Datasheet for Telink 2.4GHz RF System-On-Chip Solution TLSR8368

DS-TLSR8368-E20

Ver 2.8.0

2018/5/25

Keyword:

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

Brief:

This datasheet is dedicated for Telink 2.4GHz RF System-On-Chip (SoC) Solution TLSR8368 / TLSR8368E02 (with 2Kbit EEPROM). In this datasheet, key features, working mode, main modules, electrical specification and application 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	Initial release	2015/3	S.G.J., Y.C.Q, F.F., L.Y., Cynthia
1.1	Updated section 10	2015/3	L.X.Z., Cynthia
1.1.1	Updated section 1	2015/5	Cynthia
1.2.0	Updated package dimension figure (Figure 1-2 in Section 1.5), section 1.2.2~1.2.3, 12.2~12.4	2015/6	X.S.J., L.Y., Cynthia
1.3.0	Updated GP4&GP5 pin multiplexed function (I2C) in section 1, 8 and 13.	2015/7	S.G.J., L.X., L.Y., Cynthia
1.4.0	Added IO drive strength information	2015/7	L.Y., Cynthia
1.5.0	Added TSSOP16 package information	2015/9	X.S.J., L.Y., Cynthia
1.6.0	Added QFN24 and SOP16 package information	2015/11	X.S.J., L.Y., Cynthia
1.7.0	Updated connection relationship between GPIO and related modules.	2016/2	S.G.J., Cynthia
1.8.0	Updated DC characteristics and ordering information.	2016/7	L.J.R., X.S.J., Cynthia
1.9.0	Updated package dimension for the TLSR8368ES16	2016/7	X.S.J., Cynthia
2.0.0	Deleted TLSR8368ES16 package	2016/9	X.S.J., Cynthia
2.1.0	Deleted Telink SDK section	2016/10	L.X., Cynthia
2.2.0	Updated register configuration to select V_{GP23} or 1/3* V_{GP23} as ADC input: digital register 0x2c[2:0], and analog register afe3V_reg02<3>. Updated reference design.	2016/10	Z.J.Q., W.W.X., Cynthia
2.3.0	Added TLSR8368E02EP16 package. Added EEPROM introduction.	2016/11	L.X.Z., X.S.J., Cynthia

Version	Major Changes	Date	Author
2.4.0	Updated 3.3V analog register table (3v_reg12 ~ 3v_reg45)	2016/12	C.K.X., Cynthia
2.5.0	Updated ordering information.	2017/3	X.S.J., Cynthia
2.6.0	Updated section 8.2.2 Telink I2C communication protocol.	2017/6	S.G.J., Cynthia
2.7.0	Updated the following sections: 5.1.3 HS divider clock (0x66[4:0]), 8.1.1.1 Multiplexed functions, 1.2.2 RF Features and 13.4 AC characteristics.	2018/1	Y.C.Q., X.W.W., L.W.F., L.Y., Cynthia
2.8.0	Updated the sections below: 1.4 Ordering information, 1.6 Pin layout	2018/5	Xiashijin, Yanghuilin, Cynthia

1 Table of contents

2 Table of Figures

3 Table of Tables

1 Overview

As one member of the low-power, high-integration family of Telink wireless SoC solution, the TLSR8368/TLSR8368E02 is dedicated to 2.4GHz RF System-On-Chip solution, such as wireless keyboard, non-audio remote control applications, etc. It's completely RoHS-compliant and 100% lead (Pb)-free.

1.1 Block diagram

The TLSR8368/TLSR8368E02 is designed to offer high integration, ultra-low power application capabilities. It integrates an advanced 2.4GHz RF transceiver, a powerful 32-bit MCU, 16KB on-chip OTP, 6KB on-chip SRAM, 2Kbit EEPROM (only for TLSR8368E02), a 10bit ADC, a quadrature decoder (QDEC), four-channel PWM, abundant I/O interfaces, and nearly all of the peripheral blocks needed to construct a powerful 2.4GHz RF System-On-Chip solution.

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

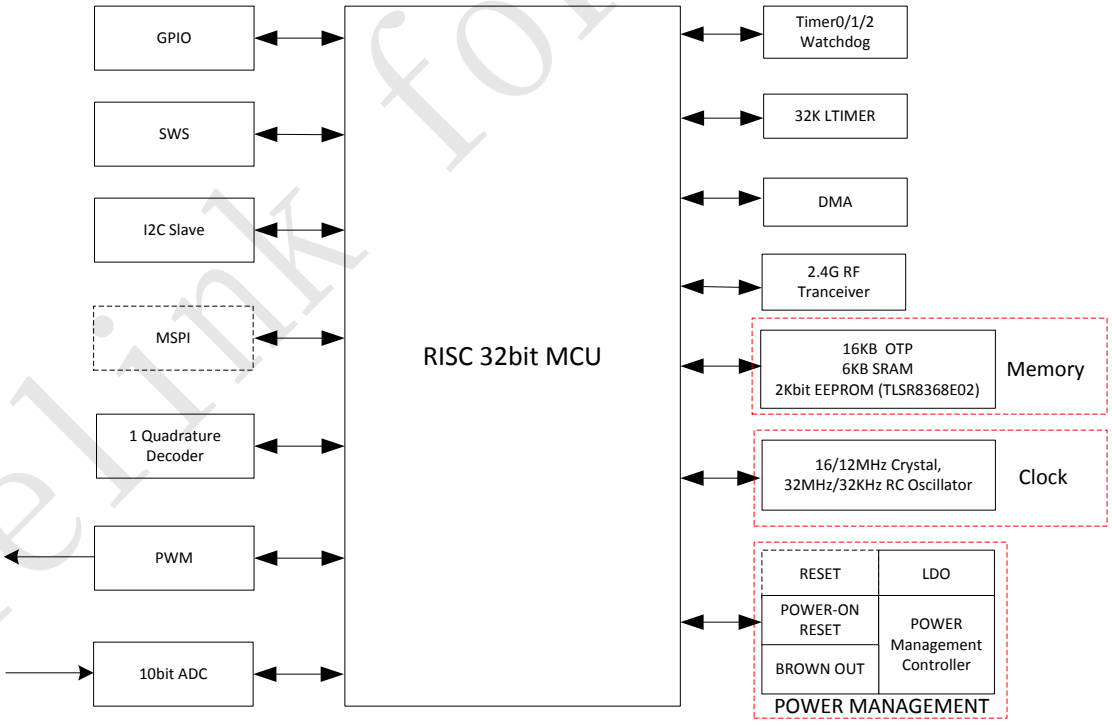


Figure 1- 1 Block diagram of the system

Based on the TLSR8368/TLSR8368E02 with high-volume-assembly and high integration, few external components are needed to satisfy customers' ultra-low cost requirement.

1.2 Key features

1.2.1 General features

General features are as follows:

1) Embed 32-bit high performance MCU with clock up to 48MHz.
2) Program memory: 16KB on-chip OTP.
3) Data memory: 6KB on-chip SRAM.
4) 2Kbit EEPROM (Electrically erasable and programmable ROM) embedded in TLSR8368E02.
5) 16/12MHz Crystal and 32KHz/32MHz embedded RC oscillator.
6) Abundant GPIO interfaces: Up to 38/14/9 GPIOs depending on package option, with configurable internal pull-up or pull-down resistors.
7) Debug interface: SWS.
8) Supports MSPI interface (only for TLSR8368ET48 & TLSR8368ET24) and I2C Slave.
9) Embeds a SAR ADC: Up to 10bit resolution and 4 input channels.
10) Embeds one quadrature decoder (QDEC).
11) Supports four-channel PWM output.
12) Embeds three general 32-bit timers Timer0~Timer2.
    - ✧ Timer0~Timer2 are available in active mode
    - ✧ Timer0~Timer1 supports four modes
    - ✧ Generally Timer2 is programmable as watchdog
13) A low-frequency 32K timer LTIMER available in suspend mode or deep sleep mode.
14) Operating temperature: -40℃~+85℃ industrial temperature range.

1.2.2 RF Features

RF features include:

1) 2.4GHz RF transceiver embedded, working in worldwide 2.4GHz ISM band.
2) Adaptive frequency hopping.
3) RF link data rate: Configurable as 2Mbps or 250Kbps.
4) Rx Sensitivity: -88dBm at 2Mbps mode, -97dBm at 250Kbps mode.
5) Tx output power: +6dBm.
6) Auto acknowledgement and retry.
7) Single-pin antenna interface.
8) RSSI monitoring.

1.2.3 Features of power management module

Features of power management module include:

1) Power supply of 1.9V~3.6V.
2) Embedded LDO.
3) Battery monitor: Embedded low battery detection.
4) Multiple stage power management to minimize power consumption.
5) Low power consumption:
   - ✧ Transmitter mode current: 15mA @ 0dBm power, 22mA @ 6dBm power
   - ✧ Receiver mode current: 12mA
   - ✧ Suspend mode current: 10uA
   - ✧ Deep sleep mode current: 0.7uA

1.3 Typical application

Typical applications for the TLSR8368/TLSR8368E02 are as follows:

- ✧ Wireless keyboard
- ✧ Non-audio remote control

1.4 Ordering information

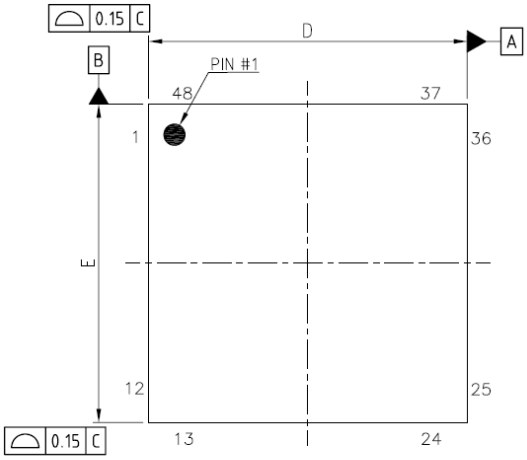
Table 1- 1 Ordering information of the TLSR8368

Product Series	Package Type	Temperature Range	Product Part No.	Packing Method	Minimum Order Quantity
TLSR8368	48-pin 7X7mm TQFN	-40℃ ~ +85℃	TLSR8368ET48	TR	3000
	24-pin 4X4mm TQFN	-40℃ ~ +85℃	TLSR8368ET24	TR	3000
	16-pin SOP16L_ 10X6 mm	-40℃ ~ +85℃	TLSR8368EP16	Tube	5000
	Dice	-40℃ ~ +85℃	TLSR8368ED	Tray	80000
TLSR8368E02	16-pin SOP16L_ 10X6 mm	-40℃ ~ +85℃	TLSR8368E02 EP16	Tube	5000

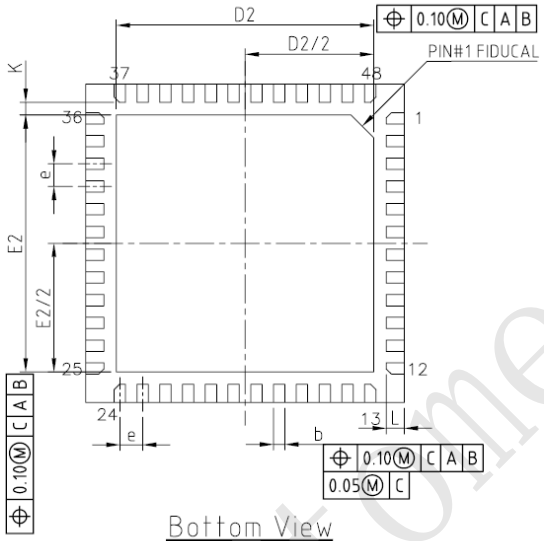
*Note: Packing method “TR” means tape and reel.

1.5 Package

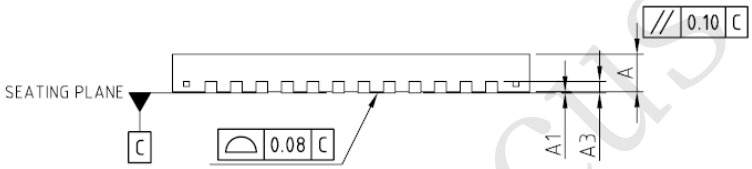
Package dimensions for the TLSR8368ET48, TLSR8368ET24 and TLSR8368EP16/ TLSR8368E02EP16 are shown as Figure 1- 2, Figure 1- 3 and Figure 1- 4, respectively.



Top View



Bottom View

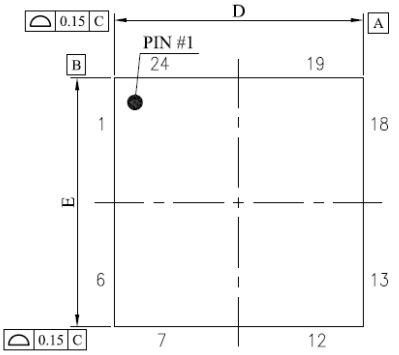


Side View

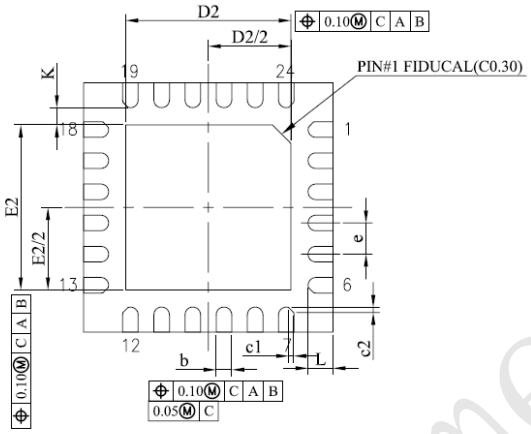
SYMBOL	DIMENSION (MM)			DIMENSION (MIL)		
	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.
A	0.70	0.75	0.80	27.6	29.5	31.5
A1	0	0.02	0.05	0	0.79	1.97
A3	0.20 REF			7.9 REF		
b	0.18	0.25	0.30	7.1	9.8	11.8
D	6.90	7.00	7.10	271.7	275.6	279.5
D2	5.60	5.65	5.70	220.5	222.4	224.4
E	6.90	7.00	7.10	271.7	275.6	279.5
E2	5.60	5.65	5.70	220.5	222.4	224.4
e	0.50 BSC			19.7BSC		
K	0.20	--	--	7.9	--	--
L	0.35	0.40	0.45	13.8	15.7	17.7

NOTE:
1. DIMENSIONING AND TOLERANCING CONFORM TO ASME Y14.5M-1994.
2. REFER TO JEDEC STD. MO-220 WKKD-4.
3. DIMENSION "b" APPLIES TO METALLIZED TERMINAL AND IS MEASURED BETWEEN 0.18 AND 0.30mm FROM TERMINAL TIP.
4. LEADFRAME MATERIAL IS 194FH AND THICKNESS IS 0.203MM (8 MIL).
5. DIMENSION"D"&"E" WILL INCLUDE ALL SIDE BURR INDUCED DURING ASSEMBLY.

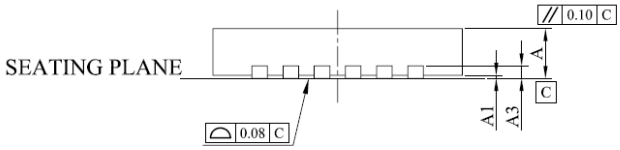
Figure 1- 2 Package dimension for the TLSR8368ET48 (Unit: mm)



Top View



Bottom View

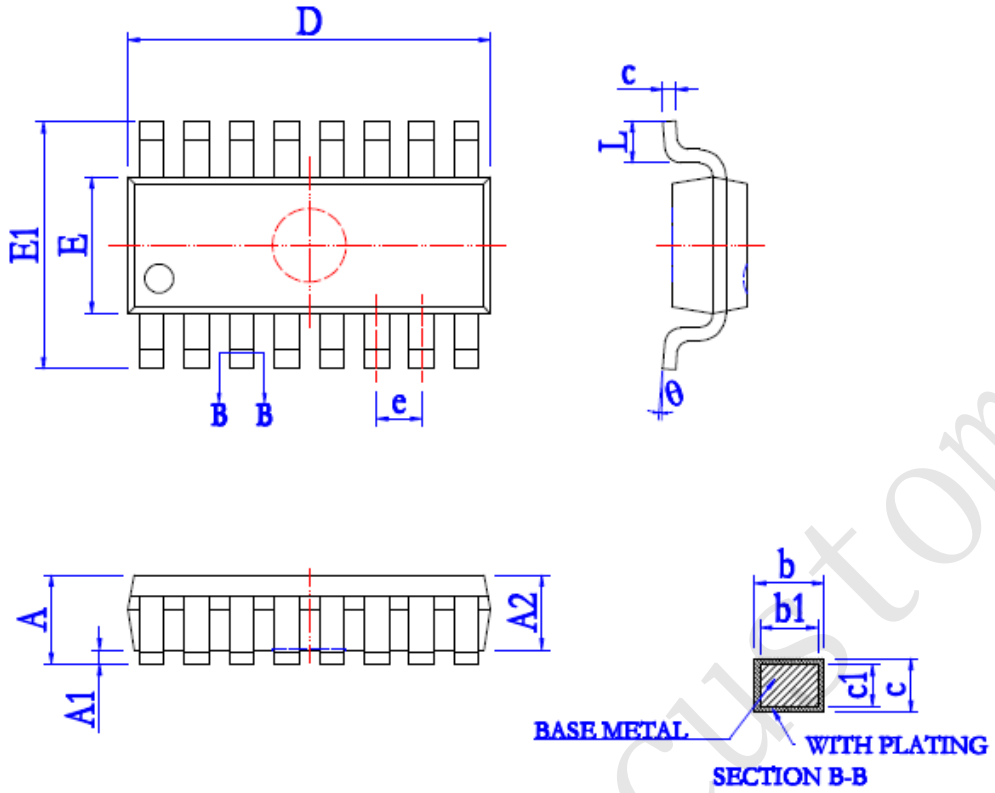


Side View

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- 3 Package dimension for the TLSR8368ET24 (Unit: mm)



SYMBOL	MILIMETER	
	MIN	MAX
A	1.350	1.750
A1	0.100	0.250
A2	1.350	1.550
b	0.330	0.510
b1	0.320	0.500
c	0.170	0.250
c1	0.160	0.240
D	9.800	10.200
E	3.800	4.000
E1	5.800	6.200
e	1.270BSC	
L	0.400	0.800
θ	0°	8°
L/F Carrier Dimension (mil)	134*91	

Figure 1- 4 Package dimension for the TLSR8368EP16/TLSR8368E02EP16

(Unit: mm)

1.6 Pin layout

Pin assignment for the TLSR8368ET48 is as shown in Figure 1-5:

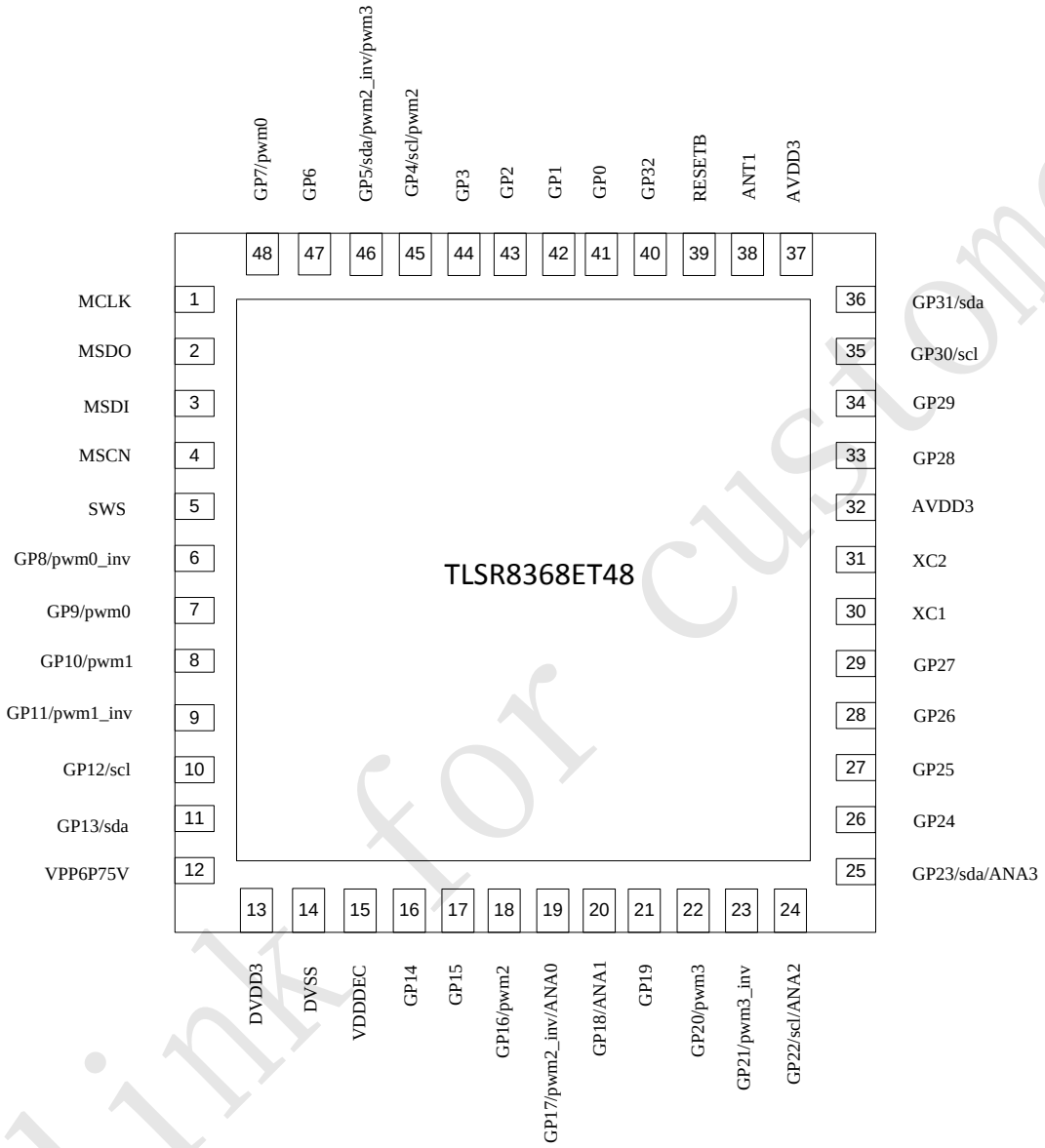


Figure 1- 5 Pin assignment for the TLSR8368ET48

Functions of 48 pins for the TLSR8368ET48 are described in Table 1-2:

Table 1- 2 Pin functions for the TLSR8368ET48

QFN48 7X7			
No.	**Pin Name**	**Pin Type**	**Description**
1	**MCLK**	Digital I/O	Memory SPI clock/GPIO
2	**MSDO**	Digital I/O	Memory SPI data output/GPIO
3	**MSDI**	Digital I/O	Memory SPI data input/GPIO
4	**MSCN**	Digital I/O	Memory SPI chip-select(Active low)/GPIO
5	**SWS**	Digital I/O	single wire slave/GPIO
6	**GP8/pwm0_inv #**	Digital I/O	GPIO8/PWM0 inverting output
7	**GP9/pwm0 #**	Digital I/O	GPIO9/PWM0 output
8	**GP10/pwm1 #**	Digital I/O	GPIO10/PWM1 output
9	**GP11/pwm1_inv #**	Digital I/O	GPIO11/PWM1 inverting output
10	**GP12/scl #**	Digital I/O	GPIO12/I2C_SCL
11	**GP13/sda #**	Digital I/O	GPIO13/I2C_SDA
12	VPP6P75V	POWER	for OTP program 6.75V power supply
13	DVDD3	PWR	3.3V IO supply
14	DVSS	GND	Digital LDO ground
15	VDDDEC	PWR	Digital LDO 1.8V output
16	**GP14 #**	Digital I/O	GPIO14
17	**GP15 #**	Digital I/O	GPIO15
18	**GP16/pwm2 #**	Digital I/O	GPIO16/PWM2 output
19	**GP17/pwm2_inv/ANA0 ***	Digital I/O	GPIO17/PWM2 inverting output/Analog input 0 for SAR ADC
20	**GP18/ANA1 ***	Digital I/O	GPIO18/Analog input 1 for SAR ADC
21	**GP19 ***	Digital I/O	GPIO19
22	**GP20/pwm3 ***	Digital I/O	GPIO20/PWM3 output
23	**GP21/pwm3_inv ***	Digital I/O	GPIO21/PWM3 inverting output
24	**GP22/scl/ANA2 ***	Digital I/O	GPIO22/I2C_SCL/Analog input 2 for SAR ADC

QFN48 7X7			
No.	**Pin Name**	**Pin Type**	**Description**
25	**GP23/sda/ANA3 ***	Digital I/O	GPIO23/I2C_SDA/Analog input 3 for SAR ADC
26	**GP24 ***	Digital I/O	GPIO24
27	**GP25**	Digital I/O	GPIO25
28	**GP26 ***	Digital I/O	GPIO26
29	**GP27 ***	Digital I/O	GPIO27
30	XC1	Analog I/O	16MHz crystal input+
31	XC2	Analog I/O	16MHz crystal input-
32	AVDD3	PWR	Analog 3.3V supply
33	**GP28**	Digital I/O	GPIO28
34	**GP29**	Digital I/O	GPIO29
35	**GP30/scl**	Digital I/O	GPIO30/I2C_SCL
36	**GP31/sda ***	Digital I/O	GPIO31/I2C_SDA
37	AVDD3	PWR	RF 3.3V supply
38	ANT1	Analog I/O	RF antenna
39	RESETB	Digital I	Power on reset, active low
40	**GP32 #**	Digital I/O	GPIO32
41	**GP0 #**	Digital I/O	GPIO0
42	**GP1 #**	Digital I/O	GPIO1
43	**GP2 #**	Digital I/O	GPIO2
44	**GP3 #**	Digital I/O	GPIO3
45	**GP4/scl/pwm2 #**	Digital I/O	GPIO4/I2C_SCL/PWM2 output
46	**GP5/sda/pwm2_inv/ pwm3 #**	Digital I/O	GPIO5/I2C_SDA/PWM2 inverting output/PWM3 output
47	**GP6 #**	Digital I/O	GPIO6
48	**GP7/pwm0 #**	Digital I/O	GPIO7/PWM0 output

*Note:

(1) Pins with bold typeface can be used as GPIOS. Please refer to **Section 8.1** GPIO for details.

(2) The pins marked with an asterisk support configurable internal 1MΩ/10KΩ pull-up resistor or 100KΩ pull-down resistor which are disabled by default. The pins marked with a pound sign support internal 100KΩ pull-down resistor which is disabled by default. Please refer to **Section 8.4** Pull-up/Pull-down resistor for details about pull-up/pull-down resistor.

(3) The I2C interface only supports Slave mode.

(4) Pin drive strength: All the GPIO pins support drive strength of 4mA or 0.7mA (4mA when "DS"=1, 0.7mA when "DS"=0) with the following exceptions: MCLK, MSDO, MSDI and MSCN support drive strength of 4mA or 2mA (4mA when "DS"=1, 2mA when "DS"=0); SWS supports drive strength of 8mA or 4mA (8mA when "DS"=1, 4mA when "DS"=0). "DS" configuration will take effect when the pin is used as output. Please refer to **Section 8.1** GPIO for corresponding "DS" register address and the default setting.

Pin assignment for the TLSR8368ET24 is as shown in Figure 1- 6:

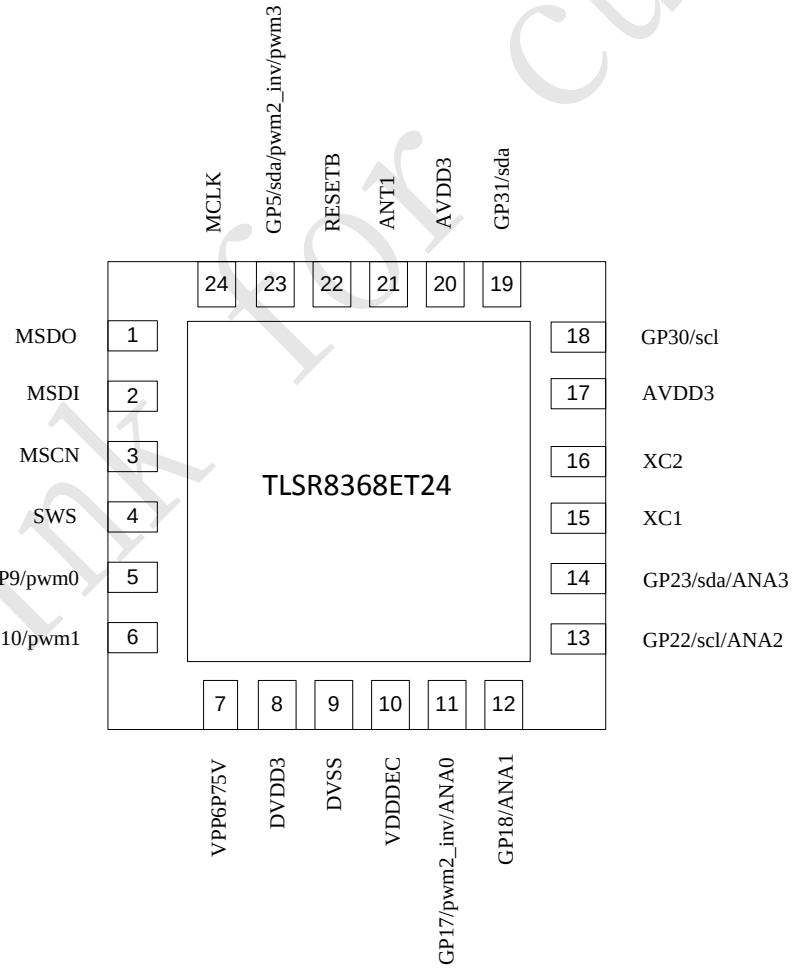


Figure 1- 6 Pin assignment for the TLSR8368ET24

Functions of 24 pins for the TLSR8368ET24 are described in Table 1-3:

Table 1- 3 Pin functions for the TLSR8368ET24

QFN24 4X4			
No.	**Pin Name**	**Pin Type**	**Description**
1	**MSDO**	Digital I/O	Memory SPI data output/GPIO
2	**MSDI**	Digital I/O	Memory SPI data input/GPIO
3	**MSCN**	Digital I/O	Memory SPI chip-select(Active low)/GPIO
4	**SWS**	Digital I/O	single wire slave/GPIO
5	**GP9/pwm0 #**	Digital I/O	GPIO9/PWM0 output
6	**GP10/pwm1 #**	Digital I/O	GPIO10/PWM1 output
7	VPP6P75V	POWER	for OTP program 6.75V power supply
8	DVDD3	PWR	3.3V IO supply
9	DVSS	GND	Digital LDO ground
10	VDDDEC	PWR	Digital LDO 1.8V output
11	**GP17/pwm2_inv/ANA0 ***	Digital I/O	GPIO17/PWM2 inverting output/Analog input 0 for SAR ADC
12	**GP18/ANA1 ***	Digital I/O	GPIO18/Analog input 1 for SAR ADC
13	**GP22/scl/ANA2 ***	Digital I/O	GPIO22/I2C_SCL/Analog input 2 for SAR ADC
14	**GP23/sda/ANA3 ***	Digital I/O	GPIO23/I2C_SDA/Analog input 3 for SAR ADC
15	XC1	Analog I/O	16MHz crystal input+
16	XC2	Analog I/O	16MHz crystal input-
17	AVDD3	PWR	Analog 3.3V supply
18	**GP30/scl**	Digital I/O	GPIO30/I2C_SCL
19	**GP31/sda ***	Digital I/O	GPIO31/I2C_SDA
20	AVDD3	PWR	RF 3.3V supply
21	ANT1	Analog I/O	RF antenna
22	RESETB	Digital I	Power on reset, active low
23	**GP5/sda/pwm2_inv/**	Digital I/O	GPIO5/I2C_SDA (not

QFN24 4X4			
No.	**Pin Name**	**Pin Type**	**Description**
	pwm3 #		recommended)/PWM2 inverting output/PWM3 output
24	**MCLK**	Digital I/O	Memory SPI clock/GPIO

*Note:

(1) Pins with bold typeface can be used as GPIOS. Please refer to **Section 8.1** for details.

(2) The pins marked with an asterisk support configurable internal 1MΩ/10KΩ pull-up resistor or 100KΩ pull-down resistor which are disabled by default. The pins marked with a pound sign support internal 100KΩ pull-down resistor which is disabled by default. Please refer to **Section 8.4** for details about pull-up/pull-down resistor.

(3) The I2C interface only supports Slave mode.

(4) Pin drive strength: All the GPIO pins support drive strength of 4mA or 0.7mA (4mA when “DS”=1, 0.7mA when “DS”=0) with the following exceptions: MCLK, MSDO, MSDI and MSCN support drive strength of 4mA or 2mA (4mA when “DS”=1, 2mA when “DS”=0); SWS supports drive strength of 8mA or 4mA (8mA when “DS”=1, 4mA when “DS”=0). “DS” configuration will take effect when the pin is used as output. Please refer to **Section 8.1** GPIO for corresponding “DS” register address and the default setting.

Pin assignment for the TLSR8368EP16 is as shown in Figure 1- 7:

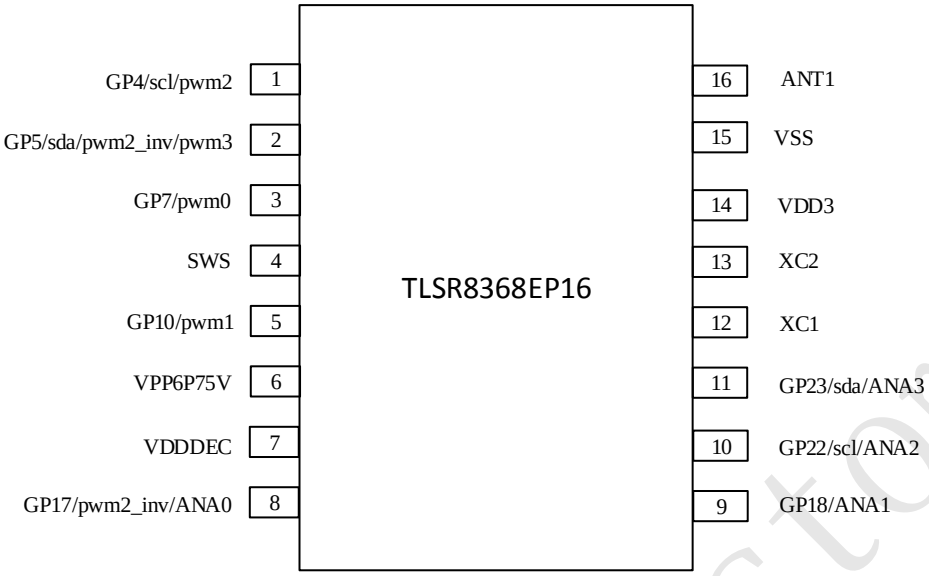


Figure 1- 7 Pin assignment for the TLSR8368EP16

Functions of 16 pins for the TLSR8368EP16 are described in Table 1-4:

Table 1- 4 Pin functions for the TLSR8368EP16

SOP16L_10X6			
No.	**Pin Name**	**Pin Type**	**Description**
1	**GP4/scl/pwm2 #**	Digital I/O	GPIO4/I2C_SCL/PWM2 output
2	**GP5/sda/pwm2_inv/ pwm3 #**	Digital I/O	GPIO5/I2C_SDA/PWM2 inverting output/PWM3 output
3	**GP7/pwm0 #**	Digital I/O	GPIO7/PWM0 output
4	**SWS**	Digital I/O	single wire slave/GPIO
5	**GP10/pwm1 #**	Digital I/O	GPIO10/PWM1 output
6	VPP6P75V	POWER	for OTP program 6.75V power supply
7	VDDDEC	PWR	Digital LDO 1.8V output
8	**GP17/pwm2_inv/ANA0 ***	Digital I/O	GPIO17/PWM2 inverting output/Analog input 0 for SAR ADC
9	**GP18/ANA1 ***	Digital I/O	GPIO18/Analog input 1 for SAR ADC
10	**GP22/scl/ANA2 ***	Digital I/O	GPIO22/I2C_SCL/Analog input 2 for SAR ADC
11	**GP23/sda/ANA3 ***	Digital I/O	GPIO23/I2C_SDA/Analog input 3 for SAR ADC

SOP16L_10X6			
No.	**Pin Name**	**Pin Type**	**Description**
12	XC1	Analog I/O	16MHz crystal input+
13	XC2	Analog I/O	16MHz crystal input-
14	VDD3	PWR	3.3V supply
15	VSS	GND	ground for the whole chip
16	ANT1	Analog I/O	RF antenna

*Note:

(1) Pins with bold typeface can be used as GPIOS. Please refer to **Section 8.1** for details.

(2) The pins marked with an asterisk support configurable internal 1MΩ/10KΩ pull-up resistor or 100KΩ pull-down resistor which are disabled by default. The pins marked with a pound sign support internal 100KΩ pull-down resistor which is disabled by default. Please refer to **Section 8.4** for details about pull-up/pull-down resistor.

(3) The I2C interface only supports Slave mode.

(4) Pin drive strength: All the GPIO pins support drive strength of 4mA or 0.7mA (4mA when “DS”=1, 0.7mA when “DS”=0) with the following exception: SWS supports drive strength of 8mA or 4mA (8mA when “DS”=1, 4mA when “DS”=0). “DS” configuration will take effect when the pin is used as output. Please refer to **Section 8.1** GPIO for corresponding “DS” register address and the default setting.

Pin assignment for the TLSR8368E02EP16 is as shown in Figure 1- 8:

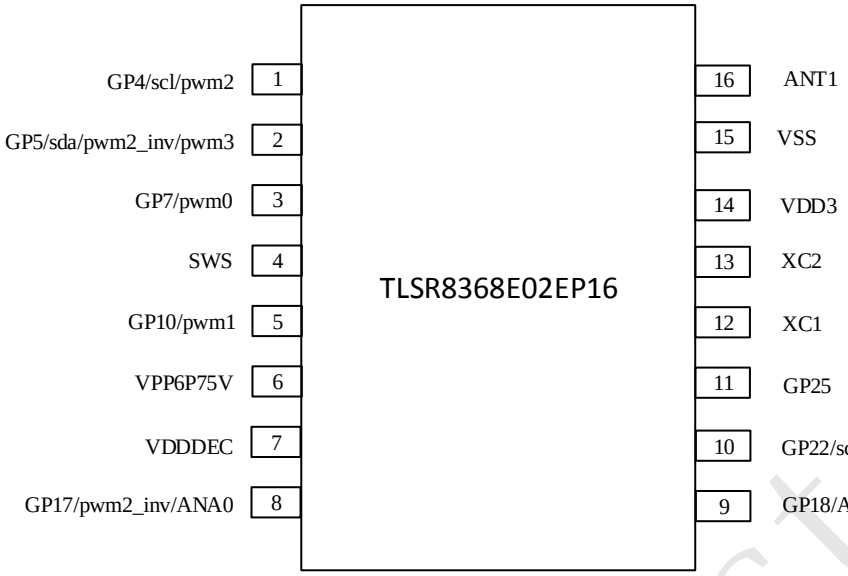


Figure 1- 8 Pin assignment for the TLSR8368E02EP16

Functions of 16 pins for the TLSR8368E02EP16 are described in Table 1- 5:

Table 1- 5 Pin functions for the TLSR8368E02EP16

SOP16L_10X6			
No.	**Pin Name**	**Pin Type**	**Description**
1	**GP4/scl/pwm2 #**	Digital I/O	GPIO4/I2C_SCL/PWM2 output
2	**GP5/sda/pwm2_inv/ pwm3 #**	Digital I/O	GPIO5/I2C_SDA/PWM2 inverting output/PWM3 output
3	**GP7/pwm0 #**	Digital I/O	GPIO7/PWM0 output
4	**SWS**	Digital I/O	single wire slave/GPIO
5	**GP10/pwm1 #**	Digital I/O	GPIO10/PWM1 output
6	VPP6P75V	POWER	for OTP program 6.75V power supply
7	VDDDEC	PWR	Digital LDO 1.8V output
8	**GP17/pwm2_inv/ANA0 ***	Digital I/O	GPIO17/PWM2 inverting output/Analog input 0 for SAR ADC
9	**GP18/ANA1 ***	Digital I/O	GPIO18/Analog input 1 for SAR ADC
10	**GP22/scl/ANA2 ***	Digital I/O	GPIO22/I2C_SCL/Analog input 2 for SAR ADC
11	**GP25**	Digital I/O	GPIO25
12	XC1	Analog I/O	16MHz crystal input+

SOP16L_10X6			
No.	**Pin Name**	**Pin Type**	**Description**
13	XC2	Analog I/O	16MHz crystal input-
14	VDD3	PWR	3.3V supply
15	VSS	GND	ground for the whole chip
16	ANT1	Analog I/O	RF antenna

*Note:

(1) Pins with bold typeface can be used as GPIOS. Please refer to **Section 8.1** for details.

(2) The pins marked with an asterisk support configurable internal 1MΩ/10KΩ pull-up resistor or 100KΩ pull-down resistor which are disabled by default. The pins marked with a pound sign support internal 100KΩ pull-down resistor which is disabled by default. Please refer to **Section 8.4** for details about pull-up/pull-down resistor.

(3) The I2C interface only supports Slave mode.

(4) Pin drive strength: All the GPIO pins support drive strength of 4mA or 0.7mA (4mA when “DS”=1, 0.7mA when “DS”=0) with the following exception: SWS supports drive strength of 8mA or 4mA (8mA when “DS”=1, 4mA when “DS”=0). “DS” configuration will take effect when the pin is used as output. Please refer to **Section 8.1** GPIO for corresponding “DS” register address and the default setting.

2 Memory

The TLSR8368/TLSR8368E02 embeds 6KB data memory (SRAM), and 16KB program memory (OTP). The TLSR8368E02 also embeds 2Kbit EEPROM, which is internally organized as 32 pages with 8 bytes each.

SRAM/Register memory map is shown as follows:

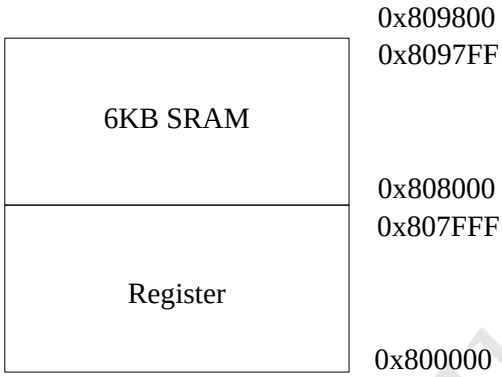


Figure 2- 1 Physical memory map

Register address: from 0x800000 to 0x807FFF;

6KB SRAM address: from 0x808000 to 0x809800.

Both register and 6KB SRAM address can be accessed via SWS interface.

OTP/External flash address mapping is configurable.

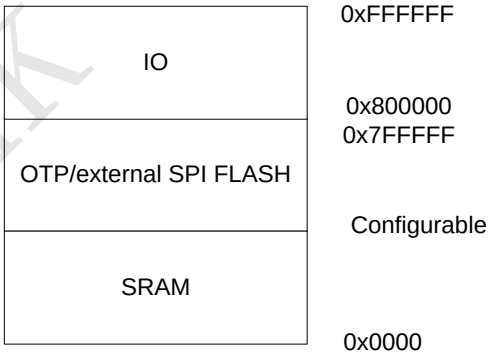


Figure 2- 2 MCU memory map

External FLASH address can be accessed via MSPI interface.

Address space starting from 0x800000 can be accessed via debug interface.

As for the EEPROM of the TLSR8368E02, please refer to **Section 12 EEPROM**.

3 MCU

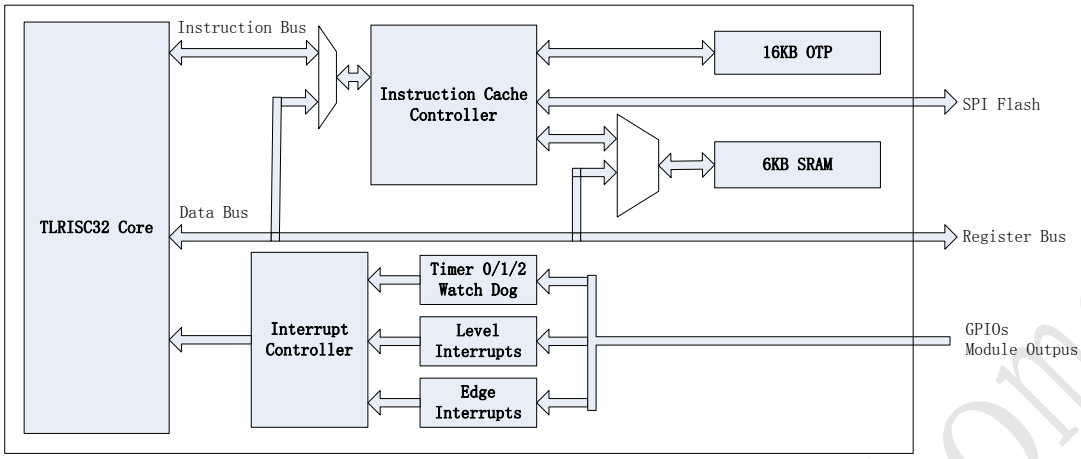


Figure 3- 1 Block diagram

The TLSR8368/TLSR8368E02 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.

3.1 Working modes

The TLSR8368/TLSR8368E02 has four working modes: Active, Idle, Suspend and Deep Sleep. This section mainly gives the description of every working mode and mode transition.

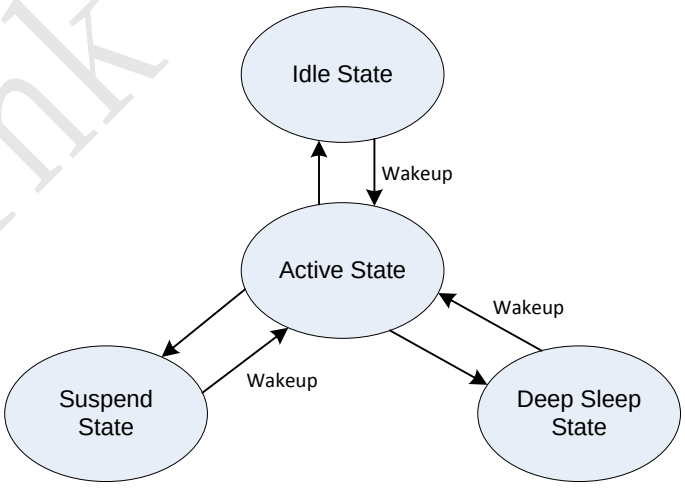


Figure 3- 2 Transition chart of working modes

3.1.1 Active mode

In active mode, the MCU block is at working state, and the TLSR8368/TLSR8368E02 can transmit or receive data via its embedded RF transceiver. The RF transceiver can also be powered down if no data transfer is needed.

3.1.2 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.

3.1.3 Power-saving mode

3.1.3.1 Brief introduction

For the TLSR8368/TLSR8368E02, 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 TLSR8368/TLSR8368E02 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.

3.1.3.2 Register configuration of power-saving mode

For the TLSR8368/TLSR8368E02, power-saving mode related registers are configurable via digital core and 3.3V analog registers.

Table 3- 1 Registers in digital core

Address	Mnemonic	Type	Description	Reset value
0x6e	WAKEUPEN	R/W	Wakeup enable [0]: enable i2c wakeup when transaction [1]: enable QDEC wakeup [2]: rsvd [3]: enable wakeup from gpio [4]: enable i2c wakeup when slave ID matched System resume control [5]: enable GPIO remote wakeup [6]: rsvd [7] sleep wakeup reset system enable	00
0x6f	PWDNEN	W	[0]: suspend enable [5]: rst all (act as power on reset) [6]: mcu low power mode [7]: stall mcu trig If bit[0] set 1, then system will go to suspend. Or only stall mcu	

Address 0x6e serves to enable various wakeup sources from power-saving mode. Please refer to **Section 3.1.3.3**Wakeup source for details.

Table 3- 2 3.3V analog registers (afe3V_reg05 ~ afe3V_reg06) (bit)

Address(bit)	Mnemonic	Reset 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>	reserved	0	
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

Address(bit)	Mnemonic	Reset value	Description
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>	reserved	1	
afe3V_reg05<6>	reserved	1	
afe3V_reg05<7>	BBPLL_LDO_pd_3V	0	Power down baseband pll LDO 1: Power down 0: Power up
afe3V_reg06<0>	comp_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

Address(bit)	Mnemonic	Reset value	Description
			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 anaog circuit ldo 1: Power down 0: power up

Table 3- 3 3.3V analog registers (3v_reg12 ~ 3v_reg45)

	Addr	Name	Description
r12	0x0c	buffer	this buffer will be reset when watch dog reset or whole chip reset(address 0x6f write 0x20)
r13	0x0d	buffer	this buffer will be reset when watch dog reset or whole chip reset(address 0x6f write 0x20)
r14	0x0e	buffer	this buffer will be reset when watch dog reset or whole chip reset(address 0x6f write 0x20)
r15	0x0f	32ktimer_cnt[7:0]	32ktimer cnt[0] = 1 means 4 cycles of 32k
r16	0x10	32ktimer_cnt[15:8]	
r17	0x11	32ktimer_cnt[23:16]	
r18	0x12[0]	32ktimer_cnt[24]	
	0x12[1]	32k timer enable	32k timer enable
r19	0x13[6:0]	r_dly	[6:0]wakeup or power on delay for digital LDO is ready.
	0x13[7]	rsvd	
r20	0x14[2:0]	wd_v	32k watch dog value
	0x14[3]	wd_en	32k watch dog enable
	0x14[7:4]	pad_pol[3:0]	pad polarity, one bit control two pad wakeup polarity. pad_pol[4] control pad_wakeup_en[1:0].
r21	0x15[3:0]	xtl_quick	xtl quick settle 0xf means disable

	Addr	Name	Description
	0x15[5:4]	wd_wkup_src	watch dog wake up source select[4]; dig wakeup source enable watch dog.[5],pad wakeup source enable watch dog
	0x15[7:6]	rsvd	
r22	0x16	pad_wakeup_en	[7:0]-->p_gpio[24:17].[0]-->p_gpio[17]
r23	0x17[2:0]	wakeup_en	[0]-> digital wakeup enable[1]-->32k timer wakeup enable,[2] pad wake up enable
	0x17[3]	32k timer reset	
	0x17[4]	rsvd	
	0x17[5]	32k timer clock select	0:32k osc, 1 16M xtl
	0x17[6]	rsvd	
	0x17[7]	rsvd	
r24	0x18[0]	pwdn_auto_en	auto pd 32k osc enable
	0x18[1]	rsvd	
	0x18[2]	pwdn_auto_en	auto pd 16m xtal enable
	0x18[3]	pwdn_auto_en	auto pd ldo_ana,BBPLL_ldo,sar_adc,rx_lnaLDO, rx_anan_ldo,rx_rfLDO,pll_bg, pll_vco_ldo,pll_cp_ldo
	0x18[4]	pwdn_en	power down sequence enable
	0x18[5]	pd_llkldo	pd low leakage ldo
	0x18[6]	pd_ldo_en	pd digital ldo enable
	0x18[7]	iso_en	isolation enable
r25	0x19	buffer	this buffer will be reset only at power on
r26	0x1a	buffer	this buffer will be reset only at power on
r27	0x1b	buffer	this buffer will be reset only at power on
r28	0x1c	buffer	this buffer will be reset only at power on
	0x1d~0x1f	rsvd	
r32	0x20	read only	32k timer_cnt[7:0](1 cycle of 32k clock will change the reslult)
r33	0x21	read only	32k timer_cnt[15:8]
r34	0x22	read only	32k timer_cnt[23:16]
r35	0x23[0]	rsvd	
	0x23[1]	w/r	write 1 to clean timer wakeup status.
	0x23[2]	w/r	write 1 to clean digital wakeup status
	0x23[3]	w/r	write 1 to clean pad wakeup status
	0x23[4]	wd_status	write 1 to clean watch dog status.
	0x23[5]	read only	rsvd
	0x23[6]	read only	32k timer_cnt[24]
	0x23[7]	w/r	32k timer enable toggle signal, write 1 to enable 32k timer
r36	0x24[0]	trk32m manul en	
	0x24[1]	trk32m en	

	Addr	Name	Description
	0x24[2]	trk32k manul en	
	0x24[3]	trk32k en	
	0x24[4]	mode_12m	1, 12M xtal, 0: 16M xtal
	0x24[7:3]	rsvd	
r37	0x25	trk32m_m_cap	
r38	0x26[6:0]	trk32k_m_cap	
	0x26[7]	rsvd	
r39	0x27	rsvd	
r40	0x28		
r41	0x29		
r42	0x2a		
r43	0x2b[2:0]	pad_wakeup en[10:8]	[0]->p_gpio[26],[1]->p_gpio[27],[2]->p_gpio[31]
	0x2b[3]	rsvd	
	0x2b[6:4]	pad_pol[6:4]	pad_pol[6:4]^pad_wakeup_en[10:8]
r44		32M rc cap value	32M calibration read only
r45		32k rc cap value	32k calibration read only

3.1.3.3 Wakeup source

3.1.3.3.1 Wakeup source – GPIO

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

First, set the right polarity of IO (0x584, 0x58c, 0x594, 0x59c, 0x5a4). Polarity 1 indicates corresponding IO is active low, while 0 indicates corresponding IO is active high.

Second, set the right mask (0x587, 0x58f, 0x597, 0x59f, 0x5a7). 1: enable this IO as wakeup source; 0: disable this IO.

Third, set both the digital core address 0x6e bit[3] and 3v_reg23 bit[0] to 1 so as to activate this mode.

Please refer to **Section 8.1** GPIO for details about polarity and mask registers of each GPIO.

3.1.3.3.2 Wakeup source – QDEC

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

First, digital core address 0x6e[1] should be set to 1b'1.

Second, address 3V_reg23 bit[0] should be set to 1b'1.

Third, addresses afe3V_reg05<0> and 3V_reg24 [0] should be cleared to power up 32K RC clock, then write 0x64[7] and 0x65[0] to 1.

After this wakeup source is enabled, once there's wheel rolling, square waves output are generated and the system is wakened.

3.1.3.3.3 Wakeup source – 32K timer

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

Address 3V_reg23 bit[1] is the enabling bit for wakeup source from 32k timer.

3.1.3.3.4 Wakeup source – pad

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

3v_reg23[2] should be set to 1b'1 to enable pad wakeup source.

3v_reg22 and 3v_reg43[2:0] are enabling signal for pad wakeup sources: 3v_reg22 bit[7:0] -> [GP24~GP17]; 3v_reg43 bit[2:0] -> [GP31, GP27, GP26]. 1: enable this IO as wakeup source; 0: disable this IO.

Polarity is controlled by 3v_reg20[7:4] and 3v_reg43[6:4]: 3v_reg43[6] controls polarity of GP31, bit[5] controls polarity of GP27, bit[4] controls polarity of GP26; 3v_reg20 bit[4] controls polarity of GP17 and GP18, bit[5] controls polarity of GP19 and GP20, bit[6] controls polarity of GP21 and GP22, bit[7] controls polarity of GP23 and GP24. Polarity 1 indicates corresponding IO is active low, while 0 indicates corresponding IO is active high.

3.1.3.4 Transition sequence

First, enable the target wakeup source, and disable other wakeup sources.

NOTE: In deep sleep mode, the wakeup_dig (including wakeup source-QDEC and

wakeup source-GPIO, shown as Figure 3-3) can't be selected as wakeup source; the effective wakeup source is 32K timer or pad wakeup source.

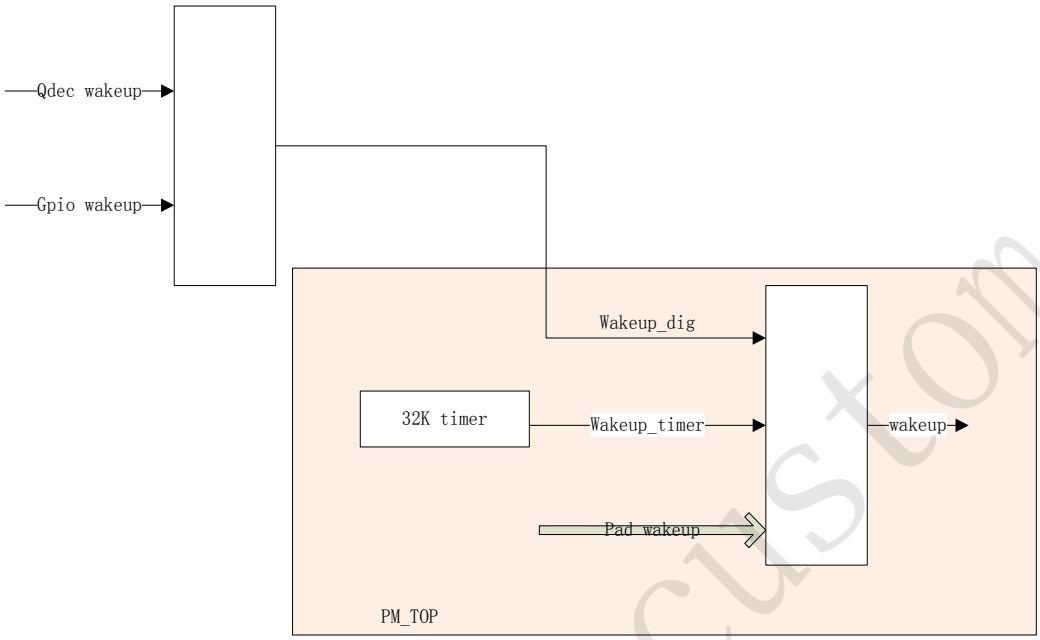


Figure 3- 3 Wakeup source

Second, select right power-saving mode: deep sleep mode or suspend mode. If deep sleep mode is to be selected, r24 bit[7] and bit[5] should be set to 1; r24 bit[7] and bit[5] should be cleared if suspend mode is to be selected.

Third, configure other enabling bits: set r23 bit[7] to 1; set r24 bits [3:0] to 1111, and also set r24 bit[6] to 1.

Fourth, Write data 0x81 to digital core address 0x6f to trigger the whole system. The system enters deep sleep mode or suspend mode (power-saving status depends on the setting of r24).

3.2 Reset

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

Table 3- 4 Register configuration for reset, wakeup and power down enabling

Address	Mnemonic	Type	Description	Reset Value
0x60	RST0	R/W	Reset control, 1 for reset, 0 for clear [0]: mcu [1]: zb [2]: rsvd [3]: dma [4]: algm [5]: sws [6]: aif [7]: rsvd	00
0x61	RST1	R/W	[0] rsvd [1]i2c [2]rsvd [3]pwm [4]rsvd [5]rsvd [6]mspi [7]bbpll	df
0x62	RST2	R/W	[0]adc [1]algs [2]mcic [3]mcic auto reset at suspend [4]systimer [5]rsvd [6]rsvd [7]rsvd	00
0x6f	PWDNEN	W	[0] suspend enable [5]:rst all (act as power on reset) [6]:mcu low power mode [7]: stall mcu trig If bit[0] set 1, then system will go to suspend. Or only stall mcu	

4 2.4G RF Transceiver

4.1 Block diagrams

The TLSR8368/TLSR8368E02 integrates an advanced 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 works in proprietary 2Mbps mode or 250Kbps mode. All modes support FSK/GFSK modulations.

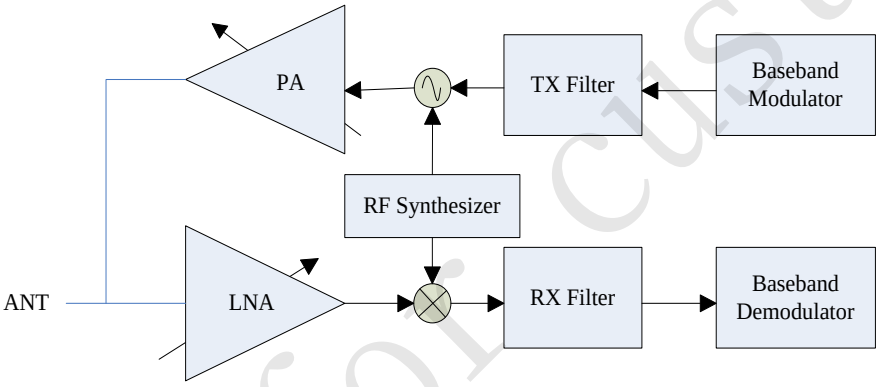


Figure 4- 1 Block diagram of RF transceiver

The internal PA can deliver a typical 6dBm output power, avoiding the needs for an external RF PA.

4.2 Function description

Air interface data rate, the modulated signaling rate for RF transceiver when transmitting and receiving data, supports 2Mbps and 250Kbps mode for the TLSR8368/TLSR8368E02.

For the TLSR8368/TLSR8368E02, RF transceiver can operate with frequency ranging from 2.400GHz to 2.4835GHz. The RF channel frequency setting determines the center of the channel.

4.3 Baseband

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

The baseband supports all features required by 2Mbps/250Kbps specification.

4.3.1 Packet format

Packet format in 2Mbps mode is shown as Table 4-1:

Table 4- 1 Packet Format in 2Mbps mode

LSB MSB

Preamble (1 octet)	Access Address (Configurable: 3, 4 or 5 octets)	PDU (Configurable: 0 to 63 octets)	CRC (Configurable: 1 octet or 2 octets)

Packet length 40bits ~ 568bits (20~284us @ 2Mbps).

Format of PPDU (presentation protocol data unit) in 250Kbps mode is shown as Table 4-2:

Table 4- 2 Packet Format in 250Kbps mode

		Octets	
		1	Variable
Preamble	SFD	Frame length (8 bits)	PSDU
SHR		PHR	PHY payload

The size of PSDU is 1~127 bytes.

4.3.2 RSSI

The TLSR8368/TLSR8368E02 provides accurate RSSI (Receiver Signal Strength Indicator) indication which can be read on per packet basis.

5 Clock

5.1 System clock

5.1.1 System clock sources

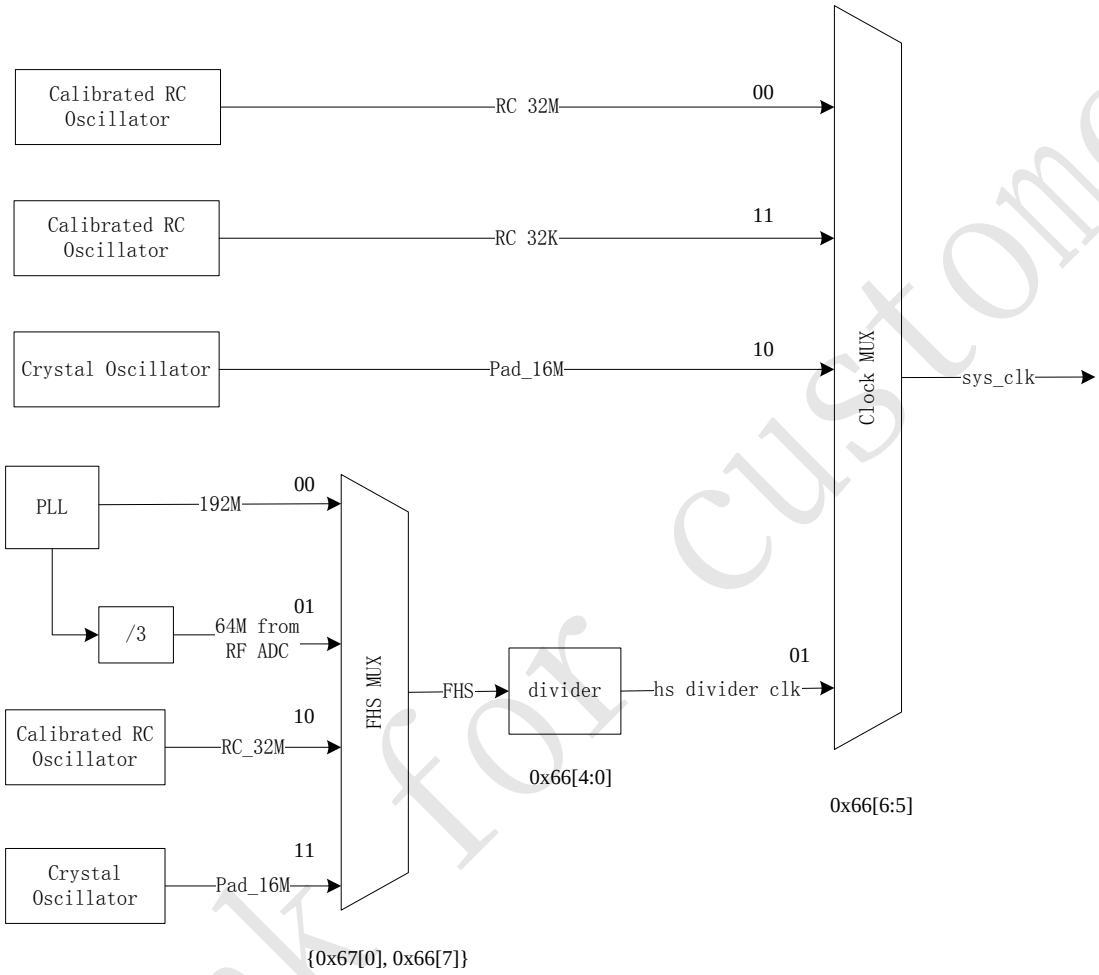


Figure 5- 1 Block diagram of system clock

There are four selectable clock sources for system clock, including: 32MHz RC oscillator, hs divider clock, 16MHz pad clock (external crystal oscillator) and 32KHz RC oscillator. Register CLKSEL (address 0x66[6:5]) is used to select system clock source.

Commonly a 16MHz crystal oscillator can be employed to generate a basic clock signal for the system. The maximum frequency tolerance of the crystal is ±60ppm. And a low-power RC oscillator can be used to generate a 32KHz clock signal for the wakeup timer.

5.1.2 FHS select

The high speed clock (FHS) is selectable via address {0x67[0], 0x66[7]} from the following sources: 192MHz clock from PLL, 64MHz clock from RF ADC, 32MHz clock from RC oscillator or 16MHz pad clock (external crystal oscillator).

5.1.3 HS divider clock

If address 0x66[6:5] is set to 2b'01 to select the HS divider clock as system clock source, system clock frequency is adjustable via address 0x66[4:0]. $F_{System\ clock}$ = F_{FHS} / (system clock divider value in address 0x66[4:0]).

5.2 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.

5.2.1 SAR ADC clock

ADC clock derives from FHS. Address 0x6b[7] should be set to 1b'1 to enable ADC clock.

ADC clock frequency dividing factor contains step and mod.

Addresses 0x6b[6:4] and 0x69[7:0] serve to configure ADC_step[10:0].

Addresses 0x6b[3:0] and 0x6a[7:0] serve to configure ADC_mod[11:0].

ADC clock is calculated according to the formula below:

$$F_{ADC\ clock} = F_{FHS} * adc\_step[10:0]/adc\_mod[11:0]$$

5.3 Register table

Table 5- 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] : mcu [1] : zb [2]: rsvd [3]: dma [4]: algm [5]: sws [6]: aif [7]: rsvd	8c
0x64	CLKEN1	R/W	[0]rsvd [1]i2c [2]rsvd [3]pwm [4]rsvd [5]rsvd [6]sys timer [7]qdec sysclk	00
0x65	CLKEN2	R/W	[0]32k for qdec [1]rsvd [2]rsvd [3]rsvd [4]rsvd [5]rsvd [6]rsvd [7]rsvd	00
0x66	CLKSEL	R/W	System clock select [4:0]: system clock divider: fhs/(CLKSEL[4:0]). Fhs refer to {0x67[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 rc [7] FHS sel (see 0x67 definition)	ff
0x67	FHS_sel	R/W	{0x67[0],0x66[7]} fhs select	00

Address	Mnemonic	Type	Description	Reset Value
			2'b00: 192M clock from pll 2'b01:64M 2'b10:32M clock from osc 2'b11:16M clock from pad	
0x68	rsvd	R/W		
0x69	Adc step[7:0]	R/W	adc_step[7:0]	00
0x6a	Adc mod[7:0]	R/W	adc_mod[7:0]	2
0x6b	adcmodstep	R/W	[3:0] adc_mod[11:8] [7:4] adc_step[11:8], adc_step[11] is enable bit. adc_clk = clk_hs *adc_step[10:0] / adc_mod[11:0]	00
0x6c	DMIC_step	R/W	rsvd	1
0x6d	DMIC_mod	R/W	rsvd	2

6 Timers

6.1 Timer0~Timer2

The TLSR8368/TLSR8368E02 supports three general 32-bit timers including Timer0~ Timer2 in active mode.

Timer0 and Timer1 support four modes: Mode 0 (System Clock Mode), Mode 1 (GPIO Trigger Mode), Mode 2 (GPIO Pulse Width Mode) and Mode 3 (Tick Mode).

Timer2 only supports Mode0 and Mode3. Generally Timer 2 is configured as “watchdog” to monitor firmware running.

6.1.1 Register table

Table 6- 1 Register configuration for Timer0~Timer2

Address	Mnemonic	Type	Description	Reset Value
0x620	TMR_CTRL0	RW	[0]Timer0 enable [2:1] Timer0 mode. 0 using sclk, 1, using gpio, 2 count widht 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	

Address	Mnemonic	Type	Description	Reset Value
			[2] timer2 status, write 1 to clear [3] watch dog status, write 1 to clear	
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	

6.1.2 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.

6.1.3 Mode1 (GPIO Trigger Mode)

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

After Timer is enabled, Timer Tick (i.e. counting value) is increased by 1 on each positive/negative 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 8.1.2** for corresponding "m0", "m1" 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.

6.1.4 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" register specifies the GPIO which generates control signal for Timer0/Timer1.

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 8.1.2** for corresponding "m0", "m1" 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 Timer0 for Mode 2 is taken as an example.

1st: Set initial Timer0 Tick value

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: Select GPIO source and edge for Timer0

Select certain GPIO to be the clock source via setting "m0" register.

Select positive edge or negative edge of GPIO input to trigger Timer0 counting start via setting "Polarity" register.

3rd: Set Timer0 to Mode 2 and enable Timer0

Set address 0x620[2:1] to 2b'10 to select Mode 2; Meanwhile set address 0x620 [0] to 1b'1 to enable Timer0.

Timer0 Tick is triggered by a positive/negative (specified during the 2nd step) edge of GPIO pulse. Timer0 starts counting upward and Timer0 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 Timer0 tick stops.

4th: Read current Timer0 Tick value to calculate GPIO pulse width

Read current Timer0 Tick value from address 0x630~0x633.

Then GPIO pulse width is calculated as follows:

$$\text{GPIO pulse width} = \text{System clock period} * (\text{current Timer0 Tick} - \text{intial Timer0 Tick})$$

For initial Timer0 Tick value set to the recommended value of 0, then:

$$\text{GPIO pulse width} = \text{System clock period} * \text{current Timer0 Tick}.$$

6.1.5 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.

6.1.6 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.

6.2 32K LTIMER

The TLSR8368/TLSR8368E02 supports a low frequency (32KHz) LTIMER in suspend mode or deep sleep mode. This timer can be used as one kind of wakeup source.

Analog register 3V_reg35[7] should be set to 1b'1 to enable the LTIMER.

3V_reg16~3V_reg18[6:0] serve to configure timing value for the LTIMER with the unit of ms.

3V_reg18[7] serves to select mode for the LTIMER: continuous mode, or single mode. In continuous mode, when the LTIMER expires, the timing value is automatically reloaded, the counting value returns to zero and starts counting upwards again. In single mode, when the LTIEMR expires, it stops counting.

3V_reg23[5] serves to select clock source for the LTIMER: 32K RC oscillator, or 16M Pad clock.

Current counting value can be read from 3V_reg32~3V_reg34 and 3V_reg35[6].

Table 6- 2 3.3V analog register table for LTIMER

Addr (Decimal)	Addr (Hexadecimal)	Name	Description	Default value
r16	0x10	32ktimer_cnt[7:0]	32ktimer cnt[0] = 1 means 4 cycles of 32k	
r17	0x11	32ktimer_cnt[15:8]		
r18	0x12[6:0]	32ktimer_cnt[22:16]		
	0x12[7]	32ktimer mode	32k timer mode,1:	

Addr (Decimal)	Addr (Hexadecimal)	Name	Description	Default value
			continuous mode, 0: single mode	
r23	0x17[2:0]	wakeup_en	[0]-> digital wakeup enable[1]-->32k timer wakeup enable,[2] pad wake up enable	
	0x17[3]	32k timer reset		
	0x17[4]	rsvd		
	0x17[5]	32k timer clock select	0:32k osc, 1 16M xtl	
	0x17[6]	rsvd		
	0x17[7]	rsvd	power down sequence enable	
r32	0x20	read only	32k timer_cnt[7:0](1 cycle of 32k clock will change the reslult)	
r33	0x21	read only	32k timer_cnt[15:8]	
r34	0x22	read only	32k timer_cnt[23:16]	
r35	0x23[0]	rsvd		
	0x23[1]	w/r	write 1 to clean timer wakeup status.	
	0x23[2]	w/r	write 1 to clean digital wakeup status	
	0x23[3]	w/r	write 1 to clean pad wakeup status	
	0x23[4]	wd_status	write 1 to clean watch dog status.	
	0x23[5]	read only	rsvd	
	0x23[6]	read only	32k timer_cnt[24]	
	0x23[7]	w/r	32k timer enable toggle signal, write 1 to enable 32k timer	

6.3 System timer

The TLSR8368/TLSR8368E02 also supports a System Timer.

Table 6- 3 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
0x743	Sys_timer[31:24]	R/W	System timer counter, write to set initial value. This is the sys timer counter	00

7 Interrupt System

7.1 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 TLSR8368/TLSR8368E02, there are 24 interrupt sources in all: 16 types are level-triggered interrupt sources and 8 types are edge-triggered interrupt sources.

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.

7.2 Register configuration

Table 7- 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_qdec, rsvd, irq_pwm, irq_dma, rsvd, time2, time1, time0} [7]: irq_host_cmd | irq_qdec [6]: rsvd [5]: irq_pwm [4]: irq_dma [3]: rsvd [2]: time2 [1]: time1 [0]: time0	00
0x641	MASK_1	RW	Byte 1 interrupt mask, level-triggered type {an_irq, irq_software, irq_zb, rsvd, rsvd, rsvd, rsvd, rsvd} [7]: an_irq [6]: irq_software [5]: irq_zb [4]: rsvd [3]: rsvd	00

Address	Mnemonic	Type	Description	Reset Value
			[2]: rsvd [1]: rsvd [0]: rsvd	
0x642	MASK_2	RW	Byte 2 interrupt mask, edge-triggered type {gpio2risc[0], rsvd, rsvd, rsvd, pm_irq, irq_gpio, rsvd, rsvd} [7]: gpio2risc[0] [6]: rsvd [5]: rsvd [4]: rsvd [3]: pm_irq [2]: irq_gpio [1]: rsvd [0]: rsvd	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	

7.2.1 Enable/Mask interrupt sources

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

7.2.2 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.

7.2.3 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 irq_gpio for example: First enable the interrupt source by setting address 0x642[2] to 1; then set address 0x643 [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[18] is 1, it means the irq_gpio interrupt is valid. Clear this interrupt source by setting address 0x64a bit[2] 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 6.1.1**).

8 Interface

8.1 GPIO

The TLSR8368ET48, TLSR8368ET24 and TLSR8368EP16/TLSR8368E02EP16 supports up to 38, 14 and 9 GPIOs. Except for dedicated GPIOs, all digital IOs can be used as general purpose IOs. Please refer to **Section 1.6** for available GPIO resources.

8.1.1 Basic configuration

8.1.1.1 Multiplexed functions

Please refer to Table 8-1 for various GPIO interface configuration.

Table 8- 1 GPIO lookup table 1

Pin Name	Default Function	Priority0	Priority1	Priority2	Act as GPIO	Act as GPIO			Input Enable	DS (Drive Strength)
						OEN	Input	Output		
MCLK	MCLK				5a6[2]	5a2[2]	5a0[2]	5a3[2]	5a1[2]	5a5[2]
MSDO	MSDO				5a6[3]	5a2[3]	5a0[3]	5a3[3]	5a1[3]	5a5[3]
MSDI	MSDI				5a6[4]	5a2[4]	5a0[4]	5a3[4]	5a1[4]	5a5[4]
MSCN	MSCN				5a6[1]	5a2[1]	5a0[1]	5a3[1]	5a1[1]	5a5[1]
SWS	SWS				5a6[5]	5a2[5]	5a0[5]	5a3[5]	5a1[5]	5a5[5]
GP8/ pwm0_inv	GPIO input				58e[0]	58a[0]	588[0]	58b[0]	589[0]	58d[0]
GP9/ pwm0	GPIO input				58e[1]	58a[1]	588[1]	58b[1]	589[1]	58d[1]
GP10/ pwm1	GPIO input				58e[2]	58a[2]	588[2]	58b[2]	589[2]	58d[2]
GP11/ pwm1_inv	GPIO input				58e[3]	58a[3]	588[3]	58b[3]	589[3]	58d[3]
GP12/ scl	GPIO input	scl			NA	58a[4]	588[4]	58b[4]	589[4]	58d[4]
GP13/	GPIO	sda			58e[5]	58a[5]	588[5]	58b[5]	589[5]	58d[5]

Pin Name	Default Function	Priority0	Priority1	Priority2	Act as GPIO	Act as GPIO			Input Enable	DS (Drive Strength)
						OEN	Input	Output		
sda	input									
GP14	GPIO input	clk32krc			58e[6]	58a[6]	588[6]	58b[6]	589[6]	58d[6]
GP15	GPIO input	clk32mrc			58e[7]	58a[7]	588[7]	58b[7]	589[7]	58d[7]
GP16/ pwm2	GPIO input	pwm2			596[0]	592[0]	590[0]	593[0]	591[0]	595[0]
GP17/ pwm2_inv/ ANA0	GPIO input	pwm2_inv			596[1]	592[1]	590[1]	593[1]	591[1]	595[1]
GP18/ ANA1	GPIO input				NA	592[2]	590[2]	593[2]	591[2]	595[2]
GP19	GPIO input				NA	592[3]	590[3]	593[3]	591[3]	595[3]
GP20/ pwm3	GPIO input				596[4]	592[4]	590[4]	593[4]	591[4]	595[4]
GP21/ pwm3_inv	GPIO input				596[5]	592[5]	590[5]	593[5]	591[5]	595[5]
GP22/ scl/ ANA2	GPIO input	scl			NA	592[6]	590[6]	593[6]	591[6]	595[6]
GP23/ sda/ ANA3	GPIO input	sda			596[7]	592[7]	590[7]	593[7]	591[7]	595[7]
GP24	GPIO input				NA	59a[0]	598[0]	59b[0]	599[0]	59d[0]
GP25	GPIO				NA	59a[1]	598[1]	59b[1]	599[1]	59d[1]

Pin Name	Default Function	Priority0	Priority1	Priority2	Act as GPIO	Act as GPIO			Input Enable	DS (Drive Strength)
						OEN	Input	Output		
	input									
GP26	GPIO input				NA	59a[2]	598[2]	59b[2]	599[2]	59d[2]
GP27	GPIO input				NA	59a[3]	598[3]	59b[3]	599[3]	59d[3]
GP28	GPIO input				NA	59a[4]	598[4]	59b[4]	599[4]	59d[4]
GP29	GPIO input				NA	59a[5]	598[5]	59b[5]	599[5]	59d[5]
GP30/ scl	GPIO input	scl			NA	59a[6]	598[6]	59b[6]	599[6]	59d[6]
GP31/ sda	GPIO input	sda			59e[7]	59a[7]	598[7]	59b[7]	599[7]	59d[7]
GP32	GPIO input				NA	5a2[0]	5a0[0]	5a3[0]	5a1[0]	5a5[0]
GP0	GPIO input	an_rxadc_dat			586[0]	582[0]	580[0]	583[0]	581[0]	585[0]
GP1	GPIO input	an_rxclk			586[1]	582[1]	580[1]	583[1]	581[1]	585[1]
GP2	GPIO input				NA	582[2]	580[2]	583[2]	581[2]	585[2]
GP3	GPIO input				NA	582[3]	580[3]	583[3]	581[3]	585[3]
GP4/ scl/ pwm2	GPIO input	5d4[0] =1 scl	pwm2		586[4]	582[4]	580[4]	583[4]	581[4]	585[4]
GP5/	GPIO	5d4[0] =1 sda	5d4[1]=1	pwm2_inv	586[5]	582[5]	580[5]	583[5]	581[5]	585[5]

Pin Name	Default Function	Priority0	Priority1	Priority2	Act as GPIO	Act as GPIO			Input Enable	DS (Drive Strength)
						OEN	Input	Output		
sda/ pwm2_inv/ pwm3	input		pwm3							
GP6	GPIO input				NA	582[6]	580[6]	583[6]	581[6]	585[6]
GP7/pwm0	GPIO input	pwm0			586[7]	582[7]	580[7]	583[7]	581[7]	585[7]

***Notes**:

(1) OEN: active low. 0: output enable.

(2) Input Enable: active high. 1: input enable.

(3) NA: no configuration.

(4) Priority0 > Priority1 > Priority2.

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

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

The pins including GP0~GP32 are used as GPIO input function by default. For a pin with multiplexed function(s), to enable the function with lower priority, other function(s) with higher priority should be disabled first.

Take the **MCLK** as an example:

(1) This pin acts as MCLK function by default.

(2) To use the pin as GPIO function, address 0x5a6[2] should be set to 1b’1.

If the pin is used as output, its “OEN” register (address 0x5a2[2]) and “Input Enable” register (address 0x5a1[2]) should be cleared.

If the pin is used as input, its “OEN” register (address 0x5a2[2]) and “Input Enable” register (address 0x5a1[2]) should be set to 1b’1.

Take the **GP5/sda/pwm2_inv/pwm3** as another example:

(1) This pin acts as GPIO input by default.

(2) To use the pin as GPIO output, address 0x586[5] should be set to 1b'1, and addresses {0x582[5], 0x581[5]} should be cleared.

(3) To use the pin as pwm3 function, addresses {0x586[5], 0x5d4[0]} should be cleared, and 0x5d4[1] should be set to 1b'1.

(4) To use the pin as pwm2_inv function, addresses {0x586[5], 0x5d4[0], 0x5d4[1]} should be cleared.

(5) As for its I2C interface function, please refer to **Section 8.2.1** for details.

8.1.1.2 Drive strength

The registers in the "**DS**" column are used to configure 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.

As shown in Table 8-2, all the GPIO pins support maximum drive level of 4mA ("DS"=1) and minimal drive level of 0.7mA ("DS"=0) with the following exceptions:

- ✧ MCLK, MSDO, MSDI and MSCN: maximum=4mA ("DS"=1), minimum=2mA ("DS"=0);
- ✧ SWS: maximum=8mA ("DS"=1), minimum=4mA ("DS"=0).

Table 8- 2 IO drive strength

No.	Pin Name	Drive Strength	
		"DS"=0	"DS"=1
1	**MCLK**	2mA	4mA
2	**MSDO**	2mA	4mA
3	**MSDI**	2mA	4mA
4	**MSCN**	2mA	4mA
5	**SWS**	4mA	8mA
6	**GP8/pwm0_inv #**	0.7mA	4mA
7	**GP9/pwm0 #**	0.7mA	4mA

No.	Pin Name	Drive Strength	
		"DS"=0	"DS"=1
8	**GP10/pwm1 #**	0.7mA	4mA
9	**GP11/pwm1_inv #**	0.7mA	4mA
10	**GP12/scl #**	0.7mA	4mA
11	**GP13/sda #**	0.7mA	4mA
16	**GP14 #**	0.7mA	4mA
17	**GP15 #**	0.7mA	4mA
18	**GP16/pwm2 #**	0.7mA	4mA
19	**GP17/pwm2_inv/ANA0 ***	0.7mA	4mA
20	**GP18/ANA1 ***	0.7mA	4mA
21	**GP19 ***	0.7mA	4mA
22	**GP20/pwm3 ***	0.7mA	4mA
23	**GP21/pwm3_inv ***	0.7mA	4mA
24	**GP22/scl/ANA2 ***	0.7mA	4mA
25	**GP23/sda/ANA3 ***	0.7mA	4mA
26	**GP24 ***	0.7mA	4mA
27	**GP25**	0.7mA	4mA
28	**GP26 ***	0.7mA	4mA
29	**GP27 ***	0.7mA	4mA
33	**GP28**	0.7mA	4mA
34	**GP29**	0.7mA	4mA
35	**GP30/scl**	0.7mA	4mA
36	**GP31/sda ***	0.7mA	4mA
40	**GP32 #**	0.7mA	4mA
41	**GP0 #**	0.7mA	4mA
42	**GP1 #**	0.7mA	4mA
43	**GP2 #**	0.7mA	4mA
44	**GP3 #**	0.7mA	4mA
45	**GP4/scl/pwm2 #**	0.7mA	4mA
46	**GP5/sda/pwm2_inv/ pwm3 #**	0.7mA	4mA
47	**GP6 #**	0.7mA	4mA
48	**GP7/pwm0 #**	0.7mA	4mA

8.1.2 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”, and generates counting signal in Mode 1 or control signal in Mode 2 for Timer0/Timer1, 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);
GPIO2RISC[0] interrupt request signal = | ((input ^ polarity) & m0);

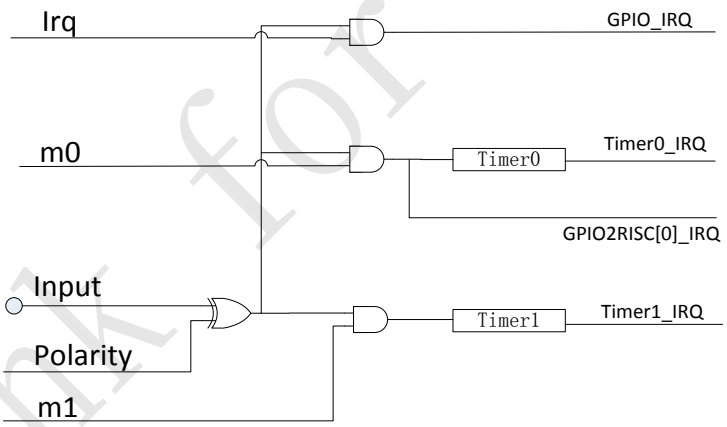


Figure 8- 1 Logic relationship between GPIO and related modules

Please refer to Table 8-3 and Table 7- 1 to learn how to configure GPIO for interrupt system or Timer0/Timer1 (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” register, and set corresponding GPIO interrupt enabling bit “Irq” register. Finally enable GPIO interrupt (irq_gpio at

address 0x642[2]).

User can read addresses {0x5c0 ~ 0x5c3, 0x5d0} to see which GPIO asserts GPIO interrupt request signal. **Note:** 0x5c0[0] --> GP0, 0x5c0[1] --> GP1, …… , 0x5c3[7] --> GP31, 0x5d0[0] --> GP32, 0x5d0[1] --> MSCN, 0x5d0[2] --> MCLK, 0x5d0[3] --> MSDO, 0x5d0[4] --> MSDI, 0x5d0[5] --> SWS, 0x5d0[7:6] --> 2b'0.

(3) Timer/Counter counting or control signal: Configure "Polarity" register (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" register.

User can read addresses {0x5c4~0x5c7, 0x5d1}/{0x5c8~0x5cb, 0x5d2} to see which GPIO asserts counting signal (in Mode 1) or control signal (in Mode 2) for Timer0/Timer1. **Note: Timer0**: 0x5c4[0] --> GP0, 0x5c4[1] --> GP1, …… , 0x5c7[7] --> GP31, 0x5d1[0] --> GP32, 0x5d1[1] --> MSCN, 0x5d1[2] --> MCLK, 0x5d1[3] --> MSDO, 0x5d1[4] --> MSDI, 0x5d1[5] --> SWS, 0x5d1[7:6] --> 2b'0; **Timer1**: 0x5c8[0] --> GP0, 0x5c8[1] --> GP1, …… , 0x5cb[7] --> GP31, 0x5d2[0] --> GP32, 0x5d2[1] --> MSCN, 0x5d2[2] --> MCLK, 0x5d2[3] --> MSDO, 0x5d2[4] --> MSDI, 0x5d2[5] --> SWS, 0x5d2[7:6] --> 2b'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". Enable GPIO2RISC[0] interrupt, i.e. "gpio2risc[0]" (address 0x642[7]).

User can read addresses {0x5c4~0x5c7, 0x5d1} to see which GPIO asserts GPIO2RISC[0] interrupt request signal. **Note:** 0x5c4[0] --> GP0, 0x5c4[1] --> GP1, …… , 0x5c7[7] --> GP31, 0x5d1[0] --> GP32, 0x5d1[1] --> MSCN, 0x5d1[2] --> MCLK, 0x5d1[3] --> MSDO, 0x5d1[4] --> MSDI, 0x5d1[5] --> SWS, 0x5d1[7:6] --> 2b'0.

Table 8- 3 GPIO lookup table 2

Pin	Input (R)	Polarity 1: active low 0: active high	Irq	m0	m1
MCLK	5a0[2]	5a4[2]	5a7[2]	5ac[2]	5b4[2]
MSDO	5a0[3]	5a4[3]	5a7[3]	5ac[3]	5b4[3]
MSDI	5a0[4]	5a4[4]	5a7[4]	5ac[4]	5b4[4]
MSCN	5a0[1]	5a4[1]	5a7[1]	5ac[1]	5b4[1]
SWS	5a0[5]	5a4[5]	5a7[5]	5ac[5]	5b4[5]
GP8/ pwm0_inv	588[0]	58c[0]	58f[0]	5a9[0]	5b1[0]
GP9/ pwm0	588[1]	58c[1]	58f[1]	5a9[1]	5b1[1]
GP10/ pwm1	588[2]	58c[2]	58f[2]	5a9[2]	5b1[2]
GP11/ pwm1_inv	588[3]	58c[3]	58f[3]	5a9[3]	5b1[3]
GP12/ scl	588[4]	58c[4]	58f[4]	5a9[4]	5b1[4]
GP13/ sda	588[5]	58c[5]	58f[5]	5a9[5]	5b1[5]
GP14	588[6]	58c[6]	58f[6]	5a9[6]	5b1[6]
GP15	588[7]	58c[7]	58f[7]	5a9[7]	5b1[7]
GP16/ pwm2	590[0]	594[0]	597[0]	5aa[0]	5b2[0]
GP17/ pwm2_inv/ ANA0	590[1]	594[1]	597[1]	5aa[1]	5b2[1]
GP18/ ANA1	590[2]	594[2]	597[2]	5aa[2]	5b2[2]
GP19	590[3]	594[3]	597[3]	5aa[3]	5b2[3]
GP20/ pwm3	590[4]	594[4]	597[4]	5aa[4]	5b2[4]
GP21/ pwm3_inv	590[5]	594[5]	597[5]	5aa[5]	5b2[5]

Pin	Input (R)	Polarity 1: active low 0: active high	Irq	m0	m1
GP22/ scl/ ANA2	590[6]	594[6]	597[6]	5aa[6]	5b2[6]
GP23/ sda/ ANA3	590[7]	594[7]	597[7]	5aa[7]	5b2[7]
GP24	598[0]	59c[0]	59f[0]	5ab[0]	5b3[0]
GP25	598[1]	59c[1]	59f[1]	5ab[1]	5b3[1]
GP26	598[2]	59c[2]	59f[2]	5ab[2]	5b3[2]
GP27	598[3]	59c[3]	59f[3]	5ab[3]	5b3[3]
GP28	598[4]	59c[4]	59f[4]	5ab[4]	5b3[4]
GP29	598[5]	59c[5]	59f[5]	5ab[5]	5b3[5]
GP30/ scl	598[6]	59c[6]	59f[6]	5ab[6]	5b3[6]
GP31/ sda	598[7]	59c[7]	59f[7]	5ab[7]	5b3[7]
GP32	5a0[0]	5a4[0]	5a7[0]	5ac[0]	5b4[0]
GP0	580[0]	584[0]	587[0]	5a8[0]	5b0[0]
GP1	580[1]	584[1]	587[1]	5a8[1]	5b0[1]
GP2	580[2]	584[2]	587[2]	5a8[2]	5b0[2]
GP3	580[3]	584[3]	587[3]	5a8[3]	5b0[3]
GP4/ scl/ pwm2	580[4]	584[4]	587[4]	5a8[4]	5b0[4]
GP5/ sda/ pwm2_inv/ pwm3	580[5]	584[5]	587[5]	5a8[5]	5b0[5]
GP6	580[6]	584[6]	587[6]	5a8[6]	5b0[6]

Pin	Input (R)	Polarity 1: active low 0: active high	Irq	m0	m1
GP7/pwm0	580[7]	584[7]	587[7]	5a8[7]	5b0[7]

8.2 I2C

The TLSR8368/TLSR8368E02 embeds I2C hardware module, which could only act as 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).

8.2.1 Pin configuration

Table 8-4 shows I2C interface configuration and priority:

Table 8- 4 I2C pin configuration

a) ~586[5] & 5d4[0]	GP4/GP5
b) ~58e[5]	GP12/GP13
c) ~596[7]	GP22/GP23
d) ~59e[7]	GP30/GP31
Priority: a) > b) > c) > d)	

For example, to use GP4 and GP5 as I2C_SCL and I2C_SDA respectively, address 0x586[5] should be cleared and 0x5d4[0] should be set to 1b'1.

To use GP12 and GP13 as I2C_SCL and I2C_SDA respectively, address 0x58e[5] should be cleared, meanwhile GP4 and GP5 should not be configured as I2C interface.

It's noted that the I2C_SCL pin must be configured as "input" via setting the corresponding "Input Enable" register to 1b'1.

8.2.2 Telink I2C 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 carry information between Master device and Slave device connected to the bus. Each device is recognized by unique address. 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. 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.

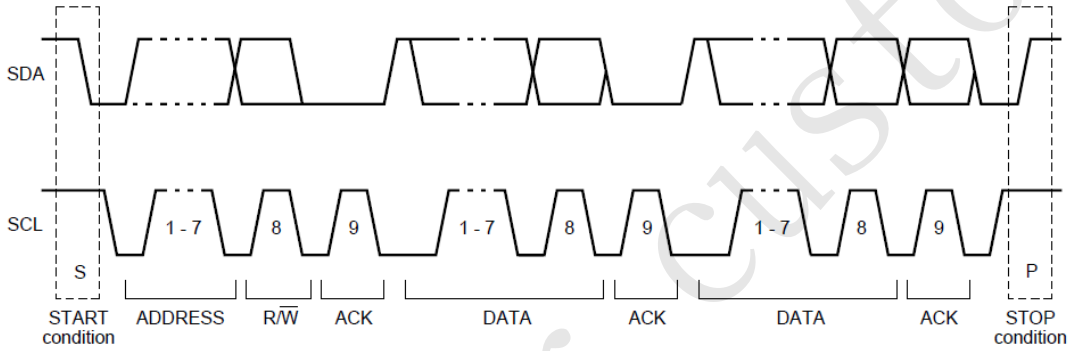


Figure 8- 2 I2C timing chart

8.2.3 Register table

Table 8- 5 Register table for I2C

Address	Name	R/W	Description	Reset Value
0x00	rsvd	RW		
0x01	I2CID	RW	I2C ID	0x5c
0x02	rsvd	RW		
0x03	I2CSCT	RW	[0]: address auto increase enable [1]: rsvd [2] enable host address	0x01
0x20	PCMD	RW	Command sent by host [6]: Host to device [7]: Device to host	
0x21	HOSTCS		W/r[0]:host_rd_clear_en: host read auto clear enable r/o[1]:host_cmd_rd:i2c host operation	0x01

Address	Name	R/W	Description	Reset Value
			have happened and is read operation r/o[2]:host_cmd_wr:i2c host operation have happened and is write operation	
0x22	irq		[0]: write 1 clear software_irq, read software irq status [1]: write 1 clear an_irq , read an_irq status [2]: write 1 clear host_pkt_irq, read host_pkt_irq status [3] rsvd [4] write 1 to trigger software irq	
0x3e	Reg_host_map_adrl	R/W	I2C mapping[7:0]: Lower byte of Mapping mode buffer address	0x80
0x3f	Reg_host_map_adrh	R/W	I2C mapping[15:8]: Higher byte of Mapping mode buffer address	0x9f

8.2.4 I2C Slave mode

The I2C of the TLSR8368/TLSR8368E02 can only be used as Slave. I2C slave address could be configured in I2CID (address 0x01) [7:1].

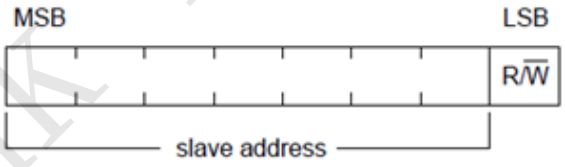


Figure 8- 3 I2C slave address

In I2C Slave mode, Master could initiate transaction anytime. I2C slave module will reply ACK automatically.

Sub modes including Direct Memory Access (DMA) mode, Mapping mode and a specific “Command Analysis” mode are supported. The latter is designed specially for the user who wants to define and use his own I2C protocol and read/write format.

8.2.4.1 DMA mode

In DMA mode, other devices (Master) could read/write Register and/or SRAM of the TLSR8368/TLSR8368E02 via I2C protocol, and initial access address is specified by I2C Master. In this mode, I2C Slave will execute the read/write command from I2C Master automatically. But user needs to notice that the lowest system clock shall be 10x faster than I2C bit rate.

The access address is offset by 0x800000. In TLSR8368/TLSR8368E02, Register address starts from 0x800000 and SRAM address starts from 0x808000. For example, if Addr(High) is 0xaa and Addr(Low) is 0xcc, the real address of accessed data is 0x80aacc.

Master could access data of the TLSR8368/TLSR8368E02 via I2C byte by byte, and access address supports automatical increment by setting address 0x03[0] to 1.

Read Format in DMA mode

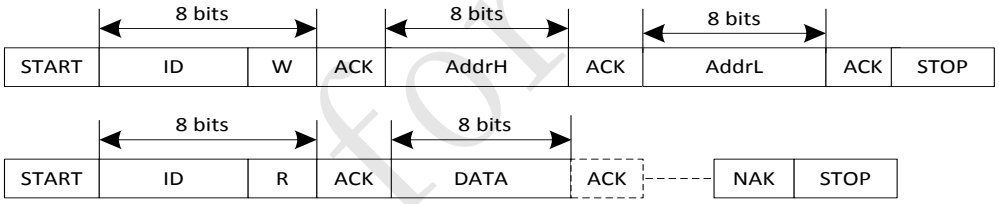


Figure 8- 4 Read format in I2C DMA mode

Write Format in DMA mode

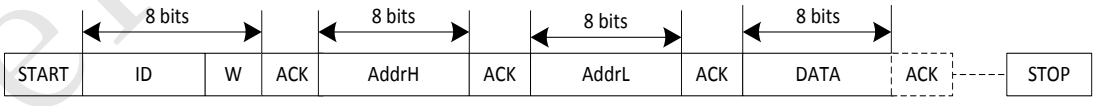


Figure 8- 5 Write format in I2C DMA mode

8.2.4.2 Mapping mode

Address 0x03[2] should be set to 1b'1 to enable Mapping mode.

In Mapping mode, data written and read by I2C master will be redirected to specified 128-byte buffer in SRAM. The initial address of the 128-byte buffer is configurable via addresses 0x3e~0x3f. Address 0x3e is lower byte and address 0x3f is higher byte. The first 64-byte buffer is for written data and following 64-byte buffer is for read data. Every time the data access will start from the beginning of the Write-buffer/Read buffer after I2C stop condition occurs.

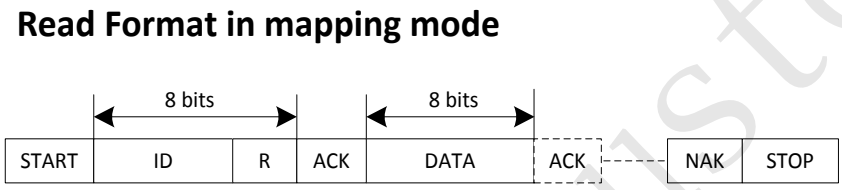


Figure 8- 6 Read format in I2C Mapping mode

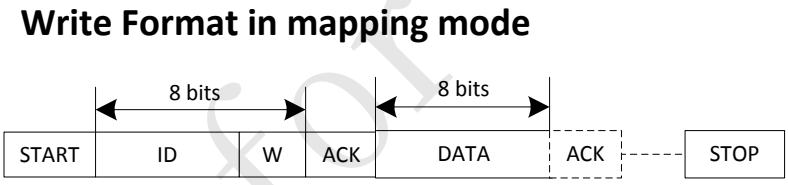


Figure 8- 7 Write format in I2C Mapping mode

8.2.4.3 Command analysis mode

For I2C Master that uses self-defined I2C protocol and read/write format, a specific "Command Analysis" mode is supported by the I2C of the TLSR8368/TLSR8368E02 (Slave).

I2C Master should specify initial access address as 0x20 (offset by 0x800000) in DMA mode, or configure mapping mode buffer address registers (addresses 0x3e~0x3f) as 0x800020 in mapping mode, by sending command to I2C Slave. I2C Slave supports command analysis function. By reading address 0x21[2:1], user can know whether the I2C Master operation that just happened is read or write operation.

8.3 SWS

The TLSR8368/TLSR8368E02 supports SWS (Single Wire Slave) interface which represents the slave device of the single wire communication system developed by Telink. The maximum data rate can be up to 2Mbps.

8.4 Pull-up/Pull-down resistor

For the TLSR8368/TLSR8368E02, the GPIOs including GP17~GP24, GP26~GP27 and GP31 support configurable 1MΩ/10KΩ pull-up resistor or 100KΩ pull-down resistor; the GPIOs including GP0~GP16 and GP32 support 100KΩ pull-down resistor. Related register configuration can be found in Table 8-6. By default the pull-up and pull-down resistors are disabled.

Take the GP17 as an example: Setting analog register afe3V_reg08<1:0> to 2b'01/2b'10/2b'11 is to enable 1MΩ pull-up resistor/10KΩ pull-up resistor/100KΩ pull-down resistor respectively for GP17; Clearing the two bits disables pull-up and pull-down resistors for GP17.

Table 8- 6 3.3V analog registers related to Pull-up/Pull-down resistor

Address(bit)	Mnemonic	Reset value	Description
afe3V_reg08<1:0>	pullupdown_ctrl <1:0>	00	Wake up mux input GP17 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<3:2>	pullupdown_ctrl <1:0>	00	Wake up mux input GP18 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(bit)	Mnemonic	Reset value	Description
afe3V_reg08<5:4>	pullupdown_ctrl <1:0>	00	Wake up mux input GP19 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 input GP20 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_reg09<1:0>	pullupdown_ctrl <1:0>	00	Wake up mux input GP21 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_reg09<3:2>	pullupdown_ctrl <1:0>	00	Wake up mux input GP22 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_reg09<5:4>	pullupdown_ctrl <1:0>	00	Wake up mux input GP23 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_reg09<7:6>	pullupdown_ctrl <1:0>	00	Wake up mux input GP24 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_reg40<7:0>	pulldown_ctrl <7:0>	00000000	GP6 ~GP0, GP32 pull down enable 0--No pull down resistor 1--enable 100kOhm pull down resistor

Address(bit)	Mnemonic	Reset value	Description
afe3V_reg41<7:0>	pulldown_ctrl <15:8>	00000000	GP14 ~GP7 pull down enable 0--No pull down resistor 1--enable 100kOhm pull down resistor
afe3V_reg42<1:0>	pulldown_ctrl <17:16>	00	GP16~GP15 pull down enable 0--No pull down resistor 1--enable 100kOhm pull down resistor
afe3V_reg42<3:2>	pullupdown_ctrl <1:0>	00	Wake up mux input GP26 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_reg42<5:4>	pullupdown_ctrl <1:0>	00	Wake up mux input GP27 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_reg42<7:6>	pullupdown_ctrl <1:0>	00	Wake up mux input GP31 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

9 Quadrature Decoder

The TLSR8368/TLSR8368E02 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 16 dedicated GPIOs including GP16~GP31 via setting address 0xd2[4:0] (for channel a)/0xd3[4:0] (for channel b).

Table 9- 1 Input pin selection

Address 0xd2[4:0]/0xd3[4:0]	Pin
0	GP16
1	GP17
2	GP18
3	GP19
4	GP20
5	GP21
6	GP22
7	GP23
8	GP24
9	GP25
10	GP26
11	GP27
12	GP28
13	GP29
14	GP30
15	GP31

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.

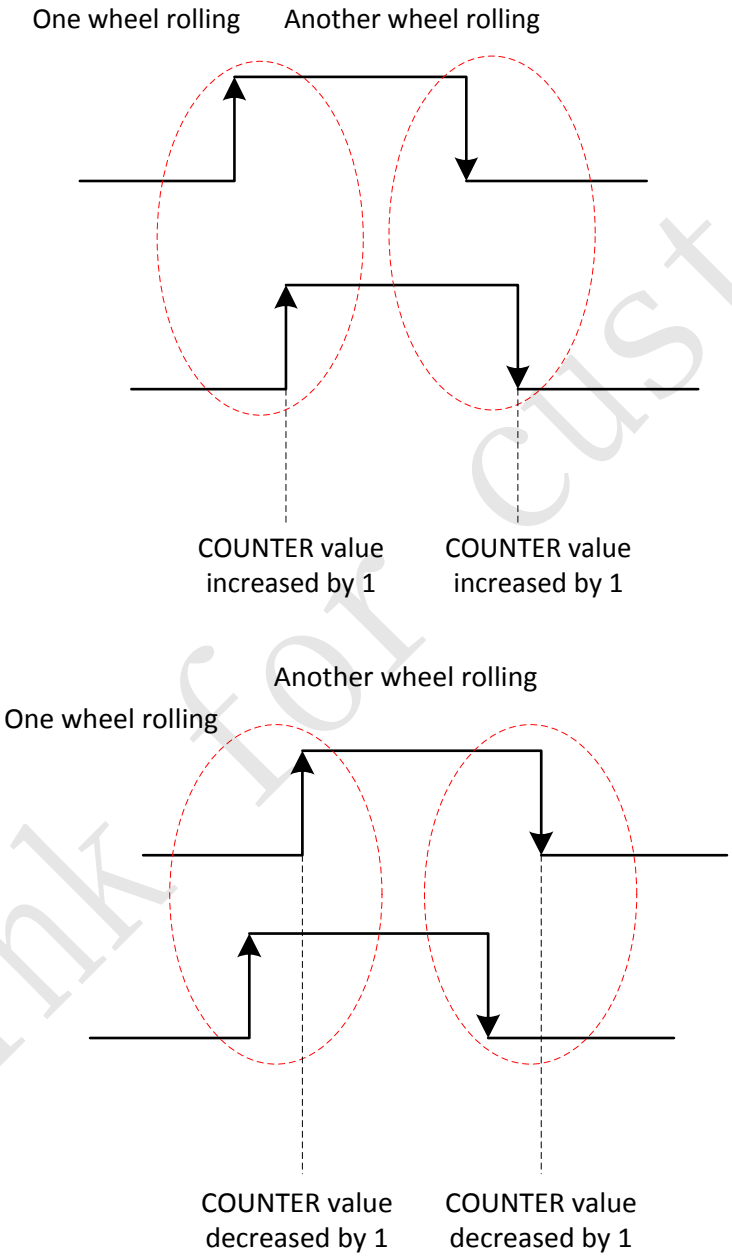


Figure 9- 1 Common mode

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 QDEC Counter value will be increased/decreased by 2 for one wheel rolling.

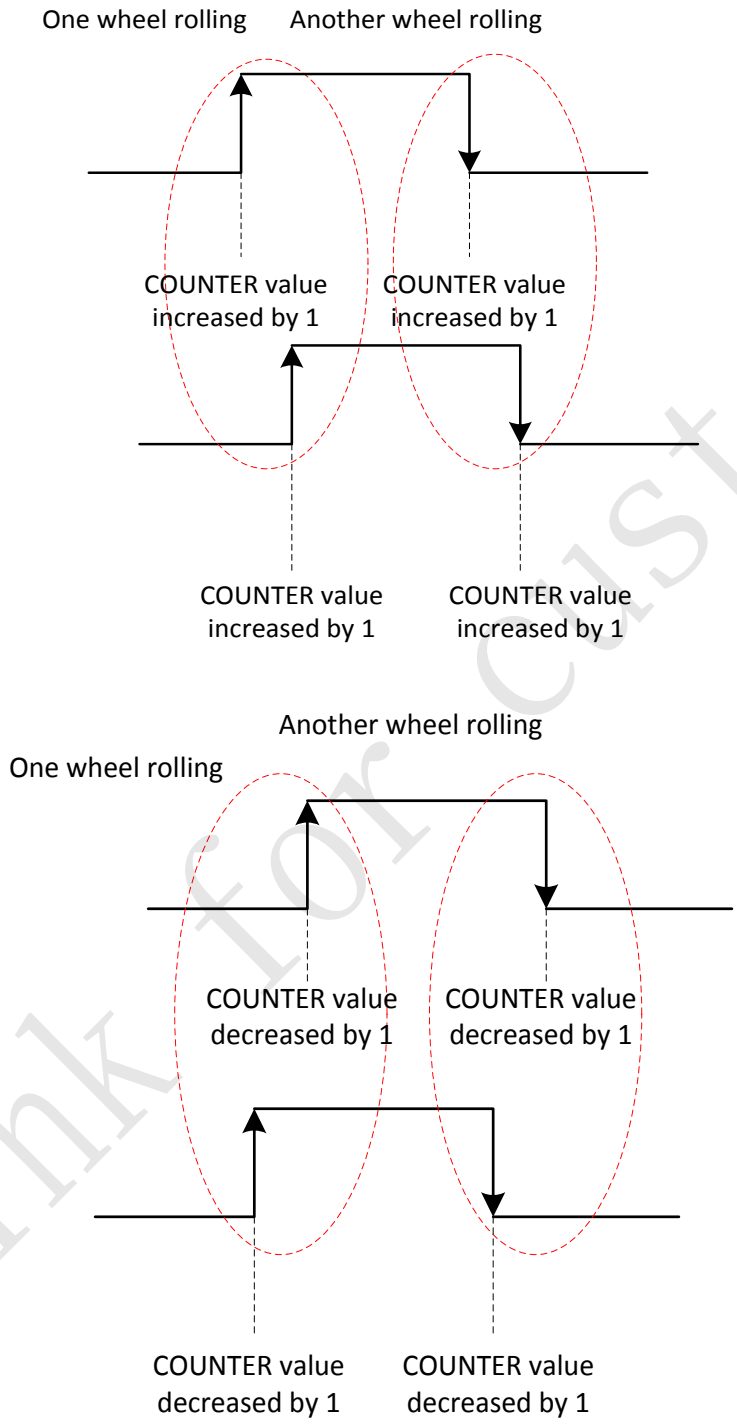


Figure 9- 2 Double accuracy mode

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.

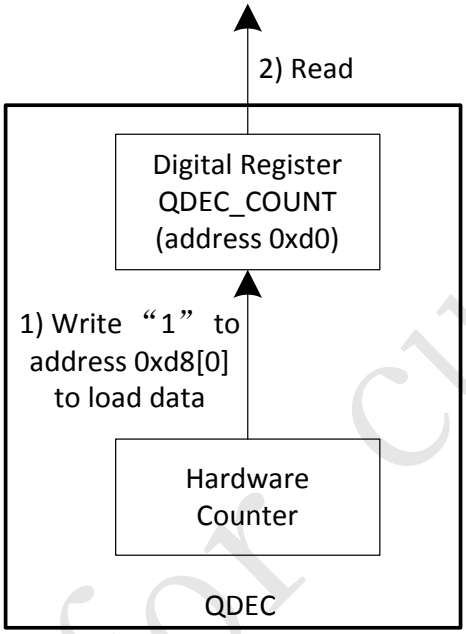


Figure 9- 3 Read real time counting value

9.4 QDEC interrupt

Address 0xd4[0] serves to enable or mask QDEC interrupt.

If address 0xd4[0] is set to 1b'1 to enable QDEC interrupt, whenever counter value changes, an QDEC IRQ is asserted and address 0xd5[0] is set to 1b'1 automatically. Writing 1b'1 to address 0xd5[0] can clear the interrupt flag bit.

9.5 QDEC reset

Address 0xd6[0] serves to reset the QDEC. The QDEC Counter value is cleared to zero.

9.6 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.

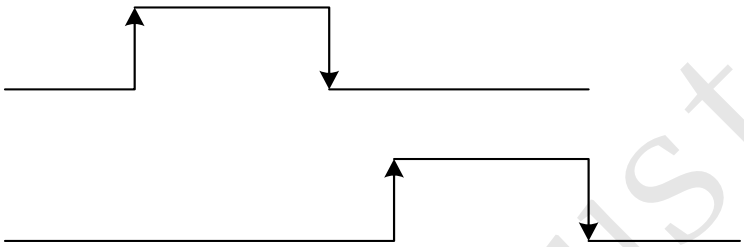


Figure 9- 4 Shuttle mode

9.7 Register table

Table 9- 2 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 de glitch) [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_CHNA0	R/W	[4:0] QDEC0 input pin select for channel a choose 1 of 16 pins for input channel a	0x00
0xd3	QDEC_CHNB0	R/W	[4:0] QDEC0 input pin select for channel b choose 1 of 16 pins for input channel b	0x01
0xd4	QDEC_MASK	R/W	[0]Interrupt mask 1: enable 0: mask	0x00

Address	Mnemonic	Type	Description	Reset value
0xd5	QDEC_INT	R	[0]Interrupt flag Write 1 to clear	
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 SAR ADC

The TLSR8368/TLSR8368E02 integrates one ADC module, which can be used to sample battery voltage and external analog input.

10.1 Register table

Table 10- 1 Register table for SAR ADC

Address	Mnemonic	Type	Description	Reset Value
Digital Registers				
0x2b	ADCREF	RW	[0]select reference 0: Vbg 1: VDDH [7:1] rsvd	0x03
0x2c	ADCMUXM	RW	Analog inputs select bit [2:0] sel ana input 000: close all 001: GP17 010: GP18 011: GP22 100: GP23 101: VDDDEC 110: V_{GP23} or 1/3*V_{GP23}. Refer to analog register afe3V_reg02<3>. 111: reserved [5:4] sel dif input choose single or diff mode and select negative input 00: single mode 01: GP18 as negative input 10: GP23 as negative input 11: VDDDEC as negative input	0x02
0x35	ADC_RUN	R/W	[7] manual mode run signal	0
0x38	ADC_DAT[7:0]	R		
0x39	ADC_DAT[9:8]	R	[1:0] ADC_DAT[9:8] [6:2] rsvd [7] adc_busy	

Address	Mnemonic	Type	Description	Reset Value
0x3c	ADC TSAPM	RW	[2:0] Select number of clock cycles for ADC sampling Setting # of clock cycles 000 3 001 6 010 9 011 12 100 18 101 24 110 48 111 144 [4:3] ADC resolution select 00:7bit 01:8bit 10:9bit 11:10bit [5] Select sign of ADC output data bit<9> 0: positive 1: negative	0x00
Analog registers				
afe3V_reg 06<0>	Power Down	RW	Power down SAR ADC 1: Power down 0: Power up	1
afe3V_reg 02<3>	power_logic_sel _atb	RW	Select V_{GP23} or 1/3*V_{GP23} as ADC input (refer to digital register 0x2c[2:0]). 0: V_{GP23} 1: 1/3*V_{GP23}	0

10.2 SAR ADC clock

ADC clock derives from FHS. Address 0x6b[7] should be set to "1" to enable ADC clock.

ADC clock must be lower than 5M when ADC reference voltage is selected as VDDH and must be lower than 4M when ADC reference voltage is selected as Vbg.

ADC clock is calculated according to the formula below:

$$F_{\text{ADC clock}} = F_{\text{FHS}} * \text{adc\_step}[10:0]/\text{adc\_mod}[11:0]$$

Refer to **Section 5.2.1 SAR ADC clock** for details.

10.3 Select ADC range, resolution and sampling time

ADC range is same as the ADC reference voltage, which is configured by register 0x2b[0]: Vbg (1.26V bandgap reference), or VDDH.

Address 0x3c[4:3] serves to set resolution: 7bit, 8bit, 9bit or 10bit. ADC data format is always 10bit no matter the conversion bit is set. Address 0x3c[5] serves to set the sign of ADC output data bit[9] as positive or negative. For example, 8 bits resolution indicates higher 8 bits are valid bits and the lower 2 bits are invalid bits.

ADC sampling time can be configured by address 0x3c[2:0], the lower sampling cycle, the shorter ADC convert time.

10.4 Select input mode and channel

The TLSR8368/TLSR8368E02 ADC has 4 input channel which can be selected by address 0x2c[2:0].

Address 0x2c[5:4] serves to select differential mode or single-end input mode.

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

For example, if address 0x2c is set to 0x06 (i.e. 8b' 00000110), and analog register afe3V_reg02<3> is set to 1b'1, $1/3*V_{GP23}$ is selected as ADC input of single-end mode.

When address 0x2c[5:4] is set to 2b'01/2b'10/2b'11, differential mode is selected, the corresponding channel identified by address 0x2c[5:4] is selected as negative input, and the positive input is selectable via address 0x2c[2:0].

For example, if address 0x2c is set to 0x11 (i.e. 8b'00010001), GP17 and GP18 are selected as positive-end and negative-end input of differential mode; actual input signal for ADC is the difference of V_{GP17} and V_{GP18} (i.e. V_{GP17} minus V_{GP18}).

10.5 ADC start

Address 0x35[7] set to "1" starts ADC sampling and conversion process.

10.6 ADC status

ADC busy flag bit, i.e. address 0x39[7], indicates whether ADC is busy.

10.7 ADC data

The real time output data ADC_DAT[9:0] can be read from addresses 0x39~0x38.

11 PWM

The TLSR8368/TLSR8368E02 supports 4-channel PWM (Pulse-Width-Modulation) output. Each PWM#n has its corresponding inverted output at PWM#n_INV pin (n=0~3).

11.1 Register table

Table 11- 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	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	0x00
0x783	PWM_CC0	R/W	[3:0]:1‘b1 invert PWM output	0x00
0x784	PWM_CC1	R/W	[3:0]:1‘b1 invert PWM_INV output	0x00
0x785	PWM_CC2	R/W	[3:0]: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
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	0x00

Address	Mnemonic	Type	Description	Reset Value
			time(if pola[1]=1)	
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
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
0x7b0	PWM_MASK	R/W	INT mask [0] PWM0 Pnum int 0: disable 1: Enable [1] rsvd [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 [7:6] rsvd	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]:rsvd [2]:PWM0 cycle done int(PWM_CNT==PWM_TMAX)	0x00

Address	Mnemonic	Type	Description	Reset Value
			[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) [7:6]: rsvd	
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	
0x7c0	PWM_NCNT0	R	[7:0]PWM0 pluse_cnt value	
0x7c1	PWM_NCNT0		[15:8]PWM0 pluse_cnt value	

11.2 Enable PWM

Register PWM_EN (address 0x780)[3:0] serves to enable PWM3~PWM0 respectively via writing “1” for the corresponding bits.

11.3 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:

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

11.4 PWM waveform, polarity and output inversion

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

11.4.1 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 0x788~0x78f). 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), 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) expires.

An interruption will be generated at the end of each signal frame if enabled via register PWM_MASK (address 0x7b0[2:5]).

11.4.2 Invert PWM output

PWM#n and PWM#n_INV output could be inverted independently via register PWM_CC0 (address 0x783[3:0]) and PWM_CC1 (address 0x784[3:0]). When the inversion bit is enabled, the corresponding PWM channel waveform will be inverted completely.

11.4.3 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[3:0]), PWM#n will output Low level at Count status and High level at Remaining status.

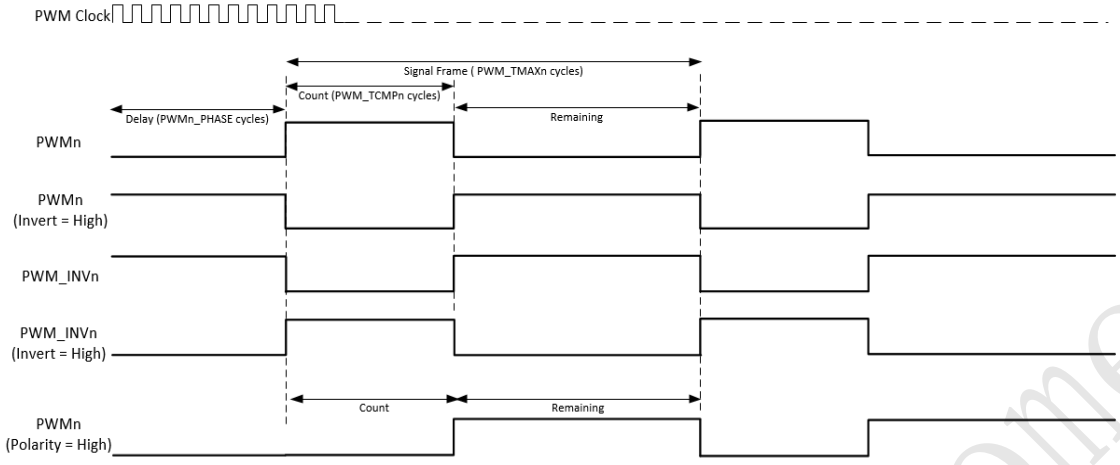


Figure 11- 1 PWM output waveform chart

11.5 PWM mode

11.5.1 Select PWM mode

PWM0 supports 3 modes, including Continuous (normal) mode, Counting mode, and IR mode. PWM1~PWM3 only support Continuous mode.

Register PWM_MODE (address 0x782[1:0]) serves to select PWM0 mode.

11.5.2 Continuous mode

PWM0~PWM3 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.

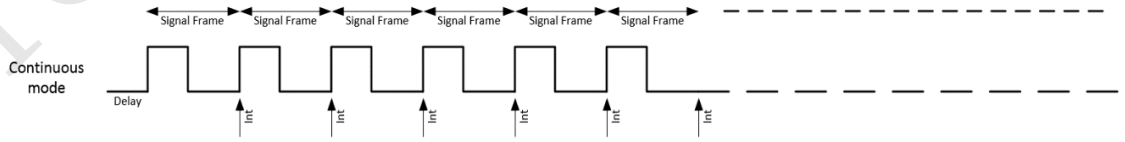


Figure 11- 2 Continuous mode

11.5.3 Counting mode

Only PWM0 supports Counting mode. In this mode, PWM0 sends out specified number of signal frames which is defined as a pulse group. The number is configured via register PWM_PNUM0 (address 0x7ac~0x7ad). After a pulse group is finished, PWM0 will be disabled automatically, and a Pnum interruption will be generated if enabled via register PWM_MASK (address 0x7b0[0]).

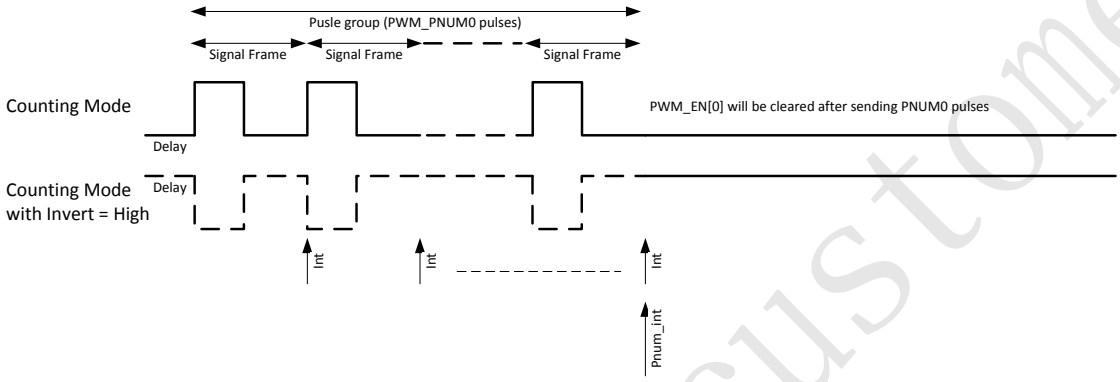


Figure 11- 3 Counting mode

Counting mode also serves to stop IR mode gracefully. Refer to **section 11.5.4** for details.

11.5.4 IR mode

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

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

To stop IR mode and complete current pulse group, user can switch PWM0 from IR mode to Counting mode so that PWM0 will stop after current pulse group is finished. If PWM0 is disabled directly via PWM_EN (0x780[0]), PWM0 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.

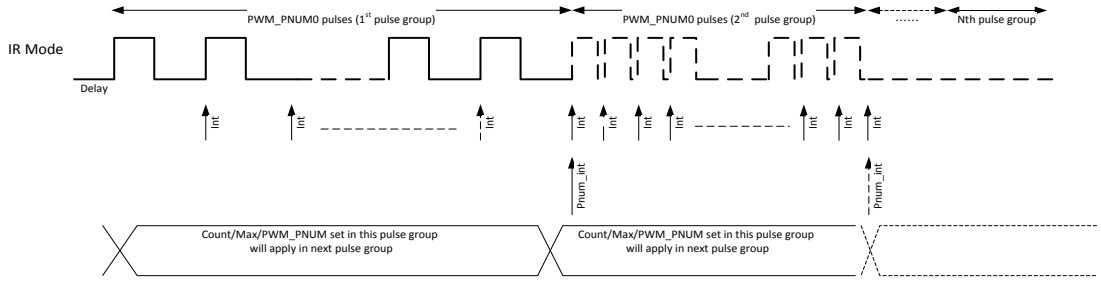


Figure 11- 4 IR mode

11.6 PWM interrupt

There are 5 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 will generate a Pnum IRQ signal after completing a pulse group. Interrupt status can be cleared via register PWM_INT (address 0x7b1).

12 EEPROM

The TLSR8368E02 also embeds 2Kbit EEPROM. To control the EEPROM, the GP12/GP13 inside the TLSR8368E02 are connected to the EEPROM and serve as I2C SCL/SDA respectively.

12.1 Communication protocol

Both SDA and SCL are bidirectional lines connected to a positive supply voltage via a pull-up resister. When the bus is free, both lines are HIGH. Data in SDA line must keep stable when SCL line is at high level, and it's only allowed to change when SCL line is at low level.

A negative/positive edge of SDA when SCL is high indicate a start/stop condition, respectively.

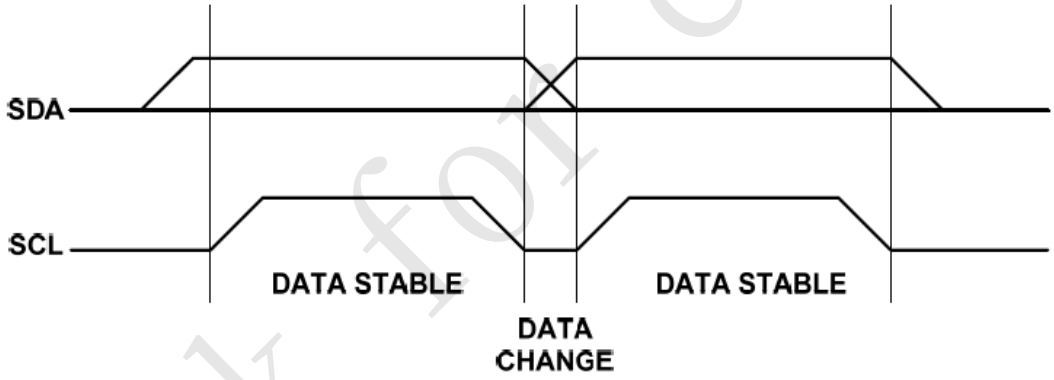


Figure 12- 1 Data validity

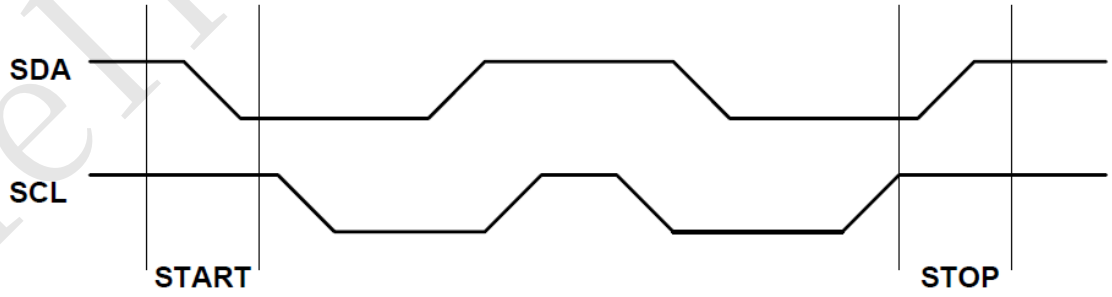


Figure 12- 2 Start and stop condition

All addresses and data words are serially transferred between the EEPROM and the MCU in 8-bit words. The EEPROM will respond with an ack “0” after it receives each word. Upon receipt of each word from the EEPROM, the MCU should also send a “0” to the EEPROM, and continue to output the next data word or send a stop condition.

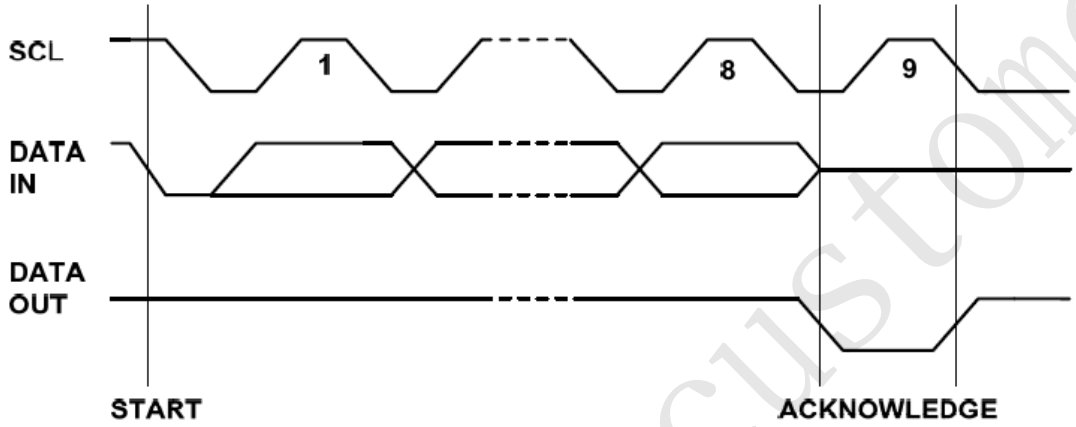


Figure 12- 3 Send Ack

Upon power-up or receipt of the stop bit and completion of any internal operations, the EEPROM will enter low-power standby mode.

The EEPROM requires an 8-bit device address word following a start condition to enable the read/write access operation. As shown in Figure 12- 4, the device address word consists of a mandatory 1, 0 sequence for the first four MSBs, device address bits A2, A1 and A0, as well as R(1)/W(0) select bit to indicate read/write operation.

1	0	1	0	A_2	A_1	A_0	R/W
MSB							LSB

Figure 12- 4 Device address

Since the device address bits “A2A1A0” are “000”, EEPROM I2C address should be 0x50.

12.2 EEPROM operation

For EEPROM read and write operations, user needs to simulate corresponding I2C read/write timing sequence via software.

12.2.1 Write operations

The EEPROM supports byte write and 8-byte page write.

For byte write, a write operation requires an 8-bit data word address following the device address word and ack. Upon receipt of this address, the EEPROM will also respond with an ack “0”, and then the first 8-bit data word is clocked in. After the 8-bit data word is received, the EEPROM will send an ack “0”. The addressing device (MCU) must terminate the write sequence with a stop condition. At this time, the EEPROM enters an internally-timed write cycle to the nonvolatile memory. All inputs are disabled during this write cycle, and the EEPROM won’t respond until the write is completed.

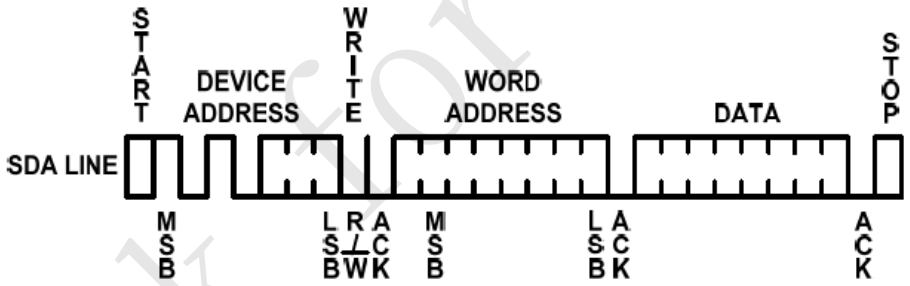


Figure 12- 5 Byte write

A page write initiation is the same as a byte write. The MCU does not send a stop condition after the first data word is clocked in and acknowledged by the EEPROM, and up to seven more data words can be transmitted. The EEPROM will respond with an ack “0” after each data word is received. The MCU must terminate the page write with a stop condition.

For the data word address, the lower three bits are internally incremented following the receipt of each data word, while the higher bits retain the memory page row location. When the word address internally generated reaches the page boundary,

the following byte is placed at the beginning of the same page. If more than eight data words are transmitted to the EEPROM, the data word address will “roll over” and previous data will be overwritten.

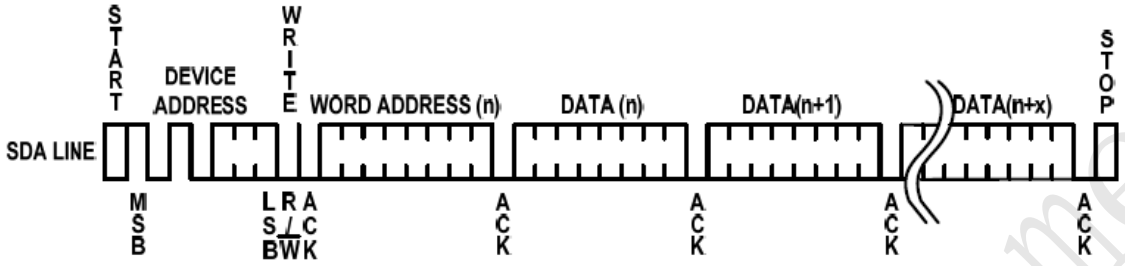


Figure 12- 6 Page write

Once the internally-timed write cycle is started and the EEPROM inputs are disabled, acknowledge polling can be initiated. This involves sending a start condition followed by the device address word. The R/W bit indicates read/write operation. Only if the internal write cycle is completed will the EEPROM respond with an ack “0” allowing the read/write sequence to continue.

12.2.2 Read operations

Read operation initiation is similar to write operation except that the R/W bit in the device address word should be set as “1”. Three read operations are supported, including current address read, random address read and sequential read.

For current address read, the internal data word address counter maintains the last address accessed during the last read/write operation, incremented by 1. This address stays valid between operations as long as the chip power is maintained. The address “roll over” during read is from the last byte of the last memory page to the first byte of the first page. The address “roll over” during write is from the last byte of the current page to the first byte of the same page. Once the device address with the R/W bit set as “1” is clocked in and acknowledged by the EEPROM, the data word of current address is serially clocked out. The MCU will generate a stop condition following NO ack.

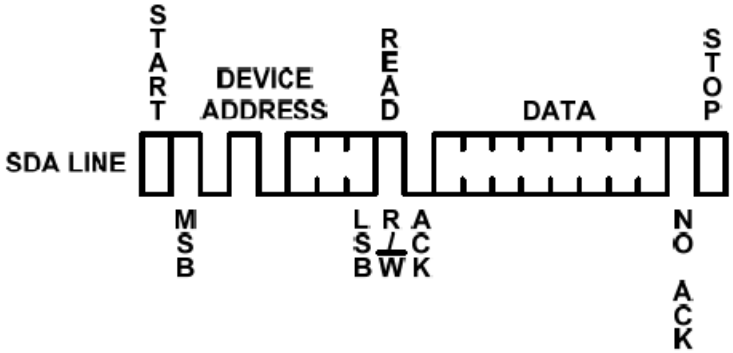


Figure 12- 7 Current address read

A random read requires a “dummy” byte write sequence to load in the data word address. Once the device address word and data word address are clocked in and acknowledged by the EEPROM, the MCU must generate another start condition. The MCU now initiates a current address read by sending a device address with the R/W bit set as “1”. The EEPROM acknowledges the device address and serially clocks out the data word. The MCU will generate a stop condition following NO ack.

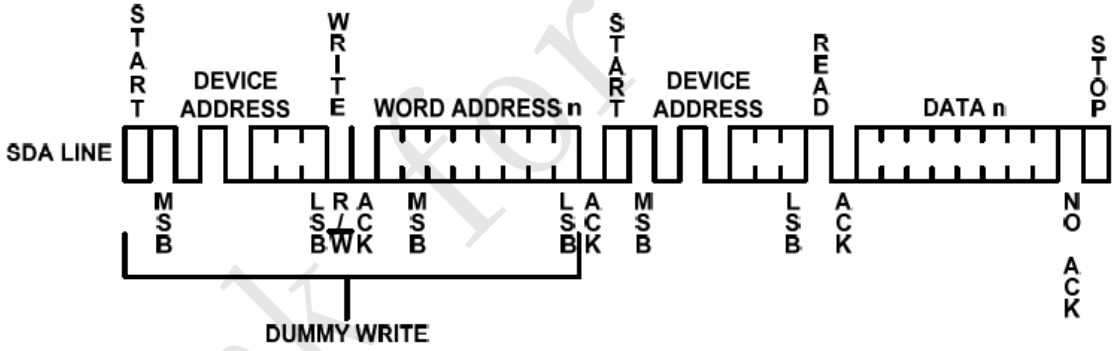


Figure 12- 8 Random read

Sequential read is initiated by either a current address read or a random address read. After the MCU receives a data word, it responds with an ack. As long as the EEPROM receives an ack, it will continue to increment the data word address and serially clock out sequential data words. When the memory address limit (2K) is reached, the data word address will “roll over” and the sequential read will continue. The sequential read operation is terminated when the MCU generates a stop condition following NO ack.

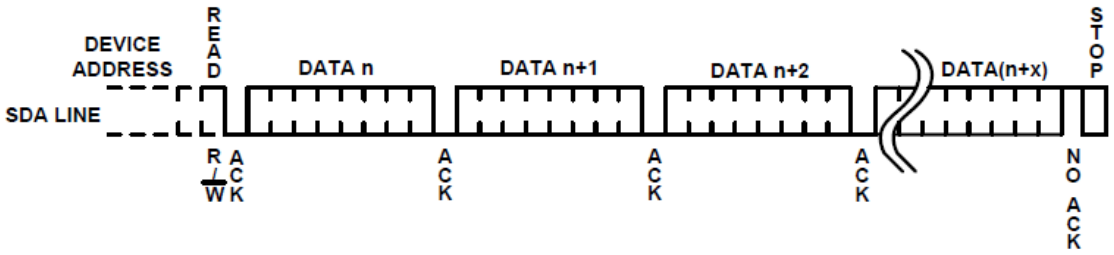


Figure 12- 9 Sequential read

13 Key Electrical Specifications

13.1 Absolute maximum ratings

Table 13- 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.

13.2 Recommended operating condition

Table 13- 2 Recommended operation condition

Item	Sym.	Min	Typ.	Max	Unit	Condition
Power-supply voltage	VDD	1.9	3.3	3.6	V	
Operating Temperature Range	T_{Opr}	-40	27	85	°C	

13.3 DC characteristics

Table 13- 3 DC characteristics

Item	Sym.	Min	Typ.	Max	Unit	Condition
Tx current	I_{Tx}	-	15	-	mA	Continuous Tx transmission @0dBm output power
		-	22	-	mA	Continuous Tx transmission @6dBm output power
Rx current	I_{Rx}	-	12	-	mA	Continuous Rx reception
Suspend current	I_{Susp}	-	10	-	uA	
Deep sleep current	I_{Deep}	-	0.7	-	uA	

*Note: All tests above are done at room temperature (T=25℃).

13.4 AC characteristics

Table 13- 4 AC Characteristics

Item	Sym.	Min	Typ.	Max	Unit	Condition
Digital inputs/outputs						
Input high voltage	VIH	0.7VDD		VDD	V	
Input low voltage	VIL	VSS		0.3VDD	V	
Output high voltage	VOH	VDD-0.3		VDD	V	
Output low voltage	VOL	VSS		0.3	V	
RF performance						

Item	Sym.	Min	Typ.	Max	Unit	Condition
Item		Min	Typ	Max	Unit	
RF_Rx performance						
Sensitivity	2Mbps		-88		dBm	
	250Kbps		-97		dBm	
Frequency Offset Tolerance		-300		+300	KHz	
Co-channel rejection			-5		dB	
In-band blocking rejection (Single Tone Interference)	±1 MHz offset		-9		dB	
	-2 MHz offset		20		dB	
	+2 MHz offset		5		dB	
	-3 MHz offset		20		dB	
	+3 MHz offset		18		dB	
	>4MHz offset		28		dB	
In-band blocking rejection (Equal Modulation Interference)	±1MHz offset		-9		dB	
	-2 MHz offset		-2		dB	
	+2 MHz offset		-3		dB	

Item	Sym.	Min	Typ.	Max	Unit	Condition
	-3 MHz offset		12		dB	
	+3 MHz offset		9		dB	
	>4MHz offset		18		dB	
Image rejection			44		dB	
RF_Tx performance						
Output power			6		dBm	
Modulation 20dB bandwidth			2.8		MHz	
16MHz crystal						
Nominal frequency (parallel resonant)	f_{NOM}		16		MHz	
Frequency tolerance	f_{TOL}	-60		+60	Ppm	
Load capacitance	C_L	5	12	18	pF	Programmable on chip load cap
Equivalent series resistance	ESR		50	100	ohm	
32MHz RC oscillator						
Nominal frequency	f_{NOM}		32		MHz	
Frequency tolerance	f_{TOL}		1		%	On chip calibration

Item	Sym.	Min	Typ.	Max	Unit	Condition
32kHz RC oscillator						
Nominal frequency	f_{NOM}		32		kHz	
Frequency tolerance	f_{TOL}		0.03		%	On chip calibration
Calibration time			3		ms	
ADC						
Differential nonlinearity	DNL		0.8		LSB	
Integral nonlinearity	INL		0.7		LSB	
Signal-to-noise and distortion ratio (fin=1kHz, fS=16kHz)	SINAD		57.8		dB	
Spurious free dynamic range (fin=1kHz, fS=16kHz)	SFDR		64.5		dB	
Effective Number of Bits	ENOB		9.2		bits	
Sampling frequency	Fs			250	KHz	VDDH reference
				200	KHz	Vbg reference

14 Application

14.1 Application example for the TLSR8368ET48

14.1.1 Schematic

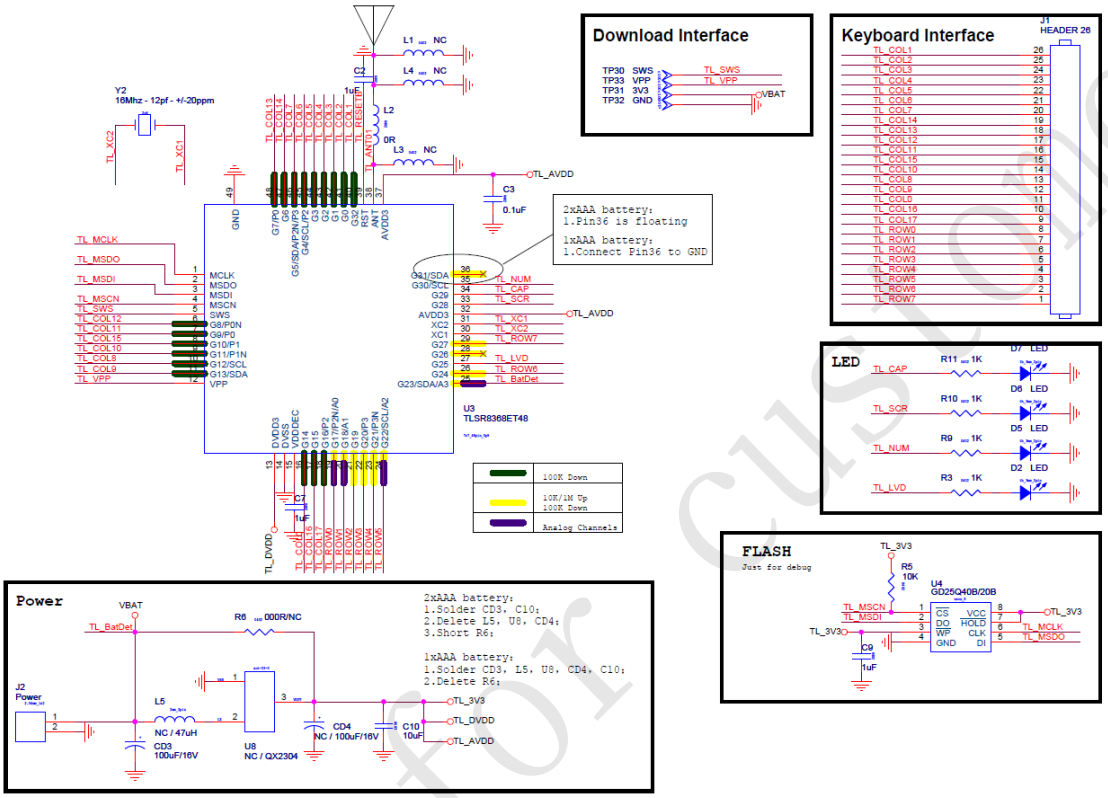


Figure 14- 1Schematic for the TLSR8368ET48

14.1.2 BOM (Bill of Material)

Table 14- 1 BOM table for the TLSR8368ET48

Quantity	Reference	Value	Footprint	Description
1	U3	TLSR8368ET48	QFN-48	
1	Y2	16Mhz - 12pf - +/-20ppm	3225	
2	C2	1uF	0402	
	C7	1uF	0402	
1	C3	0.1uF	C1005X5R1C104K	
4	D2	LED	Led_th_3mm_2pin	
	D5	LED	Led_th_3mm_2pin	
	D6	LED	Led_th_3mm_2pin	
	D7	LED	Led_th_3mm_2pin	
1	L2	0R	0402	
4	R3	1K	0402	
	R9	1K	0402	
	R10	1K	0402	
	R11	1K	0402	
Flash				
1	U4	GD25Q40B/20B	SOP-8	Just For Debug
1	R5	10K	0402	
1	C9	1uF	0402	
Power				
1	CD3	100uF/16V		2xAAA
1	C10	10uF	0603C	
1	R6	000R	0402	