Datasheet for Telink

ULP 2.4 GHz RF SoC TLSR8359

DS-TLSR8359-E8

Keyword

2.4 GHz; Memory; MCU; Working modes; Wakeup sources; RF Transceiver; Clock; Timers; Interrupt; Interface; PWM; QDEC; ADC; PGA; Temperature sensor; Low power comparator; AES

Brief

This datasheet is dedicated for Telink ULP (Ultra-low Power) 2.4 GHz RF SoC TLSR8359 (VID: 0x02). In this datasheet, key features, working mode, main modules, electrical specifications and application of the TLSR8359 are introduced.

Published by

Telink Semiconductor

Bldg 3, 1500 Zuchongzhi Rd,

Zhangjiang Hi-Tech Park, Shanghai, China

© Telink Semiconductor

All Right Reserved

Legal Disclaimer

This document is provided as-is. Telink Semiconductor reserves the right to make improvements without further notice to this document or any products herein. This document may contain technical inaccuracies or typographical errors. Telink Semiconductor disclaims any and all liability for any errors, inaccuracies or incompleteness contained herein.

Copyright © 2023 Telink Semiconductor (Shanghai) Co., Ltd.

Information

For further information on the technology, product and business term, please contact Telink Semiconductor Company (www.telink-semi.com).

For sales or technical support, please send email to the address of:

telinksales@telink-semi.com

telinksupport@telink-semi.com

Revision History

Version	Change Description
0.8.0	Preliminary release
0.8.1	• Updated sections 1.2.2 RF Features, 1.2.3 Features of Power Management Module, 2.1.2 Flash, 2.1.3 E-Fuse, 2.5.1 Power-On-Reset (POR) and Brown-Out Detect, 15.3 DC Characteristics, and 15.4 AC Characteristics • Added sections 15.5 SPI Characteristics and 15.6 I2C Characteristics
0.8.2	• Updated sections 1.3 Typical Applications, 1.4 Ordering Information, 7.3.4 I2C Master Mode, 15.1 Absolute Maximum Ratings, 15.2 Recommended Operating Conditions, and 16 Reference Design • Added Section 15.7 Flash Characteristics
0.8.3	• Updated sections 1.1 Block Diagram, 1.2 Key Features, 1.4 Ordering Information, 1.6 Pin Layout, 2.1 Memory, 2.3 Working Modes, 2.4 Reset, 2.5.1 Power-On-Reset (POR) and Brown-Out Detect, 4.2 System Clock, 4.4 Register Table, 7.1 GPIO, 9.1 Input Pin Selection, 10 SAR ADC, 12 Temperature Sensor, 15.4 AC Characteristics, and 16 Reference Design • Added Section 2.5.4 VBAT and VANT Power-Supply Mode
0.8.4	Updated sections 1.6 Pin Layout, 2.1.3 E-Fuse, 5.1.1 Register Table, 8.1 Register Table, 9.4 QDEC Reset, 9.7 Register Table, 10.3.2 "Set" State, and 14.4 Register Table
0.8.5	• An overall update on formatting • Updated UID related information: Revised UID related description in 1.2.1 General Features, removed related description in Section 2.1.2 Flash, updated related description in Section 2.1.3 E-Fuse and added Section 2.1.4 Unique ID • Removed firmware encryption related description in Section 1.2.5 Flash Features and the original Section 2.2 Firmware Encryption • Section 1.4 Ordering Information: Added ordering information for TLSR8359F512ET48 in Table 1-1 Ordering Information of TLSR8359* • Section 1.5 Package: Added package information for TLSR8359F512ET48 in description

Version	Change Description
	and Figure 1-2 Package Dimension for TLSR8359F512ET48/TLSR8359ET48 (Unit: mm) · Added Section 1.6.1 Pin Layout for TLSR8359F512ET48 and updated Section 1.6.5 Notes · Section 7.1 GPIO: Added GPIO information for TLSR8359F512ET48 · Section 7.1.1.1 GPIO Lookup Table: Added pins PA<2>, PA<3>, PA<4>, and PD<6> to Table 7-1 GPIO Lookup Table 1 · Section 7.1.1.2 Multiplexed Functions: Added DI/UART_CTS/PWM1/PA<3> and CK/UART_RTS/PWM2/PA<4> to Table 7-2 Select Multiplexed SPI/I2C · Section 7.1.2 Connection Relationship Between GPIO and Related Modules: Added pins PA<2>, PA<3>, PA<4>, and PD<6> to Table 7-3 GPIO Lookup Table 2 · Section 7.1.3 Pull-Up/Pull-Down Resistor: Removed RSVD from description of pins PA<2>, PA<3>, PA<4>, and PD<6> in Table 7-4 Analog Registers for Pull-Up/Pull-Down Resistor Control · Section 9.1 Input Pin Selection: Removed RSVD for pins PA<2>, PA<3>, and PD<6> in Table 9-1 Input Pin Selection · Section 15.4 AC Characteristics: Revised RF frequency range in Table 15-4 AC Characteristics (VDD = 3.3 V, T = 25°C) · Added Section 16.1 Application Example for TLSR8359F512ET48 · Updated Table 15-7 Flash Memory Characteristics, Figure 16-1 Schematic for TLSR8359F512ET48, Figure 16-2 Schematic for TLSR8359ET48, Figure 16-3 Schematic for TLSR8359F512ET32
0.8.6	· For SPI interface, change full-duplex to half-duplex. · For 2.4GHz proprietary payload length, change to 1~63 bytes.
0.8.7	· Updated the note for the reference schematic.

Table of Contents

List of Figures

List of Tables

1. Overview

The RoHS-compliant TLSR8359 series with internal Flash is dedicated to 2.4 GHz RF System-On-Chip solution, such as retail/logistics, private network, beacon applications, and etc.

The TLSR8359 has hardware OTA upgrades support and multiple boot switching, allowing convenient product feature roll outs and upgrades.

1.1 Block Diagram

The TLSR8359 is designed to offer high integration, ultra-low power application capabilities.

The TLSR8359 series integrates a power-balanced 32-bit MCU, 2.4 GHz Radio, 64 KB SRAM, 512 KB (TLSR8359F512) / 64 KB (TLSR8359F64) internal flash, 14-bit ADC with PGA, 6-channel PWM (1-channel IR/IR FIFO/IR DMA FIFO), one quadrature decoder (QDEC), abundant and flexible GPIO interfaces, and nearly all the peripherals needed for 2.4 GHz RF System-On-Chip application development.

The TLSR8359 also includes multi-stage power management design allowing ultra-low power operation and making it the ideal candidate for power-constraint applications.

The system's block diagram is as shown in Figure 1-1.

Figure 1-1 Block Diagram of the System
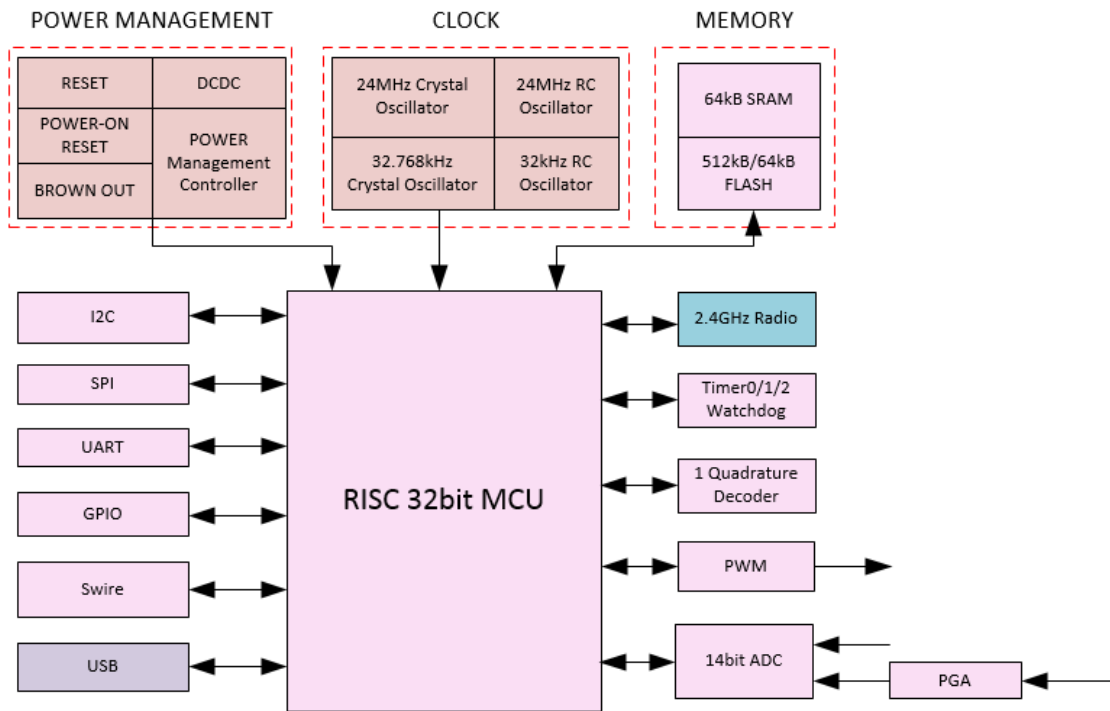
NOTE:

- Modules marked with different colors belong to different power domains. Power state of each power domain can be controlled independent of other power domains, for example, the Radio module can be independently powered on or powered down irrespective of other modules such as power management module, clock, and etc.

- The 2.4 GHz Radio and USB are powered down by default.

- The power management module and clock should be always powered on, even in deep sleep.

- In deep sleep, except for the power management and clock, all other modules should be powered down.

With the high integration level of TLSR8359, few external components are needed to satisfy customers' ultra-low cost requirements.

1.2 Key Features

1.2.1 General Features

General features are as follows:

1.   Support Unique ID (UID)

2. Embedded 32-bit proprietary microcontroller

  · Better power-balanced performance than ARM M0

  · Instruction cache controller

  · Maximum running speed up to 48 MHz

3. Program memory:

  · Internal 512 KB (TLSR8359F512)/64 KB (TLSR8359F64) flash

  · The TLSR8359ET48 has no internal flash but supports up to 16 MB external flash

4. Data memory: 64 KB on-chip SRAM, including up to 32 KB SRAMs with retention in deep sleep, and one 32 KB SRAM without retention in deep sleep

5. RTC and other timers:

  · Clock sources: 24 MHz & 32.768 kHz crystals and 24 MHz & 32 kHz embedded RC oscillators

  · Three general 32-bit timers with four selectable modes in active mode

  · Watchdog timer

  · A low-frequency 32 kHz timer available in low power mode

6. A rich set of I/Os:

  · Up to 32/17/2 GPIOs depending on package option. All digital IOs can be used as GPIOS.

  · SPI

  · I2C

  · UART with hardware flow control support

  · USB

  · Swire debug Interface

  · MSPI for external flash (only TLSR8359ET48)

7. Up to 6 channels of differential PWM:

  · PWM1~PWM5: 5-channel normal PWM output

  · PWM0: 1 channel with five modes support, including IR/IR FIFO/IR DMA FIFO mode for IR generation

8. Sensor:

  · 14-bit 10-channel (only GPIO input) SAR ADC, with 4-channel differential input PGA

  · Temperature sensor

9. One quadrature decoder

10. Embedded hardware AES and AES-CCM

11. Embedded low power comparator

12. Operating temperature range: -40°C ~ +85°C

1.2.2 RF Features

RF features include:

1. 2.4 GHz RF transceiver embedded, working in worldwide 2.4 GHz ISM band

2. 2.4 GHz proprietary 1 Mbps/2 Mbps/250 kbps/500 kbps mode with Adaptive Frequency Hopping feature support

3. RX sensitivity: -96 dBm @ 1 Mbps, -93 dBm @ 2 Mbps

4. TX output power: up to +10 dBm

5. 50 Ω matched single-pin antenna interface

6. RSSI monitoring with +/-1 dB resolution

7. Auto acknowledgement, retransmission and flow control

1.2.3 Features of Power Management Module

Features of power management module include:

1. Embedded LDO and DCDC

2. Battery monitor: Supports low battery detection

3. Power supply: 1.8 V ~ 3.6 V

4. Multiple stage power management to minimize power consumption

5. Low power consumption:

    · Whole chip RX mode: 5.3 mA

    · Whole chip TX mode: 4.8 mA @ 0 dBm with DCDC

    · Deep sleep with external wakeup (without SRAM retention): 0.4 µA

    · Deep sleep with SRAM retention: 1 µA (with 8KB SRAM retention), 1.2 µA (with 16 KB SRAM retention), 1.4 µA (with 32 KB SRAM retention)

1.2.4 USB Features

USB features include:

1. Compatible with USB2.0 Full speed mode

2. Support 9 endpoints including control endpoint 0 and 8 configurable data endpoints

3. Independent power domain

4. Support ISP (In-System Programming) via USB port

1.2.5 Flash Features

The TLSR8359F512/TLSR8359F64 embeds Flash with features below:

1. TLSR8359F512: Total 512 KB (4 Mbits)

2. TLSR8359F64: Total 64 KB (0.5 Mbits)

3. Flexible architecture: 4 KB per Sector, 64 KB/32 KB per block

4. Up to 256 bytes per programmable page

5. Write protect all or portions of memory

6. Sector erase (4 KB)

7. Block erase (32 KB/64 KB)

8. Cycle Endurance: 100,000 program/erases

9. Data Retention: typical 20-year retention

1.3 Typical Applications

The TLSR8359 is dedicated to 2.4 GHz RF System-On-Chip solution.

Its typical applications include, but are not limited to the following:

- Retail/logistics
- Private network
- Beacon

1.4 Ordering Information

Table 1-1 Ordering Information of TLSR8359[1]

Product Series	Package Type	Temperature Range	Product Part No.	Packing Method[2]	Minimum Order Quantity
TLSR8359F512	48-pin 7x7 mm TQFN	-40°C ~ +85°C	TLSR8359F512 ET48	TR	3000
	32-pin 5x5 mm TQFN	-40°C ~ +85°C	TLSR8359F512 ET32	TR	3000
TLSR8359F64	16-pin TSSOP16_ 4.96x6.4 mm	-40°C ~ +85°C	TLSR8359F64ES16	TR	5000
TLSR8359	48-pin 7x7 mm TQFN	-40°C ~ +85°C	TLSR8359ET48	TR	3000

---

[1] MSL (Moisture Sensitivity Level)： The 8359 series is applicable to MSL3 (Based on JEDEC Standard J-STD-020).

·   After the packing opened, the product shall be stored at <30°C/ <60%RH and the product shall be used within 168 hours.

·   When the color of the indicator in the packing changed, the product shall be baked before soldering.

·   If baking is required, please refer to IPC/JEDEC J-STD-033 for baking procedure.

[2] Packing method "TR" means tape and reel. The tape and reel material DO NOT support baking under high temperature.

1.5 Package

Package dimensions for the TLSR8359F512ET48, TLSR8359ET48, TLSR8359F512ET32 and TLSR8359F64ES16 are shown as below.

Figure 1-2 Package Dimension for TLSR8359F512ET48/TLSR8359ET48 (Unit: mm)
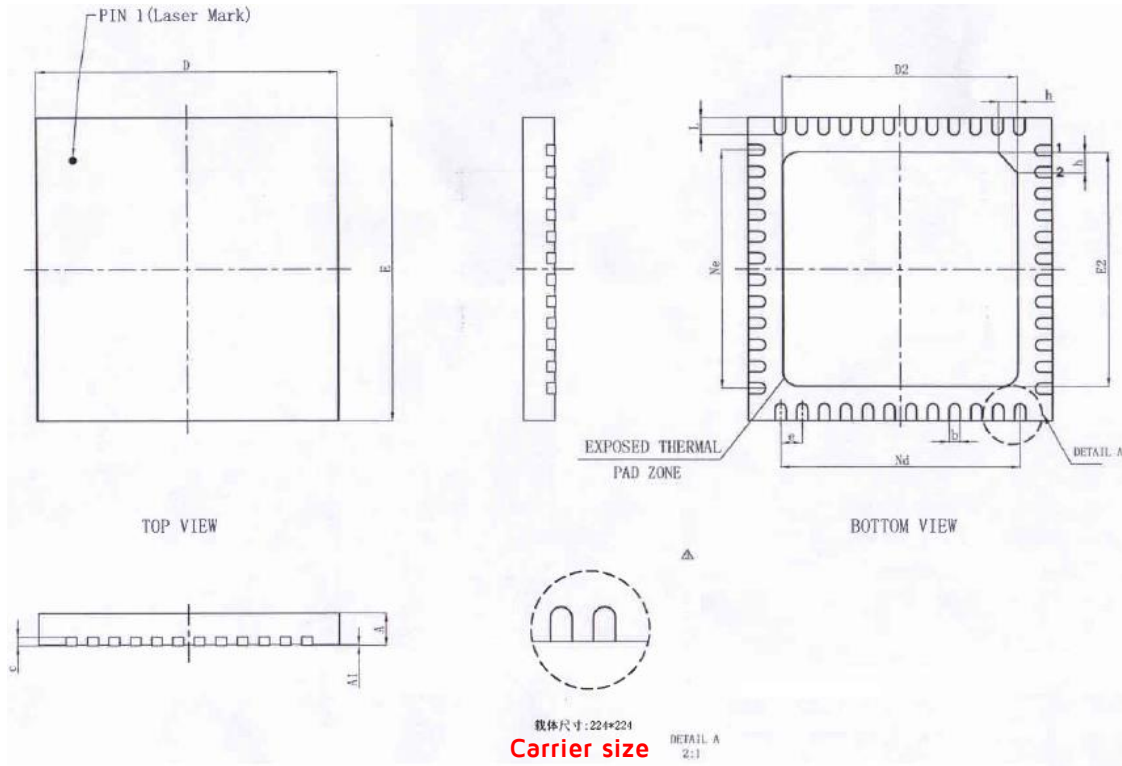
SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	0.70	0.75	0.80
	0.85	0.90	0.95
A1	0	0.02	0.05
b	0.18	0.25	0.30
b1	0.11	0.16	0.21
c	0.18	0.20	0.23
D	6.90	7.00	7.10
D2	5.30	5.40	5.50
e	0.50BSC		
Ne	5.50BSC		
Nd	5.50BSC		
E	6.90	7.00	7.10
E2	5.30	5.40	5.50
L	0.35	0.40	0.45
h	0.30	0.35	0.40
h1	0.03REF		
h2	0.10REF		

L/F carrier size

L/F载体尺寸	SYMBOL	MILLIMETER
224*224	D2	5.40±0.10
	E2	5.40±0.10

Figure 1-3 Package Dimension for TLSR8359F512ET32 (Unit: mm)

BOTTOM VIEW

EXPOSED THERMAL PAD ZONE
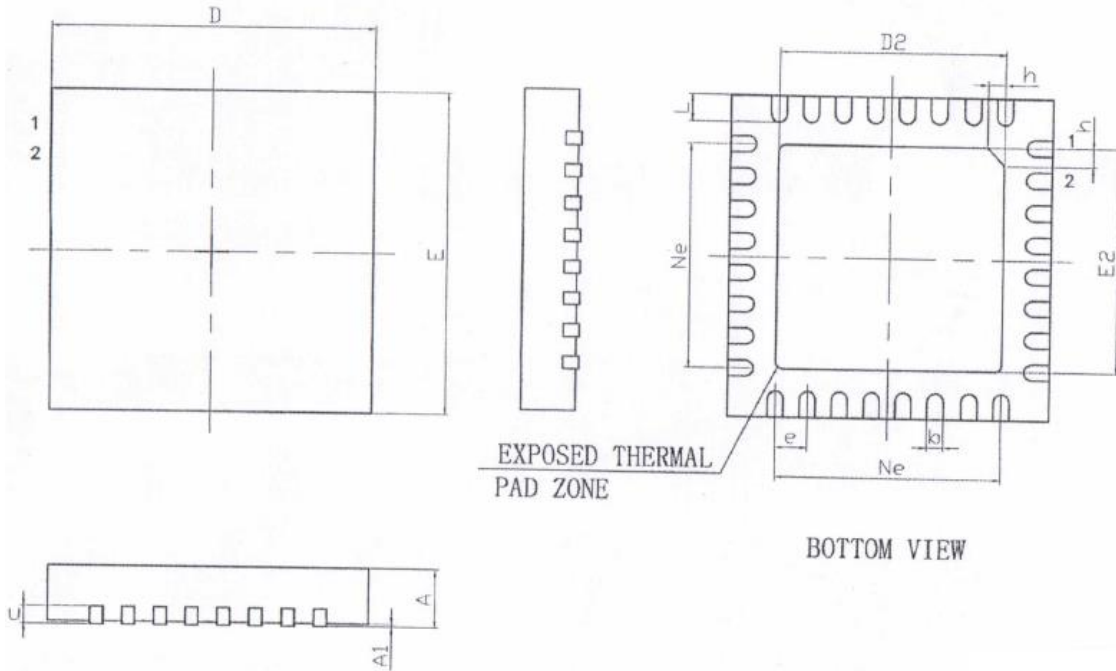
SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	0.70	0.75	0.80
A1	—	0.02	0.05
b	0.18	0.25	0.30
c	0.18	0.20	0.25
D	4.90	5.00	5.10
D2	3.40	3.50	3.60
e	0.50BSC		
Ne	3.50BSC		
E	4.90	5.00	5.10
E2	3.40	3.50	3.60
L	0.35	0.40	0.45
h	0.30	0.35	0.40
L/F载体尺寸	150x150	130x130	

L/F carrier size

Figure 1-4 Package Dimension for TLSR8359F64ES16 (Unit: mm)
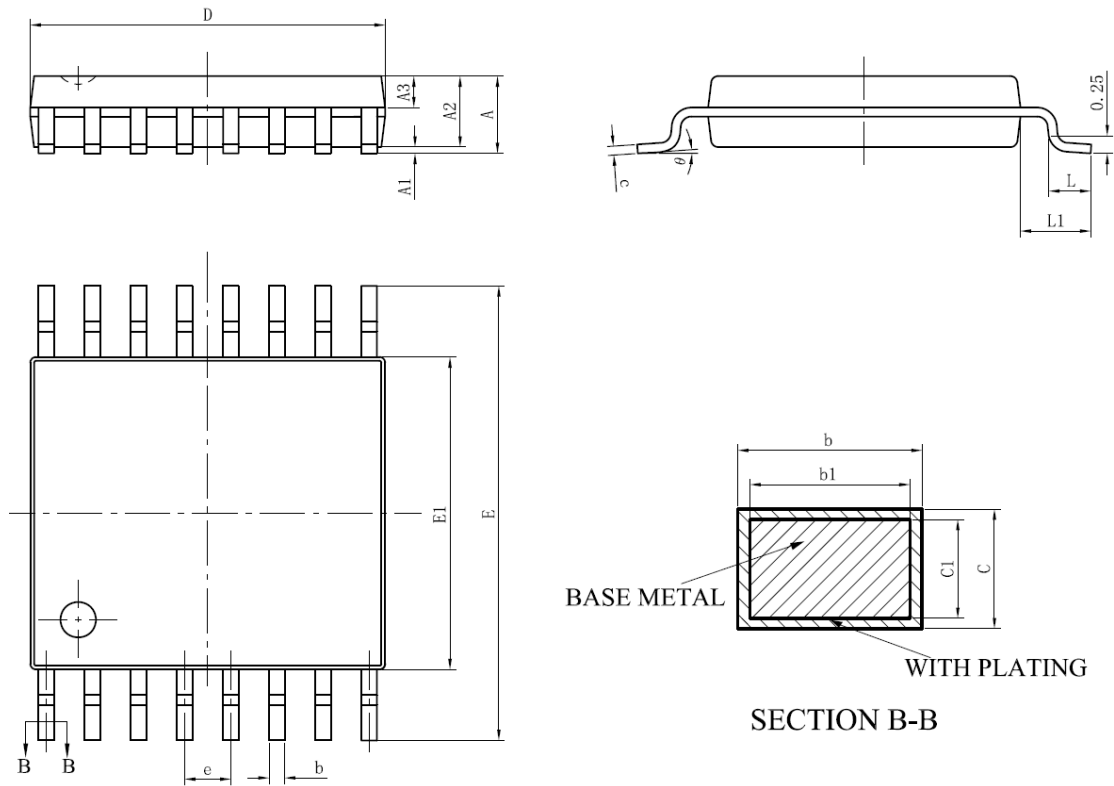
SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	—	—	1.20
A1	0.05	—	0.15
A2	0.90	1.00	1.05
A3	0.39	0.44	0.49
b	0.20	—	0.29
b1	0.19	0.22	0.25
c	0.13	—	0.18
c1	0.12	0.13	0.14
D	4.86	4.96	5.06
E	6.20	6.40	6.60
E1	4.30	4.40	4.50
e	0.65BSC		
L	0.45	0.60	0.75
L1	1.00BSC		
θ	0	—	8°

1.6 Pin Layout

1.6.1 Pin Layout for TLSR8359F512ET48

Figure 1-5 shows pin assignment for the TLSR8359F512ET48.

Figure 1-5 Pin Assignment for TLSR8359F512ET48
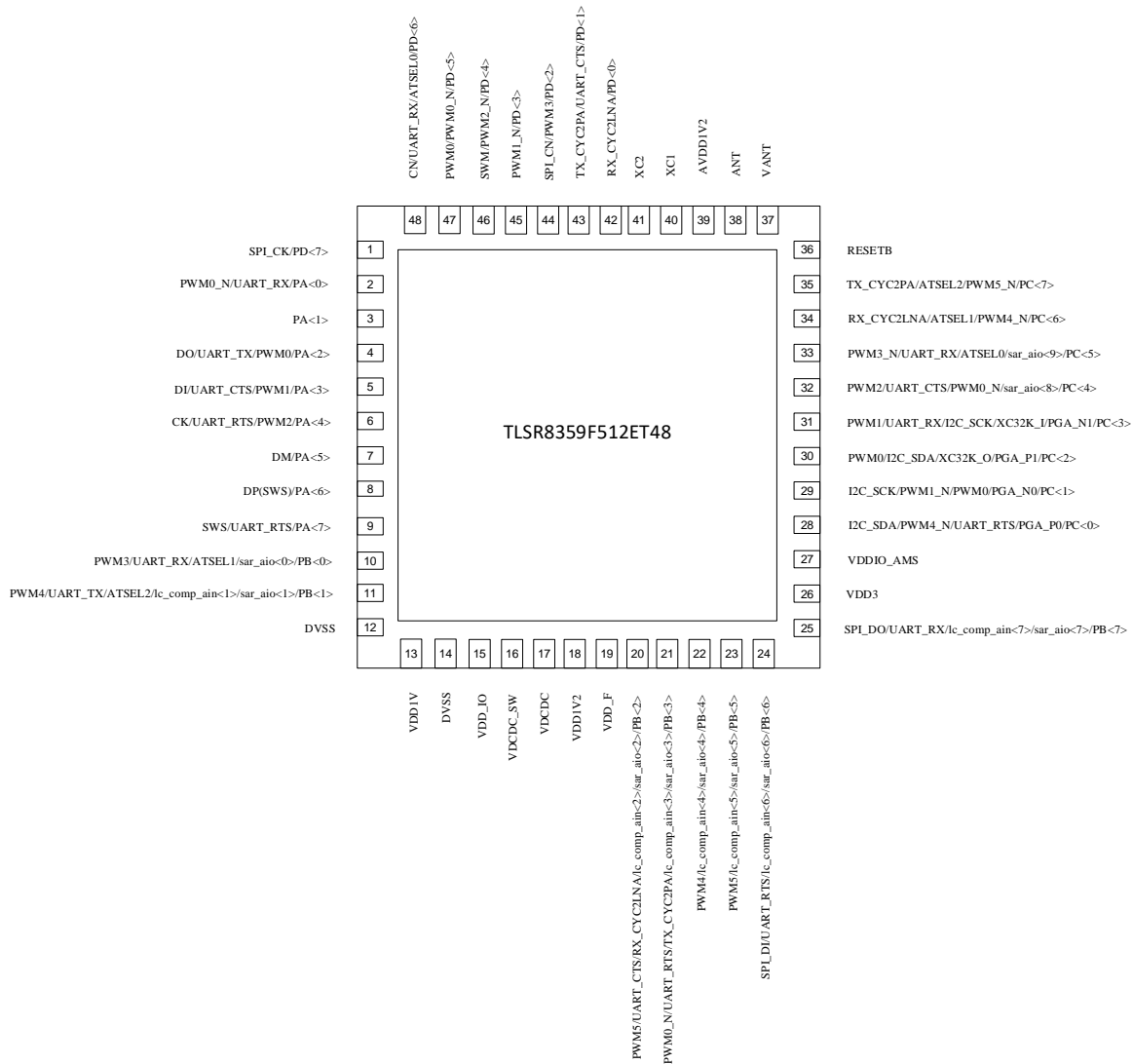
Functions of 48 pins for the TLSR8359F512ET48 are described in Table 1-2:

Table 1-2 Pin Functions for TLSR8359F512ET48

No.	Pin Name	Type	Description
1	SPI_CK/ PD<7>	Digital I/O	SPI clock (I2C_SCK) / GPIO PD[7]
2	PWM0_N/UART_RX/PA<0>	Digital I/O	PWM0 inverting output / UART_RX / GPIO PA[0]
3	PA<1>	Digital I/O	GPIO PA[1]
4	DO/UART_TX/PWM0/PA<2>	Digital I/O	SPI data output / UART_TX / PWM0 output / GPIO PA[2]
5	DI/UART_CTS/PWM1/PA<3>	Digital I/O	SPI data input (I2C_SDA) / UART_CTS / PWM1 output / GPIO PA[3]
6	CK/UART_RTS/PWM2/PA<4>	Digital I/O	SPI clock (I2C_SCK) / UART_RTS / PWM2 output / GPIO PA[4]
7	DM/PA<5>	Digital I/O	USB data minus / GPIO PA[5]
8	DP(SWS)/PA<6>	Digital I/O	USB data positive (Single wire slave) / GPIO PA[6]
9	SWS/UART_RTS/PA<7>	Digital I/O	Single wire slave/ UART_RTS / GPIO PA[7]
10	PWM3/UART_RX/ATSEL1/ sar_aio<0>/PB<0>	Digital I/O	PWM3 output / UART_RX / Antenna select pin 1 / SAR ADC input / GPIO PB[0]
11	PWM4/UART_TX/ATSEL2/lc_comp_ain<1>/sar_aio<1>/PB<1>	Digital I/O	PWM4 output / UART_TX / Antenna select pin 2 / Low power comparator input / SAR ADC input / GPIO PB[1]
12	DVSS	GND	Digital LDO ground
13	VDD1V	PWR	Internal LDO generated power supply input for digital core
14	DVSS	GND	Digital LDO ground
15	VDD_IO	PWR	External 3.3V power supply input for IO
16	VDCDC_SW	Analog	Connect with VDCDC via external inductor
17	VDCDC	Analog	Connect with VDCDC_SW via external inductor
18	VDD1V2	PWR	Internal DCDC generated power supply. Connect to GND via external capacitor. Route this 1.2V voltage power supply to AVDD1V2.

No.	Pin Name	Type	Description
19	VDD_F	PWR	Internally generated power supply to flash. Connect to GND via external capacitor.
20	PWM5/UART_CTS/RX_CYC2LNA/ lc_comp_ain<2>/sar_aio<2>/ PB<2>	Digital I/O	PWM5 output / UART_CTS / Control external LNA / Low power comparator input / SAR ADC input / GPIO PB[2]
21	PWM0_N/UART_RTS/TX_CYC2PA/ lc_comp_ain<3>/sar_aio<3>/ PB<3>	Digital I/O	PWM0 inverting output / UART_RTS / Control external PA / Low power comparator input / SAR ADC input / GPIO PB[3]
22	PWM4/lc_comp_ain<4>/sar_aio<4>/ PB<4>	Digital I/O	PWM4 output / Low power comparator input / SAR ADC input / GPIO PB[4]
23	PWM5/lc_comp_ain<5>/sar_aio<5>/ PB<5>	Digital I/O	PWM5 output / Low power comparator input / SAR ADC input / GPIO PB[5]
24	SPI_DI/UART_RTS/ lc_comp_ain<6>/sar_aio<6>/ PB<6>	Digital I/O	SPI data input (I2C_SDA) / UART_RTS / Low power comparator input / SAR ADC input / GPIO PB[6]
25	SPI_DO/UART_RX/ c_comp_ain<7> /sar_aio<7>/PB<7>	Digital I/O	SPI data output / UART_RX / Low power comparator input / SAR ADC input / GPIO PB[7]
26	VDD3	PWR	Connect to an external 3.3V power supply
27	VDDIO_AMS	PWR	External 3.3V power supply input for analog/mixed signal circuitry.
28	I2C_SDA/PWM4_N/UART_RTS/ PGA_P0/PC<0>	Digital I/O	I2C serial data / PWM4 inverting output / UART_RTS / PGA left channel positive input / GPIO PC[0]
29	I2C_SCK/PWM1_N/PWM0/ PGA_N0/PC<1>	Digital I/O	I2C serial clock / PWM1 inverting output / PWM0 output / PGA left channel negative input / GPIO PC[1]
30	PWM0/I2C_SDA/XC32K_O/PGA_P1/ PC<2>	Digital I/O	PWM0 output / I2C serial data / (optional) 32 kHz crystal output / PGA right channel positive input / GPIO PC[2]
31	PWM1/UART_RX/I2C_SCK/XC32K_I/	Digital I/O	PWM1 output / UART_RX / I2C serial clock / (optional)

No.	Pin Name	Type	Description
	PGA_N1/PC<3>		32kHz crystal input / PGA right channel negative input / GPIO PC[3]
32	PWM2/UART_CTS/PWM0_N/ sar_aio<8>/PC<4>	Digital I/O	PWM2 output / UART_CTS / PWM0 inverting output / SAR ADC input / GPIO PC[4]
33	PWM3_N/UART_RX/ATSEL0/ sar_aio<9>/PC<5>	Digital I/O	PWM3 inverting output / UART_RX / Antenna select pin 0 / SAR ADC input / GPIO PC[5]
34	RX_CYC2LNA/ATSEL1/PWM4_N/ PC<6>	Digital I/O	Control external LNA / Antenna select pin 1 / PWM4 inverting output / GPIO PC[6]
35	TX_CYC2PA/ATSEL2/PWM5_N/ PC<7>	Digital I/O	Control external PA / Antenna select pin 2 / PWM5 inverting output / GPIO PC[7]
36	RESETB	RESET	Power on reset, active low
37	VANT	PWR	Internally generated voltage output to bias internal PA. Connect to ANT using an external inductor.
38	ANT	Analog	RF antenna, connect to VANT using external inductor
39	AVDD1V2	PWR	Power supply input for internal RF Modules. Route from VDD1V2. Connect to GND via external capacitor.
40	XC1	Analog	Connect 24 MHz crystal
41	XC2	Analog	Connect 24 MHz crystal
42	RX_CYC2LNA/PD<0>	Digital I/O	Control external LNA / GPIO PD[0]
43	TX_CYC2PA/UART_CTS/PD<1>	Digital I/O	Control external PA / UART_CTS / GPIO PD[1]
44	SPI_CN/ PWM3/PD<2>	Digital I/O	SPI chip select (Active low) / PWM3 output / GPIO PD[2]
45	PWM1_N/PD<3>	Digital I/O	PWM1 inverting output / GPIO PD[3]
46	SWM/PWM2_N/PD<4>	Digital I/O	Single wire master / PWM2 inverting output / GPIO PD[4]
47	PWM0/PWM0_N/PD<5>	Digital I/O	PWM0 output / PWM0 inverting output / GPIO PD[5]
48	CN/UART_RX/ATSEL0/PD<6>	Digital I/O	SPI chip select (Active low) / UART_RX / Antenna select pin 0 / GPIO PD[6]

1.6.2 Pin Layout for TLSR8359ET48

Figure 1-6 shows pin assignment for the TLSR8359ET48.

Figure 1-6 Pin Assignment for TLSR8359ET48

Functions of 48 pins for the TLSR8359ET48 are described in Table 1-3:

Table 1-3 Pin Functions for TLSR8359ET48

No.	Pin Name	Type	Description
1	PWM0/PWM0_N/PD<5>	Digital I/O	PWM0 output / PWM0 inverting output / GPIO PD[5]

No.	Pin Name	Type	Description
2	SPI_CK/UART_TX/PD<7>	Digital I/O	SPI clock (I2C_SCK) / UART_TX / GPIO PD[7]
3	PWM0_N/UART_RX/PA<0>	Digital I/O	PWM0 inverting output / UART_RX / GPIO PA[0]
4	PA<1>	Digital I/O	GPIO PA[1]
5	DM/PA<5>	Digital I/O	USB data minus / GPIO PA[5]
6	DP(SWS)/PA<6>	Digital I/O	USB data positive (Single wire slave) / GPIO PA[6]
7	SWS/UART_RTS/PA<7>	Digital I/O	Single wire slave/ UART_RTS / GPIO PA[7]
8	PWM3/UART_RX/ATSEL1/ sar_aio<0>/PB<0>	Digital I/O	PWM3 output / UART_RX / Antenna select pin 1 / SAR ADC input / GPIO PB[0]
9	PWM4/UART_TX/ATSEL2/lc_comp_ain< 1>/sar_aio<1>/PB<1>	Digital I/O	PWM4 output / UART_TX / Antenna select pin 2 / Low power comparator input / SAR ADC input / GPIO PB[1]
10	DVSS	GND	Digital LDO ground
11	VDD1V	PWR	Internal LDO generated power supply input for digital core
12	DVSS	GND	Digital LDO ground
13	VDD_IO	PWR	External 3.3 V power supply input for IO
14	VDCDC_SW	Analog	Connect with VDCDC via external inductor
15	VDCDC	Analog	Connect with VDCDC_SW via external inductor
16	VDD1V2	PWR	Internal DCDC generated power supply. Connect to GND via external capacitor. Route this 1.2 V voltage power supply to AVDD1V2.
17	MSCN/PE<2>	Digital I/O	Memory SPI chip-select (Active low) / GPIO PE[2]
18	MSDI/PE<3>	Digital I/O	Memory SPI data input / GPIO PE[3]
19	VDD_F	PWR	Internally generated power supply to flash. Connect to GND via external capacitor.
20	PWM5/UART_CTS/RX_CYC2LNA/	Digital I/O	PWM5 output / UART_CTS / Control external LNA /

No.	Pin Name	Type	Description
	lc_comp_ain<2>/sar_aio<2>/ PB<2>		Low power comparator input / SAR ADC input / GPIO PB[2]
21	PWM0_N/UART_RTS/TX_CYC2PA/ lc_comp_ain<3>/sar_aio<3>/ PB<3>	Digital I/O	PWM0 inverting output / UART_RTS / Control external PA / Low power comparator input / SAR ADC input / GPIO PB[3]
22	PWM4/lc_comp_ain<4>/sar_aio<4>/PB <4>	Digital I/O	PWM4 output / Low power comparator input / SAR ADC input / GPIO PB[4]
23	PWM5/lc_comp_ain<5>/sar_aio<5>/PB <5>	Digital I/O	PWM5 output / Low power comparator input / SAR ADC input / GPIO PB[5]
24	SPI_DI/UART_RTS/lc_comp_ain<6>/sar_ aio<6>/PB<6>	Digital I/O	SPI data input (I2C_SDA) / UART_RTS / Low power comparator input / SAR ADC input / GPIO PB[6]
25	SPI_DO/UART_RX/lc_comp_ain<7>/sar_ aio<7>/PB<7>	Digital I/O	SPI data output / UART_RX / Low power comparator input / SAR ADC input / GPIO PB[7]
26	VDD3	PWR	Connect to an external 3.3 V power supply
27	VDDIO_AMS	PWR	External 3.3 V power supply input for analog/mixed signal circuitry.
28	I2C_SDA/PWM4_N/UART_RTS/ PGA_P0/PC<0>	Digital I/O	I2C serial data / PWM4 inverting output / UART_RTS / PGA left channel positive input / GPIO PC[0]
29	I2C_SCK/PWM1_N/PWM0/ PGA_N0/PC<1>	Digital I/O	I2C serial clock / PWM1 inverting output / PWM0 output / PGA left channel negative input / GPIO PC[1]
30	PWM0/UART_TX/I2C_SDA/XC32K_O/PG A_P1/PC<2>	Digital I/O	PWM0 output / UART_TX / I2C serial data / (optional) 32 kHz crystal output / PGA right channel positive input / GPIO PC[2]
31	PWM1/UART_RX/I2C_SCK/XC32K_I/PGA _N1/PC<3>	Digital I/O	PWM1 output / UART_RX / I2C serial clock / (optional) 32 kHz crystal input / PGA right channel negative input / GPIO PC[3]

No.	Pin Name	Type	Description
32	PWM2/UART_CTS/PWM0_N/ sar_aio<8>/PC<4>	Digital I/O	PWM2 output / UART_CTS / PWM0 inverting output / SAR ADC input / GPIO PC[4]
33	PWM3_N/UART_RX/ATSEL0/ sar_aio<9>/PC<5>	Digital I/O	PWM3 inverting output / UART_RX / Antenna select pin 0 / SAR ADC input / GPIO PC[5]
34	RX_CYC2LNA/ATSEL1/PWM4_N/ PC<6>	Digital I/O	Control external LNA / Antenna select pin 1 / PWM4 inverting output / GPIO PC[6]
35	TX_CYC2PA/ATSEL2/PWM5_N/ PC<7>	Digital I/O	Control external PA / Antenna select pin 2 / PWM5 inverting output / GPIO PC[7]
36	RESETB	RESET	Power on reset, active low
37	VANT	PWR	Internally generated voltage output to bias internal PA. Connect to ANT using an external inductor.
38	ANT	Analog	RF antenna, connect to VANT using external inductor
39	AVDD1V2	PWR	Power supply input for internal RF Modules. Route from VDD1V2. Connect to GND via external capacitor.
40	XC1	Analog	Connect 24 MHz crystal
41	XC2	Analog	Connect 24 MHz crystal
42	MSDO/PE<0>	Digital I/O	Memory SPI data output / GPIO PE[0]
43	MCLK/PE<1>	Digital I/O	Memory SPI clock / GPIO PE[1]
44	RX_CYC2LNA/UART_TX/PD<0>	Digital I/O	Control external LNA / UART_TX / GPIO PD[0]
45	TX_CYC2PA/UART_CTS/PD<1>	Digital I/O	Control external PA / UART_CTS / GPIO PD[1]
46	SPI_CN/PWM3/PD<2>	Digital I/O	SPI chip select (Active low) / PWM3 output / GPIO PD[2]
47	PWM1_N/UART_TX/PD<3>	Digital I/O	PWM1 inverting output / UART_TX / GPIO PD[3]
48	SWM/PWM2_N/PD<4>	Digital I/O	Single wire master / PWM2 inverting output / GPIO PD[4]

1.6.3 Pin Layout for TLSR8359F512ET32

Figure 1-7 shows pin assignment for the TLSR8359F512ET32.

Figure 1-7 Pin Assignment for TLSR8359F512ET32
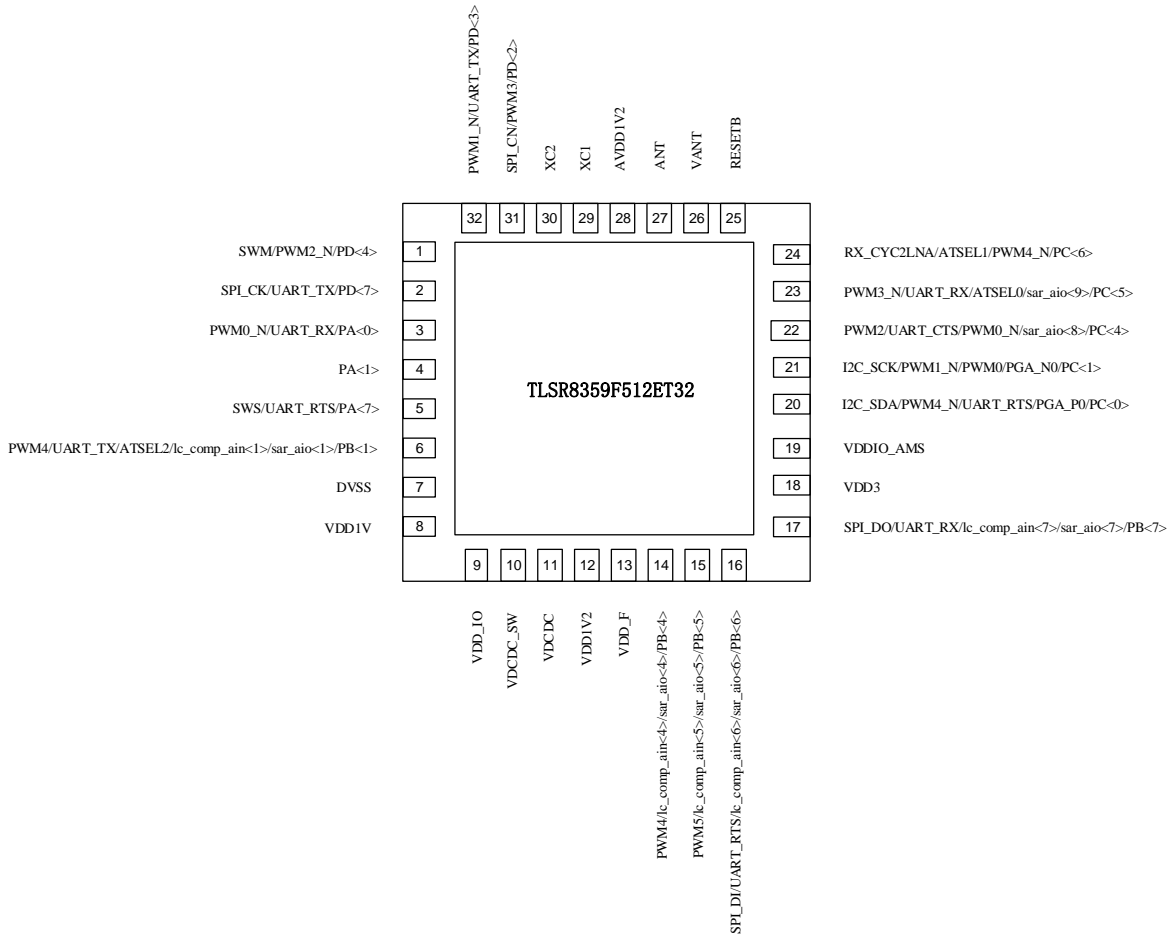
Functions of 32 pins for the TLSR8359F512ET32 are described in Table 1-4:

Table 1-4 Pin Functions for TLSR8359F512ET32

No.	Pin Name	Type	Description
1	SWM/PWM2_N/PD<4>	Digital I/O	Single wire master / PWM2 inverting output / GPIO PD[4]
2	SPI_CK/UART_TX/PD<7>	Digital I/O	SPI clock (I2C_SCK) / UART_TX / GPIO PD[7]
3	PWM0_N/UART_RX/PA<0>	Digital I/O	PWM0 inverting output / UART_RX / GPIO PA[0]

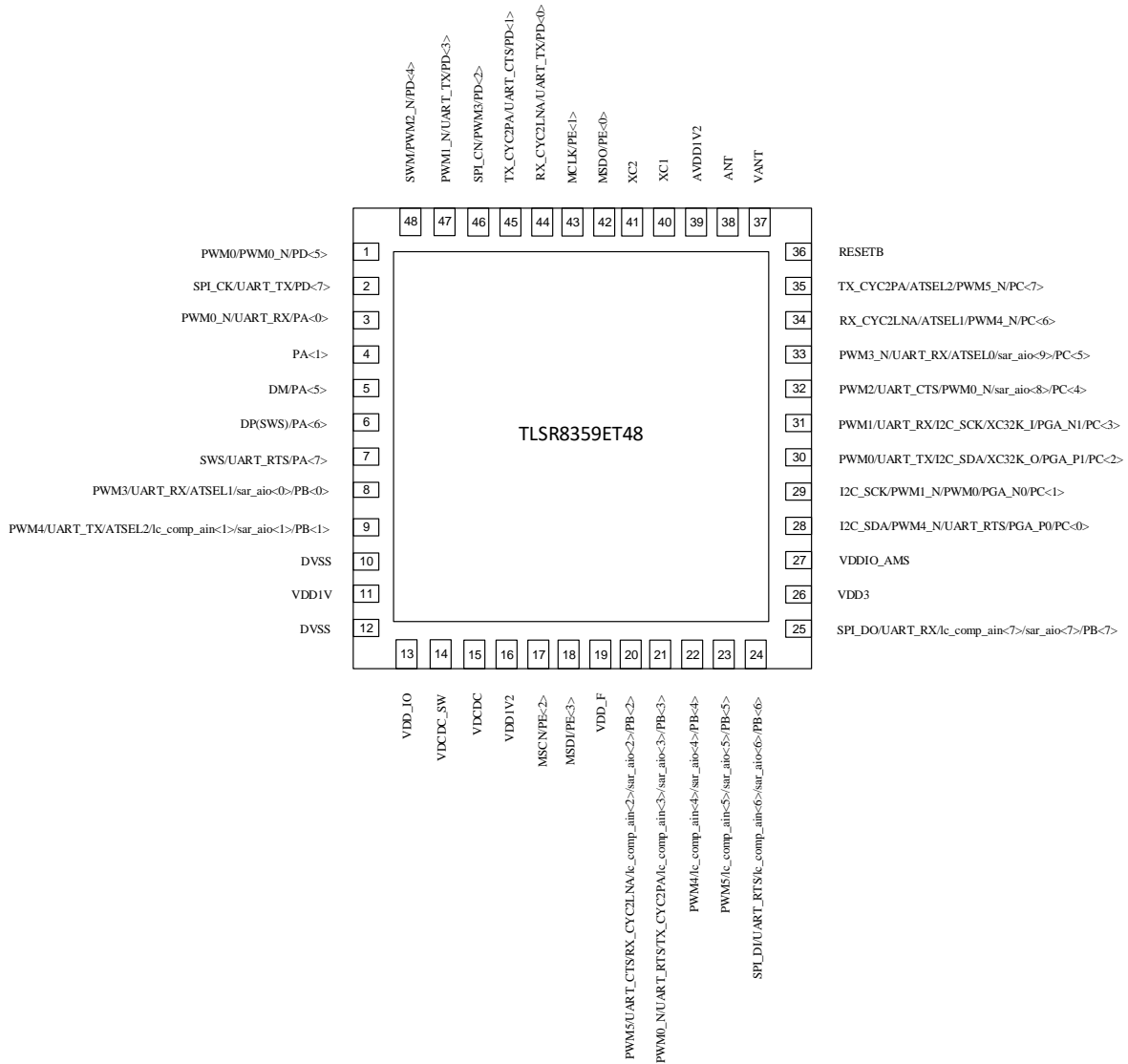| No.	Pin Name	Type	Description
4	PA<1>	Digital I/O	GPIO PA[1]
5	SWS/UART_RTS/PA<7>	Digital I/O	Single wire slave/ UART_RTS / GPIO PA[7]
6	PWM4/UART_TX/ATSEL2/lc_comp_ain<1>/sar_aio<1>/PB<1>	Digital I/O	PWM4 output / UART_TX / Antenna select pin 2 / Low power comparator input / SAR ADC input / GPIO PB[1]
7	DVSS	GND	Digital LDO ground
8	VDD1V	PWR	Internal LDO generated power supply input for digital core
9	VDD_IO	PWR	External 3.3 V power supply input for IO
10	VDCDC_SW	Analog	Connect with VDCDC via external inductor
11	VDCDC	Analog	Connect with VDCDC_SW via external inductor
12	VDD1V2	PWR	Internal DCDC generated power supply. Connect to GND via external capacitor. Route this 1.2 V voltage power supply to AVDD1V2.
13	VDD_F	PWR	Internally generated power supply to flash. Connect to GND via external capacitor.
14	PWM4/lc_comp_ain<4>/sar_aio<4>/PB<4>	Digital I/O	PWM4 output / Low power comparator input / SAR ADC input / GPIO PB[4]
15	PWM5/lc_comp_ain<5>/sar_aio<5>/PB<5>	Digital I/O	PWM5 output / Low power comparator input / SAR ADC input / GPIO PB[5]
16	SPI_DI/UART_RTS/lc_comp_ain<6>/sar_aio<6>/PB<6>	Digital I/O	SPI data input (I2C_SDA) / UART_RTS / Low power comparator input / SAR ADC input / GPIO PB[6]
17	SPI_DO/UART_RX/lc_comp_ain<7>/sar_aio<7>/PB<7>	Digital I/O	SPI data output / UART_RX / Low power comparator input / SAR ADC input / GPIO PB[7]
18	VDD3	PWR	Connect to an external 3.3 V power supply
19	VDDIO_AMS	PWR	External 3.3 V power supply input for analog/mixed signal circuitry.
20	I2C_SDA/PWM4_N/UART_RTS/	Digital I/O	I2C serial data / PWM4 inverting output / UART_RTS /

No.	Pin Name	Type	Description
	PGA_P0/PC<0>		PGA left channel positive input / GPIO PC[0]
21	I2C_SCK/PWM1_N/PWM0/ PGA_N0/PC<1>	Digital I/O	I2C serial clock / PWM1 inverting output / PWM0 output / PGA left channel negative input / GPIO PC[1]
22	PWM2/UART_CTS/PWM0_N/ sar_aio<8>/PC<4>	Digital I/O	PWM2 output / UART_CTS / PWM0 inverting output / SAR ADC input / GPIO PC[4]
23	PWM3_N/UART_RX/ATSEL0/ sar_aio<9>/PC<5>	Digital I/O	PWM3 inverting output / UART_RX / Antenna select pin 0 / SAR ADC input / GPIO PC[5]
24	RX_CYC2LNA/ATSEL1/PWM4_N/ PC<6>	Digital I/O	Control external LNA / Antenna select pin 1 / PWM4 inverting output / GPIO PC[6]
25	RESETB	RESET	Power on reset, active low
26	VANT	PWR	Internally generated voltage output to bias internal PA. Connect to ANT using an external inductor.
27	ANT	Analog	RF antenna, connect to VANT using external inductor
28	AVDD1V2	PWR	Power supply input for internal RF Modules. Route from VDD1V2. Connect to GND via external capacitor.
29	XC1	Analog	Connect 24 MHz crystal
30	XC2	Analog	Connect 24 MHz crystal
31	SPI_CN/PWM3/PD<2>	Digital I/O	SPI chip select (Active low) / PWM3 output / GPIO PD[2]
32	PWM1_N/UART_TX/PD<3>	Digital I/O	PWM1 inverting output / UART_TX / GPIO PD[3]

1.6.4 Pin Layout for TLSR8359F64ES16

Figure 1-8 shows pin assignment for the TLSR8359F64ES16.

Figure 1-8 Pin Assignment for TLSR8359F64ES16
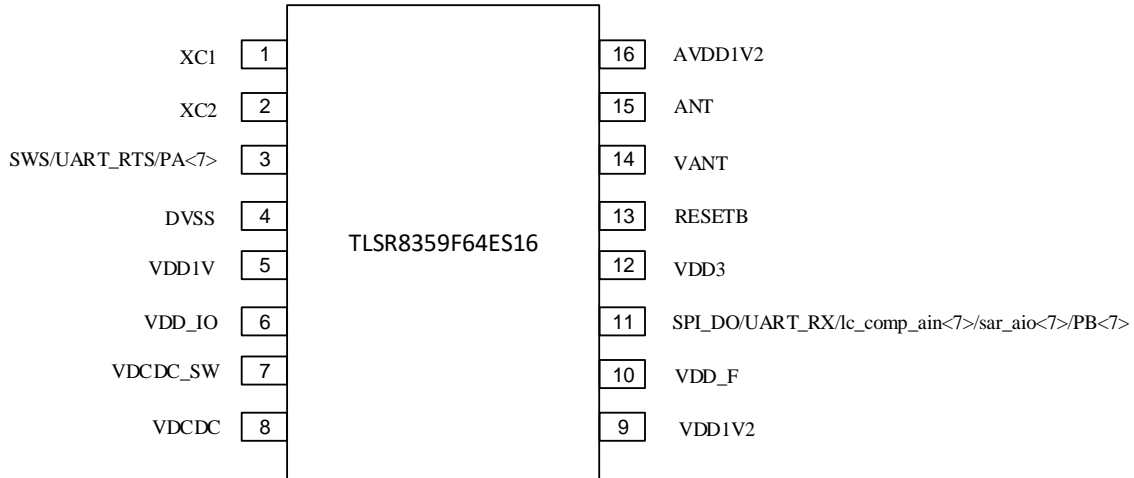
Functions of 16 pins for the TLSR8359F64ES16 are described in Table 1-5:

Table 1-5 Pin Functions for TLSR8359F64ES16

No.	Pin Name	Type	Description
1	XC1	Analog	Connect 24 MHz crystal
2	XC2	Analog	Connect 24 MHz crystal
3	SWS/UART_RTS/PA<7>	Digital I/O	Single wire slave/ UART_RTS / GPIO PA[7]
4	DVSS	GND	Digital LDO ground
5	VDD1V	PWR	Internal LDO generated power supply input for digital core
6	VDD_IO	PWR	External 3.3 V power supply input for IO
7	VDCDC_SW	Analog	Connect with VDCDC via external inductor
8	VDCDC	Analog	Connect with VDCDC_SW via external inductor
9	VDD1V2	PWR	Internal DCDC generated power supply. Connect to GND via external capacitor. Route this 1.2 V voltage

No.	Pin Name	Type	Description
			power supply to AVDD1V2.
10	VDD_F	PWR	Internally generated power supply to flash. Connect to GND via external capacitor.
11	SPI_DO/UART_RX/lc_comp_ain<7>/sar_aio<7>/PB<7>	Digital I/O	SPI data output / UART_RX / Low power comparator input / SAR ADC input / GPIO PB[7]
12	VDD3	PWR	Connect to an external 3.3 V power supply
13	RESETB	RESET	Power on reset, active low
14	VANT	PWR	Internally generated voltage output to bias internal PA. Connect to ANT using an external inductor.
15	ANT	Analog	RF antenna, connect to VANT using external inductor
16	AVDD1V2	PWR	Power supply input for internal RF Modules. Route from VDD1V2. Connect to GND via external capacitor.

1.6.5 Notes

1. All digital IOs including PA<0> ~ PD<7> can be used as GPIOs and have configurable pull-up/pull-down resistor.
2. SPI:
   - PD<7>: SPI_CK, PB<6>: SPI_DI, PB<7>: SPI_DO, PD<2>: SPI_CN
   - PA<2>: DO, PA<3>: DI, PA<4>: CK, PD<6>: CN
3. I2C:
   - PC<0> ~ PC<3> can be used as I2C. PC<0>: I2C_SDA, PC<1>: I2C_SCK, PC<2>: I2C_SDA, PC<3>: I2C_SCK
   - I2C can also be multiplexed with SPI interface, i.e. I2C_SDA/I2C_SCK can be multiplexed with SPI_DI (DI)/SPI_CK (CK) respectively.
4. PGA input: PC<0>~PC<3>. Please refer to Section 11 PGA.
5. UART:
   - PA<0>: UART_RX, PA<2>: UART_TX, PB<0>: UART_RX, PB<1>: UART_TX, PB<7>: UART_RX, PC<2>: UART_TX, PC<3>: UART_RX, PC<5>: UART_RX, PD<0>: UART_TX, PD<3>: UART_TX, PD<6>: UART_RX, PD<7>: UART_TX

- UART hardware flow control:

    PA<3>: UART_CTS, PA<4>: UART_RTS, PA<7>: UART_RTS, PB<2>: UART_CTS, PB<3>: UART_RTS,

    PB<6>: UART_RTS, PC<0>: UART_RTS, PC<4>: UART_CTS, PD<1>: UART_CTS

6. USB:

    - PA<5>: DM, PA<6>: DP

7. Single Wire debug interface:

    - PA<7>: SWS

    - SWS can also be multiplexed with DP.

    - PD<4>: SWM

8. RX_CYC2LNA & TX_CYC2PA:

    - PB<2>: RX_CYC2LNA, PB<3>: TX_CYC2PA, PC<6>: RX_CYC2LNA, PC<7>: TX_CYC2PA, PD<0>: RX_CYC2LNA, PD<1>: TX_CYC2PA

    - RX_CYC2LNA/TX_CYC2PA serves to control enabling external PA/LNA. Please refer to Section 3.1 Block Diagram.

9. ATSEL0~2:

    - PB<0>: ATSEL1, PB<1>: ATSEL2, PC<5>: ATSEL0, PC<6>: ATSEL1, PC<7>: ATSEL2, PD<6>: ATSEL0

    - ATSEL0~2 serves to select one of up to eight external antennas connected to the antenna select component. The selected antenna channel is connected to the RF_IO pin.

10. Low power comparator input: PB<1> ~ PB<7>. Please refer to Section 13 Low Power Comparator.

11. ADC GPIO input: PB<0> ~ PB<7>, PC<4> ~ PC<5>.

12. For register configuration to select pin multiplexed function, please refer to Section 7.1.1.2 Multiplexed Functions.

13. For 24 MHz crystal, the load capacitor range supported by design is 7.33 pF ~ 12.66 pF. If the crystal needs load capacitor of 15 pF, two external capacitors will be required.

14. Do not use PC<7:5> for dynamic applications. It's highly recommended to use these IOs in DC applications, e.g. as control or detect lines.

15. Pin drive strength:

    - PA<5:7> and PB<0:3> support drive strength up to 8 mA (8 mA when "DS"=1, 4 mA when "DS"=0); PB<4:7> support drive strength up to 16 mA (16 mA when "DS"=1, 12 mA when "DS"=0); other GPIOs (PA<0:4>, PC<0:7> and PD<0:7>) support drive strength up to 4 mA (4 mA when "DS"=1, 2 mA when "DS"=0).

    - "DS" configuration will take effect when the pin is used as output. Please refer to Section 7.1.1 Basic Configuration for the corresponding "DS" register address and the default setting.

2. Memory and MCU

2.1 Memory

All packages of the TLSR8359 embed 64 KB SRAM.

The TLSR8359F512 embeds 512 KB internal Flash.

The TLSR8359F64 embeds 64 KB internal Flash.

The TLSR8359ET48 has no internal Flash, but supports up to 16 MB external Flash via the MSPI interface.

2.1.1 SRAM/Register

SRAM/Register memory map is shown as follows:

Figure 2-1 Physical Memory Map
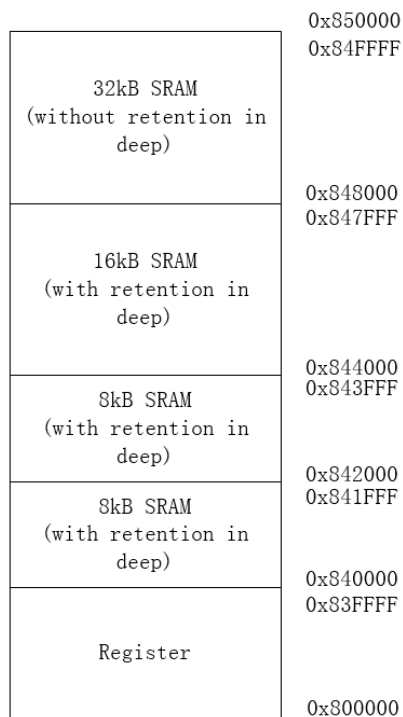
- · Register address: 0x800000 ~ 0x83FFFF.
- · SRAM space:
  Address for two independent 8 KB SRAMs with retention in deep sleep: 0x840000 ~ 0x841FFF, 0x842000 ~ 0x843FFF.

Address for 16 KB SRAM with retention in deep sleep: 0x844000 ~ 0x847FFF.

Address for 32 KB SRAM without retention in deep sleep: 0x848000 ~ 0x84FFFF.

Both register and SRAM address can be accessed (read or write) via debugging interface (SWS/SWM, SPI/I2C/USB interface).

Figure 2-2 Register Space

Register
(Base address: 0x800000)

Register	Address
RSVD	
	0x40000
Modem	
	0x01200
RSVD	
	0x01020
RSVD	
	0x01000
linklayer	
	0x00f00
RSVD	
	0x00d00
dma	
	0x00c00
DMA fifo	
	0x00b00
RSVD	
	0x00800
pwm	
	0x00780
System timer	
	0x00740
RSVD	
	0x00700
MCU	
	0x00600
gpio	
	0x00580
audio	
	0x00560
AES	
	0x00540
RSVD	
	0x00500
Baseband	
	0x00400
RSVD	
	0x00200
usb	
	0x00100
I2C address map	
	0x000e0
qdec	
	0x000d0
RSVD	
	0x000c0
RSVD	
	0x000b8
uart	
	0x000b4
swire	
	0x000b0
RSVD	
	0x000a0
uart	
	0x00090
RSVD	
	0x00080
System control	
	0x00040
RSVD	
	0x00010
RSVD	
	0x0000c
spi	
	0x00008
i2c	
	0x00000

2.1.2 Flash

For the TLSR8359ET48, external Flash address can be accessed via MSPI interface.

For the TLSR8359F512 and TLSR8359F64, the internal Flash mainly supports page program, sector/block/chip erase operations, and deep power down operation. Please refer to the corresponding SDK for Flash memory operation details.

MCU uses the system frequency to load instructions, and adopts flash driver to access (read/write) flash with the speed of half of the system clock.

2.1.3 E-Fuse

The non-volatile E-Fuse section is preloaded with 4-byte decryption key and 4-byte E-Fuse configuration, as shown below.

Table 2-1 E-Fuse Definition

E-Fuse Bit Information	Decryption Key	E-Fuse Configuration			
		Internal Information	Wafer No.	Lot No.	Internal Information
	Bit0~31	Bit32~47	Bit48~52	Bit53~55	Bit56~63

2.1.4 Unique ID

For chip identification and traceability, the flash is preloaded with 128-bit Unique ID (UID). This UID can be read via the interface in SDK.

2.2 MCU

The TLSR8359 integrates a powerful 32-bit MCU developed by Telink. The digital core is based on 32-bit RISC, and the length of instructions is 16 bits; four hardware breakpoints are supported.

2.3 Working Modes

The TLSR8359 supports six working modes, including Active, Idle, Suspend, Deep sleep with SRAM retention, deep sleep without SRAM retention, and Shutdown.

- The Power Management (PM) module is always active in all working modes.
- For modules such as MCU, RF transceiver (Radio), and SRAM, the state depends on working mode, as shown below.

Table 2-2 Working Modes

Mode	Active	Idle	Suspend	Deep Sleep with SRAM Retention	Deep Sleep Without SRAM Retention	Shutdown
MCU	active	stall	stall	off	off	off
32K normal SRAM (without retention in deep sleep)	on	on	on	off	off	off
Radio	available	available	off	off	off	off
USB	available	available	off	off	off	off
Wakeup time to Active mode	-	0 µs	100 µs	Shorter than deep sleep without retention, almost same as Suspend	1 ms	10 ms
32K retention SRAMs (with retention in deep sleep)	full	full	full	full	off	off
Wakeup on RTC (32K Timer wakeup)	-	-	available	available	available	off
Wakeup on pin (IO wakeup)	-	-	available	available	available	off
Wakeup on interrupt	-	available	-	-	-	-

Mode	Active	Idle	Suspend	Deep Sleep with SRAM Retention	Deep Sleep Without SRAM Retention	Shutdown
Wakeup on reset pin (RESETB)	-	available	available	available	available	on
Current	Please refer to Section 15.3 DC Characteristics.					

NOTE:

- *"active": MCU is at working state.*

- *"stall": In Idle and Suspend mode, MCU does not work, while its clock is still running.*

- *"available" for Modules: It's selectable to be at working state, or stall/be powered down if it does not need to work.*

- *"available"/"on" for wakeup: Corresponding wakeup method is supported.*

- *"off" for wakeup: Corresponding wakeup method is not supported.*

- *"on"/"off"/"full" for SRAMs:*

  - *"on": The 32 KB normal SRAM is powered on and works normally (can be accessed) in Active, Idle and Suspend mode.*

  - *"full": Full speed. In Active, Idle and Suspend mode, the retention SRAMs are powered on and work normally (can be accessed); in Deep sleep with SRAM retention, the retention SRAMs are powered on, however, the contents of the retention SRAMs can be retained and cannot be accessed.*

  - *"off": The 32 KB normal SRAM is powered down in two Deep sleep modes and Shutdown mode. The retention SRAMs are powered down in Deep sleep without SRAM retention and Shutdown mode.*

- *Current:*

  - *In Deep sleep without SRAM retention, only the PM module is active, all digital and analog modules are powered down, thus the power consumption is largely decreased.*

  - *In Deep sleep with SRAM retention, the PM module is active, all analog and digital modules except for the retention SRAMs are powered down, thus the power consumption is a little higher than in Deep sleep without SRAM retention, but much lower than in Suspend.*

Table 2-3 Retention Analog Registers in Deep Sleep

Address	R/W	Description	Reset Value

0x35	R/W	buffer, watch dog reset clean	0x20
0x36	R/W	buffer, watch dog reset clean	0x00
0x37	R/W	buffer, watch dog reset clean	0x00
0x38	R/W	buffer, watch dog reset clean	0x00
0x39	R/W	buffer, watch dog reset clean	0xff
0x3a	R/W	buffer, power on reset clean	0x00
0x3b	R/W	buffer, power on reset clean	0x00
0x3c	R/W	buffer, power on reset clean	0x00

Analog registers (0x35~0x3c) as shown in Table 2-3 are retained in deep sleep mode and can be used to store program state information across deep sleep cycles.

- Analog registers 0x3a~0x3c are non-volatile even when chip enters deep sleep or chip is reset by watchdog or software, i.e. the contents of these registers won't be changed by deep sleep or watchdog reset or chip software reset.
- Analog registers 0x35~0x39 are non-volatile in deep sleep, but will be cleared by watchdog reset or chip software reset.
- After POR (Power-On-Reset), all registers will be cleared to their default values, including these analog registers.

User can set flag in these analog registers correspondingly, so as to check the booting source by reading the flag.

For chip software reset, please refer to Section 2.4 Reset.

2.4 Reset

The chip supports three types of reset methods, including POR (Power-On-Reset), watchdog reset and software reset.

1. POR: After power on, the whole chip will be reset, and all registers will be cleared to their default values.
2. Watchdog reset: A programmable watchdog is supported to monitor the system. If watchdog reset is triggered, registers except for the retention analog registers 0x3a~0x3c will be cleared.

3. Software reset: It is also feasible to carry out software reset for the whole chip or some modules.

　　・　Setting address 0x6f[5] as 1b'1 is to reset the whole chip. Similar to watchdog reset, the retention analog registers 0x3a~0x3c are non-volatile, while other registers including 0x35~0x39 will be cleared by chip software reset.

　　・　Addresses 0x60~0x62 serve to reset individual modules: if some bit is set to logic "1", the corresponding module is reset.

Table 2-4 Register Configuration for Software Reset

Address	Mnemonic	Type	Description	Reset Value
0x60	RST0	R/W	Reset control, 1 for reset, 0 for clear [0]: SPI [1]: I2C [2]: UART (rs232) [3]: USB [4]: PWM [5]: QDEC [6]: RSVD [7]: Swire	0x7c
0x61	RST1	R/W	[0] ZB [1] System Timer [2] DMA [3] ALGM [4] AES [5] ADC [6] ALG [7] RSVD	0xff
0x62	RST2	R/W	[0] AIF [1] RSVD [2] DFIFO [3] RSVD	0xc7

Address	Mnemonic	Type	Description	Reset Value
			[4] RISC [5] MCIC [6] RISC1 (R) [7] MCIC1 (R)	
0x6f	PWDNEN	R/W	[0]: suspend enable (RW) [5]: rst all (act as watchdog reset) [6]: rsvd (mcu low power mode) (W) [7]: stall mcu trig If bit[0] set 1, then system will go to suspend. Or only stall mcu (W)	0x00

2.5 Power Management

The multiple-stage Power Management (PM) module is flexible to control power state of the whole chip or individual functional blocks such as MCU, RF Transceiver, and peripherals.

2.5.1 Power-On-Reset (POR) and Brown-Out Detect

Figure 2-3 Control Logic for Power Up/Down
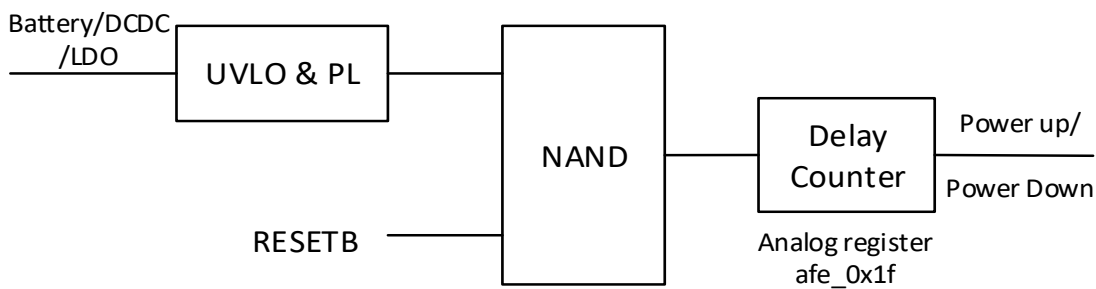
The whole chip power up and down is controlled by the UVLO (Ultra-low Voltage Lockout) & PL (Power Logic) module and the external RESETB pin via the logic shown in the above diagram. UVLO takes the external power supply as input and releases the lock only when the power supply voltage is higher than a

preset threshold. The RESETB pin has an internal pull-up resistor; an external Cap can be connected on the RESETB pin to control the POR delay.

After both UVLO and RESETB release, there is a further configurable delay before the system reset signal ("Sysrst") is released. The delay is adjusted by analog register afe_0x1f. Since the content of afe_0x1f is reset to default only after power cycle, watchdog reset, or software reset, the delay change using afe_0x1f is only applicable when the chip has not gone through these reset conditions. For example, after deep sleep wakeup, the setting in afe_0x1f will take effect.

Table 2-5 Analog Register to Control Delay Counters

Address	Name	Type	Description	Default
afe_0x1f	r_dly1	R/W	Wait for DCDC ready (16 kHz count)	0x40

Initial Power up

Figure 2-4 Initial Power-Up Sequence
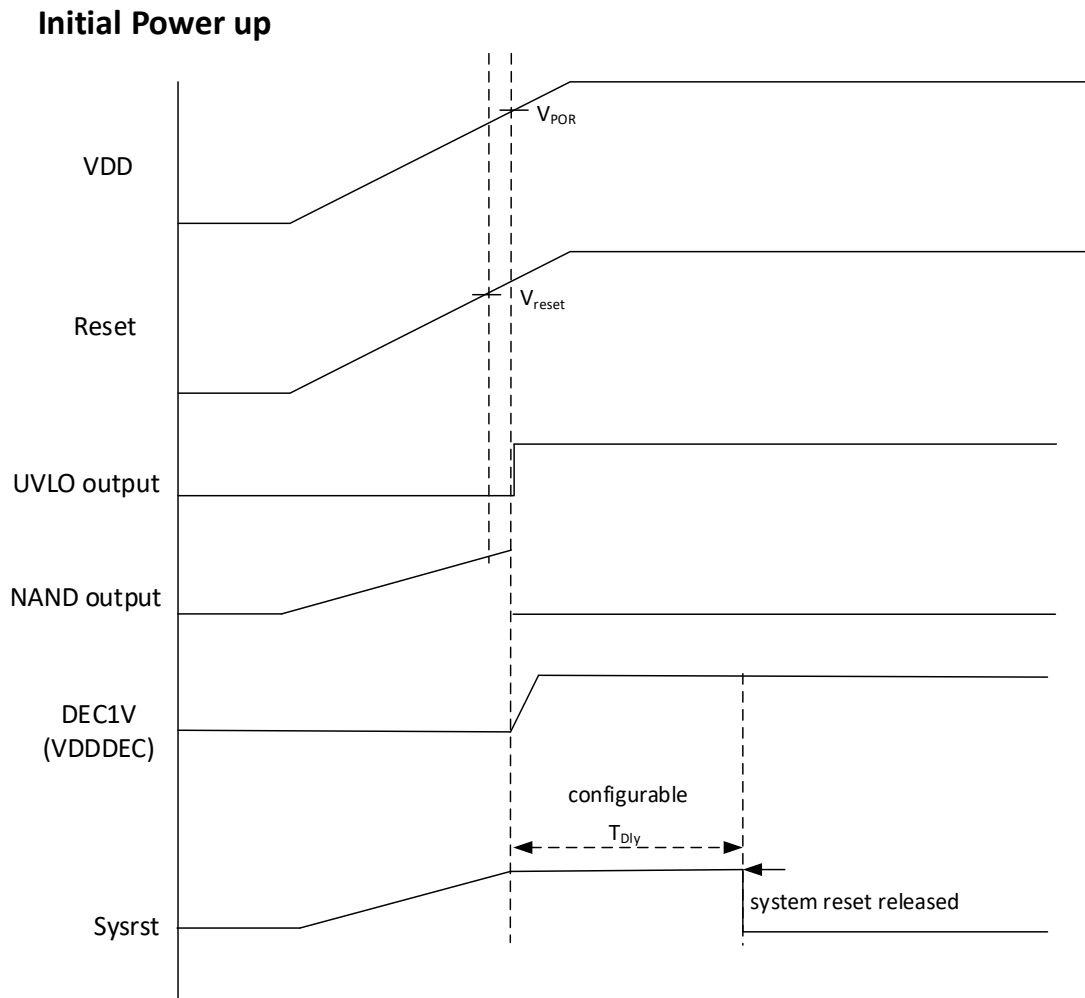
Figure 2-5 Power-Down Sequence
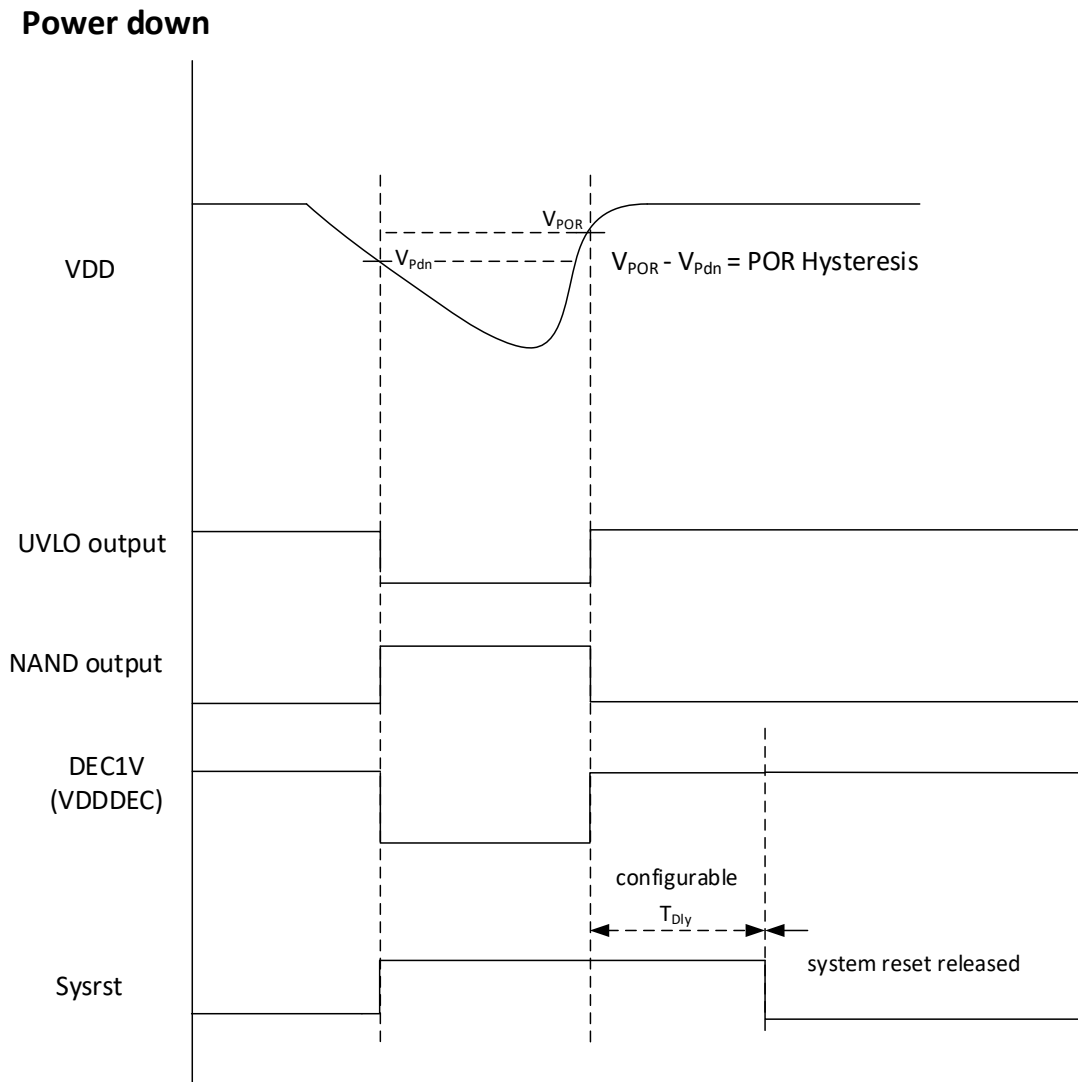
Power down

Table 2-6 Characteristics of Initial Power-Up/ Power-Down Sequence

Symbol	Parameter	Min.	Typ.	Max.	Unit
V_{POR}	VDD voltage when V_{UVLO} turns to high level	-	1.62	-	V
V_{Pdn}	VDD voltage when V_{UVLO} turns to low level	-	1.55	-	V
T_{Dly}	Delay counter value	Configurable via analog register **afe_0x1f**			

2.5.2 Working Mode Switch

In Active mode, MCU is active, all SRAMs are accessible, and other modules are selectable whether to be at working state.

The chip can switch to Idle mode to stall the MCU. In this mode, all SRAMs are still accessible, modules such as RF transceiver, USB are still selectable whether to be at working state. The chip can be triggered to Active mode by interrupt or RESETB pin, and the time to switch to Active mode is negligible.

To decrease power consumption to different levels, the chip can switch to power saving mode (Suspend, Deep sleep with SRAM retention, Deep sleep without SRAM retention, Shutdown) correspondingly. (Please refer to Table 2-2.)

- In Suspend mode, MCU stalls, all SRAMs are still accessible, the PM module is active, and modules such as RF transceiver, USB are powered down. The chip can be triggered to Active mode by 32K Timer, IO pin or RESETB pin. It takes 100 μs or so to switch from Suspend mode to Active mode.

- In Deep sleep with SRAM retention, the PM module is active, analog and digital modules except for the retention SRAMs are powered down, while the retention SRAMs can be retained and not accessible. The chip can be triggered to Active mode by 32K Timer, IO pin or RESETB pin. The time to switch to Active mode is shorter than Deep sleep without SRAM retention and close to Suspend.

- In Deep sleep without SRAM retention, only the PM module is active, while analog and digital modules including the retention SRAMs are powered down. The chip can be triggered to Active mode by 32K Timer, IO pin or RESETB pin. The time to switch to Active mode is 1 ms or so.

- In Shutdown mode, all digital and analog modules are powered down, and only the PM module is active. The chip can be triggered to Active mode by RESETB pin only. The time to switch to Active mode is 10 ms or so.

User can directly invoke corresponding library function to switch working mode of the chip.

If certain module doesn't need to work, user can power down this module in order to save power.

Table 2-7 3.3 V Analog Registers for Module Power Up/Down Control

Address	Local Name	Default	Description
afe_0x05<0>	32K_rc_pd	0	Power down 32 kHz RC oscillator 1: Power down, 0: Power up
afe_0x05<1>	32k_xtal_pd	1	Power down 32 kHz crystal 1: power down, 0: power up
afe_0x05<2>	24M_rc_pd	0	Power down of 24 MHz RC oscillator 1: Power down, 0: Power up

Address	Local Name	Default	Description
afe_0x05<3>	xtal_LDO_pd	0	Power down of 24 MHz crystal oscillator 1: Power down, 0: Power up
afe_0x05<4>	pd_pl_all_3v	0	Power down of power logic, 4.2 V VBUS_LDO and DCDC 1: Power down, 0: Power up
afe_0x05<5>	pd_pl_dcdc_3v	0	Power down of DCDC 1: Power down, 0: Power up
afe_0x05<6>	pd_pl_vbus_ldo_3v	0	Power down of VBUS_LDO 1: Power down, 0: Power up
afe_0x05<7>	ana_ldo_pd_3V	0	Power down baseband pll LDO 1: Power down, 0: Power up
afe_0x06	rsvd	-	-
afe_0x07<0>	spd_ldo_pd	1	Power down of spd ldo 1: Power down, 0: power up
afe_0x07<1>	dig_ldo_pd	0	Power down of main digital ldo 1: Power down, 0: power up
afe_0x07<2>	dig_ret_pd	1	Power down of retention ldo 1: Power down, 0: power up
afe_0x07<3>	pd_lc_comp_3v	1	Power down of low current comparator: 1: Power down, 0: Power up
afe_0x07<4>	pd_temp_sensor_3v	1	Power down of temperature sensor: 1: Power down, 0: Power up

2.5.3 LDO and DCDC

The chip embedded DCDC can generate 1.8 V output voltage and supply power for the internal Flash; the DCDC can also generate 1.4 V output voltage.

The embedded LDO regulator takes the 1.4 V voltage output from the DCDC, and generates 1.2 V regulated voltage to supply power for 1.2 V digital core and analog modules in Active/Idle/Suspend mode.

2.5.4 VBAT and VANT Power-Supply Mode

The chip provides two power-supply modes including VBAT mode and VANT mode.

- In VBAT mode, the chip is directly supplied with power by its battery voltage. The maximum output power is related to power supply voltage, for example, the maximum power is 10 dBm or so at 3.3 V power supply.

- In VANT mode, the chip is supplied with 1.2 V voltage by the embedded DCDC and LDO. In this mode, output power won't change with AVDD basically, and the maximum power is 5 dBm or so. Corresponding to the VBAT mode, the VANT mode is more power-saving at the same TX power.

2.6 Wakeup Sources

Figure 2-6 Wakeup Sources
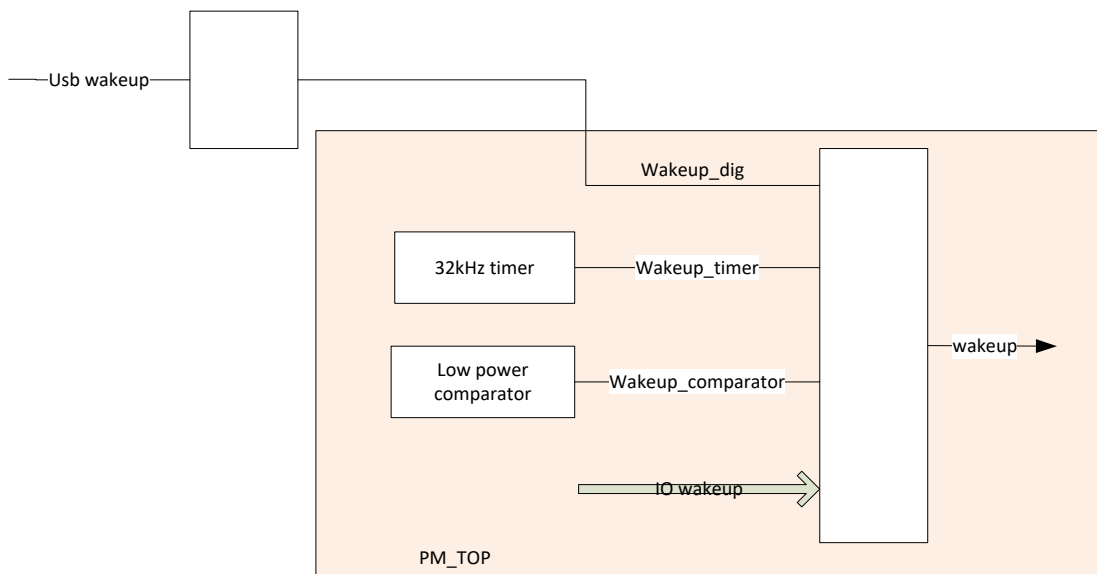
2.6.1 Wakeup Source - USB

This wakeup source can only wake up the system from suspend mode.

First, set the digital register 0x6e bit[2] as 1b'1.

To activate this mode, analog register afe_0x26[5] should also be set as 1b'1.

Once USB host sends out resuming signal, the system will be woke up.

2.6.2 Wakeup Source – 32 kHz Timer

This wakeup source is able to wake up the system from suspend mode or two deep sleep modes.

To enable the wakeup source from 32kHz timer, analog register afe_0x26[6] should be set as 1b'1.

2.6.3 Wakeup Source – Low Power Comparator

This wakeup source is able to wake up the system from suspend mode or two deep sleep modes.

To enable the wakeup source from low power comparator, analog register 0x26[7] should be set as 1b'1. The low power comparator wakeup is active high.

2.6.4 Wakeup Source – IO

This wakeup source is able to wake up the system from suspend mode or two deep sleep modes. And IO wakeup supports high level or low level wakeup which is configurable via polarity control registers.

Analog register afe_0x26[4] should be set as 1b'1 to enable IO wakeup source.

Enabling control analog registers: PA[7:0] enabling control register is afe_0x27[7:0], PB[7:0] enabling control register is afe_0x28[7:0], PC[7:0] enabling control register is afe_0x29[7:0], and PD[7:0] enabling control register is afe_0x2a[7:0]. Total wakeup pin can be up to 32.

Polarity control registers: PA[7:0] polarity control register is afe_0x21[7:0], PB[7:0] polarity control register is afe_0x22[7:0], PC[7:0] polarity control register is afe_0x23[7:0], and PD[7:0] polarity control register is afe_0x24[7:0].

The corresponding driver is available so that user can directly invoke it to use IO wakeup source.

Analog register 0x44[3:0] indicates the wakeup source which triggers system wakeup. After wakeup, the corresponding wakeup status will be set as 1b'1 automatically, and it's needed to write 1 to manually clean the status.

2.6.5 Register Table

Table 2-8 Analog Registers for Wakeup

Address	Name	Type	Description	Default
afe_0x21	PA_POL	R/W	Polarity control registers for IO wakeup	0x00
afe_0x22	PB_POL	R/W		0x00
afe_0x23	PC_POL	R/W		0x00
afe_0x24	PD_POL	R/W		0x00
afe_0x25	rsvd	R	-	-
afe_0x26	wkup_en	R/W	[7] low power comparator wakeup enable [6] 32 kHz timer wakeup enable [5] digital core (USB) wakeup enable [4] IO (pad) wake up enable [3] Enable/Mask filter for IO (Pad) wakeup 1: Select 16 µs filter to filter out jitter on IO PAD input. 0: IO Pad combinational logic output (disable filter)	0x00
afe_0x27	PA wake up enable	R/W	Enabling control registers for IO wakeup	0x00
afe_0x28	PB wake up enable	R/W		0x00
afe_0x29	PC wake up enable	R/W		0x00
afe_0x2a	PD wake up enable	R/W		0x00
afe_0x44	status	R	[7] dcdc_rdy [6] wd_status [5] cal_done_24m [4] cal_done_32k [3] IO (pad) wakeup status	-

Address	Name	Type	Description	Default
			[2] digital core (USB) wakeup status	
			[1] 32k timer wakeup status	
			[0] low power comparator wakeup status	

Table 2-9 Digital Register for Wakeup

Address	Mnemonic	Type	Description	Reset Value
0x6e	WAKEUPEN	R/W	Wakeup enable [0]: enable wakeup from I2C host [1]: enable wakeup from SPI host [2]: enable wakeup from USB [3]: enable wakeup from gpio [4]: enable wakeup from I2C synchronous interface System resume control [5]: enable GPIO remote wakeup [6]: if set to 1, system will issue USB resume signal on USB bus [7] sleep wakeup reset system enable	0x1f

3. 24 GHz RF Transceiver

3.1 Block Diagram

The TLSR8359 integrates an advanced 2.4 GHz RF transceiver. The RF transceiver works in the worldwide 2.4 GHz ISM (Industrial Scientific Medical) band.

The transceiver consists of a fully integrated RF synthesizer, a Power Amplifier (PA), a Low Noise Amplifier (LNA), a TX filter, a RX filter, a TX DAC, an ADC, a modulator and a demodulator. The transceiver can be configured to work in Proprietary 1 Mbps, 2 Mbps, 250 kbps and 500 kbps mode.

Figure 3-1 Block Diagram of RF Transceiver
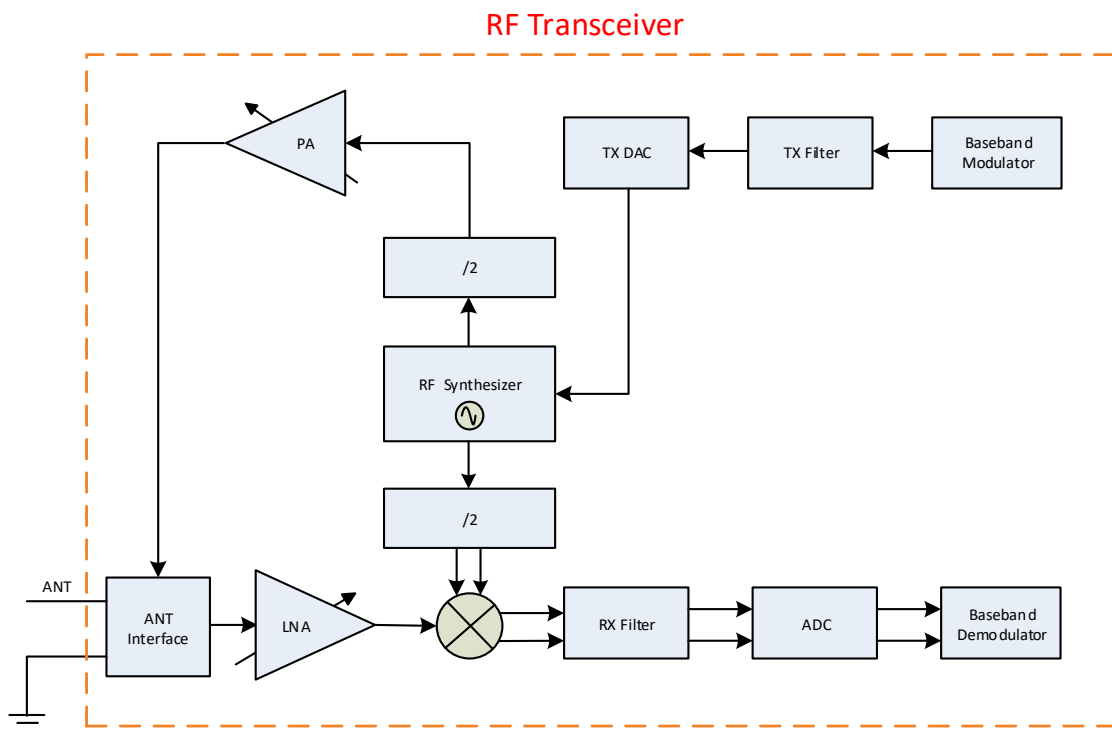
To control external PA and LNA, first follow the GPIO lookup table (see Section 7.1.1.1 GPIO Lookup Table) to configure the specific two pins as TX_CYC2PA and RX_CYC2LNA function, respectively. Note: To use TX_CYC2PA and RX_CYC2LNA function for the two pins, other functions with higher polarity should be disabled at the same time.

After the two pins are configured as TX_CYC2PA and RX_CYC2LNA function, the output function is enabled. Generally the two pins are high active: When both the two pins output low level, the external PA and LNA

are disabled; when one of the two pins output high level, the external PA/LNA are enabled correspondingly; the two pins won't output high level simultaneously.

Table 3-1 External RF Transceiver Control Example

TX_CYC2PA	RX_CYC2LNA	External RF Transceiver
L	L	Both LNA and PA OFF
L	H	LNA ON
H	L	PA ON
H	H	N/A

The internal PA can deliver a maximum 10 dBm output power, avoiding the need for an external RF PA.

3.2 Air Interface Data Rate and RF Channel Frequency

Air interface data rate, the modulated signaling rate for RF transceiver when transmitting and receiving data, is configurable via related register setting: 250 kbps, 500 kbps, 1 Mbps, 2 Mbps.

For the TLSR8359, RF transceiver can operate with frequency ranging from 2.400 GHz to 2.4835 GHz. The RF channel frequency setting determines the center of the channel.

3.3 Baseband

The baseband is disabled by default. The corresponding API is available for user to power on/down the baseband and enable/disable clock, so that the baseband can be turned on/off flexibly.

The baseband contains dedicated hardware logic to perform fast AGC control, access code correlation, CRC checking, data whitening, encryption/decryption and frequency hopping logic.

The baseband supports all features required by proprietary 2.4 GHz specification.

3.3.1 Packet Format

Packet format in 2.4 GHz proprietary mode is shown as Table 3-2:

Table 3-2 Packet Format in Proprietary Mode

LSB   MSB

Preamble (8 bits)	Address code (configurable 3~5 bytes)	Packet Controller + Payload (1~63 bytes)	CRC (1~2 bytes)

3.3.2 RSSI and Frequency Offset

The TLSR8359 provides accurate RSSI (Receiver Signal Strength Indicator) and frequency offset indication.

- RSSI can be read from the 1 byte at the tail of each received data packet.
- If no data packet is received (e.g. to perform channel energy measurement when no desired signal is present), real-time RSSI can also be read from specific registers which will be updated automatically.
- RSSI monitoring resolution can reach +/-1 dB.
- Frequency offset can be read from the 2 bytes at the tail of the data packet. Valid bits of actual frequency offset may be less than 16 bits, and different valid bits correspond to different tolerance range.

Telink supplies corresponding drivers for user to read RSSI and frequency offset as needed.

4. Clock

4.1 Clock Sources

Figure 4-1 Block Diagram of Clock
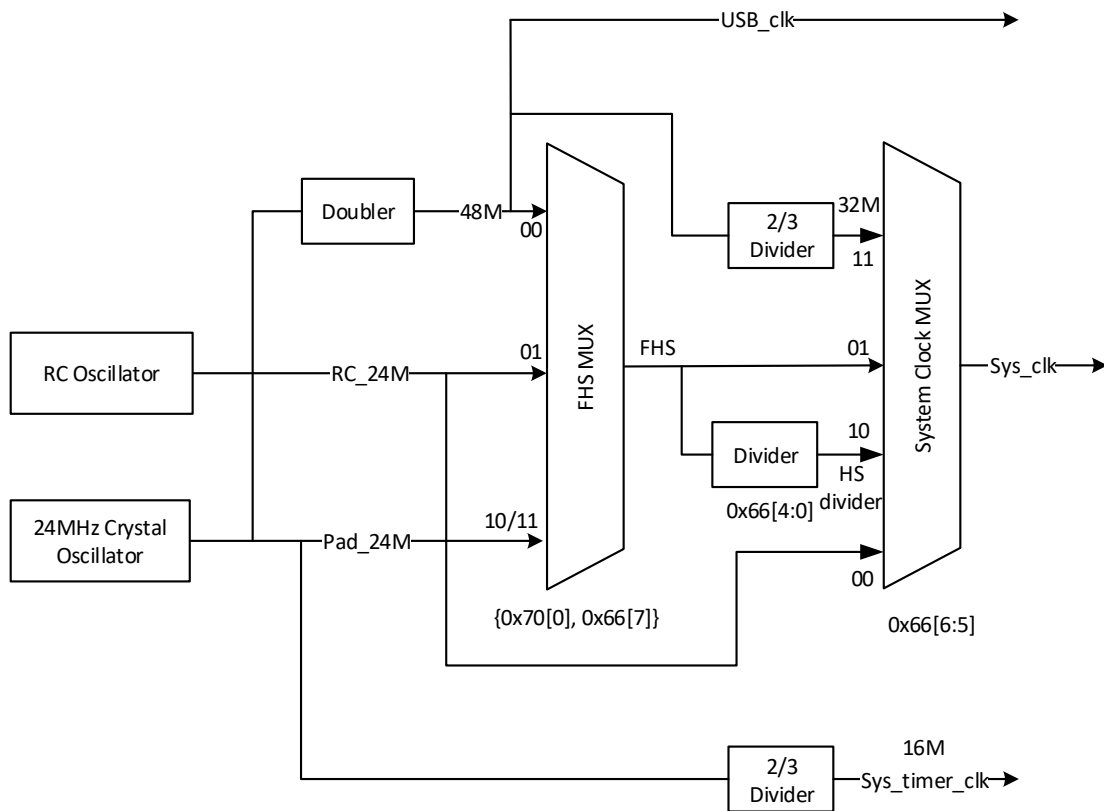
The TLSR8359 embeds a 24 MHz RC oscillator which can be used as clock source for system, as well as a 32 kHz RC oscillator to provide clock source for sleep state.

External 24 MHz crystal is available via pin XC1 and XC2, which can provide a Pad_24MHz clock source for system and System Timer, and generate a 48M clock via a frequency doubler to provide clock source for USB. External 32K crystal is available via pin PC<2:3>, which can provide a 32 kHz clock source.

4.2 System Clock

There are four selectable clock sources for MCU system clock: **RC_24M** derived from 24 MHz RC oscillator, High speed clock "**FHS**", **HS divider clock** (derived from "FHS" via a frequency divider), and **32 MHz clock** derived from 48 MHz clock via a 2/3 frequency divider (The 48M clock is derived from 24M crystal oscillator via a frequency doubler).

The high speed clock (FHS) is selectable via address {0x70[0], 0x66[7]} from the following sources: **48 MHz** clock (derived from 24M crystal oscillator via a frequency doubler), **RC_24M** (derived from 24 MHz RC oscillator), and **Pad_24M** (derived from 24M crystal oscillator).

The digital register CLKSEL (address 0x66) serves to set system clock: System clock source is selectable via bit[6:5].

- If address 0x66[6:5] is set to 2b'10 to select the HS divider clock, system clock frequency is adjustable via address 0x66[4:0]. The formula is shown as below:

  $F_{System\ clock} = F_{FHS}$ / (system clock divider value in address 0x66[4:0]).

  Note that address 0x66[4:0] should not be set as 0 or 1.

4.3 Module Clock

Registers CLKEN0~CLKEN2 (address 0x63~0x65) are used to enable or disable clock for various modules. By disable the clocks of unused modules, current consumption could be reduced.

4.3.1 System Timer Clock

System Timer clock is derived from 24M crystal oscillator via a 2/3 frequency divider. The clock frequency is fixed as 16 MHz.

4.3.2 USB Clock

USB clock is derived from 48M clock. The 48M clock is derived from 24M crystal oscillator via a frequency doubler.

4.4 Register Table

Table 4-1 Register Table Related to Clock

Address	Mnemonic	R/W	Description	Default
0x63	CLKEN0	R/W	Clock enable control: 1 for enable; 0 for disable [0]: SPI [1]: I2C [2]: UART (rs232) [3]: USB [4]: PWM [5]: QDEC [6]: RSVD [7]: Swire	0x83
0x64	CLKEN1	R/W	[0]ZB [1]System Timer [2]DMA [3]ALGM [4]AES [5:7]RSVD	0x00
0x65	CLKEN2	R/W	[0]AIF [1]RSVD [2]DFIFO [3]RSVD [4]MC [5]MCIC [6:7]RSVD	0x30

Address	Mnemonic	R/W	Description	Default
0x66	CLKSEL	R/W	System clock select [4:0]: system clock divider (must exceed 1). If 0x66[6:5] is set as 2b'10, F_{Sysclk} = F_{FHS} / (CLKSEL[4:0]). FHS: refer to 0x70 CLKSEH. [6:5]: select system clock source 2'b00: RC_24M from RC oscillator 2'b01: FHS 2'b10: HS divider (see 0x66[4:0]) 2'b11: 32M clock (48M * 2/3 divider) [7] FHS select (see 0x70[0])	0x06
0x67	-	R/W	RSVD	0x00
0x68	-	R/W	RSVD	0x02
0x6c	-	R/W	RSVD	0x01
0x6d	-	R/W	RSVD	0x02
0x70	CLKSEH	R/W	{0x70[0], 0x66[7]} FHS select 2'b00: 48M clock doubled from 24M crystal 2'b01: RC_24M from RC oscillator 2'b1x: Pad_24M from 24M crystal oscillator	0x00
0x73	SEL	R/W	[0]: clk32k select 0: select RC_32k from RC oscillator 1: select Pad_32k from 32K crystal oscillator [1]: DMIC clock select 1: select 32k (see 0x73[0] to select 32K source) 0: rsvd (select DMIC clk div)	0x04

5. Timers

5.1 Timer0 ~ Timer2

The TLSR8359 supports three timers: Timer0 ~ Timer2. The three timers all support four modes: Mode 0 (System Clock Mode), Mode 1 (GPIO Trigger Mode), Mode 2 (GPIO Pulse Width Mode) and Mode 3 (Tick Mode), which are selectable via the register TMR_CTRL0 (address 0x620) ~ TMR_CTRL1 (address 0x621).

Timer2 can also be configured as "watchdog" to monitor firmware running.

5.1.1 Register Table

Table 5-1 Register Configuration for Timer0 ~ Timer2

Address	Mnemonic	Type	Description	Reset Value
0x72	Wd_status	R/W	[0] watch dog status: verify whether it is power reset (1'b0) or watch dog reset (1'b1), write 1 to clear.	00
0x620	TMR_CTRL0	RW	[0]Timer0 enable [2:1] Timer0 mode. 0: using sclk, 1: using gpio, 2: count width of gpi, 3: tick [3]Timer1 enable [5:4] Timer1 mode. [6]Timer2 enable [7]Bit of timer2 mode	00
0x621	TMR_CTRL1	RW	[0]Bit of timer2 mode [7:1]Low bits of watch dog capture	00
0x622	TMR_CTRL2	RW	[6:0]High bits of watch dog capture. It is compared with [31:18] of timer2 ticker [7]watch dog capture	00

Address	Mnemonic	Type	Description	Reset Value
0x623	TMR_STATUS	RW	[0] timer0 status, write 1 to clear [1] timer1 status, write 1 to clear [2] timer2 status, write 1 to clear [3] watch dog status, write 1 to clear (If Watchdog is enabled, need to clear it periodically to avoid triggering watchdog reset)	00
0x624	TMR_CAPT0_0	RW	Byte 0 of timer0 capture	00
0x625	TMR_CAPT0_1	RW	Byte 1 of timer0 capture	00
0x626	TMR_CAPT0_2	RW	Byte 2 of timer0 capture	00
0x627	TMR_CAPT0_3	RW	Byte 3 of timer0 capture	00
0x628	TMR_CAPT1_0	RW	Byte 0 of timer1 capture	00
0x629	TMR_CAPT1_1	RW	Byte 1 of timer1 capture	00
0x62a	TMR_CAPT1_2	RW	Byte 2 of timer1 capture	00
0x62b	TMR_CAPT1_3	RW	Byte 3 of timer1 capture	00
0x62c	TMR_CAPT2_0	RW	Byte 0 of timer2 capture	00
0x62d	TMR_CAPT2_1	RW	Byte 1 of timer2 capture	00
0x62e	TMR_CAPT2_2	RW	Byte 2 of timer2 capture	00
0x62f	TMR_CAPT2_3	RW	Byte 3 of timer2 capture	00
0x630	TMR_TICK0_0	RW	Byte 0 of timer0 ticker	00
0x631	TMR_TICK0_1	RW	Byte 1 of timer0 ticker	00
0x632	TMR_TICK0_2	RW	Byte 2 of timer0 ticker	00
0x633	TMR_TICK0_3	RW	Byte 3 of timer0 ticker	00
0x634	TMR_TICK1_0	RW	Byte 0 of timer1 ticker	00
0x635	TMR_TICK1_1	RW	Byte 1 of timer1 ticker	00

Address	Mnemonic	Type	Description	Reset Value
0x636	TMR_TICK1_2	RW	Byte 2 of timer1 ticker	00
0x637	TMR_TICK1_3	RW	Byte 3 of timer1 ticker	00
0x638	TMR_TICK2_0	RW	Byte 0 of timer2 ticker	00
0x639	TMR_TICK2_1	RW	Byte 1 of timer2 ticker	00
0x63a	TMR_TICK2_2	RW	Byte 2 of timer2 ticker	00
0x63b	TMR_TICK2_3	RW	Byte 3 of timer2 ticker	00

5.1.2 Mode 0 (System Clock Mode)

In Mode 0, system clock is employed as clock source.

After Timer is enabled, Timer Tick (i.e. counting value) is increased by 1 on each positive edge of system clock from preset initial Tick value. Generally the initial Tick value is set to 0.

Once current Timer Tick value matches the preset Timer Capture (i.e. timing value), an interrupt is generated, Timer stops counting and Timer status is updated.

Steps of setting Timer0 for Mode 0 is taken as an example.

1st: Set initial Tick value of Timer0

Set Initial value of Tick via registers TMR_TICK0_0~TMR_TICK0_3 (address 0x630~0x633). Address 0x630 is lowest byte and 0x633 is highest byte. It's recommended to clear initial Timer Tick value to 0.

2nd: Set Capture value of Timer0

Set registers TMR_CAPT0_0~TMR_CAPT0_3 (address 0x624~0x627). Address 0x624 is lowest byte and 0x627 is highest byte.

3rd: Set Timer0 to Mode 0 and enable Timer0

Set register TMR_CTRL0 (address 0x620) [2:1] to 2b'00 to select Mode 0; Meanwhile set address 0x620[0] to 1b'1 to enable Timer0. Timer0 starts counting upward, and Tick value is increased by 1 on each positive edge of system clock until it reaches Timer0 Capture value.

5.1.3 Mode 1 (GPIO Trigger Mode)

In Mode 1, GPIO is employed as clock source. The "**m0**"/"**m1**"/"**m2**" register specifies the GPIO which generates counting signal for Timer0/Timer1/Timer2.

After Timer is enabled, Timer Tick (i.e. counting value) is increased by 1 on each positive/negative (configurable) edge of GPIO from preset initial Tick value. Generally the initial Tick value is set to 0. The "**Polarity**" register specifies the GPIO edge when Timer Tick counting increases.

NOTE: Refer to Section 7.1.2 for corresponding "**m0**", "**m1**", "**m2**" and "**Polarity**" register address.

Once current Timer Tick value matches the preset Timer Capture (i.e. timing value), an interrupt is generated and timer stops counting.

Steps of setting Timer1 for Mode 1 is taken as an example.

1st: Set initial Tick value of Timer1

Set Initial value of Tick via registers TMR_TICK1_0~TMR_TICK1_3 (address 0x634~0x637). Address 0x634 is lowest byte and 0x637 is highest byte. It's recommended to clear initial Timer Tick value to 0.

2nd: Set Capture value of Timer1

Set registers TMR_CAPT1_0~TMR_CAPT1_3 (address 0x628~0x62b). Address 0x628 is lowest byte and 0x62b is highest byte.

3rd: Select GPIO source and edge for Timer1

Select certain GPIO to be the clock source via setting "m1" register.

Select positive edge or negative edge of GPIO input to trigger Timer1 Tick increment via setting "Polarity" register.

4th: Set Timer1 to Mode 1 and enable Timer1

Set address 0x620[5:4] to 2b'01 to select Mode 1; Meanwhile set address 0x620[3] to 1b'1 to enable Timer1. Timer1 starts counting upward, and Timer1 Tick value is increased by 1 on each positive/negative (specified during the 3rd step) edge of GPIO until it reaches Timer1 Capture value.

5.1.4 Mode 2 (GPIO Pulse Width Mode)

In Mode 2, system clock is employed as the unit to measure the width of GPIO pulse. The "**m0**"/"**m1**"/"**m2**" register specifies the GPIO which generates control signal for Timer0/Timer1/Timer2.

After Timer is enabled, Timer Tick is triggered by a positive/negative (configurable) edge of GPIO pulse. Then Timer Tick (i.e. counting value) is increased by 1 on each positive edge of system clock from preset initial Tick value. Generally the initial Tick value is set to 0. The "**Polarity**" register specifies the GPIO edge when Timer Tick starts counting.

NOTE: Refer to Section 7.1.2 for corresponding "**m0**", "**m1**", "**m2**" and "**Polarity**" register address.

While a negative/positive edge of GPIO pulse is detected, an interrupt is generated and timer stops counting. The GPIO pulse width could be calculated in terms of tick count and period of system clock.

Steps of setting Timer2 for Mode 2 is taken as an example.

1st: Set initial Timer2 Tick value

Set Initial value of Tick via registers TMR_TICK2_0~TMR_TICK2_3 (address 0x638~0x63b). Address 0x638 is lowest byte and 0x63b is highest byte. It's recommended to clear initial Timer Tick value to 0.

2nd: Select GPIO source and edge for Timer2

Select certain GPIO to be the clock source via setting "m2" register.

Select positive edge or negative edge of GPIO input to trigger Timer2 counting start via setting "Polarity" register.

3rd: Set Timer2 to Mode 2 and enable Timer2

Set address 0x620[7:6] to 2b'01 and address 0x621 [0] to 1b'1.

Timer2 Tick is triggered by a positive/negative (specified during the 2nd step) edge of GPIO pulse. Timer2 starts counting upward and Timer2 Tick value is increased by 1 on each positive edge of system clock.

While a negative/positive edge of GPIO pulse is detected, an interrupt is generated and Timer2 tick stops.

4th: Read current Timer2 Tick value to calculate GPIO pulse width

Read current Timer2 Tick value from address 0x638~0x63b.

Then GPIO pulse width is calculated as follows:

$$\text{GPIO pulse width} = \text{System clock period} * (\text{current Timer2 Tick} - \text{intial Timer2 Tick})$$

For initial Timer2 Tick value is set to the recommended value of 0, then:

$$\text{GPIO pulse width} = \text{System clock period} * \text{current Timer2 Tick}.$$

5.1.5 Mode 3 (Tick Mode)

In Mode 3, system clock is employed.

After Timer is enabled, Timer Tick starts counting upward, and Timer Tick value is increased by 1 on each positive edge of system clock.

This mode could be used as time indicator. There will be no interrupt generated. Timer Tick keeps rolling from 0 to 0xffffffff. When Timer tick overflows, it returns to 0 and starts counting upward again.

Steps of setting Timer0 for Mode 3 is taken as an example.

1st: Set initial Tick value of Timer0

Set Initial value of Tick via address 0x630~0x633. Address 0x630 is lowest byte and address 0x633 is highest byte. It's recommended to clear initial Timer Tick value to 0.

2nd: Set Timer0 to Mode 3 and enable Timer0

Set address 0x620[2:1] to 2b'11 to select Mode 3, meanwhile set address 0x620[0] to 1b'1 to enable Timer0. Timer0 Tick starts to roll.

3rd: Read current Timer0 Tick value

Current Timer0 Tick value can be read from address 0x630~0x633.

5.1.6 Watchdog

Programmable watchdog could reset chip from unexpected hang up or malfunction.

Only Timer2 supports Watchdog.

Timer2 Tick has 32 bits. Watchdog Capture has only 14bits, which consists of TMR_CTRL2 (address 0x622) [6:0] as higher bits and TMR_CTRL1 (address 0x621) [7:1] as lower bits. Chip will be reset when the Timer2 Tick[31:18] matches Watch dog capture.

1st: Clear Timer2 Tick value

Clear registers TMR_TICK2_0 ~TMR_TICK2_3 (address 0x638~0x63b). Address 0x638 is lowest byte and 0x63b is highest byte.

2nd: Enable Timer2

Set register TMR_CTRL0 (address 0x620) [6] to 1b'1 to enable Timer2.

3rd: Set 14-bit Watchdog Capture value and enable Watchdog

Set address 0x622[6:0] as higher bits of watchdog capture and 0x621[7:1] as lower bits. Meanwhile set address 0x622[7] to 1b'1 to enable Watchdog.

Then Timer2 Tick starts counting upwards from 0.

If bits[31:18] of Timer2 Tick value read from address 0x638~0x63b reaches watchdog capture, the chip will be reset, and the status bit in address 0x72[0] will be set as 1b'1 automatically. User can read the watchdog status bit after chip reset to check if the reset source is watchdog, and needs to write 1b'1 to this bit to manually clear the flag.

5.2 32K LTIMER

The TLSR8359 also supports a low frequency (32 kHz) LTIMER in suspend mode or deep sleep mode. This timer can be used as one kind of wakeup source.

5.3 System Timer

The TLSR8359 also supports a System Timer. As introduced in Section 4.3.1 System Timer Clock, the clock frequency for System Timer is fixed as 16MHz irrespective of system clock.

In suspend mode, both System Timer and Timer0 ~ Timer2 stop counting, and 32K Timer starts counting. When the chip restores to active mode, Timer0 ~ Timer2 will continue counting from the number when they stops; In contrast, System Timer will continue counting from an adjusted number which is a sum of the number when it stops and an offset calculated from the counting value of 32K Timer during suspend mode.

Table 5-2 Register Table for System Timer

Address	Mnemonic	R/W	Function	Default Value
0x740	SYS_TIMER0	R/W	[7:3]	0x00
0x741	SYS_TIMER1	R/W	[7:0]	0x00
0x742	SYS_TIMER2	R/W	[7:0]	0x00
0x743	SYS_TIMER3	R/W	[7:0] System timer counter, write to set initial value. This is the sys timer counter	0x00

Address	Mnemonic	R/W	Function	Default Value
0x74c	SYS_TIMER_CTRL0	R/W	[7]:cal 32k enable (16 cycles 32k, count sys clock cycles) [6]:1:at the pos of 32k clock to set 32k timer value [5]:suspend bypass system_timer module [4]:system timer ss enable [3] manual set 32k timer mode [2]:manual set 32k timer 1:write, 0: read [1]:irq mask 1: enable, 0: disable [0] rsvd	0x90
0x74f	SYS_TIMER_CTRL1	-	[1]: timer_en, System Timer busy [2]: rsvd (clk32k_tm) [3]: m_wr_32k_en, busy reading/writing 32k Timer manually [7:4]: rsvd (ss)	0x00

NOTE: *The lower three bits of address 0x740 is invalid, therefore, the resolution should be 0.5 µs.*

6. Interrupt System

6.1 Interrupt Structure

The interrupting function is applied to manage dynamic program sequencing based on real-time events triggered by timers, pins and etc.

For the TLSR8359, there are 24 interrupt sources in all: 16 types are level-triggered interrupt sources (listed in address 0x640~0x641) and 8 types are edge-triggered interrupt sources (listed in address 0x642).

When CPU receives an interrupt request (IRQ) from some interrupt source, it will determine whether to respond to the IRQ. If CPU decides to respond, it pauses current routine and starts to execute interrupt service subroutine. Program will jump to certain code address and execute IRQ commands. After finishing interrupt service subroutine, CPU returns to the breakpoint and continues to execute main function.

6.2 Register Configuration

Table 6-1 Register Table for Interrupt System

Address	Mnemonic	Type	Description	Reset Value
0x640	MASK_0	RW	Byte 0 interrupt mask, level-triggered type {irq_mix, irq_uart, irq_dfifo, irq_dma, usb_pwdn, time2, time1, time0} [7] irq_mix, i.e. irq_host_cmd [6] irq_uart [5] irq_dfifo [4] irq_dma [3] usb_pwdn [2] time2 [1] time1 [0] time0	0x00
0x641	MASK_1	RW	Byte 1 interrupt mask, level-triggered type	0x00

Address	Mnemonic	Type	Description	Reset Value
			{rsvd, irq_pwm, irq_zb_rt, irq_udc[4:0]} [7] rsvd [6] irq_pwm [5] irq_zb_rt [4] irq_udc[4] [3] irq_udc[3] [2] irq_udc[2] [1] irq_udc[1] [0] irq_udc[0]	
0x642	MASK_2	RW	Byte 2 interrupt mask, edge-triggered type {rsvd, gpio2risc[1:0], irq_stimer, pm_irq, irq_gpio, usb_reset, usb_250us} [7] rsvd [6] gpio2risc[1] [5] gpio2risc[0] [4] irq_stimer [3] pm_irq_tm [2] irq_gpio [1] usb_reset [0] usb_250us	0x00
0x643	IRQMODE	RW	[0] interrupt enable [1] reserved (Multi-Address enable)	0x00
0x644	PRIO_0	RW	Byte 0 of priority 1: High priority; 0: Low priority	0x00
0x645	PRIO_1	RW	Byte 1 of priority	0x00
0x646	PRIO_2	RW	Byte 2 of priority	0x00
0x648	IRQSRC_0	R	Byte 0 of interrupt source	0x00

Address	Mnemonic	Type	Description	Reset Value
0x649	IRQSRC_1	R	Byte 1 of interrupt source	0x00
0x64a	IRQSRC_2	R	Byte 2 of interrupt source	0x00

6.2.1 Enable/Mask Interrupt Sources

Various interrupt sources could be enabled or masked by registers MASK_0~MASK_2 (address 0x640~0x642).

Interrupt sources of level-triggered type:

- irq_mix (0x640[7]): I2C Slave mapping mode or SPI Slave interrupt (irq_host_cmd)
- irq_uart (0x640[6]): UART interrupt
- irq_dfifo (0x640[5]): DFIFO interrupt
- irq_dma (0x640[4]): DMA interrupt
- usb_pwdn (0x640[3]): USB Host has sent power down signal
- time2, time1, timer0 (0x640[2]~0x640[0]): Timer2 ~ Timer0 interrupt
- irq_pwm (0x641[6]): PWM interrupt
- irq_zb_rt (0x641[5]): Baseband interrupt
- irq_udc[4:0] (0x641[4:0]): USB device interrupt

Interrupt sources of edge-triggered type:

- gpio2risc[1:0] (0x642[6]~0x642[5]): gpio2risc[1]~gpio2risc[0] interrupt, please refer to Section 7.1.2.
- irq_stimer (0x642[4]): System timer interrupt
- pm_irq_tm (0x642[3]): 32 kHz timer wakeup interrupt
- irq_gpio (0x642[2]): GPIO interrupt, please refer to Section 7.1.2.
- usb_reset (0x642[1]): USB Host has sent reset command.
- usb_250us (0x642[0]): USB has been in idle for 250 μs.

6.2.2 Interrupt Mode And Priority

Interrupt mode is typically-used mode. Register IRQMODE (address 0x643)[0] should be set as 1b'1 to enable interrupt function.

IRQ tasks could be set as High or Low priority via registers PRIO_0~PRIO_2 (address 0x644~0x646). When more than one interrupt sources assert interrupt requests at the same time, CPU will respond depending on respective interrupt priority levels. It's recommended not to modify priority setting.

6.2.3 Interrupt Source Flag

Three bytes in registers IRQSRC_0~IRQSRC_2 (address 0x648~0x64a) serve to indicate IRQ sources. Once IRQ occurs from certain source, the corresponding IRQ source flag will be set as "1". User could identify IRQ source by reading address 0x648~0x64a.

When handling edge-triggered type interrupt, the corresponding IRQ source flag needs to be cleared via address 0x64a. Take the interrupt source usb_250us for example: First enable the interrupt source by setting address 0x642 bit[0] as 1b'1; then set address 0x643 bit[0] as 1b'1 to enable the interrupt. In interrupt handling function, 24-bit data is read from address 0x648~0x64a to check which IRQ source is valid; if data bit[16] is 1, it means the usb_250us IRQ source is valid. Clear this interrupt source by setting address 0x64a bit[0] as 1b'1.

As for level-type interrupt, IRQ interrupt source status needs to be cleared by setting corresponding module status register. Take Timer0 IRQ interrupt source for example: First enable the interrupt source by setting address 0x640 bit[0] as 1b'1; then set address 0x643 bit[0] as 1b'1 to enable the interrupt. In interrupt handling function, 24-bit data is read from address 0x648~0x64a to check which IRQ source is valid; if data bit[0] is 1, it means the Timer0 IRQ source is valid. Register TMR_STATUS (address 0x623) [0] should be written with 1b'1 to manually clear Timer0 status (refer to Section 5.1.1 Register Table).

7. Interface

7.1 GPIO

The TLSR8359F512ET48, TLSR8359ET48, TLSR8359F512ET32 and TLSR8359F64ES16 support up to 32, 32, 17 and 2 GPIOs respectively. All digital IOs can be used as general purpose IOs.

The GPIOs including PA<0>~PD<7> have configurable pull-up/pull-down resistor. Please refer to Section 7.1.3 Pull-Up/Pull-Down Resistor for details.

7.1.1 Basic Configuration

7.1.1.1   GPIO Lookup Table

Table 7-1 GPIO Lookup Table 1

Pin	Default function	Pad Function Mux				GPIO Setting						
		Register=2	Register=1	Register=0	Register	Input (R)	IE	OEN	Output	Polarity	DS	Act as GPIO
PWM0_N/ UART_RX/ PA<0>	GPIO	UART_RX	PWM0_N	-	0x5a8[1:0]	0x580[0]	0x581[0]	0x582[0]	0x583[0]	0x584[0]	0x585[0]	0x586[0]
PA<1>	GPIO	-	-	-	0x5a8[3:2]	0x580[1]	0x581[1]	0x582[1]	0x583[1]	0x584[1]	0x585[1]	0x586[1]
DO/ UART_TX/ PWM0/ PA<2>	GPIO	PWM0	UART_TX	DO	0x5a8[5:4]	0x580[2]	0x581[2]	0x582[2]	0x583[2]	0x584[2]	0x585[2]	0x586[2]

Pin	Default function	Pad Function Mux				GPIO Setting						
		Register= 2	Register=1	Register= 0	Register	Input (R)	IE	OEN	Output	Polarity	DS	Act as GPIO
DI/ UART_CTS/ PWM1/ PA<3>	GPIO	PWM1	UART_CTS	DI	0x5a8[7: 6]	0x580[3]	0x581[3]	0x582[3]	0x583[3]	0x584[3]	0x585[3]	0x586[3]
CK/ UART_RTS/ PWM2/ PA<4>	GPIO	PWM2	UART_RTS	CK	0x5a9[1: 0]	0x580[4]	0x581[4]	0x582[4]	0x583[4]	0x584[4]	0x585[4]	0x586[4]
DM/ PA<5>	GPIO	-	-	DM	0x5a9[3: 2]	0x580[5]	0x581[5]	0x582[5]	0x583[5]	0x584[5]	0x585[5]	0x586[5]
DP(SWS)/ PA<6>	GPIO	-	-	DP(SWS)	0x5a9[5: 4]	0x580[6]	0x581[6]	0x582[6]	0x583[6]	0x584[6]	0x585[6]	0x586[6]
SWS/ UART_RTS/ PA<7>	SWS	-	UART_RTS	SWS	0x5a9[7: 6]	0x580[7]	0x581[7]	0x582[7]	0x583[7]	0x584[7]	0x585[7]	0x586[7]
PWM3/ UART_RX/ ATSEL1/ sar_aio<0>/ PB<0>	GPIO	ATSEL1	UART_RX	PWM3	0x5aa[1: 0]	0x588[0]	afe_0x bd[0]	0x58a[0]	0x58b[0]	0x58c[0]	afe_0x bf[0]	0x58e[0]

Pin	Default function	Pad Function Mux				GPIO Setting						
		Register= 2	Register=1	Register= 0	Register	Input (R)	IE	OEN	Output	Polarit y	DS	Act as GPIO
PWM4/ UART_TX/ ATSEL2/ lc_comp_ain<1 >/ sar_aio<1>/ PB<1>	GPIO	ATSEL2	UART_TX	PWM4	0x5aa[3: 2]	0x588[1]	afe_0x bd[1]	0x58a[1]	0x58b[1]	0x58c[1]	afe_0x bf[1]	0x58e[1]
PWM5/ UART_CTS/ RX_CYC2LNA/ lc_comp_ain< 2>/ sar_aio<2>/ PB<2>	GPIO	RX_CYC2L NA	UART_CTS	PWM5	0x5aa[5: 4]	0x588[2]	afe_0x bd[2]	0x58a[2]	0x58b[2]	0x58c[2]	afe_0x bf[2]	0x58e[2]
PWM0_N/ UART_RTS/ TX_CYC2PA/ lc_comp_ain< 3>/ sar_aio<3>/ PB<3>	GPIO	TX_CYC2P A	UART_RTS	PWM0_N	0x5aa[7: 6]	0x588[3]	afe_0x bd[3]	0x58a[3]	0x58b[3]	0x58c[3]	afe_0x bf[3]	0x58e[3]
PWM4/ lc_comp_ain< 4>/ sar_aio<4>/ PB<4>	GPIO	-	PWM4	-	0x5ab[1: 0]	0x588[4]	afe_0x bd[4]	0x58a[4]	0x58b[4]	0x58c[4]	afe_0x bf[4]	0x58e[4]

Pin	Default function	Pad Function Mux				GPIO Setting						
		Register=2	Register=1	Register=0	Register	Input (R)	IE	OEN	Output	Polarity	DS	Act as GPIO
PWM5/ lc_comp_ain< 5>/ sar_aio<5>/ PB<5>	GPIO	-	PWM5	-	0x5ab[3: 2]	0x588[5]	afe_0x bd[5]	0x58a[5]	0x58b[5]	0x58c[5]	afe_0x bf[5]	0x58e[5]
SPI_DI/ UART_RTS/ lc_comp_ain< 6>/ sar_aio<6>/ PB<6>	SPI_DI	UART_RTS	SPI_DI	-	0x5ab[5: 4]	0x588[6]	afe_0x bd[6]	0x58a[6]	0x58b[6]	0x58c[6]	afe_0x bf[6]	0x58e[6]
SPI_DO/ UART_RX/ lc_comp_ain< 7>/ sar_aio<7>/ PB<7>	SPI_DO	UART_RX	SPI_DO	-	0x5ab[7: 6]	0x588[7]	afe_0x bd[7]	0x58a[7]	0x58b[7]	0x58c[7]	afe_0x bf[7]	0x58e[7]
I2C_SDA/ PWM4_N/ UART_RTS/ PGA_P0/ PC<0>	GPIO	UART_RTS	PWM4_N	I2C_SDA	0x5ac[1: 0]	0x590[0]	afe_0x c0[0]	0x592[0]	0x593[0]	0x594[0]	afe_0x c2[0]	0x596[0]

Pin	Default function	Pad Function Mux				GPIO Setting						
		Register= 2	Register=1	Register= 0	Register	Input (R)	IE	OEN	Output	Polarity	DS	Act as GPIO
I2C_SCK/ PWM1_N/ PWM0/ PGA_N0/ PC<1>	GPIO	PWM0	PWM1_N	I2C_SCK	0x5ac[3: 2]	0x590[1]	afe_0x c0[1]	0x592[1]	0x593[1]	0x594[1]	afe_0x c2[1]	0x596[1]
PWM0/ UART_TX/ I2C_SDA/ XC32K_O/ PGA_P1/ PC<2>	GPIO	I2C_SDA	UART_TX	PWM0	0x5ac[5: 4]	0x590[2]	afe_0x c0[2]	0x592[2]	0x593[2]	0x594[2]	afe_0x c2[2]	0x596[2]
PWM1/ UART_RX/ I2C_SCK/ XC32K_I/ PGA_N1/ PC<3>	GPIO	I2C_SCK	UART_RX	PWM1	0x5ac[7: 6]	0x590[3]	afe_0x c0[3]	0x592[3]	0x593[3]	0x594[3]	afe_0x c2[3]	0x596[3]
PWM2/ UART_CTS/ PWM0_N/ sar_aio<8>/ PC<4>	GPIO	PWM0_N	UART_CTS	PWM2	0x5ad[1: 0]	0x590[4]	afe_0x c0[4]	0x592[4]	0x593[4]	0x594[4]	afe_0x c2[4]	0x596[4]

Pin	Default function	Pad Function Mux				GPIO Setting						
		Register= 2	Register=1	Register= 0	Register	Input (R)	IE	OEN	Output	Polarit y	DS	Act as GPIO
PWM3_N/ UART_RX/ ATSEL0/ sar_aio<9>/ PC<5>	GPIO	ATSEL0	UART_RX	PWM3_N	0x5ad[3: 2]	0x590[5]	afe_0x c0[5]	0x592[5]	0x593[5]	0x594[5]	afe_0x c2[5]	0x596[5]
RX_CYC2LNA/ ATSEL1/ PWM4_N/ PC<6>	GPIO	PWM4_N	ATSEL1	RX_CYC2L NA	0x5ad[5: 4]	0x590[6]	afe_0x c0[6]	0x592[6]	0x593[6]	0x594[6]	afe_0x c2[6]	0x596[6]
TX_CYC2PA/ ATSEL2/ PWM5_N/ PC<7>	GPIO	PWM5_N	ATSEL2	TX_CYC2P A	0x5ad[7: 6]	0x590[7]	afe_0x c0[7]	0x592[7]	0x593[7]	0x594[7]	afe_0x c2[7]	0x596[7]
RX_CYC2LNA/ UART_TX/ PD<0>	GPIO	UART_TX	-	RX_CYC2L NA	0x5ae[1: 0]	0x598[0]	0x599[0]	0x59a[0]	0x59b[0]	0x59c[0]	0x59d[0]	0x59e[0]
TX_CYC2PA/ UART_CTS/ PD<1>	GPIO	UART_CTS	-	TX_CYC2P A	0x5ae[3: 2]	0x598[1]	0x599[1]	0x59a[1]	0x59b[1]	0x59c[1]	0x59d[1]	0x59e[1]
SPI_CN/ PWM3/ PD<2>	SPI_CN	PWM3	-	SPI_CN	0x5ae[5: 4]	0x598[2]	0x599[2]	0x59a[2]	0x59b[2]	0x59c[2]	0x59d[2]	0x59e[2]

Pin	Default function	Pad Function Mux				GPIO Setting						
		Register= 2	Register=1	Register= 0	Register	Input (R)	IE	OEN	Output	Polarity	DS	Act as GPIO
PWM1_N/ UART_TX/ PD<3>	GPIO	UART_TX	-	PWM1_N	0x5ae[7: 6]	0x598[3]	0x599[3]	0x59a[3]	0x59b[3]	0x59c[3]	0x59d[3]	0x59e[3]
SWM/ PWM2_N/ PD<4>	GPIO	PWM2_N	-	SWM	0x5af[1: 0]	0x598[4]	0x599[4]	0x59a[4]	0x59b[4]	0x59c[4]	0x59d[4]	0x59e[4]
PWM0/ PWM0_N/ PD<5>	GPIO	PWM0_N	-	PWM0	0x5af[3: 2]	0x598[5]	0x599[5]	0x59a[5]	0x59b[5]	0x59c[5]	0x59d[5]	0x59e[5]
CN/ UART_RX/ ATSEL0/ PD<6>	GPIO	ATSEL0	UART_RX	CN	0x5af[5: 4]	0x598[6]	0x599[6]	0x59a[6]	0x59b[6]	0x59c[6]	0x59d[6]	0x59e[6]
SPI_CK/ UART_TX/ PD<7>	SPI_CK	UART_TX	-	SPI_CK	0x5af[7: 6]	0x598[7]	0x599[7]	0x59a[7]	0x59b[7]	0x59c[7]	0x59d[7]	0x59e[7]
MSDO/PE<0>	MSDO	-	-	MSDO	-	0x5a0[0]	0x5a1[0]	0x5a2[0]	0x5a3[0]	-	0x5a5[0]	0x5a6[0]
MCLK/PE<1>	MCLK	-	-	MCLK	-	0x5a0[1]	0x5a1[1]	0x5a2[1]	0x5a3[1]	-	0x5a5[1]	0x5a6[1]
MSCN/PE<2>	MSCN	-	-	MSCN	-	0x5a0[2]	0x5a1[2]	0x5a2[2]	0x5a3[2]	-	0x5a5[2]	0x5a6[2]

Pin	Default function	Pad Function Mux				GPIO Setting						
		Register=2	Register=1	Register=0	Register	Input (R)	IE	OEN	Output	Polarity	DS	Act as GPIO
MSDI/PE<3>	MSDI	-	-	MSDI	-	0x5a0[3]	0x5a1[3]	0x5a2[3]	0x5a3[3]	-	0x5a5[3]	0x5a6[3]

NOTE:

- IE: Input enable, high active. 1: enable input, 0: disable input.

- OEN: Output enable, low active. 0: enable output, 1: disable output.

- Register: Configure multiplexed functions in "Pad Function Mux" column.

- Output: configure GPO output.

- Input: read GPI input.

- DS: Drive strength. Default: 1 (high DS level).

- Act as GPIO: enable (1) or disable (0) GPIO function.

- Polarity: see Section 7.1.2 Connection Relationship Between GPIO and Related Modules.

- Priority: "Act as GPIO" has the highest priority. To configure as multiplexed function, disable GPIO function first.

- afe_0xbd, afe_0xbf, afe_0xc0 and afe_0xc2 marked in red color are analog registers; others are digital registers.

- For all unused GPIOs, corresponding "IE" must be set as 0.

- When SWS/PA<7> "IE" is set as 1, this pin must be fixed as pull-up/pull-down state (float state is not allowed).

- To use SAR ADC/low power comparator/PGA pin function, please refer to corresponding module sections.

7.1.1.2   Multiplexed Functions

Each pin listed in Table 7-1 acts as the function in the "**Default Function**" column by default.

- PA<7> acts as SWS function by default.
- PB<6:7>, PD<2,7> act as SPI function by default.
- PE<0:3> act as MSPI function by default.
- The other digital IOs act as GPIO function by default.

If a pin with multiplexed functions does not act as GPIO function by default, to use it as GPIO, first set the bit in "**Act as GPIO**" column as 1b'1. After GPIO function is enabled, if the pin is used as output, both the bits in "**IE**" and "**OEN**" columns should be set as 1b'0, then set the register value in the "**Output**" column; if the pin is used as input, both the bits in "**IE**" and "**OEN**" columns should be set as 1b'1, and the input data can be read from the register in the "**Input**" column.

To use a pin as certain multiplexed function (neither the default function nor GPIO function), first clear the bit in "**Act as GPIO**" column to disable GPIO function, and then configure "**Register**" in "Pad Function Mux" column to enable multiplexed function correspondingly.

Example 1: PWM0_N/ UART_RX/ PA<0>.

1. The pin acts as GPIO function by default.
   - If the pin is used as general output, both address 0x581[0] (IE) and 0x582[0] (OEN) should be set as 1b'0, then configure address 0x583[0] (Output).
   - If the pin is used as general input, both address 0x581[0] (IE) and 0x582[0] (OEN) should be set as 1b'1, and the input data can be read from address 0x580[0] (Input).
2. To use the pin as PWM0_N function, address 0x586[0] (Act as GPIO) should be set as 1b'0, and 0x5a8[1:0] (Register) should be set as 2b'01.
3. To use the pin as UART_RX function, address 0x586[0] (Act as GPIO) should be set as 1b'0, and 0x5a8[1:0] (Register) should be set as 2b'10.

Example 2: SWS/ UART_RTS/ PA<7>.

1. The pin acts as SWS function by default.
2. To use it as GPIO function, first set address 0x586[7] (Act as GPIO) as 1b'1.
   - If the pin is used as general output, both address 0x581[7] (IE) and 0x582[7] (OEN) should be set as 1b'0, then configure address 0x583[7] (Output).
   - If the pin is used as general input, both address 0x581[7] (IE) and 0x582[7] (OEN) should be set to 1b'1, and the input data can be read from address 0x580[7] (Input).
3. To use it as UART_RTS function, set address 0x586[7] (Act as GPIO) as 1b'0, and set 0x5a9[7:6] (Register) to 2b'01.

I2C can also be multiplexed with SPI interface, i.e. I2C_SDA/I2C_SCK can be multiplexed with SPI_DI (DI)/SPI_CK (CK) respectively.

To select multiplexed SPI/I2C function, please follow the steps below:

Step 1 Disable GPIO function by setting corresponding "Act as GPIO" as 1b'0.
Step 2 Select SPI/I2C function by setting corresponding "Register".
Step 3 Address 0x5b6[7:4] serve to select SPI or I2C output.

Step 4　Address 0x5b7[7:0] serve to select SPI input or I2C input.

Table 7-2 Select Multiplexed SPI/I2C

Pin with Multiplexed SPI/I2C	Act as GPIO	Register	SPI Input Select	I2C Input Select	SPI/I2C Output Select
DI/UART_CTS/PWM1/PA<3>	0x586[3]=0 Disable GPIO	0x5a8[7:6]=0 Select DI (I2C_SDA)	5b7[0] 1: as SPI input. 0: not as SPI input.	5b7[4] 1: as I2C input. 0: not as I2C input.	0x5b6[4] 1: as SPI/I2C output 0: not as SPI/I2C output
CK/UART_RTS/PWM2/PA<4>	0x586[4]=0 Disable GPIO	0x5a9[1:0]=0 Select CK (I2C_SCK)	5b7[1] 1: as SPI input. 0: not as SPI input.	5b7[5] 1: as I2C input. 0: not as I2C input.	0x5b6[5] 1: as SPI/I2C output 0: not as SPI/I2C output
SPI_DI/UART_RTS/lc_comp_ain<6>/sar_aio<6>/PB<6>	0x58e[6]=0 Disable GPIO	0x5ab[5:4]=1 Select SPI_DI (I2C_SDA) (default function)	5b7[2] 1: as SPI input. 0: not as SPI input.	5b7[6] 1: as I2C input. 0: not as I2C input.	0x5b6[6] 1: as SPI/I2C output 0: not as SPI/I2C output
SPI_CK/UART_TX/PD<7>	0x59e[7] =0 Disable GPIO	0x5af[7:6] =0 Select SPI_CK (I2C_SCK) (default function)	5b7[3] 1: as SPI input. 0: not as SPI input.	5b7[7] 1: as I2C input. 0: not as I2C input.	0x5b6[7] 1: as SPI/I2C output 0: not as SPI/I2C output

7.1.1.3　Drive Strength

The registers in the "DS" column are used to configure the corresponding pin's driving strength: "1" indicates maximum drive level, while "0" indicates minimal drive level.

The "DS" configuration will take effect when the pin is used as output. It's set as the strongest driving level by default. In actual applications, driving strength can be decreased to lower level if necessary.

- PA<5:7>, PB<0:3>: maximum = 8 mA ("DS"=1), minimum = 4 mA ("DS"=0)
- PB<4:7>: maximum = 16 mA ("DS"=1), minimum = 12 mA ("DS"=0)
- Other GPIOs (PA<0:1>, PC<0:7>, PD<0:5>, PD<7>, PE<0:3>): maximum = 4 mA ("DS"=1), minimum = 2 mA ("DS"=0)

7.1.2 Connection Relationship Between GPIO and Related Modules

GPIO can be used to generate GPIO interrupt signal for interrupt system, counting or control signal for Timer/Counter module, or GPIO2RISC interrupt signal for interrupt system.

For the "Exclusive Or (XOR)" operation result for input signal from any GPIO pin and respective "Polarity" value, on one hand, it takes "And" operation with "irq" and generates GPIO interrupt request signal; on the other hand, it takes "And" operation with "m0/m1/m2", and generates counting signal in Mode 1 or control signal in Mode 2 for Timer0/Timer1/Timer2, or generates GPIO2RISC[0]/GPIO2RISC[1] interrupt request signal.

GPIO interrupt request signal = | ((input ^ polarity) & irq);

Counting (Mode 1) or control (Mode 2) signal for Timer0 = | ((input ^ polarity) & m0);

Counting (Mode 1) or control (Mode 2) signal for Timer1 = | ((input ^ polarity) & m1);

Counting (Mode 1) or control (Mode 2) signal for Timer2 = | ((input ^ polarity) & m2);

GPIO2RISC[0] interrupt request signal = | ((input ^ polarity) & m0);

GPIO2RISC[1] interrupt request signal = | ((input ^ polarity) & m1).

Figure 7-1 Logic Relationship Between GPIO and Related Modules
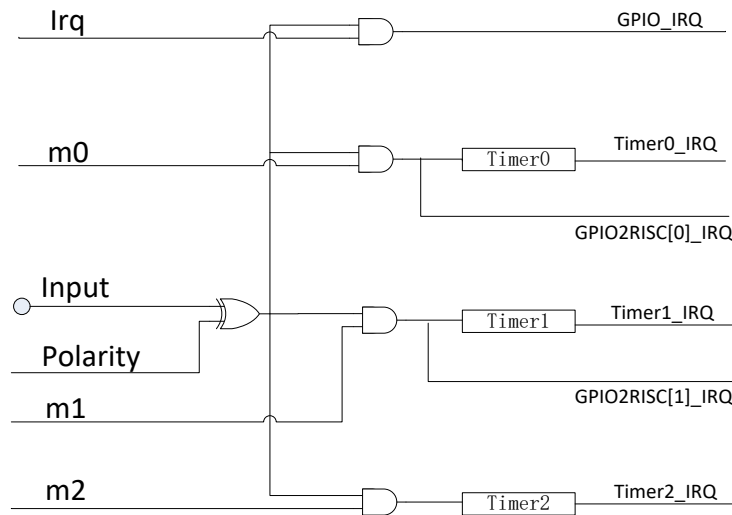
Please refer to Table 7-3 and Table 6-1 to learn how to configure GPIO for interrupt system or Timer/Counter (Mode 1 or Mode 2).

Enable GPIO function

First enable GPIO function, enable IE and disable OEN. Please see Section 7.1.1 Basic Configuration.

GPIO IRQ signal:

Select GPIO interrupt trigger edge (positive edge or negative edge) via configuring "**Polarity**", and set corresponding GPIO interrupt enabling bit "**Irq**".

Then set address 0x5b5[3] (irq_enable) to enable GPIO IRQ.

Finally enable GPIO interrupt (irq_gpio) via address 0x642[2].

User can read addresses 0x5e0 ~ 0x5e3 to see which GPIO asserts GPIO interrupt request signal. Note: 0x5e0[7:0] --> PA<7>~PA<0>, 0x5e1[7:0] --> PB<7>~PB<0>, 0x5e2[7:0] --> PC<7>~PC<0>, 0x5e3[7:0] --> PD<7>~PD<0>.

Timer/Counter counting or control signal:

Configure "**Polarity**". In Timer Mode 1, it determines GPIO edge when Timer Tick counting increases. In Timer Mode 2, it determines GPIO edge when Timer Tick starts counting.

Then set "**m0/m1/m2**" to specify the GPIO which generates counting signal (Mode 1)/control signal (Mode 2) for Timer0/Timer1/Timer2.

User can read addresses 0x5e8~0x5eb/0x5f0~0x5f3/0x5f8~0x5fb to see which GPIO asserts counting signal (in Mode 1) or control signal (in Mode 2) for Timer0/Timer1/Timer2. Note: Timer0: 0x5e8[7:0] --> PA<7>~PA<0>, 0x5e9[7:0] --> PB<7>~PB<0>, 0x5ea[7:0] --> PC<7>~PC<0>, 0x5eb[7:0] --> PD<7>~PD<0>; Timer1: 0x5f0[7:0] --> PA<7>~PA<0>, 0x5f1[7:0] --> PB<7>~PB<0>, 0x5f2[7:0] --> PC<7>~PC<0>, 0x5f3[7:0] --> PD<7>~PD<0>; Timer2: 0x5f8[7:0] --> PA<7>~PA<0>, 0x5f9[7:0] --> PB<7>~PB<0>, 0x5fa[7:0] --> PC<7>~PC<0>, 0x5fb[7:0] --> PD<7>~PD<0>.

GPIO2RISC IRQ signal:

Select GPIO2RISC interrupt trigger edge (positive edge or negative edge) via configuring "**Polarity**", and set corresponding GPIO enabling bit **"m0"/"m1"**.

Enable GPIO2RISC[0]/GPIO2RISC[1] interrupt, i.e. "gpio2risc[0]" (address 0x642[5]) / "gpio2risc[1]"(address 0x642[6]).

Table 7-3 GPIO Lookup Table 2

Pin	Input (R)	Polarity 1: Active Low 0: Active High	Irq	m0	m1	m2
PA<0>	0x580[0]	0x584[0]	0x587[0]	0x5b8[0]	0x5c0[0]	0x5c8[0]
PA<1>	0x580[1]	0x584[1]	0x587[1]	0x5b8[1]	0x5c0[1]	0x5c8[1]
PA<2>	0x580[2]	0x584[2]	0x587[2]	0x5b8[2]	0x5c0[2]	0x5c8[2]
PA<3>	0x580[3]	0x584[3]	0x587[3]	0x5b8[3]	0x5c0[3]	0x5c8[3]
PA<4>	0x580[4]	0x584[4]	0x587[4]	0x5b8[4]	0x5c0[4]	0x5c8[4]
PA<5>	0x580[5]	0x584[5]	0x587[5]	0x5b8[5]	0x5c0[5]	0x5c8[5]
PA<6>	0x580[6]	0x584[6]	0x587[6]	0x5b8[6]	0x5c0[6]	0x5c8[6]
PA<7>	0x580[7]	0x584[7]	0x587[7]	0x5b8[7]	0x5c0[7]	0x5c8[7]
PB<0>	0x588[0]	0x58c[0]	0x58f[0]	0x5b9[0]	0x5c1[0]	0x5c9[0]
PB<1>	0x588[1]	0x58c[1]	0x58f[1]	0x5b9[1]	0x5c1[1]	0x5c9[1]
PB<2>	0x588[2]	0x58c[2]	0x58f[2]	0x5b9[2]	0x5c1[2]	0x5c9[2]
PB<3>	0x588[3]	0x58c[3]	0x58f[3]	0x5b9[3]	0x5c1[3]	0x5c9[3]

Pin	Input (R)	Polarity 1: Active Low 0: Active High	Irq	m0	m1	m2
PB<4>	0x588[4]	0x58c[4]	0x58f[4]	0x5b9[4]	0x5c1[4]	0x5c9[4]
PB<5>	0x588[5]	0x58c[5]	0x58f[5]	0x5b9[5]	0x5c1[5]	0x5c9[5]
PB<6>	0x588[6]	0x58c[6]	0x58f[6]	0x5b9[6]	0x5c1[6]	0x5c9[6]
PB<7>	0x588[7]	0x58c[7]	0x58f[7]	0x5b9[7]	0x5c1[7]	0x5c9[7]
PC<0>	0x590[0]	0x594[0]	0x597[0]	0x5ba[0]	0x5c2[0]	0x5ca[0]
PC<1>	0x590[1]	0x594[1]	0x597[1]	0x5ba[1]	0x5c2[1]	0x5ca[1]
PC<2>	0x590[2]	0x594[2]	0x597[2]	0x5ba[2]	0x5c2[2]	0x5ca[2]
PC<3>	0x590[3]	0x594[3]	0x597[3]	0x5ba[3]	0x5c2[3]	0x5ca[3]
PC<4>	0x590[4]	0x594[4]	0x597[4]	0x5ba[4]	0x5c2[4]	0x5ca[4]
PC<5>	0x590[5]	0x594[5]	0x597[5]	0x5ba[5]	0x5c2[5]	0x5ca[5]
PC<6>	0x590[6]	0x594[6]	0x597[6]	0x5ba[6]	0x5c2[6]	0x5ca[6]
PC<7>	0x590[7]	0x594[7]	0x597[7]	0x5ba[7]	0x5c2[7]	0x5ca[7]
PD<0>	0x598[0]	0x59c[0]	0x59f[0]	0x5bb[0]	0x5c3[0]	0x5cb[0]
PD<1>	0x598[1]	0x59c[1]	0x59f[1]	0x5bb[1]	0x5c3[1]	0x5cb[1]
PD<2>	0x598[2]	0x59c[2]	0x59f[2]	0x5bb[2]	0x5c3[2]	0x5cb[2]
PD<3>	0x598[3]	0x59c[3]	0x59f[3]	0x5bb[3]	0x5c3[3]	0x5cb[3]
PD<4>	0x598[4]	0x59c[4]	0x59f[4]	0x5bb[4]	0x5c3[4]	0x5cb[4]
PD<5>	0x598[5]	0x59c[5]	0x59f[5]	0x5bb[5]	0x5c3[5]	0x5cb[5]
PD<6>	0x598[6]	0x59c[6]	0x59f[6]	0x5bb[6]	0x5c3[6]	0x5cb[6]
PD<7>	0x598[7]	0x59c[7]	0x59f[7]	0x5bb[7]	0x5c3[7]	0x5cb[7]

7.1.3 Pull-Up/Pull-Down Resistor

The GPIOs including PA<0> ~ PD<7> support configurable pull-up resistor of rank x1 and x100 or pull-down resistor of rank x10 which are all disabled by default. Analog registers afe_0x0e<7:0>~afe_0x15<7:0> serve to control the pull-up/pull-down resistor for each GPIO.

The DP pin also supports 1.5 kΩ pull-up resistor for USB use. The 1.5 kΩ pull up resistor is disabled by default and can be enabled by setting analog register afe_0x0b<7> as 1b'1. For the DP/PA<6> pin, user can only enable either 1.5 kΩ pull-up resistor or pull-up resistor of rank x1/x100 / pull-down resistor of rank x10 at the same time. Please refer to Table 7-4 for details.

Take the PA<0> for example: Setting analog register afe_0x0e<1:0> to 2b'01/2b'11/2b'10 is to respectively enable pull-up resistor of rank x100/pull-up resistor of rank x1/pull-down resistor of rank x10 for PA<0>; Clearing the two bits (default value) disables pull-up and pull-down resistor for PA<0>.

Table 7-4 Analog Registers for Pull-Up/Pull-Down Resistor Control

Address	Mnemonic	Default	Description
afe_0x0b<7>	dp_pullup_res_3v	0	1.5k (Typ.) pull-up resistor for USB DP PAD 0: disable 1: enable
Rank **Typical value (depend on actual application)** x1 18 kOhm x10 160 kOhm x100 1 MOhm			
afe_0x0e<7:0>	a_sel<7:0>	00000000	PA<3:0> pull up and down select: <7:6>: PA<3> <5:4>: PA<2> <3:2>: PA<1> <1:0>: PA<0> 00: Null 01: x100 pull up 10: x10 pull down 11: x1 pull up
afe_0x0f<7:0>	a_sel<15:8>	00000000	PA<7:4> pull up and down select:

Address	Mnemonic	Default	Description
			<7:6>: PA<7> <5:4>: PA<6> <3:2>: PA<5> <1:0>: PA<4> 00: Null 01: x100 pull up 10: x10 pull down 11: x1 pull up
afe_0x10<7:0>	b_sel<7:0>	00000000	PB<3:0> pull up and down select: 00: Null 01: x100 pull up 10: x10 pull down 11: x1 pull up
afe_0x11<7:0>	b_sel<15:8>	00000000	PB<7:4> pull up and down select: 00: Null 01: x100 pull up 10: x10 pull down 11: x1 pull up
afe_0x12<7:0>	c_sel<7:0>	00000000	PC<3:0> pull up and down select: 00: Null 01: x100 pull up 10: x10 pull down 11: x1 pull up
afe_0x13<7:0>	c_sel<15:8>	00000000	PC<7:4> pull up and down select: 00: Null 01: x100 pull up 10: x10 pull down 11: x1 pull up

Address	Mnemonic	Default	Description
afe_0x14<7:0>	d_sel<7:0>	00000000	PD<3:0> pull up and down select: 00: Null 01: x100 pull up 10: x10 pull down 11: x1 pull up
afe_0x15<7:0>	d_sel<15:8>	00000000	PD<7:4> pull up and down select: <7:6>: PD<7> <5:4>: PD<6> <3:2>: PD<5> <1:0>: PD<4> 00: Null 01: x100 pull up 10: x10 pull down 11: x1 pull up

7.2 SWM and SWS

The TLSR8359 supports Single Wire interface. SWM (Single Wire Master) and SWS (Single Wire Slave) represent the master and slave device of the single wire communication system developed by Telink. The maximum data rate can be up to 2 Mbps.

7.3 I2C

The TLSR8359 embeds I2C hardware module, which could act as Master mode or Slave mode. I2C is a popular inter-IC interface requiring only 2 bus lines, a serial data line (SDA) and a serial clock line (SCL).

7.3.1 Communication Protocol

Telink I2C module supports standard mode (100 kbps) and Fast-mode (400 kbps) with restriction that system clock must be by at least 10x of data rate.

Two wires, SDA and SCL (SCK) carry information between Master device and Slave device connected to the bus. Each device is recognized by unique address (ID). Master device is the device which initiates a data transfer on the bus and generates the clock signals to permit that transfer. Slave device is the device addressed by a Master.

Both SDA and SCL are bidirectional lines connected to a positive supply voltage via a pull-up resister. It's recommended to use external 3.3 kOhm pull-up resistor. For standard mode, the internal pull-up resistor of rank x1 can be used instead of the external 3.3 kOhm pull-up.

When the bus is free, both lines are HIGH. It's noted that data in SDA line must keep stable when clock signal in SCL line is at high level, and level state in SDA line is only allowed to change when clock signal in SCL line is at low level.

Figure 7-2 I2C Timing Chart
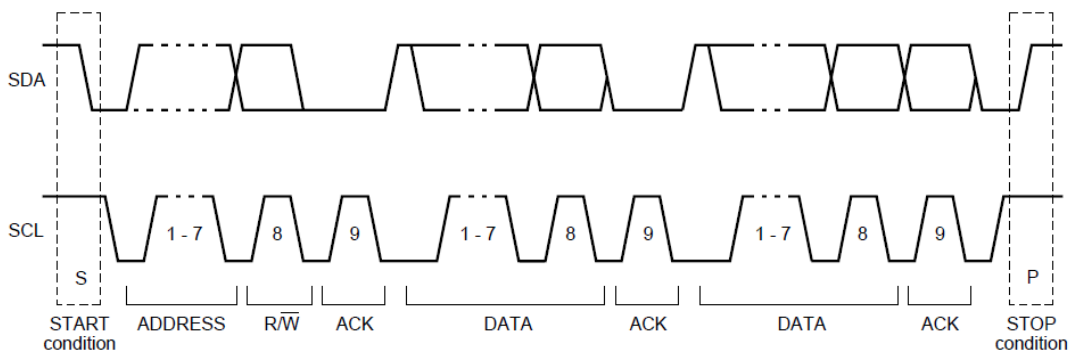
7.3.2 Register Table

Table 7-5 Register Configuration for I2C

Address	Name	R/W	Description	Reset Value
0x00	I2CSP	RW	I2C master clock speed	0x1f
0x01	I2C_ID	RW	[7:1] I2C ID	0x5c
0x02	I2CMST	RW	[0]: master busy [1]: master packet busy [2]: master received status 0 for ACK; 1 for NAK	0x00
0x03	I2CSCT0	RW	[0]: address auto increase enable	0x01

Address	Name	R/W	Description	Reset Value
			[1]: I2C master enable [2]: enable Mapping Mode [3]: r_clk_stretch_en, suspend transmission by pulling SCL down to low level, and continue transmission after SCL is released to high level	
0x04	I2CAD	RW	[7:0] data buffer in master mode	0x5a
0x05	I2CDW	RW	[7:0] Data buffer in master mode	0xf1
0x06	I2CDR	RW	[7:0] Data buffer for Read or Write in master mode	0x00
0x07	I2CSCT1	RW	[0]: launch ID cycle [1]: launch address cycle (send I2CAD data) [2]: launch data write cycle [3]: launch data read cycle For Master Write: 0: I2CAD&I2CDW, 1: I2CAD&I2CDW&I2CDR) To write 3 bytes: bit[3]=1; To write 2 bytes: bit[3]=0. For Master Read: always 1. [4]: launch start cycle [5]: launch stop cycle [6]: enable read ID [7]: enable ACK in read command	0x00
0xe0	I2CMAP_HADR	R	[6:0] I2C read address	0x00
0xe1	HOSR_ADR_L	RW	Low byte of Mapping mode buffer address	0x80
0xe2	HOSR_ADR_M	RW	Middle byte of Mapping mode buffer address	0xd7
0xe3	HOSR_ADR_H	RW	High byte of Mapping mode buffer address	0x00
0xe4	I2CMAP_HOST	RW	[0]: host_cmd_irq_o,	0x00

Address	Name	R/W	Description	Reset Value
			I2C host operation has happened. Write 1 to clear. [1]: host_rd_tag_o, I2C host operation has happened and is read operation. Write 1 to clear.	

7.3.3 I2C Slave Mode

I2C module of the TLSR8359 acts as Slave mode by default. I2C slave address can be configured via register I2C_ID (address 0x01) [7:1].

Figure 7-3 Byte Consisted of Slave Address and R/W Flag Bit
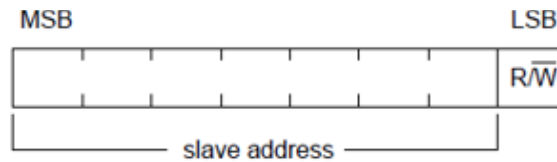
I2C slave mode supports two sub modes including Direct Memory Access (DMA) mode and Mapping mode, which is selectable via address 0x03[2].

In I2C Slave mode, Master could initiate transaction anytime. I2C slave module will reply with ACK automatically. To monitor the start of I2C transaction, user could set interrupt from GPIO for SCA or SCL.

7.3.3.1 DMA Mode

In DMA mode, other devices (Master) could access (read/write) designated address in Register and/or SRAM of the TLSR8359 according to I2C protocol. I2C module of the TLSR8359 will execute the read/write command from I2C master automatically. But user needs to notice that the system clock shall be at least 10x faster than I2C bit rate.

The access address designated by Master is offset by 0x800000. In the TLSR8359, Register address starts from 0x800000 and SRAM address starts from 0x840000. For example, if Addr High (AddrH) is 0x04, Addr Middle (AddrM) is 0x00, and Addr Low (AddrL) is 0xcc, the real address of accessed data is 0x8400cc.

In DMA mode, Master could read/write data byte by byte. The designated access address is initial address and it supports auto increment by setting address 0x03[0] to 1b'1.

Read Format in DMA mode
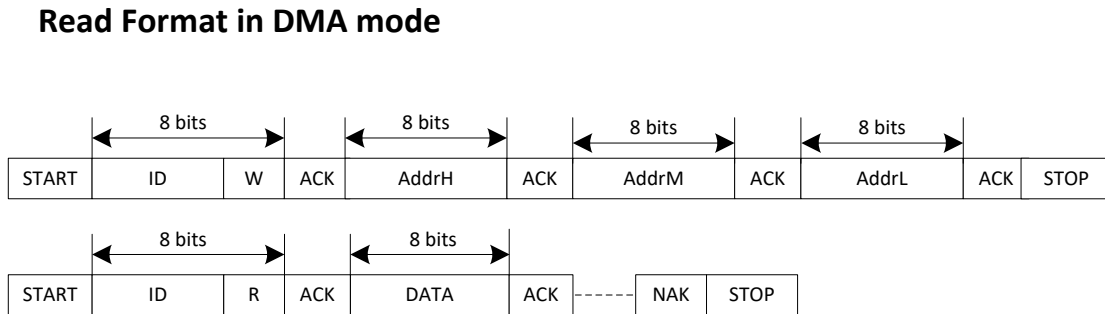
Figure 7-5 Write Format in DMA Mode

Write Format in DMA mode
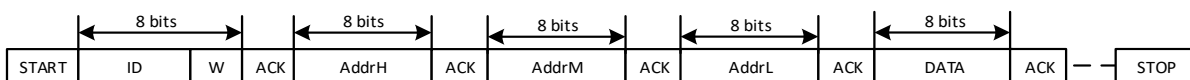
7.3.3.2 Mapping Mode

Mapping mode could be enabled via setting register I2CSCT0 (address 0x03)[2] to 1b'1.

In Mapping mode, data written and read by I2C master will be redirected to specified 128-byte buffer in SRAM. User could specify the initial address of the buffer by configuring registers HOSR_ADR_L (address 0xe1, lower byte), HOSR_ADR_M (address 0xe2, middle byte) and HOSR_ADR_H (address 0xe3, higher byte). The first 64-byte buffer is for written data and following 64-byte buffer is for read data. Every time the data access will start from the beginning of the Write-buffer/Read-buffer after I2C stop condition occurs. The last accessed data address could be checked in register I2CMAP_HADR (address 0xe0) [6:0] which is only updated after I2C STOP occurs.

Figure 7-6 Read Format in Mapping Mode

Read Format in mapping mode
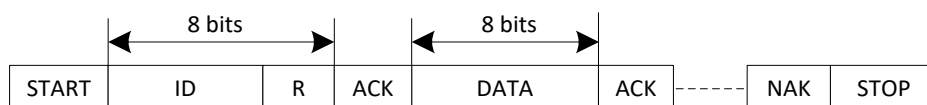
Figure 7-7 Write Format in Mapping Mode

Write Format in mapping mode
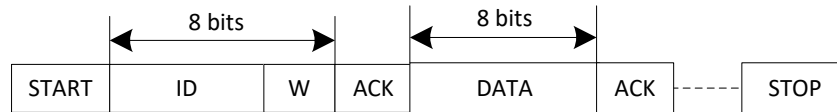
7.3.4 I2C Master Mode

Address 0x03[1] should be set to 1b'1 to enable I2C master mode for the TLSR8359.

Address 0x00 serves to set I2C Master clock: F_{I2C} = (System Clock / (4 *clock speed configured in address 0x00).

A complete I2C protocol contains START, Slave Address, R/W bit, data, ACK and STOP. Slave address could be configured via address 0x01[7:1].

I2C Master (i.e. I2C module of the TLSR8359) could send START, Slave Address, R/W bit, data and STOP cycle by configuring address 0x07. I2C master will send enabled cycles in the correct sequence.

Address 0x02 serves to indicate whether Master/Master packet is busy, as well as Master received status. Bit[0] will be set to 1 when one byte is being sent, and the bit can be automatically cleared after a start signal/ address byte/acknowledge signal/data /stop signal is sent. Bit[1] is set to 1 when the start signal is sent, and the bit will be automatically cleared after the stop signal is sent. Bit[2] indicates whether to succeed in sending acknowledgement signal.

7.3.4.1 I2C Master Write Transfer

I2C Master has 3-byte buffer for write data, which are I2CAD (0x04), I2CDW (0x05) and I2CDR (0x06). Write transfer will be completed by I2C master module.

For example, to implement an I2C write transfer with 3-byte data, which contains START, Slave Address, Write bit, ack from Slave, 1st byte, ack from slave, 2nd byte, ack from slave, 3rd byte, ack from slave and STOP, user needs to configure I2C slave address to I2C_ID (0x01) [7:1], 1st byte data to I2CAD, 2nd byte data to I2CDW and 3rd byte to I2CDR. To start I2C write transfer, I2CSCT1 (0x07) is configured to 0x3f (0011 1111). I2C Master will launch START, Slave address, Write bit, load ACK to I2CMST (0x02) [2], send I2CAD data, load ACK to I2CMST[2], send I2CDW data, load ACK to I2CMST[2], send I2CDR data, load ACK to I2CMST[2] and then STOP sequentially.

For I2C write transfer whose data are more than 3 bytes, user could split the cycles according to I2C protocol.

7.3.4.2 I2C Master Read Transfer

I2C Master has one byte buffer for read data, which is I2CDR (0x06). Read transfer will be completed by I2C Master.

For example, to implement an I2C read transfer with 1 byte data, which contains START, Slave Address, Read bit, Ack from Slave, 1st byte from Slave, Ack by master and STOP, user needs to configure I2C slave address to I2C_ID (0x01) [7:1]. To start I2C read transfer, I2CSCT1 (0x07) is configured to 0xf9 (1111 1001). I2C Master will launch START, Slave address, Read bit, load ACK to I2CMST (0x02) [2], load data to I2CDR, reply ACK and then STOP sequentially.

For I2C read transfer whose data are more than 1 byte, user could split the cycles according to I2C protocol.

7.3.5 I2C and SPI Usage

I2C hardware and SPI hardware modules in the chip share part of the hardware, as a result, when both hardware interfaces are used, the restrictions listed within this section need to be taken into consideration.

I2C and SPI hardware cannot be used as Slave at the same time.

The other cases are supported, including:

- · I2C Slave and SPI Master can be used at the same time.
- · I2C Master and SPI Slave can be used at the same time.
- · I2C and SPI can be used as Master at the same time.

Please refer to corresponding SDK instructions for details.

7.4 SPI

The TLSR8359 embeds SPI (Serial Peripheral interface), which could act as Master mode or Slave mode. SPI is a high-speed, half-duplex and synchronous communication bus requiring 4 bus lines including a chip select (CS) line, a data input (DI) line, a data output (DO) line and a clock (CK) line.

7.4.1 Register Table

Table 7-6 Register Configuration for SPI

Address	Name	R/W	Description	Reset Value
0x08	SPIDAT	RW	[7:0]: SPI data access	0x00
0x09	SPICT	RW	[0]: mst_csn, control SPI_CSN output when SPI acts as Master [1]: enable master mode [2]: spi data output disable [3]: 1 for read command; 0 for write command [4]: address auto increase [5]: share_mode [6]: busy status	0x11
0x0a	SPISP	RW	[6:0]: SPI clock speed [7]: SPI function mode, p_csn, p_scl, p_sda and p_sdo function as SPI if 1	0x05
0x0b	SPIMODE	RW	[0]: inverse SPI clock output [1]: data delay half clk	0x00

7.4.2 SPI Master Mode

SPI for the TLSR8359 supports both master mode and slave mode and acts as slave mode by default. Address 0x09 bit[1] should be set to 1b'1 to enable SPI Master mode. Register SPISP is to configure SPI pin and clock: setting address 0x0a bit[7] to 1 is to enable SPI function mode, and corresponding pins can be used as SPI pins; SPI clock = system clock/((clock speed configured in address 0x0a bit[6:0] +1)*2).

Address 0x08 serves as the data register. One reading/writing operation of 0x08 enables the SPI_CK pin to generate 8 SPI clock cycles.

Telink SPI supports four standard working modes: Mode 0~Mode 3. Register SPIMODE (address 0x0b) serves to select one of the four SPI modes:

Table 7-7 SPI Master Mode

SPI Mode	CPOL/CPHA	SPIMODE register (Address 0x0b)
Mode 0	CPOL=0, CPHA=0	bit[0]=0, bit[1]=0
Mode 1	CPOL=0, CPHA=1	bit[0]=0, bit[1]=1
Mode 2	CPOL=1, CPHA=0	bit[0]=1, bit[1]=0
Mode 3	CPOL=1, CPHA=1	bit[0]=1, bit[1]=1
CPOL: Clock Polarity When CPOL=0, SPI_CLK keeps low level in idle state; When CPOL=1, SPI_CLK keeps high level in idle state. CPHA: Clock Phase When CPHA=0, data is sampled at the first edge of clock period When CPHA=1, data is sampled at the latter edge of clock period		

Address 0x09 bit[0] is to control the CS line: when the bit is set to 1, the CS level is high; when the bit is cleared, the CS level is low.

Address 0x09 bit[2] is the disabling bit for SPI Master output. When the bit is cleared, MCU writes data into address 0x08, then the SPI_DO pin outputs the data bit by bit during the 8 clock cycles generated by the SPI_CK pin. When the bit is set to 1b'1, SPI_DO output is disabled.

Address 0x09 bit[3] is the enabling bit for SPI Master reading data function. When the bit is set to 1b'1, MCU reads the data from address 0x08, then the input data from the SPI_DI pin is shifted into address 0x08 during the 8 clock cycles generated by the SPI_CK pin. When the bit is cleared, SPI Master reading function is disabled.

Address 0x09[5] is the enabling bit for share mode, i.e. whether SPI_DI and SPI_DO share one common line.

Users can read address 0x09 bit[6] to get SPI busy status, i.e. whether the 8 clock pulses have been sent.

7.4.3 SPI Slave Mode

SPI for the TLSR8359 acts as slave mode by default. SPI Slave mode supports DMA. User could access registers of the TLSR8359 by SPI interface. It's noted that system clock of TLSR8359 shall be at least 5x faster than SPI clock for reliable connection. Address 0x0a should be written with data 0xa5 by the SPI host to activate SPI slave mode. SPI salve only supports Mode0 and Mode3.

Table 7-8 SPI Slave Mode

SPI Slave Mode	CPOL/CPHA
Mode 0	CPOL=0, CPHA=0
Mode 3	CPOL=1, CPHA=1
Receive data at positive edge of SPI MCLK clock. Send data at negative edge of SPI MCLK clock.	

Address 0x09[4] is dedicated for SPI Slave mode and indicates address auto increment. SPI write command format and read command format are illustrated in Figure 7-8:

Figure 7-8 SPI Write/Read Command Format

SPI Write Format

| SPIDI | Addr(High) | Addr(Middle) | Addr(Low) | CMD(Write) 0x00 | Data0 | Data1 | Data.... |

SPIDO

SPI Read Format

| SPIDI | Addr(High) | Addr(Middle) | Addr(Low) | CMD(Read) 0x80 |

| SPIDO | Data0 | Data1 | Data.... |
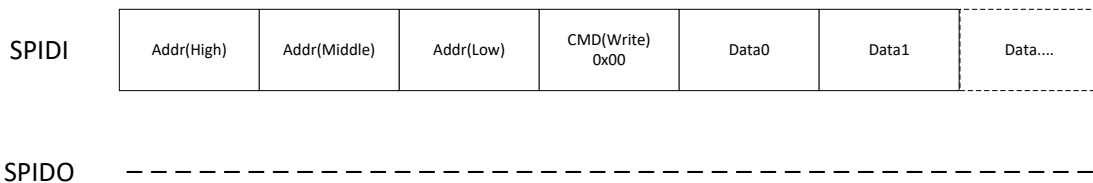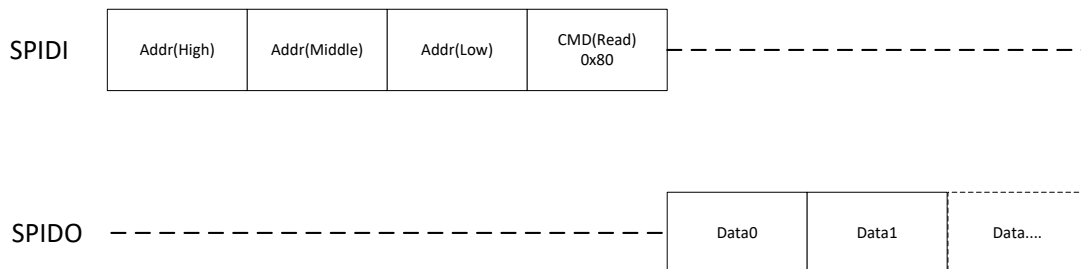
7.4.4 I2C and SPI Usage

I2C hardware and SPI hardware modules in the chip share part of the hardware, as a result, when both hardware interfaces are used, certain restrictions apply.

See Section 7.3.5 I2C and SPI Usage for detailed instructions.

7.5 UART

The TLSR8359 embeds UART (Universal Asynchronous Receiver/Transmitter) to implement full-duplex transmission and reception via UART TX and RX interface. Both TX and RX interface are 4-layer FIFO (First In First Out) interface.

Hardware flow control is supported via RTS and CTS.

Figure 7-9 UART Communication
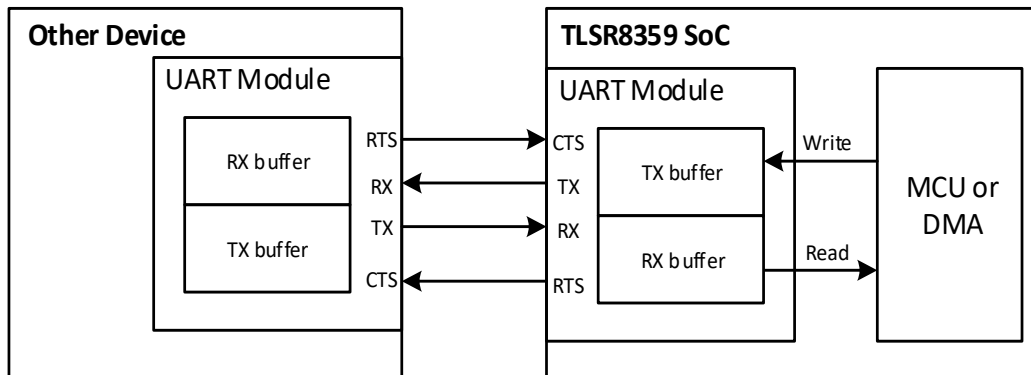
As shown in Figure 7-9, data to be sent is first written into TX buffer by MCU or DMA, then UART module transmits the data from TX buffer to other device via pin TX. Data to be read from other device is first received via pin RX and sent to RX buffer, then the data is read by MCU or DMA.

If RX buffer of the TLSR8359 UART is close to full, the TLSR8359 will send a signal (configurable high or low level) via pin RTS to inform other device that it should stop sending data. Similarly, if the TLSR8359 receives a signal from pin CTS, it indicates that RX buffer of other device is close to full and the TLSR8359 should stop sending data.

Table 7-9 Register Configuration for UART

Address	Name	R/W	Description	Reset Value
0x90	UART_DATA_BUF0	RW	write/read buffer[7:0]	0x00
0x91	UART_DATA_BUF1	RW	Write/read buffer[15:8]	0x00
0x92	UART_DATA_BUF2	RW	Write/read buffer[23:16]	0x00
0x93	UART_DATA_BUF3	RW	Write/read buffer[31:24]	0x00
0x94	UART_CLK_DIV_L	RW	[7:0]: uart clk div register uart_clk_div[7:0]	0xff
0x95	UART_CLK_DIV_H	RW	[6:0]: uart clk div register uart_clk_div[14:8] uart_sclk = sclk/(uart_clk_div[14:0]+1) [7]: uart_clk_div[15] 1: enable clock divider, 0: disable.	0x0f
0x96	UART_CTRL0	RW	[3:0] bwpc, bit width, should be larger than 2 Baudrate = uart_sclk/(bwpc+1) [4] rx dma enable [5] tx dma enable [6] rx interrupt enable [7]tx interrupt enable	0x0f
0x97	UART_CTRL1	RW	[0] cts select, 0: cts_i, 1: cts _i inverter [1]:cts enable, 1: enable, 0: disable [2]:Parity, 1: enable, 0 :disable [3]: even Parity or odd [5:4]: stop bit 00: 1 bit, 01: 1.5bit, 1x: 2bits [6]: ttl [7]: uart tx, rx loopback	0x0e
0x98	UART_CTRL2	RW	[3:0] rts trig level [4] rts Parity	0xa5

Address	Name	R/W	Description	Reset Value
			[5] rts manual value [6] rts manual enable [7] rts enable	
0x99	UART_CTRL3	RW	[3:0]: rx_irq_trig level [7:4] tx_irq_trig level	0x44
0x9a	UART_RXTIMEOUT_O_L	RW	[7:0]: R_rxtimeout_o[7:0] The setting is transfer one bytes need cycles base on uart_clk. For example, if transfer one bytes (1 start bit+8bits data+1 priority bit+2 stop bits) total 12 bits, this register setting should be (bwpc+1)*12.	0xc0
0x9b	UART_RXTIMEOUT_O_H	RW	[1:0]: R_rxtimeout_o[9:8] 2'b00:rx timeout time is r_rxtimeout[7:0] 2'b01:rx timeout time is r_rxtimeout[7:0]*2 2'b10:rx timeout time is r_rxtimeout[7:0]*3 3'b11: rx timeout time is r_rxtimeout[7:0]*4 R_rxtimeout is for rx dma to decide the end of each transaction. Supposed the interval between each byte in one transaction is very short. [5]: rsvd [6]: mask_txdone [7]: mask_err	0x01
0x9c	UART_BUFCNT	R	[3:0]: rx_buf_cnt [7:4]: tx_buf_cnt	0x00
0x9d	UART_STATUS	R	[2:0] rbcnt [3] irq_o [6:4]wbcnt [6] write 1 to clear rx	0x00

Address	Name	R/W	Description	Reset Value
			[7] rx_err, write 1 to clear tx	
0x9e	UART_TXRX_STATUS	R	[0] txdone [1] tx_buf_irq [2] rxdone [3] rx_buf_irq	0x00
0x9f	UART_STATE	R	[2:0] tstate_i [7:4] rstate_i	0x00

Addresses 0x90~0x93 serve to write data into TX buffer or read data from RX buffer.

Addresses 0x94~0x95 serve to configure UART clock.

Address 0x96 serves to set baud rate (bit[3:0]), enable RX/TX DMA mode (bit[4:5]), and enable RX/TX interrupt (bit[6:7]).

Address 0x97 mainly serves to configure CTS. Bit[1] should be set to 1b'1 to enable CTS. Bit[0] serves to configure CTS signal level. Bit[2:3] serve to enable parity bit and select even/odd parity. Bit[5:4] serve to select 1/1.5/2 bits for stop bit. Bit[6] serves to configure whether RX/TX level should be inverted.

Address 0x98 serves to configure RTS. Bit[7] and Bit[3:0] serve to enable RTS and configure RTS signal level.

Address 0x99 serves to configure the number of bytes in RX/TX buffer to trigger interrupt.

The number of bytes in RX/TX buffer can be read from address 0x9c.

7.6 USB

The TLSR8359 has a full-speed (12 Mbps) USB interface for communicating with other compatible digital devices. The USB interface acts as a USB peripheral, responding to requests from a master host controller. The chip contains internal 1.5kohm pull up resistor for the DP pin, which can be enabled via analog register afe_0x0b<7>.

Telink USB interface supports the Universal Serial Bus Specification, Revision v2.0 (USB v2.0 Specification).

The chip supports 9 endpoints, including control endpoint 0 and 8 configurable data endpoints. Endpoint 1, 2, 3, 4, 7 and 8 can be configured as input endpoint, while endpoint 5 and 6 can be configured as output endpoint. In non-audio class applications, each endpoint can be configured as bulk, interrupt and iso mode. For control endpoint 0, the chip's hardware vendor command is configurable.

Optional suspend mode:

- Selectable as USB suspend mode or chip suspend mode, support remote wakeup.
- Current draw in suspend mode complied with USB v2.0 Specification.
- USB pins (DM, DP) can be used as GPIO function in suspend mode.
- Resume and detach detect: Recognize USB device by detecting the voltage on the DP pin with configurable 1.5k pull-up resistor.
- USB pins configurable as wakeup GPIOs.

The USB interface belongs to an independent power domain, and it can be configured to power down independently.

8. PWM

The TLSR8359 supports 6-channel PWM (Pulse-Width-Modulation) output. Each PWM#n (n=0~5) has its corresponding inverted output at PWM#n_N pin.

8.1 Register Table

Table 8-1 Register Table for PWM

Address	Mnemonic	Type	Description	Reset Value
0x780	PWM_EN	R/W	[1]: 0--disable PWM1, 1--enable PWM1 [2]: 0--disable PWM2, 1--enable PWM2 [3]: 0--disable PWM3, 1--enable PWM3 [4]: 0--disable PWM4, 1--enable PWM4 [5]: 0--disable PWM5, 1--enable PWM5	0x00
0x781	PWM_EN0	R/W	[0]: 0--disable PWM0, 1--enable PWM0	0x00
0x782	PWM_CLKDIV	R/W	Set PWM_clk: (PWM_CLKDIV+1)*sys_clk	0x00
0x783	PWM_MODE	R/W	[3:0]: PWM0 mode select 0000-pwm0 normal mode 0001-pwm0 count mode 0011-pwm0 IR mode 0111-pwm0 IR FIFO mode 1111-pwm0 IR DMA FIFO mode	0x00
0x784	PWM_CC0	R/W	[5:0]:1`b1 invert PWM output	0x00
0x785	PWM_CC1	R/W	[5:0]:1`b1 invert PWM_INV output	0x00
0x786	PWM_CC2	R/W	[5:0]: Signal frame polarity of PWM5~PWM0 1b'0-high level first	0x00

Address	Mnemonic	Type	Description	Reset Value
			1b'1-low level first	
0x788~ 0x793	reserved	-	-	-
0x794	PWM_TCMP0	R/W	[7:0] bits 7-0 of PWM0's high time or low time(if pola[0]=1)	0x00
0x795	PWM_TCMP0	R/W	[15:8] bits 15-8 of PWM0's high time or low time	0x00
0x796	PWM_TMAX0	R/W	[7:0] bits 7-0 of PWM0's cycle time	0x00
0x797	PWM_TMAX0	R/W	[15:8] bits 15-8 of PWM0's cycle time	0x00
0x798	PWM_TCMP1	R/W	[7:0] bits 7-0 of PWM1's high time or low time(if pola[1]=1)	0x00
0x799	PWM_TCMP1	R/W	[15:8] bits 15-8 of PWM1's high time or low time	0x00
0x79a	PWM_TMAX1	R/W	[7:0] bits 7-0 of PWM1's cycle time	0x00
0x79b	PWM_TMAX1	R/W	[15:8] bits 15-8 of PWM1's cycle time	0x00
0x79c	PWM_TCMP2	R/W	[7:0] bits 7-0 of PWM2's high time or low time(if pola[2]=1)	0x00
0x79d	PWM_TCMP2	R/W	[15:8] bits 15-8 of PWM2's high time or low time	0x00
0x79e	PWM_TMAX2	R/W	[7:0] bits 7-0 of PWM2's cycle time	0x00
0x79f	PWM_TMAX2	R/W	[15:8] bits 15-8 of PWM2's cycle time	0x00
0x7a0	PWM_TCMP3	R/W	[7:0] bits 7-0 of PWM3's high time or low time(if pola[3]=1)	0x00
0x7a1	PWM_TCMP3	R/W	[15:8] bits 15-8 of PWM3's high time or low time	0x00
0x7a2	PWM_TMAX3	R/W	[7:0] bits 7-0 of PWM3's cycle time	0x00

Address	Mnemonic	Type	Description	Reset Value
0x7a3	PWM_TMAX3	R/W	[15:8] bits 15-8 of PWM3's cycle time	0x00
0x7a4	PWM_TCMP4	R/W	[7:0] bits 7-0 of PWM4's high time or low time(if pola[4]=1)	0x00
0x7a5	PWM_TCMP4	R/W	[15:8] bits 15-8 of PWM4's high time or low time	0x00
0x7a6	PWM_TMAX4	R/W	[7:0] bits 7-0 of PWM4's cycle time	0x00
0x7a7	PWM_TMAX4	R/W	[15:8] bits 15-8 of PWM4's cycle time	0x00
0x7a8	PWM_TCMP5	R/W	[7:0] bits 7-0 of PWM5's high time or low time(if pola[5]=1)	0x00
0x7a9	PWM_TCMP5	R/W	[15:8] bits 15-8 of PWM5's high time or low time	0x00
0x7aa	PWM_TMAX5	R/W	[7:0] bits 7-0 of PWM5's cycle time	0x00
0x7ab	PWM_TMAX5	R/W	[15:8] bits 15-8 of PWM5's cycle time	0x00
0x7ac	PWM_PNUM0	R/W	[7:0] bits 7-0 of PWM0 Pulse number in count mode and IR mode	0x00
0x7ad	PWM_PNUM0	R/W	[13:8] bits 13-8 of PWM0 Pulse number in count mode and IR mode	0x00
0x7ae~ 0x7af	reserved	-	-	-
0x7b0	PWM_MASK0	R/W	INT mask [0] PWM0 Pnum int 0: disable 1: Enable [1] PWM0 ir dma fifo mode int 0: disable 1: Enable [2] PWM0 frame int 0: disable 1: Enable	0x00

Address	Mnemonic	Type	Description	Reset Value
			[3] PWM1 frame int 0: disable 1: Enable [4] PWM2 frame int 0: disable 1: Enable [5] PWM3 frame int 0: disable 1: Enable [6] PWM4 frame int 0: disable 1: Enable [7] PWM5 frame int 0: disable 1: Enable	
0x7b1	PWM_INT0	R/W	INT status, write 1 to clear [0]: PWM0 pnum int (have sent PNUM pulses, PWM_NCNT==PWM_PNUM) [1]:PWM0 ir dma fifo mode int(pnum int &fifo empty in ir dma fifo mode) [2]: PWM0 cycle done int (PWM_CNT==PWM_TMAX) [3]: PWM1 cycle done int (PWM_CNT==PWM_TMAX) [4]: PWM2 cycle done int (PWM_CNT==PWM_TMAX) [5]: PWM3 cycle done int (PWM_CNT==PWM_TMAX) [6]: PWM4 cycle done int (PWM_CNT==PWM_TMAX) [7]: PWM5 cycle done int (PWM_CNT==PWM_TMAX)	0x00
0x7b2	PWM_MASK1	R/W	[0]: PWM0 fifo mode fifo cnt int mask 0: disable, 1: Enable	0x00

Address	Mnemonic	Type	Description	Reset Value
0x7b3	PWM_INT1	R/W	INT status, write 1 to clear [0]: fifo mode cnt int, when FIFO_NUM (0x7cd[3:0]) is less than FIFO_NUM_LVL (0x7cc[3:0])	0x00
0x7b4	PWM_CNT0	R	[7:0]PWM0 cnt value	0x00
0x7b5	PWM_CNT0	-	[15:8]PWM0 cnt value	0x00
0x7b6	PWM_CNT1	R	[7:0]PWM1 cnt value	0x00
0x7b7	PWM_CNT1	-	[15:8]PWM1 cnt value	0x00
0x7b8	PWM_CNT2	R	[7:0]PWM2 cnt value	0x00
0x7b9	PWM_CNT2	-	[15:8]PWM2 cnt value	0x00
0x7ba	PWM_CNT3	R	[7:0]PWM3 cnt value	0x00
0x7bb	PWM_CNT3	-	[15:8]PWM3 cnt value	0x00
0x7bc	PWM_CNT4	R	[7:0]PWM4 cnt value	0x00
0x7bd	PWM_CNT4	-	[15:8]PWM4 cnt value	0x00
0x7be	PWM_CNT5	R	[7:0]PWM5 cnt value	0x00
0x7bf	PWM_CNT5	-	[15:8]PWM5 cnt value	0x00
0x7c0	PWM_NCNT0	R	[7:0]PWM0 pluse_cnt value	0x00
0x7c1	PWM_NCNT0	-	[15:8]PWM0 pluse_cnt value	0x00
0x7c2 ~ 0x7c3	reserved	-	-	-
0x7c4	PWM_TCMP0_SHADOW	R/W	[7:0] bits 7-0 of PWM0's high time or low time(if pola[0]=1),if shadow bit(fifo data[14]) is 1'b1 in ir fifo mode or dma fifo mode	0x55
0x7c5	PWM_TCMP0_SHADOW	R/W	[15:8] bits 15-8 of PWM0's high time or low time ,if shadow bit(fifo data[14]) is 1'b1 in ir	0x55

Address	Mnemonic	Type	Description	Reset Value
			fifo mode or dma fifo mode	
0x7c6	PWM_TMAX0_SHADOW	R/W	[7:0] bits 7-0 of PWM0's cycle time, if shadow bit(fifo data[14]) is 1'b1 in ir fifo mode or dma fifo mode	0x00
0x7c7	PWM_TMAX0_SHADOW	R/W	[15:8] bits 15-8 of PWM0's cycle time, if shadow bit(fifo frame[14]) is 1'b1 in ir fifo mode or dma fifo mode	0x00
0x7c8	FIFO_DAT0_ENTRY	R/W	Use in ir fifo mode	0x00
0x7c9	FIFO_DAT1_ENTRY	R/W	Use in ir fifo mode	0x00
0x7ca	FIFO_DAT2_ENTRY	R/W	Use in ir fifo mode	0x00
0x7cb	FIFO_DAT3_ENTRY	R/W	Use in ir fifo mode	0x00
0x7cc	FIFO_NUM_LVL	R/W	FIFO num int trigger level	0x00
0x7cd	FIFO_SR	R	[3:0]:FIFO DATA NUM(byte) [4]:FIFO EMPTY [5]:FIFO FULL	0x10
0x7ce	FIFO_CLR	W1	[0]: write 1 to clear data in FIFO	0x00

8.2 Enable PWM

Register PWM_EN (address 0x780)[5:1] and PWM_EN0 (address 0x781)[0] serves to enable PWM5~PWM0 respectively via writing "1" for the corresponding bits.

8.3 Set PWM Clock

PWM clock derives from system clock. Register PWM_CLKDIV (address 0x782) serves to set the frequency dividing factor for PWM clock. Formula below applies:

$$F_{PWM} = F_{System\ clock} / (PWM\_CLKDIV+1)$$

8.4 PWM Waveform, Polarity and Output Inversion

Each PWM channel has independent counter and 2 status including "Count" and "Remaining". Count and Remaining status form a signal frame.

8.4.1 Waveform of Signal Frame

When PWM#n is enabled, first PWM#n enters Count status and outputs High level signal by default. When PWM#n counter reaches cycles set in register PWM_TCMP#n (address 0x794~0x795, 0x798~0x799, 0x79c~0x79d, 0x7a0~0x7a1, 0x7a4~0x7a5, 0x7a8~0x7a9) / PWM_TCMP0_SHADOW (0x7c4~0x7c5), PWM#n enters Remaining status and outputs Low level till PWM#n cycle time configured in register PWM_TMAX#n (address 0x796~0x797, 0x79a~0x79b, 0x79e~0x79f, 0x7a2~0x7a3, 0x7a6~0x7a7, 0x7aa~0x7ab) / PWM_TMAX0_SHADOW (0x7c6~0x7c7) expires.

Figure 8-1 A Signal Frame
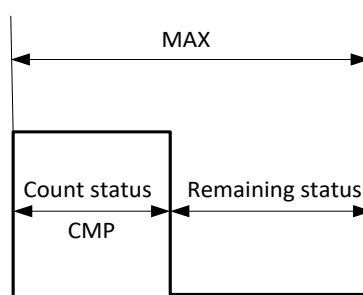
An interruption will be generated at the end of each signal frame if enabled via register PWM_MASK (address 0x7b0[2:7]).

8.4.2 Invert PWM Output

PWM#n and PWM#n_N output could be inverted independently via register PWM_CC0 (address 0x784) and PWM_CC1 (address 0x785). When the inversion bit is enabled, waveform of the corresponding PWM channel will be inverted completely.

8.4.3 Polarity for Signal Frame

By default, PWM#n outputs High level at Count status and Low level at Remaining status. When the corresponding polarity bit is enabled via register PWM_CC2 (address 0x786[5:0]), PWM#n will output Low level at Count status and High level at Remaining status.

Figure 8-2 PWM Output Waveform Chart
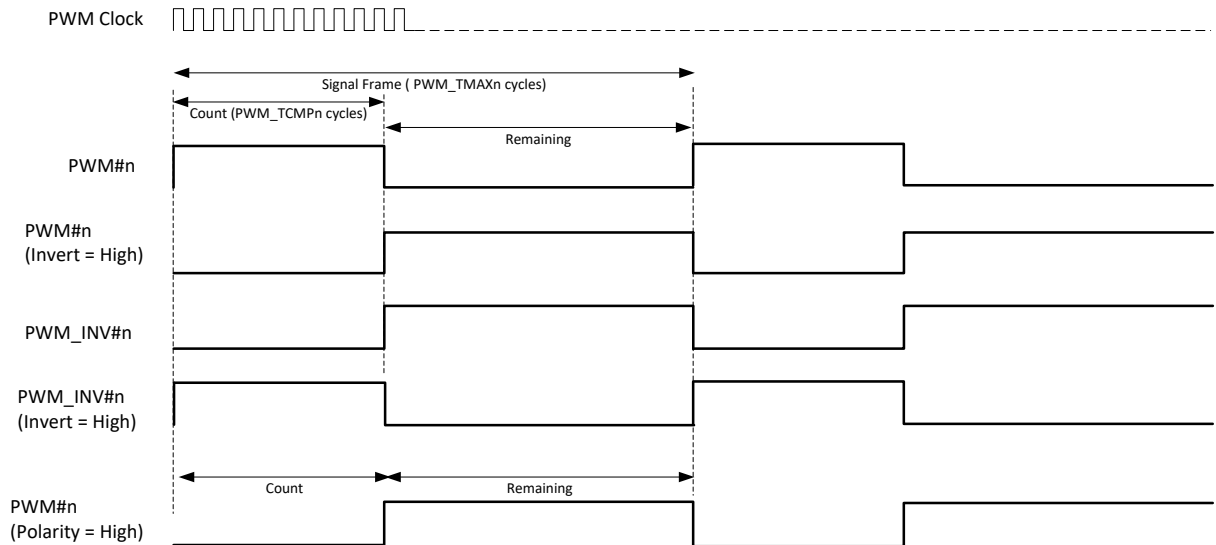
8.5 PWM Mode

8.5.1 Select PWM Mode

PWM0 supports five modes, including Continuous mode (normal mode, default), Counting mode, IR mode, IR FIFO mode, IR DMA FIFO mode.

PWM1~PWM5 only support Continuous mode.

Register PWM_MODE (address 0x783) serves to select PWM0 mode.

8.5.2 Continuous Mode

PWM0~PWM5 all support Continuous mode. In this mode, PWM#n continuously sends out signal frames. PWM#n should be disabled via address 0x780/0x781 to stop it; when stopped, the PWM output will turn low immediately.

During Continuous mode, waveform could be changed freely via PWM_TCMP#n and PWM_TMAX#n. New configuration for PWM_TCMP#n and PWM_TMAX#n will take effect in the next signal frame.

After each signal frame is finished, corresponding PWM cycle done interrupt flag bit (0x7b1[2:7]) will be automatically set to 1b'1. If the interrupt is enabled by setting PWM_MASK0 (address 0x7b0[2:7]) as 1b'1, a frame interruption will be generated. User needs to write 1b'1 to the flag bit to manually clear it.

Figure 8-3 Continuous Mode
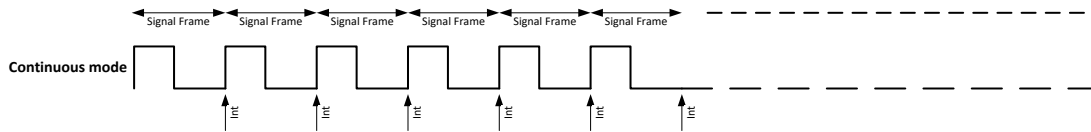
8.5.3 Counting Mode

Only PWM0 supports Counting mode. Address 0x783[3:0] should be set as 4b'0001 to select PWM0 counting mode.

In this mode, PWM0 sends out specified number of signal frames which is defined as a pulse group. The number is configured via register PWM_PNUM0 (address 0x7ac~0x7ad).

After each signal frame is finished, PWM0 cycle done interrupt flag bit (0x7b1[2]) will be automatically set to 1b'1. If the interrupt is enabled by setting PWM_MASK0 (address 0x7b0[2]) as 1b'1, a frame interruption will be generated. User needs to write 1b'1 to the flag bit to manually clear it.

After a pulse group is finished, PWM0 will be disabled automatically, and PWM0 pnum interrupt flag bit (0x7b1[0]) will be automatically set to 1b'1. If the interrupt is enabled by setting PWM_MASK0 (address 0x7b0[0]) as 1b'1, a Pnum interruption will be generated. User needs to write 1b'1 to the flag bit to manually clear it.

Figure 8-4 Counting Mode (n=0)
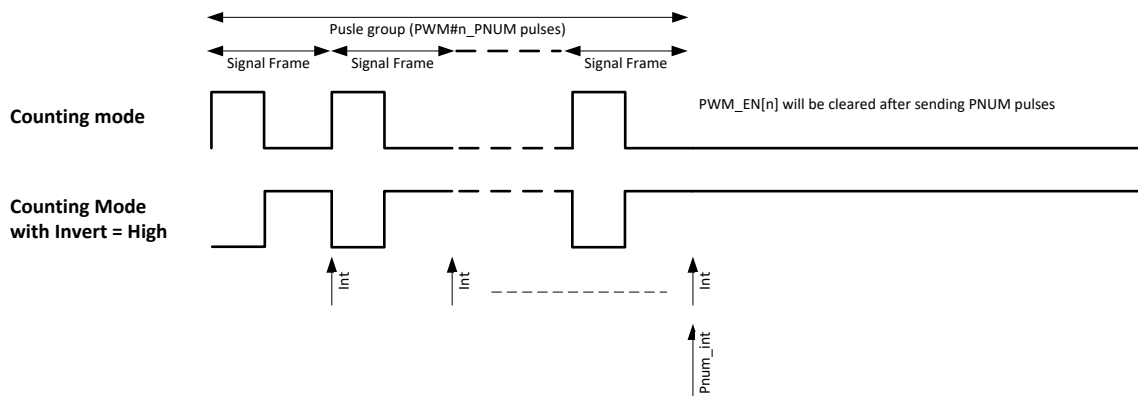
Counting mode also serves to stop IR mode gracefully. Refer to Section 8.5.4 for details.

8.5.4 IR Mode

Only PWM0 supports IR mode. Address 0x783[3:0] should be set as 4b'0011 to select PWM0 IR mode.

In this mode, specified number of frames is defined as one pulse group. In contrast to Counting mode where PWM0 stops after first pulse group is finished, PWM0 will constantly send pulse groups in IR mode.

During IR mode, PWM0 output waveform could also be changed freely via WM_TCMP0, PWM_TMAX0 and PWM_PNUM0. New configuration for PWM_TCMP0, PWM_TMAX0 and PWM_PNUM0 will take effect in the next pulse group.

To stop IR mode and complete current pulse group, user can switch PWM0 from IR mode to Counting mode so that PWM0 will stop after current pulse group is finished. If PWM0 is disabled directly via PWM_EN0 (0x781[0]), PWM0 output will turn Low immediately despite of current pulse group.

After each signal frame/pulse group is finished, PWM0 cycle done interrupt flag bit (0x7b1[2])/PWM0 pnum interrupt flag bit (0x7b1[0]) will be automatically set to 1b'1. A frame interruption/Pnum interruption will be generated (if enabled by setting address 0x7b0[2]/0x7b0[0] as 1b'1).

Figure 8-5 IR Mode (n=0)
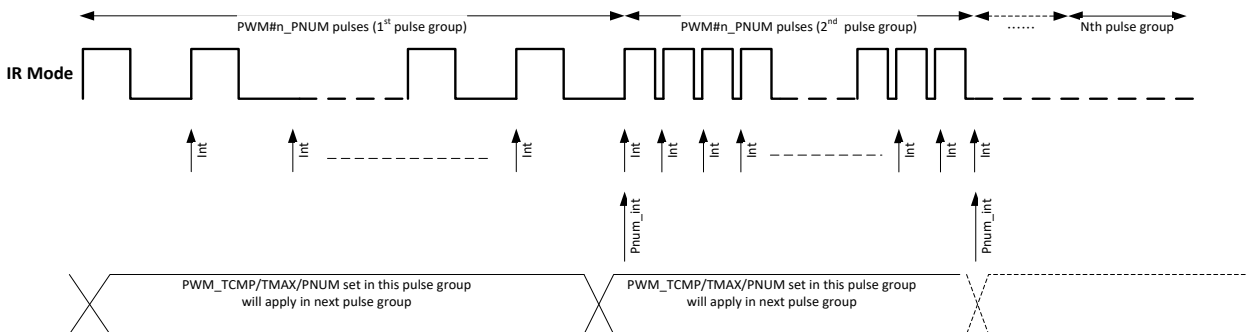
8.5.5 IR FIFO Mode

IR FIFO mode is designed to allow IR transmission of long code patterns without the continued intervention of MCU, and it is designed as a selectable working mode on PWM0. The IR carrier frequency is divided down from the system clock and can be configured as any normal IR frequencies, e.g. 36 kHz, 38 kHz, 40 kHz, or 56 kHz.

Only PWM0 supports IR FIFO mode. Address 0x783[3:0] should be set as 4b'0111 to select PWM0 IR FIFO mode.

An element ("FIFO CFG Data") is defined as basic unit of IR waveform, and written into FIFO. This element consists of 16 bits, including:

- bit[13:0] defines PWM pulse number of current group.
- bit[14] determines duty cycle and period for current PWM pulse group.

  0: use configuration of TCMP0 and TMAX0 in 0x794~0x797;

  1: use configuration of TCMP0_SHADOW and TMAX0_SHADOW in 0x7c4~0x7c7.
- bit[15] determines whether current PWM pulse group is used as carrier, i.e. whether PWM will output pulse (1) or low level (0).

User should use FIFO_DATA_ENTRY in 0x7c8~0x7cb to write the 16-bit "FIFO CFG Data" into FIFO by byte or half word or word.

- To write by byte, user should successively write 0x7c8, 0x7c9, 0x7ca and 0x7cb.
- To write by half word, user should successively write 0x7c8 and 0x7ca.
- To write by word, user should write 0x7c8.

FIFO depth is 8 bytes. User can read the register FIFO_SR in 0x7cd to view FIFO empty/full status and check FIFO data number.

Figure 8-6 IR Format Examples
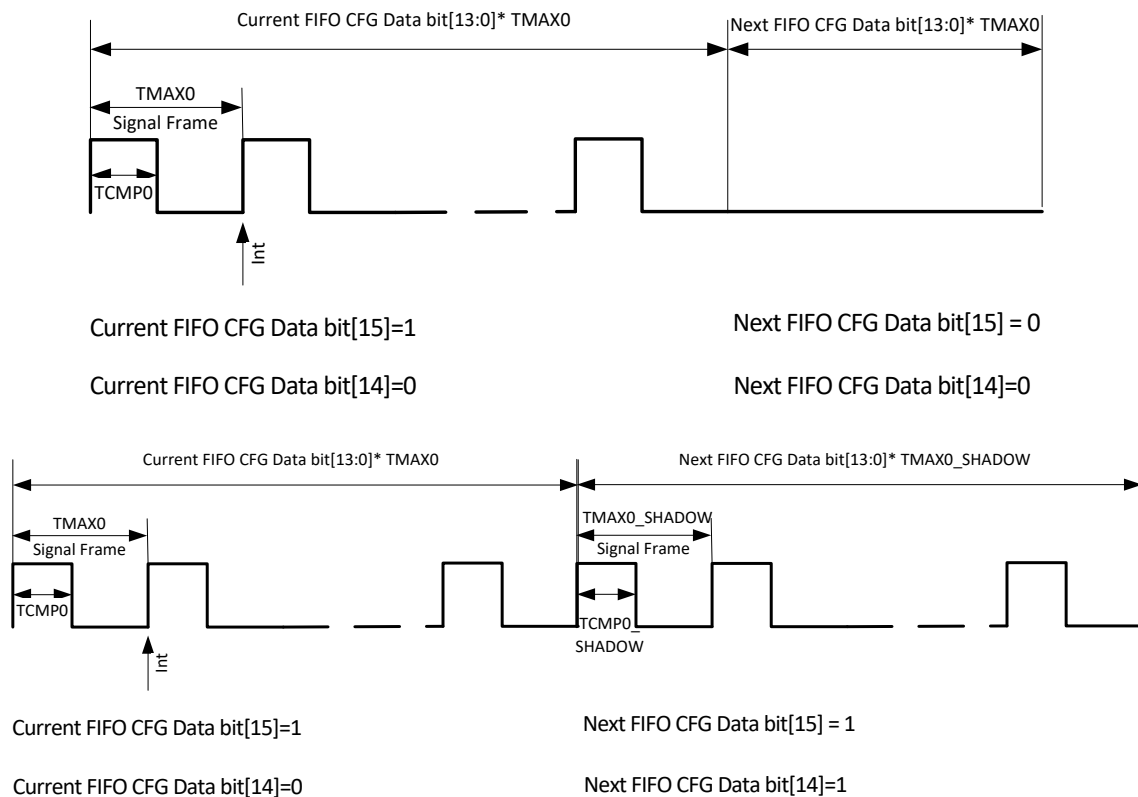
When "FIFO CFG Data" is configured in FIFO and PWM0 is enabled via PWM_EN0 (address 0x781[0]), the configured waveforms will be output from PWM0 in sequence. As long as FIFO doesn't overflow, user can continue to add waveforms during IR waveforms sending process, and long IR code that exceeds the FIFO

depth can be implemented this way. After all waveforms are sent, FIFO becomes empty, PWM0 will be disabled automatically.

The FIFO_CLR register (address 0x7ce[0]) serves to clear data in FIFO. Writing 1b'1 to this register will clear all data in the FIFO. Note that the FIFO can only be cleared when not in active transmission.

8.5.6 IR DMA FIFO Mode

IR DMA FIFO mode is designed to allow IR transmission of long code patterns without occupation of MCU, and it is designed as a selectable working mode on PWM0. The IR carrier frequency is divided down from the system clock and can be configured as any normal IR frequencies, e.g. 36 kHz, 38 kHz, 40 kHz, or 56 kHz.

Only PWM0 supports IR DMA FIFO mode. Address 0x783[3:0] should be set as 4b'1111 to select PWM0 IR DMA FIFO mode.

This mode is similar to IR FIFO mode, except that "FIFO CFG Data" is written into FIFO by DMA instead of MCU. User should write the configuration of "FIFO CFG Data" into RAM, and then enable DMA channel 5. DMA will automatically write the configuration into FIFO.

NOTE: *In this mode, when DMA channel 5 is enabled, PWM will automatically output configured waveform, without the need to manually enable PWM0 via 0x781[0] (i.e. 0x781[0] will be set as 1b'1 automatically).*

Example 1:

Suppose Mark carrier (pulse) frequency1(F1) = 40 kHz, duty cycle 1/3

　　　　Mark carrier (pulse) frequency2(F2) = 50 kHz, duty cycle 1/2
　　　　Space carrier (low level) frequency(F3) = 40 kHz

If user wants to make PWM send waveforms in following format (PWM CLK = 24 MHz):

Burst(20[F1]), i.e. 20 F1 pulses

Burst(30[F2]),

Burst(50[F1]) ,

Burst(50[F2]),

Burst(20[F1],10[F3]),

Burst(30[F2],10[F3])

Step 1   Set carrier F1 frequency as 40 kHz, set duty cycle as 1/3.

Set **PWM_TMAX0** as 0x258 (i.e. 24MHz/40kHz=600=0x258).

Since duty cycle is 1/3, set **PWM_TCMP0** as 0xc8 (i.e. 600/3=200=0xc8).

Set carrier F2 frequency as 50kHz, set duty cycle as 1/2.

Set **PWM_TMAX0_SHADOW** as 0x1e0 (i.e. 24MHz/50kHz=480=0x1e0).

Since duty cycle is 1/2, set **PWM_TCMP0_SHADOW** as 0xf0 (i.e. 480/2=240=0xf0).

Step 2   Generate "FIFO CFG Data" sequence.

Burst(20[F1]): {[15]: 1'b1, [14]: 1'b0, [13:0]: 'd20}=0x8014.

Burst(30[F2]): {[15]: 1'b1, [14]: 1'b1, [13:0]: 'd30}=0xc01e.

Burst(50[F1]) : {[15]: 1'b1, [14]: 1'b0, [13:0]: 'd50}=0x8032.

Burst(50[F2]): {[15]: 1'b1, [14]: 1'b1, [13:0]:'d50}=0xc032.

Burst(20[F1],10[F3]): {[15]: 1'b1, [14]: 1'b0, [13:0]: 'd20}=0x8014,

{[15]: 1'b0, [14]: 1'b0, [13:0]: 'd10}=0x000a.

Burst(30[F2],10[F3]): {[15]: 1'b1, [14]: 1'b1, [13:0]: 'd30}=0xc01e,

{[15]:1'b0, [14]: 1'b0, [13:0]: 'd10}=0x000a.

Step 3   Write "FIFO CFG Data" into SRAM in DMA format.

DMA SOURCE ADDRESS+0x00: 0x0000_0010 (dma transfer-length: 16byte)

DMA SOURCE ADDRESS+0x04: 0xc01e_8014 (LITTLE ENDIAN)

DMA SOURCE ADDRESS+0x08: 0xc032_8032

DMA SOURCE ADDRESS+0x0c: 0x000a_8014

DMA SOURCE ADDRESS+0x10: 0x000a_c01e

Step 4   Enable DMA channel 5 to send PWM waveforms.

Write 1'b1 to address 0x524[5] to enable DMA channel 5.

After all waveforms are sent, FIFO becomes empty, PWM0 will be disabled automatically (address 0x781[0] is automatically cleared). The FIFO mode stop interrupt flag bit (address 0x7b3[0]) will be automatically set as 1b'1. If the interrupt is enabled by setting PWM_MASK1 (address 0x7b2[0]) as 1b'1, a FIFO mode stop interrupt will be generated. User needs to write 1b'1 to the flag bit to manually clear it.

Example 2:

Suppose carrier frequency is 38 kHz, system clock frequency is 24 MHz, duty cycle is 1/3, and the format of IR code to be sent is shown as below:

1.   Preamble waveform: 9 ms carrier + 4.5 ms low level.

2.   Data 1 waveform: 0.56 ms carrier + 0.56 ms low level.

3.   Data 0 waveform: 0.56 ms carrier + 1.69 ms low level.

4. Repeat waveform: 9 ms carrier + 2.25 ms low level + 0.56 ms carrier. Repeat waveform duration is 11.81 ms, interval between two adjacent repeat waveforms is 108 ms.

5. End waveform: 0.56 ms carrier.

User can follow the steps below to configure related registers:

Step 1   Set carrier frequency as 38 kHz, set duty cycle as 1/3.

Set **PWM_TMAX0** as 0x277 (i.e. 24 MHz/38 kHz=631=0x277).

Since duty cycle is 1/3, set **PWM_TCMP0** as 0xd2 (i.e. 631/3=210=0xd2).

Step 2   Generate "FIFO CFG Data" sequence.

Preamble waveform:

9 ms carrier: {[15]:1'b1, [14]:1'b0, [13:0]: 9*38='d 342=14'h 156}=0x8156

4.5 ms low level: {[15]:1'b0, [14]:1'b0, [13:0]: 4.5*38='d 171=14'h ab}=0x00ab

Data 1 waveform:

0.56 ms carrier: {[15]:1'b1, [14]:1'b0, [13:0]: 0.56*38='d 21=14'h 15}=0x8015

0.56 ms low level: {[15]:1'b0, [14]:1'b0, [13:0]: 0.56*38='d 21=14'h 15}=0x0015

Data 0 waveform:

0.56 ms carrier: {[15]:1'b1, [14]:1'b0, [13:0]: 0.56*38='d 21=14'h 15}=0x8015

1.69 ms low level: {[15]:1'b0, [14]:1'b0, [13:0]: 1.69*38='d 64=14'h 40}=0x0040

Repeat waveform:

9 ms carrier: {[15]:1'b1, [14]:1'b0, [13:0]: 9*38='d 342=14'h 156}=0x8156

2.25 ms low level: {[15]:1'b0, [14]:1'b0, [13:0]: 2.25*38='d 86=14'h 56}=0x0056

0.56 ms carrier: {[15]:1'b1, [14]:1'b0, [13:0]: 0.56*38='d 21=14'h 15}=0x8015

108 ms -11.81 ms =96.19 ms low level:

{[15]:1'b0, [14]:1'b0, [13:0]: 96.19*38='d 3655=14'h e47}=0x0e47

End waveform:

0.56 ms carrier: {[15]:1'b1, [14]:1'b0, [13:0]: 0.56*38='d 21=14'h 15}=0x8015

Step 3   Write "IR CFG Data" into SRAM in DMA format.

If user want PWM0 to send IR waveform in following format:

Preamble+0x5a+Repeat+End

Preamble: 0x8156, 0x00ab

0x5a=8'b01011010

Data 0: 0x8015, 0x0040

Data 1: 0x8015, 0x0015

Data 0: 0x8015, 0x0040

Data 1: 0x8015, 0x0015

Data 1: 0x8015, 0x0015

Data 0: 0x8015, 0x0040

Data 1: 0x8015, 0x0015

Data 0: 0x8015, 0x0040

Repeat: 0x8156, 0x0056, 0x8015, 0x0e47

End: 0x8015.

User needs to write the configuration information above into source address of DMA channel 5, as shown below:

DMA SOURCE ADDRESS+0x00: 0x0000_002e (dma transfer-length: 46byte)

DMA SOURCE ADDRESS+0x04: 0x00ab_8156 (Preamble) (LITTLE ENDIAN)

DMA SOURCE ADDRESS+0x08: 0x0040_8015 (Data 0)

DMA SOURCE ADDRESS+0x0c: 0x0015_8015 (Data 1)

DMA SOURCE ADDRESS+0x10: 0x0040_8015 (Data 0)

DMA SOURCE ADDRESS+0x14: 0x0015_8015 (Data 1)

DMA SOURCE ADDRESS+0x18: 0x0015_8015 (Data 1)

DMA SOURCE ADDRESS+0x1c: 0x0040_8015 (Data 0)

DMA SOURCE ADDRESS+0x20: 0x0015_8015 (Data 1)

DMA SOURCE ADDRESS+0x24: 0x0040_8015 (Data 0)

DMA SOURCE ADDRESS+0x28: 0x0056_8156 (Repeat)

DMA SOURCE ADDRESS+0x2c: 0x0e47_8015 (Repeat)

DMA SOURCE ADDRESS+0x30: 0x8015 (End)

Step 4   Enable DMA channel 5 to send PWM waveforms.

Write 1'b1 to address 0x524[5] to enable DMA channel 5.

After all waveforms are sent, FIFO becomes empty, PWM0 will be disabled automatically (address 0x781[0] is automatically cleared). The FIFO mode stop interrupt flag bit (address 0x7b3[0]) will be automatically set as 1b'1. If the interrupt is enabled by setting PWM_MASK1 (address 0x7b2[0]) as 1b'1, a FIFO mode stop interrupt will be generated. User needs to write 1b'1 to the flag bit to manually clear it.

8.6 PWM Interrupt

There are 9 interrupt sources from PWM function.

After each signal frame, PWM#n (n=0~5) will generate a frame-done IRQ (Interrupt Request) signal.

In Counting mode and IR mode, PWM0 will generate a Pnum IRQ signal after completing a pulse group.

In IR FIFO mode, PWM0 will generate a FIFO mode count IRQ signal when the FIFO_NUM value is less than the FIFO_NUM_LVL, and will generate a FIFO mode stop IRQ signal after FIFO becomes empty.

In IR DMA FIFO mode, PWM0 will generate an IR waveform send done IRQ signal, after DMA has sent all configuration data, FIFO becomes empty and final waveform is sent.

To enable PWM interrupt, the total enabling bit "irq_pwm" (address 0x641[6], see Section 6 Interrupt System) should be set as 1b'1. To enable various PWM interrupt sources, PWM_MASK0 (address 0x7b0[7:0]) and PWM_MASK1 (address 0x7b2[0]) should be set as 1b'1 correspondingly.

Interrupt status can be cleared via register PWM_INT0 (address 0x7b1[7:0]) and PWM_INT1 (address 0x7b3[0]).

9. Quadrature Decoder

The TLSR8359 embeds one quadrature decoder (QDEC) which is designed mainly for applications such as wheel. The QDEC implements debounce function to filter out jitter on the two phase inputs, and generates smooth square waves for the two phase.

9.1 Input Pin Selection

The QDEC supports two phase input; each input is selectable from the 5 pins of PortD, PortC, and PortB via setting address 0xd2[2:0] (for channel a)/0xd3[2:0] (for channel b).

Table 9-1 Input Pin Selection

Address 0xd2[2:0]/0xd3[2:0]	Pin
0	PA<2>
1	PA<3>
2	PB<6>
3	PB<7>
4	PC<2>
5	PC<3>
6	PD<6>
7	PD<7>

NOTE: *To use corresponding IO as QDEC input pin, it's needed first to enable GPIO function, enable "IE" (1) and disable "OEN" (1) for this IO.*

9.2 Common Mode and Double Accuracy Mode

The QDEC embeds an internal hardware counter, which is not connected with bus.

Address 0xd7[0] serves to select common mode or double accuracy mode.

For each wheel rolling step, two pulse edges (rising edge or falling edge) are generated.

If address 0xd7[0] is cleared to select common mode, the QDEC Counter value (real time counting value) is increased/decreased by 1 only when the same rising/falling edges are detected from the two phase signals.

Figure 9-1 Common Mode
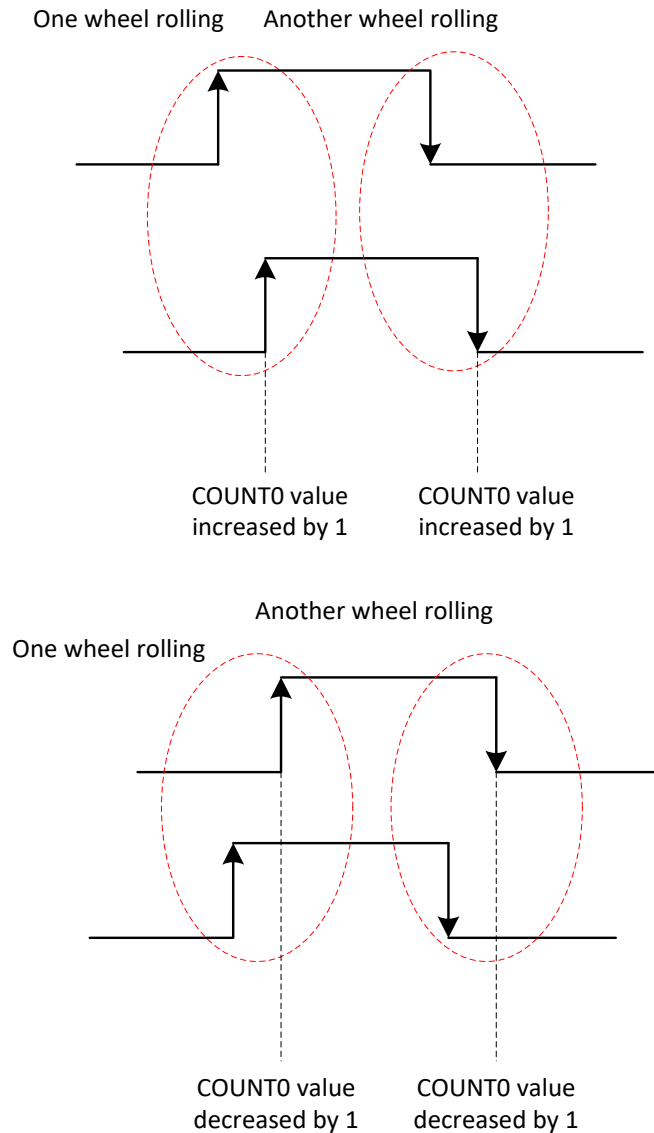
If address 0xd7[0] is set to 1b'1 to select double accuracy mode, the QDEC Counter value (real time counting value) is increased/decreased by 1 on each rising/falling edge of the two phase signals; the COUNT0 will be increased/decreased by 2 for one wheel rolling.

Figure 9-2 Double Accuracy Mode

One wheel rolling   Another wheel rolling

COUNT0 value
increased by 1

COUNT0 value
increased by 1

COUNT0 value
increased by 1

COUNT0 value
increased by 1

Another wheel rolling

One wheel rolling

COUNT0 value
decreased by 1

COUNT0 value
decreased by 1

COUNT0 value
decreased by 1

COUNT0 value
decreased by 1
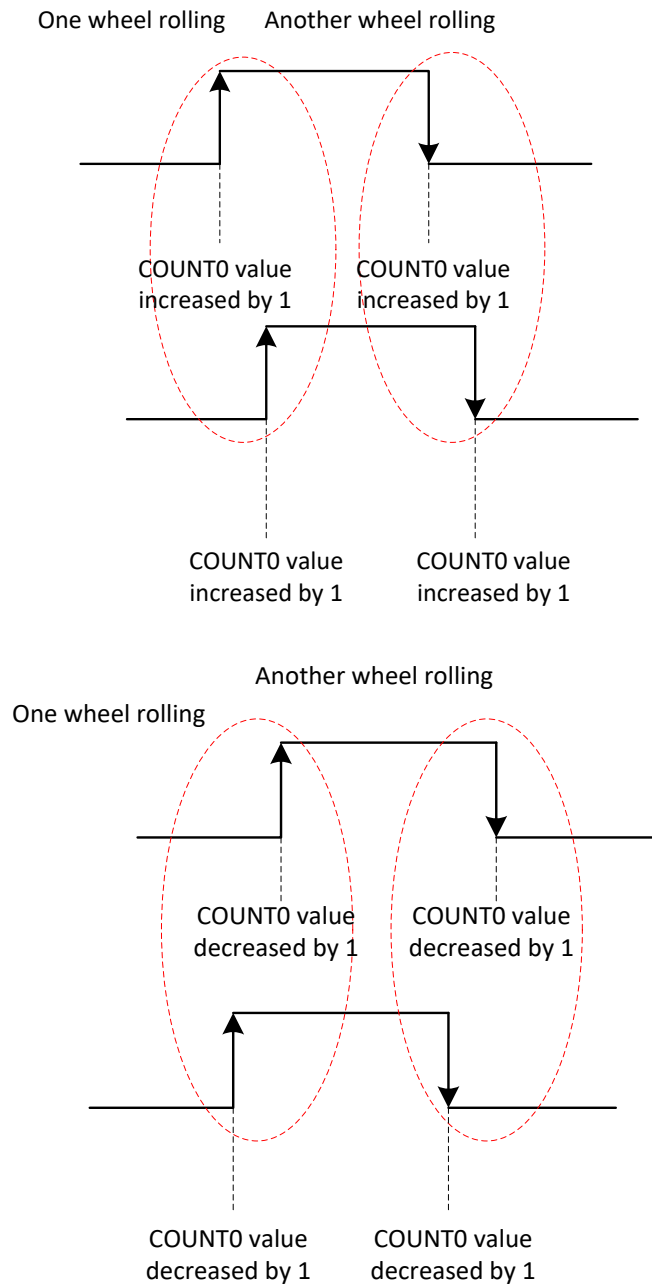
9.3 Read Real Time Counting Value

Neither can Hardware Counter value be read directly via software, nor can the counting value in address 0xd0 be updated automatically.

To read real time counting value, first write address 0xd8[0] with 1b'1 to load Hardware Counter data into the QDEC_COUNT register, then read address 0xd0.

Figure 9-3 Read Real Time Counting Value
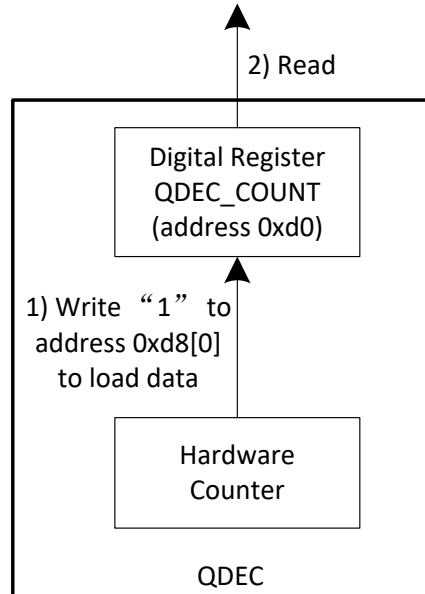
9.4 QDEC Reset

Address 0x60[5] serves to reset the QDEC. The QDEC Counter value is cleared to zero.

9.5 Other Configuration

The QDEC supports hardware debouncing. Address 0xd1[2:0] serves to set filtering window duration. All jitter with period less than the value will be filtered out and thus does not trigger count change.

Address 0xd1[4] serves to set input signal initial polarity.

Address 0xd1[5] serves to enable shuttle mode. Shuttle mode allows non-overlapping two phase signals as shown in the following figure.

Figure 9-4 Shuttle Mode
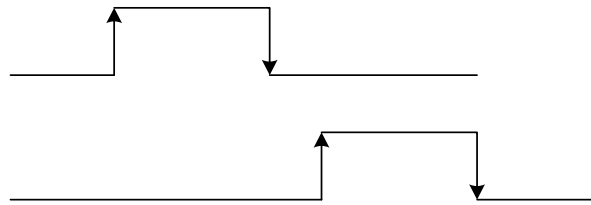
9.6 Timing Sequence

Figure 9-5 Timing Sequence Chart
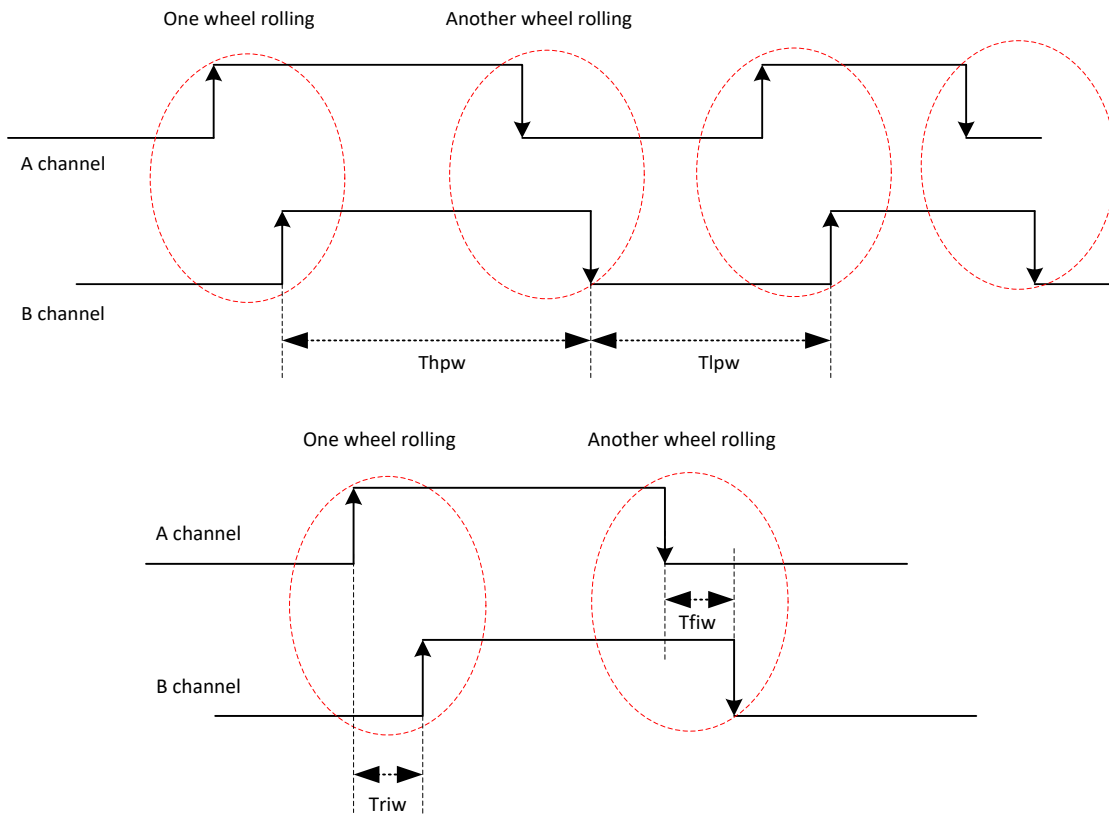
Table 9-2 Timing

Time Interval	Min Value
Thpw (High-level pulse width)	$2^{(n+1)} *clk\_32kHz *3$ (n=0xd1[2:0])

Time Interval	Min Value
Tlpw (Low-level pulse width)	$2^{(n+1)} *clk\_32kHz *3$ (n=0xd1[2:0])
Triw (Interval width between two rising edges)	$2^{(n+1)} *clk\_32kHz$ (n=0xd1[2:0])
Tfiw (Interval width between two falling edges)	$2^{(n+1)} *clk\_32kHz$ (n=0xd1[2:0])

QDEC module works based on 32 kHz clock to ensure it can work in suspend mode. QDEC module supports debouncing function, and any signal with width lower than the threshold (i.e. "$2^{(n+1)} *clk\_32kHz *3$ (n=0xd1[2:0])") will be regarded as jitter. Therefore, effective signals input from Channel A and B should contain high/low level with width Thpw/Tlpw more than the threshold. The $2^n *clk\_32kHz$ clock is used to synchronize input signal of QDEC module, so the interval between two adjacent rising/falling edges from Channel A and B, which are marked as Triw and Tfiw, should exceed "$2^{(n+1)} *clk\_32kHz$".

Only when the timing requirements above are met, can QDEC module recognize wheel rolling times correctly.

9.7 Register Table

Table 9-3 Register Table for QDEC

Address	Mnemonic	Type	Description	Reset value
0xd0	QDEC_COUNT	R	QDEC Counting value (read to clear): Pulse edge number	0x00
0xd1	QDEC_CC	R/W	[2:0] : filter time (can filter $2^n *clk\_32k*2$ width deglitch) [4]: pola, input signal pola 0: no signal is low, 1: no signal is high [5]:shuttle mode 1 to enable shuttle mode	0x00
0xd2	QDEC_CHNA	R/W	[2:0] QDEC input pin select for channel a	0x00

Address	Mnemonic	Type	Description	Reset value
			choose 1 of 5 pins for input channel a 7~0: {pd[7], rsvd (pd[6]), pc[3:2], pb[7:6], rsvd (pa[3:2])}	
0xd3	QDEC_CHNB	R/W	[2:0] QDEC input pin select for channel b choose 1 of 5 pins for input channel b 7~0: {pd[7], rsvd (pd[6]), pc[3:2], pb[7:6], rsvd (pa[3:2])}	0x01
0xd6	QDEC_RST	R/W	[0]RSVD	0x00
0xd7	QDEC_DOUBLE	R/W	[0]Enable double accuracy mode	0x01
0xd8	DATA_LOAD	R/W	[0]write 1 to load data when load completes it will be 0	0x00

10. SAR ADC

The TLSR8359 integrates one SAR ADC module, which can be used to sample analog input signals such as battery voltage and temperature sensor.

Figure 10-1 Block Diagram of ADC
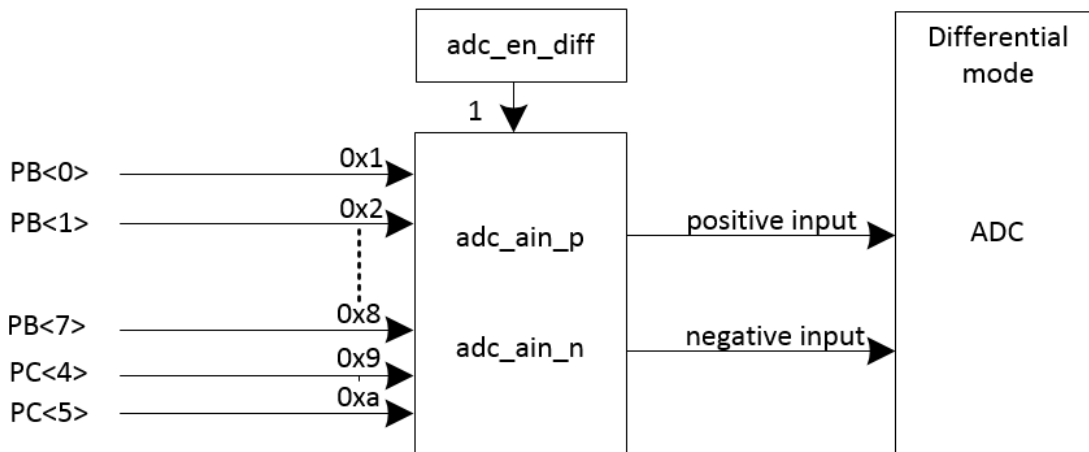
10.1 Power On/Down

The SAR ADC is disabled by default. To power on the ADC, the analog register adc_pd (afe_0xfc<5>) should be set as 1b'0.

10.2 ADC Clock

ADC clock is derived from external 24 MHz crystal source, with frequency dividing factor configurable via the analog register adc_clk_div (afe_0xf4<2:0>).

ADC clock frequency (marked as F_{ADC_clk}) = 24 MHz/(adc_clk_div+1)

10.3 ADC Control in Auto Mode

10.3.1 Set Max State and Enable Channel

The SAR ADC supports up to three channels including left channel, right channel and Misc channel. The left, right and Misc channels all consist of one "Set" state and one "Capture" state.

- The analog register r_max_scnt (afe_0xf2<6:4>) serves to set the max state index. As shown in the example below, the r_max_scnt should be set as 0x06.

1	2	3	4	5	6
Set	Capture	Set	Capture	Set	Capture
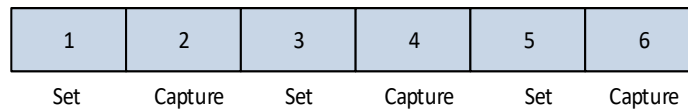
- The left/Misc channel can be enabled independently via r_en_left (afe_0xf2<0>), r_en_misc (afe_0xf2<2>).
- Only when the left channel is enabled, can the right channel be enabled via r_en_right (afe_0xf2<1>).

10.3.2 "Set" State

The length of "Set" state for left, right and Misc channel is configurable via the analog register r_max_s (afe_0xf1<3:0>).

$$\text{"Set" state duration (marked as } T_{sd}) = r\_max\_s / 24 \text{ MHz.}$$

Each "Set" state serves to set ADC control signals for current channel via corresponding analog registers, including:

- adc_en_diff: afe_0xec<4> (left channel), afe_0xec<5> (right channel), afe_0xec<6> (Misc channel). MUST set as 1b'1 to select differential input mode.
- adc_ain_p: afe_0xe8<7:4> (Misc channel), afe_0xe9<7:4> (left channel), afe_0xea<7:4> (right channel). Select positive input in differential mode.
- adc_ain_n: afe_0xe8<3:0> (Misc channel), afe_0xe9<3:0> (left channel), afe_0xea<3:0> (right channel). Select negative input in differential mode.
- adc_vref: afe_0xe7<1:0> (left channel), afe_0xe7<3:2> (right channel), afe_0xe7<5:4> (Misc channel). Set reference voltage V_{REF}. ADC maximum input range is determined by the ADC reference voltage.
- adc_sel_ai_scale: afe_0xfa<7:6>. Set scaling factor for ADC analog input as 1 (default), or 1/8. By setting this scaling factor, ADC maximum input range can be extended based on the V_{REF}. For example, suppose the V_{REF} is set as 1.2 V:

Since the scaling factor is 1 by default, the ADC maximum input range should be 0 ~ 1.2 V (negative input is GND) / -1.2 V ~ +1.2 V (negative input is ADC GPIO pin).

If the scaling factor is set as 1/8, in theory ADC maximum input range should change to 0 ~ 9.6 V (negative input is GND) / -9.6V ~ +9.6 V (negative input is ADC GPIO pin). But limited by input voltage of the chip's PAD, the actual range is narrower.

- adc_res: afe_0xeb<1:0> (left channel), afe_0xeb<5:4> (right channel), afe_0xec<1:0> (Misc channel). Set resolution as 8/10/12/14 bits.

ADC data is always 15-bit format no matter what the resolution is set. For example, 14 bits resolution indicates ADC data consists of 14-bit valid data and 1-bit sign extension bit.

- adc_tsamp: afe_0xed<3:0> (left channel), afe_0xed<7:4> (right channel), afe_0xee<3:0> (Misc channel). Set sampling time which determines the speed to stabilize input signals.

$$\text{Sampling time (marked as } T_{samp}) = adc\_tsamp / F_{ADC\_clk}.$$

The lower sampling cycle, the shorter ADC convert time.

- pga_boost, pga_gain: Set PGA gain in Boost stage and Gain stage. See Section 11 PGA.

10.3.3 "Capture" State

For the left, right and Misc channels, at the beginning of each "Capture" state, run signal is issued automatically to start an ADC sampling and conversion process; at the end of each "Capture" state, ADC output data is captured.

- The length of "Capture" state for Misc channel is configurable via the analog register r_max_mc[9:0] (afe_0xf1<7:6>, afe_0xef<7:0>).

  "Capture" state duration for Misc channel (marked as T_{cd}) = r_max_mc / 24 MHz.

- The length of "Capture" state for left and right channel is configurable via the analog register r_max_c[9:0] (afe_0xf1<5:4>, afe_0xf0<7:0>).

  "Capture" state duration for left & right channel (marked as T_{cd}) = r_max_c / 24MHz.

- The "VLD" bit (afe_0xf8<7>) will be set as 1b'1 at the end of "Capture" state to indicate the ADC data is valid, and this flag bit will be cleared automatically.

- The 15-bit ADC output data for Misc channel can be read from the analog register adc_dat[14:0] (afe_0xf8<6:0>, afe_0xf7<7:0>).

NOTE: The total duration "T_{td}", which is the sum of the length of "Set" state and "Capture" state for all channels available, determines the sampling rate.

10.3.4 Usage Cases

10.3.4.1 Case 1: 3-Channel Sampling for Left, Right and Misc

In this case, afe_0xf2<3:0> should be set as 0x7, so as to enable the left, right and Misc channels, the max state index should be set as "6" by setting afe_0xf2<6:4> as 0x6.

The total duration (marked as T_{td}) = (1*r_max_mc+3*r_max_s+2*r_max_c) / 24 MHz.

r_max_s 1	r_max_c 2	r_max_s 3	r_max_c 4	r_max_s 5	r_max_mc 6
Set	Capture	Set	Capture	Set	Capture

←————Left————→ ←————Right————→ ←————Misc————→
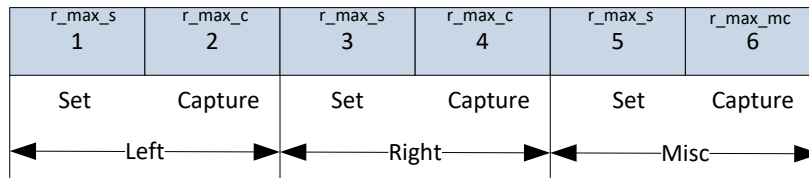
10.3.4.2 Case 2: 2-Channel Sampling for Left and Misc

In this case, afe_0xf2<3:0> should be set as 0x5, so as to enable the left and Misc channels and disable the right channel, the max state index should be set as "4" by setting afe_0xf2<6:4> as 0x4.

The total duration (marked as T_{td}) = (1*r_max_mc+2*r_max_s+1*r_max_c) / 24 MHz.

r_max_s 1	r_max_c 2	r_max_s 3	r_max_mc 4
Set	Capture	Set	Capture

←————Left————→ ←————Misc————→
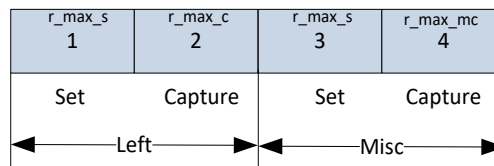
10.3.4.3 Case 3: 2-Channel Sampling for Left and Right

In this case, afe_0xf2<3:0> should be set as 0x3, so as to enable the left and right channels and disable the Misc channel, the max state index should be set as "4" by setting afe_0xf2<6:4> as 0x4.

The total duration (marked as T_{td}) = (2*r_max_s+2*r_max_c) / 24 MHz.

r_max_s 1	r_max_c 2	r_max_s 3	r_max_c 4
Set	Capture	Set	Capture

← Left → ← Right →
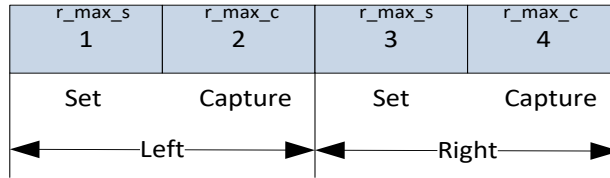
10.3.4.4 Case 4: 1-Channel Sampling for Left

In this case, afe_0xf2<3:0> should be set as 0x1, so as to enable the left channel and disable the right and Misc channels, the max state index should be set as "2" by setting afe_0xf2<6:4> as 0x2.

The total duration (marked as T_{td}) = (1*r_max_s+1*r_max_c) / 24 MHz.

r_max_s 1	r_max_c 2
Set	Capture
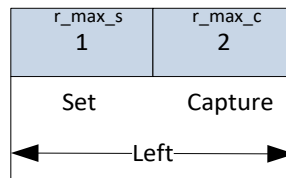
← Left →

10.3.4.5 Case 5: 1-Channel Sampling for Misc

In this case, afe_0xf2<3:0> should be set as 0x4, so as to enable the Misc channel and disable the left and right channels, the max state index should be set as "2" by setting afe_0xf2<6:4> as 0x2.

The total duration (marked as T_{td}) = (1*r_max_s+1*r_max_mc) / 24 MHz.

r_max_s 1	r_max_mc 2
Set	Capture
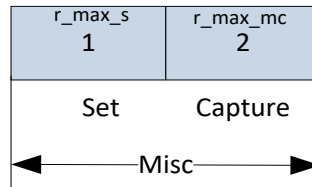
← Misc →

10.3.4.6 Case 6 with Detailed Register Setting

This case introduces the register setting details for 3-channel sampling of left, right and Misc channels.

r_max_s 1	r_max_c 2	r_max_s 3	r_max_c 4	r_max_s 5	r_max_mc 6
Set	Capture	Set	Capture	Set	Capture
Left		Right		Misc	
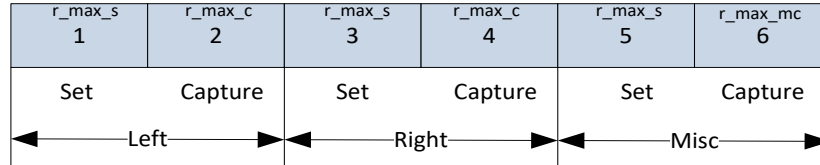
Table 10-1 Overall Register Setting

Function	Register Setting
Power on the ADC	afe_0xfc<5> = 1b'0
Set F_{ADC_clk} (ADC clock frequency) as 4 MHz	afe_0xf4<2:0> = 5 F_{ADC_clk} = 24 MHz/(5+1) = 4 MHz
Enable the left, right and Misc channels	afe_0xf2<3:0> = 0x7
Set the max state index as "6"	afe_0xf2<6:4> = 0x6

Table 10-2 Register Setting for L/R/M Channel

Function	Register setting		
	Left	**Right**	**Misc**
Set T_{sd} ("Set" state duration)	afe_0xf1<3:0> = 10 T_{sd} = r_max_s/24 MHz = 10/24 MHz = 0.417 μs		
Set T_{cd} ("Capture" state duration)	afe_0xf1<5:4>=0, afe_0xf0<7:0>=170 T_{cd} = r_max_c[9:0]/24 MHz = 170/24 MHz = 7.1 μs		afe_0xf1<7:6>=0, afe_0xef<7:0>=130 T_{cd} = r_max_mc[9:0]/24 MHz =130/24 MHz = 5.4 μs
T_{td} (total duration)	T_{td} = (1*r_max_mc+3*r_max_s+2*r_max_c) / 24 MHz = 500/24 MHz =20.83 μs		
F_s (Sampling frequency)	F_s = 1 / T_{td} = 24 MHz/500 = 48 kHz		
Select differential	afe_0xec<4>=1	afe_0xec<5>=1	afe_0xec<6>=1

Function	Register setting		
	Left	Right	Misc
input	differential input	differential input	differential input
Set input channel	afe_0xe9=0x12 Select B<0> and B<1> as positive input and negative input	afe_0xea=0x34 Select B<2> and B<3> as positive input and negative input	afe_0xe8=0xaf Select C<5> as positive input, select GND as negative input
Set reference voltage V_{REF}	afe_0xe7<1:0>=0 V_{REF} = 0.6 V	afe_0xe7<3:2>=1 V_{REF} = 0.9 V	afe_0xe7<5:4>=2 V_{REF} =1.2 V
Set scaling factor for ADC analog input	afe_0xfa<7:6>=0 scaling factor: 1		
	ADC maximum input range: -0.6 ~ +0.6 V	ADC maximum input range: -0.9 ~ +0.9 V	ADC maximum input range: 0 ~ +1.2 V
Set resolution	afe_0xeb<1:0>=1 resolution: 10 bits	afe_0xeb<5:4>=2 resolution: 12 bits	afe_0xec<1:0>=3 resolution: 14 bits
Set T_{samp} (determines the speed to stabilize input before sampling)	afe_0xed<3:0>=1 T_{samp} = adc_tsamp / F_{ADC_clk} = 6/4 MHz = 1.5 µs	afe_0xed<7:4>=2 T_{samp} = adc_tsamp / F_{ADC_clk} = 9/4 MHz = 2.25 µs	afe_0xee<3:0>=3 T_{samp} = adc_tsamp / F_{ADC_clk} = 12/4 MHz = 3 µs

10.4 Register Table

Table 10-3 Register Table Related to SAR ADC

Address	Mnemonic	Default value	Description
afe_0xe7<1:0>	adc_vrefl	00	Select V_{REF} for left channel 0x0: 0.6 V

Address	Mnemonic	Default value	Description
			0x1: 0.9 V 0x2: 1.2 V 0x3: rsvd
afe_0xe7<3:2>	adc_vrefr	00	Select V$_{REF}$ for right channel 0x0: 0.6 V 0x1: 0.9 V 0x2: 1.2 V 0x3: rsvd
afe_0xe7<5:4>	adc_vrefm	00	Select V$_{REF}$ for Misc channel 0x0: 0.6 V 0x1: 0.9 V 0x2: 1.2 V 0x3: rsvd
afe_0xe7<7:6>	RSVD	-	-
afe_0xe8<3:0>	adc_ain_m_n	0000	Select negative input for Misc channel: 0x0: No input 0x1: B<0> 0x2: B<1> ... 0x8: B<7> 0x9: C<4> 0xa: C<5> 0xb: pga_n<0> (PGA left-channel negative output) 0xc: pga_n<1> (PGA right-channel negative output) 0xd: tempsensor_n (Temperature sensor negative output) 0xe: Ground 0xf: Ground

Address	Mnemonic	Default value	Description
afe_0xe8<7:4>	adc_ain_m_p	0000	Select positive input for Misc channel: 0x0: No input 0x1: B<0> 0x2: B<1> ... 0x8: B<7> 0x9: C<4> 0xa: C<5> 0xb: pga_p<0> (PGA left-channel positive output) 0xc: pga_p<1> (PGA right-channel positive output) 0xd: tempsensor_p (Temperature sensor positive output) 0xe: rsvd 0xf: rsvd
afe_0xe9<3:0>	adc_ain_l_n	00	Select negative input for left channel 0x0: No input 0x1: B<0> 0x2: B<1> ... 0x8: B<7> 0x9: C<4> 0xa: C<5> 0xb: pga_n<0> (PGA left-channel negative output) 0xc: pga_n<1> (PGA right-channel negative output) 0xd: tempsensor_n (Temperature sensor negative output) 0xe: Ground 0xf: Ground

Address	Mnemonic	Default value	Description
afe_0xe9<7:4>	adc_ain_l_p	00	Select positive input for left channel: 0x0: No input 0x1: B<0> 0x2: B<1> ... 0x8: B<7> 0x9: C<4> 0xa: C<5> 0xb: pga_p<0> (PGA left-channel positive output) 0xc: pga_p<1> (PGA right-channel positive output) 0xd: tempsensor_p (Temperature sensor positive output) 0xe: rsvd 0xf: rsvd
afe_0xea<3:0>	adc_ain_r_n	00	Select negative input for right channel: 0x0: No input 0x1: B<0> 0x2: B<1> ... 0x8: B<7> 0x9: C<4> 0xa: C<5> 0xb: pga_n<0> (PGA left-channel negative output) 0xc: pga_n<1> (PGA right-channel negative output) 0xd: tempsensor_n (Temperature sensor negative output) 0xe: Ground 0xf: Ground
afe_0xea<7:4>	adc_ain_r_p	0000	Select positive input for right channel:

Address	Mnemonic	Default value	Description
			0x0: No input
			0x1: B<0>
			0x2: B<1>
			...
			0x8: B<7>
			0x9: C<4>
			0xa: C<5>
			0xb: pga_p<0> (PGA left-channel positive output)
			0xc: pga_p<1> (PGA right-channel positive output)
			0xd: tempsensor_p (Temperature sensor positive output)
			0xe: rsvd
			0xf: rsvd
afe_0xeb<1:0>	adc_resl	11	Set resolution for left channel 0x0: 8 bits 0x1: 10 bits 0x2: 12 bits 0x3: 14 bits
afe_0xeb<3:2>	RSVD	-	-
afe_0xeb<5:4>	adc_resr	11	Set resolution for right channel 0x0: 8 bits 0x1: 10 bits 0x2: 12 bits 0x3: 14 bits
afe_0xeb<7:6>	RSVD	-	-
afe_0xec<1:0>	adc_resm	11	Set resolution for Misc channel 0x0: 8 bits 0x1: 10 bits 0x2: 12 bits 0x3: 14 bits

Address	Mnemonic	Default value	Description
afe_0xec<3:2>	RSVD	-	-
afe_0xec<4>	adc_en_diffl	0	Select input mode for left channel. 0: rsvd 1: differential mode
afe_0xec<5>	adc_en_diffr	0	Select input mode for right channel. 0: rsvd 1: differential mode
afe_0xec<6>	adc_en_diffm	0	Select input mode for Misc channel. 0: rsvd 1: differential mode
afe_0xec<7>	RSVD	-	-
afe_0xed<3:0>	adc_tsampl	0000	Number of ADC clock cycles in sampling phase for left channel to stabilize the input before sampling: 0x0: 3 cycles 0x1: 6 cycles 0x2: 9 cycles 0x3: 12 cycles … 0xf: 48 cycles
afe_0xed<7:4>	adc_tsampr	0000	Number of ADC clock cycles in sampling phase for right channel to stabilize the input before sampling: 0x0: 3 cycles 0x1: 6 cycles 0x2: 9 cycles 0x3: 12 cycles … 0xf: 48 cycles

Address	Mnemonic	Default value	Description
afe_0xee<3:0>	adc_tsampm	0000	Number of ADC clock cycles in sampling phase for Misc channel to stabilize the input before sampling: 0x0: 3 cycles 0x1: 6 cycles 0x2: 9 cycles 0x3: 12 cycles … 0xf: 48 cycles
afe_0xef<7:0>	r_max_mc[7:0]	-	r_max_mc[9:0] serves to set length of "capture" state for Misc channel.
afe_0xf0<7:0>	r_max_c[7:0]	-	r_max_c[9:0] serves to set length of "capture" state for left and right channel.
afe_0xf1<3:0>	r_max_s	-	
afe_0xf1<5:4>	r_max_c[9:8]	-	r_max_s serves to set length of "set" state for left, right and Misc channel.
afe_0xf1<7:6>	r_max_mc[9:8]	-	Note: State length indicates number of 24M clock cycles occupied by the state.
afe_0xf2<0>	r_en_left	0	Enable left channel. 1: enable
afe_0xf2<1>	r_en_right	0	Enable right channel. 1: enable
afe_0xf2<2>	r_en_misc	-	Enable Misc channel sampling. 1: enable
afe_0xf2<3>	rsvd	0	rsvd
afe_0xf2<6:4>	r_max_scnt	00	Set total length for sampling state machine (i.e. max state index)
afe_0xf2<7>	rsvd	-	-
afe_0xf3<7:0>	rsvd	-	-
afe_0xf4<2:0>	adc_clk_div	011	ADC clock (derive from external 24M crystal) ADC clock frequency = 24M/(adc_clk_div+1)
afe_0xf4<7:3>	rsvd	-	-
afe_0xf5<7:0>	rsvd	-	rsvd

Address	Mnemonic	Default value	Description
afe_0xf6<7:0>	rsvd	-	rsvd
afe_0xf7<7:0>	adc_dat[7:0]	-	Read only, Misc adc dat[7:0]
afe_0xf8<7:0>	adc_dat[15:8]	-	Read only [7]: vld, ADC data valid status bit (This bit will be set as 1 at the end of capture state to indicate the ADC data is valid, and will be cleared when set state starts.) [6:0]: Misc adc_dat[14:8]
afe_0xf9<3:2>	rsvd	00	rsvd
afe_0xfa<7:6>	adc_sel_ai_scale	0	Analog input pre-scaling select sel_ai_scale[1:0]: scaling factor 0x0: 1 0x1: rsvd 0x2: rsvd 0x3: 1/8
afe_0xfc<4>	rsvd	0	rsvd
afe_0xfc<5>	adc_pd	1	Power down ADC 1: Power down 0: Power up

11. PGA

The TLSR8359 integrates a PGA (Programmable Gain Amplifier) module.

The PGA supports two channels including left channel and right channel, and each channel consists of Boost stage pre-amplifier and Gain stage post-amplifier.

By adjusting the gain of pre-amplifier and post-amplifier, the PGA can amplify differential analog input signals from specific pins before ADC sampling.

Figure 11-1 Block Diagram of PGA
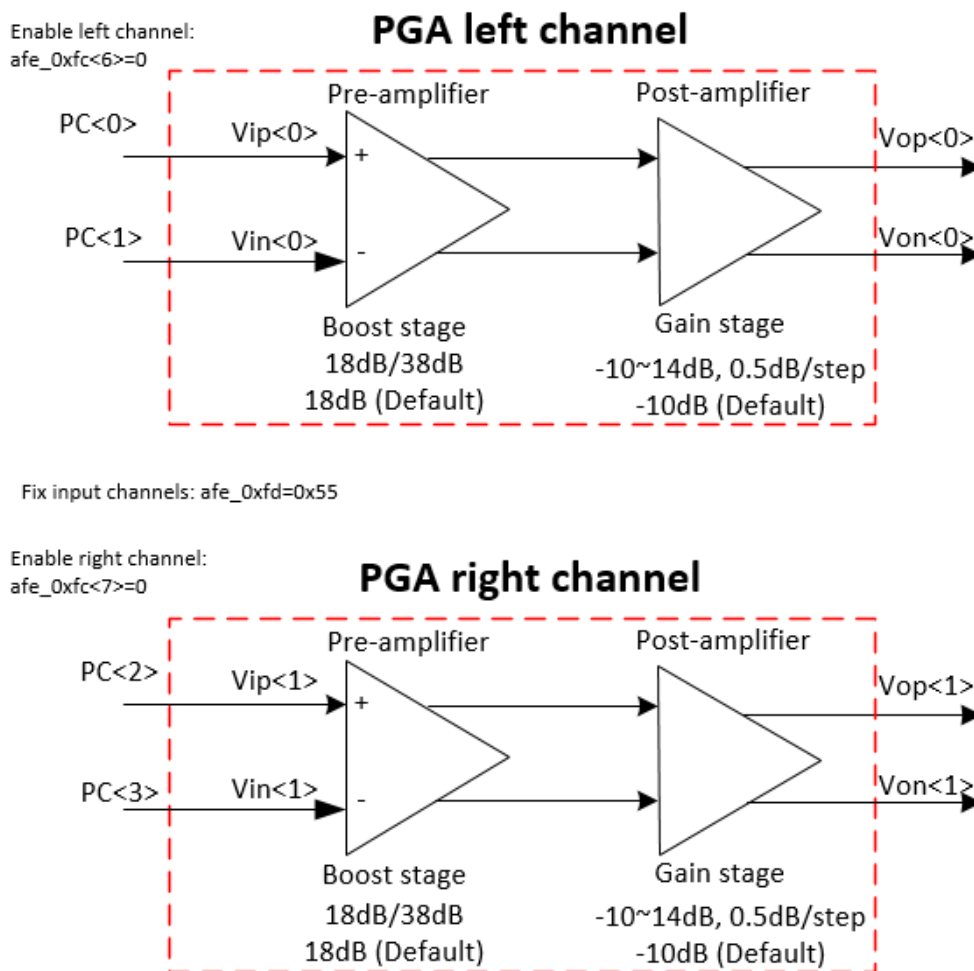
NOTE:

· Vip<0>, Vin<0>: Positive/Negative input of PGA left channel;

· Vop<0>, Von<0>: Positive/Negative output of PGA left channel;

11.1 Power On/Down

Both PGA left channel and right channel are disabled by default.

To power on PGA left/right channel, the analog register pga_pd_l (afe_0xfc<6>) / pga_pd_r (afe_0xfc<7>) should be set as 1b'0.

11.2 Input Channel

The analog register afe_0xfd must be set as 0x55, i.e. PGA input channel is fixed as PC<0> ~ PC<3>.

- PC<0>: Positive input for PGA left channel.
- PC<1>: Negative input for PGA left channel.
- PC<2>: Positive input for PGA right channel.
- PC<3>: Negative input for PGA right channel.

11.3 Adjust Gain

Address 0xb63[7] should be set as 1b'1.

The PGA gain is directly adjustable via digital register 0xb63: Address 0xb63[6] serves to set gain for the pre-amplifier as 18 dB (1b'0, default) or 38 dB (1b'1); while address 0xb63[5:0] serves to set gain for the post-amplifier as -10 dB (0x0, default) ~ 14 dB (0x30) with step of 0.5 dB. The total PGA gain should be the sum of the two gain values.

11.4 Register Table

Table 11-1 Analog Register Table Related to PGA

Address	Mnemonic	Default	Description
afe_0xfc<6>	pga_pd_l	1	Power down left channel PGA 1: Power down, 0: Power up
afe_0xfc<7>	pga_pd_r	1	Power down right channel PGA 1: Power down, 0: Power up
afe_0xfd<1:0>	pga_sel_vin_l_p	00	Select left channel positive input source. Gate off all input with pga_pd_l. 0x1: C<0> 0x0: No input 0x2: No input 0x3: No input
afe_0xfd<3:2>	pga_sel_vin_l_n	00	Select left channel negative input source. Gate off all input with pga_pd_l. 0x1: C<1> 0x0: No input 0x2: No input 0x3: No input
afe_0xfd<5:4>	pga_sel_vin_r_p	00	Select right channel positive input source. Gate off all input with pga_pd_r. 0x1: C<2> 0x0: No input 0x2: No input 0x3: No input
afe_0xfd<7:6>	pga_sel_vin_r_n	00	Select right channel negative input source. Gate off all input with pga_pd_r. 0x1: C<3> 0x0: No input

Address	Mnemonic	Default	Description
			0x2: No input
			0x3: No input

Table 11-2 Digital Register Table Related to PGA

Address	Mnemonic	Type	Description	Reset Value
0xb63	PGA_FIX_VALUE	RW	[6:0]: Pga gain fix value [7]: pga gain fix enable	0x80

12. Temperature Sensor

The TLSR8359 integrates a temperature sensor and it's used in combination with the SAR ADC to detect real-time temperature.

The temperature sensor is disabled by default. The analog register afe_0x07<4> should be set as 1b'0 to enable the temperature sensor.

Table 12-1 Analog Register for Temperature Sensor

Address	Name	Description	Default Value
afe_0x07<4>	pd_temp_sensor_3V	Power on/down temperature sensor: 0: Power up 1: Power down	1

The temperature sensor embeds two diodes. It takes the real-time temperature (T) as input, and outputs two-way forward voltage drop (V_{BE}) signals of diodes as positive and negative output respectively.

Figure 12-1 Block Diagram of Temperature Sensor
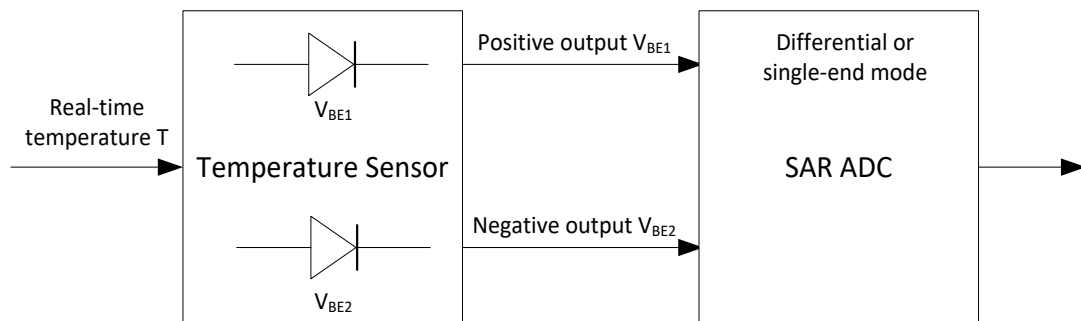
The difference of the two-way V_{BE} signals (ΔV_{BE}) is determined by the real-time temperature T, as shown below:

$$\Delta V_{BE} = 130mV + 0.51mV/°C * \left(T - (-40°C)\right)$$
$$= 130mV + 0.51mV/°C * (T + 40°C)$$

In this formula, "130mV" indicates the value of ΔV_{BE} at the temperature of "−40℃".

To detect the temperature, the positive and negative output of the temperature sensor should be enabled as the input channels of the SAR ADC. The ADC will convert the two-way V_{BE} signals into digital signal.

- When the ADC is configured as differential mode, the positive and negative output of the temperature sensor should be configured as differential input of the ADC. The ADC should initiate one operation and obtain one output signal (ADCOUT), therefore,

$$\Delta V_{BE} = \frac{ADCOUT}{2^N - 1} * V_{REF}.$$

In the formula, "N" and "V_{REF}" indicate the selected resolution and reference voltage of the SAR ADC.

Then the real-time temperature T can be calculated according to the ΔV_{BE}.

13. Low Power Comparator

The TLSR8359 embeds a low power comparator. This comparator takes two inputs: input derived from external PortB (PB<1> ~ PB<7>), and reference input derived from internal reference, PB<0>, PB<3>, AVDD3 or float.

By comparing the input voltage multiplied by selected scaling coefficient with reference input voltage, the low power comparator will output high or low level accordingly.

Figure 13-1 Block Diagram of Low Power Comparator
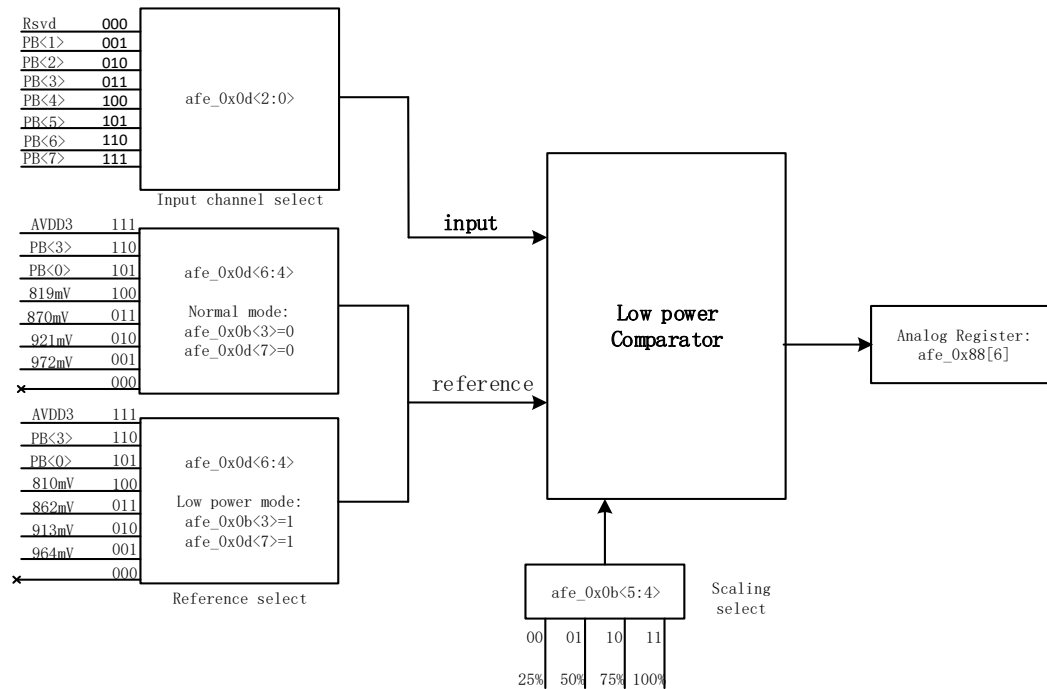
13.1 Power On/Down

The low power comparator is powered down by default.

The analog register afe_0x07<3> serves to control power state of the low power comparator: By clearing this bit, this comparator will be powered on; by setting this bit to 1b'1, this comparator will be powered down.

To use the low power comparator, first set afe_0x07<3> as 1b'0, then the 32K RC clock source is enabled as the comparator clock.

13.2 Select Input Channel

Input channel is selectable from the PortB (PB<1> ~ PB<7>) via the analog register afe_0x0d<2:0>.

13.3 Select Mode and Input Channel for Reference

Generally, it's needed to clear both the afe_0x0b<3> and afe_0x0d<7> to select the normal mode. In normal mode, the internal reference is derived from UVLO and has higher accuracy, but current bias is larger (10 µA); reference voltage input channel is selectable from internal reference of 972 mV, 921 mV, 870 mV and 819 mV, as well as PB<0>, PB<3>, AVDD3 and float.

To select the low power mode, both the afe_0x0b<3> and afe_0x0d<7> should be set as 1b'1. In low power mode, the internal reference is derived from Bandgap and has lower accuracy, but current bias is decreased to 50 nA; reference voltage input channel is selectable from internal reference of 964 mV, 913 mV, 862 mV and 810 mV, as well as PB<0>, PB<3>, AVDD3 and float.

13.4 Select Scaling Coefficient

Equivalent reference voltage equals the selected reference input voltage divided by scaling coefficient.

The analog register afe_0x0b<5:4> serves to select one of the four scaling options: 25%, 50%, 75% and 100%.

13.5 Low Power Comparator Output

The low power comparator output is determined by the comparison result of the value of [input voltage *scaling] and reference voltage input. The comparison principle is shown as below:

- If the value of [input voltage *scaling] is larger than reference voltage input, the output will be low ("0").
- If the value of [input voltage *scaling] is lower than reference voltage input, the output will be high ("1").
- If the value of [input voltage *scaling] equals reference voltage input, or input channel is selected as float, the output will be uncertain.

User can read the output of the low power comparator via the analog register afe_0x88[6].

The output of the low power comparator can be used as signal to wakeup system from low power modes.

13.6 Register Table

Table 13-1 Analog Register Table Related to Low Power Comparator

Address	Name	Description	Default Value
afe_0x07<3>	pd_lc_comp_3V	Power on/down low power comparator: 0: Power up 1: Power down	1
afe_0x0b<3>	ref_mode_sel	Reference mode select: 0: normal mode (current bias 10 μA) 1: low power mode (current bias 50 nA) See afe_0x0d<7>.	1
afe_0x0b<5:4>	comp_refscale<1:0>	Reference voltage scaling: 00: 25% 01: 50% 10: 75% 11: 100%	01
afe_0x0d<2:0>	comp_chsel<2:0>	Input Channel select: 000: rsvd 001: B<1> 010: B<2> 011: B<3> 100: B<4> 101: B<5> 110: B<6> 111: B<7>	000

Address	Name	Description	Default Value
afe_0x0d<3>	vbus_inen	rsvd	0
afe_0x0d<6:4>	comp_refesel<2:0>	Reference select: normal mode low power mode 000: Float 000: Float 001: 972 mV 001: 964 mV 010: 921 mV 010: 913 mV 011: 870 mV 011: 862 mV 100: 819 mV 100: 810 mV 101: B<0> 101: B<0> 110: B<3> 110: B<3> 111: AVDD3 111: AVDD3	000
afe_0x0d<7>	pd_l10u	Enable or disable 10 µA current bias: 0: Enable 10 µA current bias 1: Disable 10 µA current bias	1

14. AES

The TLSR8359 embeds AES module with encryption and decryption function. The input 128-bit plaintext in combination of key is converted into the final output ciphertext via encryption; the 128-bit ciphertext in combination of key can also be converted into 128-bit plaintext via decryption.

The AES hardware accelerator provides automatic encryption and decryption. It only takes (1000*system clock cycles) to implement AES encryption/decryption. Suppose system clock is 20 MHz, the time needed for AES encryption/decryption is 50 µs.

Both RISC mode and DMA mode are supported for AES operation.

14.1 RISC Mode

For RISC mode, configuration of related registers is as follows:

1. Set the value of key via writing registers AES_KEY0~ AES_KEY15 (address 0x550~0x55f).
2. Set operation method of AES module via register AES_CTRL: set address 0x540[0] as 1b'1 for decryption method, while clear this bit for encryption method.
3. For encryption method, write registers AES-DAT0~ AES-DAT3 (address 0x548~0x54b) for four times to set the 128-bit plaintext. After encryption, the 128bit ciphertext can be obtained by reading address 0x548~0x54b for four times.
4. For decryption method, write registers AES-DAT0~ AES-DAT3 (address 0x548~0x54b) for four times to set the 128-bit ciphertext. After decryption, the 128-bit plaintext can be obtained by reading address 0x548~0x54b for four times.
5. Address 0x540 bit[1] and bit[2] are read only bits: bit[1] will be cleared automatically after quartic writing of address 0x548~0x54b; bit[2] will be set as 1 automatically after encryption/decryption, and then cleared automatically after quartic reading of address 0x548~0x54b.

14.2 DMA Mode

As for DMA mode, it is only needed to configure the value of key and encryption/decryption method for AES module. Please refer to point 1 ~ 2 in Section 14.1.

14.3 AES-CCM

The AES-CCM (Counter with the CBC-MAC) mode is disabled by default. AES output is directly determined by current encryption and decryption, irrespective of previous encryption and decryption result.

If 0x540[7] is set as 1b'1 to enable AES-CCM mode, AES output will also take previous encryption and decryption result into consideration.

14.4 Register Table

Table 14-1 Register Table Related to AES

Address	Mnemonic	Type	Description	Reset Value
0x540	AES_CTRL	R/W	[0] Select decrypt/encrypt. 1: decrypt, 0: encrypt [1] Read-only. 1: input data needed, 0: input data ready. [2] Read-only. 0: output data not ready, 1: output data ready. [7] 1: enable AES-CCM mode.	0x02
0x548	AES-DAT0	-	Input/Output Data byte 0	0x00
0x549	AES-DAT1	-	Input/Output Data byte 1	0x00
0x54a	AES-DAT2	-	Input/Output Data byte 2	0x00
0x54b	AES-DAT3	-	Input/Output Data byte 3	0x00
0x550	AES_KEY0	R/W	[7:0] KEY0	0x00
0x551	AES_KEY1	R/W	[7:0] KEY1	0x00
0x552	AES_KEY2	R/W	[7:0] KEY2	0x00

Address	Mnemonic	Type	Description	Reset Value
0x553	AES_KEY3	R/W	[7:0] KEY3	0x00
0x554	AES_KEY4	R/W	[7:0] KEY4	0x00
0x555	AES_KEY5	R/W	[7:0] KEY5	0x00
0x556	AES_KEY6	R/W	[7:0] KEY6	0x00
0x557	AES_KEY7	R/W	[7:0] KEY7	0x00
0x558	AES_KEY8	R/W	[7:0] KEY8	0x00
0x559	AES_KEY9	R/W	[7:0] KEY9	0x00
0x55a	AES_KEY10	R/W	[7:0] KEY10	0x00
0x55b	AES_KEY11	R/W	[7:0] KEY11	0x00
0x55c	AES_KEY12	R/W	[7:0] KEY12	0x00
0x55d	AES_KEY13	R/W	[7:0] KEY13	0x00
0x55e	AES_KEY14	R/W	[7:0] KEY14	0x00
0x55f	AES_KEY15	R/W	[7:0] KEY15	0x00

15. Key Electrical Specifications

NOTE: *The electrical characteristics currently listed in this section are target specifications and only supplied for reference. Some data may be updated according to actual test results.*

15.1 Absolute Maximum Ratings

Table 15-1 Absolute Maximum Ratings

Item	Sym.	Min.	Max	Unit	Conditions
Supply voltage	VDD	-0.3	3.6	V	All AVDD, DVDD and VDD_IO pin must have the same voltage
Voltage on input pin	V_{In}	-0.3	VDD+0.3	V	-
Output voltage	V_{Out}	0	VDD	V	-
Storage temperature range	T_{Str}	-65	150	°C	-
Soldering temperature	T_{Sld}	-	260	°C	-

CAUTION: *Stresses above those listed in "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress only rating and operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied.*

15.2 Recommended Operating Conditions

Table 15-2 Recommended Operating Conditions

Item	Sym.	Min	Typ.	Max	Unit	Conditions
Power-supply voltage	VDD	1.8	3.3	3.6	V	All AVDD, DVDD and VDD_IO pin must have the same voltage
Supply rise time (from 1.6 V	t_R	-	-	10	ms	-

Item	Sym.	Min	Typ.	Max	Unit	Conditions
to 1.8 V)						
Operating temperature range	T$_{Opr}$	-40	-	85	℃	-

15.3 DC Characteristics

Table 15-3 DC Characteristics (VDD = 3.3 V, T = 25°C)

Item	Sym.	Min	Typ.	Max	Unit	Conditions
RX current	I$_{Rx}$	-	5.3	-	mA	Whole chip
TX current	I$_{Tx}$	-	4.8	-	mA	Whole chip @ 0 dBm with DCDC
Deep sleep with 8 KB SRAM retention	I$_{Deep1}$	-	1	-	uA	-
Deep sleep with 16 KB SRAM retention		-	1.2	-	uA	-
Deep sleep with 32 KB SRAM retention		-	1.4	-	uA	-
Deep sleep without SRAM retention	I$_{Deep2}$	-	0.4	-	uA	-

15.4 AC Characteristics

Table 15-4 AC Characteristics (VDD = 3.3 V, T = 25°C)

Item	Sym.	Min	Typ.	Max	Unit	Conditions
Digital inputs/outputs						
Input high voltage	VIH	0.7VDD	-	VDD	V	-

Item	Sym.	Min	Typ.	Max	Unit	Conditions
Input low voltage	VIL	VSS	-	0.3VDD	V	-
Output high voltage	VOH	0.9VDD	-	VDD	V	-
Output low voltage	VOL	VSS	-	0.1VDD	V	-
USB Characteristics						
USB output signal cross-over voltage	V$_{Crs}$	1.3	-	2.0	V	-
RF Performance						
Item		Min	Typ	Max	Unit	
RF frequency range	-	2400	-	2483.5	MHz	Programmable in 1 MHz step
Data rate	2.4G proprietary 1 Mbps, ±250 kHz deviation 2.4G proprietary 2 Mbps, ±500 kHz deviation 2.4G proprietary 500 kbps, ±125 kHz deviation 2.4G proprietary 250 kbps, ±62.5 kHz deviation					
1 Mbps RF_RX Performance (±250 kHz Deviation)						
Sensitivity	1 Mbps	-	-96	-	dBm	-
Frequency offset tolerance	-	-250	-	+300	kHz	-
Co-channel rejection	-	-	-11	-	dB	Wanted signal at -67 dBm
In-band blocking rejection (equal modulation interference)	+1/-1 MHz offset	-	1/3	-	dB	Wanted signal at -67 dBm
	+2/-2 MHz offset	-	37/39	-	dB	
	>=3 MHz offset	-	42	-	dB	

Item	Sym.	Min	Typ.	Max	Unit	Conditions
Image rejection	-	-	37	-	dB	Wanted signal at -67 dBm
1 Mbps RF_TX Performance						
Output power, maximum setting	-	-	10	-	dBm	-
Output power, minimum setting	-	-	-45	-	dBm	-
Programmable output power range	-		55		dB	-
Modulation 20 dB bandwidth	-	-	2.5	-	MHz	-
2 Mbps RF_RX Performance (±500kHz Deviation)						
Sensitivity	2 Mbps	-	-93	-	dBm	-
Frequency offset tolerance	-	-300	-	+200	kHz	-
Co-channel rejection	-	-	-10	-	dB	Wanted signal at -67 dBm
In-band blocking rejection	+2/-2 MHz offset	-	6/6	-	dB	Wanted signal at -67 dBm
	+4/-4 MHz offset	-	39/38		dB	
	> 4 MHz offset	-	42	-	dB	
Image rejection	-	-	25	-	dB	Wanted signal at -67 dBm
2 Mbps RF_TX Performance						

Item	Sym.	Min	Typ.	Max	Unit	Conditions
Output power, maximum setting	-	-	10	-	dBm	-
Output power, minimum setting	-	-	-45	-	dBm	-
Programmable output power range	-	55			dB	-
Modulation 20 dB bandwidth	-	-	1.4	-	MHz	-
RSSI						
RSSI range	-	-100	-	10	dBm	-
Resolution	-	-	1	-	dB	-
24 MHz Crystal						
Nominal frequency (parallel resonant)	f$_{NOM}$	-	24	-	MHz	-
Frequency tolerance	f$_{TOL}$	-20	-	+20	ppm	-
Load capacitance	C$_L$	5	12	18	pF	Programmable on chip load cap
Equivalent series resistance	ESR	-	50	100	ohm	-
32.768 kHz Crystal						
Nominal frequency (parallel resonant)	f$_{NOM}$	-	32.768	-	kHz	-
Frequency tolerance	f$_{TOL}$	-100	-	+100	ppm	-
Load capacitance	C$_L$	6	9	12.5	pF	Programmable on chip load cap
Equivalent series	ESR	-	50	80	kohm	-

Item	Sym.	Min	Typ.	Max	Unit	Conditions
resistance						
24 MHz RC Oscillator						
Nominal frequency	f_{NOM}		24		MHz	-
Frequency tolerance	f_{TOL}		1		%	On chip calibration
32 kHz RC Oscillator						
Nominal frequency	f_{NOM}	-	32	-	kHz	-
Frequency tolerance	f_{TOL}	-	0.03		%	On chip calibration
Calibration time		-	-	3	-	ms	-
ADC						
Differential nonlinearity	DNL	-	-	1	LSB	10bit resolution mode
Integral nonlinearity	INL	-	-	2	LSB	10bit resolution mode
Signal-to-noise and distortion ratio	SINAD	-	70	-	dB	f_{IN} =1 kHz, f_S = 16 kHz
Effective Number of Bits	ENOB	-	10.5	-	bits	-
Sampling frequency	Fs	-	-	200	ksps	-

15.5 SPI Characteristics

Table 15-5 SPI Characteristics

(over process, voltage 1.9 ~ 3.6 V, and T = -40 ~ +85°C)

Item	Sym.	Min	Typ.	Max	Unit	Conditions
CK frequency	F_{CK}	-	-	4	MHz	Slave
CK duty cycle clock	-	-	50	-	%	Master

Item	Sym.	Min	Typ.	Max	Unit	Conditions
DI setup time	-	30	-	-	ns	Slave
	-	90	-	-	ns	Master
DI hold time	-	10	-	-	ns	Slave
	-	90	-	-	ns	Master
CK low to DO valid time	-	-	-	30	ns	Slave
	-	-	-	120	ns	Master
CN setup time	-	60	-	-	ns	Master/Slave
CN high to DI tri-state*[3]	-	-	-	-	ns	Master

[3] Note: Master actively stops reading during transmission, and Slave releases its driver DO and turns to tri-state.

15.6 I2C Characteristics

Table 15-6 I2C Characteristics

(over process, voltage 1.9 ~ 3.6 V, and T = -40 ~ +85°C)

Item	Sym.	Standard Mode		Fast Mode		Unit	Conditions
		Min	Max	Min	Max		
SCL frequency	F_{SCL}	-	100	-	400	kHz	-
Rise time of SDA and SCL signals	T_R	-	1000	-	300	ns	-
Fall time of SDA and SCL signals	T_F	-	300	-	300	ns	-
START condition hold time	$T_{HD;STA}$	4	-	0.6	-	us	-
Data hold time	$T_{HD;DAT}$	0	3.45	-	0.9	us	-
Data setup time	$T_{SU;DAT}$	250	-	100	-	ns	-
STOP condition setup time	$T_{SU;STO}$	4	-	0.6	-	us	-

15.7 Flash Characteristics

Table 15-7 Flash Memory Characteristics

(T= -40 ~ +85°C)

Item	Sym.	Min	Typ.	Max	Unit	Conditions
Retention period	-	20	-	-	year	-
Number of erase cycles	-	100k	-	-	cycle	-
VDD for programming	-	1.65	-	2.0	V	Note this refers to the SoC supply
Sector size	-	-	4	-	KB	-
Page programming time	T_{PP}	-	2	3	ms	-
Page erase time	T_{PE}	-	16	30	ms	-
Sector erase time	T_{SE}	-	16	30	ms	-
Block erase time (32 KB/64 KB)	T_{BE}	-	16	30	ms	-
Program current	I_P	-	1.5	4.0	mA	-
Erase current	I_E	-	1.5	4.0	mA	-

16. Reference Design

16.1 Application Example for TLSR8359F512ET48

16.1.1 Schematic

Figure 16-1 Schematic for TLSR8359F512ET48
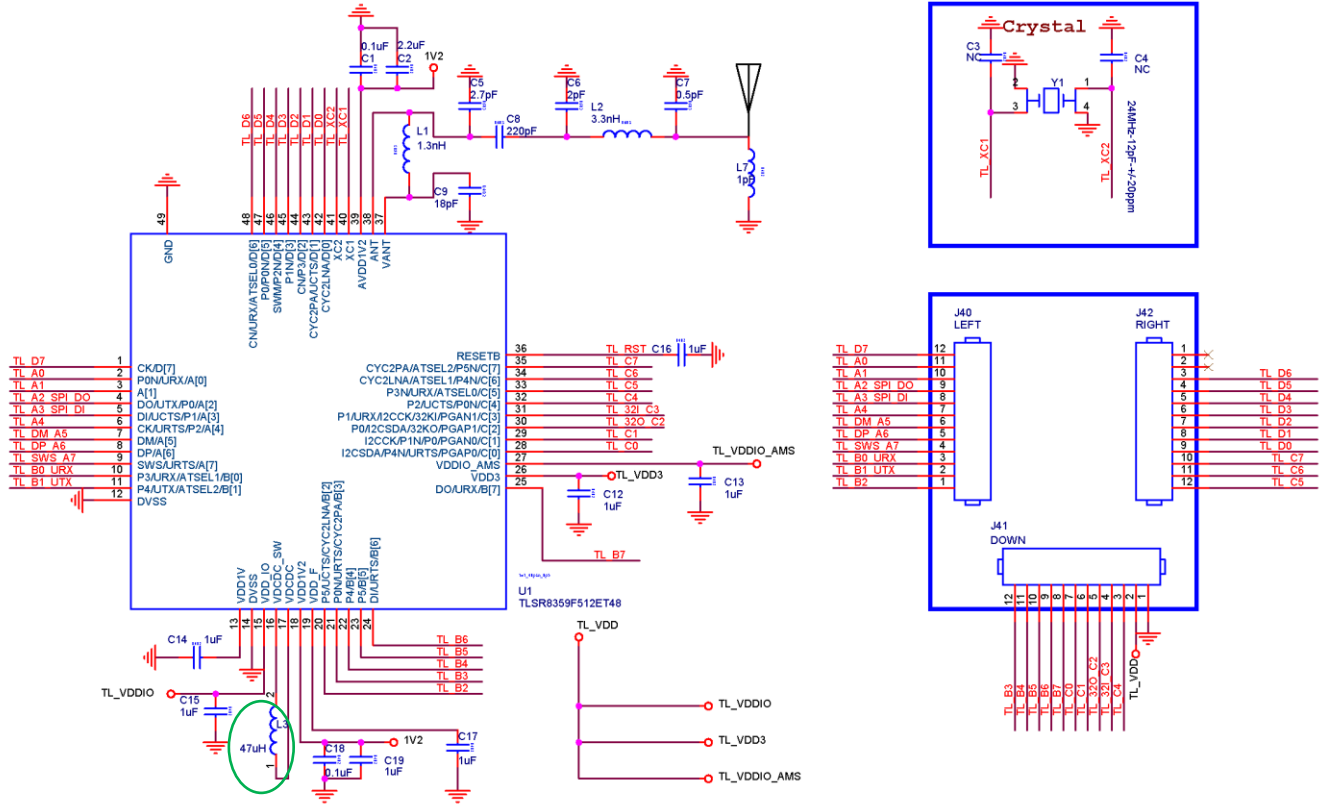
NOTE: In the reference design, the L3 adopts 47 μH inductor, which can ensure sensitivity index given in this Datasheet. User can also select 10 μH or 6.8 μH inductor instead as needed, but note that sensitivity will have 1 ~ 2 dB or so difference.

16.1.2BOM (Bill of Material)

Table 16-1 BOM Table for TLSR8359F512ET48

Quantity	Reference	Part
2	C1	0.1 μF
	C18	0.1 μF
1	C2	2.2 μF
2	C3	NC
	C4	NC
1	C5	2.7 pF
1	C6	2 pF
1	C7	0.5 pF
1	C8	220 pF
1	C9	18 pF
7	C12	1 μF
	C13	1 μF
	C14	1 μF
	C15	1 μF
	C16	1 μF
	C17	1 μF
	C19	1 μF
1	L1	1.3 nH
1	L2	3.3 nH
1	L3	47 μH
1	L7	1 pF
1	U1	TLSR8359F512ET48

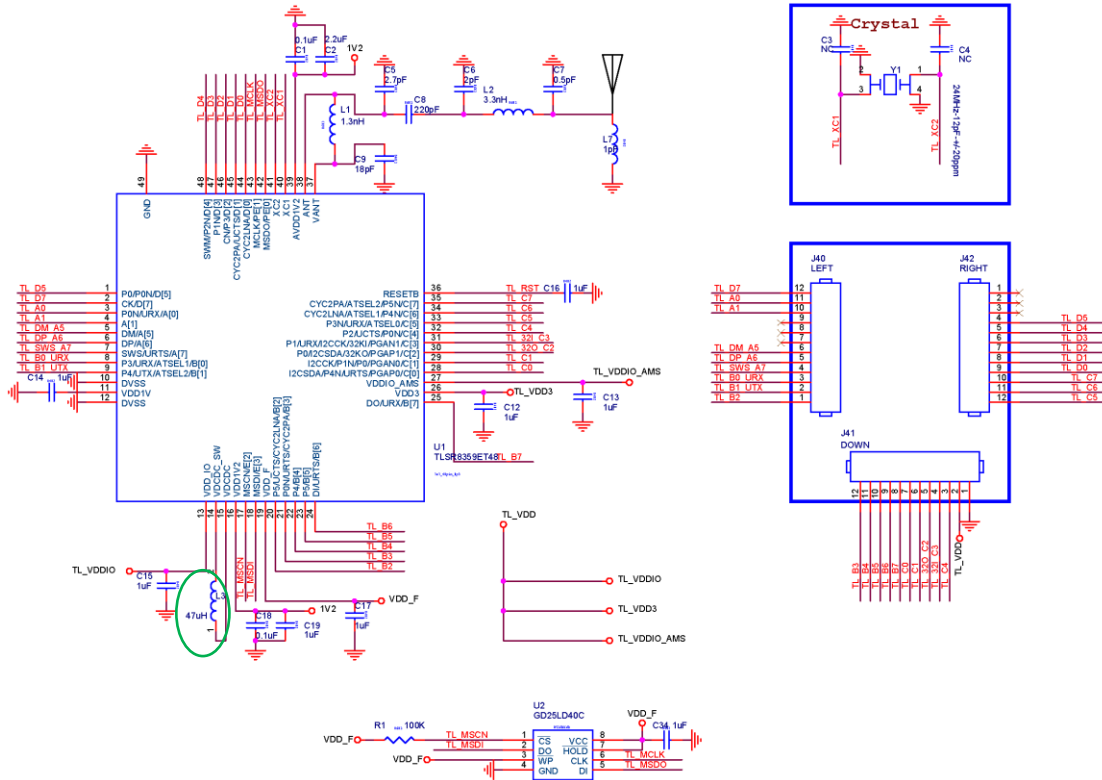| Quantity	Reference	Part
1	Y1	24MHz_12pF_+/-20 ppm

16.2 Application Example for TLSR8359ET48

16.2.1Schematic

Figure 16-2 Schematic for TLSR8359ET48

NOTE: *In the reference design, the L3 adopts 47 µH inductor, which can ensure sensitivity index given in this Datasheet. User can also select 10 µH or 6.8 µH inductor instead as needed, but note that sensitivity will have 1 ~ 2 dB or so difference.*

16.2.2 BOM (Bill of Material)

Table 16-2 BOM Table for TLSR8359ET48

Quantity	Reference	Part
2	C1	0.1 μF
	C18	0.1 μF
1	C2	2.2 μF
1	C5	2.7 pF
1	C6	2 pF
1	C7	0.5 pF
1	C8	220 pF
1	C9	18 pF
8	C12	1 μF
	C13	1 μF
	C14	1 μF
	C15	1 μF
	C16	1 μF
	C17	1 μF
	C19	1 μF
	C34	1 μF
1	L1	1.3 nH
1	L2	3.3 nH
1	L3	47 μH (alternative with 1 ~ 2 dB sensitivity difference: 10 μH or 6.8 μH)
1	L7	1 pF
1	R1	100K
1	U1	TLSR8359ET48
1	U2	GD25LD40C

Quantity	Reference	Part
1	Y1	24MHz_12pF_+/-20 ppm

16.3 Application Example for TLSR8359F512ET32

16.3.1Schematic

Figure 16-3 Schematic for TLSR8359F512ET32

NOTE: *In the reference design, the L3 adopts 47 μH inductor, which can ensure sensitivity index given in this Datasheet. User can also select 10 μH or 6.8 μH inductor instead as needed, but note that sensitivity will have 1 ~ 2 dB or so difference.*

16.3.2 BOM (Bill of Material)

Table 16-3 BOM Table for TLSR8359F512ET32

Quantity	Reference	Part
2	C1	0.1 μF
	C18	0.1 μF
1	C2	2.2 μF
2	C3	NC
	C4	NC
1	C5	2.7 pF
1	C6	2 pF
1	C7	0.5 pF
1	C8	220 pF
1	C9	18 pF
7	C12	1 μF
	C13	1 μF
	C14	1 μF
	C15	1 μF
	C16	1 μF
	C17	1 μF
	C19	1 μF
1	L1	1.3 nH
1	L2	3.3 nH
1	L3	47 μH (alternative with 1 ~ 2 dB sensitivity difference: 10 μH or 6.8 μH)
1	L7	1 pF
1	U1	TLSR8359F512ET32
1	Y1	24MHz_12pF_+/-20 ppm

16.4 Application Example for TLSR8359F64ES16

16.4.1Schematic

Figure 16-4 Schematic for TLSR8359F64ES16
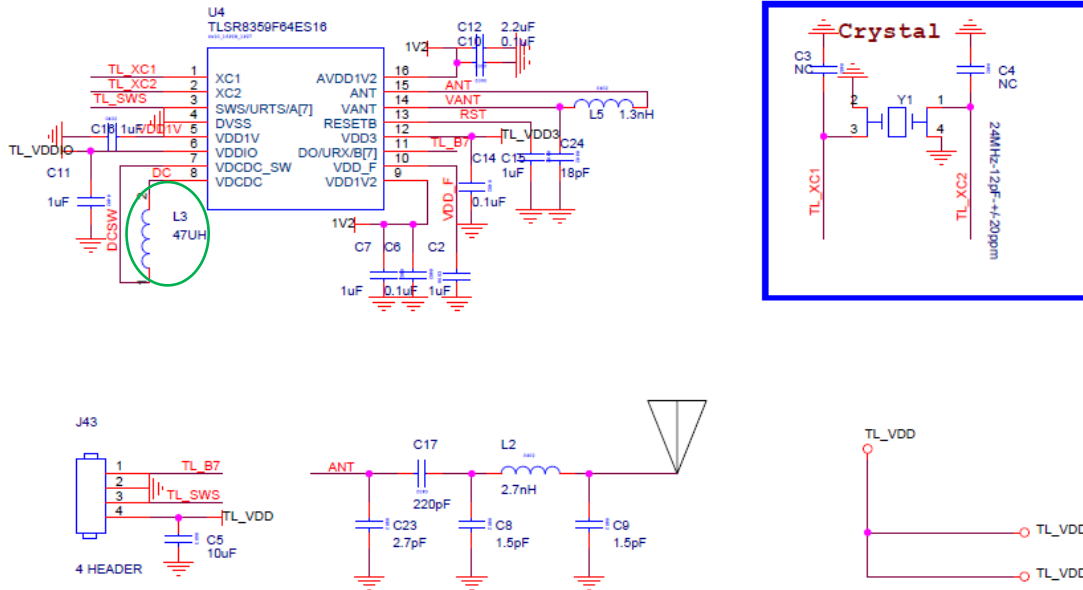
16.4.2 BOM (Bill of Material)

Table 16-4 BOM table for TLSR8359F64ES16

Quantity	Reference	Part	Spec
5	C2	1 μF	0402
	C7	1 μF	0402
	C11	1 μF	0402
	C15	1 μF	0402
	C16	1 μF	0402
1	C5	10 μF	0603C
3	C6	0.1 μF	0402
	C10	0.1 μF	0402
	C14	0.1 μF	0402
2	C8	1.5 pF	0402
	C9	1.5 pF	0402
1	C12	2.2 μF	0402
1	C17	220 pF	0402
1	C23	2.7 pF	0402
1	C24	18 pF	0402
1	J43	4 HEADER	-
1	L2	2.7 nH	0402
1	L3	47 μH (alternative with 1 ~ 2 dB sensitivity difference: 10 μH or 6.8 μH)	0805L
1	L5	1.3 nH	0402
1	U4	TLSR8359F64ES16	TSSOP_16PIN_4P4X5_0P65
1	Y1	24MHz_12pF_+/-20 ppm	OSCCC250X320X110