Datasheet for Telink BLE SoC TLSR8261

DS-TLSR8261-E14

Ver 2.3.0

2018/12/20

Keyword:

BLE; Features; Package; Pin layout; Memory; MCU; Working modes; Wakeup sources; RF Transceiver; Baseband; Clock; Timers; Interrupt; Interface; PWM; QDEC; ADC; Electrical specification; Application

Brief:

This datasheet is dedicated for Telink BLE SoC TLSR8261. In this datasheet, key features, working mode, main modules, electrical specification and application of the TLSR8261 are introduced.

TELINK SEMICONDUCTOR

Published by
Telink Semiconductor

Bldg 3, 1500 Zuchongzhi Rd,
Zhangjiang Hi-Tech Park, Shanghai, China

Legal Disclaimer

Telink Semiconductor reserves the right to make changes without further notice to any products herein to improve reliability, function or design. Telink Semiconductor disclaims any and all liability for any errors, inaccuracies or incompleteness contained herein or in any other disclosure relating to any product.

Telink Semiconductor does not assume any liability arising out of the application or use of any product or circuit described herein; neither does it convey any license under its patent rights, nor the rights of others

The products shown herein are not designed for use in medical, life-saving, or life-sustaining applications. Customers using or selling Telink Semiconductor products not expressly indicated for use in such applications do so entirely at their own risk and agree to fully indemnify Telink Semiconductor for any damages arising or resulting from such use or sale.

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:

telinkcnsales@telink-semi.com

telinkcnsupport@telink-semi.com

Revision History

Version	Major Changes	Date	Author
1.0.0	Initial release	2016/6	L.X., Cynthia
1.1.0	Updated DC characteristics and ordering information.	2016/7	L.J.R., X.S.J., Cynthia
1.2.0	Added Unique ID feature for internal 128K Flash encryption protection.	2016/8	H.Z.T., Cynthia
1.3.0	Added pull-up resistor statement for I2C.	2016/10	S.G.J., Cynthia
1.4.0	Deleted PGA related contents.	2016/10	J.H.P., Cynthia
1.5.0	Updated reference design. Deleted Wakeup source – GPIO and renamed Wakeup source – pad as Wakeup source – IO.	2016/10	H.Z.F., Z.X.D., Cynthia
1.6.0	Added watchdog status bit and flag clearing. Marked retention analog registers in deep sleep. Introduced impact to the retention analog registers by watchdog reset, POR and chip software reset.	2016/10	Z.X.D., S.G.J., Cynthia
1.7.0	Updated Reset and Power Management	2016/11	S.G.J., Cynthia
1.8.0	Modified I2C Master clock.	2016/12	Y.C.Q., Cynthia
1.9.0	Updated ordering information.	2017/3	X.S.J., Cynthia
2.0.0	Updated section 7.3.1 Communication protocol.	2017/6	S.G.J., Cynthia
2.1.0	Updated section 10.1 ADC clock. Added 9.6 Timing sequence.	2017/8	L.Y., Y.C.Q., Cynthia
2.2.0	Updated the followings sections: 4.3 System clock (0x66[4:0]), 7.1.1.3 GPIO lookup table (notes), 6.2 Register configuration and 9.7 Register table (removed QDEC interrupt), 11.2 Recommended operating condition (supply rise time).	2017/12	S.G.J., Y.C.Q., X.W.W., Cynthia

Version	Major Changes	Date	Author
2.3.0	Updated section 1.2.2 RF Features, 1.2.3 Features of power management module, 1.4 Ordering information, 2.5.1 Power-On-Reset (POR) and Brown-out detect, 4.2 Register table, 4.3 System clock, 11 Key Electrical Specifications.	2018/12	LY, YY, SGJ, LX, YHL, Cynthia

1 Table of contents

2 Table of Figures

3 Table of Tables

1 Overview

The TLSR8261F128 is Telink-developed BLE SoC solution. It's compatible with Bluetooth standard and supports BLE specification up to version 4.2. It allows easy connectivity with Bluetooth Smart Ready mobile phones, tablets, laptops, which supports BLE slave and master mode operation, including broadcast, encryption, connection updates, and channel map updates. The TLSR8261F128 is completely RoHS-compliant and 100% lead (Pb)-free.

1.1 Block diagram

The TLSR8261F128 is designed to offer high integration, ultra-low power application capabilities. It integrates strong 32-bit MCU, BLE/2.4G Radio, 16KB SRAM, 128KB internal Flash, 14bit ADC, 6-channel PWM (2-channel IR), one quadrature decoder (QDEC), abundant GPIO interfaces, multi-stage power management module and nearly all the peripherals needed for Bluetooth Low Energy applications development.

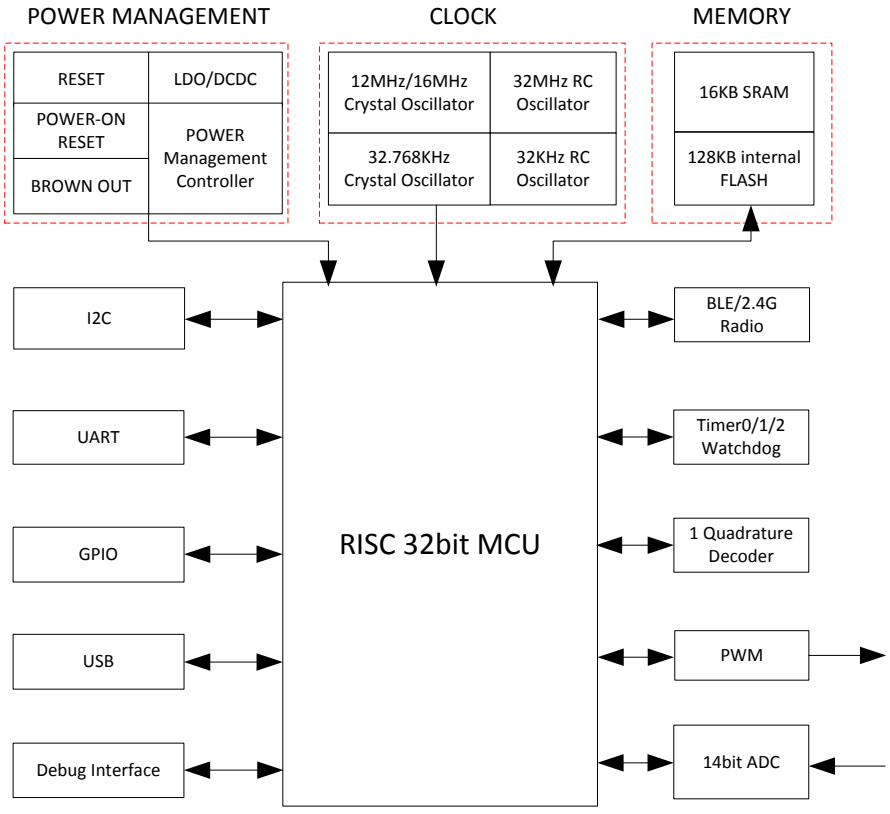


Figure 1- 1 Block diagram of the system

With the high integration level of TLSR8261F128, 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) Embedded 32-bit high performance MCU with clock up to 48MHz.
2) Program memory: internal 128KB Flash.
3) Data memory: 16KB on-chip SRAM.
4) 12MHz/16MHz & 32.768KHz Crystal and 32KHz/32MHz embedded RC oscillator.
5) A rich set of I/Os:
   - ✧ Up to 14 GPIOs;
   - ✧ I2C;
   - ✧ UART with hardware flow control;
   - ✧ USB;
   - ✧ Debug Interface: SWS.
6) Up to 6 channels of PWM, 2-channel IR.
7) Sensor:
   - ✧ 14bit ADC;
   - ✧ Temperature sensor.
8) One quadrature decoder.
9) Embedded hardware AES.
10) Operating temperature: -40℃~+85℃ temperature range.

1.2.2 RF Features

RF features include:

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

2) Bluetooth 4.2 Compliant, 1Mbps and 2Mbps LE Enhancement FIPD version.
3) Rx Sensitivity: -91dBm@BLE 1Mbps, -89dBm @BLE 2Mbps.
4) Tx output power: +7dBm.
5) Single-pin antenna interface.
6) RSSI monitoring.

1.2.3 Features of power management module

Features of power management module include:

1) Embedded LDO.
2) Battery monitor: Supports low battery detection.
3) Power supply: 1.9V~3.6V.
4) Multiple stage power management to minimize power consumption.
5) Low power consumption:
   - ✧ Tx Transceiver current: 14.7mA @0dBm power, 24.5mA @max power
   - ✧ Rx Transceiver current: 13.8mA
   - ✧ Suspend mode current: 10uA (IO wakeup), 11.4uA (32k RC wakeup)
   - ✧ Deep sleep mode current: 2.4uA

1.2.4 USB features

USB features include:

1) Compatible with USB2.0 Full speed mode.
2) Supports 9 endpoints.
3) Supports ISP (In-System Programming) via USB port.

1.2.5 Flash features

The TLSR8261F128 embeds Flash with features below:

1) Total 128KB (1Mbits);
2) Flexible architecture: 4KB per Sector, 64KB/32KB per block;
3) Up to 256 Bytes per programmable page;
4) Write protect all or portions of memory;
5) Sector erase (4KB);

6) Block erase (32KB/64KB);
7) Cycle Endurance: 100,000 program/erases;
8) Data Retention: typical 20-year retention;
9) Flash encryption protection: Pre-loaded read-only unique ID for chip identification and traceability.

1.3 Typical applications

The TLSR8261F128 can be applied to BLE systems; its typical applications include, but are not limited to the following:

- ✧ Smartphone and tablet accessories
- ✧ Remote Control and 3D glasses
- ✧ Sports and fitness tracking
- ✧ Wearable devices

1.4 Ordering information

Table 1- 1 Ordering information of the TLSR8261F128*[1]

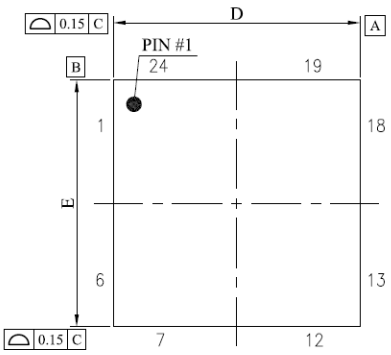
Product Series	Package Type	Temperature Range	Product Part No.	Packing Method*[2]	Minimum Order Quantity
TLSR8261F128	24-pin 4x4mm TQFN	-40℃~ +85℃	TLSR8261F128 ET24	TR	3000

[1] MSL (Moisture Sensitivity Level): The 8261 series is applicable to MSL3 (Based on JEDEC Standard J-STD-020).
- ✧ After the packing opened, the product shall be stored at <30℃/ <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.

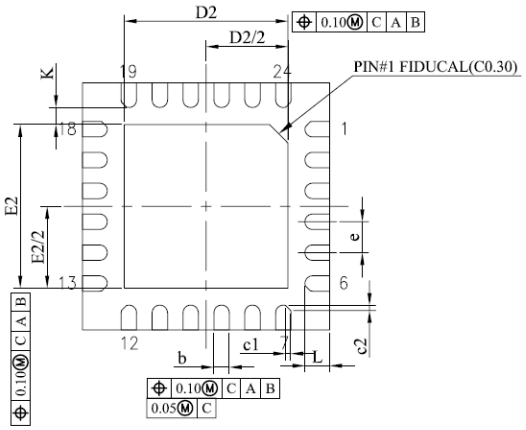
[2] Packing method “TR” means tape and reel. The tape and reel material DO NOT support baking under high temperature.

1.5 Package

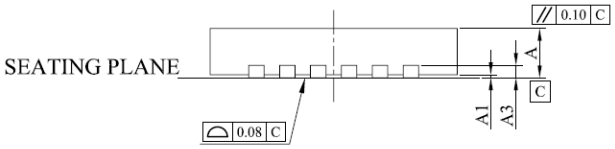
Package dimension for the TLSR8261F128ET24 is shown as Figure 1-2.



Top View



Bottom View



Side View

SYMBOL	DIMENSION (MM)			DIMENSION (MIL)		
	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.
A	0.70	0.75	0.80	27.6	29.5	31.5
A1	0	0.02	0.05	0	0.8	2.0
A3	---	0.20REF	—	---	7.9REF	—
b	0.18	0.25	0.30	7.1	9.8	11.8
D	3.90	4.00	4.10	153.5	157.5	161.4
D2	2.55	2.65	2.75	100.4	104.3	108.3
E	3.90	4.00	4.10	153.5	157.5	161.4
E2	2.55	2.65	2.75	100.4	104.3	108.3
e	—	0.50BSC	---	—	19.7BSC	---
K	0.20	—	---	7.9	—	—
L	0.35	0.40	0.45	13.8	15.7	17.7
c1	---	0.08	---	---	3.1	—
c2	---	0.08	---	---	3.1	—

NOTE:
1. DIMENSIONING AND TOLERANCING CONFORM TO ASME Y14.5M-1994.
2. REFER TO JEDEC STD.MO-220 WGGD-6
3. DIMENSION "b" APPLIES TO METALLIZED TERMINAL AND IS MEASURED BETWEEN 0.18 AND 0.30mm FROM TERMINAL TIP.
4. LEADFRAME THICKNESS IS 0.203MM (8 MIL).
5. DIMENSION"D"&"E" WILL INCLUDE ALL SIDE BURR INDUCED DURING ASSEMBLY.

Figure 1- 2 Package dimension for TLSR8261F128ET24

1.6 Pin layout

Pin assignment for the TLSR8261F128ET24 is as shown in Figure 1- 3:

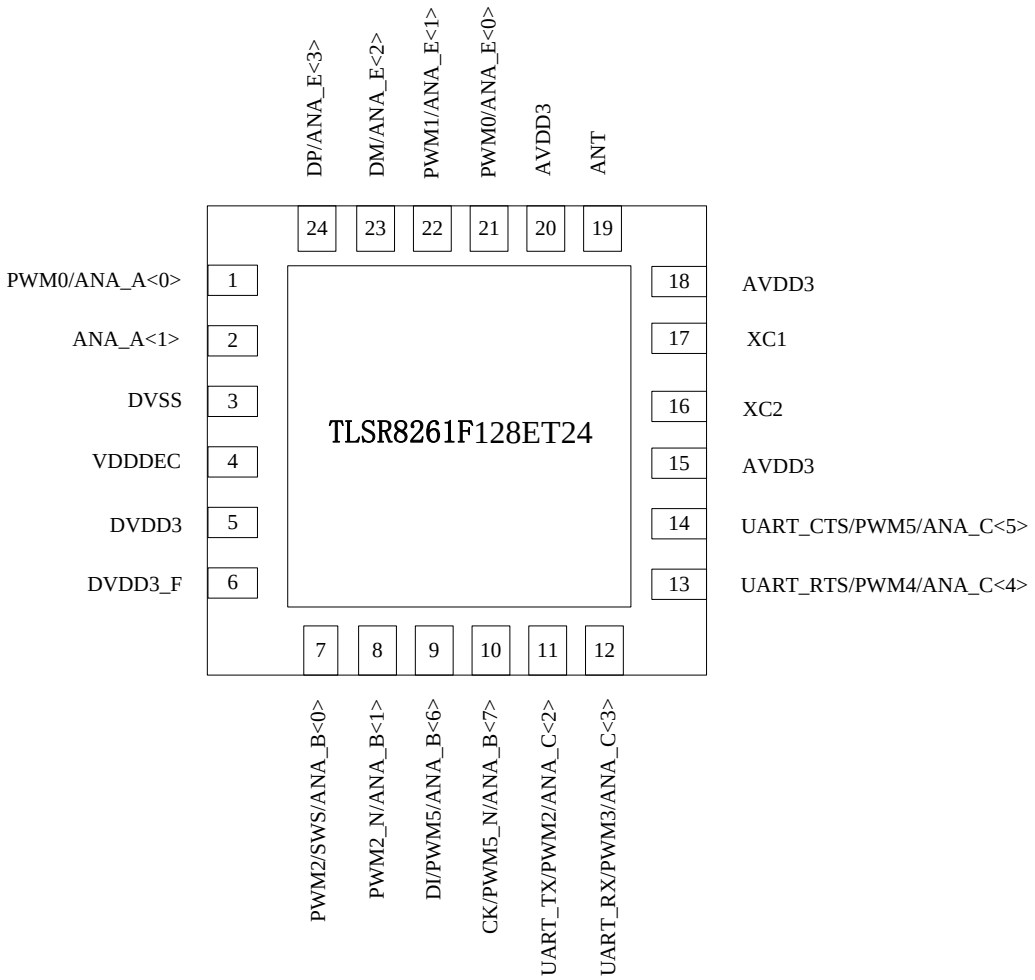


Figure 1- 3 Pin assignment for the TLSR8261F128ET24

Functions of 24 pins for the TLSR8261F128ET24 are described in Table 1- 2:

Table 1- 2 Pin functions for the TLSR8261F128ET24

No.	Pin Name	Type	Description
1	**PWM0/ANA_A<0>**	Digital I/O	PWM0/GPIO/ANA_A<0>
2	**ANA_A<1>**	Digital I/O	GPIO/ANA_A<1>
3	DVSS	GND	Digital LDO ground
4	VDDDEC	PWR	Digital LDO 1.8V output
5	DVDD3	PWR	3.3V IO supply
6	DVDD3_F	PWR	Connect external 10uF Capacitor
7	**PWM2/SWS/ANA_B<0>**	Digital I/O	PWM2 output/Single wire slave/GPIO/ ANA_B<0>
8	**PWM2_N/ANA_B<1>**	Digital I/O	PWM2 inverting output /GPIO/ANA_B<1>
9	**DI/PWM5/ANA_B<6>**	Digital I/O	I2C_SDA (I2C serial data) / PWM5 output / GPIO/ ANA_B<6>/
10	**CK/PWM5_N/ANA_B<7>**	Digital I/O	I2C_SCK (I2C serial clock) / PWM5 inverting output/ GPIO/ ANA_B<7>/
11	**UART_TX/PWM2/ANA_C<2>**	Digital I/O	UART_TX/PWM2 output/ GPIO/ANA_C<2>/ (optional) 32KHz crystal output
12	**UART_RX/PWM3/ANA_C<3>**	Digital I/O	UART_RX/PWM3 output/ GPIO /ANA_C<3>/(optional) 32KHz crystal input
13	**UART_RTS/PWM4/ANA_C<4>**	Digital I/O	UAR_RTS/PWM4 output/ GPIO /ANA_C<4>
14	**UART_CTS/PWM5/ANA_C<5>**	Digital I/O	UART_CTS/PWM5 output/ GPIO /ANA_C<5>
15	AVDD3	PWR	Analog 3.3V supply
16	XC2	Analog O	12MHz/16MHz crystal output
17	XC1	Analog I	12MHz/16MHz crystal input
18	AVDD3	PWR	Analog 3.3V supply
19	ANT	Analog O	RF antenna
20	AVDD3	PWR	Analog 3.3V supply
21	**PWM0/ANA_E<0>**	Digital I/O	PWM0 output/GPIO/ANA_E<0>
22	**PWM1/ANA_E<1>**	Digital I/O	PWM1 output/GPIO/ANA_E<1>
23	**DM/ANA_E<2>**	Digital I/O	USB data Minus/GPIO/ ANA_E<2>
24	**DP/ANA_E<3>**	Digital I/O	USB data Positive/GPIO/ ANA_E<3>

1) Pins with bold typeface can be used as GPIOS. All pins from ANA_A<0>~ANA_E<3> have configurable pull-up/pull-down resistor.

2) Pin drive strength: Please refer to section 7.1 and 11.5 for the corresponding “DS” register setting and actual test results.

1.7 Telink SDK

A full featured SDK is provided with the chip for Bluetooth Low Energy applications. The customers can easily develop rich BLE applications by employing the firmware, along with the system configuration data composed according to the specific hardware design.

2 Memory and MCU

2.1 Memory

The TLSR8261F128 embeds 16KB data memory (SRAM) and 128KB internal FLASH.

2.1.1 SRAM/Register

SRAM/Register memory map is shown as follows:

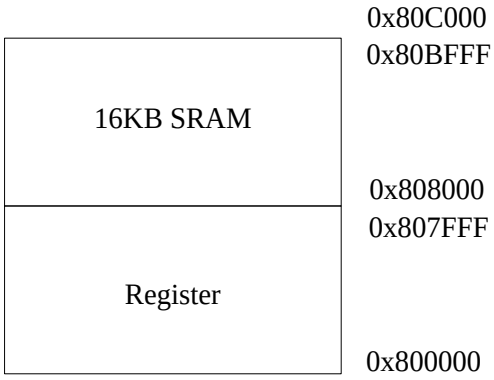


Figure 2- 1 Physical memory map

Register address: from 0x800000 to 0x807FFF;

16KB SRAM address: from 0x808000 to 0x80C000.

Both register and 16KB SRAM address can be accessed via I2C and SWS interface.

2.1.2 Flash

The internal Flash mainly supports page program, sector/block/chip erase operations, and deep power down operation.

2.1.2.1 Page program

The page program mode allows up to 256 bytes data to be programmed at memory locations that have been erased.

2.1.2.2 Sector erase

The sector erase operation serves to erase all the data of the specified sector (4KB) to all 1s.

2.1.2.3 Block erase

The block erase operation serves to erase all the data of the specified block (32KB or 64KB) to all 1s.

2.1.2.4 Chip erase

The chip erase operation serves to erase data at all memory locations to all 1s.

2.1.2.5 Unique ID

For chip identification and traceability, the 128KB Flash is preloaded with 8-byte hexadecimal Unique ID (UID). User is not allowed to modify this preloaded UID with write protection, but can read the UID via corresponding API interface.

2.2 MCU

The TLSR8261F128 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 TLSR8261F128 has four working modes: Active, Idle, Suspend and Deep Sleep. This section mainly gives the description of every working mode and mode transition.

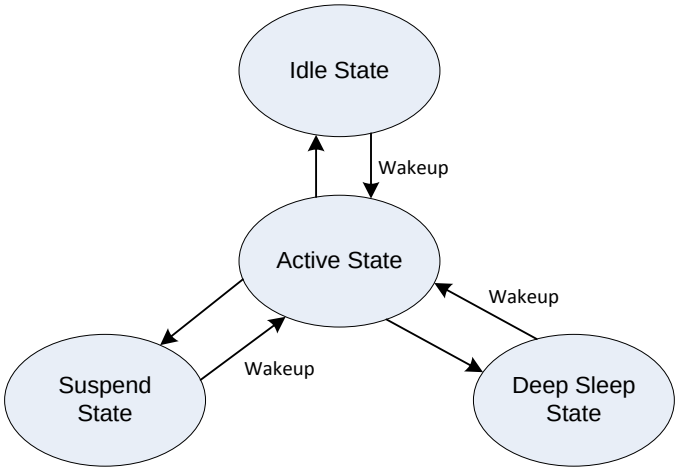


Figure 2- 2 Transition chart of working modes

2.3.1 Active mode

In active mode, the MCU block is at working state, and the TLSR8261F128 can transmit or receive data via its embedded RF transceiver. The RF transceiver can also be powered down if no data transfer is needed.

2.3.2 Idle mode

In Idle mode, the MCU block stalls, and the RF transceiver can be at working state or be powered down. The time needed for the transition from Idle mode to Active mode is negligible.

2.3.3 Power-saving mode

For the TLSR8261F128, there are two kinds of power-saving modes: suspend mode and deep sleep mode. The two modes have similar transition sequences but different register settings. For 1.8V digital core, it's still provided with the working power by 1.8V LDO in suspend mode; while in deep sleep mode, the 1.8V LDO will be turned off, and the digital core is powered down.

In suspend mode, the RF transceiver is powered down, and the clock of the MCU block is stopped. It only takes about 400us for the TLSR8261F128 to enter the active mode from suspend mode.

While in deep sleep mode, both the RF transceiver and the MCU block are powered down with only power management block being active. The transition time needed from deep sleep mode to active mode is 1ms, almost the same as power-up time.

Table 2- 1 Retention analog registers in deep sleep

Address	Description
0x34~0x39	buffer, watch dog reset clean
0x3a~0x3b	buffer, power on reset clean
0x3c	buffer, power on reset clean
0x3d	
0x3e	

Analog registers (0x34 ~ 0x3e) as shown in Table 2- 1 are retained in deep sleep mode and can be used to store program state information across deep sleep cycles.

- ✧ Analog registers 0x3a~0x3e 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 0x34~0x39 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 retention analog registers 0x3a~0x3e 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] to 1b'1 is to reset the whole chip. Similar to watchdog reset (see **section 2.3.3 Power-saving mode**), retention analog registers 0x3a~0x3e are non-volatile, while other registers including 0x34~0x39 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- 2 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]: USB [3]: rsvd [4]: MCU [5]: mac [6]: AIF [7]: zb	00
0x61	RST1	R/W	[0] system_timer [1]algm [2]dma [3]rs232 [4]pwm0 [5]aes	df

Address	Mnemonic	Type	Description	Reset Value
			[6]bbpll48m [7]swires	
0x62	RST2	R/W	[0]sbc [1]audio [2]dfifo [3]adc [4]mcic [5]soft reset to reset mcic enable [6]rsvd (mspi) [7] algs	00
0x6f	PWDNEN	W	[0]: suspend enable [5]: rst all (act as watchdog reset) [6]: rsvd (mcu low power mode) [7]: stall mcu trig If bit[0] set 1, then system will go to suspend. Or only stall mcu	

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

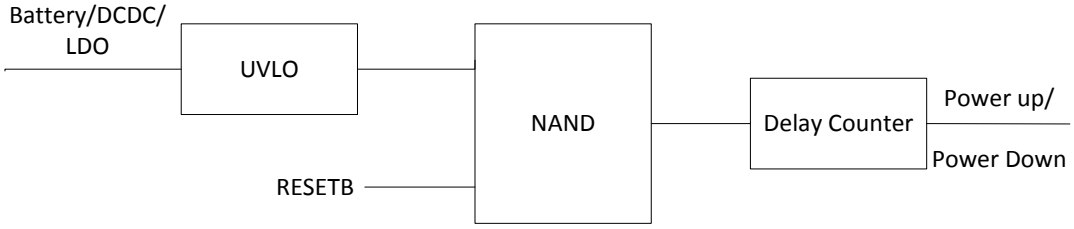


Figure 2- 3 Block diagram for power up/down

The whole chip power up and down is controlled by the UVLO (Ultra-low Voltage Lockout) 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 further configurable delay before the system is released. This delay is adjusted by analog register 0x20. Since the content of 0x20 is reset to default only after power cycle, watchdog reset, or software reset, the delay change using 0x20 is only applicable when the chip has not gone through these reset conditions. For example, after deep sleep wakeup, the setting in 0x20 will take effect.

Table 2- 3 afe 3V analog register to control delay counter

Address	Description	Default
0x20	r_dly: [7]:1: old delay mode, 0: waiting for xtal ready. [6:0]: delay, bit[0] is swapped with bit[6]. (32KHz counter). if r_dly[7] is 1, the following is the real delay: 7'b000_0000 4ms~33*0us delay 7'b100_0000 4ms~33*1us delay 7'b000_0010 4ms~33*2us delay 7'b100_0010 4ms~33*3us delay 7'b000_0001 4ms~ 2ms delay 7'b111_1110 1ms delay (32 cycles of 32K Hz) if r_dly[7] is 0, the real delay is xtal ready delay plus counter delay.	0xb0

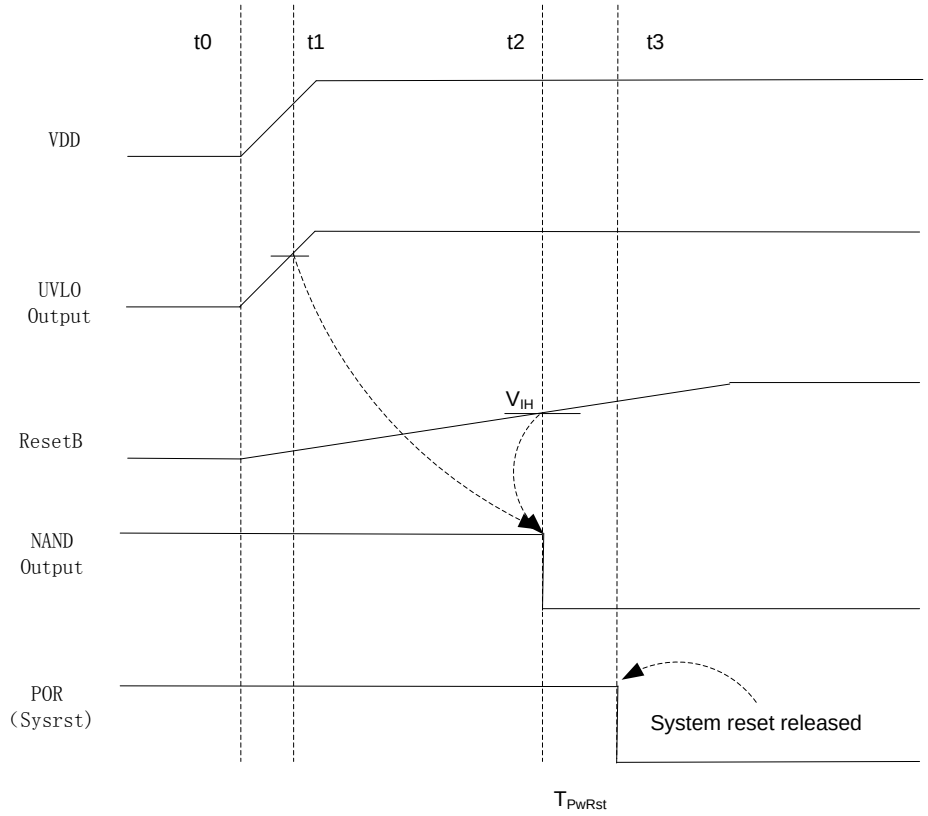


Figure 2- 4 Power-up sequence

Power down

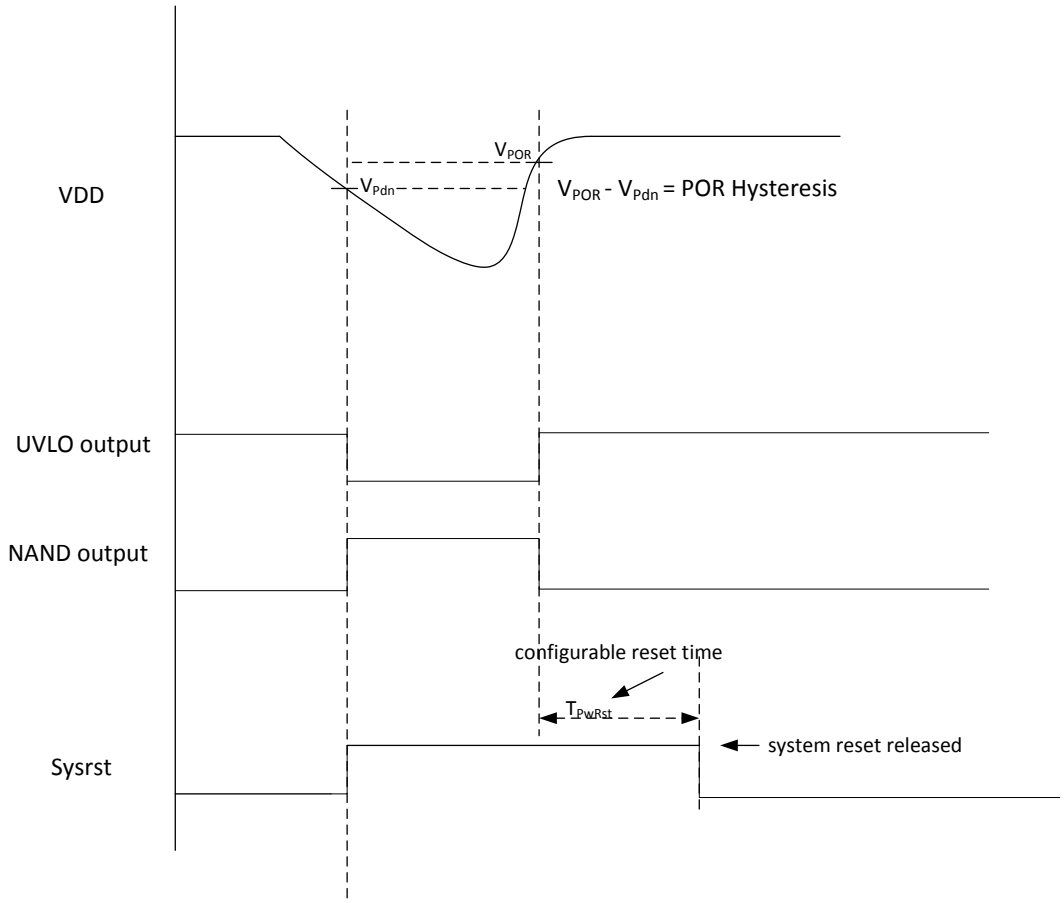


Figure 2- 5 Power-down sequence

The power up and power down sequence is shown in Table 11- 4, with the following parameters:

- V_{POR} : VDD voltage when V_{UVLO} turns to high level
- V_{Pdn} : VDD voltage when V_{UVLO} turns to low level
- T_{PwRst} : Delay counter value (Configurable via analog register 0x20)

2.5.2 Working mode switch

The chip can switch to idle mode to stall the MCU.

To minimize power consumption, the chip can switch to power saving mode (suspend or deep sleep) correspondingly. In this case, the low-power 32KHz RC oscillator is still running, and the low frequency wakeup timer LTIMER can be programmed to stay alive. The device can be activated to working state via external pin trigger or internal wakeup timer.

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- 4 3.3V analog registers for module power up/down control

Address	Local name	Default Value	Description
afe3V_reg05<0>	32K_rc_pd	0	Power down 32KHz RC oscillator 1: Power down 32KHz RC oscillator 0: Power up 32KHz RC oscillator
afe3V_reg05<1>	32k_xtal_pd	0	Power down 32k crystal 1: power down 0: power up
afe3V_reg05<2>	32M_rc_pd	0	Power down of 32MHz RC oscillator 1: Power down 32MHz RC oscillator 0: Power up 32MHz RC oscillator
afe3V_reg05<3>	xtal_LDO_pd	0	Power down of 16MHz crystal oscillator 1: Power down 0: Power up
afe3V_reg05<4>	ldo_ana_pd	0	Power down of analog LDO 1: Power down 0: Power up
afe3V_reg05<5>	pm_pd_tempsen_3V	1	Power down master Bandgap 1: Power down 0: Power up

Address	Local name	Default Value	Description
afe3V_reg05<6>	reserved	1	Power down off-chip resistor bias 1: Power down 0: Power up
afe3V_reg05<7>	BBPLL_LDO_pd_3V	1	Power down baseband pll LDO 1: Power down 0: Power up
afe3V_reg06<0>	saradc_pd	1	Power down SAR ADC 1: Power down 0: Power up
afe3V_reg06<1>	rx_lnaLDO_pd	1	Power down LNA LDO in RF transceiver 1: Power down 0: Power up
afe3V_reg06<2>	rx_anaLDO_pd	1	Power down analog LDO in RF transceiver 1: Power down 0: Power up
afe3V_reg06<3>	rx_rfLDO_pd	1	Power down RF LDO in RF transceiver 1: Power down 0: Power up
afe3V_reg06<4>	pll_BG_pd	1	Power down Bandgap in PLL 1: Power down 0: Power up
afe3V_reg06<5>	reserved		
afe3V_reg06<6>	pll_vco_ldo_pd	1	Power down VCO LDO 1: Power down 0: Power up
afe3V_reg06<7>	pll_cp_ldo_pd	1	Power down cp and prescaler analog circuit ldo 1: Power down 0: power up

2.5.3 LDO and DCDC

The chip embeds LDO regulators to generate 1.8V regulated voltage. The internal LDO regulators serve to supply power for 1.8V digital core and analog modules in Active/Idle/Suspend mode.

The chip also embeds a boost DCDC which can step up input voltage to the range of 2.7~3.6V. The DCDC output serves to supply power for flash. For internal flash, the DCDC directly supplies power for it; while for external flash, the DCDC supplies power via the DVDD3F pin of the chip.

While in deep sleep mode, the embedded 1.8V LDO regulators and the boost DCDC will be turned off.

2.6 Wakeup sources

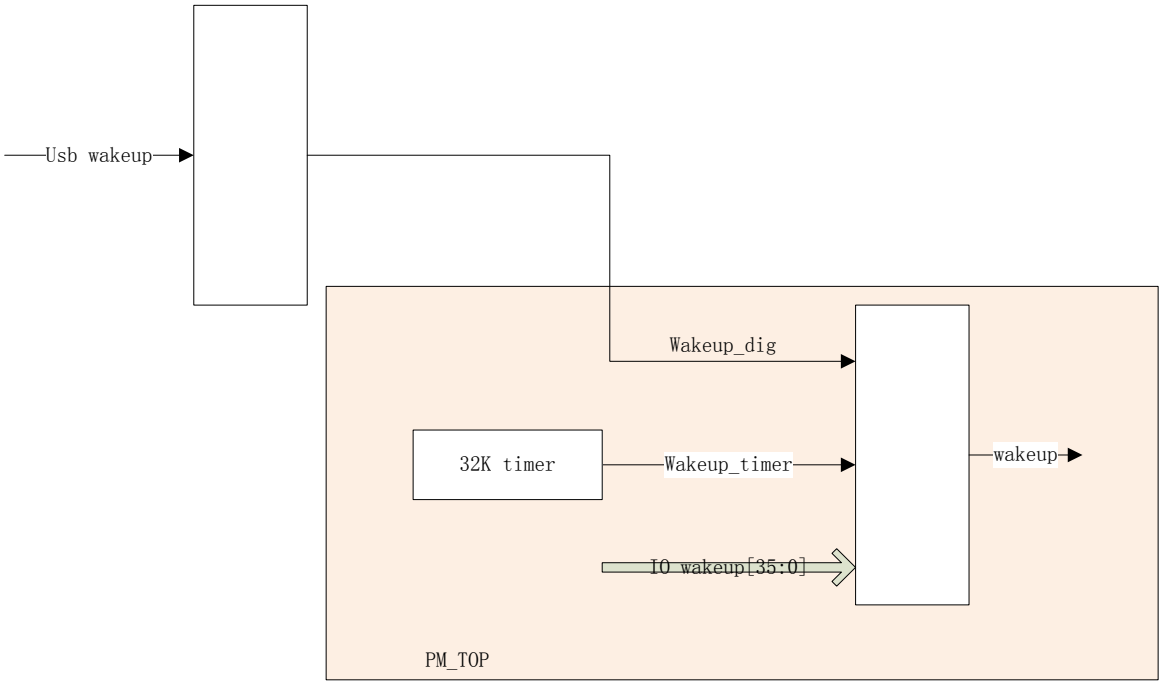


Figure 2- 6 Wakeup sources

2.6.1 Wakeup source - USB

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

First, set the digital core address 0x6e bit [2] to 1.

To activate this mode, 3V_reg38 bit[5] should also be set to 1.

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

2.6.2 Wakeup source – 32K timer

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

Address 3V_reg38 bit[6] is the enabling bit for wakeup source from 32k timer.

2.6.3 Wakeup source – IO

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

3v_reg38[4] should be set to 1b'1 to enable IO wakeup source.

Enabling control registers: PA[7:0] enabling control register is 3V_reg39[7:0], PB[7:0] enabling control register is 3V_reg40[7:0], PC[7:0] enabling control register is 3V_reg41[7:0], PD[7:0] enabling control register is 3V_reg42[7:0], and PE[3:0] enabling control register is 3V_reg43[3:0]. Total wakeup pin can be up to 36.

Polarity control registers: PA[7:0] polarity control register is 3V_reg33[7:0], PB[7:0] polarity control register is 3V_reg34[7:0], PC[7:0] polarity control register is 3V_reg35[7:0], PD[7:0] polarity control register is 3V_reg36[7:0], and PE[3:0] polarity control register is 3V_reg37[3:0].

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

2.6.4 Register table

Table 2- 5 Analog registers for Wakeup

Address Dec	Address Hex	Description	Default Value
r33	0x21	pa_pol	0x00
r34	0x22	pb_pol	0x00
r35	0x23	pc_pol	0x00
r36	0x24	pd_pol	0x00
r37[3:0]	0x25[3:0]	pe_pol[3:0]	0x00
r38[4]	0x26[4]	wkup from IO (pad)	0x00
r38[5]	0x26[5]	wkup dig (including usb)	0x00
r38[6]	0x26[6]	wkup 32k timer	0x00
r38[7]	0x26[7]	rsvd (wkup comparator)	0x00
r39	0x27	wkup_pa_en	0x00
r40	0x28	wkup_pb_en	0x00
r41	0x29	wkup_pc_en	0x00
r42	0x2a	wkup_pd_en	0x00
r43[3:0]	0x2b[3:0]	wkup_pe_en[3:0]	0x00
r68	0x44	State flag bits [0]:rsvd (wkup cmp) [1]: pm_irq (i.e. wkup_32k timer) [2]:wkup _dig [3] wkup_pad e.g. If bit[3] is 1, it indicates the system is wakened up by IO (pad) source. Write 1 to clean	0x00

Table 2- 6 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 to1, system will issue USB resume signal on USB bus [7]: sleep wakeup reset system enable	00

3 BLE/2.4G RF Transceiver

3.1 Block diagram

The TLSR8261F128 integrates an advanced BLE/2.4GHz RF transceiver. The RF transceiver works in the worldwide 2.4GHz ISM (Industrial Scientific Medical) band and contains an integrated balun with a single-ended RF Tx/Rx port pin. No matching components are needed.

The transceiver consists of a fully integrated frequency synthesizer, a power amplifier, a modulator and a receiver. The transceiver can be configured to work in standard-compliant 1Mbps BLE mode and 2Mbps enhancement BLE mode. All modes support FSK/GFSK/MSK modulations.

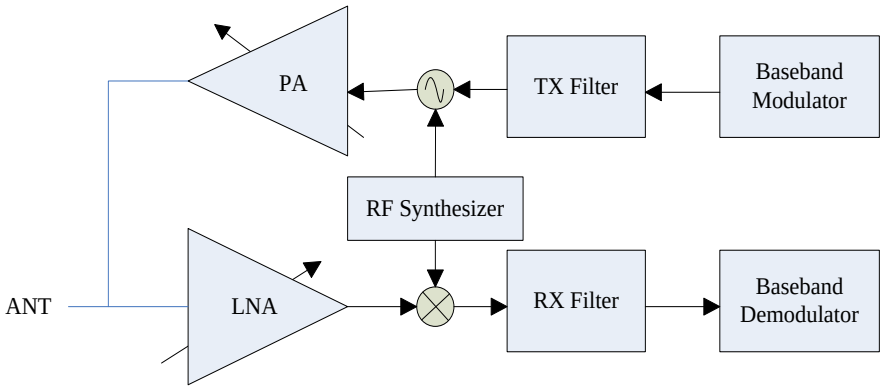


Figure 3- 1 Block diagram of RF transceiver

The internal PA can deliver a maximum 8dBm output power, avoiding the needs for an external RF PA.

3.2 Function description

3.2.1 Turn on/off

For the sake of saving power, the transceiver can be turned on/off via the software. Setting the address 0x7c bit[6] to 1 enables the RF transceiver, while clearing the bit totally disables the RF transceiver.

3.2.2 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: 1Mbps, 2Mbps.

For the TLSR8261F128, RF transceiver can operate with frequency ranging from 2.400GHz to 2.4835GHz. The RF channel frequency setting determines the center of the channel.

3.3 Baseband

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 Bluetooth V4.2 specification.

3.3.1 Packet format

Packet format in standard 1Mbps BLE mode is shown as Table 3-1:

Table 3- 1 Packet Format in standard 1Mbps BLE mode

LSB MSB

Preamble (1 octet)	Access Address (4 octets)	PDU (2 to 257 octets)	CRC (3 octets)

Packet length 80bit ~ 2120bit (80~2120us @ 1Mbps).

3.3.2 RSSI

The TLSR8261F128 provides accurate RSSI (Receiver Signal Strength Indicator) indication which can be read on per packet basis.

4 Clock

4.1 Clock sources

The TLSR8261F128 embeds a 32MHz RC oscillator which can be used as clock source for system and ADC. A 32KHz RC oscillator is also embedded to provide clock source for sleep state.

Other than the RC clock source, PLL generates a 192MHz clock source, which can be used as clock sources for system and ADC.

External 12M/16M crystal is available via pin XC1, which can provide a 12MHz/16MHz clock source for system and ADC. External 32K crystal is available via pin ANA_C<3>, which can provide a 32KHz clock source for system.

4.2 Register table

Table 4- 1 Register table for clock

Address	Mnemonic	Type	Description	Reset Value
0x63	CLKEN0	R/W	Clock enable control: 1 for enable; 0 for disable [0] : SPI [1] : I2C [2]: USB [3]: USB PHY [4]: MCU [5]: mac [6]: AIF [7]: zb	8c
0x64	CLKEN1	R/W	[0]system timer [1]algm [2]dma [3]rs232 [4]pwm0 [5]aes [6]clk32k for system timer [7]swires	00
0x65	CLKEN2	R/W	[0]32k for qdec [1]audio [2]dfifo	00

Address	Mnemonic	Type	Description	Reset Value
			[3]rsvd (key scan) [4]mcic [5]qdec [6]32k for pwm [7]rsvd (32k for keyscan)	
0x66	CLKSEL	R/W	System clock select [4:0]: system clock divider (must exceed 1): If 0x66[6:5] is set as 2b'01, $F_{Sysclk} = F_{FHS}$ / (CLKSEL[4:0]). Fhs refers to {0x70[0], 0x66[7]} FHS_sel [6:5] 2'b00:32m clock from rc 2'b01:hs divider clk 2'b10:16M clock from pad 2'b11:32k clk from pad {0x70[0], 0x66[7]}: FHS sel	ff
0x67	I2S step	R/W	Reserved	33
0x68	I2S Mod	R/W	Reserved	2
0x69	Adc step[7:0]	R/W	ADC clock step[7:0]	00
0x6a	Adc mod[7:0]	R/W	Adc clock mod[7:0]	2
0x6b	adcmodstep	R/W	[7]: adc clock enable [6:4] :adc step[10:8] [3:0] adc mod[11:8] Adc clock = fhs * step[10:0]/mod[11:0] Mod needs to be larger than or equal to 2*step Fhs refers to {0x70[0], 0x66[7]} FHS_sel	00
0x6c	DMIC_step	R/W	rsvd	1
0x6d	DMIC_mod	R/W	rsvd	2
0x70	FHS_sel	R/W	{0x70[0], 0x66[7]}: fhs select 2'b00: 192M clock from pll 2'b01: 32M clock from rc osc 2'b1x: 16M clock from pad	00
0x71	DC/DC clk mod	R/W	Reserved	
0x73	Clk mux sel		[0]: clk32k select;0:sel 32k osc 1: 32k pad [1]dmic clock select, 1:select 32k (refer to bit[0] to decide which 32k ; 0:dmic clk div [2] usb phy clock select,1 : 192M divider 0:48M pll [7:4] r_lpr_div, decide system clock speed in low power mode	0x14

4.3 System clock

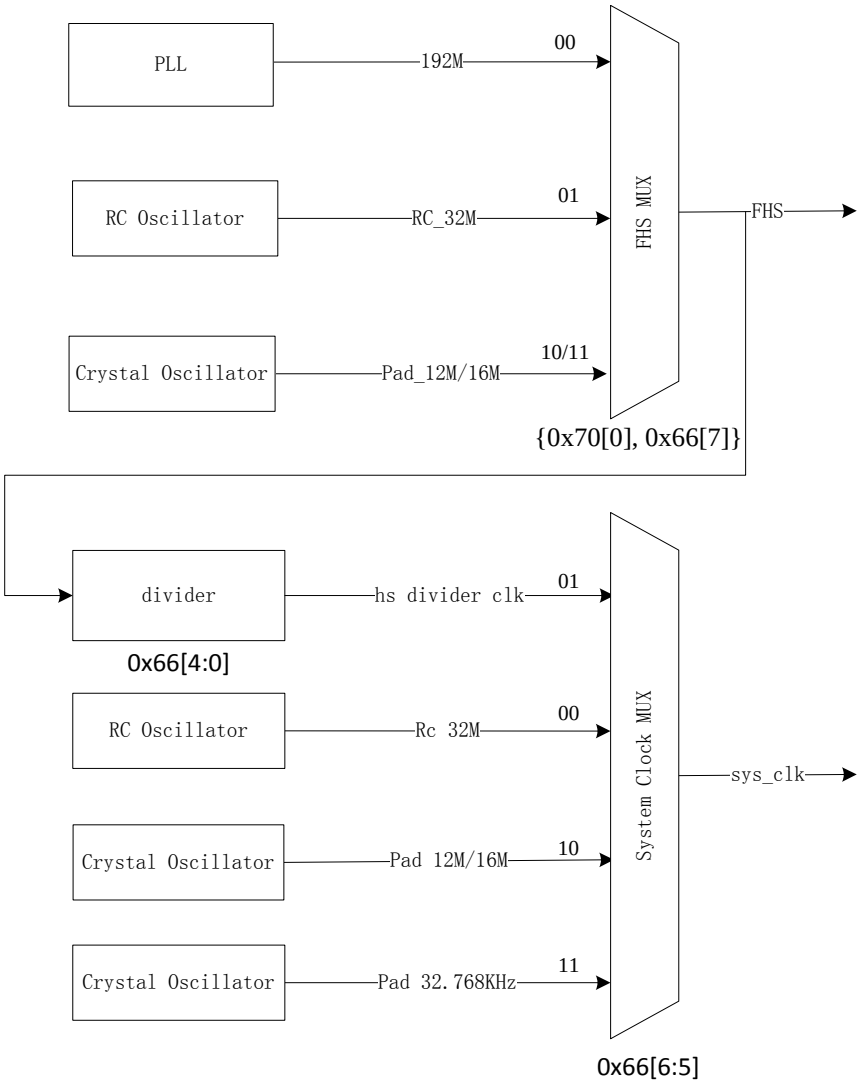


Figure 4- 1 Block diagram of system clock

There are four selectable clock sources for MCU system clock: 32MHz RC clock, HS divider clock (divided from a High speed clock), and Pad clock (12MHz/16MHz, 32.768KHz).

The high speed clock (FHS) is selectable via address {0x70[0], 0x66[7]} from the following sources: 192MHz clock from PLL, 32MHz RC clock, and 12MHz/16MHz Pad clock.

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'01 to select the HS divider clock, system clock frequency is adjustable via address 0x66[4:0].

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

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

4.4 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.

4.4.1 ADC clock

ADC clock derives from FHS. ADC clock is enabled via setting address 0x6b[7] to 1b'1.

ADC clock frequency dividing factor contains step and mod. Address 0x6b[6:4] and 0x69 serve to set ADC clock step[10:0]. Address 0x6b[3:0] and 0x6a serve to set ADC clock mod[11:0].

ADC clock frequency, $F_{ADC\ clock}$, equals to F_{FHS}* step[10:0] / mod[11:0].

5 Timers

5.1 Timer0~Timer2

The TLSR8261F128 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).

Timer 2 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, write 1 to clear. [7:1] rsvd	
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
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	

Address	Mnemonic	Type	Description	Reset Value
			[3] rsvd	
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	
0x631	TMR_TICK0_1	RW	Byte 1 of timer0 ticker	
0x632	TMR_TICK0_2	RW	Byte 2 of timer0 ticker	
0x633	TMR_TICK0_3	RW	Byte 3 of timer0 ticker	
0x634	TMR_TICK1_0	RW	Byte 0 of timer1 ticker	
0x635	TMR_TICK1_1	RW	Byte 1 of timer1 ticker	
0x636	TMR_TICK1_2	RW	Byte 2 of timer1 ticker	
0x637	TMR_TICK1_3	RW	Byte 3 of timer1 ticker	
0x638	TMR_TICK2_0	RW	Byte 0 of timer2 ticker	
0x639	TMR_TICK2_1	RW	Byte 1 of timer2 ticker	
0x63a	TMR_TICK2_2	RW	Byte 2 of timer2 ticker	
0x63b	TMR_TICK2_3	RW	Byte 3 of timer2 ticker	

5.1.2 Mode0 (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 Mode1 (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 Mode2 (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 set to the recommended value of 0, then:

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

5.1.5 Mode3 (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 32bits. 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 TLSR8261F128 also supports a low frequency (32KHz) LTIMER in suspend mode or deep sleep mode. This timer can be used as one kind of wakeup source.

5.3 System Timer

The TLSR8261F128 also supports a System Timer.

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_timer[7:0]	R/W		00
0x741	Sys_timer[15:8]	R/W		00
0x742	Sys_timer[23:16]	R/W		00

Address	Mnemonic	R/W	Function	Default Value
0x743	Sys_timer[31:24]	R/W	System timer counter, write to set initial value. This is the sys timer counter	00
0x74c	Sys_timer_ctrl	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

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 TLSR8261F128, 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 decide 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_host_cmd,irq_uart,irq_ks, irq_dma,usb_pwdn,time2,time1,time0} [7] irq_host_cmd [6] irq_uart [5] rsvd (irq_ks) [4] irq_dma [3] usb_pwdn [2] time2 [1] time1 [0] time0	00
0x641	MASK_1	RW	Byte 1 interrupt mask, level-triggered type {an_irq,irq_software|irq_pwm,irq_zb_rt,irq _udc[4:0]} [7] an_irq [6] irq_software|irq_pwm [5] irq_zb_rt	00

Address	Mnemonic	Type	Description	Reset Value
			[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 {gpio2risc[2:0],irq_stimer,pm_irq,irq_gpio,usb_reset,usb_250us} [7] gpio2risc[2] [6] gpio2risc[1] [5] gpio2risc[0] [4] irq_stimer [3] pm_irq [2] irq_gpio [1] usb_reset [0] usb_250us	00
0x643	IRQMODE	RW	[0] interrupt enable [1] reserved (Multi-Address enable)	00
0x644	PRIO_0	RW	Byte 0 of priority 1: High priority; 0: Low priority	00
0x645	PRIO_1	RW	Byte 1 of priority	00
0x646	PRIO_2	RW	Byte 2 of priority	00
0x648	IRQSRC_0	R	Byte 0 of interrupt source	
0x649	IRQSRC_1	R	Byte 1 of interrupt source	
0x64a	IRQSRC_2	R	Byte 2 of interrupt source	

6.2.1 Enable/Mask interrupt sources

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

6.2.2 Interrupt mode and priority

Interrupt mode is typically-used mode. Register IRQMODE (address 0x643)[0] should be set to 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 raised to "High". 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] to 1; then set address 0x643 bit[0] to 1 to enable the interrupt. In interrupt handling function, 24-bit data is read from address 0x648~0x64a to determine which IRQ source is valid; if data bit[16] is 1, it means the usb_250us interrupt is valid. Clear this interrupt source by setting address 0x64a bit[0] to 1.

As for level-type interrupt, IRQ interrupt source status needs to be cleared via setting corresponding module status register. Take Timer0 IRQ interrupt source for example, register TMR_STATUS (address 0x623) [0] should be written with 1b'1 to clear Timer0 status (refer to section 5.1.1).

7 Interface

7.1 GPIO

The TLSR8261F128 supports up to 14 GPIOs. Except for dedicated GPIOs, all digital IOs can be used as general purpose IOs.

All GPIOs have configurable pull-up/pull-down resistor. Please refer to **section 7.1.3** for details.

7.1.1 Basic configuration

Please refer to Table 7-1 in section 7.1.1.3 for various GPIO interface configuration.

7.1.1.1 Multiplexed functions

For a pin listed in Table 7-1, it acts as the function in the “Default Function” column by default.

If a pin with multiplexed functions does not act as GPIO function by default, to use it as GPIO function, first set the bit in “Act as GPIO” column to 1b’1. After GPIO function is enabled, if the pin is used as output, both the bits in “IE” and “OEN” columns should be cleared, then set the register value in the “Output” column; if the pin is used as input, both the bits in “IE” and “OEN” columns set to 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, and then configure register in “Pad Function Mux” column.

Take the PWM0/ANA_A<0> pin for example.

(1) The pin acts as GPIO function by default. If the pin is used as general output, both address 0x581[0] and 0x582[0] should be cleared, then configure address 0x583[0]. If the pin is used as general input, both address 0x581[0] and 0x582[0] should be set to 1b’1, and the input data can be read from address 0x580[0].

(2) Addresses 0x586[0] and 0x5b0[0] should be cleared to use the pin as PWM0 function.

Take the PWM2/SWS/ANA_B<0> pin as another example.

(1) The pin acts as SWS function by default.

(2) To use it as GPIO function, first set address 0x58e[0] to 1b'1. If the pin is used as general output, both address 0x589[0] and 0x58a[0] should be cleared, then configure address 0x58b[0]. If the pin is used as general input, both address 0x589[0] and 0x58a[0] should be set to 1b'1, and the input data can be read from address 0x588[0].

(3) To use it as PWM2 function, clear address 0x58e[0], and set 0x5b1[0] to 1b'1.

7.1.1.2 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.

All the pins support maximum drive level of 4mA ("DS"=1) and minimal drive level of 0.7mA ("DS"=0) with the following exceptions:

- ANA_E<1> and ANA_E<0>: maximum=16mA ("DS"=1), minimum=12mA ("DS"=0)
- ANA_E<3> and ANA_E<2>: maximum=12mA ("DS"=1), minimum=8mA ("DS"=0)

7.1.1.3 GPIO lookup table

Table 7- 1 GPIO lookup table

Pin	Default function	Pad Function Mux		GPIO Setting						
		bit = 1	bit = 0	Input (R)	IE	OEN	Output	Polarity	DS	Act as GPIO
PWM0/ANA_A<0>	GPIO		5b0[0] PWM0	0x580[0]	0x581[0]	0x582[0]	0x583[0]	0x584[0]	0x585[0]	0x586[0]
ANA_A<1>	GPIO			0x580[1]	0x581[1]	0x582[1]	0x583[1]	0x584[1]	0x585[1]	0x586[1]
PWM2/SWS/ ANA_B<0>	SWS	0x5b1[0]PWM2	0x5b1[0]SWS	0x588[0]	0x589[0]	0x58a[0]	0x58b[0]	0x58c[0]	0x58d[0]	0x58e[0]
PWM2_N/ ANA_B<1>	GPIO	------	0x5b1[1]PWM2_N	0x588[1]	0x589[1]	0x58a[1]	0x58b[1]	0x58c[1]	0x58d[1]	0x58e[1]
DI/PWM5/ ANA_B<6>	DI	0x5b1[6]DI	0x5b1[6]PWM5	0x588[6]	0x589[6]	0x58a[6]	0x58b[6]	0x58c[6]	0x58d[6]	0x58e[6]
CK/PWM5_N/ ANA_B<7>	CK	0x5b1[7]CK	0x5b1[7]PWM5_N	0x588[7]	0x589[7]	0x58a[7]	0x58b[7]	0x58c[7]	0x58d[7]	0x58e[7]
UART_TX/PWM2/ ANA_C<2>	GPIO	0x5b2[2]UART_TX	0x5b2[2]PWM2	0x590[2]	0x591[2]	0x592[2]	0x593[2]	0x594[2]	0x595[2]	0x596[2]
UART_RX/PWM3/ ANA_C<3>	GPIO	0x5b2[3]UART_RX	0x5b2[3]PWM3	0x590[3]	0x591[3]	0x592[3]	0x593[3]	0x594[3]	0x595[3]	0x596[3]
UAR_RTS/PWM4/ ANA_C<4>	GPIO	0x5b2[4]UART_RTS	0x5b2[4]PWM4	0x590[4]	0x591[4]	0x592[4]	0x593[4]	0x594[4]	0x595[4]	0x596[4]
UART_CTS/PWM5/ ANA_C<5>	GPIO	0x5b2[5]UART_CTS	0x5b2[5]PWM5	0x590[5]	0x591[5]	0x592[5]	0x593[5]	0x594[5]	0x595[5]	0x596[5]
PWM0/ ANA_E<0>	GPIO	0x5b4[0]PWM0		0x5a0[0]	0x5a1[0]	0x5a2[0]	0x5a3[0]	0x5a4[0]	0x5a5[0]	0x5a6[0]
PWM1/ ANA_E<1>	GPIO	0x5b4[1]PWM1		0x5a0[1]	0x5a1[1]	0x5a2[1]	0x5a3[1]	0x5a4[1]	0x5a5[1]	0x5a6[1]
DM/ANA_E<2>	DM		DM	0x5a0[2]	0x5a1[2]	0x5a2[2]	0x5a3[2]	0x5a4[2]	0x5a5[2]	0x5a6[2]
DP/ANA_E<3>	DP		DP	0x5a0[3]	0x5a1[3]	0x5a2[3]	0x5a3[3]	0x5a4[3]	0x5a5[3]	0x5a6[3]

*Notes:

(1) IE: Input enable, high active;

(2) OEN: Output enable, low active;

(3) Priority: “Act as GPIO” has the highest priority;

(4) For all unused GPIOs, corresponding “IE” must be set as 0;

(5) When SWS/ANA_B<0> “IE” is set as 1, this pin must be fixed as pull-up/pull-down state (float state is not allowed).

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 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);

GPIO2RISC[2] interrupt request signal = | ((input ^ polarity) & m2);

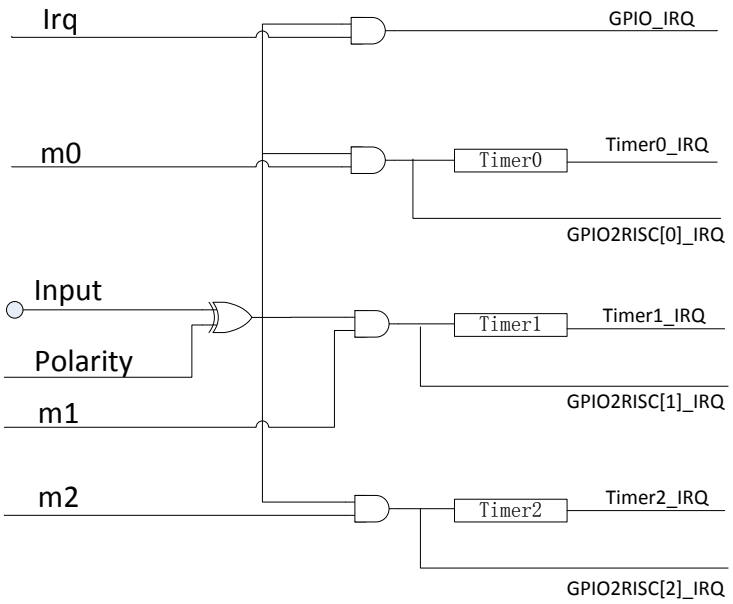


Figure 7- 1 Logic relationship between GPIO and related modules

Please refer to Table 7-2 and Table 6- 1 to learn how to configure GPIO for interrupt system or Timer/Counter (Mode 1 or Mode 2).

(1) First enable GPIO function, IE and disable OEN.

(2) 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] to enable GPIO IRQ. Finally enable GPIO interrupt (irq_gpio at address 0x642[2]). User can read addresses 0x5e0 ~ 0x5e4 to see which GPIO asserts GPIO interrupt request signal. Note: 0x5e0[1:0] --> ANA_A<1,0>, 0x5e1[7,6,1,0] --> ANA_B<7,6,1,0>, 0x5e2[5:2] -->ANA_C<5>~ANA_C<2>, 0x5e4[3:0] --> ANA_E<3>~ANA_E<0>.

(3) Timer/Counter counting or control signal: Configure “Polarity” (In Mode 1， it determines GPIO edge when Timer Tick counting increases; in Mode 2, it determines GPIO edge when Timer Tick starts counting) and set “m0/m1/m2”. User can read addresses 0x5e8~0x5ec/0x5f0~0x5f4/0x5f8~0x5fc to see which GPIO asserts counting signal (in Mode 1) or control signal (in Mode 2) for Timer0/Timer1/Timer2. Note: Timer0: 0x5e8[1:0] --> ANA_A<1,0>, 0x5e9[7,6,1,0] --> ANA_B<7,6,1,0>, 0x5ea[5:2] --> ANA_C<5>~ANA_C<2>, 0x5ec[3:0] --> ANA_E<3>~ANA_E<0>; Timer1: 0x5f0[1:0] --> ANA_A<1,0>, 0x5f1[7,6,1,0] --> ANA_B<7,6,1,0>, 0x5f2[5:2] --> ANA_C<5>~ANA_C<2>, 0x5f4[3:0] --> ANA_E<3>~ANA_E<0>; Timer2: 0x5f8[1:0] --> ANA_A<1,0>, 0x5f9[7,6,1,0] --> ANA_B<7,6,1,0>, 0x5fa[5:2] --> ANA_C<5>~ANA_C<2>, 0x5fc[3:0] --> ANA_E<3>~ANA_E<0>.

(4) GPIO2RISC IRQ signal: Select GPIO2RISC interrupt trigger edge (positive edge or negative edge) via configuring “Polarity”, and set corresponding GPIO enabling bit “m0”/“m1”/“m2”. Enable GPIO2RISC[0]/GPIO2RISC[1]/GPIO2RISC[2] interrupt, i.e. “gpio2risc[0]” (address 0x642[5]) / “gpio2risc[1]”(address 0x642[6]) /“gpio2risc[2]” (address 0x642[7]).

Table 7- 2 GPIO lookup table2

Pin	Input (R)	Polarity 1: active low 0: active high	Irq	m0	m1	m2
PWM0/ANA_A<0>	0x580[0]	0x584[0]	0x587[0]	0x5b8[0]	0x5c0[0]	0x5c8[0]
ANA_A<1>	0x580[1]	0x584[1]	0x587[1]	0x5b8[1]	0x5c0[1]	0x5c8[1]
PWM2/SWS/ ANA_B<0>	0x588[0]	0x58c[0]	0x58f[0]	0x5b9[0]	0x5c1[0]	0x5c9[0]
PWM2_N/ ANA_B<1>	0x588[1]	0x58c[1]	0x58f[1]	0x5b9[1]	0x5c1[1]	0x5c9[1]
DI/PWM5/ ANA_B<6>	0x588[6]	0x58c[6]	0x58f[6]	0x5b9[6]	0x5c1[6]	0x5c9[6]
CK/PWM5_N/ ANA_B<7>	0x588[7]	0x58c[7]	0x58f[7]	0x5b9[7]	0x5c1[7]	0x5c9[7]
UART_TX/PWM2/ ANA_C<2>	0x590[2]	0x594[2]	0x597[2]	0x5ba[2]	0x5c2[2]	0x5ca[2]
UART_RX/PWM3/ ANA_C<3>	0x590[3]	0x594[3]	0x597[3]	0x5ba[3]	0x5c2[3]	0x5ca[3]
UAR_RTS/PWM4/ ANA_C<4>	0x590[4]	0x594[4]	0x597[4]	0x5ba[4]	0x5c2[4]	0x5ca[4]
UART_CTS/PWM5/ ANA_C<5>	0x590[5]	0x594[5]	0x597[5]	0x5ba[5]	0x5c2[5]	0x5ca[5]
PWM0/ ANA_E<0>	0x5a0[0]	0x5a4[0]	0x5a7[0]	0x5bc[0]	0x5c4[0]	0x5cc[0]
PWM1/ ANA_E<1>	0x5a0[1]	0x5a4[1]	0x5a7[1]	0x5bc[1]	0x5c4[1]	0x5cc[1]
DM/ANA_E<2>	0x5a0[2]	0x5a4[2]	0x5a7[2]	0x5bc[2]	0x5c4[2]	0x5cc[2]
DP/ANA_E<3>	0x5a0[3]	0x5a4[3]	0x5a7[3]	0x5bc[3]	0x5c4[3]	0x5cc[3]

7.1.3 Pull-up/Pull-down resistor

All GPIOs support configurable 1MΩ/10KΩ pull-up resistor or 100KΩ pull-down resistor which are all disabled by default. Analog registers including afe3V_reg08<4:7>, afe3V_reg10<4:7>~afe3V_reg18 serve to control the pull-up/pull-down resistor for each GPIO.

The DP pin also supports 1.5KΩ pull-up resistor for USB use. The 1.5KΩ pull up resistor is disabled by default and can be enabled via clearing analog register afe3V_reg00<4>. For the DP pin, user can only enable either 1.5KΩ pull up or 1MΩ/10KΩ pull-up/ 100KΩ pull-down resistor at the same time.

Please refer to Table 7-3 for details.

Take the ANA_A<0> for example: Setting analog register afe3V_reg10<5:4> to 2b'01/2b'10/2b'11 is to enable 1MΩ pull-up resistor/10KΩ pull-up resistor/100KΩ pull-down resistor respectively for ANA_A<0>; Clearing the two bits (default value) disables pull-up and pull-down resistor for ANA_A<0>.

Table 7- 3 Analog registers for pull-up/pull-down resistor control

Address	Mnemonic	Default Value	Description
afe3V_reg00<4>	dp_pullup_res_enb	1	disable usb dp 1.5KOhm pull up resistor 1: disable 0: enable
afe3V_reg08<5:4>	pullupdown_ctrl<1:0>	00	Wake up mux ANA_E<2> pull up/down controls 00 -- No pull up/down resistor 01 -- 1MOhm pull-up resistor 10 – 10kOhm pull-up resistor 11 – 100kOhm pull-down resistor
afe3V_reg08<7:6>	pullupdown_ctrl<1:0>	00	Wake up mux ANA_E<3> pull up/down controls 00 -- No pull up/down resistor 01 -- 1MOhm pull-up resistor 10 – 10kOhm pull-up resistor 11 – 100kOhm pull-down resistor

Address	Mnemonic	Default Value	Description
afe3V_reg10<5:4>	pullupdown_ctrl<1:0>	00	Wake up mux ANA_A<0> pull up/down controls 00 -- No pull up/down resistor 01 -- 1MOhm pull-up resistor 10 – 10kOhm pull-up resistor 11 – 100kOhm pull-down resistor
afe3V_reg10<7:6>	pullupdown_ctrl<1:0>	00	Wake up mux ANA_A<1> pull up/down controls 00 -- No pull up/down resistor 01 -- 1MOhm pull-up resistor 10 – 10kOhm pull-up resistor 11 – 100kOhm pull-down resistor
afe3V_reg12<5:4>	pullupdown_ctrl<1:0>	00	Wake up mux ANA_B<0> pull up/down controls 00 -- No pull up/down resistor 01 -- 1MOhm pull-up resistor 10 – 10kOhm pull-up resistor 11 – 100kOhm pull-down resistor
afe3V_reg12<7:6>	pullupdown_ctrl<1:0>	00	Wake up mux ANA_B<1> pull up/down controls 00 -- No pull up/down resistor 01 -- 1MOhm pull-up resistor 10 – 10kOhm pull-up resistor 11 – 100kOhm pull-down resistor
afe3V_reg14<1:0>	pullupdown_ctrl<1:0>	00	Wake up mux ANA_B<6> pull up/down controls 00 -- No pull up/down resistor 01 -- 1MOhm pull-up resistor 10 – 10kOhm pull-up resistor 11 – 100kOhm pull-down resistor
afe3V_reg14<3:2>	pullupdown_ctrl<1:0>	00	Wake up mux ANA_B<7> pull up/down controls 00 -- No pull up/down resistor 01 -- 1MOhm pull-up resistor 10 – 10kOhm pull-up resistor 11 – 100kOhm pull-down resistor
afe3V_reg15<1:0>	pullupdown_ctrl<1:0>	00	Wake up mux ANA_C<2> pull up/down controls 00 -- No pull up/down resistor 01 -- 1MOhm pull-up resistor 10 – 10kOhm pull-up resistor 11 – 100kOhm pull-down resistor

Address	Mnemonic	Default Value	Description
afe3V_reg15<3:2>	pullupdown_ctrl<1:0>	00	Wake up mux ANA_C<3> pull up/down controls 00 -- No pull up/down resistor 01 -- 1MOhm pull-up resistor 10 – 10kOhm pull-up resistor 11 – 100kOhm pull-down resistor
afe3V_reg15<5:4>	pullupdown_ctrl<1:0>	00	Wake up mux ANA_C<4> pull up/down controls 00 -- No pull up/down resistor 01 -- 1MOhm pull-up resistor 10 – 10kOhm pull-up resistor 11 – 100kOhm pull-down resistor
afe3V_reg15<7:6>	pullupdown_ctrl<1:0>	00	Wake up mux ANA_C<5> pull up/down controls 00 -- No pull up/down resistor 01 -- 1MOhm pull-up resistor 10 – 10kOhm pull-up resistor 11 – 100kOhm pull-down resistor
afe3V_reg18<5:4>	pullupdown_ctrl<1:0>	00	Wake up mux ANA_E<0> pull up/down controls 00 -- No pull up/down resistor 01 -- 1MOhm pull-up resistor 10 – 10kOhm pull-up resistor 11 – 100kOhm pull-down resistor
afe3V_reg18<7:6>	pullupdown_ctrl<1:0>	00	Wake up mux ANA_E<1> pull up/down controls 00 -- No pull up/down resistor 01 -- 1MOhm pull-up resistor 10 – 10kOhm pull-up resistor 11 – 100kOhm pull-down resistor

7.2 Swire

The TLSR8261F128 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 2Mbps.

SWS usage is not supported in power-saving mode (deep sleep or suspend).

7.3 I2C

The TLSR8261F128 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 (100kbps), Fast-mode (400kbps) and Fast-mode plus (1Mbps) 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 the internal 10K pull-up resistor first. In order to speed up the pull-up process, user can use external pull-up resistor with smaller resistance value (e.g. 3.3K or 4.7K) instead.

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.

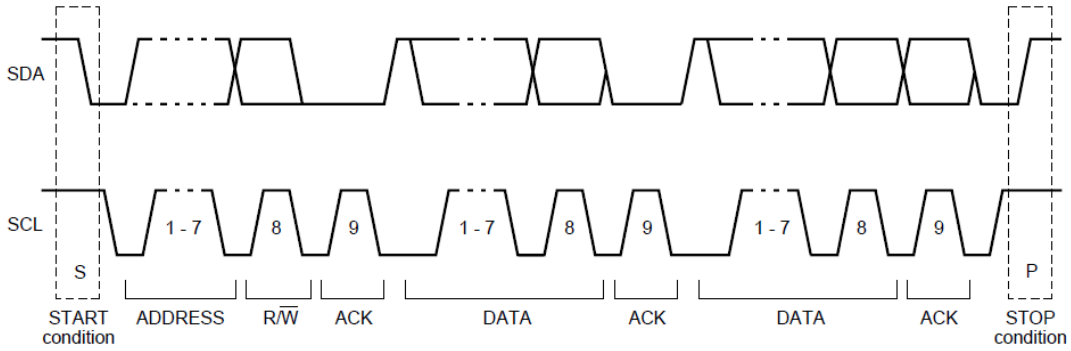


Figure 7- 2 I2C timing chart

7.3.2 Register table

Table 7- 4 Register configuration for I2C

Address	Name	R/W	Description	Reset Value
0x00	I2CSP	RW	I2C master clock speed	0x1f
0x01	I2CID	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	
0x03	I2CSCT	RW	[0]: address auto increase enable [1]: I2C master enable [2] enable Mapping Mode	0x01
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	I2CCLT	RW	[0]: launch ID cycle [1]: launch address cycle [2]: launch data write cycle [3]: launch data read cycle [4]: launch start cycle [5]: launch stop cycle [6]: enable read ID [7]: enable ACK in read command	0x00
0x20	Reg_host_map_status	R	[6:0] I2C read address	0x00
0x21	i2c_status		[0]:host_rd_clear_en [1]:host_cmd_irq_o:i2c host operation	0x01

Address	Name	R/W	Description	Reset Value
			have happened [2]:host_rd_tag_stat:i2c host operation have happened and is read operation	
0x22	clear_stats		[0]:write 1 clear software_irq [1]:write 1 clear host_cmd_irq [2]:write 1 clear host_rd_tag_stat [4]:write 1 set software_irq [5]write 1 clear ana_irq	
0x3e	Reg_host_map_adrl	R/W	Lower byte of Mapping mode buffer address	0x80
0x3f	Reg_host_map_adrh	R/W	Higher byte of Mapping mode buffer address	0xd7

7.3.3 I2C Slave mode

I2C module of the TLSR8261F128 acts as Slave mode by default. I2C slave address can be configured via register I2CID (address 0x01) [7:1].

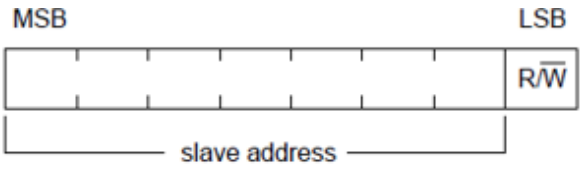


Figure 7- 3 Byte consisted of slave address and R/W flag bit

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 TLSR8261F128 according to I2C protocol. I2C module of the TLSR8261F128 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 TLSR8261F128, Register address starts from 0x800000 and SRAM address starts from 0x808000. For example, if Addr High(AddrH) is 0xaa and Addr Low (AddrL) is 0xcc, the real address of accessed data is 0x80aacc.

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

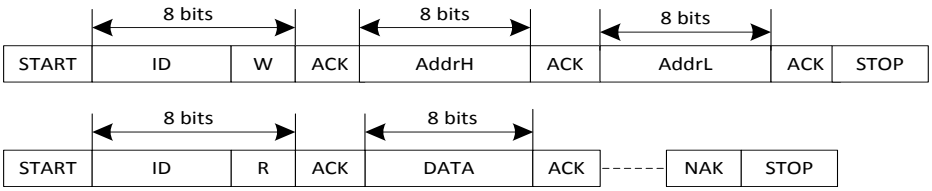


Figure 7- 4 Read format in DMA mode

Write Format in DMA mode

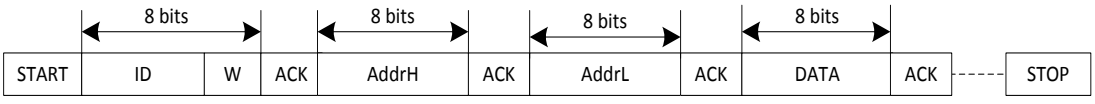


Figure 7- 5 Write format in DMA mode

7.3.3.2 Mapping mode

Mapping mode could be enabled via setting register I2CSCT (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 reg_host_map_adrl (address 0x3e, lower byte) and reg_host_map_adrh (address 0x3f, 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 previous I2C stop condition occurs. The last accessed data address during previous transfer could be checked in register reg_host_map_status (address 0x20) [6:0] which is only updated after I2C STOP occurs.

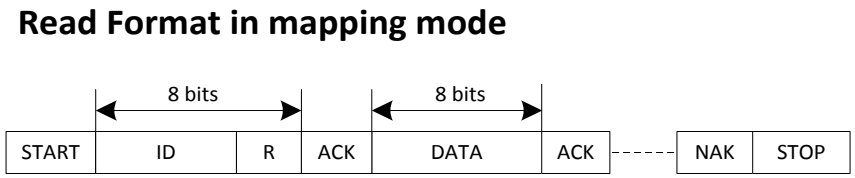


Figure 7- 6 Read format in Mapping mode

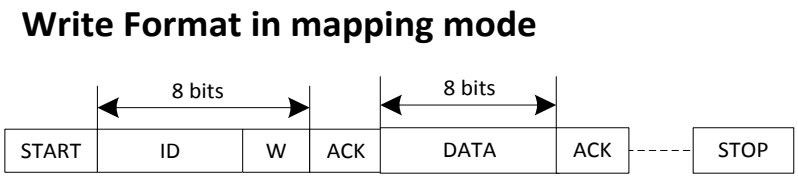


Figure 7- 7 Write format in Mapping mode

7.3.4 I2C Master mode

Address 0x03[1] should be set to 1b'1 to enable I2C master mode for the TLSR8261F128.

Address 0x00 serves to set I2C Master clock: F_{I2C} = (System Clock/(address 0x73[7:4]+1)) / (4 *clock speed configured in address 0x00). Since address 0x73[7:4] is set as 1 by default, the default F_{I2C} equals System Clock / (8 * address 0x00). If 0x73[7:4] is set as 0, F_{I2C} will change to System Clock / (4 * 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 TLSR8261F128) could send START, Slave Address, R/W bit, data and STOP by configuring address 0x07. I2C master will send enabled cycles with 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 I2CID (0x01) [7:1], 1st byte data to I2CAD, 2nd byte data to I2CDW and 3rd byte to I2CDR. To start I2C write transfer, I2CCLT (0x07) is configured to 0x3f. 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 is 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 I2CID (0x01) [7:1]. To start I2C read transfer, I2CCLT (0x07) is configured to 0xf9. 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 is more than 1 byte, user could split the cycles according to I2C protocol.

7.4 UART

The TLSR8261F128 embeds UART (Universal Asynchronous Receiver/Transmitter) to implement full-duplex transmission and reception. Both TX and RX interface are 4-layer FIFO (First In First Out) interface. Hardware flow control is also supported via RTS and CTS.

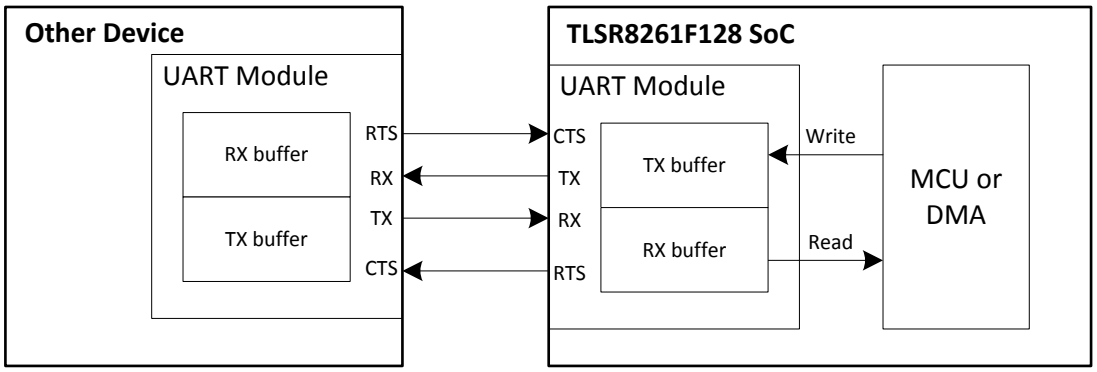


Figure 7- 8 UART communication

As shown in Figure 7- 8, 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 TLSR8261F128 UART is close to full, the TLSR8261F128 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 TLSR8261F128 receives a signal from pin CTS, it indicates that RX buffer of other device is close to full and the TLSR8261F128 should stop sending data.

Table 7- 5 Register configuration for UART

Address	Name	R/W	Description	Reset Value
0x90	uart_data_buf0	R/W	write/read buffer[7:0]	
0x91	Uart_data_buf1	R/W	Write/read buffer[15:8]	
0x92	Uart_data_buf2	RW	Write/read buffer[23:16]	
0x93	Uart_data_buf3	R/W	Write/read buffer[31:24]	

Address	Name	R/W	Description	Reset Value
0x94	uart_clk_div[7:0]	RW	uart clk div register:	0xff
0x95	Uart_clk_div[15:8]	R/W	uart_sclk = sclk/(uart_clk_div[14:0]+1) uart_clk_div[15] : 1: enable clock divider,0: disable.	0x0f
0x96	Uart ctrl0	R/W	[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	R/W	[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	R/W	[3:0] rts trig level [4] rts Parity [5] rts manual value [6] rts manual enable [7] rts enable	0xa5
0x99	Uart_ctrl3	R/W	[3:0]: rx_irq_trig level [7:4] tx_irq_trig level	0x44
0x9a	R_rxtimeout_o[7:0]	R/W	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.	0x0f
0x9b	R_rxtimeout_o[9:8]	R/W	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.	0x00
0x9c	Buf_cnt	R	[3:0]: r_buf_cnt [7:4]:t_buf_cnt	
0x9d	Uart_sts	R	[2:0] rbcnt [3] irq [6:4]wbcnt [6] write 1 clear rx [7] rx_err, write 1 clear tx	

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.

8 PWM

The TLSR8261F128 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	[0]: 0--disable PWM0, 1--enable PWM0 [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_CLK	R/W	(PWM_CLK+1)*sys_clk	0x00
0x782	PWM_MODE	R/W	[1:0]: 00-pwm0 normal mode [1:0]: 01-pwm0 count mode [1:0]: 11-pwm0 IR mode [3:2]: 00-pwm1 normal mode [3:2]: 01-pwm1 count mode [3:2]: 11-pwm1 IR mode	0x00
0x783	PWM_CC0	R/W	[5:0]:1 ‘b1 invert PWM output	0x00
0x784	PWM_CC1	R/W	[5:0]:1‘b1 invert PWM_INV output	0x00
0x785	PWM_CC2	R/W	[5:0]:1’b1 PWM’ pola, low level first	0x00
0x788	PWM_PHASE0	R/W	[7:0] bits 7-0 of PWM0's phase time	0x00
0x789	PWM_PHASE0	R/W	[15:8] bits 15-8 of PWM0's phase time	0x00
0x78a	PWM_PHASE1	R/W	[7:0] bits 7-0 of PWM1's phase time	0x00
0x78b	PWM_PHASE1	R/W	[7:8] bits 15-8 of PWM1's phase time	0x00
0x78c	PWM_PHASE2	R/W	[7:0] bits 7-0 of PWM2's phase time	0x00
0x78d	PWM_PHASE2	R/W	[15:8] bits 15-8 of PWM2's phase time	0x00
0x78e	PWM_PHASE3	R/W	[7:0] bits 7-0 of PWM3's phase time	0x00
0x78f	PWM_PHASE3	R/W	[15:8] bits 15-8 of PWM3's phase time	0x00
0x790	PWM_PHASE4	R/W	[7:0] bits 7-0 of PWM4's phase time	0x00
0x791	PWM_PHASE4	R/W	[15:8] bits 15-8 of PWM4's phase time	0x00
0x792	PWM_PHASE5	R/W	[7:0] bits 7-0 of PWM5's phase time	0x00
0x793	PWM_PHASE5	R/W	[15:8] bits 15-8 of PWM5's phase time	0x00

Address	Mnemonic	Type	Description	Reset Value
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
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		[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

Address	Mnemonic	Type	Description	Reset Value
0x7ab	PWM_TMAX5	R/W	[15:8] bits 15-8 of PWM5's cycle time	0x00
0x7ac	PWM_PNUM0	R/W	[7:0]PWM0 Pulse num in count mode and IR mode	0x00
0x7ad	PWM_PNUM0	R/W	[15:8]	0x00
0x7ae	PWM_PNUM1	R/W	[7:0]PWM1 Pulse num in count mode and IR mode	0x00
0x7af	PWM_PNUM1	R/W	[15:8]	0x00
0x7b0	PWM_MASK	R/W	INT mask [0] PWM0 Pnum int 0: disable 1: Enable [1] PWM1 Pnum int 0: disable 1: Enable [2] PWM0 frame int 0: disable 1: Enable [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	0x00
0x7b1	PWM_INT	R/W	INT status ,write 1 to clear [0]:PWM0 pnum int(have sent PNUM pulse,PWM_NCNT==PWM_PNUM) [1]:PWM1 pnum int [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

Address	Mnemonic	Type	Description	Reset Value
0x7b4	PWM_CNT0	R	[7:0]PWM 0 cnt value	
0x7b5	PWM_CNT0		[15:8]PWM 0 cnt value	
0x7b6	PWM_CNT1	R	[7:0]PWM 1 cnt value	
0x7b7	PWM_CNT1		[15:8]PWM 1 cnt value	
0x7b8	PWM_CNT2	R	[7:0]PWM 2 cnt value	
0x7b9	PWM_CNT2		[15:8]PWM 2 cnt value	
0x7ba	PWM_CNT3	R	[7:0]PWM 3 cnt value	
0x7bb	PWM_CNT3		[15:8]PWM 3 cnt value	
0x7bc	PWM_CNT4	R	[7:0]PWM 4 cnt value	
0x7bd	PWM_CNT4		[15:8]PWM 4 cnt value	
0x7be	PWM_CNT5	R	[7:0]PWM 5 cnt value	
0x7bf	PWM_CNT5		[15:8]PWM 5 cnt value	
0x7c0	PWM_NCNT0	R	[7:0]PWM0 pluse_cnt value	
0x7c1	PWM_NCNT0		[15:8]PWM0 pluse_cnt value	
0x7c2	PWM_NCNT1	R	[7:0]PWM1 pluse_cnt value	
0x7c3	PWM_NCNT1		[15:8]PWM1 pluse_cnt value	

8.2 Enable PWM

Register PWM_EN (address 0x780)[5:0] 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_CLK (address 0x781) serves to set the frequency dividing factor for PWM clock. Formula below applies:

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

8.4 PWM waveform, polarity and output inversion

Each PWM channel has independent counter and 3 status including “Delay”, “Count” and “Remaining”. Count and Remaining status form a signal frame.

8.4.1 PWM waveform

When PWM#n is enabled, PWM#n enters Delay status. By default PWM#n outputs Low level at Delay status. The Delay status duration, i.e. Phase time, is configured in register PWM_PHASE#n (address 0x788~0x793). Phase difference between PWM channels is allowed by different phase time configuration.

After Phase time expires, PWM#n exits Delay status and starts to send signal frames. First PWM#n is at 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#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) expires.

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 0x783) and PWM_CC1 (address 0x784). When the inversion bit is enabled, the corresponding PWM channel waveform 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 0x785), PWM#n will output Low level at Count status and High level at Remaining status.

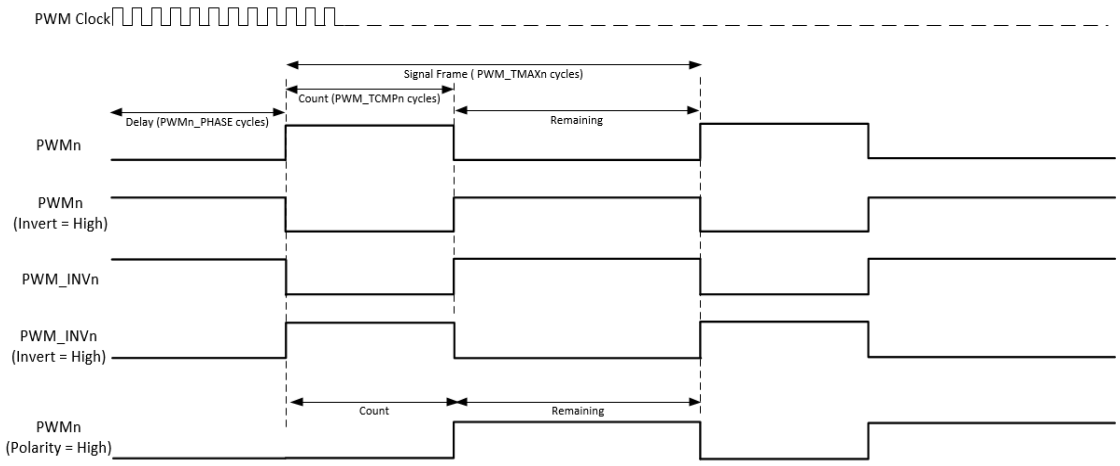


Figure 8- 1 PWM output waveform chart

8.5 PWM mode

8.5.1 Select PWM mode

PWM0 and PWM1 support 3 modes, including Continuous (normal) mode, Counting mode, and IR mode. PWM2~PWM5 only support Continuous mode.

Register PWM_MODE (address 0x782) serves to select PWM0/PWM1 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 to stop it; when stopped, the PWM output will turn low immediately.

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

A frame interruption will be generated (if enabled) after each signal frame is finished.

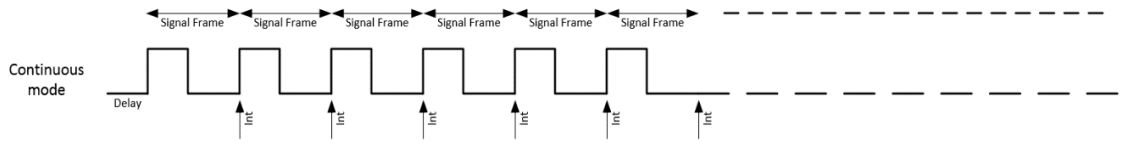


Figure 8-2 Continuous mode

8.5.3 Counting mode

Only PWM0 and PWM1 support Counting mode. In this mode, PWM#n (n=0,1) 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) and PWM_PNUM1 (address 0x7ae~0x7af). After a pulse group is finished, PWM#n will be disabled automatically, and a Pnum interruption will be generated if enabled via register PWM_MASK (address 0x7b0[0:1]).

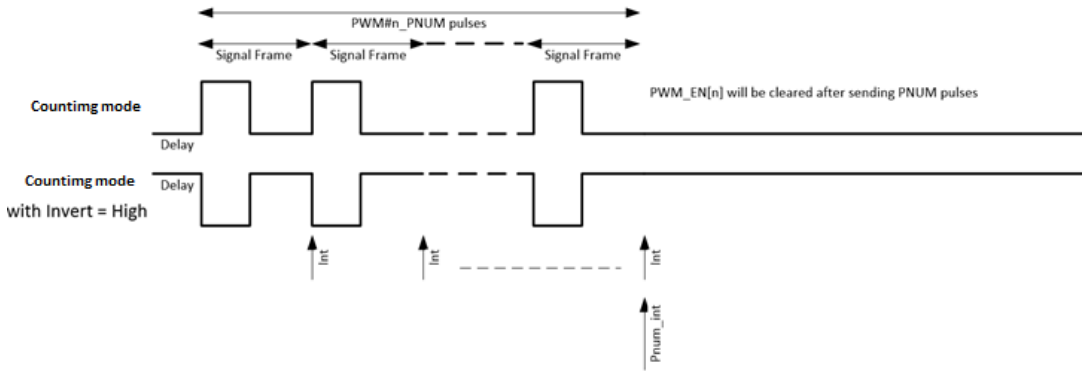


Figure 8-3 Counting mode

Counting mode also serves to stop IR mode gracefully. Refer to **section 8.5.4** for details.

8.5.4 IR mode

Only PWM0 and PWM1 support IR mode. In this mode, specified number of frames is defined as one pulse group. In contrast to Counting mode where PWM#n (n=0,1) stops after first pulse group finishes, PWM#n will constantly send pulse groups in IR mode.

During IR mode, waveform could also be changed freely. New configuration for PWM_TCMP#n and PWM_TMAX#n will take effect in the next pulse group.

To stop IR mode and complete current pulse group, user can switch PWM#n from IR mode to Counting mode so that PWM#n will stop after current pulse group is finished. If PWM#n is disabled directly via PWM_EN (0x780[0:1]), PWM#n output will turn Low immediately despite of current pulse group.

A frame interruption/Pnum interruption will be generated (if enabled) after each

signal frame/pulse group is finished.

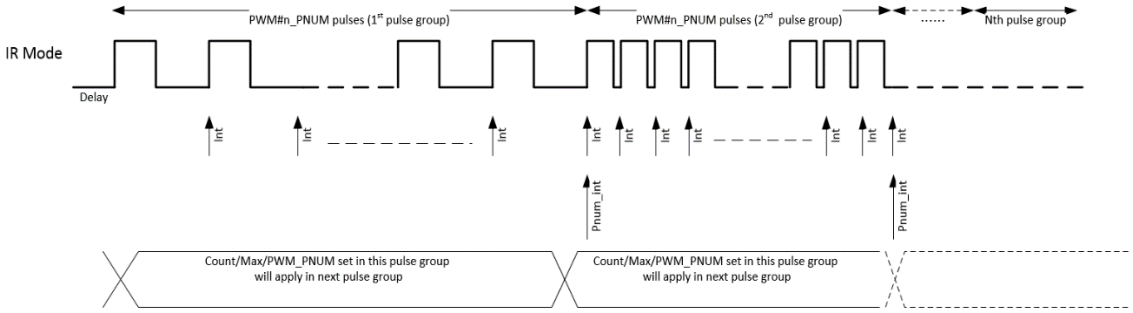


Figure 8-4 IR mode

8.6 PWM interrupt

There are 8 interrupt sources from PWM function. After each signal frame, PWM#n will generate a frame-done IRQ (Interrupt Request) signal. In Counting mode and IR mode, PWM0/PWM1 will generate a Pnum IRQ signal after completing a pulse group. Interrupt status can be cleared via register PWM_INT (address 0x7b1).

9 Quadrature Decoder

The TLSR8261F128 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 6 pins of PortE, 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	ANA_E<0>
1	ANA_E<1>
2	rsvd
3	rsvd
4	ANA_C<4>
5	ANA_C<5>
6	ANA_B<6>
7	ANA_B<7>

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.

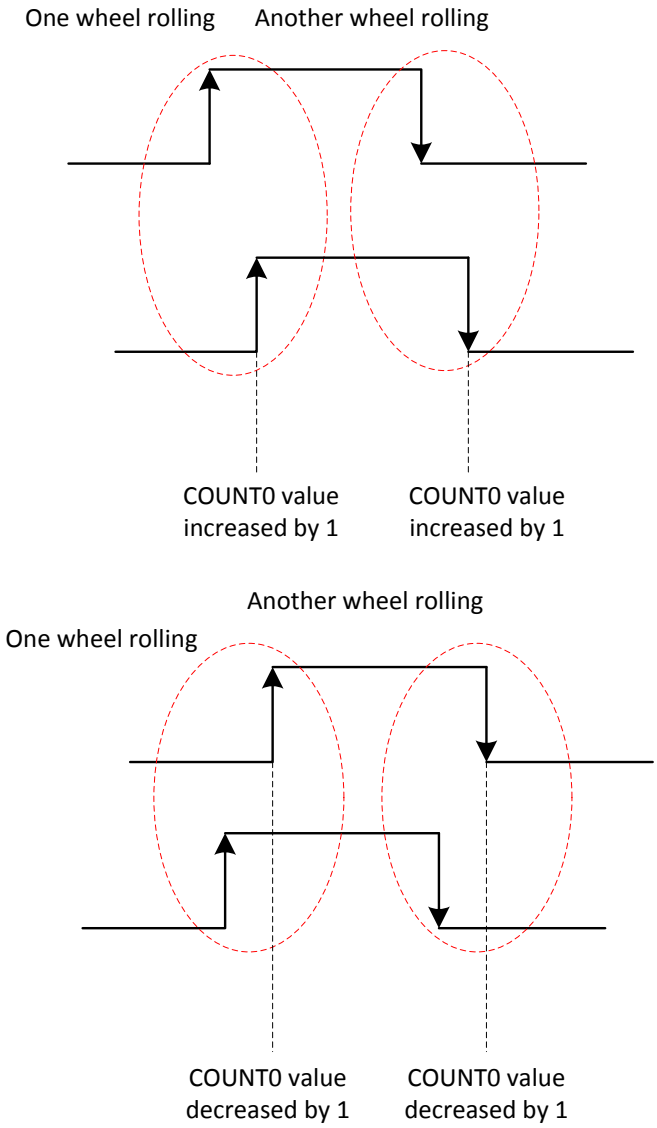


Figure 9- 1 Common mode

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.

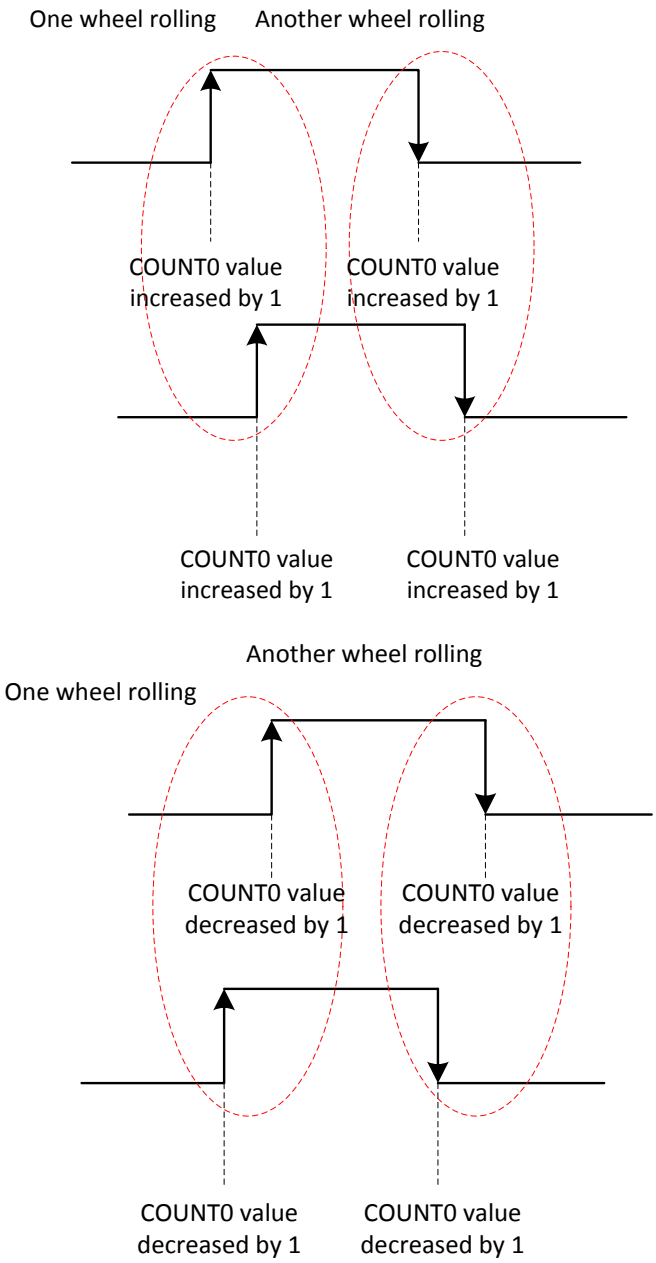


Figure 9- 2 Double accuracy mode

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.

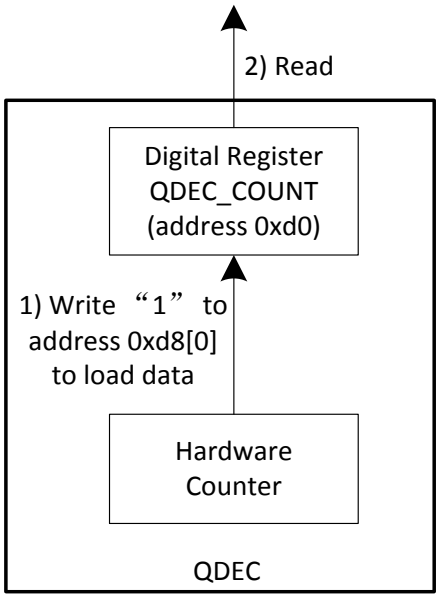


Figure 9- 3 Read real time counting value

9.4 QDEC reset

Address 0xd6[0] 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.

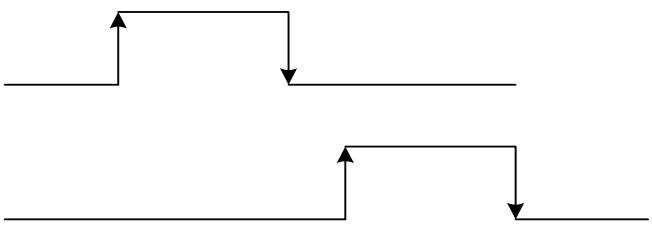


Figure 9- 4 Shuttle mode

9.6 Timing sequence

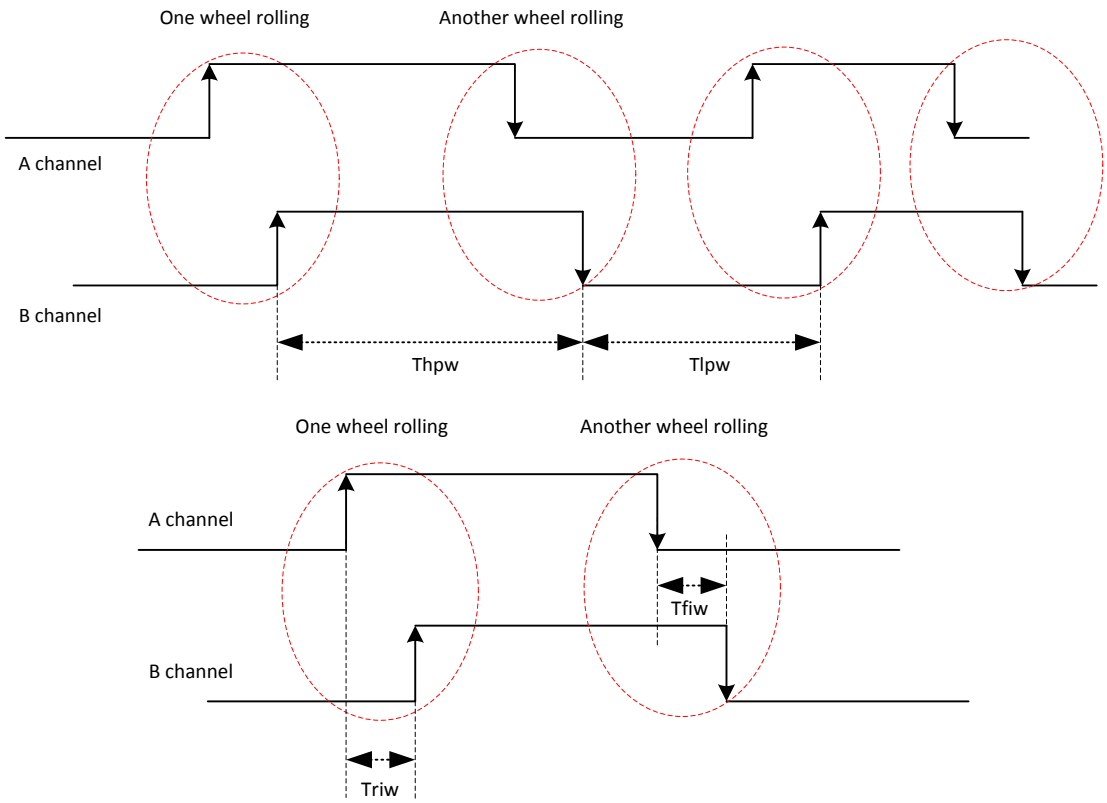


Figure 9- 5 Timing sequence chart

Table 9- 2 Timing

Time interval	Min Value
Thpw (High-level pulse width)	2^（n+1）*clk_32k *3 (n=0xd1[2:0])
Tlpw (Low-level pulse width)	2^(n+1) *clk_32k *3 (n=0xd1[2:0])
Triw (Interval width between two rising edges)	2^(n+1) *clk_32k (n=0xd1[2:0])
Tfiw (Interval width between two falling edges)	2^(n+1) *clk_32k (n=0xd1[2:0])

QDEC module works based on 32K 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_32k *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_32k 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_32k”.

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	
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	
0xd2	QDEC_CHNA	R/W	[2:0] QDEC input pin select for channel a choose 1 of 6 pins for input channel a {pb[7:6],pc[5:4], pe[1:0]}	0x00
0xd3	QDEC_CHNB	R/W	[2:0] QDEC input pin select for channel b choose 1 of 6 pins for input channel b {pb[7:6],pc[5:4], pe[1:0]}	0x01
0xd6	QDEC_RST	R/W	[0]Write 1 to reset QDEC	0x0
0xd7	QDEC_DOUBLE	R/W	[0]Enable double accuracy mode	0x0
0xd8	DATA_LOAD	R/W	[0]write 1 to load data when load completes it will be 0	

10 ADC

The TLSR8261F128 integrates one ADC module, which can be used to sample battery voltage, temperature sensor and external analog input signal.

10.1 ADC clock

ADC clock derives from FHS. Please refer to **section 4.4.1** for ADC clock configuration.

Note: ADC clock must be lower than 5MHz when ADC reference voltage is selected as AVDD and must be no more than 4MHz when ADC reference voltage is selected as 1.224V or 1.428V.

10.2 Set period

In general, the ADC Control Module in Telink MCU divides the whole sampling and conversion process into three parts via time-division: Misc corresponding to auto channel 0, L (Left) corresponding to auto channel 1, and R (Right) corresponding to auto channel 2.

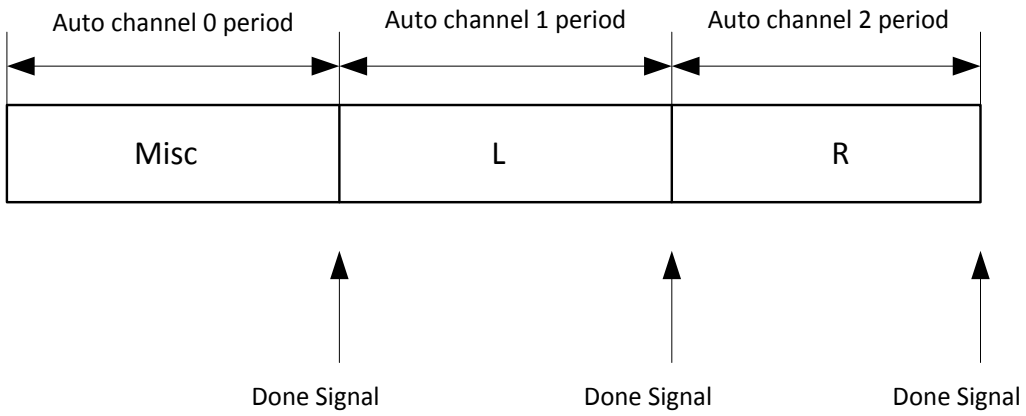


Figure 10- 1Sampling and analog-to-digital conversion process

In TLSR8261F128, only Misc channel is supported.

Addresses 0x30 and 0x31 serve to set lower byte and higher byte of the period (Sampling time plus converting time) for Misc:

Period of Misc = {ADCMAXMH, ADCMAXML} * system clock period.

Since the TLSR8261F128 doesn't support audio input, address 0x33[5:4] shall always be set to 2b'00 to skip the period for L (Left) and R (Right) channel, i.e., Auto channel 1&2.

10.3 Select ADC input range

Address 0x2b[1:0] serves to set reference voltage for Misc: 1.428V, AVDD or 1.224V.

ADC maximum input range is the same as the ADC reference voltage.

10.4 Select resolution and sampling time

Address 0x3c[5:3] serves to set resolution for Misc: 7, 9, 10, 11, 12, 13, 14bits. ADC data format is always 14bit no matter the conversion bit is set. For example, 12 bits resolution indicates higher 12 bits are valid bits and the lower 2 bits are invalid bits.

Address 0x3c[2:0] serves to set sampling time for Misc: 3, 6, 9, 12, 18, 24, 48 or 144 * ADC clock period. The lower sampling cycle, the shorter ADC convert time.

10.5 Select input mode and channel

The TLSR8261F128 ADC supports two input modes and 4 input channels.

Address 0x2c serves to select input mode and channel for Misc.

Address 0x2c[6:5] serves to select differential mode or single-end input mode for Misc.

When address 0x2c[6:5] is set to 2b'00 to select single-end mode, 0x2c[4:0] serves to select input channel.

When address 0x2c[6:5] is set to 2b'01, differential input mode is selected, the corresponding channel (i.e. ANA_B<1>) identified by address 0x2c[6:5] is selected as negative input, and the positive input is selectable via address 0x2c[4:0]. If address 0x2c is set to 0x25 (i.e. 8b'00100101), ANA_B<0> and ANA_B<1> are selected as positive-end and negative-end input of differential mode; actual input signal for ADC

is the difference of $V_{ANA\_B<0>}$ and $V_{ANA\_B<1>}$ (i.e. $V_{ANA\_B<0>}$ minus $V_{ANA\_B<1>}$).

10.6 Enable auto mode and output

Address 0x33[3] serves to enable Misc auto sampling and conversion mode. If address 0x33[3] is set as 1b'0 to select manual mode, one operation of writing address 0x35 with data "0x80" manually starts a sampling and conversion process.

Address 0x33[2] should be set as 1b'0 to disable ADC audio output, and 0x33[5:4] should be set as 2b'00 to select "no audio".

Address 0x2c[7] serves to set data format during Misc period. Real time output data can be read from addresses 0x38~0x39.

10.7 ADC done signal

ADC done signal is selectable via address 0x33[7:6]. Generally 0x33[7:6] is set to "2b'01" (or 2b'11) to select "rising" method, which means a rising edge of "ADC Valid" signal indicates one analog-to-digital conversion process is done.

10.8 ADC status

ADC busy flag bit, i.e. address 0x3a[0], indicates whether ADC is busy.

10.9 Battery detection

The TLSR8261F128 ADC can be used to sample battery voltage via the Misc channel.

In this section, two cases are introduced.

For generic configuration, such as clock, period, resolution, sampling time, and etc., please refer to sections above.

10.9.1 Case 1: Battery directly connected to chip

This case applies to voltage detection for battery power which is directly connected to the chip.

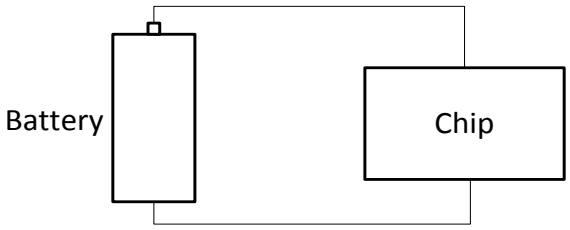


Figure 10- 2 Battery detect case 1

First write digital address **0x2c** with "**0x12**" (i.e. 8b'00010010) to select "1/3 voltage division detection" as single-end input.

Then write analog register **afe3V_reg02<5:4>** with "**2b'01**" to select IO power supply (1/3 Vddh) for 1/3 voltage division detection.

Set reference voltage (V_{REF}) as 1.428V or 1.224V via writing digital address **0x2b[1:0]** with "2b'00" or "2b'10".

Battery voltage, V_{bat}, equals to $\mathbf{V_{REF}} \times \frac{ADC\ output}{2^n} \times \mathbf{3}$.

*Note: In the formula above, ADC output is read from digital address {0x39, 0x38}, while "n" indicates the resolution configured in digital address 0x3c[5:3].

10.9.2 Case 2: Battery connected to chip via boost DCDC

This case applies to voltage detection for battery power which is not directly corrected to the chip, for example, it may be connected to the chip via a boost DCDC. In this case, the **ANA_B<7>** pin of the chip needs to be connected to the battery, where an internal 50KOhm/25KOhm divider network is used to perform voltage division. Other pins corresponding to ADC channels may be used for battery detection as well, but an external resistor divider network needs to be added.

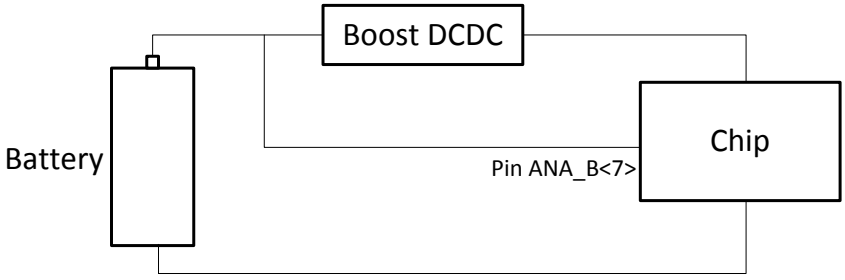


Figure 10- 3 Battery detect case2

First write digital address **0x2c** with "**0x12**" (i.e. 8b'00010010) to select "1/3 voltage division detection" as single-end input.

Then write analog register **afe3V_reg02<5:4>** with "**2b'10**" to select IO input battery voltage (1/3 ANA_B<7>) for 1/3 voltage division detection.

Set reference voltage (V_{REF}) as 1.428V or 1.224V via writing digital address **0x2b[1:0]** with "2b'00" or "2b'10".

Battery voltage, V_{bat}, equals to $\mathbf{V_{REF}} \times \frac{\boldsymbol{ADC\ output}}{\boldsymbol{2^n}} \times \mathbf{3}$.

*Note: In the formula above, ADC output is read from digital address {0x39, 0x38}, while "n" indicates the resolution configured in digital address 0x3c[5:3].

10.10 Register table

Table 10- 1 Register table related to SAR ADC

Address	Mnemonic	R/W	Description	Default value
Digital Register				
0x2b	ADCREF	RW	SAR ADC reference voltage selection [1:0]: Misc [3:2]: L 00: 1.428V 01: AVDD 10: 1.224V	0x0b
0x2c	ADCMUXM	RW	[4:0]: Analog input selection bit for Misc 00000: no input 00001: rsvd (ANA_C<0>) 00010: rsvd (ANA_C<1>) 00011: rsvd (ANA_C<6>) 00100: rsvd (ANA_C<7>) 00101: ANA_B<0> 00110: ANA_B<1> 00111: rsvd (ANA_B<2>) 01000: rsvd (ANA_B<3>) 01001: rsvd (ANA_B<4>)	0x02

Address	Mnemonic	R/W	Description	Default value
Digital Register				
			01010: rsvd (ANA_B<5>) 01011: ANA_B<6> 01100: ANA_B<7> 01101: rsvd (pga_Vom) 01110: rsvd (pga_Vop) 01111: tempsensor_n (temperature sensor negative) 10000: tempsensor_p (temperature sensor positive) 10001: AVSS 10010: 1/3 voltage division detection (selectable via analog register afe3V_reg02<5:4>) others: reserved [6:5]: Differential analog input selection bits for Misc 00: single-end 01: ANA_B<1> as inverting input 10: rsvd (ANA_B<3> as inverting input) 11: rsvd (pga_Vop as inverting input) [7]: data format setting during Misc period 0: unsigned 1: bit<14> is inverted	
0x2d	ADCMUXL	RW	[4:0]: Analog input selection bit for L [6:5]: Differential analog input selection bits for L [7]: data format setting during L period Refer to 0x2c	0x00
0x2e	ADCMUXR	RW	Reserved	0x01
0x2f	ADCRES	RW	[2:0]: SAR ADC resolution selection for L 000: 7 001: 9	0x01

Address	Mnemonic	R/W	Description	Default value
Digital Register				
			010: 10 011: 11 100: 12 101: 13 110: 14 111: 14	
0x30	ADCMAXML	RW	ADC auto channel 0 (Misc) period low byte	0xe0
0x31	ADCMAXMH	RW	ADC auto channel 0 (Misc) period high byte Period = { ADCMAXMH, ADCMAXML} system clocks	0x00
0x32	ADCMAXLR	RW	ADC auto channel 1 (L)& 2 period Period = ADCMAXLR * 16 system clocks	0x06
0x33	ADCCTRL	RW	[0]: enable auto channel 1 (L) [2]: enable audio ADC output [3]: enable auto channel 0 (Misc) [5:4]: audio ADC mode 00: no audio; 01: mono; others: reserved [7:6]: ADC done signal select 01,11: rising; 10: falling	0x27
0x38	ADCOUTPUT0	R	ADC data lower bits	
0x39	ADCOUTPUT1	R	ADC data higher bits	
0x3a	ADCBUSY	R	ADC status [0]: ADC busy flag	
0x3c	ADCMRESSAMP	RW	[5:3]: SAR ADC resolution selection for Misc Refer to 0x2f[2:0] [2:0]: Select number of clock cycles for ADC Misc sampling time 000: 3 cycles 001: 6 cycles	0x00

Address	Mnemonic	R/W	Description	Default value
Digital Register				
			010: 9 cycles 011: 12 cycles 100: 18 cycles 101: 24 cycles 110: 48 cycles 111: 144 cycles	
0x3d	ADCLSAMP	RW	[2:0]: Select number of clock cycles for ADC L sampling time Refer to 0x3c[2:0]	0x00
Analog register				
afe3V_reg02 <5:4>	batdet_ctl_3v <1:0>		choose IO power supply or IO input battery voltage for 1/3 voltage division detection 00: N/A 01: 1/3 Vddh (i.e. AVDD3) 10: 1/3 ANA_B<7> 11: N/A	00

11 Key Electrical Specifications

11.1 Absolute maximum ratings

Table 11- 1 Absolute Maximum Ratings

Characteristics	Sym.	Min.	Max	Unit	Test Condition
Supply Voltage	VDD	-0.3	3.9	V	All AVDD and DVDD 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.

11.2 Recommended operating condition

Table 11- 2 Recommended operation condition

Item	Sym.	Min	Typ.	Max	Unit	Condition
Power-supply voltage	VDD	1.9	3.3	3.6	V	
Supply rise time (from 0V to 1.9V)	t_R			28*[3]	ms	PVT
Operating Temperature Range	T_{Opr}	-40		85	°C	

[3] Note: To ensure correct chip behavior during power-up, please refer to Figure 2-4 for the POR sequences. It is recommended for the supply to rise above minimum operating voltage 1.9V no later than TpwRst (> 2ms default value) after the longer one of the following: 1). supply reaches POR threshold; 2). reset release time which is > 28ms with 1uF reset cap and proportionally longer if reset cap is increased. Normally the reset release time is the dominant factor in determining the allowed rising time. As shown in Figure 2-4, t3 is the system release time and it depends on both the UVLO output and the Reset signal. Power supply is required to reach minimum operating voltage 1.9V by time instant t3. .

11.3 Electrical characteristics

Table 11- 3 Electrical characteristics

(Test condition: VDD=3.3V and T=25℃ for Typ. value,

all Max values are defined over process and voltage 1.9~3.6V at T=25℃)

Item	Sym.	Min	Typ.	Max	Unit	Condition
Tx Transceiver Current	I_{Tx}	-	14.7	15.7	mA	Continuous Tx transmission, 0dBm output power
		-	24.5	26	mA	Continuous Tx transmission, Maximum output power
Rx Transceiver Current	I_{Rx}	-	13.8	14.7	mA	Continuous Rx reception
Suspend Current	I_{Susp}	-	10	16	uA	IO wakeup
	I_{Susp}	-	11.4	17.4	uA	32k RC wakeup
Deep sleep current	I_{Deep}	-	2.4	3	uA	

11.4 General Characteristics

Table 11- 4 General characteristics

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

Power-up/ Power-down sequence (see section 2.5.1)						
Item	**Sym.**	**Min**	**Typ.**	**Max**	**Unit**	**Condition**
VDD voltage when V_{UVLO} turns to high level	V_{POR}	1.67	1.75	1.79	V	
VDD voltage when V_{UVLO} turns to low level	V_{Pdn}	1.58	1.63	1.67	V	
Delay counter value	T_{PwRst}	Configurable via analog register 0x20				

11.5 GPIO drive strength

Table 11- 5 GPIO drive strength

(over process and T=-40~+85℃)

GPIO	"DS"=0			"DS"=1			Unit	Condition
	Min	Typ.	Max	Min	Typ.	Max		
ANA_B<0>	7.3	8	10.1	11.4	12	15.4	mA	VDD=3.3V, VOH=0.9VDD, Source current
ANA_E<1>, ANA_E<0>	9.9	12	13.7	12.4	16	17.1	mA	
ANA_E<3>, ANA_E<2>	6.8	8.0	10	8.6	12	14.1	mA	
Other GPIOs	0.6	0.7	1.1	3.3	4	6.1	mA	
ANA_B<0>	3.0	3.6	4.5	5.2	6.2	7.5	mA	VDD=2.0V, VOH=0.9VDD, Source current
ANA_E<1>, ANA_E<0>	4.2	5	6.2	5.5	6.4	7.9	mA	
ANA_E<3>, ANA_E<2>	3	3.6	4.4	4.2	5.1	6.4	mA	
Other GPIOs	0.2	0.3	0.5	1.3	1.9	2.5	mA	
ANA_B<0>	7.9	9.2	11.2	14.8	17.1	20.7	mA	VDD=3.3V, VOL=0.1VDD, Sink current
ANA_E<1>, ANA_E<0>	11.5	13.3	16.1	14.7	17.0	20.4	mA	
ANA_E<3>, ANA_E<2>	7.3	9.3	11.6	9.5	13.4	16.7	mA	
Other GPIOs	2.0	2.4	3.0	3.5	4.6	5.8	mA	
ANA_B<0>	3.2	3.8	4.9	6.1	7.3	9.1	mA	VDD=2.0V, VOL=0.1VDD, Sink current
ANA_E<1>, ANA_E<0>	4.7	5.6	7.1	6.1	6.2	9.1	mA	
ANA_E<3>, ANA_E<2>	3.2	3.9	5.0	4.3	5.6	7.2	mA	
Other GPIOs	0.8	1.0	1.3	1.5	1.8	2.4	mA	

11.6 Pull-up/Pull-down resistor

Table 11- 6 Pull-up/Pull-down resistor
(over process and T=-40~+85℃)

Pull-up/Pull-down resistor	Min	Max	Condition
Pull-up 1Mohm	0.7Mohm	1.1Mohm	VDD=3.6V
	0.9Mohm	1.4Mohm	VDD=3V
	1.6Mohm	2.7Mohm	VDD=2V
Pull-up 10kohm	25kohm	32kohm	VDD=3.6V
	26kohm	39kohm	VDD=3V
	46kohm	73kohm	VDD=2V
Pull-down 100kohm	246kohm	271kohm	VDD=3.6V
	207kohm	339kohm	VDD=3V
	398kohm	666kohm	VDD=2V

11.7 Flash memory characteristics

Table 11- 7 Flash memory characteristics
(T= -40℃~85℃)

Item	Sym.	Min	Typ.	Max	Unit	Condition
Retention period		20			year	
Number of erase cycles		100k			cycle	
VDD for programming		2.7		3.6	V	Note this refers to the SoC supply
Sector size			4		kB	
Page programming time	T_{PP}		0.8	4.0	ms	
Sector erase time	T_{SE}		120	400	ms	
Block erase time (32kB/64kB)	T_{BE}		0.4/0.8	1.2/2.4	s	
Page programming current	I_{PP}			7	mA	

Item	Sym.	Min	Typ.	Max	Unit	Condition
Sector erase current	I_{SE}			7	mA	
Block erase current	I_{BE}			7	mA	

Table 11- 8 Flash DCDC characteristics

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

Flash DCDC output						
Item	**Sym.**	**Min**	**Typ.**	**Max**	**Unit**	**Condition**
DCDC output		2.7	2.8	2.9	V	Load current = 5mA

11.8 SPI characteristics

Table 11- 9 SPI characteristics

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

Item	Sym.	Min	Typ.	Max	Unit	Condition
CK frequency	F_{CK}			4	MHz	Slave
CK duty cycle clock			50		%	Master
DI setup time		18			ns	Slave
		45			ns	Master
DI hold time		16			ns	Slave
		45			ns	Master
CK low to DO valid time				27	ns	Slave
				84	ns	Master
CN setup time		9			ns	Master/Slave
CN high to DI tri-state*[4]		206		209	ns	Master

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

11.9 I2C characteristics

Table 11- 10 I2C characteristics

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

Item	Sym.	Standard mode		Fast mode		Unit	Condition
		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	

11.10 RF performance

Table 11- 11 RF characteristics*[5]

PVT: over process, voltage 1.9~3.6V, and T=-40~+85℃

PV @ 25℃: over process and voltage at 25℃

Item		Min	Typ	Max	Unit	Condition
BLE 1Mbps RF_Rx performance (±250kHz deviation, PER≤30.8%)						
Sensitivity	1Mbps		-91	-88	dBm	Max is defined over PV @ 25℃
Frequency Offset Tolerance		-300		+300*[6]	KHz	PVT
Co-channel rejection			8	10	dB	PV @ 25℃
In-band rejection C/I (modulation interference)	±1 MHz offset		6	15	dB	PV @ 25℃
	±2 MHz offset		-20	-6	dB	
	±3 MHz offset		-29	-23	dB	
	>4MHz offset		-41	-35	dB	
Image rejection			-34	-31	dB	PV @ 25℃
BLE 1Mbps RF_Tx performance						
Output power		4	7	8	dBm	over PV @ 25℃
Harmonics	2nd		-48.2	-42	dBm	Max power and over PVT
	3rd		-56.2	-52	dBm	Max power and over PVT
Modulation 20dB bandwidth		1.1	1.2	1.3	MHz	PVT
Inband Emission	2Mhz		-42	-41	dBm	PVT
	>=3Mhz		-42	-41	dBm	
Frequency Deviation F1avg		225		275	Khz	PVT

[5] Note: Unless otherwise specified, the RF performances in this section are measured on Telink 4-layer Evaluation Board.

[6] Note: at +/-300kHz the sensitivity level may drop by 4dB; to keep the sensitivity level drop within 1dB, the frequency offset tolerance is +/-50kHz.

Item		Min	Typ	Max	Unit	Condition
Frequency Deviation F2Max		185			Khz	PVT
Frequency Deviation F2Avg/F1Avg		0.8				PVT
Spurious Emission Conducted Measurement	f < 1Ghz		-76	-52.3	dBm	T=25℃, Max Output Power, Max Voltage, fc @ 2440Mhz
	f > 1Ghz		-72.1	-59.4	dBm	
Spurious Emission Conducted Measurement	f < 1Ghz		-76	-52.3	dBm	T=25℃, Max Output Power, Max Voltage, fc @ 2480Mhz
	f > 1Ghz		-72.1	-59.4	dBm	
BLE 2Mbps RF_Rx performance (±500kHz deviation, PER≤30.8%)						
Sensitivity	2Mbps		-89	-86	dBm	Max is defined over PV @ 25℃
Frequency Offset Tolerance		-200		+200*[7]	KHz	PV @ 25℃
Co-channel rejection			10	12	dB	PV @ 25℃
In-band rejection C/I (modulation interference)	±2 MHz offset		2	8	dB	PV @ 25℃
	±4 MHz offset		-19	-14	dB	
	±6 MHz offset		-23	-18	dB	
	>8MHz offset		-26	-22	dB	
Image rejection			-26	-23	dB	PV @ 25℃
BLE 2Mbps RF_Tx performance						
Output power		4	7	8	dBm	over PV @ 25℃
Modulation 20dB bandwidth		2.2	2.5	2.8	MHz	PVT
harmonic	2nd		-48.2	-42	dBm	Max power and over PVT
	3rd		-56.2	-52	dBm	Max power and over PVT
Inband Emission	4Mhz		-42	-41	dBm	PVT

[7] Note: at +/-200kHz the sensitivity level will drop by 1dB.

Item		Min	Typ	Max	Unit	Condition
	5Mhz		-42	-41	dBm	PVT
	>=6Mhz		-42	-41	dBm	PVT
Frequency Deviation F1avg		450		550	Khz	PVT
Frequency Devation F2Max		370			Khz	PVT
Frequency Deviation F2Avg/F1Avg		0.8				PVT
Spurious Emission Conducted Measurement	f < 1Ghz		-76	-52.3	dBm	T=25℃, Max Output Power, Max Voltage, fc @ 2440Mhz
	f > 1Ghz		-71	-59.4	dBm	
Spurious Emission Conducted Measurement	f < 1Ghz		-76	-52.3	dBm	T=25℃, Max Output Power, Max Voltage, fc @ 2480Mhz
	f > 1Ghz		-71	-59.4	dBm	
RSSI @BLE mode						
RSSI linear range		-90		-54	dBm	
RSSI Resolution			+/-4		dB	

11.11 Crystal characteristics

Table 11- 12 Crystal characteristics

Item	Sym.	Min	Typ.	Max	Unit	Notes
12MHz crystal						
Nominal frequency (parallel resonant)	f_{NOM}		12		MHz	
Frequency tolerance	f_{TOL}	-20		+20	ppm	
Load capacitance	C_L	5	12	18	pF	Programmable on chip load cap (PV @ 25℃ for min and Max)
Equivalent series resistance	ESR		50	100	ohm	PVT
32.768KHz crystal						
Nominal frequency (parallel resonant)	f_{NOM}		32.768		KHz	
Frequency tolerance	f_{TOL}	-100		+100	ppm	
Load capacitance	C_L	6		12.5	pF	Programmable on chip load cap (PV @25℃ for min and max)
Equivalent series resistance	ESR		50	80	kohm	PVT

11.12 RC oscillator characteristics

Table 11- 13 RC oscillator characteristics

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

Item	Sym.	Min	Typ.	Max	Unit	Condition
32MHz RC oscillator						
Nominal frequency	f_{NOM}		32		MHz	
Frequency tolerance	f_{TOL}		1	1.2	%	With calibration
32kHz RC oscillator						
Nominal frequency	f_{NOM}		32		kHz	
Frequency tolerance	f_{TOL}		0.03	0.15*[8]	%	With calibration
Calibration time			3		ms	

[8] Note: During Bluetooth LE operation, to keep the tolerance within the +/- 500ppm required by the standard specification, other than the hardware calibration, a dynamic firmware tracking algorithm is used to adjust the sleep time between connection interval anchor points. For details on the dynamic tracking, please refer to application note AN-18021100.

11.13 ADC characteristics

Table 11- 14 ADC characteristics

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

Item	Sym.	Min	Typ.	Max	Unit	Condition
Differential nonlinearity	DNL		20.6	28.3	LSB	1.428V reference
Integral nonlinearity	INL		21.4	28.6	LSB	1.428V reference
Signal-to-noise ratio (fin=628.66Hz, fS=100kHz)	SNR	53.7	57.3		dB	1.428V reference
Total harmonic distortion(fin=628.66Hz, fS=100kHz)	THD		-74.9	-71.2	dB	1.428V reference
Effective Number of Bits	ENOB	8.6	9.2		bits	1.428V reference
Sampling frequency	Fs			200	KHz	1.224V/1.428V reference
				250	KHz	AVDD reference

11.14 Inputs/Outputs Characteristics

Table 11- 15 Inputs/Outputs Characteristics

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

Item	Sym.	Min	Typ.	Max	Unit	Condition
Input high voltage	VIH	0.7VDD		VDD	V	
Input low voltage	VIL	VSS		0.3VDD	V	
Output high voltage	VOH	0.9VDD		VDD	V	
Output low voltage	VOL	VSS		0.1VDD	V	

12 Application

12.1 Schematic

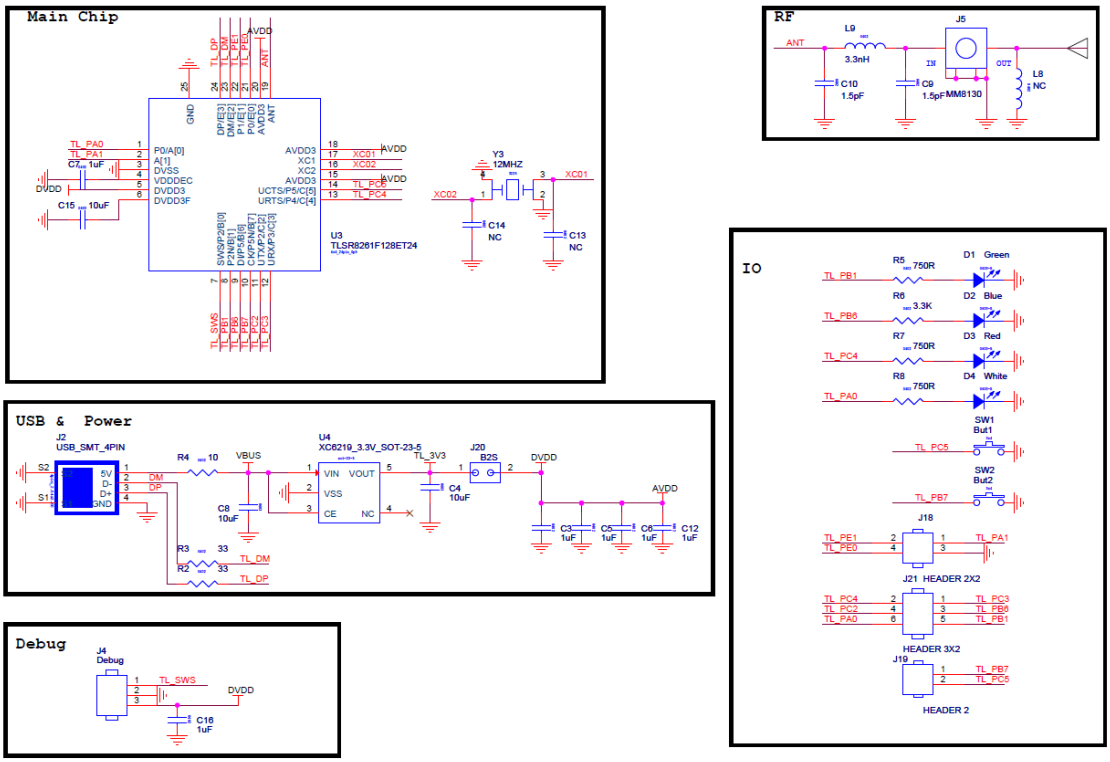


Figure 12- 1Schematic for the TLSR8261F128

12.2 Layout

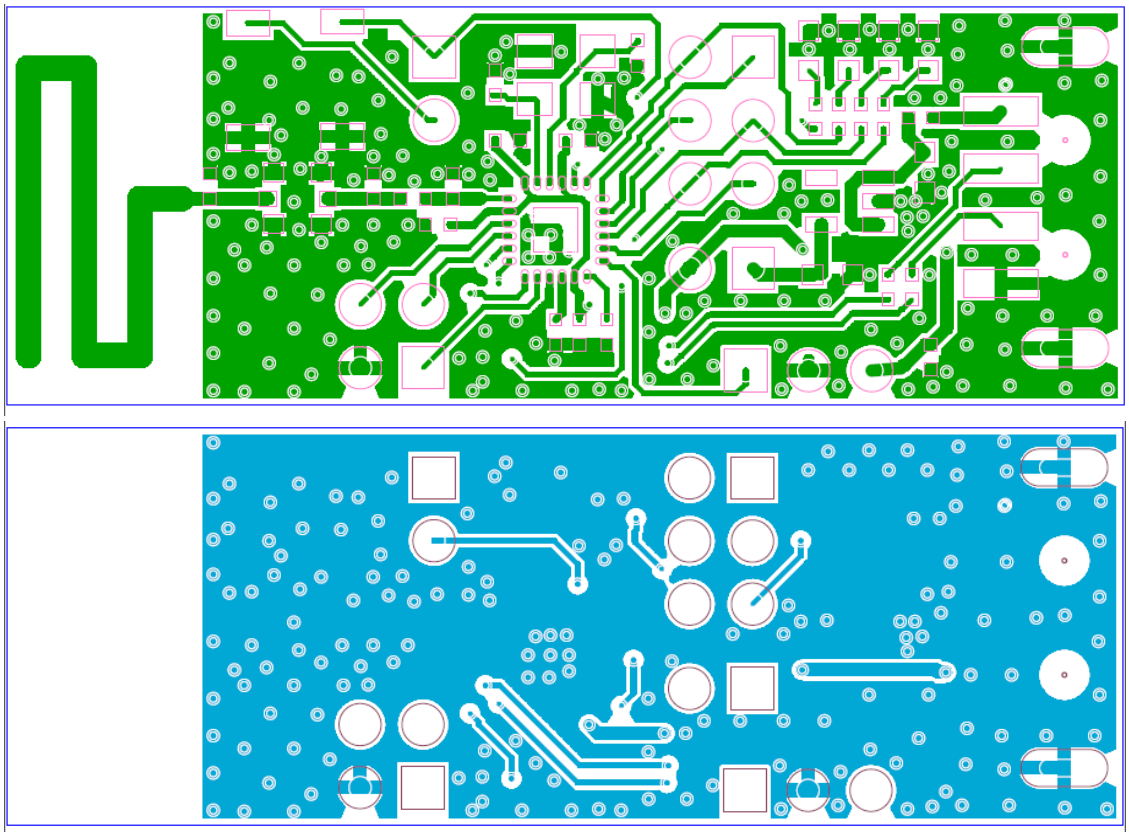


Figure 12- 2 Layout for the TLSR8261F128

(Up: Top view; Down: Bottom view)

12.3 BOM (Bill of Material)

Table 12- 1 BOM table for the TLSR8261F128

Quantity	Reference	Value	PCB Footprint	Description
6	C3	1uF	0402	capacitor
	C5	1uF	0402	capacitor
	C6	1uF	0402	capacitor
	C7	1uF	0402	capacitor
	C12	1uF	0402	capacitor
	C16	1uF	0402	capacitor
3	C4	10uF	0603C	capacitor
	C8	10uF	0603C	capacitor
	C15	10uF	0603C	capacitor

Quantity	Reference	Value	PCB Footprint	Description
2	C9	1.5pF	0402	capacitor
	C10	1.5pF	0402	capacitor
2	C13	NC	0402	capacitor
	C14	NC	0402	capacitor
1	D1	Green	0603-LED	LED
1	D2	Blue	0603-LED	LED
1	D3	Red	0603-LED	LED
1	D4	White	0603-LED	LED
1	J2	USB_SMT_4PIN	usb-m-4	USB connector
1	J4	Debug	hdr254m-1x3	header
1	J5	MM8130	MM8130	RF connector
1	J18	HEADER 2X2	HDR254F-2X2X850	header
1	J19	HEADER 2	HDR254F-1X2X850	header
1	J20	B2S	HDR254F-1X2X850	header
1	J21	HEADER 3X2	HDR254F-2X3X850	header
1	L8	NC	0402	inductor
1	L9	3.3nH	0402	inductor
2	R2	33	0402	resistor
	R3	33	0402	resistor
1	R4	10	0402	resistor
3	R5	750R	0402	resistor
	R7	750R	0402	resistor
	R8	750R	0402	resistor
1	R6	3.3K	0402	resistor
1	SW1	But1	butsmd2px4_00y3_00_nh	button
1	SW2	But2	butsmd2px4_00y3_00_nh	button
1	U3	TLSR8261F128ET24	qfn_4x4_24pin_0p5_1p8x1p8	SoC_RF
1	U4	XC6219_3.3V_SOT -23-5	sot-23-5	LDO
1	Y3	12MHZ - 12pf - +/-20ppm	OSCCC250X320X110	Crystal