

**6001 User Manual**

Security: Medium

Version 0.3 2022.04.11

# 声明

本手册由聆思科技版权所有，未经许可，任何单位和个人都不得以电子的、机械的、磁性的、光学的、化学的、手工的等形式复制、传播、转录和保存该出版物，或翻译成其它语言版本。一经发现，将追究其法律责任。

聆思科技保证本手册提供信息的准确性和可靠性。聆思科技保留更改本手册的权利，如有修改，恕不相告。请在订购时联系我们以获得产品最新信息。

对任何用户使用我们产品时侵犯第三方版权或其它权利的行为聆思科技概不负责。另外，在聆思科技未明确表示产品有该项用途时，对于产品使用在极端条件下导致一些失灵或损毁而造成的损失概不负责。

# 版本记录

|  |  |  |
| --- | --- | --- |
| Version | Description | Owner |
| 0.1 | Create new version |  |
| 0.2 | 添加除了购买的IP之外的自研模块的内容 |  |
| 0.3 | 更新量产芯片的寄存器配置内容 |  |

# 目录

# CMN\_IOMUX

## Introduction

选择GPIO的复用功能

## 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

projects\Venus\ASIC\Digital\IPs\IOMUX\Venus\_IO\_New.xlsx

## Function Description

## Basic Block Diagram

## CMN\_IOMUX Register

### PAD\_GPIOA

Offset: 000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 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 |
| 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 |

### PAD\_GPIOB

Offset: 080

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 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 |
| 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 |

### PAD\_FLASH\_WP\_N

Offset: 0C0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:26 | RO | Not used | 6'h0 |
| pad\_flash\_wp\_n\_drv | 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 |
| 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 |

### PAD\_FLASH\_MISO

Offset: 0C4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 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 |
| 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 |
| 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 |

### PAD\_FLASH\_CS\_N

Offset: 0C8

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 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 |

### PAD\_FLASH\_HOLD\_N

Offset: 0CC

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 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 |
| 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 |

### PAD\_FLASH\_CLK

Offset: 0D0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 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 |
| pad\_flash\_clk\_fsel | 4:0 | RW | Pad function selection | 5'h0 |

### PAD\_FLASH\_MOSI

Offset: 0D4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 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 |

# CMN\_SYSCTRL

## Introduction

系统控制单元配置了描述整个系统时钟频率以及每个模块的时钟选择。

## Main Features

系统的外部时钟源只有两个，外部32768Hz是可选的，内部有低频RC晶体32KHz

### 外部高频时钟

24MHz晶体是必须的

### 外部低频时钟

32768Hz是可选时钟，如果没有这个时钟，可以选择内部RC32K做为低频时钟

### 内部低频时钟

内部低频RC晶体32KHz

### 两个PLL模块

芯片内共有两个PLL模块，分别对应SYSPLL和AUDPLL，这两个PLL的最大差

异是：

(1)SYSPLL可以产生USB所需要的60MHz时钟

(2)AUDPLL可以生成音频Codec所需要的时钟22.05M，用来支持特殊的采样率入44.1K的需求

这两个PLL产生的其他可用于SYSTEM，PSRAM，Flash的时钟都可以针对对应的模块来选择。

### CBUTTON的时钟

CBUTTON模块的时钟是由其内部产生的，和外部的时钟源无关，用户不用去设置。

## Function Description

时钟区域主要分为下面的几个大的模块，在下一个章节中用不同的颜色来表示

### AP与AP对应的外设

框图中的蓝色表示，其中对应的模块有：



### CP与CP对应的外设

框图中的粉红色表示，其中对应的模块有：



### CMN双核共用对应的外设

框图中的粉红色表示，其中对应的模块有：



### Always On区域对应的外设

框图中的绿色表示，其中对应的模块有：



## Basic Block Diagram



## CMN\_SYSCTRL Register

### AON\_WAIT\_XTAL

Offset: 000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| chip\_id | 31:0 | RW | Chip ID. Default to 0x6001\_A0B0 | 32'h6001a0b0 |

### TEST\_CTRL

Offset: 004

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RO |  | 16'h0 |
| fast\_capture\_mode | 15:15 | RW | fast\_capture\_mode | 1'h1 |
| edt\_bypass | 14:14 | RW | edt\_bypass control | 1'h0 |
| dbg\_out\_sel | 13:9 | RW | select dbg\_out source | 5'h0 |
| dbg\_out\_en | 8:8 | RW | enable debug data output, active high | 1'h0 |
| dbg\_clk\_sel | 7:3 | RW | select dbg\_clk source | 5'h0 |
| dbg\_clk\_en | 2:2 | RW | enable debug clock output, active high | 1'h0 |
| flashc\_direct\_io\_mode\_frc | 1:1 | RW | force flashc to work on direct io mode, the flash I/O interface is exposed to package pins directly in this mode 1 : use register to control this mode 0: use hardware logic to control this mode | 1'h0 |
| flashc\_direct\_io\_mode\_val | 0:0 | RW | force flashc to work on direct io mode, the flash I/O interface is exposed to package pins directly in this mode, the value is only valid when the \*\_frc is set to 1 1 : direct io mode 0: normal mode | 1'h0 |

### CP\_CTRL1

Offset: 008

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:3 | RO |  | 29'h0 |
| cp\_stat\_vector\_sel | 2:2 | RW | Set the HiFi4 core StatVectorSel input. Must be set and held before HiFi4 de-reset. | 1'h0 |
| sel\_sw\_cp\_reset\_range | 1:1 | RW | Select the reset effect range of CP SW reset which is a specific keyword to write and generate 0: reset the CP core Only (default) 1: reset the CP subsystem completely | 1'h0 |
| cp\_hifi4\_runstall | 0:0 | RW | Write 1 to generate a SW controlled run stall of CP HiFi4 Core 0: the HiFi4 core is in normal running 1: the HiFi4 core is stalled. It needs SW write 0 to re-continue the HiFi4 core running. | 1'h0 |

### SW\_RESET2

Offset: 00C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:22 | RO |  | 10'h0 |
| vic\_reset | 21:21 | W1P | Write 1 to reset vic clock domain | 1'h0 |
| gpadc\_reset | 20:20 | W1P | Write 1 to generate a SW reset to GPADC digital part | 1'h0 |
| cp\_dma\_reset | 19:19 | W1P |  | 1'h0 |
| ap\_dma\_reset | 18:18 | W1P |  | 1'h0 |
| flash\_ctrl\_reset | 17:17 | W1P |  | 1'h0 |
| psram\_ctrl\_reset | 16:16 | W1P |  | 1'h0 |
| cp\_codec\_reset | 15:15 | RW | Write 1 to generate a SW reset to CP Codec digital part This bit need SW to write 0 to de-reset. | 1'h0 |
| gpio\_a\_reset | 14:14 | W1P | Write 1 to generate a SW reset to GPIO-A control | 1'h0 |
| gpio\_b\_reset | 13:13 | W1P | Write 1 to generate a SW reset to GPIO-B control | 1'h0 |
| crypto\_reset | 12:12 | W1P |  | 1'h0 |
| sdio\_reset | 11:11 | W1P |  | 1'h0 |
| gpt\_reset | 10:10 | W1P | Write 1 to generate a SW reset to general purpose timer | 1'h0 |
| usbc\_reset | 9:9 | W1P | Write 1 to generate a SW reset to USB controller | 1'h0 |
| ir\_reset | 8:8 | W1P | Write 1 to generate a SW reset to IR controller | 1'h0 |
| i2c1\_reset | 7:7 | W1P | Write 1 to generate a SW reset to I2C1 | 1'h0 |
| i2c0\_reset | 6:6 | W1P | Write 1 to generate a SW reset to I2C0 | 1'h0 |
| spi1\_reset | 5:5 | W1P | Write 1 to generate a SW reset to SPI1 | 1'h0 |
| spi0\_reset | 4:4 | W1P | Write 1 to generate a SW reset to SPI0 | 1'h0 |
| uart3\_reset | 3:3 | W1P | Write 1 to generate a SW reset to UART3 | 1'h0 |
| uart2\_reset | 2:2 | W1P | Write 1 to generate a SW reset to UART2 | 1'h0 |
| uart1\_reset | 1:1 | W1P | Write 1 to generate a SW reset to UART1 | 1'h0 |
| uart0\_reset | 0:0 | W1P | Write 1 to generate a SW reset to UART0 | 1'h0 |

### AP\_CTRL1

Offset: 014

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ap\_dma\_hs\_sel | 31:16 | RW | Each AP DMAC hardware handsharking interface has two configrations: for ap\_dma\_hs\_sel[i] == 0 : CONFIG0  ap\_dma\_hs\_sel[i] == 1 : CONFIG1 | 16'h0 |
| iomux\_reg\_clk\_frc\_on | 15:15 | RW | Added for common IOMUX register file clock force on | 1'h0 |
| gpadc\_reg\_clk\_frc\_on | 14:14 | RW | Added for AP peripheral clock force on | 1'h0 |
| gpt\_reg\_clk\_frc\_on | 13:13 | RW | Added for AP peripheral clock force on | 1'h0 |
| ir\_reg\_clk\_frc\_on | 12:12 | RW | Added for AP peripheral clock force on | 1'h0 |
| sel\_sw\_ap\_reset\_range | 11:11 | RW | Select the reset effect range of AP SW reset which is a specific keyword to write and generate 0: reset the AP core Only (default) 1: reset the AP subsystem completely | 1'h0 |
| ap\_ena\_lockup\_rst\_apsub | 10:10 | RW | Enable the reset to AP Subystem when lockup is generated | 1'h0 |
| ap\_ena\_lockup\_rst\_apcore | 9:9 | RW | Enable the reset to AP Star Core when lockup is generated | 1'h0 |
| ap\_ena\_resetreq | 8:8 | RW | Enable the reset requirement to AP subsystem from AP Star Core 0: reset the AP core Only (default) 1: reset the AP subsystem completely | 1'h0 |
| ap\_ram\_sel | 7:0 | RW | AP share memory banks selection, 64KB memory for one bank, each bank can be configured to be accessed by AHB-C or AHB-S: for ap\_ram\_sel[i] == 0: the corresponding memory bank can be accessed by AHB-C  ap\_ram\_sel[i] == 1: the corresponding memory bank can be accessed by AHB-S | 8'h0 |

### AP\_CTRL2

Offset: 018

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| usbphy\_rsv | 31:26 | RW | reserved for usbphy | 6'h0 |
| star\_cfgnsstcalib | 25:0 | RW | [25] NOREF 0: reference clock is implemented  1: reference clock is not implemented [24] SKEW 0: can guarantee an exact multiple of 10ms  1: cannot guarantee an exact multiple of 10ms [23:0] TENMS: Provides an integer value to compute a 10ms, 100Hz, delay from either the external reference clock, or CLKIN if the reference clock is not implemented. | 26'h1d4c0 |

### AP\_CTRL3

Offset: 01C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:31 | RO |  | 1'h0 |
| usbc\_st\_chrgvbus | 30:30 | RO |  | 1'h0 |
| usbc\_st\_dischrgvbus | 29:29 | RO |  | 1'h0 |
| usbc\_st\_drvvbus | 28:28 | RO |  | 1'h0 |
| usbc\_st\_speed | 27:27 | RO |  | 1'h1 |
| usbc\_st\_suspend | 26:26 | RO |  | 1'h0 |
| usbphy\_gpio\_en | 25:25 | RW |  | 1'h0 |
| usbphy\_gpio\_ie | 24:24 | RW |  | 1'h1 |
| usbphy\_didif\_bypss | 23:23 | RW |  | 1'h0 |
| usbphy\_ptrm | 22:21 | RW |  | 2'h0 |
| usbphy\_ntrm | 20:19 | RW |  | 2'h0 |
| usbphy\_tmdrv | 18:16 | RW |  | 3'h0 |
| usbphy\_speed\_frc | 15:15 | RW |  | 1'h0 |
| usbphy\_speed\_val | 14:14 | RW |  | 1'h0 |
| usbphy\_suspend\_frc | 13:13 | RW |  | 1'h0 |
| usbphy\_suspend\_val | 12:12 | RW |  | 1'h0 |
| usbphy\_mode | 11:11 | RW |  | 1'h1 |
| usbphy\_up\_p\_en\_frc | 10:10 | RW |  | 1'h0 |
| usbphy\_up\_p\_en\_val | 9:9 | RW |  | 1'h0 |
| usbphy\_dn\_p\_en\_frc | 8:8 | RW |  | 1'h0 |
| usbphy\_dn\_p\_en\_val | 7:7 | RW |  | 1'h0 |
| usbphy\_up\_m\_en | 6:6 | RW |  | 1'h0 |
| usbphy\_dn\_m\_en | 5:5 | RW |  | 1'h1 |
| usbc\_vbusvalid | 4:4 | RW |  | 1'h1 |
| usbc\_avalid | 3:3 | RW |  | 1'h1 |
| usbc\_vbuslo | 2:2 | RW |  | 1'h0 |
| usbc\_cfg\_tm1 | 1:1 | RW |  | 1'h0 |
| usbc\_cfg\_cid | 0:0 | RW |  | 1'h1 |

### AP\_CTRL4

Offset: 020

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| star\_initnsvtor\_val | 31:7 | RW | the force value of star\_initnsvtor, only take effect when the star\_initnsvtor\_frc is set to 7'h25. | 25'h0 |
| star\_initnsvtor\_frc | 6:0 | RW | when this field is set to 7'h25, the star\_initnsvtor\_val is set by the "star\_initnsvtor\_val" | 7'h0 |

### CORE\_CLK\_CFG

Offset: 030

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:31 | RO |  | 1'h0 |
| sel\_sys\_ap\_peri | 30:29 | RW | Select the ap\_peri\_clk 00: (1200/10)M 01: (1200/11)M 10: (1200/12)M 11: (1200/13)M | 2'h2 |
| dis\_sys\_pll\_unlock | 28:28 | RW | Disable SYSPLL unlock fuction 0: do not disable  1: disable | 1'h0 |
| dis\_aud\_pll\_unlock | 27:27 | RW | Disable AUDPLL unlock fuction 0: do not disable  1: disable | 1'h0 |
| div\_ap\_root\_ld | 26:26 | RW | At rising edge to load config | 1'h0 |
| div\_ap\_root\_n | 25:21 | RW | Numerator N of N/M divider of AP sytem clock frequency | 5'h1 |
| div\_ap\_root\_m | 20:16 | RW | Denominator M of N/M divider of AP sytem clock frequency Ex: N=1,M=2 mean AP HCLK is 1/2 of System Clock Frequency. Note: N M should not be 0, M >=N.M=value+1 | 5'h0 |
| Reserved | 15:15 | RO |  | 1'h0 |
| div\_sys\_root\_ld | 14:14 | RW | At rising edge to load config | 1'h0 |
| div\_sys\_root\_n | 13:11 | RW | Numerator N of N/M divider of system root clock frequency | 3'h1 |
| div\_sys\_root\_m | 10:8 | RW | Denominator M of N/M divider of system root clock frequency Ex: N=1,M=2 mean System Clock is 1/2 of Root Clock Frequency(From PLL). Note: N M should not be 0, M >=N. M=value+1 | 3'h0 |
| Reserved | 7:7 | RO |  | 1'h0 |
| ena\_npu\_clk | 6:6 | RW | 0: the clock source is gated 1: enable clock | 1'h1 |
| ena\_cp\_core\_clk | 5:5 | RW | 0: the clock source is gated 1: enable clock | 1'h1 |
| ena\_ap\_core\_clk | 4:4 | RW | 0: the clock source is gated 1: enable clock | 1'h1 |
| Reserved | 3:2 | RO |  | 2'h0 |
| sel\_sys\_root\_clk | 1:0 | RW | Select the system root clock source 00: XTAL 24MHz 01: If SYSPLL is locked or SYSPLL unlock function is disabled, use USBPLL high-speed output 10: If AUDPLL is locked or AUDPLL unlock function is disabled, use AUDPLL high-speed output 11: Reserved | 2'h0 |

### BUS\_CLK\_CFG

Offset: 034

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:21 | RO |  | 11'h0 |
| div\_cmn\_cfg\_pclk\_ld | 20:20 | RW | At rising edge to load config | 1'h0 |
| div\_cmn\_cfg\_pclk\_m | 19:16 | RW | Denominator M of 1/M divider of common PCLK Ex: M=2 mean CMN PCLK is 1/3 of AP PCLK Clock Frequency. Note: M should not be 0.M=value+1 | 4'h3 |
| Reserved | 15:15 | RO |  | 1'h0 |
| div\_ap\_peri\_pclk\_ld | 14:14 | RW | At rising edge to load config | 1'h0 |
| div\_ap\_peri\_pclk\_n | 13:11 | RW | Numerator N of N/M divider of AP PCLK | 3'h1 |
| div\_ap\_peri\_pclk\_m | 10:8 | RW | Denominator M of N/M divider of AP PCLK Ex: N=1,M=2 mean AP PCLK is 1/3 of AP HCLK Clock Frequency. Note: N M should not be 0, M >=N.M=value+1 | 3'h0 |
| Reserved | 7:7 | RO |  | 1'h0 |
| div\_cp\_pclk\_ld | 6:6 | RW | At rising edge to load config | 1'h0 |
| div\_cp\_pclk\_n | 5:3 | RW | Numerator N of N/M divider of CP PCLK | 3'h1 |
| div\_cp\_pclk\_m | 2:0 | RW | Denominator M of N/M divider of CP PCLK Ex: N=1,M=2 mean CP PCLK is 1/3 of System Clock Frequency. Note: N M should not be 0, M >=N.M=value+1 | 3'h0 |

### NVM\_CLK\_CFG

Offset: 038

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO |  | 8'h0 |
| div\_psram\_clk\_ld | 23:23 | RW |  | 1'h0 |
| div\_psram\_clk | 22:20 | RW | PLL clock divider for PSRAM clock Non-clock-gate based divider. 00: divide 1 01: divide 2 10: divide 3 11: divide 4 etc. | 3'h0 |
| ena\_psram\_clk | 19:19 | RW | 0: the clock source is gated 1: enable clock | 1'h1 |
| Reserved | 18:18 | RO |  | 1'h0 |
| sel\_psram\_clk | 17:16 | RW | Select the PSRAM clock 00: XTAL 24MHz 01: If SYSPLL is locked, use divided USBPLL high-speed output 10: If AUDPLL is locked, use divided AUDPLL high-speed output 11: Reserved | 2'h0 |
| Reserved | 15:8 | RO |  | 8'h0 |
| div\_flash\_clk\_ld | 7:7 | RW |  | 1'h0 |
| Reserved | 6:6 | RO |  | 1'h0 |
| div\_flash\_clk | 5:4 | RW | PLL clock divider for Flash clock Non-clock-gate based divider. 00: divide 1 01: divide 2 10: divide 3 11: divide 4 | 2'h0 |
| ena\_flash\_clk | 3:3 | RW | 0: the clock source is gated 1: enable clock | 1'h1 |
| inv\_flash\_clk | 2:2 | RW | 0: no inversion 1: inverte the Flash clock | 1'h0 |
| sel\_flash\_clk | 1:0 | RW | Select the Flash clock 00: XTAL 24MHz 01: If SYSPLL is locked, use divided USBPLL high-speed output 10: If AUDPLL is locked, use divided AUDPLL high-speed output 11: Reserved | 2'h0 |

### UART\_SDIO\_CLK

Offset: 03C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:31 | RO |  | 1'h0 |
| div\_uart0\_clk\_ld | 30:30 | RW | At rising edge to load config | 1'h0 |
| div\_uart0\_clk\_n | 29:25 | RW | Numerator N of N/M divider | 5'h1 |
| div\_uart0\_clk\_m | 24:20 | RW | Denominator M of N/M divider Ex: N=1,M=2 mean UART0 SCLK is 1/2 of USBPLL/24MHz Xtal Clock Frequency. Note: N M should not be 0, M >=N.M=value+1 | 5'h0 |
| ena\_uart0\_clk | 19:19 | RW | Enable SDIO function clock | 1'h0 |
| Reserved | 18:17 | RO |  | 2'h0 |
| sel\_uart0\_clk | 16:16 | RW | Select the UART0 function clock source Default=0 0: XTAL input 1: SYSPLL USB clock | 1'h0 |
| Reserved | 15:11 | RO |  | 5'h0 |
| div\_sdio\_clk\_ld | 10:10 | RW | At rising edge to load config | 1'h0 |
| div\_sdio\_clk\_n | 9:7 | RW | Numerator N of N/M divider | 3'h1 |
| div\_sdio\_clk\_m | 6:4 | RW | Denominator M of N/M divider Ex: N=1,M=2 mean SDIO CLK2x is 1/2 of USBPLL/24MHz Xtal Clock Frequency. Note: N M should not be 0, M >=N.M=value+1 | 3'h0 |
| ena\_sdio\_clk | 3:3 | RW | Enable SDIO function clock | 1'h0 |
| Reserved | 2:1 | RO |  | 2'h0 |
| sel\_sdio\_clk | 0:0 | RW | Select the SDIO function clock source Default=0 0: XTAL input 1: SYSPLL USB clock | 1'h0 |

### UART\_CLK\_CFG

Offset: 040

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:31 | RO |  | 1'h0 |
| div\_uart2\_clk\_ld | 30:30 | RW | At rising edge to load config | 1'h0 |
| div\_uart2\_clk\_n | 29:25 | RW | Numerator N of N/M divider | 5'h1 |
| div\_uart2\_clk\_m | 24:20 | RW | Denominator M of N/M divider Ex: N=1,M=2 mean UART2 SCLK is 1/2 of USBPLL/24MHz Xtal Clock Frequency. Note: N M should not be 0, M >=N.M=value+1 | 5'h0 |
| ena\_uart2\_clk | 19:19 | RW | Enable UART2 function clock | 1'h0 |
| Reserved | 18:17 | RO |  | 2'h0 |
| sel\_uart2\_clk | 16:16 | RW | Select the UART2 function clock source Default=0 0: XTAL input 1: SYSPLL USB clock | 1'h0 |
| Reserved | 15:15 | RO |  | 1'h0 |
| div\_uart1\_clk\_ld | 14:14 | RW | At rising edge to load config | 1'h0 |
| div\_uart1\_clk\_n | 13:9 | RW | Numerator N of N/M divider | 5'h1 |
| div\_uart1\_clk\_m | 8:4 | RW | Denominator M of N/M divider Ex: N=1,M=2 mean UART1 SCLK is 1/2 of USBPLL/24MHz Xtal Clock Frequency. Note: N M should not be 0, M >=N.M=value+1 | 5'h0 |
| ena\_uart1\_clk | 3:3 | RW | Enable UART1 function clock | 1'h0 |
| Reserved | 2:1 | RO |  | 2'h0 |
| sel\_uart1\_clk | 0:0 | RW | Select the UART1 function clock source Default=0 0: XTAL input 1: SYSPLL USB clock | 1'h0 |

### SPI\_CLK\_CFG

Offset: 044

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:27 | RO |  | 5'h0 |
| div\_spi1\_clk\_ld | 26:26 | RW | At rising edge to load config | 1'h0 |
| div\_spi1\_clk\_n | 25:23 | RW | Numerator N of N/M divider | 3'h1 |
| div\_spi1\_clk\_m | 22:20 | RW | Denominator M of N/M divider Ex: N=1,M=2 mean SPI1 SCLK is 1/2 of USBPLL/24MHz Xtal Clock Frequency. Note: N M should not be 0, M >=N.M=value+1 | 3'h0 |
| ena\_spi1\_clk | 19:19 | RW | Enable SPI1 function clock | 1'h0 |
| Reserved | 18:17 | RO |  | 2'h0 |
| sel\_spi1\_clk | 16:16 | RW | Select the SPI1 function clock source Default=0 0: XTAL input 1: SYSPLL USB clock | 1'h0 |
| Reserved | 15:11 | RO |  | 5'h0 |
| div\_spi0\_clk\_ld | 10:10 | RW | At rising edge to load config | 1'h0 |
| div\_spi0\_clk\_n | 9:7 | RW | Numerator N of N/M divider | 3'h1 |
| div\_spi0\_clk\_m | 6:4 | RW | Denominator M of N/M divider Ex: N=1,M=2 mean SPI0 SCLK is 1/2 of USBPLL/24MHz Xtal Clock Frequency. Note: N M should not be 0, M >=N.M=value+1 | 3'h0 |
| ena\_spi0\_clk | 3:3 | RW | Enable SPI0 function clock | 1'h0 |
| Reserved | 2:1 | RO |  | 2'h0 |
| sel\_spi0\_clk | 0:0 | RW | Select the SPI0 function clock source Default=0 0: XTAL input 1: SYSPLL USB clock | 1'h0 |

### MISC\_CLK\_CFG1

Offset: 048

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| div\_uart3\_clk\_ld | 31:31 | RW | At rising edge to load config | 1'h0 |
| div\_uart3\_clk\_n | 30:27 | RW | Numerator N of N/M divider | 4'h1 |
| div\_uart3\_clk\_m | 26:22 | RW | Denominator M of N/M divider Ex: N=1,M=2 mean UART3 SCLK is 1/2 of USBPLL/24MHz Xtal Clock Frequency. Note: N M should not be 0, M >=N.M=value+1 | 5'h0 |
| ena\_uart3\_clk | 21:21 | RW | Enable UART3 function clock | 1'h0 |
| sel\_uart3\_clk | 20:20 | RW | Select the UART3 function clock source Default=0 0: XTAL input 1: SYSPLL USB clock | 1'h0 |
| ena\_gpiob\_extclk | 19:19 | RW |  | 1'h0 |
| ena\_gpioa\_extclk | 18:18 | RW |  | 1'h0 |
| ena\_gpt\_clkt0 | 17:17 | RW |  | 1'h0 |
| ena\_gpt\_clks | 16:16 | RW |  | 1'h0 |
| div\_gpiob\_extclk\_ld | 15:15 | RW | At rising edge to load config | 1'h0 |
| div\_gpiob\_extclk | 14:12 | RW | Divider for GPIO-B clock For the requirement of = 1/2 ap\_peri\_pclk: XTAL divided by M Ex: M=2 mean GPIO BCLK is 1/4(M\*2=4) of XTAL Clock Frequency. Note: M=value+1 | 3'h1 |
| div\_gpioa\_extclk\_ld | 11:11 | RW | At rising edge to load config | 1'h0 |
| div\_gpioa\_extclk | 10:8 | RW | Divider for GPIO-A clock For the requirement of = 1/2 ap\_peri\_pclk: XTAL divided by M  Ex: M=2 mean GPIO ACLK is 1/4(M\*2=4) of XTAL Clock Frequency. Note: M=value+1 | 3'h1 |
| div\_gpt\_clkt0\_ld | 7:7 | RW | At rising edge to load config | 1'h0 |
| div\_gpt\_clkt0 | 6:4 | RW | General Purpose Timer clk\_t0 divider For the requirement of = 1/2 ap\_peri\_pclk: XTAL divided by M Ex: M=2 mean GPT T0 CLK is 1/2 of XTAL Clock Frequency. Note: M=value+1 | 3'h0 |
| div\_gpt\_clks\_ld | 3:3 | RW | At rising edge to load config | 1'h0 |
| div\_gpt\_clks | 2:0 | RW | General Purpose Timer clk\_t0 divider For the requirement of = 1/2 ap\_peri\_pclk: XTAL divided by M  Ex: M=2 mean GPT SCLK is 1/3 of XTAL Clock Frequency. Note: M=value+1 | 3'h0 |

### MISC\_CLK\_CFG2

Offset: 04C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:15 | RO |  | 17'h0 |
| codec\_reg\_clk\_frc\_on | 14:14 | RW | Force the CODEC register file clock on | 1'h0 |
| codec\_reg\_frc\_aud\_reset | 13:13 | RW | Force reset the CODEC audio modules 0: normal, de-reset 1: reset | 1'h0 |
| sel\_slow\_clock\_source | 12:12 | RW | Select the AP WatchDog clock source 0. low-power 32kHz 1. XTAL 24MHz divided to 32K | 1'h1 |
| ena\_ir\_clk | 11:11 | RW | 0: the clock source is gated 1: enable clock | 1'h0 |
| sel\_ir\_clk | 10:10 | RW | Source choice: 0. low-power 32kHz 1. XTAL 24MHz divided to 32K | 1'h1 |
| ena\_cp\_uart\_clk | 9:9 | RW | 0: the clock source is gated 1: enable clock | 1'h0 |
| ena\_apc\_pclk | 8:8 | RW | 0: the clock source is gated 1: enable clock | 1'h0 |
| ena\_codec\_clk\_dac | 7:7 | RW | 0: the clock source is gated 1: enable clock | 1'h0 |
| sel\_codec\_clk\_dac | 6:6 | RW | Default=1 0: XTAL 24MHz input 1: AUDPLL audio clock source | 1'h0 |
| ena\_codec\_clk\_adc | 5:5 | RW | 0: the clock source is gated 1: enable clock | 1'h0 |
| sel\_codec\_clk\_adc | 4:4 | RW | Default=1 0: XTAL 24MHz input 1: AUDPLL audio clock source | 1'h0 |
| ena\_xtal\_div\_32k | 3:3 | RW | Enable XTAL 24MHz divided to 32kHz 0: disable 1: enable divider (default) | 1'h1 |
| ena\_usbc\_clk | 2:2 | RW | 0: the clock source is gated 1: enable clock | 1'h0 |
| ena\_i2c1\_clk | 1:1 | RW | 0: the clock source is gated 1: enable clock | 1'h0 |
| ena\_i2c0\_clk | 0:0 | RW | 0: the clock source is gated 1: enable clock | 1'h0 |

### SW\_RESET\_AP

Offset: 050

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO |  | 31'h0 |
| ap\_reset | 0:0 | W1P | Write 0xCAFE000A to generate a SW reset to ARM Star Core or AP subsystem The reset is a high active pulse, it is clear automatically. | 1'h0 |

### SW\_RESET\_CP

Offset: 054

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO |  | 31'h0 |
| cp\_reset | 0:0 | RW | Write 0xCAFE000B to set CP to reset state. Write 0xCAFE100B to set CP to de-reset state. | 1'h1 |

### SW\_RESET\_NPU

Offset: 058

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO |  | 31'h0 |
| cp\_npu\_reset | 0:0 | RW | Write 0xCAFE000C to generate a SW reset to NPU Core The reset is a high active level, it needs SW write 0xCAFE000C again to clear. If the reset is high active, SW can read-back with 1 from this register. | 1'h0 |

### CP\_CTRL2

Offset: 05C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| cp\_boot\_addr | 31:0 | RW |  | 32'h60000000 |

### AP\_CRYPTO\_CFG0

Offset: 060

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:4 | RO |  | 28'h0 |
| crypto\_zeroize\_key\_text\_req | 3:3 | RW | active high | 1'h0 |
| crypto\_sec\_fail\_found\_to\_crypto | 2:2 | RW | active high | 1'h0 |
| crypto\_dummy\_key\_used | 1:1 | RW | active high | 1'h0 |
| crypto\_en | 0:0 | RW | active high | 1'h0 |

### CP2AP\_INTE

Offset: 064

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| cp2ap\_int\_ena | 31:0 | RW | 0: disable 1: enable the CP-to-AP interrupt status | 32'h0 |

### CP2AP\_INTS

Offset: 068

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| cp2ap\_int\_status | 31:0 | W1C | SW write 1 to clear the CP-to-AP interrupt status | 32'h0 |

### AP\_CRYPTO\_CFG1

Offset: 070

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| crypto\_key\_0\_31 | 31:0 | RW | active high | 32'h31313131 |

### AP\_CRYPTO\_CFG2

Offset: 074

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| crypto\_key\_32\_63 | 31:0 | RW | active high | 32'h31313131 |

### AP\_CRYPTO\_CFG3

Offset: 078

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| crypto\_key\_64\_95 | 31:0 | RW | active high | 32'h31313131 |

### AP\_CRYPTO\_CFG4

Offset: 07C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| crypto\_key\_96\_127 | 31:0 | RW | active high | 32'h31313131 |

# CMN\_MAILBOX

## Introduction

MailBox是一个用于多核间通信的模块，可通过对MailBox寄存器配置，在多个核之间发送中断并传输数据，完成多核之间的通信和协作。

## Main Features

* AMBA APB4.0 Slave接口
* 两个中断输出信号
* 每个中断输出分别具有16个优先级一样的中断可配置
* 有两组分别4个总共8个32bit数据寄存器
* 具有Lock功能，可将数据寄存器Lock

## Function Description

该节描述了如何使用Mailbox中断发送和清除，以及中断的屏蔽和数据锁存的工作原理。该章节出现的带“\_x”的信号和带“N”的寄存器描述均表示（0-15）共16个中断对应的序号。

### AP-to-CP Interrupt

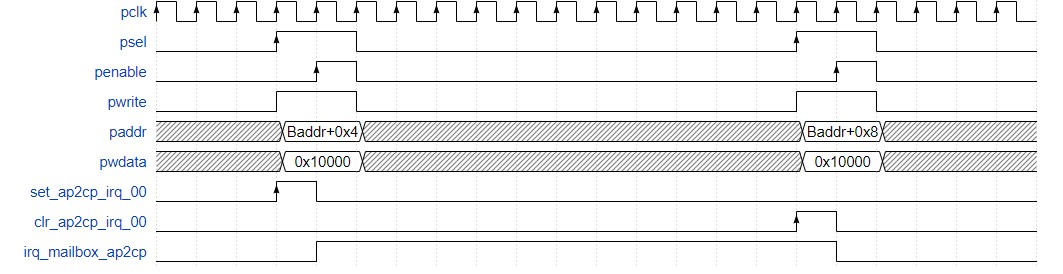
AP-to-CP中断由AP侧的Core发起，AP Core通过写AP\_MAILBOX\_CTRL寄存器[31:16]第n位发送对应序号（n-16）的中断，发送中断之前需要确保对应的中断没有被屏蔽。如果需要携带信息，须提前往AP\_DATA寄存器写入数值。待CP侧接收到中断之后，从AP\_DATA寄存器读取数据，然后查询AP\_MAILBOX\_IRQ获取mailbox内的中断号，并往对应的位写1对中断进行清除。AP-to-CP中断时序如下图：

Figure 2 AP-to-CP中断的产生和清除

由上图可知，AP Core写AP\_MAILBOX\_CTRL. ap2cp\_set\_irq\_N，产生一个set\_ap2cp\_irq\_x的脉冲，下一个时钟周期产生irq\_mailbox\_ap2cp中断;CP Core接收到中断，通过读AP\_MAILBOX\_IRQ寄存器查询Mailbox内部中断号，往对应的位写1产生了一个clr\_ap2cp\_irq\_x的脉冲，irq\_mailbox\_ap2cp中断在该脉冲出现的下一个时钟周期被清除。

### CP-to-AP Interrupt

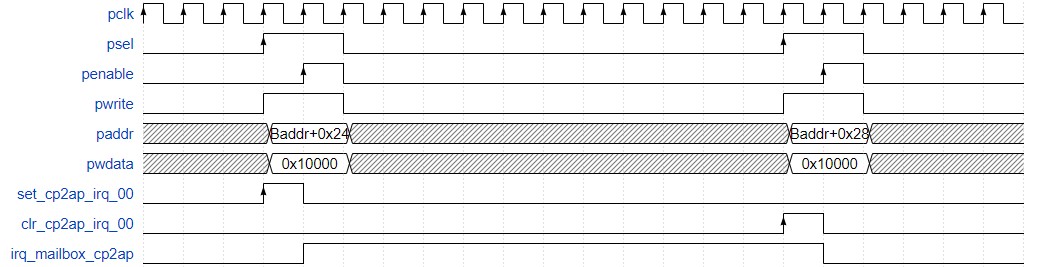
CP-to-AP中断由CP侧的Core发起，CP Core通过写CP\_MAILBOX\_CTRL寄存器[31:16]第n位发送对应序号（n-16）的中断，发送中断之前需确保对应的中断没有被屏蔽。如果需要携带信息，须提前往CP\_DATA（0-3）寄存器写入数值。待AP侧接收到中断之后，从CP\_DATA寄存器读取数据，然后查询CP\_MAILBOX\_IRQ获取mailbox内的中断号，并往对应的位写1对中断进行清除。CP-to-AP中断时序如下图：

Figure 3 CP-to-AP中断的产生和清除

由上图可知，CP Core写CP\_MAILBOX\_CTRL. cp2ap\_set\_irq\_N，产生一个set\_cp2ap\_irq\_x的脉冲，下一个时钟周期产生irq\_mailbox\_cp2ap中断;AP Core接收到中断，通过读CP\_MAILBOX\_IRQ寄存器查询Mailbox内部中断号，往对应的位写1产生了一个clr\_cp2ap\_irq\_x的脉冲，irq\_mailbox\_cp2ap中断在该脉冲出现的下一个时钟周期被清除。

### Mailbox interrupt mask

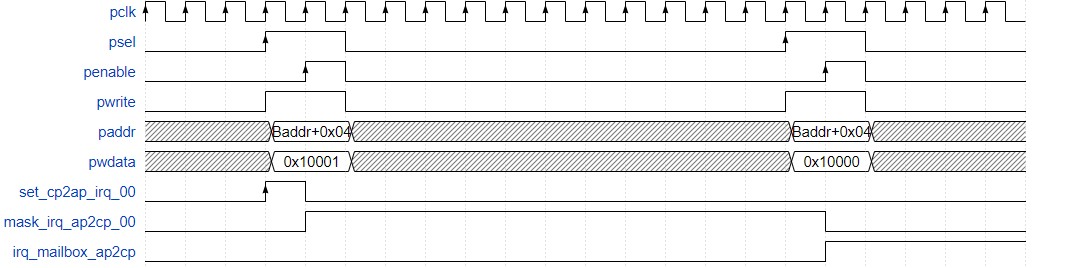
AP-to-CP中断和CP-to-AP中断均支持Mask，通过往AP\_MAILBOX\_CTRL寄存器[15:0]写1可以屏蔽AP-to-CP对应序号的中断；往CP\_MAILBOX\_CTRL寄存器[15:0]写1可以屏蔽CP-to-AP对应序号的中断。原理如下图所示：

Figure 4 Mailbox Interrupt Mask

由图1-4可知，当mask\_ap2cp\_irq\_x为高电平的时候，set\_ap2cp\_irq\_x脉冲到来并不能触发irq\_mailbox\_ap2cp。只有当mask\_ap2cp\_irq\_x为低的时候，set\_ap2cp\_irq\_x才能触发irq\_mailbox\_ap2cp。

### Mailbox lock

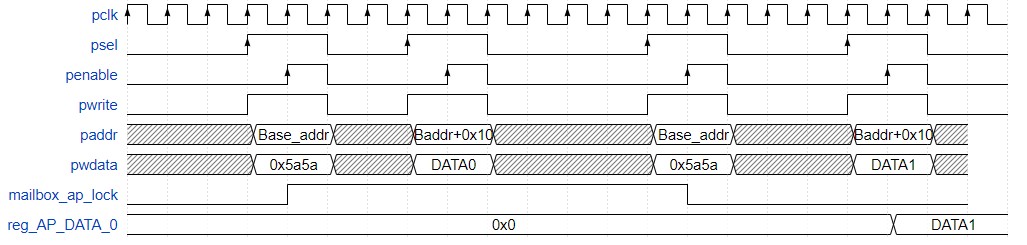
AP侧的Core可以通过对AP\_MAILBOX\_LOCK寄存器写‘h5A5A对AP侧的数据寄存器AP\_DATA（0-3）进行锁存。CP侧的Core可以通过对CP\_MAILBOX\_LOCK寄存器写’h5A5A对CP侧的数据寄存器CP\_DATA（0-3）进行锁存。AP\_MAILBOX\_LOCK和CP\_MAILBOX\_LOCK可以进行查询，为’h1的时候表明LOCK，为’h0的时候表明Un-LOCK。原理图如下所示：

Figure 5 Mailbox 数据寄存器Lock和unLock

由图1-5可知，往AP\_MAILBOX\_LOCK写入0x5a5a，AP\_DATA\_x寄存器被锁住，此时DATA0没有被成功写入reg\_AP\_DATA\_0寄存器中，当第二次往AP\_MAILBOX\_LOCK寄存器写入0x5a5a时，AP\_DATA\_x解锁，此时DATA1被成功写入reg\_AP\_DATA\_0中。

## Basic Block Diagram



## Programming Sequence

本章对Mailbox的软件使用方法作出说明。该流程中操作的寄存器细节可参考3.2章节。

### AP-to-CP Interrupt Programing Examples

AP-to-CP中断是由AP侧的Core通过写AP\_MAILBOX\_CTRL产生的中断，该中断传递到CP侧的Core，触发CP侧的中断，CP侧的Core执行完中断处理程序后，查询中断号，并将对应的中断清除，退出中断返回。



Figure 6 AP-to-CP Mailbox中断流程图

### CP-to-AP Interrupt Programing Examples

CP-to-AP中断是由CP侧的Core通过写CP\_MAILBOX\_CTRL产生的中断，该中断传递到AP侧的Core，触发AP侧的中断，AP侧的Core执行完中断处理程序后，查询中断号，并将对应的中断清除，退出中断返回。



Figure 7 CP-to-AP Mailbox中断流程图

### AP and CP Communication Programing Examples

以下流程图展示了以AP Core为发送端，CP Core为接收端，AP Core往CP Core发送一个Mailbox中断，并附带数据；CP Core接收到AP Core发送过来的Mailbox中断，读取数据，清除中断并返回一个数据，发送中断到AP Core的一个流程。该流程是一个比较典型的Mailbox应用场景，发送端不仅发送中断和数据，还需要接收中断和数据，确保接收端成功接收到中断和正确的数据。



Figure 8 AP and CP Communication Programing Examples

### Mailbox Lock Programing Examples

以下流程图展示了AP Core写完AP\_DATA\_x寄存器之后，再通过AP\_MAILBOX\_LOCK寄存器写入0x5A5A对其进行锁定的操作。锁定之后，AP\_DATA\_x内的值无法被修改，只能被读取。直到AP Core要写入新数据之前，再对AP\_MAILBOX\_LOCK寄存器写入0x5A5A对AP\_DATA\_x进行解锁。



Figure 9 Mailbox Lock Programing Examples

## CMN\_MAILBOX Register

### AP\_MAILBOX\_LOCK

Offset: 000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO | Not used | 31'h0 |
| ap\_mailbox\_lock | 0:0 | RW | Write 0x5A5A to enable or disable this bit. 0: The AP maibox register is writable; 1: The AP maibox register is locked; | 1'h0 |

### AP\_MAILBOX\_CTRL

Offset: 004

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ap2cp\_set\_irq\_15 | 31:31 | W1P | Write 1 to set AP-to-CP interrupt No.15 | 1'h0 |
| ap2cp\_set\_irq\_14 | 30:30 | W1P | Write 1 to set AP-to-CP interrupt No.14 | 1'h0 |
| ap2cp\_set\_irq\_13 | 29:29 | W1P | Write 1 to set AP-to-CP interrupt No.13 | 1'h0 |
| ap2cp\_set\_irq\_12 | 28:28 | W1P | Write 1 to set AP-to-CP interrupt No.12 | 1'h0 |
| ap2cp\_set\_irq\_11 | 27:27 | W1P | Write 1 to set AP-to-CP interrupt No.11 | 1'h0 |
| ap2cp\_set\_irq\_10 | 26:26 | W1P | Write 1 to set AP-to-CP interrupt No.10 | 1'h0 |
| ap2cp\_set\_irq\_09 | 25:25 | W1P | Write 1 to set AP-to-CP interrupt No.9 | 1'h0 |
| ap2cp\_set\_irq\_08 | 24:24 | W1P | Write 1 to set AP-to-CP interrupt No.8 | 1'h0 |
| ap2cp\_set\_irq\_07 | 23:23 | W1P | Write 1 to set AP-to-CP interrupt No.7 | 1'h0 |
| ap2cp\_set\_irq\_06 | 22:22 | W1P | Write 1 to set AP-to-CP interrupt No.6 | 1'h0 |
| ap2cp\_set\_irq\_05 | 21:21 | W1P | Write 1 to set AP-to-CP interrupt No.5 | 1'h0 |
| ap2cp\_set\_irq\_04 | 20:20 | W1P | Write 1 to set AP-to-CP interrupt No.4 | 1'h0 |
| ap2cp\_set\_irq\_03 | 19:19 | W1P | Write 1 to set AP-to-CP interrupt No.3 | 1'h0 |
| ap2cp\_set\_irq\_02 | 18:18 | W1P | Write 1 to set AP-to-CP interrupt No.2 | 1'h0 |
| ap2cp\_set\_irq\_01 | 17:17 | W1P | Write 1 to set AP-to-CP interrupt No.1 | 1'h0 |
| ap2cp\_set\_irq\_00 | 16:16 | W1P | For AP-to-CP interrupt No.0: Write 1 to set a IRQ If the IRQ is disable, the IRQ status will still be set to 1, but NO interrupt will give to AP | 1'h0 |
| ap2cp\_ena\_irq\_15 | 15:15 | RW | AP-to-CP interrupt No.15 enable, usage as below. | 1'h0 |
| ap2cp\_ena\_irq\_14 | 14:14 | RW | AP-to-CP interrupt No.14 enable, usage as below. | 1'h0 |
| ap2cp\_ena\_irq\_13 | 13:13 | RW | AP-to-CP interrupt No.13 enable, usage as below. | 1'h0 |
| ap2cp\_ena\_irq\_12 | 12:12 | RW | AP-to-CP interrupt No.12 enable, usage as below. | 1'h0 |
| ap2cp\_ena\_irq\_11 | 11:11 | RW | AP-to-CP interrupt No.11 enable, usage as below. | 1'h0 |
| ap2cp\_ena\_irq\_10 | 10:10 | RW | AP-to-CP interrupt No.10 enable, usage as below. | 1'h0 |
| ap2cp\_ena\_irq\_09 | 9:9 | RW | AP-to-CP interrupt No.9 enable, usage as below. | 1'h0 |
| ap2cp\_ena\_irq\_08 | 8:8 | RW | AP-to-CP interrupt No.8 enable, usage as below. | 1'h0 |
| ap2cp\_ena\_irq\_07 | 7:7 | RW | AP-to-CP interrupt No.7 enable, usage as below. | 1'h0 |
| ap2cp\_ena\_irq\_06 | 6:6 | RW | AP-to-CP interrupt No.6 enable, usage as below. | 1'h0 |
| ap2cp\_ena\_irq\_05 | 5:5 | RW | AP-to-CP interrupt No.5 enable, usage as below. | 1'h0 |
| ap2cp\_ena\_irq\_04 | 4:4 | RW | AP-to-CP interrupt No.4 enable, usage as below. | 1'h0 |
| ap2cp\_ena\_irq\_03 | 3:3 | RW | AP-to-CP interrupt No.3 enable, usage as below. | 1'h0 |
| ap2cp\_ena\_irq\_02 | 2:2 | RW | AP-to-CP interrupt No.2 enable, usage as below. | 1'h0 |
| ap2cp\_ena\_irq\_01 | 1:1 | RW | AP-to-CP interrupt No.1 enable, usage as below. | 1'h0 |
| ap2cp\_ena\_irq\_00 | 0:0 | RW | For AP-to-CP interrupt No.0: IRQ enable 0: Disable the IRQ; 1: Enable the IRQ from AP to AP; | 1'h0 |

### AP\_MAILBOX\_IRQ

Offset: 008

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RO | Not used | 16'h0 |
| ap2cp\_irq\_15 | 15:15 | W1C | AP-to-CP interrupt No.15 status: Write 1 to clear. | 1'h0 |
| ap2cp\_irq\_14 | 14:14 | W1C | AP-to-CP interrupt No.14 status: Write 1 to clear. | 1'h0 |
| ap2cp\_irq\_13 | 13:13 | W1C | AP-to-CP interrupt No.13 status: Write 1 to clear. | 1'h0 |
| ap2cp\_irq\_12 | 12:12 | W1C | AP-to-CP interrupt No.12 status: Write 1 to clear. | 1'h0 |
| ap2cp\_irq\_11 | 11:11 | W1C | AP-to-CP interrupt No.11 status: Write 1 to clear. | 1'h0 |
| ap2cp\_irq\_10 | 10:10 | W1C | AP-to-CP interrupt No.10 status: Write 1 to clear. | 1'h0 |
| ap2cp\_irq\_09 | 9:9 | W1C | AP-to-CP interrupt No.9 status: Write 1 to clear. | 1'h0 |
| ap2cp\_irq\_08 | 8:8 | W1C | AP-to-CP interrupt No.8 status: Write 1 to clear. | 1'h0 |
| ap2cp\_irq\_07 | 7:7 | W1C | AP-to-CP interrupt No.7 status: Write 1 to clear. | 1'h0 |
| ap2cp\_irq\_06 | 6:6 | W1C | AP-to-CP interrupt No.6 status: Write 1 to clear. | 1'h0 |
| ap2cp\_irq\_05 | 5:5 | W1C | AP-to-CP interrupt No.5 status: Write 1 to clear. | 1'h0 |
| ap2cp\_irq\_04 | 4:4 | W1C | AP-to-CP interrupt No.4 status: Write 1 to clear. | 1'h0 |
| ap2cp\_irq\_03 | 3:3 | W1C | AP-to-CP interrupt No.3 status: Write 1 to clear. | 1'h0 |
| ap2cp\_irq\_02 | 2:2 | W1C | AP-to-CP interrupt No.2 status: Write 1 to clear. | 1'h0 |
| ap2cp\_irq\_01 | 1:1 | W1C | AP-to-CP interrupt No.1 status: Write 1 to clear. | 1'h0 |
| ap2cp\_irq\_00 | 0:0 | W1C | AP-to-CP interrupt No.0 status: Write 1 to clear. | 1'h0 |

### Empty30824

Offset: 010

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ap\_mailbox\_word0 | 31:0 | RW | AP mailbox word 0: [31:0] | 32'h0 |

### Empty81485

Offset: 014

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ap\_mailbox\_word1 | 31:0 | RW | AP mailbox word 0: [63:32] | 32'h0 |

### Empty23550

Offset: 018

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ap\_mailbox\_word2 | 31:0 | RW | AP mailbox word 0: [95:64] | 32'h0 |

### Empty65712

Offset: 01C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ap\_mailbox\_word3 | 31:0 | RW | AP mailbox word 0: [127:96] | 32'h0 |

### CP\_MAILBOX\_LOCK

Offset: 020

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO | Not used | 31'h0 |
| CP\_mailbox\_lock | 0:0 | RW | Write 0x5A5A to enable or disable this bit. 0: The CP maibox register is writable; 1: The CP maibox register is locked; | 1'h0 |

### CP\_MAILBOX\_CTRL

Offset: 024

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| cp2ap\_set\_irq\_15 | 31:31 | W1P | Write 1 to set CP-to-AP interrupt No.15 | 1'h0 |
| cp2ap\_set\_irq\_14 | 30:30 | W1P | Write 1 to set CP-to-AP interrupt No.14 | 1'h0 |
| cp2ap\_set\_irq\_13 | 29:29 | W1P | Write 1 to set CP-to-AP interrupt No.13 | 1'h0 |
| cp2ap\_set\_irq\_12 | 28:28 | W1P | Write 1 to set CP-to-AP interrupt No.12 | 1'h0 |
| cp2ap\_set\_irq\_11 | 27:27 | W1P | Write 1 to set CP-to-AP interrupt No.11 | 1'h0 |
| cp2ap\_set\_irq\_10 | 26:26 | W1P | Write 1 to set CP-to-AP interrupt No.10 | 1'h0 |
| cp2ap\_set\_irq\_09 | 25:25 | W1P | Write 1 to set CP-to-AP interrupt No.9 | 1'h0 |
| cp2ap\_set\_irq\_08 | 24:24 | W1P | Write 1 to set CP-to-AP interrupt No.8 | 1'h0 |
| cp2ap\_set\_irq\_07 | 23:23 | W1P | Write 1 to set CP-to-AP interrupt No.7 | 1'h0 |
| cp2ap\_set\_irq\_06 | 22:22 | W1P | Write 1 to set CP-to-AP interrupt No.6 | 1'h0 |
| cp2ap\_set\_irq\_05 | 21:21 | W1P | Write 1 to set CP-to-AP interrupt No.5 | 1'h0 |
| cp2ap\_set\_irq\_04 | 20:20 | W1P | Write 1 to set CP-to-AP interrupt No.4 | 1'h0 |
| cp2ap\_set\_irq\_03 | 19:19 | W1P | Write 1 to set CP-to-AP interrupt No.3 | 1'h0 |
| cp2ap\_set\_irq\_02 | 18:18 | W1P | Write 1 to set CP-to-AP interrupt No.2 | 1'h0 |
| cp2ap\_set\_irq\_01 | 17:17 | W1P | Write 1 to set CP-to-AP interrupt No.1 | 1'h0 |
| cp2ap\_set\_irq\_00 | 16:16 | W1P | For CP-to-AP interrupt No.0: Write 1 to set a IRQ If the IRQ is disable, the IRQ status will still be set to 1, but NO interrupt will give to CP | 1'h0 |
| cp2ap\_ena\_irq\_15 | 15:15 | RW | CP-to-AP interrupt No.15 enable, usage as below. | 1'h0 |
| cp2ap\_ena\_irq\_14 | 14:14 | RW | CP-to-AP interrupt No.14 enable, usage as below. | 1'h0 |
| cp2ap\_ena\_irq\_13 | 13:13 | RW | CP-to-AP interrupt No.13 enable, usage as below. | 1'h0 |
| cp2ap\_ena\_irq\_12 | 12:12 | RW | CP-to-AP interrupt No.12 enable, usage as below. | 1'h0 |
| cp2ap\_ena\_irq\_11 | 11:11 | RW | CP-to-AP interrupt No.11 enable, usage as below. | 1'h0 |
| cp2ap\_ena\_irq\_10 | 10:10 | RW | CP-to-AP interrupt No.10 enable, usage as below. | 1'h0 |
| cp2ap\_ena\_irq\_09 | 9:9 | RW | CP-to-AP interrupt No.9 enable, usage as below. | 1'h0 |
| cp2ap\_ena\_irq\_08 | 8:8 | RW | CP-to-AP interrupt No.8 enable, usage as below. | 1'h0 |
| cp2ap\_ena\_irq\_07 | 7:7 | RW | CP-to-AP interrupt No.7 enable, usage as below. | 1'h0 |
| cp2ap\_ena\_irq\_06 | 6:6 | RW | CP-to-AP interrupt No.6 enable, usage as below. | 1'h0 |
| cp2ap\_ena\_irq\_05 | 5:5 | RW | CP-to-AP interrupt No.5 enable, usage as below. | 1'h0 |
| cp2ap\_ena\_irq\_04 | 4:4 | RW | CP-to-AP interrupt No.4 enable, usage as below. | 1'h0 |
| cp2ap\_ena\_irq\_03 | 3:3 | RW | CP-to-AP interrupt No.3 enable, usage as below. | 1'h0 |
| cp2ap\_ena\_irq\_02 | 2:2 | RW | CP-to-AP interrupt No.2 enable, usage as below. | 1'h0 |
| cp2ap\_ena\_irq\_01 | 1:1 | RW | CP-to-AP interrupt No.1 enable, usage as below. | 1'h0 |
| cp2ap\_ena\_irq\_00 | 0:0 | RW | For CP-to-AP interrupt No.0: IRQ enable 0: Disable the IRQ; 1: Enable the IRQ from CP to AP; | 1'h0 |

### CP\_MAILBOX\_IRQ

Offset: 028

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RO | Not used | 16'h0 |
| cp2ap\_irq\_15 | 15:15 | W1C | CP-to-AP interrupt No.15 status: Write 1 to clear. | 1'h0 |
| cp2ap\_irq\_14 | 14:14 | W1C | CP-to-AP interrupt No.14 status: Write 1 to clear. | 1'h0 |
| cp2ap\_irq\_13 | 13:13 | W1C | CP-to-AP interrupt No.13 status: Write 1 to clear. | 1'h0 |
| cp2ap\_irq\_12 | 12:12 | W1C | CP-to-AP interrupt No.12 status: Write 1 to clear. | 1'h0 |
| cp2ap\_irq\_11 | 11:11 | W1C | CP-to-AP interrupt No.11 status: Write 1 to clear. | 1'h0 |
| cp2ap\_irq\_10 | 10:10 | W1C | CP-to-AP interrupt No.10 status: Write 1 to clear. | 1'h0 |
| cp2ap\_irq\_09 | 9:9 | W1C | CP-to-AP interrupt No.9 status: Write 1 to clear. | 1'h0 |
| cp2ap\_irq\_08 | 8:8 | W1C | CP-to-AP interrupt No.8 status: Write 1 to clear. | 1'h0 |
| cp2ap\_irq\_07 | 7:7 | W1C | CP-to-AP interrupt No.7 status: Write 1 to clear. | 1'h0 |
| cp2ap\_irq\_06 | 6:6 | W1C | CP-to-AP interrupt No.6 status: Write 1 to clear. | 1'h0 |
| cp2ap\_irq\_05 | 5:5 | W1C | CP-to-AP interrupt No.5 status: Write 1 to clear. | 1'h0 |
| cp2ap\_irq\_04 | 4:4 | W1C | CP-to-AP interrupt No.4 status: Write 1 to clear. | 1'h0 |
| cp2ap\_irq\_03 | 3:3 | W1C | CP-to-AP interrupt No.3 status: Write 1 to clear. | 1'h0 |
| cp2ap\_irq\_02 | 2:2 | W1C | CP-to-AP interrupt No.2 status: Write 1 to clear. | 1'h0 |
| cp2ap\_irq\_01 | 1:1 | W1C | CP-to-AP interrupt No.1 status: Write 1 to clear. | 1'h0 |
| cp2ap\_irq\_00 | 0:0 | W1C | CP-to-AP interrupt No.0 status: Write 1 to clear. | 1'h0 |

### Empty5962

Offset: 030

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| CP\_mailbox\_word0 | 31:0 | RW | CP mailbox word 0: [31:0] | 32'h0 |

### Empty13991

Offset: 034

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| CP\_mailbox\_word1 | 31:0 | RW | CP mailbox word 0: [63:32] | 32'h0 |

### Empty51554

Offset: 038

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| CP\_mailbox\_word2 | 31:0 | RW | CP mailbox word 0: [95:64] | 32'h0 |

### Empty29862

Offset: 03C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| CP\_mailbox\_word3 | 31:0 | RW | CP mailbox word 0: [127:96] | 32'h0 |

# AON\_PMUCTRL

## Introduction

PMU模块描述芯片的低功耗模式，以及在低功耗模式下不同模块的开关。

## Main Features

### Normal mode

芯片的正常运行模式，在框图中可以看到有红色开关的地方，是可以通过软件配置来进行开关。

### Sleep mode

### Deep Sleep mode

芯片的最低功耗模式。

## Function Description

## Basic Block Diagram

### Normal mode



### Sleep mode

下图中灰色的部分，是在Sleep模式下被关闭的模块



### Deep Sleep mode

Deep Sleep mode和Sleep mode的最大差异就是PSRAM在这个模式下被强制关

闭。



## AON\_PMUCTRL Register

### AON\_WAIT\_XTAL

Offset: 000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| aon\_chip\_ver | 31:0 | RO | For Debug Purpose [31:16] 00A6: A=AI, 6=CSK600x [15:8] 0A, 0B, 0C … TSMC / UMC / SMIC [7:0] 00, 01, 02 … Version Number | 32'ha60a01 |

### PMU\_CTRL2

Offset: 008

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:30 | RO | Not used | 2'h0 |
| gpio\_out | 29:24 | RW | the output value when aon\_gpio is in output mode | 6'h0 |
| ac\_scan\_key | 23:16 | RW | Used to enable AC\_SCAN mode in scan\_mode\_2 8'hAC : scan\_mode\_2 AC\_SCAN mode Others: scan\_mode\_2 DC\_SCAN mode | 8'h0 |
| Reserved | 15:10 | RO | Not used | 6'h0 |
| mask\_cp\_enter\_sleep | 9:9 | RW |  | 1'h0 |
| mask\_ap\_enter\_sleep | 8:8 | RW |  | 1'h0 |
| ena\_deepsleep | 7:7 | RW | Make the system go-to Deep Sleep with VAD wakeup | 1'h0 |
| ena\_litesleep | 6:6 | RW | Make the system go-to Light Sleep with VAD wakeup | 1'h0 |
| wait\_pu\_core | 5:3 | RW | Wait LDV\_VCORE power-up ready 0: count to 8'd40 (default) 1: count to 8'd4 2: count to 8'd8 (typical) 3: count to 8'd16 4: count to 8'd24 5: count to 8'd32 6: count to 8'd40 7: count to 8'd48 | 3'h0 |
| wait\_pu\_dcdc | 2:0 | RW | Wait DCDC power-up ready 0: count to 8'd192 (default) 1: count to 8'd32 2: count to 8'd64 (typical) 3: count to 8'd96 4: count to 8'd128 5: count to 8'd160 6: count to 8'd192 7: count to 8'd255 | 3'h0 |

### AON\_CRM\_CTRL1

Offset: 00C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:23 | RO | Not used | 9'h0 |
| sel\_32k\_clk | 22:22 | RW | Select the aon 32K clock between osc32k and xtal32k 0: osc32k 1: xtal32k | 1'h0 |
| sel\_efuse\_clk | 21:21 | RW | Select the eFuse controller clock 0: XTAL 24MHz 1: APB PCLK from digital core | 1'h0 |
| ena\_efuse\_clk | 20:20 | RW | Gating the clock of eFuse controller 0: Disable 1: Enable clock | 1'h1 |
| Reserved | 19:19 | RO | Not used | 1'h0 |
| pd\_cbtn\_clk3m | 18:18 | RW | Power-down C-Button CLK3M. This is a SW configurable bit to C-Button module | 1'h0 |
| sel\_cbtn\_clk3m | 17:17 | RW | Select the C-Button CLK3M source 0: 3m clk 1: 6m clk | 1'h0 |
| ena\_cbtn\_clk | 16:16 | RW | Enable C-Button clock, including the CLK3M and the VSW clock gating | 1'h0 |
| Reserved | 15:15 | RO | Not used | 1'h0 |
| sel\_keysense\_clk | 14:14 | RW | Select the divided VAD clock source 0: the XTAL 24MHz divided to 32kHz 1: the RCO 32kHz | 1'h1 |
| ena\_keysense\_clk | 13:13 | RW | Enable the divided Keysense clock | 1'h0 |
| Reserved | 12:7 | RO | Not used | 6'h0 |
| sel\_vad\_clk | 6:6 | RW | Select the divided VAD clock source 0: the XTAL 24MHz divided to VAD clock 1: the Always-On PCLK | 1'h0 |
| ena\_vad\_clk | 5:5 | RW | Enable the divided VAD clock 0: the clock source is gated 1: enable clock | 1'h0 |
| div\_vad\_clk\_ld | 4:4 | RW | At rising edge to load config VAD clock divider denominator | 1'h0 |
| div\_vad\_clk\_m | 3:0 | RW | Denominator M of 1/M divider of Always-On VAD clock | 4'h0 |

### AON\_CRM\_CTRL2

Offset: 010

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:2 | RO | Not used | 30'h0 |
| sw\_reset\_aon\_adc | 1:1 | RW | Write 1 to generate a SW reset to always-on CODEC The reset is a high active level and this bit need SW to write 0 to de-reset as the Codec of Audio ADC requirement. Hence, this bit is read-back as 1 when reset active. | 1'h0 |
| sw\_reset\_efuse | 0:0 | W1P | Write 1 to generate a SW reset to eFuse controller The reset is a high active level and this bit is clear automatically. | 1'h0 |

### AON\_PCLK\_FRC\_ON

Offset: 014

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:4 | RO | Not used | 28'h0 |
| timer\_wdt\_reg\_clk\_force\_on | 3:3 | RW | Force the PCLK of register file of AON\_TIMER is enable Default is force disable, it should be work properly when NOT force on | 1'h0 |
| adc\_reg\_clk\_force\_on | 2:2 | RW | Force the PCLK of register file of Always-On Audio ADC is enable Default is force disable, it should be work properly when NOT force on | 1'h0 |
| vad\_reg\_clk\_force\_on | 1:1 | RW | Force the PCLK of register file of Always-On VAD is enable Default is force disable, it should be work properly when NOT force on | 1'h0 |
| iomux\_reg\_clk\_force\_on | 0:0 | RW | Force the PCLK of register file of Always-On IOMUX is enable Default is force disable, it should be work properly when NOT force on | 1'h0 |

### AON\_IWDT\_FEED\_KEY

Offset: 018

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| cp\_wdt\_feed\_en\_key | 31:16 | RW | set this key to 16'hCAFE to enable auto feed to IWDT when CP WDT is fed | 16'h0 |
| ap\_wdt\_feed\_en\_key | 15:0 | RW | set this key to 16'hCAFE to enable auto feed to IWDT when AP WDT is fed | 16'h0 |

### AON\_RSVD\_REG0

Offset: 01C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| aon\_rsvd\_reg\_0 | 31:0 | RW | reserved for future use | 32'h0 |

### AON\_STATUS\_1

Offset: 020

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| por\_sw\_status | 31:0 | RW | A 32bit reserved status register It can be write by SW and only can be reset by POR or Pad Reset. | 32'h0 |

### AON\_STATUS\_2

Offset: 024

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:4 | RO | Not used | 28'h0 |
| boot\_option\_val | 3:2 | RW | The set value of boot option | 2'h0 |
| boot\_option\_frc | 1:1 | RW | Force the boot option read by SW is based on a set value | 1'h0 |
| ena\_irq\_aon\_wakeup | 0:0 | RW | Enable the interrupt of Always-On Wakeup | 1'h0 |

### AON\_STATUS\_3

Offset: 028

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| sys\_rst\_cause\_7 | 23:23 | W1C | The record of AP subsystem reset casue Bit[7]:blank | 1'h0 |
| sys\_rst\_cause\_6 | 22:22 | W1C | The record of AP subsystem reset casue Bit[6]:blank | 1'h0 |
| sys\_rst\_cause\_5 | 21:21 | W1C | The record of AP subsystem reset casue Bit[5]: star\_sysresetreq | 1'h0 |
| sys\_rst\_cause\_4 | 20:20 | W1C | The record of AP subsystem reset casue Bit[4]:star\_lockup | 1'h0 |
| sys\_rst\_cause\_3 | 19:19 | W1C | The record of AP subsystem reset casue Bit[3]:ap\_wdt\_rst | 1'h0 |
| sys\_rst\_cause\_2 | 18:18 | W1C | The record of AP subsystem reset casue Bit[2]:syscfg\_p\_ap\_reset | 1'h0 |
| sys\_rst\_cause\_1 | 17:17 | W1C | The record of AP subsystem reset casue Bit[1]:syscfg\_cp\_reset | 1'h0 |
| sys\_rst\_cause\_0 | 16:16 | W1C | The record of AP subsystem reset casue Bit[0]:cp\_wdt\_reset | 1'h0 |
| aon\_RO\_aud\_aon\_ldo25\_pg08 | 15:15 | RO | Not used | 1'h0 |
| aon\_RO\_aud\_aon\_ldo18\_pg08 | 14:14 | RO | Not used | 1'h0 |
| rbk\_boot\_option | 13:12 | RO | Boot option based the 2bits boot strap pin 11 - Flash boot 10 - UART boot 01 - USB boot 00 - Xtensa debug  Strap Pin [1:0] Default to Input and Pull-Up | 2'h3 |
| rbk\_pwr\_status | 11:8 | RO |  | 4'hf |
| rbk\_irq\_keysense | 7:7 | RO | Read-back only: Keysense interrupt raw status | 1'h0 |
| rbk\_irq\_cbutton | 6:6 | RO | Read-back only: C-Button interrupt raw status | 1'h0 |
| rbk\_irq\_iwdt | 5:5 | RO | Read-back only: iWDT interrupt raw status | 1'h0 |
| rbk\_irq\_vad | 4:4 | RO | Read-back only: VAD interrupt raw status | 1'h0 |
| rbk\_irq\_rtc | 3:3 | RO | Read-back only: RTC interrupt raw status | 1'h0 |
| aon\_rst\_status | 2:2 | W1C | Read-back: status of always-on subsystem reset Reset to 1 after always-on subsystem reset. It can be ONLY clear by SW write 1.  Usage of the reset status: 1. aon\_rst\_status=0 AND por\_status=0 Both statuses are clear by SW after system boot, as this combination.  2. aon\_rst\_status=1 AND por\_status=0 AON reset happens but NO POR reset. Hence it is a iWDT reset.  3. aon\_rst\_status=1 AND por\_status=1 Both statuses are default value. Hence it is a combination after POR reset.  4. aon\_rst\_status=0 AND por\_status=1 Reset case cannot generate the combination. An invalid status. | 1'h1 |
| por\_status | 1:1 | W1C | Read-back: status of POR Reset Reset to 1 after POR or Pad Reset. It can be ONLY clear by SW write 1. | 1'h1 |
| irq\_aon\_wakeup | 0:0 | W1C | Always-On Wakeup It can be clear by SW write 1. | 1'h0 |

# AON\_IOMUX

## Introduction

Always on 区域的IO功能配置，处于这个区域的GPIO只有6个，分别是GPIOB0 — GPIOB5

## Main Features

## Function Description

## Basic Block Diagram

## AON\_IOMUX Register

### PAD\_AON\_GPIOB

Offset: 000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:26 | RO | Not used | 6'h0 |
| pad\_aon\_gpiob\_\_drv | 25:24 | RW | Pad drive strength configuration | 2'h3 |
| Reserved | 23:23 | RW | Not used | 1'h0 |
| pad\_aon\_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\_aon\_gpiob\_\_out\_reg | 21:21 | RW | Pad output value when out\_frc ==1 | 1'h0 |
| pad\_aon\_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\_aon\_gpiob\_\_oen\_reg | 19:19 | RW | Pad oen value when oen\_frc==1 | 1'h0 |
| pad\_aon\_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\_aon\_gpiob\_\_pull\_up | 17:17 | RW | Pad pullup enable control when pull\_frc ==1 0: pullup is disabled 1:pullup is enabled | 1'h0 |
| pad\_aon\_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\_aon\_gpiob\_\_ana\_sel | 8:5 | RW | Pad analog function selection, only used on Analog/Digital mixed I/Os | 4'h0 |
| pad\_aon\_gpiob\_\_fsel | 4:0 | RW | Pad function selection | 5'h0 |

# APC

## Introduction

APC属于CP的音频模块，这个部分不对用户开放

## Main Features

## Function Description

## Basic Block Diagram

## APC Register

### APC\_CFG

Offset: 000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:4 | RO | Not used | 28'h0 |
| auto\_clk\_gating | 3:3 | RW | enable auto\_clk\_gating 0: disable 1: enable | 1'h1 |
| apc\_tx\_path\_reset | 2:2 | W1C | TX path soft reset, write 1 to generate a reset pulse | 1'h0 |
| apc\_rx\_path\_reset | 1:1 | W1C | RX path soft reset, write 1 to generate a reset pulse | 1'h0 |
| apc\_enable | 0:0 | RW | APC module enable 0: disable 1: enable | 1'h0 |

### APC\_CFG\_RSVD0

Offset: 004

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:0 | RO | Not used | 32'h0 |

### APC\_CFG\_RSVD1

Offset: 008

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:0 | RO | Not used | 32'h0 |

### APC\_TX\_CH0\_CFG

Offset: 00C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:28 | RO | Not used | 4'h0 |
| tx\_ch0\_dma\_thd\_sel | 27:26 | RW | DMA burst request theshold 0: 1 word 1: 4 words 2: 8 words 3: 16 words | 2'h0 |
| tx\_ch0\_stereo\_mode | 25:25 | RW | 0: mono mode 1: stereo mode | 1'h0 |
| tx\_ch0\_r\_fifo\_cnt | 24:20 | ROI | Data count in FIFO | 5'h0 |
| tx\_ch0\_r\_fifo\_flush | 19:19 | W1C | clear FIFO data, write 1 to generate a clear pulse | 1'h0 |
| tx\_ch0\_r\_mode | 18:17 | RW | channel mode 0: 16-bit mode, {R1, R0} 1: 24-bit mode, {8'd0, R0} 2: 32-bit mode, {R0} 3: 24-bit mode, {R0, 8'd0} | 2'h0 |
| tx\_ch0\_r\_en | 16:16 | RW | channel enable 0: disable 1: enable | 1'h0 |
| Reserved | 15:9 | RO | Not used | 7'h0 |
| tx\_ch0\_l\_fifo\_cnt | 8:4 | ROI | Data count in FIFO | 5'h0 |
| tx\_ch0\_l\_fifo\_flush | 3:3 | W1C | clear FIFO data, write 1 to generate a clear pulse | 1'h0 |
| tx\_ch0\_l\_mode | 2:1 | RW | channel bit mode 0: 16-bit mode, {R0, L0}@stereo mode or {L1, L0}@mono mode 1: 24-bit mode, {8'd0,L0} 2: 32-bit mode, {L0} 3: 24-bit mode, {L0, 8'd0} | 2'h0 |
| tx\_ch0\_l\_en | 0:0 | RW | channel enable 0: disable 1: enable | 1'h0 |

### APC\_TX\_CH1\_CFG

Offset: 010

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| eq\_bypass\_reg | 31:31 | RW | 1:bypass eq | 1'h1 |
| eq\_clr\_done | 30:30 | ROI | EQ module clear done indicator, 1 for clear done | 1'h0 |
| eq\_clr | 29:29 | RW | EQ module soft clear, write 1 to generate a clear pulse | 1'h0 |
| tx\_ch1\_dst\_sel | 28:28 | RW | select the destination of TX\_CH1 0: DAC01 1: I2S2 | 1'h0 |
| tx\_ch1\_dma\_thd\_sel | 27:26 | RW | DMA burst request theshold 0: 1 word 1: 4 words 2: 8 words 3: 16 words | 2'h0 |
| tx\_ch1\_stereo\_mode | 25:25 | RW | 0: mono mode 1: stereo mode | 1'h0 |
| tx\_ch1\_r\_fifo\_cnt | 24:20 | ROI | Data count in FIFO | 5'h0 |
| tx\_ch1\_r\_fifo\_flush | 19:19 | W1C | clear FIFO data, write 1 to generate a clear pulse | 1'h0 |
| tx\_ch1\_r\_mode | 18:17 | RW | channel mode 0: 16-bit mode, {R1, R0} 1: 24-bit mode, {8'd0, R0} 2: 32-bit mode, {R0} 3: 24-bit mode, {R0, 8'd0} | 2'h0 |
| tx\_ch1\_r\_en | 16:16 | RW | channel enable 0: disable 1: enable | 1'h0 |
| tx\_ch1\_rd\_autofeed | 15:15 | RW | Auto feed 0 when reading TX channel 0: do not auto feed 1: auto feed 0 | 1'h0 |
| eq\_stage | 14:11 | RW | EQ stage setting, support up to 10 stages | 4'h0 |
| eq\_ch\_en | 10:9 | RW | equalization enable [0] L channel EQ enable -> 0: disable, 1: enable [1] R channel EQ enable-> 0: disable, 1: enable | 2'h0 |
| tx\_ch1\_l\_fifo\_cnt | 8:4 | ROI | Data count in FIFO | 5'h0 |
| tx\_ch1\_l\_fifo\_flush | 3:3 | W1C | clear FIFO data, write 1 to generate a clear pulse | 1'h0 |
| tx\_ch1\_l\_mode | 2:1 | RW | channel bit mode 0: 16-bit mode, {R0, L0}@stereo mode or {L1, L0}@mono mode 1: 24-bit mode, {8'd0,L0} 2: 32-bit mode, {L0} 3: 24-bit mode, {L0, 8'd0} | 2'h0 |
| tx\_ch1\_l\_en | 0:0 | RW | channel enable 0: disable 1: enable | 1'h0 |

### APC\_RX\_CH0\_CFG

Offset: 014

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:30 | RO | Not used | 2'h0 |
| rx\_ch0\_src\_sel | 29:28 | RW | select the source of RX\_CH0 0: ADC01 1: I2S0 2: ADC01 VAD Mode 3: Reserved | 2'h0 |
| rx\_ch0\_dma\_thd\_sel | 27:26 | RW | DMA burst request theshold 0: 1 word 1: 4 words 2: 8 words 3: 16 words | 2'h0 |
| rx\_ch0\_stereo\_mode | 25:25 | RW | 0: mono mode 1: stereo mode | 1'h0 |
| rx\_ch0\_r\_fifo\_cnt | 24:20 | ROI | Data count in FIFO | 5'h0 |
| rx\_ch0\_r\_fifo\_flush | 19:19 | W1C | clear FIFO data, write 1 to generate a clear pulse | 1'h0 |
| rx\_ch0\_r\_mode | 18:17 | RW | channel mode 0: 16-bit mode, {R1, R0} 1: 24-bit mode, {8'd0, R0} 2: 32-bit mode, {R0} 3: 24-bit mode, {R0, 8'd0} | 2'h0 |
| rx\_ch0\_r\_en | 16:16 | RW | channel enable 0: disable 1: enable | 1'h0 |
| Reserved | 15:9 | RO | Not used | 7'h0 |
| rx\_ch0\_l\_fifo\_cnt | 8:4 | ROI | Data count in FIFO | 5'h0 |
| rx\_ch0\_l\_fifo\_flush | 3:3 | W1C | clear FIFO data, write 1 to generate a clear pulse | 1'h0 |
| rx\_ch0\_l\_mode | 2:1 | RW | channel bit mode 0: 16-bit mode, {R0, L0}@stereo mode or {L1, L0}@mono mode 1: 24-bit mode, {8'd0,L0} 2: 32-bit mode, {L0} 3: 24-bit mode, {L0, 8'd0} | 2'h0 |
| rx\_ch0\_l\_en | 0:0 | RW | channel enable 0: disable 1: enable | 1'h0 |

### APC\_RX\_CH1\_CFG

Offset: 018

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:30 | RO | Not used | 2'h0 |
| rx\_ch1\_src\_sel | 29:28 | RW | select the source of RX\_CH1 0: ADC23 1: I2S1 2: ADC23 VAD Mode 3: Reserved | 2'h0 |
| rx\_ch1\_dma\_thd\_sel | 27:26 | RW | DMA burst request theshold 0: 1 word 1: 4 words 2: 8 words 3: 16 words | 2'h0 |
| rx\_ch1\_stereo\_mode | 25:25 | RW | 0: mono mode 1: stereo mode | 1'h0 |
| rx\_ch1\_r\_fifo\_cnt | 24:20 | ROI | Data count in FIFO | 5'h0 |
| rx\_ch1\_r\_fifo\_flush | 19:19 | W1C | clear FIFO data, write 1 to generate a clear pulse | 1'h0 |
| rx\_ch1\_r\_mode | 18:17 | RW | channel mode 0: 16-bit mode, {R1, R0} 1: 24-bit mode, {8'd0, R0} 2: 32-bit mode, {R0} 3: 24-bit mode, {R0, 8'd0} | 2'h0 |
| rx\_ch1\_r\_en | 16:16 | RW | channel enable 0: disable 1: enable | 1'h0 |
| Reserved | 15:9 | RO | Not used | 7'h0 |
| rx\_ch1\_l\_fifo\_cnt | 8:4 | ROI | Data count in FIFO | 5'h0 |
| rx\_ch1\_l\_fifo\_flush | 3:3 | W1C | clear FIFO data, write 1 to generate a clear pulse | 1'h0 |
| rx\_ch1\_l\_mode | 2:1 | RW | channel bit mode 0: 16-bit mode, {R0, L0}@stereo mode or {L1, L0}@mono mode 1: 24-bit mode, {8'd0,L0} 2: 32-bit mode, {L0} 3: 24-bit mode, {L0, 8'd0} | 2'h0 |
| rx\_ch1\_l\_en | 0:0 | RW | channel enable 0: disable 1: enable | 1'h0 |

### APC\_RX\_CH2\_CFG

Offset: 01C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:30 | RO | Not used | 2'h0 |
| rx\_ch2\_src\_sel | 29:28 | RW | select the source of RX\_CH2 0: TX\_CH1 Loopback 1: I2S2 2: I2S1 3: Reserved | 2'h0 |
| rx\_ch2\_dma\_thd\_sel | 27:26 | RW | DMA burst request theshold 0: 1 word 1: 4 words 2: 8 words 3: 16 words | 2'h0 |
| rx\_ch2\_stereo\_mode | 25:25 | RW | 0: mono mode 1: stereo mode | 1'h0 |
| rx\_ch2\_r\_fifo\_cnt | 24:20 | ROI | Data count in FIFO | 5'h0 |
| rx\_ch2\_r\_fifo\_flush | 19:19 | W1C | clear FIFO data, write 1 to generate a clear pulse | 1'h0 |
| rx\_ch2\_r\_mode | 18:17 | RW | channel mode 0: 16-bit mode, {R1, R0} 1: 24-bit mode, {8'd0, R0} 2: 32-bit mode, {R0} 3: 24-bit mode, {R0, 8'd0} | 2'h0 |
| rx\_ch2\_r\_en | 16:16 | RW | channel enable 0: disable 1: enable | 1'h0 |
| Reserved | 15:13 | RO | Not used | 3'h0 |
| src\_clr | 12:12 | W1C | write 1 to generate single pulse to clear SRC datapath | 1'h0 |
| src\_mode | 11:11 | RW | 0: 48KHz->8KHz 1: 48KHz->16KHz | 1'h0 |
| src\_ch\_en | 10:9 | RW | [0] ch2\_l\_src\_en, 1 for enable [1] ch2\_r\_src\_en, 1 for enable | 2'h0 |
| rx\_ch2\_l\_fifo\_cnt | 8:4 | ROI | Data count in FIFO | 5'h0 |
| rx\_ch2\_l\_fifo\_flush | 3:3 | W1C | clear FIFO data, write 1 to generate a clear pulse | 1'h0 |
| rx\_ch2\_l\_mode | 2:1 | RW | channel bit mode 0: 16-bit mode, {R0, L0}@stereo mode or {L1, L0}@mono mode 1: 24-bit mode, {8'd0,L0} 2: 32-bit mode, {L0} 3: 24-bit mode, {L0, 8'd0} | 2'h0 |
| rx\_ch2\_l\_en | 0:0 | RW | channel enable 0: disable 1: enable | 1'h0 |

### EQCOEF\_0

Offset: 020

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_0 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_1

Offset: 024

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_1 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_2

Offset: 028

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_2 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_3

Offset: 02C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_3 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_4

Offset: 030

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_4 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_5

Offset: 034

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_5 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_6

Offset: 038

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_6 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_7

Offset: 03C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_7 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_8

Offset: 040

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_8 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_9

Offset: 044

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_9 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_10

Offset: 048

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_10 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_11

Offset: 04C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_11 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_12

Offset: 050

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_12 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_13

Offset: 054

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_13 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_14

Offset: 058

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_14 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_15

Offset: 05C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_15 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_16

Offset: 060

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_16 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_17

Offset: 064

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_17 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_18

Offset: 068

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_18 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_19

Offset: 06C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_19 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_20

Offset: 070

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_20 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_21

Offset: 074

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_21 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_22

Offset: 078

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_22 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_23

Offset: 07C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_23 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_24

Offset: 080

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_24 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_25

Offset: 084

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_25 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_26

Offset: 088

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_26 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_27

Offset: 08C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_27 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_28

Offset: 090

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_28 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_29

Offset: 094

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_29 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_30

Offset: 098

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_30 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_31

Offset: 09C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_31 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_32

Offset: 0A0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_32 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_33

Offset: 0A4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_33 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_34

Offset: 0A8

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_34 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_35

Offset: 0AC

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_35 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_36

Offset: 0B0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_36 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_37

Offset: 0B4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_37 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_38

Offset: 0B8

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_38 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_39

Offset: 0BC

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_39 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_40

Offset: 0C0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_40 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_41

Offset: 0C4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_41 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_42

Offset: 0C8

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_42 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_43

Offset: 0CC

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_43 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_44

Offset: 0D0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_44 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_45

Offset: 0D4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_45 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_46

Offset: 0D8

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_46 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_47

Offset: 0DC

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_47 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_48

Offset: 0E0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_48 | 23:0 | RW |  | 24'h0 |

### EQCOEF\_49

Offset: 0E4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| eq\_coef\_49 | 23:0 | RW |  | 24'h0 |

### APC\_I2S0\_CFG0

Offset: 0E8

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| i2s0\_enable | 31:31 | RW | enable i2s interface | 1'h0 |
| i2s0\_sw\_reset | 30:30 | RW | 1: reset i2s 0: do not reset i2s | 1'h0 |
| i2s0\_rstn\_bypass | 29:29 | RW | 0: i2s will be reset when i2s disabled | 1'h0 |
| i2s0\_en\_force\_on | 28:28 | RW | 1: enable force on for slave mode | 1'h0 |
| i2s0\_bck\_force\_on | 27:27 | RW | 1: bck force on for master mode | 1'h0 |
| i2s0\_bypass\_fifovld | 26:26 | RW | 0: i2s will be enabled when tx fifo valid 1: i2s will be enabled regardless of tx fifo valid | 1'h1 |
| i2s0\_tx\_mode | 25:24 | RW | configure mono or stereo formal for Audio data out. 2'b00: STEREO\_STEREO: stereo input form IFC, stereo output to pin. 2'b01: MONOL\_STEREO\_DUPLL: mono channel L input form IFC, stereo output duplicate in both channels to pin. 2'b10: MONOR\_STEREO\_DUPLL: mono channel R input form IFC, stereo output duplicate in both channels to pin. 2'b11: MONO\_IN\_MONO\_OUT function mono input from IFC, stereo output in left channel to pin, can be implemented by setting apc\_tx\_cfg\_ch[1] or apc\_tx\_cfg\_ch[0] disable. when works in DAI, or DSD mode, always select "00" mode STEREO\_STEREO | 2'h0 |
| i2s0\_serial\_mode | 23:22 | RW | configure serial apc mode. 0: I2S 1: VOICE/PCM 2: DAI 3: DSD when mode is set DAI, the bit Master Mode should be set to "1". Data should be sent out on falling edge, which requires either Bclk\_Pol="0" and Half\_Cycle\_DLY="1" or Bclk\_Pol="1" and Half\_Cycle\_DLY="0". Bits Tx\_DLY and BCKOut\_Gate must be configured to "0" and "1". The DAI mode must NOT be modified after apc is enabled. | 2'h0 |
| i2s0\_master\_mode | 21:21 | RW | configure apc works in master mode (LRCLK and BCK timing signals are generated internally) or slave mode (LRCLK and BCK timing signals are generated externally). 0: SLAVE 1: MASTER | 1'h1 |
| i2s0\_tx\_dly | 20:20 | RW | configure the delay between serial data out MSB and LRCK edge. 0: ALIGN: Digital audio out MSB is aligned with LRCLK edge. 1: DLY\_1: Digital audio out MSB is 1 cycle delayed to LRCLK edge | 1'h0 |
| i2s0\_tx\_rx\_dly\_s | 19:19 | RW | ONLY for slave mode: configure 1 cycle supplementary Tx/RX delay. 0: NO\_DLY: No supplementary Tx/Rx delay 1: DLY: One cycle supplementary Tx/Rx delay | 1'h0 |
| i2s0\_rx\_dly | 18:17 | RW | indicates the delay between serial data in MSB and LRCK edge 2'b00: ALIGN: Digital audio in MSB is aligned with LRCLK edge 2'b01: DLY\_1: Digital audio in MSB is 1 cycle delayed to LRCLK edge 2'b10: DLY\_2: Digital audio in MSB is 2 cycle delayed to LRCLK edge 2'b11: DLY\_3: Digital audio in MSB is 3 cycle delayed to LRCLK edge | 2'h0 |
| i2s0\_tx\_half\_cycle\_dly | 16:16 | RW | delayed audio output data by half cycle. 0: NO\_DLY 1: DLY | 1'h0 |
| i2s0\_rx\_half\_cycle\_dly | 15:15 | RW | delayed audio input data by half cycle. 0: NO\_DLY 1: DLY | 1'h0 |
| i2s0\_bckout\_gate | 14:14 | RW | sets the BackOut gating. This bit dicide if apc continue to output BCK clock after data has been sent. 0: NO\_GATE 1:GATED | 1'h0 |
| i2s0\_bck\_pol | 13:13 | RW | if Master Mode, invert BCLK out is slave Mode, invert BCLK in. 0: NORMAL 1: INVERT | 1'h1 |
| i2s0\_lrck\_pol | 12:12 | RW | configure LRCK polarity. 0: LEFT\_H\_RIGHT\_L: high level on LRCK means left channel, low level on LRCK means right channel. 1: LEFT\_L\_RIGHT\_H: high level on LRCK means right channel, low level on LRCK means left channel. Note: this bit should be set to "0"(LEFT\_H\_RIGHT\_L) in voice mode | 1'h0 |
| i2s0\_wlen | 11:10 | RW | i2S data length 00: 16-bit 01: 20bit 10: 24bit 11: 32-bit | 2'h0 |
| i2s0\_bck\_lrck | 9:5 | RW | configure extend cycles of BCK after data transfer is finished. Voice\_Mode: each sample takes wlen + tx\_bck\_lrck BCLK cycle Audio\_Mode: each sample takes 2\*(wlen + tx\_bck\_lrck) BCLK cycle. 2 times than Voice | 5'h0 |
| i2s0\_right\_justified | 4:4 | RW | right justified mode | 1'h0 |
| i2s0\_lsb | 3:3 | RW | When high, the output data format is with the least significant bit first 0: MSB 1: LSB | 1'h0 |
| i2s0\_loop\_back | 2:2 | RW | sets i2s in loop back mode. The feature is for debug only and can not work in DAI mode 0: NORMAL 1: LOOPBACK | 1'h0 |
| i2s0\_swap\_chlr | 1:0 | RW | Swap Channel L and Channel R[0] [0]: for output channel swap, 0:left, 1:right [1]: for input channel swap, 0: left, 1: right | 2'h0 |

### APC\_I2S0\_CFG1

Offset: 0EC

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:28 | RO | Not used | 4'h0 |
| i2s0\_same\_edge | 27:27 | RW | 0: the edge of internal bck and bckout are inversed. 1: internal bck and bckout have same edge.  only used in master mode | 1'h0 |
| i2s0\_bck\_div\_ld | 26:26 | W1C | bck divider load control, write this bit to 1 to make divider take effect | 1'h0 |
| i2s0\_bck\_div | 25:16 | RW | bck divider, must set i2s0\_bck\_div\_ld to 1 to make the divider value take effect i2s0\_bck = codec\_clk\_adc / (i2s0\_bck\_div+1)  notice: the initial value on hardware is 5, that means i2s0\_bck = codec\_clk\_adc / 6 when i2s0\_bck\_div and i2s0\_bck\_div\_ld are not configured | 10'h5 |
| i2s0\_slot\_lrck | 15:9 | RW | The counter of empty cycles after all the slots are transmitted to get specific sample rate | 7'h0 |
| i2s0\_longsync | 8:8 | RW | longsync enable | 1'h0 |
| i2s0\_slotnum | 7:0 | RW | slots number to be transferred in one frame | 8'h0 |

### APC\_I2S1\_CFG0

Offset: 0F0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| i2s1\_enable | 31:31 | RW | enable i2s interface | 1'h0 |
| i2s1\_sw\_reset | 30:30 | RW | 1: reset i2s 0: do not reset i2s | 1'h0 |
| i2s1\_rstn\_bypass | 29:29 | RW | 0: i2s will be reset when i2s disabled | 1'h0 |
| i2s1\_en\_force\_on | 28:28 | RW | 1: enable force on for slave mode | 1'h0 |
| i2s1\_bck\_force\_on | 27:27 | RW | 1: bck force on for master mode | 1'h0 |
| i2s1\_bypass\_fifovld | 26:26 | RW | 0: i2s will be enabled when tx fifo valid 1: i2s will be enabled regardless of tx fifo valid | 1'h0 |
| i2s1\_tx\_mode | 25:24 | RW | configure mono or stereo formal for Audio data out. 2'b00: STEREO\_STEREO: stereo input form IFC, stereo output to pin. 2'b01: MONOL\_STEREO\_DUPLL: mono channel L input form IFC, stereo output duplicate in both channels to pin. 2'b10: MONOR\_STEREO\_DUPLL: mono channel R input form IFC, stereo output duplicate in both channels to pin. 2'b11: MONO\_IN\_MONO\_OUT function mono input from IFC, stereo output in left channel to pin, can be implemented by setting apc\_tx\_cfg\_ch[1] or apc\_tx\_cfg\_ch[0] disable. when works in DAI, or DSD mode, always select "00" mode STEREO\_STEREO | 2'h0 |
| i2s1\_serial\_mode | 23:22 | RW | configure serial apc mode. 0: I2S 1: VOICE/PCM 2: DAI 3: DSD when mode is set DAI, the bit Master Mode should be set to "1". Data should be sent out on falling edge, which requires either Bclk\_Pol="0" and Half\_Cycle\_DLY="1" or Bclk\_Pol="1" and Half\_Cycle\_DLY="0". Bits Tx\_DLY and BCKOut\_Gate must be configured to "0" and "1". The DAI mode must NOT be modified after apc is enabled. | 2'h0 |
| i2s1\_master\_mode | 21:21 | RW | configure apc works in master mode (LRCLK and BCK timing signals are generated internally) or slave mode (LRCLK and BCK timing signals are generated externally). 0: SLAVE 1: MASTER | 1'h1 |
| i2s1\_tx\_dly | 20:20 | RW | configure the delay between serial data out MSB and LRCK edge. 0: ALIGN: Digital audio out MSB is aligned with LRCLK edge. 1: DLY\_1: Digital audio out MSB is 1 cycle delayed to LRCLK edge | 1'h0 |
| i2s1\_tx\_rx\_dly\_s | 19:19 | RW | ONLY for slave mode: configure 1 cycle supplementary Tx/RX delay. 0: NO\_DLY: No supplementary Tx/Rx delay 1: DLY: One cycle supplementary Tx/Rx delay | 1'h0 |
| i2s1\_rx\_dly | 18:17 | RW | indicates the delay between serial data in MSB and LRCK edge 2'b00: ALIGN: Digital audio in MSB is aligned with LRCLK edge 2'b01: DLY\_1: Digital audio in MSB is 1 cycle delayed to LRCLK edge 2'b10: DLY\_2: Digital audio in MSB is 2 cycle delayed to LRCLK edge 2'b11: DLY\_3: Digital audio in MSB is 3 cycle delayed to LRCLK edge | 2'h0 |
| i2s1\_tx\_half\_cycle\_dly | 16:16 | RW | delayed audio output data by half cycle. 0: NO\_DLY 1: DLY | 1'h0 |
| i2s1\_rx\_half\_cycle\_dly | 15:15 | RW | delayed audio input data by half cycle. 0: NO\_DLY 1: DLY | 1'h0 |
| i2s1\_bckout\_gate | 14:14 | RW | sets the BackOut gating. This bit dicide if apc continue to output BCK clock after data has been sent. 0: NO\_GATE 1:GATED | 1'h0 |
| i2s1\_bck\_pol | 13:13 | RW | if Master Mode, invert BCLK out is slave Mode, invert BCLK in. 0: NORMAL 1: INVERT | 1'h1 |
| i2s1\_lrck\_pol | 12:12 | RW | configure LRCK polarity. 0: LEFT\_H\_RIGHT\_L: high level on LRCK means left channel, low level on LRCK means right channel. 1: LEFT\_L\_RIGHT\_H: high level on LRCK means right channel, low level on LRCK means left channel. Note: this bit should be set to "0"(LEFT\_H\_RIGHT\_L) in voice mode | 1'h0 |
| i2s1\_wlen | 11:10 | RW | i2S data length 00: 16-bit 01: 20bit 10: 24bit 11: 32-bit | 2'h0 |
| i2s1\_bck\_lrck | 9:5 | RW | configure extend cycles of BCK after data transfer is finished. Voice\_Mode: each sample takes wlen + tx\_bck\_lrck BCLK cycle Audio\_Mode: each sample takes 2\*(wlen + tx\_bck\_lrck) BCLK cycle. 2 times than Voice | 5'h0 |
| i2s1\_right\_justified | 4:4 | RW | right justified mode | 1'h0 |
| i2s1\_lsb | 3:3 | RW | When high, the output data format is with the least significant bit first 0: MSB 1: LSB | 1'h0 |
| i2s1\_loop\_back | 2:2 | RW | sets i2s in loop back mode. The feature is for debug only and can not work in DAI mode 0: NORMAL 1: LOOPBACK | 1'h0 |
| i2s1\_swap\_chlr | 1:0 | RW | Swap Channel L and Channel R[0] [0]: for output channel swap, 0:left, 1:right [1]: for input channel swap, 0: left, 1: right | 2'h0 |

### APC\_I2S1\_CFG1

Offset: 0F4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:28 | RO | Not used | 4'h0 |
| i2s1\_same\_edge | 27:27 | RW | 0: the edge of internal bck and bckout are inversed. 1: internal bck and bckout have same edge.  only used in master mode | 1'h0 |
| i2s1\_bck\_div\_ld | 26:26 | W1C | bck divider load control, write this bit to 1 to make divider take effect | 1'h0 |
| i2s1\_bck\_div | 25:16 | RW | bck divider, must set i2s1\_bck\_div\_ld to 1 to make the divider value take effect i2s1\_bck = codec\_clk\_adc / (1+i2s1\_bck\_div) notice: the initial value on hardware is 5, that means i2s1\_bck = codec\_clk\_adc / 6 when i2s1\_bck\_div and i2s1\_bck\_div\_ld are not configured | 10'h5 |
| i2s1\_slot\_lrck | 15:9 | RW | The counter of empty cycles after all the slots are transmitted to get specific sample rate | 7'h0 |
| i2s1\_longsync | 8:8 | RW | longsync enable | 1'h0 |
| i2s1\_slotnum | 7:0 | RW | slots number to be transferred in one frame | 8'h0 |

### APC\_I2S2\_CFG0

Offset: 0F8

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| i2s2\_enable | 31:31 | RW | enable i2s interface | 1'h0 |
| i2s2\_sw\_reset | 30:30 | RW | 1: reset i2s 0: do not reset i2s | 1'h0 |
| i2s2\_rstn\_bypass | 29:29 | RW | 0: i2s will be reset when i2s disabled | 1'h0 |
| i2s2\_en\_force\_on | 28:28 | RW | 1: enable force on for slave mode | 1'h0 |
| i2s2\_bck\_force\_on | 27:27 | RW | 1: bck force on for master mode | 1'h0 |
| i2s2\_bypass\_fifovld | 26:26 | RW | 0: i2s will be enabled when tx fifo valid 1: i2s will be enabled regardless of tx fifo valid | 1'h0 |
| i2s2\_tx\_mode | 25:24 | RW | configure mono or stereo formal for Audio data out. 2'b00: STEREO\_STEREO: stereo input form IFC, stereo output to pin. 2'b01: MONOL\_STEREO\_DUPLL: mono channel L input form IFC, stereo output duplicate in both channels to pin. 2'b10: MONOR\_STEREO\_DUPLL: mono channel R input form IFC, stereo output duplicate in both channels to pin. 2'b11: MONO\_IN\_MONO\_OUT function mono input from IFC, stereo output in left channel to pin, can be implemented by setting apc\_tx\_cfg\_ch[1] or apc\_tx\_cfg\_ch[0] disable. when works in DAI, or DSD mode, always select "00" mode STEREO\_STEREO | 2'h0 |
| i2s2\_serial\_mode | 23:22 | RW | configure serial apc mode. 0: I2S 1: VOICE/PCM 2: DAI 3: DSD when mode is set DAI, the bit Master Mode should be set to "1". Data should be sent out on falling edge, which requires either Bclk\_Pol="0" and Half\_Cycle\_DLY="1" or Bclk\_Pol="1" and Half\_Cycle\_DLY="0". Bits Tx\_DLY and BCKOut\_Gate must be configured to "0" and "1". The DAI mode must NOT be modified after apc is enabled. | 2'h0 |
| i2s2\_master\_mode | 21:21 | RW | configure apc works in master mode (LRCLK and BCK timing signals are generated internally) or slave mode (LRCLK and BCK timing signals are generated externally). 0: SLAVE 1: MASTER | 1'h1 |
| i2s2\_tx\_dly | 20:20 | RW | configure the delay between serial data out MSB and LRCK edge. 0: ALIGN: Digital audio out MSB is aligned with LRCLK edge. 1: DLY\_1: Digital audio out MSB is 1 cycle delayed to LRCLK edge | 1'h0 |
| i2s2\_tx\_rx\_dly\_s | 19:19 | RW | ONLY for slave mode: configure 1 cycle supplementary Tx/RX delay. 0: NO\_DLY: No supplementary Tx/Rx delay 1: DLY: One cycle supplementary Tx/Rx delay | 1'h0 |
| i2s2\_rx\_dly | 18:17 | RW | indicates the delay between serial data in MSB and LRCK edge 2'b00: ALIGN: Digital audio in MSB is aligned with LRCLK edge 2'b01: DLY\_1: Digital audio in MSB is 1 cycle delayed to LRCLK edge 2'b10: DLY\_2: Digital audio in MSB is 2 cycle delayed to LRCLK edge 2'b11: DLY\_3: Digital audio in MSB is 3 cycle delayed to LRCLK edge | 2'h0 |
| i2s2\_tx\_half\_cycle\_dly | 16:16 | RW | delayed audio output data by half cycle. 0: NO\_DLY 1: DLY | 1'h0 |
| i2s2\_rx\_half\_cycle\_dly | 15:15 | RW | delayed audio input data by half cycle. 0: NO\_DLY 1: DLY | 1'h0 |
| i2s2\_bckout\_gate | 14:14 | RW | sets the BackOut gating. This bit dicide if apc continue to output BCK clock after data has been sent. 0: NO\_GATE 1:GATED | 1'h0 |
| i2s2\_bck\_pol | 13:13 | RW | if Master Mode, invert BCLK out is slave Mode, invert BCLK in. 0: NORMAL 1: INVERT | 1'h1 |
| i2s2\_lrck\_pol | 12:12 | RW | configure LRCK polarity. 0: LEFT\_H\_RIGHT\_L: high level on LRCK means left channel, low level on LRCK means right channel. 1: LEFT\_L\_RIGHT\_H: high level on LRCK means right channel, low level on LRCK means left channel. Note: this bit should be set to "0"(LEFT\_H\_RIGHT\_L) in voice mode | 1'h0 |
| i2s2\_wlen | 11:10 | RW | i2S data length 00: 16-bit 01: 20bit 10: 24bit 11: 32-bit | 2'h0 |
| i2s2\_bck\_lrck | 9:5 | RW | configure extend cycles of BCK after data transfer is finished. Voice\_Mode: each sample takes wlen + tx\_bck\_lrck BCLK cycle Audio\_Mode: each sample takes 2\*(wlen + tx\_bck\_lrck) BCLK cycle. 2 times than Voice | 5'h0 |
| i2s2\_right\_justified | 4:4 | RW | right justified mode | 1'h0 |
| i2s2\_lsb | 3:3 | RW | When high, the output data format is with the least significant bit first 0: MSB 1: LSB | 1'h0 |
| i2s2\_loop\_back | 2:2 | RW | sets i2s in loop back mode. The feature is for debug only and can not work in DAI mode 0: NORMAL 1: LOOPBACK | 1'h0 |
| i2s2\_swap\_chlr | 1:0 | RW | Swap Channel L and Channel R[0] [0]: for output channel swap, 0:left, 1:right [1]: for input channel swap, 0: left, 1: right | 2'h0 |

### APC\_I2S2\_CFG1

Offset: 0FC

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:28 | RO | Not used | 4'h0 |
| i2s2\_same\_edge | 27:27 | RW | 0: the edge of internal bck and bckout are inversed. 1: internal bck and bckout have same edge.  only used in master mode | 1'h0 |
| i2s2\_bck\_div\_ld | 26:26 | W1C | bck divider load control, write this bit to 1 to make divider take effect | 1'h0 |
| i2s2\_bck\_div | 25:16 | RW | bck divider, must set i2s2\_bck\_div\_ld to 1 to make the divider value take effect i2s2\_bck = codec\_clk\_adc / (i2s2\_bck\_div+1) notice: the initial value on hardware is 6, that means i2s2\_bck = codec\_clk\_adc / 6 when i2s2\_bck\_div and i2s2\_bck\_div\_ld are not configured | 10'h5 |
| i2s2\_slot\_lrck | 15:9 | RW | The counter of empty cycles after all the slots are transmitted to get specific sample rate | 7'h0 |
| i2s2\_longsync | 8:8 | RW | longsync enable | 1'h0 |
| i2s2\_slotnum | 7:0 | RW | slots number to be transferred in one frame | 8'h0 |

### APC\_TX\_CH0\_L\_DATA

Offset: 100

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| tx\_ch0\_l\_data | 31:0 | RW | TX Channel 0 L data input | 32'h0 |

### APC\_TX\_CH0\_R\_DATA

Offset: 104

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| tx\_ch0\_r\_data | 31:0 | RW | TX Channel 0 R data input | 32'h0 |

### APC\_TX\_CH1\_L\_DATA

Offset: 108

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| tx\_ch1\_l\_data | 31:0 | RW | TX Channel 1 L data input | 32'h0 |

### APC\_TX\_CH1\_R\_DATA

Offset: 10C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| tx\_ch1\_r\_data | 31:0 | RW | TX Channel 1 R data input | 32'h0 |

### APC\_RX\_CH0\_L\_DATA

Offset: 110

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| rx\_ch0\_l\_data | 31:0 | ROI | RX Channel 0 L data output | 32'h0 |

### APC\_RX\_CH0\_R\_DATA

Offset: 114

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| rx\_ch0\_r\_data | 31:0 | ROI | RX Channel 0 R data output | 32'h0 |

### APC\_RX\_CH1\_L\_DATA

Offset: 118

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| rx\_ch1\_l\_data | 31:0 | ROI | RX Channel 1 L data output | 32'h0 |

### APC\_RX\_CH1\_R\_DATA

Offset: 11C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| rx\_ch1\_r\_data | 31:0 | ROI | RX Channel 1 R data output | 32'h0 |

### APC\_RX\_CH2\_L\_DATA

Offset: 120

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| rx\_ch2\_l\_data | 31:0 | ROI | RX Channel 2 L data output | 32'h0 |

### APC\_RX\_CH2\_R\_DATA

Offset: 124

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| rx\_ch2\_r\_data | 31:0 | ROI | RX Channel 2 R data output | 32'h0 |

### APC\_INTR\_TX\_MSK

Offset: 128

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:22 | RO | Not used | 10'h0 |
| tx\_ch0\_fifo\_vld\_msk | 21:21 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| tx\_ch1\_fifo\_vld\_msk | 20:20 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| tx\_ch1\_r\_dma\_req\_msk | 19:19 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| tx\_ch1\_r\_fifo\_ovflow\_msk | 18:18 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| tx\_ch1\_r\_fifo\_unflow\_msk | 17:17 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| tx\_ch1\_r\_fifo\_ful\_msk | 16:16 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| tx\_ch1\_r\_fifo\_emp\_msk | 15:15 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| tx\_ch1\_l\_dma\_req\_msk | 14:14 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| tx\_ch1\_l\_fifo\_ovflow\_msk | 13:13 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| tx\_ch1\_l\_fifo\_unflow\_msk | 12:12 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| tx\_ch1\_l\_fifo\_ful\_msk | 11:11 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| tx\_ch1\_l\_fifo\_emp\_msk | 10:10 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| tx\_ch0\_r\_dma\_req\_msk | 9:9 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| tx\_ch0\_r\_fifo\_ovflow\_msk | 8:8 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| tx\_ch0\_r\_fifo\_unflow\_msk | 7:7 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| tx\_ch0\_r\_fifo\_ful\_msk | 6:6 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| tx\_ch0\_r\_fifo\_emp\_msk | 5:5 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| tx\_ch0\_l\_dma\_req\_msk | 4:4 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| tx\_ch0\_l\_fifo\_ovflow\_msk | 3:3 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| tx\_ch0\_l\_fifo\_unflow\_msk | 2:2 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| tx\_ch0\_l\_fifo\_ful\_msk | 1:1 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| tx\_ch0\_l\_fifo\_emp\_msk | 0:0 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |

### APC\_INTR\_RX\_MSK

Offset: 12C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:30 | RO | Not used | 2'h0 |
| rx\_ch2\_r\_dma\_req\_msk | 29:29 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| rx\_ch2\_r\_fifo\_ovflow\_msk | 28:28 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| rx\_ch2\_r\_fifo\_unflow\_msk | 27:27 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| rx\_ch2\_r\_fifo\_ful\_msk | 26:26 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| rx\_ch2\_r\_fifo\_emp\_msk | 25:25 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| rx\_ch2\_l\_dma\_req\_msk | 24:24 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| rx\_ch2\_l\_fifo\_ovflow\_msk | 23:23 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| rx\_ch2\_l\_fifo\_unflow\_msk | 22:22 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| rx\_ch2\_l\_fifo\_ful\_msk | 21:21 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| rx\_ch2\_l\_fifo\_emp\_msk | 20:20 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| rx\_ch1\_r\_dma\_req\_msk | 19:19 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| rx\_ch1\_r\_fifo\_ovflow\_msk | 18:18 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| rx\_ch1\_r\_fifo\_unflow\_msk | 17:17 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| rx\_ch1\_r\_fifo\_ful\_msk | 16:16 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| rx\_ch1\_r\_fifo\_emp\_msk | 15:15 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| rx\_ch1\_l\_dma\_req\_msk | 14:14 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| rx\_ch1\_l\_fifo\_ovflow\_msk | 13:13 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| rx\_ch1\_l\_fifo\_unflow\_msk | 12:12 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| rx\_ch1\_l\_fifo\_ful\_msk | 11:11 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| rx\_ch1\_l\_fifo\_emp\_msk | 10:10 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| rx\_ch0\_r\_dma\_req\_msk | 9:9 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| rx\_ch0\_r\_fifo\_ovflow\_msk | 8:8 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| rx\_ch0\_r\_fifo\_unflow\_msk | 7:7 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| rx\_ch0\_r\_fifo\_ful\_msk | 6:6 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| rx\_ch0\_r\_fifo\_emp\_msk | 5:5 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| rx\_ch0\_l\_dma\_req\_msk | 4:4 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| rx\_ch0\_l\_fifo\_ovflow\_msk | 3:3 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| rx\_ch0\_l\_fifo\_unflow\_msk | 2:2 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| rx\_ch0\_l\_fifo\_ful\_msk | 1:1 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |
| rx\_ch0\_l\_fifo\_emp\_msk | 0:0 | RW | disable interrupt source 0: enable 1: disable | 1'h1 |

### APC\_INTR\_TX\_CLR

Offset: 130

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:22 | RO | Not used | 10'h0 |
| tx\_ch0\_fifo\_vld\_clr | 21:21 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| tx\_ch1\_fifo\_vld\_clr | 20:20 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| tx\_ch1\_r\_dma\_req\_clr | 19:19 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| tx\_ch1\_r\_fifo\_ovflow\_clr | 18:18 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| tx\_ch1\_r\_fifo\_unflow\_clr | 17:17 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| tx\_ch1\_r\_fifo\_ful\_clr | 16:16 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| tx\_ch1\_r\_fifo\_emp\_clr | 15:15 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| tx\_ch1\_l\_dma\_req\_clr | 14:14 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| tx\_ch1\_l\_fifo\_ovflow\_clr | 13:13 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| tx\_ch1\_l\_fifo\_unflow\_clr | 12:12 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| tx\_ch1\_l\_fifo\_ful\_clr | 11:11 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| tx\_ch1\_l\_fifo\_emp\_clr | 10:10 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| tx\_ch0\_r\_dma\_req\_clr | 9:9 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| tx\_ch0\_r\_fifo\_ovflow\_clr | 8:8 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| tx\_ch0\_r\_fifo\_unflow\_clr | 7:7 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| tx\_ch0\_r\_fifo\_ful\_clr | 6:6 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| tx\_ch0\_r\_fifo\_emp\_clr | 5:5 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| tx\_ch0\_l\_dma\_req\_clr | 4:4 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| tx\_ch0\_l\_fifo\_ovflow\_clr | 3:3 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| tx\_ch0\_l\_fifo\_unflow\_clr | 2:2 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| tx\_ch0\_l\_fifo\_ful\_clr | 1:1 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| tx\_ch0\_l\_fifo\_emp\_clr | 0:0 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |

### APC\_INTR\_RX\_CLR

Offset: 134

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:30 | RO | Not used | 2'h0 |
| rx\_ch2\_r\_dma\_req\_clr | 29:29 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| rx\_ch2\_r\_fifo\_ovflow\_clr | 28:28 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| rx\_ch2\_r\_fifo\_unflow\_clr | 27:27 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| rx\_ch2\_r\_fifo\_ful\_clr | 26:26 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| rx\_ch2\_r\_fifo\_emp\_clr | 25:25 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| rx\_ch2\_l\_dma\_req\_clr | 24:24 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| rx\_ch2\_l\_fifo\_ovflow\_clr | 23:23 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| rx\_ch2\_l\_fifo\_unflow\_clr | 22:22 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| rx\_ch2\_l\_fifo\_ful\_clr | 21:21 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| rx\_ch2\_l\_fifo\_emp\_clr | 20:20 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| rx\_ch1\_r\_dma\_req\_clr | 19:19 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| rx\_ch1\_r\_fifo\_ovflow\_clr | 18:18 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| rx\_ch1\_r\_fifo\_unflow\_clr | 17:17 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| rx\_ch1\_r\_fifo\_ful\_clr | 16:16 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| rx\_ch1\_r\_fifo\_emp\_clr | 15:15 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| rx\_ch1\_l\_dma\_req\_clr | 14:14 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| rx\_ch1\_l\_fifo\_ovflow\_clr | 13:13 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| rx\_ch1\_l\_fifo\_unflow\_clr | 12:12 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| rx\_ch1\_l\_fifo\_ful\_clr | 11:11 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| rx\_ch1\_l\_fifo\_emp\_clr | 10:10 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| rx\_ch0\_r\_dma\_req\_clr | 9:9 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| rx\_ch0\_r\_fifo\_ovflow\_clr | 8:8 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| rx\_ch0\_r\_fifo\_unflow\_clr | 7:7 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| rx\_ch0\_r\_fifo\_ful\_clr | 6:6 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| rx\_ch0\_r\_fifo\_emp\_clr | 5:5 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| rx\_ch0\_l\_dma\_req\_clr | 4:4 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| rx\_ch0\_l\_fifo\_ovflow\_clr | 3:3 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| rx\_ch0\_l\_fifo\_unflow\_clr | 2:2 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| rx\_ch0\_l\_fifo\_ful\_clr | 1:1 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |
| rx\_ch0\_l\_fifo\_emp\_clr | 0:0 | W1C | clear interrupt source, write 1 generate clear pluse | 1'h0 |

### APC\_INTR\_TX\_IRSR

Offset: 138

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:22 | RO | Not used | 10'h0 |
| tx\_ch0\_fifo\_vld\_irsr | 21:21 | ROI | interrupt status before mask | 1'h0 |
| tx\_ch1\_fifo\_vld\_irsr | 20:20 | ROI | interrupt status before mask | 1'h0 |
| tx\_ch1\_r\_dma\_req\_irsr | 19:19 | ROI | interrupt status before mask | 1'h0 |
| tx\_ch1\_r\_fifo\_ovflow\_irsr | 18:18 | ROI | interrupt status before mask | 1'h0 |
| tx\_ch1\_r\_fifo\_unflow\_irsr | 17:17 | ROI | interrupt status before mask | 1'h0 |
| tx\_ch1\_r\_fifo\_ful\_irsr | 16:16 | ROI | interrupt status before mask | 1'h0 |
| tx\_ch1\_r\_fifo\_emp\_irsr | 15:15 | ROI | interrupt status before mask | 1'h0 |
| tx\_ch1\_l\_dma\_req\_irsr | 14:14 | ROI | interrupt status before mask | 1'h0 |
| tx\_ch1\_l\_fifo\_ovflow\_irsr | 13:13 | ROI | interrupt status before mask | 1'h0 |
| tx\_ch1\_l\_fifo\_unflow\_irsr | 12:12 | ROI | interrupt status before mask | 1'h0 |
| tx\_ch1\_l\_fifo\_ful\_irsr | 11:11 | ROI | interrupt status before mask | 1'h0 |
| tx\_ch1\_l\_fifo\_emp\_irsr | 10:10 | ROI | interrupt status before mask | 1'h0 |
| tx\_ch0\_r\_dma\_req\_irsr | 9:9 | ROI | interrupt status before mask | 1'h0 |
| tx\_ch0\_r\_fifo\_ovflow\_irsr | 8:8 | ROI | interrupt status before mask | 1'h0 |
| tx\_ch0\_r\_fifo\_unflow\_irsr | 7:7 | ROI | interrupt status before mask | 1'h0 |
| tx\_ch0\_r\_fifo\_ful\_irsr | 6:6 | ROI | interrupt status before mask | 1'h0 |
| tx\_ch0\_r\_fifo\_emp\_irsr | 5:5 | ROI | interrupt status before mask | 1'h0 |
| tx\_ch0\_l\_dma\_req\_irsr | 4:4 | ROI | interrupt status before mask | 1'h0 |
| tx\_ch0\_l\_fifo\_ovflow\_irsr | 3:3 | ROI | interrupt status before mask | 1'h0 |
| tx\_ch0\_l\_fifo\_unflow\_irsr | 2:2 | ROI | interrupt status before mask | 1'h0 |
| tx\_ch0\_l\_fifo\_ful\_irsr | 1:1 | ROI | interrupt status before mask | 1'h0 |
| tx\_ch0\_l\_fifo\_emp\_irsr | 0:0 | ROI | interrupt status before mask | 1'h0 |

### APC\_INTR\_RX\_IRSR

Offset: 13C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:30 | RO | Not used | 2'h0 |
| rx\_ch2\_r\_dma\_req\_irsr | 29:29 | ROI | interrupt status before mask | 1'h0 |
| rx\_ch2\_r\_fifo\_ovflow\_irsr | 28:28 | ROI | interrupt status before mask | 1'h0 |
| rx\_ch2\_r\_fifo\_unflow\_irsr | 27:27 | ROI | interrupt status before mask | 1'h0 |
| rx\_ch2\_r\_fifo\_ful\_irsr | 26:26 | ROI | interrupt status before mask | 1'h0 |
| rx\_ch2\_r\_fifo\_emp\_irsr | 25:25 | ROI | interrupt status before mask | 1'h0 |
| rx\_ch2\_l\_dma\_req\_irsr | 24:24 | ROI | interrupt status before mask | 1'h0 |
| rx\_ch2\_l\_fifo\_ovflow\_irsr | 23:23 | ROI | interrupt status before mask | 1'h0 |
| rx\_ch2\_l\_fifo\_unflow\_irsr | 22:22 | ROI | interrupt status before mask | 1'h0 |
| rx\_ch2\_l\_fifo\_ful\_irsr | 21:21 | ROI | interrupt status before mask | 1'h0 |
| rx\_ch2\_l\_fifo\_emp\_irsr | 20:20 | ROI | interrupt status before mask | 1'h0 |
| rx\_ch1\_r\_dma\_req\_irsr | 19:19 | ROI | interrupt status before mask | 1'h0 |
| rx\_ch1\_r\_fifo\_ovflow\_irsr | 18:18 | ROI | interrupt status before mask | 1'h0 |
| rx\_ch1\_r\_fifo\_unflow\_irsr | 17:17 | ROI | interrupt status before mask | 1'h0 |
| rx\_ch1\_r\_fifo\_ful\_irsr | 16:16 | ROI | interrupt status before mask | 1'h0 |
| rx\_ch1\_r\_fifo\_emp\_irsr | 15:15 | ROI | interrupt status before mask | 1'h0 |
| rx\_ch1\_l\_dma\_req\_irsr | 14:14 | ROI | interrupt status before mask | 1'h0 |
| rx\_ch1\_l\_fifo\_ovflow\_irsr | 13:13 | ROI | interrupt status before mask | 1'h0 |
| rx\_ch1\_l\_fifo\_unflow\_irsr | 12:12 | ROI | interrupt status before mask | 1'h0 |
| rx\_ch1\_l\_fifo\_ful\_irsr | 11:11 | ROI | interrupt status before mask | 1'h0 |
| rx\_ch1\_l\_fifo\_emp\_irsr | 10:10 | ROI | interrupt status before mask | 1'h0 |
| rx\_ch0\_r\_dma\_req\_irsr | 9:9 | ROI | interrupt status before mask | 1'h0 |
| rx\_ch0\_r\_fifo\_ovflow\_irsr | 8:8 | ROI | interrupt status before mask | 1'h0 |
| rx\_ch0\_r\_fifo\_unflow\_irsr | 7:7 | ROI | interrupt status before mask | 1'h0 |
| rx\_ch0\_r\_fifo\_ful\_irsr | 6:6 | ROI | interrupt status before mask | 1'h0 |
| rx\_ch0\_r\_fifo\_emp\_irsr | 5:5 | ROI | interrupt status before mask | 1'h0 |
| rx\_ch0\_l\_dma\_req\_irsr | 4:4 | ROI | interrupt status before mask | 1'h0 |
| rx\_ch0\_l\_fifo\_ovflow\_irsr | 3:3 | ROI | interrupt status before mask | 1'h0 |
| rx\_ch0\_l\_fifo\_unflow\_irsr | 2:2 | ROI | interrupt status before mask | 1'h0 |
| rx\_ch0\_l\_fifo\_ful\_irsr | 1:1 | ROI | interrupt status before mask | 1'h0 |
| rx\_ch0\_l\_fifo\_emp\_irsr | 0:0 | ROI | interrupt status before mask | 1'h0 |

### APC\_INTR\_TX\_ISR

Offset: 140

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:22 | RO | Not used | 10'h0 |
| tx\_ch0\_fifo\_vld\_isr | 21:21 | ROI | interrupt status after mask | 1'h0 |
| tx\_ch1\_fifo\_vld\_isr | 20:20 | ROI | interrupt status after mask | 1'h0 |
| tx\_ch1\_r\_dma\_req\_isr | 19:19 | ROI | interrupt status after mask | 1'h0 |
| tx\_ch1\_r\_fifo\_ovflow\_isr | 18:18 | ROI | interrupt status after mask | 1'h0 |
| tx\_ch1\_r\_fifo\_unflow\_isr | 17:17 | ROI | interrupt status after mask | 1'h0 |
| tx\_ch1\_r\_fifo\_ful\_isr | 16:16 | ROI | interrupt status after mask | 1'h0 |
| tx\_ch1\_r\_fifo\_emp\_isr | 15:15 | ROI | interrupt status after mask | 1'h0 |
| tx\_ch1\_l\_dma\_req\_isr | 14:14 | ROI | interrupt status after mask | 1'h0 |
| tx\_ch1\_l\_fifo\_ovflow\_isr | 13:13 | ROI | interrupt status after mask | 1'h0 |
| tx\_ch1\_l\_fifo\_unflow\_isr | 12:12 | ROI | interrupt status after mask | 1'h0 |
| tx\_ch1\_l\_fifo\_ful\_isr | 11:11 | ROI | interrupt status after mask | 1'h0 |
| tx\_ch1\_l\_fifo\_emp\_isr | 10:10 | ROI | interrupt status after mask | 1'h0 |
| tx\_ch0\_r\_dma\_req\_isr | 9:9 | ROI | interrupt status after mask | 1'h0 |
| tx\_ch0\_r\_fifo\_ovflow\_isr | 8:8 | ROI | interrupt status after mask | 1'h0 |
| tx\_ch0\_r\_fifo\_unflow\_isr | 7:7 | ROI | interrupt status after mask | 1'h0 |
| tx\_ch0\_r\_fifo\_ful\_isr | 6:6 | ROI | interrupt status after mask | 1'h0 |
| tx\_ch0\_r\_fifo\_emp\_isr | 5:5 | ROI | interrupt status after mask | 1'h0 |
| tx\_ch0\_l\_dma\_req\_isr | 4:4 | ROI | interrupt status after mask | 1'h0 |
| tx\_ch0\_l\_fifo\_ovflow\_isr | 3:3 | ROI | interrupt status after mask | 1'h0 |
| tx\_ch0\_l\_fifo\_unflow\_isr | 2:2 | ROI | interrupt status after mask | 1'h0 |
| tx\_ch0\_l\_fifo\_ful\_isr | 1:1 | ROI | interrupt status after mask | 1'h0 |
| tx\_ch0\_l\_fifo\_emp\_isr | 0:0 | ROI | interrupt status after mask | 1'h0 |

### APC\_INTR\_RX\_ISR

Offset: 144

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:30 | RO | Not used | 2'h0 |
| rx\_ch2\_r\_dma\_req\_isr | 29:29 | ROI | interrupt status after mask | 1'h0 |
| rx\_ch2\_r\_fifo\_ovflow\_isr | 28:28 | ROI | interrupt status after mask | 1'h0 |
| rx\_ch2\_r\_fifo\_unflow\_isr | 27:27 | ROI | interrupt status after mask | 1'h0 |
| rx\_ch2\_r\_fifo\_ful\_isr | 26:26 | ROI | interrupt status after mask | 1'h0 |
| rx\_ch2\_r\_fifo\_emp\_isr | 25:25 | ROI | interrupt status after mask | 1'h0 |
| rx\_ch2\_l\_dma\_req\_isr | 24:24 | ROI | interrupt status after mask | 1'h0 |
| rx\_ch2\_l\_fifo\_ovflow\_isr | 23:23 | ROI | interrupt status after mask | 1'h0 |
| rx\_ch2\_l\_fifo\_unflow\_isr | 22:22 | ROI | interrupt status after mask | 1'h0 |
| rx\_ch2\_l\_fifo\_ful\_isr | 21:21 | ROI | interrupt status after mask | 1'h0 |
| rx\_ch2\_l\_fifo\_emp\_isr | 20:20 | ROI | interrupt status after mask | 1'h0 |
| rx\_ch1\_r\_dma\_req\_isr | 19:19 | ROI | interrupt status after mask | 1'h0 |
| rx\_ch1\_r\_fifo\_ovflow\_isr | 18:18 | ROI | interrupt status after mask | 1'h0 |
| rx\_ch1\_r\_fifo\_unflow\_isr | 17:17 | ROI | interrupt status after mask | 1'h0 |
| rx\_ch1\_r\_fifo\_ful\_isr | 16:16 | ROI | interrupt status after mask | 1'h0 |
| rx\_ch1\_r\_fifo\_emp\_isr | 15:15 | ROI | interrupt status after mask | 1'h0 |
| rx\_ch1\_l\_dma\_req\_isr | 14:14 | ROI | interrupt status after mask | 1'h0 |
| rx\_ch1\_l\_fifo\_ovflow\_isr | 13:13 | ROI | interrupt status after mask | 1'h0 |
| rx\_ch1\_l\_fifo\_unflow\_isr | 12:12 | ROI | interrupt status after mask | 1'h0 |
| rx\_ch1\_l\_fifo\_ful\_isr | 11:11 | ROI | interrupt status after mask | 1'h0 |
| rx\_ch1\_l\_fifo\_emp\_isr | 10:10 | ROI | interrupt status after mask | 1'h0 |
| rx\_ch0\_r\_dma\_req\_isr | 9:9 | ROI | interrupt status after mask | 1'h0 |
| rx\_ch0\_r\_fifo\_ovflow\_isr | 8:8 | ROI | interrupt status after mask | 1'h0 |
| rx\_ch0\_r\_fifo\_unflow\_isr | 7:7 | ROI | interrupt status after mask | 1'h0 |
| rx\_ch0\_r\_fifo\_ful\_isr | 6:6 | ROI | interrupt status after mask | 1'h0 |
| rx\_ch0\_r\_fifo\_emp\_isr | 5:5 | ROI | interrupt status after mask | 1'h0 |
| rx\_ch0\_l\_dma\_req\_isr | 4:4 | ROI | interrupt status after mask | 1'h0 |
| rx\_ch0\_l\_fifo\_ovflow\_isr | 3:3 | ROI | interrupt status after mask | 1'h0 |
| rx\_ch0\_l\_fifo\_unflow\_isr | 2:2 | ROI | interrupt status after mask | 1'h0 |
| rx\_ch0\_l\_fifo\_ful\_isr | 1:1 | ROI | interrupt status after mask | 1'h0 |
| rx\_ch0\_l\_fifo\_emp\_isr | 0:0 | ROI | interrupt status after mask | 1'h0 |

### I2S\_SYNC\_MODE

Offset: 148

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:2 | RO | Not used | 30'h0 |
| i2s\_sync\_mode | 1:0 | RW | 0: disbale sync mode 1: enable i2s0 and i2s1 sync mode 2: enable i2s0, i2s1 and i2s2 sync mode | 2'h0 |

### APC\_RSVD\_RW\_REG0

Offset: 150

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| apc\_rsvd\_rw\_reg0 | 31:0 | RW | Reserved for future use | 32'h0 |

### APC\_RSVD\_RW\_REG1

Offset: 154

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| apc\_rsvd\_rw\_reg1 | 31:0 | RW | Reserved for future use | 32'h0 |

### APC\_RSVD\_RO\_REG0

Offset: 158

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| apc\_rsvd\_ro\_reg0 | 31:0 | ROI | Reserved for future use | 32'h0 |

### APC\_RSVD\_RO\_REG1

Offset: 15C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| apc\_rsvd\_ro\_reg1 | 31:0 | ROI | Reserved for future use | 32'h0 |

# VAD

## Introduction

## Main Features

VAD属于CP的音频模块，这个部分不对用户开放

## Function Description

## Basic Block Diagram

## VAD Register

### CTRL

Offset: 000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RO | Not used | 16'h0 |
| VADDEC\_RESET | 15:15 | RW | 1: reset the VAD DEC logic | 1'h0 |
| MEM\_POWERDN | 14:14 | RW | 1: memory module power down | 1'h0 |
| MEM\_POWERSAVE\_SLV0 | 13:13 | RW | 1: memory module power save mode asserted. | 1'h0 |
| SQR\_EN | 12:12 | RW | SQR\_EN | 1'h1 |
| POSTAMP\_EN | 11:11 | RW | POSTAMP\_EN | 1'h1 |
| IIR5\_EN | 10:10 | RW | IIR5 Enable | 1'h1 |
| IIR4\_EN | 9:9 | RW | IIR4 Enable | 1'h1 |
| IIR3\_EN | 8:8 | RW | IIR3 Enable | 1'h1 |
| IIR2\_EN | 7:7 | RW | IIR2 Enable | 1'h1 |
| IIR1\_EN | 6:6 | RW | IIR1 Enable | 1'h1 |
| PREAMP\_EN | 5:5 | RW | Pre-amplification Enable | 1'h1 |
| Reserved | 4:4 | RO | Not used | 1'h0 |
| SRAM\_WR\_STOP | 3:3 | RW | 1: stop write ADC data into ring buf SRAM. Enable store the last data to ring buffer, and assert vad\_finish HIGH for APC taking over ADC data reading.  0: keep writing ADC data to ring buffer | 1'h0 |
| VADCH\_MODE\_SEL | 2:2 | RW | 0: 1 channel mode in VAD 1: 2 channel mode in VAD | 1'h1 |
| VAD\_EN | 1:1 | RW | 1: VAD detection enabled 0: VAD detection disabled | 1'h0 |
| AUDIO\_PATH\_EN | 0:0 | RW | 1: Audio path is enabled. The audio path will do at least protocal conversion from ADC to APC FIFOs. 0: Audio path is disabled. No data will be sent to APC. | 1'h0 |

### STATE

Offset: 004

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:4 | RO | Not used | 28'h0 |
| END\_ADDR\_VALID | 3:3 | W1C | 1: the end addr is valid | 1'h0 |
| IIR\_ACCUM\_OVERFLOW | 2:2 | W1C | 1: IIR overflow found | 1'h0 |
| MEM\_AP\_BYPASSED | 1:1 | W1C | obsolete 1: Internal memory is bypassed from audio dataflow path 0: Internal memory is used in audio dataflow path | 1'h0 |
| VAD\_FOUND | 0:0 | W1C | 1: VAD Found  0: No VAD found yet | 1'h0 |

### INTEN

Offset: 008

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:2 | RO | Not used | 30'h0 |
| MEM\_AP\_BYPASSED\_INT\_EN | 1:1 | RW | MEM\_AP\_BYPASSED interrupt enable | 1'h0 |
| VAD\_FOUND\_INT\_EN | 0:0 | RW | VAD\_FOUND interrupt enable | 1'h0 |

### TRACBK\_LEN

Offset: 00C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:15 | RO | Not used | 17'h0 |
| TRACBK\_LEN | 14:0 | RW | The maximal length of buffer samples for each channel that need trace-back to be sent to APC FIFOs. | 15'hffffff80f0 |

### FILTCTRL

Offset: 010

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:28 | RO | Not used | 4'h0 |
| POST\_AMPL\_BY\_8 | 27:27 | RW | 1： When asserted, amplified the CIC output by 8.  0： No amplification. | 1'h0 |
| POST\_AMPL\_RATIO | 26:24 | RW | Further amplification besides the possible x8 stage (always positive, no integer part) Only 8 options: ~8/8 (3’b000), ~7/8(3’b111), ~6/8(3’b110), ~5/8(3’b101), ~4/8(3b100), ~3/8(3’b011), ~2/8(3’b010), ~1/8(3’b001). | 3'h0 |
| PRE\_AMPL\_BY\_8 | 23:23 | RW | 1： When asserted, amplified the CIC output by 8.  0： No amplification. | 1'h0 |
| PRE\_AMPL\_RATIO | 22:20 | RW | Further amplification besides the possible x8 stage (always positive, no integer part). Only 8 options: ~8/8 (3’b000), ~7/8(3’b111), ~6/8(3’b110), ~5/8(3’b101), ~4/8(3b100), ~3/8(3’b011), ~2/8(3’b010), ~1/8(3’b001). | 3'h0 |
| IIR5\_ORDER | 19:18 | RW | IIR5 Filter Order | 2'h2 |
| IIR4\_ORDER | 17:16 | RW | IIR4 Filter Order | 2'h2 |
| IIR3\_ORDER | 15:14 | RW | IIR3 Filter Order | 2'h2 |
| IIR2\_ORDER | 13:12 | RW | IIR2 Filter Order | 2'h2 |
| IIR1\_ORDER | 11:10 | RW | IIR1 Filter Order | 2'h2 |
| IIR5\_B\_COEF\_SHFT | 9:8 | RW | Whether the IIR5\_Bn coefs are scaled. 2’b00: not scaled. 2’b01: scaled to 1/8 of the original values 2’b11: scaled to 8x of the original value Others: Reserved | 2'h0 |
| IIR4\_B\_COEF\_SHFT | 7:6 | RW | Whether the IIR4\_Bn coefs are scaled. 2’b00: not scaled. 2’b01: scaled to 1/8 of the original values 2’b11: scaled to 8x of the original value Others: Reserved | 2'h0 |
| IIR3\_B\_COEF\_SHFT | 5:4 | RW | Whether the IIR3\_Bn coefs are scaled. 2’b00: not scaled. 2’b01: scaled to 1/8 of the original values 2’b11: scaled to 8x of the original value Others: Reserved | 2'h0 |
| IIR2\_B\_COEF\_SHFT | 3:2 | RW | Whether the IIR2\_Bn coefs are scaled. 2’b00: not scaled. 2’b01: scaled to 1/8 of the original values 2’b11: scaled to 8x of the original value Others: Reserved | 2'h0 |
| IIR1\_B\_COEF\_SHFT | 1:0 | RW | Whether the IIR1\_Bn coefs are scaled. 2’b00: not scaled. 2’b01: scaled to 1/8 of the original values 2’b11: scaled to 8x of the original value Others: Reserved | 2'h0 |

### BLKCTRL

Offset: 014

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:10 | RO | Not used | 22'h0 |
| BLK\_TYPE | 9:9 | RW | 1:3 subblocks 0:2 subblocks | 1'h1 |
| SUBBLK\_SAMP\_LEN | 8:0 | RW | sub-block sample length | 9'h50 |

### ZCHTOL

Offset: 018

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RO | Not used | 16'h0 |
| ZCH\_TOL | 15:0 | RW | ZCH\_TOL | 16'h31 |

### ZCH\_MINAMP

Offset: 01C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RO | Not used | 16'h0 |
| ZCH\_MINAMP | 15:0 | RW | ZCH\_MINAMP | 16'h190 |

### ZCH\_MAXAMP

Offset: 020

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RO | Not used | 16'h0 |
| ZCH\_MAXAMP | 15:0 | RW | ZCH\_MAXAMP | 16'h193 |

### AMP\_L

Offset: 024

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RO | Not used | 16'h0 |
| AMP\_L | 15:0 | RW | AMP\_L | 16'h666 |

### AMP\_H

Offset: 028

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RO | Not used | 16'h0 |
| AMP\_H | 15:0 | RW | AMP\_H | 16'hb85 |

### ZCR\_L

Offset: 02C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RO | Not used | 16'h0 |
| ZCR\_L | 15:0 | RW | ZCR\_L | 16'h5 |

### ZCR\_H

Offset: 030

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RO | Not used | 16'h0 |
| ZCR\_H | 15:0 | RW | ZCR\_H | 16'h50 |

### VADDEC\_MISC

Offset: 034

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:29 | RO | Not used | 3'h0 |
| CRITR8\_EN | 28:28 | RW | CRITR8\_EN | 1'h1 |
| CRITR7\_EN | 27:27 | RW | CRITR7\_EN | 1'h1 |
| CRITR6\_EN | 26:26 | RW | CRITR6\_EN | 1'h1 |
| CRITR5\_EN | 25:25 | RW | CRITR5\_EN | 1'h1 |
| CRITR4\_EN | 24:24 | RW | CRITR4\_EN | 1'h1 |
| CRITR3\_EN | 23:23 | RW | CRITR3\_EN | 1'h1 |
| CRITR2\_EN | 22:22 | RW | CRITR2\_EN | 1'h1 |
| CRITR1\_EN | 21:21 | RW | CRITR1\_EN | 1'h1 |
| CRITR0\_EN | 20:20 | RW | CRITR0\_EN | 1'h1 |
| MINLEN\_TRANS\_POS | 19:15 | RW | MINLEN\_TRANS\_POS | 5'h5 |
| MINLEN\_TRANS\_NEG | 14:10 | RW | MINLEN\_TRANS\_NEG | 5'h3 |
| MINLEN | 9:5 | RW | MINLEN | 5'h3 |
| MAXSILENCE | 4:0 | RW | MAXSILENCE | 5'h14 |

### VAD\_IIR1\_B1

Offset: 038

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| VAD\_IIR1\_B1 | 31:0 | RW | Not used | 32'h6da3 |

### VAD\_IIR1\_B2

Offset: 03C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| VAD\_IIR1\_B2 | 31:0 | RW | Not used | 32'hffffff24ba |

### VAD\_IIR1\_B3

Offset: 040

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| VAD\_IIR1\_B3 | 31:0 | RW | Not used | 32'h6da3 |

### VAD\_IIR1\_MA2

Offset: 044

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| VAD\_IIR1\_MA2 | 31:0 | RW | Not used | 32'hf5b0 |

### VAD\_IIR1\_MA3

Offset: 048

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| VAD\_IIR1\_MA3 | 31:0 | RW | Not used | 32'hffffff89a0 |

### VAD\_IIR2\_B1

Offset: 04C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| VAD\_IIR2\_B1 | 31:0 | RW | Not used | 32'h6da3 |

### VAD\_IIR2\_B2

Offset: 050

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| VAD\_IIR2\_B2 | 31:0 | RW | Not used | 32'hffffff24ba |

### VAD\_IIR2\_B3

Offset: 054

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| VAD\_IIR2\_B3 | 31:0 | RW | Not used | 32'h6da3 |

### VAD\_IIR2\_MA2

Offset: 058

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| VAD\_IIR2\_MA2 | 31:0 | RW | Not used | 32'hf5b0 |

### VAD\_IIR2\_MA3

Offset: 05C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| VAD\_IIR2\_MA3 | 31:0 | RW | Not used | 32'hffffff89a0 |

### VAD\_IIR3\_B1

Offset: 060

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| VAD\_IIR3\_B1 | 31:0 | RW | Not used | 32'h313c |

### VAD\_IIR3\_B2

Offset: 064

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| VAD\_IIR3\_B2 | 31:0 | RW | Not used | 32'h6279 |

### VAD\_IIR3\_B3

Offset: 068

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| VAD\_IIR3\_B3 | 31:0 | RW | Not used | 32'h313c |

### VAD\_IIR3\_MA2

Offset: 06C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| VAD\_IIR3\_MA2 | 31:0 | RW | Not used | 32'h1406 |

### VAD\_IIR3\_MA3

Offset: 070

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| VAD\_IIR3\_MA3 | 31:0 | RW | Not used | 32'hffffff8f01 |

### VAD\_IIR4\_B1

Offset: 074

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| VAD\_IIR4\_B1 | 31:0 | RW | Not used | 32'h2211 |

### VAD\_IIR4\_B2

Offset: 078

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| VAD\_IIR4\_B2 | 31:0 | RW | Not used | 32'h4422 |

### VAD\_IIR4\_B3

Offset: 07C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| VAD\_IIR4\_B3 | 31:0 | RW | Not used | 32'h2211 |

### VAD\_IIR4\_MA2

Offset: 080

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| VAD\_IIR4\_MA2 | 31:0 | RW | Not used | 32'h4ad5 |

### VAD\_IIR4\_MA3

Offset: 084

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| VAD\_IIR4\_MA3 | 31:0 | RW | Not used | 32'hfffffface7 |

### VAD\_IIR5\_B1

Offset: 088

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| VAD\_IIR5\_B1 | 31:0 | RW | Not used | 32'h8e2 |

### VAD\_IIR5\_B2

Offset: 08c

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| VAD\_IIR5\_B2 | 31:0 | RW | Not used | 32'h11c4 |

### VAD\_IIR5\_B3

Offset: 090

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| VAD\_IIR5\_B3 | 31:0 | RW | Not used | 32'h8e2 |

### VAD\_IIR5\_MA2

Offset: 094

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| VAD\_IIR5\_MA2 | 31:0 | RW | Not used | 32'h94f2 |

### VAD\_IIR5\_MA3

Offset: 098

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| VAD\_IIR5\_MA3 | 31:0 | RW | Not used | 32'hffffffc786 |

### START\_ADDR

Offset: 09C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:13 | RW | Not used | 19'h0 |
| START\_ADDR | 12:0 | RW | The valid start address for ring buffer data read.  It will be updated according the TRACBK\_LEN setting when vad\_found is valid.  Note: the start\_addr is valid only if vad\_found=1. Be sure to clear vad\_found so that start\_addr can be updated @ the next vad\_found. | 13'h0 |

### END\_ADDR

Offset: 0A0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:13 | RW | Not used | 19'h0 |
| END\_ADDR | 12:0 | RW | The valid end address for ring buffer data read. It's the addr of the last data written into ring buffer when SRAM\_WR\_STOP bit asserted by software. | 13'h0 |

### CUR\_ADDR

Offset: 0A4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:13 | RW | Not used | 19'h0 |
| CUR\_ADDR | 12:0 | RW | Current address. It's the addr of the sram write address when vad\_found goes HIGH. It's recommended to use this address to decide when SRAM\_WR\_STOP bit should be asserted. | 13'h0 |

# DW\_UART

## Introduction

IP的规格内容，需要PDF转成word放进来

## Main Features

## Function Description

## Basic Block Diagram

## DW\_UART Register

### RBR

Offset: 000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:9 | RO | Not used | 23'h0 |
| rbr\_lsb8 | 8:8 | RO |  | 1'h0 |
| rbr\_lsb7to0 | 7:0 | RO |  | 8'h0 |

### THR

Offset: 000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:9 | RO | Not used | 23'h0 |
| thr\_lsb8 | 8:8 | WO |  | 1'h0 |
| thr\_lsb7to0 | 7:0 | WO |  | 8'h0 |

### DLL

Offset: 000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:8 | RO | Not used | 24'h0 |
| dll | 7:0 | RW |  | 8'h0 |

### DLH

Offset: 004

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:8 | RO | Not used | 24'h0 |
| dlh | 7:0 | RW |  | 8'h0 |

### IER

Offset: 004

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:8 | RO | Not used | 24'h0 |
| ptime | 7:7 | RW |  | 1'h0 |
| Reserved | 6:5 | RW | Not used | 2'h0 |
| echt | 4:4 | RW | enable character time out interrupt | 1'b0 |
| edssi | 3:3 | RW |  | 1'h0 |
| elsi | 2:2 | RW |  | 1'h0 |
| etbei | 1:1 | RW |  | 1'h0 |
| erbfi | 0:0 | RW | enable receive data interrupt | 1'h0 |

### IIR

Offset: 008

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:8 | RO | Not used | 24'h0 |
| fifose | 7:6 | RO |  | 2'h0 |
| Reserved | 5:4 | RO | Not used | 2'h0 |
| iid | 3:0 | RO |  | 4'h0 |

### FCR

Offset: 008

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:8 | RO | Not used | 24'h0 |
| rcvr | 7:6 | WO |  | 2'h0 |
| tet | 5:4 | WO |  | 2'h0 |
| dmam | 3:3 | WO |  | 1'h0 |
| xfifor | 2:2 | WO |  | 1'h0 |
| rfifor | 1:1 | WO |  | 1'h0 |
| fifoe | 0:0 | WO |  | 1'h0 |

### LCR

Offset: 00C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:8 | RO | Not used | 24'h0 |
| dlab | 7:7 | RW |  | 1'h |
| bc | 6:6 | RW |  | 1'h |
| stick\_parity | 5:5 | RW |  | 1'h |
| eps | 4:4 | RW |  | 1'h0 |
| pen | 3:3 | RW |  | 1'h0 |
| stop | 2:2 | RW |  | 1'h0 |
| dls | 1:0 | RW |  | 2'h0 |

### MCR

Offset: 010

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:7 | RO | Not used | 25'h0 |
| sire | 6:6 | RW |  | 1'h |
| afce | 5:5 | RW |  | 1'h |
| lb | 4:4 | RW |  | 1'h |
| out2 | 3:3 | RW |  | 1'h |
| out1 | 2:2 | RW |  | 1'h |
| rts | 1:1 | RW |  | 1'h |
| dtr | 0:0 | RW |  | 1'h0 |

### LSR

Offset: 014

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:9 | RO | Not used | 23'h0 |
| addr\_rcvd | 8:8 | RO |  | 1'h |
| rfe | 7:7 | RO |  | 1'h |
| temt | 6:6 | RO |  | 1'h |
| thre | 5:5 | RO |  | 1'h |
| bi | 4:4 | RO |  | 1'h |
| fe | 3:3 | RO |  | 1'h0 |
| pen | 2:2 | RO |  | 1'h0 |
| oe | 1:1 | RO |  | 1'h0 |
| dr | 0:0 | RO |  | 1'h0 |

### MSR

Offset: 018

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:8 | RO | Not used | 24'h0 |
| dcd | 7:7 | RO |  | 1'h |
| ri | 6:6 | RO |  | 1'h |
| dsr | 5:5 | RO |  | 1'h |
| cts | 4:4 | RO |  | 1'h |
| ddcd | 3:3 | RO |  | 1'h0 |
| teri | 2:2 | RO |  | 1'h0 |
| ddsr | 1:1 | RO |  | 1'h0 |
| dcts | 0:0 | RO |  | 1'h0 |

### SCR

Offset: 01C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:8 | RO | Not used | 24'h0 |
| scr | 7:0 | RW |  | 8'h0 |

### LPDLL

Offset: 020

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:8 | RO | Not used | 24'h0 |
| lpdll | 7:0 | RW |  | 8'h0 |

### LPDLH

Offset: 024

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:8 | RO | Not used | 24'h0 |
| lpdlh | 7:0 | RW |  | 8'h0 |

### SRBR

Offset: 030

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:9 | RO | Not used | 23'h0 |
| srbr\_lsb8 | 8:8 | RO |  | 1'h0 |
| srbr\_lsb7to0 | 7:0 | RO |  | 8'h0 |

### STHR

Offset: 030

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:9 | RO | Not used | 23'h0 |
| sthr\_lsb8 | 8:8 | WO |  | 1'h0 |
| sthr\_lsb7to0 | 7:0 | WO |  | 8'h0 |

### FAR

Offset: 070

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO | Not used | 31'h0 |
| far | 0:0 | RW |  | 1'h0 |

### TFR

Offset: 074

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:8 | RO | Not used | 24'h0 |
| tfr | 7:0 | RO |  | 8'h0 |

### RFW

Offset: 078

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:10 | RO | Not used | 22'h0 |
| rffe | 9:9 | WO |  | 1'h0 |
| rfpe | 8:8 | WO |  | 1'h0 |
| rfwd | 7:0 | WO |  | 8'h0 |

### USR

Offset: 07C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:5 | RO | Not used | 27'h0 |
| rff | 4:4 | RO |  | 1'h0 |
| rfne | 3:3 | RO |  | 1'h0 |
| tfe | 2:2 | RO |  | 1'h0 |
| tfnf | 1:1 | RO |  | 1'h0 |
| busy | 0:0 | RO |  | 1'h0 |

### TFL

Offset: 080

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| tfl | 31:0 | RO |  | 32'h0 |

### RFL

Offset: 084

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| rfl | 31:0 | RO |  | 32'h0 |

### SRR

Offset: 088

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:3 | RO | Not used | 29'h0 |
| xfr | 2:2 | WO |  | 1'h0 |
| rfr | 1:1 | WO |  | 1'h0 |
| ur | 0:0 | WO |  | 1'h0 |

### SRTS

Offset: 08C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO | Not used | 31'h0 |
| srts | 0:0 | RW |  | 1'h0 |

### SBCR

Offset: 090

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO | Not used | 31'h0 |
| sbcr | 0:0 | RW |  | 1'h0 |

### SDMAM

Offset: 094

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO | Not used | 31'h0 |
| sdmam | 0:0 | RW |  | 1'h0 |

### SFE

Offset: 098

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO | Not used | 31'h0 |
| sfe | 0:0 | RW |  | 1'h0 |

### SRT

Offset: 09C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:2 | RO | Not used | 30'h0 |
| srt | 1:0 | RW |  | 2'h0 |

### STET

Offset: 0A0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:2 | RO | Not used | 30'h0 |
| stet | 1:0 | RW |  | 2'h0 |

### HTX

Offset: 0A4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO | Not used | 31'h0 |
| htx | 0:0 | RW |  | 1'h0 |

### DMASA

Offset: 0A8

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO | Not used | 31'h0 |
| dmasa | 0:0 | WO |  | 1'h0 |

### TCR

Offset: 0AC

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:5 | RO | Not used | 27'h0 |
| xfer\_mode | 4:3 | RW |  | 2'h0 |
| de\_pol | 2:2 | RW |  | 1'h0 |
| re\_pol | 1:1 | RW |  | 1'h0 |
| rs485\_en | 0:0 | RW |  | 1'h0 |

### DE\_EN

Offset: 0B0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO | Not used | 31'h0 |
| de\_en | 0:0 | RW |  | 1'h0 |

### RE\_EN

Offset: 0B4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO | Not used | 31'h0 |
| re\_en | 0:0 | RW |  | 1'h0 |

### DET

Offset: 0B8

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| dedat | 23:16 | RW |  | 8'h0 |
| Reserved | 15:8 | RO | Not used | 8'h0 |
| deat | 7:0 | RW |  | 8'h0 |

### TAT

Offset: 0BC

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| re\_to\_de | 31:16 | RW |  | 16'h0 |
| de\_to\_re | 15:0 | RW |  | 16'h0 |

### DLF

Offset: 0C0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:4 | RO | Not used | 28'h0 |
| dlf | 3:0 | RW |  | 4'h0 |

### RAR

Offset: 0C4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:8 | RO | Not used | 24'h0 |
| rar | 7:0 | RW |  | 8'h0 |

### TAR

Offset: 0C8

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:8 | RO | Not used | 24'h0 |
| tar | 7:0 | RW |  | 8'h0 |

### LCR\_EXT

Offset: 0CC

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:4 | RO | Not used | 28'h0 |
| transmit\_mode | 3:3 | RW |  | 1'h0 |
| send\_addr | 2:2 | RW |  | 1'h0 |
| addr\_match | 1:1 | RW |  | 1'h0 |
| dls\_e | 0:0 | RW |  | 1'h0 |

### CPR

Offset: 0F4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | Not used | 8'h0 |
| fifo\_mode | 23:16 | RO |  | 8'h0 |
| Reserved | 15:14 | RO | Not used | 2'h0 |
| dma\_extra | 13:13 | RO |  | 1'h0 |
| uart\_add\_encoded\_params | 12:12 | RO |  | 1'h0 |
| shadow | 11:11 | RO |  | 1'h0 |
| fifo\_stat | 10:10 | RO |  | 1'h0 |
| fifo\_access | 9:9 | RO |  | 1'h0 |
| new\_feat | 8:8 | RO |  | 1'h0 |
| sir\_lp\_mode | 7:7 | RO |  | 1'h0 |
| sir\_mode | 6:6 | RO |  | 1'h0 |
| thre\_mode | 5:5 | RO |  | 1'h0 |
| afce\_mode | 4:4 | RO |  | 1'h0 |
| Reserved | 3:2 | RO | Not used | 2'h0 |
| apb\_data\_width | 1:0 | RO |  | 2'h0 |

### UCV

Offset: 0F8

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ucv | 31:0 | RO |  | 32'h0 |

### CTR

Offset: 0FC

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ctr | 31:0 | RO |  | 32'h0 |

# DUAL\_TIMER

## Introduction

ARM的IP，需要PDF转成word

## Main Features

## Function Description

## Basic Block Diagram

## DUAL\_TIMER Register

### LOAD0

Offset: 00

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| LOAD\_L32 | 31:0 | RW |  | 32'h0 |

### VALUE0

Offset: 04

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| VALUE\_L32 | 31:0 | RO |  | 32'hffffffff |

### CONTROL0

Offset: 08

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| RESERVED0 | 31:8 | RW |  | 24'h000000 |
| ENABLE | 7:7 | RW | Enable bit | 1'h0 |
| MODE | 6:6 | RW | Mode bit | 1'h0 |
| INT\_ENABLE | 5:5 | RW | Interrupt Enable bit | 1'h1 |
| RESERVED1 | 4:4 | RW |  | 1'h0 |
| TIMERPRE | 3:2 | RW | Prescale bits | 2'h0 |
| SIZE | 1:1 | RW | Selects 16-bit or 32- bit counter operation | 1'h0 |
| ONESHOT | 0:0 | RW | Selects one-shot or wrapping counter mode | 1'h0 |

### INTCLR0

Offset: 0C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| INTCLR\_L32 | 31:0 | WO |  | 31'h00000000 |

### RIS0

Offset: 10

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| RESERVED0 | 31:1 | RO |  | 31'h00000000 |
| RAW\_INT | 1:0 | RO | Raw interrupt status | 1'h0 |

### MIS0

Offset: 14

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| RESERVED0 | 31:1 | RO |  | 31'h00000000 |
| INT | 1:0 | RO | Interrupt status | 1'h0 |

### BGLOAD0

Offset: 18

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| BGLOAD\_L32 | 31:0 | RW |  | 32'h0 |

### RESERVED

Offset: 1C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| RESERVED0 | 31:0 | RO |  | 32'h0 |

### LOAD1

Offset: 20

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| LOAD\_L32 | 31:0 | RW |  | 32'h0 |

### VALUE1

Offset: 24

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| VALUE\_L32 | 31:0 | RO |  | 32'hffffffff |

### CONTROL1

Offset: 28

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| RESERVED0 | 31:8 | RW |  | 24'h000000 |
| ENABLE | 7:7 | RW | Enable bit | 1'h0 |
| MODE | 6:6 | RW | Mode bit | 1'h0 |
| INT\_ENABLE | 5:5 | RW | Interrupt Enable bit | 1'h1 |
| RESERVED1 | 4:4 | RW |  | 1'h0 |
| TIMERPRE | 3:2 | RW | Prescale bits | 2'h0 |
| SIZE | 1:1 | RW | Selects 16-bit or 32- bit counter operation | 1'h0 |
| ONESHOT | 0:0 | RW | Selects one-shot or wrapping counter mode | 1'h0 |

### INTCLR1

Offset: 2C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| INTCLR\_L32 | 31:0 | WO |  | 31'h00000000 |

### RIS1

Offset: 30

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| RESERVED0 | 31:1 | RO |  | 31'h00000000 |
| RAW\_INT | 1:0 | RO | Raw interrupt status | 1'h0 |

### MIS1

Offset: 34

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| RESERVED0 | 31:1 | RO |  | 31'h00000000 |
| INT | 1:0 | RO | Interrupt status | 1'h0 |

### BGLOAD1

Offset: 38

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| BGLOAD\_L32 | 31:0 | RW |  | 32'h0 |

# GPT

## Introduction

General Purpose Timer (GPT) is a programmable timer module, which is a set of compact multi-function timers.

## Main Features

The GPT provides following features：

- selectable clk source for each channel : external clk, pclk

- support 4 independent channels

- each channel provides 6 scenarios:

1. one 32bit timer

2. two 16bit timers

3. four 8bit timers

4. one 16bit PWM

5. one 16bit timer and one 8bit PWM

6. two 8bit timers and one 8bit PWM

- each timer support 3 cnt modes: up, down, up/down

- each timer support 4 run modes: single, repeat, free run, keep go

- external pause to suspend all timer/PWM activities

- input capture, configurable capture edge and capture source

- generate one pulse

- support generates interrupt for each timer, and ch\_capture\_int, ledc\_tx\_int for each channel

- support LEDC

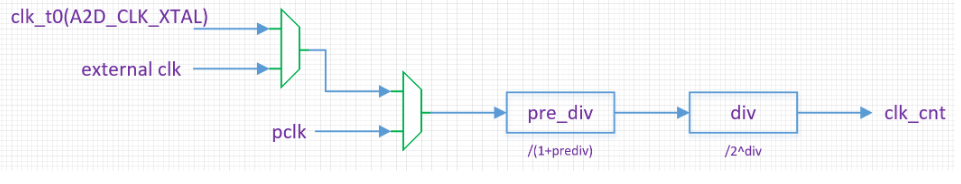
- a 4\*32-bit tx fifo for LEDC transmit

- a 4\*32-bit rx fifo for capture data

## Function Description

### Clock

The clock source for each channel is configurable. It can be selected form external clock, which comes from GPIO , internal clock clk\_t0 and PCLK. Actually, clk\_t0 and external clk are used as an enable signal of PCLK. Prescaler and divider will help to generate count clock (clk\_cnt) with lower frequency, the clk\_cnt is the work clock of counter, shown in Fig. 2.



​ Figure 2. Clock Block for each channel

### Timer

Each channel has a 32-bit counter, which can provides several timers with different bit width, such as 32-bit timer, 16-bit and 8-bit timers, see Tab.1. Using register ch\_mode to select different scenarios. There are three count directions supported, count up, count down, count up/down. In count up mode, the timer will count up to reload value setting by register CHx\_RELOAD. In count down mode the timer will count down from reload value to 0. And for count up/down mode the timer will count up to reload value first, and then count down to 0.

Also the timer run mode is programmable, user can select single, repeat, free run or keep go mode according to application. In free run mode, the timer will be a loop counter from 0 to the maximal value. And in keep go mode after re-enabling the counter, it will continue counting from the counted value when the timer is disabled.

All timer/PWM activity can be paused by external pause signal. The interrupt of each timer will be generated when time up.

### PWM

PWM output is also supported by this module. The register \*chx\_operation\* should be configured as 3 (PWM mode). The period and duty cycle of PWM is programmale by configuring register \*chx\_reload\*. When channel is disabled, PWM output value can be changed by register \*polarity\*. Central aligned and edge aligned are controled by register \*pwm\_out\_mode\*.

For central-aligned mode the period and duty cycle should be set in register \*CHx\_RELOAD\* (for detail please refer to \*Venus\_SoC\_Memory\_Mapping.xls\*). Take the following Fig. 3 central-aligned PWM as an example, setting register \*CHx\_RELOAD\* with period cycle = 5 and duty cycle = 2 and please note that the real period cycle of PWM is \*2×period cycle\*. When counter value is equal to duty cycle the PWM reverse.

For edge-aligned mode in register \*CHx\_RELOAD\* users need to configure high period and low period. The actual high and low period of PWM are \*1 + high period\* and \*1 + low period\* respectively. Take the following Fig.4 edge-aligned PWM as an example, setting register \*CHx\_RELOAD\* with low period = 3 and high period = 1. The period of PWM is \*low period + high period\*.

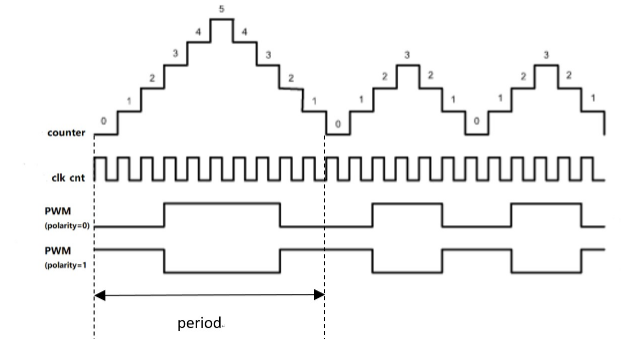


Figure 3. PWM Central-Aligned

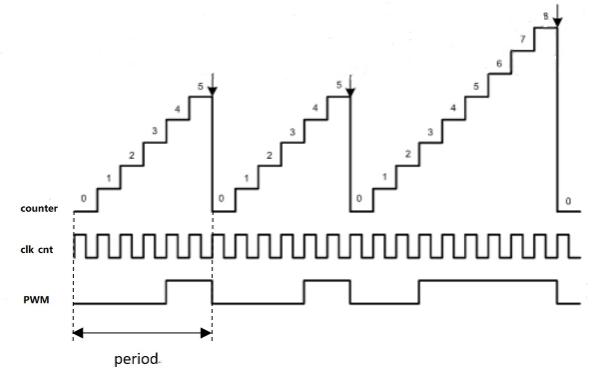
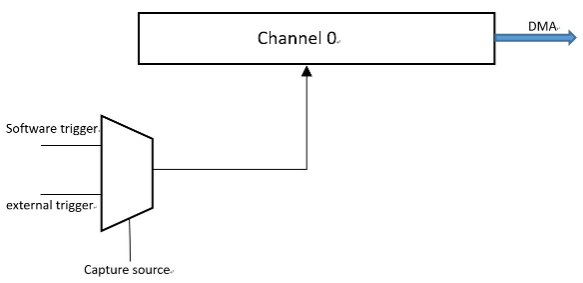


Figure 4. PWM edge-aligned

### Capture

Fig. 5 shows the simple block diagram for one channel in capture operation. Capture count mode is used to capture input pulse count, and capture time mode is used to capture pulse width. Capture source could be software trigger or external trigger. For software trigger user can set the register \*soft\_trigger\_en\*, and GPIO would be used as input pin for external trigger. User can chose rising edge, falling edge or both edge as the capture edge. The register \*chx\_operation\* should be switched to capture count mode or capture time mode, and set \*ch\_mode\* to 32-bit counter. The captured data would be saved in a 4\*32-bit rx\_fifo and can be transferd by CPU or DMA.



### One Pulse

One pulse generation is supported. Set \*operation\*=one\_pulse, and \*ch\_mode\*=1.To generate one high pulse, please set register \*polarity\* to "0" and "1" for low pulse. Register \*reload\* \*match\_0\* also need to be configured. \*reload\* value is the maxmal value of counter. When timer count up to \*match\_0\*, the pulse will be generated.

### LEDC

1. Timing of data sending

- Period of single bit data: 1.25us ~ 800kHz

- Minimal data accuracy: 0.05us = 50ns ~ 20MHz. Then the module clock should N x 20MHz.

- Uncertainty of rising / falling edge of data: ± 150ns

2. Sequence of data sending

- Single bit data output

- Sending MSB first

- Data transmit in order of GRB (Green, Red, Blue), high bit data at first

+ \*G7, G6, ......, G0, R7, ......, R0, B7, ......, B0\*

+ G7 is MSB

+ B0 is LSB

- Data length of each frame

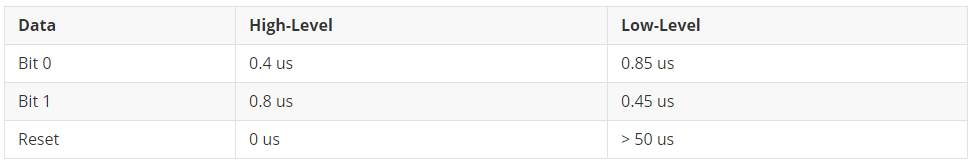
- 24bits of data to reproduce a color

- 3 groups of 8bits each that represent its RGB coding

- Time period of each frame: 24 \* 1.25us = 30us

- Time period reset code (interval of cascade frames): 50us

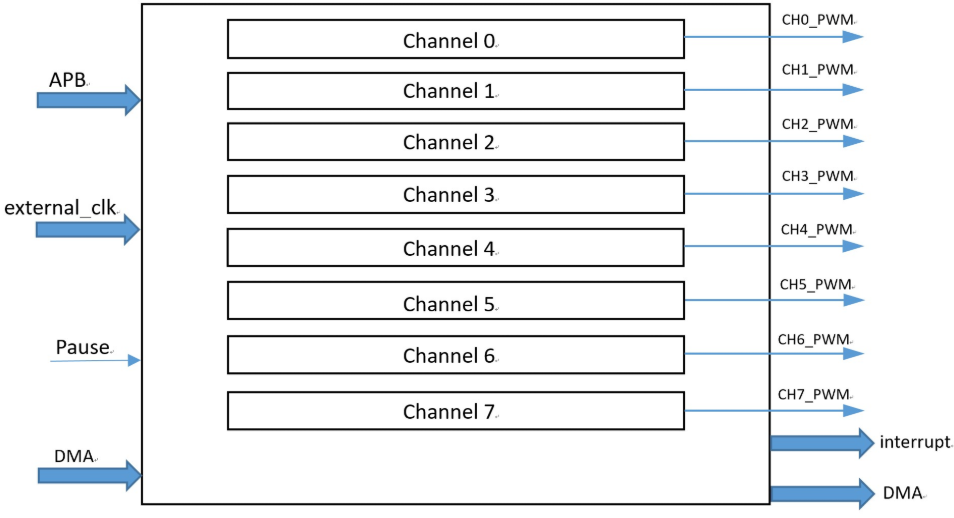
3. Data format (PWM waveform)



A 4\*32-bit tx fifo is provided for LEDC transmit. The data, which is to be send, can be transfered to tx fifo by CPU or DMA. The most significant 8-bit is flag bit: if [31:24]=8'h0 means it's a GRB data code, else it's a reset signal. Register \*chx\_reload\* and \*chx\_match\_0\* are used to configure high- and low-level duty cycle for bit 1 and bit 0. In LEDC mode, \*chx\_operation\* should be switch to LEDC mode and \*ch\_mode\* should be set to 4 (PWM). Central aligned is not supported in LEDC mode so set \*pwm\_out\_mode\* as edge aligned.

## Basic Block Diagram

Figure 1 shows the block diagram of GPT. 8 independent channels are supported. Each channel has a multi-function timer, and can also operate in capture time, capture count, one-pulse, PWM, and LEDC mode. Channel0 - Channel3 support DMA, and for Channel4 - Channel7 the data transmission can be achieved by CPU.



## GPT Register

### ID

Offset: 000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| reserved | 31:8 | RO | not used | 24'h0 |
| major | 7:4 | RO | major id value | 4'h1 |
| minor | 3:0 | RO | minor id value | 4'h0 |

### CH0\_CLK\_CTRL

Offset: 004

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| reserved | 31:25 | RO | not used | 7'h0 |
| ch0\_clk\_div\_ld | 24:24 | W1S | divider load enable. Also used as a common clk\_div\_ld for other channels in sync mode | 1'h0 |
| ch0\_clk\_prediv\_ld | 23:23 | W1S | pre divider load enable. Also used as a common clk\_prediv\_ld for other channels in sync mode | 1'h0 |
| ch0\_clk\_invert | 22:22 | RW | clock invert | 1'h0 |
| ch0\_clk\_gate | 21:21 | RW | counter clock gate enable | 1'h1 |
| ch0\_clk\_sel | 20:19 | RW | clock select. 0: clk\_t0 1: external clk , clk\_t1 2: pclk | 2'h0 |
| ch0\_clk\_div | 18:16 | RW | count\_clk=fpre/2^div | 3'h0 |
| ch0\_clk\_prediv | 15:0 | RW | pre divider, fpre=clk\_sel/(1+pre) | 16'h0 |

### CH1\_CLK\_CTRL

Offset: 008

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| reserved | 31:25 | RO | not used | 7'h0 |
| ch1\_clk\_div\_ld | 24:24 | W1S | divider load enable | 1'h0 |
| ch1\_clk\_prediv\_ld | 23:23 | W1S | pre divider load enable | 1'h0 |
| ch1\_clk\_invert | 22:22 | RW | clock invert | 1'h0 |
| ch1\_clk\_gate | 21:21 | RW | counter clock gate enable | 1'h1 |
| ch1\_clk\_sel | 20:19 | RW | clock select. 0: clk\_t0 1: external clk , clk\_t1 2: pclk | 2'h0 |
| ch1\_clk\_div | 18:16 | RW | count\_clk=fpre/2^div | 3'h0 |
| ch1\_clk\_prediv | 15:0 | RW | pre divider, fpre=clk\_sel/(1+pre) | 16'h0 |

### CH2\_CLK\_CTRL

Offset: 00C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| reserved | 31:25 | RO | not used | 7'h0 |
| ch2\_clk\_div\_ld | 24:24 | W1S | divider load enable | 1'h0 |
| ch2\_clk\_prediv\_ld | 23:23 | W1S | pre divider load enable | 1'h0 |
| ch2\_clk\_invert | 22:22 | RW | clock invert | 1'h0 |
| ch2\_clk\_gate | 21:21 | RW | counter clock gate enable | 1'h1 |
| ch2\_clk\_sel | 20:19 | RW | clock select. 0: clk\_t0 1: external clk , clk\_t1 2: pclk | 2'h0 |
| ch2\_clk\_div | 18:16 | RW | count\_clk=fpre/2^div | 3'h0 |
| ch2\_clk\_prediv | 15:0 | RW | pre divider, fpre=clk\_sel/(1+pre) | 16'h0 |

### CH3\_CLK\_CTRL

Offset: 010

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| reserved | 31:25 | RO | not used | 7'h0 |
| ch3\_clk\_div\_ld | 24:24 | W1S | divider load enable | 1'h0 |
| ch3\_clk\_prediv\_ld | 23:23 | W1S | pre divider load enable | 1'h0 |
| ch3\_clk\_invert | 22:22 | RW | clock invert | 1'h0 |
| ch3\_clk\_gate | 21:21 | RW | counter clock gate enable | 1'h1 |
| ch3\_clk\_sel | 20:19 | RW | clock select. 0: clk\_t0 1: external clk , clk\_t1 2: pclk | 2'h0 |
| ch3\_clk\_div | 18:16 | RW | count\_clk=fpre/2^div | 3'h0 |
| ch3\_clk\_prediv | 15:0 | RW | pre divider, fpre=clk\_sel/(1+pre) | 16'h0 |

### CH4\_CLK\_CTRL

Offset: 014

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| reserved | 31:25 | RO | not used | 7'h0 |
| ch4\_clk\_div\_ld | 24:24 | W1S | divider load enable | 1'h0 |
| ch4\_clk\_prediv\_ld | 23:23 | W1S | pre divider load enable | 1'h0 |
| ch4\_clk\_invert | 22:22 | RW | clock invert | 1'h0 |
| ch4\_clk\_gate | 21:21 | RW | counter clock gate enable | 1'h1 |
| ch4\_clk\_sel | 20:19 | RW | clock select. 0: clk\_t0 1: external clk , clk\_t1 2: pclk | 2'h0 |
| ch4\_clk\_div | 18:16 | RW | count\_clk=fpre/2^div | 3'h0 |
| ch4\_clk\_prediv | 15:0 | RW | pre divider, fpre=clk\_sel/(1+pre) | 16'h0 |

### CH5\_CLK\_CTRL

Offset: 018

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| reserved | 31:25 | RO | not used | 7'h0 |
| ch5\_clk\_div\_ld | 24:24 | W1S | divider load enable | 1'h0 |
| ch5\_clk\_prediv\_ld | 23:23 | W1S | pre divider load enable | 1'h0 |
| ch5\_clk\_invert | 22:22 | RW | clock invert | 1'h0 |
| ch5\_clk\_gate | 21:21 | RW | counter clock gate enable | 1'h1 |
| ch5\_clk\_sel | 20:19 | RW | clock select. 0: clk\_t0 1: external clk , clk\_t1 2: pclk | 2'h0 |
| ch5\_clk\_div | 18:16 | RW | count\_clk=fpre/2^div | 3'h0 |
| ch5\_clk\_prediv | 15:0 | RW | pre divider, fpre=clk\_sel/(1+pre) | 16'h0 |

### CH6\_CLK\_CTRL

Offset: 01C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| reserved | 31:25 | RO | not used | 7'h0 |
| ch6\_clk\_div\_ld | 24:24 | W1S | divider load enable | 1'h0 |
| ch6\_clk\_prediv\_ld | 23:23 | W1S | pre divider load enable | 1'h0 |
| ch6\_clk\_invert | 22:22 | RW | clock invert | 1'h0 |
| ch6\_clk\_gate | 21:21 | RW | counter clock gate enable | 1'h1 |
| ch6\_clk\_sel | 20:19 | RW | clock select. 0: clk\_t0 1: external clk , clk\_t1 2: pclk | 2'h0 |
| ch6\_clk\_div | 18:16 | RW | count\_clk=fpre/2^div | 3'h0 |
| ch6\_clk\_prediv | 15:0 | RW | pre divider, fpre=clk\_sel/(1+pre) | 16'h0 |

### CH7\_CLK\_CTRL

Offset: 020

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| reserved | 31:25 | RO | not used | 7'h0 |
| ch7\_clk\_div\_ld | 24:24 | W1S | divider load enable | 1'h0 |
| ch7\_clk\_prediv\_ld | 23:23 | W1S | pre divider load enable | 1'h0 |
| ch7\_clk\_invert | 22:22 | RW | clock invert | 1'h0 |
| ch7\_clk\_gate | 21:21 | RW | counter clock gate enable | 1'h1 |
| ch7\_clk\_sel | 20:19 | RW | clock select. 0: clk\_t0 1: external clk , clk\_t1 2: pclk | 2'h0 |
| ch7\_clk\_div | 18:16 | RW | count\_clk=fpre/2^div | 3'h0 |
| ch7\_clk\_prediv | 15:0 | RW | pre divider, fpre=clk\_sel/(1+pre) | 16'h0 |

### CH0\_CTRL

Offset: 024

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| reserved | 31:20 | RO | not used | 12'h0 |
| ch0\_start | 19:19 | W1S | to start channel 0 operation | 1'h0 |
| ch0\_stop | 18:18 | W1S | to stop channel 0 operation | 1'h0 |
| ch0\_chtrig\_reset\_cnt\_en | 17:17 | RW | enable channel trigger reset counter, the counter will be reset after each trigger, or when chx works as one pulse counter, it will enable trigger of chx to increase counter | 1'h0 |
| ch0\_soft\_trigger\_en | 16:16 | W1S | used in capture cnt mode, counter will be incremented by writing "1" | 1'h0 |
| ch0\_clk\_cnt\_gate | 15:15 | RW | the counter clock gating enable | 1'h1 |
| ch0\_clk\_sample\_gate | 14:14 | RW | the fast sample clock gating enable | 1'h1 |
| ch0\_capture\_filt\_thr | 13:11 | RW | filter threshod, to filter glitch, when the level counter is smaller than the value, it will be considered as a glitch | 3'h0 |
| ch0\_capture\_edgeA | 10:9 | RW | 0: rising edge,  1: falling edge,  2: both edges | 2'h0 |
| ch0\_capture\_src | 8:8 | RW | 0: external capture trigger,  1: software capture trigger | 1'h0 |
| ch0\_pwm\_out\_mode | 7:7 | RW | 0: edge aligned, 1: central aligned | 1'h0 |
| ch0\_pwm\_polarity | 6:6 | RW | output wave polarity 0: the PWM output is low when the channel is disabled 1: the PWM output is high when the channel is disabled | 1'h0 |
| ch0\_ch\_mode | 5:3 | RW | channel mode: 0: reserved 1: 32-bit timer 2:16-bit timers 3: 8-bit timers 4: PWM 5:reserved 6: mixed PWM/16-bit timer 7: mixed PWM/8-bit timers | 3'h0 |
| ch0\_operation | 2:0 | RW | 0: input capture time (ch\_mode = 1) 1: input capture count (ch\_mode = 1) 2: one pulse (ch\_mode = 1) 3. PWM (ch\_mode = 4 or 6 or 7) 4. LEDC (ch\_mode = 4) | 3'h0 |

### CH1\_CTRL

Offset: 028

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| reserved | 31:20 | RO | not used | 12'h0 |
| ch1\_start | 19:19 | W1S | to start channel 1 operation | 1'h0 |
| ch1\_stop | 18:18 | W1S | to stop channel 1 operation | 1'h0 |
| ch1\_chtrig\_reset\_cnt\_en | 17:17 | RW | enable channel trigger reset counter, the counter will be reset after each trigger, or when chx works as one pulse counter, it will enable trigger of chx to increase counter | 1'h0 |
| ch1\_soft\_trigger\_en | 16:16 | W1S | used in capture cnt mode, counter will be incremented by writing "1" | 1'h0 |
| ch1\_clk\_cnt\_gate | 15:15 | RW | the counter clock gating enable | 1'h1 |
| ch1\_clk\_sample\_gate | 14:14 | RW | the fast sample clock gating enable | 1'h1 |
| ch1\_capture\_filt\_thr | 13:11 | RW | filter threshod, to filter glitch, when the level counter is smaller than the value, it will be considered as a glitch | 3'h0 |
| ch1\_capture\_edgeA | 10:9 | RW | 0: rising edge,  1: falling edge,  2: both edges | 2'h0 |
| ch1\_capture\_src | 8:8 | RW | 0: external capture trigger,  1: software capture trigger | 1'h0 |
| ch1\_pwm\_out\_mode | 7:7 | RW | 0: edge aligned, 1: central aligned | 1'h0 |
| ch1\_pwm\_polarity | 6:6 | RW | output wave polarity 0: the PWM output is low when the channel is disabled 1: the PWM output is high when the channel is disabled | 1'h0 |
| ch1\_ch\_mode | 5:3 | RW | channel mode: 0: reserved 1: 32-bit timer 2:16-bit timers 3: 8-bit timers 4: PWM 5:reserved 6: mixed PWM/16-bit timer 7: mixed PWM/8-bit timers | 3'h0 |
| ch1\_operation | 2:0 | RW | 0: input capture time (ch\_mode = 1) 1: input capture count (ch\_mode = 1) 2: one pulse (ch\_mode = 1) 3. PWM (ch\_mode = 4 or 6 or 7) 4. LEDC (ch\_mode = 4) | 3'h0 |

### CH2\_CTRL

Offset: 02C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| reserved | 31:20 | RO | not used | 12'h0 |
| ch2\_start | 19:19 | W1S | to start channel 2 operation | 1'h0 |
| ch2\_stop | 18:18 | W1S | to stop channel 2 operation | 1'h0 |
| ch2\_chtrig\_reset\_cnt\_en | 17:17 | RW | enable channel trigger reset counter, the counter will be reset after each trigger, or when chx works as one pulse counter, it will enable trigger of chx to increase counter | 1'h0 |
| ch2\_soft\_trigger\_en | 16:16 | W1S | used in capture cnt mode, counter will be incremented by writing "1" | 1'h0 |
| ch2\_clk\_cnt\_gate | 15:15 | RW | the counter clock gating enable | 1'h1 |
| ch2\_clk\_sample\_gate | 14:14 | RW | the fast sample clock gating enable | 1'h1 |
| ch2\_capture\_filt\_thr | 13:11 | RW | filter threshod, to filter glitch, when the level counter is smaller than the value, it will be considered as a glitch | 3'h0 |
| ch2\_capture\_edgeA | 10:9 | RW | 0: rising edge,  1: falling edge,  2: both edges | 2'h0 |
| ch2\_capture\_src | 8:8 | RW | 0: external capture trigger,  1: software capture trigger | 1'h0 |
| ch2\_pwm\_out\_mode | 7:7 | RW | 0: edge aligned, 1: central aligned | 1'h0 |
| ch2\_pwm\_polarity | 6:6 | RW | output wave polarity 0: the PWM output is low when the channel is disabled 1: the PWM output is high when the channel is disabled | 1'h0 |
| ch2\_ch\_mode | 5:3 | RW | channel mode: 0: reserved 1: 32-bit timer 2:16-bit timers 3: 8-bit timers 4: PWM 5:reserved 6: mixed PWM/16-bit timer 7: mixed PWM/8-bit timers | 3'h0 |
| ch2\_operation | 2:0 | RW | 0: input capture time (ch\_mode = 1) 1: input capture count (ch\_mode = 1) 2: one pulse (ch\_mode = 1) 3. PWM (ch\_mode = 4 or 6 or 7) 4. LEDC (ch\_mode = 4) | 3'h0 |

### CH3\_CTRL

Offset: 030

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| reserved | 31:20 | RO | not used | 12'h0 |
| ch3\_start | 19:19 | W1S | to start channel 3 operation | 1'h0 |
| ch3\_stop | 18:18 | W1S | to stop channel 3 operation | 1'h0 |
| ch3\_chtrig\_reset\_cnt\_en | 17:17 | RW | enable channel trigger reset counter, the counter will be reset after each trigger, or when chx works as one pulse counter, it will enable trigger of chx to increase counter | 1'h0 |
| ch3\_soft\_trigger\_en | 16:16 | W1S | used in capture cnt mode, counter will be incremented by writing "1" | 1'h0 |
| ch3\_clk\_cnt\_gate | 15:15 | RW | the counter clock gating enable | 1'h1 |
| ch3\_clk\_sample\_gate | 14:14 | RW | the fast sample clock gating enable | 1'h1 |
| ch3\_capture\_filt\_thr | 13:11 | RW | filter threshod, to filter glitch, when the level counter is smaller than the value, it will be considered as a glitch | 3'h0 |
| ch3\_capture\_edgeA | 10:9 | RW | 0: rising edge,  1: falling edge,  2: both edges | 2'h0 |
| ch3\_capture\_src | 8:8 | RW | 0: external capture trigger,  1: software capture trigger | 1'h0 |
| ch3\_pwm\_out\_mode | 7:7 | RW | 0: edge aligned, 1: central aligned | 1'h0 |
| ch3\_pwm\_polarity | 6:6 | RW | output wave polarity 0: the PWM output is low when the channel is disabled 1: the PWM output is high when the channel is disabled | 1'h0 |
| ch3\_ch\_mode | 5:3 | RW | channel mode: 0: reserved 1: 32-bit timer 2:16-bit timers 3: 8-bit timers 4: PWM 5:reserved 6: mixed PWM/16-bit timer 7: mixed PWM/8-bit timers | 3'h0 |
| ch3\_operation | 2:0 | RW | 0: input capture time (ch\_mode = 1) 1: input capture count (ch\_mode = 1) 2: one pulse (ch\_mode = 1) 3. PWM (ch\_mode = 4 or 6 or 7) 4. LEDC (ch\_mode = 4) | 3'h0 |

### CH4\_CTRL

Offset: 034

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| reserved | 31:20 | RO | not used | 12'h0 |
| ch4\_start | 19:19 | W1S | to start channel 4 operation | 1'h0 |
| ch4\_stop | 18:18 | W1S | to stop channel 4 operation | 1'h0 |
| ch4\_chtrig\_reset\_cnt\_en | 17:17 | RW | enable channel trigger reset counter, the counter will be reset after each trigger, or when chx works as one pulse counter, it will enable trigger of chx to increase counter | 1'h0 |
| ch4\_soft\_trigger\_en | 16:16 | W1S | used in capture cnt mode, counter will be incremented by writing "1" | 1'h0 |
| ch4\_clk\_cnt\_gate | 15:15 | RW | the counter clock gating enable | 1'h1 |
| ch4\_clk\_sample\_gate | 14:14 | RW | the fast sample clock gating enable | 1'h1 |
| ch4\_capture\_filt\_thr | 13:11 | RW | filter threshod, to filter glitch, when the level counter is smaller than the value, it will be considered as a glitch | 3'h0 |
| ch4\_capture\_edgeA | 10:9 | RW | 0: rising edge,  1: falling edge,  2: both edges | 2'h0 |
| ch4\_capture\_src | 8:8 | RW | 0: external capture trigger,  1: software capture trigger | 1'h0 |
| ch4\_pwm\_out\_mode | 7:7 | RW | 0: edge aligned, 1: central aligned | 1'h0 |
| ch4\_pwm\_polarity | 6:6 | RW | output wave polarity 0: the PWM output is low when the channel is disabled 1: the PWM output is high when the channel is disabled | 1'h0 |
| ch4\_ch\_mode | 5:3 | RW | channel mode: 0: reserved 1: 32-bit timer 2:16-bit timers 3: 8-bit timers 4: PWM 5:reserved 6: mixed PWM/16-bit timer 7: mixed PWM/8-bit timers | 3'h0 |
| ch4\_operation | 2:0 | RW | 0: input capture time (ch\_mode = 1) 1: input capture count (ch\_mode = 1) 2: one pulse (ch\_mode = 1) 3. PWM (ch\_mode = 4 or 6 or 7) 4. LEDC (ch\_mode = 4) | 3'h0 |

### CH5\_CTRL

Offset: 038

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| reserved | 31:20 | RO | not used | 12'h0 |
| ch5\_start | 19:19 | W1S | to start channel 5 operation | 1'h0 |
| ch5\_stop | 18:18 | W1S | to stop channel 5 operation | 1'h0 |
| ch5\_chtrig\_reset\_cnt\_en | 17:17 | RW | enable channel trigger reset counter, the counter will be reset after each trigger, or when chx works as one pulse counter, it will enable trigger of chx to increase counter | 1'h0 |
| ch5\_soft\_trigger\_en | 16:16 | W1S | used in capture cnt mode, counter will be incremented by writing "1" | 1'h0 |
| ch5\_clk\_cnt\_gate | 15:15 | RW | the counter clock gating enable | 1'h1 |
| ch5\_clk\_sample\_gate | 14:14 | RW | the fast sample clock gating enable | 1'h1 |
| ch5\_capture\_filt\_thr | 13:11 | RW | filter threshod, to filter glitch, when the level counter is smaller than the value, it will be considered as a glitch | 3'h0 |
| ch5\_capture\_edgeA | 10:9 | RW | 0: rising edge,  1: falling edge,  2: both edges | 2'h0 |
| ch5\_capture\_src | 8:8 | RW | 0: external capture trigger,  1: software capture trigger | 1'h0 |
| ch5\_pwm\_out\_mode | 7:7 | RW | 0: edge aligned, 1: central aligned | 1'h0 |
| ch5\_pwm\_polarity | 6:6 | RW | output wave polarity 0: the PWM output is low when the channel is disabled 1: the PWM output is high when the channel is disabled | 1'h0 |
| ch5\_ch\_mode | 5:3 | RW | channel mode: 0: reserved 1: 32-bit timer 2:16-bit timers 3: 8-bit timers 4: PWM 5:reserved 6: mixed PWM/16-bit timer 7: mixed PWM/8-bit timers | 3'h0 |
| ch5\_operation | 2:0 | RW | 0: input capture time (ch\_mode = 1) 1: input capture count (ch\_mode = 1) 2: one pulse (ch\_mode = 1) 3. PWM (ch\_mode = 4 or 6 or 7) 4. LEDC (ch\_mode = 4) | 3'h0 |

### CH6\_CTRL

Offset: 03C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| reserved | 31:20 | RO | not used | 12'h0 |
| ch6\_start | 19:19 | W1S | to start channel 6 operation | 1'h0 |
| ch6\_stop | 18:18 | W1S | to stop channel 6 operation | 1'h0 |
| ch6\_chtrig\_reset\_cnt\_en | 17:17 | RW | enable channel trigger reset counter, the counter will be reset after each trigger, or when chx works as one pulse counter, it will enable trigger of chx to increase counter | 1'h0 |
| ch6\_soft\_trigger\_en | 16:16 | W1S | used in capture cnt mode, counter will be incremented by writing "1" | 1'h0 |
| ch6\_clk\_cnt\_gate | 15:15 | RW | the counter clock gating enable | 1'h1 |
| ch6\_clk\_sample\_gate | 14:14 | RW | the fast sample clock gating enable | 1'h1 |
| ch6\_capture\_filt\_thr | 13:11 | RW | filter threshod, to filter glitch, when the level counter is smaller than the value, it will be considered as a glitch | 3'h0 |
| ch6\_capture\_edgeA | 10:9 | RW | 0: rising edge,  1: falling edge,  2: both edges | 2'h0 |
| ch6\_capture\_src | 8:8 | RW | 0: external capture trigger,  1: software capture trigger | 1'h0 |
| ch6\_pwm\_out\_mode | 7:7 | RW | 0: edge aligned, 1: central aligned | 1'h0 |
| ch6\_pwm\_polarity | 6:6 | RW | output wave polarity 0: the PWM output is low when the channel is disabled 1: the PWM output is high when the channel is disabled | 1'h0 |
| ch6\_ch\_mode | 5:3 | RW | channel mode: 0: reserved 1: 32-bit timer 2:16-bit timers 3: 8-bit timers 4: PWM 5:reserved 6: mixed PWM/16-bit timer 7: mixed PWM/8-bit timers | 3'h0 |
| ch6\_operation | 2:0 | RW | 0: input capture time (ch\_mode = 1) 1: input capture count (ch\_mode = 1) 2: one pulse (ch\_mode = 1) 3. PWM (ch\_mode = 4 or 6 or 7) 4. LEDC (ch\_mode = 4) | 3'h0 |

### CH7\_CTRL

Offset: 040

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| reserved | 31:20 | RO | not used | 12'h0 |
| ch7\_start | 19:19 | W1S | to start channel 7 operation | 1'h0 |
| ch7\_stop | 18:18 | W1S | to stop channel 7 operation | 1'h0 |
| ch7\_chtrig\_reset\_cnt\_en | 17:17 | RW | enable channel trigger reset counter, the counter will be reset after each trigger, or when chx works as one pulse counter, it will enable trigger of chx to increase counter | 1'h0 |
| ch7\_soft\_trigger\_en | 16:16 | W1S | used in capture cnt mode, counter will be incremented by writing "1" | 1'h0 |
| ch7\_clk\_cnt\_gate | 15:15 | RW | the counter clock gating enable | 1'h1 |
| ch7\_clk\_sample\_gate | 14:14 | RW | the fast sample clock gating enable | 1'h1 |
| ch7\_capture\_filt\_thr | 13:11 | RW | filter threshod, to filter glitch, when the level counter is smaller than the value, it will be considered as a glitch | 3'h0 |
| ch7\_capture\_edgeA | 10:9 | RW | 0: rising edge,  1: falling edge,  2: both edges | 2'h0 |
| ch7\_capture\_src | 8:8 | RW | 0: external capture trigger,  1: software capture trigger | 1'h0 |
| ch7\_pwm\_out\_mode | 7:7 | RW | 0: edge aligned, 1: central aligned | 1'h0 |
| ch7\_pwm\_polarity | 6:6 | RW | output wave polarity 0: the PWM output is low when the channel is disabled 1: the PWM output is high when the channel is disabled | 1'h0 |
| ch7\_ch\_mode | 5:3 | RW | channel mode: 0: reserved 1: 32-bit timer 2:16-bit timers 3: 8-bit timers 4: PWM 5:reserved 6: mixed PWM/16-bit timer 7: mixed PWM/8-bit timers | 3'h0 |
| ch7\_operation | 2:0 | RW | 0: input capture time (ch\_mode = 1) 1: input capture count (ch\_mode = 1) 2: one pulse (ch\_mode = 1) 3. PWM (ch\_mode = 4 or 6 or 7) 4. LEDC (ch\_mode = 4) | 3'h0 |

### CH0\_RELOAD

Offset: 044

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch0\_reload\_up\_cfg | 31:0 | RW | channel 0 reload value: 1. when ch\_mode=1, reload value for 32-bit timer0. 2. when ch\_mode=2, [31:16]: reload value for 16-bit timer1; [15:0]: reload value for 16-bit timer0. 3. when ch\_mode=3, [31:24] reload value for 8-bit timer3; [23:16]: reload value for 8-bit timer2; [15:8]: reload value for 8-bit timer1; [7:0]:reload value for 8-bit timer0. 4. when ch\_mode=4, pwm\_out\_mode=0: [31:16]: reload value for PWM high period; [15:0]: reload value for PWM low period; pwm\_out\_mode=1: [31:16]: reload value for PWM period; [15:0]: reload value for PWM duty. 5. when ch\_mode=6, pwm\_out\_mode=0: [31:24]: reload value for PWM high period; [23:16]: reload value for PWM low period; pwm\_out\_mode=1: [31:24]: reload value for PWM period; [23:16]: reload value for PWM duty; [15:0]: reload value for 16-bit timer0. 6. when ch\_mode=7, pwm\_out\_mode=0: [31:24]: reload value for PWM high period; [23:16]: reload value for PWM low period; pwm\_out\_mode=1: [31:24]: reload value for PWM period; [23:16]: reload value for PWM duty; [15:8]: reload value for 8-bit timer1; [7:0]:reload value for 8-bit timer0. when operation=CH\_LEDC: [31:16]: for bit 1 high duty cycle (0.8us); [15:0] for bit 1 low dury cycle (0.45us). | 32'h0 |

### CH1\_RELOAD

Offset: 048

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch1\_reload\_up\_cfg | 31:0 | RW | channel 1 reload value: refer to ch0\_reload | 32'h0 |

### CH2\_RELOAD

Offset: 04C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch2\_reload\_up\_cfg | 31:0 | RW | channel 2 reload value: refer to ch0\_reload | 32'h0 |

### CH3\_RELOAD

Offset: 050

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch3\_reload\_up\_cfg | 31:0 | RW | channel 3 reload value: refer to ch0\_reload | 32'h0 |

### CH4\_RELOAD

Offset: 054

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch4\_reload\_up\_cfg | 31:0 | RW | channel 4 reload value: refer to ch0\_reload | 32'h0 |

### CH5\_RELOAD

Offset: 058

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch5\_reload\_up\_cfg | 31:0 | RW | channel 5 reload value: refer to ch0\_reload | 32'h0 |

### CH6\_RELOAD

Offset: 05C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch6\_reload\_up\_cfg | 31:0 | RW | channel 6 reload value: refer to ch0\_reload | 32'h0 |

### CH7\_RELOAD

Offset: 060

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch7\_reload\_up\_cfg | 31:0 | RW | channel 7 reload value: refer to ch0\_reload | 32'h0 |

### CNT\_MODE\_CTRL\_1

Offset: 064

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch7\_timer3\_cnt\_mode | 31:30 | RW | count direction:  0: up 1: down 2: up/down | 2'h0 |
| ch7\_timer2\_cnt\_mode | 29:28 | RW |  | 2'h0 |
| ch7\_timer1\_cnt\_mode | 27:26 | RW |  | 2'h0 |
| ch7\_timer0\_cnt\_mode | 25:24 | RW |  | 2'h0 |
| ch6\_timer3\_cnt\_mode | 23:22 | RW |  | 2'h0 |
| ch6\_timer2\_cnt\_mode | 21:20 | RW |  | 2'h0 |
| ch6\_timer1\_cnt\_mode | 19:18 | RW |  | 2'h0 |
| ch6\_timer0\_cnt\_mode | 17:16 | RW |  | 2'h0 |
| ch5\_timer3\_cnt\_mode | 15:14 | RW |  | 2'h0 |
| ch5\_timer2\_cnt\_mode | 13:12 | RW |  | 2'h0 |
| ch5\_timer1\_cnt\_mode | 11:10 | RW |  | 2'h0 |
| ch5\_timer0\_cnt\_mode | 9:8 | RW |  | 2'h0 |
| ch4\_timer3\_cnt\_mode | 7:6 | RW |  | 2'h0 |
| ch4\_timer2\_cnt\_mode | 5:4 | RW |  | 2'h0 |
| ch4\_timer1\_cnt\_mode | 3:2 | RW |  | 2'h0 |
| ch4\_timer0\_cnt\_mode | 1:0 | RW |  | 2'h0 |

### CNT\_MODE\_CTRL\_0

Offset: 068

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch3\_timer3\_cnt\_mode | 31:30 | RW | count direction:  0: up 1: down 2: up/down | 2'h0 |
| ch3\_timer2\_cnt\_mode | 29:28 | RW |  | 2'h0 |
| ch3\_timer1\_cnt\_mode | 27:26 | RW |  | 2'h0 |
| ch3\_timer0\_cnt\_mode | 25:24 | RW |  | 2'h0 |
| ch2\_timer3\_cnt\_mode | 23:22 | RW |  | 2'h0 |
| ch2\_timer2\_cnt\_mode | 21:20 | RW |  | 2'h0 |
| ch2\_timer1\_cnt\_mode | 19:18 | RW |  | 2'h0 |
| ch2\_timer0\_cnt\_mode | 17:16 | RW |  | 2'h0 |
| ch1\_timer3\_cnt\_mode | 15:14 | RW |  | 2'h0 |
| ch1\_timer2\_cnt\_mode | 13:12 | RW |  | 2'h0 |
| ch1\_timer1\_cnt\_mode | 11:10 | RW |  | 2'h0 |
| ch1\_timer0\_cnt\_mode | 9:8 | RW |  | 2'h0 |
| ch0\_timer3\_cnt\_mode | 7:6 | RW |  | 2'h0 |
| ch0\_timer2\_cnt\_mode | 5:4 | RW |  | 2'h0 |
| ch0\_timer1\_cnt\_mode | 3:2 | RW |  | 2'h0 |
| ch0\_timer0\_cnt\_mode | 1:0 | RW |  | 2'h0 |

### RUN\_MODE\_CTRL\_1

Offset: 06C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch7\_timer3\_run\_mode | 31:30 | RW | 0: single 1: repeat 2. free\_run 3. keepgo | 2'h0 |
| ch7\_timer2\_run\_mode | 29:28 | RW |  | 2'h0 |
| ch7\_timer1\_run\_mode | 27:26 | RW |  | 2'h0 |
| ch7\_timer0\_run\_mode | 25:24 | RW |  | 2'h0 |
| ch6\_timer3\_run\_mode | 23:22 | RW |  | 2'h0 |
| ch6\_timer2\_run\_mode | 21:20 | RW |  | 2'h0 |
| ch6\_timer1\_run\_mode | 19:18 | RW |  | 2'h0 |
| ch6\_timer0\_run\_mode | 17:16 | RW |  | 2'h0 |
| ch5\_timer3\_run\_mode | 15:14 | RW |  | 2'h0 |
| ch5\_timer2\_run\_mode | 13:12 | RW |  | 2'h0 |
| ch5\_timer1\_run\_mode | 11:10 | RW |  | 2'h0 |
| ch5\_timer0\_run\_mode | 9:8 | RW |  | 2'h0 |
| ch4\_timer3\_run\_mode | 7:6 | RW |  | 2'h0 |
| ch4\_timer2\_run\_mode | 5:4 | RW |  | 2'h0 |
| ch4\_timer1\_run\_mode | 3:2 | RW |  | 2'h0 |
| ch4\_timer0\_run\_mode | 1:0 | RW |  | 2'h0 |

### RUN\_MODE\_CTRL\_0

Offset: 070

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch3\_timer3\_run\_mode | 31:30 | RW | 0: single 1: repeat 2. free\_run 3. keepgo | 2'h0 |
| ch3\_timer2\_run\_mode | 29:28 | RW |  | 2'h0 |
| ch3\_timer1\_run\_mode | 27:26 | RW |  | 2'h0 |
| ch3\_timer0\_run\_mode | 25:24 | RW |  | 2'h0 |
| ch2\_timer3\_run\_mode | 23:22 | RW |  | 2'h0 |
| ch2\_timer2\_run\_mode | 21:20 | RW |  | 2'h0 |
| ch2\_timer1\_run\_mode | 19:18 | RW |  | 2'h0 |
| ch2\_timer0\_run\_mode | 17:16 | RW |  | 2'h0 |
| ch1\_timer3\_run\_mode | 15:14 | RW |  | 2'h0 |
| ch1\_timer2\_run\_mode | 13:12 | RW |  | 2'h0 |
| ch1\_timer1\_run\_mode | 11:10 | RW |  | 2'h0 |
| ch1\_timer0\_run\_mode | 9:8 | RW |  | 2'h0 |
| ch0\_timer3\_run\_mode | 7:6 | RW |  | 2'h0 |
| ch0\_timer2\_run\_mode | 5:4 | RW |  | 2'h0 |
| ch0\_timer1\_run\_mode | 3:2 | RW |  | 2'h0 |
| ch0\_timer0\_run\_mode | 1:0 | RW |  | 2'h0 |

### CH0\_CNT

Offset: 074

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch0\_counter\_value | 31:0 | RO | current counter value for channel 0 timer/PWM refer to chn\_reload description for the field definition with different ch\_mode | 32'h0 |

### CH1\_CNT

Offset: 078

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch1\_counter\_value | 31:0 | RO | current counter value for channel 1 timer/PWM refer to chn\_reload description for the field definition with different ch\_mode | 32'h0 |

### CH2\_CNT

Offset: 07C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch2\_counter\_value | 31:0 | RO | current counter value for channel 2 timer/PWM refer to chn\_reload description for the field definition with different ch\_mode | 32'h0 |

### CH3\_CNT

Offset: 080

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch3\_counter\_value | 31:0 | RO | current counter value for channel 3 timer/PWM refer to chn\_reload description for the field definition with different ch\_mode | 32'h0 |

### CH4\_CNT

Offset: 084

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch4\_counter\_value | 31:0 | RO | current counter value for channel 4 timer/PWM refer to chn\_reload description for the field definition with different ch\_mode | 32'h0 |

### CH5\_CNT

Offset: 088

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch5\_counter\_value | 31:0 | RO | current counter value for channel 5 timer/PWM refer to chn\_reload description for the field definition with different ch\_mode | 32'h0 |

### CH6\_CNT

Offset: 08C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch6\_counter\_value | 31:0 | RO | current counter value for channel 6 timer/PWM refer to chn\_reload description for the field definition with different ch\_mode | 32'h0 |

### CH7\_CNT

Offset: 090

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch7\_counter\_value | 31:0 | RO | current counter value for channel 7 timer/PWM refer to chn\_reload description for the field definition with different ch\_mode | 32'h0 |

### CH\_CNT\_EN

Offset: 094

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| reserved | 31:24 | RO | not used | 8'h0 |
| cnt\_reset | 23:16 | W1S | counter reset. [16:23]: ch7-ch0 | 8'h0 |
| cnt\_stop | 15:8 | W1S | counter stop. [15:8]: ch7-ch0 | 8'h0 |
| cnt\_start | 7:0 | W1S | counter start. [7:0]: ch7-ch0 | 8'h0 |

### CH\_NUM

Offset: 098

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| reserved | 31:8 | RO | not used | 24'h0 |
| ch\_num | 7:0 | RO | indicates occupied channels, [7:0]: ch7 - ch0 | 8'h0 |

### CH0\_MATCH\_0

Offset: 09C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch0\_match\_0 | 31:0 | RW | 1. when operation=CH\_LEDC: [31:16]: for bit 0 high duty cycle (0.4us); [15:0] for bit 0 low dury cycle (0.85us). 2. when operation=CH\_ONE\_PULSE: pulse generation point | 32'h0 |

### CH1\_MATCH\_0

Offset: 0A0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch1\_match\_0 | 31:0 | RW | 1. when operation=CH\_LEDC: [31:16]: for bit 0 high duty cycle (0.4us); [15:0] for bit 0 low dury cycle (0.85us). 2. when operation=CH\_ONE\_PULSE: pulse generation point | 32'h0 |

### CH2\_MATCH\_0

Offset: 0A4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch2\_match\_0 | 31:0 | RW | 1. when operation=CH\_LEDC: [31:16]: for bit 0 high duty cycle (0.4us); [15:0] for bit 0 low dury cycle (0.85us). 2. when operation=CH\_ONE\_PULSE: pulse generation point | 32'h0 |

### CH3\_MATCH\_0

Offset: 0A8

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch3\_match\_0 | 31:0 | RW | 1. when operation=CH\_LEDC: [31:16]: for bit 0 high duty cycle (0.4us); [15:0] for bit 0 low dury cycle (0.85us). 2. when operation=CH\_ONE\_PULSE: pulse generation point | 32'h0 |

### CH4\_MATCH\_0

Offset: 0AC

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch4\_match\_0 | 31:0 | RW | 1. when operation=CH\_LEDC: [31:16]: for bit 0 high duty cycle (0.4us); [15:0] for bit 0 low dury cycle (0.85us). 2. when operation=CH\_ONE\_PULSE: pulse generation point | 32'h0 |

### CH5\_MATCH\_0

Offset: 0B0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch5\_match\_0 | 31:0 | RW | 1. when operation=CH\_LEDC: [31:16]: for bit 0 high duty cycle (0.4us); [15:0] for bit 0 low dury cycle (0.85us). 2. when operation=CH\_ONE\_PULSE: pulse generation point | 32'h0 |

### CH6\_MATCH\_0

Offset: 0B4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch6\_match\_0 | 31:0 | RW | 1. when operation=CH\_LEDC: [31:16]: for bit 0 high duty cycle (0.4us); [15:0] for bit 0 low dury cycle (0.85us). 2. when operation=CH\_ONE\_PULSE: pulse generation point | 32'h0 |

### CH7\_MATCH\_0

Offset: 0B8

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch7\_match\_0 | 31:0 | RW | 1. when operation=CH\_LEDC: [31:16]: for bit 0 high duty cycle (0.4us); [15:0] for bit 0 low dury cycle (0.85us). 2. when operation=CH\_ONE\_PULSE: pulse generation point | 32'h0 |

### DMA\_CTRL

Offset: 0BC

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| reserved | 31:24 | RO | not used | 8'h0 |
| rx\_dma\_thres\_sel | 23:20 | RW | DMA burst request theshold, dma\_req will be triggered when fifo\_usedw >= dma\_thres. 0: 1 word 1: 4 words | 4'hf |
| tx\_dma\_thres\_sel | 19:16 | RW | DMA burst request theshold, dma\_req will be triggered when (4-fifo\_usedw) >= dma\_thres. 0: 1 word 1: 4 words | 4'hf |
| rx\_dma\_done | 15:12 | RO | indicate if rx dma transfer is done, [23:20]: ch3-ch0 | 4'h0 |
| tx\_dma\_done | 11:8 | RO | indicate if tx dma transfer is done, [19:16]: ch3-ch0 | 4'h0 |
| rx\_dma\_en | 7:4 | RW | capture rx dma enable, [7:4]: ch3 - ch0 | 4'h0 |
| tx\_dma\_en | 3:0 | RW | ledc tx dma enable, [3:0]: ch3 - ch0 | 4'h0 |

### IMR\_CH

Offset: 0C0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| tx\_fifo\_underflow\_int\_mask | 31:24 | RW | tx fifo underflow interrupt mask. [31:24]: ch7 - ch0 | 8'hff |
| rx\_fifo\_overflow\_int\_mask | 23:16 | RW | rx fifo overflow interrupt mask. [23:16]: ch7 - ch0 | 8'hff |
| ledc\_tx\_int\_mask | 15:8 | RW | ledc tx done interrupt mask. [15:8]: ch7 - ch0 | 8'hff |
| capture\_int\_mask | 7:0 | RW | input capture mask. [7:0]: ch7 - ch0 | 8'hff |

### IMR\_TIMER

Offset: 0C4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| timer3\_int\_mask | 31:24 | RW | timer 3 interrupt mask. [31:24]: ch7 - ch0 | 8'hff |
| timer2\_int\_mask | 23:16 | RW | timer 2 interrupt mask. [23:16]: ch7 - ch0 | 8'hff |
| timer1\_int\_mask | 15:8 | RW | timer 1 interrupt mask. [15:8]: ch7 - ch0 | 8'hff |
| timer0\_int\_mask | 7:0 | RW | timer 0 interrupt mask. [7:0]: ch7 - ch0 | 8'hff |

### ISR\_CH

Offset: 0C8

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| tx\_fifo\_underflow\_int\_isr | 31:24 | RO | tx fifo underflow interrupt status. [31:24]: ch7 - ch0 | 8'h0 |
| rx\_fifo\_overflow\_int\_isr | 23:16 | RO | rx fifo overflow interrupt status. [23:16]: ch7 - ch0 | 8'h0 |
| ledc\_tx\_int\_isr | 15:8 | RO | ledc tx done interrupt status. [15:8]: ch7 - ch0 | 8'h0 |
| capture\_int\_isr | 7:0 | RO | input capture interrupt status. [7:0]: ch7 - ch0 | 8'h0 |

### ISR\_TIMER

Offset: 0CC

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| timer3\_int\_isr | 31:24 | RO | timer 3 interrupt status. [31:24]: ch7 - ch0 | 8'h0 |
| timer2\_int\_isr | 23:16 | RO | timer 2 interrupt status. [23:16]: ch7 - ch0 | 8'h0 |
| timer1\_int\_isr | 15:8 | RO | timer 1 interrupt status. [15:8]: ch7 - ch0 | 8'h0 |
| timer0\_int\_isr | 7:0 | RO | timer 0 interrupt status. [7:0]: ch7 - ch0 | 8'h0 |

### ICR\_CH

Offset: 0D0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| reserved | 31:24 | RO | not used | 8'h0 |
| rx\_dma\_done\_clr | 23:20 | W1C | set 1 to clear rx\_dma\_done, [23:20]: ch3 - ch0 | 4'h0 |
| tx\_dma\_done\_clr | 19:16 | W1C | set 1 to clear tx\_dma\_done, [23:19]: ch3 - ch0 | 4'h0 |
| ledc\_tx\_int\_clr | 15:8 | W1C | ledc tx done interrupt clear. [15:8]: ch7 - ch0 | 8'h0 |
| capture\_clr | 7:0 | W1C | input capture interrupt clear. [7:0]: ch7 - ch0 | 8'h0 |

### ICR\_TIMER

Offset: 0D4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| timer3\_int\_clr | 31:24 | W1C | timer 3 interrupt clear. [31:24]: ch7 - ch0 | 8'h0 |
| timer2\_int\_clr | 23:16 | W1C | timer 2 interrupt clear. [23:16]: ch7 - ch0 | 8'h0 |
| timer1\_int\_clr | 15:8 | W1C | timer 1 interrupt clear. [15:8]: ch7 - ch0 | 8'h0 |
| timer0\_int\_clr | 7:0 | W1C | timer 0 interrupt clear. [7:0]: ch7 - ch0 | 8'h0 |

### CH\_TIMER\_ENABLE

Offset: 0D8

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| timer3\_pwm\_en | 31:24 | RW | when ch\_mode=3 set to 1 : timer 3 enable when ch\_mode=4, 6, 7 set to 1: pwm enable [31:24]: ch7 - ch0 | 8'h0 |
| timer2\_en | 23:16 | RW | when ch\_mode=3 set to 1 timer 2 enable [23:16]: ch7 - ch0 | 8'h0 |
| timer1\_en | 15:8 | RW | when ch\_mode=2, 3, 7 set to 1: timer 1 enable  [15:8]: ch7 - ch0 | 8'h0 |
| timer0\_en | 7:0 | RW | when ch\_mode=1, 2, 3, 6, 7 set to 1: timer 0 enable  [7:0]: ch7 - ch0 | 8'h0 |

### CH0\_LEDC\_TX\_FIFO

Offset: 0DC

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch0\_ledc\_tx\_fifo | 31:0 | WO | ledc tx fifo | 32'h0 |

### CH1\_LEDC\_TX\_FIFO

Offset: 0E0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch1\_ledc\_tx\_fifo | 31:0 | WO | ledc tx fifo | 32'h0 |

### CH2\_LEDC\_TX\_FIFO

Offset: 0E4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch2\_ledc\_tx\_fifo | 31:0 | WO | ledc tx fifo | 32'h0 |

### CH3\_LEDC\_TX\_FIFO

Offset: 0E8

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch3\_ledc\_tx\_fifo | 31:0 | WO | ledc tx fifo | 32'h0 |

### CH4\_LEDC\_TX\_FIFO

Offset: 0EC

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch4\_ledc\_tx\_fifo | 31:0 | WO | ledc tx fifo | 32'h0 |

### CH5\_LEDC\_TX\_FIFO

Offset: 0F0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch5\_ledc\_tx\_fifo | 31:0 | WO | ledc tx fifo | 32'h0 |

### CH6\_LEDC\_TX\_FIFO

Offset: 0F4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch6\_ledc\_tx\_fifo | 31:0 | WO | ledc tx fifo | 32'h0 |

### CH7\_LEDC\_TX\_FIFO

Offset: 0F8

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch7\_ledc\_tx\_fifo | 31:0 | WO | ledc tx fifo | 32'h0 |

### CH0\_RX\_FIFO

Offset: 0FC

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch0\_rx\_fifo | 31:0 | RO | rx fifo for capture data | 32'h0 |

### CH1\_RX\_FIFO

Offset: 100

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch1\_rx\_fifo | 31:0 | RO | rx fifo for capture data | 32'h0 |

### CH2\_RX\_FIFO

Offset: 104

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch2\_rx\_fifo | 31:0 | RO | rx fifo for capture data | 32'h0 |

### CH3\_RX\_FIFO

Offset: 108

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch3\_rx\_fifo | 31:0 | RO | rx fifo for capture data | 32'h0 |

### CH4\_RX\_FIFO

Offset: 10C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch4\_rx\_fifo | 31:0 | RO | rx fifo for capture data | 32'h0 |

### CH5\_RX\_FIFO

Offset: 110

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch5\_rx\_fifo | 31:0 | RO | rx fifo for capture data | 32'h0 |

### CH6\_RX\_FIFO

Offset: 114

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch6\_rx\_fifo | 31:0 | RO | rx fifo for capture data | 32'h0 |

### CH7\_RX\_FIFO

Offset: 118

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch7\_rx\_fifo | 31:0 | RO | rx fifo for capture data | 32'h0 |

### FIFO\_STATUS\_0

Offset: 11C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch7\_rx\_fifo\_usedw | 31:29 | RO | indicate rx fifo used data | 3'h0 |
| ch7\_tx\_fifo\_usedw | 28:26 | RO | indicate tx fifo used data | 3'h0 |
| ch6\_rx\_fifo\_usedw | 25:23 | RO | indicate rx fifo used data | 3'h0 |
| ch6\_tx\_fifo\_usedw | 22:20 | RO | indicate tx fifo used data | 3'h0 |
| ch5\_rx\_fifo\_usedw | 19:17 | RO | indicate rx fifo used data | 3'h0 |
| ch5\_tx\_fifo\_usedw | 16:14 | RO | indicate tx fifo used data | 3'h0 |
| ch4\_rx\_fifo\_usedw | 13:11 | RO | indicate rx fifo used data | 3'h0 |
| ch4\_tx\_fifo\_usedw | 10:8 | RO | indicate tx fifo used data | 3'h0 |
| rx\_fifo\_full | 7:0 | RO | indicate if rx fifo is full. [7:0]: ch7 - ch0 | 8'h0 |

### FIFO\_STATUS\_1

Offset: 120

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ch3\_rx\_fifo\_usedw | 31:29 | RO | indicate rx fifo used data | 3'h0 |
| ch3\_tx\_fifo\_usedw | 28:26 | RO | indicate tx fifo used data | 3'h0 |
| ch2\_rx\_fifo\_usedw | 25:23 | RO | indicate rx fifo used data | 3'h0 |
| ch2\_tx\_fifo\_usedw | 22:20 | RO | indicate tx fifo used data | 3'h0 |
| ch1\_rx\_fifo\_usedw | 19:17 | RO | indicate rx fifo used data | 3'h0 |
| ch1\_tx\_fifo\_usedw | 16:14 | RO | indicate tx fifo used data | 3'h0 |
| ch0\_rx\_fifo\_usedw | 13:11 | RO | indicate rx fifo used data | 3'h0 |
| ch0\_tx\_fifo\_usedw | 10:8 | RO | indicate tx fifo used data | 3'h0 |
| rx\_fifo\_empty | 7:0 | RO | indicate if rx fifo is empty. [7:0]: ch7 - ch0 | 8'hff |

### FIFO\_STATUS\_2

Offset: 124

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| tx\_fifo\_full | 31:24 | RO | indicate if tx fifo is full. [31:24]: ch7 - ch0 | 8'h0 |
| tx\_fifo\_empty | 23:16 | RO | indicate if tx fifo is empty. [23:16]: ch7 - ch0 | 8'hff |
| rx\_fifo\_wrclr\_done | 15:8 | RO | set rx\_fifo\_wr\_clr will sync rx\_fifo\_wrclr signal to ir clk domain, when wrclr signal done, sync back to pclk domain, and this field will be set. [15:8]: ch7 - ch0 | 8'h0 |
| tx\_fifo\_rdclr\_done | 7:0 | RO | set tx\_fifo\_rd\_clr will sync tx\_fifo\_rdclr signal to ir clk domain, when rdclr signal done, sync back to pclk domain, and this field will be set. [7:0]: ch7 - ch0 | 8'h0 |

### FIFO\_CONFIG

Offset: 128

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| tx\_fifo\_wr\_clr | 31:24 | W1C | set 1 to clear tx fifo wr pointer. [31:24]: ch7 - ch0 | 8'h0 |
| tx\_fifo\_rd\_clr | 23:16 | W1C | set 1 to clear tx fifo rd pointer. [23:16]: ch7 - ch0 | 8'h0 |
| rx\_fifo\_wr\_clr | 15:8 | W1C | set 1 to clear rx fifo wr pointer. [15:8]: ch7 - ch0 | 8'h0 |
| rx\_fifo\_rd\_clr | 7:0 | W1C | set 1 to clear rx fifo rd pointer. [7:0]: ch7 - ch0 | 8'h0 |

### SHADOW\_SYNC

Offset: 12C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| reserved | 31:16 | RO | not used | 16'h0 |
| shadow\_sync\_mode | 15:0 | RW | shadow register update mode, only for reload/match\_0/match\_1 in PWM operation 1: update immediately, 0: update at the period end point [0]: ch0-match0 update mode [1]: ch1-match0 update mode [2]: ch2-match0 update mode [3]: ch3-match0 update mode [4]: ch4-match0 update mode [5]: ch5-match0 update mode [6]: ch6-match0 update mode [7]: ch7-match0 update mode [8]: ch0\_reload update mode [9]: ch1\_reload update mode [10]: ch2\_reload update mode [11]: ch3\_reload update mode [12]: ch4\_reload update mode [13]: ch5\_reload update mode [14]: ch6\_reload update mode [15]: ch7\_reload update mode | 16'h0 |

### SHADOW\_LOAD

Offset: 130

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| reserved | 31:16 | RO | not used | 16'h0 |
| shadow\_load\_active | 15:0 | W1S | shadow load active, only for reload/match\_0/match\_1 in PWM operation load into shadow register [0]: ch0-match0 load active [1]: ch1-match0 load active [2]: ch2-match0 load active [3]: ch3-match0 load active [4]: ch4-match0 load active [5]: ch5-match0 load active [6]: ch6-match0 load active [7]: ch7-match0 load active [8]: ch0\_reload load active [9]: ch1\_reload load active [10]: ch2\_reload load active [11]: ch3\_reload load active [12]: ch4\_reload load active [13]: ch5\_reload load active [14]: ch6\_reload load active [15]: ch7\_reload load active | 16'h0 |

### IRSR\_CH

Offset: 134

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| tx\_fifo\_underflow\_int\_irsr | 31:24 | RO | tx fifo underflow interrupt raw status. [31:24]: ch7 - ch0 | 8'h0 |
| rx\_fifo\_overflow\_int\_irsr | 23:16 | RO | rx fifo overflow interrupt raw status. [23:16]: ch7 - ch0 | 8'h0 |
| ledc\_tx\_int\_irsr | 15:8 | RO | ledc tx done interrupt raw status. [15:8]: ch7 - ch0 | 8'h0 |
| capture\_int\_irsr | 7:0 | RO | input capture interrupt raw status. [7:0]: ch7 - ch0 | 8'h0 |

### IRSR\_TIMER

Offset: 138

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| timer3\_int\_irsr | 31:24 | RO | timer 3 interrupt raw status. [31:24]: ch7 - ch0 | 8'h0 |
| timer2\_int\_irsr | 23:16 | RO | timer 2 interrupt raw status. [23:16]: ch7 - ch0 | 8'h0 |
| timer1\_int\_irsr | 15:8 | RO | timer 1 interrupt raw status. [15:8]: ch7 - ch0 | 8'h0 |
| timer0\_int\_irsr | 7:0 | RO | timer 0 interrupt raw status. [7:0]: ch7 - ch0 | 8'h0 |

### ICR\_FIFO

Offset: 13C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| reserved | 31:16 | RO | not used | 16'h0 |
| tx\_fifo\_underflow\_int\_clr | 15:8 | W1C | tx fifo underflow interrupt clear. [15:8]: ch7 - ch0 | 8'h0 |
| rx\_fifo\_overflow\_int\_clr | 7:0 | W1C | rx fifo overflow interrupt clear. [7:0]: ch7 - ch0 | 8'h0 |

# GPIO

## Introduction

该模块用于配置GPIO相关的功能，如上拉，下拉，数据置位，数据清0，中断使能，中断触发

模式是边沿还是电平。

Andes的IP，需要PDF转word

## Main Features

## Function Description

## Basic Block Diagram

## GPIO Register

### IDREV

Offset: 000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| id | 31:12 | RO |  | 20'h2031 |
| revmajor | 11:4 | RO |  | 8'h0 |
| revminor | 3:0 | RO |  | 4'h1 |

### CFG

Offset: 010

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| pull | 31:31 | RO |  | 1'h0 |
| intr | 30:30 | RO |  | 1'h0 |
| debounce | 29:29 | RO |  | 1'h0 |
| Reserved | 28:6 | RO |  | 23'h0 |
| channelnum | 5:0 | RO |  | 6'h0 |

### DATAIN

Offset: 020

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| datain | 31:0 | RO |  | 32'h0 |

### DATAOUT

Offset: 024

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| dataout | 31:0 | RW |  | 32'h0 |

### CHANNELDIR

Offset: 028

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| channeldir | 31:0 | RW |  | 32'h0 |

### DOUTCLEAR

Offset: 02C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| doutclear | 31:0 | WO |  | 32'h0 |

### DOUTSET

Offset: 030

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| doutset | 31:0 | WO |  | 32'h0 |

### PULLEN

Offset: 040

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| pullen | 31:0 | RW |  | 32'h0 |

### PULLTYPE

Offset: 044

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| pulltype | 31:0 | RW |  | 32'h0 |

### INTREN

Offset: 050

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| inten | 31:0 | RW |  | 32'h0 |

### INTRMODE0

Offset: 054

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:31 | RO |  | 1'h0 |
| ch7intrm | 30:28 | RW |  | 3'h0 |
| Reserved | 27:27 | RO |  | 1'h0 |
| ch6intrm | 26:24 | RW |  | 3'h0 |
| Reserved | 23:23 | RO |  | 1'h0 |
| ch5intrm | 22:20 | RW |  | 3'h0 |
| Reserved | 19:19 | RO |  | 1'h0 |
| ch4intrm | 18:16 | RW |  | 3'h0 |
| Reserved | 15:15 | RO |  | 1'h0 |
| ch3intrm | 14:12 | RW |  | 3'h0 |
| Reserved | 11:11 | RO |  | 1'h0 |
| ch2intrm | 10:8 | RW |  | 3'h0 |
| Reserved | 7:7 | RO |  | 1'h0 |
| ch1intrm | 6:4 | RW |  | 3'h0 |
| Reserved | 3:3 | RO |  | 1'h0 |
| ch0intrm | 2:0 | RW |  | 3'h0 |

### INTRMODE1

Offset: 058

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:31 | RO |  | 1'h0 |
| ch15intrm | 30:28 | RW |  | 3'h0 |
| Reserved | 27:27 | RO |  | 1'h0 |
| ch14intrm | 26:24 | RW |  | 3'h0 |
| Reserved | 23:23 | RO |  | 1'h0 |
| ch13intrm | 22:20 | RW |  | 3'h0 |
| Reserved | 19:19 | RO |  | 1'h0 |
| ch12intrm | 18:16 | RW |  | 3'h0 |
| Reserved | 15:15 | RO |  | 1'h0 |
| ch11intrm | 14:12 | RW |  | 3'h0 |
| Reserved | 11:11 | RO |  | 1'h0 |
| ch10intrm | 10:8 | RW |  | 3'h0 |
| Reserved | 7:7 | RO |  | 1'h0 |
| ch9intrm | 6:4 | RW |  | 3'h0 |
| Reserved | 3:3 | RO |  | 1'h0 |
| ch8intrm | 2:0 | RW |  | 3'h0 |

### INTRMODE2

Offset: 05C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:31 | RO |  | 1'h0 |
| ch23intrm | 30:28 | RW |  | 3'h0 |
| Reserved | 27:27 | RO |  | 1'h0 |
| ch22intrm | 26:24 | RW |  | 3'h0 |
| Reserved | 23:23 | RO |  | 1'h0 |
| ch21intrm | 22:20 | RW |  | 3'h0 |
| Reserved | 19:19 | RO |  | 1'h0 |
| ch20intrm | 18:16 | RW |  | 3'h0 |
| Reserved | 15:15 | RO |  | 1'h0 |
| ch19intrm | 14:12 | RW |  | 3'h0 |
| Reserved | 11:11 | RO |  | 1'h0 |
| ch18intrm | 10:8 | RW |  | 3'h0 |
| Reserved | 7:7 | RO |  | 1'h0 |
| ch17intrm | 6:4 | RW |  | 3'h0 |
| Reserved | 3:3 | RO |  | 1'h0 |
| ch16intrm | 2:0 | RW |  | 3'h0 |

### INTRMODE3

Offset: 060

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:31 | RO |  | 1'h0 |
| ch31intrm | 30:28 | RW |  | 3'h0 |
| Reserved | 27:27 | RO |  | 1'h0 |
| ch30intrm | 26:24 | RW |  | 3'h0 |
| Reserved | 23:23 | RO |  | 1'h0 |
| ch29intrm | 22:20 | RW |  | 3'h0 |
| Reserved | 19:19 | RO |  | 1'h0 |
| ch28intrm | 18:16 | RW |  | 3'h0 |
| Reserved | 15:15 | RO |  | 1'h0 |
| ch27intrm | 14:12 | RW |  | 3'h0 |
| Reserved | 11:11 | RO |  | 1'h0 |
| ch26intrm | 10:8 | RW |  | 3'h0 |
| Reserved | 7:7 | RO |  | 1'h0 |
| ch25intrm | 6:4 | RW |  | 3'h0 |
| Reserved | 3:3 | RO |  | 1'h0 |
| ch24intrm | 2:0 | RW |  | 3'h0 |

### INTRSTATUS

Offset: 064

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| intrstatus | 31:0 | W1C |  | 32'h0 |

### DEBOUNCEEN

Offset: 070

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| debounceen | 31:0 | RW |  | 32'h0 |

### DEBOUNCECTRL

Offset: 074

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| dbclksel | 31:31 | RW |  | 1'h0 |
| Reserved | 30:8 | RO |  | 23'h0 |
| dbprescale | 7:0 | RW |  | 8'h0 |

# SPI

## Introduction

Andes的IP，需要PDF转WORD

## Main Features

## Function Description

## Basic Block Diagram

## SPI Register

### IDREV

Offset: 000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| id | 31:12 | RO |  | 20'h2002 |
| revmajor | 11:4 | RO |  | 8'h0 |
| revminor | 3:0 | RO |  | 4'h1 |

### TRANSFMT

Offset: 010

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| fast\_md | 31:31 | RW | 0: mode0/mode2 slow mode, the frequency ratio of spi\_clk and SCLK should be larger than 6 1: mode0/mode2 fast mode, the freqency ration of spi\_clk and SCLK should be larger than 2 | 1'h0 |
| Reserved | 30:18 | RO |  | 13'h0 |
| addrlen | 17:16 | RW |  | 2'h0 |
| Reserved | 15:13 | RO |  | 3'h0 |
| datalen | 12:8 | RW |  | 5'h0 |
| datamerge | 7:7 | RW |  | 1'h0 |
| Reserved | 6:5 | RO |  | 2'h0 |
| mosibidir | 4:4 | RW |  | 1'h0 |
| lsb | 3:3 | RW |  | 1'h0 |
| slvmode | 2:2 | RW |  | 1'h0 |
| cpol | 1:1 | RW |  | 1'h0 |
| cpha | 0:0 | RW |  | 1'h0 |

### DIRECTIO

Offset: 014

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:25 | RO |  | 7'h0 |
| directioen | 24:24 | RW |  | 1'h0 |
| Reserved | 23:22 | RO |  | 2'h0 |
| hold\_oe | 21:21 | RW |  | 1'h0 |
| wp\_oe | 20:20 | RW |  | 1'h0 |
| miso\_oe | 19:19 | RW |  | 1'h0 |
| mosi\_oe | 18:18 | RW |  | 1'h0 |
| sclk\_oe | 17:17 | RW |  | 1'h0 |
| cs\_oe | 16:16 | RW |  | 1'h0 |
| Reserved | 15:14 | RO |  | 2'h0 |
| hold\_o | 13:13 | RW |  | 1'h0 |
| wp\_o | 12:12 | RW |  | 1'h0 |
| miso\_o | 11:11 | RW |  | 1'h0 |
| mosi\_o | 10:10 | RW |  | 1'h0 |
| sclk\_o | 9:9 | RW |  | 1'h0 |
| cs\_o | 8:8 | RW |  | 1'h0 |
| Reserved | 7:6 | RO |  | 2'h0 |
| hold\_i | 5:5 | RO |  | 1'h0 |
| wp\_i | 4:4 | RO |  | 1'h0 |
| miso\_i | 3:3 | RO |  | 1'h0 |
| mosi\_i | 2:2 | RO |  | 1'h0 |
| sclk\_i | 1:1 | RO |  | 1'h0 |
| cs\_i | 0:0 | RO |  | 1'h0 |

### TRANSCTRL

Offset: 020

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:31 | RO |  | 1'h0 |
| cmden | 30:30 | RW |  | 1'h0 |
| addren | 29:29 | RW |  | 1'h0 |
| addrfmt | 28:28 | RW |  | 1'h0 |
| transmode | 27:24 | RW |  | 4'h0 |
| dualquad | 23:22 | RW |  | 2'h0 |
| tokenen | 21:21 | RW |  | 1'h0 |
| wrtrancnt | 20:12 | RW |  | 9'h0 |
| tokenvalue | 11:11 | RW |  | 1'h0 |
| dummycnt | 10:9 | RW |  | 2'h0 |
| rdtrancnt | 8:0 | RW |  | 9'h0 |

### CMD

Offset: 024

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:8 | RO |  | 24'h0 |
| cmd | 7:0 | RW |  | 8'h0 |

### ADDR

Offset: 028

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| addr | 31:0 | RW |  | 32'h0 |

### DATA

Offset: 02C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| data | 31:0 | RW |  | 32'h0 |

### CTRL

Offset: 030

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:21 | RO |  | 11'h0 |
| txthres | 20:16 | RW |  | 5'h0 |
| Reserved | 15:13 | RO |  | 3'h0 |
| rxthres | 12:8 | RW |  | 5'h0 |
| Reserved | 7:5 | RO |  | 3'h0 |
| txdmaen | 4:4 | RW |  | 1'h0 |
| rxdmaen | 3:3 | RW |  | 1'h0 |
| txfiforst | 2:2 | RW |  | 1'h0 |
| rxfiforst | 1:1 | RW |  | 1'h0 |
| spirst | 0:0 | RW |  | 1'h0 |

### STATUS

Offset: 034

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO |  | 8'h0 |
| txfull | 23:23 | RO |  | 1'h0 |
| txempty | 22:22 | RO |  | 1'h0 |
| Reserved | 21:21 | RO |  | 1'h0 |
| txnum | 20:16 | RO |  | 5'h0 |
| rxfull | 15:15 | RO |  | 1'h0 |
| rxempty | 14:14 | RO |  | 1'h0 |
| Reserved | 13:13 | RO |  | 1'h0 |
| rxnum | 12:8 | RO |  | 5'h0 |
| Reserved | 7:1 | RO |  | 7'h0 |
| spiactive | 0:0 | RO |  | 1'h0 |

### INTREN

Offset: 038

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:6 | RO |  | 26'h0 |
| slvcmden | 5:5 | RW |  | 1'h0 |
| endinten | 4:4 | RW |  | 1'h0 |
| txfifointen | 3:3 | RW |  | 1'h0 |
| rxfifointen | 2:2 | RW |  | 1'h0 |
| txfifourinten | 1:1 | RW |  | 1'h0 |
| rxfifoorinten | 0:0 | RW |  | 1'h0 |

### INTRST

Offset: 03C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:6 | RO |  | 26'h0 |
| slvcmdint | 5:5 | W1C |  | 1'h0 |
| endint | 4:4 | W1C |  | 1'h0 |
| txfifoint | 3:3 | W1C |  | 1'h0 |
| rxfifoint | 2:2 | W1C |  | 1'h0 |
| txfifourint | 1:1 | W1C |  | 1'h0 |
| rxfifoorint | 0:0 | W1C |  | 1'h0 |

### TIMING

Offset: 040

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:14 | RO |  | 18'h0 |
| cs2sclk | 13:12 | RW |  | 2'h0 |
| csht | 11:8 | RW |  | 4'h0 |
| sclk\_div | 7:0 | RW |  | 8'h0 |

### MEMCTRL

Offset: 050

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:9 | RO |  | 23'h0 |
| memctrlchg | 8:8 | RO |  | 1'h0 |
| Reserved | 7:4 | RO |  | 4'h0 |
| memrdcmd | 3:0 | RW |  | 4'h0 |

### SLVST

Offset: 060

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:19 | RO |  | 13'h0 |
| underrun | 18:18 | W1C |  | 1'h0 |
| overrun | 17:17 | W1C |  | 1'h0 |
| ready | 16:16 | RW |  | 1'h0 |
| usr\_status | 15:0 | RW |  | 16'h0 |

### SLVDATACNT

Offset: 064

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| wcnt\_clr | 31:31 | RW | wcnt counter clear. High active | 1'h0 |
| Reserved | 30:26 | RO |  | 5'h0 |
| wcnt | 25:16 | RO |  | 10'h0 |
| rcnt\_clr | 15:15 | RO | rcnt counter clear. High active | 1'h0 |
| Reserved | 14:10 | RO |  | 5'h0 |
| rcnt | 9:0 | RO |  | 10'h0 |

### CONFIG

Offset: 07C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:15 | RO |  | 17'h0 |
| slave | 14:14 | RO |  | 1'h0 |
| eilmmem | 13:13 | RO |  | 1'h0 |
| AHBmem | 12:12 | RO |  | 1'h0 |
| directio | 11:11 | RO |  | 1'h0 |
| Reserved | 10:10 | RO |  | 1'h0 |
| quadspi | 9:9 | RO |  | 1'h0 |
| dualspi | 8:8 | RO |  | 1'h0 |
| Reserved | 7:6 | RO |  | 2'h0 |
| txfifosize | 5:4 | RO |  | 2'h0 |
| Reserved | 3:2 | RO |  | 2'h0 |
| rxfifosize | 1:0 | RO |  | 2'h0 |

# CRYPTO

## Introduction

CRYPTO 模块主要用于AES加解密和SHA运算

## Main Features

AES只支持AES128，密钥128比特，每次处理一个4 words块。

AES支持ECB，CBC模式

AES加解密一个块需要22周期

SHA为SHA256运算器，每次处理一个16words块，结果为256bits hash。

SHA处理一个块需要67周期

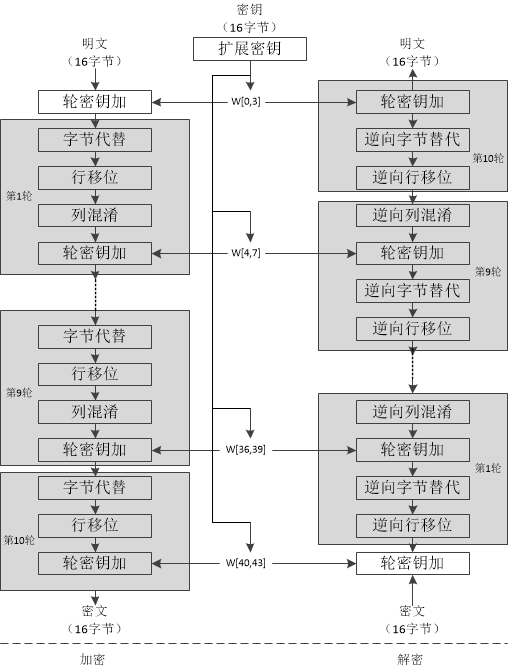
CRYPTO工作模式有以下几种

* AES encrypt only
* AES decrypt only
* SHA only
* AES encrypt with SHA
* AES decrypt with SHA

## Function Description

## Basic Block Diagram

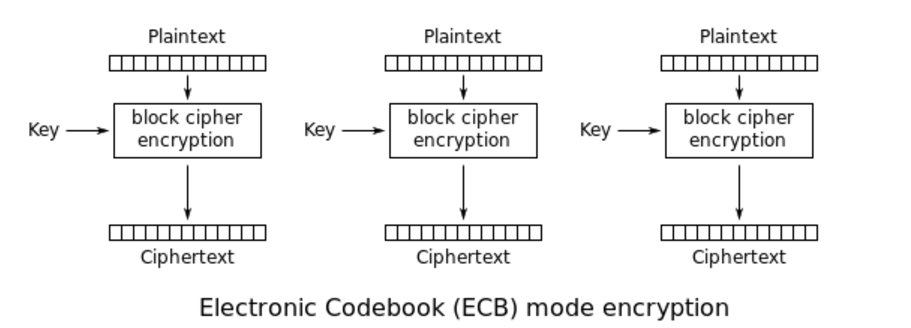
### AES加解密流程

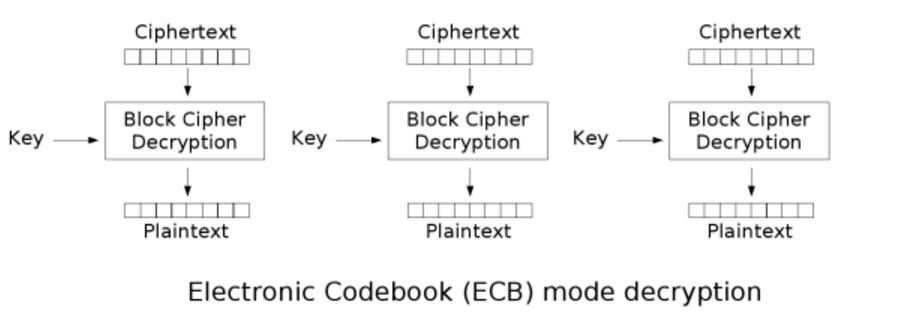


### AES mode

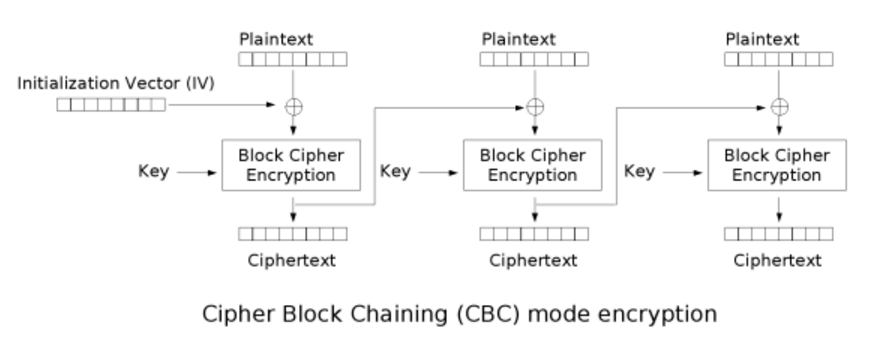
目前AES128只支持ECB和CBC模式

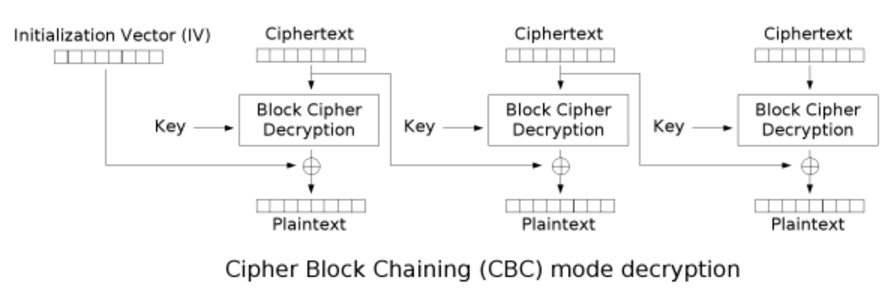
AES ECB mode





AES CBC mode





## CRYPTO Register

### CPT\_CTRL

Offset: 000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| KEY\_PRO | 31:24 | RW | Write protect KEY = 0xF5 | 8'h0 |
| Reserved | 23:1 | RO |  | 23'h0 |
| CPT\_START | 0:0 | RW | 1:start the CRYPTO engine 0:stop the CRYPTO engine | 1'h0 |

### CPT\_MODE

Offset: 004

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| KEY\_PRO | 31:24 | RW | Write protect KEY = 0xF5 | 8'h0 |
| Reserved | 23:17 | RO |  | 7'h0 |
| KEY\_UPDATE | 16:16 | RW | 1:start the update derived key procedure,  When the conditions below meet: In AES mode SHA mode disabled AES Decrypt Key\_sel == 0x3 | 1'h0 |
| Reserved | 15:10 | RO |  | 6'h0 |
| KEY\_SEL | 9:8 | RW | AES key selection: 0: user\_key 1: derived\_key, cannot be read 2: fixed test\_key 0x3:root\_key, only used in update\_derived\_key procedure | 2'h0 |
| Reserved | 7:6 | RO |  | 2'h0 |
| AES\_MODE | 5:4 | RW | 0:AES in ECB mode 1:AES in CBC mode 2:reserved 3:reserved | 2'h0 |
| Reserved | 3:3 | RO |  | 1'h0 |
| AES\_DEC | 2:2 | RW | 1: AES in decrypt mode, 0:AES in encrypt mode | 1'h0 |
| SHA\_EN | 1:1 | RW | 1: SHA enabled, 0:SHA disabled | 1'h0 |
| AES\_EN | 0:0 | RW | 1: AES enabled, 0: AES disabled | 1'h0 |

### CPT\_STATUS

Offset: 008

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:4 | RO |  | 28'h0 |
| SIZE\_ERROR | 3:3 | RO | 1: size is not 4 word aligned 0: OK | 1'h0 |
| DST\_ADDR\_ERROR | 2:2 | RO | 1: destination address not word aligned 0: no error | 1'h0 |
| ERROR | 1:1 | RO | 1: there is some error occurred. Error source is reported by Error\_id, cleared when a new start 0: No error | 1'h0 |
| DONE | 0:0 | RO | 1: Operation done or not started 0: the FCC is in operation, then some registers (CPT\_SRC\_ADDR)can’t be written | 1'h1 |

### CPT\_INT\_EN

Offset: 010

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:2 | RO |  | 30'h0 |
| ERROR | 1:1 | RW | 1:enable error report interrupt. 0:ignore | 1'h0 |
| DONE | 0:0 | RW | 1: enable done interrupt. 0: ignore | 1'h0 |

### CPT\_INT\_DIS

Offset: 014

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:2 | RO |  | 30'h0 |
| ERROR | 1:1 | RW | 1:enable error report interrupt. 0:ignore | 1'h0 |
| DONE | 0:0 | RW | 1: enable done interrupt. 0: ignore | 1'h0 |

### CPT\_INT\_MASK

Offset: 018

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:2 | RO |  | 30'h0 |
| ERROR | 1:1 | RW | 1:enable error report interrupt. 0:ignore | 1'h0 |
| DONE | 0:0 | RW | 1: enable done interrupt. 0: ignore | 1'h0 |

### CPT\_SRC\_ADDR

Offset: 020

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| SRC\_ADDR | 31:0 | RO | The start address of data to be processed | 32'h0 |

### CPT\_SRC\_SIZE

Offset: 024

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:26 | RO |  | 6'h0 |
| SRC\_SIZE | 25:4 | RW | If SHA is enabled, the SIZE must multiple of 16 WORDS Else Must multiple of 4 WORDS(for AES) | 22'h0 |
| Reserved | 3:0 | RO |  | 4'h0 |

### CPT\_DST\_ADDR

Offset: 028

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| DST\_ADDR | 31:2 | RW | The start address of AES result, must be word aligned | 30'h0 |
| Reserved | 1:0 | RO |  | 2'h0 |

### CPT\_SHA\_ST0

Offset: 030

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| SHA\_ST0 | 31:0 | RO | SHA256 status 0 | 32'h0 |

### CPT\_SHA\_ST1

Offset: 034

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| SHA\_ST1 | 31:0 | RO | SHA256 status 1 | 32'h0 |

### CPT\_SHA\_ST2

Offset: 038

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| SHA\_ST2 | 31:0 | RO | SHA256 status 2 | 32'h0 |

### CPT\_SHA\_ST3

Offset: 03C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| SHA\_ST3 | 31:0 | RO | SHA256 status 3 | 32'h0 |

### CPT\_SHA\_ST4

Offset: 040

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| SHA\_ST4 | 31:0 | RO | SHA256 status 4 | 32'h0 |

### CPT\_SHA\_ST5

Offset: 044

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| SHA\_ST5 | 31:0 | RO | SHA256 status 5 | 32'h0 |

### CPT\_SHA\_ST6

Offset: 048

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| SHA\_ST6 | 31:0 | RO | SHA256 status 6 | 32'h0 |

### CPT\_SHA\_ST7

Offset: 04C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| SHA\_ST7 | 31:0 | RO | SHA256 status 7 | 32'h0 |

### CPT\_AES\_IV0

Offset: 050

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| AES\_IV0 | 31:0 | RO | AES initialization vector 0 | 32'h0 |

### CPT\_AES\_IV1

Offset: 054

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| AES\_IV1 | 31:0 | RO | AES initialization vector 1 | 32'h0 |

### CPT\_AES\_IV2

Offset: 058

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| AES\_IV2 | 31:0 | RO | AES initialization vector 2 | 32'h0 |

### CPT\_AES\_IV3

Offset: 05C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| AES\_IV3 | 31:0 | RO | AES initialization vector 3 | 32'h0 |

### CPT\_AES\_KEY0

Offset: 060

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| AES\_KEY0 | 31:0 | RO | AES userkey 0 The key is used when key\_sel == 0x0 | 32'h0 |

### CPT\_AES\_KEY1

Offset: 064

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| AES\_KEY1 | 31:0 | RO | AES userkey 1 The key is used when key\_sel == 0x1 | 32'h0 |

### CPT\_AES\_KEY2

Offset: 068

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| AES\_KEY2 | 31:0 | RO | AES userkey 2 The key is used when key\_sel == 0x2 | 32'h0 |

### CPT\_AES\_KEY3

Offset: 06C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| AES\_KEY3 | 31:0 | RO | AES userkey 3 The key is used when key\_sel == 0x3 | 32'h0 |

### CPT\_CONFIG

Offset: 070

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| IV\_WORD\_INV | 31:31 | RW | Inv inside 4 words for initial vector only | 1'h0 |
| IV\_BYTE\_INV | 30:30 | RW | Inv inside a word for initial vector only | 1'h0 |
| KEY\_WORD\_INV | 29:29 | RW | Inv inside 4 words for key only | 1'h0 |
| KEY\_BYTE\_INV | 28:28 | RW | Inv inside a word for key only | 1'h0 |
| OUT\_WORD\_INV | 27:27 | RW | Inv inside 4 words for AES result output | 1'h0 |
| OUT\_BYTE\_INV | 26:26 | RW | Inv the bytes in output word | 1'h0 |
| WORD\_INV | 25:25 | RW | Inv inside 4 words for input | 1'h0 |
| BYTE\_INV | 24:24 | RW | Inv inside a word for input | 1'h0 |
| SHA\_INV | 23:23 | RW | Inv inside a word for SHA status register only | 1'h0 |
| Reserved | 22:18 | RW |  | 5'h0 |
| READ\_LOCK | 17:17 | RW | 0: the aes\_userkey can be read out through apb 1: return 0 when read the aes\_userkey | 1'h0 |
| WRITE\_LOCK | 16:16 | RW | 0: the aes\_userkey can be written 1:the aes\_userkey can not be written | 1'h0 |
| Reserved | 15:10 | RW |  | 6'h0 |
| SHA\_CG\_DIS | 9:9 | RW | 1:SHA global clock gated disable | 1'h0 |
| AES\_CG\_DIS | 8:8 | RW | 1:AES global clock gater disable | 1'h0 |
| Reserved | 7:2 | RW |  | 6'h0 |
| CROSS\_1K | 1:1 | RW | 1:don’t care the 1k boundary 0:split the req during across 1k boundary | 1'h0 |
| BUFFABLE | 0:0 | RW | 1:AHB req with buffable 0:AHB req with non-buffable | 1'h0 |

### CPT\_VERSION

Offset: 078

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| VERSION\_MINOR | 31:16 | RO | The current CRYPTO MINOR version ECO version | 16'h0 |
| VERSION\_MAJOR | 15:0 | RO | The current CRYPTO MAJOR version 0x200:Venus | 16'h200 |

### CPT\_FEATURE

Offset: 07C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:0 | RO |  | 32'h0 |

# WDT

## Introduction

从IP中来，需要把PDF转成WORD

## Main Features

## Function Description

## Basic Block Diagram

## WDT Register

### IDREV

Offset: 000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| id | 31:12 | RO |  | 20'h3002 |
| revmajor | 11:4 | RO |  | 8'h0 |
| revminor | 3:0 | RO |  | 4'h1 |

### CTRL

Offset: 010

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:11 | RO |  | 21'h0 |
| rsttime | 10:8 | RW |  | 3'h0 |
| inttime | 7:4 | RW |  | 4'h0 |
| rsten | 3:3 | RW |  | 1'h0 |
| inten | 2:2 | RW |  | 1'h0 |
| clksel | 1:1 | RW |  | 1'h0 |
| en | 0:0 | RW |  | 1'h0 |

### RESTART

Offset: 014

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RO |  | 16'h0 |
| restart | 15:0 | WO | ATCWDT200\_RESTART\_NUM : 16'hcafe | 16'h0 |

### WREN

Offset: 018

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RO |  | 16'h0 |
| wen | 15:0 | WO | ATCWDT200\_WP\_NUM : 16'h5aa5 | 16'h0 |

### ST

Offset: 01C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO |  | 31'h0 |
| intexpired | 0:0 | W1C |  | 1'h0 |

# IR

## Introduction

IR (Infrared Radiation) module is designed to send and receive infrared remote control signals.

　　It supports NEC, Toshiba 9012 and Philips RC5 protocols in hardware mode. It also supports other protocols when configured in software mode. The transmitted signals are modulated with carrier frequency. The typical frequency of carrier wave is 38KHz, and duty cycle is 1/3.

## Main Features

- Support AMBA APB bus access

- Support IR transmitting and receiving

- Support NEC, Toshiba 9012 and Philips RC5 protocols in hardware mode

- Support other IR protocols with software programming

- Support programmable carrier frequency and duty cycle

- Support two independent 32\*8 FIFOs

- Use clk\_ir as function clock, pclk as apb clock, and clk\_tx\_ir to generate carrier clock

- Support DMA transfer in software program mode

## Function Description

### Carrier

Why we need carrier? There are many source of Infra-Red light. The sun is the brightest source of all, while light bulbs, candles, central heating system are also the sources of Infra-Red light. Even our bodies radiate Infra-Red light. In fact everything radiates heat, and Infra-Red light.

　　To make sure our IR message receiving without errors, modulation is needed. With modulation we make IR light source blink in a particular frequency. The IR receiver will be turned to that frequency, and ignore others. In modulation, "marks" represent IR light pulsing on and off in a particular frequency. While "space" is the off state, no light emitted.

　　The carrier frequency regularly used are 38KHz, 39.2KHz, 37.916KHz, 42KHz, 40KHz, 44KHz, 38.38KHz and so on. Among them, 38KHz is the most widely used one. 90% infrared radiation transmitters/receivers use 38KHz frequency.

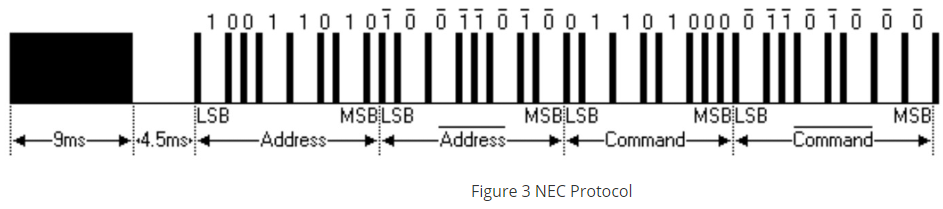
### Protocol

The design support 3 IR protocols: NEC, Toshiba 9012 and Philips RC5.

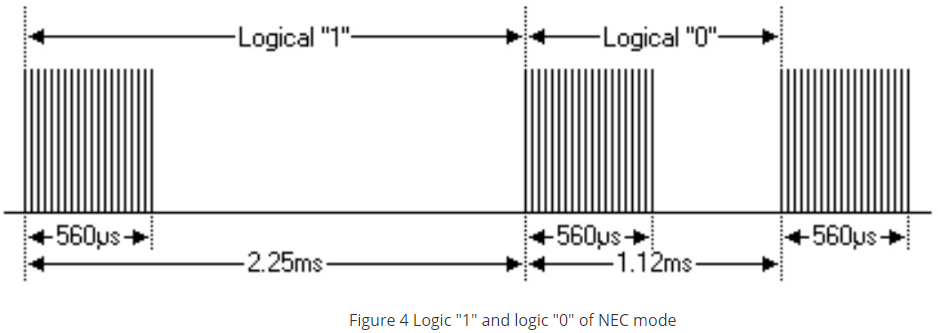
#### Hardware NEC mode

　The NEC protocol contains 8-bit address and 8-bit command. Both address and command are tranmitted twice for reliability. The carrier frequency is 38KHz. Bit time of logic "1" is 2.25ms, and bit time of logic "0" is 1.12ms.

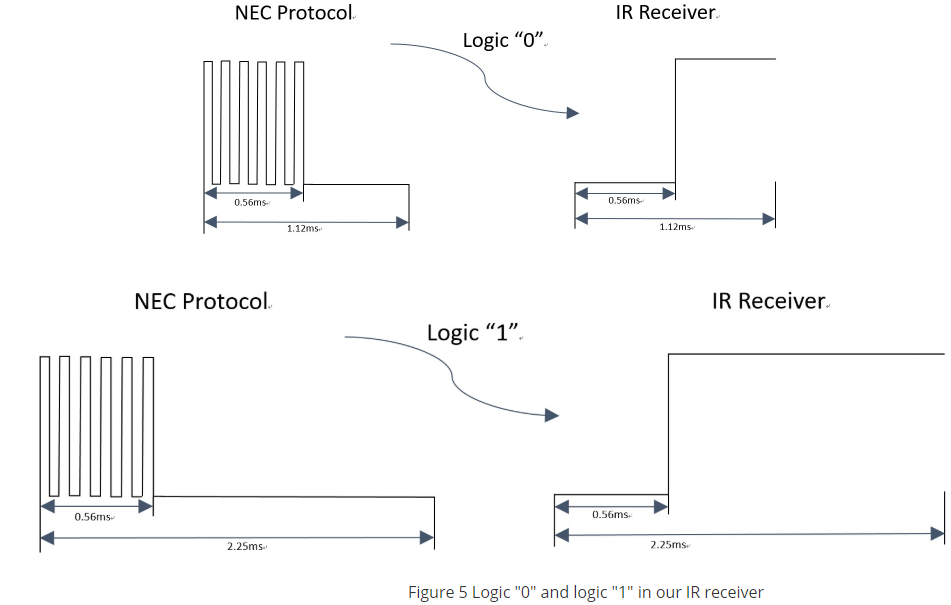
　　The NEC protocol shows below. A message is started with 9ms pulse and 4.5ms space. It's the header of a frame of a message. After that the address and command are both transmitted twice. The second time all bits are inverted and can be used for verification of the received message. The total transmission time is constant because every bit is repeated with its inverted length. During transmission the LSB is transmitted first.



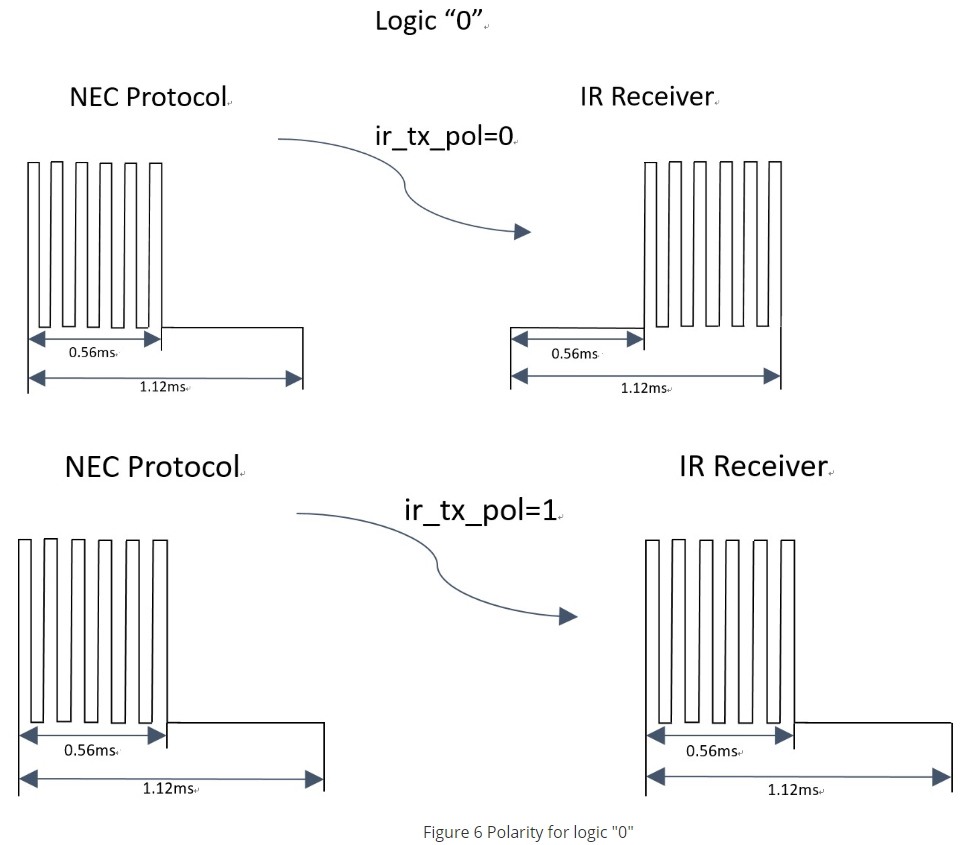
Logic "1" transmitted with 0.56ms pulse first, and total period is 2.25ms. Logic "0" transmitted with 0.56ms pulse first, and total period is 1.12ms. The picture below shows logic "1" and logic "0".



To be noticed, the "pulse" is recognized as level low, while "space" is represented as level high in our IR receivers. The logic "0" and logic "1" in our receiver is recognized as the following picture.

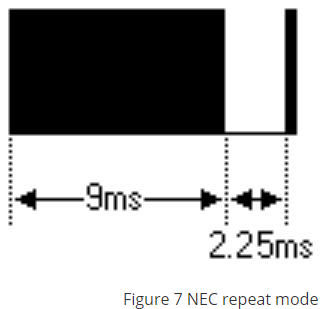


In IR transmitter the output is transmitted with carrier, and the polarity is controlled by register "ir\_tx\_pol". When "ir\_tx\_pol" is "1", the output is just the same as IR protocol. When "ir\_tx\_pol" is "0", the output "mark" and "space" is reversed. Take logic "0" as an example.



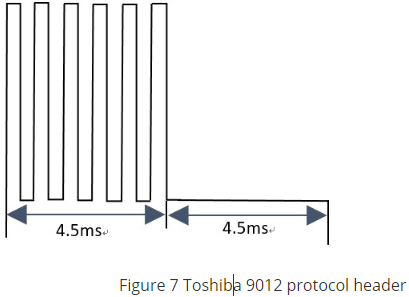
In our design the 8-bit address and 8-bit reversed address are called "usercode". To be honesty, you can write 8-bit address + 8 bit reversed bits into this register, or you can just write 16 bits. It's controlled by software. The 8-bit command and 8-bit reversed command are named as "datacode". Like the usercode, you can also write expand 16 bits or 8-bit + 8-bit reversed.

　　We also support repeat mode in NEC protocol. If the key on the remote control remains pressed, the repeat mode is triggered. Every 110ms a repeat code is transmitted as long as the key remains pressed. The repeat code is simply 9ms pulse followed by a 2.25ms space and 560us pulse.



#### Toshiba

Toshiba 9012 is very similar to NEC mode, it also contains header, usercode and datacode. Unlike NEC protocol, the header of Toshiba 9012 is 4.5ms "mark" + 4.5ms "space".

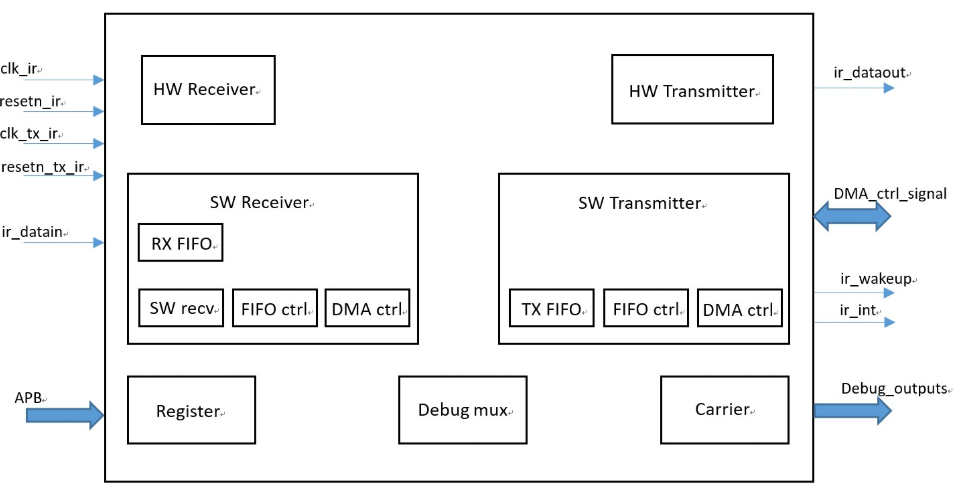


After the header usercode and datacode are transmitted with LSB first. The logic "0" and logic "1" are represented just same as NEC protocol.

#### Philips RC5

The RC-5 code from Philips is possibly the most used protocol by hobbyists, probably because of the wide availability of cheap remote controls. It contains 5-bit address and 6-bit command length. Typical carrier frequency is 36kHz, and constant bit time is 1.778ms.

## Basic Block Diagram



The IR module owns a receiver and a transmitter, both could be configured to software control mode or hardware control mode. In other words, the IR module could be configured into software TX mode, hardware TX mode, software RX mode, hardware RX mode.

　　In software mode, it performs serial-to-parallel conversion on data received from input pad, and parallel-to-serial conversion on data transmitted to output pad. CPU or DMA is responsible for dealing with parallel data between IR FIFO and external memory.

　　The RX FIFO and TX FIFO are both 32\*8, which could be written and read by the processor core over APB bus. CPU can directly write or read these entries. However, DMA operation is more recommended.



　　　　　　　　　　　　　　　　Figure 2 Data Structure

　　The data structure is shown in Figure 2. Each 32-bit value contains two field, "level" indicates whether a high/low level value was received or is going to be transmitted, and "period" is the duration (numbers of clk\_ir clk cycles) for which the level lasts. As in transmit mode, a zero period is interpreted as an end-marker. The transmitter will transmit "thres" (in register "ir\_idle\_thres") instead once it reads this value, and generated transmit ok interrupt.

　　Once the receiver is enable, it measures the duration between input signal edges. The duration measured by clk\_ir clock cycle will be written as "period", the input signal level will also be written into "level" field. The data structure is the same with transmitting. When receiver detects no change in signal level for more than "thres" clock cycle in register "ir\_idle\_thres", the receiver interrupt will be generated.

　　The carrier is generated with input clock clk\_tx\_ir, eg 6MHz, "ir\_carrier\_high" and "ir\_carrier\_low" in register "ir\_carrier\_config" are used to configure the numbers of clk\_tx\_ir clock cycles of output high and output level respectively. "ir\_tx\_pol" in register "ir\_tx\_config" controls the polarity of ir transmit output.

　　After IR transmitted or recevied, an interrupt will be raised. In hardware received mode, an interrupt generated after a complete NEC, Toshiba 9012 or Philips RC5 signals detected. In hardware transmit mode , an interrupt generated after signals transmitted. In software recevied mode, after idle threshold clock cycles of no edge detected, an interrupt is generated. In software transmit mode, a transmit data of zero length indicates the end of transmission (notice: set \*txdata\_hi\_lo\* to "1"), and an interrupt will be generated.

## Programming Guide

### NEC receive

- Configure GPIO

- IR module enable

- Set NEC mode

- Set RX mode

- Enable interrupt

- Enable IR

- Wait for interrupt

- Check IR received\_ok status, clear interrupt

### NEC transmit

- Configure GPIO

- IR module enable

- Set NEC mode

- Set TX mode

- Configure carry high and carry low

- Set time s1, s3, bit time 1, 2 and bit cycle

- Set tx usercode and datacode

- Set tx polarity

- Enable interrupt

- Enable IR

- Set tx\_start

- Wait for interrupt

- Check IR transmit\_ok status, clear interrupt

### NEC repeat receive

- Configure GPIO

- IR module enable

- Set NEC mode

- Set RX mode

- Enable interrupt

- Enable IR

- Wait for interrupt

- Check IR received\_ok status, clear interrupt

- Enable interrupt

- Check IR repest status, clear interrupt

### NEC repeat transmit

- Configure GPIO

- IR module enable

- Set NEC mode

- Set TX mode

- Configure carry high and carry low

- Set time s1, s3, bit time 1, 2 and bit cycle

- Set tx usercode and datacode

- Set tx polarity

- Enable interrupt

- Enable IR

- Set tx\_start

- Wait for interrupt

- Check IR transmit\_ok status, clear interrupt

- Set time s2, s4

- Set tx repeat mode

- Set tx\_start

- Wait for interrupt

- Check IR repest status, clear interrupt

### Toshiba 9012 receive

- Configure GPIO

- IR module enable

- Set 9012 mode

- Set RX mode

- Set time s1, s2, s3, s4

- Enable interrupt

- Enable IR

- Wait for interrupt

- Check IR received\_ok status, clear interrupt

### Toshiba 9012 transmit

- Configure GPIO

- IR module enable

- Set 9012 mode

- Set TX mode

- Configure carry high and carry low

- Set time s1, s3, bit time 1, 2 and bit cycle

- Set tx usercode and datacode

- Set tx polarity

- Enable interrupt

- Enable IR

- Set tx\_start

- Wait for interrupt

- Check IR transmit\_ok status, clear interrupt

### Philips RC5 receive

- Configure GPIO

- IR module enable

- Set RC5 mode

- Set RX mode

- Set time s1, s2, s3, s4

- Enable interrupt

- Enable IR

- Wait for interrupt

- Check IR received\_ok status, clear interrupt

### Philips RC5 transmit

* Configure GPIO
* IR module enable
* Set RC5 mode
* Set TX mode
* Configure carry high and carry low
* Set time s1, bit time 1 and bit cycle
* Set tx usercode and datacode
* Set tx polarity
* Enable interrupt
* Enable IR
* Set tx\_start
* Wait for interrupt
* Check IR transmit\_ok status, clear interrupt

### software receive

- Configure DMA for ir software receiving

- IR module enable

- Set software mode

- Set RX mode

- Configure rx DMA reg condition

- Enable rx DMA

- Enable interrupt

- Enable IR

- Wait for interrupt

- Check IR received\_ok status, clear interrupt

### software transmit

- preparing data to be transmitted

- Configure DMA for ir software transmitting

- IR module enable

- Set software mode

- Set TX mode

- Configure carry high and carry low

- Set time s1, bit time 1 and bit cycle

- Set tx usercode and datacode

- Set tx polarity

- Configure tx DMA reg condition

- Enable tx DMA

- Enable interrupt

- Enable IR

- Wait for interrupt

- Check IR transmit\_ok status, clear interrupt

## IR Register

### IR\_CTRL

Offset: 000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| sw\_hw\_mode | 31:31 | RW | control ir in software or hardware mode 0: hardware mode 1: software mode | 1'h0 |
| ir\_bit\_cycle | 30:24 | RW | bit cycle in NEC/9012 tx mode, it represents 0.56ms in bit sending. In RC5 tx mode, it represents 0.9ms, that is half cycle of one bit. | 7'h49 |
| txrx\_mode | 23:23 | RW | control tx rx mode 0: rx mode enable 1: tx mode enable | 1'h0 |
| bit\_time\_2 | 22:16 | RW | in NEC/9012, it represents cycle numbers in transmitting logic 1. | 7'h26 |
| tx\_repeat\_mode | 15:15 | RW | 0: disable tx repeat mode 1: enable tx repeat mode | 1'h0 |
| bit\_time\_1 | 14:8 | RW | in NEC/9012, it represents cycle numbers in transmitting logic 0. in RC5, it represents cycle numbers in transmitting one bit, either logic 0 or logic 1. | 7'ha |
| ir\_datacode\_verify | 7:7 | RW | 0: disable verify datacode 1: enable verify datacode | 1'h0 |
| ir\_usercode\_verify | 6:6 | RW | 0: disable verify usercode 1: enable verify usercode | 1'h0 |
| ir\_int\_verify\_en | 5:5 | RW | 0: disable ir interrupt generation by verify 1: enable ir interrupt generation by verify | 1'h0 |
| ir\_int\_en | 4:4 | RW | 0: disable ir interrupt generation 1: enable ir interrupt generation | 1'h1 |
| ir\_end\_detect\_en | 3:3 | RW | 0: disable ir detect end bit 1: disable ir detect end bit | 1'h0 |
| ir\_mode | 2:1 | RW | 0: NEC mode 1: 9012 mode 2: RC5 mode | 2'h0 |
| ir\_en | 0:0 | RW | 0: disable ir function 1: enable ir function | 1'h1 |

### IR\_TX\_CONFIG

Offset: 004

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:4 | RO | not used | 28'h0 |
| carrier\_cnt\_clr | 3:3 | W1C | write 1 clear carrier counter to 0 | 1'h0 |
| ir\_debug\_sel | 2:2 | RW | ir debug mux signal sel 0: {5'h0, ir\_tx\_out, ir\_tx\_cst} 1: {14'h0, ir\_tx\_out\_sync, ir\_tx\_carry} | 1'h0 |
| ir\_tx\_pol | 1:1 | RW | 0: change polarity 1: default | 1'h1 |
| tx\_start | 0:0 | RW | in ir hardware mode, write 1 to this bit will start ir transmit, in hardware repeat mode, will start a repeat. In software mode, no need to set this bit. | 1'h0 |

### IR\_CARRY\_CONFIG

Offset: 008

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:18 | RO | not used | 14'h0 |
| ir\_carry\_high | 17:9 | RW | config numbers of ir clk cycles of high carry | 9'he4 |
| ir\_carry\_low | 8:0 | RW | config numbers of ir clk cycles of low carry | 9'h72 |

### IR\_TIME\_1

Offset: 00C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:12 | RO | not used | 20'h0 |
| ir\_time\_s1 | 11:0 | RW | 9ms high time in NEC/9012 tx mode, it represents cycle numbers in state s1. in RC5 tx mode, it represents cycle numbers sending a bit, either logic 0 or logic 1, in state s1, s2 and toggle | 12'hff |

### IR\_TIME\_2

Offset: 010

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:12 | RO | not used | 20'h0 |
| ir\_time\_s2 | 11:0 | RW | 2.25ms low time | 12'h13f |

### IR\_TIME\_3

Offset: 014

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:12 | RO | not used | 20'h0 |
| ir\_time\_s3 | 11:0 | RW | 4.5ms low time in NEC/9012 tx mode, it represents cycle numbers in state s2. | 12'h19f |

### IR\_TIME\_4

Offset: 018

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:12 | RO | not used | 20'h0 |
| ir\_time\_s4 | 11:0 | RW | s3 max value | 12'h1bf |

### IR\_TIME\_5

Offset: 01C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:12 | RO | not used | 20'h0 |
| ir\_time\_s5 | 11:0 | RW | one frame time | 12'hc7f |

### IR\_RX\_CODE

Offset: 020

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ir\_rx\_datacode | 31:16 | RO | ir rx data code (command) | 16'h0 |
| ir\_rx\_usercode | 15:0 | RO | ir rx user code (address) | 16'h0 |

### IR\_TX\_CODE

Offset: 024

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ir\_tx\_datacode | 31:16 | RW | ir tx data code (command) | 16'h0 |
| ir\_tx\_usercode | 15:0 | RW | ir tx user code (address) | 16'h0 |

### IR\_FSM

Offset: 028

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:22 | RO | not used | 10'h0 |
| ir\_rx\_fifo\_wrclr\_done | 21:21 | RO | set rx\_fifo\_wr\_clr will sync rx\_fifo\_wrclr signal to ir clk domain, when wrclr signal done, sync back to pclk domain, and this field will be set. | 1'h0 |
| ir\_tx\_fifo\_rdclr\_done | 20:20 | RO | set rx\_fifo\_rd\_clr will sync tx\_fifo\_rdclr signal to ir clk domain, when rdclr signal done, sync back to pclk domain, and this field will be set. | 1'h0 |
| ir\_rx\_dma\_done | 19:19 | RO | indicate if rx dma transfer is done | 1'h0 |
| ir\_tx\_dma\_done | 18:18 | RO | indicate if tx dma transfer is done | 1'h0 |
| ir\_rx\_fifo\_usedw | 17:14 | RO | indicate write fifo used data | 4'h0 |
| ir\_rx\_fifo\_full | 13:13 | RO | indicate if write fifo is full | 1'h0 |
| ir\_rx\_fifo\_empty | 12:12 | RO | indicate if write fifo is empty | 1'h1 |
| ir\_tx\_fifo\_usedw | 11:8 | RO | indicate read fifo used data | 4'h0 |
| ir\_tx\_fifo\_full | 7:7 | RO | indicate if read fifo is full | 1'h0 |
| ir\_tx\_fifo\_empty | 6:6 | RO | indicate if read fifo is empty | 1'h1 |
| ir\_tx\_repeat | 5:5 | RO | indicate if ir in hardware transmit repeat mode | 1'h0 |
| ir\_transmit\_ok | 4:4 | RO | indicate if ir transmit is ok | 1'h0 |
| ir\_repeat | 3:3 | RO | indicate if ir in hardware receive mode | 1'h0 |
| ir\_datacode\_verify | 2:2 | RO | indicate if datacode is verified | 1'h0 |
| ir\_usercode\_verify | 1:1 | RO | indicate if usercode is verified | 1'h0 |
| ir\_received\_ok | 0:0 | RO | indicate if ir receive is ok | 1'h0 |

### IR\_TX\_FIFO

Offset: 02C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| txdata\_hi\_lo | 31:31 | WO | indicate high or low of transmit trans | 1'h0 |
| tx\_data\_len | 30:0 | WO | tx\_data\_len \* ir\_clock\_cycle = actual data length | 31'h0 |

### IR\_RX\_FIFO

Offset: 030

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| rxdata\_hi\_lo | 31:31 | RO | indicate high or low of receive trans | 1'h0 |
| rx\_data\_len | 30:0 | RO | rx\_data\_len \* ir\_clock\_cycle = actual data length | 31'h0 |

### IR\_DMA\_CONFIG

Offset: 034

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:10 | RO | not used | 22'h0 |
| rx\_dma\_thres\_sel | 9:6 | RW | dma requst trigger condition,  dma req will be triggered when fifo\_usedw >= thres\_sel | 4'h8 |
| tx\_dma\_thres\_sel | 5:2 | RW | dma requst trigger condition,  dma req will be triggered when (8- fifo\_usedw) >= thres\_sel | 4'h8 |
| rx\_dma\_enable | 1:1 | RW | 0: disable dma receive mode 1: enable dma receive mode | 1'h0 |
| tx\_dma\_enable | 0:0 | RW | 0: disable dma transfer mode 1: enable dma transfer mode | 1'h0 |

### IR\_FIFO\_CONFIG

Offset: 038

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:4 | RO | not used | 28'h0 |
| rx\_fifo\_wr\_clr | 3:3 | W1S | set 1 to clear rx fifo wr pointer | 1'h0 |
| rx\_fifo\_rd\_clr | 2:2 | W1S | set 1 to clear rx fifo rd pointer | 1'h0 |
| tx\_fifo\_wr\_clr | 1:1 | W1S | set 1 to clear tx fifo wr pointer | 1'h0 |
| tx\_fifo\_rd\_clr | 0:0 | W1S | set 1 to clear tx fifo rd pointer | 1'h0 |

### IR\_CLEAR\_STATUS

Offset: 03C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:7 | RO | not used | 25'h0 |
| ir\_int\_clr | 6:6 | W1S | set 1 to clear ir interrupt | 1'h0 |
| ir\_rx\_dma\_done\_clr | 5:5 | W1S | set 1 to clear rx\_dma\_done | 1'h0 |
| ir\_tx\_dma\_done\_clr | 4:4 | W1S | set 1 to clear tx\_dma\_done | 1'h0 |
| ir\_tx\_repeat\_clr | 3:3 | W1S | set 1 to clear tx repeat status | 1'h0 |
| ir\_transmit\_ok\_clr | 2:2 | W1S | set 1 to clear transmit ok status | 1'h0 |
| ir\_repeat\_clr | 1:1 | W1S | set 1 to clear rx repeat status | 1'h0 |
| ir\_received\_ok\_clr | 0:0 | W1S | set 1 to clear receive ok status | 1'h0 |

### IR\_IDLE\_THRES

Offset: 040

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:31 | RO | not used | 1'h0 |
| thres | 30:0 | RW | config this threshold to identify the cycles of idle/end state during software transmitting or receiving | 31'h500 |

# I2C

## Introduction

Andes IP 需要PDF转word

## Main Features

## Function Description

## Basic Block Diagram

## I2C Register

### IDREV

Offset: 000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ID | 31:12 | RO | ID number | 20'h20021 |
| RevMajor | 11:4 | RO | Major revision number | 8'h0 |
| RevMinor | 3:0 | RO | Minor revision number | 4'h0 |

### CFG

Offset: 010

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:2 | RW | Not used | 30'h0 |
| FIFOSIZE | 1:0 | RW | FIFO size | 2'h3 |

### INTEN

Offset: 014

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:10 | RO | Not used | 22'h0 |
| CMPL | 9:9 | RW | Set to enable the Completion Interrupt. Master: interrupts when a transaction is issued from this master and completed without losing the bus arbitration. Slave: interrupts when a transaction addressing the controller is completed. | 1'h0 |
| ByteRecv | 8:8 | RW | Set to enable the Byte Receive Interrupt. Interrupts when a byte of data is received Auto-ACK will be disabled if this interrupt is enabled, that is, the software needs to ACK/NACK the received byte manually. | 1'h0 |
| ByteTrans | 7:7 | RW | Set to enable the Byte Transmit Interrupt. Interrupts when a byte of data is transmitted. | 1'h0 |
| Start | 6:6 | RW | Set to enable the STOP Condition Interrupt. Interrupts when a STOP condition is detected. | 1'h0 |
| Stop | 5:5 | RW | Set to enable the STOP Condition Interrupt. Interrupts when a STOP condition is detected. | 1'h0 |
| ArbLos | 4:4 | RW | Set to enable the Arbitration Lose Interrupt. Master: interrupts when the controller loses the bus arbitration Slave: not available in this mode | 1'h0 |
| AddrHit | 3:3 | RW | Set to enable the Address Hit Interrupt. Master: interrupts when the addressed slave returned an ACK. Slave: interrupts when the controller is addressed. | 1'h0 |
| FIFOHALF | 2:2 | RW | Set to enable the FIFO Half Interrupt. Receiver: Interrupts when the FIFO is half-full Transmitter: Interrupts when the FIFO is half-empty This interrupt depends on the transaction direction; don’t enable this interrupt unless the transfer direction is determined, otherwise unintended interrupts may be triggered. | 1'h0 |
| FIFOFULL | 1:1 | RW | Set to enable the FIFO Full Interrupt. Interrupts when the FIFO is full. | 1'h0 |
| FIFOEMPTY | 0:0 | RW | Set to enable the FIFO Empty Interrupt. Interrupts when the FIFO is empty. | 1'h0 |

### STATUS

Offset: 018

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:15 | RO | Not used | 17'h0 |
| LineSDA | 14:14 | RO | Indicates the current status of the SDA line on the bus. 1: High 0: Low | 1'h1 |
| LineSCL | 13:13 | RO | Indicates the current status of the SCL line on the bus. 1: High 0: Low | 1'h1 |
| GenCall | 12:12 | RO | Indicates that the address of the current transaction is a general call address. This status is only valid in slave mode. 1: General call 0: Not general call | 1'h0 |
| BusBusy | 11:11 | RO | Indicates that the bus is busy. The bus is busy when a START condition is on bus and it ends when a STOP condition is seen on bus. 1: Busy 0: Not busy | 1'h0 |
| ACK | 10:10 | RO | Indicates the type of the last received/transmitted acknowledgement bit. 1: ACK 0: NACK | 1'h0 |
| CMPL | 9:9 | W1C | Transaction Completion Master: Indicates that a transaction has been issued from this master and completed without losing the bus arbitration. Slave: Indicates that a transaction addressing the controller has been completed. This status bit must be cleared to receive the next transaction; otherwise, the next incoming transaction will be blocked. | 1'h0 |
| ByteRevc | 8:8 | W1C | Indicates that a byte of data has been received. | 1'h0 |
| ByteTrans | 7:7 | W1C | Indicates that a byte of data has been transmitted. | 1'h0 |
| Start | 6:6 | W1C | Indicates that a START Condition or a repeated START condition has been transmitted/received. | 1'h0 |
| Stop | 5:5 | W1C | Indicates that a STOP Condition has been transmitted/received. | 1'h0 |
| ArbLos | 4:4 | W1C | Indicates that the controller has lost the bus arbitration (master mode only). | 1'h0 |
| AddrHit | 3:3 | W1C | Master: indicates that a slave has responded to the transaction. Slave: indicates that a transaction is targeting the controller (including the General Call). | 1'h0 |
| FIFOHALF | 2:2 | RO | Transmitter: Indicates that the FIFO is half-full. Receiver: Indicates that the FIFO is half-empty. | 1'h0 |
| FIFOFULL | 1:1 | RO | Indicates that the FIFO is full. | 1'h0 |
| FIFO\_EMPTY | 0:0 | RO | Indicates that the FIFO is empty. | 1'h1 |

### ADDR

Offset: 01C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:10 | RO | Not used | 22'h0 |
| ADDR | 9:0 | RW | The slave address. For 7-bit addressing mode, the most significant 3 bits are ignored and only the least-significant 7 bits of Addr are valid. | 10'h0 |

### DATA

Offset: 020

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:8 | RO | Not used | 24'h0 |
| Data | 7:0 | RW | Write this register to put one byte of data to the FIFO. Read this register to get one byte of data from the FIFO. | 8'h0 |

### CTRL

Offset: 024

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:13 | RO | Not used | 19'h0 |
| Phase\_start | 12:12 | RW | Enable this bit to send a START condition at the beginning of transaction. Master mode only. | 1'h1 |
| Phase\_addr | 11:11 | RW | Enable this bit to send the address after START condition. Master mode only. | 1'h1 |
| Phase\_data | 10:10 | RW | Enable this bit to send the data after Address phase. Master mode only. | 1'h1 |
| Phase\_stop | 9:9 | RW | Enable this bit to send a STOP condition at the end of a transaction. Master mode only. | 1'h1 |
| Dir | 8:8 | RW | Transaction direction Master: Set this bit to determine the direction for the next transaction. 0: Transmitter 1: Receiver Slave: The direction of the last received transaction. 0: Receiver 1: Transmitter | 1'h0 |
| DataCnt | 7:0 | RW | Data counts in bytes. Master: The number of bytes to transmit/receive. 0 means 256 bytes. DataCnt will be decreased by one for each byte transmitted/received. Slave: the meaning of DataCnt depends on the DMA mode: If DMA is not enabled, DataCnt is the number of bytes transmitted/received from the bus master. It is reset to 0 when the controller is addressed and then increased by one for each byte of data transmitted/received. If DMA is enabled, DataCnt is the number of bytes to transmit/receive. It will not be reset to 0 when the slave is addressed and it will be decreased by one for each byte of data transmitted/received. | 8'h0 |

### CMD

Offset: 028

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:3 | RO | Not used | 29'h0 |
| CMD | 2:0 | RW | Write this register with the following values to perform the corresponding actions: 0x0: no action 0x1: issue a data transaction (Master only) 0x2: respond with an ACK to the received byte 0x3: respond with a NACK to the received byte 0x4: clear the FIFO 0x5: reset the I2C controller (abort current transaction, set the SDA and SCL line to the open-drain mode, reset the Status Register and the Interrupt Enable Register, and empty the FIFO) When issuing a data transaction by writing 0x1 to this register, the CMD field stays at 0x1 for the duration of the entire transaction, and it is only cleared to 0x0 after when the transaction has completed or when the controller loses the arbitration. Note: No transaction will be issued by the controller when all phases (Start, Address, Data and Stop) are disabled. | 3'h0 |

### SETUP

Offset: 02C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:29 | RO | Not used | 3'h0 |
| T\_SUDAT | 28:24 | RW | T\_SUDAT defines the data setup time before releasing the SCL. Setup time = (4 + T\_SP + T\_SUDAT) \* tpclk tpclk = PCLK period | 5'h5 |
| T\_SP | 23:21 | RW | T\_SP defines the pulse width of spikes that must be suppressed by the input filter. Pulse width = T\_SP \* tpclk | 3'h1 |
| T\_HDDAT | 20:16 | RW | T\_SUDAT defines the data hold time after SCL goes LOW Hold time = (4 + T\_SP + T\_HDDAT) \* tpclk | 5'h5 |
| Reserved | 15:14 | RO |  | 2'h0 |
| T\_SCLRatio | 13:13 | RW | The LOW period of the generated SCL clock is defined by the combination of T\_SCLRatio and T\_SCLHi values. When T\_SCLRatio = 0, the LOW period is equal to HIGH period. When T\_SCLRatio = 1, the LOW period is roughly two times of HIGH period. SCL LOW period = (4 + T\_SP + T\_SCLHi \* ratio) \* tpclk 1: ratio = 2 0: ratio = 1 This field is only valid when the controller is in the master mode. | 1'h1 |
| T\_SCLHi | 12:4 | RW | The HIGH period of generated SCL clock is defined by T\_SCLHi. SCL HIGH period = (4 + T\_SP + T\_SCLHi) \* tpclk The T\_SCLHi value must be greater than T\_SP and T\_HDDAT values. This field is only valid when the controller is in the master mode. | 9'h10 |
| DMAEN | 3:3 | RW | Enable the direct memory access mode data transfer. 1: Enable 0: Disable | 1'h0 |
| Master | 2:2 | RW | Configure this device as a master or a slave. 1: Master mode 0: Slave mode | 1'h0 |
| Addressing | 1:1 | RW | I2C addressing mode: 1: 10-bit addressing mode 0: 7-bit addressing mode | 1'h0 |
| IICEN | 0:0 | RW | Enable the ATCIIC100 I2C controller. 1: Enable 0: Disable | 1'h0 |

# CP\_UART0

## Introduction

CP的UART0，不开放

## Main Features

## Function Description

## Basic Block Diagram

## CP\_UART0 Register

### DATA

Offset: 000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:8 | RO | Not used | 24'h0 |
| data | 7:0 | RW | Data value. Read Received data. Write Transmit data. | 8'h0 |

### STATE

Offset: 004

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:4 | RO | Not used | 28'h0 |
| rx\_buf\_overrun | 3:3 | W1C | RX buffer overrun, write 1 to clear. | 1'h0 |
| tx\_buf\_overrun | 2:2 | W1C | TX buffer overrun, write 1 to clear. | 1'h0 |
| rx\_buf\_full | 1:1 | RO | RX buffer full, read-only. | 1'h0 |
| tx\_buf\_full | 0:0 | RO | TX buffer full, read-only. | 1'h0 |

### CTRL

Offset: 008

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:7 | RO | Not used | 25'h0 |
| test\_mode | 6:6 | RW | High-speed test mode for TX only. | 1'h0 |
| rx\_overrun\_inten | 5:5 | RW | RX overrun interrupt enable. | 1'h0 |
| tx\_overrun\_inten | 4:4 | RW | TX overrun interrupt enable. | 1'h0 |
| rx\_inten | 3:3 | RW | RX interrupt enable. | 1'h0 |
| tx\_inten | 2:2 | RW | TX interrupt enable. | 1'h0 |
| rx\_en | 1:1 | RW | RX enable. | 1'h0 |
| tx\_en | 0:0 | RW | TX enable. | 1'h0 |

### INTSTATUS

Offset: 00C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:4 | RO | Not used | 28'h0 |
| rx\_overrun\_int | 3:3 | W1C | RX overrun interrupt. Write 1 to clear. | 1'h0 |
| tx\_overrun\_int | 2:2 | W1C | TX overrun interrupt. Write 1 to clear. | 1'h0 |
| rx\_int | 1:1 | RO | RX interrupt. Write 1 to clear. | 1'h0 |
| tx\_int | 0:0 | RO | TX interrupt. Write 1 to clear. | 1'h0 |

### BAUDDIV

Offset: 010

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:20 | RO | Not used | 12'h0 |
| bauddiv | 19:0 | RW | Baud rate divider. The minimum number is 16. | 20'h0 |

# SDIO

## Introduction

IP的内容，需要PDF转WORD

## Main Features

## Function Description

## Basic Block Diagram

## SDIO Register

### SDMADR

Offset: 000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| sdma\_sys\_addr\_r | 31:0 | RW |  | 32'h1 |

### BSR\_BCR

Offset: 004

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| blk\_cnt\_r | 31:16 | RW |  | 16'h0 |
| Reserved | 15:15 | RO |  | 1'h0 |
| sdma\_buf\_bound | 14:12 | RW |  | 3'h0 |
| blk\_size\_r | 11:0 | RW |  | 12'h0 |

### ARG1

Offset: 008

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| arg1\_r | 31:0 | RW |  | 32'h0 |

### TMR\_CR

Offset: 00C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:30 | RO |  | 2'h0 |
| cmd\_idx | 29:24 | RW |  | 6'h0 |
| cmd\_type | 23:22 | RW |  | 2'h0 |
| data\_pres\_sel | 21:21 | RW |  | 1'h0 |
| cmd\_idx\_chk\_en | 20:20 | RW |  | 1'h0 |
| cmd\_crc\_chk\_en | 19:19 | RW |  | 1'h0 |
| Reserved | 18:18 | RO |  | 1'h0 |
| rsp\_type\_sel | 17:16 | RW |  | 2'h0 |
| Reserved | 15:6 | RO |  | 10'h0 |
| multi\_blk\_rw | 5:5 | RW |  | 1'h0 |
| tran\_dir\_sel | 4:4 | RW |  | 1'h0 |
| auto\_cmd\_en | 3:2 | RW |  | 2'h0 |
| blk\_cnt\_en | 1:1 | RW |  | 1'h0 |
| dma\_en | 0:0 | RW |  | 1'h0 |

### RESP0

Offset: 010

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| rsp\_r | 31:0 | ROC |  | 32'h0 |

### RESP1

Offset: 014

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| rsp\_r | 31:0 | ROC |  | 32'h0 |

### RESP2

Offset: 018

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| rsp\_r | 31:0 | ROC |  | 32'h0 |

### RESP3

Offset: 01C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| rsp\_r | 31:0 | ROC |  | 32'h0 |

### BDP

Offset: 020

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| data\_port\_r | 31:0 | RW |  | 32'h0 |

### PSR

Offset: 024

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:25 | RO |  | 7'h0 |
| cmd\_lin\_lv | 24:24 | RO |  | 1'h0 |
| data\_lin\_lv | 23:20 | RO |  | 4'h0 |
| wr\_prot\_lv | 19:19 | RO |  | 1'h0 |
| cd\_pin\_lv | 18:18 | RO |  | 1'h0 |
| sys\_card\_stable | 17:17 | RO |  | 1'h0 |
| sys\_card\_insert | 16:16 | RO |  | 1'h0 |
| Reserved | 15:12 | RO |  | 4'h0 |
| buf\_ren\_r | 11:11 | ROC |  | 1'h0 |
| buf\_wen\_r | 10:10 | ROC |  | 1'h0 |
| rd\_tran\_act\_r | 9:9 | ROC |  | 1'h0 |
| wr\_tran\_act\_r | 8:8 | ROC |  | 1'h0 |
| Reserved | 7:3 | RO |  | 5'h0 |
| data\_lin\_act\_r | 2:2 | ROC |  | 1'h0 |
| cmd\_inhibit\_d | 1:1 | ROC |  | 1'h0 |
| cmd\_inhibit\_c | 0:0 | ROC |  | 1'h0 |

### HC1\_PCR\_BGCR

Offset: 028

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:20 | RO |  | 12'h0 |
| int\_at\_blk\_gap | 19:19 | RW |  | 1'h0 |
| read\_wait | 18:18 | RW |  | 1'h0 |
| cont\_req | 17:17 | RWAC |  | 1'h0 |
| sp\_blk\_gap\_req | 16:16 | RW |  | 1'h0 |
| Reserved | 15:12 | RO |  | 4'h0 |
| sd\_bus\_vol | 11:9 | RW |  | 3'h0 |
| sd\_bus\_pow | 8:8 | RW |  | 1'h0 |
| cd\_sel | 7:7 | RW |  | 1'h0 |
| cd\_test\_lv | 6:6 | RW |  | 1'h0 |
| ext\_data\_width | 5:5 | RW |  | 1'h0 |
| dma\_type | 4:3 | RW |  | 2'h0 |
| hi\_speed | 2:2 | RW |  | 1'h0 |
| data\_width | 1:1 | RW |  | 1'h0 |
| led\_ctrl | 0:0 | RW |  | 1'h0 |

### CCR\_TCR\_SRR

Offset: 02C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:27 | RO |  | 5'h0 |
| soft\_rst\_dat | 26:26 | RWAC |  | 1'h0 |
| soft\_rst\_cmd | 25:25 | RWAC |  | 1'h0 |
| soft\_rst\_all | 24:24 | RWAC |  | 1'h0 |
| Reserved | 23:20 | RO |  | 4'h0 |
| data\_timer | 19:16 | RW |  | 4'h0 |
| low\_bit\_sd\_clk\_sel | 15:8 | RW |  | 8'h0 |
| upper\_bit\_sd\_clk\_sel | 7:6 | RW |  | 2'h0 |
| clk\_gen\_sel | 5:5 | ROC |  | 1'h0 |
| Reserved | 4:3 | RO |  | 2'h0 |
| sd\_clk\_en | 2:2 | RW |  | 1'h0 |
| clk\_stable | 1:1 | ROC |  | 1'h0 |
| inter\_clk\_en | 0:0 | RW |  | 1'h0 |

### NISR\_EISR

Offset: 030

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:27 | RO |  | 5'h0 |
| tuning\_err\_r | 26:26 | RW1C |  | 1'h0 |
| adma\_err\_r | 25:25 | RW1C |  | 1'h0 |
| auto\_cmd12\_err\_r | 24:24 | RW1C |  | 1'h0 |
| cur\_lim\_err\_r | 23:23 | RW1C |  | 1'h0 |
| data\_end\_bit\_err\_r | 22:22 | RW1C |  | 1'h0 |
| data\_crc\_err\_r | 21:21 | RW1C |  | 1'h0 |
| data\_timeout\_err\_r | 20:20 | RW1C |  | 1'h0 |
| cmd\_idx\_err\_r | 19:19 | RW1C |  | 1'h0 |
| cmd\_end\_bit\_err\_r | 18:18 | RW1C |  | 1'h0 |
| cmd\_crc\_err\_r | 17:17 | RW1C |  | 1'h0 |
| cmd\_timeout\_err\_r | 16:16 | RW1C |  | 1'h0 |
| err\_interrupt\_r | 15:15 | ROC |  | 1'h0 |
| Reserved | 14:13 | RO |  | 2'h0 |
| re\_tuning\_int\_r | 12:12 | ROC |  | 1'h0 |
| int\_c\_r | 11:11 | ROC |  | 1'h0 |
| int\_b\_r | 10:10 | ROC |  | 1'h0 |
| int\_a\_r | 9:9 | ROC |  | 1'h0 |
| card\_int\_r | 8:8 | ROC |  | 1'h0 |
| card\_remove\_r | 7:7 | RW1C |  | 1'h0 |
| card\_insert\_r | 6:6 | RW1C |  | 1'h0 |
| buf\_r\_rdy\_r | 5:5 | RW1C |  | 1'h0 |
| buf\_w\_rdy\_r | 4:4 | RW1C |  | 1'h0 |
| dma\_interrupt\_r | 3:3 | RW1C |  | 1'h0 |
| blk\_gap\_evt\_r | 2:2 | RW1C |  | 1'h0 |
| tran\_complete\_r | 1:1 | RW1C |  | 1'h0 |
| cmd\_complete\_r | 0:0 | RW1C |  | 1'h0 |

### NISER\_EISER

Offset: 034

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:27 | RO |  | 5'h0 |
| tuning\_err\_st\_en | 26:26 | RW |  | 1'h0 |
| adma\_err\_st\_en | 25:25 | RW |  | 1'h0 |
| auto\_cmd12\_err\_st\_en | 24:24 | RW |  | 1'h0 |
| cur\_lim\_err\_st\_en | 23:23 | RW |  | 1'h0 |
| data\_end\_bit\_err\_st\_en | 22:22 | RW |  | 1'h0 |
| data\_crc\_err\_st\_en | 21:21 | RW |  | 1'h0 |
| data\_timeout\_err\_st\_en | 20:20 | RW |  | 1'h0 |
| cmd\_idx\_err\_st\_em | 19:19 | RW |  | 1'h0 |
| cmd\_end\_bit\_err\_st\_en | 18:18 | RW |  | 1'h0 |
| cmd\_crc\_err\_st\_en | 17:17 | RW |  | 1'h0 |
| cmd\_timeout\_err\_st\_en | 16:16 | RW |  | 1'h0 |
| Reserved | 15:13 | RO |  | 3'h0 |
| re\_tuning\_st\_en | 12:12 | RW |  | 1'h0 |
| int\_c\_st\_en | 11:11 | RO |  | 1'h0 |
| int\_b\_st\_en | 10:10 | RO |  | 1'h0 |
| int\_a\_st\_en | 9:9 | RO |  | 1'h0 |
| card\_int\_st\_en | 8:8 | RW |  | 1'h0 |
| card\_remove\_st\_en | 7:7 | RW |  | 1'h0 |
| card\_insert\_st\_en | 6:6 | RW |  | 1'h0 |
| buf\_r\_rdy\_st\_en | 5:5 | RW |  | 1'h0 |
| buf\_w\_rdy\_st\_en | 4:4 | RW |  | 1'h0 |
| dma\_interrupt\_st\_en | 3:3 | RW |  | 1'h0 |
| blk\_gap\_evt\_st\_en | 2:2 | RW |  | 1'h0 |
| tran\_complete\_st\_en | 1:1 | RW |  | 1'h0 |
| cmd\_complete\_st\_en | 0:0 | RW |  | 1'h0 |

### NISEN\_EISEN

Offset: 038

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:27 | RO |  | 5'h0 |
| tuning\_err\_sign\_en | 26:26 | RW |  | 1'h0 |
| adma\_err\_sig\_en | 25:25 | RW |  | 1'h0 |
| auto\_cmd12\_err\_sig\_en | 24:24 | RW |  | 1'h0 |
| cur\_lim\_err\_sign\_en | 23:23 | RW |  | 1'h0 |
| data\_end\_bit\_err\_sig\_en | 22:22 | RW |  | 1'h0 |
| data\_crc\_err\_sig\_en | 21:21 | RW |  | 1'h0 |
| data\_timeout\_err\_sig\_en | 20:20 | RW |  | 1'h0 |
| cmd\_idx\_err\_sig\_en | 19:19 | RW |  | 1'h0 |
| cmd\_end\_bit\_err\_sig\_en | 18:18 | RW |  | 1'h0 |
| cmd\_crc\_err\_sig\_en | 17:17 | RW |  | 1'h0 |
| cmd\_timeout\_err\_sig\_en | 16:16 | RW |  | 1'h0 |
| Reserved | 15:13 | RO |  | 3'h0 |
| re\_tuning\_sig\_en | 12:12 | RW |  | 1'h0 |
| int\_c\_sig\_en | 11:11 | RO |  | 1'h0 |
| int\_b\_sig\_en | 10:10 | RO |  | 1'h0 |
| int\_a\_sig\_en | 9:9 | RO |  | 1'h0 |
| card\_int\_sig\_en | 8:8 | RW |  | 1'h0 |
| card\_remove\_sig\_en | 7:7 | RW |  | 1'h0 |
| card\_insert\_sig\_en | 6:6 | RW |  | 1'h0 |
| buf\_r\_rdy\_sig\_en | 5:5 | RW |  | 1'h0 |
| buf\_w\_rdy\_sig\_en | 4:4 | RW |  | 1'h0 |
| dma\_interrupt\_sig\_en | 3:3 | RW |  | 1'h0 |
| blk\_gap\_evt\_sig\_en | 2:2 | RW |  | 1'h0 |
| tran\_complete\_sig\_en | 1:1 | RW |  | 1'h0 |
| cmd\_complete\_sig\_en | 0:0 | RW |  | 1'h0 |

### AC12ES\_HC2

Offset: 03C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| present\_val\_en | 31:31 | RW |  | 1'h0 |
| asyn\_int\_en | 30:30 | RW |  | 1'h0 |
| Reserved | 29:24 | RO |  | 6'h0 |
| sample\_clk\_sel | 23:23 | RW |  | 1'h0 |
| execute\_tuning | 22:22 | RWAC |  | 1'h0 |
| driver\_str\_sel | 21:20 | RW |  | 2'h0 |
| v18\_sig\_en | 19:19 | RW |  | 1'h0 |
| uhs\_mode\_sel | 18:16 | RW |  | 3'h0 |
| Reserved | 15:8 | RO |  | 8'h0 |
| cmd\_no\_ex\_by\_cmd12\_r | 7:7 | ROC |  | 1'h0 |
| Reserved | 6:5 | RO |  | 2'h0 |
| auto\_cmd\_idx\_err\_r | 4:4 | ROC |  | 1'h0 |
| auto\_cmd\_end\_bit\_err\_r | 3:3 | ROC |  | 1'h0 |
| auto\_cmd\_crc\_err\_r | 2:2 | ROC |  | 1'h0 |
| auto\_cmd\_timeout\_err\_r | 1:1 | ROC |  | 1'h0 |
| auto\_cmd12\_no\_ex\_r | 0:0 | ROC |  | 1'h0 |

### CAP0

Offset: 040

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| slot\_type | 31:30 | RW |  | 2'h0 |
| async\_int\_support | 29:29 | RO |  | 1'h0 |
| bus\_64\_support | 28:28 | RO |  | 1'h0 |
| Reserved | 27:27 | RO |  | 1'h0 |
| voltage\_1\_8\_support | 26:26 | RW |  | 1'h0 |
| voltage\_3\_0\_support | 25:25 | RW |  | 1'h0 |
| voltage\_3\_3\_support | 24:24 | RW |  | 1'h0 |
| suspend\_resume\_support | 23:23 | RW |  | 1'h0 |
| sdma\_support | 22:22 | RW |  | 1'h0 |
| hi\_speed\_support | 21:21 | RW |  | 1'h0 |
| adma1\_supoort | 20:20 | RO |  | 1'h0 |
| adma2\_support | 19:19 | RW |  | 1'h0 |
| bit\_support\_8 | 18:18 | RW |  | 1'h0 |
| max\_blk\_len | 17:16 | RW |  | 2'h0 |
| base\_clk\_for\_sd\_clk | 15:8 | RW |  | 8'h0 |
| timeout\_clk\_unit | 7:7 | RW |  | 1'h0 |
| Reserved | 6:6 | RO |  | 1'h0 |
| timeout\_clk\_freq | 5:0 | RW |  | 6'h0 |

### CAP1

Offset: 044

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO |  | 8'h0 |
| clk\_multiplier | 23:16 | RO |  | 8'h0 |
| re\_tuning\_mode | 15:14 | RO |  | 2'h0 |
| tuning\_sdr50 | 13:13 | RW |  | 1'h0 |
| Reserved | 12:12 | RO |  | 1'h0 |
| timer\_cnt\_for\_re\_tuning | 11:8 | RO |  | 4'h0 |
| Reserved | 7:7 | RO |  | 1'h0 |
| driver\_d\_support | 6:6 | RW |  | 1'h0 |
| driver\_c\_support | 5:5 | RW |  | 1'h0 |
| driver\_a\_support | 4:4 | RW |  | 1'h0 |
| Reserved | 3:3 | RO |  | 1'h0 |
| ddr50\_support | 2:2 | RW |  | 1'h0 |
| sdr104\_support | 1:1 | RW |  | 1'h0 |
| sdr50\_support | 0:0 | RW |  | 1'h0 |

### MCC0

Offset: 048

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO |  | 8'h0 |
| max\_cur\_for\_vlg\_1\_8 | 23:16 | RO |  | 8'h0 |
| max\_cur\_for\_vlg\_3\_0 | 15:8 | RO |  | 8'h0 |
| max\_cur\_for\_vlg\_3\_3 | 7:0 | RW |  | 8'h0 |

### MCC1

Offset: 04C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:0 | RO |  | 32'h0 |

### FACERR\_FERR

Offset: 050

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:29 | RO |  | 3'h0 |
| f\_ahb\_resp\_err | 28:28 | WO |  | 1'h0 |
| Reserved | 27:26 | RO |  | 2'h0 |
| f\_adma\_err | 25:25 | WO |  | 1'h0 |
| f\_auto\_cmd\_err | 24:24 | WO |  | 1'h0 |
| f\_cur\_lim\_err | 23:23 | WO |  | 1'h0 |
| f\_data\_end\_bit\_err | 22:22 | WO |  | 1'h0 |
| f\_data\_crc\_err | 21:21 | WO |  | 1'h0 |
| f\_data\_timeout\_err | 20:20 | WO |  | 1'h0 |
| f\_cmd\_idx\_err | 19:19 | WO |  | 1'h0 |
| f\_cmd\_end\_bit\_err | 18:18 | WO |  | 1'h0 |
| f\_cmd\_crc\_err | 17:17 | WO |  | 1'h0 |
| f\_cmd\_timeout\_err | 16:16 | WO |  | 1'h0 |
| Reserved | 15:8 | RO |  | 8'h0 |
| f\_auto\_cmd\_no\_ex\_by\_cmd12 | 7:7 | WO |  | 1'h0 |
| Reserved | 6:5 | RO |  | 2'h0 |
| f\_auto\_cmd\_idx\_err | 4:4 | WO |  | 1'h0 |
| f\_auto\_cmd\_end\_bit\_err | 3:3 | WO |  | 1'h0 |
| f\_auto\_cmd\_crc\_err | 2:2 | WO |  | 1'h0 |
| f\_auto\_cmd\_timeout\_err | 1:1 | WO |  | 1'h0 |
| f\_auto\_cmd12\_no\_ex | 0:0 | WO |  | 1'h0 |

### AESR

Offset: 054

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:3 | RO |  | 29'h0 |
| adma\_len\_err | 2:2 | ROC |  | 1'h0 |
| adma\_err\_st\_r | 1:0 | ROC |  | 2'h0 |

### ALSAR

Offset: 058

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| adma\_lo\_addr\_r | 31:0 | RW |  | 32'h0 |

### AHSAR

Offset: 05C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| adma\_hi\_addr\_r | 31:0 | RW |  | 32'h0 |

### PV0

Offset: 060

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RO |  | 16'h0 |
| driver\_str\_sel | 15:14 | RO |  | 2'h0 |
| Reserved | 13:11 | RO |  | 3'h0 |
| clk\_gen\_sel | 10:10 | RO |  | 1'h0 |
| sdclk\_freq\_sel | 9:0 | RO |  | 10'h0 |

### PV1

Offset: 064

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RO |  | 16'h0 |
| driver\_str\_sel | 15:14 | RO |  | 2'h0 |
| Reserved | 13:11 | RO |  | 3'h0 |
| clk\_gen\_sel | 10:10 | RO |  | 1'h0 |
| sdclk\_freq\_sel | 9:0 | RO |  | 10'h0 |

### PV2

Offset: 068

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RO |  | 16'h0 |
| driver\_str\_sel | 15:14 | RO |  | 2'h0 |
| Reserved | 13:11 | RO |  | 3'h0 |
| clk\_gen\_sel | 10:10 | RO |  | 1'h0 |
| sdclk\_freq\_sel | 9:0 | RO |  | 10'h0 |

### PV3

Offset: 06C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RO |  | 16'h0 |
| driver\_str\_sel | 15:14 | RO |  | 2'h0 |
| Reserved | 13:11 | RO |  | 3'h0 |
| clk\_gen\_sel | 10:10 | RO |  | 1'h0 |
| sdclk\_freq\_sel | 9:0 | RO |  | 10'h0 |

### HCVR

Offset: 0FC

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| vendor\_ver\_num | 31:24 | RO |  | 8'h0 |
| spec\_ver\_num | 23:16 | RW |  | 8'h0 |
| Reserved | 15:0 | RO |  | 16'h0 |

### VR0

Offset: 100

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:28 | RO |  | 4'h0 |
| ncrc | 27:24 | RW |  | 4'h0 |
| Reserved | 23:17 | RO |  | 7'h0 |
| int\_edge\_sel | 16:16 | RW |  | 1'h0 |
| Reserved | 15:14 | RO |  | 2'h0 |
| p\_lat\_off | 13:8 | RW |  | 6'h0 |
| Reserved | 7:1 | RO |  | 7'h0 |
| p\_lat\_en | 0:0 | RW |  | 1'h0 |

### VR1

Offset: 104

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:25 | RO |  | 7'h0 |
| cmd\_conflict\_en | 24:24 | RW |  | 1'h0 |
| Reserved | 23:19 | RO |  | 5'h0 |
| nsb | 18:16 | RW |  | 3'h0 |
| Reserved | 15:12 | RO |  | 4'h0 |
| ncr | 11:8 | RW |  | 4'h0 |
| Reserved | 7:4 | RO |  | 4'h0 |
| lo\_sd\_rstn | 3:3 | RO |  | 1'h0 |
| mmc\_boot\_ack\_en | 2:2 | RW |  | 1'h0 |
| mmc\_boot | 1:0 | RW |  | 2'h0 |

### VR2

Offset: 108

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO |  | 31'h0 |
| clk\_ctrl\_sw\_rst | 0:0 | RWAC |  | 1'h0 |

### VR3

Offset: 10C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:29 | RO |  | 3'h0 |
| sd\_delay\_sel\_bound | 28:24 | RW |  | 5'h0 |
| Reserved | 23:21 | RO |  | 3'h0 |
| sd\_delay\_val | 20:16 | RW |  | 5'h0 |
| Reserved | 15:13 | RO |  | 3'h0 |
| crc16\_error\_thres | 12:8 | RW |  | 5'h0 |
| Reserved | 7:5 | RO |  | 3'h0 |
| tuning\_success\_thres | 4:0 | RW |  | 5'h0 |

### VR4

Offset: 110

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| tuning\_record | 31:0 | RO |  | 32'h0 |

### VR5

Offset: 114

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:4 | RO |  | 28'h0 |
| db\_timeout | 3:0 | RW |  | 4'h0 |

### VR6

Offset: 118

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO |  | 31'h0 |
| hburst\_incr | 0:0 | RW |  | 1'h0 |

### VR7

Offset: 11C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:6 | RO |  | 26'h0 |
| stop\_sdclk\_by\_user | 5:5 | RW1C |  | 1'h0 |
| stop\_sdclk\_insert\_unstable | 4:4 | RW1C |  | 1'h0 |
| stop\_sdclk\_wait\_buf\_ava\_rd | 3:3 | RW1C |  | 1'h0 |
| stop\_sdclk\_wr\_fifo\_empty | 2:2 | RW1C |  | 1'h0 |
| stop\_sdcllk\_rd\_fifo\_afull | 1:1 | RW1C |  | 1'h0 |
| ahb\_resp\_err\_sts | 0:0 | RW1C |  | 1'h0 |

### VR8

Offset: 120

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:6 | RO |  | 26'h0 |
| stop\_sdcllk\_by\_user\_en | 5:5 | RW |  | 1'h0 |
| stop\_sdclk\_insert\_unstable\_en | 4:4 | RW |  | 1'h0 |
| stop\_sdclk\_wait\_buf\_ava\_rd\_en | 3:3 | RW |  | 1'h0 |
| stop\_sdclk\_wr\_fifo\_empty\_en | 2:2 | RW |  | 1'h0 |
| stop\_sdclk\_rd\_fifo\_afull\_en | 1:1 | RW |  | 1'h0 |
| ahb\_resp\_err\_sts\_en | 0:0 | RW |  | 1'h0 |

### VR9

Offset: 124

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:6 | RO |  | 26'h0 |
| stop\_sdclk\_by\_user\_sig\_en | 5:5 | RW |  | 1'h0 |
| stop\_sdclk\_insert\_unstable\_sig\_en | 4:4 | RW |  | 1'h0 |
| stop\_sdclk\_wait\_buf\_ava\_rd\_sig\_en | 3:3 | RW |  | 1'h0 |
| stop\_sdclk\_wr\_fifo\_empty\_sig\_en | 2:2 | RW |  | 1'h0 |
| stop\_sdclk\_rd\_fifo\_afull\_sig\_en | 1:1 | RW |  | 1'h0 |
| ahb\_resp\_err\_sig\_en | 0:0 | RW |  | 1'h0 |

### DHER

Offset: 128

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO |  | 31'h0 |
| dma\_hsk\_en | 0:0 | RW |  | 1'h0 |

### HWA

Offset: 178

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:9 | RO |  | 23'h0 |
| hw\_config | 8:0 | RO |  | 9'h0 |

### IPRR

Offset: 17C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| rev\_num | 31:0 | RO |  | 32'h0 |

### CMCR

Offset: 180

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:11 | RO |  | 21'h0 |
| swap\_hl | 10:10 | RW |  | 1'h0 |
| ch\_endian | 9:9 | RW |  | 1'h0 |
| sec\_access\_en | 8:8 | RW |  | 1'h0 |
| auto\_c2\_dcbc\_en | 7:7 | RW |  | 1'h0 |
| auto\_c2\_ecbc\_en | 6:6 | RW |  | 1'h0 |
| rngc2\_g\_en | 5:5 | RW |  | 1'h0 |
| c2\_dcbc\_en | 4:4 | RW |  | 1'h0 |
| c2\_d\_en | 3:3 | RW |  | 1'h0 |
| c2\_ecbc\_en | 2:2 | RW |  | 1'h0 |
| c2\_e\_en | 1:1 | RW |  | 1'h0 |
| c2\_g\_en | 0:0 | RW |  | 1'h0 |

### CMSR

Offset: 184

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO |  | 31'h0 |
| cp\_rdy | 0:0 | RW1C |  | 1'h0 |

### CMSE

Offset: 188

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO |  | 31'h0 |
| cp\_rdy\_sig\_en | 0:0 | RW |  | 1'h0 |

### LWID

Offset: 18C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| R\_r | 31:0 | RW |  | 32'h0 |

### HWID

Offset: 190

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| L\_r | 31:0 | RW |  | 32'h0 |

### LWIK

Offset: 194

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| key\_b\_r | 31:0 | RW |  | 32'h0 |

### HWIK

Offset: 198

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO |  | 8'h0 |
| key\_a\_r | 23:0 | RW |  | 24'h0 |

### LWOD

Offset: 19C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| cp\_out\_low\_r | 31:0 | RO |  | 32'h0 |

### HWOD

Offset: 1A0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| cp\_out\_hi\_r | 31:0 | RO |  | 32'h0 |

### SCTDP

Offset: 1A4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:8 | RO |  | 24'h0 |
| sec\_data\_port | 7:0 | RW |  | 8'h0 |

# CBUTTON

## Introduction

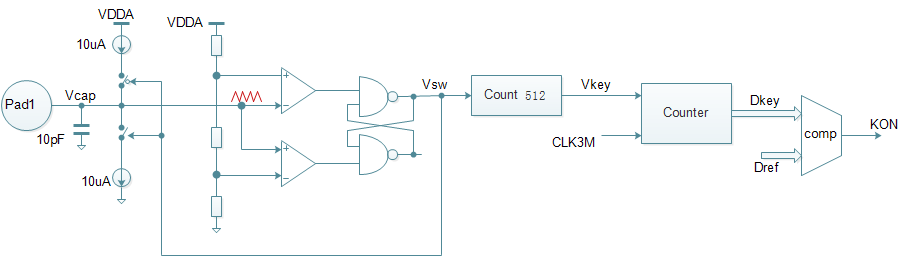
This block is used to control the touch so that correct key data can be sampled and provided to CPU through the co-working of hardware and software.

## Main Features

* APB interface to configure the touch time and configurations
* Interrupt enable is configurable through APB register
* Interrupt signal would be high to notify the CPU key press has been detected when interrupt is enabled
* Multi-key press at the same time is supported

## Function Description

## Basic Block Diagram



**Figure 3-1 Block Diagram**

Figure 3-1 shows the overall scheme of the touch detection and figure 3-2 shows the working waveform. Basically, the analog circuit before vsw is an RC oscillator. When key is pressed, the capacitance would be changed so that the frequency of the oscillator would reflect this change. The count 512 is configurable count. It is used to low the frequency and generate square waveform. The counter is used to count the time when vkey is high to detect the frequency of the analog RC oscillator. Then, by comparing the counter value with the reference value Dref, we can know whether there is key pressing or not. The reference value Dref can be got after initial power on when no key is pressed.

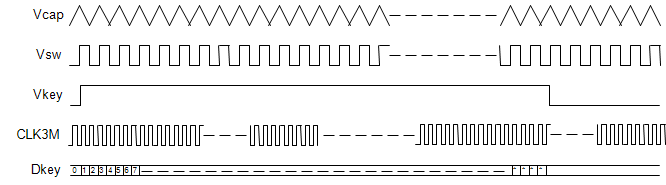


Figure 3-2 working waveform

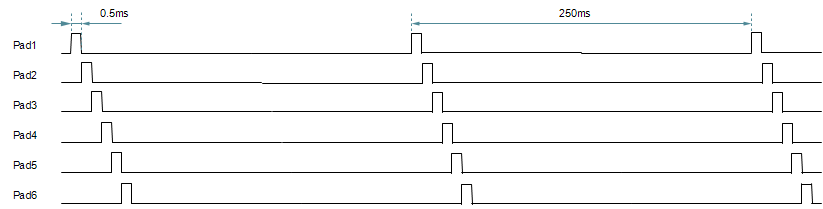


Figure 3-3, Key checking interval

Figure 3-3 shows the key checking sequence. Checking of each key can be done in a short time such as 0.5ms. The key checking of all keys can be done in adjacent time slot one by one. Then, we can power down the circuit to save power consumption for a long time such as 250ms and then start key checking again. When there is key pressing after key checking by using hardware. We can further use software to keep checking the dkey value of one key several times like the waveform shown in Figure 3-4 to make sure there is no incorrect key pressing detected.

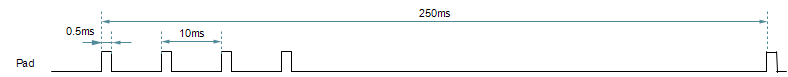


Figure 3-4, Continuous key detection of one key four times

## CBUTTON Register

### KEY0\_CONFIG

Offset: 000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:30 | RO |  | 2'h0 |
| hysteresis0 | 29:16 | RW | Hysteresis window for touch 0. Actual window is two times of this value | 14'h14 |
| Reserved | 15:15 | RO |  | 1'h0 |
| threshold0 | 14:0 | RW | Threshold value for touch 0 | 15'h64 |

### KEY1\_CONFIG

Offset: 004

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:30 | RO |  | 2'h0 |
| hysteresis1 | 29:16 | RW | Hysteresis window for touch 1. Actual window is two times of this value | 14'h14 |
| Reserved | 15:15 | RO |  | 1'h0 |
| threshold1 | 14:0 | RW | Threshold value for touch 1 | 15'h64 |

### KEY2\_CONFIG

Offset: 008

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:30 | RO |  | 2'h0 |
| hysteresis2 | 29:16 | RW | Hysteresis window for touch 2. Actual window is two times of this value | 14'h14 |
| Reserved | 15:15 | RO |  | 1'h0 |
| threshold2 | 14:0 | RW | Threshold value for touch 2 | 15'h64 |

### KEY3\_CONFIG

Offset: 00C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:30 | RO |  | 2'h0 |
| hysteresis3 | 29:16 | RW | Hysteresis window for touch 3. Actual window is two times of this value | 14'h14 |
| Reserved | 15:15 | RO |  | 1'h0 |
| threshold3 | 14:0 | RW | Threshold value for touch 3 | 15'h64 |

### KEY4\_CONFIG

Offset: 010

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:30 | RO |  | 2'h0 |
| hysteresis4 | 29:16 | RW | Hysteresis window for touch 4. Actual window is two times of this value | 14'h14 |
| Reserved | 15:15 | RO |  | 1'h0 |
| threshold4 | 14:0 | RW | Threshold value for touch 4 | 15'h64 |

### KEY5\_CONFIG

Offset: 014

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:30 | RO |  | 2'h0 |
| hysteresis5 | 29:16 | RW | Hysteresis window for touch 5. Actual window is two times of this value | 14'h14 |
| Reserved | 15:15 | RO |  | 1'h0 |
| threshold5 | 14:0 | RW | Threshold value for touch 5 | 15'h64 |

### TRAINING

Offset: 018

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:29 | RO | not used | 3'h0 |
| soft\_reset | 28:28 | WO | Write 1 to initiate a pulse to reset the touch logic so that the touch logic can start from beginning including the key training process | 1'h0 |
| cnt\_cfg | 27:20 | RW | Settings used for the clock divider to generate square wave of vkey from analog input vsw. actual divider value is two times of this value | 8'h21 |
| training\_time | 19:0 | RW | Initial training time for each touch pad after initial power on. 150ms when 3M clock is used. | 20'h6ddd0 |

### WAIT

Offset: 01C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:30 | RO | not used | 2'h0 |
| stable\_time | 29:20 | RW | stable time after changing channle (sel) of the touch 0.2ms when 3M clock is used | 10'h258 |
| wait\_time | 19:0 | RW | wait time to enter key checking after clk3m is stable, suggest to have different value during normal power and power down state. During poer down state, suggest to give this time a little bit large than half of the time when clk3m (ROSC3M) is on. 250ms in normal power state when 3M clock is always on | 20'hb71b0 |

### CONFIG

Offset: 020

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:14 | RO | not used | 18'h0 |
| sw\_sel | 13:11 | RW | Select which touch is used to detect the key pressing when sw\_detect is high | 3'h0 |
| sw\_start | 10:10 | RW | High effective signal to start key checking when sw\_detect is high | 1'h0 |
| sw\_detect | 9:9 | RW | High effective signal to mean that software is used to control the analog circuit and detect the key value | 1'h0 |
| hw\_en | 8:8 | RW | High effective signal to enable the hardware detection of the touch key | 1'h0 |
| cpu\_clear | 7:7 | RW | High effective signal to clear the kon signal so that the next round of key search can begin. Software need to write 1 to clear the kon signal and then write 0 after kon has been cleared so that the following key detection can begin. 200K if PCLK is 200M | 1'h0 |
| cbutton\_int\_en | 6:6 | RW | High effective interrupt enable | 1'h0 |
| touch\_sel | 5:0 | RW | High effective signal to select which touch pad is enabled | 6'h0 |

### ANALOG\_CTRL

Offset: 024

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:20 | RO | not used | 12'h0 |
| rco3m\_cfg | 19:17 | RW | Configure control of the oscillator.  \*\_CFG[2]: Hysteresis Control of the internal level-shifter.  1’b0: enable the hysteresis;  1’b1: disable the hysteresis. \*\_CFG[1]: target frequency control  1’b0: set the target frequency to 4MHz;  1’b1: set the target frequency to 3MHz  \*\_CFG[0]: Reserved | 3'h2 |
| rco3m\_tune | 16:11 | RW | Frequency tuning control of the oscillator. The step is about 1% typically.  For example, when the target frequency is 3MHz,  6’b100001: +1%  6’b100000: 0 (Default);  6’b011111: -1%; | 6'h20 |
| ibctl | 10:8 | RW | Control the bias current of amplifiers | 3'h4 |
| ictl | 7:5 | RW | Control the charge/discharge current | 3'h4 |
| cctl | 4:0 | RW | Control the different input cap. | 5'h4 |

### REF0

Offset: 028

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RO | not used | 16'h0 |
| dref0 | 15:0 | RO | read out reference value of touch0. Reset value is the reset value of the threshold | 16'hc8 |

### REF1

Offset: 02C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RO | not used | 16'h0 |
| dref1 | 15:0 | RO | read out reference value of touch1. Reset value is the reset value of the threshold | 16'hc8 |

### REF2

Offset: 030

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RO | not used | 16'h0 |
| dref2 | 15:0 | RO | read out reference value of touch2. Reset value is the reset value of the threshold | 16'hc8 |

### REF3

Offset: 034

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RO | not used | 16'h0 |
| dref3 | 15:0 | RO | read out reference value of touch3. Reset value is the reset value of the threshold | 16'hc8 |

### REF4

Offset: 038

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RO | not used | 16'h0 |
| dref4 | 15:0 | RO | read out reference value of touch4. Reset value is the reset value of the threshold | 16'hc8 |

### REF5

Offset: 03C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RO | not used | 16'h0 |
| dref5 | 15:0 | RO | read out reference value of touch0. Reset value is the reset value of the threshold | 16'hc8 |

### INT\_IRSR

Offset: 040

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:18 | RO | not used | 14'h0 |
| krel\_int\_irsr | 17:12 | RO | read out key being released raw momentary status for touch[5:0] | 6'h0 |
| kpre\_int\_irsr | 11:6 | RO | read out key being pressed raw momentary status for touch[5:0] | 6'h0 |
| kon\_int\_irsr | 5:0 | RO | read out key being pressed value for touch[5:0]. Level triggered | 6'h0 |

### RAW\_DATA0

Offset: 044

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:19 | RO | not used | 13'h0 |
| vsw | 18:18 | RO | the raw vsw value from the analog circuit | 1'h0 |
| vkey | 17:17 | RO | the raw vkey value just after the first clock divider | 1'h0 |
| raw | 16:0 | RO | read out raw data of touch0 | 17'h0 |

### RAW\_DATA1

Offset: 048

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:19 | RO | not used | 13'h0 |
| vsw | 18:18 | RO | the raw vsw value from the analog circuit | 1'h0 |
| vkey | 17:17 | RO | the raw vkey value just after the first clock divider | 1'h0 |
| raw | 16:0 | RO | read out raw data of touch1 | 17'h0 |

### RAW\_DATA2

Offset: 04C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:19 | RO | not used | 13'h0 |
| vsw | 18:18 | RO | the raw vsw value from the analog circuit | 1'h0 |
| vkey | 17:17 | RO | the raw vkey value just after the first clock divider | 1'h0 |
| raw | 16:0 | RO | read out raw data of touch1 | 17'h0 |

### RAW\_DATA3

Offset: 050

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:19 | RO | not used | 13'h0 |
| vsw | 18:18 | RO | the raw vsw value from the analog circuit | 1'h0 |
| vkey | 17:17 | RO | the raw vkey value just after the first clock divider | 1'h0 |
| raw | 16:0 | RO | read out raw data of touch1 | 17'h0 |

### RAW\_DATA4

Offset: 054

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:19 | RO | not used | 13'h0 |
| vsw | 18:18 | RO | the raw vsw value from the analog circuit | 1'h0 |
| vkey | 17:17 | RO | the raw vkey value just after the first clock divider | 1'h0 |
| raw | 16:0 | RO | read out raw data of touch1 | 17'h0 |

### RAW\_DATA5

Offset: 058

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:19 | RO | not used | 13'h0 |
| vsw | 18:18 | RO | the raw vsw value from the analog circuit | 1'h0 |
| vkey | 17:17 | RO | the raw vkey value just after the first clock divider | 1'h0 |
| raw | 16:0 | RO | read out raw data of touch1 | 17'h0 |

### STEP\_SIZE

Offset: 05C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RO | not used | 16'h0 |
| up\_size | 15:8 | RW | Move up size when key reference value is adjusted upside. Actual size is up\_size+1, 0 means move up by 1 | 8'h0 |
| down\_size | 7:0 | RW | Move down size when key reference value is adjusted upside. Actual size is down\_size+1, 0 means move down by 1 | 8'h0 |

### INT\_ISR

Offset: 060

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:18 | RO | not used | 14'h0 |
| krel\_int\_isr | 17:12 | RO | read out key being released raw momentary status for touch[5:0] | 6'h0 |
| kpre\_int\_isr | 11:6 | RO | read out key being pressed raw momentary status for touch[5:0] | 6'h0 |
| kon\_int\_isr | 5:0 | RO | read out key being pressed value for touch[5:0]. Level triggered | 6'h0 |

### INT\_IMR

Offset: 064

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:18 | RO | not used | 14'h0 |
| krel\_int\_mask | 17:12 | RW | Key release int mask for touch[5:0] | 6'h3f |
| kpre\_int\_mask | 11:6 | RW | Key press int mask for touch[5:0] | 6'h3f |
| kon\_int\_mask | 5:0 | RW | Key on int mask for touch[5:0] | 6'h3f |

### INT\_ICR

Offset: 068

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:12 | RO | not used | 20'h0 |
| krel\_int\_clr | 11:6 | W1C | Key release int clear for touch[5:0] | 6'h0 |
| kpre\_int\_clr | 5:0 | W1C | Key press int clear for touch[5:0] | 6'h0 |

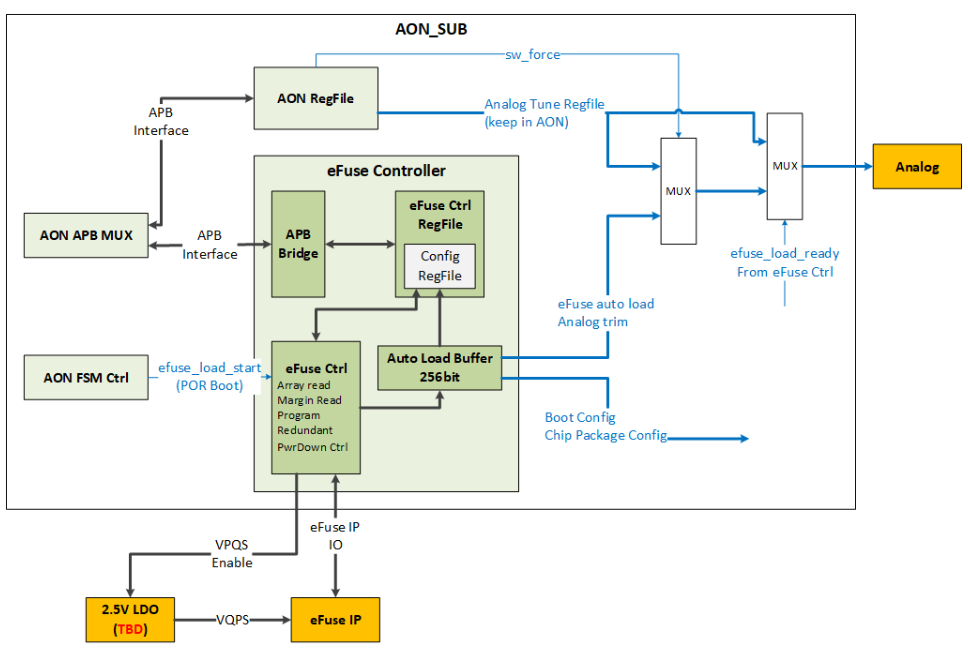
# AON\_EFUSE\_CTRL

## Introduction

## Main Features

## Function Description

## Basic Block Diagram



## Programming Guide

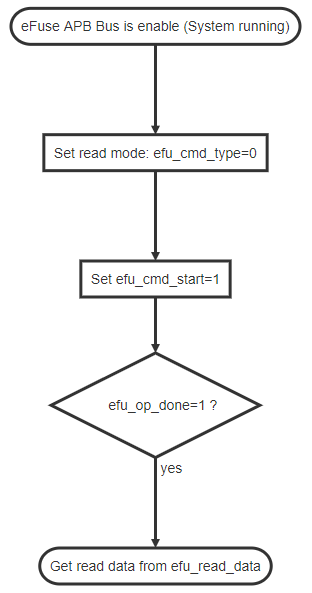
### Array and Margin Read

Note:

1. For register field efu\_cmd\_addr, as the assignment of eFuse bits are always 32-bit. Hence:

- Address [3:0] are 32bit word selection 0 ~ 15.

- Address [8:4] are NOT used in read operation



### Program Mode

Note:

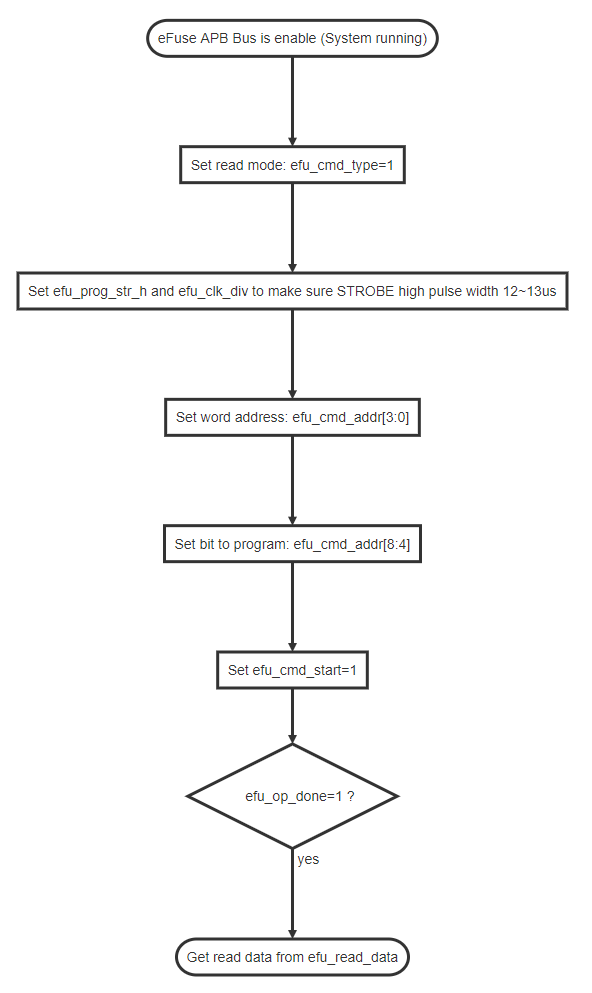
1. For register field efu\_cmd\_addr in program operation:

- Address [8:4] are 32bit selection to program a single bit in each STROBE high pulse.

- Address [3:0] are still 32bit word selection 0 ~ 15

2. eFuse bit NOT programmed, is \*\*default to 0\*\*

3. A bit can ONLY programmed to \*\*1\*\* and \*\*ONLY program once\*\*. Multiple times programming a single bit will cause device error.



### Auto-Load Mode

Note

1. This mode is only start in POR procedure.
2. **First 256 bits** including analog trim, boot configuration and Chip ID are auto-loaded.
3. The auto-load bits can be read from auto\_load\_buffer31:0 ~ auto\_load\_buffer31:0 in eFuse regfile.
4. When auto-load is done, a efuse\_auto\_load\_ready bit can be read from Always-On regfile.

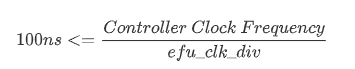
### eFuse Controller Clock Config

#### eFuse Access Timing Requirements

1. CSB/STROBE/PGENB/LOAD信号的切换，均是基于： eFuse controller clock / efu\_clk\_div

2. eFuse Read Timing

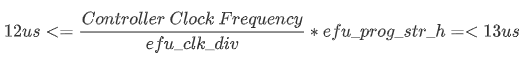
- CSB/STROBE的Setup/Hold分别需要有100ns以上的margin，只需要配置 \*\*efu\_clk\_div\*\*



3. eFuse Program Timing

- 要配置时钟满足eFuse Program STROBE=1 高电平为12-13us

- 要配置 \*\*efu\_clk\_div\*\* 和 \*\*efu\_prog\_str\_h\*\*

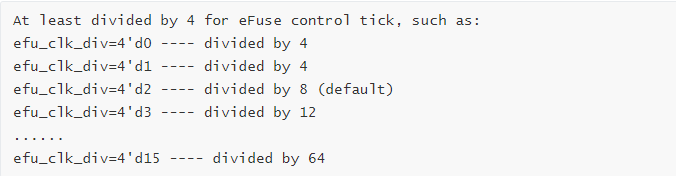


#### Default PCLK: XTAL 24MHz

##### eFuse Controller Clock default: XTAL 24MHz

XTAL 24MHz是系统初始时钟，确保eFuse auto-load正常

##### efu\_clk\_div = 2 (default)

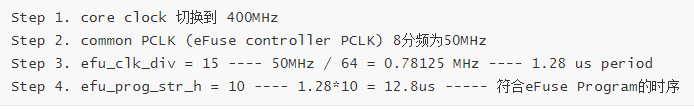


##### XTAL 24MHz / 8 = 3 MHz (default)\*\*

eFuse Read STROBE=1 高电平持续时间333ns，也就是3 MHz的一个tick cycle，确保符合eFuse Read的时序（worst case需要100ns）

#### High Speed PCLK: based on PLL core clock

如果System core clock使用400MHz PLL clock source，eFuse controller PCLK配置参考如下：



## AON\_EFUSE\_CTRL Register

### STA

Offset: 000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:10 | RO | Not used | 22'h0 |
| efu\_auto\_ld\_ready | 9:9 | ROI | Status of auto-load data ready. | 1'h0 |
| efu\_auto\_ld\_busy | 8:8 | ROI | Status of auto-load busy flag. | 1'h0 |
| Reserved | 7:4 | RO | Not used | 4'h0 |
| efu\_ctrl\_state | 3:0 | ROI | eFuse control state for SW read-only | 4'h0 |

### INT\_EN

Offset: 004

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:17 | RO | Not used | 15'h0 |
| efu\_done\_intr | 16:16 | W1C | fuse program/read command done. It doesn’t mean that the command is successful. | 1'h0 |
| Reserved | 15:1 | RW | Not used | 15'h0 |
| efu\_done\_intr\_en | 0:0 | RW | 1: generate interrupt after EFuse program done; 0: don’t generate interrupt | 1'h0 |

### CMD\_CTL

Offset: 008

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:23 | RO | Not used | 9'h0 |
| efu\_redundancy\_ena\_b | 22:22 | RW | eFuse RSB Input Redundancy enable/disable (active low). | 1'h1 |
| efu\_redundancy\_row\_sel | 21:21 | RW | eFuse RWL Input Redundancy information row select (active high) | 1'h0 |
| efu\_margin\_rd | 20:20 | RW | Read trip point setting: MR = L for normal read mode; MR = H for margin read1 mode | 1'h0 |
| efu\_pwd\_down | 19:19 | RW | eFuse power-down control. This bit needs to set or clear by SW 1: eFuse in power-down mode 0: eFuse normal operation | 1'h0 |
| efu\_cmd\_type | 18:18 | RW | eFuse command mode 1: program eFuse 0: read eFuse | 1'h0 |
| imm\_shadow | 17:17 | RW | If asserted to be 1’b1, the programed ( when programming is successful ) or readd bits will be shown in the corresponding apb-accessible shadow registers immediately, unless prevented by the security schemes of the eFuse. | 1'h0 |
| efu\_cmd\_start | 16:16 | W1S | Fuse cmd start. It will be auto cleared when either no programing is done, or program/read are done. | 1'h0 |
| Reserved | 15:9 | RW | Not used | 7'h1 |
| efu\_cmd\_addr | 8:0 | RW | Fuse program/read byte address.  As the assignment of eFuse bits are always 32-bit. Address [3:0]: 32bit word selection 0 ~ 15  Address [8:4]: Not used in read operation For 32bit selection in program operation. Program a single bit in each STROBE high pulse. | 9'h0 |

### PARA1

Offset: 014

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| efu\_prog\_standby | 31:28 | RW | standby cycles needed before exit program operations | 4'h1 |
| efu\_prog\_str\_l | 27:18 | RW | fuse parameter, program strobe low time | 10'h26 |
| efu\_prog\_str\_h | 17:8 | RW | fuse parameter, program strobe high time | 10'h26 |
| efu\_vdd\_hd | 7:4 | RW | fuse parameter, program voltage source VDDQ hold time | 4'h1 |
| efu\_vdd\_su | 3:0 | RW | fuse parameter, program voltage source VDDQ setup time | 4'h1 |

### PARA2

Offset: 018

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:4 | RO | Not used | 28'h0 |
| efu\_clk\_div | 3:0 | RW | Divide the eFuse controller clock to a slower frequency for eFuse control At least divided by 4 for eFuse control tick, such as: efu\_clk\_div=4'd0 ---- divided by 4 efu\_clk\_div=4'd1 ---- divided by 4 efu\_clk\_div=4'd2 ---- divided by 8 (default) efu\_clk\_div=4'd3 ---- divided by 12 ...... efu\_clk\_div=4'd15 ---- divided by 64 | 4'h2 |

### RD\_DATA

Offset: 01C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| efu\_read\_data | 31:0 | ROI | eFuse read data | 32'h0 |

### AUTO\_LOAD\_START

Offset: 020

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO | Not used | 31'h0 |
| efu\_auto\_ld\_start | 0:0 | W1P | Should be written as 0xCAFEEF01 to start a eFuse auto-load procedure.  This start signal is a pulse to eFuse controller and the auto-load procedure will be automatically finished, hence this bit will always read-back 0. If the auto-load procedure is NOT finished, SW can read efu\_auto\_ld\_busy as 1. | 1'h0 |

### PROG\_PROTECT

Offset: 028

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:4 | RO | Not used | 28'h0 |
| efu\_redundancy\_rf | 3:0 | ROI | RF3~RF0 Output Output high once the redundancy bit has been used. | 4'h0 |

### PROG\_PROTECT

Offset: 028

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:4 | RO | Not used | 28'h0 |
| efu\_redundancy\_rf | 3:0 | ROI | RF3~RF0 Output Output high once the redundancy bit has been used. | 4'h0 |

### AUTO\_LOAD

Offset: 080

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| auto\_load\_word | 31:0 | ROI | eFuse auto-load word bit [31:0]+n\*32 | 32'h0 |

### DIRECT\_RD

Offset: 1000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| direct\_rd\_data | 31:0 | ROI | Directly read eFuse address via the address mapping. E.g. 0x1000 - eFuse 32bit word 0, eFuse address[3:0] 0x0 0x1004 - eFuse 32bit word 1, eFuse address[3:0] 0x1 0x1008 - eFuse 32bit word 2, eFuse address[3:0] 0x2 …… 0x103C - eFuse 32bit word F, eFuse address[3:0] 0xF | 32'h0 |

# PSRAM\_MC

## Introduction

## Main Features

1. support octal SPI with DDR mode(APM protocol PSRAM)
2. clock up to 200Mhz
3. asynchronous working mode with async-FIFO
4. 8-bit command/address/data DQ bus
5. Support 1 chips, support up to 16Mbytes
6. Register configurable write and read initial latencies
7. Support wide operating temperature range by tCEM config register to guarantee robust self-refreshing
8. Support DDR PSRAM with 1K/2K Byte page only.
9. DDR PSRAM controller’s register base address:0xF0800000
10. DDR PSRAM’s memory start:0x6000\_0000(CP), 0x3000\_0000(AP)
11. PVT variation resistance(occasionally re-calibration of delay lines)
12. APM protocol bus protocol compatible
13. Supports up to 5 AHB masters with configurable priority

* fixed priority, round-robin priority and hybrid priority

1. Supports AHB write auto-concatenation
2. Supports AHB read auto-concatenation

## Function Description

## Basic Block Diagram

Below in Fig 1 is the structure of this module.



Fig. 1 Detail Structure

Notes: The accesses of PSRAM mode registers and memory share the same read data FIFO, which has a risk of reading or writing an unknown data when a register and memory access switching happens. A solution is to use separated read FIFOs.

DDR PSRAM controller has some major modules:

DDR AHB slave: AHB memory access from CPU, DMA, Luna, AP2CP master, etc.

DDR PSRAM register: APB register access

DDR interface： DDR signal generation

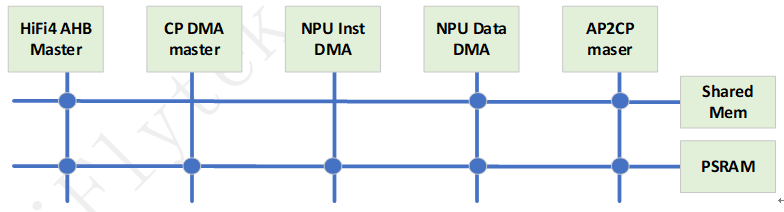
Rd and wr delay chains: delay chains for write and read clock in PHY

FIFOs: 8 FIFOs used for command store and read data buffer

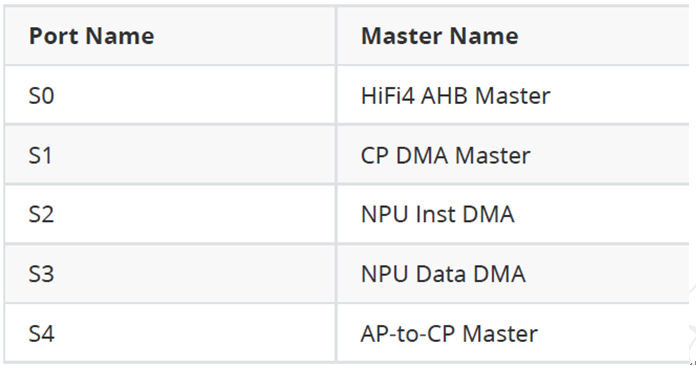
## Sub-modules

### PSRAM controller

#### 系统连接框图



5个Master port各自通过AHB总线连接PSRAM controller，经控制器内部的arbiter访问 PSRAM，port0～port5在Venus项目中实际连接的master如下表所示。



各个端口访问PSRAM的地址，参见Venus\_Chip\_Sepecification的address mapping章节。

### arbiter和FIFO

三种优先级配置：

1. Fixed priority （priority\_ctrl寄存器配置为全0）

Port0 具有最高优先级。port0 至 port5优先级依次降低。

1. 混合优先级

各个port根据priority\_ctrl寄存器对应位所配置的值，分成两组：配置值为1的归并到group1，反之为group0.

group0和group1内各自为round robin；group1和group0为固定优先级：group1高于group1。

3）round-roubin （priority\_ctrl寄存器配置为全1）

各个port的AHB读/写命令，经arbiter之后：

1. 读或写命令进入深度为8的cmd FIFO; 写数据进入32x32-bit wdata FIFO.
2. 每一个port有独立的32x32-bit read FIFO。可最多缓存两个burst16读自动拼接时读取的数据。
3. 所有的port共享一个32x32-bit wdata FIFO.

无正在执行的读命令的情况下，可缓存AHB 写数据个数：

最多2个burst16，或4个burst8，或8个burst4/sngl

### AHB读访问

|  |  |
| --- | --- |
| **读访问类型** | **支持特性** |
| 32-bit WORD  1）sngl 访问 2）INCR 访问（不定长） 3）burst 4/8/16 4）wrap 4/8/16 | 1) wrap类型的访问，必须配置psram mode register 8 bit[2:0]选择合适的wrap模式  2) 读自动拼接：  - 使能ddr\_preft寄存器的rconcat位，支持burst 4/8/16和不定长INCR；设置psram mode register 8 的bit[2:0]==3’b011,即wrap 1K。  - 不支持wrap 4/8/16、SNGL类型的读拼接 |
| BYTE/hword： 1）sngl 访问 2）INCR 访问（不定长） 3）burst 4/8/16 4）wrap 4/8/16 | 不支持自动拼接，且所有的访问都拆分为SNGL读访问。 |
| RAW （read after write）hit | 支持简单的RAW （read after write）hit功能：  若读和写都保持相同长度的访问，如同为burst4，则可使能RAW hit功能。  当新的读命令进入cmd fifo时，若cmd fifo中还有等待的psram 写命令，而且其地址和读命令的地址相同，则直接从 wdata fifo中读取数据，提高访问速度。 |

* + 1. AHB写访问

|  |  |
| --- | --- |
| **写访问类型** | **支持特性** |
| 32-bit WORD 1）sngl 访问 2）INCR 访问（不定长） 3）burst 4/8/16 4）wrap 4/8/16 | 1) wrap类型的访问，必须配置psram mode register 8 bit[2:0]选择合适的wrap模式 2) 写自动拼接：  - 使能ddr\_preft寄存器的wconcat位，支持burst 4/8/16；设置psram mode register 8 的bit[2:0]==3’b011,即wrap 1K。  - 不支持wrap 4/8/16，不定长INCR，SNGL。 |
| AHB 写访问(BYTE/hword)： 1）sngl 访问 2）INCR 访问（不定长） 3）burst 4/8/16 4）wrap 4/8/16 | 1）支持自动将byte/hword burst4/8/16 数据拼接成32位后一次写入psram，但不支持前后地址连续burst的自动拼接。 |

注意：

当write dta/cmd buffer中有写数据未清空时，新的读命令必须等待。

## PHY

PHY本身不主动更新DLL。MC通过ctrlupd\_req来发起一次slave DLL更新：复位read FIFO，更新DLL delay。MC可在如发起类似auto-refresh命令时，同时发起ctrlupd\_req请求来更新DLL，适应温度、电压的漂移。

## PSRAM\_MC Register

### MEM\_CMD

Offset: 000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| read\_reg\_cmd | 31:24 | RW | This INST will be used when reading data from DDR PSRAM chip’s Mode register | 8'h40 |
| write\_reg\_cmd | 23:16 | RW | This INST will be used when reading data from DDR PSRAM chip’s Mode register | 8'hc0 |
| read\_mem\_cmd | 15:8 | RW | Memory read INST 0x00: sync read mode 0x20: linear read mode | 8'h20 |
| write\_mem\_cmd | 7:0 | RW | Memory Write INST 0x80: sync write mode 0xa0: linear write mode | 8'ha0 |

### DDR\_TIM\_CONFIG

Offset: 004

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| tcem\_cfg | 31:20 | RW | cycle accurate Tcem config for read only | 12'h50 |
| tchd\_cfg | 19:16 | RW | obsolete | 4'h0 |
| tcsp\_cfg | 15:12 | RW | ce# setup time to clk rising edge | 4'h0 |
| Reserved | 11:8 | RW | not used | 4'h5 |
| tcph\_cfg | 7:0 | RW | CE# HIGH between subsequent burst operations | 8'h4 |

### CLK\_CS\_OE\_CTRL

Offset: 008

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:18 | RO | not used | 14'h0 |
| dqs\_local\_inv | 17:17 | RW | local dqs invertion 1:inverter of psram clk 0: psram clk | 1'h0 |
| dqs\_local | 16:16 | RW | dqs seletion 1: use local generated dqs (the ddr clk or inverted ddr clk) 0: use outer chip’s dqs The dqs signal is used to capture the input data | 1'h0 |
| Reserved | 15:13 | RO | unused | 3'h0 |
| clk\_sel | 12:12 | RW | DDR\_clk selection 1: from pll 0: from hclk Default is from pll. Config the PLL and its divider to get an appropriate clock. | 1'h1 |
| clk\_div | 9:8 | RW | Obsolete | 2'h0 |
| reg\_clk\_en | 4:4 | RW | ddr clk on/off control 0： ddr clk forced off 1：ddr clk on/off control by psram wr/rd access | 1'h1 |
| dm\_en | 3:3 | RW | DM (data mask) output enable. High active | 1'h0 |
| cs\_neg | 2:2 | RW | 1: cs output aligned with negedge clock 0:cs output aligned with poedge clock | 1'h0 |
| cs\_en | 1:1 | RW | CS output enable | 1'h1 |
| clk\_en | 0:0 | RW | clock output enable | 1'h1 |

### DDR\_WAIT

Offset: 00C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| mem\_r\_wait | 31:24 | RW | memory read wait cycles | 8'h6 |
| mem\_w\_wait | 23:16 | RW | Memory write wait cycles The wait cycles equal to mem\_w\_wait +1. For APM protocal PSRAM @200MHz, set this bits to be 1, so that write latency is 2 . | 8'h0 |
| mode\_r\_wait | 15:8 | RW | mode register read wait cycles | 8'h6 |
| Reserved | 7:0 | RO | unused | 8'h0 |

### DDR\_MASTER\_CTRL

Offset: 010

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:17 | RO | unused | 15'h0 |
| dll\_en | 16:16 | RW | DLL enable, high active | 1'h0 |
| Reserved | 15:15 | RO | not used | 1'h0 |
| dll\_bypass | 14:14 | RW | Controls the bypass mode of the master DLL. If this bit is high, rdlvl\_delay value would directly mean the number of delay cells used for the DQS path and wrlvl\_delay value would directly mean the number of delay cells used for the memory clock | 1'h0 |
| phase\_detect\_sel | 13:11 | RW | Phase detect selector to choose the number of delay cells to be inserted for the delay line in the edge detection circuit of phase detector. | 3'h0 |
| dll\_lock\_num | 10:8 | RW | Waiting cycles before asserting lock done | 3'h0 |
| dll\_start\_point | 7:0 | RW | Initial delay value for the DLL master delay line. | 8'h4 |

### DLL\_DELAY

Offset: 014

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:17 | RO | unused | 15'h0 |
| tchd\_en\_for\_wr | 16:16 | RW | tchd wait cycles enable for psram write access | 1'h0 |
| rdlvl\_delay | 15:8 | RW | Delays of the read DQS | 8'h40 |
| wrlvl\_delay | 7:0 | RW | Delays of the DDR write clock | 8'h40 |

### DLL\_OBS\_REG0

Offset: 018

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| lock\_inc | 31:24 | RO | inc (master delay line should have more delay cells) value in the past 8 cycles | 8'h0 |
| lock\_dec | 23:16 | RO | dec (master delay line should have less delay cells) value in the past 8 cycles | 8'h0 |
| dll\_lock\_value | 15:8 | RO | DLL lock value | 8'h0 |
| dll\_unlock\_cnt | 7:3 | RO | counter clock gate enable | 5'h0 |
| half\_clock\_mode | 2:2 | RO | The delay cells are only enough to lock half cycle. | 1'h0 |
| force\_lock\_overflow | 1:1 | RO | All delay cells have been used but still fails to lock DLL | 1'h0 |
| dll\_lock | 0:0 | RO | master DLL has locked | 1'h0 |

### DLL\_OBS\_REG1

Offset: 01C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RO | unused | 16'h0 |
| decoder\_out\_wr | 15:8 | RO | Number of delay cells used to generate the delay for memory clock | 8'h0 |
| decoder\_out\_rd | 7:0 | RO | Number of delay cells used to generate the delay for DQS | 8'h0 |

### LOCK\_DONE

Offset: 020

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO | unused | 31'h0 |
| lock\_done | 0:0 | RO | High active signal to mean that DLL has locked | 1'h0 |

### DLL\_RSTN

Offset: 024

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO | unused | 31'h0 |
| dll\_rst\_n | 0:0 | WO | 0: reset DLL 1: ignore Step1:write 0 to reset DLL Step2:check if DLL is lock\_done Step3:set rd/wrlvl\_delay Step4:DLL resync | 1'h1 |

### DLL\_RESYNC

Offset: 028

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO | not used | 31'h0 |
| dll\_resync | 0:0 | WO | Write 1: resync high active pulse used to resync the slave delay line, need to be issued by software after new value of rdlvl\_delay/wrlvl\_delay has been updated。 | 1'h0 |

### CHIP\_CONFIG

Offset: 030

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| chip\_init | 31:31 | RW | 0x0: initialization is not stated, AHB read ignored 0x1: write 1 to enable PSRAM memory access. | 1'h0 |
| Reserved | 30:20 | RO | unused | 11'h0 |
| cmd\_extra\_ce | 19:19 | RW | CS polarity during PSRAM mode register access 1: low 0: high during user specify cmd execution | 1'h0 |
| uhs\_extra\_clk | 18:16 | RW | obsolete | 3'h2 |
| Reserved | 15:13 | RO | unused | 3'h0 |
| chip\_uhs\_map | 12:11 | RW | obsolete | 2'h0 |
| chip\_uhs | 10:10 | RW | obsolete | 1'h0 |
| Reserved | 9:9 | RO | unused | 1'h0 |
| chip\_page | 8:8 | RW | obsolete | 1'h0 |
| Reserved | 7:6 | RO | unused | 2'h0 |
| chip\_size | 5:4 | RW | obsolete | 2'h0 |
| Reserved | 3:1 | RO | unused | 3'h0 |
| chip\_num | 0:0 | RW | obsolete | 1'h0 |

### DDR\_PREFT

Offset: 034

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:27 | RO | unused | 5'h0 |
| war\_en | 26:26 | RW | high active. Enable auto-detection of WAR(write after read). It's valid only if rfifo\_clr\_en=0 | 1'h0 |
| rfifo\_clr\_en | 25:25 | RW | high active. Enable clear read fifo for each psram write | 1'h1 |
| preft\_en | 24:24 | RW | high active prefetch enable | 1'h0 |
| Reserved | 23:15 | RO | unused | 9'h0 |
| preft\_pred\_num | 14:8 | RW | The additional number of data being prefetched after a write has been issured. It's used for more accurately detecting the WAR. | 7'h8 |
| Reserved | 7:7 | RO | unused | 1'h0 |
| preft\_cnt | 6:0 | RW | prefetch depth | 7'h40 |

### DDR\_FIFO

Offset: 038

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:4 | RO | unused | 28'h0 |
| fifo\_cnt | 3:0 | RO | the number of commands in cmd FIFO | 4'h0 |

### DDR\_MISC

Offset: 03C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:10 | RO | unused | 22'h0 |
| wbuf\_flush\_dis | 9:9 | RW | 1: disable auto-flush wrtie buffer 0: auto write the data in write buffer to psram when AHB keep in idle or busy state more than 16 HCLK clocks | 1'h0 |
| wbuf\_hit\_dis | 8:8 | RW | 1: write concatenation enable 0: write concatenation disable | 1'h0 |
| Reserved | 7:1 | RO | unused | 7'h0 |
| fifo\_rd | 0:0 | RW | 0: nomal mode; 1: power down mode | 1'h0 |

### PRIO\_CTRL

Offset: 040

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:5 | RO | unused | 27'h0 |
| port\_prio | 4:0 | RW | priority setting of 5 AHB port  1: in group 1 0: in group 0 | 5'h0 |

### DDR\_CMD

Offset: 084

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| special\_cmd | 31:0 | RW | Special command Will be issued in sequence of Byte3->Byte2->Byte1->Byte0 Byte3: bit 31-24, PSRAM command Byte2~Byte0: bit23-0, PSRAM address | 32'h0 |

### DDR\_CMD\_EN

Offset: 088

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO | unused | 31'h0 |
| special\_cmd\_en | 0:0 | RW | Write 1 to issue the special\_cmd Auto return to 0 when the cmd is done | 1'h0 |

### DEBUG

Offset: 0EC

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:3 | RO | unused | 29'h0 |
| debug\_sel | 2:0 | RW | 00: debug disable 01~04: enable debug others: debug disable | 3'h0 |

### DDR\_VERSION

Offset: 0FC

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| version | 31:0 | RO | 0x101: MPW version 0x102: Mass product version | 32'h102 |

### MR0

Offset: 100

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:8 | RO | unused | 24'h0 |
| MR0 | 7:0 | RW | psram mode register 0 | 8'h0 |

### MR1

Offset: 104

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:8 | RO | unused | 24'h0 |
| MR1 | 7:0 | RO | psram mode register 1 64Mbit APM protocol PSRAM: 0xAD | 8'had |

### MR2

Offset: 108

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:8 | RO | unused | 24'h0 |
| MR2 | 7:0 | RO | psram mode register 2 64Mbit APM protocol PSRAM: 0xD1 | 8'hd1 |

### MR3

Offset: 10C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:8 | RO | unused | 24'h0 |
| MR3 | 7:0 | RO | psram mode register 3 64Mbit APM protocol PSRAM: 0x20 (fast refresh) | 8'h20 |

### MR4

Offset: 110

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:8 | RO | unused | 24'h0 |
| MR4 | 7:0 | RW | psram mode register 4 | 8'h0 |

### MR6

Offset: 118

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:8 | RO | unused | 24'h0 |
| MR6 | 7:0 | WO | psram mode register 6 | 8'h0 |

### MR8

Offset: 120

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:8 | RO | unused | 24'h0 |
| MR8 | 7:0 | RW | psram mode register 8 | 8'h0 |

# TRNG

## Introduction

TRNG 模块用于产生随机数，分别由两个部分组成：模拟TRNG模块和数字TRNG模块。模拟TRNG是一个1bit的随机产生器（主要电路为一个振荡周期易受温度、电压以及随机噪声等影响的振荡环）。数字TRNG连续采样 模拟TRNG的 输出，采样完成后，数字TRNG模块自动Power Down 模拟TRNG模块 ，并产生中断，上报软件。

## Main Features

* APB Bus总线的分频时钟作为数字TRNG的采样时钟
* 模拟TRNG 输出的 单比特随机数采样频率不大于200kbps。可由用户通过寄存器配置为 2^9~2^16（指数为9~16的整数，共8个值可选）；
* 支持数字 TRNG模块在冷热delay时间后，开始采样；
* 支持单次采样时间间隔可配置；
* 支持通过AHB Bus读取 数字TRNG 采集的32bit随机数；
* 支持自动低功耗模式。AHB总线读取TRNG 数据后，完成32bit采样后自动Power Down模拟TRNG电路，进入power down模式
* 用户可通过降低采样率提高TRNG熵率（entropy rate）

## Function Description

TRNG的模块主要信号的时序图如下图所示：

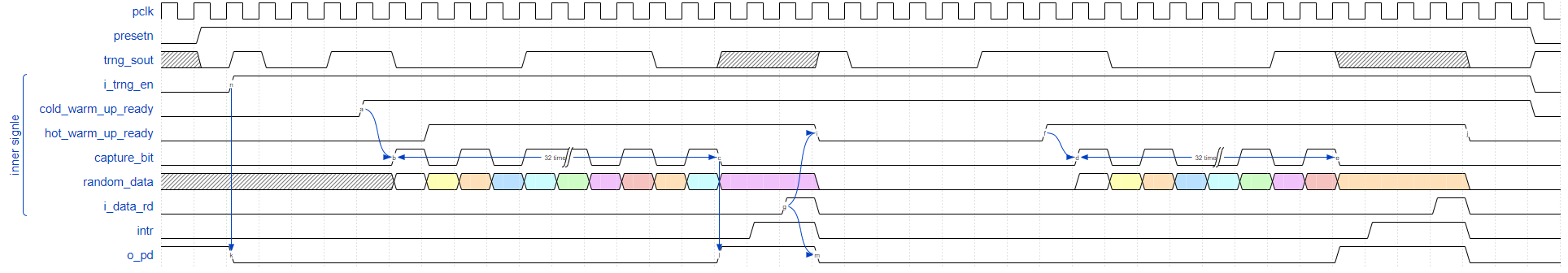


Figure 2 TRNG 时序图

* 通过APB配置i\_trng\_en=1，数字TRNG 模块输出的o\_pd信号拉低，模拟TRNG模块在o\_pd拉低后，开始输出trng\_sout（刚开始模拟输出的trng\_sout信号不稳定，需要延迟采集 ）。
* 在内部计数器计数到CPLD\_TIME时，拉高cold\_warm\_up\_ready信号，此时冷启动完成 。TRNG Core开始采集trng\_sout信号。采样间隔为寄存器配置的DELAY\_TIME。
* 在冷启动完成后，同步开启热启动计数，当内部计数器计数达到 HOT\_TIME时， hot\_warm\_up\_delay信号拉高。
* 在采样数达到32次后，采样结束，此时TRNG Core拉高o\_pd信号，模拟TRNG停止输出trng\_sout信号。
* 通过读取采集到的random\_data数据后，hot\_warm\_up\_ready信号拉低，内部计数器重新开始计数。同时o\_pd信号拉低，模拟TRNG 开始输出trng\_sout信号。
* 在hot\_warm\_up\_ready信号再次拉高后，再次采集32次trng\_sout信号的值 。

## Basic Block Diagram



Figure 1 TRNG Diagram

TRNG 主要 由以下几个主要 模块组成：

* APB Interface：寄存器控制，状态机上报等；
* TRNG CORE：TRNG 控制逻辑，采样逻辑；
* Analog TRNG：TRNG 模拟模块；

## TRNG Register

### TRNG\_CTRL

Offset: 000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| KEY\_PRO | 31:24 | RW | Write protect KEY = 0xF5 | 8'h0 |
| Reserved | 23:1 | RO |  | 23'h0 |
| TRNG\_EN | 0:0 | RW | 1:start the CRYPTO engine 0:stop the CRYPTO engine | 1'h0 |

### TRNG\_CONFIG

Offset: 004

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| KEY\_PRO | 31:24 | RW | Write protect KEY = 0xF5 | 8'h0 |
| Reserved | 23:17 | RO |  | 7'h0 |
| seed\_ext | 16:16 | RW | ext seed control | 1'h0 |
| Reserved | 15:10 | RO |  | 6'h0 |
| COLD\_TIME | 9:8 | RW | Cold warm up time in apb clock cycles, wait cycles after Power On Reset 0x0:2^22 0x1:2^23 0x2:2^24 0x3:2^25 | 2'h2 |
| Reserved | 7:6 | RO |  | 2'h0 |
| HOT\_TIME | 5:4 | RW | Hot warm up time in apb clock cycles, wait cycles after trng analog part power on 0x0:2^16 0x1:2^17 0x2:2^18 0x3:2^19 | 2'h2 |
| Reserved | 3:3 | RO |  | 1'h0 |
| DELAY\_TIME | 2:0 | RW | Delay time in apb clock cycles, wait cycles before next bit capture, capture 32 bits total 0x0:2^9 0x1:2^10 0x2:2^11 0x3:2^12 0x4:2^13 0x5:2^14 0x6:2^15 0x7:2^16 | 3'h4 |

### TRNG\_STATUS

Offset: 008

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO |  | 31'h0 |
| DREADY | 0:0 | RO | 1: Random data is ready for read 0: not ready Return to 0 when the TRNG\_DATA is read | 1'h0 |

### TRNG\_INT\_EN

Offset: 010

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO |  | 31'h0 |
| DREADY | 0:0 | RW | 1: enable DREADY interrupt. 0: ignore | 1'h0 |

### TRNG\_INT\_DIS

Offset: 014

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO |  | 31'h0 |
| DREADY | 0:0 | RW | 1: disable DREADY interrupt. 0: ignore | 1'h0 |

### TRNG\_INT\_MASK

Offset: 018

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO |  | 31'h0 |
| DREADY | 0:0 | RW | 1: DREADY interrupt enabled 0: DREADY interrupt disabled | 1'h0 |

### TRNG\_DATA

Offset: 020

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| TRNG\_DATA | 31:0 | RO | The True Random data, read only | 32'h0 |

### TRNG\_TEST

Offset: 030

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:12 | RO |  | 20'h0 |
| TEST\_CFG | 11:8 | RW | Test Mode：control config | 4'h0 |
| Reserved | 7:2 | RO |  | 6'h0 |
| TEST\_PD | 1:1 | RW | Test Mode：Power Down Control | 1'h0 |
| TEST\_CTRL | 0:0 | RW | Test Mode：1 - enable；0-disable | 1'h0 |

### TRNG\_VERSION

Offset: 078

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| VERSION\_MINOR | 31:16 | RO | The current TRNG MINOR version ECO version | 16'h0 |
| VERSION\_MAJOR | 15:0 | RO | The current TRNGO MAJOR version 0x200:Venus | 16'h101 |

### TRNG\_FEATURE

Offset: 07C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:0 | RO | Which feature parameter is enabled or disabled | 32'h0 |

# KEYSENSE

## Introduction

The keysense generates the key sense wakeup signal and ADC hardware trigger signal for key measure.

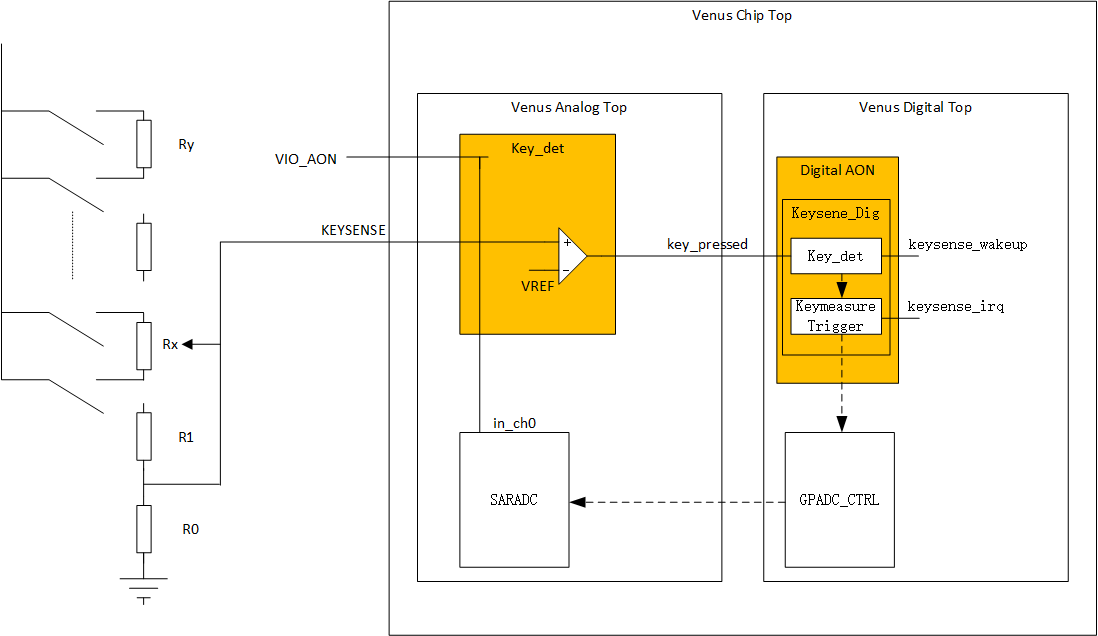


Fig. 1 System application block diagram

The features:

1. Selectable KHz counting clk sources: 32KHz RC osc, 32KHz xtal, a KHz clock divided from 24MHz xtal, etc.
2. Configurable counting threshold for both wakeup and key-measure.
3. Supports key-sense : wakeup and irq.
4. Supports key-measure: hardware SAR ADC triggering control.
5. Supports key-press and key-release irq.

## Main Features

## Function Description

## Basic Block Diagram

Below in Fig 2 is the structure of this module.



Fig. 2 Module Detail Structure

## Programming Guide

1. Set proper value for the wakeup threshold and ADC trigger threshold.
2. Enable the interrupt source needed.
3. Enable keysense.
4. In sleep mode, waking for the key pressing operation and wake up the chip when the key pressing time exceed the wake up threshold. (The wake up signal should be ignored by the AON FSM, when the chip is not in sleep mode. There’s no method to disable the wake up signal in Keysense IP itself. The disable control should be done at the top level.)
5. A wake up/adc\_trigger/keypress/keyrelease irq will be generated if the corresponding interrupt is enabled. AP can respond to the interrupt when it is in active or idle mode.
6. Both in sleep mode or normal working mode, a sensed key can trigger a key-measure operation. Refer to ADC controller spec for more details.

## KEYSENSE Register

### KS\_CFG

Offset: 000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:3 | RO | Not used | 29'h0 |
| ks\_dbg\_en | 2:2 | RW | keysense debug output enable, active high | 1'h0 |
| ks\_dbg\_sel | 1:1 | RW | keysense debug output select: dbg\_datin0 = {ks\_cnt[6:0], ks\_adc\_trig\_intr\_trg, ks\_adc\_trig, ks\_wakeup\_intr\_trg, ks\_wakeup, keyin\_detect\_sync, keyin\_detect, cfg\_ks\_en, ks\_rstn, ks\_clk}; dbg\_datin1 = {ks\_cnt[15:5], ks\_adc\_trig, ks\_wakeup, keyin\_detect, cfg\_ks\_en, ks\_clk}; | 1'h0 |
| ks\_en | 0:0 | RW | keysense enable, active high | 1'h0 |

### KS\_STAT

Offset: 004

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ks\_cnt | 31:16 | ROI | ks counter output, only used for debug purpose | 16'h0 |
| Reserved | 15:6 | RO | Not used | 10'h0 |
| ks\_adc\_trig\_intr\_trg | 5:5 | ROI | keysense adc trigger interrupt source, only used for debug purpose | 1'h0 |
| ks\_adc\_trig | 4:4 | ROI | keysense trigger to GPADC, only used for debug purpose | 1'h0 |
| ks\_wakeup\_intr\_trg | 3:3 | ROI | keysense wakeup interrupt source, only used for debug purpose | 1'h0 |
| ks\_wakeup | 2:2 | ROI | keysense wakeup, only used for debug purpose | 1'h0 |
| keyin\_detect\_sync | 1:1 | ROI | keyin detect signal synced to ks\_clk domain, only used for debug purpose | 1'h0 |
| keyin\_detect | 0:0 | ROI | keyin detect signal from analog SARADC, only used for debug purpose | 1'h0 |

### KS\_THD

Offset: 008

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| ks\_thd\_adc\_trig | 31:16 | RW | adc trigger counter threshold, in the unit of ks\_clk(32KHz clock) cycles | 16'h40 |
| ks\_thd\_wakeup | 15:0 | RW | wakeup counter threshold, in the unit of ks\_clk(32KHz clock) cycles | 16'h20 |

### KS\_IMR

Offset: 00C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:4 | RO |  | 28'h0 |
| ks\_press\_imr | 3:3 | RW | 1 : disable interrrupt source; 0: enable interrupt source | 1'h1 |
| ks\_release\_imr | 2:2 | RW | 1 : disable interrrupt source; 0: enable interrupt source | 1'h1 |
| ks\_adc\_trig\_imr | 1:1 | RW | 1 : disable interrrupt source; 0: enable interrupt source | 1'h1 |
| ks\_wakeup\_imr | 0:0 | RW | 1 : disable interrrupt source; 0: enable interrupt source | 1'h1 |

### KS\_ICR

Offset: 010

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:4 | RO |  | 28'h0 |
| ks\_press\_icr | 3:3 | W1C | write 1 to generate a pulse to clear interrupt status | 1'h0 |
| ks\_release\_icr | 2:2 | W1C | write 1 to generate a pulse to clear interrupt status | 1'h0 |
| ks\_adc\_trig\_icr | 1:1 | W1C | write 1 to generate a pulse to clear interrupt status | 1'h0 |
| ks\_wakeup\_icr | 0:0 | W1C | write 1 to generate a pulse to clear interrupt status | 1'h0 |

### KS\_IRSR

Offset: 014

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:4 | RO |  | 28'h0 |
| ks\_press\_irsr | 3:3 | ROI | raw status of interrupt source | 1'h0 |
| ks\_release\_irsr | 2:2 | ROI | raw status of interrupt source | 1'h0 |
| ks\_adc\_trig\_irsr | 1:1 | ROI | raw status of interrupt source | 1'h0 |
| ks\_wakeup\_irsr | 0:0 | ROI | raw status of interrupt source | 1'h0 |

### KS\_ISR

Offset: 018

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:4 | RO |  | 28'h0 |
| ks\_press\_isr | 3:3 | ROI | status of interrupt source after mask | 1'h0 |
| ks\_release\_isr | 2:2 | ROI | status of interrupt source after mask | 1'h0 |
| ks\_adc\_trig\_isr | 1:1 | ROI | status of interrupt source after mask | 1'h0 |
| ks\_wakeup\_isr | 0:0 | ROI | status of interrupt source after mask | 1'h0 |

# GPADC

## Introduction

* 12-bit resolution, up to 8 channels, up to 1Msps, 24MHz ADC clock
* Configurable hardware ADC trigger sources
* User configurable n-times ADC sampling
* Dedicated ADC Data FIFO for each ADC channel
* Configurable ADC sampling duration
* Configurable waiting time for next Round A/D conversion
* ADC configuration and test logic
* switch on/off control
* ADC trimming
* ADC channel selection
* External/internal VREF selection

## Main Features

## Function Description

* 1. 14-bit to 12-bit ADC data conversion

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| bit num | weight | part2 | part1 | part0 |
| 13 | 1536 | 1024 | 512 | 0 |
| 12 | 1152 | 1024 | 128 | 0 |
| 11 | 640 | 512 | 128 | 0 |
| 10 | 352 | 256 | 64 | 32 |
| 9 | 192 | 128 | 64 | 0 |
| 8 | 104 | 64 | 32 | 8 |
| 7 | 56 | 32 | 16 | 8 |
| 6 | 28 | 16 | 8 | 4 |
| 5 | 16 | 16 | 0 | 0 |
| 4 | 8 | 8 | 0 | 0 |
| 3 | 6 | 4 | 2 | 0 |
| 2 | 3 | 2 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 |
|  | weight sum=4095 | |  |  |

The weight sum of the 14-bit ADC data is 4095, which means it can be exactly represented by a 12-bit binary data.

The bit weight of each bit of the 14-bit ADC data can be further divided into 3 parts as listed in part2/1/0 columns in the above table, so that each part is a 2n number to simplify the calculation.

## Basic Block Diagram

Below in Fig 1 is the structure of this module.



Fig. 1 Detail Structure



## Programming Guide

* 1. ADC application example

1. ADC setup preparation

- Set ADC setup wait time ( ADC\_CR.adc\_setup\_wait register)

- Turn on or off the VREF/VIN buffers

- Select the VREF source; Enable the used ADC channels

- Set an appropriate waiting time before each new round ADC conversion.

- Set the run-round number

- Set FIFO threshold for each ADC channel

2. Enable ADC and wait for the ADC to be setup

- If adc\_complete is HIGH, clear this bit by write 1 to it. Otherwise ADC conversion won’t start even adc\_en=1.

3. Issue a software trigger or waiting for a hardware trigger from keysense

4. Read ADC FIFO data if FIFO threshold reaches

5. Disable ADC if all the n-round conversion is done

## GPADC Register

### ADC\_CR

Offset: 000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| adc\_rwait\_high | 31:24 | RW | ADC periodical sampling wait time (high) The total wait time: (adc\_rwait\_high +1)\* adc\_rwait\_low \* (SAMPLE time + conversion time). e.g., For 24MHz ADC clock, the maximum wait time is 256\*0.255 = 65.28 ms, and the default waiting time is 78\*0.255 = 20ms | 8'h4e |
| adc\_rwait\_low | 23:16 | RW | ADC periodical sampling wait time (lower) 0: no wait time  1: 1 \* (SAMPLE time + conversion time) 2: 2 \* (SAMPLE time + conversion time) … 255: 255 \* (SAMPLE time + conversion time) Note: for 24MHz adc clock, the default lower waiting time is 0.255ms | 8'hff |
| adc\_setup\_wait | 15:8 | RW | Set waiting time after setting adc\_en=1 for ADC to be setup. After waiting, adc\_ready flag in adc\_sr will be set HIGH. 0: no wait time  1~255: (2\*\*(n-1)) \* (SAMPLE time + conversion time) | 8'h1 |
| Reserved | 7:7 | RO | not used | 1'h0 |
| round\_wait\_unit | 6:6 | RW | Select N Round wait time unit 0:(sample+conversion) time as Nround wait timer count unit 1: clock cycle as Nround wait timer count unit | 1'h0 |
| gpt\_trig\_en | 5:5 | RW | gpt trigger enable 0:gpt trigger disable 1:gpt trigger enable | 1'h0 |
| hard\_trig\_en | 4:4 | RW | hardware trigger enable 0：disable hardware trigger from keysense 1: enable hardware trigger from keysense | 1'h1 |
| soft\_trig | 3:3 | W1P | software trigger write 1 create a soft trigger pluse write 0 no effect | 1'h0 |
| Reserved | 2:2 | RO | not used | 1'h0 |
| force\_fifo\_wclk\_on | 1:1 | RW | Reserved | 1'h0 |
| adc\_en | 0:0 | RW | 1: adc enable 0: adc\_disable Note：ADC will be automatically enabled if hard\_trig\_en=1 and keysense is detected | 1'h0 |

### ADC\_SR

Offset: 004

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:2 | RO | not used | 30'h0 |
| hard\_trig | 1:1 | W1C | A hardware trigger detected. | 1'h0 |
| adc\_busy | 0:0 | W1C | ADC conversion is ongoing. | 1'h0 |

### ADC\_CONFIG

Offset: 008

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | not used | 8'h0 |
| sample\_t\_set | 23:22 | RW | set ADC sample time 00: 4 adc clocks 01: 8 adc clocks 10: 16 adc clocks 11: 32 adc clocks | 2'h1 |
| adc\_clk\_set | 21:19 | RW | Set ADC clock frequency 000: 24MHz 001: 12MHz 010: 6MHz 011: 3MHz 100:1.5MHz 101:750KHz 110:375KHz 111:187.5kHz | 3'h0 |
| adc\_trig\_num | 18:8 | RW | ADC periodical sampling  0: infinite trigger mode,ADC will infinitely trigger the selected channel. 1~1024: multi-trigger. After n-round ADC sampling done, stops the ADC | 11'h1 |
| Reserved | 7:7 | RO | not used | 1'h0 |
| differ\_vinn\_sel | 6:6 | RW | Select analog VINN input channel: 0: VINN=SAR\_VIN1 1: VINN=SAR\_VIN3 | 1'h0 |
| single | 5:5 | RW | Single-end or differential analog inputs 0:differential 1:single end | 1'h1 |
| vref\_buf\_en | 4:4 | RW | Enable VREF buffer for improving VREF driving strength Notes: enable vref\_buf\_en only if vref\_sel=2’b00 or 2’b01 | 1'h1 |
| vref\_sel | 3:2 | RW | VREF selection 00,11: 1.25V internal VREF with optional buffer 01: 1/3 VDDIO (2.7~3.3V) with optional amplifier 10: an external VREF without amplifier (e.g., an external high-resolution voltage which should have large enough driving capability) | 2'h1 |
| en\_vin\_buf\_amp | 1:1 | RW | Enable the shared Vin amplifier (a buffer for enhancing the driving capability) for the 8 ADC channel inputs 0: disable 1: enable | 1'h1 |
| bypass\_vin\_buf | 0:0 | RW | Bypass the shared Vin amplifier for the 8 ADC channel inputs | 1'h0 |

### ADC\_CH\_SEL

Offset: 00C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| dma\_ch\_en | 31:16 | RW | DMA Channel Enable.One bit for each of the 13 channels,and supports more than one channel to be selected. Bit16~29:Selete ADC Channel 0-13 for DMA bit30~31:Reserved | 16'h0 |
| adc\_ch\_sel | 15:0 | RW | Select ADC input channels. One bit for each of the 8 channels, and supports more than one channel to be selected.  0: de-select; 1, select Note: Bit3~0: ADC channel 3 ~ 0 Bit4: 1/6 VDD\_IO  Bit5: 1/8 VCC Bit6: keysense Bit7: Reserved Bit8~13:CVD channel 0~5 Bit14~15:Reserved | 16'h40 |

### ADC\_IMR0

Offset: 010

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:30 | RO | not used | 2'h0 |
| fifo\_empty\_imr\_ch13 | 29:29 | RW | Mask FIFO empty interrupt of ADC channel 13. High active. | 1'h1 |
| fifo\_empty\_imr\_ch12 | 28:28 | RW | Mask FIFO empty interrupt of ADC channel 12. High active. | 1'h1 |
| fifo\_empty\_imr\_ch11 | 27:27 | RW | Mask FIFO empty interrupt of ADC channel 11. High active. | 1'h1 |
| fifo\_empty\_imr\_ch10 | 26:26 | RW | Mask FIFO empty interrupt of ADC channel 10. High active. | 1'h1 |
| fifo\_empty\_imr\_ch9 | 25:25 | RW | Mask FIFO empty interrupt of ADC channel 9. High active. | 1'h1 |
| fifo\_empty\_imr\_ch8 | 24:24 | RW | Mask FIFO empty interrupt of ADC channel 8. High active. | 1'h1 |
| fifo\_empty\_imr\_ch7 | 23:23 | RW | Mask FIFO empty interrupt of ADC channel 7. High active. | 1'h1 |
| fifo\_empty\_imr\_ch6 | 22:22 | RW | Mask FIFO empty interrupt of ADC channel 6. High active. | 1'h1 |
| fifo\_empty\_imr\_ch5 | 21:21 | RW | Mask FIFO empty interrupt of ADC channel 5. High active. | 1'h1 |
| fifo\_empty\_imr\_ch4 | 20:20 | RW | Mask FIFO empty interrupt of ADC channel 4. High active. | 1'h1 |
| fifo\_empty\_imr\_ch3 | 19:19 | RW | Mask FIFO empty interrupt of ADC channel 3. High active. | 1'h1 |
| fifo\_empty\_imr\_ch2 | 18:18 | RW | Mask FIFO empty interrupt of ADC channel 2. High active. | 1'h1 |
| fifo\_empty\_imr\_ch1 | 17:17 | RW | Mask FIFO empty interrupt of ADC channel 1. High active. | 1'h1 |
| fifo\_empty\_imr\_ch0 | 16:16 | RW | Mask FIFO empty interrupt of ADC channel 0. High active. | 1'h1 |
| Reserved | 15:4 | RO | not used | 12'h0 |
| adc\_complete\_imr | 3:3 | RW | Mask ADC complete interrupt | 1'h0 |
| eoc\_err\_imr | 2:2 | RW | Mask no EOC generated error interrupt | 1'h0 |
| adc\_ready\_imr | 1:1 | RW | Mask adc ready interrupt | 1'h0 |
| channel\_err\_imr | 0:0 | RW | Mask channel select error interrupt | 1'h0 |

### ADC\_IMR1

Offset: 014

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:30 | RO | not used | 2'h1 |
| fifo\_thd\_imr\_ch13 | 29:29 | RW | Mask FIFO threshold reach interrupt of ADC channel13.High active. | 1'h1 |
| fifo\_thd\_imr\_ch12 | 28:28 | RW | Mask FIFO threshold reach interrupt of ADC channel12.High active. | 1'h1 |
| fifo\_thd\_imr\_ch11 | 27:27 | RW | Mask FIFO threshold reach interrupt of ADC channel11.High active. | 1'h1 |
| fifo\_thd\_imr\_ch10 | 26:26 | RW | Mask FIFO threshold reach interrupt of ADC channel10.High active. | 1'h1 |
| fifo\_thd\_imr\_ch9 | 25:25 | RW | Mask FIFO threshold reach interrupt of ADC channel9.High active. | 1'h1 |
| fifo\_thd\_imr\_ch8 | 24:24 | RW | Mask FIFO threshold reach interrupt of ADC channel8.High active. | 1'h1 |
| fifo\_thd\_imr\_ch7 | 23:23 | RW | Mask FIFO threshold reach interrupt of ADC channel7.High active. | 1'h1 |
| fifo\_thd\_imr\_ch6 | 22:22 | RW | Mask FIFO threshold reach interrupt of ADC channel6.High active. | 1'h1 |
| fifo\_thd\_imr\_ch5 | 21:21 | RW | Mask FIFO threshold reach interrupt of ADC channel5.High active. | 1'h1 |
| fifo\_thd\_imr\_ch4 | 20:20 | RW | Mask FIFO threshold reach interrupt of ADC channel4.High active. | 1'h1 |
| fifo\_thd\_imr\_ch3 | 19:19 | RW | Mask FIFO threshold reach interrupt of ADC channel3.High active. | 1'h1 |
| fifo\_thd\_imr\_ch2 | 18:18 | RW | Mask FIFO threshold reach interrupt of ADC channel2.High active. | 1'h1 |
| fifo\_thd\_imr\_ch1 | 17:17 | RW | Mask FIFO threshold reach interrupt of ADC channel1.High active. | 1'h1 |
| fifo\_thd\_imr\_ch0 | 16:16 | RW | Mask FIFO threshold reach interrupt of ADC channel0.High active. | 1'h1 |
| Reserved | 15:14 | RO | not used | 2'h0 |
| fifo\_full\_imr\_ch13 | 13:13 | RW | Mask FIFO full interrupt of ADC channel 13. High active. | 1'h0 |
| fifo\_full\_imr\_ch12 | 12:12 | RW | Mask FIFO full interrupt of ADC channel 12. High active. | 1'h0 |
| fifo\_full\_imr\_ch11 | 11:11 | RW | Mask FIFO full interrupt of ADC channel 11. High active. | 1'h0 |
| fifo\_full\_imr\_ch10 | 10:10 | RW | Mask FIFO full interrupt of ADC channel 10. High active. | 1'h0 |
| fifo\_full\_imr\_ch9 | 9:9 | RW | Mask FIFO full interrupt of ADC channel 9. High active. | 1'h0 |
| fifo\_full\_imr\_ch8 | 8:8 | RW | Mask FIFO full interrupt of ADC channel 8. High active. | 1'h0 |
| fifo\_full\_imr\_ch7 | 7:7 | RW | Mask FIFO full interrupt of ADC channel 7. High active. | 1'h0 |
| fifo\_full\_imr\_ch6 | 6:6 | RW | Mask FIFO full interrupt of ADC channel 6. High active. | 1'h0 |
| fifo\_full\_imr\_ch5 | 5:5 | RW | Mask FIFO full interrupt of ADC channel 5. High active. | 1'h0 |
| fifo\_full\_imr\_ch4 | 4:4 | RW | Mask FIFO full interrupt of ADC channel 4. High active. | 1'h0 |
| fifo\_full\_imr\_ch3 | 3:3 | RW | Mask FIFO full interrupt of ADC channel 3. High active. | 1'h0 |
| fifo\_full\_imr\_ch2 | 2:2 | RW | Mask FIFO full interrupt of ADC channel 2. High active. | 1'h0 |
| fifo\_full\_imr\_ch1 | 1:1 | RW | Mask FIFO full interrupt of ADC channel 1. High active. | 1'h0 |
| fifo\_full\_imr\_ch0 | 0:0 | RW | Mask FIFO full interrupt of ADC channel 0. High active. | 1'h0 |

### ADC\_ICR0

Offset: 018

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:30 | RO | not used | 2'h0 |
| fifo\_empty\_icr\_ch13 | 29:29 | W1C | Clear FIFO empty status of ADC channel 13. High active. | 1'h0 |
| fifo\_empty\_icr\_ch12 | 28:28 | W1C | Clear FIFO empty status of ADC channel 12. High active. | 1'h0 |
| fifo\_empty\_icr\_ch11 | 27:27 | W1C | Clear FIFO empty status of ADC channel 11. High active. | 1'h0 |
| fifo\_empty\_icr\_ch10 | 26:26 | W1C | Clear FIFO empty status of ADC channel 10. High active. | 1'h0 |
| fifo\_empty\_icr\_ch9 | 25:25 | W1C | Clear FIFO empty status of ADC channel 9. High active. | 1'h0 |
| fifo\_empty\_icr\_ch8 | 24:24 | W1C | Clear FIFO empty status of ADC channel 8. High active. | 1'h0 |
| fifo\_empty\_icr\_ch7 | 23:23 | W1C | Clear FIFO empty status of ADC channel 7. High active. | 1'h0 |
| fifo\_empty\_icr\_ch6 | 22:22 | W1C | Clear FIFO empty status of ADC channel 6. High active. | 1'h0 |
| fifo\_empty\_icr\_ch5 | 21:21 | W1C | Clear FIFO empty status of ADC channel 5. High active. | 1'h0 |
| fifo\_empty\_icr\_ch4 | 20:20 | W1C | Clear FIFO empty status of ADC channel 4. High active. | 1'h0 |
| fifo\_empty\_icr\_ch3 | 19:19 | W1C | Clear FIFO empty status of ADC channel 3. High active. | 1'h0 |
| fifo\_empty\_icr\_ch2 | 18:18 | W1C | Clear FIFO empty status of ADC channel 2. High active. | 1'h0 |
| fifo\_empty\_icr\_ch1 | 17:17 | W1C | Clear FIFO empty status of ADC channel 1. High active. | 1'h0 |
| fifo\_empty\_icr\_ch0 | 16:16 | W1C | Clear FIFO empty status of ADC channel 0. High active. | 1'h0 |
| Reserved | 15:4 | RO | not used | 12'h0 |
| adc\_complete\_icr | 3:3 | W1C | Clear ADC complete status | 1'h0 |
| eoc\_err\_icr | 2:2 | W1C | Clear no EOC generated error status | 1'h0 |
| adc\_ready\_icr | 1:1 | W1C | Clear adc ready status | 1'h0 |
| channel\_err\_icr | 0:0 | W1C | Clear channel select error status | 1'h0 |

### ADC\_ICR1

Offset: 01C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:30 | RO | not used | 2'h0 |
| fifo\_thd\_icr\_ch13 | 29:29 | W1C | Clear FIFO threshold reach status of ADC channnel 13. High active | 1'h0 |
| fifo\_thd\_icr\_ch12 | 28:28 | W1C | Clear FIFO threshold reach status of ADC channnel 12. High active | 1'h0 |
| fifo\_thd\_icr\_ch11 | 27:27 | W1C | Clear FIFO threshold reach status of ADC channnel 11. High active | 1'h0 |
| fifo\_thd\_icr\_ch10 | 26:26 | W1C | Clear FIFO threshold reach status of ADC channnel 10. High active | 1'h0 |
| fifo\_thd\_icr\_ch9 | 25:25 | W1C | Clear FIFO threshold reach status of ADC channnel 9. High active | 1'h0 |
| fifo\_thd\_icr\_ch8 | 24:24 | W1C | Clear FIFO threshold reach status of ADC channnel 8. High active | 1'h0 |
| fifo\_thd\_icr\_ch7 | 23:23 | W1C | Clear FIFO threshold reach status of ADC channnel 7. High active | 1'h0 |
| fifo\_thd\_icr\_ch6 | 22:22 | W1C | Clear FIFO threshold reach status of ADC channnel 6. High active | 1'h0 |
| fifo\_thd\_icr\_ch5 | 21:21 | W1C | Clear FIFO threshold reach status of ADC channnel 5. High active | 1'h0 |
| fifo\_thd\_icr\_ch4 | 20:20 | W1C | Clear FIFO threshold reach status of ADC channnel 4. High active | 1'h0 |
| fifo\_thd\_icr\_ch3 | 19:19 | W1C | Clear FIFO threshold reach status of ADC channnel 3. High active | 1'h0 |
| fifo\_thd\_icr\_ch2 | 18:18 | W1C | Clear FIFO threshold reach status of ADC channnel 2. High active | 1'h0 |
| fifo\_thd\_icr\_ch1 | 17:17 | W1C | Clear FIFO threshold reach status of ADC channnel 1. High active | 1'h0 |
| fifo\_thd\_icr\_ch0 | 16:16 | W1C | Clear FIFO threshold reach status of ADC channnel 0. High active | 1'h0 |
| Reserved | 15:14 | RO | not used | 2'h0 |
| fifo\_full\_icr\_ch13 | 13:13 | W1C | Clear FIFO full status of ADC channel 13. High active. | 1'h0 |
| fifo\_full\_icr\_ch12 | 12:12 | W1C | Clear FIFO full status of ADC channel 12. High active. | 1'h0 |
| fifo\_full\_icr\_ch11 | 11:11 | W1C | Clear FIFO full status of ADC channel 11. High active. | 1'h0 |
| fifo\_full\_icr\_ch10 | 10:10 | W1C | Clear FIFO full status of ADC channel 10. High active. | 1'h0 |
| fifo\_full\_icr\_ch9 | 9:9 | W1C | Clear FIFO full status of ADC channel 9. High active. | 1'h0 |
| fifo\_full\_icr\_ch8 | 8:8 | W1C | Clear FIFO full status of ADC channel 8. High active. | 1'h0 |
| fifo\_full\_icr\_ch7 | 7:7 | W1C | Clear FIFO full status of ADC channel 7. High active. | 1'h0 |
| fifo\_full\_icr\_ch6 | 6:6 | W1C | Clear FIFO full status of ADC channel 6. High active. | 1'h0 |
| fifo\_full\_icr\_ch5 | 5:5 | W1C | Clear FIFO full status of ADC channel 5. High active. | 1'h0 |
| fifo\_full\_icr\_ch4 | 4:4 | W1C | Clear FIFO full status of ADC channel 4. High active. | 1'h0 |
| fifo\_full\_icr\_ch3 | 3:3 | W1C | Clear FIFO full status of ADC channel 3. High active. | 1'h0 |
| fifo\_full\_icr\_ch2 | 2:2 | W1C | Clear FIFO full status of ADC channel 2. High active. | 1'h0 |
| fifo\_full\_icr\_ch1 | 1:1 | W1C | Clear FIFO full status of ADC channel 1. High active. | 1'h0 |
| fifo\_full\_icr\_ch0 | 0:0 | W1C | Clear FIFO full status of ADC channel 0. High active. | 1'h0 |

### ADC\_IRSR0

Offset: 020

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:30 | RO | not used | 2'h0 |
| fifo\_empty\_irsr\_ch13 | 29:29 | RO | FIFO empty raw status of ADC channel 13. High active. | 1'h0 |
| fifo\_empty\_irsr\_ch12 | 28:28 | RO | FIFO empty raw status of ADC channel 12. High active. | 1'h0 |
| fifo\_empty\_irsr\_ch11 | 27:27 | RO | FIFO empty raw status of ADC channel 11. High active. | 1'h0 |
| fifo\_empty\_irsr\_ch10 | 26:26 | RO | FIFO empty raw status of ADC channel 10. High active. | 1'h0 |
| fifo\_empty\_irsr\_ch9 | 25:25 | RO | FIFO empty raw status of ADC channel 9. High active. | 1'h0 |
| fifo\_empty\_irsr\_ch8 | 24:24 | RO | FIFO empty raw status of ADC channel 8. High active. | 1'h0 |
| fifo\_empty\_irsr\_ch7 | 23:23 | RO | FIFO empty raw status of ADC channel 7. High active. | 1'h0 |
| fifo\_empty\_irsr\_ch6 | 22:22 | RO | FIFO empty raw status of ADC channel 6. High active. | 1'h0 |
| fifo\_empty\_irsr\_ch5 | 21:21 | RO | FIFO empty raw status of ADC channel 5. High active. | 1'h0 |
| fifo\_empty\_irsr\_ch4 | 20:20 | RO | FIFO empty raw status of ADC channel 4. High active. | 1'h0 |
| fifo\_empty\_irsr\_ch3 | 19:19 | RO | FIFO empty raw status of ADC channel 3. High active. | 1'h0 |
| fifo\_empty\_irsr\_ch2 | 18:18 | RO | FIFO empty raw status of ADC channel 2. High active. | 1'h0 |
| fifo\_empty\_irsr\_ch1 | 17:17 | RO | FIFO empty raw status of ADC channel 1. High active. | 1'h0 |
| fifo\_empty\_irsr\_ch0 | 16:16 | RO | FIFO empty raw status of ADC channel 0. High active. | 1'h0 |
| Reserved | 15:4 | RO | not used | 12'h0 |
| adc\_complete\_irsr | 3:3 | RO | N-round ADC conversion complete raw status | 1'h0 |
| eoc\_err\_irsr | 2:2 | RO | no EOC generated error raw status | 1'h0 |
| adc\_ready\_irsr | 1:1 | RO | adc ready raw status | 1'h0 |
| channel\_err\_irsr | 0:0 | RO | channel select error raw status | 1'h0 |

### ADC\_IRSR1

Offset: 024

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:30 | RO | not used | 2'h0 |
| fifo\_thd\_irsr\_ch13 | 29:29 | RO | FIFO threshold reach raw status of ADC channnel 13. High active | 1'h0 |
| fifo\_thd\_irsr\_ch12 | 28:28 | RO | FIFO threshold reach raw status of ADC channnel 12. High active | 1'h0 |
| fifo\_thd\_irsr\_ch11 | 27:27 | RO | FIFO threshold reach raw status of ADC channnel 11. High active | 1'h0 |
| fifo\_thd\_irsr\_ch10 | 26:26 | RO | FIFO threshold reach raw status of ADC channnel 10. High active | 1'h0 |
| fifo\_thd\_irsr\_ch9 | 25:25 | RO | FIFO threshold reach raw status of ADC channnel 9. High active | 1'h0 |
| fifo\_thd\_irsr\_ch8 | 24:24 | RO | FIFO threshold reach raw status of ADC channnel 8. High active | 1'h0 |
| fifo\_thd\_irsr\_ch7 | 23:23 | RO | FIFO threshold reach raw status of ADC channnel 7. High active | 1'h0 |
| fifo\_thd\_irsr\_ch6 | 22:22 | RO | FIFO threshold reach raw status of ADC channnel 6. High active | 1'h0 |
| fifo\_thd\_irsr\_ch5 | 21:21 | RO | FIFO threshold reach raw status of ADC channnel 5. High active | 1'h0 |
| fifo\_thd\_irsr\_ch4 | 20:20 | RO | FIFO threshold reach raw status of ADC channnel 4. High active | 1'h0 |
| fifo\_thd\_irsr\_ch3 | 19:19 | RO | FIFO threshold reach raw status of ADC channnel 3. High active | 1'h0 |
| fifo\_thd\_irsr\_ch2 | 18:18 | RO | FIFO threshold reach raw status of ADC channnel 2. High active | 1'h0 |
| fifo\_thd\_irsr\_ch1 | 17:17 | RO | FIFO threshold reach raw status of ADC channnel 1. High active | 1'h0 |
| fifo\_thd\_irsr\_ch0 | 16:16 | RO | FIFO threshold reach raw status of ADC channnel 0. High active | 1'h0 |
| Reserved | 15:14 | RO | not used | 2'h0 |
| fifo\_full\_irsr\_ch13 | 13:13 | RO | FIFO empty raw status of ADC channel 13. High active. | 1'h0 |
| fifo\_full\_irsr\_ch12 | 12:12 | RO | FIFO empty raw status of ADC channel 12. High active. | 1'h0 |
| fifo\_full\_irsr\_ch11 | 11:11 | RO | FIFO empty raw status of ADC channel 11. High active. | 1'h0 |
| fifo\_full\_irsr\_ch10 | 10:10 | RO | FIFO empty raw status of ADC channel 10. High active. | 1'h0 |
| fifo\_full\_irsr\_ch9 | 9:9 | RO | FIFO empty raw status of ADC channel 9. High active. | 1'h0 |
| fifo\_full\_irsr\_ch8 | 8:8 | RO | FIFO empty raw status of ADC channel 8. High active. | 1'h0 |
| fifo\_full\_irsr\_ch7 | 7:7 | RO | FIFO empty raw status of ADC channel 7. High active. | 1'h0 |
| fifo\_full\_irsr\_ch6 | 6:6 | RO | FIFO empty raw status of ADC channel 6. High active. | 1'h0 |
| fifo\_full\_irsr\_ch5 | 5:5 | RO | FIFO empty raw status of ADC channel 5. High active. | 1'h0 |
| fifo\_full\_irsr\_ch4 | 4:4 | RO | FIFO empty raw status of ADC channel 4. High active. | 1'h0 |
| fifo\_full\_irsr\_ch3 | 3:3 | RO | FIFO empty raw status of ADC channel 3. High active. | 1'h0 |
| fifo\_full\_irsr\_ch2 | 2:2 | RO | FIFO empty raw status of ADC channel 2. High active. | 1'h0 |
| fifo\_full\_irsr\_ch1 | 1:1 | RO | FIFO empty raw status of ADC channel 1. High active. | 1'h0 |
| fifo\_full\_irsr\_ch0 | 0:0 | RO | FIFO empty raw status of ADC channel 0. High active. | 1'h0 |

### ADC\_ISR0

Offset: 028

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:30 | RO | not used | 2'h0 |
| fifo\_empty\_isr\_ch13 | 29:29 | RO | FIFO empty status after mask of ADC channel 13. High active. | 1'h0 |
| fifo\_empty\_isr\_ch12 | 28:28 | RO | FIFO empty status after mask of ADC channel 12. High active. | 1'h0 |
| fifo\_empty\_isr\_ch11 | 27:27 | RO | FIFO empty status after mask of ADC channel 11. High active. | 1'h0 |
| fifo\_empty\_isr\_ch10 | 26:26 | RO | FIFO empty status after mask of ADC channel 10. High active. | 1'h0 |
| fifo\_empty\_isr\_ch9 | 25:25 | RO | FIFO empty status after mask of ADC channel 9. High active. | 1'h0 |
| fifo\_empty\_isr\_ch8 | 24:24 | RO | FIFO empty status after mask of ADC channel 8. High active. | 1'h0 |
| fifo\_empty\_isr\_ch7 | 23:23 | RO | FIFO empty status after mask of ADC channel 7. High active. | 1'h0 |
| fifo\_empty\_isr\_ch6 | 22:22 | RO | FIFO empty status after mask of ADC channel 6. High active. | 1'h0 |
| fifo\_empty\_isr\_ch5 | 21:21 | RO | FIFO empty status after mask of ADC channel 5. High active. | 1'h0 |
| fifo\_empty\_isr\_ch4 | 20:20 | RO | FIFO empty status after mask of ADC channel 4. High active. | 1'h0 |
| fifo\_empty\_isr\_ch3 | 19:19 | RO | FIFO empty status after mask of ADC channel 3. High active. | 1'h0 |
| fifo\_empty\_isr\_ch2 | 18:18 | RO | FIFO empty status after mask of ADC channel 2. High active. | 1'h0 |
| fifo\_empty\_isr\_ch1 | 17:17 | RO | FIFO empty status after mask of ADC channel 1. High active. | 1'h0 |
| fifo\_empty\_isr\_ch0 | 16:16 | RO | FIFO empty status after mask of ADC channel 0. High active. | 1'h0 |
| Reserved | 15:4 | RO | not used | 12'h0 |
| adc\_complete\_isr | 3:3 | RO | N-round ADC conversion complete status after mask | 1'h0 |
| eoc\_err\_isr | 2:2 | RO | no EOC generated error status after mask | 1'h0 |
| adc\_ready\_isr | 1:1 | RO | adc ready status after mask | 1'h0 |
| channel\_err\_isr | 0:0 | RO | channel select error status after mask | 1'h0 |

### ADC\_ISR1

Offset: 02C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:30 | RO | not used | 2'h0 |
| fifo\_thd\_isr\_ch13 | 29:29 | RO | FIFO threshold reach status after mask of ADC channnel 13. High active | 1'h0 |
| fifo\_thd\_isr\_ch12 | 28:28 | RO | FIFO threshold reach status after mask of ADC channnel 12. High active | 1'h0 |
| fifo\_thd\_isr\_ch11 | 27:27 | RO | FIFO threshold reach status after mask of ADC channnel 11. High active | 1'h0 |
| fifo\_thd\_isr\_ch10 | 26:26 | RO | FIFO threshold reach status after mask of ADC channnel 10. High active | 1'h0 |
| fifo\_thd\_isr\_ch9 | 25:25 | RO | FIFO threshold reach status after mask of ADC channnel 9. High active | 1'h0 |
| fifo\_thd\_isr\_ch8 | 24:24 | RO | FIFO threshold reach status after mask of ADC channnel 8. High active | 1'h0 |
| fifo\_thd\_isr\_ch7 | 23:23 | RO | FIFO threshold reach status after mask of ADC channnel 7. High active | 1'h0 |
| fifo\_thd\_isr\_ch6 | 22:22 | RO | FIFO threshold reach status after mask of ADC channnel 6. High active | 1'h0 |
| fifo\_thd\_isr\_ch5 | 21:21 | RO | FIFO threshold reach status after mask of ADC channnel 5. High active | 1'h0 |
| fifo\_thd\_isr\_ch4 | 20:20 | RO | FIFO threshold reach status after mask of ADC channnel 4. High active | 1'h0 |
| fifo\_thd\_isr\_ch3 | 19:19 | RO | FIFO threshold reach status after mask of ADC channnel 3. High active | 1'h0 |
| fifo\_thd\_isr\_ch2 | 18:18 | RO | FIFO threshold reach status after mask of ADC channnel 2. High active | 1'h0 |
| fifo\_thd\_isr\_ch1 | 17:17 | RO | FIFO threshold reach status after mask of ADC channnel 1. High active | 1'h0 |
| fifo\_thd\_isr\_ch0 | 16:16 | RO | FIFO threshold reach status after mask of ADC channnel 0. High active | 1'h0 |
| Reserved | 15:14 | RO | not used | 2'h0 |
| fifo\_full\_isr\_ch13 | 13:13 | RO | FIFO full status after mask of ADC channel 13. High active. | 1'h0 |
| fifo\_full\_isr\_ch12 | 12:12 | RO | FIFO full status after mask of ADC channel 12. High active. | 1'h0 |
| fifo\_full\_isr\_ch11 | 11:11 | RO | FIFO full status after mask of ADC channel 11. High active. | 1'h0 |
| fifo\_full\_isr\_ch10 | 10:10 | RO | FIFO full status after mask of ADC channel 10. High active. | 1'h0 |
| fifo\_full\_isr\_ch9 | 9:9 | RO | FIFO full status after mask of ADC channel 9. High active. | 1'h0 |
| fifo\_full\_isr\_ch8 | 8:8 | RO | FIFO full status after mask of ADC channel 8. High active. | 1'h0 |
| fifo\_full\_isr\_ch7 | 7:7 | RO | FIFO full status after mask of ADC channel 7. High active. | 1'h0 |
| fifo\_full\_isr\_ch6 | 6:6 | RO | FIFO full status after mask of ADC channel 6. High active. | 1'h0 |
| fifo\_full\_isr\_ch5 | 5:5 | RO | FIFO full status after mask of ADC channel 5. High active. | 1'h0 |
| fifo\_full\_isr\_ch4 | 4:4 | RO | FIFO full status after mask of ADC channel 4. High active. | 1'h0 |
| fifo\_full\_isr\_ch3 | 3:3 | RO | FIFO full status after mask of ADC channel 3. High active. | 1'h0 |
| fifo\_full\_isr\_ch2 | 2:2 | RO | FIFO full status after mask of ADC channel 2. High active. | 1'h0 |
| fifo\_full\_isr\_ch1 | 1:1 | RO | FIFO full status after mask of ADC channel 1. High active. | 1'h0 |
| fifo\_full\_isr\_ch0 | 0:0 | RO | FIFO full status after mask of ADC channel 0. High active. | 1'h0 |

### ADC\_FIFO\_THD0

Offset: 030

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| fifo\_thd\_ch7 | 31:28 | RW | FIFO threshold for generating IRQ of ADC channel 7 | 4'h8 |
| fifo\_thd\_ch6 | 27:24 | RW | FIFO threshold for generating IRQ of ADC channel 6 | 4'h8 |
| fifo\_thd\_ch5 | 23:20 | RW | FIFO threshold for generating IRQ of ADC channel 5 | 4'h8 |
| fifo\_thd\_ch4 | 19:16 | RW | FIFO threshold for generating IRQ of ADC channel 4 | 4'h8 |
| fifo\_thd\_ch3 | 15:12 | RW | FIFO threshold for generating IRQ of ADC channel 3 | 4'h8 |
| fifo\_thd\_ch2 | 11:8 | RW | FIFO threshold for generating IRQ of ADC channel 2 | 4'h8 |
| fifo\_thd\_ch1 | 7:4 | RW | FIFO threshold for generating IRQ of ADC channel 1 | 4'h8 |
| fifo\_thd\_ch0 | 3:0 | RW | FIFO threshold for generating IRQ of ADC channel 0 | 4'h8 |

### ADC\_FIFO\_THD1

Offset: 034

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | not used | 8'h0 |
| fifo\_thd\_ch13 | 23:20 | RW | FIFO threshold for generating IRQ of ADC channel 13 | 4'h8 |
| fifo\_thd\_ch12 | 19:16 | RW | FIFO threshold for generating IRQ of ADC channel 12 | 4'h8 |
| fifo\_thd\_Ch11 | 15:12 | RW | FIFO threshold for generating IRQ of ADC channel 11 | 4'h8 |
| fifo\_thd\_ch10 | 11:8 | RW | FIFO threshold for generating IRQ of ADC channel 10 | 4'h8 |
| fifo\_thd\_ch9 | 7:4 | RW | FIFO threshold for generating IRQ of ADC channel 9 | 4'h8 |
| fifo\_thd\_ch8 | 3:0 | RW | FIFO threshold for generating IRQ of ADC channel 8 | 4'h8 |

### ADC\_FIFO\_DATA\_CNT0

Offset: 038

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| rdata\_cnt\_ch7 | 31:28 | RO | number of ADC data in fifo of ADC channel 7 | 4'h0 |
| rdata\_cnt\_ch6 | 27:24 | RO | number of ADC data in fifo of ADC channel 6 | 4'h0 |
| rdata\_cnt\_ch5 | 23:20 | RO | number of ADC data in fifo of ADC channel 5 | 4'h0 |
| rdata\_cnt\_ch4 | 19:16 | RO | number of ADC data in fifo of ADC channel 4 | 4'h0 |
| rdata\_cnt\_ch3 | 15:12 | RO | number of ADC data in fifo of ADC channel 3 | 4'h0 |
| rdata\_cnt\_ch2 | 11:8 | RO | number of ADC data in fifo of ADC channel 2 | 4'h0 |
| rdata\_cnt\_ch1 | 7:4 | RO | number of ADC data in fifo of ADC channel 1 | 4'h0 |
| rdata\_cnt\_ch0 | 3:0 | RO | number of ADC data in fifo of ADC channel 0 | 4'h0 |

### ADC\_FIFO\_DATA\_CNT1

Offset: 03C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | not used | 8'h0 |
| rdata\_cnt\_ch13 | 23:20 | RO | number of ADC data in fifo of ADC channel 13 | 4'h0 |
| rdata\_cnt\_ch12 | 19:16 | RO | number of ADC data in fifo of ADC channel 12 | 4'h0 |
| rdata\_cnt\_ch11 | 15:12 | RO | number of ADC data in fifo of ADC channel 11 | 4'h0 |
| rdata\_cnt\_ch10 | 11:8 | RO | number of ADC data in fifo of ADC channel 10 | 4'h0 |
| rdata\_cnt\_ch9 | 7:4 | RO | number of ADC data in fifo of ADC channel 9 | 4'h0 |
| rdata\_cnt\_ch8 | 3:0 | RO | number of ADC data in fifo of ADC channel 8 | 4'h0 |

### ADC\_OFFSET\_A

Offset: 040

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | not used | 8'h0 |
| cal\_offset\_a1 | 23:16 | RW | calibration offset for VREF=1/2 VDDIO, and VIN amplifier on bit7: sign bit6: 32 LSBs bit5: 16 LSBs bit4: 8 LSBs bit3: 4 LSBs bit2: 2 LSBs bit1: 1 LSB bit0: 0.5 LSB the offset range：+/- 63.5 LSBs | 8'h0 |
| Reserved | 15:8 | RO | not used | 8'h0 |
| cal\_offset\_a0 | 7:0 | RW | calibration offset for VREF=1/2 VDDIO, and VIN amplifier off bit7: sign bit6: 32 LSBs bit5: 16 LSBs bit4: 8 LSBs bit3: 4 LSBs bit2: 2 LSBs bit1: 1 LSB bit0: 0.5 LSB the offset range：+/- 63.5 LSBs | 8'h0 |

### ADC\_GAIN\_A

Offset: 044

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RO | not used | 16'h0 |
| cal\_gain\_a | 15:0 | RW | calibration gain for VREF=1/2 VDDIO (the same gain for VIN amplifier on/off ) bit15: sign bit14-0: lower 15 bits of 19-bit fraction the gain range: +/- 6.25% exampls: for 3.1%, cal\_gain = (0.031 \* (2\*\*19))=0x3f7d | 16'h0 |

### ADC\_OFFSET\_B

Offset: 048

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | not used | 8'h0 |
| cal\_offset\_b1 | 23:16 | RW | calibration offset for VREF=VREF\_V1P25, and VIN amplifier on bit7: sign bit6: 32 LSBs bit5: 16 LSBs bit4: 8 LSBs bit3: 4 LSBs bit2: 2 LSBs bit1: 1 LSB bit0: 0.5 LSB the offset range：+/- 63.5 LSBs | 8'h0 |
| Reserved | 15:8 | RO | not used | 8'h0 |
| cal\_offset\_b0 | 7:0 | RW | calibration offset for VREF=VREF\_V1P25, and VIN amplifier off bit7: sign bit6: 32 LSBs bit5: 16 LSBs bit4: 8 LSBs bit3: 4 LSBs bit2: 2 LSBs bit1: 1 LSB bit0: 0.5 LSB the offset range：+/- 63.5 LSBs | 8'h0 |

### ADC\_GAIN\_B

Offset: 04C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RO | not used | 16'h0 |
| cal\_gain\_b | 15:0 | RW | calibration gain for VREF=VREF\_V1P25 (the same gain for VIN amplifier on/off ) bit15: sign bit14-0: lower 15 bits of 19-bit fraction the gain range: +/- 6.25% exampls: for 3.1%, cal\_gain = (0.031 \* (2\*\*19))=0x3f7d | 16'h0 |

### ADC\_OFFSET\_C

Offset: 050

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:24 | RO | not used | 8'h0 |
| cal\_offset\_c1 | 23:16 | RW | calibration offset for VREF=VREF\_EXT, and VIN amplifier on bit7: sign bit6: 32 LSBs bit5: 16 LSBs bit4: 8 LSBs bit3: 4 LSBs bit2: 2 LSBs bit1: 1 LSB bit0: 0.5 LSB the offset range：+/- 63.5 LSBs | 8'h0 |
| Reserved | 15:8 | RO | not used | 8'h0 |
| cal\_offset\_c0 | 7:0 | RW | calibration offset for VREF=VREF\_EXT, and VIN amplifier off bit7: sign bit6: 32 LSBs bit5: 16 LSBs bit4: 8 LSBs bit3: 4 LSBs bit2: 2 LSBs bit1: 1 LSB bit0: 0.5 LSB the offset range：+/- 63.5 LSBs | 8'h0 |

### ADC\_GAIN\_C

Offset: 054

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RO | not used | 16'h0 |
| cal\_gain\_c | 15:0 | RW | calibration gain for VREF=VREF\_EXT (the same gain for VIN amplifier on/off ) bit15: sign bit14-0: lower 15 bits of 19-bit fraction the gain range: +/- 6.25% exampls: for 3.1%, cal\_gain = (0.031 \* (2\*\*19))=0x3f7d | 16'h0 |

### ADC\_TEST

Offset: 058

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| test\_en | 31:25 | RW | Reserved | 7'h0 |
| force\_vin\_sel | 24:17 | RW | sar\_vin\_sel will be or will original o\_sar\_vin\_sel[7:0] | 8'h0 |
| cbutton\_test\_en | 16:16 | RW | Cbutton test enable 0:disable 1:enable | 1'h0 |
| Reserved | 15:5 | RO | not used | 11'h0 |
| calib\_opt | 4:4 | RW | ADC calibration option 0: ADC calibration parameter from register 1: ADC calibration parameter from efuse | 1'h1 |
| debug\_sel | 3:2 | RW | Debug output selection 00: output i\_eoc,i\_soc, i\_din[13:0] to debug pins 01: output dma\_rx\_ack, dma\_rx\_single,o\_sar\_vin\_sel[6:0],i\_gpt\_trig,i\_ks\_adc\_trig, ADC\_CLK, SINGLE, RESET\_N, SOC, EOC, to debug pins 10:output i\_adc\_sample,i\_adc\_comp,i\_adc\_swa[13:0] to debug pins 11:output adc\_clk,state\_c[1:0],o\_sar\_vin\_sel[7],i\_adc\_sample,i\_adc\_comp,cvd\_ch\_sel\_o[5:0],guard\_o,acquisition\_o,precharge\_b\_o,precharge\_a\_o to debug pins. | 2'h0 |
| ate\_sel | 1:0 | RW | 2-bit ADC test selection (connect analog ADC IP) 00: normal user mode 01: test internal reference voltage VREF\_BUF: output to sar\_vin\_3 10: test internal vin buffer VIN\_BUF: output to sar\_vin\_2 11: test internal keyin\_detect: output to sar\_vin\_1 | 2'h0 |

### ADC\_RDR0

Offset: 060

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:12 | RO |  | 20'h0 |
| read\_data\_ch0 | 11:0 | RO | FIFO read data register of ADC channel 0 | 12'h0 |

### ADC\_RDR1

Offset: 064

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:12 | RO |  | 20'h0 |
| read\_data\_ch1 | 11:0 | RO | FIFO read data register of ADC channel 1 | 12'h0 |

### ADC\_RDR2

Offset: 068

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:12 | RO |  | 20'h0 |
| read\_data\_ch2 | 11:0 | RO | FIFO read data register of ADC channel 2 | 12'h0 |

### ADC\_RDR3

Offset: 06c

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:12 | RO |  | 20'h0 |
| read\_data\_ch3 | 11:0 | RO | FIFO read data register of ADC channel 3 | 12'h0 |

### ADC\_RDR4

Offset: 070

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:12 | RO |  | 20'h0 |
| read\_data\_ch4 | 11:0 | RO | FIFO read data register of ADC channel 4 | 12'h0 |

### ADC\_RDR5

Offset: 074

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:12 | RO |  | 20'h0 |
| read\_data\_ch5 | 11:0 | RO | FIFO read data register of ADC channel 5 | 12'h0 |

### ADC\_RDR6

Offset: 078

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:12 | RO |  | 20'h0 |
| read\_data\_ch6 | 11:0 | RO | FIFO read data register of ADC channel 6 | 12'h0 |

### ADC\_RDR7

Offset: 07c

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:12 | RO |  | 20'h0 |
| read\_data\_ch7 | 11:0 | RO | FIFO read data register of ADC channel 7 | 12'h0 |

### ADC\_RDR8

Offset: 080

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:28 | RO | not used | 4'h0 |
| read\_data1\_ch8 | 27:16 | RO | FIFO read data1 of ADC channel 8 | 12'h0 |
| Reserved | 15:12 | RO | not used | 4'h0 |
| read\_data0\_ch8 | 11:0 | RO | FIFO read data0 of ADC channel 8 | 12'h0 |

### ADC\_RDR9

Offset: 084

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:28 | RO | not used | 4'h0 |
| read\_data1\_ch9 | 27:16 | RO | FIFO read data1 of ADC channel 9 | 12'h0 |
| Reserved | 15:12 | RO | not used | 4'h0 |
| read\_data0\_ch9 | 11:0 | RO | FIFO read data0 of ADC channel 9 | 12'h0 |

### ADC\_RDR10

Offset: 088

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:28 | RO | not used | 4'h0 |
| read\_data1\_ch10 | 27:16 | RO | FIFO read data1 of ADC channel 10 | 12'h0 |
| Reserved | 15:12 | RO | not used | 4'h0 |
| read\_data0\_ch10 | 11:0 | RO | FIFO read data0 of ADC channel 10 | 12'h0 |

### ADC\_RDR11

Offset: 08c

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:28 | RO | not used | 4'h0 |
| read\_data1\_ch11 | 27:16 | RO | FIFO read data1 of ADC channel 11 | 12'h0 |
| Reserved | 15:12 | RO | not used | 4'h0 |
| read\_data0\_ch11 | 11:0 | RO | FIFO read data0 of ADC channel 11 | 12'h0 |

### ADC\_RDR12

Offset: 090

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:28 | RO | not used | 4'h0 |
| read\_data1\_ch12 | 27:16 | RO | FIFO read data1 of ADC channel 12 | 12'h0 |
| Reserved | 15:12 | RO | not used | 4'h0 |
| read\_data0\_ch12 | 11:0 | RO | FIFO read data0 of ADC channel 12 | 12'h0 |

### ADC\_RDR13

Offset: 094

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:28 | RO | not used | 4'h0 |
| read\_data1\_ch13 | 27:16 | RO | FIFO read data1 of ADC channel 13 | 12'h0 |
| Reserved | 15:12 | RO | not used | 4'h0 |
| read\_data0\_ch13 | 11:0 | RO | FIFO read data0 of ADC channel 13 | 12'h0 |

### DMA\_RDR

Offset: 0A0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| rdata\_dma | 31:0 | RO | Read data Register of DMA | 32'h0 |

### CVD\_CTRL

Offset: 0A4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:3 | RO | not used | 29'h0 |
| guard\_ring\_en | 2:2 | RW | Guard Ring Enable | 1'h0 |
| double\_samp\_en | 1:1 | RW | Double sample enable 0:Disable double sample function 1:Enable double sample function | 1'h0 |
| adipen | 0:0 | RW | When double\_samp\_en =1: 0:second charge direction same as first charge 1:second charge direction inverted first charge | 1'h0 |

### PRE\_CHA\_TIMER

Offset: 0A8

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:8 | RO | not used | 24'h0 |
| precha\_timer | 7:0 | RW | set PrechargeA count number if set N to this register，Precharge A will pull up for N+1 clock cycle | 8'h0 |

### PRE\_CHB\_TIMER

Offset: 0AC

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:8 | RO | not used | 24'h0 |
| prechb\_timer | 7:0 | RW | set PrechargeB count number if set N to this register，Precharge B will pull up for N+1 clock cycle | 8'h0 |

### ACQ\_TIMER

Offset: 0B0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:8 | RO | not used | 24'h0 |
| acq\_timer | 7:0 | RW | set Acquicition count number if set N to this register，Acquicition will pull up for N+1 clock cycle | 8'h0 |

### ADD\_CAP

Offset: 0B4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:3 | RO | not used | 29'h0 |
| add\_cap\_sel | 2:0 | RW | Additional Capacitor Select | 3'h0 |

# CODEC

## Introduction

在CP端，不开放给用户

## Main Features

## Function Description

## Basic Block Diagram

## CODEC Register

### AUD\_R0\_RSVD\_REG0

Offset: 000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RW | not used | 16'h0 |
| aud\_reg\_rsvd | 15:2 | RW | reserved for future | 14'h3c00 |
| aud\_adc\_deskew\_clk\_inv | 1:1 | RW | the edge selection for ADC deskew clock (24Mhz source clock) 0: negedge 1: posedge | 1'h0 |
| aud\_reg\_rsvd\_0 | 0:0 | RW | reserved for future | 1'h0 |

### AUD\_R1\_GLOBAL0

Offset: 004

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RW | Resverved for further | 16'h0 |
| adc\_anaclk\_inv | 15:15 | RW | ADC analog clock invert (option) 1: invert 0: buffer | 1'h0 |
| test\_out\_sela | 14:10 | RW | test signal selection output from pin aud\_debug\_clk (for test). 1:   adc\_mclk\_test 2:   aud\_adc\_clk\_ana  3:   aud\_adc\_fsclk  4:   aud\_dmic\_clk  5:   adc\_clk\_cic  6:   adc\_clk\_fil  7: aud\_adc\_clk\_idac\_bias 8:   adclrc\_core  9: aud\_adc\_lp\_rst 10:  dac\_mclk\_test  11:  aud\_dac\_clk\_test  12:  aud\_dac\_clk\_ana  13:  aud\_dac\_fsclk  14:  dac\_clk\_fil  15:  dac\_clk\_sdm 16: daclrc\_core 17: aud\_adc\_cap\_cali\_flag 18: aud\_lpga\_zcflag 19: aud\_rpga\_zcflag 20: aud\_dacl\_zcflag 21: aud\_dacr\_zcflag others: 1'b0 | 5'h0 |
| test\_out\_selb | 9:6 | RW | selects the internal signals as output from aud\_debug\_data[15:0] 3: { 9'd0, aud\_adc\_clk\_ana, aud\_adc\_sdm\_out\_r[2:0], aud\_adc\_sdm\_out\_l[2:0] } 4: { 6'd0, soft\_mute\_state\_r, soft\_mute\_state\_l, dac\_dig\_debug\_data[7:0] } 5: {2'b00, aud\_dac\_tc\_r[13:0]} 6: {2'b00, aud\_dac\_tc\_l[13:0]} 7: dac\_dig\_debug\_data[15:0] Others: 16'd0 | 4'h0 |
| test\_in\_sel | 5:3 | RW | Input data mux selection from aud\_debug\_data\_in[7:0]; 1: dac analog debug data[7:0] is selected from debug\_data\_in[7:0] 2: adc digital input data comes from debug\_data\_in[5:0]; others: invalid | 3'h0 |
| debug\_mode | 2:0 | RW | Debug mode selection 1: adc\_ana\_debug\_mode 2: adc\_dig\_debug\_mode 4. dac\_fil\_debug\_mode (dac\_dig\_debug\_data = dac\_out\_l\_vmux[23:8]) 5: dac\_ana\_debug\_mode 6: dac\_dig\_debug\_mode (dac\_dig\_debug\_data = {8'd0, sd\_out\_r[3:0], sd\_out\_l[3:0]}) Others: invalid | 3'h0 |

### AUD\_R2\_GLOBAL1

Offset: 008

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:11 | RW | Resverved for further | 21'h0 |
| aud\_vmiden | 10:10 | RW | Codec VMID enable 1: enable 0: disable | 1'h0 |
| aud\_vmidr\_sel | 9:9 | RW | Codec VMID Resistor Select  0: 50Kohm(default) 1: 200Kohm | 1'h0 |
| aud\_irefen | 8:8 | RW | Codec REF IREF GEN enable 1: enable 0: disable | 1'h0 |
| aud\_dac\_irefsel | 7:6 | RW | Codec REF DAC path IREF Adjust 00: 1.5uA 01: 2uA (default)  10: 2.5uA  11: 3.0uA | 2'h1 |
| aud\_mb1\_set | 5:4 | RW | Codec MicBias1 setting 00: disble 01: 1.8v 10: 2.0V 11: 2.3v | 2'h2 |
| aud\_mb1\_miller | 3:3 | RW | Codec MicBias1 compensation mode 0: external cap without Miller Comp;  1: No external Cap with Miller Comp; | 1'h0 |
| aud\_mb0\_set | 2:1 | RW | Codec MicBias0 setting 00: disble 01: 1.8v 10: 2.0V 11: 2.3v | 2'h2 |
| aud\_mb0\_miller | 0:0 | RW | Codec MicBias0 compensation mode 0: external cap without Miller Comp; 1: No external Cap with Miller Comp; | 1'h0 |

### AUD\_R3\_RSVD\_REG1

Offset: 00C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:0 | RW | Resverved for further | 32'h0 |

### AUD\_R4\_RSVD\_REG2

Offset: 010

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:0 | RO | not used | 32'h0 |

### AUD\_R5\_ADC\_CTRL0

Offset: 014

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:14 | RW | not used | 18'h0 |
| adc\_cap\_cali\_go | 13:13 | RW | A write of 1 will start the ADC CAP calibration process. And no influence if a write of 0 to the bit. Needs to clear this bit before starting cap calibration. Please refer to AU\_ADC\_TRIM.docx for details. | 1'h0 |
| adc\_rc\_cali\_go | 12:12 | RW | A write of 1 will start the ADC QUAR calibration process. And no influence if a write of 0 to the bit. Needs to clear this bit before starting quar calibration. Please refer to AU\_ADC\_TRIM.docx for details. | 1'h0 |
| aud\_adc\_mclk\_sel | 11:11 | RW | 0: 24MHz 1: 22.05MHz | 1'h0 |
| lpga\_toen\_p | 10:10 | RW | Zero Crossing Time Out enable for ADC Left PGA gain setting. 1: enable 0: disable | 1'h1 |
| rpga\_toen\_p | 9:9 | RW | Zero Crossing Time Out enable for ADC Right PGA gain setting. 1: enable 0: disable | 1'h1 |
| adc\_resetn\_p | 8:8 | RW | ADC digital reset control. All the digital data-path of ADC will be reset execept their registers. 0: reset 1: release | 1'h0 |
| adcclk\_en\_p | 7:7 | RW | ADC internal clock generation enable, including adc\_clk\_ana, adc\_clk\_fil, adc\_clk\_cic and adc\_clk\_ibias. 1: enable 0: disabled | 1'h0 |
| adcosr\_p | 6:4 | RW | ADC Over Sample ratio settings. 000: 500x 001: 250x  010: 125x 011: 100x  100: 50x others: invalid | 3'h1 |
| adcsr\_p | 3:0 | RW | ADC Sample Rate settings. 0000: 8Khz 0011: 16Khz 1000: 44.1khz/48Khz  others: invalid for 44.1KHz PLL should be used to supply codec adc clk at 22.05MHz | 4'h3 |

### AUD\_R6\_ADC\_CTRL1

Offset: 018

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:31 | RW | not used | 1'h0 |
| adc\_hpfout\_sel | 30:30 | RW | Select the output of High Pass filter 1 as ADC output (for test) 0: normal output 1: selected. | 1'h0 |
| adc\_single\_ch\_mode | 29:29 | RW | ADC single channel mode enable (for test). 1: enable. The output data from right channel is same with that of ADC left channel. 0: disable. | 1'h0 |
| hpfcut | 28:26 | RW | The cut off frequency setttings of the High pass filter2  48Khz: 000: 122Hz 001: 153Hz  010: 156Hz 011: 245Hz 100: 306Hz 101: 392Hz  110: 490Hz 111: 612Hz 44.1KHz: 000: 112Hz 001: 140Hz  010: 143Hz 011: 225Hz 100: 281Hz 101: 360Hz  110: 450Hz 111: 562Hz 32KHz: 000: 81Hz 001: 102Hz  010: 104Hz 011: 163Hz 100: 204Hz 101: 261Hz  110: 372Hz 111: 408Hz 24KHz: 000: 61Hz 001: 77Hz  010: 78Hz 011: 123Hz 100: 153Hz 101: 192Hz  110: 245Hz 111: 306Hz 16KHz: 000: 41Hz 001: 51Hz  010: 52Hz 011: 82Hz 100: 102Hz 101: 131Hz  110: 186Hz 111: 204Hz 8KHz: 000: 21Hz 001: 25Hz  010: 26Hz 011: 41Hz 100: 51Hz 101: 66Hz  110: 93Hz 111: 102Hz For other ADCSR setting, Please zoom in/out linearly and calculate their cut-off frequency; | 3'h3 |
| adc\_pga\_level\_l | 25:21 | RW | ADC Left PGA volume control. -12dB to +36dB @ 2db/step 00000: -12dB 00001: -10dB 00010: -8dB … 2dB@step … 10111: +34dB 11000: +36dB others: invalid | 5'h6 |
| adc\_pga\_level\_r | 20:16 | RW | ADC Right PGA volume control. -12dB to +36dB @ 2db/step 00000: -12dB 00001: -10dB 00010: -8dB … 2dB@step … 10111: +34dB 11000: +36dB others: invalid | 5'h6 |
| hpf1en | 15:15 | RW | 1st HPF enable. (typical 3.7Hz) 1: enable 0: disable | 1'h0 |
| hpf2en | 14:14 | RW | 2nd HPF enable. 1: enable 0: disable | 1'h0 |
| adcvol\_l | 13:7 | RW | ADC Left channel digital gain. 0000000: digital mute 0000001: digital mute 0000010: -83dB 0000011:-82dB … 1dB@step… 1010101: 0dB … 1dB@step… 1111101:+40dB 1111110: +41dB 1111111: +42dB | 7'h55 |
| adcvol\_r | 6:0 | RW | ADC Right channel digital gain. 0000000: digital mute 0000001: digital mute 0000010: -83dB 0000011:-82dB … 1dB@step… 1010101: 0dB … 1dB@step… 1111101:+40dB 1111110: +41dB 1111111: +42dB | 7'h55 |

### AUD\_R7\_ADC\_CTRL2

Offset: 01C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:30 | RW | not used | 2'h0 |
| NFA1 | 29:16 | RW | ADC Notch filter NFA1 coefficient | 14'h0 |
| Reserved | 15:15 | RW | not used | 1'h0 |
| NFEN | 14:14 | RW | ADC Notch filter enable 1: enable 0: disable | 1'h0 |
| NFA0 | 13:0 | RW | ADC Notch filter NFA0 coefficient | 14'h0 |

### AUD\_R8\_ADC\_CTRL3

Offset: 020

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| tolerance | 31:29 | RW | ADC ALC target error tolerance setting. 011: +/- 3dB 010: +/-2dB 100: +/- 4dB 001: +/-1dB 000: +/- 0dB …… | 3'h3 |
| target\_l | 28:24 | RW | ADC Left channel ALC target level. 00000: -1dB 00001: -3dB …-2dB@step 10110: -45dB 10111: -47dB other: invalid | 5'h3 |
| target\_r | 23:19 | RW | ADC Right channel ALC target level. 00000: -1dB 00001: -3dB …-2dB@step 10110: -45dB 10111: -47dB other: invalid | 5'h3 |
| alcmode | 18:18 | RW | ADC ALC mode selection. 1: limiter mode 0: normal mode | 1'h0 |
| alcsel\_l | 17:17 | RW | ADC Left ALC function enable. 1: ALC enable 0: ALC disable | 1'h0 |
| alcsel\_r | 16:16 | RW | ADC Right ALC function enable. 1: ALC enable 0: ALC disable | 1'h0 |
| ng\_en | 15:15 | RW | ADC ALC Noise Gate enable. 1: Enable 0: Disable | 1'h0 |
| ng | 14:10 | RW | ADC ALC noise floor level setting. 00000: -40dB 00001: -42dB 00010: -44dB …-2dB@step... 10101: -82dB 10110: -84dB others: invalid | 5'ha |
| alcmax | 9:5 | RW | Max ADC PGA gain used in ALC mode. Min: 00000 Max: 11000 | 5'h18 |
| alcmin | 4:0 | RW | Min ADC PGA gain used in ALC mode Min: 00000 Max: 11000 | 5'h0 |

### AUD\_R9\_ADC\_CTRL4

Offset: 024

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:28 | RW | not used | 4'h0 |
| autorst\_en\_l | 27:27 | RW | Auto reset the analog SDM of ADC left channel if a predefined amout of all zeros or all ones are matched. 1: enable, reset will hold 8 clock cycles and self\_clearing. 0: disable | 1'h0 |
| autorst\_en\_r | 26:26 | RW | Auto reset the analog SDM of ADC Right channel if a predefined amout of all zeros or all ones are matched. 1: enable, reset will hold 8 clock cycles and self\_clearing. 0: disable | 1'h0 |
| autorst\_type | 25:23 | RW | predefined time range for auto reset function (16Khz). 000: 128us 001: 256us  010: 512us 011: 1ms 100: 2ms 101: 4ms | 3'h1 |
| dmic\_enable | 22:22 | RW | Digital Microphone mode enable 0: Audio DSP input is from ADC SDM, and DMIC input is disabled. 1: Audio DSP input is from digital microphone input. | 1'h0 |
| dmic\_src | 21:21 | RW | Selects digital microphone data nput if dmic\_mode is set to 1. 0: from DMIC0\_IN 1: from DMIC1\_IN | 1'h0 |
| dmic\_latch\_adj | 20:19 | RW | DMIC input data timing adjust range. 00: 0 degree delay 01: 90 degree delay 10: 180 degree delay 11: 270 degree delay | 2'h0 |
| unconnect | 18:17 | RW |  | 2'h0 |
| dmic\_mode | 16:16 | RW | Left data is sampled by the falling edge of dmic\_clk, and Right data is sampled by the rising edge of dmic\_clk. 0: Single edge on DMIC0\_IN and DMIC1\_IN; 1: Double edge on either DMIC0\_IN or DMIC1\_IN, which decided by dmic\_srcr; | 1'h1 |
| Reserved | 15:13 | RW | not used | 3'h0 |
| peak\_fastalc\_en | 12:12 | RW | ALC enable at once if peak amplitude of input signal is greater than -1.5 dBFS. | 1'h0 |
| alchld | 11:8 | RW | ADC ALC hold time before gain is increased (ADCSR=48Khz). 0000 = 1.34ms 0001 = 2.67ms  0010 = 5.33ms 0011 = 10.66ms  …time doubles with every step 0111 = 0.17s 1000 = 0.34s  1001 = 0.68s  ….. 1111 = 43s | 4'h1 |
| alcatk | 7:4 | RW | ADC ALC attack (gain ramp-down) time (Fs = 48kHz and ALCMODE == 0) . 0000 = 83.2us 0001 = 166.4us  0010 = 333us 0011 = 666us  …time doubles with every step 1111 = 2.7s ADC ALC attack (gain ramp-down) time (Fs = 48Khz and ALCMODE == 1) . 0000 = 20.8us 0001 = 41.6us  0010 = 83.2us 0011 = 166.4us  …time doubles with every step 1111 = 0.68s | 4'h4 |
| alcdcy | 3:0 | RW | ADC Decay (gain ramp-up) time (Fs=48kHz and ALCMODE ==0) . 0000 = 333us 0001 = 666us  0010 = 1.33ms 0011 = 2.66ms  …time doubles with every step 1111 = 10.8s ADC Decay (gain ramp-up) time (Fs=48kHz and ALCMODE ==1) . 0000 = 83.2us 0001 = 166.4us  0010 = 333us 0011 = 666us  …time doubles with every step 1111 = 2.7s | 4'h6 |

### AUD\_R10\_ADC\_CTRL5

Offset: 028

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| offset\_regen | 31:31 | RW | Add an offset to ADC output (for test). 1: enable 0: disable | 1'h0 |
| offset\_reg | 30:21 | RW | offset value from register settings. | 10'h0 |
| filgain\_regen | 20:20 | RW | Add a gain to ADC data path (for debug). 1: enable 0: disable | 1'h0 |
| filgain\_reg | 19:0 | RW | gain setting from register. | 20'h188b4 |

### AUD\_R11\_ADC\_CTRL6

Offset: 02C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:31 | RW | not used | 1'h0 |
| rset\_adc\_cap\_cali | 30:30 | RW | register select enable for ADC cap calibration. 1: calibration setting from register, and not from internal state machine. 0: calibration value is from internal state machine. | 1'h0 |
| rval\_adc\_cap\_cali | 29:25 | RW | register setting value. | 5'h0 |
| rctrl\_adc\_cap\_cali | 24:24 | RW | register control enable. 1: enable 0: disable | 1'h0 |
| rval\_adc\_cap\_cali\_en | 23:23 | RW | calibraton enable from register. | 1'h0 |
| rset\_adc\_rc\_cali | 22:22 | RW | rc calibration setting from register. | 1'h0 |
| rval\_adc\_quar\_cov | 21:16 | RW | quar calibration value from register. | 6'h0 |
| rctrl\_adc\_rc\_cali | 15:15 | RW | rc calibration setting from register. | 1'h0 |
| rval\_adc\_quar\_cov\_en | 14:14 | RW | ADC vref setting time calibration enable. | 1'h0 |
| aud\_adc\_rc\_ctrl | 13:9 | RW | [0]: SAR ADC Comparator bias current control  0:20uA，1:10uA [1]: Integrator input resistance, feedback capacitor control.   1: 12MHz FS   0: 04MHz FS (4MHz/16K default) others: invalid (only bit[0] bit[1] are valid) | 5'h0 |
| aud\_adc\_idac\_ctrl | 8:7 | RW | Feedback IDAC Control 00: x-5% 01: 7.5uA (expected default)---Full scale 1.5Vrms 10: x+6% 11: x+13% | 2'h0 |
| aud\_adc\_ib\_ctrl | 6:3 | RW | ADC amp ibias control 0000: 1.5uA 0001: 2uA (expected default) 0010: 2.5uA 0011: 3uA bit[2]=1: amp1 low power mode, bit[3]=1: amp2 low power mode | 4'h0 |
| aud\_adc\_lp\_rst | 2:2 | RW | ADC Loop path reset. 0: no reset 1: Loop reset | 1'h1 |
| adc\_en\_l | 1:1 | RW | ADC Left enable. 0: Power Down; 1: Enabled | 1'h0 |
| adc\_en\_r | 0:0 | RW | ADC Right enable. 0: Power Down; 1: Enabled | 1'h0 |

### AUD\_R12\_ADC\_CTRL7

Offset: 030

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:26 | RW | not used | 6'h0 |
| rset\_aud\_idac\_bias | 25:25 | RW | register control enable. 1: enable 0: disable | 1'h0 |
| rval\_aud\_idac\_bias | 24:24 | RW | register value for aud\_idac\_bias. | 1'h0 |
| aud\_adc\_idac\_trim | 23:21 | RW |  | 3'h0 |
| aud\_adc\_vref\_ctrl | 20:19 | RW | Control vref voltage. Control vref power consumption 00: default VREF\_CTRL0>:1—— ADC VREF 40uA；0—— ADC VREF 20uA； VREF\_CTRL1>:1—— ADC VCM BIAS 30uA； 0—— ADC VCM BIAS 10uA | 2'h0 |
| aud\_adc\_vref\_mode | 18:18 | RW | Control generate vref mode. | 1'h0 |
| aud\_adc\_vref\_en | 17:17 | RW | VREF\_EN 1: enable 0: disable | 1'h0 |
| aud\_adc\_idac\_offset | 16:15 | RW | feed back idac DC offset.  00: 0uA(default); 01: LSB/16;  10: LSB/8; 11: LSB/4 | 2'h0 |
| aud\_adc\_atb | 14:12 | RW | ADC atb control 000:ATB output H-Z; 001: test ADC0 IDAC\_NMOS; 010: test ADC0 IDAC\_PMOS; 011: test ADC1 IDAC\_NMOS; 100: test ADC1 IDAC\_PMOS 000:default | 3'h0 |
| lpga\_zcen\_p | 11:11 | RW | Zero Crossing enable for ADC Left PGA. 1: enable 0: disable | 1'h1 |
| aud\_lpga\_lpr | 10:10 | RW | ADC Left PGA power mode,  1: Low Power Mode 0: normal mode. | 1'h0 |
| aud\_lpga\_en\_single | 9:9 | RW | ADC Left PGA input mode. 0: differetial 1: Single | 1'h1 |
| aud\_lpga\_mute | 8:8 | RW | ADC Left PGA Mute,  1: mute 0: working | 1'h1 |
| lpga\_en | 7:7 | RW | ADC Left Input PGA enable;  0= Power Down; 1= Enable; | 1'h0 |
| rpga\_zcen\_p | 6:6 | RW | Zero Crossing enable for ADC Right PGA 1: enable 0: disable | 1'h1 |
| aud\_rpga\_lpr | 5:5 | RW | ADC Right PGA power mode,  1: Low Power Mode 0: normal mode | 1'h0 |
| aud\_rpga\_en\_single | 4:4 | RW | ADC Right PGA input mode,  0: differetial 1: Single | 1'h1 |
| aud\_rpga\_mute | 3:3 | RW | ADC Right PGA Mute,  1: mute 0: working | 1'h1 |
| rpga\_en | 2:2 | RW | ADC Right Input PGA enable;  0: Power Down; 1: Enable; | 1'h0 |
| aud\_pgabuf\_lp | 1:1 | RW | ADC LPGA/RPGA Buffer low power mode 1: low power mode 0: normal mode | 1'h0 |
| aud\_pgabufen | 0:0 | RW | LPGA/RPGA Buffer Enable 1: enable 0: disable | 1'h0 |

### AUD\_R13\_RSVD\_REG3

Offset: 034

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:0 | RO | not used | 32'h0 |

### AUD\_R14\_RSVD\_REG4

Offset: 038

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:0 | RO | not used | 32'h0 |

### AUD\_R15\_DAC\_CTRL0

Offset: 03C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:10 | RW |  | 22'h0 |
| aud\_dac\_mclk\_sel | 9:9 | RW | audio main clock selection for DAC Left/Right channel 1: 22.05Mhz 0: 24Mhz | 1'h0 |
| dacl\_toen\_p | 8:8 | RW | DAC Left Zero Crossing Time-out enable 1: enable 0: disable | 1'h1 |
| dacr\_toen\_p | 7:7 | RW | DAC Right Zero Crossing Time-out enable 1: enable 0: disable | 1'h1 |
| dac\_rstn\_p | 6:6 | RW | DAC digital reset except registers. 0: reset 1: release | 1'h0 |
| dacclk\_en\_p | 5:5 | RW | DAC clock generation enable, including dac\_clk\_fil, dac\_clk\_sdm and dac\_clk\_ana. 1: enable 0: disabled | 1'h0 |
| dacosr\_p | 4:4 | RW | DAC Over Sample ratio. 0: 250x 1:125x | 1'h0 |
| dacsr\_p | 3:0 | RW | DAC sampling rate setting. 0000: fs = 8k 0001: fs = 11.025k  0010: fs = 12k 0011: fs = 16k  0100: fs = 22.05k 0101: fs = 24k  0110: fs = 32k 0111: fs = 44.1k  1000: fs = 48k 1001: fs = 96k  1010: fs = 88.2k 1011~1111: reserved for 11.025KHz, 44.1KHz and 88.2KHz PLL should be used to supply codec dac clk at 22.05MHz | 4'h8 |

### AUD\_R16\_DAC\_CTRL1

Offset: 040

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RW | not used | 16'h0 |
| dac\_gain\_l | 15:8 | RW | DAC Left Digital Volume Control Others = Digital Mute 01110000: -113dB 01110001: -112dB ... 1dB@step… 11111110: +29dB 11111111: +30dB | 8'he1 |
| dac\_gain\_r | 7:0 | RW | DAC Right Digital Volume Control Others = Digital Mute 01110000: -113dB 01110001: -112dB ... 1dB@step… 11111110: +29dB 11111111: +30dB | 8'he1 |

### AUD\_R17\_DAC\_CTRL2

Offset: 044

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:11 | RW | not used | 21'h0 |
| dacl\_inv\_bf\_sdm | 10:10 | RW | DAC Left Data Invert before DAC SDM. 1: invert 0: buffer | 1'h0 |
| dacr\_inv\_bf\_sdm | 9:9 | RW | DAC Right Data Invert before DAC SDM 1: invert 0: buffer | 1'h0 |
| dacmu\_l | 8:8 | RW | DAC Left Digital Mute;  1: mute; 0: no mute (signal active) | 1'h1 |
| dacmu\_r | 7:7 | RW | DAC Right Digital Mute;  1: mute; 0: no mute (signal active) | 1'h1 |
| dacl\_dwaen | 6:6 | RW | DAC Left DWA Enable Signal. 0: disable; 1: enable. | 1'h1 |
| dacr\_dwaen | 5:5 | RW | DAC Right DWA Enable Signal. 0: disable; 1: enable. | 1'h1 |
| softmute\_en | 4:4 | RW | DAC Digital soft mute enable signal;  0: disable; 1: enable. | 1'h1 |
| soft\_speed | 3:0 | RW | DAC Soft mute decay/attach speed settings 0000: 1dB adjust / (2\*T\_DACLRC) 0001: 1dB adjust / (4\*T\_DACLRC) … double adjust time per step … 1111: 1dB adjust / (2^16\*T\_DACLRC) (where T\_DACLRC is the period of DACLRC, 20.8us for 48KHz) | 4'h0 |

### AUD\_R18\_DAC\_CTRL3

Offset: 048

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:15 | RW | not used | 17'h0 |
| dac\_dwa\_type | 14:12 | RW | DAC DWA Type (only for test) 000: circling DWA starting address; 001: increment DWA starting at every clock cycles; 010: shifting two times at every clock cycles; 011: random increment DWA starting address at every clock cycle; 100: add 2bit random increment into DWA starting address; 101: add 3bit random increment into DWA starting address; | 3'h1 |
| dac\_dith\_ntf\_en | 11:11 | RW | DAC Dither Noise-shaping Enable 1: enable 0: disable | 1'h1 |
| dac\_sd\_level\_sel | 10:10 | RW | DAC SDM level option (only for testing) 1: 15 level setting. The feedback path of DAC SDM is a little increase in its amplitude. 0: 15 level and the middle level is sited at level7(default) | 1'h0 |
| dac\_sd\_amute\_en\_l | 9:9 | RW | DAC Left SDM auto bypass enable when all zero’s input; 0: disable 1: enable; | 1'h1 |
| dac\_sd\_amute\_en\_r | 8:8 | RW | DAC Right SDM auto bypass enable when all zero’s input; 0: disable 1: enable; | 1'h1 |
| dac\_sd\_amute\_type | 7:5 | RW | DAC SDM auto mute threshold (# of all zero’s input): 000: shorter … 101: longer Others: not used. | 3'h2 |
| dac\_sd\_rstn | 4:4 | RW | DAC SDM reset, active low 0: reset 1: release | 1'h0 |
| dac\_sd\_nz | 3:3 | RW | DAC SDM no-zero input signal (for test) 1: no-zero 0: normal | 1'h0 |
| dac\_dith\_bypass | 2:2 | RW | DAC SDM dither function bypass 1: bypass 0: function used | 1'h0 |
| dac\_dith\_type | 1:0 | RW | DAC SDM dither type selection (for testing) | 2'h0 |

### AUD\_R19\_DAC\_CTRL4

Offset: 04C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RW | not used | 16'h0 |
| dac\_offset | 15:0 | RW | Added DAC offset (option for testing) | 16'h0 |

### AUD\_R20\_DAC\_CTRL5

Offset: 050

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:26 | RW | not used | 6'h0 |
| dac\_dith\_pow | 25:0 | RW | DAC digital SDM dither power (for testing) | 26'h300000 |

### AUD\_R21\_DAC\_CTRL6

Offset: 054

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:11 | RW | not used | 21'h0 |
| aud\_dac\_clk\_test\_en | 5:5 | RW | enable control for DAC test clock output (for test) 1: enable 0: disable | 1'h0 |
| aud\_dac\_dct\_rstn | 4:4 | RW | DAC dct reset (analog), active low 0: reset 1: release | 1'h0 |
| dacl\_zc\_en\_p | 3:3 | RW | DAC Left Zero Crossing enable 1: enable 0: disable | 1'h1 |
| dacr\_zc\_en\_p | 2:2 | RW | DAC Right Zero Crossing enable 1: enable 0: disable | 1'h1 |
| dacen\_l | 1:1 | RW | DAC Left enable;  0: Power Down; 1: Enabled | 1'h0 |
| dacen\_r | 0:0 | RW | DAC Right enable;  0: Power Down; 1: Enabled | 1'h0 |

### AUD\_R22\_DAC\_CTRL7

Offset: 058

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RW | not used | 16'h0 |
| lolvol\_p | 15:12 | RW | DAC Left Line Out Volume Control 0000 = -24dB 0001 = -22dB 0010 = -20dB … 2dB steps up to ... 1110 = +4dB 1111 = +6dB | 4'h9 |
| lorvol\_p | 11:8 | RW | DAC Right Line Out Volume Control 0000 = -24dB 0001 = -22dB 0010 = -20dB … 2dB steps up to ... 1110 = +4dB 1111 = +6dB | 4'h9 |
| lolp\_mute | 7:7 | RW | 0: DAC Left Line-Out normal work(P) 1: DAC Left Line-Out mute(P) | 1'h1 |
| loln\_mute | 6:6 | RW | 0: DAC Left Line-Out normal work (N) 1: DAC Left Line-Out mute(N) | 1'h1 |
| lorp\_mute | 5:5 | RW | 0: DAC Right Line-Out normal work(P) 1: DAC Right Line-Out mute(P) | 1'h1 |
| lorn\_mute | 4:4 | RW | 0: DAC Right Line-Out normal work(N) 1: DAC Right Line-Out mute(N) | 1'h1 |
| lolp\_en | 3:3 | RW | DAC Left Line Out enable (P) 1: enable 0:disable | 1'h0 |
| loln\_en | 2:2 | RW | DAC Left Line Out enable (N) 1: enable 0: disable | 1'h0 |
| lorp\_en | 1:1 | RW | DAC Right Line Out enable (P) 1: enable 0: disable | 1'h0 |
| lorn\_en | 0:0 | RW | DAC Right Line Out enable (N) 1: enable 0: disable | 1'h0 |

### AUD\_R25\_STATUS0

Offset: 064

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:17 | RO | not used | 15'h0 |
| adc\_cap\_cali | 16:12 | RO |  | 5'h10 |
| adc\_quar\_cov | 11:6 | RO |  | 6'h20 |
| adc\_cap\_cali\_fail | 5:5 | RO |  | 1'h0 |
| adc\_cap\_cali\_done | 4:4 | RO |  | 1'h0 |
| adc\_cap\_cov\_cali\_flag\_inst | 3:3 | RO |  | 1'h0 |
| adc\_rc\_cali\_fail | 2:2 | RO |  | 1'h0 |
| adc\_rc\_cali\_done | 1:1 | RO |  | 1'h0 |
| adc\_quar\_cov\_cali\_flag\_inst | 0:0 | RO |  | 1'h0 |

# AON\_CODEC

## Introduction

在CP端，不开放给用户

## Main Features

## Function Description

## Basic Block Diagram

## AON\_CODEC Register

### AUD\_R0\_RSVD\_REG0

Offset: 000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RW | not used | 16'h0 |
| aon\_aud\_reg\_rsvd | 15:1 | RW | reserved for future | 15'h7800 |
| aon\_aud\_deskew\_clk\_inv | 0:0 | RW | the edge selection for ADC deskew clock (24Mhz source clock) 0: negedge 1: posedge | 1'h0 |

### AUD\_R1\_GLOBAL0

Offset: 004

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:16 | RW | Resverved for further | 16'h0 |
| aon\_adc\_anaclk\_inv | 15:15 | RW | ADC analog clock invert (option) 1: invert 0: buffer | 1'h0 |
| aon\_test\_out\_sela | 14:10 | RW | test signal selection output from pin aud\_debug\_clk (for test). 1:   adc\_mclk\_test 2:   aud\_adc\_clk\_ana  3:   aud\_adc\_fsclk  4:   aud\_dmic\_clk  5:   adc\_clk\_cic  6:   adc\_clk\_fil  7: aud\_adc\_clk\_idac\_bias 8:   adclrc\_core  9: aud\_adc\_lp\_rst 10: aud\_adc\_cap\_cali\_flag 11: aud\_lpga\_zcflag 12: aud\_rpga\_zcflag others: 1'b0 | 5'h0 |
| aon\_test\_out\_selb | 9:6 | RW | selects the internal signals as output from aud\_debug\_data[15:0] 3: { 9'd0, aud\_adc\_clk\_ana, aud\_adc\_sdm\_out\_r[2:0], aud\_adc\_sdm\_out\_l[2:0] } Others: 16'd0 | 4'h0 |
| aon\_test\_in\_sel | 5:3 | RW | Input data mux selection from aud\_debug\_data\_in[7:0]; 2: adc digital input data comes from debug\_data\_in[5:0]; others: invalid | 3'h0 |
| aon\_debug\_mode | 2:0 | RW | Debug mode selection 1: adc\_ana\_debug\_mode 2: adc\_dig\_debug\_mode Others: invalid | 3'h0 |

### AUD\_R2\_GLOBAL1

Offset: 008

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:11 | RW | Resverved for further | 21'h0 |
| aon\_aud\_vmiden | 10:10 | RW | Codec VMID enable 1: enable 0: disable | 1'h0 |
| aon\_aud\_vmidr\_sel | 9:9 | RW | Codec VMID Resistor Select  0: 50Kohm(default) 1: 200Kohm | 1'h0 |
| aon\_aud\_irefen | 8:8 | RW | Codec REF IREF GEN enable 1: enable 0: disable | 1'h0 |
| aon\_aud\_dac\_irefsel | 7:6 | RW | Codec REF DAC path IREF Adjust 00: 1.5uA 01: 2uA (default)  10: 2.5uA  11: 3.0uA | 2'h1 |
| aon\_aud\_mb1\_set | 5:4 | RW | Codec MicBias1 setting 00: disble 01: 1.8v 10: 2.0V 11: 2.3v | 2'h2 |
| aon\_aud\_mb1\_miller | 3:3 | RW | Codec MicBias1 compensation mode 0: external cap without Miller Comp;  1: No external Cap with Miller Comp; | 1'h0 |
| aon\_aud\_mb0\_set | 2:1 | RW | Codec MicBias0 setting 00: disble 01: 1.8v 10: 2.0V 11: 2.3v | 2'h2 |
| aon\_aud\_mb0\_miller | 0:0 | RW | Codec MicBias0 compensation mode 0: external cap without Miller Comp; 1: No external Cap with Miller Comp; | 1'h0 |

### AUD\_R3\_RSVD\_REG1

Offset: 00C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:0 | RW | Resverved for further | 32'h0 |

### AUD\_R4\_RSVD\_REG2

Offset: 010

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:0 | RO | not used | 32'h0 |

### AUD\_R5\_ADC\_CTRL0

Offset: 014

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:14 | RW | not used | 18'h0 |
| aon\_adc\_cap\_cali\_go | 13:13 | RW | A write of 1 will start the ADC CAP calibration process. And no influence if a write of 0 to the bit. Needs to clear this bit before starting cap calibration. Please refer to AU\_ADC\_TRIM.docx for details. | 1'h0 |
| aon\_adc\_rc\_cali\_go | 12:12 | RW | A write of 1 will start the ADC QUAR calibration process. And no influence if a write of 0 to the bit. Needs to clear this bit before starting quar calibration. Please refer to AU\_ADC\_TRIM.docx for details. | 1'h0 |
| aon\_aud\_adc\_mclk\_sel | 11:11 | RW | only 24MHz is available for adc01 0: 24Mhz | 1'h0 |
| aon\_lpga\_toen\_p | 10:10 | RW | Zero Crossing Time Out enable for ADC Left PGA gain setting. 1: enable 0: disable | 1'h1 |
| aon\_rpga\_toen\_p | 9:9 | RW | Zero Crossing Time Out enable for ADC Right PGA gain setting. 1: enable 0: disable | 1'h1 |
| aon\_adc\_resetn\_p | 8:8 | RW | ADC digital reset control. All the digital data-path of ADC will be reset execept their registers. 0: reset 1: release | 1'h0 |
| aon\_adcclk\_en\_p | 7:7 | RW | ADC internal clock generation enable, including adc\_clk\_ana, adc\_clk\_fil, adc\_clk\_cic and adc\_clk\_ibias. 1: enable 0: disabled | 1'h0 |
| aon\_adcosr\_p | 6:4 | RW | ADC Over Sample ratio settings. 000: 500x 001: 250x  010: 125x 011: 100x  100: 50x others: invalid | 3'h1 |
| aon\_adcsr\_p | 3:0 | RW | ADC Sample Rate settings. 0000: 8Khz 0011: 16Khz 1000: 44.1khz/48Khz  others: invalid for 44.1KHz PLL should be used to supply codec adc clk at 22.05MHz | 4'h3 |

### AUD\_R6\_ADC\_CTRL1

Offset: 018

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:31 | RW | not used | 1'h0 |
| aon\_adc\_hpfout\_sel | 30:30 | RW | Select the output of High Pass filter 1 as ADC output (for test) 0: normal output 1: selected. | 1'h0 |
| aon\_adc\_single\_ch\_mode | 29:29 | RW | ADC single channel mode enable (for test). 1: enable. The output data from right channel is same with that of ADC left channel. 0: disable. | 1'h0 |
| aon\_hpfcut | 28:26 | RW | The cut off frequency setttings of the High pass filter2  48Khz: 000: 122Hz 001: 153Hz  010: 156Hz 011: 245Hz 100: 306Hz 101: 392Hz  110: 490Hz 111: 612Hz 44.1KHz: 000: 112Hz 001: 140Hz  010: 143Hz 011: 225Hz 100: 281Hz 101: 360Hz  110: 450Hz 111: 562Hz 32KHz: 000: 81Hz 001: 102Hz  010: 104Hz 011: 163Hz 100: 204Hz 101: 261Hz  110: 372Hz 111: 408Hz 24KHz: 000: 61Hz 001: 77Hz  010: 78Hz 011: 123Hz 100: 153Hz 101: 192Hz  110: 245Hz 111: 306Hz 16KHz: 000: 41Hz 001: 51Hz  010: 52Hz 011: 82Hz 100: 102Hz 101: 131Hz  110: 186Hz 111: 204Hz 8KHz: 000: 21Hz 001: 25Hz  010: 26Hz 011: 41Hz 100: 51Hz 101: 66Hz  110: 93Hz 111: 102Hz For other ADCSR setting, Please zoom in/out linearly and calculate their cut-off frequency; | 3'h3 |
| aon\_adc\_pga\_level\_l | 25:21 | RW | ADC Left PGA volume control. -12dB to +36dB @ 2db/step 00000: -12dB 00001: -10dB 00010: -8dB … 2dB@step … 10111: +34dB 11000: +36dB others: invalid | 5'h6 |
| aon\_adc\_pga\_level\_r | 20:16 | RW | ADC Right PGA volume control. -12dB to +36dB @ 2db/step 00000: -12dB 00001: -10dB 00010: -8dB … 2dB@step … 10111: +34dB 11000: +36dB others: invalid | 5'h6 |
| aon\_hpf1en | 15:15 | RW | 1st HPF enable. (typical 3.7Hz) 1: enable 0: disable | 1'h0 |
| aon\_hpf2en | 14:14 | RW | 2nd HPF enable. 1: enable 0: disable | 1'h0 |
| aon\_adcvol\_l | 13:7 | RW | ADC Left channel digital gain. 0000000: digital mute 0000001: digital mute 0000010: -83dB 0000011:-82dB … 1dB@step… 1010101: 0dB … 1dB@step… 1111101:+40dB 1111110: +41dB 1111111: +42dB | 7'h55 |
| aon\_adcvol\_r | 6:0 | RW | ADC Right channel digital gain. 0000000: digital mute 0000001: digital mute 0000010: -83dB 0000011:-82dB … 1dB@step… 1010101: 0dB … 1dB@step… 1111101:+40dB 1111110: +41dB 1111111: +42dB | 7'h55 |

### AUD\_R7\_ADC\_CTRL2

Offset: 01C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:30 | RW | not used | 2'h0 |
| aon\_NFA1 | 29:16 | RW | ADC Notch filter NFA1 coefficient | 14'h0 |
| Reserved | 15:15 | RW | not used | 1'h0 |
| aon\_NFEN | 14:14 | RW | ADC Notch filter enable 1: enable 0: disable | 1'h0 |
| aon\_NFA0 | 13:0 | RW | ADC Notch filter NFA0 coefficient | 14'h0 |

### AUD\_R8\_ADC\_CTRL3

Offset: 020

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| aon\_tolerance | 31:29 | RW | ADC ALC target error tolerance setting. 011: +/- 3dB 010: +/-2dB 100: +/- 4dB 001: +/-1dB 000: +/- 0dB …… | 3'h3 |
| aon\_target\_l | 28:24 | RW | ADC Left channel ALC target level. 00000: -1dB 00001: -3dB …-2dB@step 10110: -45dB 10111: -47dB other: invalid | 5'h3 |
| aon\_target\_r | 23:19 | RW | ADC Right channel ALC target level. 00000: -1dB 00001: -3dB …-2dB@step 10110: -45dB 10111: -47dB other: invalid | 5'h3 |
| aon\_alcmode | 18:18 | RW | ADC ALC mode selection. 1: limiter mode 0: normal mode | 1'h0 |
| aon\_alcsel\_l | 17:17 | RW | ADC Left ALC function enable. 1: ALC enable 0: ALC disable | 1'h0 |
| aon\_alcsel\_r | 16:16 | RW | ADC Right ALC function enable. 1: ALC enable 0: ALC disable | 1'h0 |
| aon\_ng\_en | 15:15 | RW | ADC ALC Noise Gate enable. 1: Enable 0: Disable | 1'h0 |
| aon\_ng | 14:10 | RW | ADC ALC noise floor level setting. 00000: -40dB 00001: -42dB 00010: -44dB …-2dB@step... 10101: -82dB 10110: -84dB others: invalid | 5'ha |
| aon\_alcmax | 9:5 | RW | Max ADC PGA gain used in ALC mode. Min: 00000 Max: 11000 | 5'h18 |
| aon\_alcmin | 4:0 | RW | Min ADC PGA gain used in ALC mode Min: 00000 Max: 11000 | 5'h0 |

### AUD\_R9\_ADC\_CTRL4

Offset: 024

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:28 | RW | not used | 4'h0 |
| aon\_autorst\_en\_l | 27:27 | RW | Auto reset the analog SDM of ADC left channel if a predefined amout of all zeros or all ones are matched. 1: enable, reset will hold 8 clock cycles and self\_clearing. 0: disable | 1'h0 |
| aon\_autorst\_en\_r | 26:26 | RW | Auto reset the analog SDM of ADC Right channel if a predefined amout of all zeros or all ones are matched. 1: enable, reset will hold 8 clock cycles and self\_clearing. 0: disable | 1'h0 |
| aon\_autorst\_type | 25:23 | RW | predefined time range for auto reset function (16Khz). 000: 128us 001: 256us  010: 512us 011: 1ms 100: 2ms 101: 4ms | 3'h1 |
| aon\_dmic\_enable | 22:22 | RW | Digital Microphone mode enable 0: Audio DSP input is from ADC SDM, and DMIC input is disabled. 1: Audio DSP input is from digital microphone input. | 1'h0 |
| aon\_dmic\_src | 21:21 | RW | Selects digital microphone data nput if dmic\_mode is set to 1. 0: from DMIC0\_IN 1: from DMIC1\_IN | 1'h0 |
| aon\_dmic\_latch\_adj | 20:19 | RW | DMIC input data timing adjust range. 00: 0 degree delay 01: 90 degree delay 10: 180 degree delay 11: 270 degree delay | 2'h0 |
| aon\_unconnect | 18:17 | RW |  | 2'h0 |
| aon\_dmic\_mode | 16:16 | RW | Left data is sampled by the falling edge of dmic\_clk, and Right data is sampled by the rising edge of dmic\_clk. 0: Single edge on DMIC0\_IN and DMIC1\_IN; 1: Double edge on either DMIC0\_IN or DMIC1\_IN, which decided by dmic\_srcr; | 1'h1 |
| Reserved | 15:13 | RW | not used | 3'h0 |
| aon\_peak\_fastalc\_en | 12:12 | RW | ALC enable at once if peak amplitude of input signal is greater than -1.5 dBFS. | 1'h0 |
| aon\_alchld | 11:8 | RW | ADC ALC hold time before gain is increased (ADCSR=48Khz). 0000 = 1.34ms 0001 = 2.67ms  0010 = 5.33ms 0011 = 10.66ms  …time doubles with every step 0111 = 0.17s 1000 = 0.34s  1001 = 0.68s  ….. 1111 = 43s | 4'h1 |
| aon\_alcatk | 7:4 | RW | ADC ALC attack (gain ramp-down) time (Fs = 48kHz and ALCMODE == 0) . 0000 = 83.2us 0001 = 166.4us  0010 = 333us 0011 = 666us  …time doubles with every step 1111 = 2.7s ADC ALC attack (gain ramp-down) time (Fs = 48Khz and ALCMODE == 1) . 0000 = 20.8us 0001 = 41.6us  0010 = 83.2us 0011 = 166.4us  …time doubles with every step 1111 = 0.68s | 4'h4 |
| aon\_alcdcy | 3:0 | RW | ADC Decay (gain ramp-up) time (Fs=48kHz and ALCMODE ==0) . 0000 = 333us 0001 = 666us  0010 = 1.33ms 0011 = 2.66ms  …time doubles with every step 1111 = 10.8s ADC Decay (gain ramp-up) time (Fs=48kHz and ALCMODE ==1) . 0000 = 83.2us 0001 = 166.4us  0010 = 333us 0011 = 666us  …time doubles with every step 1111 = 2.7s | 4'h6 |

### AUD\_R10\_ADC\_CTRL5

Offset: 028

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| aon\_offset\_regen | 31:31 | RW | Add an offset to ADC output (for test). 1: enable 0: disable | 1'h0 |
| aon\_offset\_reg | 30:21 | RW | offset value from register settings. | 10'h0 |
| aon\_filgain\_regen | 20:20 | RW | Add a gain to ADC data path (for debug). 1: enable 0: disable | 1'h0 |
| aon\_filgain\_reg | 19:0 | RW | gain setting from register. | 20'h188b4 |

### AUD\_R11\_ADC\_CTRL6

Offset: 02C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:31 | RW | not used | 1'h0 |
| aon\_rset\_adc\_cap\_cali | 30:30 | RW | register select enable for ADC cap calibration. 1: calibration setting from register, and not from internal state machine. 0: calibration value is from internal state machine. | 1'h0 |
| aon\_rval\_adc\_cap\_cali | 29:25 | RW | register setting value. | 5'h0 |
| aon\_rctrl\_adc\_cap\_cali | 24:24 | RW | register control enable. 1: enable 0: disable | 1'h0 |
| aon\_rval\_adc\_cap\_cali\_en | 23:23 | RW | calibraton enable from register. | 1'h0 |
| aon\_rset\_adc\_rc\_cali | 22:22 | RW | rc calibration setting from register. | 1'h0 |
| aon\_rval\_adc\_quar\_cov | 21:16 | RW | quar calibration value from register. | 6'h0 |
| aon\_rctrl\_adc\_rc\_cali | 15:15 | RW | rc calibration setting from register. | 1'h0 |
| aon\_rval\_adc\_quar\_cov\_en | 14:14 | RW | ADC vref setting time calibration enable. | 1'h0 |
| aon\_aud\_adc\_rc\_ctrl | 13:9 | RW | [0]: SAR ADC Comparator bias current control  0:20uA，1:10uA [1]: Integrator input resistance, feedback capacitor control.   1: 12MHz FS   0: 04MHz FS (4MHz/16K default) others: invalid (only bit[0] bit[1] are valid) | 5'h0 |
| aon\_aud\_adc\_idac\_ctrl | 8:7 | RW | Feedback IDAC Control 00: x-5% 01: 7.5uA (expected default)---Full scale 1.5Vrms 10: x+6% 11: x+13% | 2'h0 |
| aon\_aud\_adc\_ib\_ctrl | 6:3 | RW | ADC amp ibias control 0000: 1.5uA 0001: 2uA (expected default) 0010: 2.5uA 0011: 3uA bit[2]=1: amp1 low power mode, bit[3]=1: amp2 low power mode | 4'h0 |
| aon\_aud\_adc\_lp\_rst | 2:2 | RW | ADC Loop path reset. 0: no reset 1: Loop reset | 1'h1 |
| aon\_adc\_en\_l | 1:1 | RW | ADC Left enable. 0: Power Down; 1: Enabled | 1'h0 |
| aon\_adc\_en\_r | 0:0 | RW | ADC Right enable. 0: Power Down; 1: Enabled | 1'h0 |

### AUD\_R12\_ADC\_CTRL7

Offset: 030

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:26 | RW | not used | 6'h0 |
| aon\_rset\_aud\_idac\_bias | 25:25 | RW | register control enable. 1: enable 0: disable | 1'h0 |
| aon\_rval\_aud\_idac\_bias | 24:24 | RW | register value for aud\_idac\_bias. | 1'h0 |
| aon\_aud\_adc\_idac\_trim | 23:21 | RW |  | 3'h0 |
| aon\_aud\_adc\_vref\_ctrl | 20:19 | RW | Control vref voltage. Control vref power consumption 00: default VREF\_CTRL0>:1—— ADC VREF 40uA；0—— ADC VREF 20uA； VREF\_CTRL1>:1—— ADC VCM BIAS 30uA； 0—— ADC VCM BIAS 10uA | 2'h0 |
| aon\_aud\_adc\_vref\_mode | 18:18 | RW | Control generate vref mode. | 1'h0 |
| aon\_aud\_adc\_vref\_en | 17:17 | RW | VREF\_EN 1: enable 0: disable | 1'h0 |
| aon\_aud\_adc\_idac\_offset | 16:15 | RW | feed back idac DC offset.  00: 0uA(default); 01: LSB/16;  10: LSB/8; 11: LSB/4 | 2'h0 |
| aon\_aud\_adc\_atb | 14:12 | RW | ADC atb control 000:ATB output H-Z; 001: test ADC0 IDAC\_NMOS; 010: test ADC0 IDAC\_PMOS; 011: test ADC1 IDAC\_NMOS; 100: test ADC1 IDAC\_PMOS 000:default | 3'h0 |
| aon\_lpga\_zcen\_p | 11:11 | RW | Zero Crossing enable for ADC Left PGA. 1: enable 0: disable | 1'h1 |
| aon\_aud\_lpga\_lpr | 10:10 | RW | ADC Left PGA power mode,  1: Low Power Mode 0: normal mode. | 1'h0 |
| aon\_aud\_lpga\_en\_single | 9:9 | RW | ADC Left PGA input mode. 0: differetial 1: Single | 1'h1 |
| aon\_aud\_lpga\_mute | 8:8 | RW | ADC Left PGA Mute,  1: mute 0: working | 1'h1 |
| aon\_lpga\_en | 7:7 | RW | ADC Left Input PGA enable;  0= Power Down; 1= Enable; | 1'h0 |
| aon\_rpga\_zcen\_p | 6:6 | RW | Zero Crossing enable for ADC Right PGA 1: enable 0: disable | 1'h1 |
| aon\_aud\_rpga\_lpr | 5:5 | RW | ADC Right PGA power mode,  1: Low Power Mode 0: normal mode | 1'h0 |
| aon\_aud\_rpga\_en\_single | 4:4 | RW | ADC Right PGA input mode,  0: differetial 1: Single | 1'h1 |
| aon\_aud\_rpga\_mute | 3:3 | RW | ADC Right PGA Mute,  1: mute 0: working | 1'h1 |
| aon\_rpga\_en | 2:2 | RW | ADC Right Input PGA enable;  0: Power Down; 1: Enable; | 1'h0 |
| aon\_aud\_pgabuf\_lp | 1:1 | RW | ADC LPGA/RPGA Buffer low power mode 1: low power mode 0: normal mode | 1'h0 |
| aon\_aud\_pgabufen | 0:0 | RW | LPGA/RPGA Buffer Enable 1: enable 0: disable | 1'h0 |

### AUD\_R25\_STATUS0

Offset: 064

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:17 | RO | not used | 15'h0 |
| aon\_adc\_cap\_cali | 16:12 | RO | Trim RC product 10000:default | 5'h10 |
| aon\_adc\_quar\_cov | 11:6 | RO | QUAT\_COV3:0>= delay trim code QUAT\_COV5:4>= delay\_v2 trim code 0000:default | 6'h20 |
| aon\_adc\_cap\_cali\_fail | 5:5 | RO |  | 1'h0 |
| aon\_adc\_cap\_cali\_done | 4:4 | RO |  | 1'h0 |
| aon\_adc\_cap\_cov\_cali\_flag\_inst | 3:3 | RO |  | 1'h0 |
| aon\_adc\_rc\_cali\_fail | 2:2 | RO |  | 1'h0 |
| aon\_adc\_rc\_cali\_done | 1:1 | RO |  | 1'h0 |
| aon\_adc\_quar\_cov\_cali\_flag\_inst | 0:0 | RO |  | 1'h0 |

# FLASH\_DL

## Introduction

不开放用户

## Main Features

## Function Description

## Basic Block Diagram

## FLASH\_DL Register

### CTRL

Offset: 000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:17 | RO | unused | 15'h0 |
| clk\_inv\_en | 16:16 | RW | clock inversion enable. High active | 1'h0 |
| cycle\_cal\_en | 15:15 | RW | enable clk cycle time calculation. The result is stored in dly\_lock\_value register. 1: enable  0: disable | 1'h0 |
| dly\_bypass | 14:14 | RW | Controls the bypass mode of the master dly. If this bit is high, rdlvl\_delay value would directly mean the number of delay cells used for the input clock | 1'h1 |
| phase\_detect\_sel | 13:11 | RW | Phase detect selector to choose the number of delay cells to be inserted for the delay line in the edge detection circuit of phase detector. | 3'h0 |
| dly\_lock\_num | 10:8 | RW | Waiting cycles before asserting lock done | 3'h0 |
| dly\_start\_point | 7:0 | RW | Initial delay value for the clk cycle time calculation | 8'h4 |

### DELAY

Offset: 004

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:8 | RO | unused | 24'h0 |
| clk\_delay | 7:0 | RW | Delays of the input clock | 8'h0 |

### OBS\_REG0

Offset: 008

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| lock\_inc | 31:24 | RO | inc value in the past 8 cycles(for debug only) | 8'h0 |
| lock\_dec | 23:16 | RO | dec value in the past 8 cycles(for debug only) | 8'h0 |
| dly\_lock\_value | 15:8 | RO | dly lock value | 8'h0 |
| dly\_unlock\_cnt | 7:3 | RO | counter clock gate enable | 5'h0 |
| half\_clock\_mode | 2:2 | RO | The delay cells are only enough to lock half cycle. | 1'h0 |
| force\_lock\_overflow | 1:1 | RO | All delay cells have been used but still fails to calculate the whole clock cycle | 1'h0 |
| dly\_lock | 0:0 | RO | dly has locked for calculating one or half cycle of input clock | 1'h0 |

### OBS\_REG1

Offset: 00C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:8 | RO | unused | 24'h0 |
| decoder\_out\_rd | 7:0 | RO | Number of delay cells used to generate the delay for input clock | 8'h0 |

### LOCK\_DONE

Offset: 010

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO | unused | 31'h0 |
| lock\_done | 0:0 | RO | High active signal to mean that dly has locked | 1'h0 |

### RSTN

Offset: 014

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO | unused | 31'h0 |
| dly\_rst\_n | 0:0 | WO | 0: reset dly 1: ignore Step1:write 0 to reset dly Step2:check if dly is lock\_done Step3:set clk\_delay Step4:dly resync | 1'h1 |

### RESYNC

Offset: 018

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Bit | Type | Description | Reset |
| Reserved | 31:1 | RO | not used | 31'h0 |
| dly\_resync | 0:0 | WO | Write 1: resync high active pulse used to resync the clk\_delay to delay line, need to be issued by software after new value of clk\_delay has been updated | 1'h0 |