD	*[IO_CFG6]*	UA2_RXD_ENPUD || !UA2_RXD_PUD	UA2_RXD_ENPUD || UA2_RXD_PUD
MCU_UA2_TXD		UA2_TXD_ENPUD || !UA2_TXD_PUD	UA2_TXD_ENPUD || UA2_TXD_PUD
MCU_UA2_RTS_N		UA2_RTS_N_ENPUD || !UA2_RTS_N_PUD	UA2_RTS_N_ENPUD || UA2_RTS_N_PUD
MCU_UA2_CTS_N		UA2_CTS_N_ENPUD || !UA2_CTS_N_PUD	UA2_CTS_N_ENPUD || UA2_CTS_N_PUD
MCU_SPIM0_CS_N	*[IO_CFG7]*	SPIM0_CS_N_ENPUD || !SPIM0_CS_N_PUD	SPIM0_CS_N_ENPUD || SPIM0_CS_N_PUD
MCU_SPIM0_CLK		SPIM0_CLK_ENPUD || !SPIM0_CLK_PUD	SPIM0_CLK_ENPUD || SPIM0_CLK_PUD
MCU_SPIM0_MOSI		SPIM0_MOSI_ENPUD || !SPIM0_MOSI_PUD	SPIM0_MOSI_ENPUD || SPIM0_MOSI_PUD
MCU_SPIM0_MISO		SPIM0_MISO_ENPUD || !SPIM0_MISO_PUD	SPIM0_MISO_ENPUD || SPIM0_MISO_PUD
MCU_SPIM1_CS_N		SPIM1_CS_N_ENPUD || !SPIM1_CS_N_PUD	SPIM1_CS_N_ENPUD || SPIM1_CS_N_PUD
MCU_SPIM1_CLK		SPIM1_CLK_ENPUD || !SPIM1_CLK_PUD	SPIM1_CLK_ENPUD || SPIM1_CLK_PUD
MCU_SPIM1_MOSI		SPIM1_MOSI_ENPUD || !SPIM1_MOSI_PUD	SPIM1_MOSI_ENPUD || SPIM1_MOSI_PUD
MCU_SPIM1_MISO		SPIM1_MISO_ENPUD || !SPIM1_MISO_PUD	SPIM1_MISO_ENPUD || SPIM1_MISO_PUD
MCU_SPIM2_CS_N	*[IO_CFG8]*	SPIM2_CS_N_ENPUD || !SPIM2_CS_N_PUD	SPIM2_CS_N_ENPUD || SPIM2_CS_N_PUD
MCU_SPIM2_CLK		SPIM2_CLK_ENPUD || !SPIM2_CLK_PUD	SPIM2_CLK_ENPUD || SPIM2_CLK_PUD
MCU_SPIM2_MOSI		SPIM2_MOSI_ENPUD || !SPIM2_MOSI_PUD	SPIM2_MOSI_ENPUD || SPIM2_MOSI_PUD
MCU_SPIM2_MISO		SPIM2_MISO_ENPUD || !SPIM2_MISO_PUD	SPIM2_MISO_ENPUD || SPIM2_MISO_PUD
MCU_SPIM3_CS_N		SPIM3_CS_N_ENPUD || !SPIM3_CS_N_PUD	SPIM3_CS_N_ENPUD || SPIM3_CS_N_PUD
MCU_SPIM3_CLK		SPIM3_CLK_ENPUD || !SPIM3_CLK_PUD	SPIM3_CLK_ENPUD || SPIM3_CLK_PUD
MCU_SPIM3_MOSI		SPIM3_MOSI_ENPUD || !SPIM3_MOSI_PUD	SPIM3_MOSI_ENPUD || SPIM3_MOSI_PUD
MCU_SPIM3_MISO		SPIM3_MISO_ENPUD || !SPIM3_MISO_PUD	SPIM3_MISO_ENPUD || SPIM3_MISO_PUD
MCU_SPIC_CS_N	*[IO_CFG9]*	SPIC_CS_N_ENPUD || !SPIC_CS_N_PUD	SPIC_CS_N_ENPUD || SPIC_CS_N_PUD
MCU_SPIC_CLK		SPIC_CLK_ENPUD || !SPIC_CLK_PUD	SPIC_CLK_ENPUD || SPIC_CLK_PUD
MCU_SPIC_MOSI		SPIC_MOSI_ENPUD || !SPIC_MOSI_PUD	SPIC_MOSI_ENPUD || SPIC_MOSI_PUD
MCU_SPIC_MISO		SPIC_MISO_ENPUD || !SPIC_MISO_PUD	SPIC_MISO_ENPUD || SPIC_MISO_PUD
MCU_SPIC_IO2		SPIC_IO2_ENPUD || !SPIC_IO2_PUD	SPIC_IO2_ENPUD || SPIC_IO2_PUD
MCU_SPIC_IO3		SPIC_IO3_ENPUD || !SPIC_IO3_PUD	SPIC_IO3_ENPUD || SPIC_IO3_PUD
MCU_ADC24_SYNC	*[IO_CFG10]*	ADC24_SYNC_ENPUD || !ADC24_SYNC_PUD	ADC24_SYNC_ENPUD || ADC24_SYNC_PUD

Table 3.3 IO configuration registers and IO cell control signals (Drivability)

GCONF Register		IO Cell Control		Drivability
[IO_CFGn].*_CTL2	[IO_CFGn].*_CTL1	CTL2	CTL1	
0	0	0	0	2 mA (*1)
0	1	0	1	4 mA (*1)
1	0	1	0	6 mA (*1)
1	1	1	1	8 mA (*1)

(*1) The each current value is different depending on VDD33 voltage. Refer to DC Characteristics of each product Technical Data sheet.

Table 3.4 IO configuration registers and MCU chip pins (Drivability)

Pin Name	Register Name	IO Cell Control	
		CTL2	CTL1
MCU_DBG_TCK	*[IO_CFG11]*	DBG_CTL2	DBG_CTL1
MCU_DBG_TDO			
MCU_DBG_TDI			
MCU_DBG_TMS			
MCU_DBG_TRST_N			
MCU_GPIO_0	*[IO_CFG0]*	—	—
MCU_GPIO_1		GPIO_1_CTL2	GPIO_1_CTL1
MCU_GPIO_2		GPIO_2_CTL2	GPIO_2_CTL1
MCU_GPIO_3		GPIO_3_CTL2	GPIO_3_CTL1
MCU_GPIO_4		GPIO_4_CTL2	GPIO_4_CTL1
MCU_GPIO_5		GPIO_5_CTL2	GPIO_5_CTL1
MCU_GPIO_6		GPIO_6_CTL2	GPIO_6_CTL1
MCU_GPIO_7		GPIO_7_CTL2	GPIO_7_CTL1
MCU_GPIO_8	*[IO_CFG1]*	GPIO_8_CTL2	GPIO_8_CTL1
MCU_GPIO_9		GPIO_9_CTL2	GPIO_9_CTL1
MCU_GPIO_10		GPIO_10_CTL2	GPIO_10_CTL1
MCU_GPIO_11		GPIO_11_CTL2	GPIO_11_CTL1
MCU_GPIO_12		GPIO_12_CTL2	GPIO_12_CTL1
MCU_GPIO_13		GPIO_13_CTL2	GPIO_13_CTL1
MCU_GPIO_14		GPIO_14_CTL2	GPIO_14_CTL1
MCU_GPIO_15		GPIO_15_CTL2	GPIO_15_CTL1
MCU_GPIO_24	*[IO_CFG3]*	GPIO_24_CTL2	GPIO_24_CTL1
MCU_GPIO_25		GPIO_25_CTL2	GPIO_25_CTL1
MCU_GPIO_26		GPIO_26_CTL2	GPIO_26_CTL1
MCU_GPIO_27		GPIO_27_CTL2	GPIO_27_CTL1
MCU_GPIO_28		GPIO_28_CTL2	GPIO_28_CTL1
MCU_GPIO_29		GPIO_29_CTL2	GPIO_29_CTL1
MCU_GPIO_30		GPIO_30_CTL2	GPIO_30_CTL1
MCU_GPIO_31		GPIO_31_CTL2	GPIO_31_CTL1
MCU_I2C0_DATA	*[IO_CFG4]*	I2C0_DATA_CTL2	I2C0_DATA_CTL1
MCU_I2C0_CLK		I2C0_CLK_CTL2	I2C0_CLK_CTL1
MCU_I2C1_DATA		I2C1_DATA_CTL2	I2C1_DATA_CTL1
MCU_I2C1_CLK		I2C1_CLK_CTL2	I2C1_CLK_CTL1
MCU_I2C2_DATA		I2C2_DATA_CTL2	I2C2_DATA_CTL1
MCU_I2C2_CLK		I2C2_CLK_CTL2	I2C2_CLK_CTL1

Pin Name	Register Name	IO Cell Control	
		CTL2	CTL1
MCU_UA0_RXD	*[IO_CFG5]*	UA0_RXD_CTL2	UA0_RXD_CTL1
MCU_UA0_TXD		UA0_TXD_CTL2	UA0_TXD_CTL1
MCU_UA1_RXD		UA1_RXD_CTL2	UA1_RXD_CTL1
MCU_UA1_TXD		UA1_TXD_CTL2	UA1_TXD_CTL1
MCU_UA1_RTS_N		UA1_RTS_N_CTL2	UA1_RTS_N_CTL1
MCU_UA1_CTS_N		UA1_CTS_N_CTL2	UA1_CTS_N_CTL1
MCU_UA2_RXD	*[IO_CFG6]*	UA2_RXD_CTL2	UA2_RXD_CTL1
MCU_UA2_TXD		UA2_TXD_CTL2	UA2_TXD_CTL1
MCU_UA2_RTS_N		UA2_RTS_N_CTL2	UA2_RTS_N_CTL1
MCU_UA2_CTS_N		UA2_CTS_N_CTL2	UA2_CTS_N_CTL1
MCU_SPIM0_CS_N	*[IO_CFG7]*	SPIM0_CS_N_CTL2	SPIM0_CS_N_CTL1
MCU_SPIM0_CLK		SPIM0_CLK_CTL2	SPIM0_CLK_CTL1
MCU_SPIM0_MOSI		SPIM0_MOSI_CTL2	SPIM0_MOSI_CTL1
MCU_SPIM0_MISO		SPIM0_MISO_CTL2	SPIM0_MISO_CTL1
MCU_SPIM1_CS_N		SPIM1_CS_N_CTL2	SPIM1_CS_N_CTL1
MCU_SPIM1_CLK		SPIM1_CLK_CTL2	SPIM1_CLK_CTL1
MCU_SPIM1_MOSI		SPIM1_MOSI_CTL2	SPIM1_MOSI_CTL1
MCU_SPIM1_MISO		SPIM1_MISO_CTL2	SPIM1_MISO_CTL1
MCU_SPIM2_CS_N	*[IO_CFG8]*	SPIM2_CS_N_CTL2	SPIM2_CS_N_CTL1
MCU_SPIM2_CLK		SPIM2_CLK_CTL2	SPIM2_CLK_CTL1
MCU_SPIM2_MOSI		SPIM2_MOSI_CTL2	SPIM2_MOSI_CTL1
MCU_SPIM2_MISO		SPIM2_MISO_CTL2	SPIM2_MISO_CTL1
MCU_SPIM3_CS_N		SPIM3_CS_N_CTL2	SPIM3_CS_N_CTL1
MCU_SPIM3_CLK		SPIM3_CLK_CTL2	SPIM3_CLK_CTL1
MCU_SPIM3_MOSI		SPIM3_MOSI_CTL2	SPIM3_MOSI_CTL1
MCU_SPIM3_MISO		SPIM3_MISO_CTL2	SPIM3_MISO_CTL1
MCU_SPIC_CS_N	*[IO_CFG9]*	SPIC_CS_N_CTL2	SPIC_CS_N_CTL1
MCU_SPIC_CLK		SPIC_CLK_CTL2	SPIC_CLK_CTL1
MCU_SPIC_MOSI		SPIC_MOSI_CTL2	SPIC_MOSI_CTL1
MCU_SPIC_MISO		SPIC_MISO_CTL2	SPIC_MISO_CTL1
MCU_SPIC_IO2		SPIC_IO2_CTL2	SPIC_IO2_CTL1
MCU_SPIC_IO3		SPIC_IO3_CTL2	SPIC_IO3_CTL1
MCU_ADC24_SYNC	*[IO_CFG10]*	ADC24_SYNC_CTL2	ADC24_SYNC_CTL1

Table 3.5 IO configuration registers and MCU chip pins (ZIHD, ZH / RESX / LATI / STBX)

Pin Name	IO Cell Control			
	ZIHD,ZH	RESX	LATI	STBX
MCU_DBG_TCK	—	***[CTRL_IO_AON_0]***. CTRL_IO_AON_PM_RESX	***[CTRL_IO_AON_1]***. CTRL_IO_AON_PM_LATI	(Fixed to 1)
MCU_DBG_TDO	—			(Fixed to 0)
MCU_DBG_TDI	—			(Fixed to 1)
MCU_DBG_TMS	—			(Fixed to 1)
MCU_DBG_TRST_N	—			(Fixed to 1)
MCU_GPIO_0	***[IRQ_SETTING_0]***. IRQ_GPIO0_setting	—	—	—
MCU_GPIO_1	***[IRQ_SETTING_0]***. IRQ_GPIO1_setting	***[CTRL_IO_AON_0]***. CTRL_IO_AON_PM_RESX	***[CTRL_IO_AON_1]***. CTRL_IO_AON_PM_LATI	***[CTRL_IO_AON_3]***. CTRL_IO_GPIO_1_STBX
MCU_GPIO_2	***[IRQ_SETTING_0]***. IRQ_GPIO2_setting			***[CTRL_IO_AON_3]***. CTRL_IO_GPIO_2_STBX
MCU_GPIO_3	***[IRQ_SETTING_0]***. IRQ_GPIO3_setting			***[CTRL_IO_AON_3]***. CTRL_IO_GPIO_3_STBX
MCU_GPIO_4	***[IRQ_SETTING_0]***. IRQ_GPIO4_setting			***[CTRL_IO_AON_3]***. CTRL_IO_GPIO_4_STBX
MCU_GPIO_5	***[IRQ_SETTING_0]***. IRQ_GPIO5_setting			***[CTRL_IO_AON_3]***. CTRL_IO_GPIO_5_STBX
MCU_GPIO_6	***[IRQ_SETTING_0]***. IRQ_GPIO6_setting			***[CTRL_IO_AON_3]***. CTRL_IO_GPIO_6_STBX
MCU_GPIO_7	***[IRQ_SETTING_0]***. IRQ_GPIO7_setting			***[CTRL_IO_AON_3]***. CTRL_IO_GPIO_7_STBX
MCU_GPIO_8	—			***[CTRL_IO_AON_3]***. CTRL_IO_GPIO_8_STBX
MCU_GPIO_9	—			***[CTRL_IO_AON_3]***. CTRL_IO_GPIO_9_STBX
MCU_GPIO_10	—			***[CTRL_IO_AON_3]***. CTRL_IO_GPIO_10_STBX
MCU_GPIO_11	—			***[CTRL_IO_AON_3]***. CTRL_IO_GPIO_11_STBX
MCU_GPIO_12	—			***[CTRL_IO_AON_3]***. CTRL_IO_GPIO_12_STBX
MCU_GPIO_13	—			***[CTRL_IO_AON_3]***. CTRL_IO_GPIO_13_STBX

Pin Name	IO Cell Control			
	ZIHD,ZH	RESX	LATI	STBX
MCU_GPIO_14	—			***[CTRL_IO_AON_3]***. CTRL_IO_GPIO_14_STBX
MCU_GPIO_15	—			***[CTRL_IO_AON_3]***. CTRL_IO_GPIO_15_STBX
MCU_GPIO_24	***[IRQ_SETTING_1]***. IRQ_GPIO24_setting			***[CTRL_IO_AON_3]***. CTRL_IO_GPIO_24_STBX
MCU_GPIO_25	***[IRQ_SETTING_1]***. IRQ_GPIO25_setting			***[CTRL_IO_AON_3]***. CTRL_IO_GPIO_25_STBX
MCU_GPIO_26	***[IRQ_SETTING_1]***. IRQ_GPIO26_setting			***[CTRL_IO_AON_3]***. CTRL_IO_GPIO_26_STBX
MCU_GPIO_27	***[IRQ_SETTING_1]***. IRQ_GPIO27_setting			***[CTRL_IO_AON_3]***. CTRL_IO_GPIO_27_STBX
MCU_GPIO_28	—			***[CTRL_IO_AON_3]***. CTRL_IO_GPIO_28_STBX
MCU_GPIO_29	—			***[CTRL_IO_AON_3]***. CTRL_IO_GPIO_29_STBX
MCU_GPIO_30	***[IRQ_SETTING_1]***. IRQ_GPIO30_setting			***[CTRL_IO_AON_3]***. CTRL_IO_GPIO_30_STBX
MCU_GPIO_31	—			***[CTRL_IO_AON_3]***. CTRL_IO_GPIO_31_STBX
MCU_I2C0_DATA	—	***[CTRL_IO_AON_4]***. CTRL_IO_AON_PP1_RESX	***[CTRL_IO_AON_5]***. CTRL_IO_AON_PP1_LATI	***[CTRL_IO_AON_6]***. CTRL_IO_I2C0_DATA_STBX
MCU_I2C0_CLK	—			***[CTRL_IO_AON_6]***. CTRL_IO_I2C0_CLK_STBX
MCU_I2C1_DATA	—			***[CTRL_IO_AON_6]***. CTRL_IO_I2C1_DATA_STBX
MCU_I2C1_CLK	—			***[CTRL_IO_AON_6]***. CTRL_IO_I2C1_CLK_STBX
MCU_I2C2_DATA	—	***[CTRL_IO_AON_0]***. CTRL_IO_AON_PM_RESX	***[CTRL_IO_AON_1]***. CTRL_IO_AON_PM_LATI	***[CTRL_IO_AON_2]***. CTRL_IO_I2C2_STBX
MCU_I2C2_CLK	—			
MCU_UA0_RXD	—	***[CTRL_IO_AON_4]***. CTRL_IO_AON_PP1_RESX	***[CTRL_IO_AON_5]***. CTRL_IO_AON_PP1_LATI	***[CTRL_IO_AON_6]***. CTRL_IO_UART0_STBX
MCU_UA0_TXD	—			
MCU_UA1_RXD	—			***[CTRL_IO_AON_6]***. CTRL_IO_UA1_RXD_STBX
MCU_UA1_TXD	—			***[CTRL_IO_AON_6]***. CTRL_IO_UA1_TXD_STBX

Pin Name	IO Cell Control			
	ZIHD,ZH	RESX	LATI	STBX
MCU_UA1_RTS_N	—			***[CTRL_IO_AON_6]***. CTRL_IO_UA1_RTS_N_STBX
MCU_UA1_CTS_N	—			***[CTRL_IO_AON_6]***. CTRL_IO_UA1_CTS_N_STBX
MCU_UA2_RXD	—	***[CTRL_IO_AON_0]***. CTRL_IO_AON_PM_RESX	***[CTRL_IO_AON_1]***. CTRL_IO_AON_PM_LATI	***[CTRL_IO_AON_2]***. CTRL_IO_UA2D_STBX
MCU_UA2_TXD	—			***[CTRL_IO_AON_2]***. CTRL_IO_UA2D_STBX
MCU_UA2_RTS_N	—			***[CTRL_IO_AON_2]***. CTRL_IO_UA2_RTS_N_STBX
MCU_UA2_CTS_N	—			***[CTRL_IO_AON_2]***. CTRL_IO_UA2_CTS_N_STBX
MCU_SPIM0_CS_N	—	***[CTRL_IO_AON_4]***. CTRL_IO_AON_PP1_RESX	***[CTRL_IO_AON_5]***. CTRL_IO_AON_PP1_LATI	***[CTRL_IO_AON_6]***. CTRL_IO_SPIM0_CS_N_STBX
MCU_SPIM0_CLK	—			***[CTRL_IO_AON_6]***. CTRL_IO_SPIM0_CLK_STBX
MCU_SPIM0_MOSI	—			***[CTRL_IO_AON_6]***. CTRL_IO_SPIM0_MOSI_STBX
MCU_SPIM0_MISO	—			***[CTRL_IO_AON_6]***. CTRL_IO_SPIM0_MISO_STBX
MCU_SPIM1_CS_N	—			***[CTRL_IO_AON_6]***. CTRL_IO_SPIM1_CS_N_STBX
MCU_SPIM1_CLK	—			***[CTRL_IO_AON_6]***. CTRL_IO_SPIM1_CLK_STBX
MCU_SPIM1_MOSI	—			***[CTRL_IO_AON_6]***. CTRL_IO_SPIM1_MOSI_STBX
MCU_SPIM1_MISO	—			***[CTRL_IO_AON_6]***. CTRL_IO_SPIM1_MISO_STBX
MCU_SPIM2_CS_N	—	***[CTRL_IO_AON_0]***. CTRL_IO_AON_PM_RESX	***[CTRL_IO_AON_1]***. CTRL_IO_AON_PM_LATI	***[CTRL_IO_AON_2]***. CTRL_IO_SPIM2_STBX
MCU_SPIM2_CLK	—			
MCU_SPIM2_MOSI	—			
MCU_SPIM2_MISO	—			
MCU_SPIM3_CS_N	—			***[CTRL_IO_AON_2]***. CTRL_IO_SPIM3_STBX
MCU_SPIM3_CLK	—			
MCU_SPIM3_MOSI	—			
MCU_SPIM3_MISO	—			

Pin Name	IO Cell Control			
	ZIHD,ZH	RESX	LATI	STBX
MCU_SPIC_CS_N	—	(Fixed to 1)	(Fixed to 0)	(Fixed to 1)
MCU_SPIC_CLK	—	(Fixed to 1)	(Fixed to 0)	(Fixed to 1)
MCU_SPIC_MOSI	—	(Fixed to 1)	(Fixed to 0)	(Fixed to 1)
MCU_SPIC_MISO	—	(Fixed to 1)	(Fixed to 0)	(Fixed to 1)
MCU_SPIC_IO2	—	(Fixed to 1)	(Fixed to 0)	(Fixed to 1)
MCU_SPIC_IO3	—	(Fixed to 1)	(Fixed to 0)	(Fixed to 1)
MCU_ADC24_SYNC	—	***[CTRL_IO_AON_0]***. CTRL_IO_AON_PM_RESX	***[CTRL_IO_AON_1]***. CTRL_IO_AON_PM_LATI	***[CTRL_IO_AON_2]***. CTRL_IO_ADC24_SYNC_STBX

4. Usage

4.1. Power Up

At the power-up, software should do the following settings regarding to IO.

(1) Input standby deassertion

All bidirectional IO cells except the MCU_DBG_*/MCU_SPIC_* are in the input standby state (STBX = 0). The IO cell used as an input pin should be set to STBX = 1 in the following PMU registers.

PMU registers	Bit name	value
[CTRL_IO_AON_2]	CTRL_IO_*_STBX	(Set the IO cell to be used as input to STBX = 1)
[CTRL_IO_AON_3]	CTRL_IO_*_STBX	(Set the IO cell to be used as input to STBX = 1)
[CTRL_IO_AON_6]	CTRL_IO_*_STBX	(Set the IO cell to be used as input to STBX = 1)

(2) IO function setting

The initial state of all IO cells except the SPIC pin, which have the multiplex function, is in "No Function (No connected modules to the input)." The appropriate settings should be done using the following GCONF registers.

GCONF registers	Bit name	value
[FMODE_CFG0]	*_FMODE	(Set according to the function to use)
[FMODE_CFG1]	*_FMODE	(Set according to the function to use)
[FMODE_CFG2]	*_FMODE	(Set according to the function to use)
[FMODE_CFG3]	*_FMODE	(Set according to the function to use)
[FMODE_CFG4]	*_FMODE	(Set according to the function to use)
[FMODE_CFG5]	*_FMODE	(Set according to the function to use)
[FMODE_CFG6]	*_FMODE	(Set according to the function to use)

Depending on the function to use, should be set to 1 (output enable) using the following GCONF register.

GCONF registers	Bit name	value
[OE_CTRL]	*_OE	(Set 1 to the controller to be used)

(3) IO pull-up or pull-down setting and output drivability setting

All bidirectional IO cells except the MCU_DBG_*/MCU_SPIC_* are in the initial state of the pull-up or pull-down selection (depending on the cell types) and the smallest drivability. The appropriate settings should be done using the following GCONF registers.

GCONF registers	Bit name	value
[IO_CFG0]	*_ENPUD, *_PUD, *_CTL1, *_CTL2	(Set according to the need)
[IO_CFG1]	*_ENPUD, *_PUD, *_CTL1, *_CTL2	(Set according to the need)
[IO_CFG2]	*_ENPUD, *_PUD, *_CTL1, *_CTL2	(Set according to the need)
[IO_CFG3]	*_ENPUD, *_PUD, *_CTL1, *_CTL2	(Set according to the need)
[IO_CFG4]	*_ENPUD, *_PUD, *_CTL1, *_CTL2	(Set according to the need)
[IO_CFG5]	*_ENPUD, *_PUD, *_CTL1, *_CTL2	(Set according to the need)
[IO_CFG6]	*_ENPUD, *_PUD, *_CTL1, *_CTL2	(Set according to the need)
[IO_CFG7]	*_ENPUD, *_PUD, *_CTL1, *_CTL2	(Set according to the need)
[IO_CFG8]	*_ENPUD, *_PUD, *_CTL1, *_CTL2	(Set according to the need)
[IO_CFG9]	*_ENPUD, *_PUD, *_CTL1, *_CTL2	(Set according to the need)
[IO_CFG10]	*_ENPUD, *_PUD, *_CTL1, *_CTL2	(Set according to the need)
[IO_CFG11]	*_ENPUD, *_PUD, *_CTL1, *_CTL2	(Set according to the need)

4.2. RTC/STOP Mode

- Mode Transition

(1) IO cell reset or protection setting

In the RTC or STOP mode, the Power Domains operating at VDD12 is shut down. The VDD12 signals become unstable including the signals connected to the IO. To prevent the outputs of the external pins from being unstable, the protection signal (LATI) becomes 1 by the hardware sequencer, and the states of the IO cell output buffer and the pull-up or pull-down are protected.

By setting the following PMU registers by the software, the reset signal (RESX) becomes to 0, it is possible to reset the states of the IO cell output buffer and the pull-up or pull-down. (When RESX = 0, the output from the external pin is not protected.)

PMU registers	Bit name	value
[CTRL_IO_AON_0]	CTRL_IO_AON_PM_RESX	0x00000000
[CTRL_IO_AON_4]	CTRL_IO_AON_PP1_RESX	0x00000000

(2) Input Standby

Some MCU's IO pins receive signals from external devices whose power is shut down in the RTC or STOP mode. Before transition to the mode, set the following PMU registers in the software, the standby control signal of the IO cell (STBX) should be 0 to prevent the input buffer from generating the penetrate current.

PMU registers	Bit name	value
[CTRL_IO_AON_2]	CTRL_IO_*_STBX	(Set 0 if necessary)
[CTRL_IO_AON_3]	CTRL_IO_*_STBX	(Set 0 if necessary)
[CTRL_IO_AON_6]	CTRL_IO_*_STBX	(Set 0 if necessary)

- Return from RTC/STOP modes

(3) IO Cell

The GPIO and the peripheral blocks are in the initial state because the Power Domains operating at VDD12 has been shut down. The IO cells should be set to the normal operation by the procedure as the power-up. (If the software deasserts the protection signal (LATI) before the peripheral circuits and other settings, the external pins (IO cells) become to the initial state.)

(4) IO cell reset and protection deassertion
The IO cell reset or protection signal should be deasserted by the software. RESX = 1 and LATI = 0 should be set in the PMU registers.

PMU registers	Bit name	value
[CTRL_IO_AON_1]	CTRL_IO_AON_PM_LATI	0x00000000
[CTRL_IO_AON_5]	CTRL_IO_AON_PP1_LATI	0x00000000

If the software sets 0 to the reset signal (RESX) at the mode transition, the following PMU register settings is required by the software.

PMU registers	Bit name	value
[CTRL_IO_AON_0]	CTRL_IO_AON_PM_RESX	0x00000001
[CTRL_IO_AON_4]	CTRL_IO_AON_PP1_RESX	0x00000001

4.3. RETENTION Mode

- Mode Transition
  The protection signal (LATI) becomes 1 by the hardware sequencer, and the states of the IO cell output buffer and the pull-up or pull-down are protected. Setting by the software is not required.

- Return from RETENTION Mode
  In the RETENTION mode, the states of Power Domains operating at VDD12are protected. Since the protection signal (LATI) becomes 0 by the hardware sequencer at the time of return, setting by the software is not required.

4.4. SLEEP/WAIT Mode

In the SLEEP or WAIT mode, the power for Power Domains operating at VDD12 is supplied. The IO cell state is protected. No procedures at the mode transition or return are necessary.

4.5. WAIT-RETENTION Mode

In the WAIT-RETENTION mode, the PP1 domain is in the RETENTION state. Since the protection signal (LATI) for the external signals of the PPI domain becomes 1 at the mode transition and becomes 0 at the return by the hardware sequencer, setting by the software is not required.

5. Revision History

Table 5.1 Revision History

Revision	Date	Description
0.1	2014-09-29	Newly released
0.2	2015-01-07	Section 4 Corrected error of description
0.3	2015-01-20	Section 4.2 Added description to (3) IO Cell
1.0	2015-01-21	Official version
1.1	2015-05-28	Section 3.1.1. 3.1.2. Table 3.3 Corrected current value and added description.
1.2	2018-02-02	Changed header, footer and the last page. Changed corporate name and descriptions. Modified trademark description. Corrected typo.

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, 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, devices related to electric power, and equipment used in finance-related fields. **IF YOU USE PRODUCT FOR UNINTENDED USE, TOSHIBA ASSUMES NO LIABILITY FOR PRODUCT.** For details, please contact your TOSHIBA sales representative.
- 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.**