# MT2511 Technical Reference Manual

Version: 1.0

Release date: 5 May 2017

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s). Airoha cannot grant you permission for any material that is owned by third parties. You may only use or reproduce this document if you have agreed to and been bound by the applicable license agreement with Airoha (“License Agreement”) and been granted explicit permission within the License Agreement (“Permitted User”). If you are not a Permitted User, please cease any access or use of this document immediately. Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited. THIS DOCUMENT IS PROVIDED ON AN “AS-IS” BASIS ONLY. AIROHA EXPRESSLY DISCLAIMS ANY AND ALL WARRANTIES OF ANY KIND AND SHALL IN NO EVENT BE LIABLE FOR ANY CLAIMS RELATING TO OR ARISING OUT OF THIS DOCUMENT OR ANY USE OR INABILITY TO USE THEREOF. Specifications contained herein are subject to change without notice.

## Document Revision History

| Revision | Date | Description |
|---|---|---|
| 0.1 | 19 April 2015 | Initial draft |
| 0.2 | 28 April 2016 | Editing, adding addition chapters |
| 0.3 | 2 May 2016 | Editing for coda format |
| 0.31 | 21 June 2016 | MP version release. |
| 0.35 | 3 September 2016 | Refine programming guide. |
| 1.0 | 5 May 2017 | Official version. |

# Table of contents

**1 MCU Interface Selection ........ 7**

1.1 Interface Selection between I2C and SPI ........ 7

1.2 Access Speed of SPI/I2C ........ 7

**2 Interrupts ........ 8**

2.1 General Description ........ 8

2.2 Programming Guide ........ 8

2.2.1 Interrupt Enable ........ 8

2.2.2 Interrupt Status checking ........ 8

2.2.3 Interrupt Clear ........ 9

2.3 Interrupt Register Definitions ........ 9

**3 Serial Peripheral Interface Slave Controller ........ 13**

3.1 General Description ........ 13

3.1.1 Features ........ 14

**4 Inter-Integrated Circuit Controller ........ 17**

4.1 General Description ........ 17

4.1.1 Features ........ 17

4.1.2 Transfer Format Support ........ 17

4.1.3 Programming Guide ........ 19

**5 Timing Control Module ........ 20**

5.1 General Operation ........ 20

5.2 Dynamic Power-Down Mode ........ 20

5.3 Timing Module Operation ........ 22

5.4 Timing Module Register Definitions ........ 25

5.5 Programming Guide ........ 35

**6 PPG Channel ........ 36**

6.1 General Description ........ 36

6.1.1 Transmitter Front-End ........ 37

6.1.2 Receiver Front-End ........ 38

6.1.3 PPG AVERAGE Function Introduce ........ 39

6.1.4 PPG Data Format ........ 40

6.2 PPG Register Definitions ........ 41

6.3 Programming Guide ........ 52

**7 EKG Channel ........ 54**

7.1 General Description ........ 54

7.1.1 EKG Data Format ........ 55

7.2 EKG Channel Register Definitions........ 55

7.3 Programming Guide ........ 61

**8 Heartbeat Interval Estimation ........ 62**

8.1 General Description ........ 62

8.2 Heartbeat Interval Estimation Register Definitions ........ 63

8.3 Programming Guide ........ 67

# Lists of tables and figures

Table 1-1. Register Control Interface Selection ........ 7
Table 2-1. MT2511 Interrupts ........ 8
Table 3-1. SPI Slave Controller Interface ........ 13
Table 3-2. SPI Slave Command Description ........ 15
Table 3-3. SPI Slave Status Description (Use RS Command to Poll SPI Slave Status) ........ 15
Table 5-1-1 Example timing module register setting ........ 23
Figure 3-1. Pin Connection between SPI Master and SPI Slave ........ 13
Figure 3-2. SPI Transmission Formats ........ 13
Figure 3-3. SPI Slave Controller Commands Waveform ........ 14
Figure 3-4. SPI Slave Control Flow Diagram ........ 14
Figure 3-5. Config Read/Write (CR/CW) Command Format ........ 16
Figure 5-1. Timing Module General Operation ........ 20
Figure 5-2. Dynamic Power Down Mode ........ 21
Figure 5-3. Timing Module Operation Example ........ 22
Figure 6-1. PPG Channel Analog Part Block Diagram ........ 36
Figure 6-2. PPG Channel Digital Part Block Diagram ........ 37
Figure 6-3. Push-Pull Mode and H-Bridge Mode ........ 38
Figure 6-4. Receiver Front-End ........ 39
Figure 6-5. ADC Data with Averaging Enabled ........ 40
Figure 7-1. EKG Channel Analog Part Block Diagram ........ 54
Figure 7-2. EKG Channel Digital Part Block Diagram ........ 54
Figure 8-1. Heartbeat Detection Flow ........ 62
Figure 8-2. The Built-in Heartbeat Interval Detector Block Diagram ........ 62

# Documentation General Conventions

**Abbreviations for Control Moudules**

| Abbreviation | Full name |
|---|---|
| SPI master | Serial peripheral interface master controller |
| SPI slave | Serial peripheral interface slave controller |
| I2C | Inter-integrated circuit interface |
| TCM | Timing Control Moudle |
| PPG | Photoplethysmography |
| EKG | Electrocardiography |
| BI | Beat Interval |
| TCRL | Timing control logic |

**Abbreviations for Registers**

| Abbreviation | Full name |
|---|---|
| RW | Read and write |
| RO | Read only |
| WO | Write only |
| RC | Read 1 to clear |
| WC | Write 1 to clear |
| RWC | Read or write 1 to clear |
| FM | Frequency measurement |
| FRC | Free running counter |

# 1 MCU Interface Selection

## 1.1 Interface Selection between I2C and SPI

***Table 1-1*** *summarizes the MCU register access interface selection.*

***Table 1-1. Register Control Interface Selection***

| Interface | I2C_SEL | SPI_MOSI | Register Group | Register Group |
|---|---|---|---|---|
| | | | top_reg base | top_reg_1 base |
| I2C | Tied to logic high | Tied to logic high | I2C slave addr. 0x37 | I2C slave addr. 0x27 |
| | | Tied to logic low | I2C slave addr. 0x33 | I2C slave addr. 0x23 |
| SPI | Tied to logic low | as SPI serial in master out | SPI addr. 0x33 | SPI addr. 0x23 |

## 1.2 Access Speed of SPI/I2C

The maximum SPI Clock is designed to operate at a frequency of 2M Hz.

The maximum I2C Clock depends on board-level parameters such as trace latency. Typically, the maximum speed is 400K Hz.

Actual operating clock frequency may depending on board level condition, like trace length and total loading,

Either MT2511 I2C or MT2511 SPI does not support burst access.

# 2 Interrupts

## 2.1 General Description

To facilitate MCU programming and sensor data flow control, MT2511 supports various interrupts for each function. The table here lists supported interrupts and their usage.

*Table 2-1. MT2511 Interrupts*

| Interrupt | Description |
|---|---|
| MEM_EKG | When number of data in EKG memory is greater than pre-configured number AFE_MEM_CON3::rg_irq_th. |
| MEM_EKG_WFULL | When putting data to EKG SRAM but it’s full. |
| MEM_EKG_REMPTY | When trying to read data from EKG SRAM but it’s empty. |
| MEM_PPG1 | When number of data inPPG1 memory is greater than pre-configured number AFE_MEM_CON7::rg_irq_th. |
| MEM_PPG1_WFULL | When putting data to PPG1 SRAM but it’s full. |
| MEM_PPG1_REMPTY | When trying to read data from PPG1 SRAM but it’s empty. |
| MEM_PPG2 | When number of data in PPG2 memory is greater than pre-configured number AFE_MEM_CON11::rg_irq_th. |
| MEM_PPG2_WFULL | When putting data to PPG2 SRAM but it’s full. |
| MEM_PPG2_REMPTY | When trying to read data from PPG2 SRAM but it’s empty. |
| MEM_BISI | When number of data in BISI memory is greater than pre-configured number AFE_MEM_CON15::rg_irq_th. |
| MEM_BISI_WFULL | When putting data to BISI SRAM but it’s full. |
| MEM_BISI_REMPTY | When trying to read data from BISI SRAM but it’s empty. |

## 2.2 Programming Guide

Taking MEM_EKG_REMPTY as example, the following sections provide programming guide to enable, check and disable interrupt.

### 2.2.1 Interrupt Enable

To enable some interrupt, set RG_INT_EN_* to 1. For MEM_EKG_REMPY interrupt, set INT_CON[3] = 1.

### 2.2.2 Interrupt Status checking

On receiving MT2511 interrupt, host MCU can identify the exact cause of interrupt by reading the INT_STATUS register. When MEM_EKG_REMPTY interrupt is received, INT_STATUS[3] is asserted.

### 2.2.3 Interrupt Clear

To clear a specific interrupt, RG_INT_EN_* control bit should be set to 0 and RG_INT_CLR* clear bit should be set to 1 then set to 0.

For MEM_EKG_REMPTY interrupt, INT_CON[3] should be set to 0 and INT_STATUS[19] should be set to 0 then set to 0.

## 2.3 Interrupt Register Definitions

**Base name: top_reg**

| Address | Name | Width | Register Function |
|---|---|---|---|
| 0000004C | **INT_CON** | 32 | **Interrupt control register** |
| 00000054 | **INT_STATUS** | 32 | **Interrupt status register** |

**0000004C INT_CON Interrupt control register 00000000**

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Name** | Reserved | | | | | | | | Reserved | | | | | | | |
| **Type** | RW | | | | | | | | RW | | | | | | | |
| **Reset** | 0 | 0 | 0 | 0 | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | RG_INT_POLARITY | | | RG_INT_EN_MEM_BISI_REMPTY | RG_INT_EN_MEM_BISI_WFULL | RG_INT_EN_MEM_BISI | RG_INT_EN_MEM_PPG2_REMPTY | RG_INT_EN_MEM_PPG2_WFULL | RG_INT_EN_MEM_PPG2 | RG_INT_EN_MEM_PPG1_REMPTY | RG_INT_EN_MEM_PPG1_WFULL | RG_INT_EN_MEM_PPG1 | RG_INT_EN_MEM_EKG_REMPTY | RG_INT_EN_MEM_EKG_WFULL | RG_INT_EN_MEM_EKG | Reserved |
| **Type** | RW | | | RW | RW | RW | RW | RW | RW | RW | RW | RW | RW | RW | RW | RW |
| **Reset** | 0 | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit(s) | Name | Description |
|---|---|---|
| 31:28 | Reserved | |
| 23:16 | Reserved | |
| 15 | RG_INT_POLARITY | **Interrupt level inverting**<br>0: do not invert interrupt source<br>1: invert interrupt source |
| 12 | RG_INT_EN_MEM_BISI_REMPTY | **Issue interrupt when number of data when trying** |

| Bit(s) | Name | Description |
|---|---|---|
| | | **to read data from empty BI memory**<br>0: do not issue interrupt<br>1: issue interrupt |
| 11 | RG_INT_EN_MEM_BISI_WFULL | **Issue interrupt when number of data in BI memory is full**<br>0: do not issue interrupt<br>1: issue interrupt |
| 10 | RG_INT_EN_MEM_BISI | **Issue interrupt when number of data in BI memory is greater than pre-configured number AFE_MEM_CON15::rg_irq_th**<br>0: do not issue interrupt<br>1: issue interrupt |
| 9 | RG_INT_EN_MEM_PPG2_REMPTY | **Issue interrupt when number of data when trying to read data from empty PPG2 memory**<br>0: do not issue interrupt<br>1: issue interrupt |
| 8 | RG_INT_EN_MEM_PPG2_WFULL | **Issue interrupt when PPG2 memory is full**<br>0: do not issue interrupt<br>1: issue interrupt |
| 7 | RG_INT_EN_MEM_PPG2 | **Issue interrupt when number of data in PPG2 memory is greater than pre-configured number AFE_MEM_CON11::rg_irq_th**<br>0: do not issue interrupt<br>1: issue interrupt |
| 6 | RG_INT_EN_MEM_PPG1_REMPTY | **Issue interrupt when number of data when trying to read data from empty PPG1 memory**<br>0: do not issue interrupt<br>1: issue interrupt |
| 5 | RG_INT_EN_MEM_PPG1_WFULL | **Issue interrupt when PPG1 memory is full**<br>0: do not issue interrupt<br>1: issue interrupt |
| 4 | RG_INT_EN_MEM_PPG1 | **Issue interrupt when number of data in PPG1 memory is greater than pre-configured number AFE_MEM_CON7::rg_irq_th**<br>0: do not issue interrupt<br>1: issue interrupt |
| 3 | RG_INT_EN_MEM_EKG_REMPTY | **Issue interrupt when number of data when trying to read data from empty EKG memory**<br>0: do not issue interrupt<br>1: issue interrupt |
| 2 | RG_INT_EN_MEM_EKG_WFULL | **Issue interrupt when EKG memory is full**<br>0: do not issue interrupt<br>1: issue interrupt |
| 1 | RG_INT_EN_MEM_EKG | **Issue interrupt when number of data in EKG memory is greater than pre-configured number AFE_MEM_CON3::rg_irq_th**<br>0: do not issue interrupt<br>1: issue interrupt |

**00000054** **INT_STATUS** **Interrupt status register** **FFFFFFFF**

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | | | | RG_INT_CLR_MEM_BISI_REMPTY | RG_INT_CLR_MEM_BISI_WFULL | RG_INT_CLR_MEM_BISI | RG_INT_CLR_MEM_PPG2_REMPTY | RG_INT_CLR_MEM_PPG2_WFULL | RG_INT_CLR_MEM_PPG2 | RG_INT_CLR_MEM_PPG1_REMPTY | RG_INT_CLR_MEM_PPG1_WFULL | RG_INT_CLR_MEM_PPG1 | RG_INT_CLR_MEM_EKG_REMPTY | RG_INT_CLR_MEM_EKG_WFULL | RG_INT_CLR_MEM_EKG | Reserved |
| Type | | | | RW | RW | RW | RW | RW | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Name | | | | RG_INT_STATUS_MEM_BISI_REMPTY | RG_INT_STATUS_MEM_BISI_WFULL | RG_INT_STATUS_MEM_BISI | RG_INT_STATUS_MEM_PPG2_REMPTY | RG_INT_STATUS_MEM_PPG2_WFULL | RG_INT_STATUS_MEM_PPG2 | RG_INT_STATUS_MEM_PPG1_REMPTY | RG_INT_STATUS_MEM_PPG1_WFULL | RG_INT_STATUS_MEM_PPG1 | RG_INT_STATUS_MEM_EKG_REMPTY | RG_INT_STATUS_MEM_EKG_WFULL | RG_INT_STATUS_MEM_EKG | Reserved |
| Type | | | | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO | RO |
| Reset | | | | x | x | x | x | x | x | X | X | x | x | x | x | x |

| Bit(s) | Name | Description |
|---|---|---|
| 28 | RG_INT_CLR_MEM_BISI_REMPTY | **To clear interrupt source MEM_BISI_REMPTY** |
| 27 | RG_INT_CLR_MEM_BISI_WFULL | **To clear interrupt source MEM_BISI_WFULL** |
| 26 | RG_INT_CLR_MEM_BISI | **To clear interrupt source MEM_BISI** |
| 25 | RG_INT_CLR_MEM_PPG2_REMPTY | **To clear interrupt source MEM_PPG2_REMPTY** |
| 24 | RG_INT_CLR_MEM_PPG2_WFULL | **To clear interrupt source MEM_PPG2_WFULL** |
| 23 | RG_INT_CLR_MEM_PPG2 | **To clear interrupt source MEM_PPG2** |
| 22 | RG_INT_CLR_MEM_PPG1_REMPTY | **To clear interrupt source** |

| Bit(s) | Name | Description |
|---|---|---|
| | | **MEM_PPG1_REMPTY** |
| 21 | RG_INT_CLR_MEM_PPG1_WFULL | **To clear interrupt source MEM_PPG1_WFULL** |
| 20 | RG_INT_CLR_MEM_PPG1 | **To clear interrupt source MEM_PPG1** |
| 19 | RG_INT_CLR_MEM_EKG_REMPTY | **To clear interrupt source MEM_EKG_REMPTY** |
| 18 | RG_INT_CLR_MEM_EKG_WFULL | **To clear interrupt source MEM_EKG_WFULL** |
| 17 | RG_INT_CLR_MEM_EKG | **To clear interrupt source MEM_EKG** |
| 12 | RG_INT_STATUS_MEM_BISI_REMPTY | **Status of interrupt MEM_BISI_REMPTY** |
| 11 | RG_INT_STATUS_MEM_BISI_WFULL | **Status of interrupt MEM_BISI_WFULL** |
| 10 | RG_INT_STATUS_MEM_BISI | **Status of interrupt MEM_BISI** |
| 9 | RG_INT_STATUS_MEM_PPG2_REMPTY | **Status of interrupt MEM_PPG2_REMPTY** |
| 8 | RG_INT_STATUS_MEM_PPG2_WFULL | **Status of interrupt MEM_PPG2_WFULL** |
| 7 | RG_INT_STATUS_MEM_PPG2 | **Status of interrupt MEM_PPG2** |
| 6 | RG_INT_STATUS_MEM_PPG1_REMPTY | **Status of interrupt MEM_PPG1_REMPTY** |
| 5 | RG_INT_STATUS_MEM_PPG1_WFULL | **Status of interrupt MEM_PPG1_WFULL** |
| 4 | RG_INT_STATUS_MEM_PPG1 | **Status of interrupt MEM_PPG1** |
| 3 | RG_INT_STATUS_MEM_EKG_REMPTY | **Status of interrupt MEM_EKG_REMPTY** |
| 2 | RG_INT_STATUS_MEM_EKG_WFULL | **Status of interrupt MEM_EKG_WFULL** |
| 1 | RG_INT_STATUS_MEM_EKG | **Status of interrupt MEM_EKG** |

# 3 Serial Peripheral Interface Slave Controller

## 3.1 General Description

The SPI (Serial Peripheral Interface) is a bit-serial, four-pin transmission protocol. Figure 3-1 is an example of the connection between the SPI master and SPI slave. The SPI slave controller, configured by the SPI master transmit data, is a slave responsible of data transmission with the master.

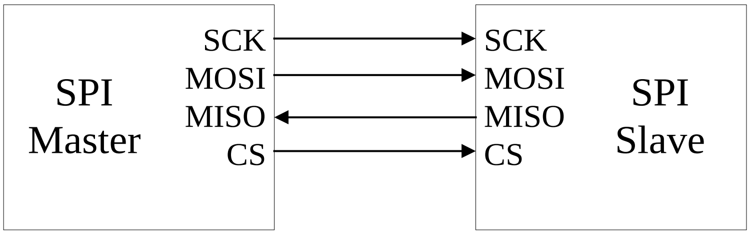


*Figure 3-1. Pin Connection between SPI Master and SPI Slave*

Figure 3-2 shows the waveform during the SPI transmission. The low active CS_N determines the start point and end point of one transaction. The CS_N setup time, hold time and idle time are also depicted.

CPOL defines the clock polarity in the transmission. CPHA defines the legal timing to sample MOSI and MISO. MT2511 only support CPOL = 0 and CPHA = 0 mode. The setup/ hold/ idle time should be greater than 1 us.

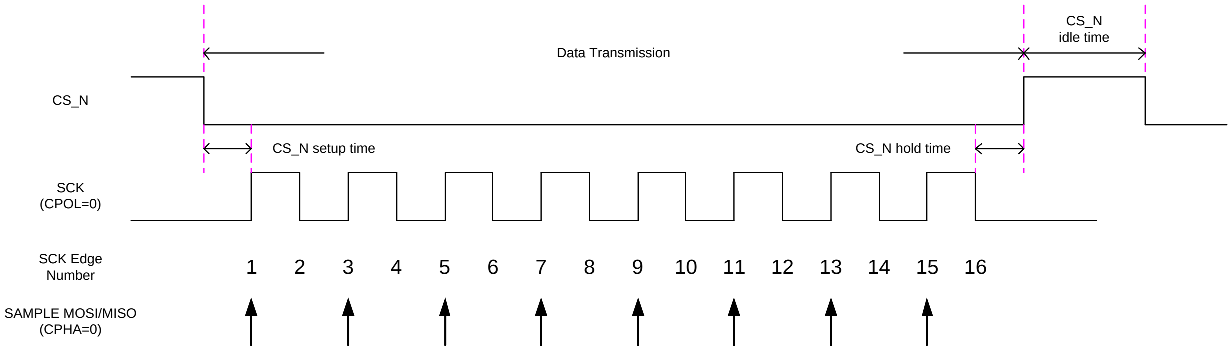


*Figure 3-2. SPI Transmission Formats*

*Table 3-1. SPI Slave Controller Interface*

| Signal name | Type | Description |
|---|---|---|
| CS | I | Low active chip selection signal |
| SCK | I | The (bit) serial clock (Max SCK clock rate is 2MHz.) |
| MOSI | I | Data signal from master output to slave input |

| Signal name | Type | Description |
|---|---|---|
| MISO | O | Data signal from slave output to master input |

### 3.1.1 Features

The SPI slave controller has eight commands that can be configured by SPI master transmit data. The commands include “configure-write”, “configure-read”, “write-data”, “read-data”, “write-status” and “read-status”. The command waveform is shown here.

Note: The SPI slave controller data format is LSBF ( least significant bit first ).

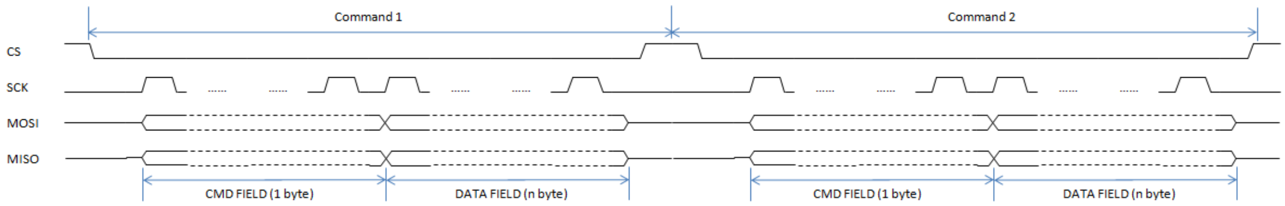


*Figure 3-3. SPI Slave Controller Commands Waveform*

**SPI slave control flow**

The SPI slave control flow is shown here.

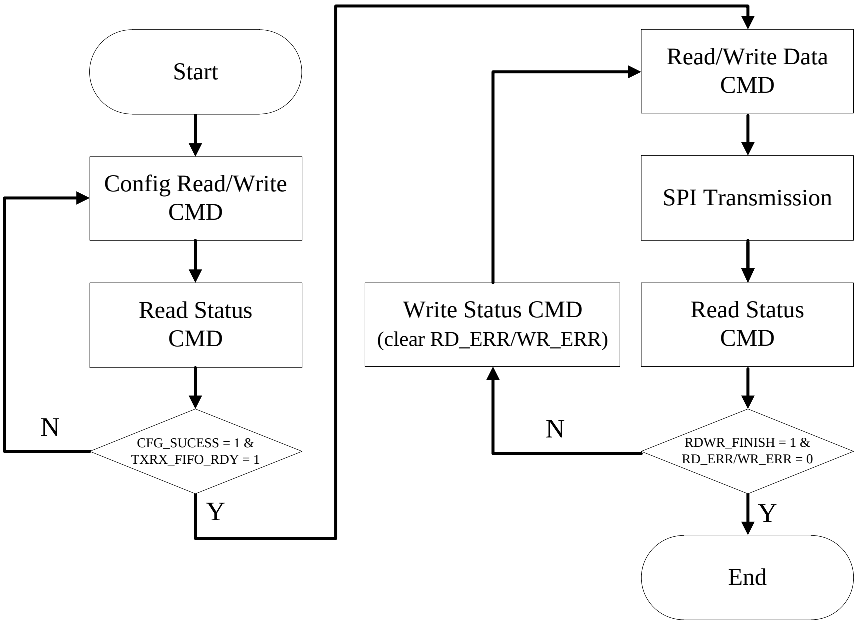

Start
Config Read/Write CMD
Read Status CMD
CFG_SUCESS = 1 & TXRX_FIFO_RDY = 1
N
Y
Read/Write Data CMD
SPI Transmission
Read Status CMD
RDWR_FINISH = 1 & RD_ERR/WR_ERR = 0
N
Y
Write Status CMD (clear RD_ERR/WR_ERR)
End

*Figure 3-4. SPI Slave Control Flow Diagram*

First, SPI slave controller transmits “config-read/write” command to configure the transfer data length and read/write address of the memory. After the SPI slave is configured, it can send/receive data package with SPI master by “read/write-data” command. In each state, SPI master transmits “read-status” command to poll SPI slave situation. If SPI master detects error flag bit of state, it should send “write-status” command to clear the bit and poll this bit until it turns low. Detailed descriptions of SPI slave command are shown in Table 3-2 and the SPI slave status in Table 3-3 below.

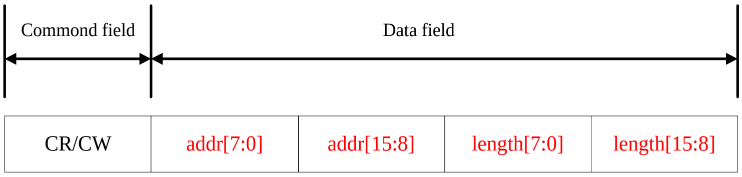
*Table 3-2. SPI Slave Command Description*

| Cmd Field [7:0] | CMD Default Code | Data Field | Usage |
|---|---|---|---|
| Read Data (RD) | 0x81 | N bytes. Burst data payload | Master read data |
| Write Data (WD) | 0x06 | N bytes. Burst data payload | Master write data |
| Read Status (RS) | 0x0A | 1 byte | Master reads slave status register |
| Write Status (WS) | 0x08 | 1 byte | Master writes slave status register to clear error bit (i.e. write 1 to clear). |
| Config Read (CR) | 0x02 | 2 bytes addr,<br>2 bytes data length | Master configure slave to start read data. |
| Config Write (CW) | 0x04 | 2 bytes addr,<br>2 bytes data length | Master configures slave to start write data. |

*Table 3-3. SPI Slave Status Description (Use RS Command to Poll SPI Slave Status)*

| Function | Bit | Usage |
|---|---|---|
| SR_CFG_SUCCESS | 1 | Master checks this bit to know if CW/CR command is successful. |
| SR_TXRX_FIFO_RDY | 2 | If master configures read/write, when slave is ready to send/receive data, the master can send RD/WD command. Clean: After SPI slave receives CR/CW command. |
| SR_RD_ERR | 3 | After a RD command, master can read this bit to know if there is error in the read transfer. If there is error, master should send WS command to clear this bit and poll this bit until this bit turns 0. |
| SR_WR_ERR | 4 | After a WD command, master can read this bit to know if there is error in the write transfer. If there is error, master should send WS command to clear this bit and poll this bit until this bit turns 0. |
| SR_RDWR_FINISH | 5 | After RD/WD transaction, master can poll this bit to know if the read/write transfer is finished. Clean: After SPI slave receives CR/CW command. |
| SR_TIMOUT_ERR | 6 | SPI slave does not receive or send data over 31.75 us, the flag of timeout will rise. If there is error, master should send WS command to clear this bit and poll this bit until this bit turns 0. |
| SR_CMD_ERR | 7 | If master sends an error CMD at the first byte, master can know the error status |

| Function | Bit | Usage |
|---|---|---|
| | | through the received data. Clean: After SPI slave receives  correct command. |



| | | | | |
|---|---|---|---|---|
| CR/CW | addr[7:0] | addr[15:8] | length[7:0] | length[15:8] |

*Figure 3-5. Config Read/Write (CR/CW) Command Format*

# 4 Inter-Integrated Circuit Controller

## 4.1 General Description

Inter-Integrated Circuit (I2C) is a two-wire serial interface. The two signals are SCL and SDA. SCL is a clock signal driven by the master. SDA is a bi-directional data signal that can be driven by either the master or the slave. This generic controller supports the slave role and conforms to the I2C specification.

### 4.1.1 Features

- I2C compliant slave mode operation
- 7-bit addressing
- START/STOP/REPEATED START condition
- Manual transfer mode
- Multi-transfer per transaction
- Active drive/wired-and I/O configuration

### 4.1.2 Transfer Format Support

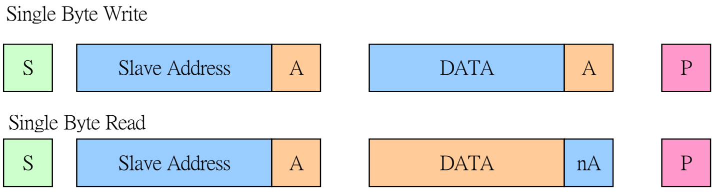
Wording convention note:

- Transfer = Anything encapsulated within a Start and Stop or Repeated Start.
- Transfer length = Number of bytes within the transfer
- Transaction = This is the top unit. Everything combined equals one transaction.
- Transaction length = Number of transfers to be conducted.

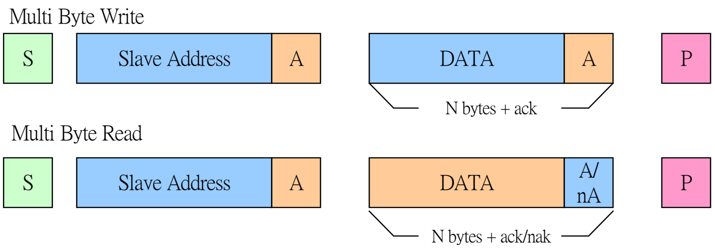
🟦 Master to slave dir

🟧 Slave to master dir

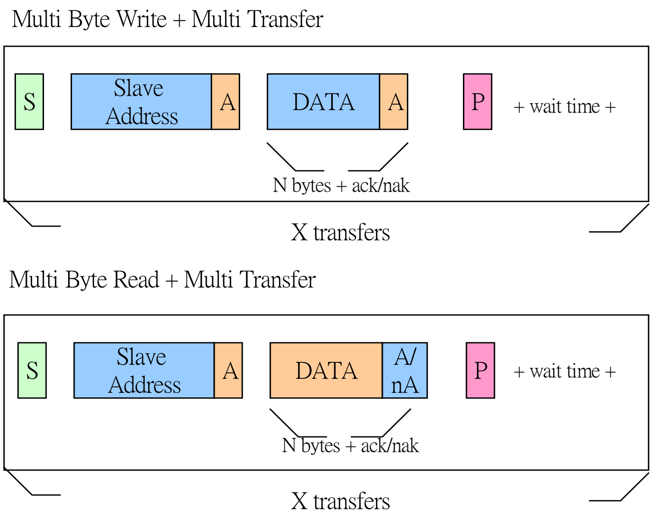
**Single-byte access**



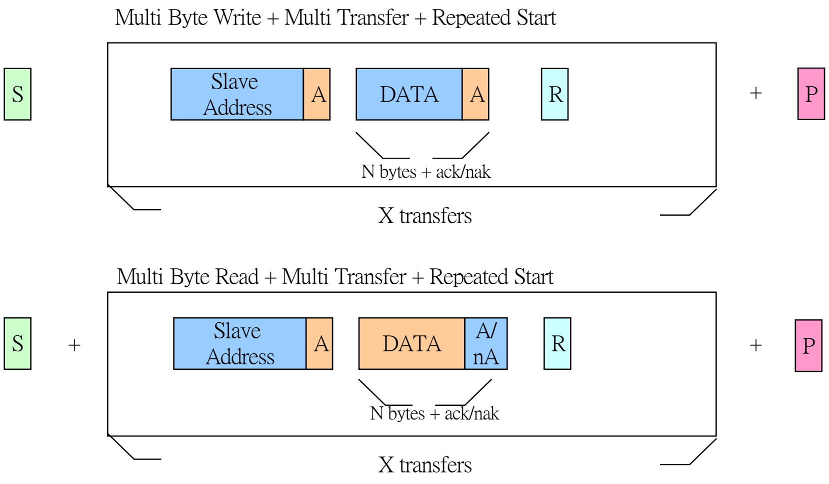
### Multi-byte access



### Multi-byte transfer + multi-transfer (same direction)



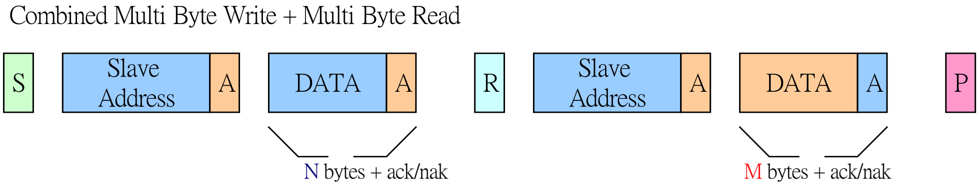
### Multi-byte transfer + multi-transfer w RS (same direction)



### Combined write/read with Repeated Start (direction change)

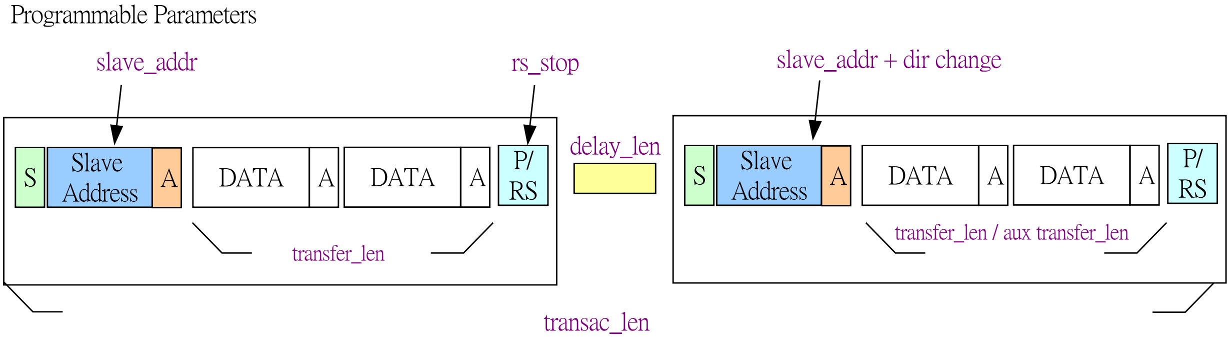
Note:

- Only supports write and then read sequence. Read and then write is not supported.
- In this format, transaction is 2



### 4.1.3 Programming Guide

Common transfer programmable parameters



# 5 Timing Control Module

## 5.1 General Operation

- LED and ambient (AMB) data are stored at the low pass filter (LPF) separately and then digitized by 24-bit incremental ADC.
- Flexible timing module: PRF: 64~4096 SPS, duty: 1~25%

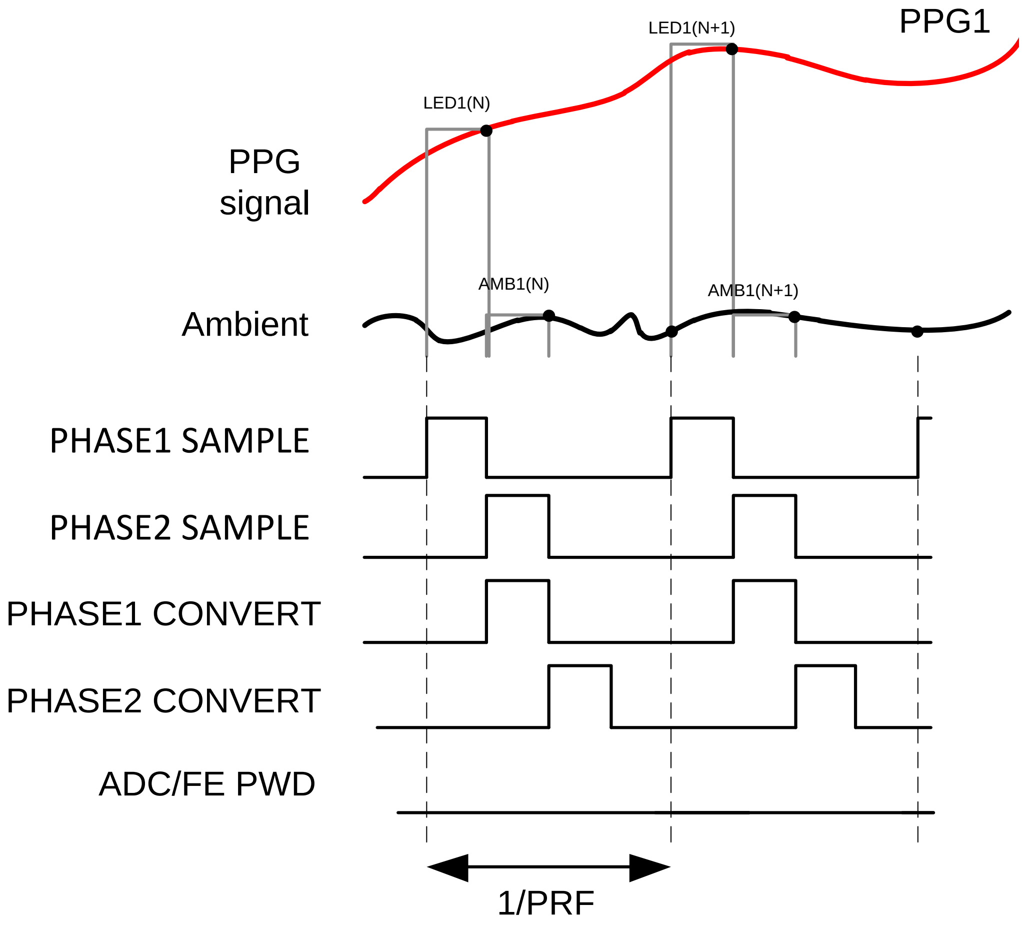


*Figure 5-1. Timing Module General Operation*

## 5.2 Dynamic Power-Down Mode

- The flexible timing control enables the users to control the device timing for different applications and to power down the device for power saving.

- $I_{RX+TX}$=480µA → 100µA at PRF=125 and duty=1.5%

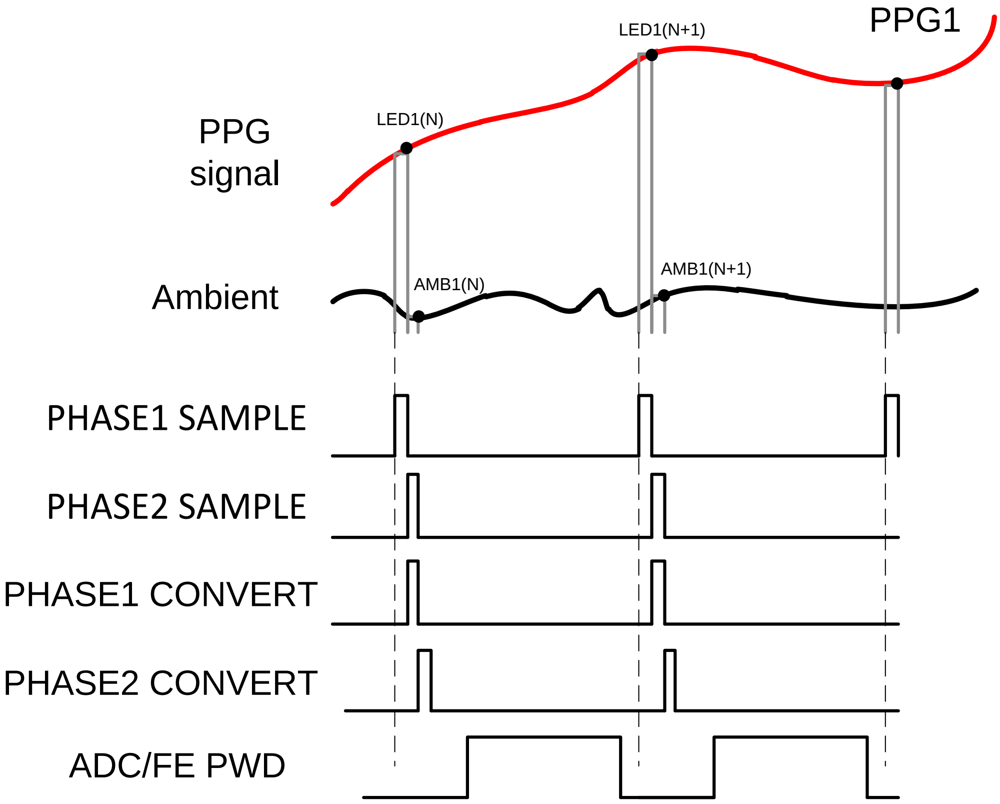


***Figure 5-2. Dynamic Power Down Mode***

## 5.3 Timing Module Operation

The rising and falling edge position of below 14 signals can be set separately. LED1 and LED2 can be set at the same phase (phase1) or the different phase (phase1 & phase3). Figure 5-3 shows some timing operation example, and corresponding register settings are listed in Table 5-1-1 Example timing module register setting .

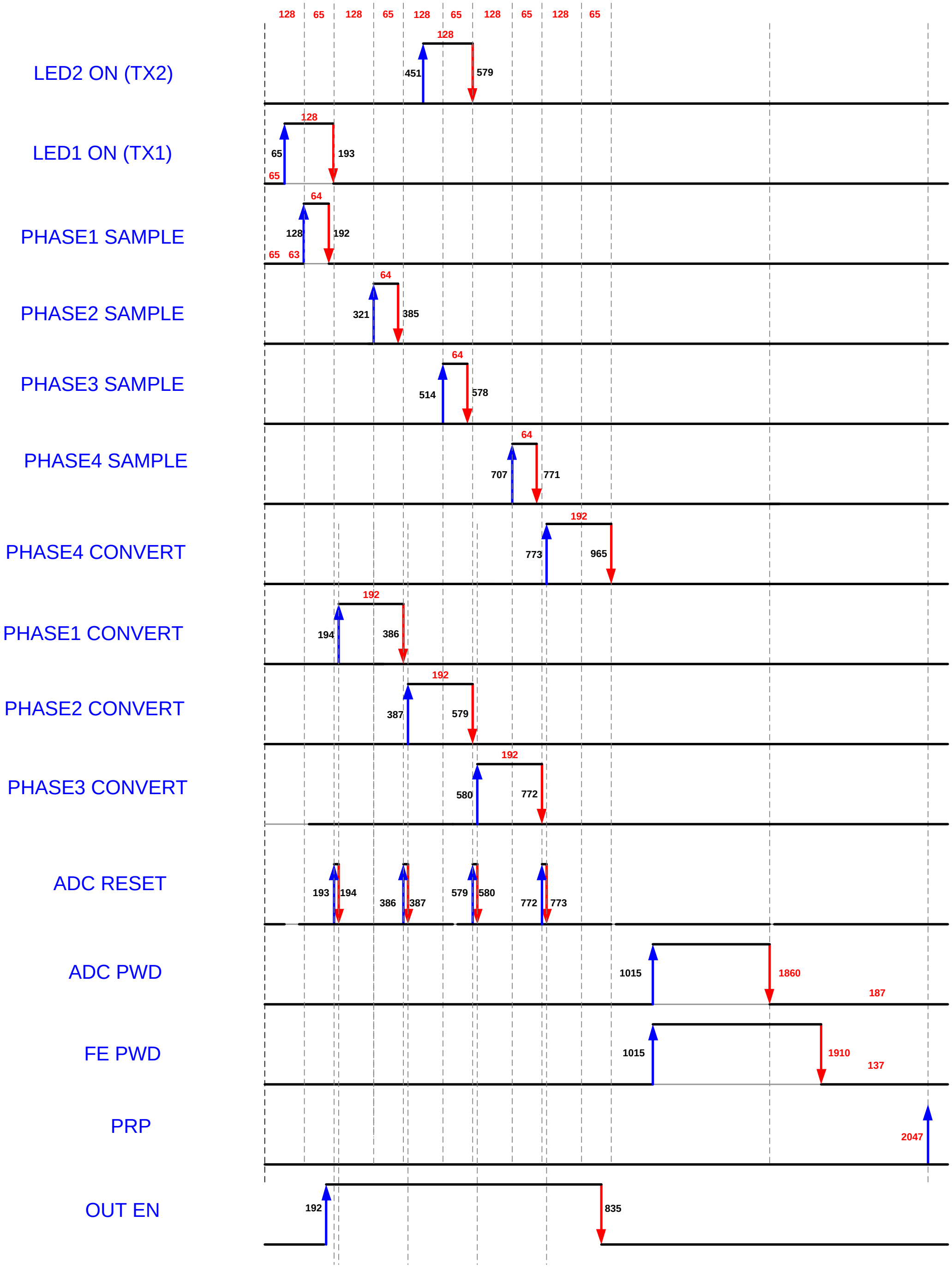


*Figure 5-3. Timing Module Operation Example*

*Table 5-1-1 Example timing module register setting*

| Phase | Register | Field | Value |
|---|---|---|---|
| LED2 On | AFE_TCTRL_CON3 | RG_LED2_START | 451 |
| | | RG_LED2_END | 579 |
| LED1 On | AFE_TCTRL_CON2 | RG_LED1_START | 65 |
| | | RG_LED1_END | 193 |
| Phase1 sample | AFE_TCTRL_CON4 | RG_SPHASE1_START | 128 |
| | | RG_SPHASE1_END | 192 |
| Phase2 sample | AFE_TCTRL_CON6 | RG_SPHASE2_START | 321 |
| | | RG_SPHASE2_END | 385 |
| Phase3 sample | AFE_TCTRL_CON5 | RG_SPHASE3_START | 514 |
| | | RG_SPHASE3_END | 578 |
| Phase4 sample | AFE_TCTRL_CON7 | RG_SPHASE4_START | 707 |
| | | RG_SPHASE4_END | 771 |
| Phase4 convert | AFE_TCTRL_CON11 | RG_CONVPHASE4_START | 773 |
| | | RG_CONVPHASE4_END | 965 |
| Phase1 convert | AFE_TCTRL_CON8 | RG_CONVPHASE1_START | 194 |
| | | RG_CONVPHASE1_END | 386 |
| Phase2 convert | AFE_TCTRL_CON10 | RG_CONVPHASE2_START | 387 |
| | | RG_CONVPHASE2_END | 579 |
| Phase3 convert | AFE_TCTRL_CON9 | RG_CONVPHASE3_START | 580 |
| | | RG_CONVPHASE3_END | 772 |
| ADC reset | AFE_TCTRL_CON12 | RG_ADC_RST1_START | 128 |
| | | RG_ADC_RST1_END | 192 |
| | AFE_TCTRL_CON13 | RG_ADC_RST2_START | 321 |
| | | RG_ADC_RST2_END | 385 |
| | AFE_TCTRL_CON14 | RG_ADC_RST3_START | 514 |
| | | RG_ADC_RST3_END | 578 |
| | AFE_TCTRL_CON15 | RG_ADC_RST4_START | 707 |
| | | RG_ADC_RST4_END | 771 |
| ADC PWD | AFE_TCTRL_CON16 | PPGADC_PWD_START | 1015 |
| | | PPGADC_PWD_END | 1860 |
| FE PWD | AFE_TCTRL_CON17 | RG_PPGFE_PWD_START | 1015 |
| | | RG_PPGFE_PWD_END | 1910 |

| **Phase** | **Register** | Field | Value |
|---|---|---|---|
| PRP | AFE_TCTRL_CON1 | RG_PRP | 2047 |
| OUT_EN | AFE_TCTRL_CON18 | RG_PPG_OUT_EN_START | 192 |
| | | RG_PPG_OUT_EN_START | 835 |

## 5.4 Timing Module Register Definitions

**BASE name: top_reg**

| Address | Name | Width | Register Function |
|---|---|---|---|
| 00000060 | AFE_DIG_ENABLE | 32 | **AFE Digital Part Enable** |

**00000060 AFE_DIG_ENABLE 00000000**

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Name** | | | | | | | | | | | | | | | | |
| **Type** | | | | | | | | | | | | | | | | |
| **Reset** | | | | | | | | | | | | | | | | |
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | | | | | | RG_P2S_MON_EN | | AD_TIMING_EN | RG_AFE_MEM_BISI_EN | RG_AFE_MEM_PPG2_EN | RG_AFE_MEM_PPG1_EN | RG_AFE_MEM_EKG_EN | RG_AFE_EKG_ON | RG_AFE_PPG_ON | | RG_AFE_BI_ON |
| **Type** | | | | | | RW | | RW | RW | RW | RW | RW | RW | RW | | RW |
| **Reset** | | | | | | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 |

| Bit(s) | Name | Description |
|---|---|---|
| 10 | RG_P2S_MON_EN | Enable P2S monitor module |
| 8 | AD_TIMING_EN | Enable timing module |
| 7 | RG_AFE_MEM_BISI_EN | Enable BI FIFO memory |
| 6 | RG_AFE_MEM_PPG2_EN | Enable PPG2 FIFO memory |
| 5 | RG_AFE_MEM_PPG1_EN | Enable PPG1 FIFO memory |
| 4 | RG_AFE_MEM_EKG_EN | Enable EKG FIFO memory |
| 3 | RG_AFE_EKG_ON | Enable EKG circuit |
| 2 | RG_AFE_PPG_ON | Enable PPG circuit |
| 0 | RG_AFE_BI_ON | Enable BI circuit |

**BASE name: top_reg_1**

| Address | Name | Width | Register Function |
|---|---|---|---|
| 00000028 | AFE_TCTRL_CON0 | 32 | **Timing Control Module Control 0** |
| 0000002C | AFE_TCTRL_CON1 | 32 | **Timing Control Module Control 1** |
| 00000030 | AFE_TCTRL_CON2 | 32 | **Timing Control Module Control 2** |
| 00000034 | AFE_TCTRL_CON3 | 32 | **Timing Control Module Control 3** |
| 00000038 | AFE_TCTRL_CON4 | 32 | **Timing Control Module Control 4** |
| 0000003C | AFE_TCTRL_CON5 | 32 | **Timing Control Module Control 5** |
| 00000040 | AFE_TCTRL_CON6 | 32 | **Timing Control Module Control 6** |
| 00000044 | AFE_TCTRL_CON7 | 32 | **Timing Control Module Control 7** |
| 00000048 | AFE_TCTRL_CON8 | 32 | **Timing Control Module Control 8** |
| 0000004C | AFE_TCTRL_CON9 | 32 | **Timing Control Module Control 9** |
| 00000050 | AFE_TCTRL_CON10 | 32 | **Timing Control Module Control 10** |
| 00000054 | AFE_TCTRL_CON11 | 32 | **Timing Control Module Control 11** |
| 00000058 | AFE_TCTRL_CON12 | 32 | **Timing Control Module Control 12** |
| 0000005C | AFE_TCTRL_CON13 | 32 | **Timing Control Module Control 13** |
| 00000060 | AFE_TCTRL_CON14 | 32 | **Timing Control Module Control 14** |
| 00000064 | AFE_TCTRL_CON15 | 32 | **Timing Control Module Control 15** |
| 00000068 | AFE_TCTRL_CON16 | 32 | **Timing Control Module Control 16** |
| 0000006C | AFE_TCTRL_CON17 | 32 | **Timing Control Module Control 17** |
| 00000070 | AFE_TCTRL_CON18 | 32 | **Timing Control Module Control 18** |

**00000028 AFE_TCTRL_CON0 FFFF0000**

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | AFE_TCTRL_DEBUG | | | | | | | | | | | | | | | |
| Type | RO | | | | | | | | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | x |
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Name | | | | | | | | | | | | | RG_TX_B2_SEL | RG_TX_B1_SEL | RG_SET_DATA | |
| Type | | | | | | | | | | | | | RW | RW | RW | |
| Reset | | | | | | | | | | | | | 0 | 0 | 0 | |

| Bit(s) | Name | Description |
|---|---|---|
| 31:16 | AFE_TCTRL_DEBUG | Reserved for chip debug |
| 3 | RG_TX_B2_SEL | **DA_TX_B2 selection**<br>1: from RG_TX_B2<br>0: use DA_PPG_LED2 falling edge to latch |
| 2 | RG_TX_B1_SEL | **DA_TX_B1 selection**<br>1: from RG_TX_B1<br>0: use DA_PPG_LED1 falling edge to latch |
| 1 | RG_SET_DATA | **software update trigger** |

**0000002C AFE_TCTRL_CON1 00001FFF**

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | | | | | | | | | | | | | | | | |
| Type | | | | | | | | | | | | | | | | |
| Reset | | | | | | | | | | | | | | | | |
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Name | | | RG_PRP | | | | | | | | | | | | | |
| Type | | | RW | | | | | | | | | | | | | |
| Reset | | | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

| Bit(s) | Name | Description |
|---|---|---|
| 13:0 | RG_PRP | **PRP Length** |

### 00000030 AFE_TCTRL_CON2 07FF0FFF

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | | | RG_LED1_START | | | | | | | | | | | | | |
| Type | | | RW | | | | | | | | | | | | | |
| Reset | | | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Name | | | RG_LED1_END | | | | | | | | | | | | | |
| Type | | | RW | | | | | | | | | | | | | |
| Reset | | | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

| Bit(s) | Name | Description |
|---|---|---|
| 29:16 | RG_LED1_START | LED1 ON START POSITION |
| 13:0 | RG_LED1_END | LED1 ON END POSITION |

### 00000034 AFE_TCTRL_CON3 17FF1FFF

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | | | RG_LED2_START | | | | | | | | | | | | | |
| Type | | | RW | | | | | | | | | | | | | |
| Reset | | | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Name | | | RG_LED2_END | | | | | | | | | | | | | |
| Type | | | RW | | | | | | | | | | | | | |
| Reset | | | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

| Bit(s) | Name | Description |
|---|---|---|
| 29:16 | RG_LED2_START | LED2 ON START POSITION |
| 13:0 | RG_LED2_END | LED2 ON END POSITION |

### 00000038 AFE_TCTRL_CON4 07FF0FFE

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | | | RG_SPHASE1_START | | | | | | | | | | | | | |
| Type | | | RW | | | | | | | | | | | | | |
| Reset | | | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Name | | | RG_ SPHASE1_END | | | | | | | | | | | | | |
| Type | | | RW | | | | | | | | | | | | | |

**00000038 AFE_TCTRL_CON4 07FF0FFE**

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Reset** | | | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

| Bit(s) | Name | Description |
|---|---|---|
| 29:16 | RG_ SPHASE1_START | PHASE1 SAMPLE START POSITION |
| 13:0 | RG_ SPHASE1_END | PHASE1 SAMPLE END POSITION |

**0000003C AFE_TCTRL_CON5 17FF1FFE**

| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Name** | | | RG_ SPHASE3_START | | | | | | | | | | | | | |
| **Type** | | | RW | | | | | | | | | | | | | |
| **Reset** | | | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | | | RG_ SPHASE3_END | | | | | | | | | | | | | |
| **Type** | | | RW | | | | | | | | | | | | | |
| **Reset** | | | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

| Bit(s) | Name | Description |
|---|---|---|
| 29:16 | RG_ SPHASE3_START | PHASE3 SAMPLE START POSITION |
| 13:0 | RG_ SPHASE3_END | PHASE3 SAMPLE END POSITION |

**00000040 AFE_TCTRL_CON6 0FFF17FE**

| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Name** | | | RG_ SPHASE2_START | | | | | | | | | | | | | |
| **Type** | | | RW | | | | | | | | | | | | | |
| **Reset** | | | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | | | RG_ SPHASE2_END | | | | | | | | | | | | | |
| **Type** | | | RW | | | | | | | | | | | | | |
| **Reset** | | | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

| Bit(s) | Name | Description |
|---|---|---|
| 29:16 | RG_ SPHASE2_START | PHASE2 SAMPLE START POSITION |
| 13:0 | RG_ SPHASE2_END | PHASE2 SAMPLE END POSITION |

**00000044 AFE_TCTRL_CON7 1FFF07FE**

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Name** | | | **RG_ SPHASE4_START** | | | | | | | | | | | | | |
| **Type** | | | RW | | | | | | | | | | | | | |
| **Reset** | | | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | | | **RG_ SPHASE4_END** | | | | | | | | | | | | | |
| **Type** | | | RW | | | | | | | | | | | | | |
| **Reset** | | | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

| Bit(s) | Name | Description |
|---|---|---|
| 29:16 | **RG_ SPHASE4_START** | **PHASE4 SAMPLE START POSITION** |
| 13:0 | **RG_ SPHASE4_END** | **PHASE4 SAMPLE END POSITION** |

### 00000048 AFE_TCTRL_CON8 100017FF

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | | | RG_CONVPHASE1_START | | | | | | | | | | | | | |
| Type | | | RW | | | | | | | | | | | | | |
| Reset | | | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Name | | | RG_CONVPHASE1_END | | | | | | | | | | | | | |
| Type | | | RW | | | | | | | | | | | | | |
| Reset | | | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

| Bit(s) | Name | Description |
|---|---|---|
| 29:16 | RG_CONVPHASE1_START | PHASE1 CONVERSION START POSITION |
| 13:0 | RG_CONVPHASE1_END | PHASE1 CONVERSION END POSITION |

### 0000004C AFE_TCTRL_CON9 000007FF

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | | | RG_CONVPHASE3_START | | | | | | | | | | | | | |
| Type | | | RW | | | | | | | | | | | | | |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Name | | | RG_CONVPHASE3_END | | | | | | | | | | | | | |
| Type | | | RW | | | | | | | | | | | | | |
| Reset | | | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

| Bit(s) | Name | Description |
|---|---|---|
| 29:16 | RG_CONVPHASE3_START | PHASE3 CONVERSION START POSITION |
| 13:0 | RG_CONVPHASE3_END | PHASE3 CONVERSION END POSITION |

### 00000050 AFE_TCTRL_CON10 18001FFF

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | | | RG_CONVPHASE2_START | | | | | | | | | | | | | |
| Type | | | RW | | | | | | | | | | | | | |
| Reset | | | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Name | | | RG_CONVPHASE2_END | | | | | | | | | | | | | |
| Type | | | RW | | | | | | | | | | | | | |

**00000050 AFE_TCTRL_CON10 18001FFF**

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset | | | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

| Bit(s) | Name | Description |
|---|---|---|
| 29:16 | RG_CONVPHASE2_START | PHASE2 CONVERSION START POSITION |
| 13:0 | RG_CONVPHASE2_END | PHASE2 CONVERSION END POSITION |

**00000054 AFE_TCTRL_CON11 08000FFF**

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | | | RG_CONVPHASE4_START | | | | | | | | | | | | | |
| Type | | | RW | | | | | | | | | | | | | |
| Reset | | | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Name | | | RG_CONVPHASE4_END | | | | | | | | | | | | | |
| Type | | | RW | | | | | | | | | | | | | |
| Reset | | | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

| Bit(s) | Name | Description |
|---|---|---|
| 29:16 | RG_CONVPHASE4_START | PHASE4 CONVERSION START POSITION |
| 13:0 | RG_CONVPHASE4_END | PHASE4 CONVERSION END POSITION |

**00000058 AFE_TCTRL_CON12 1FFF0000**

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | | | RG_ADC_RST1_START | | | | | | | | | | | | | |
| Type | | | RW | | | | | | | | | | | | | |
| Reset | | | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Name | | | RG_ADC_RST1_END | | | | | | | | | | | | | |
| Type | | | RW | | | | | | | | | | | | | |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit(s) | Name | Description |
|---|---|---|
| 29:16 | RG_ADC_RST1_START | ADC RESET1 START POSITION |
| 13:0 | RG_ADC_RST1_END | ADC RESET1 END POSITION |

**0000005C AFE_TCTRL_CON13 07FF0800**

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | | | RG_ADC_RST2_START | | | | | | | | | | | | | |
| Type | | | RW | | | | | | | | | | | | | |
| Reset | | | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Name | | | RG_ADC_RST2_END | | | | | | | | | | | | | |
| Type | | | RW | | | | | | | | | | | | | |
| Reset | | | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit(s) | Name | Description |
|---|---|---|
| 29:16 | RG_ADC_RST2_START | ADC RESET2 START POSITION |
| 13:0 | RG_ADC_RST2_END | ADC RESET2 END POSITION |

**00000060 AFE_TCTRL_CON14 0FFF1000**

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | | | RG_ADC_RST3_START | | | | | | | | | | | | | |
| Type | | | RW | | | | | | | | | | | | | |
| Reset | | | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Name | | | RG_ADC_RST3_END | | | | | | | | | | | | | |
| Type | | | RW | | | | | | | | | | | | | |
| Reset | | | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit(s) | Name | Description |
|---|---|---|
| 29:16 | RG_ADC_RST3_START | ADC RESET3 START POSITION |
| 13:0 | RG_ADC_RST3_END | ADC RESET3 END POSITION |

**00000064 AFE_TCTRL_CON15 17FF1800**

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | | | RG_ADC_RST4_START | | | | | | | | | | | | | |
| Type | | | RW | | | | | | | | | | | | | |
| Reset | | | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

**00000064 AFE_TCTRL_CON15 17FF1800**

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Name** | | | RG_ADC_RST4_END | | | | | | | | | | | | | |
| **Type** | | | RW | | | | | | | | | | | | | |
| **Reset** | | | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit(s) | Name | Description |
|---|---|---|
| 29:16 | RG_ADC_RST4_START | ADC RESET4 START POSITION |
| 13:0 | RG_ADC_RST4_END | ADC RESET4 END POSITION |

**00000068 AFE_TCTRL_CON16 00000000**

| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Name** | | | RG_PPGADC_PWD_START | | | | | | | | | | | | | |
| **Type** | | | RW | | | | | | | | | | | | | |
| **Reset** | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | | | RG_PPGADC_PWD_END | | | | | | | | | | | | | |
| **Type** | | | RW | | | | | | | | | | | | | |
| **Reset** | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit(s) | Name | Description |
|---|---|---|
| 29:16 | RG_PPGADC_PWD_START | PPGADC PWD START POSITION |
| 13:0 | RG_PPGADC_PWD_END | PPGADC PWD END POSITION |

**0000006C AFE_TCTRL_CON17 00000000**

| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Name** | | | RG_PPGFE_PWD_START | | | | | | | | | | | | | |
| **Type** | | | RW | | | | | | | | | | | | | |
| **Reset** | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | | | RG_PPGFE_PWD_END | | | | | | | | | | | | | |
| **Type** | | | RW | | | | | | | | | | | | | |
| **Reset** | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit(s) | Name | Description |
|---|---|---|
| 29:16 | RG_PPGFE_PWD_START | PPGFE PWD START POSITION |
| 13:0 | RG_PPGFE_PWD_END | PPGFE PWD END POSITION |

**00000070** **AFE_TCTRL_CON18** **3FFF3FFF**

| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Name** | | | **RG_PPG_OUT_EN_START** | | | | | | | | | | | | | |
| **Type** | | | RW | | | | | | | | | | | | | |
| **Reset** | | | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | | | **RG_PPG_OUT_EN_END** | | | | | | | | | | | | | |
| **Type** | | | RW | | | | | | | | | | | | | |
| **Reset** | | | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

| **Bit(s)** | **Name** | **Description** |
|---|---|---|
| 29:16 | **RG_PPG_OUT_EN_START** | **PPG OUT EN START POSITION** |
| 13:0 | **RG_PPG_OUT_EN_END** | **PPG OUT EN END POSITION** |

## 5.5 Programming Guide

The setting value of AFE_TCTRL_CON1~18 (0x2C~0x70) should be set before enabling EKG channel.

1. Setting the value to AFE_TCTRL_CON1~18 (0x2C~0x70)

2. Enable rg_set_data (setting AFE_TCTRL_CON0[1] (0x28) = 1 and AFE_DIG_ENABLE.

# 6 PPG Channel

## 6.1 General Description

Figure 6-1 illustrates the system block diagram for PPG acquisition.[1] The PPG channel is separated into two parts: Transmitter (TX) and Receiver (RX). The TX part consist of LED driver. The light emitted by LED is penetrated /reflected by the skin, and received by photodiode of the RX. The RX consists of a trans-impedance amplifier (TIA), a programmable gain amplifier (PGA), an ambient cancellation digital-to-analog converter (AMBDAC), and a 24-bit incremental ADC. It amplifies and digitizes the received current. The blocks are described in more detail in the following sections.

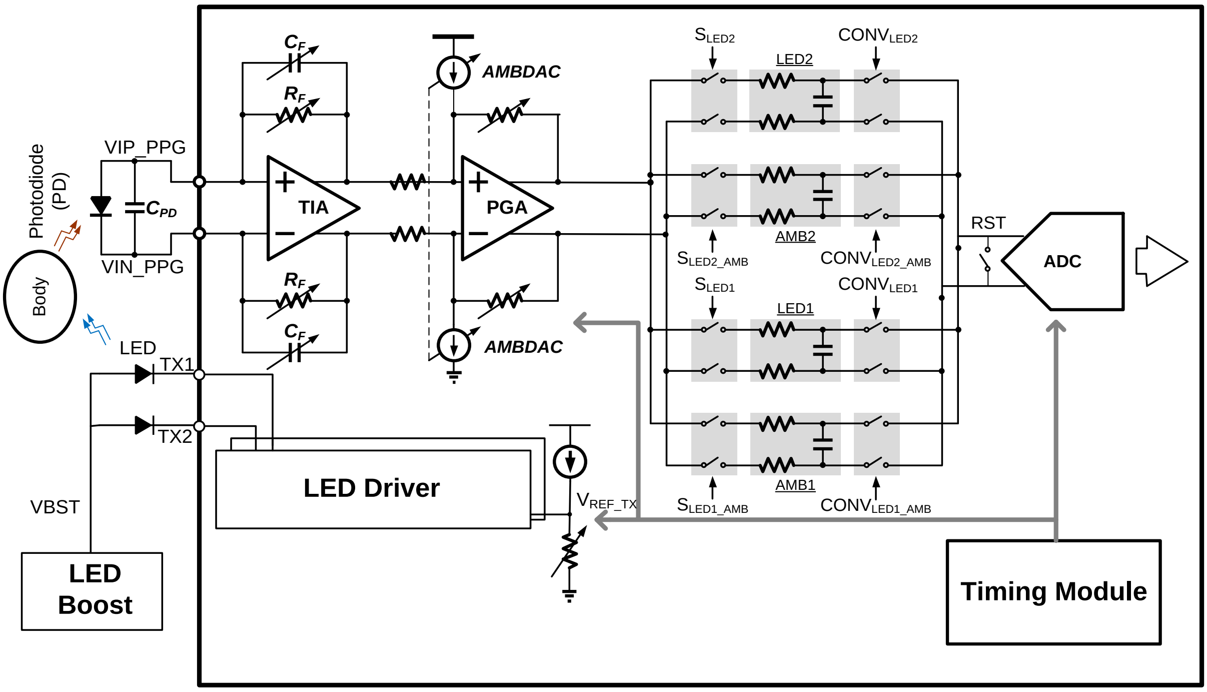


*Figure 6-1. PPG Channel Analog Part Block Diagram*

[1] "A **photoplethysmogram** (PPG) is an optically obtained plethysmogram, a volumetric measurement of an organ. " Wikipedia.

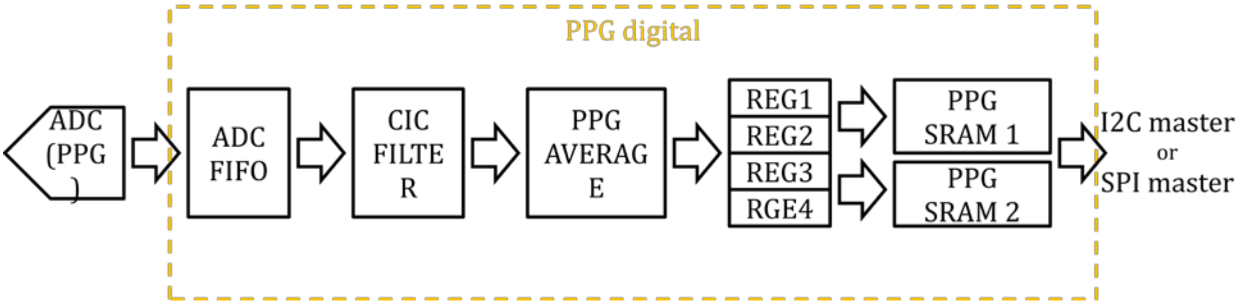


*Figure 6-2. PPG Channel Digital Part Block Diagram*

### 6.1.1 Transmitter Front-End

The LED driver and the voltage boost are used to light up external LED and to provide the voltage drop of LED. The two current DACs can enable two wavelength LEDs, which are turned on in a time-interleaved manner. The LED full scale current range is from 9.5 mA to 103 mA with a 3-bit current step of 13.4mA. The LED current can be set by the following equation: (equation 1 and equation 2)

$$\text{LED1 current (TX1)} = \frac{\text{Full scale range of LED current}}{256} * \text{DAC1 code} \qquad (1)$$

$$\text{LED2 current (TX2)} = \frac{\text{Full scale range of LED current}}{256} * \text{DAC2 code} \qquad (2)$$

As shown in Figure 6-3 below, two LED driver schemes are supported:

- Push-Pull Mode: A push-pull drive for a three-terminal LED package. The minimum external supply voltage = 0.7V + (maximum voltage drop across the LED).  This value is depends on the registry LED current settings.
- H-Bridge Mode: An H-bridge drive for two-terminal back-to-back LED package. The minimum external supply voltage (LED_SUP) = 0.7V + (maximum voltage drop across the LED). This value is depends on the registry LED current settings.

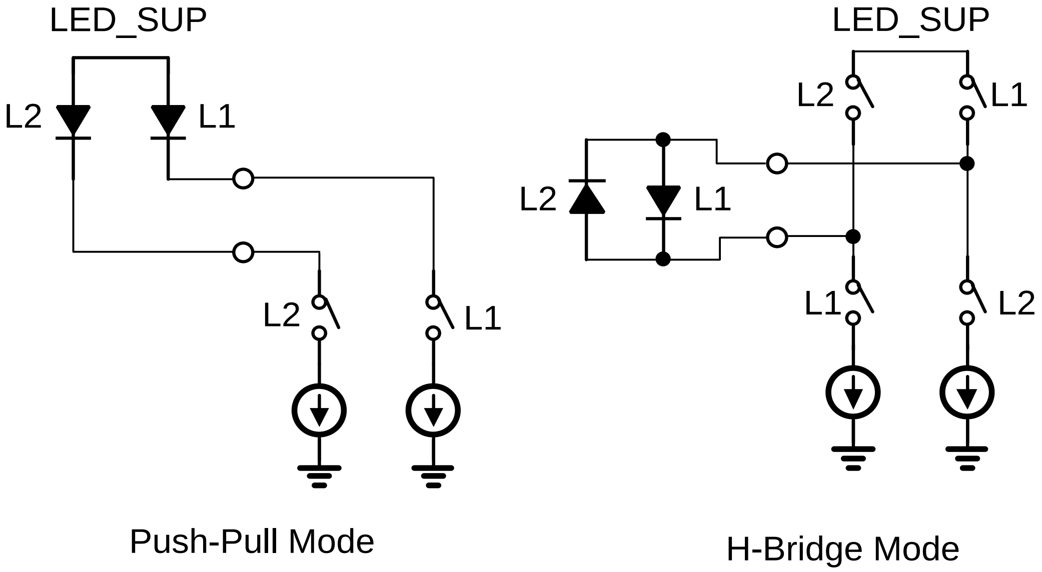


***Figure 6-3. Push-Pull Mode and H-Bridge Mode***

### 6.1.2 Receiver Front-End

As shown in Figure 6-4 the photo detector transfers the reflected light into current, $I_{PD,}$ and then amplifies by TIA that converts the input photodiode current into an appropriate voltage. The feedback resistor of the amplifier ($R_F$) has seven gain settings (1 MΩ, 500KΩ, 250KΩ, 100KΩ, 50KΩ, 25KΩ, 10KΩ) and is programmable to support a wide input range of photodiode currents from 0.5-µA to 50-µA.

The feedback resistor ($R_F$) and the feedback capacitor ($C_F$) form a low pass filter for the input current. Please ensure that the low pass filter ($R_F$ and $C_F$) has sufficiently high bandwidth (as shown in Equation 3).

$$R_F * C_F \le \frac{\text{LED Sample Time}}{10} \le \frac{\text{Duty}}{\text{PRF} * 10} = \frac{\text{LED ON Time}}{10} \qquad (3)$$

The output voltage of the I-V amplifier includes the pleth component ($I_{Pleth}$) and a component resulting from the ambient light leakage ($I_{Amb}$). The second stage consists of an AMBDAC that sources the cancellation current ($I_{DC\ offset}$) and a PGA.

The AMBDAC has a cancellation current range of 6 µA with six steps (1 µA each) for two phases (LED1/LED2 phase and AMB1/AMB2 phase). The PGA amplifier gains up the pleth component and has five programmable gain settings: 1, 1.5, 2, 3, 4, and 6 (V/V). Then, the signals are sampled by the corresponding LPFs and digitized by a 24-bit incremental ADC.

$$V_{DIFF} = 2 * \left[(I_{Pleth} + I_{Amb}) * \frac{R_F}{R_I} - I_{DC\ offset}\right] * R_G \qquad (4)$$

Where $\mathrm{R_I} = 100\mathrm{K\Omega}$

$$\mathrm{R_G} = 100\mathrm{K\Omega} * \mathrm{PGA\ gain}$$

The PPG control logic (timing module) can adjust the sampling rate (Equation 5) and duty cycle (Equation 6) of the LED currents and also power down the AFE when the LEDs are off. The output of the ambient cancellation amplifier is separated into LED1, AMB1, LED2, AMB2 channels. When LED2 is on, the amplifier output is filtered and sampled on capacitor $C_{LED2}$. Similarly, the LED1 signal is sampled on the $C_{LED1}$ capacitor when LED1 is ON. In between the LED2 and LED1 pulses, the idle amplifier output is sampled to estimate the ambient signal on capacitors $C_{LED2\_amb}$ and $C_{LED1\_amb}$. The minimum supported RX sampling time is 50 µs.

$$\mathrm{PRF\ (Sampling\ rate, Hz)} = \frac{\mathrm{1MHz}}{(PRP + 1)} = \mathrm{64Hz} \sim \mathrm{4096Hz} \qquad (5)$$

$$\mathrm{Duty\ (\%)} = \frac{\mathrm{LED\ ON\ Time}}{\mathrm{PRP} * 100} = 1.5\% \sim 25\% \qquad (6)$$

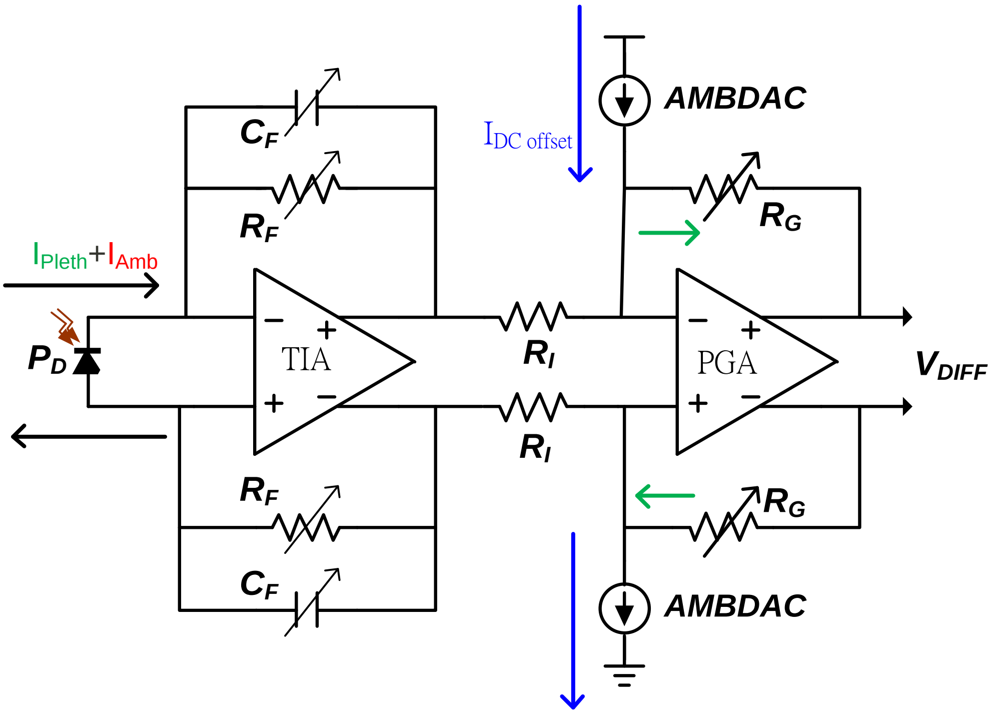


*Figure 6-4. Receiver Front-End*

### 6.1.3 PPG AVERAGE Function Introduce

The PPG AVERAGE module is used to average multiple ADC samples and reduce noise to improve dynamic range. At the next rising edge of the ADC reset signal, the average value (23-bit) is written into the output registers sequentially as follows (see Figure 6-5):

- At the 25% reset signal, the averaged 23-bit word is written to reg1.
- At the 50% reset signal, the averaged 23-bit word is written to reg2.
- At the 75% reset signal, the averaged 23-bit word is written to reg3.
- At the next 0% reset signal, the averaged 23-bit word is written to reg4. The contents of reg2 and reg3 are written to reg5 and the contents of reg4 and reg1 are written to reg6.

At the rising edge of the ADC_RDY signal, the contents of all six result registers can be read out.

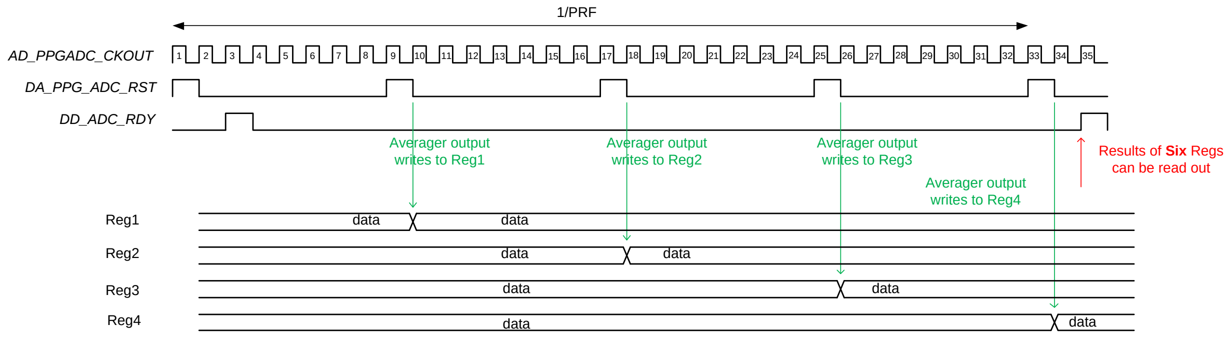


*Figure 6-5. ADC Data with Averaging Enabled*

### 6.1.4 PPG Data Format

The MT2511 outputs 24 bits of data per channel in binary twos complement format, MSB first. A positive full-scale input produces an output code of 7FFFFFh and the negative full-scale input produces an output code of 800000h. The output clips at these codes for signals exceeding full-scale. All 24 bits toggle when the analog input is at positive or negative full-scale.

$$\begin{aligned} \mathrm{PPG\ (mV)} &= \frac{\mathrm{PPGADC\ output\ code} * \mathrm{LSB}}{1000} \\ &= \frac{\mathrm{PPGADC\ output\ code} * \frac{V_{ref}}{2^{16}}}{1000} \\ &= \frac{\mathrm{PPGADC\ output\ code} * \frac{3.2V}{2^{16}}}{1000} \qquad (7) \end{aligned}$$

## 6.2 PPG Register Definitions

**Module name: top_reg**

| Address | Name | Width | Register Function |
|---|---|---|---|
| 00000018 | PPGFE_CON0 | 32 | PPGFE Control 0 |
| 0000001C | PPGFE_CON1 | 32 | PPGFE Control 1 |
| 00000020 | PPGADC_CON0 | 32 | PPGADC Control 0 |
| 00000028 | LEDDRV_CON0 | 32 | LED Driving Control 0 |
| 0000002C | LEDDRV_CON1 | 32 | LED Driving Control 1 |
| 00000068 | AFE_PPG_CON | 32 | PPG Digital Part Control |
| 000000D0 | AFE_MEM_CON4 | 32 | PPG1 SRAM Control 0 |
| 000000D4 | AFE_MEM_CON5 | 32 | PPG1 SRAM Control 1 |
| 000000D8 | AFE_MEM_CON6 | 32 | PPG1 SRAM Control 2 |
| 000000DC | AFE_MEM_CON7 | 32 | PPG1 SRAM Control 3 |
| 000000E0 | AFE_MEM_CON8 | 32 | PPG2 SRAM Control 0 |
| 000000E4 | AFE_MEM_CON9 | 32 | PPG2 SRAM Control 1 |
| 000000E8 | AFE_MEM_CON10 | 32 | PPG2 SRAM Control 2 |
| 000000EC | AFE_MEM_CON11 | 32 | PPG2 SRAM Control 3 |

**00000018 PPGFE_CON0 PPGFE Control 0 70244212**

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | | RG_PPG_PD | RG_PPG_LPF_CT | | RG_AMBDAC2 | | | RG_AMBDAC1 | | | | RG_PGA_GAIN | | | RG_PPG_ENSEPGAIN | |
| Type | | RW | RW | | RW | | | RW | | | | RW | | | RW | |
| Reset | | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 1 | 0 | |
| **Bit** | **15** | **14** | **13** | **12** | **11** | **10** | **9** | **8** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| Name | RG_TIA_CF2 | | | | | RG_TIA_CF1 | | | | | RG_TIA_RF2 | | | RG_TIA_RF1 | | |
| Type | RW | | | | | RW | | | | | RW | | | RW | | |
| Reset | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |

| Bit(s) | Name | Description |
|---|---|---|
| 30 | RG_PPG_PD | **PPG power down**<br>1: power down |

| Bit(s) | Name | Description |
|---|---|---|
| | | 0: power on |
| 29:28 | RG_PPG_LPF_CT | **PPG LPF corner adjustment**<br>11: 500Hz (default)<br>10: 1KHz<br>01: 2KHz<br>00: 4KHz |
| 27:25 | RG_AMBDAC2 | **AMB phase current adjustment ($I_{DC\ Offset}$):**<br>000/001: 1uA,<br>010: 2uA,<br>011: 3uA,<br>100: 4uA,<br>101: 5uA,<br>110: 6uA. |
| 24:22 | RG_AMBDAC1 | **LED phase current adjustment ($I_{DC\ Offset}$):**<br>000/001: 1uA,<br>010: 2uA,<br>011: 3uA,<br>100: 4uA,<br>101: 5uA,<br>110: 6uA. |
| 20:18 | RG_PGA_GAIN | **PGA gain adjustment (V/V)**<br>111/110/101: 6<br>100: 4<br>011: 3<br>010: 2<br>001: 1.5 (default)<br>000: 1 |
| 17 | RG_PPG_ENSEPGAIN | **enable separate gain for LED1 phase and LED2 phase:**<br>1: enable,<br>0: disable |
| 15:11 | RG_TIA_CF2 | **CF2<4:0>:TIA CF adjustment for LED2 phase and AMB2 phase if RG_PPG_ENSEPGAIN=H:**<br>bit4=1: +150p,<br>bit3=1: +50p,<br>bit2=1: +25p,<br>bit1=1: +15p,<br>bit0=1: +5p<br>00000: 5pF |
| 10:6 | RG_TIA_CF1 | **CF1<4:0>:TIA CF adjustment for LED1 phase and AMB1 phase**<br>bit4=1: +150p,<br>bit3=1: +50p,<br>bit2=1: +25p,<br>bit1=1: +15p,<br>bit0=1: +5p,<br>00000: 5pF |
| 5:3 | RG_TIA_RF2 | **TIA RF adjustment for LED2 phase and AMB2 phase if RG_PPG_ENSEPGAIN=H:**<br>000: 500k,<br>001: 250k,<br>010: 100k,<br>011: 50k,<br>100: 25k,<br>101: 10k,<br>110/111: 1M |
| 2:0 | RG_TIA_RF1 | **TIA RF adjustment for LED1 phase and AMB1 phase**<br>000: 500k,<br>001: 250k, |

| Bit(s) | Name | Description |
|---|---|---|
| | | 010: 100k,<br>011: 50k,<br>100: 25k,<br>101: 10k,<br>110/111: 1M |

**0000001C** **PPGFE_CON1** **PPGFE Control 1** **0048CC29**

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | | | | | | | | | | | | | | | | |
| Type | | | | | | | | | | | | | | | | |
| Reset | | | | | | | | | | | | | | | | |
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Name | | | | | RG_AMBDAC_PWDB | | | | | | | | | | | |
| Type | | | | | RW | | | | | | | | | | | |
| Reset | | | | | 1 | | | | | | | | | | | |

| Bit(s) | Name | Description |
|---|---|---|
| 11 | RG_AMBDAC_PWDB | **AMBDAC_PWDB:**<br>1: power on,<br>0: power down, also need to set<br>PPGFE_CON0::RG_AMBDAC1=0<br>PPGFE_CON0::RG_AMBDAC2=0 |

**00000020** **PPGADC_CON0** **000F5555**

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | | | | | | | | | | | | | | | | |
| Type | | | | | | | | | | | | | | | | |
| Reset | | | | | | | | | | | | | | | | |
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Name | | | | | | | | | | | | | | | | RG_PPGADC_PD |
| Type | | | | | | | | | | | | | | | | RW |
| Reset | | | | | | | | | | | | | | | | 1 |

| Bit(s) | Name | Description |
|---|---|---|

| Bit(s) | Name | Description |
|---|---|---|
| 0 | RG_PPGADC_PD | **PPGADC power down:**<br>1: power down,<br>0: power on |

**00000028 LEDDRV_CON0 LED Driving Control 0 00002900**

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Name** | | | | | | | | | | | | | | | | |
| **Type** | | | | | | | | | | | | | | | | |
| **Reset** | | | | | | | | | | | | | | | | |
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | | | | | | | | | **RG_TX_SEL** | | | | | | | **RG_TX_HBRIDGE_PUSHPULL** |
| **Type** | | | | | | | | | RW | | | | | | | RW |
| **Reset** | | | | | | | | | 0 | 0 | 0 | | | | | 0 |

| Bit(s) | Name | Description |
|---|---|---|
| 7:5 | RG_TX_SEL | **Select full-scale range of LED current:**<br>000: 9.5 mA<br>111: 103.3 mA<br>a step is 13.4 mA |
| 0 | RG_TX_HBRIDGE<br>_PUSHPULL | **1: Hbridge mode; 0: Push-pull mode** |

**0000002C** **LEDDRV_CON1** **LED Driving Control 1** **00002020**

| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Name** | | | | | | | | | | | | | | | | |
| **Type** | | | | | | | | | | | | | | | | |
| **Reset** | | | | | | | | | | | | | | | | |
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | **RG_TX_B2** | | | | | | | | **RG_TX_B1** | | | | | | | |
| **Type** | RW | | | | | | | | RW | | | | | | | |
| **Reset** | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |

| **Bit(s)** | **Name** | **Description** |
|---|---|---|
| 15:8 | RG_TX_B2 | **DAC2 code 00000000:minimum; 11111111:maximum** |
| 7:0 | RG_TX_B1 | **DAC1 code 00000000:minimum; 11111111:maximum** |

**00000068 AFE_PPG_CON PPG Digital Part Control 04B6270A**

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | RG_NUM_OF_AVG | | | | | | RG_PPG_FIFO | | | RG_REG5_CTL_MINUEND | | RG_REG5_CTL_SUBTRAHEND | | RG_REG6_CTL_MINUEND | | RG_REG6_CTL_SUBTRAHEND |
| Type | RW | | | | | | RW | | | RW | | RW | | RW | | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Name | RG_REG6_CTL_SUBTRAHEND | RG_NUM_TO_L1 | | | RG_NUM_TO_L2 | | | RG_NUM_TO_L3 | | | RG_NUM_TO_L4 | | | RG_NUM_TO_BI | | |
| Type | RW | RW | | | RW | | | RW | | | RW | | | RW | | |
| Reset | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |

| Bit(s) | Name | Description |
|---|---|---|
| 31:26 | RG_NUM_OF_AVG | Reference of number of average in PPGAVG |
| 25:23 | RG_PPG_FIFO | PPG FIFO start point |
| 22:21 | RG_REG5_CTL_MINUEND | Set minuend of reg5, default = reg2 |
| 20:19 | RG_REG5_CTL_SUBTRAHEND | Set subtrahend of reg5, default = reg3 |
| 18:17 | RG_REG6_CTL_MINUEND | Set minuend of reg6, default = reg4 |
| 16:15 | RG_REG6_CTL_SUBTRAHEND | Set subtrahend of reg6, default = reg1 |
| 14:12 | RG_NUM_TO_L1 | Set l1 register, default = reg2 |
| 11:9 | RG_NUM_TO_L2 | Set l2 register, default = reg3 |
| 8:6 | RG_NUM_TO_L3 | Set l3 register, default = reg4 |
| 5:3 | RG_NUM_TO_L4 | Set l4 register, default = reg1 |
| 2:0 | RG_NUM_TO_BI | Set bi register, default = reg2 |

**000000D0 AFE_MEM_CON4 PPG1 SRAM Control 0 00000000**

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | UPSRAM_CE | UPSRAM_RW_ADDR_AUTO_INC | UPSRAM_ATRIG | | UPSRAM_WTRIG | UPSRAM_RTRIG | UPSRAM_CUR_ADDR | | | | | | | | | |
| Type | RW | RW | RW | | RO | RO | RO | | | | | | | | | |
| Reset | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Name | | | | | | | UPSRAM_RW_ADDR | | | | | | | | | |
| Type | | | | | | | RW | | | | | | | | | |
| Reset | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit(s) | Name | Description |
|---|---|---|
| 31 | UPSRAM_CE | **PPG1 SRAM chip select** |
| 30 | UPSRAM_RW_ADDR_AUTO_INC | **Automatically increase I2C/SPI access PPG1 SRAM address**<br>0: no effect<br>1: automatically increase |
| 29 | UPSRAM_ATRIG | **Enable access PPG1 SRAM from I2C/SPI**<br>0: no effect<br>1: toggle access trigger |
| 27 | UPSRAM_WTRIG | **Write PPG1 SRAM from I2C/SPI**<br>0: no effect<br>1: write trigger |
| 26 | UPSRAM_RTRIG | **Read PPG1 SRAM from I2C/SPI**<br>0: no effect<br>1: read trigger |
| 25:16 | UPSRAM_CUR_ADDR | **The current access address of PPG1 SRAM from I2C/SPI** |
| 9:0 | UPSRAM_RW_ADDR | **The access address of PPG1 SRAM from I2C/SPI** |

**000000D4 AFE_MEM_CON5 PPG1 SRAM Control 1 00000000**

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | UPSRAM_WR_DATA[31:16] | | | | | | | | | | | | | | | |
| Type | RW | | | | | | | | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Name | UPSRAM_WR_DATA[15:0] | | | | | | | | | | | | | | | |
| Type | RW | | | | | | | | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit(s) | Name | Description |
|---|---|---|
| 31:0 | UPSRAM_WR_DATA | The write data from I2C/SPI to PPG1 SRAM |

**000000D8 AFE_MEM_CON6 PPG1 SRAM Control 2 00000000**

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | UPSRAM_RD_DATA[31:16] | | | | | | | | | | | | | | | |
| Type | RO | | | | | | | | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Name | UPSRAM_RD_DATA[15:0] | | | | | | | | | | | | | | | |
| Type | RO | | | | | | | | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit(s) | Name | Description |
|---|---|---|
| 31:0 | UPSRAM_RD_DATA | The read data from PPG1 SRAM to I2C/SPI |

**000000DC AFE_MEM_CON7 PPG1 SRAM Control 3 00000000**

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | | | | | | | FILTER_CUR_ADDR | | | | | | | | | |
| Type | | | | | | | RO | | | | | | | | | |
| Reset | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Name | | | | | | | | RG_IRG_TH | | | | | | | | |
| Type | | | | | | | | RW | | | | | | | | |
| Reset | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit(s) | Name | Description |
|---|---|---|
| 25:16 | FILTER_CUR_ADDR | The current access address of PPG1 SRAM from PPG1 FIFO |
| 8:0 | RG_IRG_TH | The threshold of PPG1 SRAM to set irq signal |

**000000E0 AFE_MEM_CON8 PPG2 SRAM Control 0 00000000**

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | UPSRAM_CE | UPSRAM_RW_ADDR_AUTO_INC | UPSRAM_ATRIG | | UPSRAM_WTRIG | UPSRAM_RTRIG | UPSRAM_CUR_ADDR | | | | | | | | | |
| Type | RW | RW | RW | | RO | RO | RO | | | | | | | | | |
| Reset | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Name | | | | | | | UPSRAM_RW_ADDR | | | | | | | | | |
| Type | | | | | | | RW | | | | | | | | | |
| Reset | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit(s) | Name | Description |
|---|---|---|
| 31 | UPSRAM_CE | PPG2 SRAM chip select |
| 30 | UPSRAM_RW_ADDR_AUTO_INC | Automatically increase I2C/SPI access PPG2 SRAM address<br>0: no effect<br>1: automatically increase |
| 29 | UPSRAM_ATRIG | Enable access PPG2 SRAM from I2C/SPI<br>0: no effect<br>1: toggle access trigger |
| 27 | UPSRAM_WTRIG | Write PPG2 SRAM from I2C/SPI<br>0: no effect<br>1: write trigger |
| 26 | UPSRAM_RTRIG | Read PPG2 SRAM from I2C/SPI<br>0: no effect<br>1: read trigger |
| 25:16 | UPSRAM_CUR_ADDR | The current access address of PPG2 SRAM from I2C/SPI |
| 9:0 | UPSRAM_RW_ADDR | The access address of PPG2 SRAM from I2C/SPI |

**000000E4 AFE_MEM_CON9 PPG2 SRAM Control 1 00000000**

**000000E4** **AFE_MEM_CON9** **PPG2 SRAM Control 1** **00000000**

| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Name** | **UPSRAM_WR_DATA[31:16]** | | | | | | | | | | | | | | | |
| **Type** | RW | | | | | | | | | | | | | | | |
| **Reset** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | **UPSRAM_WR_DATA[15:0]** | | | | | | | | | | | | | | | |
| **Type** | RW | | | | | | | | | | | | | | | |
| **Reset** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| **Bit(s)** | **Name** | **Description** |
|---|---|---|
| 31:0 | **UPSRAM_WR_DATA** | **The write data from I2C/SPI to PPG2 SRAM** |

**000000E8** **AFE_MEM_CON10** **PPG2 SRAM Control 2** **00000000**

| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Name** | **UPSRAM_RD_DATA[31:16]** | | | | | | | | | | | | | | | |
| **Type** | RO | | | | | | | | | | | | | | | |
| **Reset** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | **UPSRAM_RD_DATA[15:0]** | | | | | | | | | | | | | | | |
| **Type** | RO | | | | | | | | | | | | | | | |
| **Reset** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| **Bit(s)** | **Name** | **Description** |
|---|---|---|
| 31:0 | **UPSRAM_RD_DATA** | **The read data from PPG2 SRAM to I2C/SPI** |

**000000EC** **AFE_MEM_CON11** **PPG2 SRAM Control 3** **00000000**

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| **Name** | | | | | | | **FILTER_CUR_ADDR** | | | | | | | | | |
| **Type** | | | | | | | RO | | | | | | | | | |
| **Reset** | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | | | | | | | | **RG_IRG_TH** | | | | | | | | |
| **Type** | | | | | | | | RW | | | | | | | | |
| **Reset** | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit(s) | Name | Description |
|---|---|---|
| 25:16 | **FILTER_CUR_ADDR** | **The current access address of PPG2 SRAM from PPG2 FIFO** |
| 8:0 | **RG_IRG_TH** | **The threshold of PPG2 SRAM to set irq signal** |

## 6.3 Programming Guide

The setting value of PPGFE_CON0 (0x18), PPGFE_CON1 (0x1C), PPGADC_CON0 (0x20), LEDDRV_CON0 (0x28), LEDDRV_CON1 (0x2C), LEDSUP_CON1 (0x34), AFE_PPG_CON (0x68) should be set before enabling PPG channel.

1. Setting the value to PPGFE_CON0 (0x18), PPGFE_CON1 (0x1C), PPGADC_CON0 (0x20), LEDDRV_CON0 (0x28), LEDDRV_CON1 (0x2C), LEDSUP_CON1 (0x34), AFE_PPG_CON (0x68).

2. Enable PPG channel (setting AFE_DIG_ENABLE (0x60) = 0x564)

**Read/Write PPG1 SRAM from I2C/SPI**

1. READ:

    a. Setting AFE_MEM_CON4 (0xD0) = 0x60000000 | start_addr[9:0]

    b. HW will put read_data from PPG1 SRAM to AFE_MEM_CON6 (0xD8)

2. WRITE:

    a. Setting AFE_MEM_CON4 (0xD0) = 0x60000000 | start_addr[9:0]

    b. Setting AFE_MEM_CON5 (0xD4) = write_data[31:0]

c. HW will write write_data to PPG1 SRAM

**Read/Write PPG2 SRAM from I2C/SPI**

1. READ:

    a. Setting AFE_MEM_CON8 (0xE0) = 0x60000000 | start_addr[9:0]

    b. HW will put read_data from PPG2 SRAM to AFE_MEM_CON10 (0xE8)

2. WRITE:

    a. Setting AFE_MEM_CON8 (0xE0) = 0x60000000 | start_addr[9:0]

b. Setting AFE_MEM_CON9 (0xE4) = write_data[31:0]

c. HW will write write_data to PPG2 SRAM

# 7 EKG Channel

## 7.1 General Description

Figure 7-1 and Figure 7-2 show the analog/digital parts of EKG system. The EKG channel supports two-electrode (2E) mode and right leg drive (RLD) mode, and acts as a buffer between human and circuit. It integrates a programmable gain amplifier (PGA), a right leg drive amplifier, and a 24-bit sigma-delta analog-to-digital converter (ADC) to sense and digitize the EKG signal. The PGA is a differential input/differential output, and has seven gain settings (1, 2, 3, 4, 6, 8, and 12). The sampling frequency of EKG ADC is adjustable from 64Hz to 4096Hz. There are two extra samples (data=0) when doing first time sample EKG.

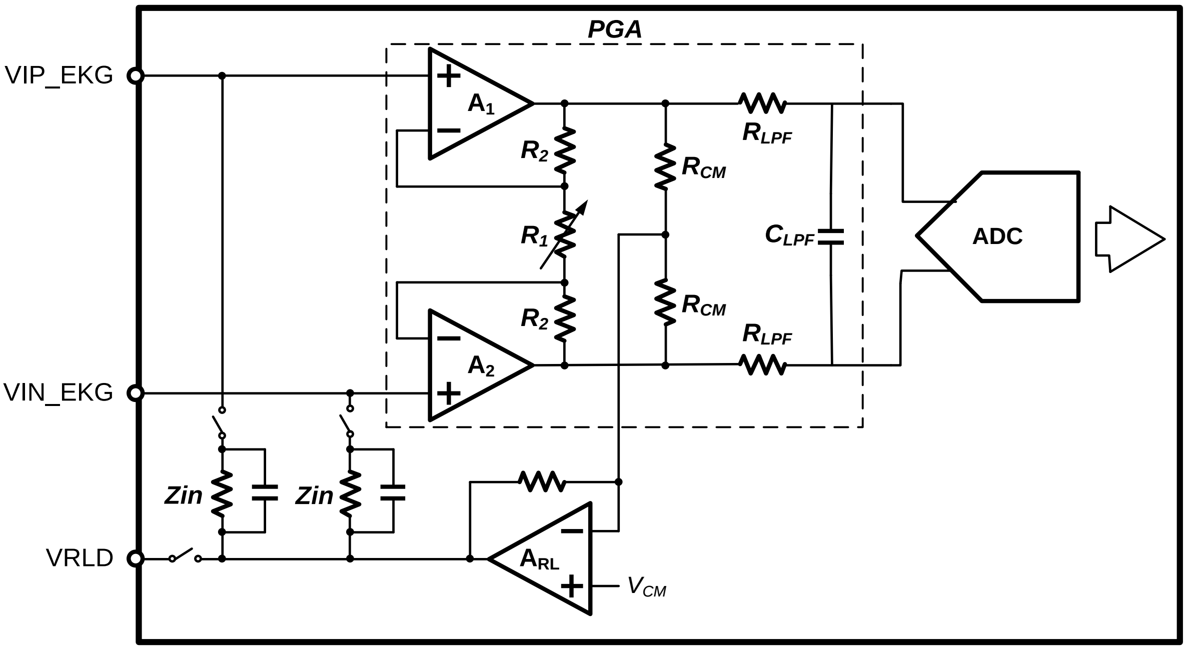


*Figure 7-1. EKG Channel Analog Part Block Diagram*

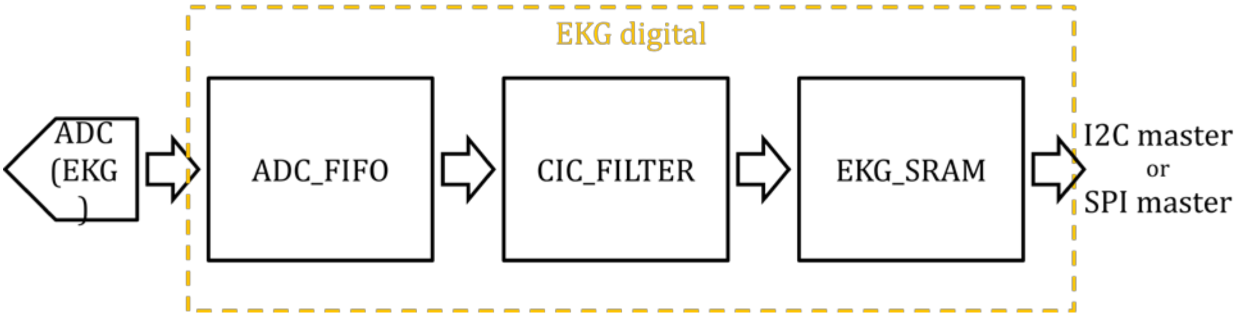


*Figure 7-2. EKG Channel Digital Part Block Diagram*

### 7.1.1 EKG Data Format

The MT2511 outputs 24 bits of data per channel in binary twos complement format, MSB first. A positive full-scale input produces an output code of 7FFFFFh and the negative full-scale input produces an output code of 800000h. The output clips at these codes for signals exceeding full-scale. All 24 bits toggle when the analog input is at positive or negative full-scale.

$$\begin{aligned}\text{EKG (mV)} &= \frac{\text{EKGADC output code} * \text{LSB}}{1000 * EKG\ PGA\ Gain} \\ &= \frac{\text{EKGADC output code} * \frac{V_{ref}}{2^{23}}}{1000 * EKG\ PGA\ Gain} \\ &= \frac{\text{EKGADC output code} * \frac{4V}{2^{23}}}{1000 * EKG\ PGA\ Gain} \qquad (7)\end{aligned}$$

## 7.2 EKG Channel Register Definitions

**Module name: top_reg**

| Address | Name | Width | Register Function |
|---|---|---|---|
| 00000008 | EKGFE_CON0 | 32 | EKGFE Control |
| 00000010 | EKGADC_CON0 | 32 | EKGADC Control |
| 00000064 | AFE_EKG_CON | 32 | EKG Digital Part Control |
| 000000C0 | AFE_MEM_CON0 | 32 | EKG SRAM Control 0 |
| 000000C4 | AFE_MEM_CON1 | 32 | EKG SRAM Control 1 |
| 000000C8 | AFE_MEM_CON2 | 32 | EKG SRAM Control 2 |
| 000000CC | AFE_MEM_CON3 | 32 | EKG SRAM Control 3 |

**00000008 EKGFE_CON0 EKGFE Control 010AD443**

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Name** | | | | | | | | | | | | | | | | |
| **Type** | | | | | | | | | | | | | | | | |
| **Reset** | | | | | | | | | | | | | | | | |
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | **RG_RL_ZIN** | | | | | **RG_RL_MODE** | | | **RG_IA_GAIN** | | | | | | | **RG_EKGFE_PD** |
| **Type** | RW | | | | | RW | | | RW | | | | | | | RW |
| **Reset** | 1 | 1 | | | | 1 | | | 0 | 1 | 0 | 0 | | | | 1 |

| Bit(s) | Name | Description |
|---|---|---|
| 15:14 | RG_RL_ZIN | **Input impedance adjustment (Only @2E mode)**<br>11: Zin=500M (default)<br>10/01: Zin=250M<br>00: Zin=125M |
| 10 | RG_RL_MODE | **2E or RLD mode control.**<br>1: 2E mode<br>0: RLD mode (default) |
| 7:4 | RG_IA_GAIN | **PGA gain adjustment (V/V)**<br>0000: 1<br>0001: 2<br>0010: 3<br>0011: 4<br>0100: 6 (default)<br>0101: 8<br>0110: 12 |
| 0 | RG_EKGFE_PD | **EKGFE power down:**<br>1: power down,<br>0: power on |

**00000010 EKGADC_CON0 EKGADC Control 002F5555**

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Name** | | | | | | | | | | | | **RG_EKGADC_MODESEL** | | | | |
| **Type** | | | | | | | | | | | | RW | | | | |
| **Reset** | | | | | | | | | | | | 0 | 1 | 1 | | |
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | | | | | | | | | | | | | | | | **RG_EKGADC_PD** |
| **Type** | | | | | | | | | | | | | | | | RW |
| **Reset** | | | | | | | | | | | | | | | | 1 |

| Bit(s) | Name | Description |
|---|---|---|
| 20:18 | RG_EKGADC_MODESEL | **EKGADC Mode Select:**<br>000->fs=64Hz,<br>001->fs=128Hz,<br>010->fs=256Hz,<br>011->fs=512 or 1024 or 2048 or 4096 Hz<br>Set corresponding RG_EKG_DIG_MODESEL at the same time. |
| 0 | RG_EKGADC_PD | **EKGADC power down:**<br>1->power down,<br>0->power on |

**00000064 AFE_EKG_CON EKG Digital Part Control 0000000B**

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Name** | | | | | | | | | | | | | | | | |
| **Type** | | | | | | | | | | | | | | | | |
| **Reset** | | | | | | | | | | | | | | | | |
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | | | | | | | | | | | **RG_EKG_FIFO** | | | **RG_EKG_DIG_MODESEL** | | |
| **Type** | | | | | | | | | | | RW | | | RW | | |
| **Reset** | | | | | | | | | | | 0 | 0 | 1 | 0 | 1 | 1 |

| Bit(s) | Name | Description |
|---|---|---|
| 5:3 | **RG_EKG_FIFO** | **EKG FIFO start point** |
| 2:0 | **RG_EKG_DIG_MODESEL** | **mode selection in EKG:**<br>3b'000->fs=64Hz,<br>3b'001->fs=128Hz,<br>3b'010->fs=256Hz, |

| Bit(s) | Name | Description |
|---|---|---|
| | | 3b'011->fs=512Hz,<br>3b'100->fs=1024Hz,<br>3b'101->fs=2048Hz,<br>3b'110->fs=4096Hz.<br>Set corresponding RG_EKGADC_MODESEL at the same time. |

**000000C0 AFE_MEM_CON0 EKG SRAM Control 0 00000000**

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | UPSRAM_CE | UPSRAM_RW_ADDR_AUTO_INC | UPSRAM_ATRIG | | UPSRAM_WTRIG | UPSRAM_RTRIG | UPSRAM_CUR_ADDR | | | | | | | | | |
| Type | RW | RW | RW | | RO | RO | RO | | | | | | | | | |
| Reset | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Name | | | | | | | UPSRAM_RW_ADDR | | | | | | | | | |
| Type | | | | | | | RW | | | | | | | | | |
| Reset | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit(s) | Name | Description |
|---|---|---|
| 31 | UPSRAM_CE | **EKG SRAM chip select** |
| 30 | UPSRAM_RW_ADDR_AUTO_INC | **Automatically increase I2C/SPI access EKG SRAM address**<br>0: no effect<br>1: automatically increase |
| 29 | UPSRAM_ATRIG | **Enable access EKG SRAM from I2C/SPI**<br>0: no effect<br>1: toggle access trigger |
| 27 | UPSRAM_WTRIG | **Write EKG SRAM from I2C/SPI**<br>0: no effect<br>1: write trigger |
| 26 | UPSRAM_RTRIG | **Read EKG SRAM from I2C/SPI**<br>0: no effect<br>1: read trigger |
| 25:16 | UPSRAM_CUR_ADDR | **The current access address of EKG SRAM from I2C/SPI** |
| 9:0 | UPSRAM_RW_ADDR | **The access address of EKG SRAM from I2C/SPI** |

### 000000C4 AFE_MEM_CON1 EKG SRAM Control 1 00000000

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | UPSRAM_WR_DATA[31:16] | | | | | | | | | | | | | | | |
| Type | RW | | | | | | | | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Name | UPSRAM_WR_DATA[15:0] | | | | | | | | | | | | | | | |
| Type | RW | | | | | | | | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit(s) | Name | Description |
|---|---|---|
| 31:0 | UPSRAM_WR_DATA | The write data from I2C/SPI to EKG SRAM |

### 000000C8 AFE_MEM_CON2 EKG SRAM Control 2 00000000

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | UPSRAM_RD_DATA[31:16] | | | | | | | | | | | | | | | |
| Type | RO | | | | | | | | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Name | UPSRAM_RD_DATA[15:0] | | | | | | | | | | | | | | | |
| Type | RO | | | | | | | | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit(s) | Name | Description |
|---|---|---|
| 31:0 | UPSRAM_RD_DATA | The read data from EKG SRAM to I2C/SPI |

### 000000CC AFE_MEM_CON3 EKG SRAM Control 3 00000000

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | | | | | | | FILTER_CUR_ADDR | | | | | | | | | |
| Type | | | | | | | RO | | | | | | | | | |
| Reset | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Name | | | | | | | | RG_IRG_TH | | | | | | | | |
| Type | | | | | | | | RW | | | | | | | | |
| Reset | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit(s) | Name | Description |
|---|---|---|
| 25:16 | FILTER_CUR_ADDR | The current access address of EKG SRAM from EKG FIFO |
| 8:0 | RG_IRG_TH | The threshold of EKG SRAM to set irq signal |

## 7.3 Programming Guide

The setting value of EKGFE_CON0 (0x08), EKGADC_CON0 (0x10), and AFE_EKG_CON (0x64) should be set before enabling EKG channel.

1. Setting the value to EKGFE_CON0 (0x08), EKGADC_CON0 (0x10), and AFE_EKG_CON (0x64)
2. Enable EKG channel (setting AFE_DIG_ENABLE (0x60) = 0x418)

**Read/Write EKG SRAM from I2C/SPI**

1. READ:

   a. Setting AFE_MEM_CON0 (0xC0) = 0x60000000 | start_addr[9:0]

   b. HW will put read_data from EKG SRAM to AFE_MEM_CON2 (0xC8)

2. WRITE:

   a. Setting AFE_MEM_CON0 (0xC0) = 0x60000000 | start_addr[9:0]

   b. Setting AFE_MEM_CON1 (0xC4) = write_data[31:0]

   c. HW will write write_data to EKG SRAM

# 8 Heartbeat Interval Estimation

## 8.1 General Description

MT2511 has a built-in heartbeat interval (BI) detector to reduce power consumption of hear-beat detection. When the signal quality of PPG is good enough, the BI detector is able to approximate wavelength of major tone. Instead of complete PPG signals, only beat time intervals are recorded in SRAM. The MCU is freed from estimating heart rate; as a result the amount of data read from MT2511 SRAM via I2C/SPI is reduced significantly.

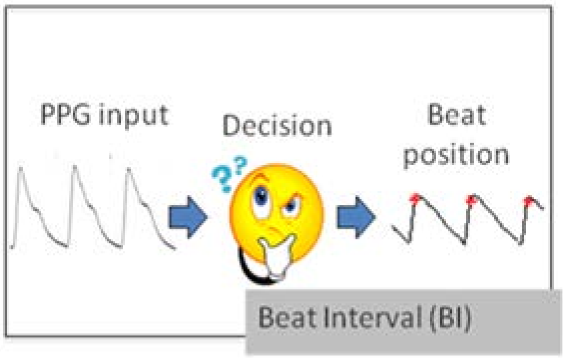


*Figure 8-1. Heartbeat Detection Flow*

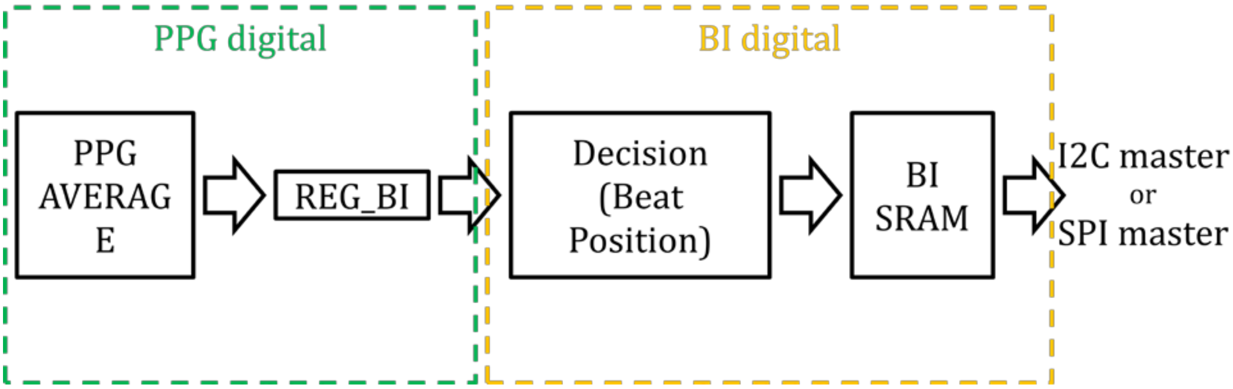


***Figure** 8-2. **The Built-in Heartbeat Interval Detector Block Diagram***

## 8.2 Heartbeat Interval Estimation Register Definitions

**BASE name: top_reg**

| Address | Name | Width | Register Function |
|---|---|---|---|
| 000000F0 | **AFE_MEM_CON12** | 32 | **BI SRAM Control 0** |
| 000000F4 | **AFE_MEM_CON13** | 32 | **BI SRAM Control 1** |
| 000000F8 | **AFE_MEM_CON14** | 32 | **BI SRAM Control 2** |
| 000000FC | **AFE_MEM_CON15** | 32 | **BI SRAM Control 3** |

**000000F0** **AFE_MEM_CON12** **00000000**

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Name** | UPSRAM_CE | UPSRAM_RW_ADDR_AUTO_INC | UPSRAM_ATRIG | | UPSRAM_WTRIG | UPSRAM_RTRIG | UPSRAM_CUR_ADDR | | | | | | | | | |
| **Type** | RW | RW | RW | | RO | RO | RO | | | | | | | | | |
| **Reset** | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | | | | | | | UPSRAM_RW_ADDR | | | | | | | | | |
| **Type** | | | | | | | RW | | | | | | | | | |
| **Reset** | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit(s) | Name | Description |
|---|---|---|
| 31 | UPSRAM_CE | **BI SRAM chip select** |
| 30 | UPSRAM_RW_ADDR_AUTO_INC | **Automatically increase I2C/SPI access BI SRAM address**<br>0: no effect<br>1: automatically increase |
| 29 | UPSRAM_ATRIG | **Enable access BI SRAM from I2C/SPI**<br>0: no effect<br>1: toggle access trigger |
| 27 | UPSRAM_WTRIG | **Write BI SRAM from I2C/SPI**<br>0: no effect<br>1: write trigger |
| 26 | UPSRAM_RTRIG | **Read BI SRAM from I2C/SPI**<br>0: no effect<br>1: read trigger |
| 25:16 | UPSRAM_CUR_ADDR | **The current access address of BI SRAM from I2C/SPI** |
| 9:0 | UPSRAM_RW_ADDR | **The access address of BI SRAM from I2C/SPI** |

### 000000F4 AFE_MEM_CON13 00000000

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | UPSRAM_WR_DATA[31:16] | | | | | | | | | | | | | | | |
| Type | RW | | | | | | | | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Name | UPSRAM_WR_DATA[15:0] | | | | | | | | | | | | | | | |
| Type | RW | | | | | | | | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit(s) | Name | Description |
|---|---|---|
| 31:0 | UPSRAM_WR_DATA | The write data from I2C/SPI to BI SRAM |

### 000000F8 AFE_MEM_CON14 00000000

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | UPSRAM_RD_DATA[31:16] | | | | | | | | | | | | | | | |
| Type | RO | | | | | | | | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Name | UPSRAM_RD_DATA[15:0] | | | | | | | | | | | | | | | |
| Type | RO | | | | | | | | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit(s) | Name | Description |
|---|---|---|
| 31:0 | UPSRAM_RD_DATA | The read data from BI SRAM to I2C/SPI |

**000000FC AFE_MEM_CON15 00000000**

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Name** | | | | | | | **FILTER_CUR_ADDR** | | | | | | | | | |
| **Type** | | | | | | | RO | | | | | | | | | |
| **Reset** | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **Bit** | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | | | | | | | | **RG_IRG_TH** | | | | | | | | |
| **Type** | | | | | | | | RW | | | | | | | | |
| **Reset** | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit(s) | Name | Description |
|---|---|---|
| 25:16 | FILTER_CUR_ADDR | The current access address of BI SRAM from BI FIFO |
| 8:0 | RG_IRG_TH | The threshold of BI SRAM to set irq signal |

## 8.3 Programming Guide

**Enable BI feature**:

1. Setting AFE_DIG_ENABLE (0x60) = 0x585

   Read/Write BI SRAM from I2C/SPI

2. READ:

   a. Setting AFE_MEM_CON12 (0xF0) = 0x60000000 | start_addr[9:0]

   b. HW will put read_data from BI SRAM to AFE_MEM_CON14 (0xF8)

3. WRITE:

   a. Setting AFE_MEM_CON12 (0xF0) = 0x60000000 | start_addr[9:0]

   b. Setting AFE_MEM_CON13 (0xF4) = write_data[31:0]

   c. HW will write write_data to BI SRAM