CSK6012 Voice Recognition SoC Datasheet

CSDS-22001-021_V1.0   **November 22, 2022**

DISCLAIMER

Update History

Version	Date	Update Description
V1.0	November 22, 2022	Initial release.

Table of Contents

No matter where you go, there you are.

—Banzai

1　版權所有

2　章節自動編號

```
.. sectnum::
```

3 Topic

If CLOCK_MODE equals 1, there is no idle time between back-to-back characters if data is ready in the transmit FIFO. In this case, because *sync_delay* equals one pclk as described in Section 8 *Equation Example*, the requirement to avoid idle time between consecutive characters is met for all {DLH,DLL} values.

This paragraph might be rendered in a custom way.

If CLOCK_MODE equals 1, there is no idle time between back-to-back characters if data is ready in the transmit FIFO. In this case, because *sync_delay* equals one pclk as described in Section 8 *Equation Example*, the requirement to avoid idle time between consecutive characters is met for all {DLH,DLL} values.

試試行不行

If CLOCK_MODE equals 1, there is no idle time between back-to-back characters if data is ready in the transmit FIFO. In this case, because *sync_delay* equals one pclk as described in Section 8 *Equation Example*, the requirement to avoid idle time between consecutive characters is met for all {DLH,DLL} values.

出現的位置

出現在哪裏呢這個 sidebar

---

4 分（不管用?)

- good
- bad
- excellent
- normal
- qualified
- good
- bad
- excellent
- normal
- qualified
- good
- bad
- excellent
- normal
- qualified
- good
- bad
- excellent
- normal
- qualified
- good
- bad
- excellent
- normal
- qualified

5 加粗居中

confidential

Confidential

Confidential

6   鏈接 1

If CLOCK_MODE equals 1, there is no idle time between back-to-back characters if data is ready in the transmit FIFO. In this case, because *sync_delay* equals one pclk as described in Section 8 *Equation Example*, the requirement to avoid idle time between consecutive characters is met for all {DLH,DLL} values.

Section 8 *Equation Example* is okay,

Section 8 *Equation Example* is okay.

Section 8 *Equation Example* is not okay.

Section 8 *Equation Example* is not okay.

Section 8 *Equation Example* is not okay.

refer to `dmas` *DMA Support*.

7   鏈接 2

If CLOCK_MODE equals 1, there is no idle time between back-to-back characters if data is ready in the transmit FIFO. In this case, because *sync_delay* equals one pclk as described in Section 8.1 *Error*, the requirement to avoid idle time between consecutive characters is met for all {DLH,DLL} values.

8 Equation Example

8.1 Error

說明：如果 text 大括號內有短線，這個命令的使用會導致報錯。在前面加上反斜線就 okay.

$$\frac{\text{Rate of SSI data transmission}}{Rate of DW\_ahb\_dmac response to destination burst requests}$$

$$\frac{\text{Rate of SSI data transmission}}{\text{Rate of DW\_ahb\_dmac response to destination burst requests}}$$

8.2 空格的寫法

$$14 \ (round off \ value)$$

$$14 \ (\text{roundoff value})$$

8.3 短線的寫法

前面加 slash

$$\text{DLF} = BRD_F * 2^{DLF\_SIZE}$$

$$BRD_F * 2^{DLF\_SIZE}$$

$$\text{DLF} = BRD_F * 2^{DLF\_SIZE}$$

This equation ref{equ1} is not okay?

$$\text{DLF} = BRD_F * 2^{DLF\_SIZE} = 0.866132364 * 16 = 13.858117824 = 14 \ (\text{roundoff value})$$

Therefore, the Generated Baud Rate (GBR) is as follows:

$$GBR = \frac{\text{Serial Clock}}{(16 \times GD)} = \frac{133}{16 \times 1.875} = 4433333.333$$

Now the error is calculated as follows:

$$Error = \frac{\text{GBR - RBR}}{RBR} = 0.004729$$

The error percentage is as follows:

$$Error\% = 0.004729 \times 100 = 0.473$$

9 CMN_IOMUX

9.1 Introduction

选择 GPIO 的复用功能。

9.2 Main Features

每个 GPIO 有 16 个复用功能可以配置，具体的配置功能需要参考 projectsVenusASICDigitalIP-sIOMUXVenus_IO_New.xlsx。

9.3 Function Description

9.4 Basic Block Diagram

9.5 CMN_IOMUX Register

9.5.1 PAD_GPIOA (000)

Table 9.1 PAD_GPIOA Register

Name	Bit	Type	Description	Reset
Reserved	31:26	RO	Not used	6' h0
pad_gpioa___drv	25:24	RW	Pad drive strength configuration	2' h3
Reserved	23:23	RW	Not used	1' h0
pad_gpioa___out_frc	22:22	RW	Pad output value force control 1: "out_reg"value will overwrite output value of normal function 0: output value is controlled by normal function	1' h0
pad_gpioa___out_reg	21:21	RW	Pad output value when out_frc ==1	1' h0

Continued on next page

Table 9.1 – continued from previous page

Name	Bit	Type	Description	Reset
pad_gpioa___oen_frc	20:20	RW	Pad output enable force control 1: "oen_reg" value will overwrite the oen control logic of normal function 0: oen is controlled by normal function	1' h0
pad_gpioa___oen_reg	19:19	RW	Pad oen value when oen_frc==1	1' h0
pad_gpioa___pull_frc	18:18	RW	Pad Pullup/Pulldown force control 1: pull_up/pull_dn value below will overwrite the Pullup/Pulldown of normal function 0: Pullup/Pulldown is controlled by normal function	1' h0
pad_gpioa___pull_up	17:17	RW	Pad pullup enable control when pull_frc ==1 0: pullup is disabled 1:pullup is enabled	1' h0
pad_gpioa___pull_dn	16:16	RW	Pad pulldown eanble control when pull_frc==1 0: pulldown is disabled 1: pulldown is enabled	1' h0
Reserved	15:9	RO	Not used	7' h0
pad_gpioa___ana_sel	8:5	RW	Pad analog function selection, only used on Analog/Digital mixed I/Os	4' h0
pad_gpioa___fsel	4:0	RW	Pad function selection	5' h0

9.5.2 PAD_GPIOB (080)

Table 9.2 PAD_GPIOB Register

Name	Bit	Type	Description	Reset
Reserved	31:26	RO	Not used	6' h0
pad_gpiob___drv	25:24	RW	Pad drive strength configuration	2' h3
Reserved	23:23	RW	Not used	1' h0
pad_gpiob___out_frc	22:22	RW	Pad output value force control 1: "out_reg"value will overwrite output value of normal function 0: output value is controlled by normal function	1' h0
pad_gpiob___out_reg	21:21	RW	Pad output value when out_frc ==1	1' h0
pad_gpiob___oen_frc	20:20	RW	Pad output enable force control 1: "oen_reg" value will overwrite the oen control logic of normal function 0: oen is controlled by normal function	1' h0
pad_gpiob___oen_reg	19:19	RW	Pad oen value when oen_frc==1	1' h0
pad_gpiob___pull_frc	18:18	RW	Pad Pullup/Pulldown force control 1: pull_up/pull_dn value below will overwrite the Pullup/Pulldown of normal function 0: Pullup/Pulldown is controlled by normal function	1' h0
pad_gpiob___pull_up	17:17	RW	Pad pullup enable control when pull_frc ==1 0: pullup is disabled 1:pullup is enabled	1' h0

Continued on next page

Table 9.2 – continued from previous page

Name	Bit	Type	Description	Reset
pad_gpiob___pull_dn	16:16	RW	Pad pulldown eanble control when pull_frc==1 0: pulldown is disabled 1: pulldown is enabled	1' h0
Reserved	15:9	RO	Not used	7' h0
pad_gpiob___ana_sel	8:5	RW	Pad analog function selection, only used on Analog/Digital mixed I/Os	4' h0
pad_gpiob___fsel	4:0	RW	Pad function selection	5' h0

9.5.3 PAD_FLASH_WP_N (0C0)

Table 9.3 PAD_FLASH_WP_N Register

Name	Bit	Type	Description	Reset
Reserved	31:26	RO	Not used	6' h0
pad_flash_wp_n_ds	25:24	RW	Pad drive strength configuration	2' h3
Reserved	23:23	RW	Not used	1' h0
pad_flash_wp_n_out_frc	22:22	RW	Pad output value force control 1: "out_reg"value will overwrite output value of normal function 0: output value is controlled by normal function	1' h0
pad_flash_wp_n_out_reg	21:21	RW	Pad output value when out_frc ==1	1' h0
pad_flash_wp_n_oen_frc	20:20	RW	Pad output enable force control 1: "oen_reg" value will overwrite the oen control logic of normal function 0: oen is controlled by normal function	1' h0
pad_flash_wp_n_oen_reg	19:19	RW	Pad oen value when oen_frc==1	1' h0

Continued on next page

Table 9.3 – continued from previous page

Name	Bit	Type	Description	Reset
pad_flash_wp_n_pull_frc	18:18	RW	Pad Pullup/Pulldown force control 1: pull_up/pull_dn value below will overwrite the Pullup/Pulldown of normal function 0: Pullup/Pulldown is controlled by normal function	1' h0
pad_flash_wp_n_pull_up	17:17	RW	Pad pullup enable control when pull_frc ==1 0: pullup is disabled 1:pullup is enabled	1' h0
pad_flash_wp_n_pull_dn	16:16	RW	Pad pulldown eanble control when pull_frc==1 0: pulldown is disabled 1: pulldown is enabled	1' h0
Reserved	15:9	RO	Not used	7' h0
pad_flash_wp_n_ana_sel	8:5	RW	Pad analog function selection, only used on Analog/Digital mixed I/Os	4' h0
pad_flash_wp_n_fsel	4:0	RW	Pad function selection	5' h0

9.5.4 PAD_FLASH_MISO (0C4)

Table 9.4 PAD_FLASH_MISO Register

Name	Bit	Type	Description	Reset
Reserved	31:26	RO	Not used	6' h0
pad_flash_miso_drv	25:24	RW	Pad drive strength configuration	2' h3
Reserved	23:23	RW	Not used	1' h0

Continued on next page

Table 9.4 – continued from previous page

Name	Bit	Type	Description	Reset
pad_flash_miso_out_frc	22:22	RW	Pad output value force control 1: "out_reg"value will overwrite output value of normal function 0: output value is controlled by normal function	1' h0
pad_flash_miso_out_reg	21:21	RW	Pad output value when out_frc ==1	1' h0
pad_flash_miso_oen_frc	20:20	RW	Pad output enable force control 1: "oen_reg" value will overwrite the oen control logic of normal function 0: oen is controlled by normal function	1' h0
pad_flash_miso_oen_reg	19:19	RW	Pad oen value when oen_frc==1	1' h0
pad_flash_miso_pull_frc	18:18	RW	Pad Pullup/Pulldown force control 1: pull_up/pull_dn value below will overwrite the Pullup/Pulldown of normal function 0: Pullup/Pulldown is controlled by normal function	1' h0
pad_flash_miso_pull_up	17:17	RW	Pad pullup enable control when pull_frc ==1 0: pullup is disabled 1:pullup is enabled	1' h0
pad_flash_miso_pull_dn	16:16	RW	Pad pulldown eanble control when pull_frc==1 0: pulldown is disabled 1: pulldown is enabled	1' h0
Reserved	15:9	RO	Not used	7' h0

Continued on next page

Table 9.4 – continued from previous page

Name	Bit	Type	Description	Reset
pad_flash_miso_ana_sel	8:5	RW	Pad analog function selection, only used on Analog/Digital mixed I/Os	4'h0
pad_flash_miso_fsel	4:0	RW	Pad function selection	5'h0

9.5.5 PAD_FLASH_CS_N (0C8)

Table 9.5 PAD_FLASH_CS_N Register

Name	Bit	Type	Description	Reset
Reserved	31:26	RO	Not used	6'h0
pad_flash_cs_n_drv	25:24	RW	Pad drive strength configuration	2'h3
Reserved	23:23	RW	Not used	1'h0
pad_flash_cs_n_out_frc	22:22	RW	Pad output value force control 1: "out_reg" value will overwrite output value of normal function 0: output value is controlled by normal function	1'h0
pad_flash_cs_n_out_reg	21:21	RW	Pad output value when out_frc ==1	1'h0
pad_flash_cs_n_oen_frc	20:20	RW	Pad output enable force control 1: "oen_reg" value will overwrite the oen control logic of normal function 0: oen is controlled by normal function	1'h0
pad_flash_cs_n_oen_reg	19:19	RW	Pad oen value when oen_frc==1	1'h0
pad_flash_cs_n_pull_frc	18:18	RW	Pad Pullup/Pulldown force control 1: pull_up/pull_dn value below will overwrite the Pullup/Pulldown of normal function 0: Pullup/Pulldown is controlled by normal function	1'h0
pad_flash_cs_n_pull_up	17:17	RW	Pad pullup enable control when pull_frc ==1 0: pullup is disabled 1:pullup is enabled	1'h0
pad_flash_cs_n_pull_dn	16:16	RW	Pad pulldown eanble control when pull_frc==1 0: pulldown is disabled 1: pulldown is enabled	1'h0
Reserved	15:9	RO	Not used	7'h0
pad_flash_cs_n_ana_sel	8:5	RW	Pad analog function selection, only used on Analog/Digital mixed I/Os	4'h0
pad_flash_cs_n_fsel	4:0	RW	Pad function selection	5'h0

9.5.6 PAD_FLASH_HOLD_N (0CC)

Table 9.6 PAD_FLASH_HOLD_N Register

Name	Bit	Type	Description	Reset
Reserved	31:26	RO	Not used	6'h0
pad_flash_hold_n_drv	25:24	RW	Pad drive strength configuration	2'h3
Reserved	23:23	RW	Not used	1'h0
pad_flash_hold_n_out_frc	22:22	RW	Pad output value force control 1: "out_reg" value will overwrite output value of normal function 0: output value is controlled by normal function	1'h0

Continued on next page

Table 9.6 – continued from previous page

Name	Bit	Type	Description	Reset
pad_flash_hold_n_out_reg	21:21	RW	Pad output value when out_frc ==1	1'h0
pad_flash_hold_n_oen_frc	20:20	RW	Pad output enable force control 1: "oen_reg" value will overwrite the oen control logic of normal function 0: oen is controlled by normal function	1'h0
pad_flash_hold_n_oen_reg	19:19	RW	Pad oen value when oen_frc==1	1'h0
pad_flash_hold_n_pull_frc	18:18	RW	Pad Pullup/Pulldown force control 1: pull_up/pull_dn value below will overwrite the Pullup/Pulldown of normal function 0: Pullup/Pulldown is controlled by normal function	1'h0
pad_flash_hold_n_pull_up	17:17	RW	Pad pullup enable control when pull_frc ==1 0: pullup is disabled 1:pullup is enabled	1'h0
pad_flash_hold_n_pull_dn	16:16	RW	Pad pulldown eanble control when pull_frc==1 0: pulldown is disabled 1: pulldown is enabled	1'h0
Reserved	15:9	RO	Not used	7'h0
pad_flash_hold_n_ana_sel	8:5	RW	Pad analog function selection, only used on Analog/Digital mixed I/Os	4'h0
pad_flash_hold_n_fsel	4:0	RW	Pad function selection	5'h0

9.5.7 PAD_FLASH_CLK (0D0)

Table 9.7 PAD_FLASH_CLK Register

Name	Bit	Type	Description	Reset
Reserved	31:26	RO	Not used	6'h0
pad_flash_clk_drv	25:24	RW	Pad drive strength configuration	2'h3
Reserved	23:23	RW	Not used	1'h0
pad_flash_clk_out_frc	22:22	RW	Pad output value force control 1: "out_reg" value will overwrite output value of normal function 0: output value is controlled by normal function	1'h0
pad_flash_clk_out_reg	21:21	RW	Pad output value when out_frc ==1	1'h0
pad_flash_clk_oen_frc	20:20	RW	Pad output enable force control 1: "oen_reg" value will overwrite the oen control logic of normal function 0: oen is controlled by normal function	1'h0
pad_flash_clk_oen_reg	19:19	RW	Pad oen value when oen_frc==1	1'h0
pad_flash_clk_pull_frc	18:18	RW	Pad Pullup/Pulldown force control 1: pull_up/pull_dn value below will overwrite the Pullup/Pulldown of normal function 0: Pullup/Pulldown is controlled by normal function	1'h0
pad_flash_clk_pull_up	17:17	RW	Pad pullup enable control when pull_frc ==1 0: pullup is disabled 1:pullup is enabled	1'h0
pad_flash_clk_pull_dn	16:16	RW	Pad pulldown eanble control when pull_frc==1 0: pulldown is disabled 1: pulldown is enabled	1'h0
Reserved	15:9	RO	Not used	7'h0
pad_flash_clk_ana_sel	8:5	RW	Pad analog function selection, only used on Analog/Digital mixed I/Os	4'h0

Continued on next page

Table 9.7 – continued from previous page

Name	Bit	Type	Description	Reset
pad_flash_clk_fsel	4:0	RW	Pad function selection	5'h0

9.5.8 PAD_FLASH_MOSI (0D4)

Table 9.8 PAD_FLASH_MOSI Register

Name	Bit	Type	Description	Reset
Reserved	31:26	RO	Not used	6'h0
pad_flash_mosi_drv	25:24	RW	Pad drive strength configuration	2'h3
Reserved	23:23	RW	Not used	1'h0
pad_flash_mosi_out_frc	22:22	RW	Pad output value force control 1: "out_reg" value will overwrite output value of normal function 0: output value is controlled by normal function	1'h0
pad_flash_mosi_out_reg	21:21	RW	Pad output value when out_frc ==1	1'h0
pad_flash_mosi_oen_frc	20:20	RW	Pad output enable force control 1: "oen_reg" value will overwrite the oen control logic of normal function 0: oen is controlled by normal function	1'h0
pad_flash_mosi_oen_reg	19:19	RW	Pad oen value when oen_frc==1	1'h0
pad_flash_mosi_pull_frc	18:18	RW	Pad Pullup/Pulldown force control 1: pull_up/pull_dn value below will overwrite the Pullup/Pulldown of normal function 0: Pullup/Pulldown is controlled by normal function	1'h0
pad_flash_mosi_pull_up	17:17	RW	Pad pullup enable control when pull_frc ==1 0: pullup is disabled 1:pullup is enabled	1'h0
pad_flash_mosi_pull_dn	16:16	RW	Pad pulldown eanble control when pull_frc==1 0: pulldown is disabled 1: pulldown is enabled	1'h0
Reserved	15:9	RO	Not used	7'h0
pad_flash_mosi_ana_sel	8:5	RW	Pad analog function selection, only used on Analog/Digital mixed I/Os	4'h0
pad_flash_mosi_fsel	4:0	RW	Pad function selection	5'h0