A31L214, A31L213 User's Manual

Ultra-Low Power Cortex-M0+ Microcontroller
Flash 128/64KB, SRAM 20KB, Data Flash 4KB, ADC, AES

UM Rev. 1.00

Introduction

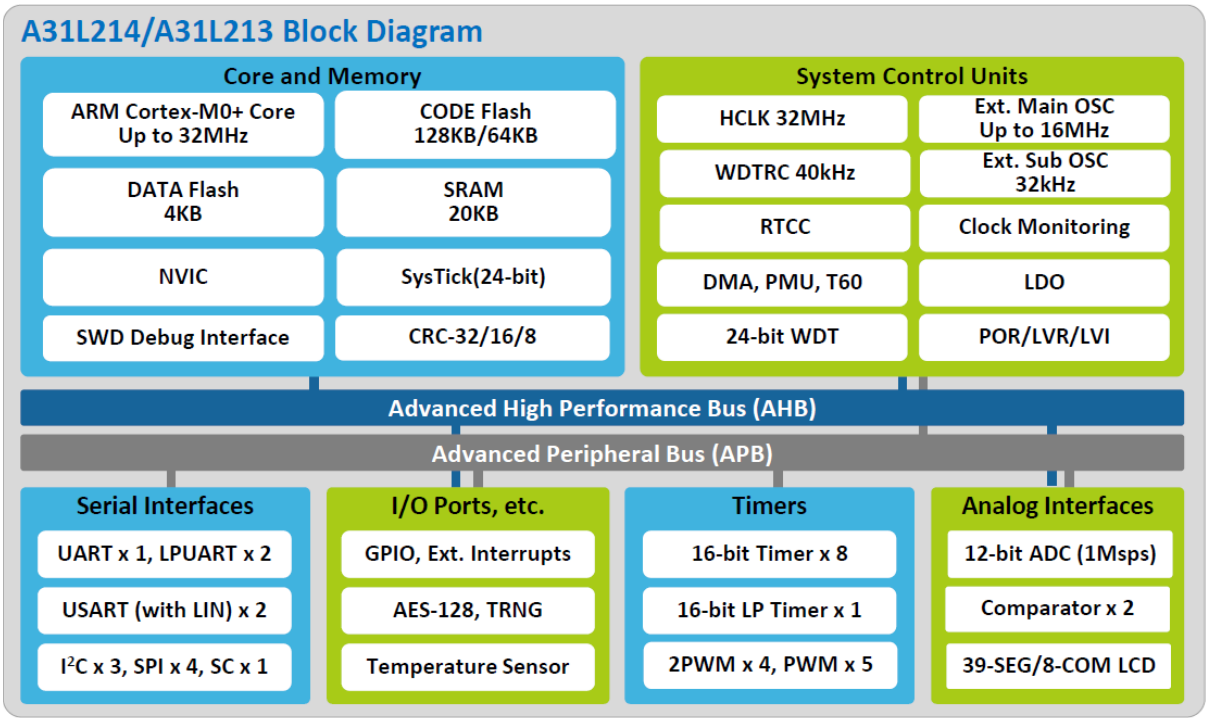
A31L21x User's manual contains complete information of the A31L21x MCU for application developers who use A31L214 or A31L213 for their specific needs.

The ultra-low power A31L21x series is a 32-bit general purpose microcontroller for various appliances. To meet the requirements for the complexity and high performance in consumer electronics, the ultra-low power A31L21x series incorporates ARM's high-speed 32-bit Cortex-M0+ Core, and has up to 128KB of Flash memory, 4KB of Data Flash, and 20KB of SRAM.

As shown in Figure 1, the ultra-low power A31L21x series has various peripherals such as 16-bit timers, Real timer and calendar, 12-bit ADC, Comparator, CRC generator, UART, USART, LPUART, I2C, SPI, Smart card interface, LCD driver/controller, AES 128, True random number generator, DMA, etc. It also has a POR, LVR, LVI, and an internal RC oscillator.

The A31L21x series supports SLEEP mode and DEEP SLEEP mode to reduce power consumption.

Figure 1. Conceptual Block Diagram of A31L21x Series



Contents

List of Figures

List of Tables

1 Description

The ultra-low power A31L21x series is a microcontroller based on ARM Cortex-M0+ core with a Flash memory of up to 128KB, a Data Flash of 4KB, and an SRAM of 20KB.

Operation voltage of the device ranges from 1.71V to 3.6V. It provides a highly flexible and cost-effective solution for many embedded control applications.

This device offers 16-bit timers, Real timer and calendar, 12-bit ADC, Comparator, CRC generator, UART, USART, LPUART, I2C, SPI, Smart card interface, LCD driver/controller, AES 128, True random number generator, DMA, etc. The A31L21x series also has a POR, LVR, LVI, and an internal RC oscillator.

The ultra-low power A31L21x series supports SLEEP mode and DEEP SLEEP mode to reduce power consumption. The A31L21x series is suitable for ultra-low power applications.

1.1 Device overview

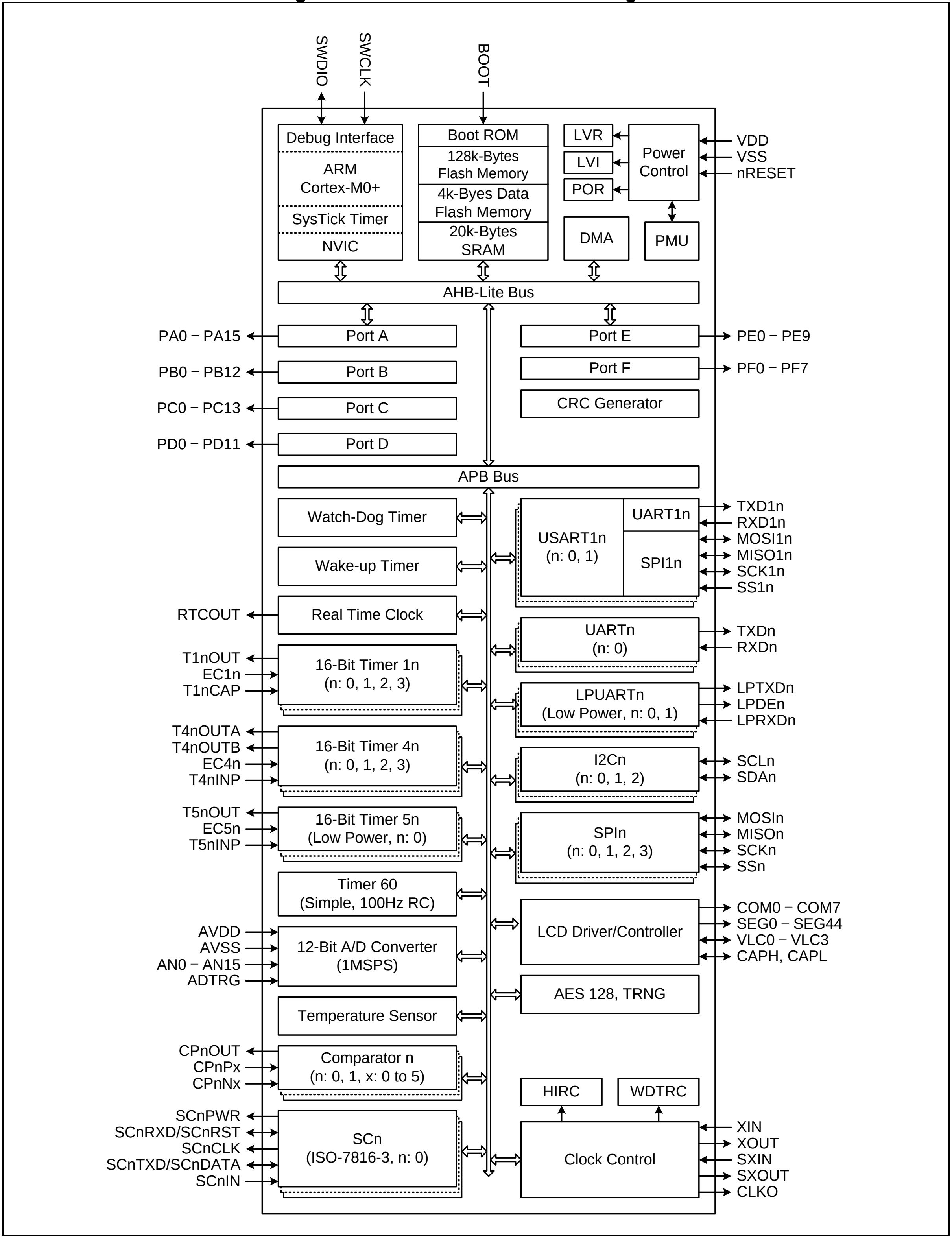
Table 1. A31L21x Series Features and Peripheral Counts

Peripheral	Description
CPU	Cortex-M0+
Memory	• Flash memory: 128/64 Kbytes • Data Flash: 4 Kbytes • SRAM: 20 Kbytes, 32-byte backup register
I/O	73 programmable
Timers	• Watchdog Timer, Real Time Clock and Calendar • Eight general purpose timers and one low power timer — Periodic, one-shot, PWM, capture mode
LCD driver	• 39 segments and 8 commons • Resistor/cap bias, and 16-step contrast control
DMA	Seven DMA channels, ADC/USART/UART/I2C/SPI/SC/AES
ADC	16-channel input, 12-bit ADC with 1Msps, down to 1.71V
Comparator	Two comparators, down to 1.71V
Temperature sensor	Frequency variation: 3.2 kHz/℃
CRC generator	8/16/32-bit CRC generator, CRC-8/16/32, CRC-CCITT
Security	AES 128-bit encryption/decryption, True random number generator
External communication ports	• 2 USARTs (UART + SPI), 1 UART, 1 SC • 2 LPUARTs, up to 9600bps with 32.768kHz • 3 I^{2}Cs up to 1Mbps, 4 SPIs up to 16Mbps
128-bit Unique ID	Supported
System fail-safe function	Clock monitoring
Debug interface	SWD debug interface
Ultra-low power tech	• 1.71V to 3.6V supply voltage • 90uA/MHz in RUN mode, 13uA in RUN mode (32.768kHz, 40kHz) • 0.99uA DEEP SLEEP + RTCC + SRAM retention • 0.35uA DEEP SLEEP with power control • 45nA shutdown (DEEP SLEEP mode 3) • 5us wakeup time from all power modes
Packages	• LQFP 80-1212 (0.5mm pitch) • LQFP 64-1010 (0.5mm pitch) • LQFP 48-0707 (0.5mm pitch) • LQFP 32-0707 (0.8mm pitch) • QFN 32-0505 (0.5mm pitch)
Operating temperature	-40℃ to +85℃ (commercial grade)
	-40℃ to +105℃ (industrial grade)

1.2 Block diagram

Figure 2 shows a block diagram of the A31L21x series.

Figure 2. A31L21x Series Block Diagram



1.3 Functional description

The following sections provide a brief description of the features of the A31L21x series microcontroller.

1.3.1 ARM Cortex-M0+

The Cortex-M0+ processor has a very low gate count. It is a highly energy efficient processor for microcontrollers and deeply embedded applications that require an area-optimized, low-power processor.

In the core, the system timer (SYSTICK) provides a simple 24-bit timer that can be used as a real time operating system (RTOS) or as a simple counter.

The processor implements the ARMv6-M Thumb instruction set including a number of 32-bit instructions, which are introduced with Thumb-2 technology. Hardware single-cycle multiplication is available.

Integrated Nested Vectored Interrupt Controller (NVIC) provides deterministic interrupt handling.

It also supports SWD debugging features.

1.3.2 Nested Vector Interrupt Controller (NVIC)

External interrupt signals connect to the NVIC, and the NVIC prioritizes the interrupts. Software can set the priority of each interrupt.

The NVIC embedded in the Cortex-M0+ processor core achieves low latency interrupts processing and efficient processing of late arriving interrupts.

All NVIC registers can only be accessed through word transfers.

1.3.3 128KB internal Code Flash memory

The A31L21x series has built-in 128KB Flash memory.

It supports self-programming feature. In addition, ISP and JTAG programming in boot or debug mode are supported.

1.3.4 20KB internal SRAM

On-chip 20KB SRAM is used as a working memory space and as a program code area temporarily.

1.3.5 4KB Data Flash memory

The A31L21x series has 4KB Data Flash memory. It is used for temporary information storage purpose.

1.3.6 Boot logic

A boot logic supports Flash programming. The boot logic is activated when the external boot pin is set to boot mode.

1.3.7 System Control Unit (SCU)

An SCU block manages internal power, clock, reset and operation mode. It also controls the analog blocks (Oscillator Block, VDC and LVR).

1.3.8 Power Management Unit (PMU)

A PMU block manages power of internal core, Code Flash, Data Flash, SRAM, logic, and peripheral blocks in RUN, SLEEP, and DEEP SLEEP modes.

It also controls the wake-up time from SLEEP and DEEP SLEEP modes.

1.3.9 24-bit Watchdog Timer (WDT)

A Watchdog Timer monitors the system. It generates internal resets or interrupts to detect abnormal status of the system.

1.3.10 Multi-purpose 16-bit timer

Eight-channel 16-bit timers and one-channel low power general-purposed 16-bit timer support the functions introduced below:

- Periodic timer mode
- Counter mode
- PWM mode
- Capture mode

1.3.11 Real Time Clock and Calendar (RTCC)

A real time clock and a calendar can run in SLEEP and DEEP SLEEP modes. The RTCC is not reset by a system reset except in the event of a power-on reset.

1.3.12 USART (UART and SPI)

USART supports UART and SPI modes. The A31L21x series has 2 channel USART module.

Boot mode uses this USART10 block to download Flash program.

1.3.13 Inter-integrated Circuit (I2C) interface

The A31L21x series has three channels of I2C block and supports up to 1MHz I2C communication.

Master and slave modes are available.

1.3.14 Serial Peripheral Interface (SPI)

The A31L21x series has four channels of SPI block and supports up to 16MHz communication.

Master and slave modes are available.

1.3.15 Universal Asynchronous Receiver/Transmitter (UART)

The A31L21x series has one channel of UART block.

For accurate baud rate control, a fractional baud-rate generation feature is supported.

1.3.16 Low Power Universal Asynchronous Receiver/Transmitter (LPUART)

The A31L21x series has two channels of Low Power UART block. This LPUART is available at 32.768kHz sub oscillator with up to 9600bps.

1.3.17 Smartcard interface (SC)

The A31L21x series has one channel of SC block. This block supports UART and smartcard modes.

The SC block has also baud-rate compensation, receive time out data, and extra guard time registers.

1.3.18 General PORT I/Os (GPIO)

16-bit PA port, 13-bit PB port, 14-bit PC port, 12-bit PD port, 10-bit PE port, and 8-bit PF port are available and provide multiple functions.

- General I/O port
- External interrupt input port and on-chip input debounce filter
- Programmable pull-up, pull-down, and open-drain selection

1.3.19 12-bit Analog-to-Digital Converter (ADC)

ADC of the A31L21x series can convert analog signals to digital signals at a conversion rate of up to 1Msps. 20-channel analog MUX provides various combinations of data from external and internal analog signals.

1.3.20 Comparator

The A31L21x series has two comparator blocks. The block has an internal reference for channels.

1.3.21 LCD driver/controller

An LCD driver supports a resistor bias, capacitor bias with contrast control, and various duties.

1.3.22 Cyclic Redundancy Check (CRC) generator

The A31L21x series has four polynomials for the CRC generator: CRC-CCITT, CRC-8/-16/-32.

1.3.23 Advanced Encryption Standard (AES-128)

The AES-128 is used to encrypt and decrypt data (complies with FIPS PUB 197, 2001 November 26).

1.3.24 True Random Number Generator (TRNG)

The RNG continuously provides 32-bit samples based on an analog noise source.

1.3.25 Temperature Sensor (TS)

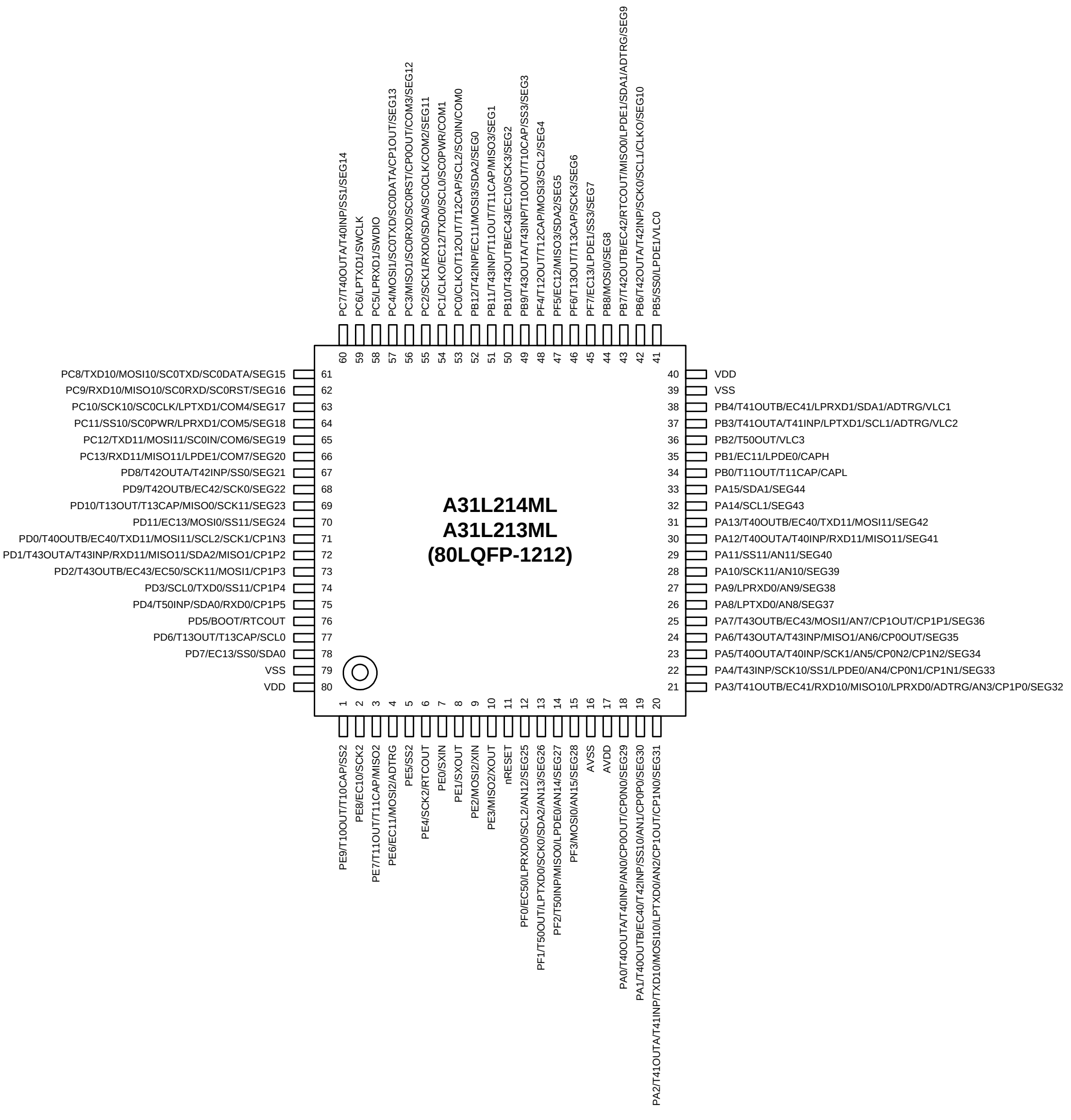
The Temperature Sensor consists of a ring-oscillator. Its frequency varies with temperature.

2 Pinouts and pin descriptions

In chapter 2, pinouts and pin descriptions of the A31L21x series are introduced.

2.1 Pinouts

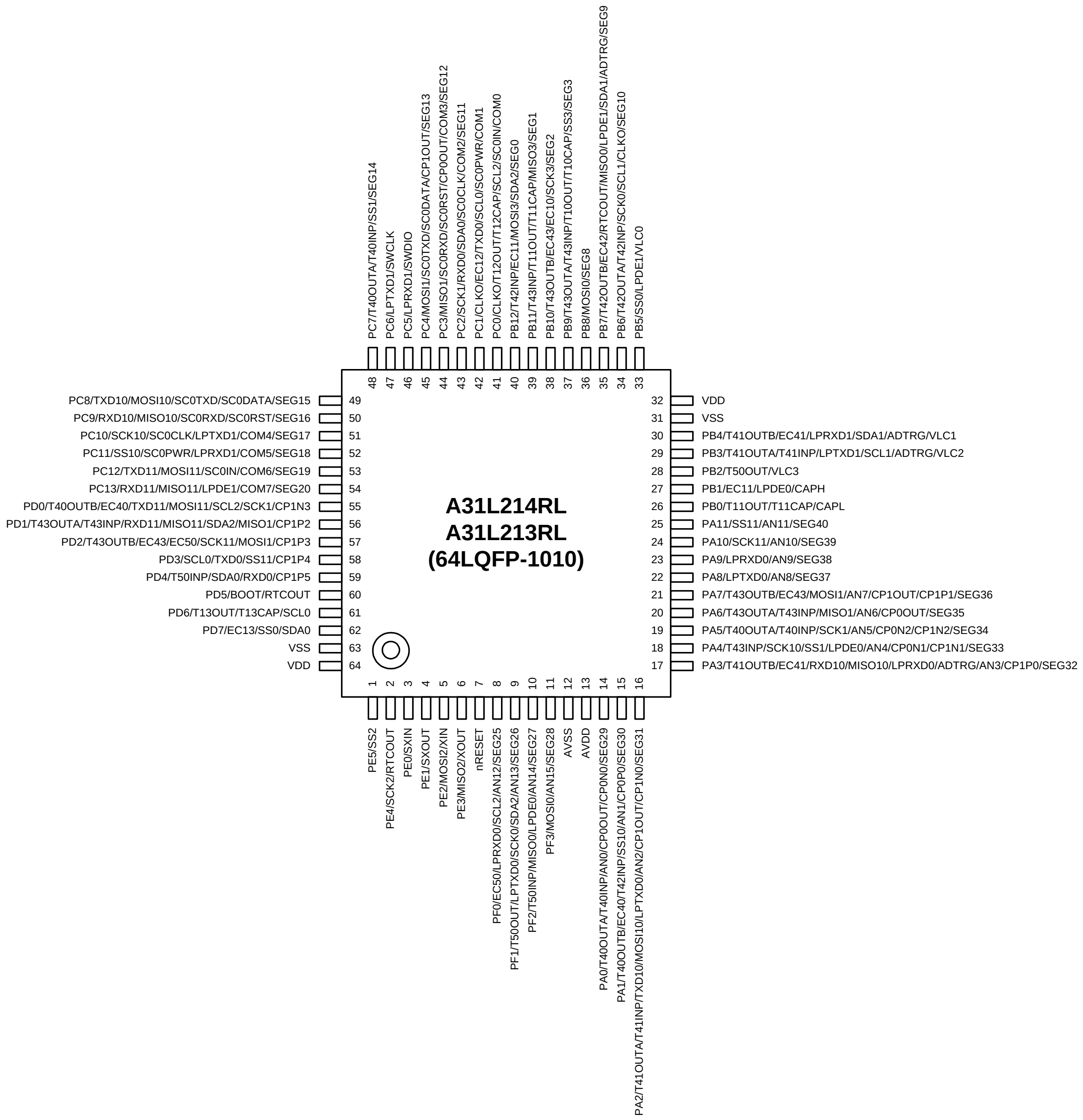
Figure 3. LQFP-80 Pinouts


A31L214ML
A31L213ML
(80LQFP-1212)
1 PE9/T10OUT/T10CAP/SS2
2 PE8/EC10/SCK2
3 PE7/T11OUT/T11CAP/MISO2
4 PE6/EC11/MOSI2/ADTRG
5 PE5/SS2
6 PE4/SCK2/RTCOUT
7 PE0/SXIN
8 PE1/SXOUT
9 PE2/MOSI2/XIN
10 PE3/MISO2/XOUT
11 nRESET
12 PF0/EC50/LPRXD0/SCL2/AN12/SEG25
13 PF1/T50OUT/LPTXD0/SCK0/SDA2/AN13/SEG26
14 PF2/T50INP/MISO0/LPDE0/AN14/SEG27
15 PF3/MOSI0/AN15/SEG28
16 AVSS
17 AVDD
18 PA0/T40OUTA/T40INP/AN0/CP0OUT/CP0N0/SEG29
19 PA1/T40OUTB/EC40/T42INP/SS10/AN1/CP0P0/SEG30
20 PA2/T41OUTA/T41INP/TXD10/MOSI10/LPTXD0/AN2/CP1OUT/CP1N0/SEG31
21 PA3/T41OUTB/EC41/RXD10/MISO10/LPRXD0/ADTRG/AN3/CP1P0/SEG32
22 PA4/T43INP/SCK10/SS1/LPDE0/AN4/CP0N1/CP1N1/SEG33
23 PA5/T40OUTA/T40INP/SCK1/AN5/CP0N2/CP1N2/SEG34
24 PA6/T43OUTA/T43INP/MISO1/AN6/CP0OUT/SEG35
25 PA7/T43OUTB/EC43/MOSI1/AN7/CP1OUT/CP1P1/SEG36
26 PA8/LPTXD0/AN8/SEG37
27 PA9/LPRXD0/AN9/SEG38
28 PA10/SCK11/AN10/SEG39
29 PA11/SS11/AN11/SEG40
30 PA12/T40OUTA/T40INP/RXD11/MISO11/SEG41
31 PA13/T40OUTB/EC40/TXD11/MOSI11/SEG42
32 PA14/SCL1/SEG43
33 PA15/SDA1/SEG44
34 PB0/T11OUT/T11CAP/CAPL
35 PB1/EC11/LPDE0/CAPH
36 PB2/T50OUT/VLC3
37 PB3/T41OUTA/T41INP/LPTXD1/SCL1/ADTRG/VLC2
38 PB4/T41OUTB/EC41/LPRXD1/SDA1/ADTRG/VLC1
39 VSS
40 VDD
41 PB5/SS0/LPDE1/VLC0
42 PB6/T42OUTA/T42INP/SCK0/SCL1/CLKO/SEG10
43 PB7/T42OUTB/EC42/RTCOUT/MISO0/LPDE1/SDA1/ADTRG/SEG9
44 PB8/MOSI0/SEG8
45 PF7/EC13/LPDE1/SS3/SEG7
46 PF6/T13OUT/T13CAP/SCK3/SEG6
47 PF5/EC12/MISO3/SDA2/SEG5
48 PF4/T12OUT/T12CAP/MOSI3/SCL2/SEG4
49 PB9/T43OUTA/T43INP/T10OUT/T10CAP/SS3/SEG3
50 PB10/T43OUTB/EC43/EC10/SCK3/SEG2
51 PB11/T43INP/T11OUT/T11CAP/MISO3/SEG1
52 PB12/T42INP/EC11/MOSI3/SDA2/SEG0
53 PC0/CLKO/T12OUT/T12CAP/SCL2/SC0IN/COM0
54 PC1/CLKO/EC12/TXD0/SCL0/SC0PWR/COM1
55 PC2/SCK1/RXD0/SDA0/SC0CLK/COM2/SEG11
56 PC3/MISO1/SC0RXD/SC0RST/CP0OUT/COM3/SEG12
57 PC4/MOSI1/SC0TXD/SC0DATA/CP1OUT/SEG13
58 PC5/LPRXD1/SWDIO
59 PC6/LPTXD1/SWCLK
60 PC7/T40OUTA/T40INP/SS1/SEG14
61 PC8/TXD10/MOSI10/SC0TXD/SC0DATA/SEG15
62 PC9/RXD10/MISO10/SC0RXD/SC0RST/SEG16
63 PC10/SCK10/SC0CLK/LPTXD1/COM4/SEG17
64 PC11/SS10/SC0PWR/LPRXD1/COM5/SEG18
65 PC12/TXD11/MOSI11/SC0IN/COM6/SEG19
66 PC13/RXD11/MISO11/LPDE1/COM7/SEG20
67 PD8/T42OUTA/T42INP/SS0/SEG21
68 PD9/T42OUTB/EC42/SCK0/SEG22
69 PD10/T13OUT/T13CAP/MISO0/SCK11/SEG23
70 PD11/EC13/MOSI0/SS11/SEG24
71 PD0/T40OUTB/EC40/TXD11/MOSI11/SCL2/SCK1/CP1N3
72 PD1/T43OUTA/T43INP/RXD11/MISO11/SDA2/MISO1/CP1P2
73 PD2/T43OUTB/EC43/EC50/SCK11/MOSI1/CP1P3
74 PD3/SCL0/TXD0/SS11/CP1P4
75 PD4/T50INP/SDA0/RXD0/CP1P5
76 PD5/BOOT/RTCOUT
77 PD6/T13OUT/T13CAP/SCL0
78 PD7/EC13/SS0/SDA0
79 VSS
80 VDD

NOTES:

1. PD[4:0] pins are 5V tolerant I/Os. The corresponding internal resistors of these pins must be disabled by S/W to use these pins as 5V I/Os. To use these pins as 5V tolerant outputs, external pull-up resistors must be used by applications and the corresponding pins must be configured as open-drain outputs by S/W.
2. The VDD should be greater than or equal to 2.0V and the corresponding pins are set to push-pull outputs by the PD_TYP[4:0] bits if CP1P[5:2] and CP1N3 are used as comparator pins for the alternative function.
3. The PA0, PA2, PE4, and PE6 pins can be wakeup sources from DEEP SLEEP with ALLPWR = 1/2/3.

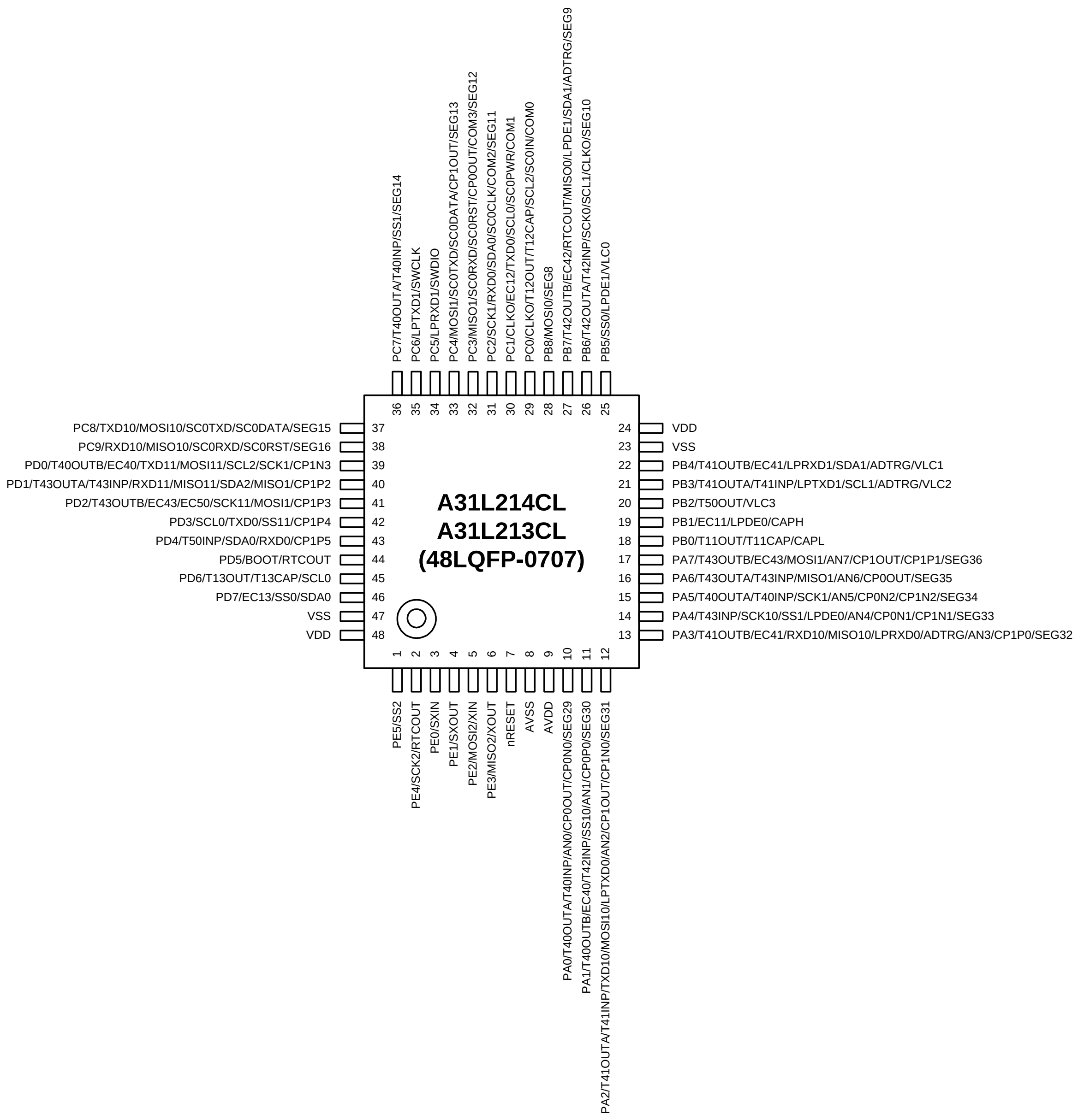
Figure 4. LQFP-64 Pinouts


A31L214RL
A31L213RL
(64LQFP-1010)
48 PC7/T40OUTA/T40INP/SS1/SEG14
47 PC6/LPTXD1/SWCLK
46 PC5/LPRXD1/SWDIO
45 PC4/MOSI1/SC0TXD/SC0DATA/CP1OUT/SEG13
44 PC3/MISO1/SC0RXD/SC0RST/CP0OUT/COM3/SEG12
43 PC2/SCK1/RXD0/SDA0/SC0CLK/COM2/SEG11
42 PC1/CLKO/EC12/TXD0/SCL0/SC0PWR/COM1
41 PC0/CLKO/T12OUT/T12CAP/SCL2/SC0IN/COM0
40 PB12/T42INP/EC11/MOSI3/SDA2/SEG0
39 PB11/T43INP/T11OUT/T11CAP/MISO3/SEG1
38 PB10/T43OUTB/EC43/EC10/SCK3/SEG2
37 PB9/T43OUTA/T43INP/T10OUT/T10CAP/SS3/SEG3
36 PB8/MOSI0/SEG8
35 PB7/T42OUTB/EC42/RTCOUT/MISO0/LPDE1/SDA1/ADTRG/SEG9
34 PB6/T42OUTA/T42INP/SCK0/SCL1/CLKO/SEG10
33 PB5/SS0/LPDE1/VLC0
49 PC8/TXD10/MOSI10/SC0TXD/SC0DATA/SEG15
50 PC9/RXD10/MISO10/SC0RXD/SC0RST/SEG16
51 PC10/SCK10/SC0CLK/LPTXD1/COM4/SEG17
52 PC11/SS10/SC0PWR/LPRXD1/COM5/SEG18
53 PC12/TXD11/MOSI11/SC0IN/COM6/SEG19
54 PC13/RXD11/MISO11/LPDE1/COM7/SEG20
55 PD0/T40OUTB/EC40/TXD11/MOSI11/SCL2/SCK1/CP1N3
56 PD1/T43OUTA/T43INP/RXD11/MISO11/SDA2/MISO1/CP1P2
57 PD2/T43OUTB/EC43/EC50/SCK11/MOSI1/CP1P3
58 PD3/SCL0/TXD0/SS11/CP1P4
59 PD4/T50INP/SDA0/RXD0/CP1P5
60 PD5/BOOT/RTCOUT
61 PD6/T13OUT/T13CAP/SCL0
62 PD7/EC13/SS0/SDA0
63 VSS
64 VDD
32 VDD
31 VSS
30 PB4/T41OUTB/EC41/LPRXD1/SDA1/ADTRG/VLC1
29 PB3/T41OUTA/T41INP/LPTXD1/SCL1/ADTRG/VLC2
28 PB2/T50OUT/VLC3
27 PB1/EC11/LPDE0/CAPH
26 PB0/T11OUT/T11CAP/CAPL
25 PA11/SS11/AN11/SEG40
24 PA10/SCK11/AN10/SEG39
23 PA9/LPRXD0/AN9/SEG38
22 PA8/LPTXD0/AN8/SEG37
21 PA7/T43OUTB/EC43/MOSI1/AN7/CP1OUT/CP1P1/SEG36
20 PA6/T43OUTA/T43INP/MISO1/AN6/CP0OUT/SEG35
19 PA5/T40OUTA/T40INP/SCK1/AN5/CP0N2/CP1N2/SEG34
18 PA4/T43INP/SCK10/SS1/LPDE0/AN4/CP0N1/CP1N1/SEG33
17 PA3/T41OUTB/EC41/RXD10/MISO10/LPRXD0/ADTRG/AN3/CP1P0/SEG32
1 PE5/SS2
2 PE4/SCK2/RTCOUT
3 PE0/SXIN
4 PE1/SXOUT
5 PE2/MOSI2/XIN
6 PE3/MISO2/XOUT
7 nRESET
8 PF0/EC50/LPRXD0/SCL2/AN12/SEG25
9 PF1/T50OUT/LPTXD0/SCK0/SDA2/AN13/SEG26
10 PF2/T50INP/MISO0/LPDE0/AN14/SEG27
11 PF3/MOSI0/AN15/SEG28
12 AVSS
13 AVDD
14 PA0/T40OUTA/T40INP/AN0/CP0OUT/CP0N0/SEG29
15 PA1/T40OUTB/EC40/T42INP/SS10/AN1/CP0P0/SEG30
16 PA2/T41OUTA/T41INP/TXD10/MOSI10/LPTXD0/AN2/CP1OUT/CP1N0/SEG31

NOTES:

1. Each of the PA[15:12], PD[11:8], PE[9:6] and PF[7:4] pins should be selected as a push-pull output or an input with pull-up or pull-down resistor by software control when the 64LQFP package is used.
2. PD[4:0] pins are 5V tolerant I/Os. The corresponding internal resistors of these pins must be disabled by S/W to use these pins as 5V I/Os. To use these pins as 5V tolerant outputs, external pull-up resistors must be used by applications and the corresponding pins must be configured as open-drain outputs by S/W
3. The VDD should be greater than or equal to 2.0V and the corresponding pins are set to push-pull outputs by the PD_TYP[4:0] bits if CP1P[5:2] and CP1N3 are used as comparator pins for the alternative function.
4. The PA0, PA2, and PE4 pins can be wakeup sources from DEEP SLEEP with ALLPWR = 1/2/3.

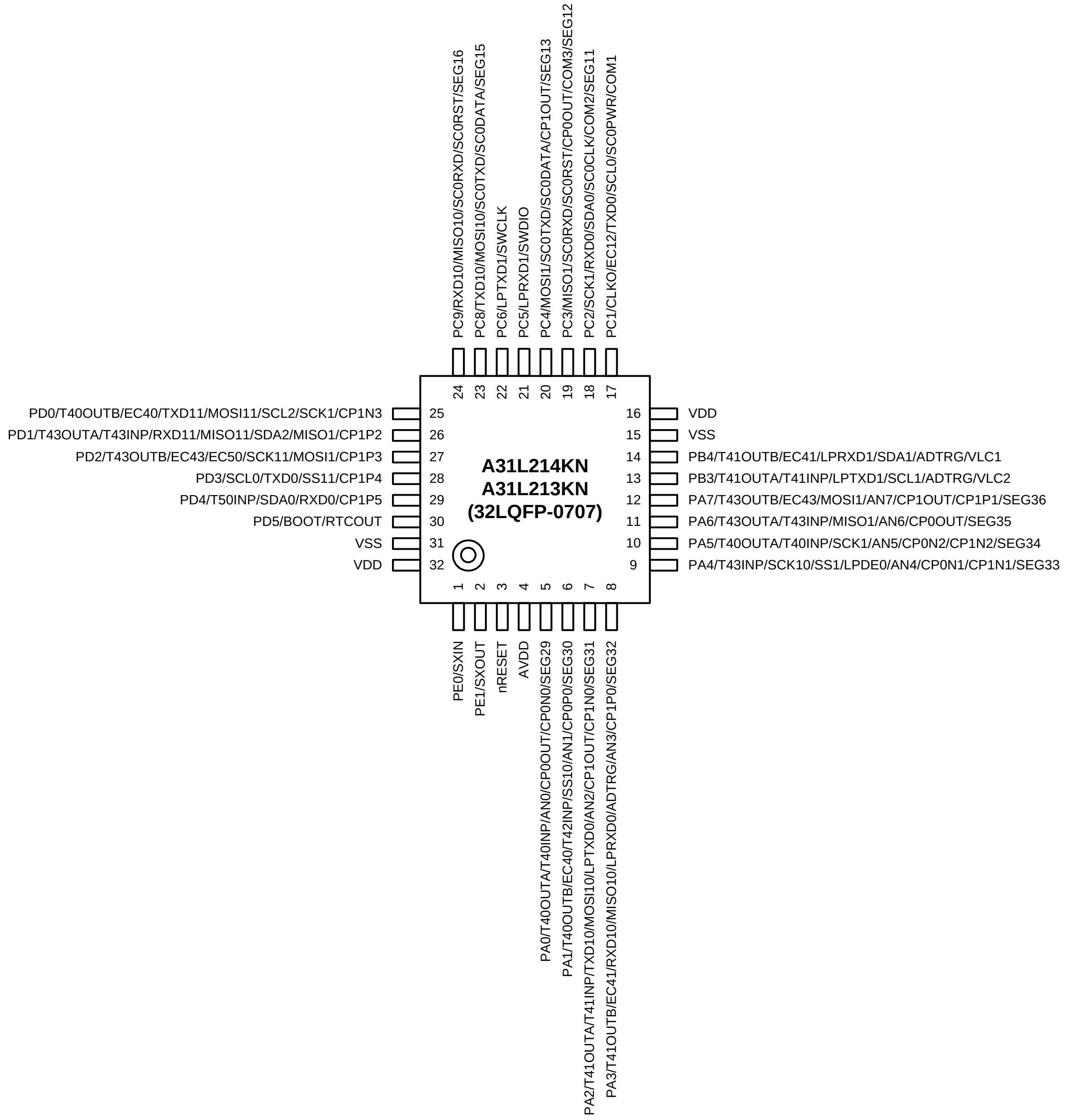
Figure 5. LQFP-48 Pinouts


A31L214CL
A31L213CL
(48LQFP-0707)
1 PE5/SS2
2 PE4/SCK2/RTCOUT
3 PE0/SXIN
4 PE1/SXOUT
5 PE2/MOSI2/XIN
6 PE3/MISO2/XOUT
7 nRESET
8 AVSS
9 AVDD
10 PA0/T40OUTA/T40INP/AN0/CP0OUT/CP0N0/SEG29
11 PA1/T40OUTB/EC40/T42INP/SS10/AN1/CP0P0/SEG30
12 PA2/T41OUTA/T41INP/TXD10/MOSI10/LPTXD0/AN2/CP1OUT/CP1N0/SEG31
13 PA3/T41OUTB/EC41/RXD10/MISO10/LPRXD0/ADTRG/AN3/CP1P0/SEG32
14 PA4/T43INP/SCK10/SS1/LPDE0/AN4/CP0N1/CP1N1/SEG33
15 PA5/T40OUTA/T40INP/SCK1/AN5/CP0N2/CP1N2/SEG34
16 PA6/T43OUTA/T43INP/MISO1/AN6/CP0OUT/SEG35
17 PA7/T43OUTB/EC43/MOSI1/AN7/CP1OUT/CP1P1/SEG36
18 PB0/T11OUT/T11CAP/CAPL
19 PB1/EC11/LPDE0/CAPH
20 PB2/T50OUT/VLC3
21 PB3/T41OUTA/T41INP/LPTXD1/SCL1/ADTRG/VLC2
22 PB4/T41OUTB/EC41/LPRXD1/SDA1/ADTRG/VLC1
23 VSS
24 VDD
25 PB5/SS0/LPDE1/VLC0
26 PB6/T42OUTA/T42INP/SCK0/SCL1/CLKO/SEG10
27 PB7/T42OUTB/EC42/RTCOUT/MISO0/LPDE1/SDA1/ADTRG/SEG9
28 PB8/MOSI0/SEG8
29 PC0/CLKO/T12OUT/T12CAP/SCL2/SC0IN/COM0
30 PC1/CLKO/EC12/TXD0/SCL0/SC0PWR/COM1
31 PC2/SCK1/RXD0/SDA0/SC0CLK/COM2/SEG11
32 PC3/MISO1/SC0RXD/SC0RST/CP0OUT/COM3/SEG12
33 PC4/MOSI1/SC0TXD/SC0DATA/CP1OUT/SEG13
34 PC5/LPRXD1/SWDIO
35 PC6/LPTXD1/SWCLK
36 PC7/T40OUTA/T40INP/SS1/SEG14
37 PC8/TXD10/MOSI10/SC0TXD/SC0DATA/SEG15
38 PC9/RXD10/MISO10/SC0RXD/SC0RST/SEG16
39 PD0/T40OUTB/EC40/TXD11/MOSI11/SCL2/SCK1/CP1N3
40 PD1/T43OUTA/T43INP/RXD11/MISO11/SDA2/MISO1/CP1P2
41 PD2/T43OUTB/EC43/EC50/SCK11/MOSI1/CP1P3
42 PD3/SCL0/TXD0/SS11/CP1P4
43 PD4/T50INP/SDA0/RXD0/CP1P5
44 PD5/BOOT/RTCOUT
45 PD6/T13OUT/T13CAP/SCL0
46 PD7/EC13/SS0/SDA0
47 VSS
48 VDD

NOTES:

1. Each of the PA[15:8], PB[12:9], PC[13:10], PD[11:8], PE[9:6] and PF[7:0] pins should be selected as a push-pull output or an input with pull-up or pull-down resistor by software control when the 48LQFP package is used.
2. PD[4:0] pins are 5V tolerant I/Os. The corresponding internal resistors of these pins must be disabled by S/W to use these pins as 5V I/Os. To use these pins as 5V tolerant outputs, external pull-up resistors must be used by applications and the corresponding pins must be configured as open-drain outputs by S/W
3. The VDD should be greater than or equal to 2.0V and the corresponding pins are set to push-pull outputs by the PD_TYP[4:0] bits if CP1P[5:2] and CP1N3 are used as comparator pins for the alternative function.
4. The PA0, PA2, and PE4 pins can be wakeup sources from DEEP SLEEP with ALLPWR = 1/2/3.

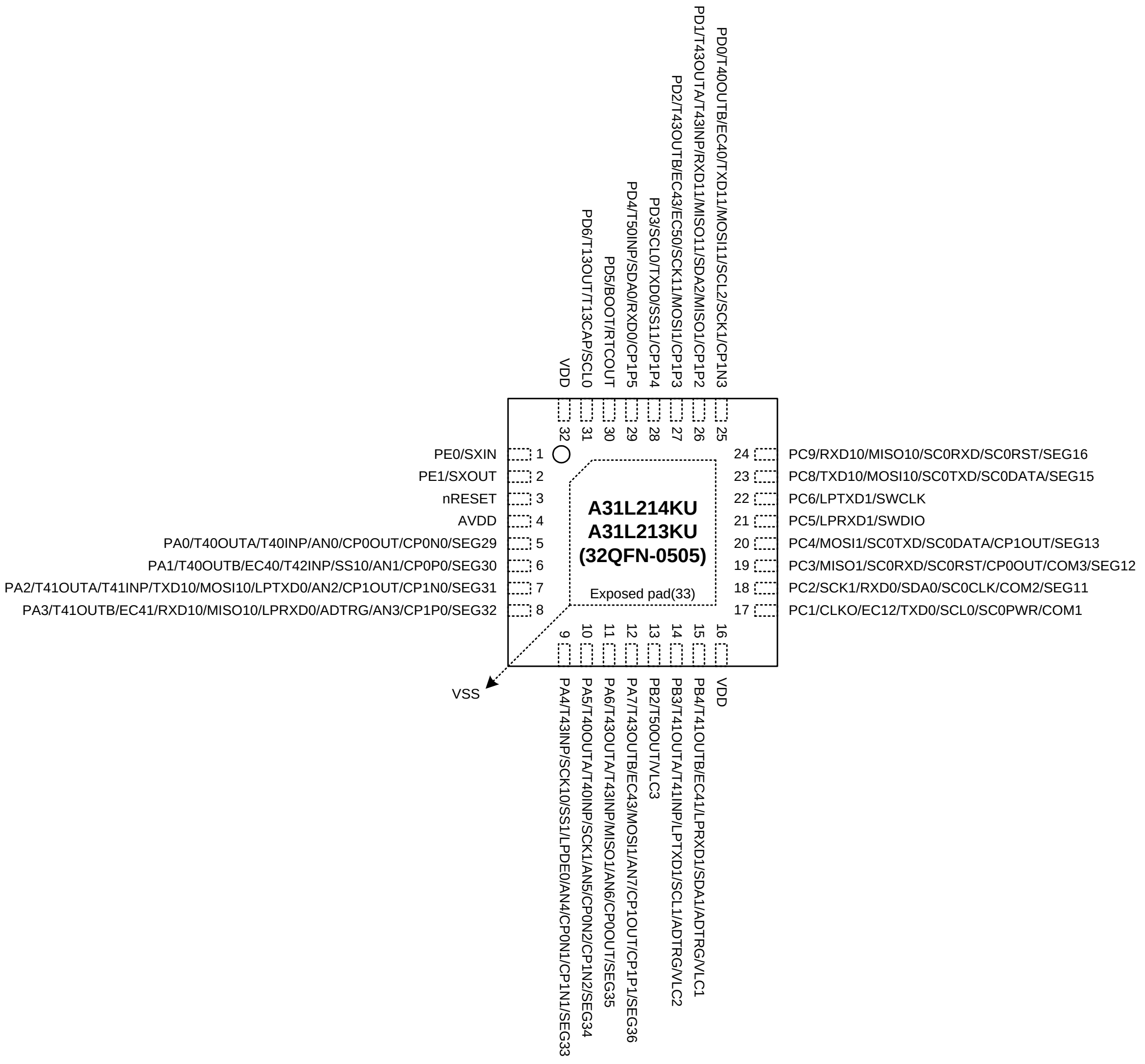
Figure 6. LQFP-32 Pinouts


PC9/RXD10/MISO10/SC0RXD/SC0RST/SEG16
PC8/TXD10/MOSI10/SC0TXD/SC0DATA/SEG15
PC6/LPTXD1/SWCLK
PC5/LPRXD1/SWDIO
PC4/MOSI1/SC0TXD/SC0DATA/CP1OUT/SEG13
PC3/MISO1/SC0RXD/SC0RST/CP0OUT/COM3/SEG12
PC2/SCK1/RXD0/SDA0/SC0CLK/COM2/SEG11
PC1/CLKO/EC12/TXD0/SCL0/SC0PWR/COM1
24
23
22
21
20
19
18
17
PD0/T40OUTB/EC40/TXD11/MOSI11/SCL2/SCK1/CP1N3
PD1/T43OUTA/T43INP/RXD11/MISO11/SDA2/MISO1/CP1P2
PD2/T43OUTB/EC43/EC50/SCK11/MOSI1/CP1P3
PD3/SCL0/TXD0/SS11/CP1P4
PD4/T50INP/SDA0/RXD0/CP1P5
PD5/BOOT/RTCOUT
VSS
VDD
25
26
27
28
29
30
31
32
A31L214KN
A31L213KN
(32LQFP-0707)
16
15
14
13
12
11
10
9
VDD
VSS
PB4/T41OUTB/EC41/LPRXD1/SDA1/ADTRG/VLC1
PB3/T41OUTA/T41INP/LPTXD1/SCL1/ADTRG/VLC2
PA7/T43OUTB/EC43/MOSI1/AN7/CP1OUT/CP1P1/SEG36
PA6/T43OUTA/T43INP/MISO1/AN6/CP0OUT/SEG35
PA5/T40OUTA/T40INP/SCK1/AN5/CP0N2/CP1N2/SEG34
PA4/T43INP/SCK10/SS1/LPDE0/AN4/CP0N1/CP1N1/SEG33
1
2
3
4
5
6
7
8
PE0/SXIN
PE1/SXOUT
nRESET
AVDD
PA0/T40OUTA/T40INP/AN0/CP0OUT/CP0N0/SEG29
PA1/T40OUTB/EC40/T42INP/SS10/AN1/CP0P0/SEG30
PA2/T41OUTA/T41INP/TXD10/MOSI10/LPTXD0/AN2/CP1OUT/CP1N0/SEG31
PA3/T41OUTB/EC41/RXD10/MISO10/LPRXD0/ADTRG/AN3/CP1P0/SEG32

NOTES:

1. Each of the PA[15:8], PB[12:5, 2:0], PC[13:10, 7, 0], PD[11:6], PE[9:2] and PF[7:0] pins should be selected as a push-pull output or an input with pull-up or pull-down resistor by software control when the 32LQFP package is used.
2. PD[4:0] pins are 5V tolerant I/Os. The corresponding internal resistors of these pins must be disabled by S/W to use these pins as 5V I/Os. To use these pins as 5V tolerant outputs, external pull-up resistors must be used by applications and the corresponding pins must be configured as open-drain outputs by S/W
3. The VDD should be greater than or equal to 2.0V and the corresponding pins are set to push-pull outputs by the PD_TYP[4:0] bits if CP1P[5:2] and CP1N3 are used as comparator pins for the alternative function.
4. The PA0 and PA2 pins can be wakeup sources from DEEP SLEEP with ALLPWR = 1/2/3.

Figure 7. QFN-32 Pinouts


PD0/T40OUTB/EC40/TXD11/MOSI11/SCL2/SCK1/CP1N3
PD1/T43OUTA/T43INP/RXD11/MISO11/SDA2/MISO1/CP1P2
PD2/T43OUTB/EC43/EC50/SCK11/MOSI1/CP1P3
PD3/SCL0/TXD0/SS11/CP1P4
PD4/T50INP/SDA0/RXD0/CP1P5
PD5/BOOT/RTCOUT
PD6/T13OUT/T13CAP/SCL0
VDD
32
31
30
29
28
27
26
25
PE0/SXIN 1
PE1/SXOUT 2
nRESET 3
AVDD 4
PA0/T40OUTA/T40INP/AN0/CP0OUT/CP0N0/SEG29 5
PA1/T40OUTB/EC40/T42INP/SS10/AN1/CP0P0/SEG30 6
PA2/T41OUTA/T41INP/TXD10/MOSI10/LPTXD0/AN2/CP1OUT/CP1N0/SEG31 7
PA3/T41OUTB/EC41/RXD10/MISO10/LPRXD0/ADTRG/AN3/CP1P0/SEG32 8
A31L214KU
A31L213KU
(32QFN-0505)
Exposed pad(33)
24 PC9/RXD10/MISO10/SC0RXD/SC0RST/SEG16
23 PC8/TXD10/MOSI10/SC0TXD/SC0DATA/SEG15
22 PC6/LPTXD1/SWCLK
21 PC5/LPRXD1/SWDIO
20 PC4/MOSI1/SC0TXD/SC0DATA/CP1OUT/SEG13
19 PC3/MISO1/SC0RXD/SC0RST/CP0OUT/COM3/SEG12
18 PC2/SCK1/RXD0/SDA0/SC0CLK/COM2/SEG11
17 PC1/CLKO/EC12/TXD0/SCL0/SC0PWR/COM1
9
10
11
12
13
14
15
16
VSS
PA4/T43INP/SCK10/SS1/LPDE0/AN4/CP0N1/CP1N1/SEG33
PA5/T40OUTA/T40INP/SCK1/AN5/CP0N2/CP1N2/SEG34
PA6/T43OUTA/T43INP/MISO1/AN6/CP0OUT/SEG35
PA7/T43OUTB/EC43/MOSI1/AN7/CP1OUT/CP1P1/SEG36
PB2/T50OUT/VLC3
PB3/T41OUTA/T41INP/LPTXD1/SCL1/ADTRG/VLC2
PB4/T41OUTB/EC41/LPRXD1/SDA1/ADTRG/VLC1
VDD

NOTES:

1. Each of the PA[15:8], PB[12:5, 1:0], PC[13:10, 7, 0], PD[11:7], PE[9:2] and PF[7:0] pins should be selected as a push-pull output or an input with pull-up or pull-down resistor by software control when the 32QFN package is used.
2. PD[4:0] pins are 5V tolerant I/Os. The corresponding internal resistors of these pins must be disabled by S/W to use these pins as 5V I/Os. To use these pins as 5V tolerant outputs, external pull-up resistors must be used by applications and the corresponding pins must be configured as open-drain outputs by S/W
3. The VDD should be greater than or equal to 2.0V and the corresponding pins are set to push-pull outputs by the PD_TYP[4:0] bits if CP1P[5:2] and CP1N3 are used as comparator pins for the alternative function.
4. The PA0 and PA2 pins can be wakeup sources from DEEP SLEEP with ALLPWR = 1/2/3.

2.2 Pin description

Table 2 shows pin configuration containing several pairs of power/ground and other dedicated pins. Multi-function pins have up to nine selections of functions including GPIO.

Table 2. Pin Description

Pin number					Pin name	Type	Description	Remark
LQFP-80	LQFP-64	LQFP-48	LQFP-32	QFN-32				
1	-	-	-	-	PE9*	IOUDS	PORT E Bit 9 Input/Output	
					T10OUT	O	Timer 10 pulse output	
					T10CAP	I	Timer 10 capture input	
					SS2	I	SPI slave select input	
2	-	-	-	-	PE8*	IOUDS	PORT E Bit 8 Input/Output	
					EC10	I	Timer 10 event count input	
					SCK2	I/O	SPI clock input/output	
3	-	-	-	-	PE7*	IOUDS	PORT E Bit 7 Input/Output	
					T11OUT	O	Timer 11 pulse output	
					T11CAP	I	Timer 11 capture input	
					MISO2	I/O	SPI master input, slave output	
4	-	-	-	-	PE6*	IOUDS	PORT E Bit 6 Input/Output	Wake-up possible from DEEP SLEEP with ALLPWR=1/2/3
					EC11	I	Timer 11 event count input	
					MOSI2	I/O	SPI master output, slave input	
					ADTRG	I	ADC trigger input	
5	1	1	-	-	PE5*	IOUDS	PORT E Bit 5 Input/Output	
					SS2	I	SPI slave select input	
6	2	2	-	-	PE4*	IOUDS	PORT E Bit 4 Input/Output	Wake-up possible from DEEP SLEEP with ALLPWR=1/2/3
					SCK2	I/O	SPI clock input/output	
					RTCOUT	O	Real time clock output	
7	3	3	1	1	PE0*	IOUDS	PORT E Bit 0 Input/Output	
					SXIN	IA	Sub Oscillator Input	
8	4	4	2	2	PE1*	IOUDS	PORT E Bit 1 Input/Output	
					SXOUT	OA	Sub Oscillator Output	
9	5	5	-	-	PE2*	IOUDS	PORT E Bit 2 Input/Output	
					MOSI2	I/O	SPI master output, slave input	
					XIN	IA	Main oscillator input	
10	6	6	-	-	PE3*	IOUDS	PORT E Bit 3 Input/Output	
					MISO2	I/O	SPI master input, slave output	
					XOUT	OA	Main oscillator output	
11	7	7	3	3	nRESET	Input	External Reset Input	Always pull-up

Table 2. Pin Description (continued)

Pin number					Pin name	Type	Description	Remark
LQFP-80	LQFP-64	LQFP-48	LQFP-32	QFN-32				
12	8	-	-	-	PF0*	IOUDS	PORT F Bit 0 Input/Output	
					EC50	I	Timer 50 event count input	
					LPRXD0	I	Low power UART data input	
					SCL2	I/O	I2C clock input/output	
					AN12	IA	A/D converter analog input channel	
					SEG25	OA	LCD segment signal output	
13	9	-	-	-	PF1*	IOUDS	PORT F Bit 1 Input/Output	
					T50OUT	O	Timer 50 pulse output	
					LPTXD0	O	Low power UART data output	
					SCK0	I/O	SPI clock input/output	
					SDA2	I/O	I2C data input/output	
					AN13	IA	A/D converter analog input channel	
					SEG26	OA	LCD segment signal output	
14	10	-	-	-	PF2*	IOUDS	PORT F Bit 2 Input/Output	
					T50INP	I	Timer 50 capture/clear input	
					MISO0	I/O	SPI master input, slave output	
					LPDE0	O	Low power UART DE signal output	
					AN14	IA	A/D converter analog input channel	
					SEG27	OA	LCD segment signal output	
15	11	-	-	-	PF3*	IOUDS	PORT F Bit 3 Input/Output	
					MOSI0	I/O	SPI master output, slave input	
					AN15	IA	A/D converter analog input channel	
					SEG28	OA	LCD segment signal output	
16	12	8	-	-	AVSS	PA	Analog Ground	
17	13	9	4	4	AVDD	PA	Analog Power	
18	14	10	5	5	PA0*	IOUDS	PORT A Bit 0 Input/Output	Wake-up possible from DEEP SLEEP with ALLPWR=1/2/3
					T40OUTA	O	Timer 40 pulse output	
					T40INP	I	Timer 40 capture/force input	
					AN0	IA	A/D converter analog input channel	
					CP0OUT	OA	Comparator 0 output	
					CP0N0	IA	Comparator 0 negative input	
					SEG29	OA	LCD segment signal output	
19	15	11	6	6	PA1*	IOUDS	PORT A Bit 1 Input/Output	
					T40OUTB	O	Timer 40 pulse output	
					EC40	I	Timer 40 event count input	
					T42INP	I	Timer 42 capture/force input	
					SS10	I	SPI slave select input	
					AN1	IA	A/D converter analog input channel	
					CP0P0	IA	Comparator 0 positive input	
					SEG30	OA	LCD segment signal output	

Table 2. Pin Description (continued)

Pin number					Pin name	Type	Description	Remark
LQFP-80	LQFP-64	LQFP-48	LQFP-32	QFN-32				
20	16	12	7	7	PA2*	IOUDS	PORT A Bit 2 Input/Output	Wake-up possible from DEEP SLEEP with ALLPWR=1/2/3
					T41OUTA	O	Timer 41 pulse output	
					T41INP	I	Timer 41 capture/force input	
					TXD10	O	UART data output	
					MOSI10	I/O	SPI master output, slave input	
					LPTXD0	O	Low power UART data output	
					AN2	IA	A/D converter analog input channel	
					CP1OUT	OA	Comparator 1 output	
					CP1N0	IA	Comparator 1 negative input	
					SEG31	OA	LCD segment signal output	
21	17	13	8	8	PA3*	IOUDS	PORT A Bit 3 Input/Output	
					T41OUTB	O	Timer 41 pulse output	
					EC41	I	Timer 41 event count input	
					RXD10	I	UART data input	
					MISO10	I/O	SPI master input, slave output	
					LPRXD0	I	Low power UART data input	
					ADTRG	I	ADC trigger input	
					AN3	IA	A/D converter analog input channel	
					CP1P0	IA	Comparator 1 positive input	
					SEG32	OA	LCD segment signal output	
22	18	14	9	9	PA4*	IOUDS	PORT A Bit 4 Input/Output	
					T43INP	I	Timer 43 capture/force input	
					SCK10	I/O	SPI clock input/output	
					SS1	I	SPI slave select input	
					LPDE0	O	Low power UART DE signal output	
					AN4	IA	A/D converter analog input channel	
					CP0N1	IA	Comparator 0 negative input	
					CP1N1	IA	Comparator 1 negative input	
					SEG33	OA	LCD segment signal output	
23	19	15	10	10	PA5*	IOUDS	PORT A Bit 5 Input/Output	
					T40OUTA	O	Timer 40 pulse output	
					T40INP	I	Timer 40 capture/force input	
					SCK1	I/O	SPI clock input/output	
					AN5	IA	A/D converter analog input channel	
					CP0N2	IA	Comparator 0 negative input	
					CP1N2	IA	Comparator 1 negative input	
					SEG34	OA	LCD segment signal output	

Table 2. Pin Description (continued)

Pin number					Pin name	Type	Description	Remark
LQFP-80	LQFP-64	LQFP-48	LQFP-32	QFN-32				
24	20	16	11	11	PA6*	IOUDS	PORT A Bit 6 Input/Output	
					T43OUTA	O	Timer 43 pulse output	
					T43INP	I	Timer 43 capture/force input	
					MISO1	I/O	SPI master input, slave output	
					AN6	IA	A/D converter analog input channel	
					CP0OUT	OA	Comparator 0 output	
					SEG35	OA	LCD segment signal output	
25	21	17	12	12	PA7*	IOUDS	PORT A Bit 7 Input/Output	
					T43OUTB	O	Timer 43 pulse output	
					EC43	I	Timer 43 event count input	
					MOSI1	I/O	SPI master output, slave input	
					AN7	IA	A/D converter analog input channel	
					CP1OUT	OA	Comparator 1 output	
					CP1P1	IA	Comparator 1 positive input	
					SEG36	OA	LCD segment signal output	
26	22	-	-	-	PA8*	IOUDS	PORT A Bit 8 Input/Output	
					LPTXD0	O	Low power UART data output	
					AN8	IA	A/D converter analog input channel	
					SEG37	OA	LCD segment signal output	
27	23	-	-	-	PA9*	IOUDS	PORT A Bit 9 Input/Output	
					LPRXD0	I	Low power UART data input	
					AN9	IA	A/D converter analog input channel	
					SEG38	OA	LCD segment signal output	
28	24	-	-	-	PA10*	IOUDS	PORT A Bit 10 Input/Output	
					SCK11	I/O	SPI clock input/output	
					AN10	IA	A/D converter analog input channel	
					SEG39	OA	LCD segment signal output	
29	25	-	-	-	PA11*	IOUDS	PORT A Bit 11 Input/Output	
					SS11	I	SPI slave select input	
					AN11	IA	A/D converter analog input channel	
					SEG40	OA	LCD segment signal output	
30	-	-	-	-	PA12*	IOUDS	PORT A Bit 12 Input/Output	
					T40OUTA	O	Timer 40 pulse output	
					T40INP	I	Timer 40 capture/force input	
					RXD11	I	UART data input	
					MISO11	I/O	SPI master input, slave output	
					SEG41	OA	LCD segment signal output	

Table 2. Pin Description (continued)

Pin number					Pin name	Type	Description	Remark
LQFP-80	LQFP-64	LQFP-48	LQFP-32	QFN-32				
31	-	-	-	-	PA13*	IOUDS	PORT A Bit 13 Input/Output	
					T40OUTB	O	Timer 40 pulse output	
					EC40	I	Timer 40 event count input	
					TXD11	O	UART data output	
					MOSI11	I/O	SPI master output, slave input	
					SEG42	OA	LCD segment signal output	
32	-	-	-	-	PA14*	IOUDS	PORT A Bit 14 Input/Output	
					SCL1	I/O	I2C clock input/output	
					SEG43	OA	LCD segment signal output	
33	-	-	-	-	PA15*	IOUDS	PORT A Bit 15 Input/Output	
					SDA1	I/O	I2C data input/output	
					SEG44	OA	LCD segment signal output	
34	26	18	-	-	PB0*	IOUDS	PORT B Bit 0 Input/Output	
					T11OUT	O	Timer 11 pulse output	
					T11CAP	I	Timer 11 capture input	
					CAPL	I	Capacitor terminal for voltage booster	
35	27	19	-	-	PB1*	IOUDS	PORT B Bit 1 Input/Output	
					EC11	I	Timer 11 event count input	
					LPDE0	O	Low power UART DE signal output	
					CAPH	I	Capacitor terminal for voltage booster	
36	28	20	-	13	PB2*	IOUDS	PORT B Bit 2 Input/Output	
					T50OUT	O	Timer 50 pulse output	
					VLC3	IA/OA	LCD bias voltage input/output	
37	29	21	13	14	PB3*	IOUDS	PORT B Bit 3 Input/Output	
					T41OUTA	O	Timer 41 pulse output	
					T41INP	I	Timer 41 capture/force input	
					LPTXD1	O	Low power UART data output	
					SCL1	I/O	I2C clock input/output	
					ADTRG	I	ADC trigger input	
					VLC2	IA/OA	LCD bias voltage input/output	
38	30	22	14	15	PB4*	IOUDS	PORT B Bit 4 Input/Output	
					T41OUTB	O	Timer 41 pulse output	
					EC41	I	Timer 41 event count input	
					LPRXD1	I	Low power UART data input	
					SDA1	I/O	I2C data input/output	
					ADTRG	I	ADC trigger input	
					VLC1	IA/OA	LCD bias voltage input/output	
39	31	23	15	-	VSS	P	Ground	
40	32	24	16	16	VDD	P	VDD	

Table 2. Pin Description (continued)

Pin number					Pin name	Type	Description	Remark
LQFP-80	LQFP-64	LQFP-48	LQFP-32	QFN-32				
41	33	25	-	-	PB5*	IOUDS	PORT B Bit 5 Input/Output	
					SS0	I	SPI slave select input	
					LPDE1	O	Low power UART DE signal output	
					VLC0	IA/OA	LCD bias voltage input/output	
42	34	26	-	-	PB6*	IOUDS	PORT B Bit 6 Input/Output	
					T42OUTA	O	Timer 42 pulse output	
					T42INP	I	Timer 42 event count input	
					SCK0	I/O	SPI clock input/output	
					SCL1	I/O	I2C clock input/output	
					CLKO	O	System clock output	
					SEG10	OA	LCD segment signal output	
43	35	27	-	-	PB7*	IOUDS	PORT B Bit 7 Input/Output	
					T42OUTB	O	Timer 42 pulse output	
					EC42	I	Timer 42 event count input	
					RTCOUT	O	Real time clock output	
					MISO0	I/O	SPI master input, slave output	
					LPDE1	O	Low power UART DE signal output	
					SDA1	I/O	I2C data input/output	
					ADTRG	I	ADC trigger input	
					SEG9	OA	LCD segment signal output	
44	36	28	-	-	PB8*	IOUDS	PORT B Bit 8 Input/Output	
					MOSI0	I/O	SPI master output, slave input	
					SEG8	OA	LCD segment signal output	
45	-	-	-	-	PF7*	IOUDS	PORT F Bit 7 Input/Output	
					EC13	I	Timer 13 event count input	
					LPDE1	O	Low power UART DE signal output	
					SS3	I	SPI slave select input	
					SEG7	OA	LCD segment signal output	
46	-	-	-	-	PF6*	IOUDS	PORT F Bit 6 Input/Output	
					T13OUT	O	Timer 13 pulse output	
					T13CAP	I	Timer 13 capture input	
					SCK3	I/O	SPI clock input/output	
					SEG6	OA	LCD segment signal output	
47	-	-	-	-	PF5*	IOUDS	PORT F Bit 5 Input/Output	
					EC12	I	Timer 12 event count input	
					MISO3	I/O	SPI master input, slave output	
					SDA2	I/O	I2C data input/output	
					SEG5	OA	LCD segment signal output	

Table 2. Pin Description (continued)

Pin number					Pin name	Type	Description	Remark
LQFP-80	LQFP-64	LQFP-48	LQFP-32	QFN-32				
48	-	-	-	-	PF4*	IOUDS	PORT F Bit 4 Input/Output	
					T12OUT	O	Timer 12 pulse output	
					T12CAP	I	Timer 12 capture input	
					MOSI3	I/O	SPI master output, slave input	
					SCL2	I/O	I2C clock input/output	
					SEG4	OA	LCD segment signal output	
49	37	-	-	-	PB9*	IOUDS	PORT B Bit 9 Input/Output	
					T43OUTA	O	Timer 43 pulse output	
					T43INP	I	Timer 43 event count input	
					T10OUT	O	Timer 10 pulse output	
					T10CAP	I	Timer 10 capture input	
					SS3	I	SPI slave select input	
					SEG3	OA	LCD segment signal output	
50	38	-	-	-	PB10*	IOUDS	PORT B Bit 10 Input/Output	
					T43OUTB	O	Timer 43 pulse output	
					EC43	I	Timer 43 event count input	
					EC10	I	Timer 10 event count input	
					SCK3	I/O	SPI clock input/output	
					SEG2	OA	LCD segment signal output	
51	39	-	-	-	PB11*	IOUDS	PORT B Bit 11 Input/Output	
					T43INP	I	Timer 43 event count input	
					T11OUT	O	Timer 11 pulse output	
					T11CAP	I	Timer 11 capture input	
					MISO3	I/O	SPI master input, slave output	
					SEG1	OA	LCD segment signal output	
52	40	-	-	-	PB12*	IOUDS	PORT B Bit 12 Input/Output	
					T42INP	I	Timer 42 event count input	
					EC11	I	Timer 11 event count input	
					MOSI3	I/O	SPI master output, slave input	
					SDA2	I/O	I2C data input/output	
					SEG0	OA	LCD segment signal output	
53	41	29	-	-	PC0*	IOUDS	PORT C Bit 0 Input/Output	
					CLKO	O	System clock output	
					T12OUT	O	Timer 12 pulse output	
					T12CAP	I	Timer 12 capture input	
					SCL2	I/O	I2C clock input/output	
					SC0IN	I	Smartcard detection input	
					COM0	OA	LCD common signal output	

Table 2. Pin Description (continued)

Pin number					Pin name	Type	Description	Remark
LQFP-80	LQFP-64	LQFP-48	LQFP-32	QFN-32				
54	42	30	17	17	PC1*	IOUDS	PORT C Bit 1 Input/Output	
					CLKO	O	System clock output	
					EC12	I	Timer 12 event count input	
					TXD0	O	UART data output	
					SCL0	I/O	I2C clock input/output	
					SC0PWR	O	Smartcard power control output	
					COM1	OA	LCD common signal output	
55	43	31	18	18	PC2*	IOUDS	PORT C Bit 2 Input/Output	
					SCK1	I/O	SPI clock input/output	
					RXD0	I	UART data input	
					SDA0	I/O	I2C data input/output	
					SC0CLK	O	Smartcard clock output	
					COM2	OA	LCD common signal output	
					SEG11	OA	LCD segment signal output	
56	44	32	19	19	PC3*	IOUDS	PORT C Bit 3 Input/Output	
					MISO1	I/O	SPI master input, slave output	
					SC0RXD	I	SC0's UART data input	
					SC0RST	O	Smartcard reset output	
					CP0OUT	OA	Comparator 0 output	
					COM3	OA	LCD common signal output	
					SEG12	OA	LCD segment signal output	
57	45	33	20	20	PC4*	IOUDS	PORT C Bit 4 Input/Output	
					MOSI1	I/O	SPI master output, slave input	
					SC0TXD	O	SC0's UART data output	
					SC0DATA	I/O	Smartcard data input/output	
					CP1OUT	O	Comparator 1 output	
					SEG13	OA	LCD segment signal output	
58	46	34	21	21	PC5	IOUDS	PORT C Bit 5 Input/Output	
					LPRXD1	Input	Low power UART data input	
					SWDIO*	I/O	SWD data input/output	Pull-up when reset
59	47	35	22	22	PC6	IOUDS	PORT C Bit 6 Input/Output	
					LPTXD1	Output	Low power UART data output	
					SWCLK*	Input	SWD clock input	Pull-down when reset
60	48	36	-	-	PC7*	IOUDS	PORT C Bit 7 Input/Output	
					T40OUTA	O	Timer 40 pulse output	
					T40INP	I	Timer 40 event count input	
					SS1	I	SPI slave select input	
					SEG14	OA	LCD segment signal output	

Table 2. Pin Description (continued)

Pin number					Pin name	Type	Description	Remark
LQFP-80	LQFP-64	LQFP-48	LQFP-32	QFN-32				
61	49	37	23	23	PC8*	IOUDS	PORT C Bit 8 Input/Output	
					TXD10	O	UART data output	
					MOSI10	I/O	SPI master output, slave input	
					SC0TXD	O	SC0’s UART data output	
					SC0DATA	I/O	Smartcard data input/output	
					SEG15	OA	LCD segment signal output	
62	50	38	24	24	PC9*	IOUDS	PORT C Bit 9 Input/Output	
					RXD10	I	UART data input	
					MISO10	I/O	SPI master input, slave output	
					SC0RXD	I	SC0’s UART data input	
					SC0RST	O	Smartcard reset output	
					SEG16	OA	LCD segment signal output	
63	51	-	-	-	PC10*	IOUDS	PORT C Bit 10 Input/Output	
					SCK10	I/O	SPI clock input/output	
					SC0CLK	O	Smartcard clock output	
					LPTXD1	O	Low power UART data output	
					COM4	OA	LCD common signal output	
					SEG17	OA	LCD segment signal output	
64	52	-	-	-	PC11*	IOUDS	PORT C Bit 11 Input/Output	
					SS10	I	SPI slave select input	
					SC0PWR	O	Smartcard power control output	
					LPRXD1	I	Low power UART data input	
					COM5	OA	LCD common signal output	
					SEG18	OA	LCD segment signal output	
65	53	-	-	-	PC12*	IOUDS	PORT C Bit 12 Input/Output	
					TXD11	O	UART data output	
					MOSI11	I/O	SPI master output, slave input	
					SC0IN	I	Smartcard detection input	
					COM6	OA	LCD common signal output	
					SEG19	OA	LCD segment signal output	
66	54	-	-	-	PC13*	IOUDS	PORT C Bit 13 Input/Output	
					RXD11	I	UART data input	
					MISO11	I/O	SPI master input, slave output	
					LPDE1	O	Low power UART DE signal output	
					COM7	OA	LCD common signal output	
					SEG20	OA	LCD segment signal output	

Table 2. Pin Description (continued)

Pin number					Pin name	Type	Description	Remark
LQFP-80	LQFP-64	LQFP-48	LQFP-32	QFN-32				
67	-	-	-	-	PD8*	IOUDS	PORT D Bit 8 Input/Output	
					T42OUTA	O	Timer 42 pulse output	
					T42INP	I	Timer 42 event count input	
					SS0	I	SPI slave select input	
					SEG21	OA	LCD segment signal output	
68	-	-	-	-	PD9*	IOUDS	PORT D Bit 9 Input/Output	
					T42OUTB	O	Timer 42 pulse output	
					EC42	I	Timer 42 event count input	
					SCK0	I/O	SPI clock input/output	
					SEG22	OA	LCD segment signal output	
69	-	-	-	-	PD10*	IOUDS	PORT D Bit 10 Input/Output	
					T13OUT	O	Timer 13 pulse output	
					T13CAP	I	Timer 13 capture input	
					MISO0	I/O	SPI master input, slave output	
					SCK11	I/O	SPI clock input/output	
					SEG23	OA	LCD segment signal output	
70	-	-	-	-	PD11*	IOUDS	PORT D Bit 11 Input/Output	
					EC13	I	Timer 13 event count input	
					MOSI0	I/O	SPI master output, slave input	
					SS11	I	SPI slave select input	
					SEG24	OA	LCD segment signal output	
71	55	39	25	25	PD0*	IOUDS	PORT D Bit 0 Input/Output	5V tolerant I/O (The internal pull-up resistor must be disabled to use 5V I/O) VDD ≥ 2.0V and PD_TYP[1:0]=000b when CP1N3, CP1P2
					T40OUTB	O	Timer 40 pulse output	
					EC40	I	Timer 40 event count input	
					TXD11	O	UART data output	
					MOSI11	I/O	SPI master output, slave input	
					SCL2	I/O	I2C clock input/output	
					SCK1	I/O	SPI clock input/output	
					CP1N3	IA	Comparator 1 negative input	
72	56	40	26	26	PD1*	IOUDS	PORT D Bit 1 Input/Output	
					T43OUTA	O	Timer 43 pulse output	
					T43INP	I	Timer 43 event count input	
					RXD11	I	UART data input	
					MISO11	I/O	SPI master input, slave output	
					SDA2	I/O	I2C data input/output	
					MISO1	I/O	SPI master input, slave output	
					CP1P2	IA	Comparator 1 positive input	

Table 2. Pin Description (continued)

Pin number					Pin name	Type	Description	Remark
LQFP-80	LQFP-64	LQFP-48	LQFP-32	QFN-32				
73	57	41	27	27	PD2*	IOUDS	PORT D Bit 2 Input/Output	5V tolerant I/O (The internal pull-up resistor must be disabled to use 5V I/O) VDD ≥ 2.0V and PD_TYP[4:2]=00b when CP1P3, CP1P4, CP1P5
					T43OUTB	O	Timer 43 pulse output	
					EC43	I	Timer 43 event count input	
					EC50	I	Timer 50 event count input	
					SCK11	I/O	SPI clock input/output	
					MOSI1	I/O	SPI master output, slave input	
					CP1P3	IA	Comparator 1 positive input	
74	58	42	28	28	PD3*	IOUDS	PORT D Bit 3 Input/Output	
					SCL0	I/O	I2C clock input/output	
					TXD0	O	UART data output	
					SS11	I	SPI slave select input	
					CP1P4	IA	Comparator 1 positive input	
75	59	43	29	29	PD4*	IOUDS	PORT D Bit 4 Input/Output	
					T50INP	I	Timer 50 capture/clear input	
					SDA0	I/O	I2C data input/output	
					RXD0	I	UART data input	
					CP1P5	IA	Comparator 1 positive input	
76	60	44	30	30	PD5	IOUDS	PORT D Bit 5 Input/Output	
					BOOT*	I	Boot mode input	Pull-up when reset
					RTCOUT	O	Real time clock output	
77	61	45	-	31	PD6*	IOUDS	PORT D Bit 6 Input/Output	
					T13OUT	O	Timer 13 pulse output	
					T13CAP	I	Timer 13 capture input	
					SCL0	I/O	I2C clock input/output	
78	62	46	-	-	PD7*	IOUDS	PORT D Bit 7 Input/Output	
					EC13	I	Timer 13 event count input	
					SS0	I	SPI slave select input	
					SDA0	I/O	I2C data input/output	
79	63	47	31	-	VSS	P	Ground	
80	64	48	32	32	VDD	P	VDD	
-	-	-	-	33	VSS	P	Ground (Exposed pad)	

NOTES:

1. Notation: I=Input, O=Output, U=Pull-up, D=Pull-down, S=Schmitt-Trigger Input Type, C=CMOS Input Type, A=Analog, P=Power
2. (*) Selected pin function after reset condition
3. Pin order may be changed with revision notice.

3 Central Processing Unit (CPU)

The A31L21x series uses Cortex® –M0+ as its CPU and includes an interrupt controller named NVIC.

3.1 Cortex®–M0+ core

The Cortex-M0+ processor is the most energy-efficient ARM processor available. It builds on the very successful Cortex-M0+ processor, retaining full instruction set and tool compatibility, while further reducing energy consumption and increasing performance.

Please refer to the technical reference manual "ARM DDI 0484C" provided by ARM for detail information of Cortex-M0+.

3.2 Interrupt controller

The Cortex-M0+ processor has an embedded interrupt controller named Nested Vector Interrupt Controller (NVIC). The A31L21x series has an additional interrupt control block for controlling 32 interrupt sources generated by internal peripherals.

To use interrupts from internal peripherals, both the NVIC and the interrupt control block must be configured properly.

This document only describes the peripheral interrupt controller, therefore for more information on NVIC inside the Cortex-M0+ processor, please refer to the technical reference manual “ARM DDI 0484C” on the ARM technical document site.

Table 3. Interrupt Vector Map

Priority	Vector Address	Interrupt Source
-16	0x0000_0000	Stack Pointer
-15	0x0000_0004	Reset Address
-14	0x0000_0008	NMI Exception
-13	0x0000_000C	Hard Fault Exception
-12	0x0000_0010	Reserved
-11	0x0000_0014	
-10	0x0000_0018	
-9	0x0000_001C	
-8	0x0000_0020	
-7	0x0000_0024	
-6	0x0000_0028	
-5	0x0000_002C	SVCall Exception
-4	0x0000_0030	Reserved
-3	0x0000_0034	
-2	0x0000_0038	PenSV Exception
-1	0x0000_003C	SysTick Exception
0	0x0000_0040	LVI Interrupt
1	0x0000_0044	WUT Interrupt
2	0x0000_0048	WDT Interrupt
3	0x0000_004C	EINT0 Interrupt
4	0x0000_0050	EINT1 Interrupt
5	0x0000_0054	EINT2 Interrupt
6	0x0000_0058	EINT3 Interrupt
7	0x0000_005C	TIMER40 Interrupt

Table 3. Interrupt Vector Map (continued)

Priority	Vector Address	Interrupt Source
8	0x0000_0060	TIMER41 Interrupt
9	0x0000_0064	TIMER42 Interrupt
10	0x0000_0068	I2C0 Interrupt
11	0x0000_006C	USART1[1:0] Interrupt
12	0x0000_0070	SPI[1:0] Interrupt
13	0x0000_0074	SPI[3:2] Interrupt
14	0x0000_0078	I2C1 Interrupt
15	0x0000_007C	TIMER50 Interrupt
16	0x0000_0080	SC0 Interrupt
17	0x0000_0084	Deep Sleep Mode 1 Wakeup Interrupt
18	0x0000_0088	ADC Interrupt
19	0x0000_008C	UART0 Interrupt
20	0x0000_0090	Temperature Sensor Interrupt
21	0x0000_0094	TIMER43 Interrupt
22	0x0000_0098	CMP[1:0] Interrupt
23	0x0000_009C	DMACH[3:0] Interrupt
24	0x0000_00A0	DMACH[6:4] Interrupt
25	0x0000_00A4	LPUART[1:0] Interrupt
26	0x0000_00A8	TIMER1[1:0] Interrupt
27	0x0000_00AC	TIMER1[3:2] Interrupt
28	0x0000_00B0	RTCC Interrupt TIMER60 Interrupt
29	0x0000_00B4	I2C2 Interrupt
30	0x0000_00B8	AES-128 Interrupt
31	0x0000_00BC	Random Number Generator Interrupt

3.3 Registers

Base address and register map of the interrupt registers are shown in Table 4 and Table 5.

Table 4. Base Address of Interrupt Registers

Name	Base address
Interrupt register	0x4000_1000

Table 5. Interrupt Controller Register Map

Name	Offset	Type	Description	Reset Value
INTC_PnTRIG	0x0000-0x00FF	RW	Port n Interrupt Trigger Selection Register	0000_0000
INTC_PnCR	0x0100-0x01FF	RW	Port n Interrupt Control Register	0000_0000
INTC_PnFLAG	0x0200-0x02FF	RW	Port n Interrupt Flag Register	0000_0000
INTC_EINTxCONF1 INTC_EINTxCONF2	0x0300-0x03FF	RW	External Interrupt Configuration Register1,2	0000_0000
INTC_MSK	0x0400	RW	Interrupt Source Mask Register	0000_0000

NOTES:

1. n = A to F
2. x = 0 to 3

3.3.1 INTC_PnTRIG: port n interrupt trigger selection register

INTC_PnTRIG register is 32-bit size and accessible in 32/16/8-bit (n= A to F).

INTC_PATRIG =0x4000_1000, INTC_PBTRIG =0x4000_1004, INTC_PCTRIG =0x4000_1008
INTC_PDTRIG =0x4000_100C, INTC_PETRIG =0x4000_1010, INTC_PFTRIG =0x4000_1014

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																ITRIG15	ITRIG14	ITRIG13	ITRIG12	ITRIG11	ITRIG10	ITRIG9	ITRIG8	ITRIG7	ITRIG6	ITRIG5	ITRIG4	ITRIG3	ITRIG2	ITRIG1	ITRIG0
0x0000																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
–																RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

x	ITRIGx	Port n Interrupt Trigger Selection bit, x= 0 to 15	
		0	Edge trigger interrupt
		1	Level trigger interrupt

3.3.2 INTC_PnCR: port n interrupt control register

INTC_PnCR register is 32-bit size and accessible in 32/16/8-bit (n= A to F).

INTC_PACR=0x4000_1100, INTC_PBCR=0x4000_1104, INTC_PCCR=0x4000_1108
INTC_PDCR=0x4000_110C, INTC_PECR=0x4000_1110, INTC_PFCR=0x4000_1114

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INTCTL15		INTCTL14		INTCTL13		INTCTL12		INTCTL11		INTCTL10		INTCTL9		INTCTL8		INTCTL7		INTCTL6		INTCTL5		INTCTL4		INTCTL3		INTCTL2		INTCTL1		INTCTL0	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW	

2x+1 2x	INTCTLx	Port n Interrupt Control bits, x= 0 to 15	
		00	Disable external interrupt (The flag bit won’t be set)
		01	Interrupt on falling edge or on low level
		10	Interrupt on rising edge or on high level
		11	Interrupt on both falling and rising edge, No level interrupt

NOTE: Do not write “11” to the corresponding INTCTLx[1:0] bits when the ITRIGx bit of INTC_PnTRIG is ‘1’. If so, it may cause a malfunction.

3.3.3 INTC_PnFLAG: port n interrupt flag register

INTC_PnFLAG register is 32-bit size and accessible in 32/16/8-bit (n= A to F).

INTC_PAFLAG=0x4000_1200, INTC_PBFLAG=0x4000_1204, INTC_PCFLAG=0x4000_1208
INTC_PDFLAG=0x4000_120C, INTC_PEFLAG=0x4000_1210, INTC_PFFLAG=0x4000_1214

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																FLAG15	FLAG14	FLAG13	FLAG12	FLAG11	FLAG10	FLAG9	FLAG8	FLAG7	FLAG6	FLAG5	FLAG4	FLAG3	FLAG2	FLAG1	FLAG0
0x0000																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
–																RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

x	FLAGx	Port n Interrupt Flag bit, x: 0 to 15	
		0	No request occurred
		1	Request occurred. The bit is cleared to ‘0’ when ‘1’ is written.

3.3.4 INTC_EINTnCONF1: external interrupt n configuration register 1 (n= 0 to 3)

INTC_EINTnCONF1 register is 32-bit size and accessible in 32/16/8-bit.

INTC_EINT0CONF1=0x4000_1300, INTC_EINT1CONF1=0x4000_1304
INTC_EINT2CONF1=0x4000_1308, INTC_EINT3CONF1=0x4000_130C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CONF7				CONF6				CONF5				CONF4				CONF3				CONF2				CONF1				CONF0			
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW				RW				RW				RW				RW				RW				RW				RW			

4x+3 4x	CONFx	Configuration bits for External Interrupt Group n, x: 0 to 7	
		0000	PAx
		0001	PBx
		0010	PCx
		0011	PDx
		0100	PEx
		0101	PFx
		Others	Reserved

3.3.5 INTC_ EINTnCONF2: external interrupt n configuration register 2 (n= 0 to 3)

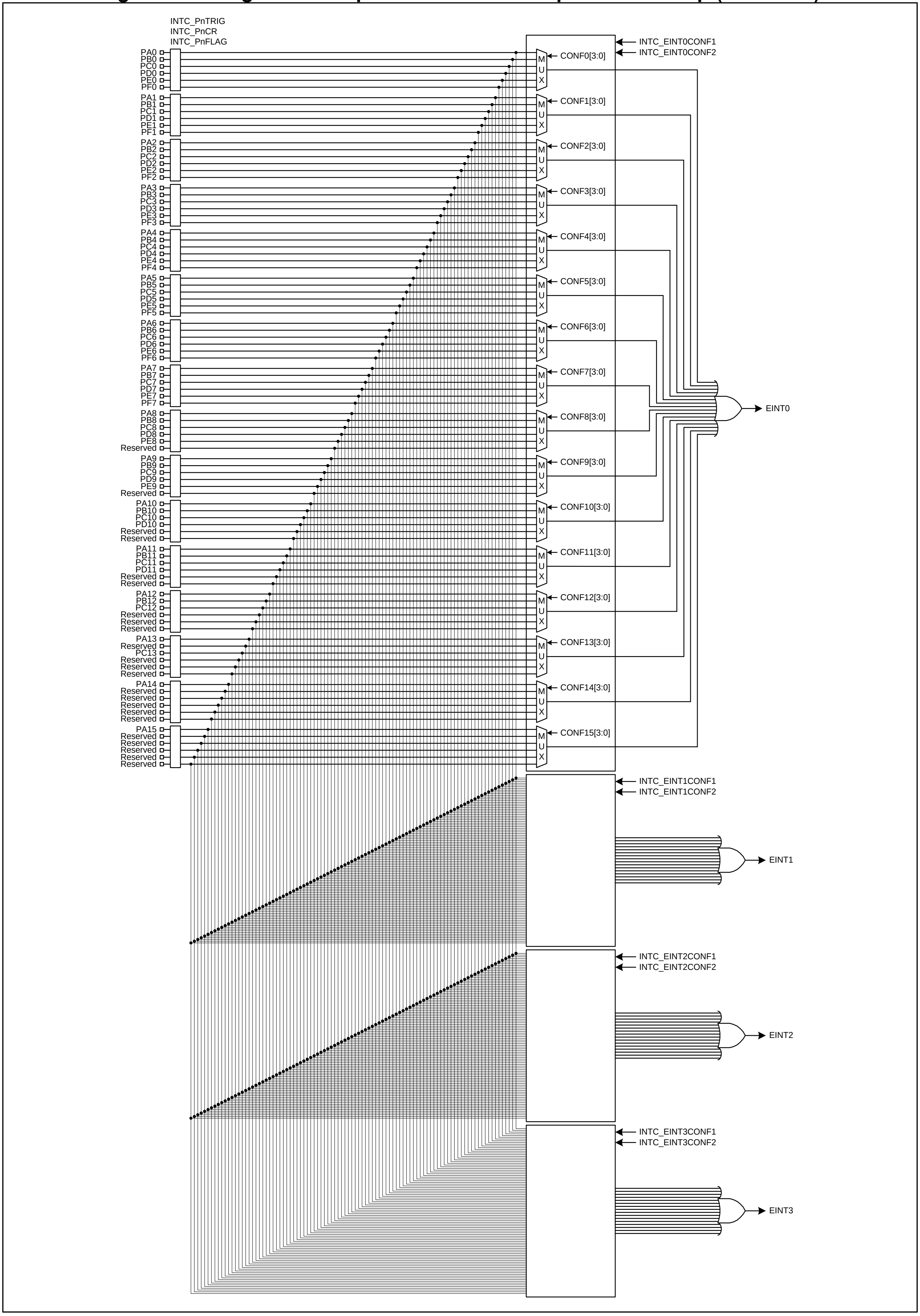
INTC_EINTnCONF2 register is 32-bit size and accessible in 32/16/8-bit.

INTC_EINT0CONF2=0x4000_1310, INTC_EINT1CONF2=0x4000_1314
INTC_EINT2CONF2=0x4000_1318, INTC_EINT3CONF2=0x4000_131C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CONF15				CONF14				CONF13				CONF12				CONF11				CONF10				CONF9				CONF8			
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW				RW				RW				RW				RW				RW				RW				RW			

4(x-8)+3 4(x-8)	CONFx	Configuration bits for External Interrupt Group n, x: 8 to 15	
		0000	PAx
		0001	PBx
		0010	PCx
		0011	PDx
		0100	PEx
		0101	PFx
		Others	Reserved

Figure 8. Configuration Map for External Interrupt 0/1/2/3 Group (n = A to F)



3.3.6 INTC_MSK: interrupt source mask register

INTC_MSK register is 32-bit size and accessible in 32/16/8-bit.

INTC_MSK=0x4000_1400

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IMSK31	IMSK30	IMSK29	IMSK28	IMSK27	IMSK26	IMSK25	IMSK24	IMSK23	IMSK22	IMSK21	IMSK20	IMSK19	IMSK18	IMSK17	IMSK16	IMSK15	IMSK14	IMSK13	IMSK12	IMSK11	IMSK10	IMSK9	IMSK8	IMSK7	IMSK6	IMSK5	IMSK4	IMSK3	IMSK2	IMSK1	IMSK0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

x	IMSKx	Interrupt Source Mask bit, x: 0 to 31	
		0	Mask. The corresponding interrupt is disabled.
		1	Unmask.

NOTES:

1. A mask interrupt source is not used as a wake-up source on “sleep”/”deep sleep” mode.
2. The corresponding interrupts of IMSKx are listed below:

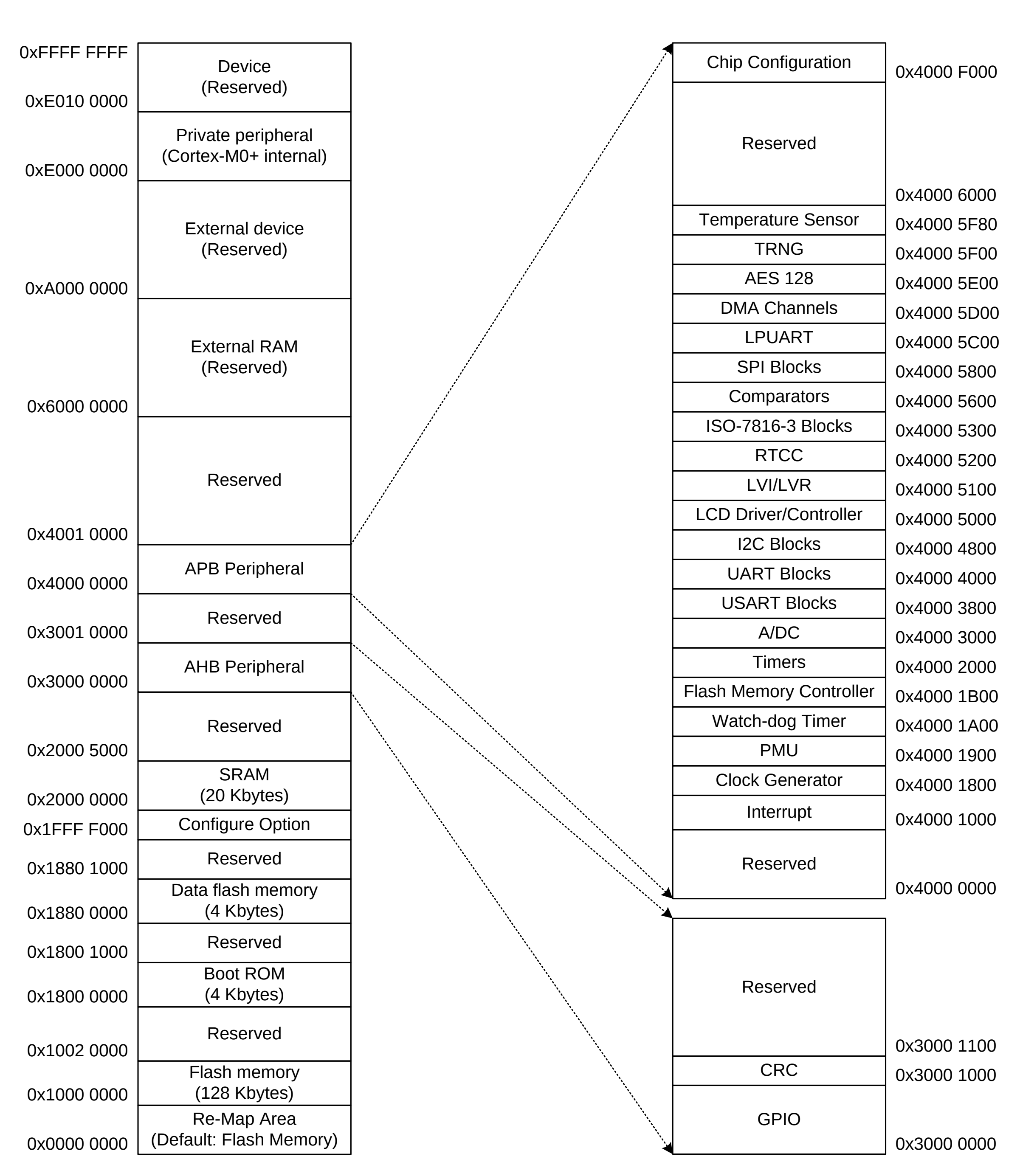
Table 6. Corresponding Interrupts of IMSKx

Source mask	Interrupt source name
IMSK0	LVI
IMSK1	WUT
IMSK2	WDT
IMSK3	EINT0
IMSK4	EINT1
IMSK5	EINT2
IMSK6	EINT3
IMSK7	TIMER40
IMSK8	TIMER41
IMSK9	TIMER42
IMSK10	I2C0
IMSK11	USART1[1:0]
IMSK12	SPI[1:0]
IMSK13	SPI[3:2]
IMSK14	I2C1
IMSK15	TIMER50
IMSK16	SC0
IMSK17	DS1WK
IMSK18	ADC
IMSK19	UART0
IMSK20	TS
IMSK21	TIMER43
IMSK22	CMP[1:0]
IMSK23	DMACH[3:0]
IMSK24	DMACH[6:4]
IMSK25	LPUART[1:0]
IMSK26	T1[1:0]
IMSK27	T1[3:2]
IMSK28	RTCC, T60
IMSK29	I2C2
IMSK30	AES-128
IMSK31	RNG

4 Control Memory Organization

Figure 9 shows addressable memory space in memory map.

Figure 9. Main Memory Map



NOTE: Access to the reserved area may cause a malfunction.

4.1 Internal SRAM

The A31L21x series has a block of 0-wait on-chip SRAM. Its size is 20KB, and its base address is 0x2000_0000. The SRAM's memory area is mainly used for data memory and stack memory. It is possible to locate code area in the SRAM memory for fast operation or for Flash erase or program operation for self-program.

This device does not support memory remapping. Therefore, the jump and return are required to process the code in SRAM memory area.

4.2 Boot mode

4.2.1 Boot mode pins

The A31L21x series has Boot mode to program the internal Flash memory. Boot mode is activated when the BOOT pin is set to "Low" level at reset timing. (For normal operation mode, the BOOT pin is set to "High" level.)

Boot mode supports the UART boot using the TXD10/RXD10 ports.

Table 7 introduces pins used in the Boot mode.

Table 7. Boot Mode Pin List

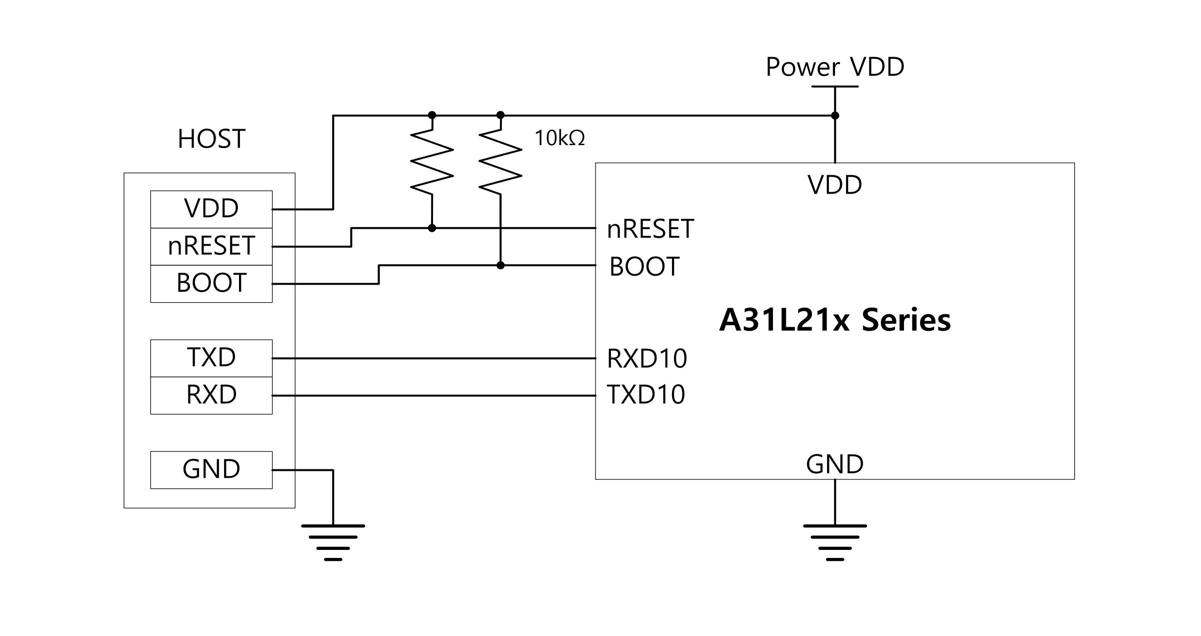
Block	Pin name	Direction	Description
SYSTEM	nRESET	I	Reset Input signal
	BOOT/PD5	I	'0' to enter Boot mode
UART mode of USART10	RXD10/PA3	I	UART Boot Receive Data
	TXD10/PA2	O	UART Boot Transmit Data

4.2.2 Boot mode connection

Users can design the target board using Boot mode ports – UART mode of USART10.

Figure 10 shows an example diagram of connections in Boot mode.

Figure 10. Connection Diagram of UART Boot



4.3 Flash memory

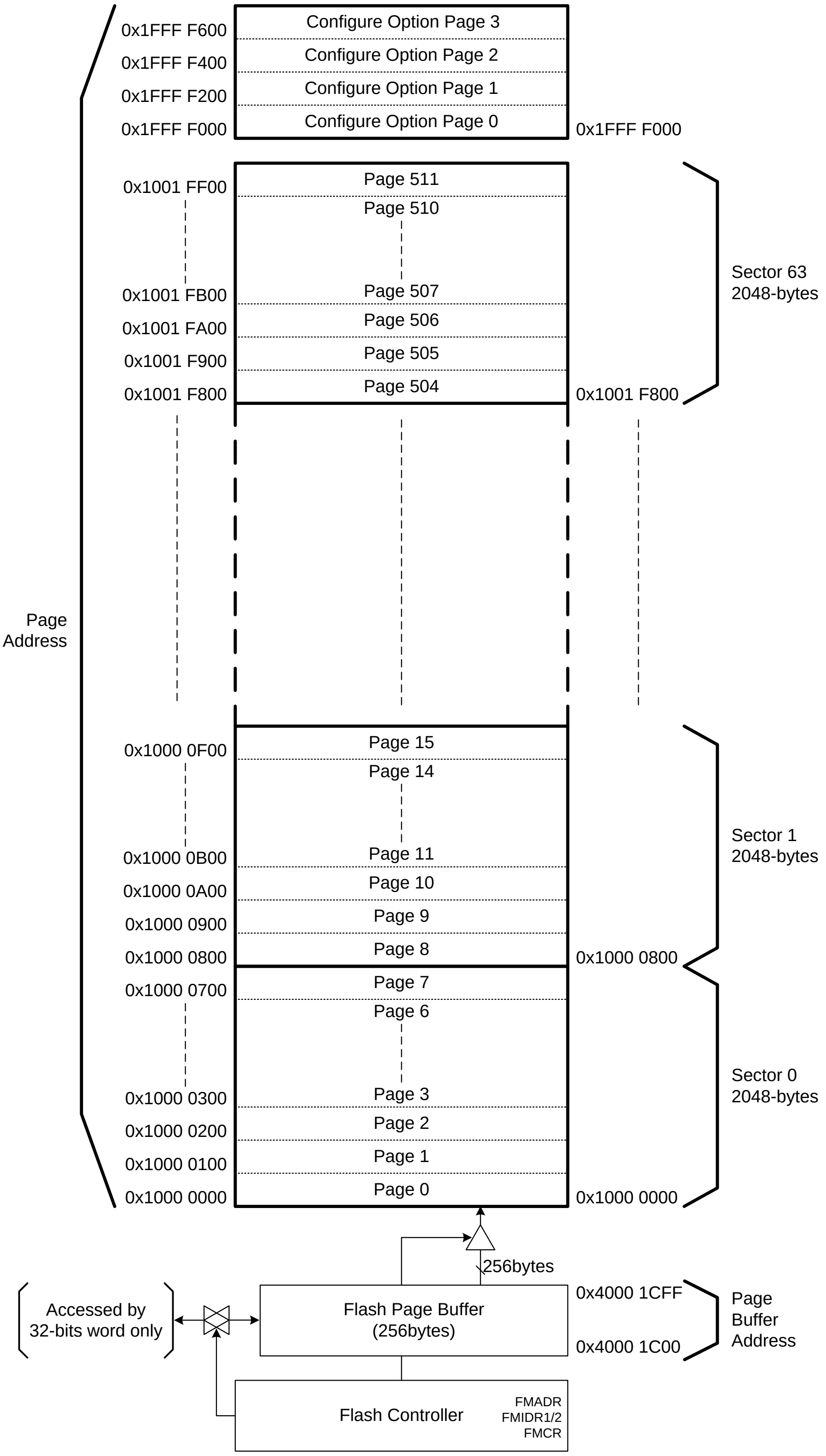
The A31L21x series has built-in Flash memory with the following features:

- 128 or 64KB Flash memory
- 32-bit read data bus width
- 256-byte page size
- Page erase and bulk erase available
- 256-byte unit program

Table 8. Internal Flash Memory Specification

Item	Description
Size	128KB
Start address	0x1000_0000
End address	0x1001_FFFF
Page size	256-byte
Total page count	512 pages
PGM unit	256-byte
Erase unit	256-byte or bulk

Figure 11. Internal Flash Memory Block Diagram



NOTES:

1. If an instruction of a size other than the 32-bit size is used to write data to the buffer, incorrect data is written.
2. Each of Configure Option Pages (page0/1/2/3) is 256-byte size, respectively.

4.3.1 Registers

Base address and register map of the Flash memory controller are shown in Table 9 and Table 10.

Table 9. Base Address of Flash Memory Controller

Name	Base address
Flash memory controller	0x4000_1B00

Table 10. Flash Memory Controller Register Map

Name	Offset	Type	Description	Reset Value
FMC_ADR	0x0000	RW	Flash Memory Address Register	0x5FFFFF80
FMC_IDR1	0x0004	RW	Flash Memory Identification Register 1	0x00000000
FMC_IDR2	0x0008	RW	Flash Memory Identification Register 2	0x00000000
FMC_CR	0x000C	RW	Flash Memory Control Register	0x00000000
FMC_BCR	0x0010	RW	Flash Memory Configure Area Bulk Erase Control Register	0x00000000
FMC_ERFLAG	0x0014	RW	Flash Memory Error Flag	0x00000000
FMC_PAGEBUF	0x0100-0x01FF	WO	Flash Memory Page Buffer Area	0x00000000

4.3.1.1 FMC_ADR: Flash memory address register

FMC_ADR register is used to remember the internal Flash memory address. This register is 32-bit size.

FMC_ADR=0x4000_1B00

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="32">ADDR</td></tr>
<tr><td colspan="32">0x5FFF_FF80
RW</td></tr>
</table>

31 0	ADDR	Flash Memory Address Pointer. This register is reset to 0x5FFFFF80 immediately after a single operation.

NOTE: The LSB-byte of the target Flash address is always considered to “0x00”.

4.3.1.2 FMC_IDR1: Flash memory identification register 1

FMC_IDR1 register is an internal Flash memory identification register for Flash mode. This register is 32-bit size.

FMC_IDR1=0x4000_1B04

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="32">ID1</td></tr>
<tr><td colspan="32">0x0000_0000
RW</td></tr>
</table>

31 0	ID1	Flash Memory Identification 1	
		0x08192A3B	Identification value for a Flash mode
		Others	No identification value

4.3.1.3 FMC_IDR2: Flash memory identification register 2

FMC_IDR2 register is an internal Flash memory identification register for Flash mode. This register is 32-bit size.

FMC_IDR2=0x4000_1B08

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="32">ID2</td></tr>
<tr><td colspan="32">0x0000_0000
RW</td></tr>
</table>

31 0	ID2	Flash Memory Identification 2	
		0x4C5D6E7F	Identification value for a Flash mode
		Others	No identification value

NOTES:

1. The FMC_IDR1/2 registers are automatically cleared to logic 0x00000000 immediately after one time operation except “Flash page buffer reset mode”.
2. The FMC_IDR1/2 registers should be written with correct values in turn.
3. If incorrect values are written to the FMC_IDR1/2 registers, the registers are cleared to logic 0x00000000.

4.3.1.4 FMC_CR: Flash memory control register

FMC_CR register is an internal Flash memory control register. This register is 32-bit size.

FMC_CR=0x4000_1B0C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WTIDKY																**FMKEY**								**FMBUSY**	**Reserved**			**FMOD**			
0x0000																**0x00**								**0**	**000**			**0000**			
WO																**RW**								**RO**	**-**			**RW**			

31 16	WTIDKY	Write Identification Key.	
		When writing, write 0x6C93 to these bits, or else writing is ignored.	
15 8	FMKEY	Flash Memory Operation Area Selection.	
		0x00	Selects no area but for Flash page buffer reset mode.
		0x38	Selects “configure option area” for Flash memory erase/write.
		0xA4	Selects “Flash memory area” for Flash memory erase/write.
		Others	Not allowed. FMOPFLAG will be set.
7	FMBUSY	Flash Memory Operation Mode Busy.	
		0	No effect.
		1	Busy.
3 0	FMOD	Flash Memory Operation Mode Selection.	
		0001	“Flash page buffer reset mode” and start regardless of the Flash operation rule. (Clear all 256bytes page buffer to 0xFFFFFFFF)
		0010	“Flash page erase mode” and start when the Flash operation rule is satisfied.
		0100	“Flash page write mode” and start when the Flash operation rule is satisfied.
		1000	“Flash bulk erase mode” and start when the Flash operation rule is satisfied.
		Others	Not allowed. FMOPFLAG will be set.

NOTES

1. During a Flash memory operation mode, all interrupts are on disable regardless of enable bits.
2. The FMKEY[7:0] and FMOD[3:0] bits are automatically cleared to logic “0x00” immediately after a single operation.
3. In Read Protection level 1/2, the Data Flash memory must be erased before performing Flash bulk erase.

4.3.1.5 FMC_BCR: Flash memory configure area bulk erase control register

FMC_BCR register is used to permit bulk erase. This register is 32-bit size.

FMC_BCR=0x4000_1B10

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0
WTIDKY	**Reserved**	**CNF3BEN**	**CNF2BEN**	**CNF1BEN**
0x0000	**0000**	**0000**	**0000**	**0000**
WO	**–**	**RW**	**RW**	**RW**

31 16	WTIDKY	Write Identification Key.	
		When writing, write 0xC1BE to these bits, or else writing is ignored.	
11 8	CNF3BEN	Configure Option Page 3 Bulk Erase Enable.	
		0x5	Permit “Configure Option Page 3” erase at bulk erase
		Others	Protect “Configure Option Page 3” erase at bulk erase
7 4	CNF2BEN	Configure Option Page 2 Bulk Erase Enable.	
		0x5	Permit “Configure Option Page 2” erase at bulk erase
		Others	Protect “Configure Option Page 2” erase at bulk erase
3 0	CNF1BEN	Configure Option Page 1 Bulk Erase Enable.	
		0x5	Permit “Configure Option Page 1” erase at bulk erase
		Others	Protect “Configure Option Page 1” erase at bulk erase

NOTE: This register is automatically cleared to logic “0x00” immediately after one time operation.

4.3.1.6 FMC_ERFLAG: Flash memory error flag register

FMC_ERFLAG is 32-bit size, and accessible in 32/16/8-bit.

FMC_ERFLAG=0x4000_1B14

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8	7 6 5 4 3 2	1	0
Reserved		**INSTFLAG**	**FMOPFLAG**
0x000000	**000000**	**0**	**0**
–		**RW**	**RW**

1	INSTFLAG	Don’t care	
0	FMOPFLAG	Error bit of Flash Memory Operation Procedure. This bit is set to logic 1 if there is a wrong procedure for Flash memory operation.	
		0	No wrong procedure.
		1	A wrong procedure occurred. The bit is cleared to ‘0’ when ‘1’ is written.

4.3.2 Procedure for Flash memory operation

- The high frequency internal RC oscillator (HIRC) should be enabled by S/W for Flash memory operation.
- The procedure will be cleared, the related registers will be reset, and FMOPFLAG will be set if wrong sequence is detected.
- The address range is 0x10000000 to 0x17FFFFFF when “Flash memory area” is selected.
- The address range is 0x1FFFF000 to 0x1FFFFFFF when “configure option area” is selected.
- If the CPU is in the Flash memory, the CPU will halt while the Flash memory is programmed.
- The “Configure Option Page 0” won’t be erased at Flash bulk erase mode.
- The “Configure Option Page 1/2/3” can be erased at Flash bulk erase mode if the CNFxBEN has correct values
- The CPU should not be in the Flash memory area on Flash bulk erase mode.
- A write to the Flash related register is ignored during Flash operation.
- An NMI source should not be selected during Flash memory operation is activated.
- The LVR should be enabled during Flash memory operation is activated (Recommended: 2.17V over).
- The global interrupt should be disabled.
- The CPU should not enter SLEEP and DEEP SLEEP mode during Flash erase/write mode.
- It should not be entered the Flash erase/write modes during Data Flash program. So, before the Flash erase/write operation, the Data Flash Busy bit (DFMOPFLAG) should be checked whether the bit is “1b” or not. If the bit is “1b”, do not start Flash program.

4.3.2.1 Page Erase procedure

1. Write 0x5FFFFFFF to FMC_ADR when the register is equal to 0x5FFFFF80.
2. Write 0x08192A3B to FMC_IDR1 register when the FMC_ADR register is equal to 0x5FFFFFFF.
3. Write 0x4C5D6E7F to FMC_IDR2 register when the FMC_ADR register is equal to 0x5FFFFFFF.
4. Write 0x6C930001 to FMC_CR register for page buffer reset when the FMC_ADR register is equal to 0x5FFFFFFF.
5. Clear page buffer (128bytes) by writing 0xFFFFFFFF repeatedly during the FMC_ADR register is 0x5FFFFFFF.
6. Write a page address to FMC_ADR register.
7. Read and check the FMC_IDR1 and FMC_IDR2 registers in turn.
8. Write 0x6C93A402 (Flash memory area) or 0x6C933802 (configure option area) to FMC_CR register.
9. Check whether the FMBUSY bit is ‘0’ or not.
10. Verify the erased page of Flash memory.

4.3.2.2 Byte/Page Write procedure

1. Write 0x5FFFFFFF to FMC_ADR when the register is equal to 0x5FFFFF80.
2. Write 0x08192A3B to FMC_IDR1 register when the FMC_ADR register is equal to 0x5FFFFFFF.
3. Write 0x4C5D6E7F to FMC_IDR2 register when the FMC_ADR register is equal to 0x5FFFFFFF.
4. Write 0x6C930001 to FMC_CR register for page buffer reset when the FMC_ADR register is equal to 0x5FFFFFFF.
5. Write data to page buffer (any bytes) when the FMC_ADR register is equal to 0x5FFFFFFF.
6. Write a page address to FMC_ADR register.
7. Read and check the FMC_IDR1 and FMC_IDR2 registers in turn.
8. Write 0x6C93A404 (Flash memory area) or 0x6C933804 (configure option area) to FMC_CR register.
9. Check whether the FMBUSY bit is ‘0’ or not.
10. Verify the written page of Flash memory.

4.3.2.3 Flash Bulk Erase procedure

1. Write 0x5FFFFFFF to FMC_ADR when the register is equal to 0x5FFFFF80.
2. Write 0x08192A3B to FMC_IDR1 register when the FMC_ADR register is equal to 0x5FFFFFFF.
3. Write 0x4C5D6E7F to FMC_IDR2 register when the FMC_ADR register is equal to 0x5FFFFFFF.
4. Write 0x6C930001 to FMC_CR register for page buffer reset when the FMC_ADR register is equal to 0x5FFFFFFF.
5. Write the value 0x5F9A30D7 to FMC_ADR register.
6. Read and check the FMC_IDR1 and FMC_IDR2 register in turn.
7. Write 0x6C93A408 to FMC_CR register.
8. Check whether the FMBUSY bit is ‘0’ or not.
9. Verify all the pages of Flash memory.

4.3.2.4 Flash Bulk Erase procedure including configure option area

1. Write 0x5FFFFFFF to FMC_ADR when the register is equal to 0x5FFFFF80.
2. Write 0x08192A3B to FMC_IDR1 register when the FMC_ADR register is equal to 0x5FFFFFFF.
3. Write 0x4C5D6E7F to FMC_IDR2 register when the FMC_ADR register is equal to 0x5FFFFFFF.
4. Write 0x6C930001 to FMC_CR register for page buffer reset when the FMC_ADR register is equal to 0x5FFFFFFF.
5. Write the value 0xC1BE0VVV to FMC_BCR register. If V==5, the corresponding option page will be erased.
6. Write the value 0x5F9A30D7 to FMC_ADR register.
7. Read and check the FMC_IDR1 and FMC_IDR2 register in turn.
8. Write 0x6C93A408 to FMC_CR register.
9. Check whether the FMBUSY bit is ‘0’ or not.
10. Verify all the pages of Flash memory.

4.4 Data Flash memory

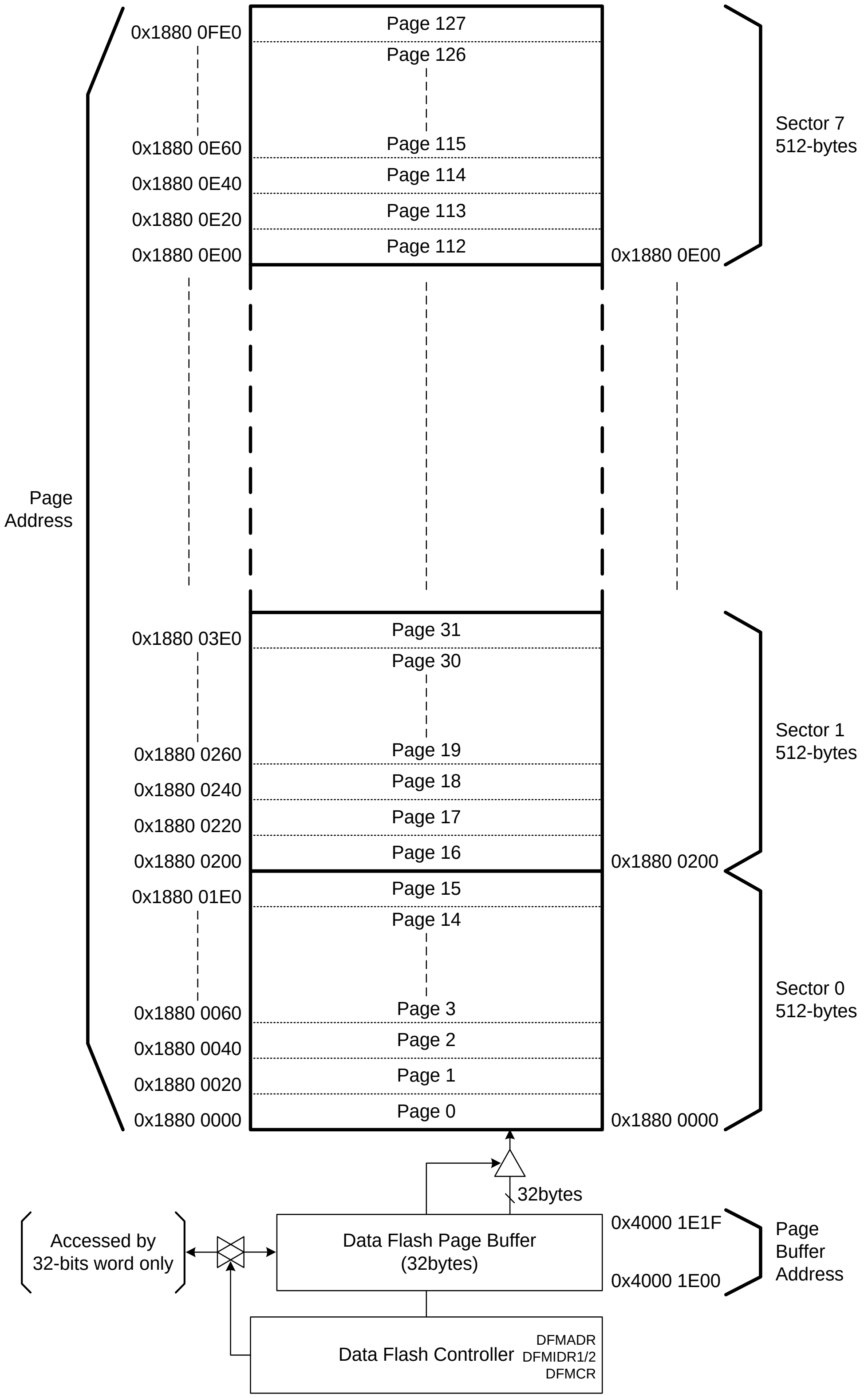
The A31L21x series has an internal Data Flash memory featuring the followings:

- 4KB Data Flash memory
- 32-bit read data bus width
- 32-byte page size
- Page erase and bulk erase available
- 32-byte unit program

Table 11. Data Flash Memory Specification

Item	Description
Size	4KB
Start address	0x1880_0000
End address	0x1880_0FFF
Page size	32-byte
Total page count	128 pages
PGM unit	32-byte
Erase unit	32-byte or bulk

Figure 12. Data Flash Memory Block Diagram



NOTE: If an instruction of a size other than the 32-bit size is used to write data to the buffer, incorrect data is written.

4.4.1 Registers

Base address and register map of the Data Flash memory controller are shown in Table 12 and Table 13.

Table 12. Base Address of Data Flash Memory Controller

Name	Base address
Data Flash memory controller	0x4000_1D00

Table 13. Flash Memory Controller Register Map

Name	Offset	Type	Description	Reset value
DFMC_DADR	0x0000	RW	Data Flash Memory Address Register	0x5FFF9990
DFMC_DIDR1	0x0004	RW	Data Flash Memory Identification Register 1	0x00000000
DFMC_DIDR2	0x0008	RW	Data Flash Memory Identification Register 2	0x00000000
DFMC_DCR	0x000C	RW	Data Flash Memory Control Register	0x00000000
DFMC_DERFLAG	0x0014	RW	Data Flash Memory Error Flag	0x00000000
DFMC_DPAGEBUF	0x0100- 0x011F	WO	Data Flash Memory Page Buffer Area	0x00000000

4.4.1.1 DFMC_DADR: Data Flash memory address register

DFMC_DADR register is used to remember the internal Data Flash memory address. This register is 32-bit size.

DFMC_DADR=0x4000_1D00

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDR																															
0x5FFF_9990																															
RW																															

31 0	ADDR	Data Flash Memory Address Pointer. This register is reset to 0x5FFF9990 immediately after a single operation.

NOTE: The LSB-5bits of the target Flash address is always considered to “00000”.

4.4.1.2 DFMC_DIDR1: Data Flash memory identification register 1

DFMC_DIDR1 register is an internal Data Flash memory identification register for Data Flash mode. This register is 32-bit size.

DFMC_DIDR1=0x4000_1D04

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID1																															
0x0000_0000 RW																															

31	ID1	Data Flash Memory Identification 1	
0		0xA9C46E91	Identification value for a Data Flash mode
		Others	No identification value

4.4.1.3 DFMC_DIDR2: Data Flash memory identification register 2

DFMC_DIDR2 register is an internal Data Flash memory identification register for Data Flash mode. This register is 32-bit size.

DFMC_DIDR2=0x4000_1D08

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID2																															
0x0000_0000 RW																															

31	ID2	Data Flash Memory Identification 2	
0		0x78F5B3D2	Identification value for a Data Flash mode
		Others	No identification value

NOTES:

1. The DFMC_DIDR1/2 registers are automatically cleared to logic 0x00000000 immediately after one time operation except “Data Flash page buffer reset mode”.
2. The DFMC_DIDR1/2 registers should be written with correct values in turn.
3. If incorrect values are written to the DFMC_DIDR1/2 registers, the registers are cleared to logic 0x00000000.

4.4.1.4 DFMC_DCR: Data Flash memory control register

DFMC_DCR register is an internal Data Flash memory control register. This register is 32-bit size.

DFMC_DCR=0x4000_1D0C

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="16">WTIDKY</td><td colspan="8">DFMKEY</td><td>DFMBUSY</td><td colspan="3">Reserved</td><td colspan="4">DFMOD</td></tr>
<tr><td colspan="16">0x0000</td><td colspan="8">0x00</td><td>0</td><td colspan="3">000</td><td colspan="4">0000</td></tr>
<tr><td colspan="16">WO</td><td colspan="8">RW</td><td>RO</td><td colspan="3">-</td><td colspan="4">RW</td></tr>
</table>

31 16	WTIDKY	Write Identification Key.	
		When writing, write 0xB69C to these bits, or else writing is ignored.	
15 8	DFMKEY	Data Flash Memory Operation Area Selection	
		0x00	Selects no area but for Data Flash page buffer reset mode.
		0xA4	Selects “Data Flash memory area” for Data Flash memory erase/write.
		Others	Not allowed. DFMOPFLAG will be set.
7	DFMBUSY	Data Flash Memory Operation Mode Busy	
		0	No effect.
		1	Busy.
3 0	DFMOD	Data Flash Memory Operation Mode Selection	
		0001	“Data Flash page buffer reset mode” and start regardless of operation rule. (Clear all 32bytes page buffer to 0xFF)
		0010	“Data Flash page erase mode” and start when operation rule is satisfied.
		0100	“Data Flash page write mode” and start when operation rule is satisfied.
		1000	“Data Flash bulk erase mode” and start when operation rule is satisfied.
		Others	Not allowed. DFMOPFLAG will be set.

NOTE: The DFMKEY[7:0] and DFMOD[3:0] bits are automatically cleared to logic “0x00” immediately after a single operation

4.4.1.5 DFMC_DERFLAG: Data Flash memory error flag register

DFMC_DERFLAG is 32-bit size, and accessible in 32/16/8-bit.

DFMC_DERFLAG=0x4000_1D14

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="31">Reserved</td><td>DFMOPFLAG</td></tr>
<tr><td colspan="24">0x000000</td><td colspan="7">0000000</td><td>0</td></tr>
<tr><td colspan="31">-</td><td>RW</td></tr>
</table>

0	DFMOPFLAG		Error bit of Data Flash memory operation procedure. This bit is set to logic 1 if there is a wrong procedure for Data Flash memory operation.
		0	No wrong procedure.
		1	A wrong procedure occurred. The bit is cleared to ‘0’ when ‘1’ is written.

4.4.2 Procedure for Data Flash memory operation

- The high frequency internal RC oscillator (HIRC) should be enabled by S/W for Data Flash memory operation.
- The procedure will be cleared, the related registers will be reset, and DFMOPFLAG will be set if wrong sequence is detected.
- The address range is 0x18800000 to 0x188FFFFF when “Data Flash memory area” is selected.
- A write to the Data Flash related register is ignored during Data Flash operation.
- The LVR should be enabled during Data Flash memory operation is activated (Recommended: 2.17V over).
- The CPU should not enter SLEEP and DEEP SLEEP mode during Data Flash erase/write mode.
- It should not be entered the Data Flash erase/write modes during Flash program. So, before the Data Flash erase/write operation, the Flash busy bit (FMOPFLAG) should be checked whether the bit is “1b” or not. If the bit is “1b”, do not start Data Flash program.

4.4.2.1 Page Erase procedure

1. Write 0x59999990 to DFMC_DADR when the register is equal to 0x5FFF9990.
2. Write 0xA9C46E91 to DFMC_DIDR1 register when the DFMC_DADR register is equal to 0x59999990.
3. Write 0x78F5B3D2 to DFMC_DIDR2 register when the DFMC_DADR register is equal to 0x59999990.
4. Write 0xB69C0001 to DFMC_DCR register for page buffer reset when the DFMC_DADR register is equal to 0x59999990.
5. Clear page buffer (32bytes) by writing 0xFFFFFFFF repeatedly during the DFMC_DADR register is 0x59999990.
6. Write a page address to DFMC_DADR register.
7. Read and check the DFMC_DIDR1 and DFMC_DIDR2 registers in turn.
8. Write 0xB69CA402 (Data Flash memory area) to DFMC_DCR register.
9. Check whether the DFMBUSY bit is ‘0’ or not.
10. Verify the erased page of Data Flash memory.

4.4.2.2 Page Write procedure

1. Write 0x59999990 to DFMC_DADR when the register is equal to 0x5FFF9990.
2. Write 0xA9C46E91 to DFMC_DIDR1 register when the DFMC_DADR register is equal to 0x59999990.
3. Write 0x78F5B3D2 to DFMC_DIDR2 register when the DFMC_DADR register is equal to 0x59999990.
4. Write 0xB69C0001 to DFMC_DCR register for page buffer reset when the DFMC_DADR register is equal to 0x59999990.
5. Write data to Data Flash page buffer (any bytes) when the DFMC_DADR register is equal to 0x59999990.
6. Write a page address to DFMC_DADR register.
7. Read and check the DFMC_DIDR1 and DFMC_DIDR2 registers in turn.
8. Write 0xB69CA404 (Data Flash memory area) to DFMC_DCR register.
9. Check whether the DFMBUSY bit is ‘0’ or not.
10. Verify the written page of Data Flash memory.

4.4.2.3 Data Flash Bulk Erase procedure

1. Write 0x59999990 to DFMC_DADR when the register is equal to 0x5FFF9990.
2. Write 0xA9C46E91 to DFMC_DIDR1 register when the DFMC_DADR register is equal to 0x59999990.
3. Write 0x78F5B3D2 to DFMC_DIDR2 register when the DFMC_DADR register is equal to 0x59999990.
4. Write 0xB69C0001 to DFMC_DCR register for page buffer reset when the DFMC_DADR register is equal to 0x59999990.
5. Write the value 0x4BC27F54 to DFMC_DADR register.
6. Read and check the DFMC_DIDR1 and DFMC_DIDR2 register in turn.
7. Write 0xB69CA408 to DFMC_DCR register.
8. Check whether the DFMBUSY bit is ‘0’ or not.
9. Verify all the pages of Flash memory.

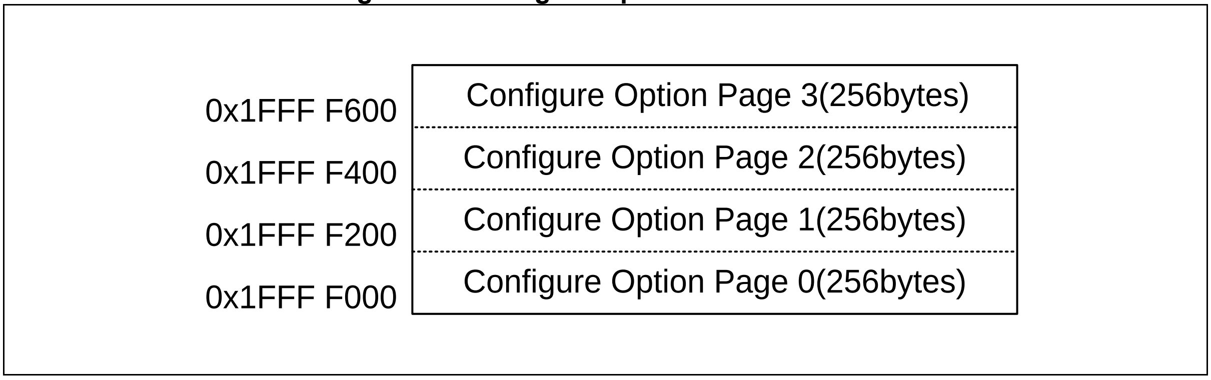
4.5 Configure option area

Configuration option area of the A31L21x series is used for system related trimming values, user option, and user data. The configure option area consists of four pages in the Flash memory, which can be erased and written by the Flash memory controller. This area can be read by any instruction.

The four pages of the configuration option area are listed in the followings:

- Page 0: System related trimming values and 128-bit unique device ID registers
- Page 1: User option for Read Protection, watchdog timer, and LVR voltage level configurations
- Page 2: User data 0 area
- Page 3: User data 1 area

Figure 13. Configure Option Area Structure


0x1FFF F600
Configure Option Page 3(256bytes)
0x1FFF F400
Configure Option Page 2(256bytes)
0x1FFF F200
Configure Option Page 1(256bytes)
0x1FFF F000
Configure Option Page 0(256bytes)

4.5.1 Configure option page

Base address of the configuration option area ranges from 0x1FFF_F000 to 0x1FFF_F600. The area map is shown in Table 14.

Table 14. Configuration Option Area Map

Page	Name	Address	Description
0	-	0x1FFF_F000 to 0x1FFF_F047 0x1FFF_F060 to 0x1FFF_F07F	System Trimming Values
	TS_FREQ_T30	0x1FFF_F048	Temperature Sensor Output Frequency acquired at 30℃ [Hz]
	TS_FREQ_T85	0x1FFF_F04C	Temperature Sensor Output Frequency acquired at 85℃ [Hz]
	TS_FREQ_T105	0x1FFF_F06C	Temperature Sensor Output Frequency acquired at 105℃ [Hz]
	CONF_MF1CNFIG	0x1FFF_F050	Manufacture Information 1 for 128-bit unique ID
	CONF_MF2CNFIG	0x1FFF_F054	Manufacture Information 2 for 128-bit unique ID
	CONF_MF3CNFIG	0x1FFF_F058	Manufacture Information 3 for 128-bit unique ID
	CONF_MF4CNFIG	0x1FFF_F05C	Manufacture Information 4 for 128-bit unique ID
1	CONF_RPCNFIG	0x1FFF_F200	Configuration for Read Protection
	CONF_WDTCNFIG	0x1FFF_F20C	Configuration for Watch-Dog Timer
	CONF_LVRCNFIG	0x1FFF_F210	Configuration for Low Voltage Reset
	CONF_CNFIGWTP1	0x1FFF_F214	Erase/Write Protection for Configure Option Page 1/2/3
	CONF_FMWTP1	0x1FFF_F240	Erase/Write Protection 1 for Flash Memory
	CONF_FMWTP2	0x1FFF_F244	Erase/Write Protection 2 for Flash Memory
	CONF_DFMWTP1	0x1FFF_F270	Erase/Write Protection 1 for Data Flash Memory
2	**-**	0x1FFF_F400 to 0x1FFF_F4FF	User Data Area 0
3	**-**	0x1FFF_F600 to 0x1FFF_F6FF	User Data Area 1

4.5.1.1 TS_FREQ_T30: Temperature Sensor output frequency 1

The Temperature Sensor Output Frequency 1 is 32-bit Flash memory. This is accessible in 32/16/8-bit.

TS_FREQ_T30=0x1FFF_F048

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								**F_T30**																							

23 0	F_T30	Temperature Sensor Output Frequency acquired at 30℃ [Hz].

4.5.1.2 TS_FREQ_T85: Temperature Sensor output frequency 2

The Temperature Sensor Output Frequency 2 is 32-bit Flash memory. This is accessible in 32/16/8-bit.

TS_FREQ_T85=0x1FFF_F04C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								**F_T85**																							

23 0	F_T85	Temperature Sensor Output Frequency acquired at 85℃ [Hz]. (Commercial grade)

4.5.1.3 TS_FREQ_T105: Temperature Sensor output frequency 3

The Temperature Sensor Output Frequency 3 is 32-bit Flash memory. This is accessible in 32/16/8-bit.

TS_FREQ_T105=0x1FFF_F06C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								**F_T105**																							

23 0	F_T105	Temperature Sensor Output Frequency acquired at 105℃ [Hz]. (Industrial grade)

4.5.1.4 CONF_MF1CNFIG: configuration for manufacture information 1

The Configuration for Manufacture Information 1 is 32-bit Flash memory. This is accessible in 32/16/8-bit.

CONF_MF1CNFIG=0x1FFF_F050

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
XYCDN																															

31 0	XYCDN	X and Y Coordinates.

4.5.1.5 CONF_MF2CNFIG: configuration for manufacture information 2

The Configuration for Manufacture Information 2 is 32-bit Flash memory. This is accessible in 32/16/8-bit.

CONF_MF2CNFIG=0x1FFF_F054

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LOTNO[23:0]																								**WAFNO**							

31 8	LOTNO[23:0]	Lot Number.
7 0	WAFNO	Wafer Number.

4.5.1.6 CONF_MF3CNFIG: configuration for manufacture information 3

The Configuration for Manufacture Information 3 is 32-bit Flash memory. This is accessible in 32/16/8-bit.

CONF_MF3CNFIG=0x1FFF_F058

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LOTNO[55:24]																															

31 0	LOTNO[55:24]	Lot Number.

4.5.1.7 CONF_MF4CNFIG: configuration for manufacture information 4

The Configuration for Manufacture Information 4 is 32-bit Flash memory. This is accessible in 32/16/8-bit.

CONF_MF4CNFIG=0x1FFF_F05C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LOTNO[87:56]																															

31 0	LOTNO[87:56]	Lot Number.

4.5.1.8 CONF_RPCNFIG: configuration for Read Protection

The configuration for the Flash Memory Read Protection is 32-bit. This is accessible in 32/16/8-bit.

CONF_RPCNFIG=0x1FFF_F200

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WTIDKY																												**Reserved**		**READP**	

31 4	WTIDKY	Write Identification Key These bits are the write key for “Read Protection”. So, The WTIDKY[27:0] should be kept with the 0x69C8A27. Otherwise, the Read Protection will be on level 2.	
1 0	READP	Read Protection for Flash Memory and Data Flash Memory Area.	
		11	Read Protection level 0, No restriction for read/erase/write.
		10	Read Protection level 1, Not readable/erasable/writable by “Debug” Bulk erasable only by ”Debug” Readable/erasable/writable by “Instruction from Flash Memory and RAM”
		0x	Read Protection level 2, Where x is don’t care Not readable/erasable/writable by “Debug”/”Instruction from RAM” Bulk erasable only by “Instruction from RAM”/”Debug” Readable/erasable/writable by “Instruction from Flash Memory”

NOTES:

1. The Read Protection level can be changed from lower level to higher level only.
2. The “Configure Option Page 1” cannot be erased by “Debug” unit on “Read Protection level 1/2” and by “Instruction from RAM” on “Read Protection level 2.
3. The configure option area may be read even if the “Read Protection” is on level 1 and 2.
4. A page unit erase/write except a bulk erase isn’t executable by “Instruction from RAM” regardless of the CONF_FMWTP1/2 and CONF_DFMWTP1 registers on Read Protection level 2.
5. A page unit erase/write except a bulk erase isn’t executable by “Debug” regardless of the CONF_FMWTP1/2 and CONF_DFMWTP1 registers on Read Protection level 1/2.
6. The Read Protection level will be ‘0’ on operation after bulk erase.

4.5.1.9 CONF_WDTCNFIG: configuration for Watchdog Timer

The configuration for watchdog timer is 32-bit Flash memory. This is accessible in 32/16/8-bit.

CONF_WDTCNFIG=0x1FFF_F20C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																**WRCMF**												**Reserved**	**WCLKMF**	**WRSTMF**	**WCNTMF**

15 4	WRCMF	Watchdog Timer RC Oscillator Master Configuration	
		0x96D	The WDTRC oscillation is decided by the WDTRCEN of SCU_CLKSRCR register.
		0x2A7	Master enable WDTRC but disabled at DEEP SLEEP mode regardless the PMU_PWRCR.ALLPWR bits.
		Others	Master enable WDTRC but power off at shutdown mode.
		NOTE: If the WDTRC is selected for MCLK by SCU_SCCR register when the bits are not 0x96D, the CPU cannot wake up at DEEP SLEEP mode. So, only SLEEP mode on the above case should be used for power down.	
2	WCLKMF	Watchdog Timer Clock Selection Master Configuration	
		0	Watchdog Timer clock is selected by the WDTCLK of SCU_PPCLKSR register.
		1	Master selection WDTRC for Watchdog Timer clock
1	WRSTMF	Watchdog Timer Reset Enable Master Configuration	
		0	Master enable WDT reset
		1	Disable/Enable of WDT reset is decided by the RSTEN[5:0] of WDT_CR register.
0	WCNTMF	Watchdog Timer Counter Enable Master Configuration	
		0	Master enable WDT counter
		1	Disable/Enable WDT counter is decided by the CNTEN[5:0] of WDT_CR register.

4.5.1.10 CONF_LVRCNFIG: configuration for Low Voltage Reset

The configuration for low voltage reset is 32-bit Flash memory. This is accessible in 32/16/8-bit.

CONF_LVRCNFIG=0x1FFF_F210

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3	2 1 0
WTIDKY		**LVRENM**	**Reserved**	**LVRVS**

31 16	WTIDKY	Write Identification Key : 0x9D58	
		These bits are the write key for “LVR controller”.	
15 8	LVRENM	LVR Reset Operation Control Master Configuration	
		0xAA	LVR operation is decided by the LVREN of SCU_LVRCR register
		Others	Master enable LVR operation
2 0	LVRVS	LVR Voltage Selection.	
		111	1.50V
		110	1.87V
		101	2.02V
		100	2.17V
		011	2.32V
		010	2.47V
		001	2.64V
		000	2.78V

4.5.1.11 CONF_CNFIGWTP1: Erase/Write Protection for Configure Option Page 1/2/3

The Erase/Write Protection for Configure Option Page is 32-bit Flash memory. This is accessible in 32/16/8-bit.

CONF_CNFIGWTP1=0x1FFF_F214

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4	3	2	1	0
Reserved				**Reserved**	**CP3WP**	**CP2WP**	**CP1WP**

2	CP3WP	Configure Option Page 3 Erase/	
		0	Enable protection (Not erasable/writable by instruction)
		1	Disable protection (Erasable/writable by instruction)
1	CP2WP	Configure Option Page 2 Erase/Write Protection	
		0	Enable protection (Not erasable/writable by instruction)
		1	Disable protection (Erasable/writable by instruction)
0	CP1WP	Configure Option Page 1 Erase/Write Protection	
		0	Enable protection (Not erasable/writable by instruction)
		1	Disable protection (Erasable/writable by instruction)

NOTE: The Configure Option Page which is protected cannot be erased by page unit.

4.5.1.12 CONF_FMWTP1 Erase/Write Protection 1 for Flash memory

The Erase/Write Protection 1 for Flash memory is 32-bit Flash memory. This is accessible in 32/16/8-bit.

CONF_FMWTP1=0x1FFF_F240

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SWTP31	SWTP30	SWTP29	SWTP28	SWTP27	SWTP26	SWTP25	SWTP24	SWTP23	SWTP22	SWTP21	SWTP20	SWTP19	SWTP18	SWTP17	SWTP16	SWTP15	SWTP14	SWTP13	SWTP12	SWTP11	SWTP10	SWTP9	SWTP8	SWTP7	SWTP6	SWTP5	SWTP4	SWTP3	SWTP2	SWTP1	SWTP0

n	SWTPn	Flash Memory Erase/Write Protection bits, n: 0 to 31 (Sector 0 to Sector 31)	
		0	Protect “Flash memory sector n erase/write”
		1	Permit “Flash memory sector n erase/write”

4.5.1.13 CONF_FMWTP2 Erase/Write Protection 2 for Flash memory

The Erase/Write Protection 2 for Flash memory is 32-bit Flash memory. This is accessible in 32/16/8-bit.

CONF_FMWTP2=0x1FFF_F244

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SWTP63	SWTP62	SWTP61	SWTP60	SWTP59	SWTP58	SWTP57	SWTP56	SWTP55	SWTP54	SWTP53	SWTP52	SWTP51	SWTP50	SWTP49	SWTP48	SWTP47	SWTP46	SWTP45	SWTP44	SWTP43	SWTP42	SWTP41	SWTP40	SWTP39	SWTP38	SWTP37	SWTP36	SWTP35	SWTP34	SWTP33	SWTP32

n-32	SWTPn	Flash Memory Erase/Write Protection bits, n: 32 to 63 (Sector 32 to Sector 63)	
		0	Protect “Flash memory sector n erase/write”
		1	Permit “Flash memory sector n erase/write”

4.5.1.14 CONF_DFMWTP1 Erase/Write Protection 1 for Data Flash memory

The Erase/Write Protection 1 for Data Flash memory is 32-bit Flash memory. This is accessible in 32/16/8-bit.

CONF_DFMWTP1=0x1FFF_F270

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																								SWTP7	SWTP6	SWTP5	SWTP4	SWTP3	SWTP2	SWTP1	SWTP0

n	DSWTPn	Data Flash Memory Erase/Write Protection bits, n: 0 to 7 (Sector 0 to Sector 7)	
		0	Protect “Data Flash memory sector n erase/write”
		1	Permit “Data Flash memory sector n erase/write”

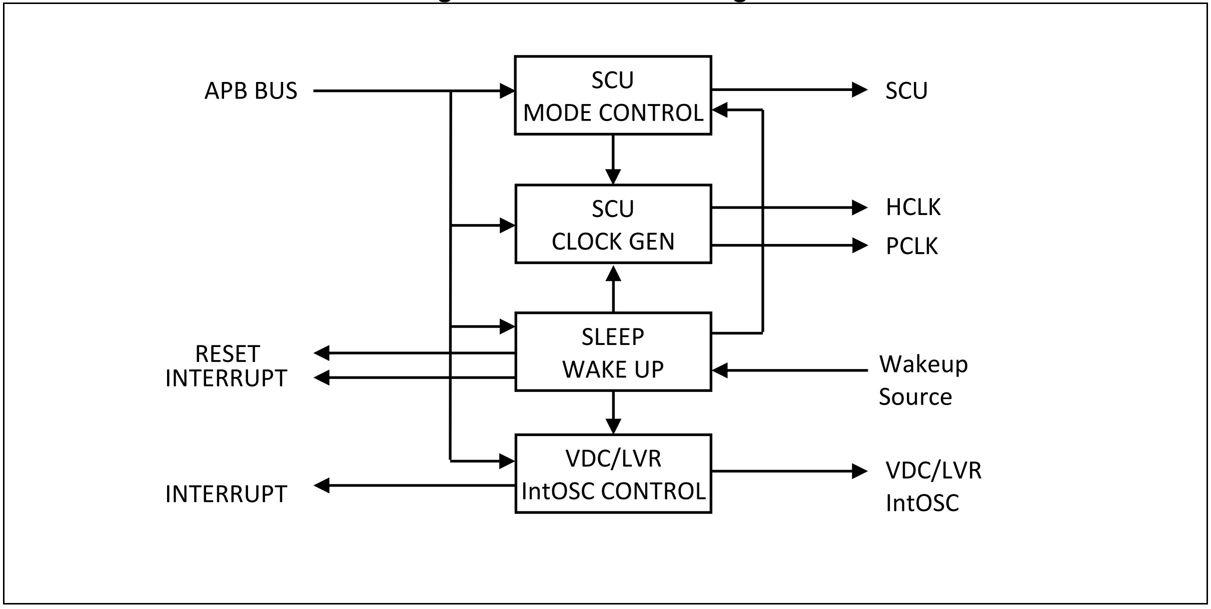
5 System Control Unit (SCU)

The A31L21x series has a built-in intelligent power control block, which manages analog blocks and operating modes. Internal reset and clock signals are controlled by SCU block to maintain optimized system performance and power dissipation.

5.1 SCU block diagram

Figure 14 shows the SCU block diagram.

Figure 14. SCU Block Diagram

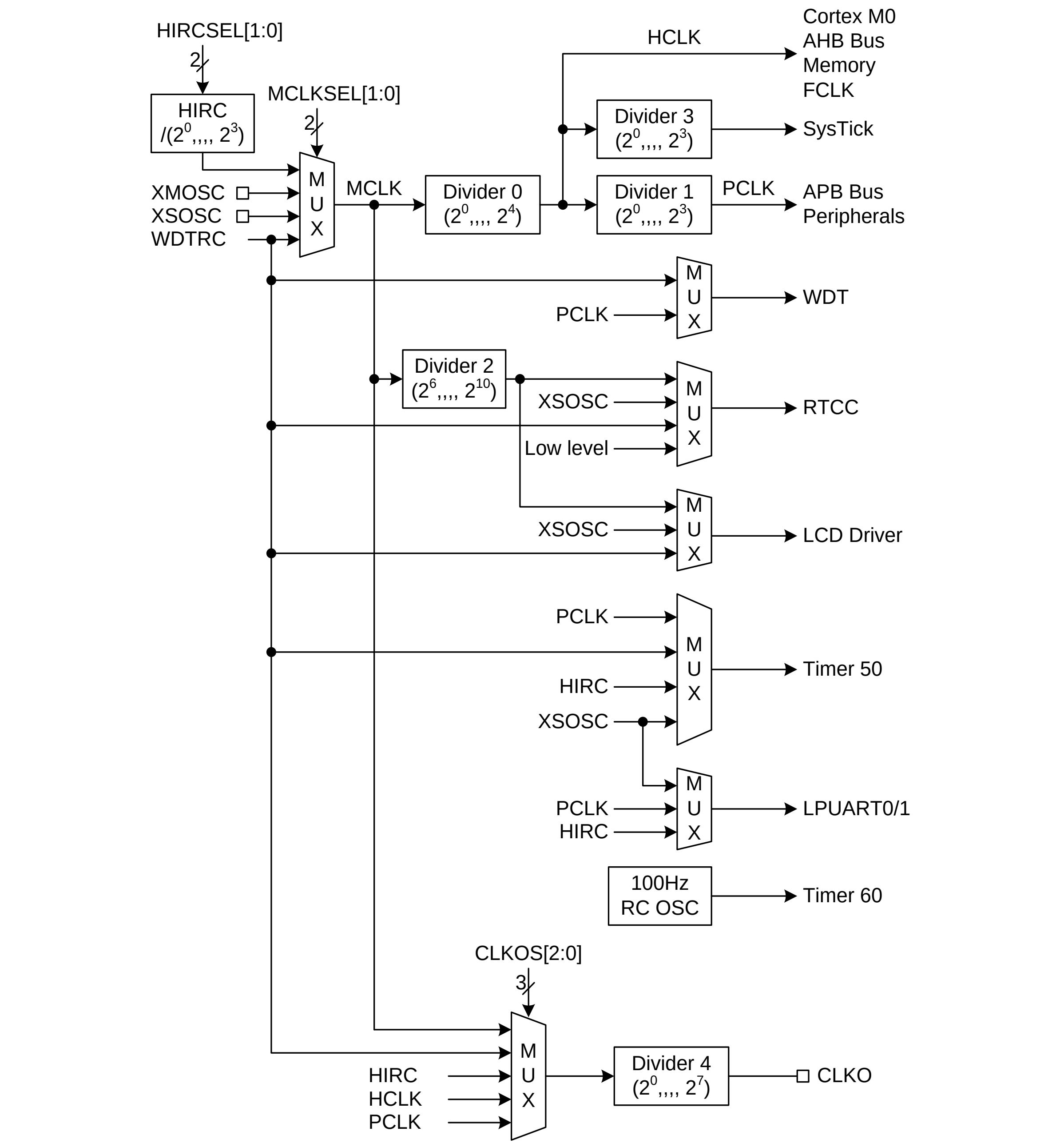


5.2 Clock system

The A31L21x series has two main operating clocks. One is HCLK, which supplies the clock to the CPU and AHB bus system. The other one is PCLK, which supplies the clock to the peripheral systems.

Users can control the clock system variation by software. Figure 15 shows the clock system of the A31L21x series and Table 15 shows the descriptions for clock sources.

Figure 15. Clock Source Configuration



NOTE: If the WRCMF[11:0] bits of configuration are not equal to 0x96D, the WDTRC oscillation can't be disabled by the WDTRCEN bit of SCU_CLKSRCR register.

Each mux to switch clock source has a glitch-free circuit. So a clock can be switched without glitch risks. When you change the clock mux control, be sure both clock sources are alive. If either is not alive, clock change operation stops and system will shut down and not recover.

Table 15. Clock Sources

Clock name	Mnemonic	Frequency	Description
Main OSC	XMOSC	• X-TAL (2MHz to 16MHz) • External Clock (2MHz to 32MHz)	• External Main Crystal OSC • External Main Clock
Sub OSC	XSOSC	X-TAL (32.768kHz)	External Sub Crystal OSC
Internal RC OSC	HIRC	2MHz to 32MHz	High Frequency Internal RC OSC
WDT RC OSC	WDTRC	40kHz	Watchdog Timer RC OSC

5.2.1 HCLK clock domain

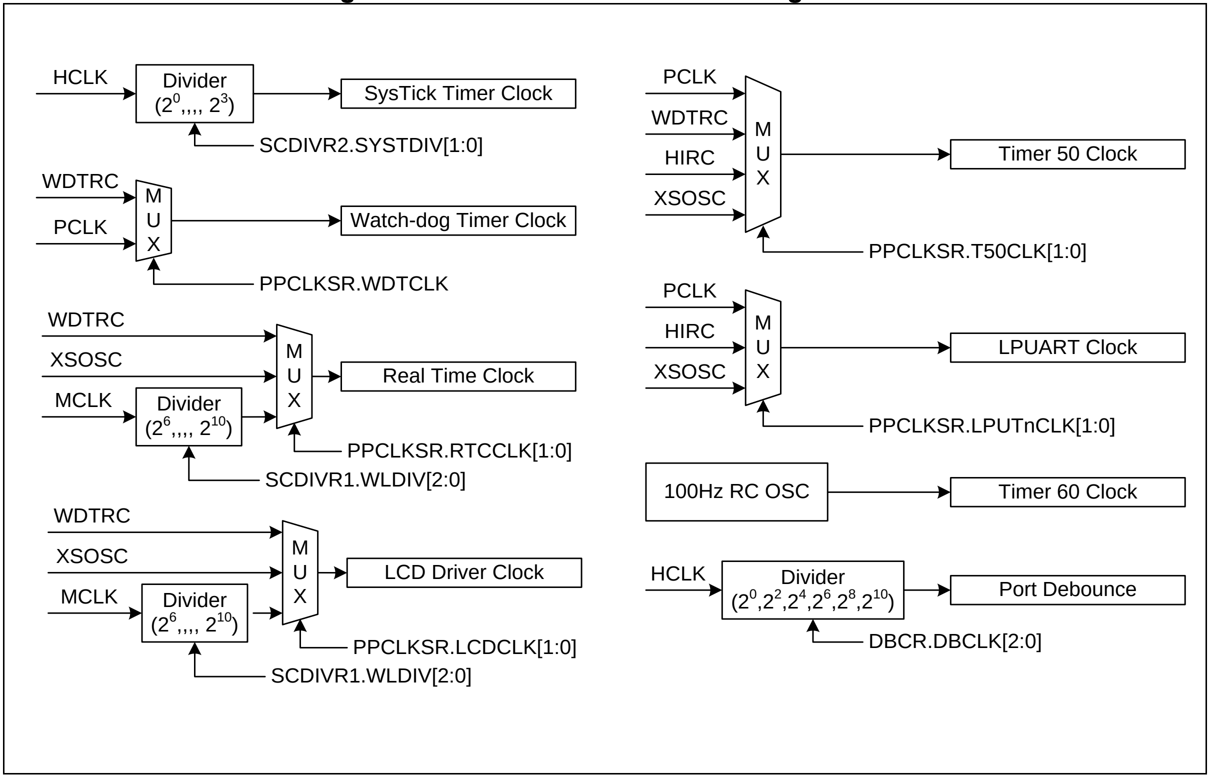
HCLK clock feeds the clock to the CPU and AHB bus. Cortex-M0+ CPU requires 2 clocks, FCLK and HCLK. FCLK is a free running clock and is always running except during power down mode. HCLK can be stopped during SLEEP mode.

The HCLK clock operates the BUS system and memory systems. Max BUS operating clock speed is 32MHz. HCLK frequency should be limited to a frequency of 32MHz or lower.

5.2.2 Miscellaneous clock domain

Various clock sources are required for each functional block. The SCU provides clock source selectivity with dedicated pre-scaler for each functional block. The clock selection mux does not support glitch-free function, so the clock is unpredictable during clock selection. Figure 16 shows the configurations for miscellaneous clocks.

Figure 16. Miscellaneous Clock Configuration



5.2.3 PCLK clock domain

PCLK is the master clock for all the peripherals except for the CRC generator and ports. It can shut down during power down mode. Each peripheral clock is generated by SCU_PPCLKEN1 and SCU_PPCLKEN2 register set. Figure 15 illustrates the PCLK clock distributions. The peripherals are not accessible even by reading its registers until each PCLK clock of each block is enabled.

5.2.4 Clock configuration procedure

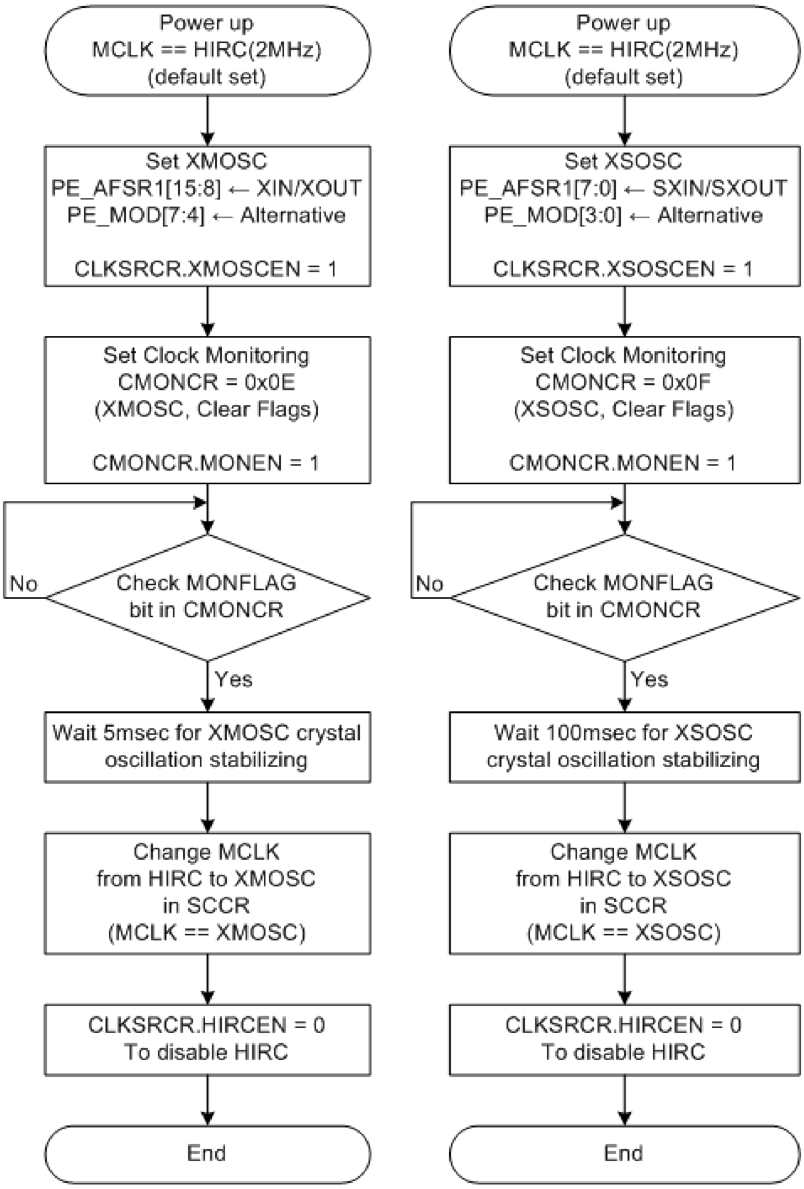
After power on the device, a default system clock is generated by HIRC (2MHz) clock. The HIRC is enabled by default during power up sequence. Other clock sources are enabled by user controls and configuration options with a system clock.

XMOSC and XSOSC clocks are enabled by XMOSCEN and XSOSCEN bits of SCU_CLKSRCR register respectively. Before enabling XMOSC and XSOSC blocks, the pin mux configuration should be set for XIN/XOUT and SXIN/SXOUT functions. PE2/PE3 and PE0/PE1 pins are shared by XMOSC's XIN/XOUT function and XSOSC's SXIN/SXOUT function – PE_MOD and PE_AFSR1 registers should be configured properly.

After enabling the XMOSC and XSOSC blocks, a user can check stability of crystal oscillation through a clock monitoring control register, SCU_CMONCR. It takes more than 1ms to ensure stable crystal oscillation before changing the system clock.

Figure 17 shows an example flow chart to configure the system clock to XMOSC and XSOSC clock.

Figure 17. Clock Configuration Procedure



5.3 Reset

The A31L21x series has two system resets. One is the cold reset by POR, which is effective during power up or down sequence. The other is the warm reset, generated by several reset sources. The reset event makes the device to turn back to its initial state.

The cold reset has only one reset source, which is POR, while the warm reset has several reset sources as shown below:

- nRESET pin
- WDT reset
- LVR reset
- MON reset
- S/W reset
- CPU request reset
- WAKUP3 reset

5.3.1 Cold reset

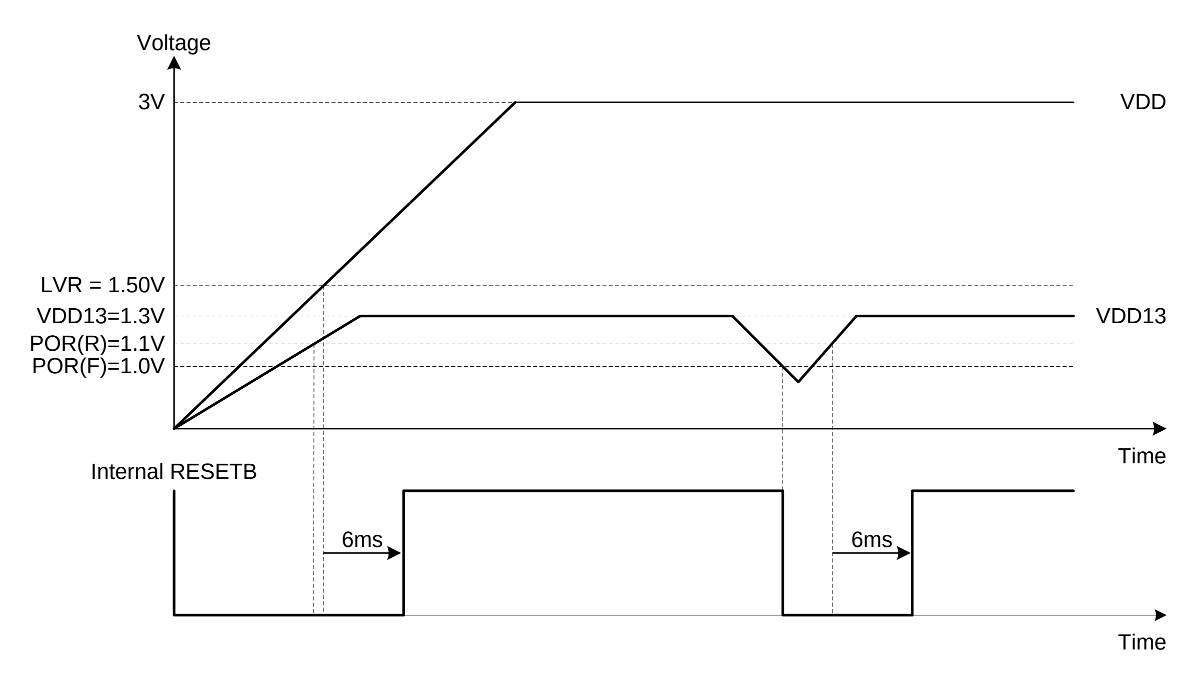
The cold reset is one of important feature of the A31L21x series when it powers up. This characteristic will globally affect the system boot.

Internal VDC is enabled when VDD power is turned on. Internal VDD level slope follows the External VDD power slope. Internal POR trigger level is at 1.1V of the internal VDC voltage. At this time, boot operation begins.

Internal RC clock turns on and counts 6ms for internal VDC level to stabilize. At this time, external VDD voltage level should be bigger than initial LVR level (1.50V). After 6ms of counting, the CPU reset is released and operation begins.

Figure 18 shows waveform of power up sequence and internal reset.

Figure 18. Power-up POR Sequence



A register SCU_RSTSSR shows the POR reset status. The last reset comes from the POR. SCU_RSTSSR.PORSTA is set to '1'. After power on, this bit is always '1' if the bit is not cleared by S/W. If abnormal internal voltage drop is detected during normal operation, the system will be reset and this bit also will be set to '1'.

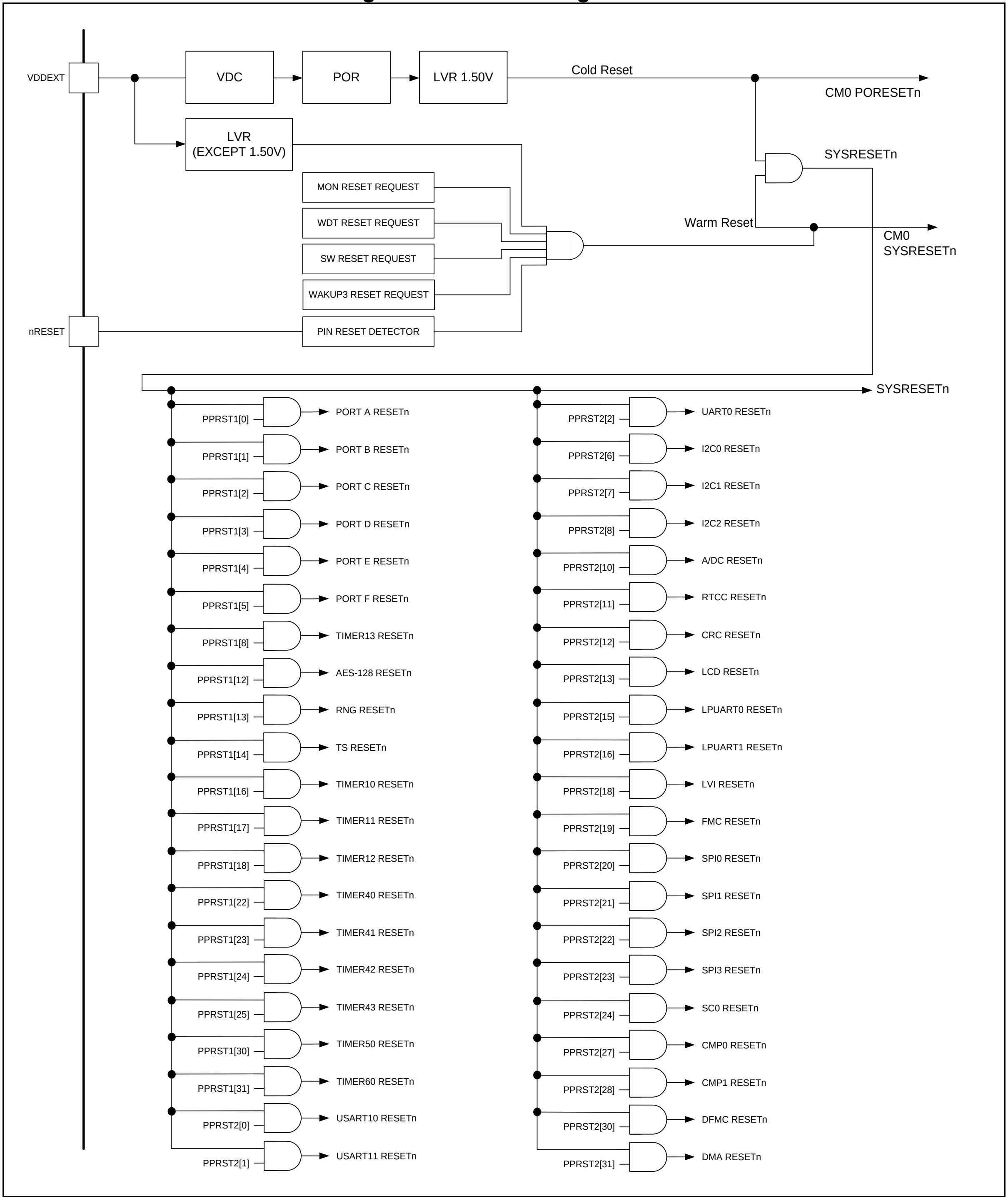
When the cold reset is applied, the entire device returns to its initial state.

5.3.2 Warm reset

The warm reset event has several reset sources and some parts of the device return to their initial states when the warm reset takes place.

The warm reset status appears in a register SCU_RSTSSR. A reset for each peripheral block is controlled by a register SCU_PPRST. The reset can be masked independently.

Figure 19. Reset Configuration

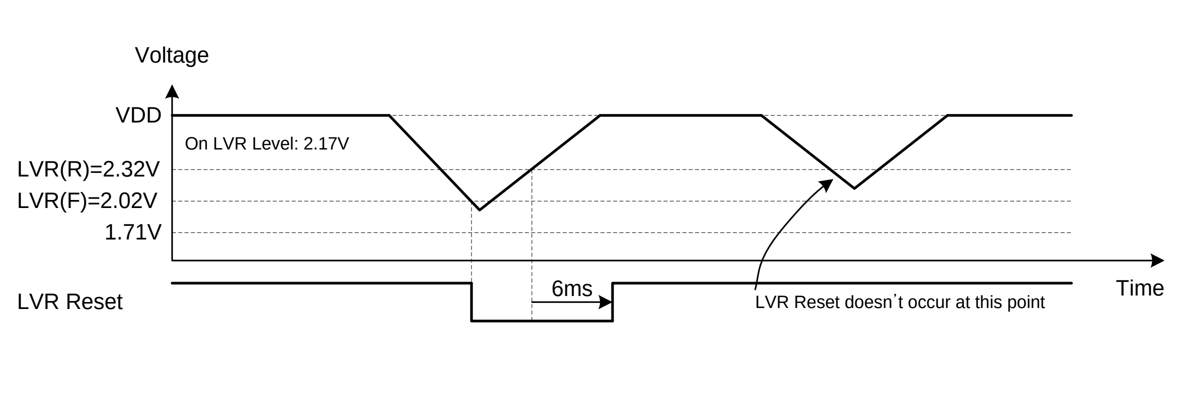


5.3.3 LVR reset

The LVR voltage level is set by a low voltage reset configuration register (CONF_LVRCNFIG) in the Configure Option Page 1. The LVR reset status appears in the register SCU_RSTSSR.

The LVR reset is controlled by the register SCU_LVRCR. The register is cleared to “0x00” when the POR/WAKUP3 reset occurs.

Figure 20. LVR Reset Timing Diagram

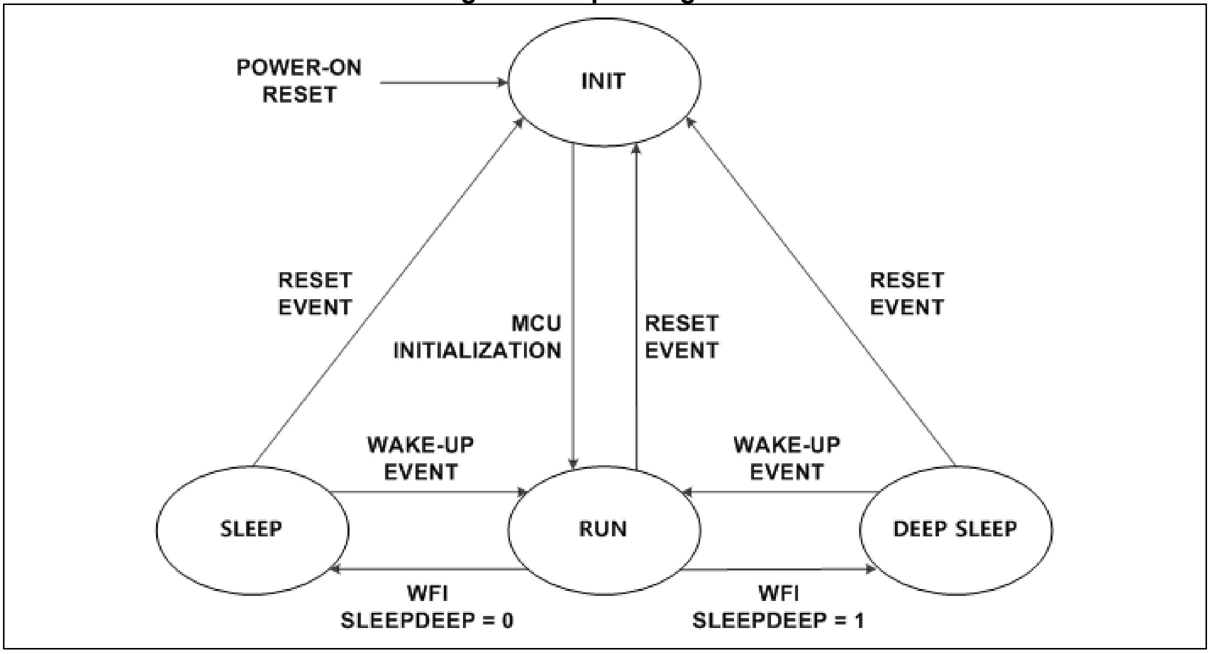


5.4 Operation mode

INIT mode is the initial state of the device when reset. At RUN mode, the chip runs at its max CPU performance with a high-speed clock system. At SLEEP and DEEP SLEEP mode, the chip runs at a low power consumption mode. The system saves power by halting the processor core and unused peripherals.

Figure 21 shows the operation mode transition diagram.

Figure 21. Operating Mode



5.4.1 RUN mode

This mode is to operate CPU core and peripheral hardware with a high-speed clock. The device enters in the INIT state after reset, and then enters in the RUN mode.

5.4.2 SLEEP mode

The device stops only CPU in this mode. Each peripheral function turns on by a function enable bit and a clock enable bit of the register SCU_PPCLKEN.

5.4.3 DEEP SLEEP mode

The device stops not only CPU but also a selected system clock (MCLK) in this mode. RTCC with sub clock, T60, and watchdog timer with WDTRC still operate in DEEP SLEEP mode 0/1/2.

5.4.4 SHUT DOWN mode

The device stops CPU, a selected system clock (MCLK), and most of the peripherals in this mode (DEEP SLEEP mode 3). The T60 only can operate in SHUT DOWN mode.

5.5 Pin description for SCU

Table 16. Pins and External Signals for SCU

Pin name	Type	Description
nRESET	I	External Reset Input
XIN/XOUT	OSC	External Crystal Oscillator for Main Clock
SXIN/SXOUT	OSC	External Crystal Oscillator for Sub Clock
CLKO	O	Clock Output Monitoring Signal

5.6 Registers

Base address and register map of SCU (chip configuration) are shown in Table 17 and Table 18.

Table 17. Base Address of SCU (Chip Configuration)

Name	Base address
SCU (chip configuration)	0x4000_F000

Table 18. SCU Register Map (Chip Configuration)

Name	Offset	Type	Description	Reset value
SCU_VENDORID	0x0000	R	Vendor Identification Register	0x41424F56
SCU_CHIPID	0x0004	R	Chip Identification Register, Where n = 0 or 1.	0x4D31F01n
SCU_REVNR	0x0008	R	Revision Number Register	0x000000xx
–	–	–	Reserved	–
SCU_PMREMAP	0x0014	RW	Program Memory Remap Register	0x00000000
SCU_BTPSCR	0x0018	RW	Boot Pin Status and Control Register	0x000000xx
SCU_RSTSSR	0x001C	RW	Reset Source Status Register	0x000000xx
SCU_NMISRCR	0x0020	RW	NMI Source Selection Register	0x00000000
SCU_SWRSTR	0x0024	R	Software Reset Register	0x00000000
SCU_SRSTVR	0x0028	R	System Reset Validation Register	0x00000055
SCU_WUTCR	0x002C	RW	Wake-up Timer Control Register	0x00000000
SCU_WUTDR	0x0030	RW	Wake-up Timer Data Register	0x00001F40
–	–	–	Reserved	–
SCU_HIRCTRM	0x00A8	RW	High Frequency Internal RC Trim Register (HIRCNFIG)	0x0000xxxx
SCU_WDTRCTRM	0x00AC	RW	Watchdog Timer RC Trim Register (WDTRCNFIG)	0x000000xx

NOTE: The CHIPID is written by H/W if the proper configure address is read.

Base address and register map of SCU (clock generation) are shown in Table 19 and Table 20.

Table 19. Base Address of SCU (Clock Generation)

Name	Base address
SCU (clock generation)	0x4000_1800

Table 20. SCU Register Map (Clock Generation)

Name	Offset	Type	Description	Reset value
SCU_SCCR	0x0000	RW	System Clock Control Register	0x00000000
SCU_CLKSRCR	0x0004	RW	Clock Source Control Register	0x0000000C
SCU_SCDIVR1	0x0008	RW	System Clock Divide Register 1	0x00000000
SCU_SCDIVR2	0x000C	RW	System Clock Divide Register 2	0x00000000
SCU_CLKOCR	0x0010	RW	Clock Output Control Register	0x00000000
SCU_CMONCR	0x0014	RW	Clock Monitoring Control Register	0x00000000
SCU_PPCLKEN1	0x0020	RW	Peripheral Clock Enable Register 1	0x00000000
SCU_PPCLKEN2	0x0024	RW	Peripheral Clock Enable Register 2	0x00020000
SCU_PPCLKSR	0x0040	RW	Peripheral Clock Selection Register	0x00000000
SCU_PPRST1	0x0060	RW	Peripheral Reset Register 1	0x00000000
SCU_PPRST2	0x0064	RW	Peripheral Reset Register 2	0x00000000
SCU_XSOSC	0x0084	RW	Sub Oscillator Control Register	0x00000038

Base address and register map of SCU (LVR/LVI) are shown in Table 21 and Table 22.

Table 21. Base Address of SCU (LVR/LVI)

Name	Base address
SCU (LVR/LVI)	0x4000_5100

Table 22. SCU Register Map (LVR/LVI)

Name	Offset	Type	Description	Reset value
SCU_LVICR	0x0000	RW	Low Voltage Indicator Control Register	0x00000000
SCU_LVRCR	0x0004	RW	Low Voltage Reset Control Register	0x00000000

5.6.1 SCU_VENDORID: vendor id register

SCU_VENDORID register shows Vendor identification information. This register is a 32-bit read-only register.

SCU_CIDR=0x4000_F000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VENDID																															
0x4142_4F56																															
RO																															

31 0	VENDID	Vendor Identification bits. 0x4142_4F56

5.6.2 SCU_CHIPID: chip ID register

SCU_CHIPID register shows chip identification information. This register is a 32-bit read-only register.

SCU_CHIPID=0x4000_F004

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CHIPID																															
0x4D31F010 or 0x4D31F011																															
RO																															

31 0	CHIPID	Chip Identification bits.	
		0x4D31F010	A31L214 (128KB Flash memory for program)
		0x4D31F011	A31L213 (64KB Flash memory for program)

5.6.3 SCU_REVNR: revision number register

SCU_REVNR register is a 32-bit read-only register. This register is accessible in 32/16/8-bit.

SCU_REVNR=0x4000_F008

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																								REVNO							
0x000000																								xx							
–																								RO							

7 0	REVNOChip Revision Number. This value is assigned by the manufacturer.

5.6.4 SCU_PMREMAP: program memory remap register

SCU_PMREMAP register is 32-bit size.

SCU_PMREMAP=0x4000_F014

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="16">WTIDKY</td><td colspan="8">nPMREM</td><td colspan="8">PMREM</td></tr>
<tr><td colspan="16">0x0000</td><td colspan="8">0x00</td><td colspan="8">0x00</td></tr>
<tr><td colspan="16">WO</td><td colspan="8">WO</td><td colspan="8">RW</td></tr>
</table>

31 16	WTIDKY	Write Identification Key When writing, write 0xE2F1 to these bits, or else writing is ignored.
15 8	nPMREM	Write Complement Key When writing, write the complement value of PMREM[7:0], or else writing is ignored.
7 0	PMREM	Program Memory Remap. 0x69 Boot ROM is re-mapped to address 0x00000000. 0x10001000 of Flash memory is re-mapped to address 0x00001000. Others Flash memory is re-mapped to address 0x00000000. **NOTE**: The remapped program memory can be accessed from the original address.

5.6.5 SCU_BTPSCR: boot pin status and control register

SCU_BTPSCR register is 32-bit size and accessible in 32/16/8-bit.

SCU_BTPSCR=0x4000_F018

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="24">Reserved</td><td>Reserved</td><td colspan="2">BFIND</td><td colspan="4">Reserved</td><td>BTPSTA</td></tr>
<tr><td colspan="24">0x000000</td><td>0</td><td colspan="2">xx</td><td colspan="4">0000</td><td>x</td></tr>
<tr><td colspan="24">-</td><td>-</td><td colspan="2">RW</td><td colspan="4">-</td><td>RO</td></tr>
</table>

6 5	BFIND	BOOT Pin Function Indicator. The BFIND[1:0] bits are cleared to “00” by POR, the BFIND[1] bit is cleared to ‘0’ by nRESET, and the bits are not cleared by other system reset. One of the two of the following must be set in the BFIND[1:0] bits to check whether ISP is needed or not. 10 Check the BOOT pin when the system resets by nRESET including POR. 11 Check the BOOT pin when the system resets by POR and WAKUP3.
0	BTPSTA	BOOT Pin Status. 0 The BOOT pin is low level. 1 The BOOT pin is high level. Note) This bit is always ‘1’ if the BOOT pin is not selected for alternative function.

NOTE: When a system reset occurs, the PD5 pin is configured as alternative function for BOOT, the pull-up resistor is enabled, and the debounce filter is enabled.

5.6.6 SCU_RSTSSR: reset source status register

SCU_RSTSSR register shows reset source information when reset event is occurred. ‘1’ implies a reset event exists, while ‘0’ means a reset event does not exist for a corresponding reset source.

When a reset source is detected, ‘1’ is written into the corresponding bit position and reset status will be cleared.

SCU_RSTSSR register is 32-bit size and accessible in 32/16/8-bit.

SCU_RSTSSR=0x4000_F01C

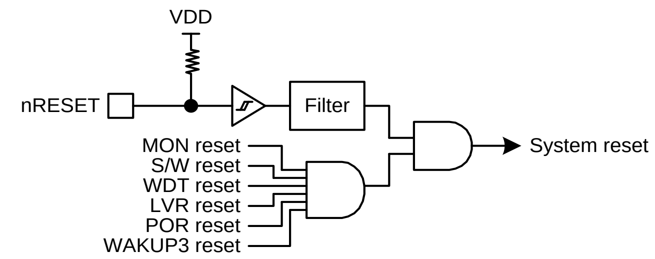
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																								Reserved		MONSTA	SWSTA	EXTSTA	WDTSTA	LVRSTA	PORSTA
0x000000																								00		x	x	x	x	x	x
-																								-		RW	RW	RW	RW	RW	RW

5	MONSTA	Clock Monitoring Reset Status.	
		0	Not detected
		1	Clock monitoring reset is detected. The bit is cleared to ‘0’ when ‘1’ is written.
4	SWSTA	Software Reset Status.	
		0	Not detected
		1	Software reset is detected. The bit is cleared to ‘0’ when ‘1’ is written.
3	EXTSTA	External Pin Reset Status.	
		0	Not detected
		1	External pin reset is detected. The bit is cleared to ‘0’ when ‘1’ is written.
2	WDTSTA	Watchdog Timer Reset Status.	
		0	Not detected
		1	Watchdog Timer reset is detected. The bit is cleared to ‘0’ when ‘1’ is written.
1	LVRSTA	LVR Reset Status.	
		0	Not detected
		1	LVR reset is detected. The bit is cleared to ‘0’ when ‘1’ is written.
0	PORSTA	POR Reset Status.	
		0	Not detected
		1	POR reset is detected. The bit is cleared to ‘0’ when ‘1’ is written.

NOTES:

1. The PORSTA bit is set to “1b” and the other bits are cleared to “0b” when power-on reset occurs.
2. In case of wake-up in deep sleep mode 3 (shutdown), the PORSTA bit may not be set to “1b”. So, check the WAKUPST3 bit of PMU_PWRSR register.
3. The corresponding reset status bit may be set to “1b” if any reset signal is asserted during power-on reset occurs. For example, The EXTSTA bit may be set if the external reset is asserted during POR.

Figure 22. Reset Circuit Diagram



NOTE: A core S/W reset on debugger will reset only CPU.

5.6.7 SCU_NMISRCR: NMI source selection register

SCU_NMISRCR is the non-maskable interrupt configuration register, which can be set by software. SCU_NMISRCR register is 32-bit size, and accessible in 32/16/8-bit.

SCU_NMISRCR=0x4000_F020

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8	7	6	5	4 3 2 1 0
Reserved	**NMICON**	**MONINT**	**Reserved**	**NMISRC**
0x000000	**0**	**0**	**0**	**00000**
-	**RW**	**RW**	**-**	**RW**

7	NMICON	Non-Maskable Interrupt (NMI) Control.	
		0	Disable NMI
		1	Enable NMI
6	MONINT	Clock Monitoring Interrupt Selection.	
		0	Non-select clock monitoring interrupt for NMI source
		1	Select clock monitoring interrupt for NMI source
4	NMISRC	Non-Maskable Interrupt Source Selection.	
0		Select one of the interrupt sources 0 to 31 for NMI source.	

NOTE: The interrupt source which is selected for NMI should be disabled in NVIC to avoid both generation of the normal and NMI interrupts.

5.6.8 SCU_SWRSTR: software reset register

SCU_SWRSTR register is 32-bit size.

SCU_SWRSTR=0x4000_F024

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
WTIDKY	**Reserved**	**SWRST**
0x0000	**0x00**	**0x00**
WO	-	**WO**

31	WTIDKY	Write Identification Key	
16		When writing, write 0x9EB3 to these bits, or else writing is ignored.	
7	SWRST	Software Reset (System Reset)	
0		0x2D	A software reset will be generated for all peripheral and core.
		Others	No effect

5.6.9 SCU_SRSTVR: system reset validation register

SCU_SRSTVR register is 32-bit size, and accessible in 32/16/8-bit.

SCU_SRSTVR=0x4000_F028

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																								VALID							
0x000000																								0x55							
-																								RO							

7 0	VALID	System Reset Validation.	
		0x55	System reset is O.K.
		Others	A weak system reset. A system reset must be generated by S/W

5.6.10 SCU_WUTCR: wake-up timer control register

Wake-up timer always works on operating mode. This timer gives a stable time for clock generation during Power on and DEEP SLEEP mode release. The main purpose of this timer is periodical tick timer or a wake-up source.

SCU_WUTCR register is 32-bit size, and accessible in 32/16/8-bit.

SCU_WUTCR=0x4000_F02C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																								WUTIEN	Reserved					CNTRLD	WUTIFLAG
0x000000																								0	00000					0	0
–																								RW	-					RW	RW

7	WUTIEN	Wake-up Timer Interrupt Enable bit	
		0	Disable wake-up timer interrupt
		1	Enable wake-up timer interrupt
1	CNTRLD	Counter Reload bit	
		0	No effect
		1	Reload data to counter (Automatically cleared to '0' after operation)
0	WUTIFLAG	Wake-up Timer Interrupt Flag bit	
		0	No request occurred
		1	Request occurred. The bit is cleared to '0' when '1' is written.
		NOTE: This bit may not be set to "1b" if the PCLK frequency is slower than the HCLK frequency. So, in order for WUT interrupt to occur normally, the SCU_SCDIVR2.PDIV[1:0] bits must be set to "00b" so that the PCLK frequency is the same as the HCLK frequency.	

5.6.11 SCU_WUTDR: wake-up timer data register

SCU_WUTDR register is 32-bit size and accessible in 32/16/8-bit.

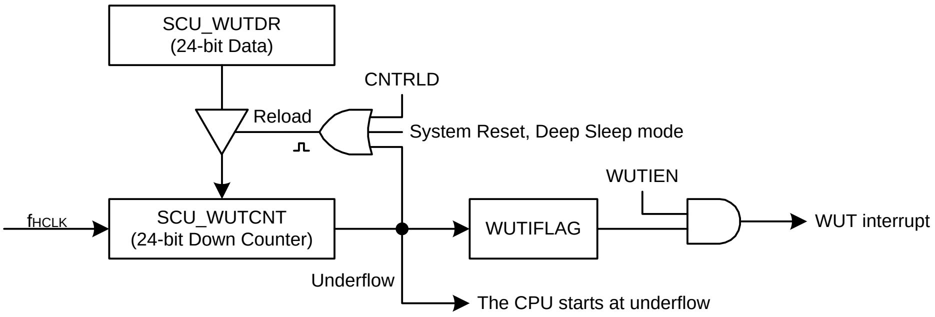
SCU_WUTDR=0x4000_F030

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								**WUTDATA**																							
0x00								**0x001F40**																							
–								**RW**																							

23 WUTDATA Wake-up Timer Data. The range is 0x000000 to 0xFFFFFF.
0 **NOTES**:

1. When HIRC is system clock, its value should be set to be at least more than 1us.
2. When WDTRC is system clock, its value should be set to be at least more than 100us.

Figure 23. Wake-up Timer Block Diagram



NOTE: The WUTIFLAG bit may not be set to "1b" if the PCLK frequency is slower than the HCLK frequency. So, in order for WUT interrupt to occur normally, the SCU_SCDIVR2.PDIV[1:0] bits must be set to "00b" so that the PCLK frequency is the same as the HCLK frequency.

5.6.12 SCU_HIRCTRM: high frequency internal RC trim register

SCU_HIRCTRM register may be used for user trimming of HIRC by s/w. This register is 32-bit size.

SCU_HIRCTRM=0x4000_F0A8

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WTIDKY																**Reserved**							**CTRMH**			**FTRMH**					
0x0000																x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
WO																I	I	I	I	I	I	I	**RW**			**RW**					

31 16	WTIDKY	Write Identification Key When writing, write 0xA6B5 to these bits, or else writing is ignored.
8 6	CTRMH	Factory HIRC Coarse Trim. These bits are fixed by manufacturer and read from “Configure Option Page 0” when a system reset occurs. These bits provide a user programmable trimming value on operation. The range is -4 to +3, the CTRMH[2] is sign bit, and the frequency is changed by 2.3MHz step-by-step.
5 0	FTRMH	Factory HIRC Fine Trim. These bits are fixed by manufacturer and read from “Configure Option Page 0” when a system reset occurs. These bits provide a user programmable trimming value on operation. The range is -32 to +31, the FTRMH[5] is sign bit, and the frequency is changed by 80kHz step-by-step.

5.6.13 SCU_WDTRCTRM: Watchdog Timer RC trim register

SCU_WDTRCTRM register may be used for user trimming of WDTRC by s/w. This register is 32-bit size.

SCU_WDTRCTRM=0x4000_F0AC

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="16">WTIDKY</td><td colspan="8">nTRMW</td><td colspan="4">CTRMW</td><td>Reserved</td><td colspan="3">FTRMW</td></tr>
<tr><td colspan="16">0x0000</td><td colspan="8">xx</td><td>x</td><td>x</td><td>x</td><td>x</td><td>0</td><td>x</td><td>x</td><td>x</td></tr>
<tr><td colspan="16">WO</td><td colspan="8">WO</td><td colspan="4">RW</td><td>–</td><td colspan="3">RW</td></tr>
</table>

31 16	WTIDKY	Write Identification Key When writing, write 0x4C3D to these bits, or else writing is ignored.
15 8	nTRMW	Write Complement Key When writing, write the complement value of LSB(CTRMW+FTRMW), otherwise the write is ignored.
7 4	CTRMW	Factory WDTRC Coarse Trim. These bits are fixed by manufacturer and read from “Configure Option Page 0” when the system resets. These bits provide a user-programmable trimming value on operation. The range is -8 to +7, the CTRMW[3] is sign bit, and the frequency is changed by about 2.8kHz steps.
2 0	FTRMW	Factory WDTRC Fine Trim. These bits are fixed by manufacturer and read from “Configure Option Page 0” when the system resets. These bits provide a user-programmable trimming value on operation. The range is -4 to +3, the FTRMW[2] is sign bit, and the frequency is changed by about 0.6kHz steps.

5.6.14 SCU_SCCR: system clock control register

The A31L21x series has multiple clock sources to generate internal operating clocks. SCU_SCCR register controls such a clock source.

This register is 32-bit size.

SCU_SCCR=0x4000_1800

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2	1 0
WTIDKY	**Reserved**		**MCLKSEL**
0x0000	**0x00**	**0 0 0 0 0 0**	**0 0**
WO	**–**	**–**	**RW**

31 16	WTIDKY	Write Identification Key	
		When writing, write 0x570A to these bits, or else writing is ignored.	
1 0	MCLKSEL	Main Clock Selection, MCLK	
		00	High frequency Internal RC oscillator (32MHz), HIRC
		01	External main oscillator (2 – 32MHz), XMOSC
		10	External sub oscillator (32.768kHz), XSOSC
		11	Internal watchdog timer RC oscillator (40kHz), WDTRC

NOTES:

1. The MCLKSEL bits will not be changed on selecting the clock which is disabled by SCU_CLKSRCR register.
2. If the MCLKSEL bits are “10” or “11”, the HDIV[2:0] bits of SCU_SCDIVR1 register should be “100” for non-divided system clock.

5.6.15 SCU_CLKSRCR: clock source control register

The A31L21x series has multiple clock sources to generate internal operating clocks. SCU_CLKSRCR register controls each clock source.

This register is 32-bit size.

SCU_CLKSRCR=0x4000_1804

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WTIDKY																**Reserved**		**HIRCSEL**		**Reserved**			**XMFRNG**	**Reserved**				**WDTRCEN**	**HIRCEN**	**XMOSCEN**	**XSOSCEN**
0x0000																**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**1**	**0**	**0**
WO																**-**	**-**	**RW**		**-**	**-**	**-**	**RW**	**-**	**-**	**-**	**-**	**RW**	**RW**	**RW**	**RW**

31 16	WTIDKY	Write Identification Key When writing, write 0xA507 to these bits, or else writing is ignored.
13 12	HIRCSEL	HIRC Frequency Selection bits
		00 32MHz HIRC
		01 16MHz HIRC
		10 8MHz HIRC
		11 4MHz HIRC
8	XMFRNG	Main Oscillator Type and Frequency Range Selection bit
		0 x-tal for XMOSC, 2 to 16MHz
		1 External clock for XMOSC, 2MHz to 32MHz
3	WDTRCEN	WDTRC Enable bit, Watchdog Timer RC oscillator
		0 Disable WDTRC
		1 Enable WDTRC
2	HIRCEN	HIRC Enable bit, High frequency internal RC oscillator
		0 Disable HIRC
		1 Enable HIRC
1	XMOSCEN	XMOSC Enable bit, External main oscillator
		0 Disable XMOSC
		1 Enable XMOSC
0	XSOSCEN	XSOSC Enable bit, External sub oscillator. The bit is cleared to “0b” by the reset of POR/WAKUP3 but retained by the other reset.
		0 Disable XSOSC
		1 Enable XSOSC

NOTE: The clock selected as a main system clock by SCU_SCCR register will not be changed by the corresponding bit.

5.6.16 SCU_SCDIVR1: system clock divide register 1

SCU_SCDIVR1 register is 32-bit size and accessible in 32/16/8-bit.

SCU_SCDIVR1=0x4000_1808

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																									**WLDIV**			**Reserved**	**HDIV**		
0x000000																									**0**	**0**	**0**	**0**	**0**	**0**	**0**
–																									**RW**			**–**	**RW**		

6	WLDIV	Clock Divide bits for RTCC and LCD Driver, Divider 2 (Refer to Figure 15)	
4		000	MCLK÷64
		001	MCLK÷128
		010	MCLK÷256
		011	MCLK÷512
		100	MCLK÷1024
		others	Reserved
2	HDIV	Clock Divide bits for HCLK, Divider 0 (Refer to Figure 15)	
0		000	MCLK÷16
		001	MCLK÷8
		010	MCLK÷4
		011	MCLK÷2
		100	MCLK÷1
		others	Reserved (MCLK÷1)

NOTES:

1. If the selected MCLK is XSOSC or WDTRC, the HDIV[2:0] bits should be set to “100”.
2. The frequency range of HCLK should be 2.0 to 32[MHz] by s/w while the HIRC is the system clock.
3. After changing the value of HDIV[2:0] bits for system clock speed, a delay 10us is required before changing the value again immediately.

5.6.17 SCU_SCDIVR2: system clock divide register 2

SCU_SCDIVR2 register is 32-bit size and accessible in 32/16/8-bit.

SCU_SCDIVR2=0x4000_180C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																										**SYSTDIV**		**Reserved**		**PDIV**	
0x000000																										**0**	**0**	**0**	**0**	**0**	**0**
–																										**RW**		**–**	**–**	**RW**	

5 4	SYSTDIV	Clock Divide bits for SysTick Timer, Divider 3 (Refer to Figure 15)	
		00	HCLK÷1
		01	HCLK÷2
		10	HCLK÷4
		11	HCLK÷8
1 0	PDIV	Clock Divide bits for PCLK, Divider 1 (Refer to Figure 15)	
		00	HCLK÷1
		01	HCLK÷2
		10	HCLK÷4
		11	HCLK÷8

NOTE: If the selected MCLK is XSOSC or WDTRC, the PDIV[1:0] should be set to “00”.

5.6.18 SCU_CLKOCR: clock output control register

The A31L21x series can drive the clock from a selected clock (CLKOS) with a dedicated post divider.

SCU_CLKOCR register is 32-bit size and accessible in 32/16/8-bit.

SCU_CLKOCR=0x4000_1810

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																								**CLKOEN**	**POLSEL**	**CLKODIV**			**CLKOS**		
0x000000																								**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
–																								**RW**	**RW**	**RW**			**RW**		

7	CLKOEN	Clock Output Enable bit	
		0	Disable clock output
		1	Enable clock output
6	POLSEL	Clock Output Polarity Selection bit when disable	
		0	Low level during disable
		1	High level during disable
5	CLKODIV	Output Clock Divide bits, Divider 4 (Refer to Figure 15)	
3		000	“Selected clock”÷1
		001	“Selected clock”÷2
		010	“Selected clock”÷4
		011	“Selected clock”÷8
		100	“Selected clock”÷16
		101	“Selected clock”÷32
		110	“Selected clock”÷64
		111	“Selected clock”÷128
2	CLKOS	Clock Output Selection bits	
0		000	MCLK
		001	WDTRC
		010	HIRC
		011	HCLK
		100	PCLK
		others	Reserved (None)

5.6.19 SCU_CMONCR: clock monitoring control register

Internal clock can be monitored by using internal WDTRC for security purpose.

SCU_CMONCR register is 32-bit size and accessible in 32/16/8-bit.

SCU_CMONCR=0x4000_1814

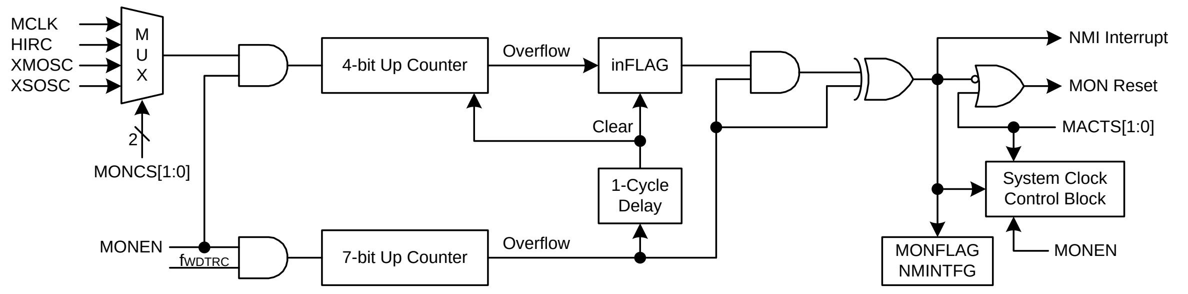
31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
Reserved	**MONEN**	**MACTS**		**Reserved**	**MONFLAG**	**NMINTFG**	**MONCS**	
0x000000	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
–	**RW**	**RW**		**-**	**RW**	**RW**	**RW**	

7	MONEN	Clock Output Enable bit	
		0	Disable clock monitoring
		1	Enable clock monitoring
		NOTE: When this bit is reset to '0', the block clears the 4/7-bit counter, inFLAG, and flags.	
6 5	MACTS	Clock Monitoring Action Selection bits	
		00	No action by clock monitoring, but flags will be set/cleared on condition
		01	Reset generation by clock monitoring
		10	The system clock will be changed to the WDTRC regardless of MCLKSEL[1:0] bits of system clock control register (SCU_SCCR) only when the MCLK is selected for monitoring.
		11	Not used
3	MONFLAG	Clock Monitoring Result Flag bit	
		0	The clock under monitoring is not ready.
		1	The clock under monitoring is ready. This bit is cleared to '0' when '1' is written.
2	NMINTFG	Clock Monitoring Interrupt Flag bit (only when the MCLK is selected for monitoring)	
		0	No request occurred
		1	Request occurred. The bit is cleared to '0' when '1' is written.
		NOTE: When the bit is set, the system clock must be switched to WDTRC by S/W.	
1 0	MONCS	Monitored Clock Selection bits	
		00	MCLK
		01	HIRC
		10	XMOSC
		11	XSOSC

NOTES:

1. The block should be enabled after disable to clear the internal status for new clock monitoring.
2. This block must be disabled by S/W before entering DEEP SLEEP mode.

Figure 24. Clock Monitoring Circuit Diagram



NOTE: WDTRC should be enabled to use this block since it is the watch clock for monitoring.

5.6.20 SCU_PPCLKEN1: peripheral clock enable register 1

To use a certain peripheral unit, its clock should be activated by writing ‘1’ to the corresponding bit in SCU_PPCLKEN1/SCU_PPCLKEN2 register. Until enabling the clock, the peripheral does not operate properly. To stop the clock of the peripheral unit, write ‘0’ to the corresponding bit in the SCU_PPCLKEN1/PPCLKEN2 register.

SCU_PPCLKEN1 register is 32-bit size and accessible in 32/16/8-bit.

SCU_PPCLKEN1=0x4000_1820

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
T60CLKE	T50CLKE	Reserved				T43CLKE	T42CLKE	T41CLKE	T40CLKE	Reserved			T12CLKE	T11CLKE	T10CLKE	Reserved	TSCLKE	RNGCLKE	AESCLKE	Reserved			T13CLKE	Reserved		PFCLKE	PECLKE	PDCLKE	PCCLKE	PBCLKE	PACLKE
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	-	-	-	-	RW	RW	RW	RW	-	-	-	RW	RW	RW	-	RW	RW	RW	-	-	-	RW	-	-	RW	RW	RW	RW	RW	RW

31	T60CLKE	TIMER60 clock enable
30	T50CLKE	TIMER50 clock enable
25	T43CLKE	TIMER43 clock enable
24	T42CLKE	TIMER42 clock enable
23	T41CLKE	TIMER41 clock enable
22	T40CLKE	TIMER40 clock enable
18	T12CLKE	TIMER12 clock enable
17	T11CLKE	TIMER11 clock enable
16	T10CLKE	TIMER10 clock enable
14	TSCLKE	Temperature Sensor clock enable
13	RNGCLKE	Random Number Generator clock enable
12	AESCLKE	AES-128 clock enable
8	T13CLKE	TIMER13 clock enable
5	PFCLKE	Port F clock enable
4	PECLKE	Port E clock enable
3	PDCLKE	Port D clock enable
2	PCCLKE	Port C clock enable
1	PBCLKE	Port B clock enable
0	PACLKE	Port A clock enable

NOTE: The peripheral registers may not be read/written by software when the peripheral clock is disabled.

5.6.21 SCU_PPCLKEN2: peripheral clock enable register 2

SCU_PPCLKEN2 register is 32-bit size and accessible in 32/16/8-bit.

SCU_PPCLKEN2=0x4000_1824

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DMACLKE	DFMCLCKE	Reserved	CMP1CLKE	CMP0CLKE	Reserved		SC0CLKE	SPI3CLKE	SPI2CLKE	SPI1CLKE	SPI0CLKE	FMCLKE	LVICLKE	WDTCLKE	LPUT1CLKE	LPUT0CLKE	Reserved	LCDCLKE	CRCLKE	RTCCLKE	ADCLKE	Reserved	I2C2CLKE	I2C1CLKE	I2C0CLKE	Reserved			UT0CLKE	UST11CLKE	UST10CLKE
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	-	RW	RW	-	-	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	-	RW	RW	RW	RW	-	RW	RW	RW	-	-	-	RW	RW	RW

31	DMACLKE	DMA clock enable
30	DFMCLKE	Data Flash Memory Control clock enable. Ignored during Data Flash operation.
28	CMP1CLKE	Comparator 1 clock enable
27	CMP0CLKE	Comparator 0 clock enable
24	SC0CLKE	Smart Card Interface 0 (ISO 7816-3) clock enable
23	SPI3CLKE	SPI 3 clock enable
22	SPI2CLKE	SPI 2 clock enable
21	SPI1CLKE	SPI 1 clock enable
20	SPI0CLKE	SPI 0 clock enable
19	FMCLKE	Flash Memory Control clock enable. Ignored during Flash operation.
18	LVICLKE	LVI (Low Voltage Indicator) clock enable
17	WDTCLKE	WDT (Watchdog Timer) clock enable. The WDTRC won't be disabled if the clock is enabled by watchdog timer configuration register (CONF_WDTCNFIG) in "Configure Option Page 1"
16	LPUT1CLKE	Low Power UART 1 clock enable
15	LPUT0CLKE	Low Power UART 0 clock enable
13	LCDCLKE	LCD Controller clock enable
12	CRCLKE	CRC (Cyclic Redundancy Check) clock enable
11	RTCCLKE	Real Time Clock/Calendar clock enable. The bit is cleared to "0b" by the reset of POR/WAKUP3 but retained by the other reset.
10	ADCLKE	ADC (Analog to Digital Converter) clock enable
8	I2C2CLKE	I2C2 (Inter-integrated Circuit) clock enable
7	I2C1CLKE	I2C1 (Inter-integrated Circuit) clock enable
6	I2C0CLKE	I2C0 (Inter-integrated Circuit) clock enable
2	UT0CLKE	UART0 clock enable
1	UST11CLKE	USART11 clock enable
0	UST10CLKE	USART10 clock enable

NOTE: The peripheral registers may not be read/written by software when the peripheral clock is disabled.

5.6.22 SCU_PPCLKSR: peripheral clock selection register

SCU_PPCLKSR register is 32-bit size and accessible in 32/16/8-bit.

SCU_PPCLKSR=0x4000_1840

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved						T50CLK		Reserved										LPUT1CLK		LPUT0CLK		RTCCLK		LCDCLK		Reserved					WDTCLK
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-	-	-	-	-	-	RW	RW	-	-	-	-	-	-	-	-	-	-	RW	RW	RW	RW	RW	RW	RW	RW	-	-	-	-	-	RW

25 24	T50CLK	Timer 50 Clock Selection.	
		00	PCLK clock
		01	WDTRC clock
		10	HIRC clock
		11	XSOSC clock
13 12	LPUT1CLK	Low Power UART 1 Clock Selection.	
		00	PCLK clock
		01	HIRC clock
		10	XSOSC clock
		11	Reserved
11 10	LPUT0CLK	Low Power UART 0 Clock Selection.	
		00	PCLK clock
		01	HIRC clock
		10	XSOSC clock
		11	Reserved
9 8	RTCCLK	Real Time Clock/Calendar Clock Selection. The bits are cleared to “00b” by the reset of POR/WAKUP3 but retained by the other reset.	
		00	Low level (RTC stuck)
		01	XSOSC clock
		10	WDTRC clock
		11	A clock of the MCLK which is divided by divider 2
7 6	LCDCLK	LCD Driver Clock Selection.	
		00	A clock of the MCLK which is divided by divider 2 (Refer to Figure 15)
		01	XSOSC clock
		10	WDTRC clock
		11	Reserved
0	WDTCLK	Watchdog Timer Clock Selection.	
		0	WDTRC clock
		1	PCLK clock

5.6.23 SCU_PPRST1: peripheral reset register 1

SCU_PPRST1/PPRST2 register can make a peripheral reset. If a specific bit in this register is set to ‘1’, the peripheral corresponded with this bit occurs a reset event and the registers of the peripheral are initialized with reset values.

SCU_PPRST1 register is 32-bit size and accessible in 32/16/8-bit.

SCU_PPRST1=0x4000_1860

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
T60RST	T50RST	Reserved				T43RST	T42RST	T41RST	T40RST	Reserved			T12RST	T11RST	T10RST	Reserved	TSRST	RNGRST	AESRST	Reserved			T13RST	Reserved		PFRST	PERST	PDRST	PCRST	PBRST	PARST
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	-	-	-	-	RW	RW	RW	RW	-	-	-	RW	RW	RW	-	RW	RW	RW	-	-	-	RW	-	-	RW	RW	RW	RW	RW	RW

31	T60RST	Timer 60 Reset bit	
		0	No effect
		1	Reset Timer 60, Cleared by software
30	T50RST	Timer 50 Reset bit	
		0	No effect
		1	Reset Timer 50, Cleared by software
25	T43RST	Timer 43 Reset bit	
		0	No effect
		1	Reset Timer 43, Cleared by software
24	T42RST	Timer 42 Reset bit	
		0	No effect
		1	Reset Timer 42, Cleared by software
23	T41RST	Timer 41 Reset bit	
		0	No effect
		1	Reset Timer 41, Cleared by software
22	T40RST	Timer 40 Reset bit	
		0	No effect
		1	Reset Timer 40, Cleared by software
18	T12RST	Timer 12 Reset bit	
		0	No effect
		1	Reset Timer 12, Cleared by software
17	T11RST	Timer 11 Reset bit	
		0	No effect
		1	Reset Timer 11, Cleared by software
16	T10RST	Timer 10 Reset bit	
		0	No effect
		1	Reset Timer 10, Cleared by software
14	TSRST	Temperature Sensor Reset bit	
		0	No effect
		1	Reset Temperature Sensor, Cleared by software
13	RNGRST	Random Number Generator Reset bit	
		0	No effect
		1	Reset RNG, Cleared by software
12	AESRST	AES-128 Reset bit	
		0	No effect
		1	Reset AES-128, Cleared by software
8	T13RST	Timer 13 Reset bit	
		0	No effect
		1	Reset Timer 13, Cleared by software
5	PFRST	Port F Reset bit	
		0	No effect
		1	Reset Port F, Cleared by software

4	PERST	Port E Reset bit	
		0	No effect
		1	Reset Port E, Cleared by software
3	PDRST	Port D Reset bit	
		0	No effect
		1	Reset Port D, Cleared by software
2	PCRST	Port C Reset bit	
		0	No effect
		1	Reset Port C, Cleared by software
1	PBRST	Port B Reset bit	
		0	No effect
		1	Reset Port B, Cleared by software
0	PARST	Port A Reset bit	
		0	No effect
		1	Reset Port A, Cleared by software

5.6.24 SCU_PPRST2: peripheral reset register 2

SCU_PPRST2 register is 32-bit size and accessible in 32/16/8-bit.

SCU_PPRST2=0x4000_1864

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DMARST	DFMCRST	Reserved	CMP1RST	CMP0RST	Reserved		SC0RST	SPI3RST	SPI2RST	SPI1RST	SPI0RST	FMCRST	LVIRST	Reserved	LPUT1RST	LPUT0RST	Reserved	LCDRST	CRRST	RTCRST	ADRST	Reserved	I2C2RST	I2C1RST	I2C0RST	Reserved			UT0RST	UST11RST	UST10RST
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	-	RW	RW	-	-	RW	RW	RW	RW	RW	RW	RW	-	RW	RW	-	RW	RW	RW	RW	-	RW	RW	RW	-	-	-	RW	RW	RW

31	DMARST	DMA Reset bit	
		0	No effect
		1	Reset DMA, Cleared by software
30	DFMCRST	Data Flash Memory Control Reset bit. Ignored during Data Flash operation.	
		0	No effect
		1	Reset Data Flash memory control, Cleared by software.
28	CMP1RST	Comparator 1 Reset bit	
		0	No effect
		1	Reset Comparator 1, Cleared by software
27	CMP0RST	Comparator 0 Reset bit	
		0	No effect
		1	Reset Comparator 0, Cleared by software
24	SC0RST	Smart Card Interface 0 (ISO 7816-3) Reset bit	
		0	No effect
		1	Reset Smart Card Interface 0, Cleared by software
23	SPI3RST	SPI3 Reset bit	
		0	No effect
		1	Reset SPI3, Cleared by software
22	SPI2RST	SPI2 Reset bit	
		0	No effect
		1	Reset SPI2, Cleared by software
21	SPI1RST	SPI1 Reset bit	
		0	No effect
		1	Reset SPI1, Cleared by software
20	SPI0RST	SPI0 Reset bit	
		0	No effect
		1	Reset SPI0, Cleared by software
19	FMCRST	Flash Memory Control (FMC) Reset bit. Ignored during Flash operation.	
		0	No effect
		1	Reset Flash memory control, Cleared by software.
18	LVIRST	Low Voltage Indicator (LVI) Reset bit	
		0	No effect
		1	Reset LVI, Cleared by software
16	LPUT1RST	Low Power UART 1 Reset bit	
		0	No effect
		1	Reset LPUART1, Cleared by software
15	LPUT0RST	Low Power UART 0 Reset bit	
		0	No effect
		1	Reset LPUART0, Cleared by software
13	LCDRST	LCD Controller Reset bit	
		0	No effect
		1	Reset LCD Controller, Cleared by software
12	CRRST	Cyclic Redundancy Check (CRC) Reset bit	
		0	No effect
		1	Reset CRC, Cleared by software
11	RTCRST	Real Time Clock and Calendar (RTCC) Reset bit	
		0	No effect

		1	Reset RTCC, Cleared by software
10	ADRST	Analog to Digital Converter (ADC) Reset bit	
		0	No effect
		1	Reset ADC, Cleared by software
8	I2C2RST	Inter-integrated Circuit (I2C2) Reset bit	
		0	No effect
		1	Reset I2C2, Cleared by software
7	I2C1RST	Inter-integrated Circuit (I2C1) Reset bit	
		0	No effect
		1	Reset I2C1, Cleared by software
6	I2C0RST	Inter-integrated Circuit (I2C0) Reset bit	
		0	No effect
		1	Reset I2C0, Cleared by software
2	UT0RST	UART0 Reset bit	
		0	No effect
		1	Reset UART0 , Cleared by software
1	UST11RST	USART11 Reset bit	
		0	No effect
		1	Reset USART11 , Cleared by software
0	UST10RST	USART10 Reset bit	
		0	No effect
		1	Reset USART10 , Cleared by software

5.6.25 SCU_XSOSC: sub oscillator control register

SCU_XSOSC register is used to select driving current of sub oscillator.

SCU_XSOSC register is 32-bit size and accessible in 32/16/8-bit.

SCU_XSOSC=0x4000_1884

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
Reserved	Reserved		ISET_I			Reserved		
0x000000	0	0	1	1	1	0	0	0
-	-	-	RW	RW	RW	-	-	-

5 3	ISET_I	Sub Oscillator Driving Current Selection.	
		000	Reserved (3rd level driving current)
		001	Reserved (3rd level driving current)
		010	3rd level driving current
		011	4th level driving current
		100	5th level driving current
		101	6th level driving current
		110	7th level driving current
		111	The highest driving current

NOTES:

1. The “111b” should be set when the sub oscillator is started by s/w and the value should be kept during sub oscillator stabilization.
2. After sub oscillator stabilization, the ISET_I bits of SCU_XSOSC register may be changed to a lower value in order to reduce the current consumption due to the sub oscillator.

5.6.26 SCU_LVICR: low voltage indicator control register

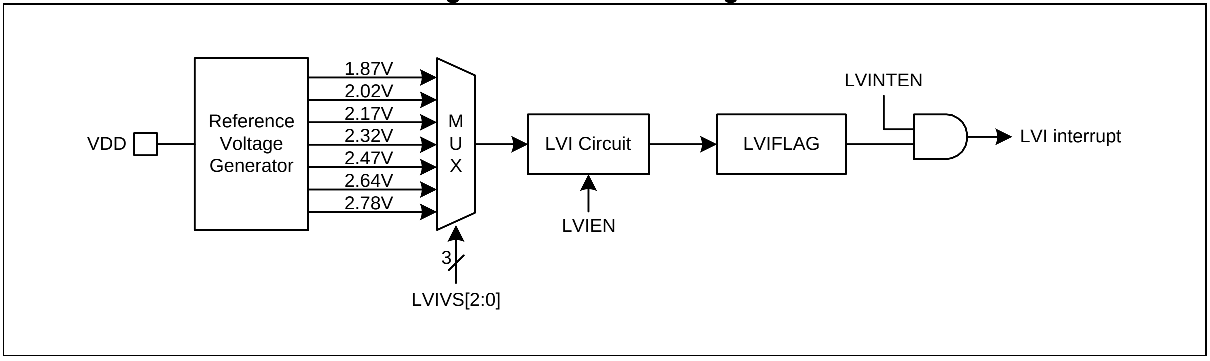
SCU_LVICR register is 32-bit size and accessible in 32/16/8-bit.

SCU_LVICR=0x4000_5100

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
Reserved	**LVIEN**	**Reserved**	**LVINTEN**	**LVIFLAG**	**Reserved**	**LVIVS**		
0x000000	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
–	**RW**	**-**	**RW**	**RW**	**-**	**RW**	**RW**	**RW**

7	LVIEN	LVI Enable.	
		0	Disable low voltage indicator.
		1	Enable low voltage indicator.
5	LVINTEN	LVI Interrupt Enable.	
		0	Disable low voltage indicator interrupt.
		1	Enable low voltage indicator interrupt.
4	LVIFLAG	LVI Interrupt Flag.	
		0	No request occurred.
		1	Request occurred. The bit is cleared to ‘0’ when ‘1’ is written.
2 0	LVIVS	LVI Voltage Selection.	
		000	1.87V
		001	1.87V
		010	2.02V
		011	2.17V
		100	2.32V
		101	2.47V
		110	2.64V
		111	2.78V

Figure 25. LVI Block Diagram



5.6.27 SCU_LVRCR: low voltage reset control register

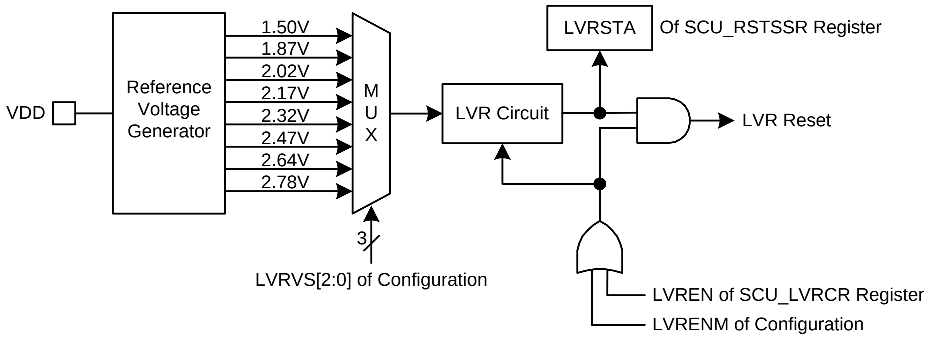
SCU_LVRCR register is 32-bit size and accessible in 32/16/8-bit.

SCU_LVRCR=0x4000_5104

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved			**LVREN**
0x000000			**0x00**
–			**RW**

7 0	LVREN	LVR Enable. These bits are cleared to 0x00 by POR/WAKUP3 only and retained by other reset signals.	
		0x55	Disable low voltage reset.
		Others	Enable low voltage reset.

Figure 26. LVR Block Diagram



NOTE: Voltage level of LVR is set by the low voltage reset configuration register (CONF_LVRCNFIG) in the Configure Option Page 1.

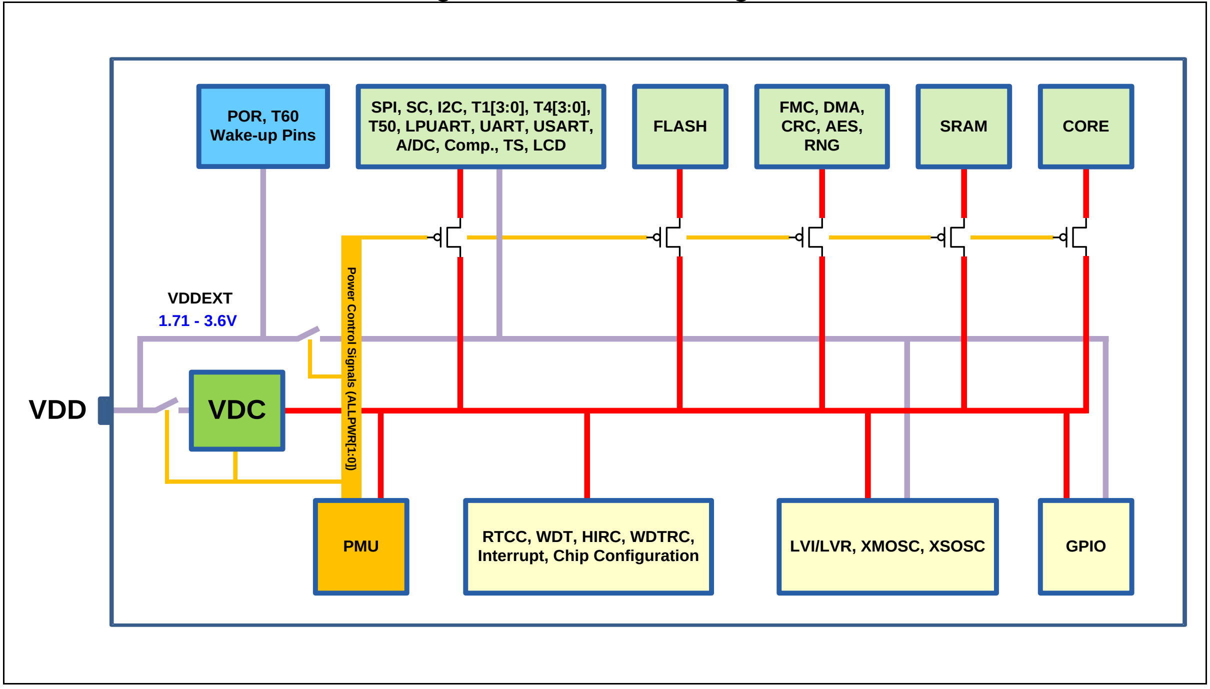
6 Power Management Unit (PMU)

The A31L21x series has a built-in Power Management Unit (PMU), which manages the internal power supply of the system control and peripheral parts and a wake-up time from SLEEP and DEEP SLEEP modes.

This PMU has 32-byte sized backup registers to retain data during DEEP SLEEP mode 0/1/2 except DEEP SLEEP mode 3 (SHUT DOWN mode).

6.1 PMU block diagram

Figure 27. PMU Block Diagram


POR, T60
Wake-up Pins
SPI, SC, I2C, T1[3:0], T4[3:0], T50, LPUART, UART, USART, A/DC, Comp., TS, LCD
FLASH
FMC, DMA, CRC, AES, RNG
SRAM
CORE
VDDEXT
1.71 - 3.6V
Power Control Signals (ALLPWR[1:0])
VDD
VDC
PMU
RTCC, WDT, HIRC, WDTRC, Interrupt, Chip Configuration
LVI/LVR, XMOSC, XSOSC
GPIO

6.2 Functional table on current mode

Table 23. Functional Table on Current Mode

IP	Main RUN (IDD1)	Main SLEEP (IDD2)	Sub RUN (IDD3)	Sub SLEEP (IDD4)	DEEP SLEEP Mode (ALLPWR = x)			
					0 (IDD5)	1 (IDD6)	2 (IDD7)	3 (IDD8)
CPU	O	X	O	X	X	X	Power off	Power off
Flash	O	X	O	X	X	X	Power off	Power off
SRAM	O	X	O	X	X	X	Power off	Power off
Back-up Reg.	O	X	O	X	X	X	X	Power off
PMU	O	O	O	O	O	O	O	Power off
FMC/DFMC	OPT	X	OPT	X	X	X	Power off	Power off
DMA	OPT	OPT	OPT	OPT	X	X	Power off	Power off
CRC	OPT	X	OPT	X	X	X	Power off	Power off
POR	O	O	O	O	O	O	O	O
LVR/LVI	OPT	OPT	OPT	OPT	OPT	OPT	OPT	Power off
GPIO	O	O	O	O	O	X	X	Power off
Wakeup pins	X	X	X	X	X	OPT	OPT	OPT
SCU	O	O	O	O	O	O	O	Power off
SPI	OPT	OPT	OPT	OPT	X	X	Power off	Power off
I2C	OPT	OPT	OPT	OPT	X	X	Power off	Power off
USART	OPT	OPT	OPT	OPT	OPT	X	Power off	Power off
UART	OPT	OPT	OPT	OPT	X	X	Power off	Power off
LPUART	OPT	OPT	OPT	OPT	OPT	MBOFF	Power off	Power off
ISO-7816-3	OPT	OPT	OPT	OPT	X	X	Power off	Power off
SysTick	OPT	OPT	OPT	OPT	X	X	Power off	Power off
T10 – T13	OPT	OPT	OPT	OPT	X	X	Power off	Power off
T40 – T43	OPT	OPT	OPT	OPT	X	X	Power off	Power off
T50	OPT	OPT	OPT	OPT	OPT	MBOFF	Power off	Power off
T60 (100Hz)	OPT	OPT	OPT	OPT	OPT	OPT	OPT	OPT
WDT	OPT	OPT	OPT	OPT	OPT	OPT	OPT	Power off
WUT	O	O	O	O	X	X	X	Power off
A/DC	OPT	X	X	X	X	X	Power off	Power off
Comparator	OPT	OPT	OPT	OPT	OPT	MBOFF	Power off	Power off
TS	OPT	OPT	X	X	X	X	Power off	Power off
LCD Driver	OPT	OPT	OPT	OPT	OPT	MBOFF	Power off	Power off
RTCC	OPT	OPT	OPT	OPT	OPT	OPT	OPT	Power off
AES-128	OPT	OPT	OPT	X	X	X	Power off	Power off
RNG	OPT	OPT	X	X	X	X	Power off	Power off

Table 23. Functional Table on Current Mode (continued)

IP	Main RUN (IDD1)	Main SLEEP (IDD2)	Sub RUN (IDD3)	Sub SLEEP (IDD4)	DEEP SLEEP Mode (ALLPWR = x)			
					0 (IDD5)	1 (IDD6)	2 (IDD7)	3 (IDD8)
HIRC	O	O	OPT	X	X	X	X	Power off
WDTRC	OPT	OPT	OPT	OPT	OPT	OPT	OPT	Power off
XMOSC	OPT	OPT	OPT	X	X	X	X	Power off
XSOSC	OPT	OPT	OPT	OPT	OPT	OPT	OPT	Power off

NOTES:

1. O: Enable, X: Must be disabled (Retention), Opt: Optional (A function can be disabled/enabled by s/w)
2. It can be woken up from SLEEP and DEEP SLEEP modes by an interrupt source of the optional peripherals.
3. The “MBOFF” means that the related peripherals are disabled by each control register bit and the clock must be disabled using SCU_PPCLKEN1/2 registers before entering deep sleep mode 1.
4. When wake-up in deep sleep mode 3 (shutdown), if the BOOT(PD5) pin is low level, it enters boot mode, so it must be set to high level to prevent unintentional entry into boot mode.

6.3 Wake-up time table

Table 24. Wake-up Time Table

Parameter	Symbol	Conditions	Typ	Max	Unit
Wake-up from main sleep	t_{WUMS}	HCLK=32MHz HIRC, Included stabilization	3	6	us
Wake-up from sub sleep	t_{WUSS}	HCLK=40kHz WDTRC	440	600	
Wake-up from deep sleep	t_{WUDS0}	HCLK=32MHz HIRC, Included stabilization PMU_PWRCR.ALLPWR=0	5	8	
	t_{WUDS1}	HCLK=32MHz HIRC PMU_PWRCR.ALLPWR=1	5	8	
	t_{WUDS2}	HCLK=32MHz HIRC, PMU_PWRCR.ALLPWR=2 A CPU reset will occur when wake-up.	5	8	
	t_{WUDS3}	HCLK=32MHz HIRC, PMU_PWRCR.ALLPWR=3 A system reset will occur when wake-up.	8.5	9.8	ms

NOTES:

1. A wake-up source will generate a CPU reset after maximum 8usec when the system is in DEEP SLEEP mode 2 with PMU_PWRCR.ALLPWR=0x2.
2. A wake-up source will generate a system reset after maximum 9.8ms when the system is in DEEP SLEEP mode 3 (Shutdown) with PMU_PWRCR.ALLPWR=0x3. This is called a WAKUP3 reset, and it resets the system in the same way as a POR reset. So, to prevent entering boot mode, care must be taken to ensure that the BOOT(PD5) pin is set to high level.

6.4 Registers

Base address and register map of PMU are shown in Table 25 and Table 26.

Table 25. Base Address of PMU

Name	Base address
PMU	0x4000_1900

Table 26. PMU Register Map

Name	Offset	Type	Description	Reset Value
PMU_PWRCR	0x0000	RW	Power Management control register	0x00000000
PMU_PWRSR	0x0008	RW	Power Management Status Register	0x0000000x
PMU_APUPDCR	0x0010	RW	Power Port A Pull-up/down Control Register	0x00000000
PMU_BPUPDCR	0x0014	RW	Power Port B Pull-up/down Control Register	0x00000000
PMU_CPUPDCR	0x0018	RW	Power Port C Pull-up/down Control Register	0x00000000
PMU_DPUPDCR	0x001C	RW	Power Port D Pull-up/down Control Register	0x00000000
PMU_EPUPDCR	0x0020	RW	Power Port E Pull-up/down Control Register	0x00000000
PMU_FPUPDCR	0x0024	RW	Power Port F Pull-up/down Control Register	0x00000000
PMU_BKR0 to 31	0x0040 to 0x005F	RW	Back-up registers 0 to 31	POR: Unknown Shutdown: Unknown Others: retained

6.4.1 PMU_PWRCR: power management control register

PMU_PWRCR register is used to manage power shut-off of system and peripherals except always-on region.

PMU_PWRCR register is 32-bit size and accessible in 32/16/8-bit.

PMU_PWRCR=0x4000_1900

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WTIDKY																**ALLPWR**		**Reserved**	**WKUP3**	**WKUP2**	**WKUP1**	**WKUP0**	**FLASHPWR**	**DFLASHPWR**	**Reserved**						
0x0000																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
WO																RW	RW	-	RW	RW	RW	RW	RW	RW	-	-	-	-	-	-	-

31 16	WTIDKY	Write Identification Key When writing, write 0x5A3C to these bits, or else writing is ignored
15 14	ALLPWR	All System and Peripheral Except Always-on Region Power Control 00 DEEP SLEEP mode 0, can be waked-up by optional and enabled peripherals 01 DEEP SLEEP mode 1, can be waked-up by optional peripherals and wakeup pins 10 DEEP SLEEP mode 2, can be waked-up by optional peripherals and wakeup pins 11 DEEP SLEEP mode 3 (Shutdown), can be waked-up by T60 and wakeup pins **NOTE**: These bits are automatically cleared to “00b” by wake-up signal.
9+x	WKUPx	Wake-up function control bit(when the ALLPWR bits are not “00b”), x: 0 to 3 WKUP[3:0]: [PE6:PE4:PA2:PA0] 0 Disable wake-up function from DEEP SLEEP mode. 1 Enable wake-up function from DEEP SLEEP mode **NOTES**: 1. The WKUPx pin should be configured as input mode by s/w. 2. The pin is forced in rising edge wake-up pin regardless of interrupt registers. 3. After wake-up, Check the real state of the pin.
8	FLASHPWR	Flash Memory Power Control 0 Power on 1 Power off **NOTE**: If this bit is set to “1b”, the Flash memory is immediately turned off. The bit can be written only during the code is operating in SRAM. The Flash memory has 2usec power-on time. The s/w should wait at least 2usec before Flash memory access after writing “0b” to the bit for power-up Flash memory.
7	DFLASHPWR	Data Flash Memory Power Control 0 Power on 1 Power off **NOTE**: The Data Flash memory has 2usec power-on time. The s/w should wait at least 2usec before Data Flash memory access after writing “0b” to the bit for power-up Data Flash memory.
6 0	-	Reserved. This value should be set to 0. Otherwise, a malfunction may occur.

NOTE: There is thing to watch out for before entering a deep sleep mode 1. Refer to “functional table on current mode” in the PMU block.

6.4.2 PMU_PWRSR: power management status register

PMU_PWRSR register is 32-bit size and accessible in 32/16/8-bit.

PMU_PWRSR=0x4000_1908

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13	12	11 10 9	8	7 6 5 4	3	2	1 0
Reserved	**Reserved**	**DS1WKIFLAG**	**Reserved**	**DS1WKIEN**	**Reserved**	**WAKUPST3**	**WAKUPST2**	**Reserved**
0x0000	**000**	**0**	**000**	**0**	**0000**	**x**	**x**	**00**
–	**–**	**RW**	**–**	**RW**	**–**	**RW**	**RW**	**–**

12	DS1WKIFLAG	Deep Sleep Mode 1 Wakeup Interrupt Flag 0 No request occurred 1 Request occurred by wakeup pins, This bit is cleared to ‘0’ when write ‘1’.
8	DS1WKIEN	Deep Sleep Mode 1 Wakeup Interrupt Enable by wakeup pins 0 Disable deep sleep mode 1 wakeup interrupt 1 Enable deep sleep mode 1 wakeup interrupt
3	WAKUPST3	Wake-up Reset Status 3 bit. A system reset will occur and this bit is set when the system wakes up from DEEP SLEEP mode 3 (PMU_PWRCR.ALLPWR = 3, Shutdown). 0 Not detected. 1 Wake-up reset is detected, This bit is cleared to ‘0’ when write ‘1’.
2	WAKUPST2	Wake-up Reset Status 2 bit. A core reset will occur and this bit is set when the system wakes up from DEEP SLEEP mode 2 (PMU_PWRCR.ALLPWR = 2). 0 Not detected. 1 Wake-up reset is detected, This bit is cleared to ‘0’ when write ‘1’.

NOTE: The WAKUPST2 and WAKUPST3 bit is cleared to “0b” by a power-on reset but are retained at the other reset.

6.4.3 PMU_APUPDCR: power port A pull-up/down control register

PMU_APUPDCR register is 32-bit size and accessible in 32/16/8-bit.

PMU_APUPDCR=0x4000_1910

31 30	29 28	27 26	25 24	23 22	21 20	19 18	17 16	15 14	13 12	11 10	9 8	7 6	5 4	3 2	1 0
APUPD15	APUPD14	APUPD13	APUPD12	APUPD11	APUPD10	APUPD9	APUPD8	APUPD7	APUPD6	APUPD5	APUPD4	APUPD3	APUPD2	APUPD1	APUPD0
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

2x+1 2x	APUPDx	PA[x] Pull-up/down control bits (when the ALLPWR bits are “11b”), x: 0 to 15	
		00	Disable the PA[x]’s pull-up/down resistors
		01	Enable the PA[x]’s pull-up resistor
		10	Enable the PA[x]’s pull-down resistor
		11	Reserved

NOTE: The PMU_APUPDCR register may be used to not make a floating state when DEEP SLEEP mode 3.

6.4.4 PMU_BPUPDCR: power port B pull-up/down control register

PMU_BPUPDCR register is 32-bit size and accessible in 32/16/8-bit.

PMU_BPUPDCR=0x4000_1914

31	30	29	28	27	26	25 24	23 22	21 20	19 18	17 16	15 14	13 12	11 10	9 8	7 6	5 4	3 2	1 0
Reserved						BPUPD12	BPUPD11	BPUPD10	BPUPD9	BPUPD8	BPUPD7	BPUPD6	BPUPD5	BPUPD4	BPUPD3	BPUPD2	BPUPD1	BPUPD0
0	0	0	0	0	0	00	00	00	00	00	00	00	00	00	00	00	00	00
I	I	I	I	I	I	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

2x+1 2x	BPUPDx	PB[x] Pull-up/down control bits (when the ALLPWR bits are “11b”), x: 0 to 12	
		00	Disable the PB[x]’s pull-up/down resistors
		01	Enable the PB[x]’s pull-up resistor
		10	Enable the PB[x]’s pull-down resistor
		11	Reserved

NOTE: The PMU_BPUPDCR register may be used to not make a floating state when DEEP SLEEP mode 3.

6.4.5 PMU_CPUPDCR: power port C pull-up/down control register

PMU_CPUPDCR register is 32-bit size and accessible in 32/16/8-bit.

PMU_CPUPDCR=0x4000_1918

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				CPUPD13		CPUPD12		CPUPD11		CPUPD10		CPUPD9		CPUPD8		CPUPD7		CPUPD6		CPUPD5		CPUPD4		CPUPD3		CPUPD2		CPUPD1		CPUPD0	
0	0	0	0	00		00		00		00		00		00		00		00		00		00		00		00		00		00	
-	-	-	-	RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW	

2x+1 2x	CPUPDx	PC[x] Pull-up/down control bits (when the ALLPWR bits are “11b”), x: 0 to 13	
		00	Disable the PC[x]’s pull-up/down resistors
		01	Enable the PC[x]’s pull-up resistor
		10	Enable the PC[x]’s pull-down resistor
		11	Reserved

NOTE: The PMU_CPUPDCR register may be used to not make a floating state when DEEP SLEEP mode 3.

6.4.6 PMU_DPUPDCR: power port D pull-up/down control register

PMU_DPUPDCR register is 32-bit size and accessible in 32/16/8-bit.

PMU_DPUPDCR=0x4000_191C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								DPUPD11		DPUPD10		DPUPD9		DPUPD8		DPUPD7		DPUPD6		DPUPD5		DPUPD4		DPUPD3		DPUPD2		DPUPD1		DPUPD0	
0	0	0	0	0	0	0	0	00		00		00		00		00		00		00		00		00		00		00		00	
-	-	-	-	-	-	-	-	RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW	

2x+1 2x	DPUPDx	PD[x] Pull-up/down control bits (when the ALLPWR bits are “11b”), x: 0 to 11	
		00	Disable the PD[x]’s pull-up/down resistors
		01	Enable the PD[x]’s pull-up resistor
		10	Enable the PD[x]’s pull-down resistor
		11	Reserved

NOTE: The PMU_DPUPDCR register may be used to not make a floating state when DEEP SLEEP mode 3.

6.4.7 PMU_EPUPDCR: power port E pull-up/down control register

PMU_EPUPDCR register is 32-bit size and accessible in 32/16/8-bit.

PMU_EPUPDCR=0x4000_1920

31	30	29	28	27	26	25	24	23	22	21	20	19 18	17 16	15 14	13 12	11 10	9 8	7 6	5 4	3 2	1 0
Reserved												**EPUPD9**	**EPUPD8**	**EPUPD7**	**EPUPD6**	**EPUPD5**	**EPUPD4**	**EPUPD3**	**EPUPD2**	**EPUPD1**	**EPUPD0**
0	0	0	0	0	0	0	0	0	0	0	0	00	00	00	00	00	00	00	00	00	00
ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

2x+1 2x	EPUPDx	PE[x] Pull-up/down control bits (when the ALLPWR bits are "11b"), x: 0 to 9
		00 Disable the PE[x]'s pull-up/down resistors
		01 Enable the PE[x]'s pull-up resistor
		10 Enable the PE[x]'s pull-down resistor
		11 Reserved

NOTE: The PMU_EPUPDCR register may be used to not make a floating state when DEEP SLEEP mode 3.

6.4.8 PMU_FPUPDCR: power port F pull-up/down control register

PMU_FPUPDCR register is 32-bit size and accessible in 32/16/8-bit.

PMU_FPUPDCR=0x4000_1924

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15 14	13 12	11 10	9 8	7 6	5 4	3 2	1 0
Reserved																**FPUPD7**	**FPUPD6**	**FPUPD5**	**FPUPD4**	**FPUPD3**	**FPUPD2**	**FPUPD1**	**FPUPD0**
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	00	00	00	00	00	00	00	00
ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	RW	RW	RW	RW	RW	RW	RW	RW

2x+1 2x	FPUPDx	PF[x] Pull-up/down control bits (when the ALLPWR bits are "11b"), x: 0 to 7
		00 Disable the PF[x]'s pull-up/down resistors
		01 Enable the PF[x]'s pull-up resistor
		10 Enable the PF[x]'s pull-down resistor
		11 Reserved

NOTE: The PMU_FPUPDCR register may be used to not make a floating state when DEEP SLEEP mode 3.

6.4.9 PMU_BKRx: back-up register x (x = 0 to 31)

PMU_BKRx register is 8-bit size and accessible in 32/16/8-bit (x = 0 to 31).

PMU_BKRx=0x4000_1940 to 0x4000_195F

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="32">BACKUP</td></tr>
<tr><td colspan="32">Unknown by POR/shutdown mode, and Retained by others</td></tr>
<tr><td colspan="32">RW</td></tr>
</table>

31 0	BACKUP	Back-up Data bits. This register is used for data back-up on power off mode except of shutdown mode.

7 Port Control Unit (PCU) and GPIO

Port Control Unit (PCU) configures and controls external I/Os as shown below:

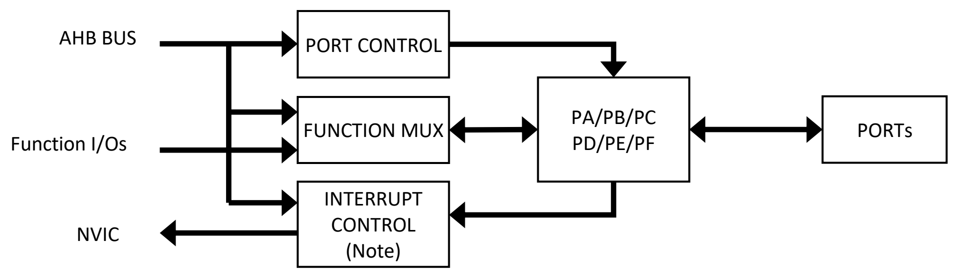
- It configures direction of an external signal of each pin.
- It sets Interrupt trigger mode for each pin.
- The PCU sets internal pull-up/down register control and open drain control.

Most pins, except for dedicated function pins, can be used as General Purpose I/O (GPIO) ports. GPIO block controls the GPIO as shown below:

- Output signal level (H/L) select
- External interrupt interface
- Pull-up/down enable or disable

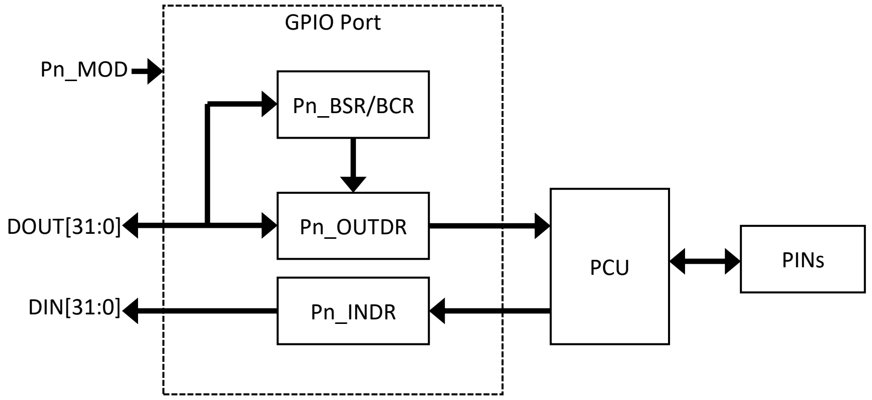
7.1 PCU and GPIO block diagrams

Figure 28. PCU Block Diagram



NOTE: Refer to the section3.2 Interrupt controller.

Figure 29. GPIO Block Diagram



7.2 I/O port block diagram

Figure 30. I/O Port Block Diagram (General Purpose I/O Pins)

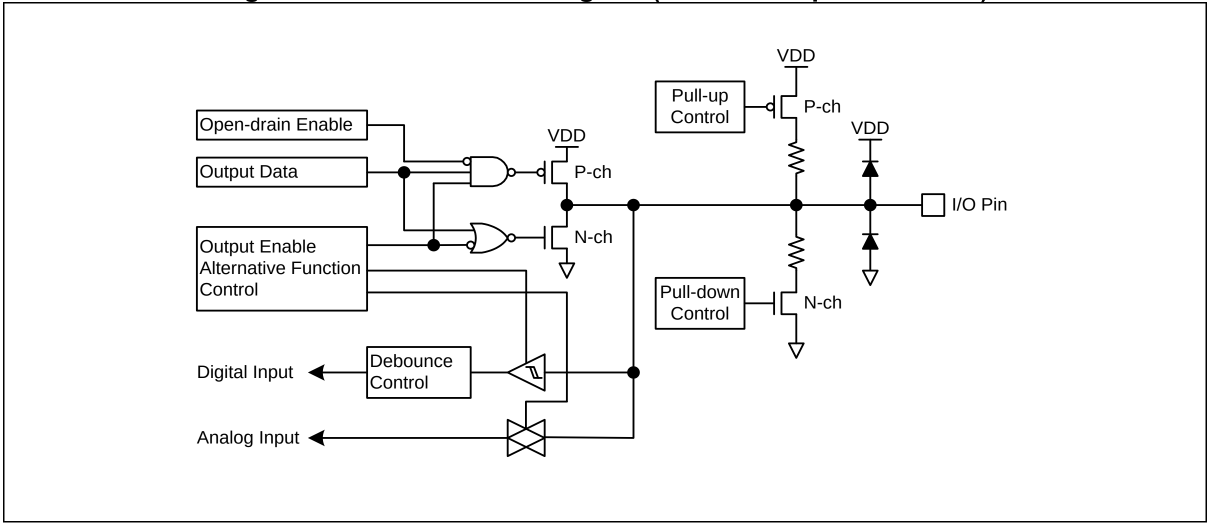
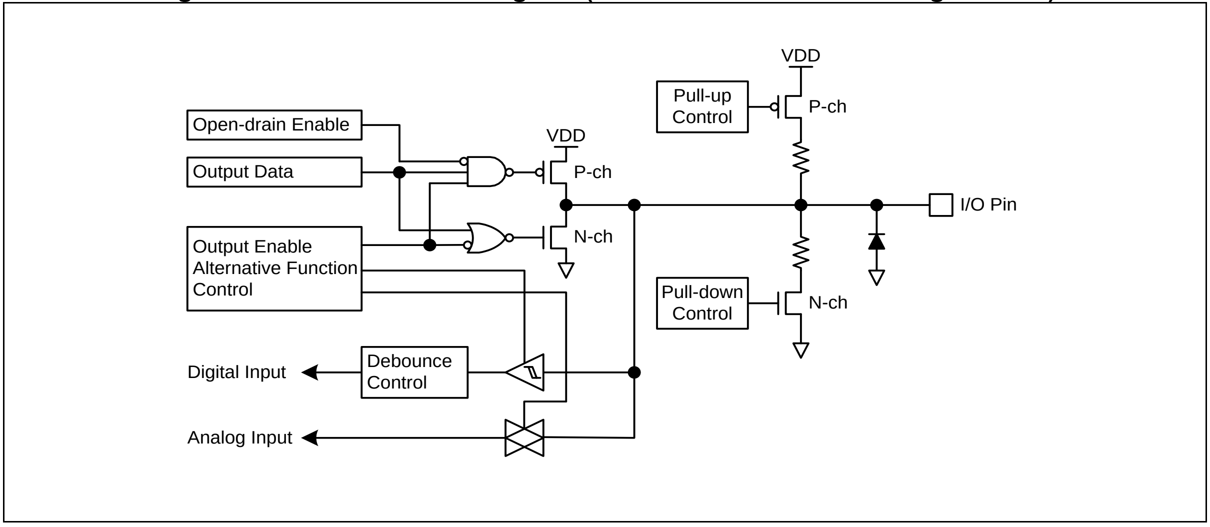


Figure 31. I/O Port Block Diagram (5V Tolerant I/O and LCD Signal Pins)



7.3 Pin multiplexing

GPIO pins support alternative functions. Table 27 shows pin multiplexing information.

Table 27. GPIO Alternative Functions

Port	Pin	Function								
		AF0	AF1	AF2	AF3	AF4	AF5	AF6	AF7	AF8
PA	0	T40OUTA	T40INP	–	–	AN0	CP0N0	CP0OUT	SEG29	–
	1	T40OUTB	EC40	T42INP	SS10	AN1	CP0P0	–	SEG30	–
	2	T41OUTA	T41INP	TXD10	MOSI10	AN2	CP1N0	CP1OUT	SEG31	LPTXD0
	3	T41OUTB	EC41	RXD10	MISO10	AN3	CP1P0	ADTRG	SEG32	LPRXD0
	4	–	T43INP	SS1	SCK10	AN4	CP0N1	CP1N1	SEG33	LPDE0
	5	T40OUTA	T40INP	–	SCK1	AN5	CP0N2	CP1N2	SEG34	–
	6	T43OUTA	T43INP	–	MISO1	AN6	–	CP0OUT	SEG35	–
	7	T43OUTB	EC43	–	MOSI1	AN7	CP1P1	CP1OUT	SEG36	–
	8	–	–	LPTXD0	–	AN8	–	–	SEG37	–
	9	–	–	LPRXD0	–	AN9	–	–	SEG38	–
	10	–	–	–	SCK11	AN10	–	–	SEG39	–
	11	–	–	–	SS11	AN11	–	–	SEG40	–
	12	T40OUTA	T40INP	RXD11	MISO11	–	–	–	SEG41	–
	13	T40OUTB	EC40	TXD11	MOSI11	–	–	–	SEG42	–
	14	–	–	–	–	SCL1	–	–	SEG43	–
	15	–	–	–	–	SDA1	–	–	SEG44	–
PB	0	T11OUT	T11CAP	–	–	–	–	–	CAPL	–
	1	–	EC11	–	–	–	LPDE0	–	CAPH	–
	2	T50OUT	–	–	–	–	–	–	VLC3	–
	3	T41OUTA	T41INP	LPTXD1	–	SCL1	–	ADTRG	VLC2	–
	4	T41OUTB	EC41	LPRXD1	–	SDA1	–	ADTRG	VLC1	–
	5	–	–	–	SS0	–	LPDE1	–	VLC0	–
	6	T42OUTA	T42INP	CLKO	SCK0	SCL1	–	–	SEG10	–
	7	T42OUTB	EC42	RTCOUT	MISO0	SDA1	LPDE1	ADTRG	SEG9	–
	8	–	–	–	MOSI0	–	–	–	SEG8	–
	9	T43OUTA	T43INP	T10CAP	SS3	–	–	T10OUT	SEG3	–
	10	T43OUTB	EC43	EC10	SCK3	–	–	–	SEG2	–
	11	T43INP	–	T11CAP	MISO3	–	–	T11OUT	SEG1	–
	12	T42INP	–	EC11	MOSI3	–	SDA2	–	SEG0	–

Table 27. GPIO Alternative Functions (continued)

Port	Pin	Function							
		AF0	**AF1**	**AF2**	**AF3**	**AF4**	**AF5**	**AF6**	**AF7**
PC	0	CLKO	T12CAP	–	–	SC0IN	SCL2	T12OUT	COM0
	1	CLKO	EC12	TXD0	–	SC0PWR	SCL0	–	COM1
	2	–	–	RXD0	SCK1	SC0CLK	SDA0	–	COM2/SEG11
	3	–	–	–	MISO1	SC0RST	SC0RXD	CP0OUT	COM3/SEG12
	4	–	–	–	MOSI1	SC0DATA	SC0TXD	CP1OUT	SEG13
	5	SWDIO	–	LPRXD1	–	–	–	–	–
	6	SWCLK	–	LPTXD1	–	–	–	–	–
	7	T40OUTA	T40INP	–	–	–	–	SS1	SEG14
	8	–	–	TXD10	MOSI10	SC0DATA	SC0TXD	–	SEG15
	9	–	–	RXD10	MISO10	SC0RST	SC0RXD	–	SEG16
	10	–	–	LPTXD1	SCK10	SC0CLK	–	–	COM4/SEG17
	11	–	–	LPRXD1	SS10	SC0PWR	–	–	COM5/SEG18
	12	–	–	TXD11	MOSI11	SC0IN	–	–	COM6/SEG19
	13	–	–	RXD11	MISO11	–	LPDE1	–	COM7/SEG20
PD	0	T40OUTB	EC40	SCL2	SCK1	MOSI11	CP1N3	–	TXD11
	1	T43OUTA	T43INP	SDA2	MISO1	MISO11	–	CP1P2	RXD11
	2	T43OUTB	EC43	EC50	MOSI1	SCK11	–	CP1P3	–
	3	–	–	TXD0	–	SS11	SCL0	CP1P4	–
	4	–	T50INP	RXD0	–	–	SDA0	CP1P5	–
	5	BOOT	–	RTCOUT	–	–	–	–	–
	6	T13OUT	T13CAP	–	–	SCL0	–	–	–
	7	–	EC13	–	SS0	SDA0	–	–	–
	8	T42OUTA	T42INP	–	SS0	–	–	–	SEG21
	9	T42OUTB	EC42	–	SCK0	–	–	–	SEG22
	10	T13OUT	T13CAP	–	MISO0	SCK11	–	–	SEG23
	11	–	EC13	–	MOSI0	SS11	–	–	SEG24

Table 27. GPIO Alternative Functions (continued)

Port	Pin	Function							
		AF0	AF1	AF2	AF3	AF4	AF5	AF6	AF7
PE	0	SXIN	–	–	–	–	–	–	–
	1	SXOUT	–	–	–	–	–	–	–
	2	XIN	–	–	MOSI2	–	–	–	–
	3	XOUT	–	–	MISO2	–	–	–	–
	4	RTCOUT	–	–	SCK2	–	–	–	–
	5	–	–	–	SS2	–	–	–	–
	6	–	EC11	–	MOSI2	–	–	ADTRG	–
	7	T11OUT	T11CAP	–	MISO2	–	–	–	–
	8	–	EC10	–	SCK2	–	–	–	–
	9	T10OUT	T10CAP	–	SS2	–	–	–	–
PF	0	–	EC50	LPRXD0	–	AN12	SCL2	–	SEG25
	1	T50OUT	–	LPTXD0	SCK0	AN13	SDA2	–	SEG26
	2	–	T50INP	–	MISO0	AN14	LPDE0	–	SEG27
	3	–	–	–	MOSI0	AN15	–	–	SEG28
	4	T12OUT	T12CAP	–	MOSI3	–	SCL2	–	SEG4
	5	–	EC12	–	MISO3	–	SDA2	–	SEG5
	6	T13OUT	T13CAP	–	SCK3	–	–	–	SEG6
	7	–	EC13	–	SS3	–	LPDE1	–	SEG7

NOTES:

1. The PC[13:10] and PC[3:2] are automatically configured as common or segment signal according to the duty of the LCD control register when the pins are selected as alternative functions for common/segment.
2. The SWCLK and SWDIO pins shouldn't be changed as other alternative functions by software during the pins are connected with debugger host.
3. The VDD should be greater than or equal to 2.0V if CP1N3, CP1P2, CP1P3, CP1P4, and CP1P5 are used as comparator pins for alternative function.

7.4 Registers

Base address and register map of PCU and GPIO block are shown in Table 28 and Table 29.

Table 28. Base Address of Port

Port name	Address range	Size (bytes)	Description
PA	0x3000 0000 – 0x3000 00FF	256	General Purpose I/O Port A
PB	0x3000 0100 – 0x3000 01FF	256	General Purpose I/O Port B
PC	0x3000 0200 – 0x3000 02FF	256	General Purpose I/O Port C
PD	0x3000 0300 – 0x3000 03FF	256	General Purpose I/O Port D
PE	0x3000 0400 – 0x3000 04FF	256	General Purpose I/O Port E
PF	0x3000 0500 – 0x3000 05FF	256	General Purpose I/O Port F

Table 29. PCU and GPIO Register Map

Name	Offset	Type	Description	Reset Value
Pn_MOD	0x0000	RW	Port n mode register	0xFFFFFFFF
Pn_TYP	0x0004	RW	Port n output type selection register	0x00000000
Pn_AFSR1	0x0008	RW	Port n alternative function selection register 1	0x00000000
Pn_AFSR2	0x000C	RW	Port n alternative function selection register 2	0x00000000
Pn_PUPD	0x0010	RW	Port n pull-up/down resistor selection register	0x00000000
Pn_INDR	0x0014	RO	Port n input data register	0x0000FFFF
Pn_OUTDR	0x0018	RW	Port n output data register	0x00000000
Pn_BSR	0x001C	WO	Port n output bit set register	0x00000000
Pn_BCR	0x0020	WO	Port n output bit clear register	0x00000000
Pn_OUTDMSK	0x0024	RW	Port n output data mask register	0x00000000
Pn_DBCR	0x0028	RW	Port n debounce control register	0x00000000

NOTE: Where n=A, B, C, D, E, and F.

7.4.1 Pn_MOD: port n mode register

Pn_MOD register selects one from input mode and output mode for each port pin. Each pin can be configured as an input pin, an output pin or an Alternative Function pin.

This register is 32-bit size and accessible in 32/16/8-bit. (n = A to F).

PA_MOD=0x3000_0000, PB_MOD=0x3000_0100, PC_MOD=0x3000_0200
PD_MOD=0x3000_0300, PE_MOD=0x3000_0400, PF_MOD=0x3000_0500

31 30	29 28	27 26	25 24	23 22	21 20	19 18	17 16	15 14	13 12	11 10	9 8	7 6	5 4	3 2	1 0
MODE15	MODE14	MODE13	MODE12	MODE11	MODE10	MODE9	MODE8	MODE7	MODE6	MODE5	MODE4	MODE3	MODE2	MODE1	MODE0
11	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

2x+1 2x	MODEx	Port n Mode Selection bits, x: 0 to 15	
		00	Input mode
		01	Output mode
		10	Alternative function mode
		11	Off mode (Both input and output are disabled, and the input value is “1b”)

NOTES:

1. The MODEx bits for PE0 – PE3 won’t be changed during the corresponding clock (XMOSC/XSOSC) is selected as the system clock (MCLK).
2. The MODEx bits for PE[1:0] are set to “11b” by the reset of POR/WAKUP3 but retained by the other reset.
3. The MODEx bits for PC[6:5] and PD5 are set to “10b” for alternative function by reset.
4. PC5: SWDIO, PC6: SWCLK, PD5: BOOT
5. Exceptionally, the MODEx bits for PA[7:6] are set to “00b” for input by a system reset.

7.4.2 Pn_TYP: port n output type selection register

Pn_TYP register selects an output type of a port pin from Push-pull output and Open-drain output.

This register is 32-bit size and accessible in 32/16/8-bit. (n = A to F)

PA_TYP=0x3000_0004, PB_TYP=0x3000_0104, PC_TYP=0x3000_0204
PD_TYP=0x3000_0304, PE_TYP=0x3000_0404, PF_TYP=0x3000_0504

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	TYP15	TYP14	TYP13	TYP12	TYP11	TYP10	TYP9	TYP8	TYP7	TYP6	TYP5	TYP4	TYP3	TYP2	TYP1	TYP0
0x0000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
–	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

x	TYPx	Port n Output Type Selection bit, x: 0 to 15	
		0	Push-pull output
		1	Open-drain output

7.4.3 PA_AFSR1/2: port A alternative function selection register 1/2

PA_AFSR1/2 register must be set properly before using the port. Otherwise, the port may not function properly.

This register is 32-bit size and accessible in 32/16/8-bit.

PA_AFSR1=0x3000_0008

31 30 29 28	27 26 25 24	23 22 21 20	19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0
AFSR7	AFSR6	AFSR5	AFSR4	AFSR3	AFSR2	AFSR1	AFSR0
0000	0000	0000	0000	0000	0000	0000	0000
RW	RW	RW	RW	RW	RW	RW	RW

4x+3 4x	AFSRx	Port A Alternative Function Selection bits, x: 0 to 7	
		0000	Alternative Function 0 (AF0)
		0001	Alternative Function 1 (AF1)
		0010	Alternative Function 2 (AF2)
		0011	Alternative Function 3 (AF3)
		0100	Alternative Function 4 (AF4)
		0101	Alternative Function 5 (AF5)
		0110	Alternative Function 6 (AF6)
		0111	Alternative Function 7 (AF7)
		1000	Alternative Function 8 (AF8)
		Others	Reserved

PA_AFSR2=0x3000_000C

31 30 29 28	27 26 25 24	23 22 21 20	19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0
AFSR15	AFSR14	AFSR13	AFSR12	AFSR11	AFSR10	AFSR9	AFSR8
0000	0000	0000	0000	0000	0000	0000	0000
RW	RW	RW	RW	RW	RW	RW	RW

4(x-8)+3 4(x-8)	AFSRx	Port A Alternative Function Selection bits, x: 8 to 15	
		0000	Alternative Function 0 (AF0)
		0001	Alternative Function 1 (AF1)
		0010	Alternative Function 2 (AF2)
		0011	Alternative Function 3 (AF3)
		0100	Alternative Function 4 (AF4)
		0101	Alternative Function 5 (AF5)
		0110	Alternative Function 6 (AF6)
		0111	Alternative Function 7 (AF7)
		1000	Alternative Function 8 (AF8)
		Others	Reserved

Table 30. Functions of PA Port

Port	Pin	Function								
		AF0	**AF1**	**AF2**	**AF3**	**AF4**	**AF5**	**AF6**	**AF7**	**AF8**
PA	0	T40OUTA	T40INP	–	–	AN0	CP0N0	CP0OUT	SEG29	–
	1	T40OUTB	EC40	T42INP	SS10	AN1	CP0P0	–	SEG30	–
	2	T41OUTA	T41INP	TXD10	MOSI10	AN2	CP1N0	CP1OUT	SEG31	LPTXD0
	3	T41OUTB	EC41	RXD10	MISO10	AN3	CP1P0	ADTRG	SEG32	LPRXD0
	4	–	T43INP	SS1	SCK10	AN4	CP0N1	CP1N1	SEG33	LPDE0
	5	T40OUTA	T40INP	–	SCK1	AN5	CP0N2	CP1N2	SEG34	–
	6	T43OUTA	T43INP	–	MISO1	AN6	–	CP0OUT	SEG35	–
	7	T43OUTB	EC43	–	MOSI1	AN7	CP1P1	CP1OUT	SEG36	–
	8	–	–	LPTXD0	–	AN8	–	–	SEG37	–
	9	–	–	LPRXD0	–	AN9	–	–	SEG38	–
	10	–	–	–	SCK11	AN10	–	–	SEG39	–
	11	–	–	–	SS11	AN11	–	–	SEG40	–
	12	T40OUTA	T40INP	RXD11	MISO11	–	–	–	SEG41	–
	13	T40OUTB	EC40	TXD11	MOSI11	–	–	–	SEG42	–
	14	–	–	–	–	SCL1	–	–	SEG43	–
	15	–	–	–	–	SDA1	–	–	SEG44	–

7.4.4 PB_AFSR1/2: port B alternative function selection register 1/2

PB_AFSR1/2 register must be set properly before using the port. Otherwise, the port may not function properly.

This register is 32-bit size and accessible in 32/16/8-bit.

PB_AFSR1=0x3000_0108

31 30 29 28	27 26 25 24	23 22 21 20	19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0
AFSR7	AFSR6	AFSR5	AFSR4	AFSR3	AFSR2	AFSR1	AFSR0
0000	0000	0000	0000	0000	0000	0000	0000
RW	RW	RW	RW	RW	RW	RW	RW

4x+3 4x	AFSRx	Port B Alternative Function Selection bits, x: 0 to 7	
		0000	Alternative Function 0 (AF0)
		0001	Alternative Function 1 (AF1)
		0010	Alternative Function 2 (AF2)
		0011	Alternative Function 3 (AF3)
		0100	Alternative Function 4 (AF4)
		0101	Alternative Function 5 (AF5)
		0110	Alternative Function 6 (AF6)
		0111	Alternative Function 7 (AF7)
		Others	Reserved

PB_AFSR2=0x3000_010C

31 30 29 28	27 26 25 24	23 22 21 20	19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0
AFSR15	AFSR14	AFSR13	AFSR12	AFSR11	AFSR10	AFSR9	AFSR8
0000	0000	0000	0000	0000	0000	0000	0000
RW	RW	RW	RW	RW	RW	RW	RW

4(x-8)+3 4(x-8)	AFSRx	Port B Alternative Function Selection bits, x: 8 to 15	
		0000	Alternative Function 0 (AF0)
		0001	Alternative Function 1 (AF1)
		0010	Alternative Function 2 (AF2)
		0011	Alternative Function 3 (AF3)
		0100	Alternative Function 4 (AF4)
		0101	Alternative Function 5 (AF5)
		0110	Alternative Function 6 (AF6)
		0111	Alternative Function 7 (AF7)
		Others	Reserved

Table 31. Functions of PB Port

Port	Pin	Function							
		AF0	**AF1**	**AF2**	**AF3**	**AF4**	**AF5**	**AF6**	**AF7**
PB	0	T11OUT	T11CAP	–	–	–	–	–	CAPL
	1	–	EC11	–	–	–	LPDE0	–	CAPH
	2	T50OUT	–	–	–	–	–	–	VLC3
	3	T41OUTA	T41INP	LPTXD1	–	SCL1	–	ADTRG	VLC2
	4	T41OUTB	EC41	LPRXD1	–	SDA1	–	ADTRG	VLC1
	5	–	–	–	SS0	–	LPDE1	–	VLC0
	6	T42OUTA	T42INP	CLKO	SCK0	SCL1	–	–	SEG10
	7	T42OUTB	EC42	RTCOUT	MISO0	SDA1	LPDE1	ADTRG	SEG9
	8	–	–	–	MOSI0	–	–	–	SEG8
	9	T43OUTA	T43INP	T10CAP	SS3	–	–	T10OUT	SEG3
	10	T43OUTB	EC43	EC10	SCK3	–	–	–	SEG2
	11	T43INP	–	T11CAP	MISO3	–	–	T11OUT	SEG1
	12	T42INP	–	EC11	MOSI3	–	SDA2	–	SEG0

7.4.5 PC_AFSR1/2: port C alternative function selection register 1/2

PC_AFSR1/2 register must be set properly before using the port. Otherwise, the port may not function properly.

This register is 32-bit size and accessible in 32/16/8-bit.

PC_AFSR1=0x3000_0208

31 30 29 28	27 26 25 24	23 22 21 20	19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0
AFSR7	**AFSR6**	**AFSR5**	**AFSR4**	**AFSR3**	**AFSR2**	**AFSR1**	**AFSR0**
0000	**0000**	**0000**	**0000**	**0000**	**0000**	**0000**	**0000**
RW	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**

4x+3 4x	AFSRx	Port C Alternative Function Selection bits, x: 0 to 7	
		0000	Alternative Function 0 (AF0)
		0001	Alternative Function 1 (AF1)
		0010	Alternative Function 2 (AF2)
		0011	Alternative Function 3 (AF3)
		0100	Alternative Function 4 (AF4)
		0101	Alternative Function 5 (AF5)
		0110	Alternative Function 6 (AF6)
		0111	Alternative Function 7 (AF7)
		Others	Reserved

PC_AFSR2=0x3000_020C

31 30 29 28	27 26 25 24	23 22 21 20	19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0
AFSR15	**AFSR14**	**AFSR13**	**AFSR12**	**AFSR11**	**AFSR10**	**AFSR9**	**AFSR8**
0000	**0000**	**0000**	**0000**	**0000**	**0000**	**0000**	**0000**
RW	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**

4(x-8)+3 4(x-8)	AFSRx	Port C Alternative Function Selection bits, x: 8 to 15	
		0000	Alternative Function 0 (AF0)
		0001	Alternative Function 1 (AF1)
		0010	Alternative Function 2 (AF2)
		0011	Alternative Function 3 (AF3)
		0100	Alternative Function 4 (AF4)
		0101	Alternative Function 5 (AF5)
		0110	Alternative Function 6 (AF6)
		0111	Alternative Function 7 (AF7)
		Others	Reserved

Table 32. Functions of PC Port

Port	Pin	Function							
		AF0	AF1	AF2	AF3	AF4	AF5	AF6	AF7
PC	0	CLKO	T12CAP	–	–	SC0IN	SCL2	T12OUT	COM0
	1	CLKO	EC12	TXD0	–	SC0PWR	SCL0	–	COM1
	2	–	–	RXD0	SCK1	SC0CLK	SDA0	–	COM2/SEG11
	3	–	–	–	MISO1	SC0RST	SC0RXD	CP0OUT	COM3/SEG12
	4	–	–	–	MOSI1	SC0DATA	SC0TXD	CP1OUT	SEG13
	5	SWDIO	–	LPRXD1	–	–	–	–	–
	6	SWCLK	–	LPTXD1	–	–	–	–	–
	7	T40OUTA	T40INP	–	–	–	–	SS1	SEG14
	8	–	–	TXD10	MOSI10	SC0DATA	SC0TXD	–	SEG15
	9	–	–	RXD10	MISO10	SC0RST	SC0RXD	–	SEG16
	10	–	–	LPTXD1	SCK10	SC0CLK	–	–	COM4/SEG17
	11	–	–	LPRXD1	SS10	SC0PWR	–	–	COM5/SEG18
	12	–	–	TXD11	MOSI11	SC0IN	–	–	COM6/SEG19
	13	–	–	RXD11	MISO11	–	LPDE1	–	COM7/SEG20

7.4.6 PD_AFSR1/2: port D alternative function selection register 1/2

PD_AFSR1/2 register must be set properly before using the port. Otherwise, the port may not function properly.

This register is 32-bit size and accessible in 32/16/8-bit.

PD_AFSR1=0x3000_0308

31 30 29 28	27 26 25 24	23 22 21 20	19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0
AFSR7	**AFSR6**	**AFSR5**	**AFSR4**	**AFSR3**	**AFSR2**	**AFSR1**	**AFSR0**
0000	**0000**	**0000**	**0000**	**0000**	**0000**	**0000**	**0000**
RW	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**

4x+3 4x	AFSRx	Port D Alternative Function Selection bits, x: 0 to 7	
		0000	Alternative Function 0 (AF0)
		0001	Alternative Function 1 (AF1)
		0010	Alternative Function 2 (AF2)
		0011	Alternative Function 3 (AF3)
		0100	Alternative Function 4 (AF4)
		0101	Alternative Function 5 (AF5)
		0110	Alternative Function 6 (AF6)
		0111	Alternative Function 7 (AF7)
		Others	Reserved

PD_AFSR2=0x3000_030C

31 30 29 28	27 26 25 24	23 22 21 20	19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0
AFSR15	**AFSR14**	**AFSR13**	**AFSR12**	**AFSR11**	**AFSR10**	**AFSR9**	**AFSR8**
0000	**0000**	**0000**	**0000**	**0000**	**0000**	**0000**	**0000**
RW	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**

4(x-8)+3 4(x-8)	AFSRx	Port D Alternative Function Selection bits, x: 8 to 15	
		0000	Alternative Function 0 (AF0)
		0001	Alternative Function 1 (AF1)
		0010	Alternative Function 2 (AF2)
		0011	Alternative Function 3 (AF3)
		0100	Alternative Function 4 (AF4)
		0101	Alternative Function 5 (AF5)
		0110	Alternative Function 6 (AF6)
		0111	Alternative Function 7 (AF7)
		Others	Reserved

Table 33. Functions of PD Port

Port	Pin	Function							
		AF0	AF1	AF2	AF3	AF4	AF5	AF6	AF7
PD	0	T40OUTB	EC40	SCL2	SCK1	MOSI11	CP1N3	–	TXD11
	1	T43OUTA	T43INP	SDA2	MISO1	MISO11	–	CP1P2	RXD11
	2	T43OUTB	EC43	EC50	MOSI1	SCK11	–	CP1P3	–
	3	–	–	TXD0	–	SS11	SCL0	CP1P4	–
	4	–	T50INP	RXD0	–	–	SDA0	CP1P5	–
	5	BOOT	–	RTCOUT	–	–	–	–	–
	6	T13OUT	T13CAP	–	–	SCL0	–	–	–
	7	–	EC13	–	SS0	SDA0	–	–	–
	8	T42OUTA	T42INP	–	SS0	–	–	–	SEG21
	9	T42OUTB	EC42	–	SCK0	–	–	–	SEG22
	10	T13OUT	T13CAP	–	MISO0	SCK11	–	–	SEG23
	11	–	EC13	–	MOSI0	SS11	–	–	SEG24

7.4.7 PE_AFSR1/2: port E alternative function selection register 1/2

PE_AFSR1/2 register must be set properly before using the port. Otherwise, the port may not function properly.

This register is 32-bit size and accessible in 32/16/8-bit.

PE_AFSR1=0x3000_0408

31 30 29 28	27 26 25 24	23 22 21 20	19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0
AFSR7	AFSR6	AFSR5	AFSR4	AFSR3	AFSR2	AFSR1	AFSR0
0000	0000	0000	0000	0000	0000	0000	0000
RW	RW	RW	RW	RW	RW	RW	RW

4x+3 4x	AFSRx	Port E Alternative Function Selection bits, x: 0 to 7	
		0000	Alternative Function 0 (AF0)
		0001	Alternative Function 1 (AF1)
		0010	Alternative Function 2 (AF2)
		0011	Alternative Function 3 (AF3)
		0100	Alternative Function 4 (AF4)
		0101	Alternative Function 5 (AF5)
		0110	Alternative Function 6 (AF6)
		0111	Alternative Function 7 (AF7)
		Others	Reserved

NOTE: The AFSRx bits for PE[3:0] won't be changed during the corresponding clock (XMOSC/XSOSC) is selected as the system clock (MCLK).

PE_AFSR2=0x3000_040C

31 30 29 28	27 26 25 24	23 22 21 20	19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0
AFSR15	AFSR14	AFSR13	AFSR12	AFSR11	AFSR10	AFSR9	AFSR8
0000	0000	0000	0000	0000	0000	0000	0000
RW	RW	RW	RW	RW	RW	RW	RW

4(x-8)+3 4(x-8)	AFSRx	Port E Alternative Function Selection bits, x: 8 to 15	
		0000	Alternative Function 0 (AF0)
		0001	Alternative Function 1 (AF1)
		0010	Alternative Function 2 (AF2)
		0011	Alternative Function 3 (AF3)
		0100	Alternative Function 4 (AF4)
		0101	Alternative Function 5 (AF5)
		0110	Alternative Function 6 (AF6)
		0111	Alternative Function 7 (AF7)
		Others	Reserved

Table 34. Functions of PE Port

Port	Pin	Function							
		AF0	AF1	AF2	AF3	AF4	AF5	AF6	AF7
PE	0	SXIN	–	–	–	–	–	–	–
	1	SXOUT	–	–	–	–	–	–	–
	2	XIN	–	–	MOSI2	–	–	–	–
	3	XOUT	–	–	MISO2	–	–	–	–
	4	RTCOUT	–	–	SCK2	–	–	–	–
	5	–	–	–	SS2	–	–	–	–
	6	–	EC11	–	MOSI2	–	–	ADTRG	–
	7	T11OUT	T11CAP	–	MISO2	–	–	–	–
	8	–	EC10	–	SCK2	–	–	–	–
	9	T10OUT	T10CAP	–	SS2	–	–	–	–

7.4.8 PF_AFSR1/2: port F alternative function selection register 1/2

PF_AFSR1/2 register must be set properly before using the port. Otherwise, the port may not function properly.

This register is 32-bit size and accessible in 32/16/8-bit.

PF_AFSR1=0x3000_0508

31 30 29 28	27 26 25 24	23 22 21 20	19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0
AFSR7	AFSR6	AFSR5	AFSR4	AFSR3	AFSR2	AFSR1	AFSR0
0000	0000	0000	0000	0000	0000	0000	0000
RW	RW	RW	RW	RW	RW	RW	RW

4x+3 4x	AFSRx	Port F Alternative Function Selection bits, x: 0 to 7	
		0000	Alternative Function 0 (AF0)
		0001	Alternative Function 1 (AF1)
		0010	Alternative Function 2 (AF2)
		0011	Alternative Function 3 (AF3)
		0100	Alternative Function 4 (AF4)
		0101	Alternative Function 5 (AF5)
		0110	Alternative Function 6 (AF6)
		0111	Alternative Function 7 (AF7)
		Others	Reserved

PF_AFSR2=0x3000_050C

31 30 29 28	27 26 25 24	23 22 21 20	19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0
AFSR15	AFSR14	AFSR13	AFSR12	AFSR11	AFSR10	AFSR9	AFSR8
0000	0000	0000	0000	0000	0000	0000	0000
RW	RW	RW	RW	RW	RW	RW	RW

4(x-8)+3 4(x-8)	AFSRx	Port F Alternative Function Selection bits, x: 8 to 15	
		0000	Alternative Function 0 (AF0)
		0001	Alternative Function 1 (AF1)
		0010	Alternative Function 2 (AF2)
		0011	Alternative Function 3 (AF3)
		0100	Alternative Function 4 (AF4)
		0101	Alternative Function 5 (AF5)
		0110	Alternative Function 6 (AF6)
		0111	Alternative Function 7 (AF7)
		Others	Reserved

Table 35. Functions of PF Port

Port	Pin	Function							
		AF0	AF1	AF2	AF3	AF4	AF5	AF6	AF7
PF	0	–	EC50	LPRXD0	–	AN12	SCL2	–	SEG25
	1	T50OUT	–	LPTXD0	SCK0	AN13	SDA2	–	SEG26
	2	–	T50INP	–	MISO0	AN14	LPDE0	–	SEG27
	3	–	–	–	MOSI0	AN15	–	–	SEG28
	4	T12OUT	T12CAP	–	MOSI3	–	SCL2	–	SEG4
	5	–	EC12	–	MISO3	–	SDA2	–	SEG5
	6	T13OUT	T13CAP	–	SCK3	–	–	–	SEG6
	7	–	EC13	–	SS3	–	LPDE1	–	SEG7

7.4.9 Pn_PUPD: port n Pull-up/down resistor selection register

Every pin of the port has an on-chip pull-up/down resistor, which can be configured by Pn_PUPD registers.

This register is 32-bit size and accessible in 32/16/8-bit. (n = A to F).

PA_PUPD=0x3000_0010, PB_PUPD=0x3000_0110, PC_PUPD=0x3000_0210
PD_PUPD=0x3000_0310, PE_PUPD=0x3000_0410, PF_PUPD=0x3000_0510

31 30	29 28	27 26	25 24	23 22	21 20	19 18	17 16	15 14	13 12	11 10	9 8	7 6	5 4	3 2	1 0
PUPD15	PUPD14	PUPD13	PUPD12	PUPD11	PUPD10	PUPD9	PUPD8	PUPD7	PUPD6	PUPD5	PUPD4	PUPD3	PUPD2	PUPD1	PUPD0
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

2x+1 2x	PUPDx	Port n Pull-up/down Resistor Selection bits, x: 0 to 15	
		00	Disable pull-up/down resistor
		01	Enable pull-up resistor
		10	Enable pull-down resistor
		11	Reserved

NOTES:

1. The pull-up/down resistor of PE[3:0] are automatically disabled regardless of the corresponding PUPDx value if the pins are configured as alternative function pins for x-tal (XIN, XOUT, SXIN, and SXOUT).
2. The PUPDx bits for PC5, PC6, and PD5 are set to “01b”, “10b”, and “01b” for SWDIO/SWCLK/BOOT by reset, respectively.
3. PC5: SWDIO, PC6: SWCLK, PD5: BOOT

7.4.10 Pn_INDR: port n input data register

Each pin level status can be read in the Pn_INDR register. Except for analog input and alternative mode output, the pin level can be detected in the Pn_INDR register.

This register is 32-bit size and accessible in 32/16/8-bit. (n = A to F).

PA_INDR=0x3000_0014, PB_INDR=0x3000_0114, PC_INDR=0x3000_0214
PD_INDR=0x3000_0314, PE_INDR=0x3000_0414, PF_INDR=0x3000_0514

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	INDR15	INDR14	INDR13	INDR12	INDR11	INDR10	INDR9	INDR8	INDR7	INDR6	INDR5	INDR4	INDR3	INDR2	INDR1	INDR0
0x0000	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
–	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO

x	INDRx	Port n Input Data bit, x: 0 to 15

7.4.11 Pn_OUTDR: port n output data register

When a pin is set as an output in GPIO mode, output level of the pin is defined by Pn_OUTDR registers.

This register is 32-bit size and accessible in 32/16/8-bit. (n = A to F).

PA_OUTDR=0x3000_0018, PB_OUTDR=0x3000_0118, PC_OUTDR=0x3000_0218
PD_OUTDR=0x3000_0318, PE_OUTDR=0x3000_0418, PF_OUTDR=0x3000_0518

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																OUTDR15	OUTDR14	OUTDR13	OUTDR12	OUTDR11	OUTDR10	OUTDR9	OUTDR8	OUTDR7	OUTDR6	OUTDR5	OUTDR4	OUTDR3	OUTDR2	OUTDR1	OUTDR0
0x0000																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
–																RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

x OUTDRx	Port n Output Data bit, x: 0 to 15. The OUTDR bits can be individually set/cleared by writing to the Pn_BSR/Pn_BCR register.

7.4.12 Pn_BSR: port n output bit set register

Pn_BSR are used for controlling each bit of the Pn_OUTDR register. Writing a '1' into the specific bit position will set a corresponding bit of Pn_OUTDR to '1'. Writing '0' in the register has no effect.

This register is 32-bit size and accessible in 32/16/8-bit. (n = A to F)

PA_BSR=0x3000_001C, PB_BSR=0x3000_011C, PC_BSR=0x3000_021C
PD_BSR=0x3000_031C, PE_BSR=0x3000_041C, PF_BSR=0x3000_051C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																BSR15	BSR14	BSR13	BSR12	BSR11	BSR10	BSR9	BSR8	BSR7	BSR6	BSR5	BSR4	BSR3	BSR2	BSR1	BSR0
0x0000																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
–																WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO

x BSRx	Port n Output Set bit, x: 0 to 15. These bits are always read to 0x00.	
	0	No effect
	1	Set the corresponding OUTDRx bit (Automatically cleared to 0)

7.4.13 Pn_BCR: port n output bit clear register

Pn_BCR are used for controlling each bit of Pn_OUTDR register. Writing a ‘1’ into the specific bit will set a corresponding bit of Pn_OUTDR to ‘0’. Writing ‘0’ in this register has no effect.

This register is 32-bit size and accessible in 32/16/8-bit. (n = A to F).

PA_BCR=0x3000_0020, PB_BCR=0x3000_0120, PC_BCR=0x3000_0220
PD_BCR=0x3000_0320, PE_BCR=0x3000_0420, PF_BCR=0x3000_0520

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																BCR15	BCR14	BCR13	BCR12	BCR11	BCR10	BCR9	BCR8	BCR7	BCR6	BCR5	BCR4	BCR3	BCR2	BCR1	BCR0
0x0000																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
–																WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO

x BCRx	Port n Output Clear bit, x: 0 to 15. These bits are always read to 0x00.	
	0	No effect
	1	Clear the corresponding OUTDRx bit (Automatically cleared to 0)

7.4.14 Pn_OUTDMSK: port n output data mask register

Pn_OUTDMSK are used for protecting each bit of Pn_OUTDR register. Writing a ‘1’ into the specific bit will protect a corresponding bit of Pn_OUTDR. Writing ‘0’ in this register is unmask.

This register is 32-bit size and accessible in 32/16/8-bit. (n = A to F).

PA_OUTDMSK=0x3000_0024, PB_OUTDMSK=0x3000_0124, PC_OUTDMSK=0x3000_0224
PD_OUTDMSK=0x3000_0324, PE_OUTDMSK=0x3000_0424, PF_OUTDMSK=0x3000_0524

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																OUTDMSK15	OUTDMSK14	OUTDMSK13	OUTDMSK12	OUTDMSK11	OUTDMSK10	OUTDMSK9	OUTDMSK8	OUTDMSK7	OUTDMSK6	OUTDMSK5	OUTDMSK4	OUTDMSK3	OUTDMSK2	OUTDMSK1	OUTDMSK0
0x0000																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
–																RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

x OUTDMSKx	Port n Output Data Mask bit, x: 0 to 15.	
	0	Unmask. The corresponding OUTDRx bit can be changed.
	1	Mask. The corresponding OUTDRx bit is protected.

7.4.15 Pn_DBCR: port n debounce control register

This register is 32-bit size and accessible in 32/16/8-bit. (n = A to F).

PA_DBCR=0x3000_0028, PB_DBCR=0x3000_0128, PC_DBCR=0x3000_0228
PD_DBCR=0x3000_0328, PE_DBCR=0x3000_0428, PF_DBCR=0x3000_0528

31 30 29 28 27 26 25 24	23 22 21 20 19	18 17 16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		**DBCLK**	**DBEN15**	**DBEN14**	**DBEN13**	**DBEN12**	**DBEN11**	**DBEN10**	**DBEN9**	**DBEN8**	**DBEN7**	**DBEN6**	**DBEN5**	**DBEN4**	**DBEN3**	**DBEN2**	**DBEN1**	**DBEN0**
0x00	**00000**	**000**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
–	**–**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**

18 16	DBCLK	Port n Debounce Filter Sampling Clock Selection	
		000	HCLK/1
		001	HCLK/4
		010	HCLK/16
		011	HCLK/64
		100	HCLK/256
		101	HCLK/1024
		110	Reserved
		111	Reserved
x	DBENx	Port n Debounce Enable bit, x: 0 to 15.	
		0	Disable debounce filter
		1	Enable debounce filter

NOTES:

1. If a level is not detected on an enabled pin three or more times in a row at the sampling clock, the signal is eliminated as noise.
2. The port debounce should be disabled before DEEP SLEEP mode.
3. The debounce of the PD5 (BOOT) Pin will be enabled on system reset.

7.5 Functional description

When input function of an I/O port is used by the Pin Control Register, output function of the I/O port is disabled.

Each port functions differently according to the Alternative Function Selection Register.

The Input Data Register captures current data on the I/O pin or debounced input data at every GPIO clock cycle.

Figure 32. Port Structure Block Diagram

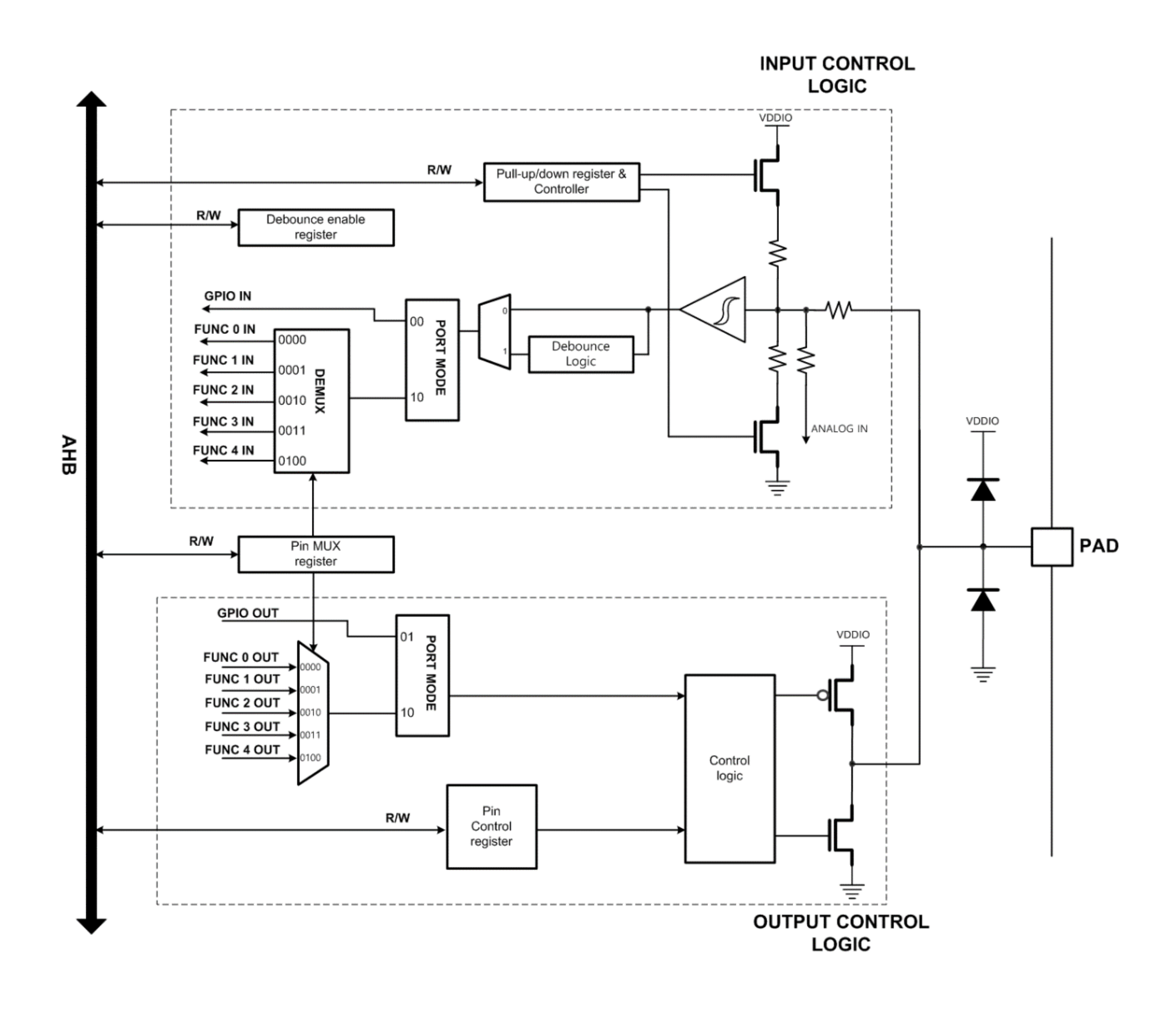
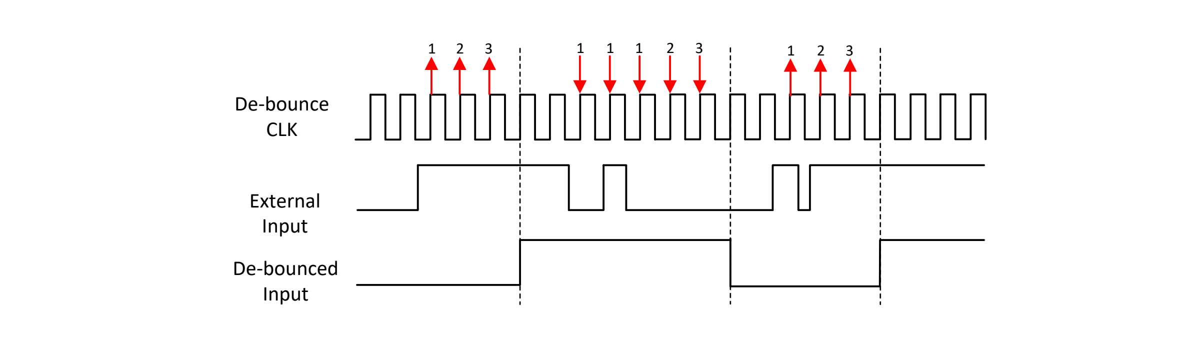
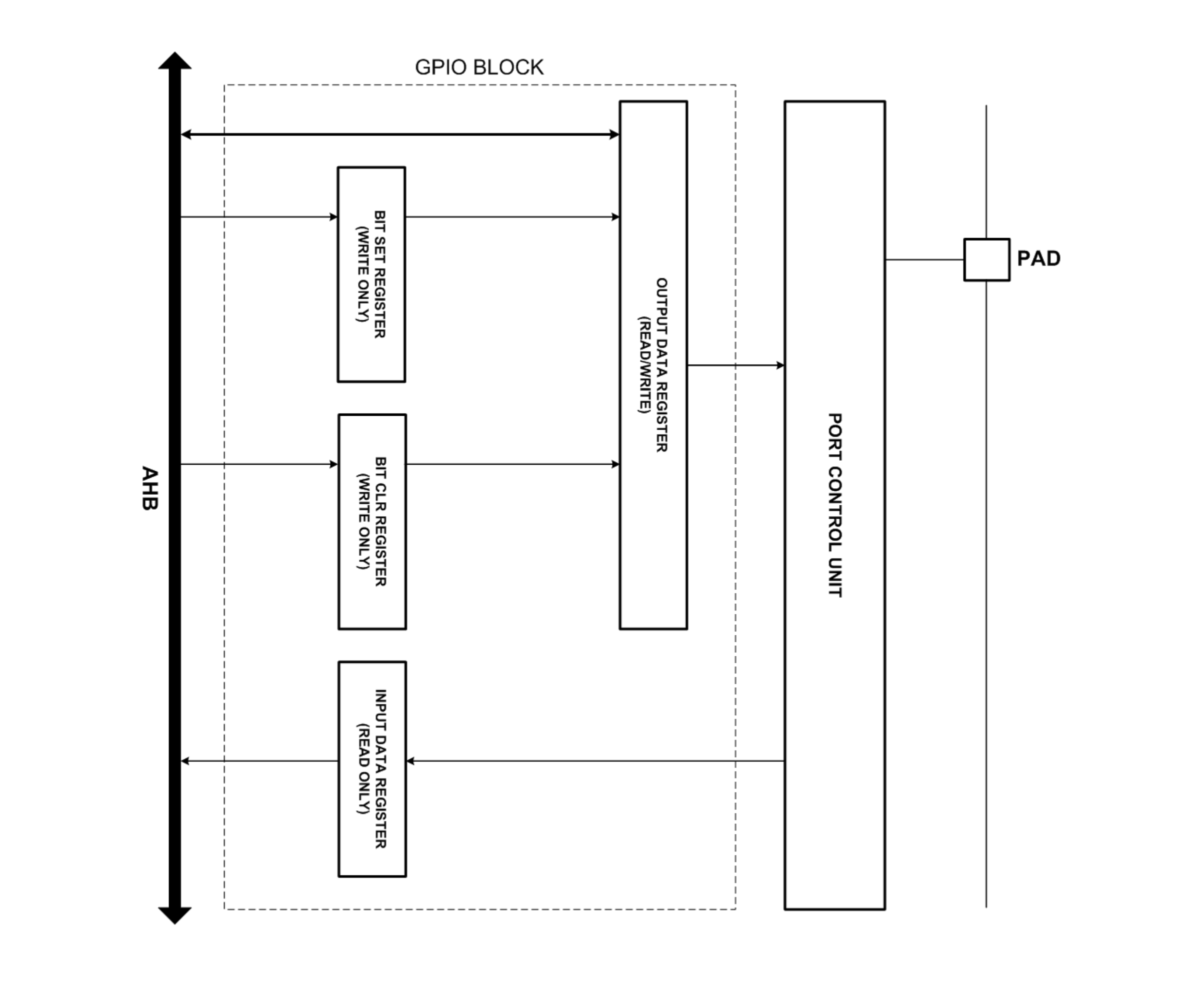


Figure 33. Debounce Function Timing Diagram



When an I/O port is configured as an output, the value written to the GPIO Output Data Register is output on the I/O Pin. When the Bit Set Register is set, the GPIO Output Data Register is set to High. When the Bit Clr Register is set, the GPIO Output Data Register is set to Low. The Input Data Register captures current data on the I/O pin or debounced input data at every GPIO clock cycle.

Figure 34. GPIO Block Diagram


GPIO BLOCK
AHB
BIT SET REGISTER (WRITE ONLY)
BIT CLR REGISTER (WRITE ONLY)
INPUT DATA REGISTER (READ ONLY)
OUTPUT DATA REGISTER (READ/WRITE)
PORT CONTROL UNIT
PAD

8 Watchdog Timer (WDT)

Watchdog Timer (WDT) rapidly detects CPU malfunctions such as endless loops caused by noise and recovers the CPU to the normal state. WDT signal for malfunction detection can be used as either a CPU reset or an interrupt request.

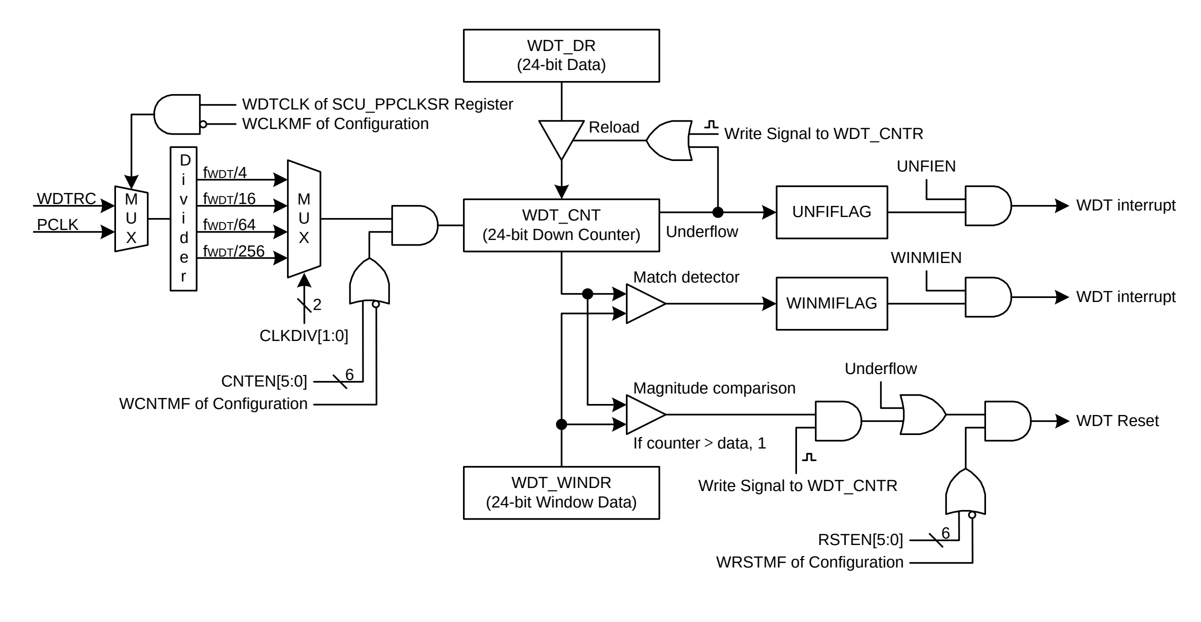
When the WDT is not being used for malfunction detection, it can be used as a timer to generate interrupts at fixed intervals. When WDT_CNT value reaches WDT_WINDR value, a watchdog interrupt can be generated. The underflow time of the WDT can be set by WDT_DR. If an underflow occurs, an internal reset may be generated. The WDT operates at 40kHz which is the embedded RC oscillator's clock.

The WDT operations are listed in the followings:

- 24-bit down counter (WDT_CNT)
- Select reset or periodic interrupt
- Count clock selection
- Watchdog overflow output signal
- Includes Counter Window function

8.1 WDT block diagram

Figure 35. WDT Block Diagram



NOTE: When a reload signal is generated, the WDT_DR value is reloaded to WDT_CNT register, after 1-clock delay, with the clock selected by the CLKDIV bits.

8.2 Registers

Base address and register map of WDT are shown in Table 36 and Table 37.

Table 36. Base Address of WDT

Name	Base address
WDT	0x4000_1A00

Table 37. WDT Register Map

Name	Offset	Type	Description	Reset value
WDT_CR	0x0000	RW	Watchdog Timer Control Register	0x00000000
WDT_SR	0x0004	RW	Watchdog Timer Status Register	0x00000080
WDT_DR	0x0008	RW	Watchdog Timer Data Register	0x00000FFF
WDT_CNT	0x000C	RO	Watchdog Timer Counter Register	0x00000FFF
WDT_WINDR	0x0010	RW	Watchdog Timer Window Data Register	0x00001FFF
WDT_CNTR	0x0014	WO	Watchdog Timer Counter Reload Register	0x00000000

8.2.1 WDT_CR: Watchdog Timer control register

WDT module should be configured properly before running. The WDT module can reset the system or assert an interrupt signal to the system.

This register is 32-bit size.

WDT_CR=0x4000_1A00

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10	9 8 7 6 5 4	3	2	1 0
WTIDKY	**RSTEN**	**CNTEN**	**WINMIEN**	**UNFIEN**	**CLKDIV**
0x0000	**000000**	**000000**	**0**	**0**	**00**
WO	**RW**	**RW**	**RW**	**RW**	**RW**

31 16	WTIDKY	Write Identification Key.	
		When writing, write 0x5A69 to these bits, or else writing is ignored.	
15 10	RSTEN	Watchdog Timer Reset Enable.	
		0x25	Disable watchdog timer reset.
		Others	Enable watchdog timer reset.
9 4	CNTEN	Watchdog Timer Counter Enable.	
		0x1A	Disable watchdog timer counter.
		Others	Enable watchdog timer counter.
3	WINMIEN	Watchdog Timer Window Match Interrupt Enable.	
		0	Disable window data match interrupt.
		1	Enable window data match interrupt.
2	UNFIEN	Watchdog Timer Underflow Interrupt Enable.	
		0	Disable watchdog timer underflow interrupt.
		1	Enable watchdog timer underflow interrupt.
1 0	CLKDIV	Watchdog Timer Clock Divider. The watchdog timer clock is selected by SCU_PPCLKSR[0] bit of clock generation and CONF_WDTCNFIG[2] bit of Configure Option Page 1.	
		00	fWDT/4
		01	fWDT/16
		10	fWDT/64
		11	fWDT/256

8.2.2 WDT_SR: Watchdog Timer status register

WDT_SR register is 32-bit size and accessible in 32/16/8-bit.

WDT_SR=0x4000_1A04

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8	7	6 5 4 3 2	1	0
Reserved	**DBGCNTEN**	**Reserved**	**WINMIFLAG**	**UNFIFLAG**
0x000000	**1**	**00000**	**0**	**0**
–	**RW**	**–**	**RW**	**RW**

7	DBGCNTEN	Watchdog Timer Counter Enable bit when the core is halted in debug mode.
		0 The watchdog timer counter continues operation even if the core is halted.
		1 The watchdog timer counter stops when the core is halted.
		NOTE: This bit is set to '1' by POR/WAKUP3 reset.
1	WINMIFLAG	Watchdog Timer Window Match Interrupt Flag.
		0 No request occurred.
		1 Request occurred. The bit is cleared to '0' when '1' is written.
0	UNFIFLAG	Watchdog Timer Underflow Interrupt Flag.
		0 No request occurred.
		1 Request occurred. The bit is cleared to '0' when '1' is written.

8.2.3 WDT_DR: Watchdog Timer data register

WDT_DR register is used to update WDT_CNT register.

This register is 32-bit size.

WDT_DR=0x4000_1A08

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Reserved	**DATA**
0x00	**0x000FFF**
–	**RW**

23 0	DATA	Watchdog Timer Data. The range is 0x000000 to 0xFFFFFF.

NOTE: Once any value is written to this data register, the register cannot be changed until system reset.

8.2.4 WDT_CNT: Watchdog Timer counter register

WDT_CNT register represents current count value of the 32-bit down counter. When the counter value reaches 0, an interrupt or a reset will be asserted.

This register is 32-bit size.

WDT_CNT=0x4000_1A0C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved	**CNT**		
0x00	**0x000FFF**		
–	**RO**		

23 0	CNT	Watchdog Timer Counter

8.2.5 WDT_WINDR: Watchdog Timer window data register

WDT_WINDR register is used to compare to WDT_CNT for WINDOW function.

This register is 32-bit size.

WDT_WINDR=0x4000_1A10

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved	**WDATA**		
0x00	**0x001FFF**		
-	**RW**		

23 0	WDATA	Watchdog Timer Window Data. The range is 0x000000 to 0xFFFFFF.

NOTE: Once any value is written to this window data register, the register cannot be changed until system reset.

8.2.6 WDT_CNTR: Watchdog Timer counter reload register

WDT_CNTR register is used to generate a reload signal. When a reload signal is generated, the WDT_DR value is reloaded to WDT_CNT.

This register is 32-bit size.

WDT_CNTR=0x4000_1A14

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved			**CNTR**
0x000000			**0x00**
-			**WO**

7 0	CNTR	Watchdog Timer Counter Reload bits.	
		0x6A	Reload the WDT_DR value to watchdog timer counter and re-start. (Automatically cleared to “0x00” after operation)
		Others	No effect

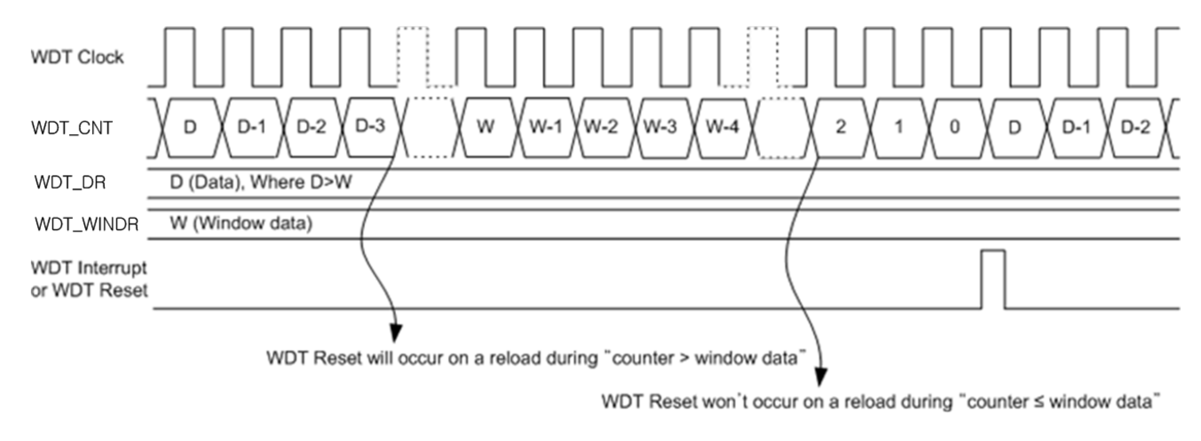
8.3 Functional description

Watchdog Timer count is enabled by CNTEN (WDT_CR[9:4]) settings which can be any value other than 0x1A. As the WDT activates, the down counter will start counting from the load value. If the RSTEN (WDT_CR[15:10]) is set as any value other than 0x25, WDT reset would be asserted when the WDT counter value reaches 0 (underflow event) from WDT_DR value.

Before WDT counter reaches 0, software can write 0x6A to WDT_CNTR register in order to reload WDT counter when the counter value is less than or equal to the value of window data register. WDT reset may be asserted if the reload occurs when counter > window data.

8.3.1 Timing diagram

Figure 36. WDT Interrupt and WDT Reset Timing Diagram


WDT Clock
WDT_CNT
D
D-1
D-2
D-3
W
W-1
W-2
W-3
W-4
2
1
0
D
D-1
D-2
WDT_DR
D (Data), Where D>W
WDT_WINDR
W (Window data)
WDT Interrupt or WDT Reset
WDT Reset will occur on a reload during "counter > window data"
WDT Reset won't occur on a reload during "counter ≤ window data"

8.3.2 Pre-scale table

The WDT includes a 24-bit down counter with programmable pre-scaler to define different time-out intervals.

Clock sources of the WDT can be WDTRC or PCLK. The PCLK can be selected by setting WDTCLK (SCU_PPCLKSR[0]) to ‘1’. Then CONF_WDTCNFIG[2] bit of Configure Option Page 1 is cleared to logic ‘0’.

A WDT counter can be set as a base clock by controlling a 2-bit pre-scaler CLKDIV [1:0] in the WDT_CR register. The maximum pre-scaled value is “clock source frequency/256”. The pre-scaled WDT counter clock frequency values are listed in Table 38.

Selectable clock source (40kHz ~ 32MHz) and time-out interval at a single count

Time-out period = (Load Value + 1) * (1/pre-scaled WDT counter clock frequency)

*Time out period (when the Load Value reaches 0, underflow flag is set to ‘1’)

Table 38. Pre-scaled WDT Counter Clock Frequency

Clock source	WDTCLKIN	WDTCLKIN/4	WDTCLKIN/16	WDTCLKIN/64	WDTCLKIN/256
WDTRC	40kHz	10kHz	2.5kHz	0.625kHz	0.156kHz
PCLK	PCLK	PCLK/4	PCLK/16	PCLK/64	PCLK/256

9 Real Timer Clock and Calendar (RTCC)

Real Timer Clock and Calendar (RTCC) has a function for RTC (Real Time Clock) and calendar operations. Internal structure of the RTCC is implemented with the clock source select circuit, second/minute/hour/day/week/month/year counter circuits, alarm circuit, output select circuit, and error correction circuit.

The RTCC is an independent BCD counter. The RTCC circuitry and the related control bits are not reset by a system reset other than POR/WAKUP3.

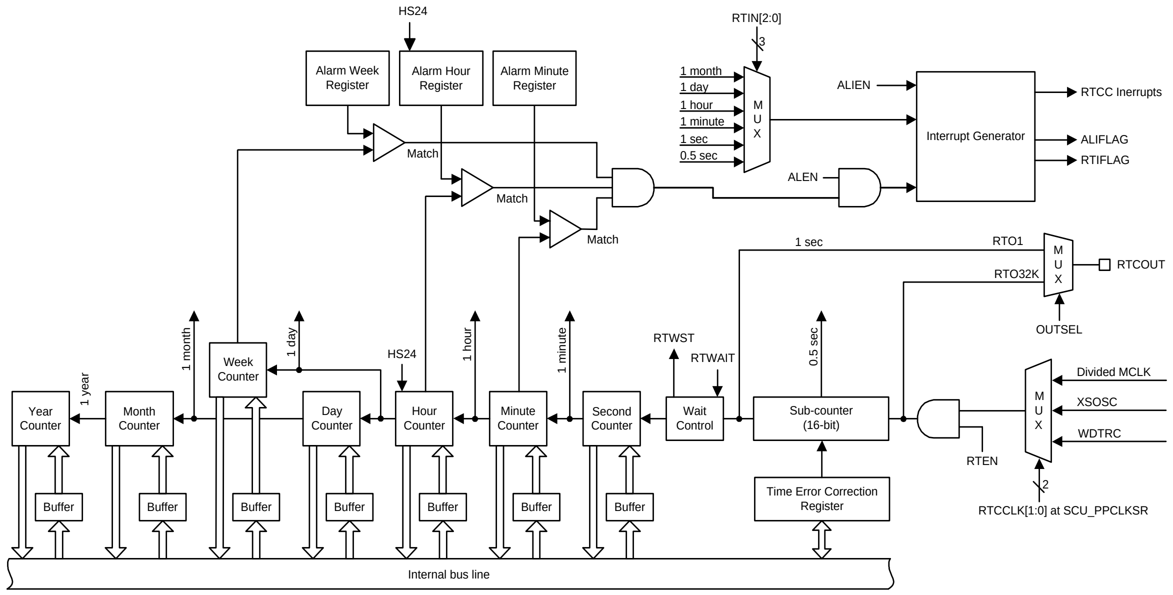
Main operations of the RTCC are introduced in the following list:

- Calendar with 0.5 seconds, seconds, minutes, hours, day, week, month, and year up to 2099
- Time error correction function
- Alarm function with interrupt
- Wake-up possible from DEEP SLEEP mode

9.1 RTCC block diagram

Figure 37 shows a block diagram of the RTCC block.

Figure 37. RTCC Block Diagram



NOTE: Time in the block diagram is based on 32.768 kHz.

9.2 Registers

Base address and register map of the RTCC are shown in Table 39 and Table 40.

Table 39. Base Address of RTCC

Name	Base address
RTCC	0x4000_5200

Table 40. RTCC Register Map

Name	Offset	Type	Description	POR/WAKUP3 Reset value (Retained at the other reset)
RTC_CR	0x0000	RW	RTCC Control Register	0x00000000
RTC_ECR	0x0004	RW	RTCC Time Error Correction Register	0x00000000
RTC_SCNT	0x0008	RO	RTCC Sub-counter Register	0x00000000
RTC_SEC	0x000C	RW	RTCC Second Counter Register	0x00000000
RTC_MIN	0x0010	RW	RTCC Minute Counter Register	0x00000000
RTC_HOUR	0x0014	RW	RTCC Hour Counter Register	0x00000012
RTC_DAY	0x0018	RW	RTCC Day Counter Register	0x00000001
RTC_WEEK	0x001C	RW	RTCC Week Counter Register	0x00000000
RTC_MONTH	0x0020	RW	RTCC Month Counter Register	0x00000001
RTC_YEAR	0x0024	RW	RTCC Year Counter Register	0x00000000
RTC_ALMIN	0x0028	RW	RTCC Alarm Minute Register	0x00000000
RTC_ALHOUR	0x002C	RW	RTCC Alarm Hour Register	0x00000012
RTC_ALWEEK	0x0030	RW	RTCC Alarm Week Register	0x00000000

9.2.1 RTC_CR: RTCC control register

RTC_CR register is 32-bit size and accessible in 32/16/8-bit.

RTC_CR=0x4000_5200

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																RTEN	RTIN			RTIFLAG	HS24	Reserved	OUTSEL	ALEN	ALIEN	ALIFLAG	Reserved			RTWST	RTWAIT
0x0000																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
–																RW	RW	RW	RW	RW	RW	-	RW	RW	RW	RW	-	-	-	RO	RW

15	RTEN	RTCC Enable.	
		0	RTCC Disable
		1	RTCC Enable
14 12	RTIN	RTCC Interrupt Interval Selection.	
		000	Disable RTCC Interval Interrupt.(NOTE1)
		001	Once per 0.5 sec
		010	Once per 1 sec
		011	Once per 1 min
		100	Once per 1 hour
		101	Once per 1 day
		110	Once per 1 month
		111	Reserved. Value is not changed.
11	RTIFLAG	RTCC Interval Interrupt Flag.	
		0	No request occurred.
		1	Request occurred. The bit is cleared to '0' when '1' is written.
10	HS24	12/24-hour System Selection. (NOTE2)	
		0	12-hour system.
		1	24-hour system.
8	OUTSEL	RTCOUT Selection.	
		0	RTO 1 (1Hz).
		1	RTO 32K (32kHz).
7	ALEN	RTCC Alarm Match Operation Enable. (NOTE3)	
		0	Disable RTCC alarm match operation.
		1	Enable RTCC alarm match operation
6	ALIEN	RTCC Alarm Match Interrupt Enable.	
		0	Disable
		1	Enable
5	ALIFLAG	RTCC Alarm Match Interrupt Flag.	
		0	No request occurred.
		1	Request occurred. The bit is cleared to '0' when '1' is written.
1	RTWST	RTCC Wait Status Flag. (NOTE4)	
		0	Counter is operating.
		1	Mode to read/write counter value.
0	RTWAIT	RTCC Wait Status Flag. (NOTE5)	
		0	Set counter Operation
		1	Stop RTSEC to RTYEAR counters for read/write counter value

NOTES:

1. When changing the values of RTIN[2:0] while the counter operates (RTEN = 1), rewrite the values of RTIN[2:0] after disabling interrupt servicing RTCC Interrupt by using the Interrupt & Wake-up Source Mask Register (INTC_MSK). Furthermore, after rewriting the values of RTIN[2:0], enable interrupt servicing after clearing the RTIFLAG flag.
2. Rewrite the HS24 value after setting RTWAIT (bit 0 of RTC_CR) to 1. If the HS24 value is changed, the values of the RTCC hour counter register (RTC_HOUR) and RTCC alarm hour register (RTC_ALHOUR) change according to the specified time system.
   Table 41. Value of RTC_HOUR/RTC_ALHOUR by HS24 bit shows the displayed time digits.
3. When setting a value to the ALEN bit while the counter operates (RTEN = 1) and ALIEN = 1, rewrite the ALEN bit after disabling interrupt servicing RTCC Interrupt by using Interrupt & Wake-up Source Mask Register (INTC_MSK). Furthermore, clear the ALIFLAG flag after

rewriting the ALEN bit. When setting each alarm register (ALIEN flag of RTC_CR, the RTC_ALMIN register, the RTC_ALHOUR register, and the RTC_ALWEEK register), set match operation to be invalid (“0”) for the ALEN bit.

4. This status flag indicates whether the setting of RTWAIT is valid. Before reading or writing the counter value, confirm that the value of this flag is 1.
5. This bit controls the operation of the counter. Be sure to write “1” to it to read or write the counter value. Because the RTCC sub-counter (RTC_SCNT) continues operation, complete reading or writing of it in 1 second, and clear this bit back to 0. When RTWAIT = 1, it takes up to 2 clocks (RTCC clock) until the counter value can be read or written. If the RTCC sub-counter (RTC_SCNT) overflows when RTWAIT = 1, it counts up after RTWAIT = 0. If the RTCC second counter register (RTC_SEC) is written, however, it does not count up because RTCC sub-counter (RTC_SCNT) is cleared.

9.2.2 RTC_ECR: RTCC time error correction register

RTC_ECR register is 32-bit size and accessible in 32/16/8-bit.

RTC_ECR=0x4000_5204

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
Reserved			**ECTM**	**ECSIGN**	**ECV**					
0x000000			**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
–			**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**

7	ECTM	Time Error Correction Timing Selection. (NOTE1)	
		0	Corrects time error when the second digits are at 00H, 20H, or 40H (every 20 seconds).
		1	Corrects watch error only when the second digits are at 00H (every 60 seconds).
6	ECSIGN	Time Error Correction Data Sign. (NOTE2~4)	
		0	Increases by {(ECV5, ECV4, ECV3, ECV2, ECV1, ECV0) – 1} × 2
		1	Decreases by {(/ECV5, /ECV4, /ECV3, /ECV2, /ECV1, /ECV0) + 1} × 2
5 0	ECV	Time Error Correction Data.	

NOTES:

1. Do not write to the RTC_ECR register at the following timing.
   - When ECTM = 0 is set: For a period of SEC = 00H, 20H, 40H
   - When ECTM = 1 is set: For a period of SEC = 00H
2. When (ECSIGN, ECV5, ECV4, ECV3, ECV2, ECV1, ECV0) = (n, 0, 0, 0, 0, 0, n), the watch error is not corrected. (Where n = 0 or 1)
3. /ECV5 to /ECV0 are the inverted values of the corresponding bits (000011 when 111100).
4. Range of correction value: (when ECSIGN = 0) 2, 4, 6, 8, … , 120, 122, 124
   (when ECSIGN = 1) –2, –4, –6, –8, … , –120, –122, –124

9.2.3 RTC_SCNT: RTCC sub counter register

RTC_SCNT register is 32-bit size and accessible in 32/16-bit.

RTC_SCNT=0x4000_5208

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		**RTCNT**	
0x0000		**0x0000**	
–		**RO**	

15 0	RTCNT	RTCC Sub-counter.

NOTES:

1. When a correction is made by using the RTC_ECR register, the value of RTC_SCNT may become 8000H or more.
2. The RTC_SCNT is also cleared by writing the RTCC second counter register.
3. The value read from this register is not guaranteed if it is read during operation, because a value that is changing is read.

9.2.4 RTC_SEC: RTCC second counter register

RTC_SEC register is 32-bit size and accessible in 32/16/8-bit.

RTC_SEC=0x4000_520C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
Reserved			**Reserved**	**RSEC**						
0x000000			**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
–			**-**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**

6 0	RSEC	RTCC Second counter.

NOTES:

1. The RTC_SEC register takes a value of 0 to 59 (decimal) and indicates the count value of seconds. It counts up when the RTCC sub-counter overflows.
2. Set a decimal value of 00 to 59 to this register in BCD code. If a value outside this range is tried to be written in the register, the value is ignored.

9.2.5 RTC_MIN: RTCC minute counter register

RTC_MIN register is 32-bit size and accessible in 32/16/8-bit.

RTC_MIN=0x4000_5210

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
Reserved			Reserved	RMIN						
0x000000			0	0	0	0	0	0	0	0
–			-	RW	RW	RW	RW	RW	RW	RW

6 0	RMIN	RTCC Minute counter.

NOTES:

1. The RTC_MIN register takes a value of 0 to 59 (decimal) and indicates the count value of minutes. It counts up when the RTCC second counter register overflows.
2. Even if the RTCC second counter register overflows while this register is being written, this register ignores the overflow and is set to the value written.
3. Set a decimal value of 00 to 59 to this register in BCD code. If a value outside this range is tried to be written in the register, the value is ignored.

9.2.6 RTC_HOUR: RTCC hour counter register

RTC_HOUR register is 32-bit size and accessible in 32/16/8-bit.

RTC_HOUR=0x4000_5214

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
Reserved			Reserved		RHOUR					
0x000000			0	0	0	1	0	0	1	0
–			-	-	RW	RW	RW	RW	RW	RW

5 0	RHOUR	RTCC Hour counter.

NOTES:

1. The RTC_HOUR register takes a value of 00 to 23 or 01 to 12, 21 to 32 (decimal) and indicates the count value of hours. It counts up when the RTCC minute counter register overflows.
2. Even if the RTCC minute counter register overflows while this register is being written, this register ignores the overflow and is set to the value written. Specify a decimal value of 00 to 23, 01 to 12, or 21 to 32 by using BCD code according to the time system specified using HS24 bit of RTC_CR(RTCC control register). If a value outside this range is tried to be written in the register, the value is ignored.
3. RTHOUR5 bit of RTC_HOUR indicates AM(0)/PM(1) if HS24(RTC_CR[10]) = 0 (if the 12-hour system is selected).

Table 41. Value of RTC_HOUR/RTC_ALHOUR by HS24 bit

24-Hour Display (HS24 bit = 1)		**12-Hour Display (HS24 bit = 0)**	
Time	**RTC_HOUR Register** **RTC_ALHOUR Register**	**Time**	**RTC_HOUR Register** **RTC_ALHOUR Register**
0	00H	0 a.m.	12H
1	01H	1 a.m.	01H
2	02H	2 a.m.	02H
3	03H	3 a.m.	03H
4	04H	4 a.m.	04H
5	05H	5 a.m.	05H
6	06H	6 a.m.	06H
7	07H	7 a.m.	07H
8	08H	8 a.m.	08H
9	09H	9 a.m.	09H
10	10H	10 a.m.	10H
11	11H	11 a.m.	11H
12	12H	0 p.m.	32H
13	13H	1 p.m.	21H
14	14H	2 p.m.	22H
15	15H	3 p.m.	23H
16	16H	4 p.m.	24H
17	17H	5 p.m.	25H
18	18H	6 p.m.	26H
19	19H	7 p.m.	27H
20	20H	8 p.m.	28H
21	21H	9 p.m.	29H
22	22H	10 p.m.	30H
23	23H	11 p.m.	31H

9.2.7 RTC_DAY: RTCC day counter register

RTC_DAY register is 32-bit size and accessible in 32/16/8-bit.

RTC_DAY=0x4000_5218

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																								Reserved		RDAY					
0x000000																								0	0	0	0	0	0	0	1
–																								-	-	RW	RW	RW	RW	RW	RW

5 0	RDAY	RTCC Day counter.

NOTES:

1. The RTC_DAY register takes a value of 1 to 31 (decimal) and indicates the count value of days. It counts up when the RTCC hour counter register overflows.
2. Even if the RTCC hour counter register overflows while this register is being written, this register ignores the overflow and is set to the value written.
3. Set a decimal value of 01 to 31 to this register in BCD code. If a value outside this range is tried to be written in the register, the value is ignored.
4. The RTC_DAY register counts as follows.
   - — 01 to 31 (January, March, May, July, August, October, December)
   - — 01 to 30 (April, June, September, November)
   - — 01 to 29 (February of leap year)
   - — 01 to 28 (February of normal year)

9.2.8 RTC_WEEK: RTCC week counter register

RTC_WEEK register is 32-bit size and accessible in 32/16/8-bit.

RTC_WEEK=0x4000_521C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
Reserved			**Reserved**					**RWEEK**		
0x000000			**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
–			-	-	-	-	-	**RW**	**RW**	**RW**

2 0	RWEEK	RTCC Week counter.	
		000	Sunday
		001	Monday
		010	Tuesday
		011	Wednesday
		100	Thursday
		101	Friday
		110	Saturday
		111	Ignored. Value is not changed

NOTES:

1. The RTC_WEEK register takes a value of 0 to 6 (decimal) and indicates the count value of weekdays. It counts up in synchronization with the RTCC day counter register.
2. Set a decimal value of 00 to 06 to this register in BCD code. If a value outside this range is tried to be written in the register, the value is ignored.
3. Values corresponding to the month count register and day count register are not automatically stored to the RTCC week counter register.

9.2.9 RTC_MONTH: RTCC month counter register

RTC_MONTH register is 32-bit size and accessible in 32/16/8-bit.

RTC_MONTH=0x4000_5220

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
Reserved			**Reserved**			**RMONTH**				
0x000000			**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1**
–			-	-	-	**RW**	**RW**	**RW**	**RW**	**RW**

4 0	RMONTH	RTCC Month counter.

NOTES:

1. The RTC_MONTH register takes a value of 1 to 12 (decimal) and indicates the count value of months. It counts up when the RTCC day counter register overflows.
2. Even if the RTCC day counter register overflows while this register is being written, this register ignores the overflow and is set to the value written.
3. Set a decimal value of 01 to 12 to this register in BCD code. If a value outside this range is tried to be written in the register, the value is ignored.

9.2.10 RTC_YEAR: RTCC year counter register

RTC_YEAR register is 32-bit size and accessible in 32/16/8-bit.

RTC_YEAR=0x4000_5224

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
Reserved			RYEAR							
0x000000			0	0	0	0	0	0	0	0
–			RW	RW	RW	RW	RW	RW	RW	RW

7 0	RYEAR	RTCC Year counter.

NOTES:

1. The RTC_YEAR register takes a value of 0 to 99 (decimal) and indicates the count value of years. It counts up when the RTCC month counter register overflows. Values 00, 04, 08, …, 92, and 96 indicate a leap year.
2. Even if the RTCC month counter register overflows while this register is being written, this register ignores the overflow and is set to the value written.
3. Set a decimal value of 00 to 99 to this register in BCD code. If a value outside this range is tried to be written in the register, the value is ignored.

9.2.11 RTC_ALMIN: RTCC alarm minute counter register

RTC_ALMIN register is 32-bit size and accessible in 32/16/8-bit.

RTC_ALMIN=0x4000_5228

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
Reserved			Reserved	AMIN						
0x000000			0	0	0	0	0	0	0	0
–			-	RW	RW	RW	RW	RW	RW	RW

6 0	AMIN	RTCC Alarm Minute counter.

NOTE: This register is used to set minutes of alarm. Set a decimal value of 00 to 59 to this register in BCD code. If a value outside this range is tried to be written in the register, the value is ignored.

9.2.12 RTC_ALHOUR: RTCC alarm hour counter register

RTC_ALHOUR register is 32-bit size and accessible in 32/16/8-bit.

RTC_ALHOUR=0x4000_522C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																								Reserved		AHOUR					
0x000000																								0	0	0	1	0	0	1	0
–																								-	-	RW	RW	RW	RW	RW	RW

5 0	AHOUR	RTCC Alarm Hour counter.

NOTES:

1. This register is used to set hours of alarm. Specify a decimal value of 00 to 23, 01 to 12, or 21 to 32 by using BCD code according to the time system specified using HS24 bit of RTC_CR(RTCC control register). If a value outside this range is tried to be written in the register, the value is ignored.
2. AHOUR5 bit of RTC_ALHOUR indicates AM(0)/PM(1) if HS24 = 0 (if the 12-hour system is selected).

9.2.13 RTC_ALWEEK: RTCC alarm week counter register

RTC_ALWEEK register is 32-bit size and accessible in 32/16/8-bit.

RTC_ALWEEK=0x4000_5230

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
Reserved	**Reserved**	**AWEEK6**	**AWEEK5**	**AWEEK4**	**AWEEK3**	**AWEEK2**	**AWEEK1**	**AWEEK0**
0x000000	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
–	**-**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**

6	AWEEK6	Saturday Alarm Setting
		0 Disable Saturday Alarm
		1 Enable Saturday Alarm
5	AWEEK5	Friday Alarm Setting
		0 Disable Friday Alarm
		1 Enable Friday Alarm
4	AWEEK4	Thursday Alarm Setting
		0 Disable Thursday Alarm
		1 Enable Thursday Alarm
3	AWEEK3	Wednesday Alarm Setting
		0 Disable Wednesday Alarm
		1 Enable Wednesday Alarm
2	AWEEK2	Tuesday Alarm Setting
		0 Disable Tuesday Alarm
		1 Enable Tuesday Alarm
1	AWEEK1	Monday Alarm Setting
		0 Disable Monday Alarm
		1 Enable Monday Alarm
0	AWEEK0	Sunday Alarm Setting
		0 Disable Sunday Alarm
		1 Enable Sunday Alarm

9.3 Functional description

9.3.1 Time error correction

The time of RTCC can be corrected with high accuracy when it is slow or fast, by setting a value to the RTCC time error correction register.
The range of value that can be corrected by using the RTCC time error correction register (RTC_ECR) is shown below.

Table 42. Correctable Range of Time Error

	ECTM = 0 (correction every 20 sec)	ECTM = 1 (correction every 60 sec)
Correctable range	–189.2 ppm to 189.2 ppm	–63.1 ppm to 63.1 ppm
Maximum excludes quantization error	± 1.53 ppm	± 0.51 ppm
Minimum resolution	± 3.05 ppm	± 1.02 ppm

NOTE: If a correctable range is –63.1 ppm or lower and 63.1 ppm or higher, set ECTM to 0.

The correction value used when correcting the count value of the RTCC sub-counter register (RTC_SCNT) is calculated by using the following expression.

Set ECTM to 0 when the correction range is −63.1 ppm or less, or 63.1 ppm or more.

- When ECTM = 0, Correction value = Number of correction counts in 1 minute ÷ 3 = (Oscillation frequency ÷ Target frequency − 1) X 32768 X 60 ÷ 3
- When ECTM = 1, Correction value = Number of correction counts in 1 minute = (Oscillation frequency ÷ Target frequency − 1) X 32768 X 60

NOTES:

1. The correction value is 2, 4, 6, 8, … 120, 122, 124 or −2, −4, −6, −8, … −120, −122, −124.
2. The oscillation frequency is the external sub oscillator clock (XSOSC) value. It can be got through the RTCOUT pin. (when the RTC_ECR value is 0x00000000).
3. The target frequency is the frequency resulting after correction performed by using the time error

The correction value is the time error correction value calculated by RTC_ECR[6:0].

- When ECSIGN = 0, Correction value = {(ECV5, ECV4, ECV3, ECV2, ECV1, ECV0) − 1} X 2
- When ECSIGN = 1, Correction value = − {(/ECV5, /ECV4, /ECV3, /ECV2, /ECV1, /ECV0) + 1} X 2

When (ECSIGN, ECV5, ECV4, ECV3, ECV2, ECV1, ECV0) is (x, 0, 0, 0, 0, 0, x), time error correction is not performed. (x = 0 or 1).

/ECV5, /ECV4, /ECV3, /ECV2, /ECV1, /ECV0 are bit-inverted values (000011 when 111100).

9.3.2 Time error correction example 1

Example of correcting from 32772.3 Hz to 32768 Hz (32772.3 Hz － 131.2 ppm)

9.3.2.1 Measuring the oscillation frequency

The oscillation frequency of each product is measured by outputting about 32 kHz from the RTCOUT pin or outputting about 1 Hz from the RTCOUT pin when the time error correction register is set to its initial value (00H).

9.3.2.2 Calculating the correction value

If the target frequency is assumed to be 32768 Hz (32772.3 Hz － 131.2 ppm), the correction range for －131.2 ppm is －63.1 ppm or less, so assume ECTM to be 0.

The expression for calculating the correction value when ECTM is 0 is applied.

Correction value = Number of correction counts in 1 minute ÷ 3
= (Oscillation frequency ÷ Target frequency － 1) X 32768 X 60 ÷ 3
= (32772.3 ÷ 32768 － 1) X 32768 X 60 ÷ 3
= 86

9.3.2.3 Calculating the values to be set to registers

If the correction value is 0 or more (when delaying), assume ECSIGN to be 0.

Calculate (ECV5, ECV4, ECV3, ECV2, ECV1, ECV0) from the correction value.

{ (ECV5, ECV4, ECV3, ECV2, ECV1, ECV0) － 1} X 2 = 86
(ECV5, ECV4, ECV3, ECV2, ECV1, ECV0) = 44
(ECV5, ECV4, ECV3, ECV2, ECV1, ECV0) = (1, 0, 1, 1, 0, 0)

Consequently, when correcting from 32772.3 Hz to 32768 Hz (32772.3 Hz － 131.2 ppm), setting the correction register such that ECTM is 0 and the correction value is 86 (RTC_ECR[6:0] = 0101100) results in 32768 Hz (0ppm).

9.3.3 Time error correction example 2

Example of correcting from 32767.4 Hz to 32768 Hz (32767.4 Hz + 18.3 ppm)

9.3.3.1 Measuring the oscillation frequency

The oscillation frequency of each product is measured by outputting about 32 kHz from the RTCOUT pin or outputting about 1 Hz from the RTCOUT pin when the time error correction register is set to its initial value (00H).

9.3.3.2 Calculating the correction value

Assume the target frequency to be 32768 Hz (32767.4 Hz + 18.3 ppm) and ECTM to be 1.

The expression for calculating the correction value when ECTM is 1 is applied.

$$\begin{aligned}\text{Correction value} &= \text{Number of correction counts in 1 minute}\\ &= (\text{Oscillation frequency} \div \text{Target frequency} - 1) \text{ X } 32768 \text{ X } 60\\ &= (32767.4 \div 32768 - 1) \text{ X } 32768 \text{ X } 60\\ &= -36\end{aligned}$$

9.3.3.3 Calculating the values to be set to registers

If the correction value is 0 or less (when quickening), assume ECSIGN to be 1.

Calculate (ECV5, ECV4, ECV3, ECV2, ECV1, ECV0) from the correction value.

$$\begin{aligned}&-\{(/\text{ECV5}, /\text{ECV4}, /\text{ECV3}, /\text{ECV2}, /\text{ECV1}, /\text{ECV0}) + 1\} \text{ X } 2 = -36\\ &(/\text{ECV5}, /\text{ECV4}, /\text{ECV3}, /\text{ECV2}, /\text{ECV1}, /\text{ECV0}) = 17\\ &(/\text{ECV5}, /\text{ECV4}, /\text{ECV3}, /\text{ECV2}, /\text{ECV1}, /\text{ECV0}) = (0, 1, 0, 0, 0, 1)\\ &(\text{ECV5}, \text{ECV4}, \text{ECV3}, \text{ECV2}, \text{ECV1}, \text{ECV0}) = (1, 0, 1, 1, 1, 0)\end{aligned}$$

Consequently, when correcting from 32767.4 Hz to 32768 Hz (32767.4 Hz + 18.3 ppm), setting the correction register such that ECTM is 1 and the correction value is －36 (RTC_ECR[6:0] = 1101110) results in 32768 Hz (0ppm).

10 Timer Counter 10/11/12/13

The timer block comprises 4 channels of 16-bit general purpose timers. Each has an independent 16-bit counter and a dedicated prescaler that feeds counting clock. They support periodic timer, PWM pulse, one-shot and capture mode.

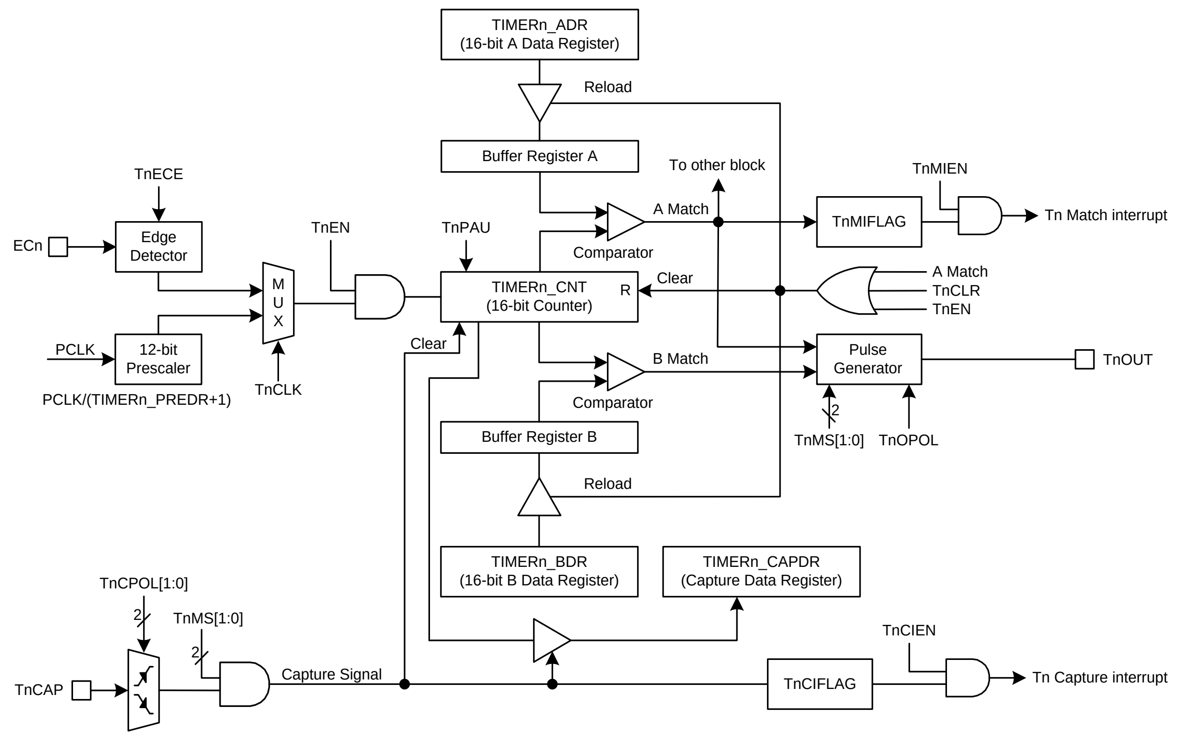
One more optional free-run timer is provided. The main purpose of this timer is a periodical tick timer or a wake-up source. The timer counter 10/11/12/13 features the followings:

- 16-bit up-counter and 12-bit prescaler
- Periodic timer, One-shot timer, PWM pulse, and Capture mode
- Synchronous start and clear function

10.1 Timer counter 10/11/12/13 block diagram

Figure 38 shows the block diagram of a timer block unit.

Figure 38. Timer Counter n Block Diagram (n = 10, 11, 12, and 13)



NOTES:

1. TnEN bit is automatically cleared to logic “0b” after one pulse is generated at a PPG one-shot mode.
2. After the TnEN bit is set, counting of TIMERn_CNT may be delayed 2 clocks or more.

10.2 Pin description for timer counter 10/11/12/13

Table 43. Pins and External Signals for Timer Counter n (n = 10, 11, 12, and 13)

Pin name	Type	Description
ECn	I	External clock input
TnCAP	I	Capture input
TnOUT	O	PWM/one-shot output

10.3 Registers

Base address and register map of the Timer 10/11/12/13 are shown in Table 44 and Table 45.

Table 44. Base Address of Timer 10/11/12/13

Name	Base address	Size	Description
TIMER10	0x4000_2100	128	Timer/Counter 10
TIMER11	0x4000_2180	128	Timer/Counter 11
TIMER12	0x4000_2200	128	Timer/Counter 12
TIMER13	0x4000_2280	128	Timer/Counter 13

Table 45. Timer Register Map (n = 10, 11, 12, and 13)

Name	Offset	Type	Description	Reset value
TIMERn_CR	0x00	RW	Timer/Counter n Control Register	0x00000000
TIMERn_ADR	0x04	RW	Timer/Counter n A Data Register	0x0000FFFF
TIMERn_BDR	0x08	RW	Timer/Counter n B Data Register	0x0000FFFF
TIMERn_CAPDR	0x0C	RO	Timer/Counter n Capture Data Register	0x00000000
TIMERn_PREDR	0x10	RW	Timer/Counter n Prescaler Data Register	0x00000FFF
TIMERn_CNT	0x14	RO	Timer/Counter n Counter Register	0x00000000

10.3.1 TIMERn_CR: timer/counter n control register

Timer module should be configured properly before running. The timer should be configured with the appropriate value in TIMERn_CR register for designated operating mode. After configuring this register, a user can start or stop the timer function by using this register.

TIMERn_CR register is 32-bit size and accessible in 32/16/8-bit (n = 10, 11, 12, and 13).

TIMER10_CR=0x4000_2100, TIMER11_CR=0x4000_2180
TIMER12_CR=0x4000_2200, TIMER13_CR=0x4000_2280

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																TnEN	TnCLK	TnMS		TnECE	Reserved		TnOPOL	TnCPOL		TnMIEN	TnCIEN	TnMIFLAG	TnCIFLAG	TnPAU	TnCLR
0x0000																0	0	00		0	00		0	00		0	0	0	0	0	0
–																RW	RW	RW		RW	-		RW	RW		RW	RW	RW	RW	RW	RW

15	TnEN	Timer n Operation Enable.	
		0	Disable timer n operation.
		1	Enable timer n operation. (Counter clear and start)
14	TnCLK	Timer n Clock Selection.	
		0	Select an internal prescaler clock.
		1	Select an external clock.
		NOTE: This bit should be changed while TnEN bit is '0'.	
13 12	TnMS	Timer n Operation Mode Selection.	
		00	Timer/Counter mode. (TnOUT: toggle at A-match)
		01	Capture mode. (The A-match interrupt can occur)
		10	PPG one-shot mode. (TnOUT: Programmable pulse output)
		11	PPG repeat mode. (TnOUT: Programmable pulse output)
		NOTE: These bits should be changed while TnEN bit is '0'.	
11	TnECE	Timer n External Clock Edge Selection.	
		0	Select falling edge of external clock.
		1	Select rising edge of external clock.
8	TnOPOL	TnOUT Polarity Selection.	
		0	Start high. (TnOUT is low level at disable)
		1	Start low. (TnOUT is high level at disable)
7 6	TnCPOL	Timer n Capture Polarity Selection.	
		00	Capture on falling edge.
		01	Capture on rising edge.
		10	Capture on both falling and rising edge.
		11	Reserved.
5	TnMIEN	Timer n Match Interrupt Enable.	
		0	Disable timer n match interrupt.
		1	Enable timer n match interrupt.
4	TnCIEN	Timer n Capture Interrupt Enable.	
		0	Disable timer n capture interrupt.
		1	Enable timer n capture interrupt.
3	TnMIFLAG	Timer n Match Interrupt Flag.	
		0	No request occurred.
		1	Request occurred. The bit is cleared to '0' when '1' is written.
2	TnCIFLAG	Timer n Capture Interrupt Flag.	
		0	No request occurred.
		1	Request occurred. The bit is cleared to '0' when '1' is written.
1	TnPAU	Timer n Counter Temporary Pause Control.	
		0	Continue counting.
		1	Temporary pause.
0	TnCLR	Timer n Counter and Prescaler Clear.	
		0	No effect.

	1	Clear timer n counter and prescaler. (Automatically cleared to ‘0’ after operation)

10.3.2 TIMERn_ADR: timer/counter n A data register

TIMERn_ADR register is 32-bit size and accessible in 32/16/8-bit (n = 10, 11, 12, and 13).

TIMER10_ADR=0x4000_2104, TIMER11_ADR=0x4000_2184
TIMER12_ADR=0x4000_2204, TIMER13_ADR=0x4000_2284

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		**ADATA**	
0x0000		**0xFFFF**	
-		**RW**	

15 0	ADATA	Timer/Counter n A Data. The range is 0x0002 to 0xFFFF.

NOTE: Do not write “0x0000” in the TIMERn_ADR register under PPG mode.

10.3.3 TIMERn_BDR: timer/counter n B data register

TIMERn_BDR register is 32-bit size and accessible in 32/16/8-bit (n = 10, 11, 12, and 13).

TIMER10_BDR=0x4000_2108, TIMER11_BDR=0x4000_2188
TIMER12_BDR=0x4000_2208, TIMER13_BDR=0x4000_2288

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		**BDATA**	
0x0000		**0xFFFF**	
–		**RW**	

15 0	BDATA	Timer/Counter n B Data. The range is 0x0000 to 0xFFFF.

10.3.4 TIMERn_CAPDR: timer/counter n capture data register

TIMERn_CAPDR register is 32-bit size and accessible in 32/16/8-bit (n = 10, 11, 12, and 13).

TIMER10_CAPDR=0x4000_210C, TIMER11_CAPDR=0x4000_218C
TIMER12_CAPDR=0x4000_220C, TIMER13_CAPDR=0x4000_228C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		**CAPD**	
0x0000		**0x0000**	
–		**RO**	

15 0	CAPD	Timer/Counter n Capture Data.

10.3.5 TIMERn_PREDR: timer/counter n prescaler data register

TIMERn_PREDR register is 32-bit size and accessible in 32/16/8-bit (n = 10, 11, 12, and 13).

TIMER10_PREDR=0x4000_2110, TIMER11_PREDR=0x4000_2190
TIMER12_PREDR=0x4000_2210, TIMER13_PREDR=0x4000_2290

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																				PRED											
0x00000																				0xFFF											
–																				RW											

11 0	PRED	Timer/Counter n Prescaler Data.

10.3.6 TIMERn_CNT: timer/counter n counter register

TIMERn_CNT register is 32-bit size and accessible in 32/16/8-bit (n = 10, 11, 12, and 13).

TIMER10_CNT=0x4000_2114, TIMER11_CNT=0x4000_2194
TIMER12_CNT=0x4000_2214, TIMER13_CNT=0x4000_2294

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																CNT															
0x0000																0x0000															
-																RO															

15 0	CNT	Timer/Counter n Counter.

10.4 Functional description

10.4.1 Timer counter 10/11/12/13

Timer/counter n can use an internal or an external clock source (ECn). A clock selection logic can select a clock source and it is controlled by clock selection bits (TnCLK).

Timer n clock source: {PCLK/(TIMERn_PREDR+1)}, ECn

In Capture mode, by TnCAP, data is captured into a corresponding capture data register (TIMERn_CAPDR). Timer n outputs the comparison result between counter and data register through TnOUT port in Timer/counter mode. Also, timer n output PWM waveform through TnOUT port under PPG mode. (n = 10, 11, 12, and 13)

Table 46. Timer n Operating Modes (n = 10, 11, 12, and 13)

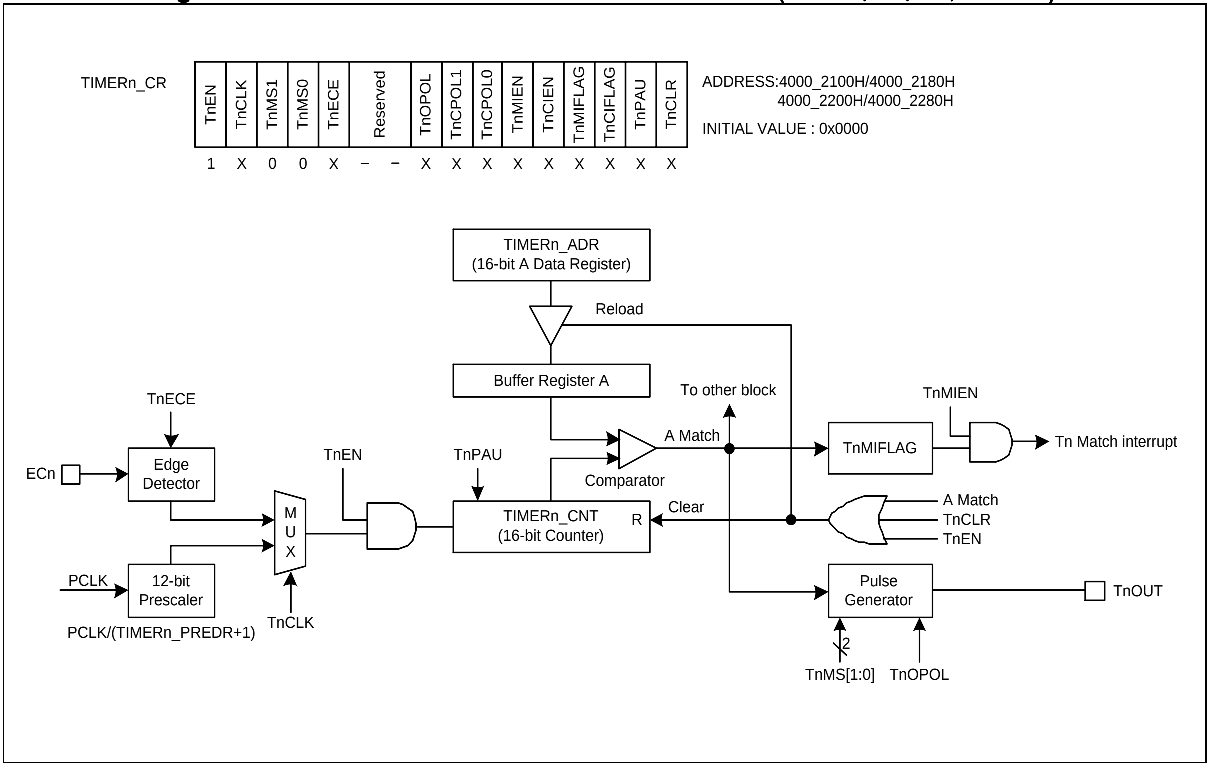
TnEN	Alternative mode	TnMS	TIMERn_PREDR	Timer n
1	TIMER10: T10OUT TIMER11: T11OUT TIMER12: T12OUT TIMER13: T13OUT	00	0xXXX	Timer/Counter Mode
1	TIMER10: T10CAP TIMER11: T11CAP TIMER12: T12CAP TIMER13: T13CAP	01	0xXXX	Capture Mode
1	TIMER10: T10OUT TIMER11: T11OUT TIMER12: T12OUT TIMER13: T13OUT	10	0xXXX	PPG Mode (one-shot mode)
1		11	0xXXX	PPG Mode (repeat mode)

10.4.2 16-bit Timer/counter mode

16-bit Timer/counter mode is selected by control register as shown in Figure 39. The 16-bit timer has a counter register and a data register. The counter register is increased by internal or external clock input. Timer n can use an input clock with 12-bit prescaler division rates (TIMERn_PREDR) and an external clock (ECn). When the values of TIMERn_CNT and TIMERn_ADR are the same in the timer n, a match signal is generated and the interrupt of Timer n takes place.

The TIMERn_CNT values are automatically cleared by the match signal. It can also be cleared by software (TnCLR).

Figure 39. 16-bit Timer/Counter Mode for Timer n (n = 10, 11, 12, and 13)

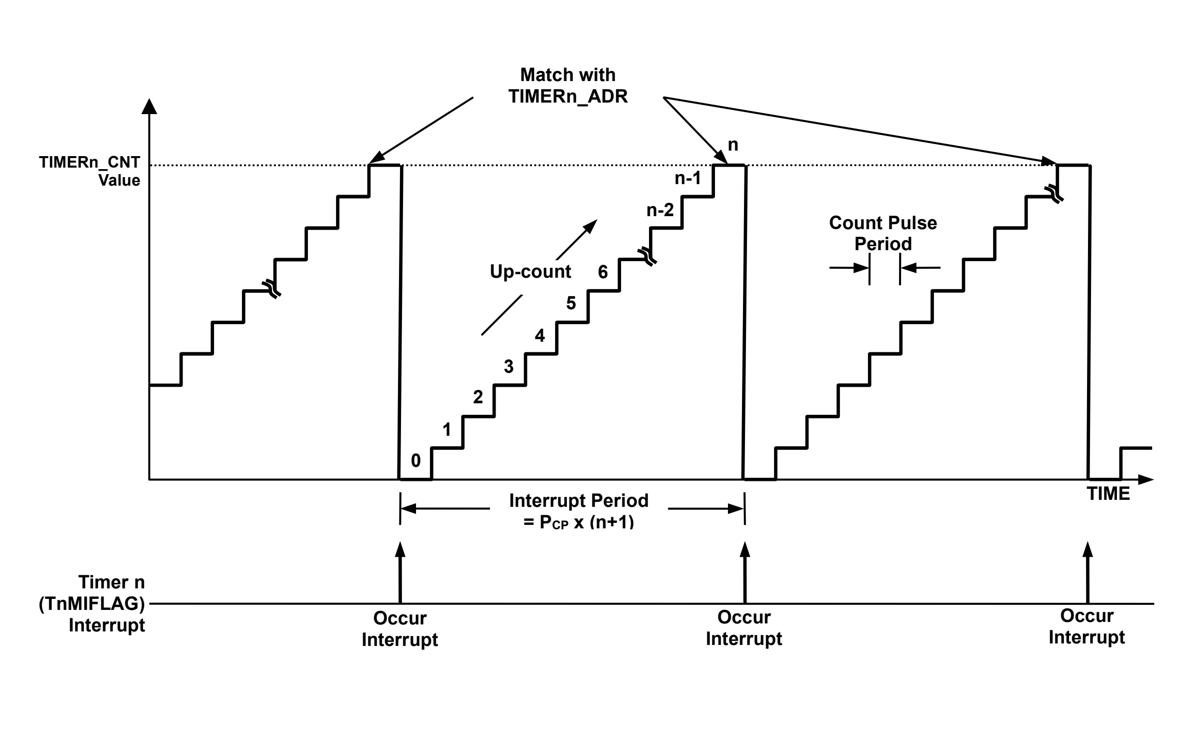


TIMERn_CR

TnEN	TnCLK	TnMS1	TnMS0	TnECE	Reserved		TnOPOL	TnCPOL1	TnCPOL0	TnMIEN	TnCIEN	TnMIFLAG	TnCIFLAG	TnPAU	TnCLR
1	X	0	0	X	–	–	X	X	X	X	X	X	X	X	X

ADDRESS:4000_2100H/4000_2180H
4000_2200H/4000_2280H
INITIAL VALUE : 0x0000

Figure 40. 16-bit Timer/Counter n Example (n = 10, 11, 12, and 13)



10.4.3 16-bit Capture mode

Timer n Capture mode is evoked by configuring TnMS[1:0] as ‘01’. The internal clock can be used as a clock source. It basically has the same function as the 16-bit timer/counter mode and an interrupt takes place when TIMERn_CNT becomes equal to TIMERn_ADR.

This timer interrupt in Capture mode is very useful when the pulse width of captured signal is wider than the maximum period of timer. The capture result is loaded into TIMERn_CAPDR. In the timer n capture mode, timer n output (TnOUT) waveform is not available.

Figure 41. 16-bit Capture Mode for Timer n (n = 10, 11, 12, and 13)

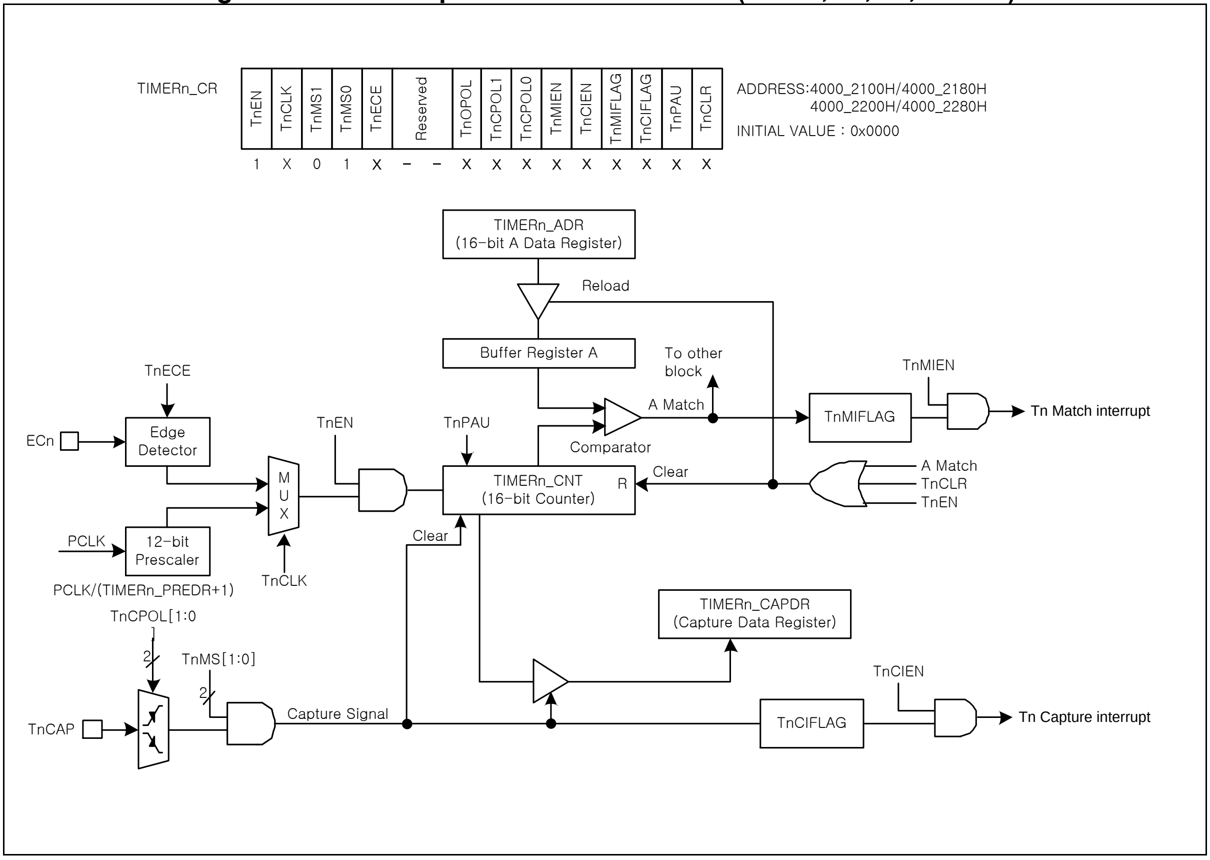


Figure 42. 16-bit Capture Mode for Timer n (n = 10, 11, 12, and 13)

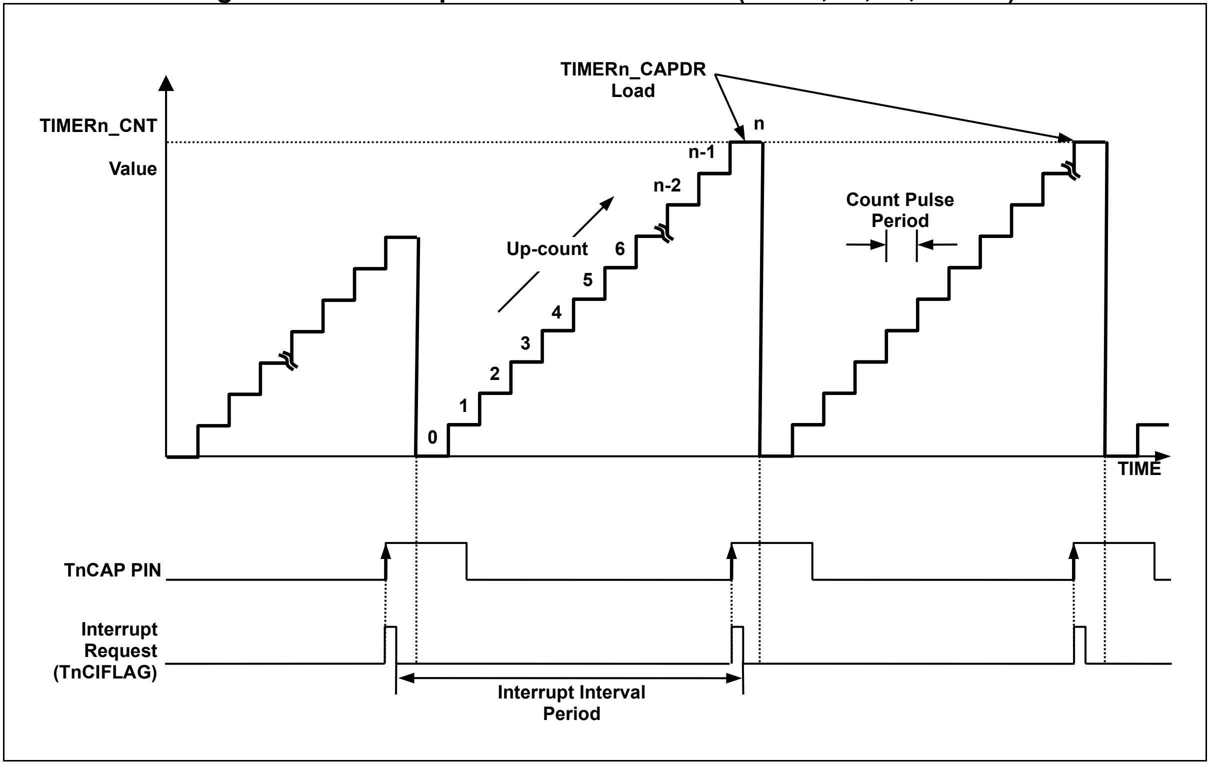
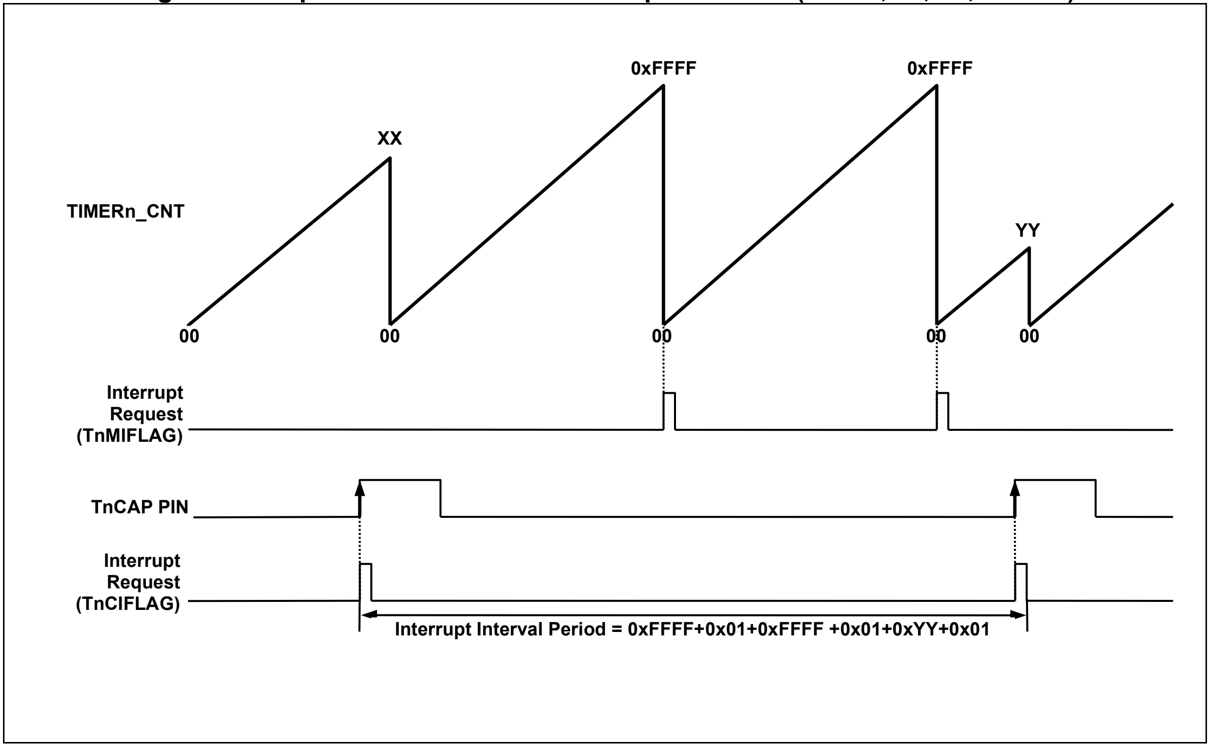


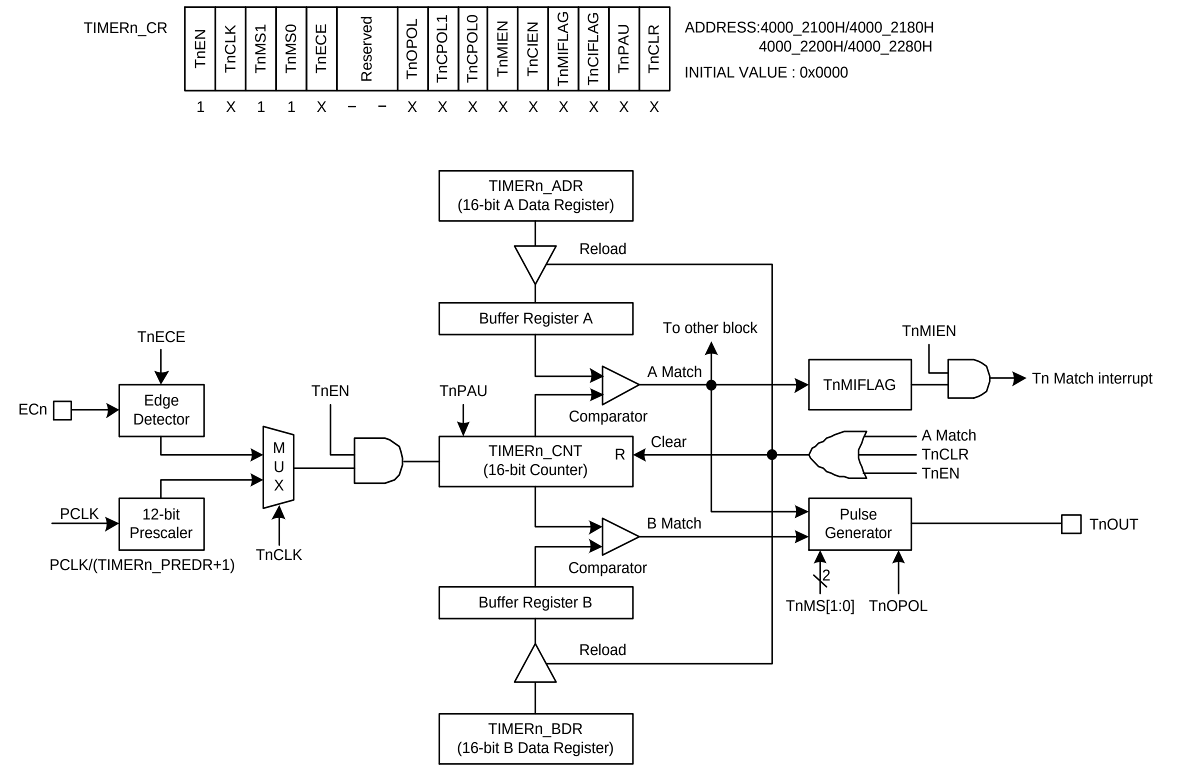
Figure 43. Express Timer Overflow in Capture Mode (n = 10, 11, 12, and 13)



10.4.4 16-bit PPG mode

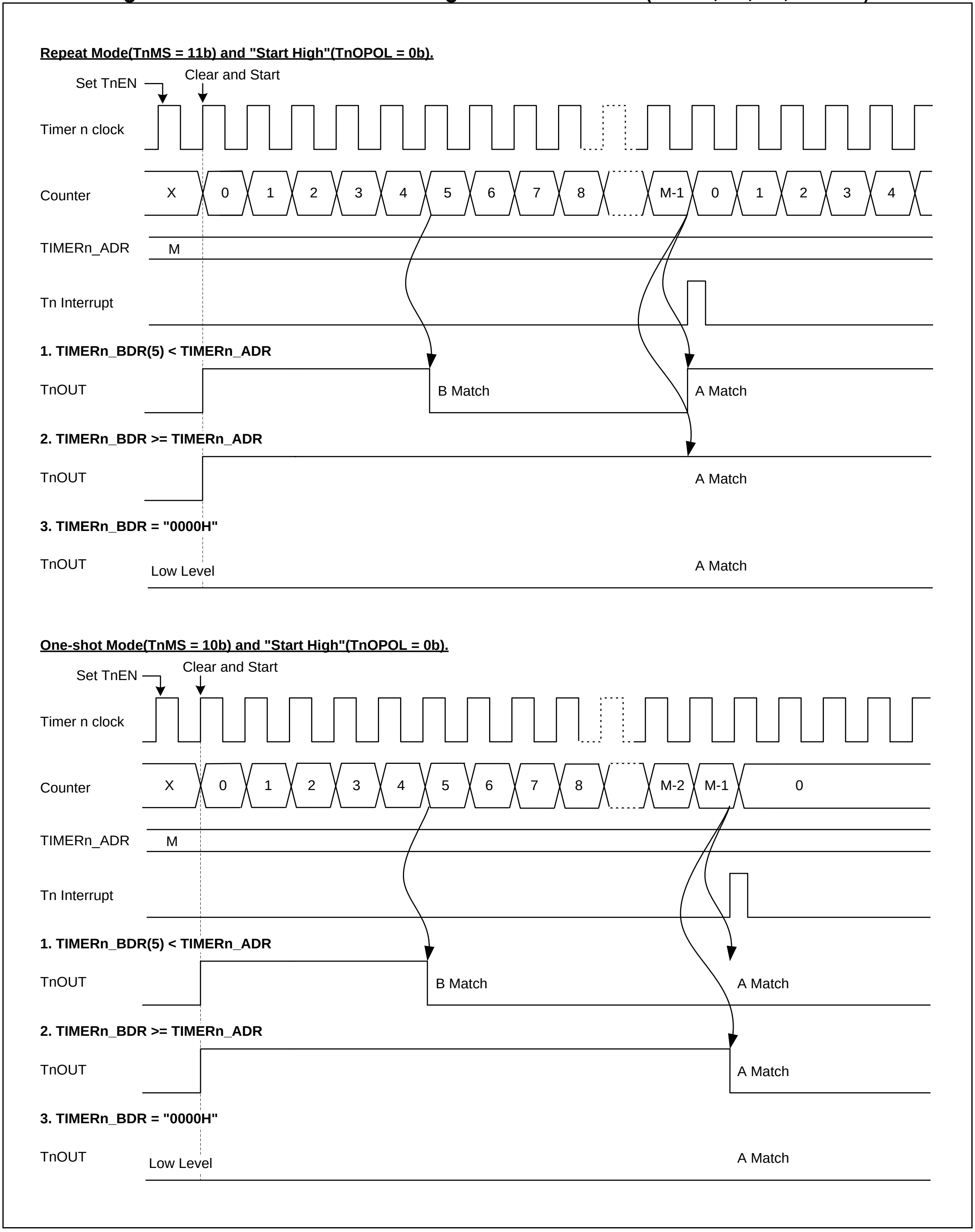
Timer n has a PPG (Programmable Pulse Generation) function. In PPG mode, the TnOUT pin generates PWM output of up to 16-bit resolution. This pin should be configured as TnOUT function in the Px_AFSR1/Px_AFSR2 for PWM output. The period of PWM output is determined by the TIMERn_ADR. The duty of PWM output is determined by the TIMERn_BDR. (x = A to F)

Figure 44. 16-bit PPG Repeat and One-shot Mode for Timer n (n = 10, 11, 12 and 13)



NOTE: The TnEN is automatically cleared to logic ‘0’ after one pulse is generated in PPG one-shot mode.

Figure 45. 16-bit PPG Mode Timing Chart for Timer n (n = 10, 11, 12, and 13)



11 Timer Counter 40/41/42/43

Each of Timer counter 40/41/42/43 is a 16-bit general purpose timer with two outputs. It has an independent 16-bit counter and a dedicated prescaler that feeds counting clock. It supports periodic timer, PWM pulse, one-shot and capture mode.

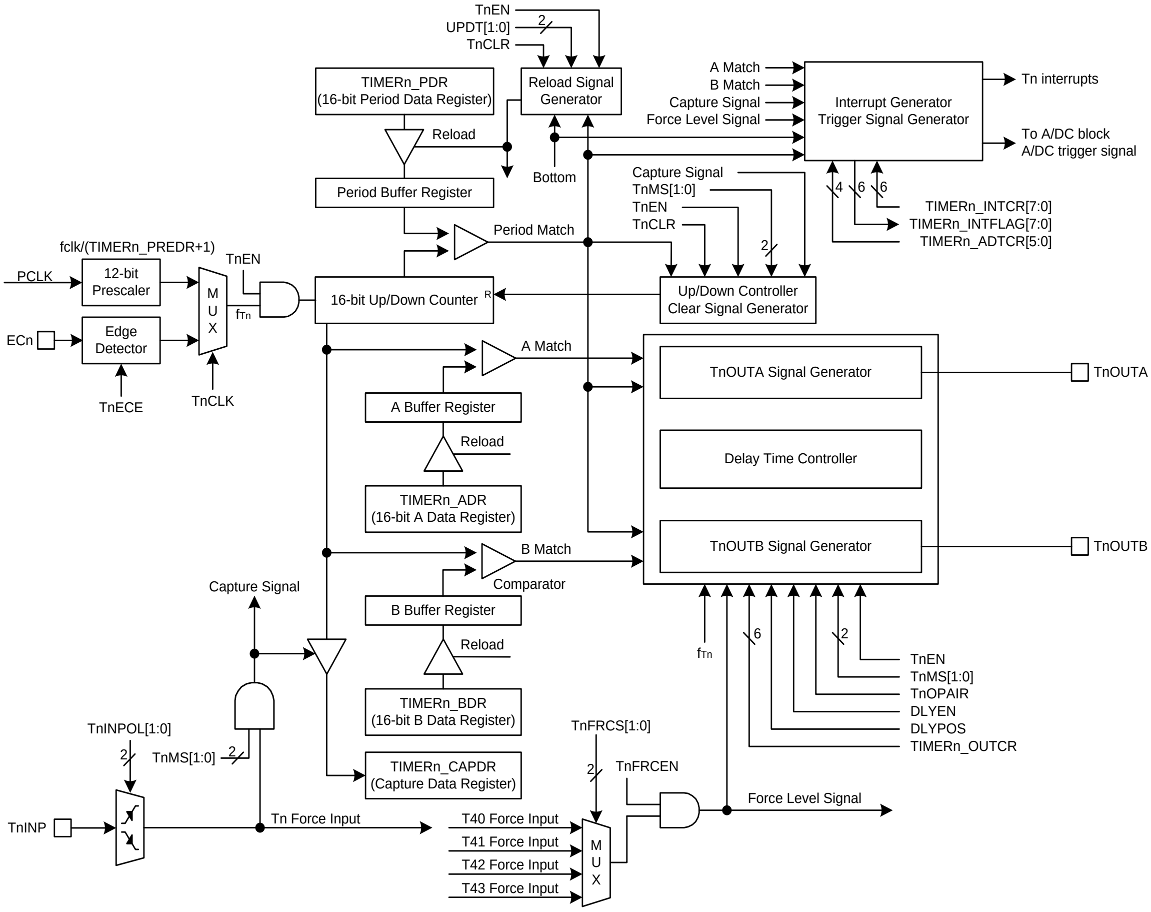
Main purpose of this timer is to work as a periodical tick timer or to provide a wake-up source. The operations of Timer counter 40/41/42/43 are listed in the followings:

- 12-bit prescaler and 16-bit up-counter
- Interval timer, One-shot timer, Back-to-back, and Capture mode
- Counter sharing function to connect each other
- Synchronous start and clear function

11.1 Timer counter 40/41/42/43 block diagram

Figure 46 shows the block diagram of a timer block unit.

Figure 46. Timer Counter n Block Diagram (n = 40, 41, 42, and 43)



NOTES:

1. Period/A/B match interrupts except for bottom can occur in the interval mode.
2. Period match interrupt only can occur in the capture mode.
3. 16-bit up/down counter in the capture mode is cleared to “0000H” after the counter value is loaded to the TIMERn_CAPDR register at a valid edge.
4. TnEN bit is automatically cleared to logic “0b” after one pulse is generated at a one-shot interval mode.
5. After the TnEN bit is set, the counting of TIMERn_CNT may be delayed 2 clocks or more.
6. High/low width of ECn clock should be longer than PCLK clock period.

11.2 Pin description for timer counter 40/41/42/43

Table 47. Pins and External Signals for Timer Counter n (n = 40, 41, 42, and 43)

Pin name	Type	Description
ECn	I	External clock input
TnINP	I	Capture or force input
TnOUTA	O	Timer A output
TnOUTB	O	Timer B output

11.3 Registers

Base address and register map of the Timer 40/41/42/43 are shown in Table 48 and Table 49.

Table 48. Base Address of Timer 40/41/42/43

Name	Base address	Size	Description
TIMER40	0x4000_2700	128	Timer/Counter 40
TIMER41	0x4000_2780	128	Timer/Counter 41
TIMER42	0x4000_2800	128	Timer/Counter 42
TIMER43	0x4000_2880	128	Timer/Counter 43

Table 49. Timer Register Map (n = 40, 41, 42, and 43)

Name	Offset	Type	Description	Reset value
TIMERn_CR	0x00	RW	Timer/Counter n Control Register	0x00000000
TIMERn_PDR	0x04	RW	Timer/Counter n Period Data Register	0x0000FFFF
TIMERn_ADR	0x08	RW	Timer/Counter n A Data Register	0x0000FFFF
TIMERn_BDR	0x0C	RW	Timer/Counter n B Data Register	0x0000FFFF
TIMERn_CAPDR	0x10	RO	Timer/Counter n Capture Data Register	0x00000000
TIMERn_PREDR	0x14	RW	Timer/Counter n Prescaler Data Register	0x00000FFF
TIMERn_CNT	0x18	RO	Timer/Counter n Counter Register	0x00000000
TIMERn_OUTCR	0x1C	RW	Timer/Counter n Output Control Register	0x00000000
TIMERn_DLY	0x20	RW	Timer/Counter n Output Delay Data Register	0x00000000
TIMERn_INTCR	0x24	RW	Timer/Counter n Interrupt Control Register	0x00000000
TIMERn_INTFLAG	0x28	RW	Timer/Counter n Interrupt Flag Register	0x00000000
TIMERn_ADTCR	0x2C	RW	Timer/Counter n ADC Trigger Control Register	0x00000000

11.3.1 TIMERn_CR: timer/counter n control register

Timer module should be configured properly before running. Once target purpose is defined, the timer can be configured in the TIMERn_CR register. After configuring this register, a user can start or stop the timer function by using this register.

TIMERn_CR register is 32-bit size and accessible in 32/16/8-bit (n = 40, 41, 42, and 43).

TIMERn_CR=0x4000_2700, TIMERn_CR=0x4000_2780
TIMERn_CR=0x4000_2800, TIMERn_CR=0x4000_2880

31 30 29 28 27 26 25 24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	TnFRCEN	Reserved	TnFRCS		CNTSHEN	Reserved	CNTSH		TnEN	TnCLK	TnMS		TnECE	TnOPAIR	DLYEN	DLYPOS	UPDT		TnINPOL		Reserved		TnPAU	TnCLR
0x00	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0
–	RW	-	RW	RW	RW	-	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	-	-	RW	RW

23	TnFRCEN	Timer n Output Force Level Enable.	
		0	Disable output force level.
		1	Enable output force level during the valid level of the selected Tn force input.
		NOTE: The output force level depends on the LVLA and LVLB bits.	
21 20	TnFRCS	Timer n Force Input Selection.	
		00	T40 force input.
		01	T41 force input.
		10	T42 force input.
		11	T43 force input.
19	CNTSHEN	Timer Counter Sharing Enable.	
		0	Disable counter sharing.
		1	Enable counter sharing
17 16	CNTSH	Timer Counter Sharing Selection.	
		00	Timer n uses timer 40's counter instead of itself.
		01	Timer n uses timer 41's counter instead of itself.
		10	Timer n uses timer 42's counter instead of itself.
		11	Timer n uses timer 43's counter instead of itself.
		NOTES: 1. When using the timer's own counter, timer counter sharing function should be disabled by CNTSHEN = "0b". 2. When using the counter sharing, the TnMS[1:0], UPDT[1:0], and TIMERn_PDR of the sharing timers must have the same value, respectively. That is, the same value must be written to the registers of the sharing timers. 3. When using of the counter sharing, the TIMERn_PDR of the master is copied to the TIMERn_PDR of the slave and can't be written to the slave's TIMERn_PDR.	
15	TnEN	Timer n Operation Enable.	
		0	Disable timer n operation.
		1	Enable timer n operation (Counter clear and start)
14	TnCLK	Timer n Clock Selection.	
		0	Select an internal prescaler clock.
		1	Select an external clock.
		NOTE: This bit should be changed while TnEN bit is '0'.	
13 12	TnMS	Timer n Operation Mode Selection.	
		00	Interval mode. (All match interrupts can occur)
		01	Capture mode. (The Period-match interrupt can occur)
		10	Back-to-back mode. (All match and bottom interrupts can occur)
		11	One-shot interval mode (All match interrupts can occur)
		NOTE: These bits should be changed during TnEN bit is '0'.	
11	TnECE	Timer n External Clock Edge Selection.	
		0	Select falling edge of external clock.

		1	Select rising edge of external clock.
10	TnOPAIR	Timer n Output Pair Selection	
		0	No output pair
		1	Output pair (The TnOUTB signal depends on TIMERn_ADR register)
9	DLYEN	Delay Time Insertion Enable. This bit is effective on the TnOPAIR = ‘1’.	
		0	Disable to insert delay time to the TnOUTA/TnOUTB.
		1	Enable to insert delay time to the TnOUTA/TnOUTB.
		NOTE:	The “delay time insertion” doesn’t work in one-shot interval mode.
8	DLYPOS	Delay Time Insertion Position.	
		0	Insert at front of TnOUTA and at back of TnOUTB pins.
		1	Insert at back of TnOUTA and at front of TnOUTB pins.
7	UPDT	Data Reload Time Selection.	
6		00	Update data to buffer at the time of writing.
		01	Update data to buffer at period match.
		10	Update data to buffer at bottom.
		11	Not used
5	TnINPOL	Timer n Input Capture/”Force level” Polarity Selection.	
4		00	Capture on falling edge, Force level on low level.
		01	Capture on rising edge, Force level on high level.
		10	Capture on both of falling and rising edge, Not available for force level.
		11	Reserved
		NOTE:	The counter of timer n is cleared to 0x0000 at valid edge in capture mode.
1	TnPAU	Timer n Counter Temporary Pause Control.	
		0	Continue counting.
		1	Temporary pause
0	TnCLR	Timer n Counter and Prescaler Clear.	
		0	No effect.
		1	Clear timer n counter and prescaler (Automatically cleared to ‘0’ after operation)

11.3.2 TIMERn_PDR: timer/counter n period data register

TIMERn_PDR register is 32-bit size and accessible in 32/16/8-bit (n = 40, 41, 42 and 43).

TIMER40_PDR=0x4000_2704, TIMER41_PDR=0x4000_2784
TIMER42_PDR=0x4000_2804, TIMER43_PDR=0x4000_2884

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		PDATA	
0x0000		0xFFFF	
–		RW	

15 0	PDATA	Timer/Counter n Period Data. The range is 0x0002 to 0xFFFF. Period match time: (PDATA[15:0]+1)÷fTn

11.3.3 TIMERn_ADR: timer/counter n A data register

TIMERn_ADR register is 32-bit size and accessible in 32/16/8-bit (n = 40, 41, 42 and 43).

TIMER40_ADR=0x4000_2708, TIMER41_ADR=0x4000_2788
TIMER42_ADR=0x4000_2808, TIMER43_ADR=0x4000_2888

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		ADATA	
0x0000		0xFFFF	
-		RW	

15 0	ADATA	Timer/Counter n A Data. The range is 0x0000 to 0xFFFF. A match time: (ADATA[15:0])÷fTn

11.3.4 TIMERn_BDR: timer/counter n B data register

TIMERn_BDR register is 32-bit size and accessible in 32/16/8-bit (n = 40, 41, 42 and 43).

TIMER40_BDR=0x4000_270C, TIMER41_BDR=0x4000_278C
TIMER42_BDR=0x4000_280C, TIMER43_BDR=0x4000_288C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		BDATA	
0x0000		0xFFFF	
–		RW	

15 0	BDATA	Timer/Counter n B Data. The range is 0x0000 to 0xFFFF. B match time: (BDATA[15:0])÷fTn

11.3.5 TIMERn_CAPDR: timer/counter n capture data register

TIMERn_CAPDR register is 32-bit size and accessible in 32/16/8-bit (n = 40, 41, 42 and 43).

TIMER40_CAPDR=0x4000_2710, TIMER41_CAPDR=0x4000_2790
TIMER42_CAPDR=0x4000_2810, TIMER43_CAPDR=0x4000_2890

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="16">Reserved</td><td colspan="16">CAPD</td></tr>
<tr><td colspan="16">0x0000</td><td colspan="16">0x0000</td></tr>
<tr><td colspan="16">–</td><td colspan="16">RO</td></tr>
</table>

15 0	CAPD	Timer/Counter n Capture Data.

11.3.6 TIMERn_PREDR: timer/counter n prescaler data register

TIMERn_PREDR register is 32-bit size and accessible in 32/16/8-bit (n = 40, 41, 42 and 43).

TIMER40_PREDR=0x4000_2714, TIMER41_PREDR=0x4000_2794
TIMER42_PREDR=0x4000_2814, TIMER43_PREDR=0x4000_2894

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="20">Reserved</td><td colspan="12">PRED</td></tr>
<tr><td colspan="20">0x00000</td><td colspan="12">0xFFF</td></tr>
<tr><td colspan="20">–</td><td colspan="12">RW</td></tr>
</table>

11 0	PRED	Timer/Counter n Prescaler Data.

11.3.7 TIMERn_CNT: timer/counter n counter register

TIMERn_CNT register is 32-bit size and accessible in 32/16/8-bit (n = 40, 41, 42 and 43).

TIMER40_CNT=0x4000_2718, TIMER41_CNT=0x4000_2798
TIMER42_CNT=0x4000_2818, TIMER43_CNT=0x4000_2898

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="16">Reserved</td><td colspan="16">CNT</td></tr>
<tr><td colspan="16">0x0000</td><td colspan="16">0x0000</td></tr>
<tr><td colspan="16">-</td><td colspan="16">RO</td></tr>
</table>

15 0	CNT	Timer/Counter n Counter.

11.3.8 TIMERn_OUTCR: timer/counter n output control register

TIMERn_OUTCR register is 32-bit size and accessible in 32/16/8-bit (n = 40, 41, 42 and 43).

TIMER40_OUTCR=0x4000_271C, TIMER41_OUTCR=0x4000_279C
TIMER42_OUTCR=0x4000_281C, TIMER43_OUTCR=0x4000_289C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																Reserved						POLB	POLA	Reserved		TnBOE	TnAOE	Reserved		LVLB	LVLA
0x0000																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
–																-	-	-	-	-	-	RW	RW	-	-	RW	RW	-	-	RW	RW

9	POLB	TnOUTB Output Polarity Selection.	
		0	Low level start (The TnOUTB pin is started with low level after counting)
		1	High level start (The TnOUTB pin is started with high level after counting)
8	POLA	TnOUTA Output Polarity Selection.	
		0	Low level start (The TnOUTA pins are started with low level after counting)
		1	High level start (The TnOUTA pins are started with high level after counting)
5	TnBOE	TnOUTB Output Enable.	
		0	Disable output.
		1	Enable output.
4	TnAOE	TnOUTA Output Enable.	
		0	Disable output.
		1	Enable output.
1	LVLB	Configure TnOUTB output When Disable.	
		0	Low level
		1	High level
0	LVLA	Configure TnOUTA output When Disable.	
		0	Low level
		1	High level

11.3.9 TIMERn_DLY timer/counter n output delay data register

TIMERn_DLY register is 32-bit size and accessible in 32/16/8-bit (n = 40, 41, 42 and 43).

TIMER40_DLY=0x4000_2720, TIMER41_DLY=0x4000_27A0
TIMER42_DLY=0x4000_2820, TIMER43_DLY=0x4000_28A0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																						DLY									
0x00000																						0x000									
–																						RW									

9	DLY	Timer/Counter n Output Delay Data.
0		Delay time: (DLY[9:0]+1)÷fTn

11.3.10 TIMERn_INTCR: timer/counter n interrupt control register

TIMERn_INTCR register is 32-bit size and accessible in 32/16/8-bit (n = 40, 41, 42 and 43).

TIMER40_INTCR=0x4000_2724, TIMER41_INTCR=0x4000_27A4
TIMER42_INTCR=0x4000_2824, TIMER43_INTCR=0x4000_28A4

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																				TnFRCIEN	TnCIEN	TnBTIEN	TnPMIEN	Reserved				TnBMIEN		TnAMIEN	
0x00000																				0	0	0	0	0	0	0	0	0	0	0	0
–																				RW	RW	RW	RW	-	-	-	-	RW	RW	RW	RW

11	TnFRCIEN	Timer n Output Force Level Interrupt Enable.	
		0	Disable timer n output hold interrupt.
		1	Enable timer n output hold interrupt.
10	TnCIEN	Timer n Capture Interrupt Enable.	
		0	Disable timer n capture interrupt.
		1	Enable timer n capture interrupt.
9	TnBTIEN	Timer n Bottom Interrupt Enable.	
		0	Disable timer n bottom interrupt.
		1	Enable timer n bottom interrupt.
8	TnPMIEN	Timer n Period Match Interrupt Enable.	
		0	Disable timer n period interrupt.
		1	Enable timer n period interrupt.
3 2	TnBMIEN	Timer n B Match Interrupt Enable.	
		00	Disable B match interrupt.
		01	Enable B match interrupt on up counting.
		10	Disable B match interrupt on down counting.
		11	Disable B match interrupt on up and down counting.
1 0	TnAMIEN	Timer n A Match Interrupt Enable.	
		00	Disable A match interrupt.
		01	Enable A match interrupt on up counting.
		10	Disable A match interrupt on down counting.
		11	Disable A match interrupt on up and down counting.

11.3.11 TIMERn_INTFLAG: timer/counter n interrupt flag register

TIMERn_INTFLAG register is 32-bit size and accessible in 32/16/8-bit (n = 40, 41, 42 and 43).

TIMER40_INTFLAG=0x4000_2728, TIMER41_INTFLAG=0x4000_27A8
TIMER42_INTFLAG=0x4000_2828, TIMER43_INTFLAG=0x4000_28A8

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
Reserved			TnFRCIFLAG	TnCIFLAG	TnBTIFLAG	TnPMIFLAG	Reserved		TnBMIFLAG	TnAMIFLAG
0x000000			0	0	0	0	0	0	0	0
–			RW	RW	RW	RW	-	-	RW	RW

7	TnFRCIFLAG	Timer n Output Force Level Interrupt Flag.	
		0	No request occurred.
		1	Request occurred. The bit will be cleared to ‘0’ when ‘1’ is written to this bit.
6	TnCIFLAG	Timer n Capture Interrupt Flag.	
		0	No request occurred.
		1	Request occurred. The bit will be cleared to ‘0’ when ‘1’ is written to this bit.
5	TnBTIFLAG	Timer n Bottom Interrupt Flag bit. This bit is effective only at back-to-back mode.	
		0	No request occurred.
		1	Request occurred. The bit will be cleared to ‘0’ when ‘1’ is written to this bit.
4	TnPMIFLAG	Timer n Period Match Interrupt Flag.	
		0	No request occurred.
		1	Request occurred. The bit will be cleared to ‘0’ when ‘1’ is written to this bit.
1	TnBMIFLAG	Timer n B Match Interrupt Flag.	
		0	No request occurred.
		1	Request occurred. The bit will be cleared to ‘0’ when ‘1’ is written to this bit.
0	TnAMIFLAG	Timer n A Match Interrupt Flag.	
		0	No request occurred.
		1	Request occurred. The bit will be cleared to ‘0’ when ‘1’ is written to this bit.

11.3.12 TIMERn_ADTCR: timer/counter n ADC trigger control register

TIMERn_ADTCR register is 32-bit size and accessible in 32/16/8-bit (n = 40, 41, 42 and 43).

TIMER40_ADTCR=0x4000_272C, TIMER41_ADTCR=0x4000_27AC
TIMER42_ADTCR=0x4000_282C, TIMER43_ADTCR=0x4000_28AC

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																						TnBTTG	TnPMTG	Reserved				TnBMTG		TnAMTG	
0x000000																						0	0	0	0	0	0	0	0	0	0
–																						RW	RW	-	-	-	-	RW	RW	RW	RW

9	TnBTTG	Select Timer n Bottom for ADC Trigger Signal Generator.	
		0	Disable ADC trigger signal generator by bottom.
		1	Enable ADC trigger signal generator by bottom.
8	TnPMTG	Select Timer n Period Match for ADC Trigger Signal Generator.	
		0	Disable ADC trigger signal generator by period match.
		1	Enable ADC trigger signal generator by period match.
3	TnBMTG	Select Timer n B Match for ADC Trigger Signal Generator.	
2		00	Disable ADC trigger signal generator by B match.
		01	Enable ADC trigger signal generator by B match on up counting
		10	Enable ADC trigger signal generator by B match on down counting
		11	Enable ADC trigger signal generator by B match on up and down counting.
1	TnAMTG	Select Timer n A Match for ADC Trigger Signal Generator.	
0		00	Disable ADC trigger signal generator by A match.
		01	Enable ADC trigger signal generator by A match on up counting
		10	Enable ADC trigger signal generator by A match on down counting.
		11	Enable ADC trigger signal generator by A match on up and down counting.

11.4 Functional description

11.4.1 Timer counter 40/41/42/43

Timer/counter n can use an internal or an external clock source (ECn). A clock selection logic can select a clock source and it is controlled by clock selection bits (TnCLK).

Timer n clock source: {PCLK/(TIMERn_PREDR+1)}, ECn

In Capture mode, by TnINP, data is captured into a corresponding capture data register (TIMERn_CAPDR). Timer n outputs the comparison result between counter and data register through TnOUTA/TnOUTB ports in Timer/counter and Back-to-back mode.

The outputs, TnOUTA/TnOUTB, can be forced to a fixed level during an external force input signal by hardware when TnFRCEN=1. (n = 40, 41, 42 and 43)

Table 50. Timer n Operating Modes (n = 40, 41, 42, and 43)

TnEN	TnMS	Timer n
1	00	16-bit Interval mode
1	01	16-bit Capture mode
1	10	16-bit back-to-back mode
1	11	16-bit one-shot interval mode

11.4.2 Timer 40/41/42/43 Capture mode

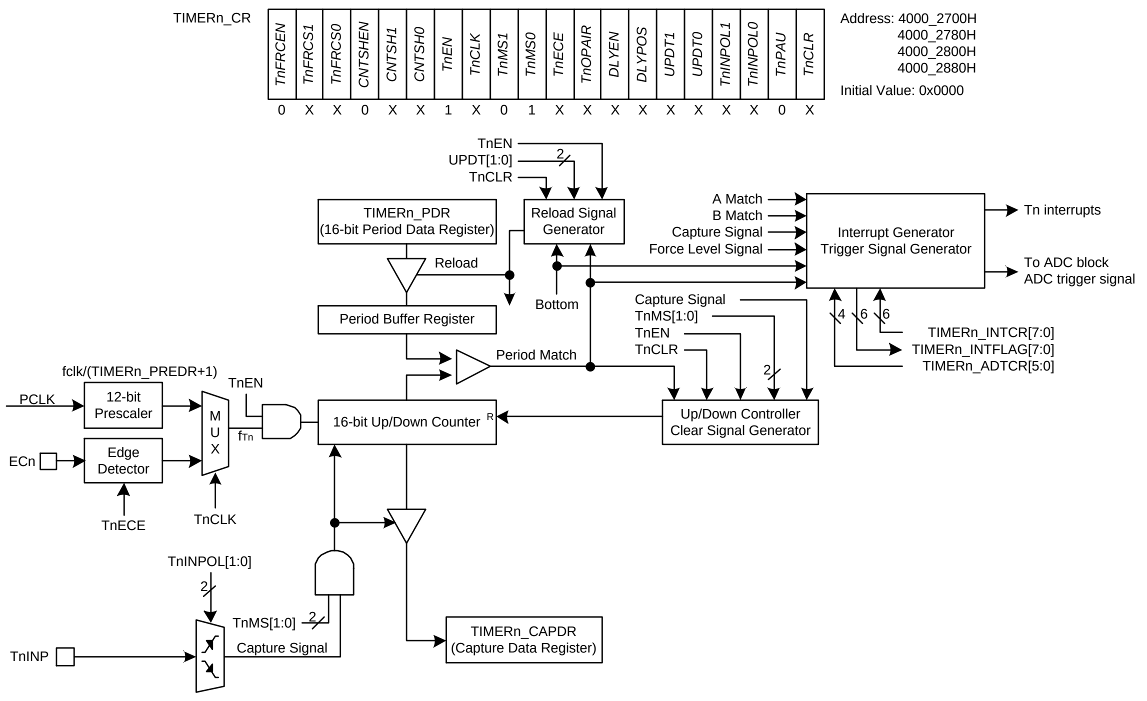
16-bit timer capture mode is set by configuring the TnMS[1:0] as ‘01’. The clock source can use internal or external clock input.

This 16-bit timer capture mode basically has the same function as the 16-bit interval mode. An interrupt takes place when the 16-bit up/down counter and the TIMERn_PDR have the same values. The 16-bit up/down counter value is automatically cleared by a match signal. It can also be cleared by software (TnCLR).

A timer interrupt in Capture mode is very useful when the pulse width of captured signal is wider than the maximum period of timer.

The capture result is loaded into TIMERn_CAPDR. (n = 40, 41, 42 and 43)

Figure 47. 16-bit Capture Mode for Timer 40/41/42/43



NOTES:

1. The Period match only can occur in capture mode.
2. The 16-bit up/down counter in capture mode is cleared to “0x0000” after the counter value is loaded to the TIMERn_CAPDR register at valid edge.

11.4.3 Timer 40/41/42/43 Interval mode

Interval mode is set by configuring the TnMS[1:0] as ‘00’. Each of Timer 40/41/42/43 has a counter and data registers.

The 16-bit up/down counter is increased by internal or external clock input. The timer can use an input clock with 12-bit prescaler division rates (TIMERn_PREDR[11:0]). When the values of 16-bit up/down counter and the TIMERn_PDR are the same in the timer, a match signal is generated and the period match interrupt of the timer is occurred. The 16-bit up/down counter value is automatically cleared by the match signal. It can also be cleared by software (TnCLR).

The timer has 2-channel pins that generate PWM outputs of up to 16-bit resolution. The match signals and interrupts of period/A/B can be generated when the 16-bit counter value is the same as the value of TIMERn_PDR, TIMERn_ADR, and TIMERn_BDR, respectively. The period and duty of the output is determined by the TIMERn_PDR (period register), TIMERn_ADR (A channel duty register), and TIMERn_BDR (B channel duty register).

TnOUTA and TnOUTB’s Period = [TIMERn_PDR + 1] X Source Clock
TnOUTA Duty = [TIMERn_ADR] X Source Clock
TnOUTB Duty = [TIMERn_BDR] X Source Clock

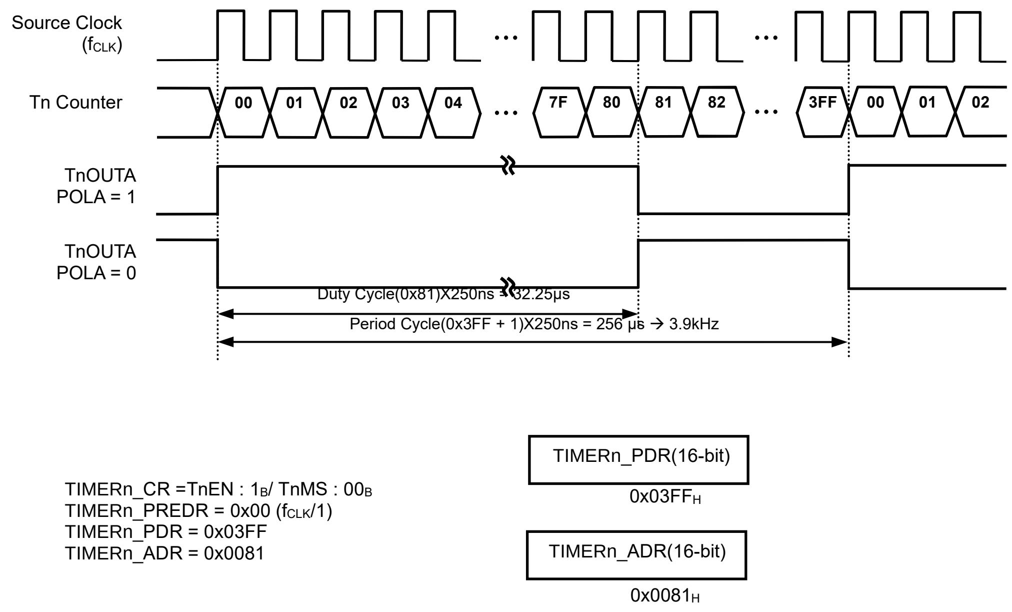
POLA/POLB bit of TIMERn_OUTCR register decides the polarity of output. If the POLA/POLB bit is set to ‘1’, the TnOUTA/TnOUTB output is high level start and if the POLA/POLB bit is cleared to ‘0’, the TnOUTA/TnOUTB output is low level start, respectively. (n = 40, 41, 42 and 43)

Table 51. TnOUTA/B Channel Polarity

PnAOE	PnBOE	POLA	POLB	TnOUTA Pin Output	PnOUTB Pin Output
1	1	0	0	Low level start	Low level start
		0	1	Low level start	High level start
		1	0	High level start	Low level start
		1	1	High level start	High level start

NOTE: Where n = 40, 41, 42 and 43.

Figure 48. Example of TnOUTA at 4MHz (n = 40, 41, 42 and 43)



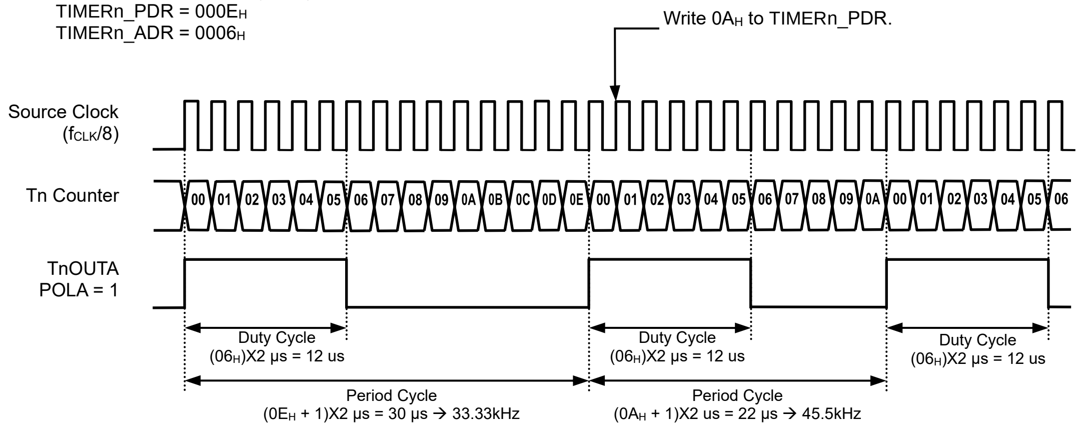
TIMERn_CR =TnEN : 1_B/ TnMS : 00_B
TIMERn_PREDR = 0x00 (f_{CLK}/1)
TIMERn_PDR = 0x03FF
TIMERn_ADR = 0x0081

11.4.3.1 Data reload time selection

Data reload time can be selected from “update data to buffer at the time of writing”, “update data to buffer at period match”, or “update data to buffer at bottom”. The UPDT[1:0] bits of TIMERn_CR register is used to select the data reload time to upload into buffer.

Figure 49. Example of Changing the Period in Absolute Duty Cycle at 4MHz (n = 40, 41, 42 and 43)

TIMERn_CR =TnEN : 1_B/ TnMS : 00_B
TIMERn_PREDR = 07_H (f_{CLK}/8)
TIMERn_PDR = $000E_H$
TIMERn_ADR = 0006_H



NOTE: This example timing wave shows the data reload time is chosen as the time of writing mode (UPDT[1:0] = “00”).

11.4.3.2 Timer output delay

Using the DLYEN bit, DLYPOS bit, the TIMERn_DLY register can delay the PWM output. When DLYPOS is set to ‘0’, the delay is inserted in front of TnOUTA and behind TnOUTB pins.

When DLYPOS is set to ‘1’, the delay is inserted behind TnOUTA and in front of TnOUTB pins.

Figure 50 and Figure 51 show example timing waveforms. (n = 40, 41, 42 and 43)

11.4.3.3 Output force level on the TnINP input

This is used to maintain the TnOUTA and the TnOUTB inactive level under overload condition. The output level of TnOUTA and TnOUTB can be driven to the levels selected by LVLB and LVLA bits during the input signal selected by TnFRCS[1:0] and TnINPOL[1:0] bits when TnFRCEN=1. The output signal remains at the selected level until the next cycle.

The TnFRCS[1:0] bits select an input pin for a given channel and the TnINPOL[1:0] bits select the valid level of input signal. As an example, see b of item 2 in Figure 50 and Figure 51 (n = 40, 41, 42 and 43).

Figure 50. Interval Mode Timing Chart With “DLYPOS = 0” (n = 40, 41, 42 and 43)

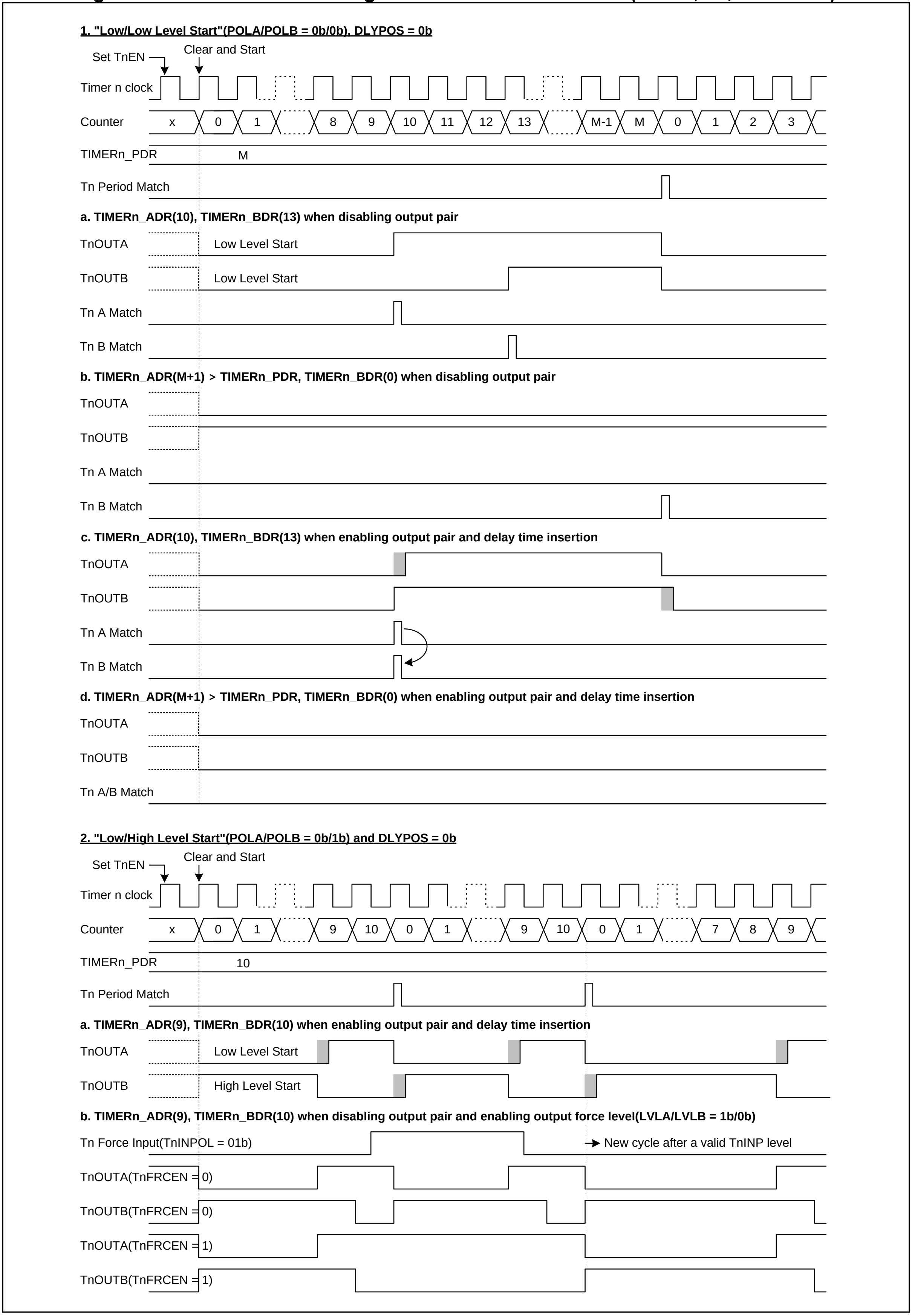
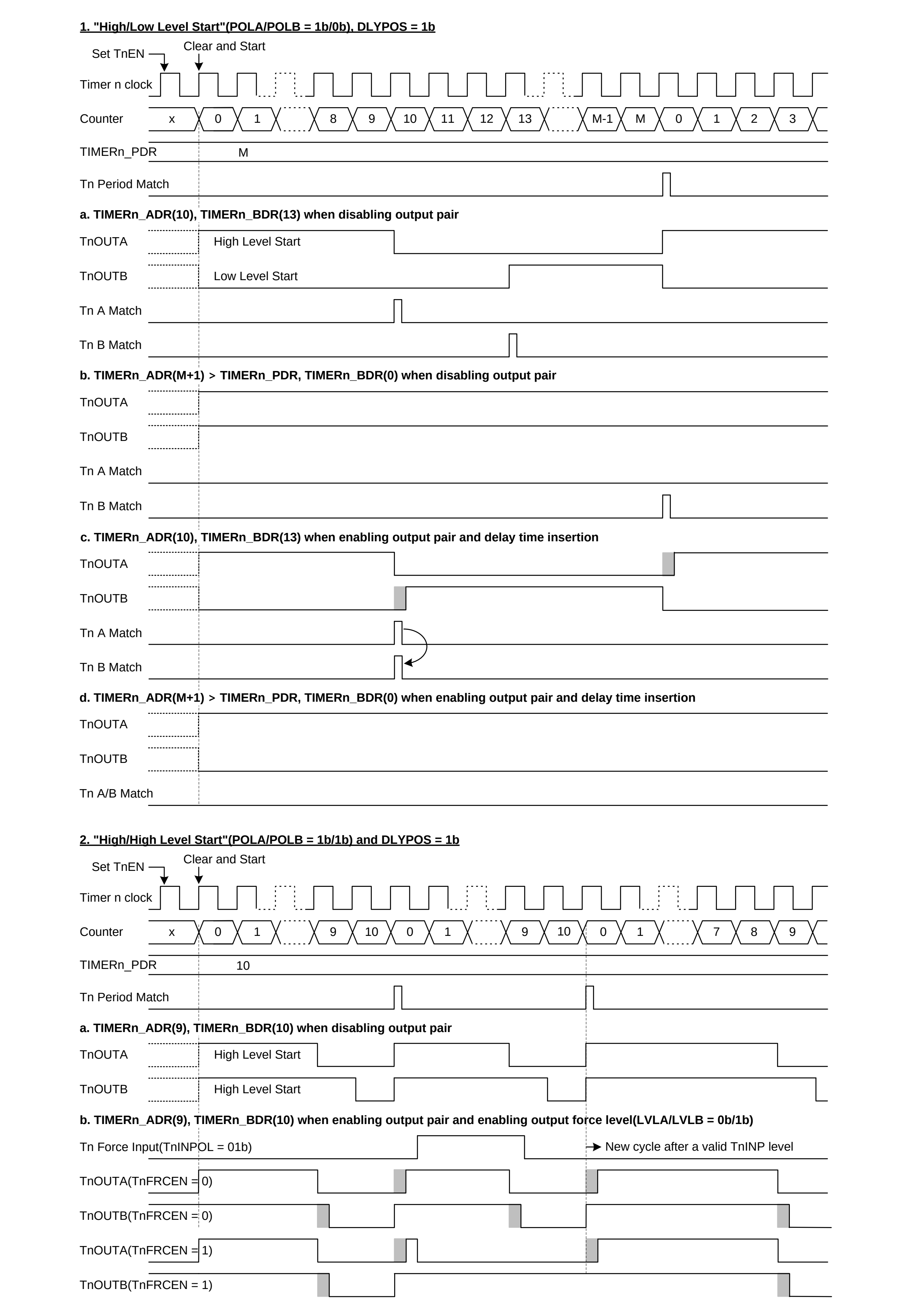

1. "Low/Low Level Start"(POLA/POLB = 0b/0b), DLYPOS = 0b
Set TnEN
Clear and Start
Timer n clock
Counter
x
0
1
8
9
10
11
12
13
M-1
M
0
1
2
3
TIMERn_PDR
M
Tn Period Match
a. TIMERn_ADR(10), TIMERn_BDR(13) when disabling output pair
TnOUTA
Low Level Start
TnOUTB
Low Level Start
Tn A Match
Tn B Match
b. TIMERn_ADR(M+1) > TIMERn_PDR, TIMERn_BDR(0) when disabling output pair
TnOUTA
TnOUTB
Tn A Match
Tn B Match
c. TIMERn_ADR(10), TIMERn_BDR(13) when enabling output pair and delay time insertion
TnOUTA
TnOUTB
Tn A Match
Tn B Match
d. TIMERn_ADR(M+1) > TIMERn_PDR, TIMERn_BDR(0) when enabling output pair and delay time insertion
TnOUTA
TnOUTB
Tn A/B Match
2. "Low/High Level Start"(POLA/POLB = 0b/1b) and DLYPOS = 0b
Set TnEN
Clear and Start
Timer n clock
Counter
x
0
1
9
10
0
1
9
10
0
1
7
8
9
TIMERn_PDR
10
Tn Period Match
a. TIMERn_ADR(9), TIMERn_BDR(10) when enabling output pair and delay time insertion
TnOUTA
Low Level Start
TnOUTB
High Level Start
b. TIMERn_ADR(9), TIMERn_BDR(10) when disabling output pair and enabling output force level(LVLA/LVLB = 1b/0b)
Tn Force Input(TnINPOL = 01b)
New cycle after a valid TnINP level
TnOUTA(TnFRCEN = 0)
TnOUTB(TnFRCEN = 0)
TnOUTA(TnFRCEN = 1)
TnOUTB(TnFRCEN = 1)

Figure 51. Interval Mode Timing Chart With “DLYPOS = 1” (n = 40, 41, 42 and 43)


1. "High/Low Level Start"(POLA/POLB = 1b/0b), DLYPOS = 1b
Set TnEN
Clear and Start
Timer n clock
Counter
x
0
1
8
9
10
11
12
13
M-1
M
0
1
2
3
TIMERn_PDR
M
Tn Period Match
a. TIMERn_ADR(10), TIMERn_BDR(13) when disabling output pair
TnOUTA
High Level Start
TnOUTB
Low Level Start
Tn A Match
Tn B Match
b. TIMERn_ADR(M+1) > TIMERn_PDR, TIMERn_BDR(0) when disabling output pair
TnOUTA
TnOUTB
Tn A Match
Tn B Match
c. TIMERn_ADR(10), TIMERn_BDR(13) when enabling output pair and delay time insertion
TnOUTA
TnOUTB
Tn A Match
Tn B Match
d. TIMERn_ADR(M+1) > TIMERn_PDR, TIMERn_BDR(0) when enabling output pair and delay time insertion
TnOUTA
TnOUTB
Tn A/B Match
2. "High/High Level Start"(POLA/POLB = 1b/1b) and DLYPOS = 1b
Set TnEN
Clear and Start
Timer n clock
Counter
x
0
1
9
10
0
1
9
10
0
1
7
8
9
TIMERn_PDR
10
Tn Period Match
a. TIMERn_ADR(9), TIMERn_BDR(10) when disabling output pair
TnOUTA
High Level Start
TnOUTB
High Level Start
b. TIMERn_ADR(9), TIMERn_BDR(10) when enabling output pair and enabling output force level(LVLA/LVLB = 0b/1b)
Tn Force Input(TnINPOL = 01b)
New cycle after a valid TnINP level
TnOUTA(TnFRCEN = 0)
TnOUTB(TnFRCEN = 0)
TnOUTA(TnFRCEN = 1)
TnOUTB(TnFRCEN = 1)

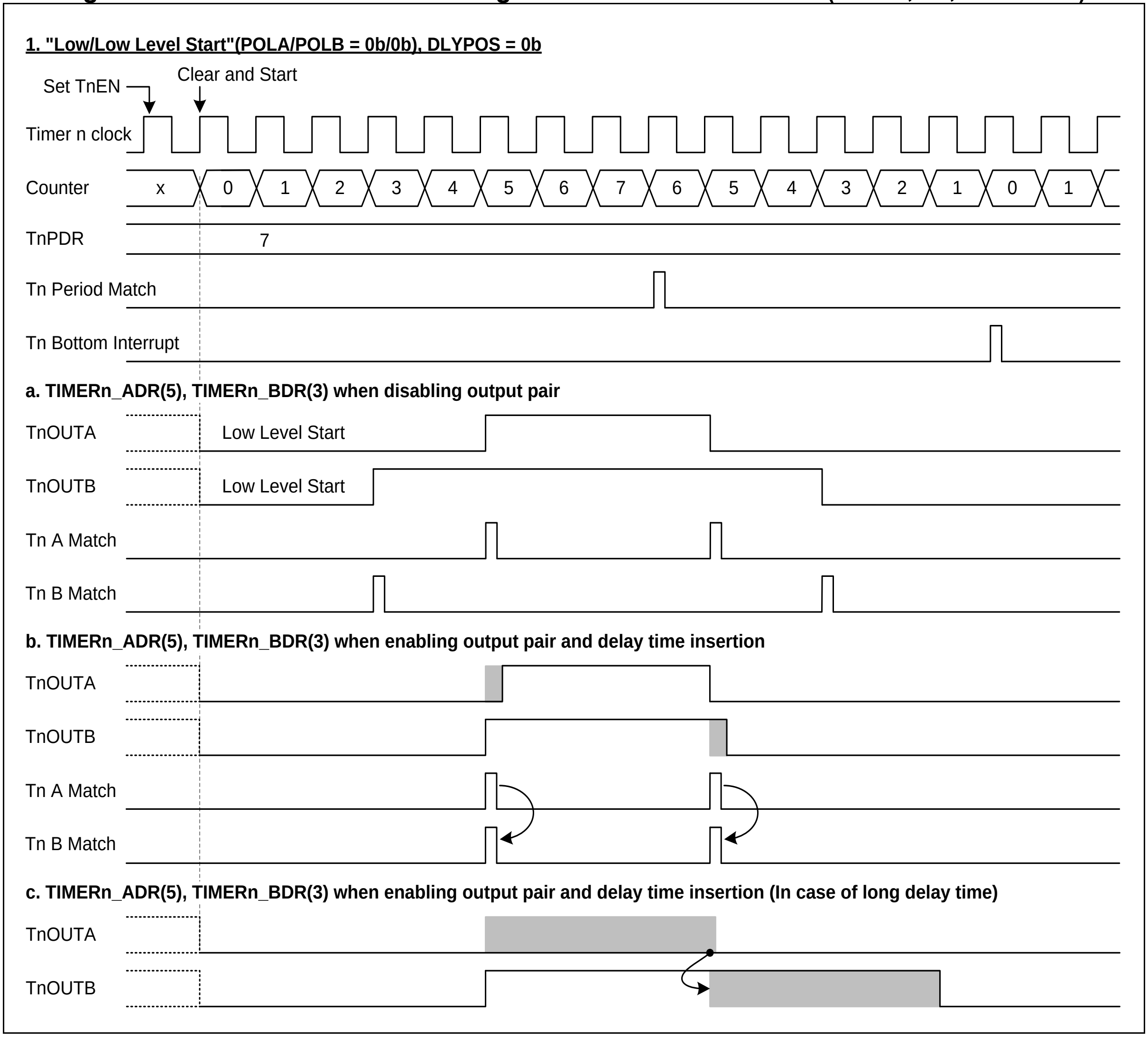
11.4.4 Back-to-back mode

Back-to-back mode is set by configuring the TnMS[1:0] as ‘10’. In the Back-to-back mode, the 16-bit up/down counter repeats the up/down counting. In fact, the effective duty and period becomes twice the register setting.

If the TIMERn_PDR’s data value is set to “0x3210”, 16-bit up/down counter will increment until it reaches 0x3210. At this point, a period match signal is generated and the period match interrupt takes place. Then the 16-bit up/down counter will decrement until it reaches 0x0000. At this point, the bottom interrupt takes place. This process repeats.

Since other functions operate similar to the interval mode, a user can refer to the interval mode for information of them. (n = 40, 41, 42 and 43)

Figure 52. Back-to-Back Mode Timing Chart with “DLYPOS = 0” (n = 40, 41, 42 and 43)

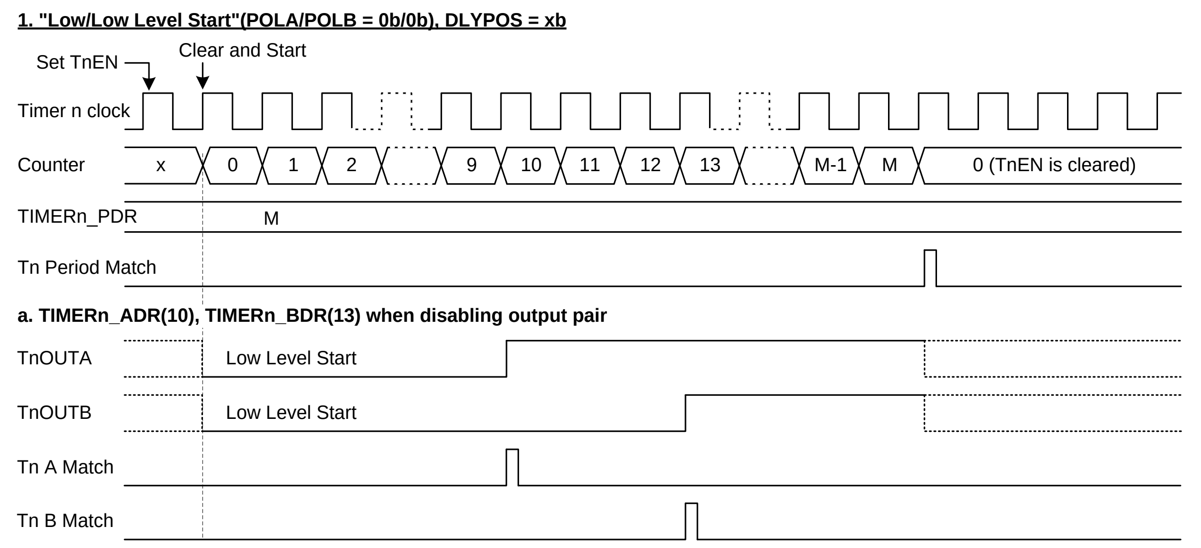

1. "Low/Low Level Start"(POLA/POLB = 0b/0b), DLYPOS = 0b
Set TnEN
Clear and Start
Timer n clock
Counter
x
0
1
2
3
4
5
6
7
6
5
4
3
2
1
0
1
TnPDR
7
Tn Period Match
Tn Bottom Interrupt
a. TIMERn_ADR(5), TIMERn_BDR(3) when disabling output pair
TnOUTA
Low Level Start
TnOUTB
Low Level Start
Tn A Match
Tn B Match
b. TIMERn_ADR(5), TIMERn_BDR(3) when enabling output pair and delay time insertion
TnOUTA
TnOUTB
Tn A Match
Tn B Match
c. TIMERn_ADR(5), TIMERn_BDR(3) when enabling output pair and delay time insertion (In case of long delay time)
TnOUTA
TnOUTB

11.4.5 One-shot interval mode

One-shot interval mode is set by configuring the TnMS[1:0] as ‘11’. When the value of 16-bit up/down counter reaches the value of the TIMERn_PDR after start, a match signal is generated. The period match interrupt is occurred, the TnEN bit is automatically cleared to “0b”, and the one-shot interval mode is finished successively.

Since other functions operate similar to the interval mode, a user can refer to the interval mode for information of them. (n = 40, 41, 42 and 43)

Figure 53. One-Shot Interval Timing Chart (n: 40, 41, 42 and 43)



11.4.6 Timer counter sharing function

The timer can be linked together internally for synchronization. The timer to be used as a master must clear the CNTSHEN bit of the TIMERn_CR register to “0b”. On the other hand, the timer to be used as slave should set the CNTSHEN bit of TIMERn_CR register to “1b”.

The counter sharing timers, a master and slaves, must have the same values in the TnMS[1:0] and UPDT[1:0] bits of TIMERn_CR register and in the TIMERn_PDR register. If the values are different, the counter sharing function may not work correctly. The clock frequency of the timers must also be set to the same value for good chaining. (n = 40, 41, 42 and 43)

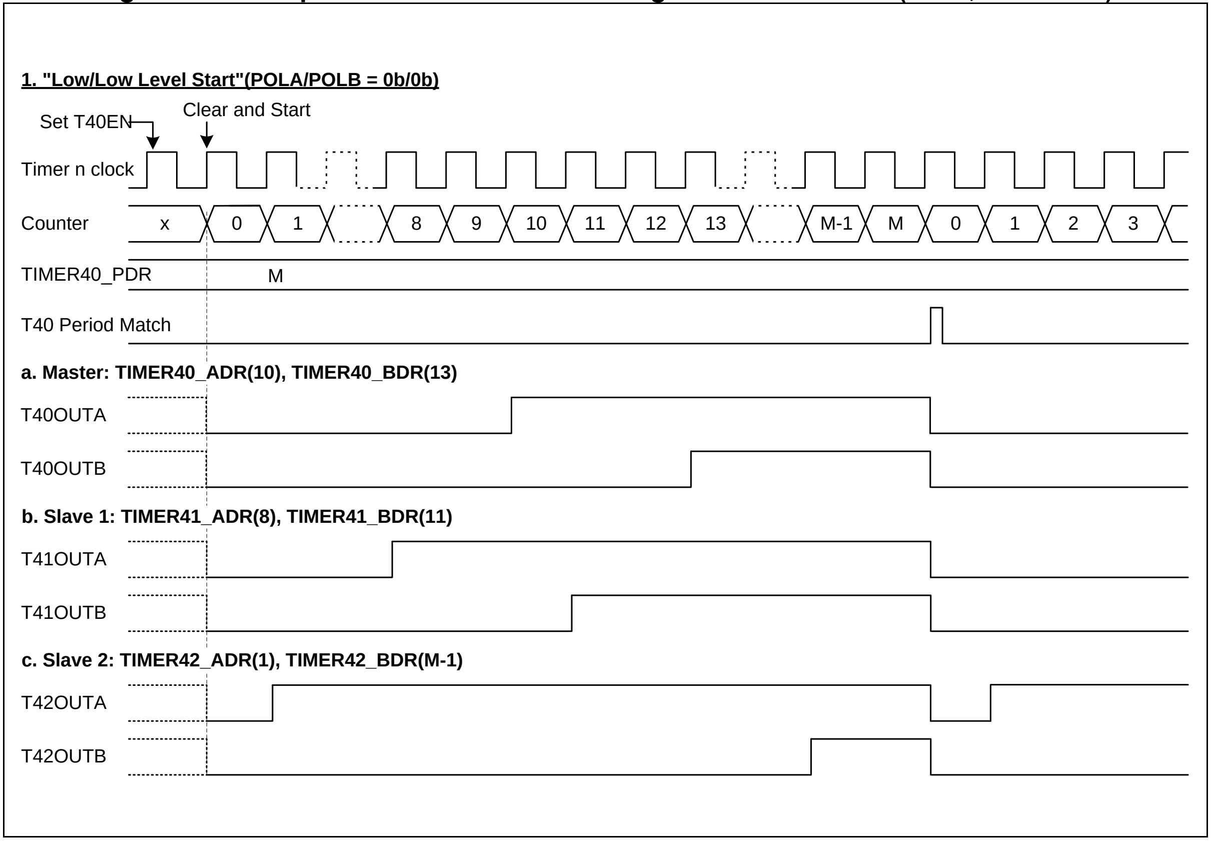
Table 52. Example of Timer Counter Sharing On Interval Mode

	TnEN	TnCLK	TnMS[1:0]	CNTSHEN	CNTSH[1:0]	UPDT[1:0]	PDR	PREDR
Master (T40)	1	Don't care		0	Don't care	One of 0x0 – 0x2	Don't care	
Slave 1 (T41)	1	Same as T40's		1	0x0 (T40)	Same as above	Same as T40's	
Slave 2 (T42)	1	Same as T40's		1	0x0 (T40)	Same as above	Same as T40's	

NOTES:

1. T40: Master, T41 and T42: Slave.
2. The TnMS[1:0] bits shall be set to the same value for the master and slave.
3. The TnCLK bit and TIMERn_PREDR register must also set to the same value for the same frequency of counter input.
4. The TIMERn_PDR register should also be set to the same value for the same period of timer outputs.

Figure 54. Example of Timer Counter Sharing On Interval Mode (n: 40, 41 and 42)



1. "Low/Low Level Start"(POLA/POLB = 0b/0b)
Set T40EN
Clear and Start
Timer n clock
Counter
x
0
1
8
9
10
11
12
13
M-1
M
0
1
2
3
TIMER40_PDR
M
T40 Period Match
a. Master: TIMER40_ADR(10), TIMER40_BDR(13)
T40OUTA
T40OUTB
b. Slave 1: TIMER41_ADR(8), TIMER41_BDR(11)
T41OUTA
T41OUTB
c. Slave 2: TIMER42_ADR(1), TIMER42_BDR(M-1)
T42OUTA
T42OUTB

12 Timer Counter 50

A timer block includes a single channel 16-bit general purpose timer. This timer has an independent 16-bit counter and a dedicated prescaler that feeds counting clock. It supports periodic timer, PWM pulse, one-shot timer and capture mode.

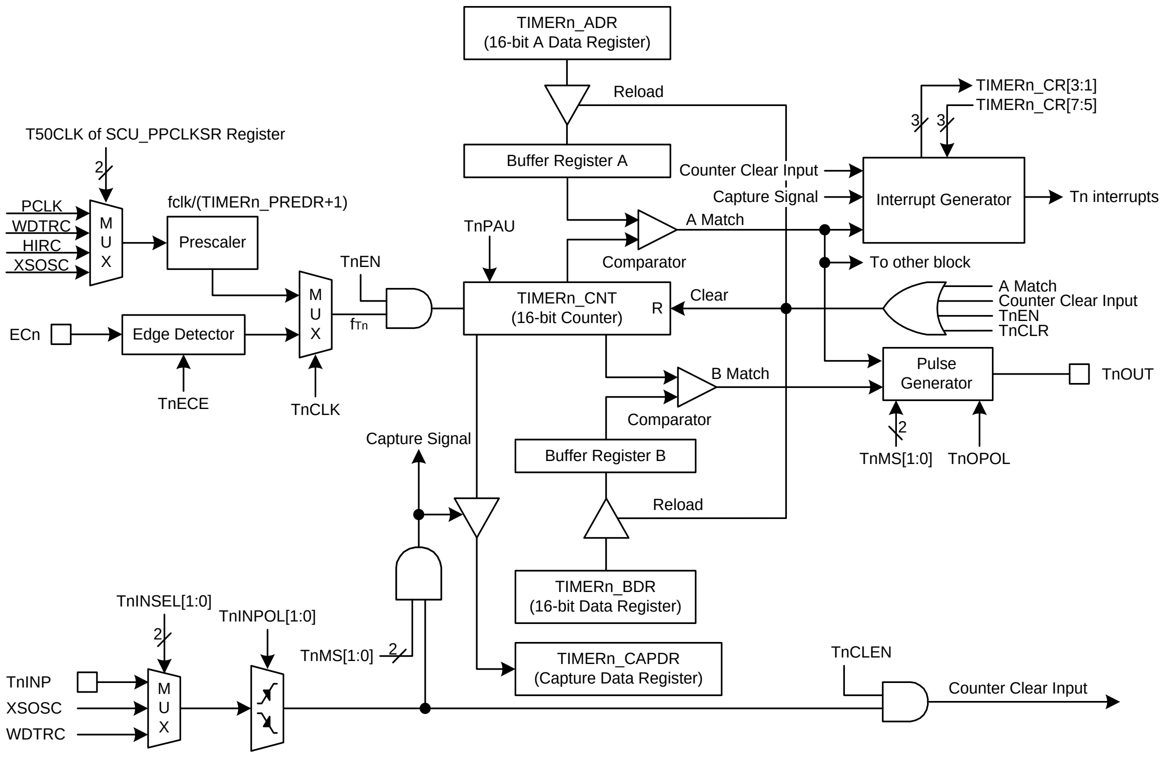
Additional free-run timer is optionally provided. Main purpose of this timer is to work as a periodical tick timer or to provide a wake-up source. The Timer counter 50 features the followings:

- 16-bit up-counter and 8-bit prescaler
- Interval timer, One-shot timer, PWM pulse, and Capture mode
- Synchronous start and clear function
- Low power operation with WDTRC or XSOSC

12.1 Timer counter 50 block diagram

Figure 55 shows the block diagram of a timer block unit.

Figure 55. Timer Counter n Block Diagram (n = 50)



NOTES:

1. TnEN is automatically cleared to logic ‘0’ after one pulse is generated in PPG one-shot interval mode.
2. After TnEN bit is set, counting of TIMERn_CNT may be delayed by 2 clocks or more.
3. Counter clear input can control wake-up from SLEEP mode by suppressing A-match interrupt.
4. High/low width of ECn clock should be longer than PCLK clock period.
5. The capture and counter clear input interrupts cannot be used as a wake-up source in DEEP SLEEP mode. That is, the interrupts will not be generated in DEEP SLEEP mode.

12.2 Pin description for timer counter 50

Table 53. Pins and External Signals for Timer Counter 50 (n = 50)

Pin name	Type	Description
ECn	I	External clock input
TnINP	I	Capture/Clear input
TnOUT	O	PWM/one-shot output

12.3 Registers

Base address and register map of the Timer 50 are shown in Table 54 and Table 55.

Table 54. Base Address of Timer 50

Name	Base address
TIMER50	0x4000_2B00

Table 55. Timer Register Map (n = 50)

Name	Offset	Type	Description	Reset value
TIMERn_CR	0x0000	RW	Timer/Counter n Control Register	0x00000000
TIMERn_ADR	0x0004	RW	Timer/Counter n A Data Register	0x0000FFFF
TIMERn_BDR	0x0008	RW	Timer/Counter n B Data Register	0x0000FFFF
TIMERn_CAPDR	0x000C	RO	Timer/Counter n Capture Data Register	0x00000000
TIMERn_PREDR	0x0010	RW	Timer/Counter n Prescaler Data Register	0x000000FF
TIMERn_CNT	0x0014	RO	Timer/Counter n Counter Register	0x00000000

12.3.1 TIMERn_CR: timer/counter n control register

Timer module should be configured properly before running. Once target purpose is defined, the timer can be configured in the TIMERn_CR register. After configuring this register, a user can start or stop the timer function by using this register.

TIMERn_CR register is 32-bit size and accessible in 32/16/8-bit. (n = 50)

TIMER50_CR=0x4000_2B00

31 30 29 28 27 26 25 24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	**Reserved**			**TnCLEN**	**TnINSEL**		**TnINPOL**		**TnEN**	**TnCLK**	**TnMS**		**TnECE**	**Reserved**	**TnOPOL**	**TnPAU**	**TnMIEN**	**TnCIEN**	**TnCLIEN**	**Reserved**	**TnMIFLAG**	**TnCIFLAG**	**TnCLIFLAG**	**TnCLR**
0x00	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
–	**-**	**-**	**-**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**-**	**RW**	**RW**	**RW**	**RW**	**RW**	**-**	**RW**	**RW**	**RW**	**RW**

20	TnCLEN	Timer n Counter Clear Input Enable.
		0 Disable counter clear input.
		1 Enable counter clear input at a valid edge by TnINPOL[1:0] bits
19 18	TnINSEL	Timer n Input Signal Selection.
		00 Select an external input signal.
		01 Select the XSOSC (External sub oscillator) signal
		10 Select the WDTRC (Watch-dog timer RC oscillator) signal
		11 Not used
		NOTE: This bit should be changed during T50EN bit is ‘0’.
17 16	TnINPOL	Timer n Input Capture/”Counter Clear Input” Polarity Selection.
		00 Capture/”Counter clear input” on falling edge.
		01 Capture/”Counter clear input” on rising edge
		10 Capture/”Counter clear input” on both of falling and rising edge
		11 Reserved
15	TnEN	Timer n Operation Enable.
		0 Disable timer n operation.
		1 Enable timer n operation. (Counter clear and start)
14	TnCLK	Timer n Clock Selection.
		0 Select an internal prescaler clock.
		1 Select an external clock.
		NOTE: This bit should be changed while TnEN bit is ‘0’.
13 12	TnMS	Timer n Operation Mode Selection.
		00 Timer/Counter mode. (TnOUT: toggle at A-match)
		01 Capture mode. (The A-match interrupt can occur)
		10 PPG one-shot mode. (TnOUT: Programmable pulse output)
		11 PPG repeat mode. (TnOUT: Programmable pulse output)
		NOTE: This bit should be changed while TnEN bit is ‘0’.
11	TnECE	Timer n External Clock Edge Selection.
		0 Select falling edge of external clock.
		1 Select rising edge of external clock.
9	TnOPOL	TnOUT Polarity Selection.
		0 Start high. (TnOUT is low level at disable)
		1 Start low. (TnOUT is high level at disable)
8	TnPAU	Timer n Counter Temporary Pause Control.
		0 Continue counting.
		1 Temporary pause.
7	TnMIEN	Timer n Match Interrupt Enable.
		0 Disable timer n match interrupt.
		1 Enable timer n match interrupt.
6	TnCIEN	Timer n Capture Interrupt Enable.
		0 Disable timer n capture interrupt.
		1 Enable timer n capture interrupt.
5	TnCLIEN	Timer n Counter Clear Input Interrupt Enable.

		0	Disable timer n Counter Clear Input interrupt.
		1	Enable timer n Counter Clear Input interrupt.
3	TnMIFLAG	Timer n Match Interrupt Flag.	
		0	No request occurred.
		1	Request occurred. The bit is cleared to ‘0’ when ‘1’ is written.
2	TnCIFLAG	Timer n Capture Interrupt Flag.	
		0	No request occurred.
		1	Request occurred. The bit is cleared to ‘0’ when ‘1’ is written.
		NOTE: This bit may not be set to ‘1’ by capture input signal in DEEP SLEEP mode.	
1	TnCLIFLAG	Timer n Counter Clear Input Interrupt Flag.	
		0	No request occurred.
		1	Request occurred. The bit is cleared to ‘0’ when ‘1’ is written.
		NOTE: This bit may not be set to ‘1’ by counter clear input signal in DEEP SLEEP mode.	
0	TnCLR	Timer n Counter and Prescaler Clear.	
		0	No effect.
		1	Clear timer n counter and prescaler. (Automatically cleared to ‘0’ after operation)

12.3.2 TIMERn_ADR: timer/counter n A data register

TIMERn_ADR register is 32-bit size and accessible in 32/16/8-bit. (n = 50)

TIMER50_ADR=0x4000_2B04

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																ADATA															
0x0000																0xFFFF															
-																RW															

15 0	ADATA	Timer/Counter n A Data. The range is 0x0002 to 0xFFFF. A match time: (ADATA[15:0]+1) ÷ f_{Tn}
		NOTE: Do not write “0x0000” in the TIMERn_ADR register under PPG mode.

12.3.3 TIMERn_BDR: timer/counter n B data register

TIMERn_BDR register is 32-bit size and accessible in 32/16/8-bit. (n = 50)

TIMER50_BDR=0x4000_2B08

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																BDATA															
0x0000																0xFFFF															
-																RW															

15 0	BDATA	Timer/Counter n B Data. The range is 0x0000 to 0xFFFF. B match time: (BDATA[15:0]) ÷ f_{Tn}

12.3.4 TIMERn_CAPDR: timer/counter n capture data register

TIMERn_CAPDR register is 32-bit size and accessible in 32/16/8-bit. (n = 50)

TIMER50_CAPDR=0x4000_2B0C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																CAPD															
0x0000																0x0000															
-																RO															

15 0	CAPD	Timer/Counter n Capture Data.

12.3.5 TIMERn_PREDR: timer/counter n prescaler data register

TIMERn_PREDR register is 32-bit size and accessible in 32/16/8-bit. (n = 50)

TIMER50_PREDR=0x4000_2B10

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved	**PRED**
0x000000	**0xFF**
-	**RW**

7 0	PRED	Timer/Counter n Prescaler Data.

12.3.6 TIMERn_CNT: timer/counter n counter register

TIMERn_CNT register is 32-bit size and accessible in 32/16/8-bit. (n = 50)

TIMER50_CNT=0x4000_2B14

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Reserved	**CNT**
0x0000	**0x0000**
-	**RO**

15 0	CNT	Timer/Counter n Counter.

12.4 Functional description

12.4.1 Timer counter 50

Timer/counter n can use an internal or an external clock as a clock source (ECn). A clock selection logic selects the clock source and the clock selection logic is controlled by clock selection bits (TnCLK). (n = 50)

- TIMER n clock sources are listed as followings:
  - PCLK/(TIMERn_PREDR +1)
  - WDTRC/(TIMERn_PREDR +1)
  - HIRC/(TIMERn_PREDR +1)
  - XSOSC/(TIMERn_PREDR +1)
  - ECn

In capture mode, by TnINP, XSOSC or WDTRC, data is captured into input capture data register (TIMERn_CAPDR). Timer n outputs the comparison result between counter and data register through TnOUT port in Timer/counter mode. In addition, Timer n outputs PWM waveform through TnOUT port in PPG mode. (n = 50)

Table 56. Timer n Operating Modes (n = 50)

TnEN	Alternative mode	TnMS[1:0]	TIMERn_PREDR	Timer n
1	T50OUT	00	0xXX	16-bit Timer/Counter Mode
1	T50INP	01	0xXX	16-bit Capture Mode
1	T50OUT	10	0xXX	16-bit PPG Mode(one-shot mode)
1	T50OUT	11	0xXX	16-bit PPG Mode(repeat mode)

12.4.2 16-bit Timer/counter mode

16-bit Timer/counter mode is selected by control register as shown in Figure 56. The 16-bit timer has a counter register and a data register.

The counter register is increased by internal or external clock input. Timer n can use an input clock with 8-bit prescaler division rates (TIMERn_PREDR) and an external Clock (ECn). When the values of TIMERn_CNT and TIMERn_ADR are the same in the timer n, a match signal is generated and the interrupt of Timer n takes place.

The TIMERn_CNT values are automatically cleared by the match signal. It can also be cleared by software (TnCLR).

Figure 56. 16-bit Timer/Counter Mode for Timer n (n = 50)

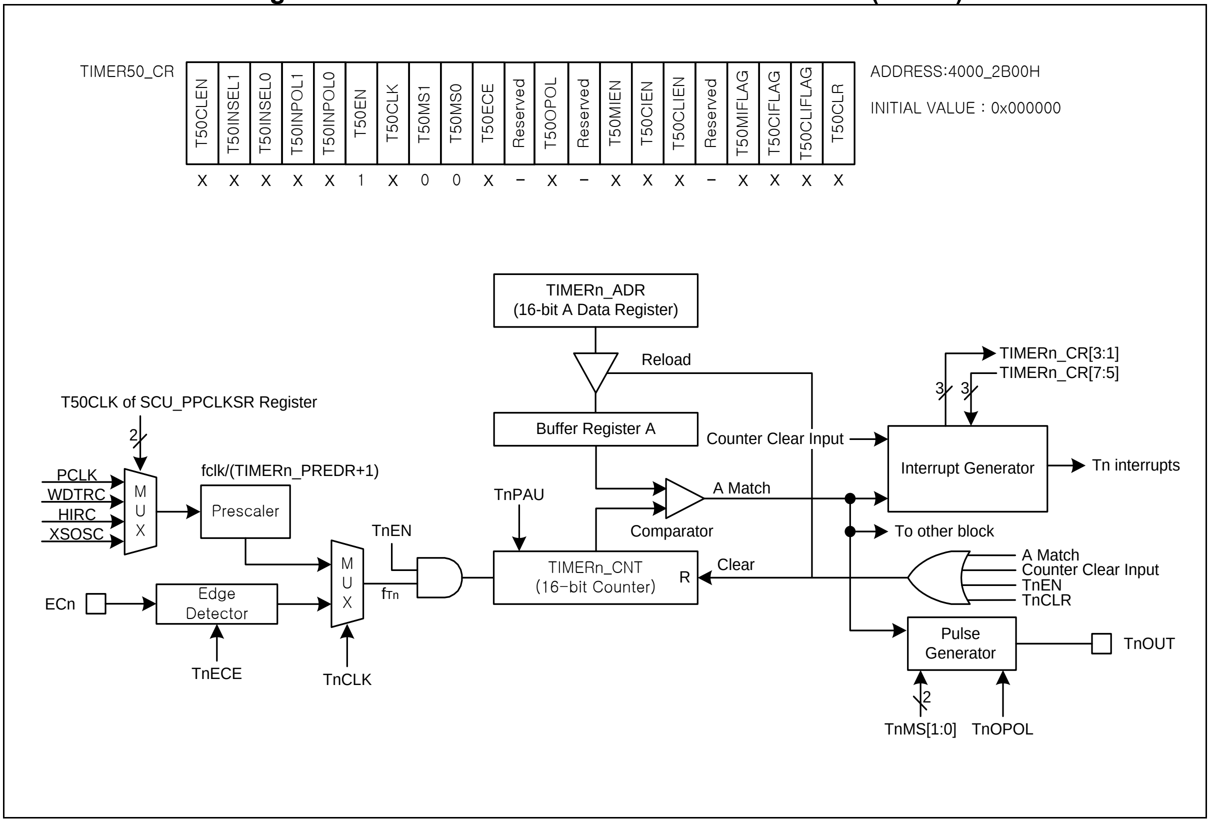
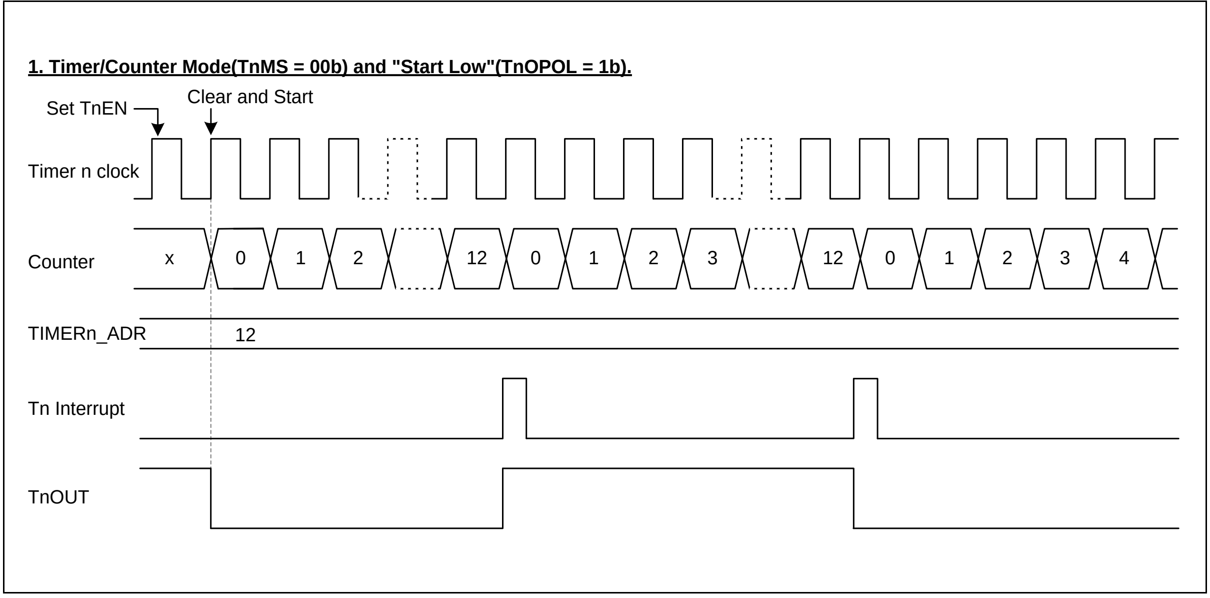


Figure 57. 16-bit Timer/Counter n Example (n = 50)

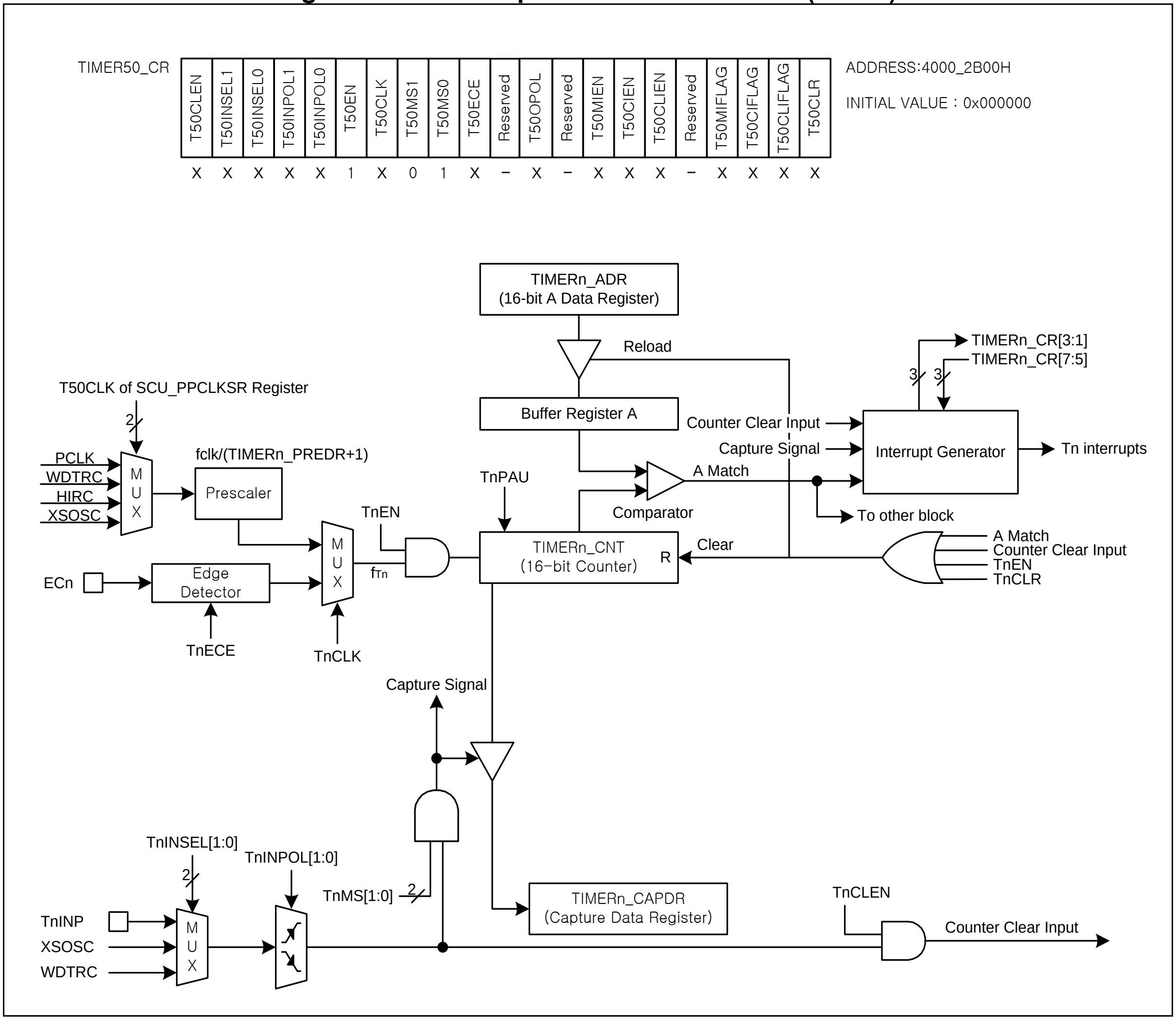


12.4.3 16-bit Capture mode

Timer n Capture mode is evoked by configuring TnMS[1:0] as ‘01’. The internal clock can be used as a clock source. It basically has the same function as the 16-bit timer/counter mode and an interrupt takes place when TIMERn_CNT becomes equal to TIMERn_ADR. (n = 50).

This timer interrupt in Capture mode is very useful when the pulse width of captured signal is wider than the maximum period of timer. The capture result is loaded into TIMERn_CAPDR. In the timer n capture mode, timer n output (TnOUT) waveform is not available.

Figure 58. 16-bit Capture Mode for Timer n (n = 50)



TIMER50_CR

T50CLEN	T50INSEL1	T50INSEL0	T50INPOL1	T50INPOL0	T50EN	T50CLK	T50MS1	T50MS0	T50ECE	Reserved	T50OPOL	Reserved	T50MIEN	T50CIEN	T50CLIEN	Reserved	T50MIFLAG	T50CIFLAG	T50CLIFLAG	T50CLR
X	X	X	X	X	1	X	0	1	X	-	X	-	X	X	X	-	X	X	X	X

ADDRESS:4000_2B00H

INITIAL VALUE : 0x000000

Figure 59. 16-bit Capture Mode for Timer 50

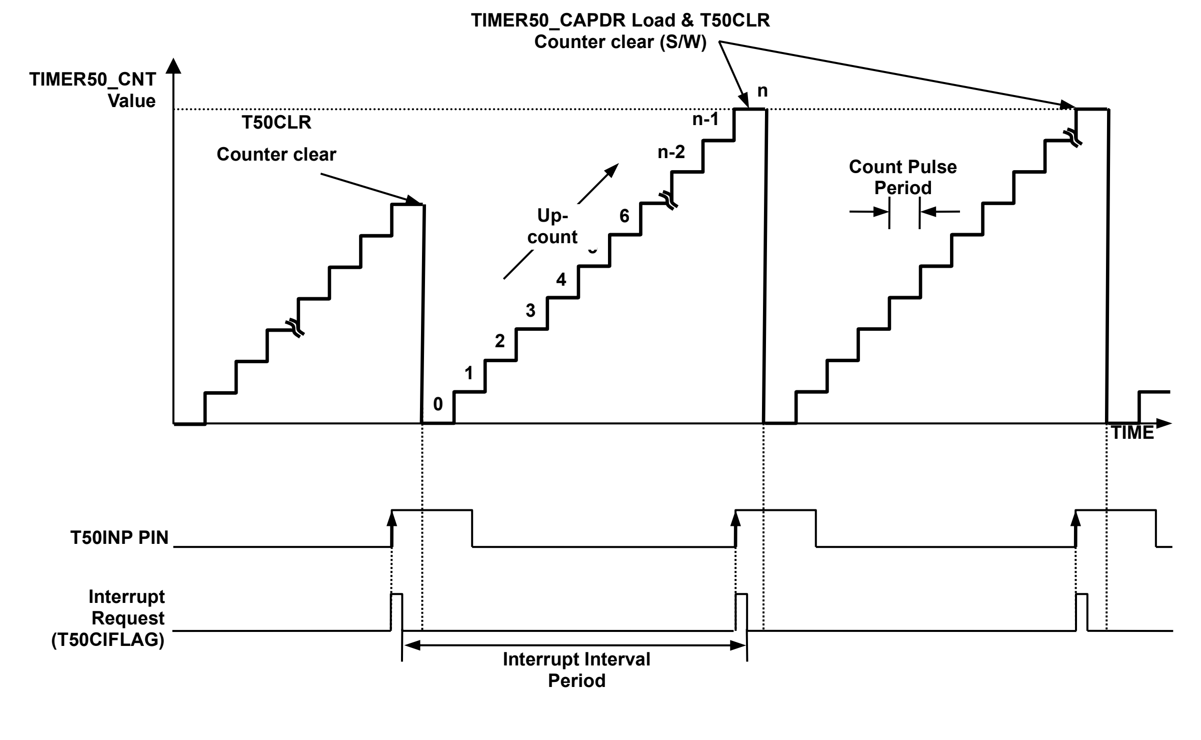
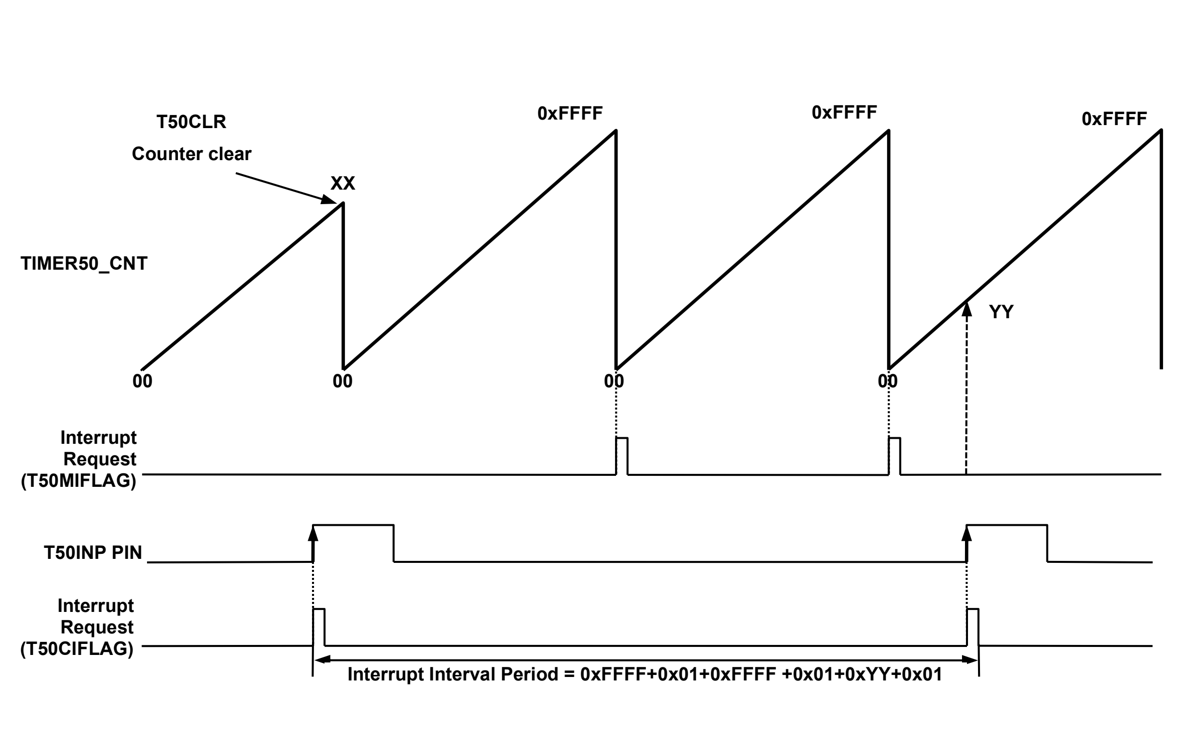


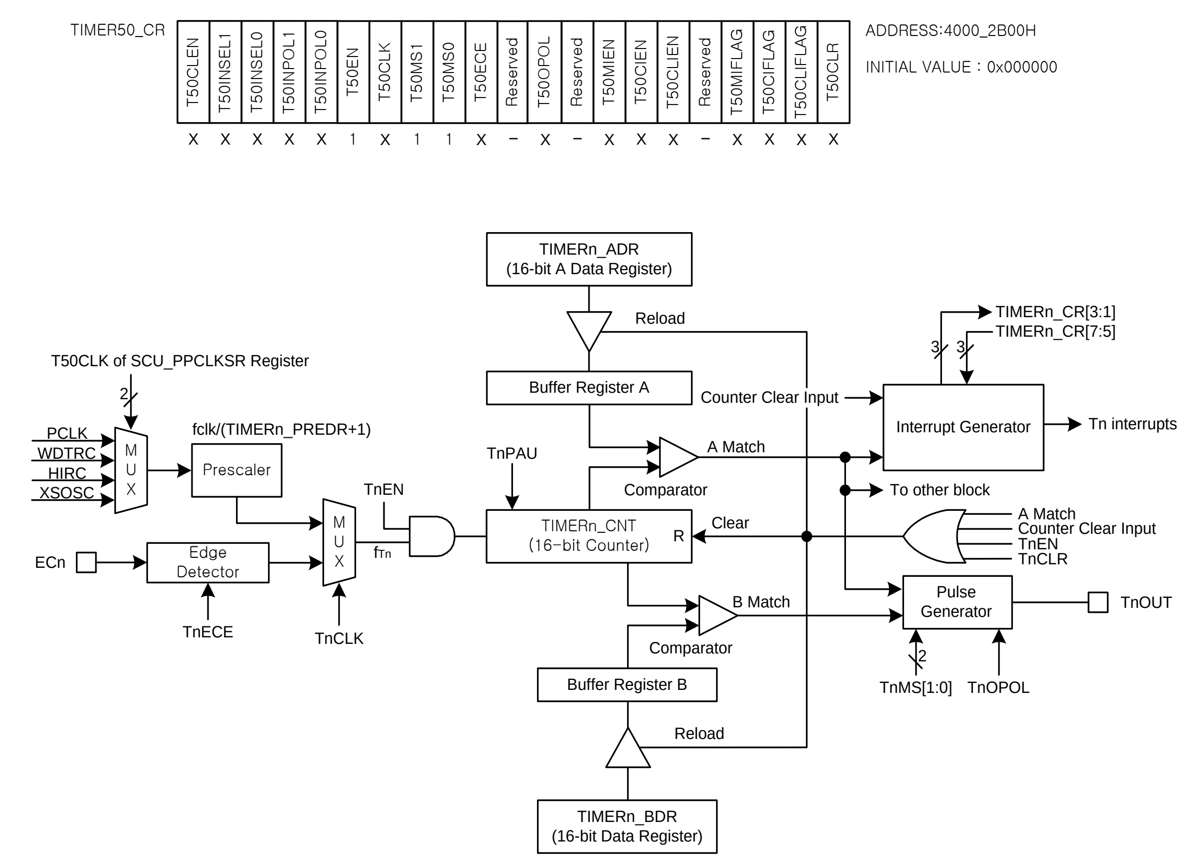
Figure 60. Express Timer Overflow in Capture Mode



12.4.4 16-bit PPG mode

Timer n has a PPG (Programmable Pulse Generation) function. In PPG mode, the TnOUT pin generates PWM output of up to 16-bit resolution. This pin should be configured as TnOUT function in the Px_AFSR1/Px_AFSR2 for PWM output. The period of PWM output is determined by the TIMERn_ADR. The duty of PWM output is determined by TIMERn_BDR. (x = A to F)

Figure 61. 16-bit PPG Repeat and One-shot Mode for Timer n (n = 50)



TIMER50_CR

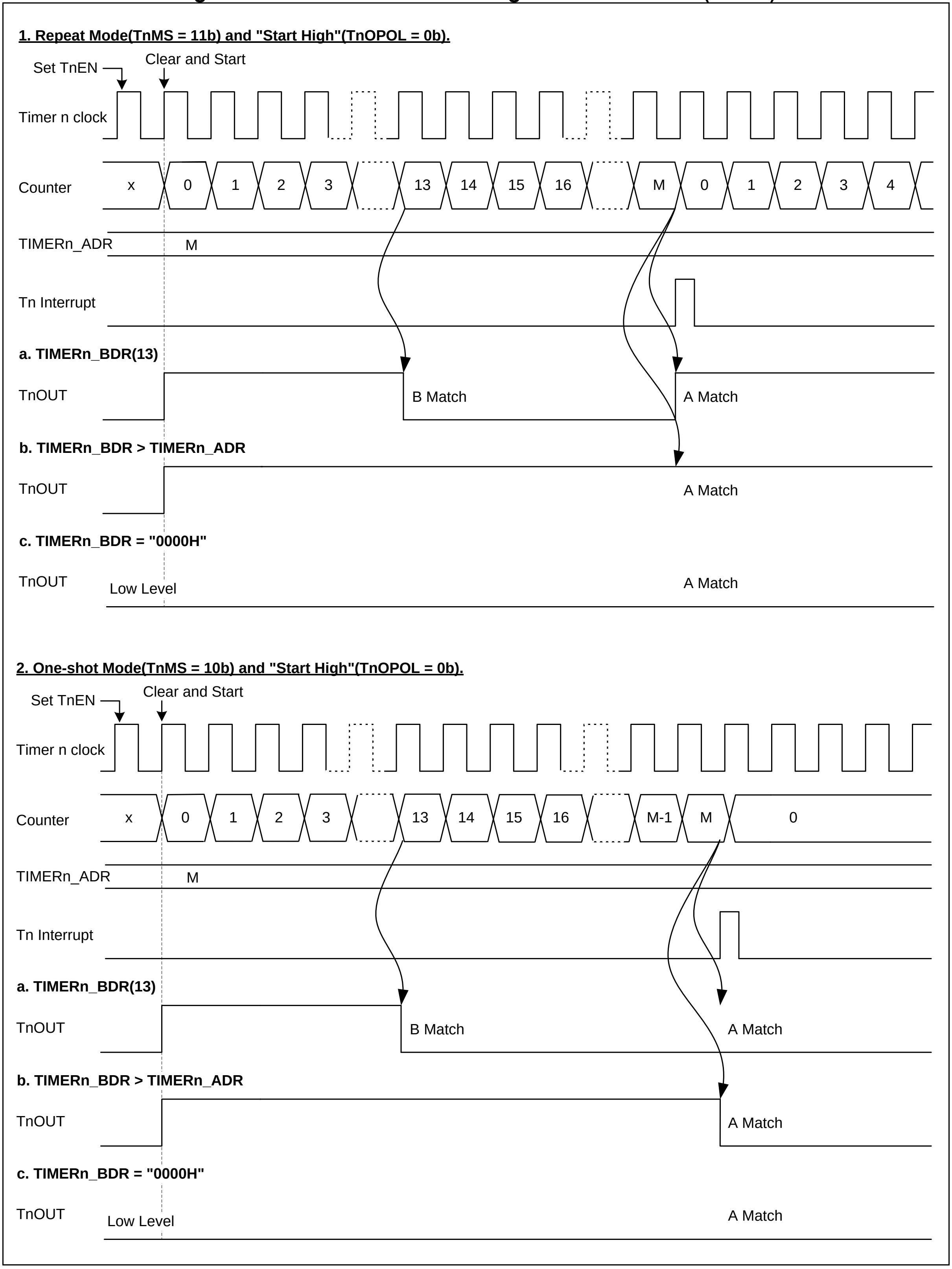
ADDRESS:4000_2B00H

INITIAL VALUE : 0x000000

T50CLEN	T50INSEL1	T50INSEL0	T50INPOL1	T50INPOL0	T50EN	T50CLK	T50MS1	T50MS0	T50ECE	Reserved	T50OPOL	Reserved	T50MIEN	T50CIEN	T50CLIEN	Reserved	T50MIFLAG	T50CIFLAG	T50CLIFLAG	T50CLR
X	X	X	X	X	1	X	1	1	X	-	X	-	X	X	X	-	X	X	X	X

NOTE: The TnEN is automatically cleared to logic ‘0’ after one pulse is generated in PPG one-shot mode.

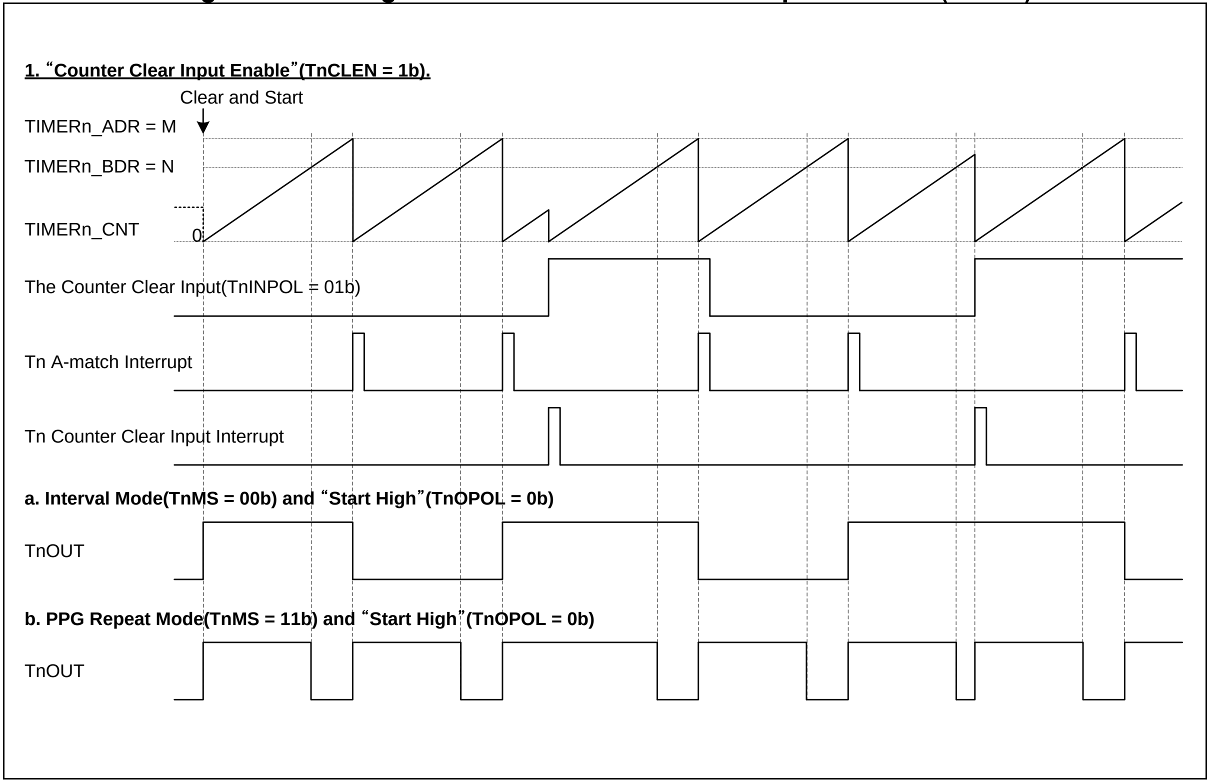
Figure 62. 16-bit PPG Mode Timing Chart for Timer n (n = 50)



12.4.5 Counter clear input enable

TIMERn_CNT value can be automatically cleared by the "Counter clear input signal" when the "Counter clear input" is enabled by configuring TnCLEN as '1'. So, the TnOUT waveform can be modified by TnINP pin. (n = 50)

Figure 63. Timing Chart When "Counter Clear Input Enable" (n = 50)


1. "Counter Clear Input Enable"(TnCLEN = 1b).
Clear and Start
TIMERn_ADR = M
TIMERn_BDR = N
TIMERn_CNT
0
The Counter Clear Input(TnINPOL = 01b)
Tn A-match Interrupt
Tn Counter Clear Input Interrupt
a. Interval Mode(TnMS = 00b) and "Start High"(TnOPOL = 0b)
TnOUT
b. PPG Repeat Mode(TnMS = 11b) and "Start High"(TnOPOL = 0b)
TnOUT

13 Timer Counter 60

A timer block includes a single channel 16-bit general purpose timer. This timer has an independent 16-bit counter and 100Hz RC oscillator that feeds counting clock. It supports only a periodic timer.

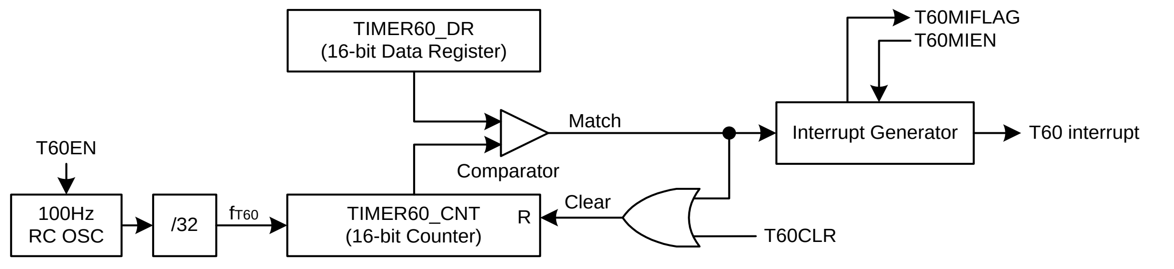
Main purpose of this timer is to provide a wake-up source from DEEP SLEEP mode 3 (shutdown). The Timer counter 60 features the followings:

- 16-bit up-counter and interval timer mode
- Synchronous start and clear function
- Low power operation with an internal 100Hz RC oscillator

13.1 Timer counter 60 block diagram

Figure 64 shows the block diagram of the timer counter 60.

Figure 64. Timer Counter 60 Block Diagram



NOTES:

1. The TIMER60_DR register must be written while the timer counter 60 is disabled.
2. This timer is primarily intended to be used as a wake-up source in DEEP SLEEP mode 3 (shutdown).
3. When this timer is used in RUN mode, pay attention to the frequency deviation of 100Hz RC oscillator.

13.2 Registers

Base address and register map of the timer counter 60 block are shown in Table 57 and Table 58.

Table 57. Base Address of Timer 60

Name	Base address
TIMER60	0x4000_2F00

Table 58. Timer 60 Register Map

Name	Offset	Type	Description	Reset value
TIMER60_CR	0x0000	RW	Timer/Counter 60 Control Register	0x00000000
TIMER60_DR	0x0004	RW	Timer/Counter 60 Data Register	0x0000FFFF
TIMER60_CNT	0x0008	RO	Timer/Counter 60 Counter Register	0x00000000

13.2.1 TIMER60_CR: timer/counter 60 control register

TIMER60_CR register is 32-bit size and accessible in 32/16/8-bit. This register may not reset by wakeup in shutdown mode.

TIMER60_CR=0x4000_2F00

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="16">Reserved</td><td>T60EN</td><td colspan="7">Reserved</td><td>T60MIEN</td><td colspan="3">Reserved</td><td>T60MIFLAG</td><td colspan="2">Reserved</td><td>T60CLR</td></tr>
<tr><td colspan="16">0x0000</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td colspan="16">-</td><td>RW</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>RW</td><td>-</td><td>-</td><td>-</td><td>RW</td><td>-</td><td>-</td><td>RW</td></tr>
</table>

15	T60EN	Timer 60 Operation Enable.	
		0	Disable timer 60 operation and 100Hz RC oscillator
		1	Enable timer 60 operation and 100Hz RC oscillator
7	T60MIEN	Timer 60 Match Interrupt Enable.	
		0	Disable timer 60 match interrupt
		1	Enable timer 60 match interrupt
3	T60MIFLAG	Timer 60 Match Interrupt Flag.	
		0	No request occurred
		1	Request occurred. This bit is cleared to ‘0’ when write ‘1’.
0	T60CLR	Timer 60 Counter Clear.	
		0	No effect
		1	Clear timer 60 counter (Automatically cleared to “0b” after operation)

13.2.2 TIMER60_DR: timer/counter 60 data register

TIMER60_DR register is 32-bit size and accessible in 32/16/8-bit. This register may not reset by wakeup in shutdown mode.

TIMER60_DR=0x4000_2F04

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																DATA															
0x0000																0xFFFF															
-																RW															

15 0	DATA	Timer/Counter 60 Data. The range is 0x0001 to 0xFFFF. Match time: (DATA[15:0]+1)÷fT60

13.2.3 TIMER60_CNT: timer/counter 60 counter register

TIMER60_CNT register is 32-bit size and accessible in 32/16/8-bit. This register may not reset by wakeup in shutdown mode.

TIMER60_CNT=0x4000_2F08

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																CNT															
0x0000																0x0000															
-																RO															

15 0	CNT	Timer/Counter 60 Counter.

13.3 Functional description

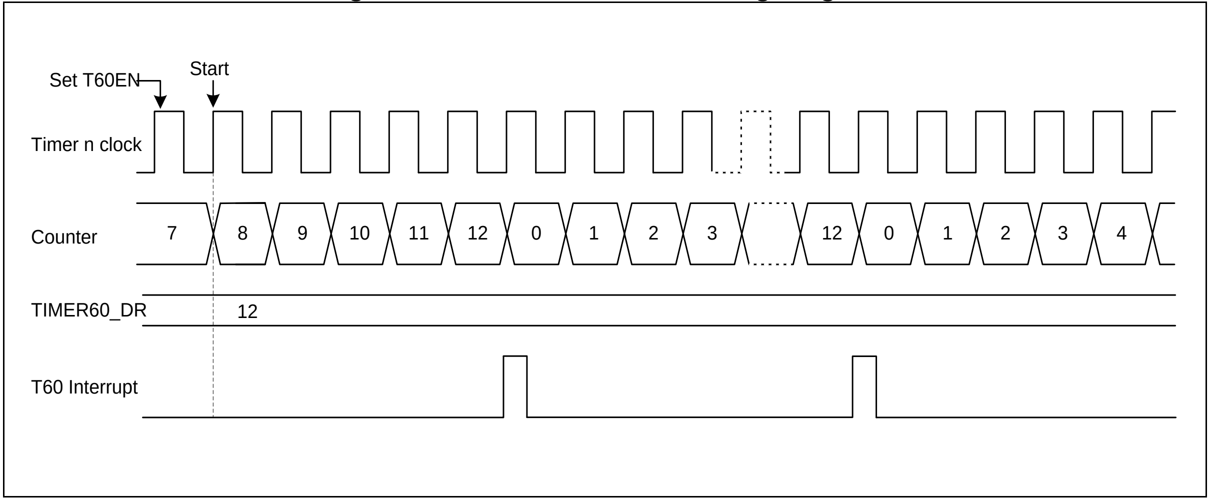
The timer/counter 60 has an internal RC oscillator and its frequency is about 100Hz. Also, it is divided into 32 and input to a 16-bit counter.

Although the input frequency of timer/counter 60 has a large error rate, but the frequency is 3.125Hz (100Hz÷32) and it has a 16-bit counter. So, it is useful when SHUT DOWN mode for low power mode is required for a long time.

13.3.1 Timing diagram

The 16-bit counter of timer/counter 60 is incremented by 1 every rising edge of f_{T60} during the T60EN bit of TIMER60_CR register is set. The counter is cleared to 0x0000 by writing 1 to the T60CLR bit of TIMER60_CR register. When enabled for the first time after reset (System reset and timer 60 reset by the T60RST of SCU_PPRST1 register), the interrupt of T60 may occur 0.5 clock earlier with the T60 clock. Figure 65 shows timer/counter 60 timing diagram.

Figure 65. Timer/Counter 60 Timing Diagram



14 High Speed 12-bit ADC

ADC (Analog-to-Digital Converter) of the A31L21x series allows conversion of an analog input signal to a corresponding 12-bit digital value. Its A/D module has sixteen analog inputs as shown in Figure 66. Output of the multiplexer is the input into the converter, which generates the result through successive approximation.

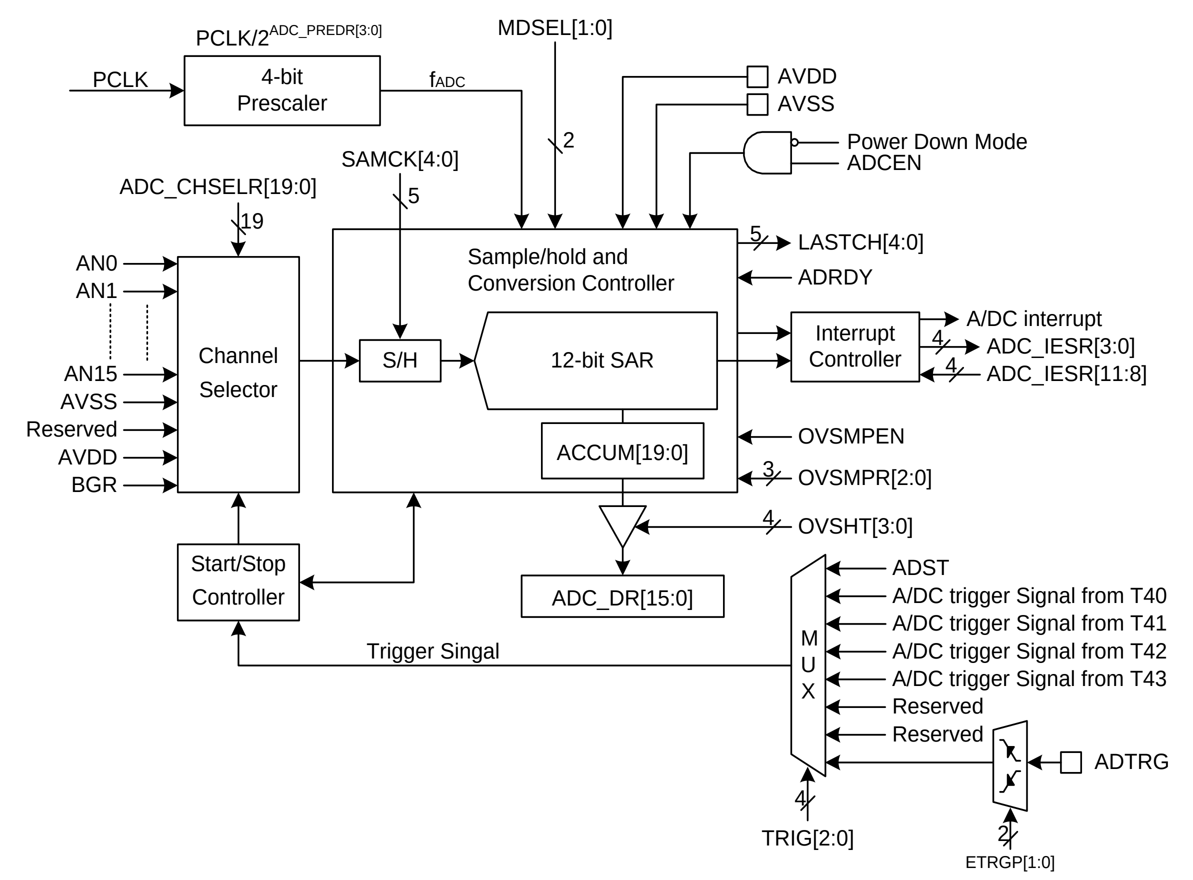
The A/D module has seven registers such as a control register (ADC_CR), a data register (ADC_DR), a prescaler data register (ADC_PREDR), an oversampling control register (ADC_OVSCR), an interrupt enable and status register (ADC_IESR), a sampling time register (ADC_SAMR), and a channel selection register (ADC_CHSELR). The A/D module supports single, sequential, and continuous conversion modes. Main features of the ADC are listed in the followings:

- 16-channel of analog inputs
- S/W (ADST), Timer trigger (T40/41/42/43 ADC trigger signal), and external trigger support
- Conversion time: Up to 1us with 12 clocks + at least 4 sample/hold clocks
- 4-bit Prescaler and 16-bit data registers
- Up to 256 over sampling
- Single, sequential, and continuous conversion mode

14.1 12-bit ADC block diagram

Figure 66 shows a block diagram of an ADC block.

Figure 66. 12-bit ADC Block Diagram



NOTE: The external trigger input, ADTRG, signal should be slower than a system clock frequency.

14.2 Pin description for 12-bit ADC

Table 59. Pins and External Signals for 12-bit ADC

Pin name	Type	Description
AN0	A	ADC Input 0
AN1	A	ADC Input 1
AN2	A	ADC Input 2
AN3	A	ADC Input 3
AN4	A	ADC Input 4
AN5	A	ADC Input 5
AN6	A	ADC Input 6
AN7	A	ADC Input 7
AN8	A	ADC Input 8
AN9	A	ADC Input 9
AN10	A	ADC Input 10
AN11	A	ADC Input 11
AN12	A	ADC Input 12
AN13	A	ADC Input 13
AN14	A	ADC Input 14
AN15	A	ADC Input 15
AVSS	AP	Analog GND
AVDD	AP	Analog Power

NOTE: Where A=Analog, AP= Analog Power

14.3 Registers

Base address and register map of the ADC are shown in Table 60 and Table 61.

Table 60. Base Address of ADC

Name	Base address
ADC	0x4000_3000

Table 61. High Speed ADC Register Map

Name	Offset	Type	Description	Reset value
ADC_CR	0x0000	RW	A/D Converter Control Register	0x00000000
ADC_OVSCR	0x0004	RW	A/D Converter Oversampling Control Register	0x00000000
ADC_IESR	0x0008	RW	A/D Converter Interrupt Enable and Status Register	0x00000000
ADC_DR	0x000C	RO	A/D Converter Data Register	Unknown
ADC_PREDR	0x0010	RW	A/D Converter Prescaler Data Register	0x00000000
ADC_SAMR	0x0014	RW	A/D Converter Sampling Time Register	0x00000000
ADC_CHSELR	0x0018	RW	A/D Converter Channel Selection Register	0x00000000

14.3.1 ADC_CR: A/D converter control register

A/D Converter module should be configured properly before running.

ADC_CR register is 32-bit size and accessible in 32/16/8-bit.

ADC_CR=0x4000_3000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																ADCEN	Reserved		TRIG			ETRGP		ADRDY	Reserved		MDSEL		Reserved		ADST
0x0000																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
–																RW	–	–	RW	RW	RW	RW	RW	RW	–	–	RW	RW	–	–	RW

15	ADCEN	ADC Module Enable.
		0 Disable ADC module operation.
		1 Enable ADC module operation.
		NOTE: This ADC module is disabled in DEEP SLEEP mode.
12	TRIG	ADC Trigger Signal Selection.
10		000 ADST.
		001 ADC trigger signal from timer 40.
		010 ADC trigger signal from timer 41.
		011 ADC trigger signal from timer 42.
		100 ADC trigger signal from timer 43.
		111 External ADC trigger input (ADTRG)
		OthersReserved
9	ETRGP	ADC External Trigger Input Polarity Selection.
8		00 Disable ADC external trigger function
		01 Trigger on falling edge
		10 Trigger on rising edge
		11 Trigger on both of falling and rising edge
7	ADRDY	ADC Conversion Ready.
		0 Stop subsequent steps.
		1 Ready to convert.
4	MDSEL	ADC Conversion Mode Selection.
3		00 Single conversion mode.
		01 Sequential conversion mode
		10 Continuous conversion mode
		11 Reserved.
0	ADST	ADC Conversion S/W Start. This bit is automatically cleared to ‘0’ after operation.
		0 No effect.
		1 S/W Trigger signal generation for conversion start.

14.3.2 ADC_OVSCR: A/D converter oversampling control register

ADC_OVSCR register is 32-bit size and accessible in 32/16/8-bit.

ADC_OVSCR=0x4000_3004

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																OVSMPEN	Reserved							OVSMPR			Reserved	OVSHT			
0x0000																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
–																RW	-	-	-	-	-	-	-	RW	RW	RW	-	RW	RW	RW	RW

15	OVSMPEN	Oversampling Enable.	
		0	Disable oversampling.
		1	Enable oversampling.
7 5	OVSMPR	Oversampling Ratio Selection.	
		Oversampling ratio: $2^{OVSMPR[2:0]+1}$, Ex) On OVSMPR[2:0] = 010b, 2^{2+1} = x8	
3 0	OVSHT	Oversampling Data Shift.	
		0000	No shift
		0001	Shift right 1-bit
		0010	Shift right 2-bit
		0011	Shift right 3-bit
		0100	Shift right 4-bit
		0101	Shift right 5-bit
		0110	Shift right 6-bit
		0111	Shift right 7-bit
		1000	Shift right 8-bit
		Others	reserved

14.3.3 ADC_IESR: A/D converter interrupt enable and status register

ADC_IESR register is 32-bit size and accessible in 32/16/8-bit.

ADC_IESR=0x4000_3008

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								**Reserved**			**LASTCH**					**Reserved**				**STBIEN**	**OVRUNIEN**	**EOCIEN**	**EOSIEN**	**Reserved**				**STBIFLAG**	**OVRUNIFLAG**	**EOCIFLAG**	**EOSIFLAG**
0x00								**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
–								**-**	**-**	**-**	**RO**	**RO**	**RO**	**RO**	**RO**	**-**	**-**	**-**	**-**	**RW**	**RW**	**RW**	**RW**	**-**	**-**	**-**	**-**	**RW**	**RW**	**RW**	**RW**

20	LASTCH	ADC Last Conversion Channel Number.	
16		The LASTCH[4:0] indicates the last converted channel number	
11	STBIEN	ADC Stabilization Interrupt Enable.	
		0	Disable stabilization interrupt.
		1	Enable stabilization interrupt.
10	OVRUNIEN	ADC Data Overrun Interrupt Enable.	
		0	Disable overrun interrupt.
		1	Enable overrun interrupt.
9	EOCIEN	ADC End of Conversion Interrupt Enable.	
		0	Disable end of conversion interrupt
		1	Enable end of conversion interrupt
8	EOSIEN	ADC End of Sequence Interrupt Enable.	
		0	Disable end of sequence interrupt.
		1	Enable end of sequence interrupt.
3	STBIFLAG	ADC Stabilization Interrupt Flag.	
		0	No request occurred.
		1	Request occurred. This bit is cleared to ‘0’ when ‘1’ is written.
		NOTE:	This bit will be set to “1b” after about 16/f_{ADC} time when the ADC module is enabled by ADCEN bit. So, the ADC conversion should start after reviewing whether this bit is “1b”.
2	OVRUNIFLAG	ADC Data Overrun Interrupt Flag.	
		0	No request occurred.
		1	Request occurred. This bit is cleared to ‘0’ when ‘1’ is written.
1	EOCIFLAG	ADC End of Conversion Interrupt Flag.	
		00	No request occurred.
		01	Request occurred, This bit is cleared to ‘0’ when ‘1’ is written or the result data are read by s/w or DMA.
0	EOSIFLAG	ADC End of Sequence Interrupt Flag.	
		0	No request occurred.
		1	Request occurred, This bit is cleared to ‘0’ when write ‘1’.

14.3.4 ADC_DR: A/D converter data register

ADC_DR register is 32-bit size and accessible in 32/16/8-bit.

ADC_DR=0x4000_300C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		**ADATA**	
0x0000		**0xXXXX**	
-		**RO**	

15 0	ADATA	A/D Converter Result Data. On OVSMPEN = 0 - ADATA[15:12] = 0x0 and ADATA[11:0] = 12-bit data converted On OVSMPEN = 1 - OVSHT[3:0] = 0: ACCUM[15:0] → ADATA[15:0] - OVSHT[3:0] = 1: ACCUM[16:1] → ADATA[15:0] - OVSHT[3:0] = 2: ACCUM[17:2] → ADATA[15:0] - OVSHT[3:0] = 3: ACCUM[18:3] → ADATA[15:0] - OVSHT[3:0] = 4: ACCUM[19:4] → ADATA[15:0] - OVSHT[3:0] = 5: ACCUM[19:5] → ADATA[15] = 0x0 and ADATA[14:0] - OVSHT[3:0] = 6: ACCUM[19:6] → ADATA[15:14] = 0x0 and ADATA[13:0] - OVSHT[3:0] = 7: ACCUM[19:7] → ADATA[15:13] = 0x0 and ADATA[12:0] - OVSHT[3:0] = 8: ACCUM[19:8] → ADATA[15:12] = 0x0 and ADATA[11:0]

NOTE: After waking up in power down mode, this data register is uncertain.

14.3.5 ADC_PREDR: A/D converter prescaler data register

ADC_PREDR register is 32-bit size and accessible in 32/16/8-bit.

ADC_PREDR=0x4000_3010

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4	3	2	1	0
Reserved				**PRED**			
0x0000000				**0**	**0**	**0**	**0**
-				**RW**	**RW**	**RW**	**RW**

3 0	PRED	A/D Converter Prescaler Data. The prescaler sets the A/D conversion clock. The frequency of A/D converter should be less than or equal to 16MHz. The range is 0x0 to 0x8. $f_{ADC} = PCLK/2^{PRED[3:0]}$ Recommend Max. f_{ADC}: 16MHz If 2.7V ≤ AVDD Recommend Max. f_{ADC}: 8MHz If 1.71V ≤ AVDD

14.3.6 ADC_SAMR: A/D converter sampling time register

ADC_SAMR register is 32-bit size and accessible in 32/16/8-bit.

ADC_SAMR=0x4000_3014

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																								**Reserved**			**SAMCK**				
0x000000																								0	0	0	0	0	0	0	0
-																								-	-	-	RW	RW	RW	RW	RW

4 0	SAMCK	Sampling cycles for sample/hold circuit. The range is 0x0 to 0x1E. Sampling cycles: SAMCK[4:0] + 2. Conversion cycles: 12.

14.3.7 ADC_CHSELR: A/D converter channel selection register

ADC_CHSELR register is 32-bit size and accessible in 32/16/8-bit.

ADC_CHSELR=0x4000_3018

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								**Reserved**				**AN19(BGR)**	**AN18(AVDD)**	**Reserved**	**AN16(AVSS)**	**AN15**	**AN14**	**AN13**	**AN12**	**AN11**	**AN10**	**AN9**	**AN8**	**AN7**	**AN6**	**AN5**	**AN4**	**AN3**	**AN2**	**AN1**	**AN0**
0x00								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
–								-	-	-	-	RW	RW	-	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

x	ANx	A/D Converter Channel Selection, x : 0 to 19	
		0	ANx is not selected for conversion
		1	ANx is selected for conversion

NOTES:

1. This register should be not written on going conversion.
2. When entering DEEP SLEEP mode with ADC channel selected as AN19(BGR), the current increases a lot. So, the ADC_CHSELR.AN19(BGR) bit must be set to "0b" before entering DEEP SLEEP mode.

14.4 Functional description

14.4.1 ADC enable/disable control

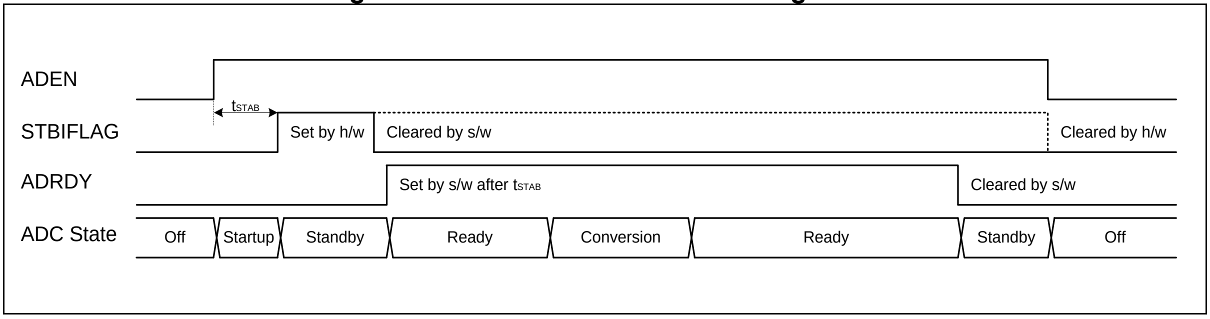
A/D converter needs a stabilization time of about 16/f_{ADC}, t_{STAB}, before it starts converting. The following procedure is required for an ADC conversion.

1. Set the ADCEN bit of ADC_CR register to “1b” for enabling ADC module operation.
2. Wait until the STBIFLAG bit of ADC_IESR register is set to “1b”. The bit is set after the ADC stabilization time.
3. Set the ADRDY bit of ADC_CR register to “1b” for converting.

The following procedure is required to disable the ADC module.

1. Clear the ADRDY bit of ADC_CR register to “0b” for conversion stop.
2. Clear the ADCEN bit of ADC_CR register to “0b” for disabling ADC module operation.

Figure 67. ADC Enable/Disable Timing Chart



14.4.2 Channel selection

The ADC has 16 input channels from GPIO pins and 3 internal channels. It is possible to convert a single channel or to scan a sequence of channels. The channels to be converted should be programmed in the ADC_CHSELR register. The conversion order is always from AN0 to AN19.

14.4.3 ADC conversion timing

Conversion clock of the ADC is the sum of sampling and converting. The sampling clock is equal to the ADC_SAMR register + 2 and converting clock is always 12 clocks. The ADC clock should be set appropriately by the ADC_PREDR register according to the AVDD voltage. In addition, the ADC_SAMR register must be set carefully for accurate conversion.

If the ADC_SAMR register has value of "0x2", the sampling clock is 4 clocks. Since the converting clock is always 12 clocks, the conversion clock of the ADC is calculated as shown in the followings:

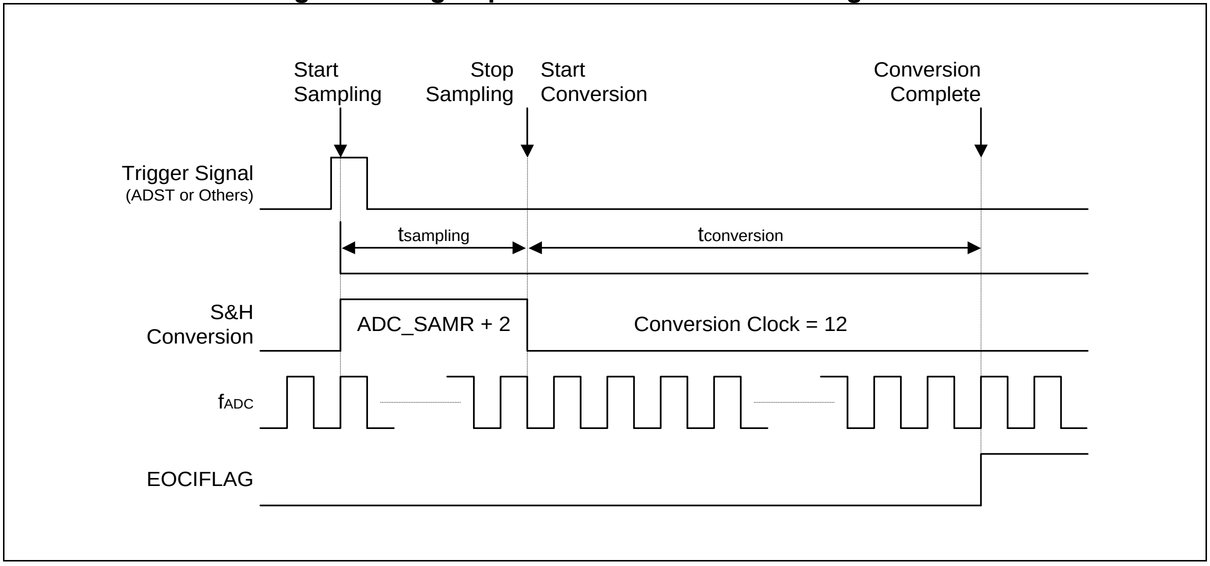
Conversion clock = (ADC_SAMR + 2) + 12 [clocks]

Table 62. ADC Frequency Set according to AVDD

AVDD Range	Max. f_{ADC}	ADC_PREDR (Ex: PCLK = 32MHz)
2.7V ≤ AVDD ≤ 3.6V	Up to 16MHz	0x1 or more
1.71V ≤ AVDD ≤ 3.6V	Up to 8MHz	0x2 or more

NOTE: On low or high temperature, set the ADC frequency lower than the above table.

Figure 68. High Speed ADC Conversion Timing Chart



14.4.4 ADC conversion mode

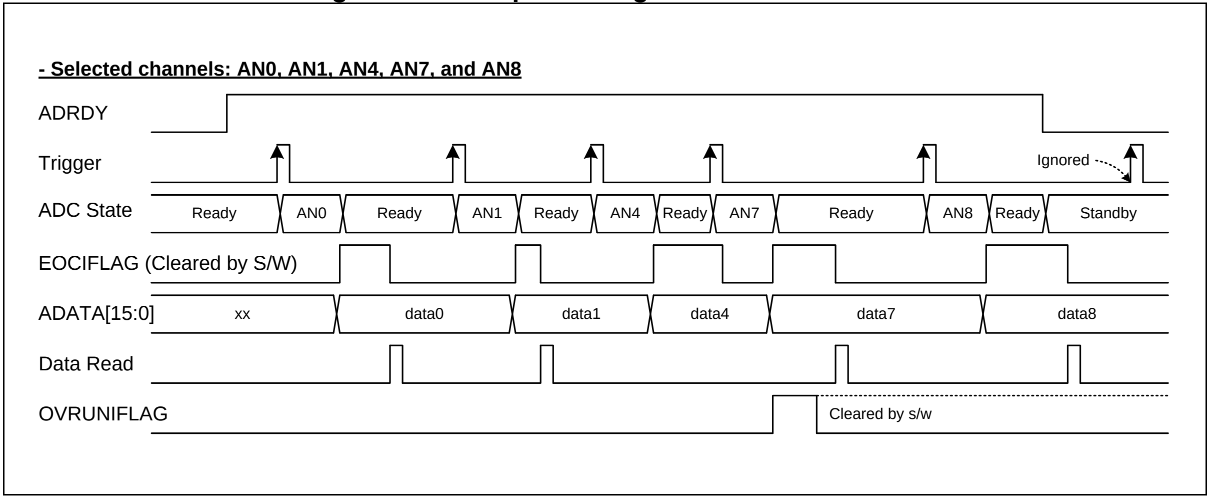
There are three modes for ADC such as Single conversion mode, Sequential conversion mode and Continuous conversion mode. A mode is selected by the MDSEL[1:0] bits of ADC_CR register.

14.4.4.1 Single conversion mode

The ADC converts one of the selected channels in order every trigger signal during single conversion mode. Analog input signal is selected by ADC_CHSELR register.

The end of conversion interrupt flag, the EOCIFLAG bit of ADC_IESR register, is set to “1b” as soon as a new conversion data result is available. The EOCIFLAG bit is cleared by software by writing “1b” to it. An ADC data overrun interrupt flag is set to “1b” if a trigger finishes a new conversion while the previous conversion data are not read.

Figure 69. Example of Single Conversion Mode

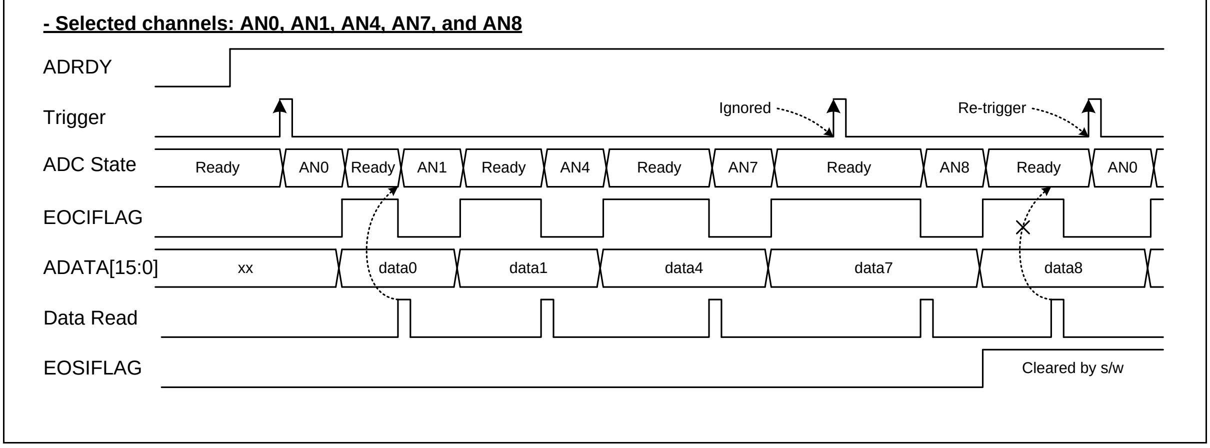


14.4.4.2 Sequential conversion mode

The ADC converts all selected channels in order by a trigger signal during Sequential conversion mode. All trigger signals are ignored during a sequence procedure. The next conversion starts immediately after data read.

The conversion sequence is terminated after all selected channels are converted. The end of sequence interrupt flag, the EOSIFLAG bit of ADC_IESR register, is set to “1b” as soon as the last data result of a sequence is available. The EOSIFLAG bit is cleared by software by writing “1b” to it.

Figure 70. Example of Sequential Conversion Mode

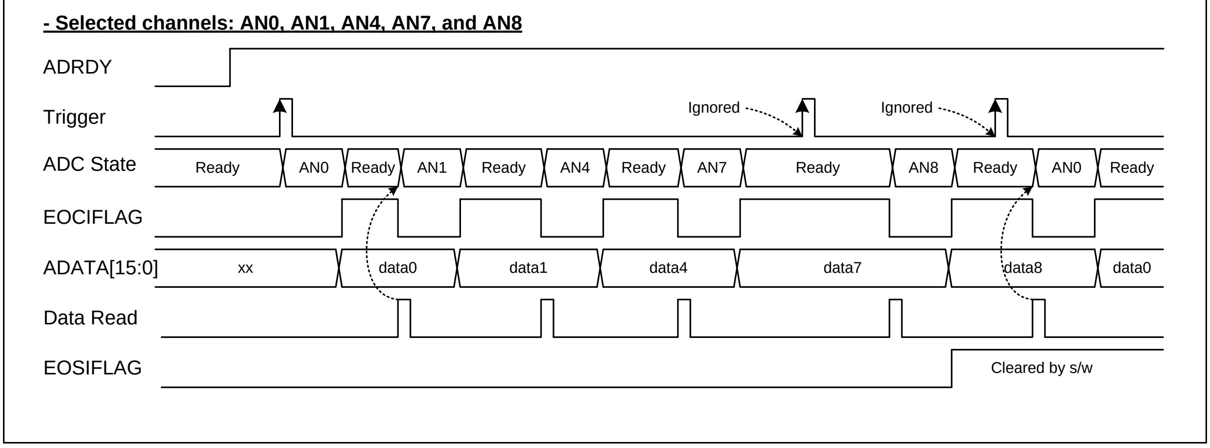


14.4.4.3 Continuous conversion mode

The ADC repeatedly converts all selected channels in order by a trigger signal during Continuous conversion mode. All trigger signals are ignored on the Continuous conversion mode. The next conversion starts immediately after data read as in the sequential conversion mode.

The end of sequence interrupt flag, the EOSIFLAG bit of ADC_IESR register, is also set to “1b” as soon as the last data result of a sequence is available, but the next conversion sequence is continued until a termination by software. The continuous conversion can be terminated by writing “0b” to the ADRDY bit of ADC_CR register.

Figure 71. Example of Continuous Conversion Mode



14.4.5 ADC oversampling

The ADC has oversampling function by hardware for averaging, SNR improvement, and filtering. The function can handle multiple conversions and average them into a single data width, up to 16-bit. The oversampling ratio is configured by the OVSMPR[2:0] bits of ADC_OVSCR register with enabling the oversampling. The range is from x2 to x256. The ADC block has 20-bit accumulator for all sums of sampling data (256 x 12-bit: 20-bit). The average result consists of a right bit shift up to 8-bit. The right bit shift is selected by the OVSHT[3:0] bits of ADC_OVSCR register.

$$\text{Average Result} = \sum_{n=1}^{n=2^{OVSMPR[2:0]+1}} Data_n \gg \text{OVSHT[3:0]}$$

The upper bits of the average result are truncated with only the 16 least significant bits before being transferred into the ADC_DR register.

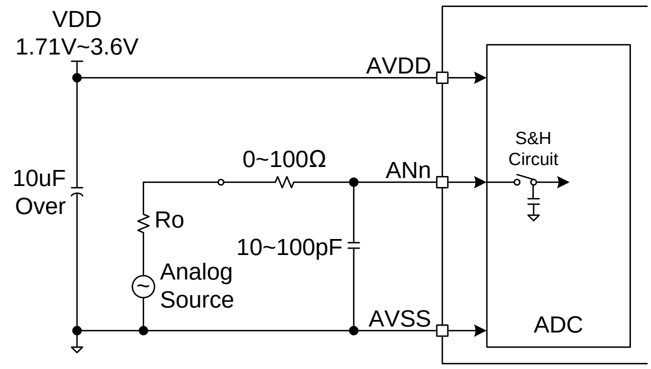
Table 63. ADC Result Data

OVSMPEN	OVSHT[3:0]	ADC_DR Register															
		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Don’t care	0	0	0	0	ADATA[11:0]											
1	0	ACCUM[15:0] → ADATA[15:0]															
1	1	ACCUM[16:1] → ADATA[15:0]															
1	2	ACCUM[17:2] → ADATA[15:0]															
1	3	ACCUM[18:3] → ADATA[15:0]															
1	4	ACCUM[19:4] → ADATA[15:0]															
1	5	0	ACCUM[19:5] → ADATA[14:0]														
1	6	0	0	ACCUM[19:6] → ADATA[13:0]													
1	7	0	0	0	ACCUM[19:7] → ADATA[12:0]												
1	8	0	0	0	0	ACCUM[19:8] → ADATA[11:0]											

14.4.6 ADC recommend circuit

An output resistor (Ro) of analog source increases the capacitor charging time of the circuit. It may degrade the accuracy of ADC. The charging time depends on the resistor and capacitor of an input circuit. So, the sampling time should be adjusted appropriately by the ADC sampling time register (ADC_SAMR). The interval time of conversion should also be adjusted for accuracy.

Figure 72. Recommend Circuit for ADC Input



15 Comparator 0/1

The A31L21x series includes two comparator modules. Each comparator module has three registers such as a control register (CMP_CR), a status register (CMP_SR), and a reference control register (CMP_RCR). The comparator module has an internal reference circuit too.

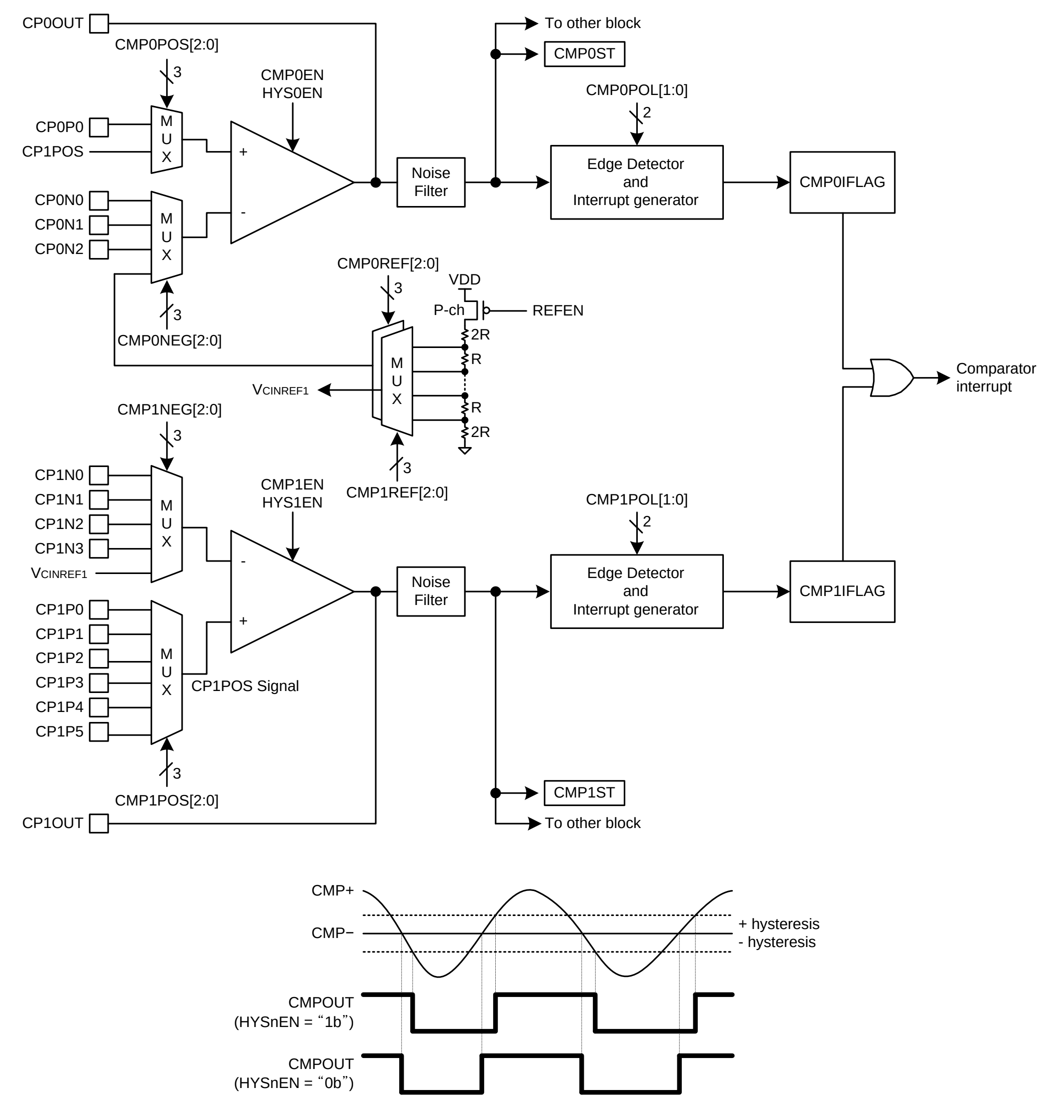
The comparator module features the followings:

- External analog inputs
- Hysteresis function
- Low and fast speed selectable
- Wake-up possible from DEEP SLEEP mode

15.1 Comparator 0/1 block diagram

Figure 73 shows a block diagram of the comparator block.

Figure 73. Comparator 0/1 Block Diagram



NOTE: The VDD should be greater than or equal to 2.0V and the corresponding pins are set to push-pull output by the PD_TYP[4:0] bits if CP1P[5:2] and CP1N3 are used as comparator pins for alternative function.

15.2 Pin description for Comparator 0/1

Table 64. Pins and External Signals for Comparator 0/1

Pin name	Type	Description
CP0P0	A	Comparator 0 positive input
CP0N0	A	Comparator 0 negative input
CP0N1	A	Comparator 0 negative input
CP0N2	A	Comparator 0 negative input
CP0OUT	A	Comparator 0 output
CP1P0	A	Comparator 1 positive input
CP1P1	A	Comparator 1 positive input
CP1P2	A	Comparator 1 positive input
CP1P3	A	Comparator 1 positive input
CP1P4	A	Comparator 1 positive input
CP1P5	A	Comparator 1 positive input
CP1N0	A	Comparator 1 negative input
CP1N1	A	Comparator 1 negative input
CP1N2	A	Comparator 1 negative input
CP1N3	A	Comparator 1 negative input
CP1OUT	A	Comparator 1 output

15.3 Registers

Base address and register map of the Comparator 0/1 are shown in Table 65 and Table 66.

Table 65. Base Address of Comparator 0/1

Name	Base address	Size	Description
CMP0	0x4000_5600	128	Comparator 0
CMP1	0x4000_5680	128	Comparator 1

Table 66. Comparator n Register Map (n = 0 and 1)

Name	Offset	Type	Description	Reset value
CMPn_CR	0x0000	RW	Comparator n Control Register	0x00000000
CMPn_SR	0x0004	RW	Comparator n Status Register	0x00000000
CMPn_RCR	0x0008	RW	Comparator n Reference Control Register	0x00000000

15.3.1 CMPn_CR: comparator n control register

CMPn_CR register is 32-bit size and accessible in 32/16/8-bit. (n = 0 and 1)

CMP0_CR=0x4000_5600, CMP1_CR=0x4000_5680

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																**CMPnEN**	**CMPnNEG**			**CMPnPOS**			**HYSnEN**	**Reserved**	**CMPnSPD**	**CMPnPOL**		**Reserved**	**NFCKn**		
0x0000																**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
–																**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**-**	**RW**	**RW**	**RW**	**-**	**RW**	**RW**	**RW**

15	CMPnEN	Comparator n Enable. (It isn't automatically disabled at power down)	
		0	Disable comparator n operation.
		1	Enable comparator n operation.
14	CMPnNEG	Comparator n Negative Input Selection.	
12		000	Select external CPnN0 pin.
		001	Select external CPnN1 pin.
		010	Select external CPnN2 pin.
		011	Select external CPnN3 pin (Reserved on comparator 0).
		111	Select internal reference.
		Others	Reserved
11	CMPnPOS	Comparator n Positive Input Selection.	
9		000	Select external CPnP0 pin.
		001	Select external CPnP1 pin (CP1POS signal on comparator 0).
		010	Select external CPnP2 pin (Reserved on comparator 0).
		011	Select external CPnP3 pin (Reserved on comparator 0).
		100	Select external CPnP4 pin (Reserved on comparator 0)
		101	Select external CPnP5 pin (Reserved on comparator 0).
		Others	Reserved
8	HYSnEN	Comparator n Hysteresis Enable.	
		0	Disable hysteresis function.
		1	Enable hysteresis function.
6	CMPnSPD	Comparator n Speed Selection.	
		0	Slow speed.
		1	Fast speed.
5	CMPnPOL	Comparator n Interrupt Polarity Selection.	
4		00	No interrupt at any edge.
		01	Interrupt on falling edge
		10	Interrupt on rising edge
		11	Interrupt on both of falling and rising edge
2	NFCKn	Comparator n Noise Filter Sampling Clock Selection.	
0		000	PCLK/1
		001	PCLK/2
		010	PCLK/4
		011	PCLK/8
		100	PCLK/16
		101	PCLK/32
		110	PCLK/64
		111	Reserved

NOTES:

3. If a level is not detected three or more times in a row at the sampling clock, the signal is eliminated as noise.
4. A pulse level should be input for the duration of 3 clocks or more to be actually detected as a valid edge.
5. The comparator noise filter is automatically disabled at DEEP SLEEP mode and recovered after DEEP SLEEP mode release.

15.3.2 CMPn_SR: comparator n status register

CMPn_SR register is 32-bit size and accessible in 32/16/8-bit. (n = 0 and 1)

CMP0_SR=0x4000_5604, CMP1_SR=0x4000_5684

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
Reserved			Reserved			CMPnIFLAG	Reserved			CMPnST
0x000000			0	0	0	0	0	0	0	0
-			-	-	-	RW	-	-	-	RO

4	CMPnIFLAG	Comparator n Interrupt Flag.
		0 No request occurred.
		1 Request occurred, This bit is cleared to ‘0’ when write ‘1’.
0	CMPnST	Comparator n Output Status.
		0 Comparator n output is low.
		1 Comparator n output is high

15.3.3 CMPn_RCR: comparator n reference control register

CMPn_RCR register is 32-bit size and accessible in 32/16/8-bit. (n = 0 and 1)

CMP0_RCR=0x4000_5608, CMP1_RCR=0x4000_5688

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
Reserved			REFnEN	Reserved				CMPnREF		
0x000000			0	0	0	0	0	0	0	0
-			RW	-	-	-	-	RW	RW	RW

7	REFnEN	Comparator n Internal Reference Enable.
		0 Disable internal reference.
		1 Enable internal reference.
		NOTE: This bit is only in the comparator 0 reference control register (CMP0RCR).
2	CMPnREF	Comparator n Reference Voltage Level Selection.
0		000 Select reference voltage level 0
		001 Select reference voltage level 1
		010 Select reference voltage level 2
		011 Select reference voltage level 3
		100 Select reference voltage level 4
		101 Select reference voltage level 5
		110 Select reference voltage level 6
		111 Select reference voltage level 7
		NOTE: Reference voltage = (2+k)xVDD÷11, k: 0 to 7

16 USART 10/11

USART (Universal Synchronous and Asynchronous serial Receiver and Transmitter) is a highly flexible serial communication device. The USART of the A31L21x series features the followings:

- Full Duplex Operation. (Independent Serial Receive and Transmit Registers)
- Asynchronous or Synchronous Operation
- Baud Rate Generator
- Supports Serial Frames with 5,6,7,8, or 9 Data bits and 1 or 2 Stop bits
- Odd or Even Parity Generation, and Parity Check Supported by Hardware.
- Supports Receive Character Detection and Receive Time Out Function
- Supports Local Interconnection Network (LIN)
- Data OverRun Detection
- Framing Error Detection
- Three Separate Interrupts on TX Completion, TX Data Register Empty and RX Completion
- Double Speed Asynchronous communication mode
- Up to 16MHz data transfer for SPI

16.1 USART 10/11 block diagram

Figure 74 shows a block diagram of the UART and LIN block.

Figure 74. UART and LIN Block Diagram of USART (n = 10 and 11)

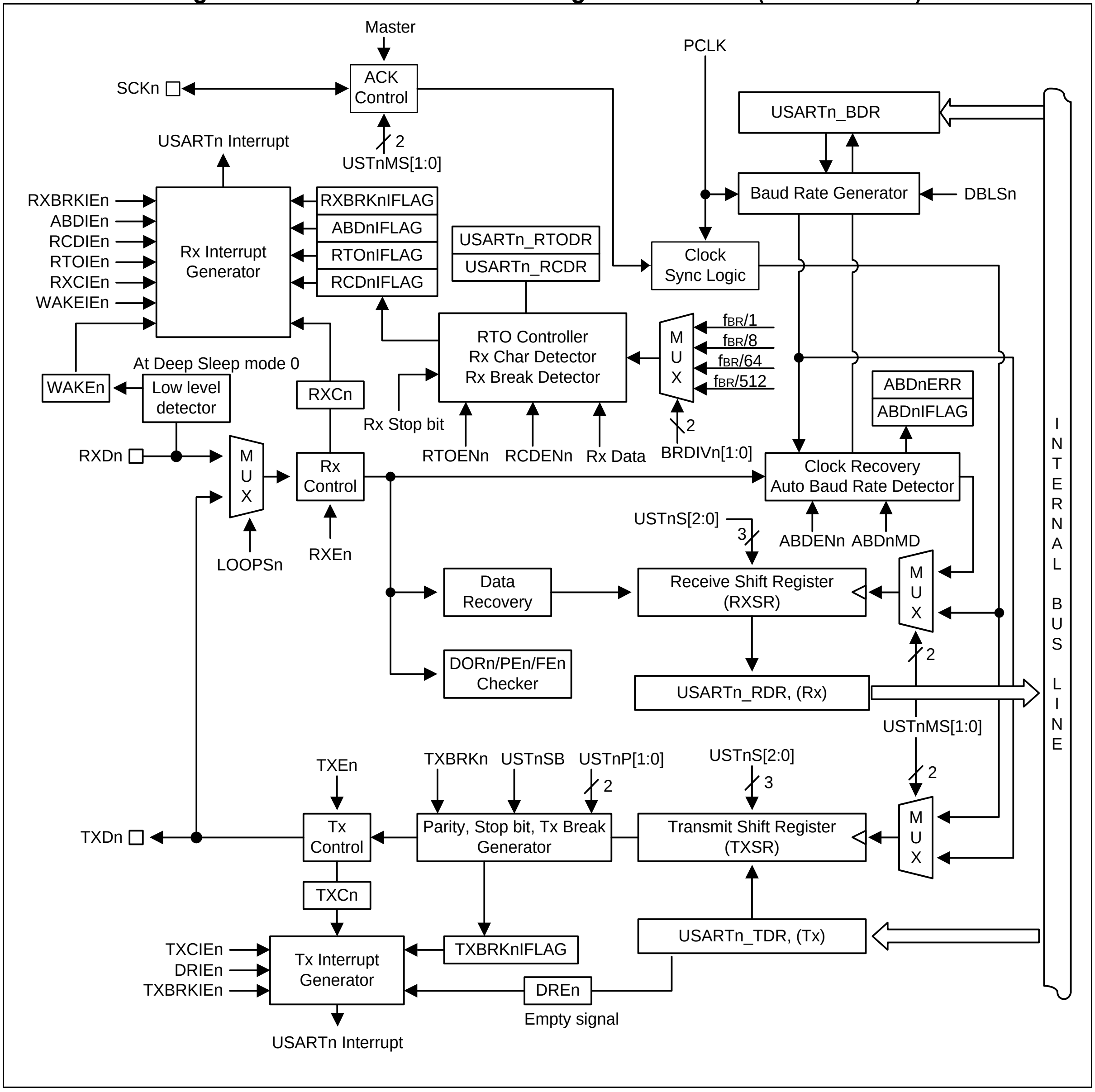
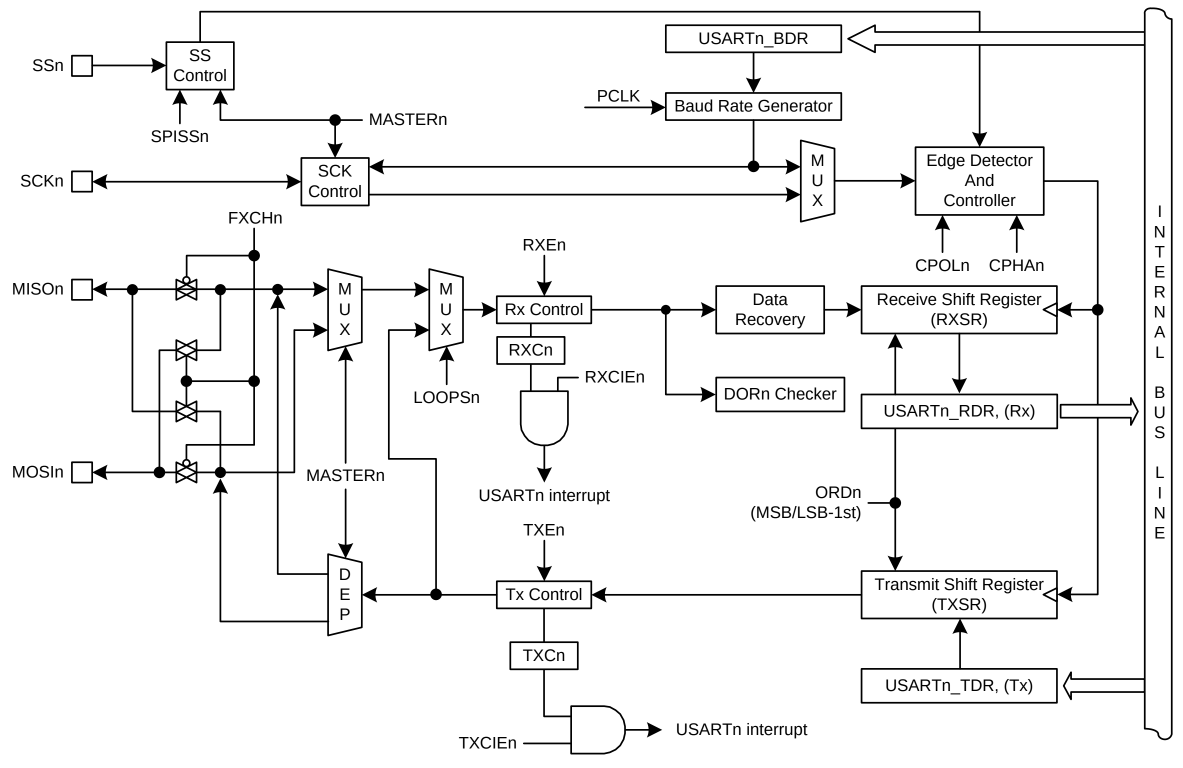


Figure 75 shows a block diagram of the SPI block.

Figure 75. SPIn Block Diagram of USART (n = 10 and 11)



NOTES:

1. The clock frequency of SCKn should be less than or equal to PCLK/2.
2. When DMA is used, the interval of data to be received requires at least 32 clocks of PCLK.

16.2 Pin description for USART 10/11

Table 67. Pins and External Signals for USART 10/11

Pin name	Type	Description
TXDn	O	UART Channel n transmit output
RXDn	I	UART Channel n receive input
SSn	I/O	SPIn Slave select input /output
SCKn	I/O	SPIn Serial clock input/output
MOSIn	I/O	SPIn Serial data (Master output, Slave input)
MISOn	I/O	SPIn Serial data (Master input, Slave output)

16.3 Registers

Base address and register map of the USART 10/11 are shown in Table 68 and Table 69.

Table 68. Base Address of USART 10/11

Name	Base address	Size	Description
USART 10	0x4000_3800	256	USART 10 block (UART 10 + SPI 10)
USART 11	0x4000_3900	256	USART 11 block (UART 11 + SPI 11)

Table 69. USART n Register Map (n = 10 and 11)

Name	Offset	Type	Description	Reset value
USARTn_CR1	0x00	RW	USARTn control register 1	0x00000000
USARTn_CR2	0x04	RW	USARTn control register 2	0x00000000
USARTn_CR3	0x08	RW	USARTn control register 3	0x00000000
USARTn_ST	0x0C	RW	USARTn status register	0x00000080
USARTn_BDR	0x10	RW	USARTn baud rate generation register	0x00000FFF
USARTn_RDR	0x14	RO	USARTn receive data register	0x00000000
USARTn_TDR	0x18	RW	USARTn transmit data register	0x00000000
USARTn_RTODR	0x1C	RW	USARTn receive time out data register	0x000000FF
USARTn_RCDR	0x20	RW	USARTn receive character detection data register	0x00000000

16.3.1 USARTn_CR1: USARTn control register 1

USART module should be configured properly before running.

USARTn_CR1 register is 32-bit size and accessible in 32/16/8-bit. (n = 10 and 11)

USART10_CR1=0x4000_3800, USART11_CR1=0x4000_3900

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14	13 12	11 10 9	8	7	6	5	4	3	2	1	0
Reserved	**USTnMS**	**USTnP**	**USTnS**	**ORDn**	**CPOLn**	**CPHAn**	**DRIEn**	**TXCIEn**	**RXCIEn**	**WAKEIEn**	**TXEn**	**RXEn**
0x0000	**00**	**00**	**000**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
-	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**

15 14	USTnMS	USARTn Operation Mode Selection.			
		00	Asynchronous Mode. (UART)		
		01	Synchronous Mode.		
		10	Local Interconnection Network (LIN)		
		11	SPI mode		
		NOTE: The LIN transmission is fixed as “Start, D0, D1, D2, D3, D4, D5, D6, D7, Stop1”. So, the USTnP[1:0], LOOPSn, and USTnSB bits must be cleared to ‘0’ and the USTnS[2:0] bits should be set to “011b”.			
13 12	USTnP	Selects Parity Generation and Check method. (only UART mode)			
		00	No parity.		
		01	Reserved.		
		10	Even parity.		
		11	Odd parity.		
11 9	USTnS	Selects the length of data bit in a frame at Asynchronous or Synchronous mode.			
		000	5 bit.		
		001	6 bit.		
		010	7 bit.		
		011	8 bit.		
		111	9 bit.		
		Others	Reserved.		
8	ORDn	Selects the first data bit to be transmitted. (only SPI mode)			
		0	LSB-first.		
		1	MSB-first.		
7	CPOLn	Selects the clock polarity of SCK in synchronous or SPI mode.			
		0	SCK to 0 when idle.		
		1	SCK to 1 when idle.		
6	CPHAn	CPOLn and this bit determine if data are sampled on the leading or the trailing edge of SCK. (only SPI mode)			
		CPOLn	CPHAn	Leading edge	Trailing edge
		0	0	Sample (Rising)	Setup (Falling)
		0	1	Setup (Rising)	Sample (Falling)
		1	0	Sample (Falling)	Setup (Rising)
		1	1	Setup (Falling)	Sample (Rising)
5	DRIEn	Transmit Data Register Empty Interrupt Enable.			
		0	Disable transmit data empty interrupt.		
		1	Enable transmit data empty interrupt.		
4	TXCIEn	Transmit Complete Interrupt Enable.			
		0	Disable transmit complete interrupt.		
		1	Enable transmit complete interrupt.		
3	RXCIEn	Receive Complete Interrupt Enable.			
		0	Disable receive complete interrupt.		
		1	Enable receive complete interrupt.		
2	WAKEIEn	Asynchronous Wake-up Interrupt Enable in DEEP SLEEP Mode. When the device is in DEEP SLEEP mode, if RXDn goes to low level, an interrupt can be requested to wake-up system (only UART and LIN mode). This bit should be cleared to ‘0’ to receive Rx data.			

		0	Disable asynchronous wake-up interrupt.
		1	Enable asynchronous wake-up interrupt. (Only used for wake-up)
1	TXEn	Enables the Transmitter unit.	
		0	Transmitter is disabled.
		1	Transmitter is enabled.
0	RXEn	Enables the Receiver unit.	
		0	Receiver is disabled.
		1	Receiver is enabled.

NOTE: The CPOLn and CPHAn bits should be changed while TXEn and RXEn bits are ‘0’.

16.3.2 USARTn_CR2: USARTn control register 2

USART module should be configured properly before running.

USARTn_CR2 register is 32-bit size and accessible in 32/16/8-bit. (n = 10 and 11)

USART10_CR2=0x4000_3804, USART11_CR2=0x4000_3904

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="22">Reserved</td><td>USTnEN</td><td>DBLSn</td><td>MASTERn</td><td>LOOPSn</td><td>DISSCKn</td><td>USTnSSEN</td><td>FXCHn</td><td>USTnSB</td><td colspan="2">Reserved</td></tr>
<tr><td colspan="22">0x00000</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td colspan="22">-</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>-</td><td>-</td></tr>
</table>

9	USTnEN	Enable USARTn block. This bit can be cleared to ‘0b’ during the corresponding TXEn and RXEn bits are all ‘0b’. **NOTE**: This bit should be set to “1b” after setting the related registers.	
		0	Disable USARTn block.
		1	Enable USARTn block.
8	DBLSn	Selects receiver sampling rate. (only asynchronous and LIN mode)	
		0	Normal asynchronous operation.
		1	Double speed asynchronous operation.
7	MASTERn	Selects master or slave in SPIn or Synchronous mode and controls the direction of SCKn pin.	
		0	Slave operation. (External clock for SCKn)
		1	Master operation. (Internal clock for SCKn)
6	LOOPSn	1. 1-wire Half-Duplex Communication on Asynchronous Mode.	
		0	Normal operation.
		1	1-wire half-duplex communication (The TXD and RXD lines are internally connected, the RXD pin is not used, and the TXD pin is always an input when no transmitted. So, the TXD pin must be configured to open-drain with an external pull-up resistor)
		2. Loop Back for Test on SPI and Synchronous Mode	
		0	Normal operation.
		1	Loop back (The “MOSI and MISO”/”TXD and RXD” lines are internally connected and the receive input is not used).
5	DISSCKn	In synchronous mode operation, selects the waveform of SCKn output.	
		0	SCKn is free-running while USARTn is enabled in synchronous master mode.
		1	SCKn is active while any frame is transferring.
4	USTnSSEN	This bit controls the SSn pin operation. (only SPI mode)	
		0	Disable.
		1	Enable. (The SS pin should be configured as an alternative function)
3	FXCHn	SPIn port function exchange control. (only SPI mode)	
		0	No effect.
		1	Exchange MOSIn and MISOn function.
2	USTnSB	Selects the length of stop bit in Asynchronous or Synchronous mode.	
		0	1 Stop bit.
		1	2 Stop bit.

16.3.3 USARTn_CR3: USARTn control register 3

USART module should be configured properly before running. This register is used only for UART and LIN mode. USARTn_CR3 register is 32-bit size and accessible in 32/16/8-bit. (n = 10 and 11)

USART10_CR3=0x4000_3808, USART11_CR3=0x4000_3908

31 30 29 28 27 26 25 24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1 0
Reserved	**RXBRKn**	**TXBRKn**	**RXBRKIEn**	**TXBRKIEn**	**RXBRKnIFLAG**	**TXBRKnIFLAG**	**Reserved**	**ABDENn**	**ABDnMD**	**ABDIEn**	**ABDnIFLAG**	**ABDnERR**	**Reserved**	**RCDENn**	**RTOENn**	**Reserved**	**RCDIENn**	**RTOIENn**	**Reserved**	**RCDnIFLAG**	**RTOnIFLAG**	**Reserved**	**BRDIVn**
0x00	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**00**
-	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**-**	**RW**	**RW**	**RW**	**RW**	**RW**	**-**	**RW**	**RW**	**-**	**RW**	**RW**	**-**	**RW**	**RW**	**-**	**RW**

23 RXBRKn	Rx Break Field Length (Only LIN mode)	
	0	10 ‘0’ bits or more
	1	11 ‘0’ bits or more
22 TXBRKn	Tx Break Field (Only LIN mode)	
	0	No effect
	1	Transmit Tx break (13 ‘0’ bits + 2 ‘1’ bits, 000000000000011b, automatically cleared)
21 RXBRKIEn	Rx Break Field Detection Interrupt Enable bit (Only LIN mode)	
	0	Disable Rx break detection interrupt
	1	Enable Rx break detection interrupt
20 TXBRKIEn	Tx Break Field Completion Interrupt Enable bit (Only LIN mode)	
	0	Disable Tx break completion interrupt
	1	Enable Tx break completion interrupt
19 RXBRKnIFLAG	Rx Break Field Detection Interrupt flag. This bit is set when LIN break is detected.	
	0	No request occurred
	1	Request occurred. This bit is cleared to ‘0’ when write ‘1’.
18 TXBRKnIFLAG	Tx Break Field Completion Interrupt flag. This bit is set when Tx break field is completely transmitted.	
	0	No request occurred
	1	Request occurred. This bit is cleared to ‘0’ when write ‘1’.
16 ABDENn	Auto Baud Rate Detection Enable bit.	
	0	Disable auto baud rate detection
	1	Enable auto baud rate detection (This bit is automatically cleared after operation)
	NOTE: In the LIN mode, the Rx break field may not be detected while this bit is “1b”. So, it is recommended to set this bit to “1b” for auto baud rate detection after the Rx break field detection.	
15 ABDnMD	Auto Baud Rate Detection Mode	
	0	Mode 0, The start bit is used to measure the baud rate (The 1ST bit must be “1b”)
	1	Mode 1, The 0x55 character is used to measure the baud rate detection
14 ABDIEn	Auto Baud Rate Detection Interrupt Enable bit	
	0	Disable auto baud rate detection interrupt
	1	Enable auto baud rate detection interrupt
13 ABDnIFLAG	Auto Baud Rate Detection Interrupt Flag. This bit is set to “1b” when the auto baud rate detection finishes, whether an error occurs or not.	
	0	No request occurred
	1	Request occurred. This bit is cleared to ‘0’ when ‘1’ is written
12 ABDnERR	Auto Baud Rate Detection Error bit. This bit is set to “1b” if the clock counting values are not between 16 and 65536 on the normal speed operation (DBLSn = 0) or between 8 and 32768 on the double speed operation (DBLSn = 1).	
	0	No error occurs

		1	An error occurs. This bit is cleared to ‘0’ when write ‘1’.
		NOTE:	If an error occurs, the USARTn_BDR register will not be updated.
10	RCDENn		Receive Character Detection Function Enable bit. This function is to compare the value of USARTn_RCDR register with the value just received.
		0	Disable receive detection function.
		1	Enable receive detection function.
9	RTOENn		Receive Time Out Function Enable bit. This function is to count time with baud rate units from the leading edge of a start bit to a new start bit. The receive time out controller counts down from the value of USARTn_RTODR register every start bit and set this bit. The RTOnIFLAG bit is set to “1b” at the counter underflow (only asynchronous mode).
		0	Disable receive time out function.
		1	Enable receive time out function.
7	RCDIEn		Receive Character Detection Interrupt Enable.
		0	Disable receive character detection interrupt
		1	Enable receive character detection interrupt
6	RTOIEn		Receive Time Out Interrupt Enable.
		0	Disable receive time out interrupt.
		1	Enable receive time out interrupt
4	RCDnIFLAG		Receive Character detection Interrupt Flag. This bit is set to “1b” if the value in the USARTn_RCDR register matches the value received in the non-error state of frame and parity. On match of them, the bit may be set even if data overrun occurs.
		0	No request occurred.
		1	Request occurred. This bit is cleared to ‘0’ when ‘1’ is written.
3	RTOnIFLAG		Receive Time Out Interrupt Flag
		0	No request occurred.
		1	Request occurred. This bit is cleared to ‘0’ when ‘1’ is written.
1 0	BRDIVn		Baud Rate Clock Dividing Selection for Receive Time Out. (only asynchronous and LIN mode).
		00	f_{BR}/1.
		01	f_{BR}/8
		10	f_{BR}/64
		11	f_{BR}/512

16.3.4 USARTn_ST: USARTn status register

USARTn_ST register is 32-bit size and accessible in 32/16/8-bit. (n = 10 and 11)

USART10_ST=0x4000_380C, SART11_ST=0x4000_390C

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
Reserved	**DREn**	**TXCn**	**RXCn**	**WAKEn**	**Reserved**	**DORn**	**FEn**	**PEn**
0x000000	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
-	**RO**	**RW**	**RO**	**RW**	**-**	**RW**	**RW**	**RW**

7	DREn	Transmit Data Register Empty Interrupt Flag. The flag is set to “1b” when the data in the USARTn_TDR register has been transferred to the transmit shift register. This bit is cleared by a write to the USARTn_TDR register (only UART mode).	
		0	Not transferred to the transmit shift register.
		1	Transferred to the transmit shift register.
6	TXCn	Transmit Complete Interrupt Flag. This flag is set to “1b” when the data in the transmit shift register has been shifted out and when the DREn = 1.	
		0	No request occurred.
		1	The data in the transmit shift register are shifted out completely. This bit is cleared to ‘0’ when write ‘1’.
5	RXCn	Receive Data Register Not Empty Interrupt Flag. This bit is set to “1b” when the data in the receive shift register has been transferred to the USARTn_RDR register. The bit is cleared by a read to the USARTn_RDR register.	
		0	No request occurred.
		1	There is data in the receive data register. This bit is cleared to ‘0’ when write ‘1’.
4	WAKEn	Asynchronous Wake-up Interrupt Flag. This flag is set when the RXD pin is detected low while the CPU is in DEEP SLEEP mode (only UART mode)	
		0	No request occurred.
		1	Request occurred. This bit is cleared to ‘0’ when write ‘1’.
2	DORn	Data Overrun bit. This bit is set when the receive shift register is transferred to the USARTn_RDR register while the RXCn=1. The data of the shift register are ignored. This bit must be cleared by S/W to receive new data (only UART mode).	
		0	No Data OverRun.
		1	Data overrun detected. This bit is cleared to ‘0’ when write ‘1’.
1	FEn	Frame Error bit. This bit is set when the received data have not a valid stop bit. That is, the stop bit following the last data bit is detected as “0b”. The bit will be cleared by H/W if new data are received (only UART mode).	
		0	No Frame Error.
		1	Frame error detected. This bit is cleared to ‘0’ when write ‘1’.
0	PEn	Parity Error bit. This bit is set when the received data has a parity error on parity enable. The bit will be cleared by H/W if new data are received (only UART mode).	
		0	No Parity Error.
		1	Parity error detected. This bit is cleared to ‘0’ when write ‘1’.

16.3.5 USARTn_BDR: USARTn baud rate generation register

USARTn_BDR register is 32-bit size and accessible in 32/16/8-bit. (n = 10 and 11)

USART10_BDR=0x4000_3810, USART11_BDR=0x4000_3910

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																				BDATA											
0x00000																				0xFFF											
-																				RW											

11 0	BDATA	The value in this register is used to generate internal baud rate in UART mode or to generate SCK clock in SPI mode. The range is 0x000 to 0xFFF in asynchronous UART and SPI mode but the range is 0x002 to 0xFFF in synchronous mode.

16.3.6 USARTn_RDR: USARTn receive data register

USARTn_RDR register is 32-bit size and accessible in 32/16/8-bit. (n = 10 and 11)

USART10_RDR=0x4000_3814, USART11_RDR=0x4000_3914

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																							RDATA								
0x000000																							0	0	0	0	0	0	0	0	0
-																							RO								

8 0	RDATA	Receive Data bits. A receive shift register is moved to this register after stop bit.

NOTE: When asynchronous or synchronous mode, the RDATA[8] bit is the received 9th bit.

16.3.7 USARTn_TDR: USARTn transmit data register

USARTn_TDR register is 32-bit size and accessible in 32/16/8-bit. (n = 10 and 11)

USART10_TDR=0x4000_3818, USART11_TDR=0x4000_3918

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																							TDATA								
0x000000																							0	0	0	0	0	0	0	0	0
-																							RW								

8 0	TDATA	Transmit Data bits. This register is moved to the transmit shift register after a previous character is completely shifted out. In SPI master mode, the SCK clock is generated when data are moved to the shift register. Do not write to this transmit data register while transmitting in SPI mode.

NOTES:

1. When asynchronous or synchronous mode, the TDATA[8] bit is the 9th bit to be transmitted.
2. The data to be transmitted should be written after all control registers are set.

16.3.8 USARTn_RTODR: USARTn receive time out data register

USARTn_RTODR register is 32-bit size and accessible in 32/16/8-bit. (n = 10 and 11)

USART10_RTODR=0x4000_381C, USART11_RTODR=0x4000_391C

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="24">Reserved</td><td colspan="8">RTOD</td></tr>
<tr><td colspan="24">0x000000</td><td colspan="8">0xFF</td></tr>
<tr><td colspan="24">-</td><td colspan="8">RW</td></tr>
</table>

7 0	RTOD	USARTn Receive Time Out Data. Counting number: RTOD[7:0] +1

16.3.9 USARTn_RCDR: USARTn receive character detection data register

USARTn_RCDR register is 32-bit size and accessible in 32/16/8-bit. (n = 10 and 11)

USART10_RCDR=0x4000_3820, USART11_RCDR=0x4000_3920

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="24">Reserved</td><td colspan="8">RCDD</td></tr>
<tr><td colspan="24">0x000000</td><td colspan="8">0x00</td></tr>
<tr><td colspan="24">-</td><td colspan="8">RW</td></tr>
</table>

7 0	RCDD	USARTn Receive Character Detection Data.

16.4 Functional description

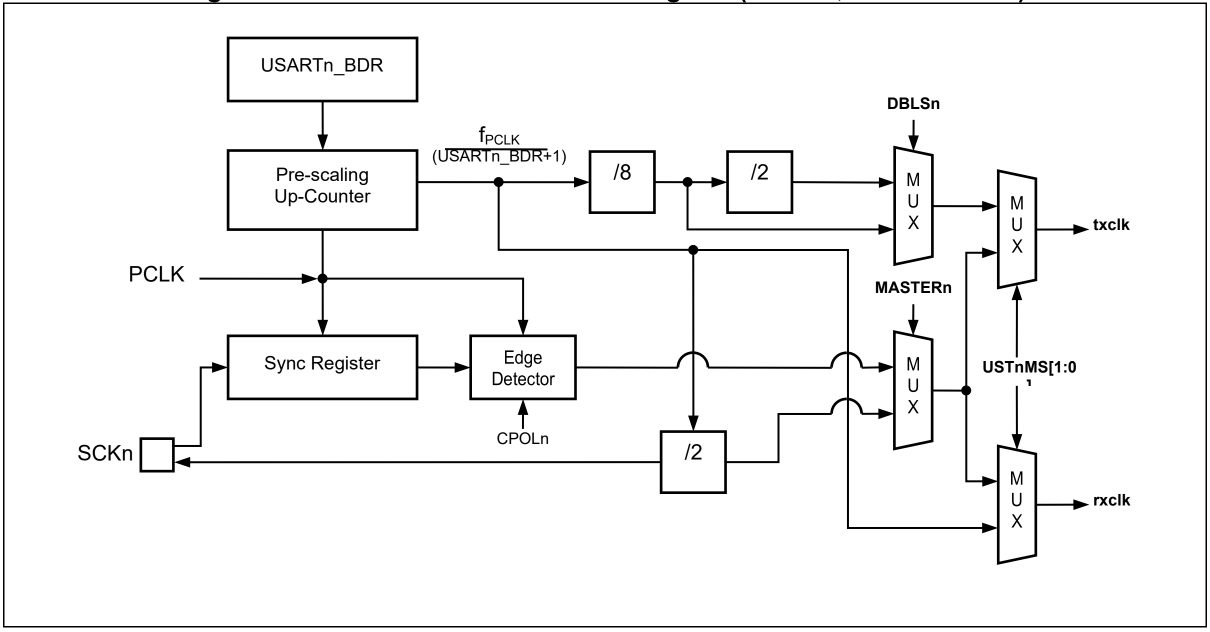
The USART comprises a clock generator, a transmitter, and a receiver. The clock generation logic includes synchronization logic for external clock input, which is used for synchronizing or SPI slave operation. Baud rate generator in the clock generation logic is for asynchronous or master (synchronous or SPI) operation.

The Transmitter consists of a write buffer, a serial shift register, a parity generator, and a control logic. Using DMA allows continuous transfer of data without any s/w involvement between frames.

The receiver is the most complex part of the USART module due to its clock and data recovery units. The recovery unit is used for asynchronous data reception. In addition, the receiver has a parity checker, a shift register, and a control logic. The receiver supports the same frame formats as the transmitter and can detect frame error, data overrun and parity errors. (n = 10 and 11)

16.4.1 USART clock generation

Figure 76. Clock Generation Block Diagram (USART, n = 10 and 11)



The clock generation logic generates the base clock for the transmitter and receiver. The USART supports four modes of clock operation, which are Normal asynchronous mode, Double speed asynchronous mode, Master synchronous mode and Slave synchronous mode.

The clock generation scheme for master SPI and slave SPI mode is the same as master synchronous and slave synchronous operation mode. The USTnMS[1:0] bits in USARTn_CR1 register selects asynchronous or synchronous operation. Asynchronous double speed mode is controlled by the DBLS bit in the USARTn_CR2 register.

The MASTER bit in USARTn_CR2 register controls whether the clock source is internal (master mode, output pin) or external (slave mode, input pin). The SCKn pin is active only when the USART operates in synchronous or SPI mode.

Table 70 shows the equations for calculating the baud rate (in bps).

Table 70. Equations for Calculating USART Baud Rate Register Settings (n = 10 and 11)

Operating mode	Equation for calculating baud rate
Asynchronous Normal Mode (DBLSn=0)	Baud Rate = PCLK/(16(USARTn_BDR+1))
Asynchronous Double Speed Mode (DBLSn=1)	Baud Rate = PCLK/(8(USARTn_BDR+1))
Synchronous or SPI Master Mode	Baud Rate = PCLK/(2(USARTn_BDR+1))

16.4.2 External clock (SCKn)

External clock is used in the Synchronous mode or in the SPI slave mode. External clock input from the SCKn pin is sampled by the synchronization logic to remove meta-stability. The output from the synchronization logic must be passed through an edge detector before it is used by the transmitter and the receiver.

This process introduces two CPU clock period delay. The maximum frequency of the external SCKn pin is limited up to 16MHz.

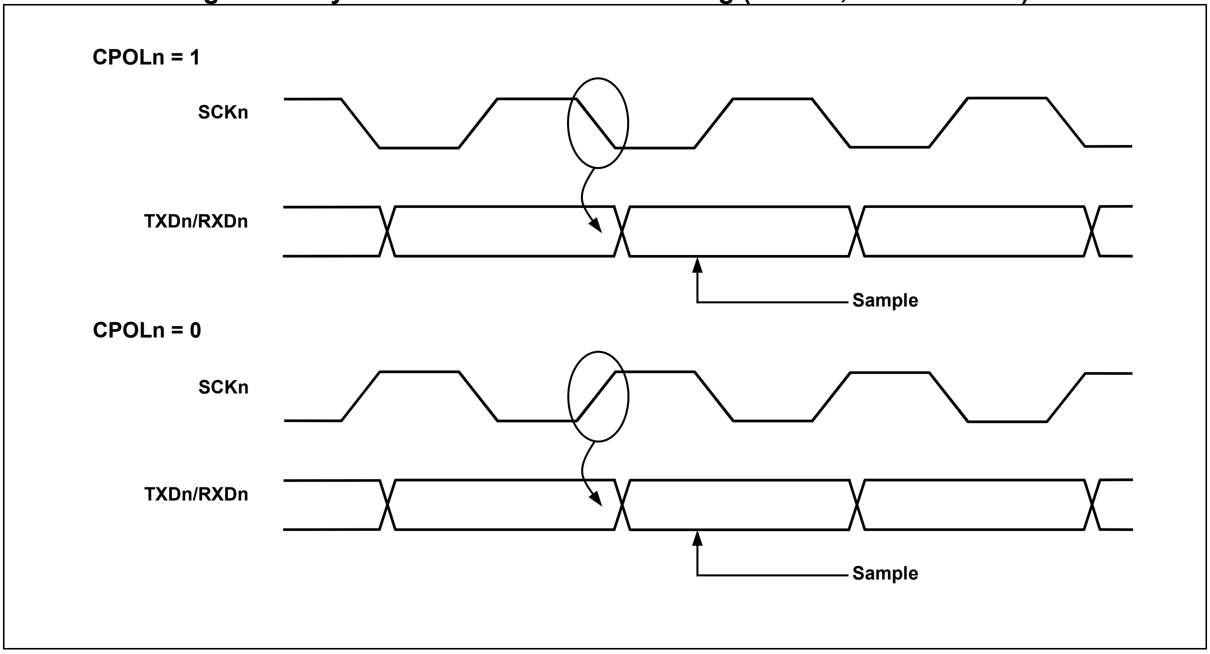
16.4.3 Synchronous mode operation

External clock is used in the Synchronous mode or in the SPI slave mode. When the Synchronous or the SPI mode is used, the SCKn pin will be used as either clock input (slave) or clock output (master).

Data sampling and transmission are issued on different edges of SCKn clock respectively. For example, if data input on RXDn (MISOn in SPI mode) pin is sampled on the rising edge of SCKn clock, data output on TXDn (MOSIn in SPI mode) pin is altered on the falling edge.

The CPOLn bit in USARTn_CR1 register selects which SCKn clock edge is used for data sampling and which is used for data change. As shown in Figure 77 below, when CPOLn is zero, the data will be changed at rising SCKn edge and sampled at falling SCKn edge.

Figure 77. Synchronous Mode SCKn Timing (USART, n = 10 and 11)



16.4.4 UART data format

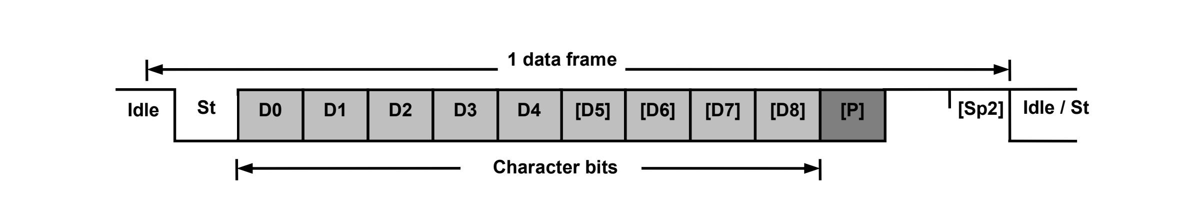
A serial frame is defined to be one character of data bits with synchronization bits (start and stop bits), and optionally a parity bit for error detection. The USART supports all 30 combinations of the followings as valid frame formats.

- 1 start bit
- 5, 6, 7, 8 or 9 data bits
- No, even or odd parity bit.
- 1 or 2 stop bits.

A frame starts with the start bit followed by the least significant data bit (LSB). Then the next data bits, up to nine, follow, ending with the most significant bit (MSB). If a parity function is enabled, the parity bit is inserted between the last data bit and the stop bit. A high-to-low transition on data pin is considered as a start bit.

When a complete frame is transmitted, it can be directly followed by a new frame, or the communication line can be set to an idle state. The idle state means high state of data pin. Figure 78 shows a possible combination of the frame formats. Bits inside brackets are optional.

Figure 78. Frame Format (UART)



1 data frame consists of the following bits

- Idle: No communication on communication line (TXDn/RXDn)
- St: Start bit (low)
- Dm: Data bits (0 to 8)
- P: Parity bit (even parity, odd parity, no parity)
- Sp: Stop bit (1 bit or 2 bits)

The frame format used by the UART is set by USTnS[2:0], USTnP[1:0] bits in the USARTn_CR1 register and USTnSB bit in the USARTn_CR2 register. The transmitter and the receiver use the same values. (n = 10 and 11)

16.4.5 UART parity bit

The parity bit is calculated by doing an exclusive-OR of all data bits. If odd parity is used, the result of the exclusive-OR is inverted. The parity bit is located between the MSB and the first stop bit of a serial frame.

- Peven = Dm-1 ^ … ^ D3 ^ D2 ^ D1 ^ D0 ^ 0
- Podd = Dm-1 ^ … ^ D3 ^ D2 ^ D1 ^ D0 ^ 1
- Peven: Parity bit using even parity
- Podd: Parity bit using odd parity
- Dm: Data bit n of the character

16.4.6 UART transmitter

The UART transmitter is enabled by setting TXEn bit in the USARTn_CR1 register. When the Transmitter is enabled, the TXDn pin should be set to TXDn function for the serial output pin in UART mode by the GPIO registers. Baud-rate, operation mode and frame format must be set up before starting any transmission. In Synchronous operation mode, the SCKn pin is used for transmission clock, so it should be selected to do SCKn function by the GPIO registers. (n = 10 and 11)

16.4.6.1 UART sending TX data

A data transmission is initiated by loading data to the transmit data register (USARTn_TDR). The data to be written in transmit data register is moved to the shift register when the shift register is ready to send a new frame.

The shift register is loaded with the new data if it is in idle state or immediately after the last stop bit of the previous frame is transmitted. When the shift register is loaded to the new data, it will transfer one complete frame according to the settings of the control registers.

If 9-bit characters are used in the Asynchronous or the Synchronous operation mode, the 9th bit must be written to TDATA[8] bit in the USARTn_TDR register. (n = 10 and 11)

16.4.6.2 UART transmitter flag and interrupt

The UART transmitter has two flags that indicate its state. One is USART data register empty flag (DREn) and the other is transmit completion flag (TXCn). Both flags can be used as interrupt sources.

DREn flag indicates whether the transmit buffer is ready to receive new data. This bit is set when the transmit buffer is empty and cleared when the transmit buffer contains data to be transmitted but has not yet been moved into the shift register.

When the data register empty interrupt enable (DRIEn) bit in USARTnCR1 register is set and the global interrupt is enabled, USARTn_ST status register empty interrupt is generated while DREn flag is set.

Transmit complete (TXCn) flag bit is set when the entire frame in the transmit shift register has been shifted out. The TXCn flag can be cleared by writing ‘1’ to TXCn bit in the USARTn_ST register.

When transmit complete interrupt enable (TXCIEn) bit in the USARTn_CR1 register is set and the global interrupt is enabled, UART transmit complete interrupt is generated while TXCn flag is set. (n = 10 and 11)

16.4.6.3 UART parity generator

The parity generator calculates the parity bit for the serial frame data to be sent. When parity bit is enabled (USTnP1=1), the transmitter control logic inserts the parity bit between the MSB and the first stop bit of the frame to be sent. (n = 10 and 11)

16.4.6.4 UART disabling transmitter

Disabling the transmitter by clearing the TXEn bit will not become effective until the current transmission is completed. When the transmitter is disabled, the TXDn pin can be used as a normal general purpose I/O (GPIO). (n = 10 and 11)

16.4.7 UART receiver

The UART receiver is enabled by setting RXEn bit in the USARTn_CR1 register. When the receiver is enabled, the RXDn pin should be set to RXDn function for the serial input pin in the UART mode by the GPIO registers. Baud-rate, operation mode and frame format must be set before serial reception. In Synchronous or SPI operation mode, the SCKn pin is used as a transfer clock input, so it should be selected to do SCKn function by the GPIO registers. (n = 10 and 11)

16.4.7.1 UART receiving RX data

When the UART is in Synchronous mode or in Asynchronous mode, the receiver starts data reception if it detects a valid start bit (LOW) on RXDn pin. Each bit after the start bit is sampled at predefined baud-rate (asynchronous) or at sampling edge of SCKn (synchronous), and shifted into the receive shift register until the first stop bit of a frame is received.

Even if there is a second stop bit in the frame, the second stop bit is ignored by the receiver. That is, receiving the first stop bit means that a complete serial frame is presented in the receiver shift register and contents of the shift register are to be moved into the receive data register (USARTn_RDR). (n = 10 and 11)

16.4.7.2 UART receiver flag and interrupt

The UART receiver has a flag that indicates the receiver’s state. The receive complete (RXCn) flag indicates whether there are unread data in the receive buffer. This flag is set when there is unread data in the receive buffer and cleared when the receive buffer is empty. If the receiver is disabled (RXEn=0), the receiver buffer is flushed and the RXCn flag is cleared.

When the receive complete interrupt enable (RXCIEn) bit in the USARTn_CR1 register is set and global interrupt is enabled, the UART receiver complete interrupt is generated while RXCn flag is set.

The UART receiver has three error flags, which are frame error (FEn), data overrun (DORn) and parity error (PEn). These error flags can be read from the USARTn_ST register.

The frame error (FEn) flag indicates state of the first stop bit. The FEn flag is ‘0’ when the stop bit was correctly detected as ‘1’, while the FEn flag is ‘1’ when the stop bit was incorrect, i.e. detected as ‘0’. This flag can be used for detecting out-of-sync conditions between data frames.

The data overrun (DORn) flag indicates data loss due to a full receive buffer condition. DORn occurs when the receive buffer is full, and another new data is presented in the receive shift register to be stored into the receive buffer. After the DORn flag is set, all the incoming data are lost. To avoid data loss or to clear this flag, receive buffer must be read.

The parity error (PEn) flag indicates that the frame in the receive buffer had a parity error during reception. If parity check function is not enabled (USTnP1=0), the PEn bit is always read as ‘0’. (n = 10 and 11)

16.4.7.3 UART parity checker

If parity bit is enabled (USTnP1=1), the parity checker calculates the parity of the data bits in incoming frame and compares the result with the parity bit from the received serial frame. (n = 10 and 11)

16.4.7.4 UART disabling receiver

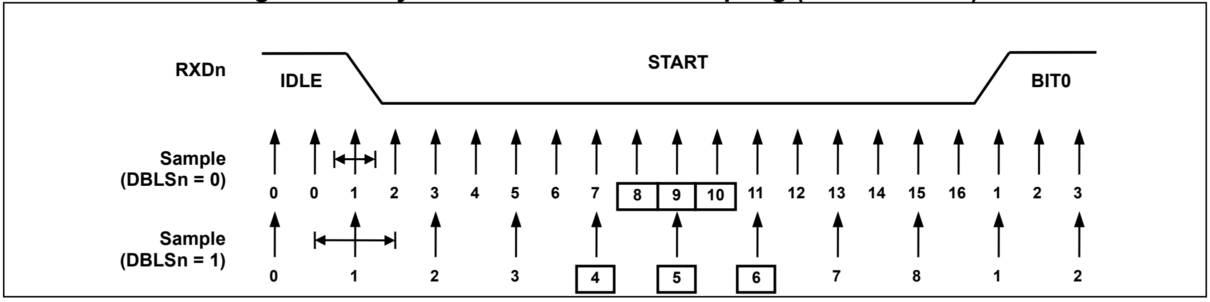
Unlike the transmitter, the receiver becomes inactive immediately after it is disabled by clearing RXEn bit. When the receiver is disabled, the receiver flushes the receive buffer, the remaining data in the buffer is all reset, and the RXDn pin can be used as a normal general purpose I/O (GPIO). (n = 10 and 11)

16.4.7.5 Asynchronous data reception

To receive asynchronous data frame, the USART includes a clock and data recovery unit. The clock recovery logic is used for synchronizing the internally generated baud-rate clock to the incoming asynchronous serial frame on the RXDn pin. The data recovery logic samples and filters the incoming bits with a low pass filter, and removes the noise of RXDn pin.

Figure 79 illustrates the sampling process of the start bit of an incoming frame. The sampling rate is 16 times the baud-rate in normal mode and 8 times the baud-rate for double speed mode (DBLSn=1). The horizontal arrows show the synchronization variation due to the asynchronous sampling process. Note that larger time variation is seen using the double speed mode. (n = 10 and 11)

Figure 79. Asynchronous Start Bit Sampling (n = 10 and 11)



When the receiver is enabled (RXEn=1), the clock recovery logic tries to find a high-to-low transition on the RXDn line, which is the start bit condition. After detecting the high-to-low transition on RXDn line, the clock recovery logic uses samples 8, 9 and 10 for normal mode to detect whether valid start bit is received.

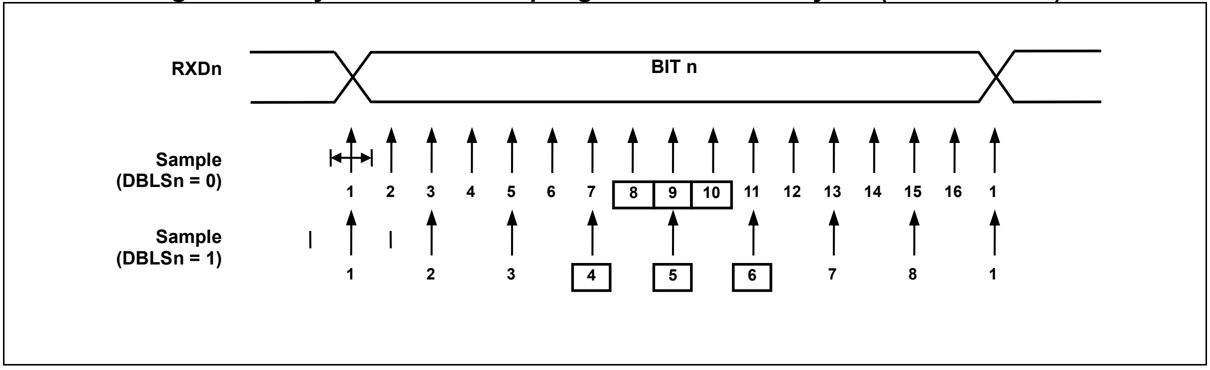
If more than 2 samples have logical low level, it is considered that a valid start bit is detected and the internally generated clock is synchronized to the incoming data frame. Then the data recovery can begin. The synchronization process is repeated for each start bit.

As described above, when the receiver clock is synchronized to the start bit, the data recovery can begin. Data recovery process is almost the same as clock recovery process.

The data recovery logic samples each incoming bit 16 times for normal mode and 8 times for double speed mode, and uses sample 8, 9 and 10 to decide data value. If more than 2 samples have low levels, the received bit is considered as a logic ‘0’ and if more than 2 samples have high levels, the received bit is considered as a logic ‘1’.

The data recovery process is then repeated until a complete frame is received, including the first stop bit. The decided bit value is stored in the receive shift register in order. Note that the Receiver only uses the first stop bit of a frame. Internally, after receiving the first stop bit, the receiver is in idle state and waits to find the start bit. (n = 10 and 11)

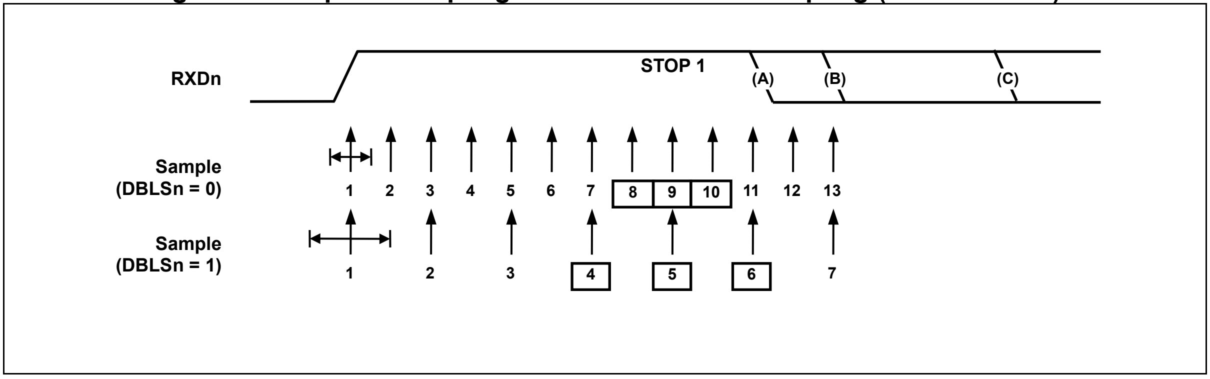
Figure 80. Asynchronous Sampling of Data and Parity Bit (n = 10 and 11)



The process for detecting stop bit is the same as clock and data recovery process. That is, if 2 or more samples of 3 center values have high level, correct stop bit is detected, or else a frame error (FEn) flag is set.

After deciding whether the first stop bit is valid or not, the receiver goes to idle state and monitors the RXDn line to check whether a valid high to low transition is detected (start bit detection). (n = 10 and 11)

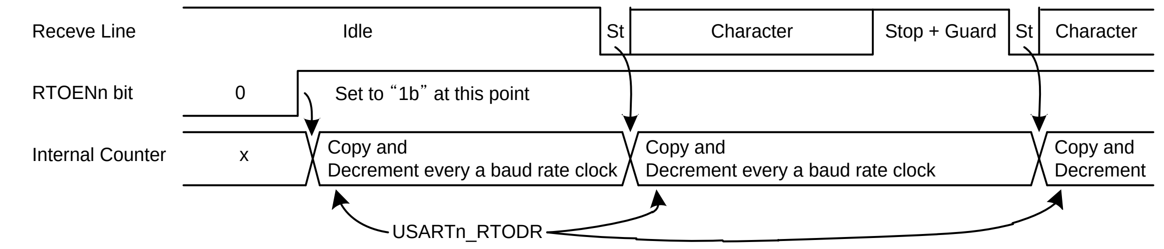
Figure 81. Stop Bit Sampling and Next Start Bit Sampling (n = 10 and 11)



16.4.7.6 Receive time out function

The receive time out function is used for checking a frame finish. This function is to count time with baud rate unit between the last start bit and a new start bit, and between setting the RTOENn bit of USARTn_CR3 register and a new start bit. The USARTn_RTODR register should have duration time value before using the receive time out function. (n = 10 and 11)

Figure 82. Receive Time Out Function (n = 10 and 11)


Receve Line
Idle
St
Character
Stop + Guard
St
Character
RTOENn bit
0
Set to "1b" at this point
Internal Counter
x
Copy and Decrement every a baud rate clock
Copy and Decrement every a baud rate clock
Copy and Decrement
USARTn_RTODR

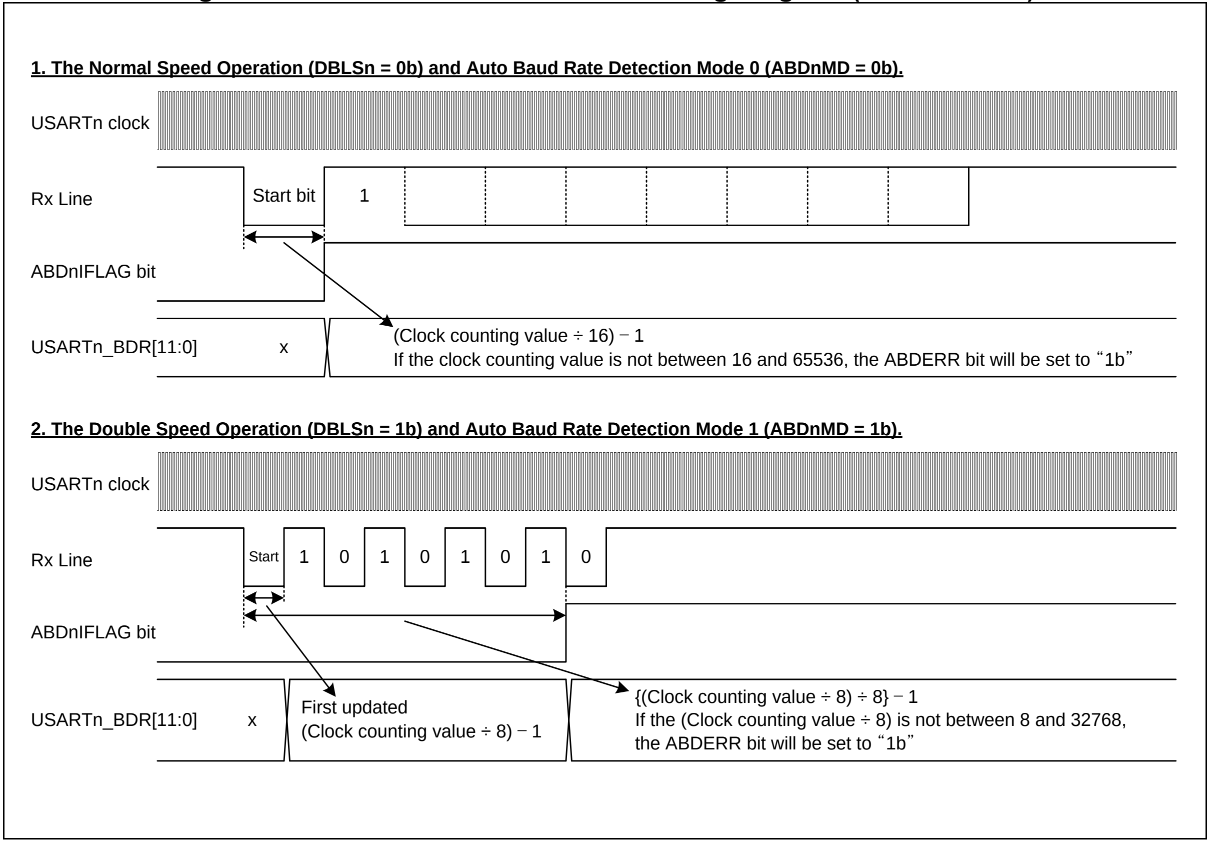
NOTES:

1. The value of USARTn_RTODR register is transferred to an internal counter when the RTOENn bit is set to “1b” or when a start bit is received to.
2. If the RTOENn bit is set to “1b”, the internal counter counts down until underflow.
3. The RTOnIFLAG bit of USARTn_CR3 register is set to “1b” on the internal counter underflow and the receive time out function is finished. So, to re-use the receive time out function, the RTOENn bit must be cleared to “0b” and then set to “1b” again.

16.4.7.7 UART auto baud rate detection

The auto baud rate detection is enabled by setting “1b” to the ABDENn bit of the USARTn_CR3 register. The function is useful when using clock source with relatively low accuracy. There are two auto baud rate detection modes, “Start bit to measure” and “0x55 character to measure”. (n = 10 and 11)

Figure 83. Auto Baud Rate Detection Timing Diagram (n = 10 and 11)


1. The Normal Speed Operation (DBLSn = 0b) and Auto Baud Rate Detection Mode 0 (ABDnMD = 0b).
USARTn clock
Rx Line
Start bit
1
ABDnIFLAG bit
USARTn_BDR[11:0]
x
(Clock counting value ÷ 16) − 1
If the clock counting value is not between 16 and 65536, the ABDERR bit will be set to “1b”
2. The Double Speed Operation (DBLSn = 1b) and Auto Baud Rate Detection Mode 1 (ABDnMD = 1b).
USARTn clock
Rx Line
Start
1
0
1
0
1
0
1
0
ABDnIFLAG bit
USARTn_BDR[11:0]
x
First updated
(Clock counting value ÷ 8) − 1
{(Clock counting value ÷ 8) ÷ 8} − 1
If the (Clock counting value ÷ 8) is not between 8 and 32768, the ABDERR bit will be set to “1b”

16.4.8 SPI mode

The USART can be set to operate in industrial standard SPI compliant mode. The SPI mode has the following features.

- Full Duplex, Three-wire synchronous data transfer.
- Mater and Slave Operation.
- Supports all four SPI modes of operation (mode 0, 1, 2, and 3).
- Selectable LSB first or MSB first data transfer.
- Double buffered transmit and receive.
- Programmable transmit bit rate.
- Up to 16MHz data transfer for SPI

When the SPI mode is enabled by configuring USTnMS[1:0] as ”11”, the slave select (SSn) pin becomes active LOW input in Slave mode operation if USTnSSEN bit is set to ‘1’. The SSn function is not automatically controlled in master mode operation even if USTnSSEN bit is set to ‘1’.

Note that during SPI mode of operation, the pin RXDn is renamed as MISOn and TXDn is renamed as MOSIn for compatibility to other SPI devices. (n = 10 and 11)

16.4.9 SPI clock formats and timing

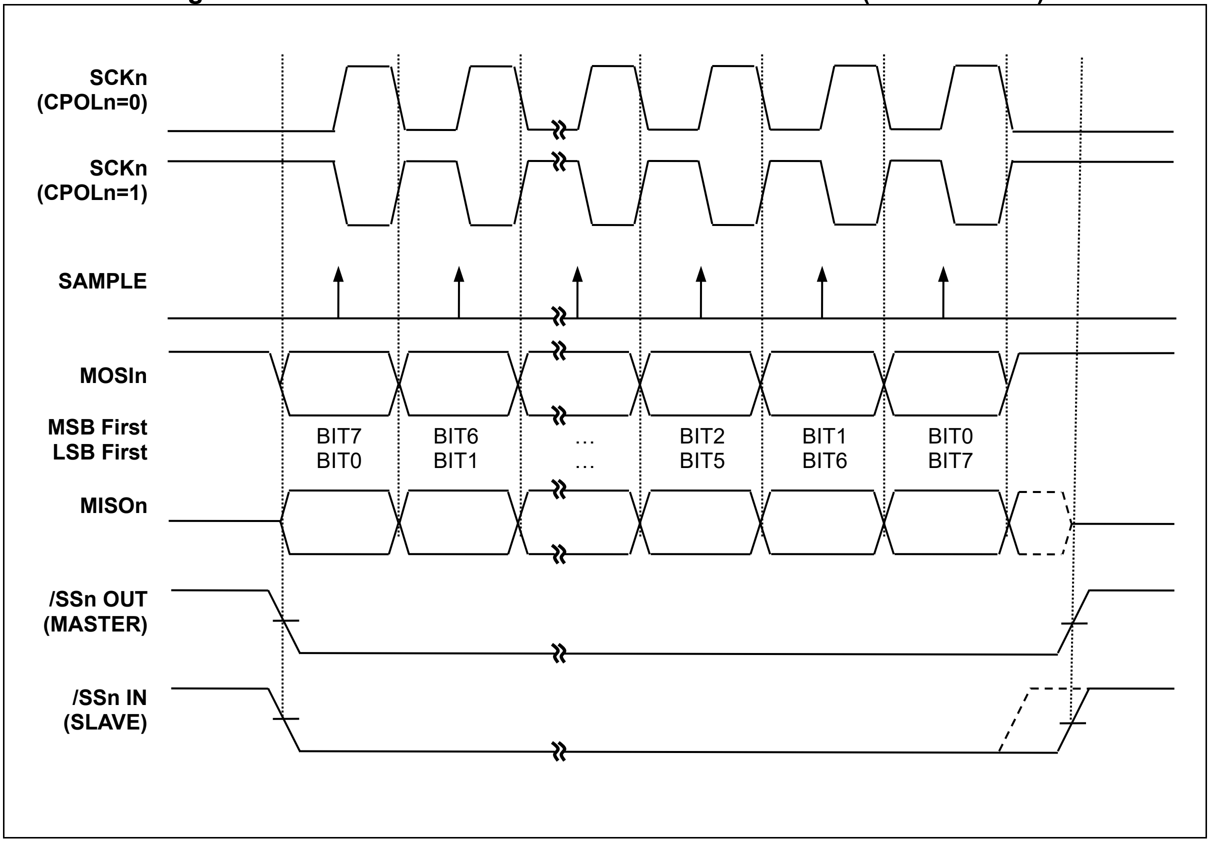
To accommodate a wide variety of synchronous serial peripherals from different manufacturers, the USART has a clock polarity bit (CPOLn) and a clock phase control bit (CPHAn) to select one of four clock formats for data transfers. CPOLn selectively inserts an inverter in series with the clock. CPHAn chooses between two different clock phase relationships between the clock and data. Note that CPHAn and CPOLn bits in USTnCR0 register have different meanings according to the USTnMS[1:0] bits, which decide the operating mode of USART.

Table 71 shows four combinations of CPOLn and CPHAn for SPI mode 0, 1, 2, and 3. (n = 10 and 11)

Table 71. CPOL Functionality (n = 10 and 11)

SPIn mode	CPOLn	CPHAn	Leading edge	Trailing edge
0	0	0	Sample (Rising)	Setup (Falling)
1	0	1	Setup (Rising)	Sample (Falling)
2	1	0	Sample (Falling)	Setup (Rising)
3	1	1	Setup (Falling)	Sample (Rising)

Figure 84. USART SPIn Clock Formats when CPHAn=0 (n = 10 and 11)

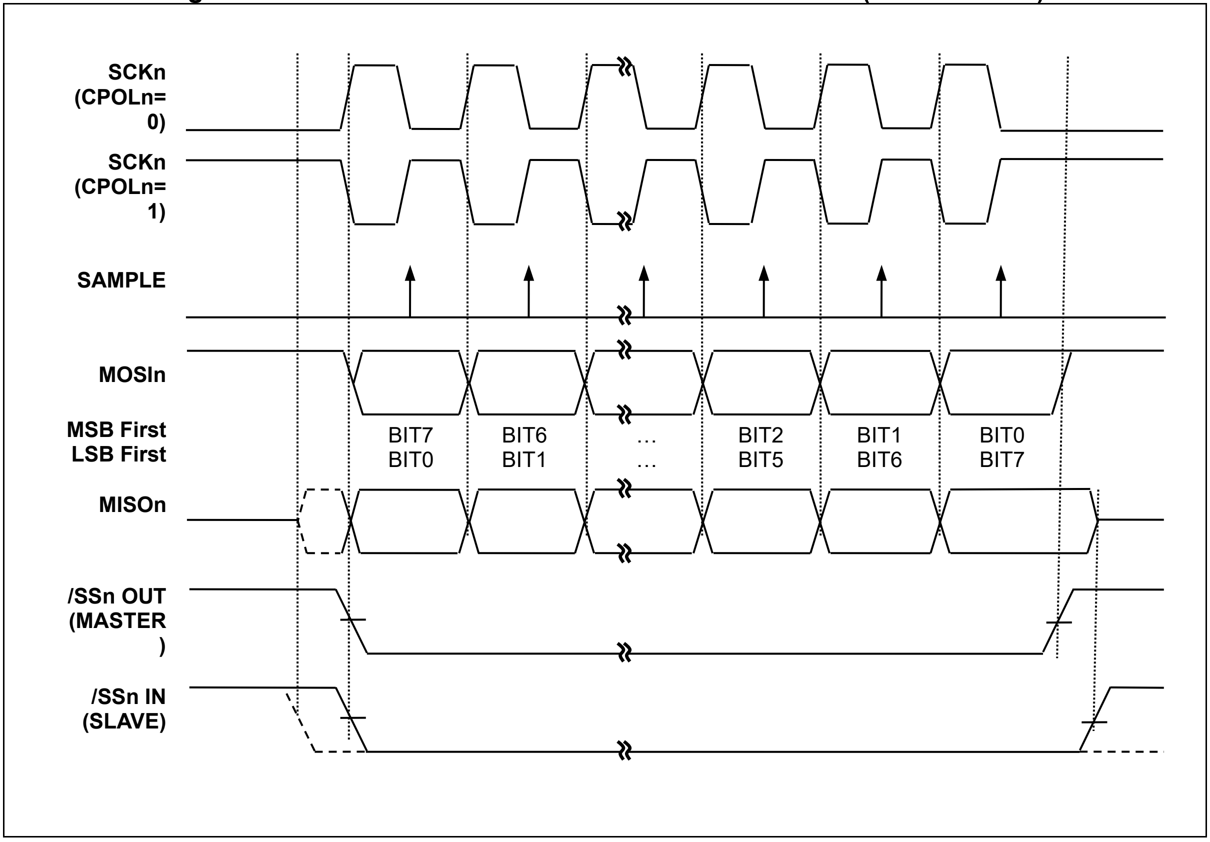


When CPHAn=0, the slave begins to drive its MISOn output with the first data bit value when SSn goes to active low.

The first SCKn edge causes both the master and the slave to sample the data bit value on their MISOn and MOSIn inputs, respectively.

At the second SCKn edge, the USART shifts the second data bit value out to the MOSIn and MISOn outputs of the master and slave, respectively. (n = 10 and 11)

Figure 85. USART SPIn Clock Formats when CPHAn=1 (n = 10 and 11)



When CPHAn=1, the slave begins to drive its MISOn output when SSn goes active low, but the data is not defined until the first SCKn edge.

The first SCKn edge shifts the first bit of data from the shifter onto the MOSIn output of the master and the MISOn output of the slave.

The next SCKn edge causes both the master and slave to sample the data bit value on their MISOn and MOSIn inputs, respectively.

At the third SCKn edge, the USART shifts the second data bit value out to the MOSIn and MISOn output of the master and slave respectively.

Because the SPIn logic reuses USART resources, SPIn mode of operation is similar to that of synchronous or asynchronous operation.

A SPIn transfer is initiated by checking for the USART Data Register Empty flag (DREn=1) and then writing a byte of data to the USARTn_TDR Register. In master mode of operation, even when transmission is not enabled (TXEn=0), writing data to the USARTn_TDR register is necessary because the clock SCKn is generated from the transmitter block.

16.4.10 Local interconnection network (LIN) mode

The LIN mode is selected by writing “10b” to the USTnMS[1:0] bits of the USARTn_CR1 register. The LIN transmission is fixed as start bit, 8-bits data length, 1 stop bit, and no parity. So, it should be set as follows.

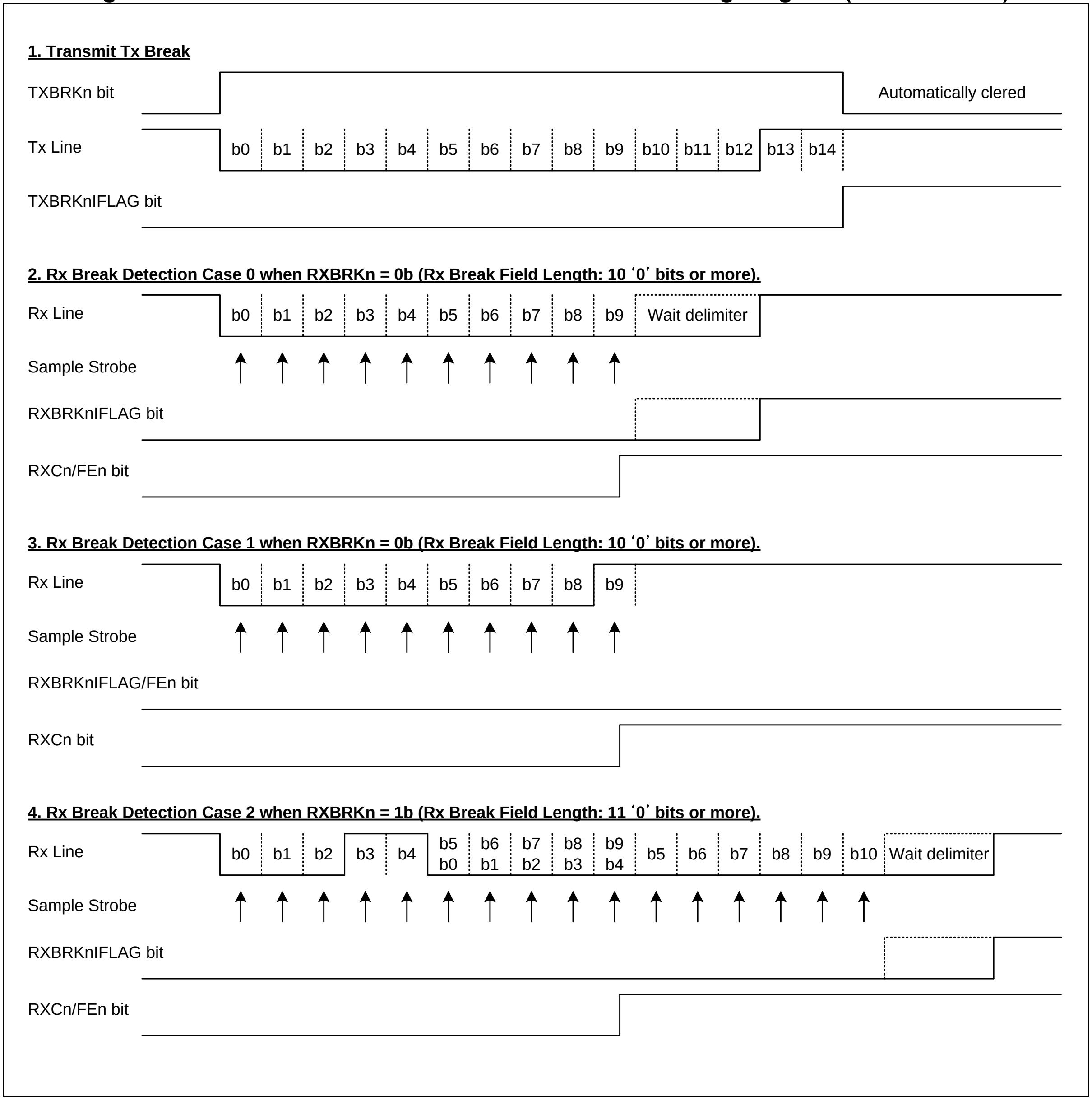
- USTnP[1:0] bits of USARTn_CR1: cleared to “00b” for no parity.
- USTnS[2:0] bits of USARTn_CR1: set to “011b” for 8-bit data length.
- USTnSB and LOOPSn bits USARTn_CR2: cleared to “0b” for 1 stop bit and normal operation.

During LIN mode is enabled, the break field detection circuit is activated and it is independent from the UART receiver. A break field can be detected whenever it occurs during idle state or during a frame.

When the auto baud rate detection is enabled by setting “1b” to the ABDENn bit of USARTn_CR3 register, the Rx break field may not be detected. So, the auto baud rate detection function should be enabled after the Rx break field detection if needed.

Figure 86 shows the timing diagram for LIN break field.

Figure 86. LIN Break Field Detection and Transmit Timing Diagram (n = 10 and 11)


1. Transmit Tx Break
TXBRKn bit
Automatically clered
Tx Line
b0 b1 b2 b3 b4 b5 b6 b7 b8 b9 b10 b11 b12 b13 b14
TXBRKnIFLAG bit
2. Rx Break Detection Case 0 when RXBRKn = 0b (Rx Break Field Length: 10 '0' bits or more).
Rx Line
b0 b1 b2 b3 b4 b5 b6 b7 b8 b9 Wait delimiter
Sample Strobe
RXBRKnIFLAG bit
RXCn/FEn bit
3. Rx Break Detection Case 1 when RXBRKn = 0b (Rx Break Field Length: 10 '0' bits or more).
Rx Line
b0 b1 b2 b3 b4 b5 b6 b7 b8 b9
Sample Strobe
RXBRKnIFLAG/FEn bit
RXCn bit
4. Rx Break Detection Case 2 when RXBRKn = 1b (Rx Break Field Length: 11 '0' bits or more).
Rx Line
b0 b1 b2 b3 b4 b5 b0 b6 b1 b7 b2 b8 b3 b9 b4 b5 b6 b7 b8 b9 b10 Wait delimiter
Sample Strobe
RXBRKnIFLAG bit
RXCn/FEn bit

17 UART 0

The A31L21x series has a built-in 1-channel UART module (Universal Asynchronous Receiver/Transmitter).

Users can read the UART operation status including the error status from the status register.

A baud rate generator, which generates proper baud rate, exists for each UART channel. This baud rate generator divides down the PCLK to the frequency ranging from 1 to 65536. Then, the baud rate is generated using a 1:16 clock and an 8-bit precision clock tuning function.

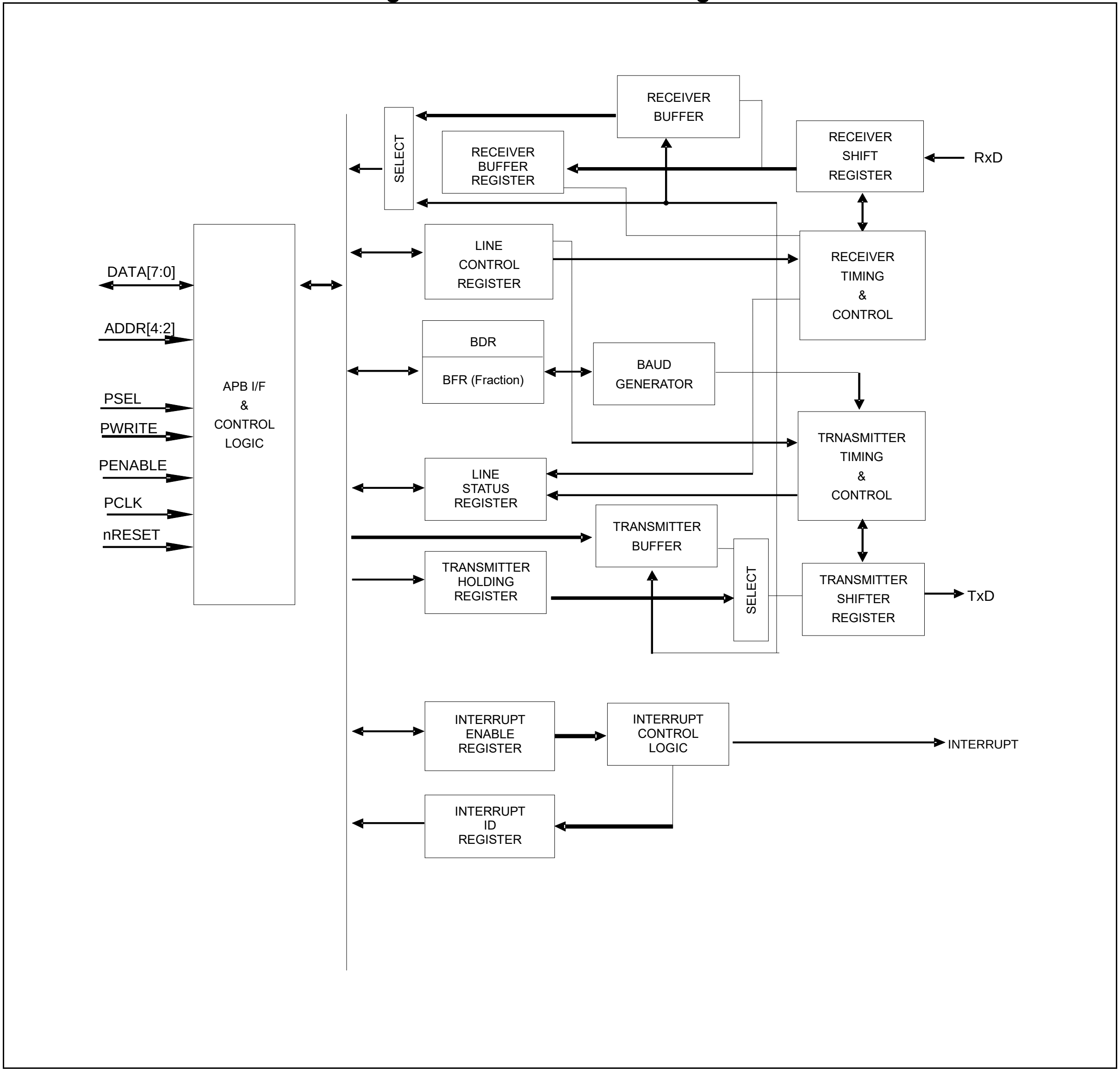
The UART 0 of the A31L21x series features the followings:

- Compatible with 16450 UART
- Configurable standard asynchronous control bit (Start, Stop, and Parity)
- Programmable 16-bit fractional baud generator
- Programmable serial communication
- 5-, 6-, 7- or 8- bit data transfer
- Even, odd, or no-parity bit insertion and detection
- 1-, 1.5- or 2-Stop bit-insertion and detection
- 16-bit baud rate generation with 8-bit fraction control
- Hardware inter-frame delay function
- Stop bit error detection
- Detail status register

17.1 UART 0 block diagram

Figure 87 shows a block diagram of the UART block.

Figure 87. UART 0 Block Diagram



17.2 Pin description for UART 0

Table 72. Pins and External Signals for UART 0 (n = 0)

Pin name	Type	Description
TXDn	O	UART Channel n transmit output
RXDn	I	UART Channel n receive input

17.3 Registers

Base address and register map of the UART are shown in Table 73 and Table 74.

Table 73. Base Address of UART

Name	Base address	Size	Description
UART0	0x4000_4000	256	UART0 Block

Table 74. UART n Register Map (n = 0)

Name	Offset	Type	Description	Reset value
UARTn_RBR	0x00	RO	UARTn Receive Data Buffer Register	0x00000000
UARTn_THR	0x00	WO	UARTn Transmit Data Hold Register	0x00000000
UARTn_IER	0x04	RW	UARTn Interrupt Enable Register	0x00000000
UARTn_IIR	0x08	RO	UARTn Interrupt ID Register	0x00000001
UARTn_LCR	0x0C	RW	UARTn Line Control Register	0x00000000
UARTn_DCR	0x10	RW	UARTn Data Control Register	0x00000000
UARTn_LSR	0x14	RO	UARTn Line Status Register	0x00000060
UARTn_BDR	0x20	RW	UARTn Baud Rate Divisor Latch Register	0x00000000
UARTn_BFR	0x24	RW	UARTn Baud Rate Fractional Counter Value	0x00000000
UARTn_IDTR	0x30	RW	UARTn Inter-frame Delay Time Register	0x000000C0

17.3.1 UARTn_RBR: UARTn receive data buffer register

Received data will be read from UARTn_RBR register. The maximum length of data is 8 bits. The last data received will stay in this register until a new byte is received.

UARTn_RBR register is 32-bit size and accessible in 32/16/8-bit. (n = 0)

UART0_RBR=0x4000_4000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																								RBR							
0x000000																								0x00							
-																								RO							

7 0	RBR	UARTn Receive Data Buffer.

17.3.2 UARTn_THR: UARTn transmit data hold register

UARTn_THR register is 32-bit size and accessible in 32/16/8-bit. (n = 0)

UART0_THR=0x4000_4000

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="24">Reserved</td><td colspan="8">THR</td></tr>
<tr><td colspan="24">0x000000</td><td colspan="8">0x00</td></tr>
<tr><td colspan="24">-</td><td colspan="8">WO</td></tr>
</table>

7 0	THR	UARTn Transmit Data Hold.

17.3.3 UARTn_IER: UARTn interrupt enable register

UARTn_IER register is 32-bit size and accessible in 32/16/8-bit. (n = 0)

UART0_IER=0x4000_4004

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="28">Reserved</td><td>TXEIE</td><td>RLSIE</td><td>THREIE</td><td>DRIE</td></tr>
<tr><td colspan="28">0x0000000</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td colspan="28">-</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td></tr>
</table>

3	TXEIE	Transmit Register Empty Interrupt Enable.
		0Disable transmit register empty interrupt.
		1Enable transmit register empty interrupt.
2	RLSIE	Receiver Line Status Interrupt Enable.
		0Disable receiver line status interrupt.
		1Enable receiver line status interrupt.
1	THREIE	Transmit Holding Register Empty Interrupt Enable.
		0Disable transmit hold register empty interrupt.
		1Enable transmit hold register empty interrupt.
0	DRIE	Data Receive Interrupt Enable.
		0Disable data receive interrupt.
		1Enable data receive interrupt.

17.3.4 UARTn_IIR: UARTn interrupt ID register

UARTn_IIR register is 32-bit size and accessible in 32/16/8-bit. (n = 0)

UART0_IIR=0x4000_4008

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5	4	3	2	1	0
Reserved	**TXE**	**Reserved**	**IID**		**IPEN**
0x000000	**0**	**0**	**00**		**1**
-	**RO**	-	**RO**		**RO**

4 TXE Transmit Complete Interrupt Source ID.

2 1 IID UARTn Interrupt ID.

NOTE: The UARTn supports 3-priority interrupt generation and the interrupt source ID register shows one interrupt source which has highest priority among pending interrupts.

The priority is defined as below.

— Receive line status interrupt.

— Receive data ready interrupt and Character timeout interrupt.

— Transmit hold register empty interrupt.

0 IPEN Interrupt Pending.

0 Interrupt is pending.

1 No interrupt is pending.

Table 75. Interrupt ID and Control of UARTn_IIR

Priority	**TXE**	**IID**		**IPEN**	**Interrupt sources**		
	Bit 4	**Bit 2**	**Bit 1**	**Bit 0**	**Interrupt**	**Interrupt condition**	**Interrupt clear**
-	0	0	0	1	None	-	-
1	0	1	1	0	Receiver Line Status	Overrun, Parity, Framing or Break Error	Read LSR register
2	0	1	0	0	Receiver Data Available	Receive data is available.	Read receive register or read IIR register
3	0	0	1	0	Transmitter Holding Register Empty	Transmit buffer empty	Write transmit hold register or read IIR register
4	1	X	X	X	Transmitter Register Empty	Transmit register empty	Write transmit hold register or read IIR register

NOTE: After check the above bits, Read data buffer to avoid losing interrupt source.

17.3.5 UARTn_LCR: UARTn line control register

UARTn_LCR register is 32-bit size and accessible in 32/16/8-bit. (n = 0)

UART0_LCR=0x4000_400C

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7	6	5	4	3	2	1	0
Reserved	**BREAK**	**STICKP**	**PARITY**	**PEN**	**STOPBIT**	**DLEN**	
0x000000	**0**	**0**	**0**	**0**	**0**	**00**	
-	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	

6	BREAK	Transfer Break Control. The TXDn pin will be driven at low state to notice the alert to the receiver. 0 Normal transfer mode. 1 Break transmit mode.
5	STICKP	Force Parity. This bit is effective when the PEN bit is set to ‘1’. 0 Disable parity stuck. 1 Enable parity stuck. A parity is always the value of the PARITY bit.
4	PARITY	Parity Mode and Parity Stuck Selection. 0 Odd parity mode. 1 Even parity mode.
3	PEN	Parity Bit Transfer Enable. 0 Disable parity transfer. 1 Enable parity transfer.
2	STOPBIT	Stop Bit Length Selection. 0 1 stop bit. 1 1.5 or 2 stop bit. 1.5 stop bit in case of 5-bit data and 2 stop bit in case of 6/7/8-bit data.
1 0	DLEN	Data Length Selection. 00 5-bit data length 01 6-bit data length 10 7-bit data length 11 8-bit data length

Parity bit will be generated according to bit 3,4,5 of UARTn_LCR register.

Table 76 shows the variation of parity bit generation.

Table 76. Interrupt ID and UARTn_LCR’s Control

STICKP	PARITY	PEN	Parity
X	X	0	No Parity
0	0	1	Odd Parity
0	1	1	Even Parity
1	0	1	Force parity as ‘1’
1	1	1	Force parity as ‘0’

17.3.6 UARTn_DCR: UARTn data control register

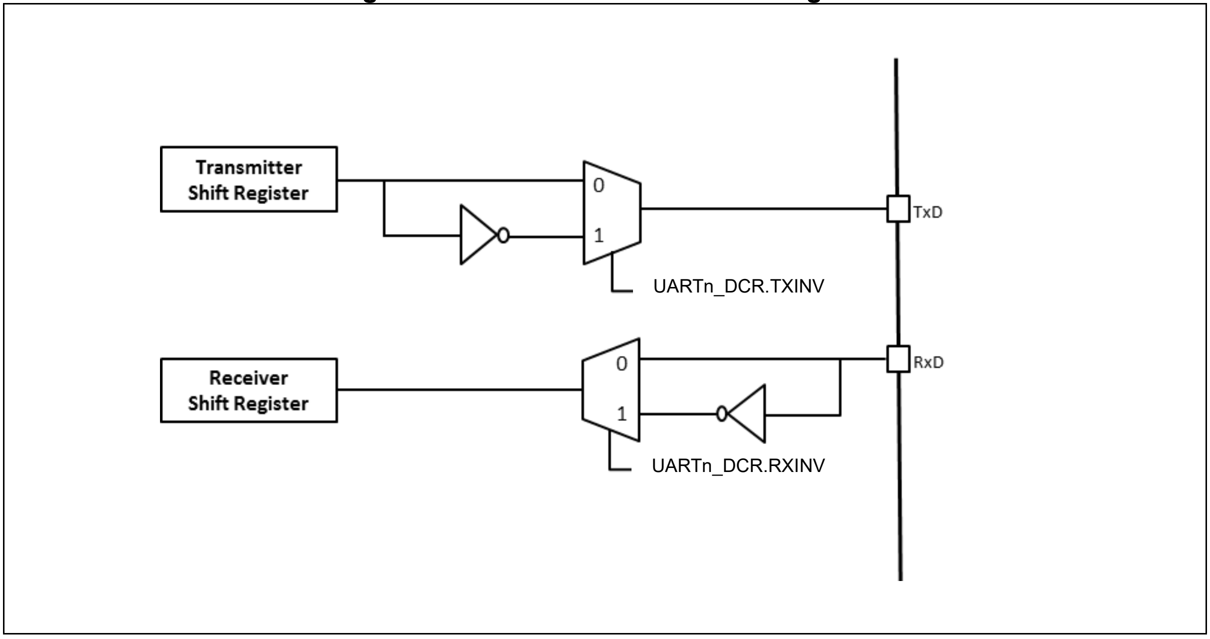
UARTn_DCR register is 32-bit size and accessible in 32/16/8-bit. (n = 0)

UART0_DCR=0x4000_4010

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5	4	3	2	1 0
Reserved	**LBON**	**RXINV**	**TXINV**	**Reserved**
0x000000	**0**	**0**	**0**	**00**
-	**RW**	**RW**	**RW**	**-**

4	LBON	Local Loopback Test Mode Enable.	
		0	Normal mode.
		1	Local loopback mode. TXDn connected to RXDn internally.
3	RXINV	Receive Data Inversion Selection.	
		0	Normal receive data input.
		1	Inverted receive data input.
2	TXINV	Transmit Data Inversion Selection.	
		0	Normal transmit output.
		1	Inverted transmit output.

Figure 88. Data Inversion Control Diagram



17.3.7 UARTn_LSR: UARTn line status register

UARTn_LSR register is 32-bit size and accessible in 32/16/8-bit. (n = 0)

UART0_LSR=0x4000_4014

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																									**TEMT**	**THRE**	**BI**	**FE**	**PE**	**OE**	**DR**
0x000000																									**1**	**1**	**0**	**0**	**0**	**0**	**0**
-																									**RO**	**RO**	**RO**	**RO**	**RO**	**RO**	**RO**

6	TEMT	Transmit Empty.	
		0	Transmit register has data or is transmitting.
		1	Transmit register is empty.
5	THRE	Transmit Holding Empty.	
		0	Transmit hold register is not empty.
		1	Transmit hold register is empty
		NOTE: This bit will be set to ‘1’ when it starts transmission.	
4	BI	Break Condition Indication.	
		0	Normal status.
		1	Break condition is detected.
3	FE	Frame Error Indicator.	
		0	No frame error.
		1	Frame error takes place. The receive character did not have a valid stop.
2	PE	Parity Error Indicator.	
		0	No parity error.
		1	Parity error takes place. The receive character does not have correct parity information.
1	OE	Overrun Error Indicator.	
		0	No overrun error.
		1	Overrun error takes place. Additional data arrived while RHR is full.
0	DR	Data Receive Indicator.	
		0	No data in receive hold register.
		1	Data has been received and is saved in the receive hold register.

This register provides the status of data transfers between transmitter and receiver. A user can check the line status from this register. Bit 1,2,3,4 will raise the line status interrupt when RLSIE bit in UARTn_IER register is set. Other bits can generate interrupts when their interrupt enable bits in UARTn_IER register are set.

17.3.8 UARTn_BDR: UARTn baud rate divisor latch register

UARTn_BDR register is 32-bit size and accessible in 32/16/8-bit. (n = 0)

UART0_BDR=0x4000_4020

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Reserved	**BDR**
0x0000	**0x0000**
-	**RW**

15 0	BDR	Baud Rate Divider Latch Value Baud rate = PCLK/(16 x (BDR[15:0] + 1)). The range is 0x0000 to 0xFFFF.

To establish communication with the UART channel, baud rate should be set properly. The programmable baud rate generator provides divider number from 0 to 65535. Expected baud rate should be written to the 16-bit divider register (UARTn_BDR). UART$_{clock}$ is PCLK.

Baud rate calculation formula is as follows:

$$\mathrm{BDR} = \frac{UART_{clock}}{16 \times BaudRate} - 1$$

In case of 32MHz UART$_{clock}$ speed, the divider value and error rate is shown in table

Table 77. Example of Baud Rate Calculation (without BFR)

UART$_{clock}$= 32MHz		
Baud rate	**Divider**	**Error (%)**
1200	1665	0.04%
2400	832	0.04%
4800	415	0.16%
9600	207	0.16%
19200	103	0.16%
38400	51	0.16%
57600	33	2.12%
115200	16	2.12%

17.3.9 UARTn_BFR: UARTn baud rate fraction counter register

UARTn_BFR register is 32-bit size and accessible in 32/16/8-bit. (n = 0)

UART0_BFR=0x4000_4024

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved			**BFR**
0x000000			**0x00**
-			**RW**

7 0	BFR	Fraction Counter value.	
		0	Disable fraction counter.
		N	Fraction compensation mode under operation. Fraction counter is incremented by FCNT. FCNT = Float * 256
		NOTE:	8-bit fractional counter will count up by FCNT value every (baud rate)/16 period and whenever fractional counter overflow is happen, the divisor value will increment by 1. So this period will be compensated. Then next period, the divisor value will return to original set value.

Table 78. Example of Baud Rate Calculation

$\text{UART}_{\text{clock}}$ = 32MHz			
Baud rate	**Divider**	**FCNT**	**Error (%)**
1200	1665	170	0.00%
2400	832	85	0.00%
4800	415	170	0.00%
9600	207	85	0.00%
19200	103	42	0.00%
38400	51	21	0.00%
57600	33	184	0.01%
115200	16	92	0.01%

FCNT value can be calculated using the equation below:

$$\text{FCNT} = \text{Float} * 256$$

For example, when the target baud rate is 4800 bps and $\text{UART}_{\text{clock}}$ is 32MHz, the BDR value is 415.6666. The integer 415 is the BDR value and floating number 0.6666 leads to an FNCT value as follows:

FCNT = 0.6666 * 256 = 170.6496, and thus the FCNT value is 170.

8-bit fractional counter will count up by FCNT value every (baud rate)/16 periods and whenever fractional counter overflow takes place, the divisor value will increment by 1 and compensate this period. Then, the divisor value will return to its original value.

17.3.10 UARTn_IDTR: UARTn inter-frame delay time register

UARTn_IDTR register is 32-bit size and accessible in 32/16/8-bit. (n = 0)

UART0_IDTR=0x4000_4030

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8	7	6	5 4 3	2 1 0
Reserved	**SMS**	**DMS**	**Reserved**	**WAITVAL**
0x000000	**1**	**1**	**000**	**000**
-	**RW**	**RW**	**-**	**RW**

7	SMS	Start Bit Multi Sampling Enable.	
		0	Multi sampling is disabled for start bit, Single sampling will be done at 8/16 baud rate for the start bit.
		1	Multi sampling is enabled for start bit. Sampling is done 3 times at 7/16, 8/16, and 9/16 baud rate. Dominant value among 3 samples will be selected for the start bit.
6	DMS	Data Bit Multi sampling enable.	
		0	Multi sampling is disabled for data bit, Single sampling will be done at 8/16 baud rate for the data bit.
		1	Multi sampling is enabled for data bit. Sampling is done 3 times at 7/16, 8/16, and 9/16 baud rate. Dominant value among 3 samples will be selected for the data bit.
2 0	WAITVAL	Wait Time Value. Dummy delay can be inserted between 2 Continuous Transmits. Wait Time = WAITVAL[2:0]/(Baud Rate)	

17.4 Functional description

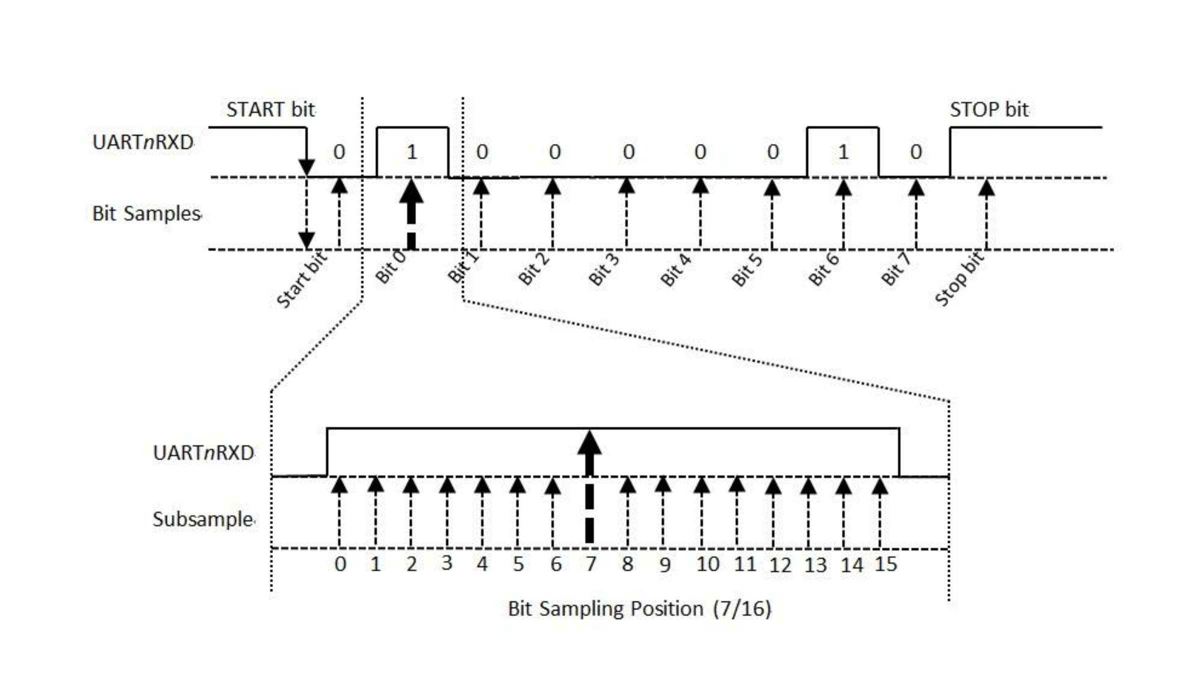
The UART module is compatible with 16450 UART. Additionally, fractional baud rate compensation logic is provided. It does not have an internal FIFO block.

17.4.1 Receiver sampling timing

The UART of the A31L21x series operates at the following timing as shown in Figure 89.

If falling edge is detected on the receive line, the UART considers it as a start bit. From then on, the UART oversamples 1-bit 16 times and detects the bit value at the 7th sample.

Figure 89. Sampling Timing of UART Receiver



It is recommended to enable debounce settings in the PCU block to enhance the immunity to external glitch noise.

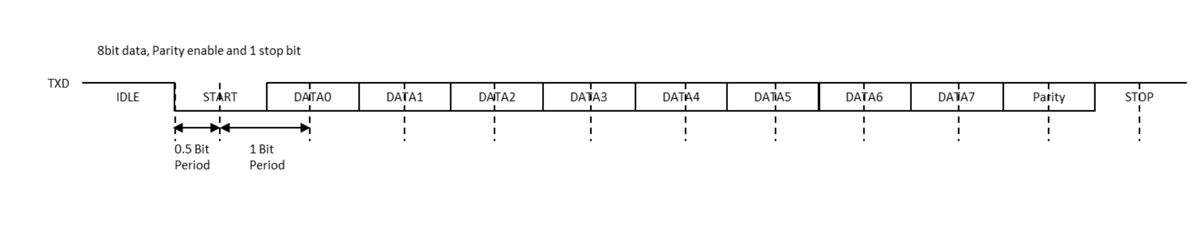
17.4.2 Transmitter

The transmitter has a data transmission function. The start bit, data bits, optional parity bit, and stop bit shift in series, the least significant bit shifting first.

The number of data bit is selected in DLEN[1:0] in the UARTn_LCR register. The parity bit is set according to the PARITY and PEN bits in the UARTn_LCR register. If the parity type is even, the parity bit depends on one-bit sum of all data bits. For odd parity, the parity bit is an inverted sum of all data bits. The number of stop bits is selected in the STOPBIT in the UARTn_LCR register.

The example of transmission data format is introduced in Figure 90.

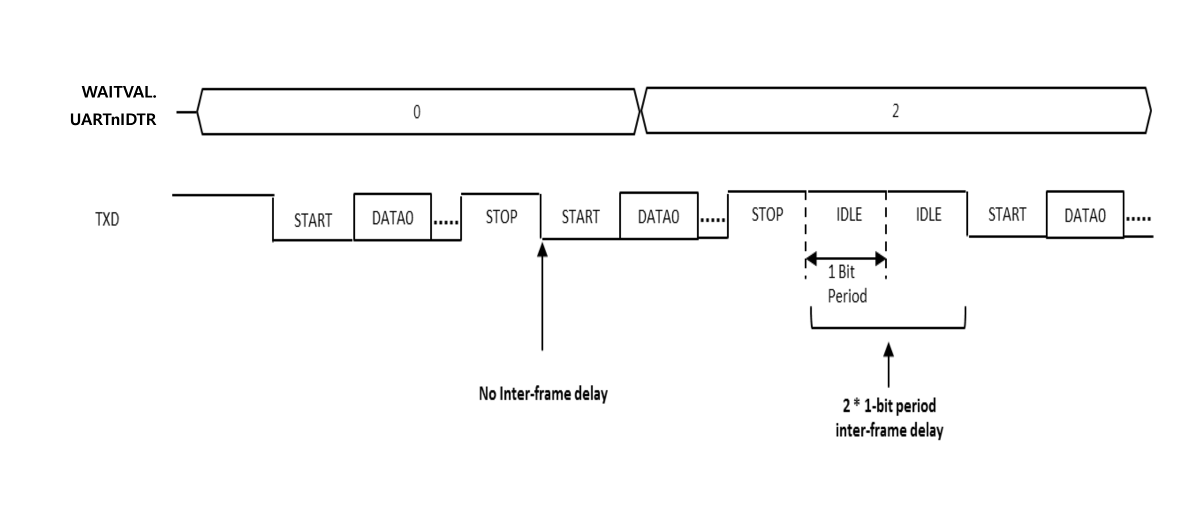
Figure 90. Transmission Data Format Example



17.4.3 Inter-frame delay transmission

The inter-frame delay function allows the transmitter to insert an idle state on the TXD line between 2 characters. The width of the idle state is defined in WAITVAL field of the UARTn_IDTR register. When this field is set as 0, no time-delay is generated. Otherwise, the transmitter holds a high level on TXD after each transmitted character during the number of bit periods defined in WAITVAL field.

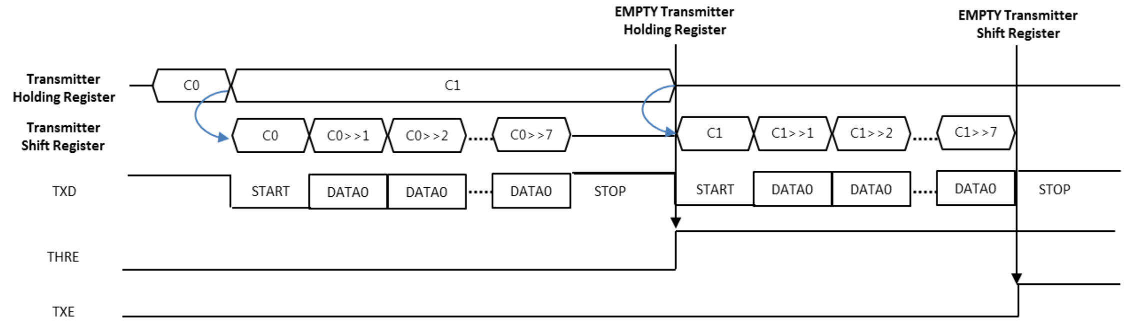
Figure 91. Inter-frame Delay Timing Diagram



17.4.4 Transmit interrupt

The transmission operation makes some kinds of interrupt flags. When transmitter hold register is empty, the THRE interrupt flag will be raised. When transmitter shifter register is empty, the TXE interrupt flag will be raised. User can select an interrupt timing that works the best for the application.

Figure 92. Transmit Interrupt Timing Diagram


EMPTY Transmitter Holding Register
EMPTY Transmitter Shift Register
Transmitter Holding Register
Transmitter Shift Register
TXD
THRE
TXE
C0
C1
C0>>1
C0>>2
C0>>7
C1>>1
C1>>2
C1>>7
START
DATA0
STOP

18 LPUART 0/1

The A31L21x series has a built-in 2-channel low power UART module (Universal Asynchronous Receiver/Transmitter).

This Low Power UART (LPUART) supports asynchronous serial communication up to 9600bps in DEEP SLEEP mode with 32.768kHz sub-oscillator. It also supports 1-wire half-duplex communication.

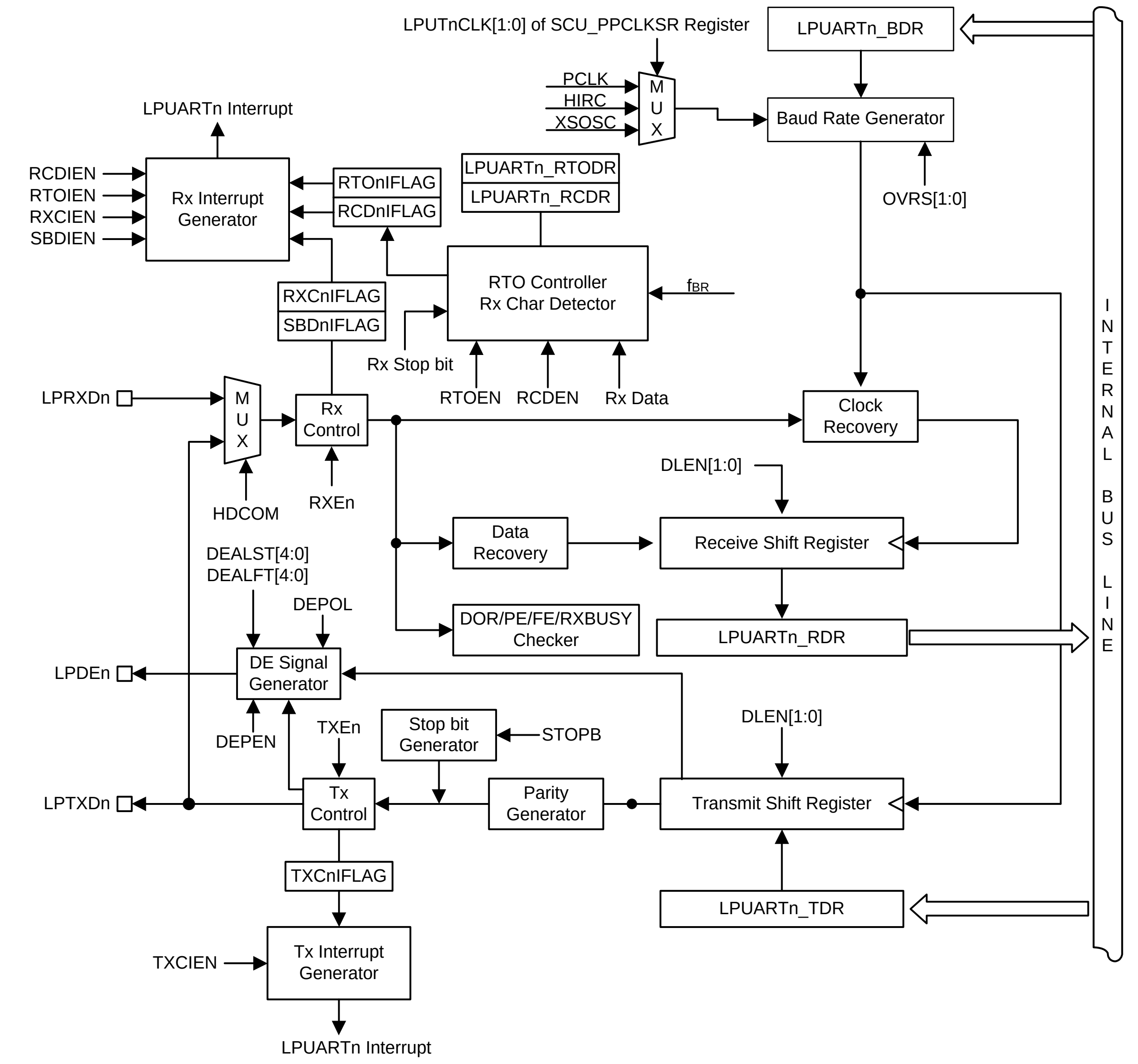
The LPUART 0/1 of the A31L21x series features the followings:

- Full-Duplex and Half-Duplex Operations
- Baud Rate Generator
- Serial Frames with 5,6,7, or 8 Data bits and 1 or 2 Stop bits supported
- Odd or Even Parity Generation, and Parity Check Supported by Hardware
- Receive Character Detection and Receive Time Out Function supported
- Baud Rate Compensation Function
- Up to 9600pbs with 32.768kHz sub-oscillator supported
- Data OverRun Detection
- Framing Error Detection
- Double Speed Asynchronous Communication Mode

18.1 LPUART block diagram

Figure 93 shows a block diagram of the LPUART block.

Figure 93. LPUART Block Diagram (n = 0 and 1)



NOTES:

1. If XSOSC (32.768kHz) is used for a clock of the LPUARTn, the maximum baud rate may be up to 9600bps.
2. A clock of PCLK should be faster than or equal to a clock of the LPUARTn.
3. Data to be transmitted should be written to the LPUARTn_TDR register after checking if the TXCIFLAG bit is set to “1b”.

18.2 Pin description for LPUART

Table 79. Pins and External Signals for LPUART (n = 0 and 1)

Pin name	Type	Description
LPTXDn	O	Low Power UART n transmit output
LPRXDn	I	Low Power UART n receive input
LPDEn	O	Low Power UART n DE signal output

18.3 Registers

Base address and register map of the LPUART are shown in Table 80 and Table 81.

Table 80. Base Address of LPUART

Name	Base address
LPUART0	0x4000_5C00
LPUART1	0x4000_5C80

Table 81. LPUART Register Map (n = 0 and 1)

Name	Offset	Type	Description	Reset Value
LPUARTn_CR1	0x00	RW	LPUARTn Control Register 1	0x00000000
LPUARTn_CR2	0x04	RW	LPUARTn Control Register 2	0x00000000
LPUARTn_IER	0x10	RW	LPUARTn Interrupt Enable Register	0x00000000
LPUARTn_IFSR	0x14	RW	LPUARTn Interrupt Flag and Status Register	0x00000004
LPUARTn_RDR	0x18	RO	LPUARTn Receive Data Register	0x00000000
LPUARTn_TDR	0x1C	RW	LPUARTn Transmit Data Register	0x00000000
LPUARTn_BDR	0x20	RW	LPUARTn Baud Rate Data Register	0x0000FFFF
LPUARTn_BCMP	0x24	RW	LPUARTn Baud Rate Compensation Register	0x00000000
LPUARTn_RTODR	0x28	RW	LPUARTn Receive Time Out Data Register	0x0000FFFF
LPUARTn_RCDR	0x2C	RW	LPUARTn Receive Character Detection Data Register	0x00000000
LPUARTn_DLYDR	0x30	RW	LPUARTn Tx Delay Time Data Register	0x00000000

18.3.1 LPUARTn_CR1: LPUARTn control register 1

Low power UARTn module should be configured properly before running.

LPUARTn_CR1 register is 32-bit size and accessible in 32/16/8-bit. (n = 0 and 1)

LPUART0_CR1=0x4000_5C00, LPUART1_CR1=0x4000_5C80

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	**Reserved**	**PEN**	**STKPEN**	**PSEL**	**Reserved**	**DLEN**		**Reserved**	**STOPB**	**OVRS**		**HDCOM**	**TXE**	**RXE**	**WAKEN**	**LPUEN**
0x0000	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
–	**-**	**RW**	**RW**	**RW**	**-**	**RW**	**RW**	**-**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**

14	PEN	Parity Enable.	
		0	Disable parity bit generation and detection.
		1	Enable parity bit generation and detection.
13	STKPEN	Stick Parity Enable.	
		0	Disable stick parity.
		1	Enable stick parity.
		NOTE:	On PEN = 1 and STKPEN = 1, The parity bit is 0 if PSEL = 0 and 1 if PSEL = 1.
12	PSEL	Parity Selection.	
		0	Odd parity (Odd number of logic ‘1’).
		1	Even parity (Even number of logic ‘1’).
10 9	DLEN	Data Length Selection.	
		00	5 bit (Start, D0, D1, D2, D3, D4, Parity or not, Stop1, Stop2 or not).
		01	6 bit (Start, D0, D1, D2, D3, D4, D5, Parity or not, Stop1, Stop2 or not)
		10	7 bit (Start, D0, D1, D2, D3, D4, D5, D6 Parity or not, Stop1, Stop2 or not)
		11	8 bit (Start, D0, D1, D2, D3, D4, D5, D6, D7, Parity or not, Stop1, Stop2 or not).
7	STOPB	Stop bit.	
		0	1 Stop bit.
		1	2 Stop bit.
6 5	OVRS	Oversampling Selection.	
		00	16 oversampling.
		01	8 oversampling.
		10	No oversampling (Only 1 sampling).
		11	reserved (No oversampling).
4	HDCOM	1-wire Half-Duplex Communication.	
		0	Normal operation.
		1	1-wire half-duplex communication (The TXD and RXD lines are internally connected, the RXD pin is not used, and the TXD pin is always an input when no transmitted. So, the TXD pin must be configured to open-drain with an external pull-up resistor)
3	TXE	Enable the Transmitter unit.	
		0	Transmitter is disabled.
		1	Transmitter is enabled.
2	RXE	Enable the Receiver unit.	
		0	Receiver is disabled.
		1	Receiver is enabled.
1	WAKEN	Wake-up Function bit in DEEP SLEEP Mode. The LPUARTn clock to wake-up from DEEP SLEEP mode must be selected as XSOSC by the SCU_PPCLKSR register. This bit should be set just before entering DEEP SLEEP mode and cleared on exit.	
		0	Disable wake-up function in DEEP SLEEP mode.
		1	Enable wake-up function in DEEP SLEEP mode.
		If XSOSC is for clock of LPUARTn, the XSOSC shouldn’t be off in DEEP SLEEP mode.	
0	LPUEN	Low Power UARTn Enable bit. This bit can be cleared to “0b” during the corresponding TXE and RXE bits are all “0b”.	
		0	Disable LPUARTn block.
		1	Enable LPUARTn block.

NOTE:	If this bit is cleared, the LPUARTn current operations are discarded, the configuration is kept, and all the status flags are set to reset values.

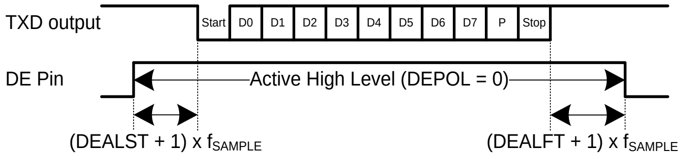
18.3.2 LPUARTn_CR2: LPUARTn control register 2

Low power UARTn module should be configured properly before running.

LPUARTn_CR2 register is 32-bit size and accessible in 32/16/8-bit. (n = 0 and 1)

LPUART0_CR2=0x4000_5C04, LPUART1_CR2=0x4000_5C84

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								Reserved			DEALST					Reserved			DEALFT					DEPOL	DEPEN	Reserved	RCDEN	RTOEN	Reserved		
0x00								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
–								-	-	-	RW	RW	RW	RW	RW	-	-	-	RW	RW	RW	RW	RW	RW	RW	-	RW	RW	-	-	-

20 16	DEALST	DE Pin Active Level Start Time. The range is 0x00 to 0x1F. These bits define the time in low power UARTn clock from the active level of DE signal to the beginning of the start bit.	
12 8	DEALFT	DE Pin Active Level Finish Time. The range is 0x00 to 0x1F. These bits define the time in low power UARTn clock from the end of the stop bit to the de-active level of DE signal.	
7	DEPOL	DE Pin Polarity Selection.	
		0	Active high level. The DE pin is a high level during transmit a frame, else low level.
		1	Active low level. The DE pin is a low during transmit a frame, else high level.
		 TXD output Start D0 D1 D2 D3 D4 D5 D6 D7 P Stop DE Pin Active High Level (DEPOL = 0) (DEALST + 1) x f_{SAMPLE} (DEALFT + 1) x f_{SAMPLE} Where $f_{SAMPLE} = f_{LPUART}/(LPUARTn\_BDR[15:0] + 1)$ **NOTE**: A TXCIFLAG bit will be set to “1b” at stop bit and the transmit of next character may start after the end of active level.	
6	DEPEN	DE Pin Function Enable.	
		0	Disable DE pin function.
		1	Enable DE pin function.
4	RCDEN	Receive Character Detection Function Enable. This function is to compare the value of LPUARTn_RCDR register with the value just received.	
		0	Disable receive detection function.
		1	Enable receive detection function.
3	RTOEN	Receive Time Out Function Enable. This function is to count time with baud rate units from the leading edge of a start bit to a new start bit. The receive time out controller counts down from the value of LPUARTn_RTODR register every start bit and set this bit. The RTOIFLAG bit is set to “1b” at the counter underflow. The counter clock is a baud-rate bit unit.	
		0	Disable receive time out function.
		1	Enable receive time out function.

18.3.3 LPUARTn_IER: LPUARTn interrupt enable register

LPUARTn_IER register is 32-bit size and accessible in 32/16/8-bit. (n = 0 and 1)

LPUART0_IER=0x4000_5C10, LPUART1_IER=0x4000_5C90

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																								**RCDIEN**	**RTOIEN**	**Reserved**	**SBDIEN**	**Reserved**	**TXCIEN**	**Reserved**	**RXCIEN**
0x000000																								**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
–																								**RW**	**RW**	**-**	**RW**	**-**	**RW**	**-**	**RW**

7	RCDIEN	Receive Character Detection Interrupt Enable. On DEEP SLEEP mode, the receive character detection can wake-up system.	
		0	Disable receive character detection interrupt.
		1	Enable receive character detection interrupt.
6	RTOIEN	Receive Time Out Interrupt Enable.	
		0	Disable receive time out interrupt.
		1	Enable receive time out interrupt.
4	SBDIEN	Start Bit Detection Interrupt Enable bit in DEEP SLEEP mode. On DEEP SLEEP mode, the detection of start bit can wake-up system.	
		0	Disable start bit detection interrupt.
		1	Enable start bit detection interrupt.
2	TXCIEN	Transmit Complete Interrupt Enable.	
		0	Disable transmit complete interrupt.
		1	Enable transmit complete interrupt.
0	RXCIEN	Receive Data Register Not Empty Interrupt Enable bit. On DEEP SLEEP mode, it can wake-up system if there is a received character.	
		0	Disable receive data not empty interrupt.
		1	Enable receive data not empty interrupt.

18.3.4 LPUARTn_IFSR: LPUARTn interrupt flag and status register

LPUARTn_IFSR register is 32-bit size and accessible in 32/16/8-bit. (n = 0 and 1)

LPUART0_IFSR=0x4000_5C14, LPUART1_IFSR=0x4000_5C94

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																DOR	FE	PE	RXBUSY	Reserved				RCDnIFLAG	RTOnIFLAG	Reserved	SBDnIFLAG	Reserved	TXCnIFLAG	Reserved	RXCnIFLAG
0x0000																0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
–																RW	RW	RW	RO	-	-	-	-	RW	RW	-	RW	-	RW	-	RW

15	DOR	Data Overrun. This bit is set when the receive shift register is transferred to the LPUARTn_RDR register while the RXCnIFLAG=1. The data of the shift register are ignored. This bit must be cleared by S/W to receive new data.	
		0	No data overrun.
		1	Data overrun detected. This bit is cleared to '0' when write '1'.
14	FE	Frame Error bit. This bit is set when the received data have not a valid stop bit (That is, the stop bit following the last data bit is detected as "0b"). The bit will be cleared by H/W if new data are received.	
		0	No frame error.
		1	Frame error detected, This bit is cleared to '0' when write '1'.
13	PE	Parity Error bit. This bit is set when the received data has a parity error on parity enable. The bit will be cleared by H/W if new data are received.	
		0	No parity error.
		1	Parity error detected, This bit is cleared to '0' when write '1'.
12	RXBUSY	RXD Line Busy bit. This bit is set at a start bit and reset at the end of the reception.	
		0	Receive line (RXD) is not busy.
		1	Reception on going.
7	RCDnIFLAG	Receive Character detection Interrupt Flag. This bit is set to "1b" when the value of LPUARTn_RCDR register matches the value received in the non-error state of frame and parity. On match of them, the bit may be set even if data overrun occurs.	
		0	No request occurred.
		1	Request occurred. This bit is cleared to '0' when write '1'.
6	RTOnIFLAG	Receive Time Out Interrupt Flag. This bit is set to "1b" at the counter underflow of the receive time out controller.	
		0	No request occurred.
		1	Request occurred, This bit is cleared to '0' when write '1'.
4	SBDnIFLAG	Start Bit Detection Interrupt Flag. This bit is set to "1b" when a start bit is detected in DEEP SLEEP mode.	
		0	No request occurred.
		1	Request occurred. This bit is cleared to '0' when write '1'.
2	TXCnIFLAG	Transmit Complete Interrupt Flag. This flag is set to "1b" when the data in the transmit shift register has been shifted out.	
		0	No request occurred.
		1	The data in the transmit shift register are shifted out completely. This bit is cleared to '0' when write '1'.
0	RXCnIFLAG	Receive Data Register Not Empty Interrupt Flag. This bit is set to "1b" when the data in the receive shift register has been transferred to the LPUARTn_RDR register. The bit is cleared by a read to the LPUARTn_RDR register.	
		0	No request occurred.

	1	There is data in the receive data register. This bit is cleared to ‘0’ when write ‘1’.

18.3.5 LPUARTn_RDR: LPUARTn receive data register

LPUARTn_RDR register is 32-bit size and accessible in 32/16/8-bit. (n = 0 and 1)

LPUART0_RDR=0x4000_5C18, LPUART1_RDR=0x4000_5C98

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="24">Reserved</td><td colspan="8">RDATA</td></tr>
<tr><td colspan="24">0x000000</td><td colspan="8">0x00</td></tr>
<tr><td colspan="24">-</td><td colspan="8">RO</td></tr>
</table>

7 0	RDATA	Receive Data. A receive shift register is moved to this register after stop bit.

18.3.6 LPUARTn_TDR: LPUARTn transmit data register

LPUARTn_TDR register is 32-bit size and accessible in 32/16/8-bit. (n = 0 and 1)

LPUART_TDR0=0x4000_5C1C, LPUART_TDR1=0x4000_5C9C

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="24">Reserved</td><td colspan="8">TDATA</td></tr>
<tr><td colspan="24">0x000000</td><td colspan="8">0x00</td></tr>
<tr><td colspan="24">–</td><td colspan="8">RW</td></tr>
</table>

7 0	TDATA	Transmit Data bits. This register is moved to the transmit shift register after a previous character is completely shifted out.

18.3.7 LPUARTn_BDR: LPUARTn baud rate generation register

LPUARTn_BDR register is 32-bit size and accessible in 32/16/8-bit. (n = 0 and 1)

LPUART0_BDR=0x4000_5C20, LPUART1_BDR=0x4000_5CA0

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		BDATA	
0x0000		0xFFFF	
-		RW	

15
0 BDATA These bits are used to generate baud rate.

16 oversampling:

- Baud Rate = f_{LPUART}/{16 x (BDATA[15:0] +1)}
- BDATA[15:0] range: 0x0 to 0xFFFF

8 oversampling:

- Baud Rate = f_{LPUART}/{8 x (BDATA[15:0] +1)}
- BDATA[15:0] range: 0x0 to 0xFFFF

No oversampling: This can be used with XSOSC (32.768kHz).

- Baud Rate = f_{LPUART}/(BDATA[15:0] +1)
- BDATA[15:0] range: 0x2 to 0xFFFF
- If this register is 0x0002 on the no oversampling, the LPUARTn_BCMP[15] bit (BCMPS) shouldn't be set to "1b" for minus compensation.

18.3.8 LPUARTn_BCMP: LPUARTn baud rate compensation register

LPUARTn_BCMP register is 32-bit size and accessible in 32/16/8-bit. (n = 0 and 1)

LPUART0_BCMP=0x4000_5C24, LPUART1_BCMP=0x4000_5CA4

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		BCMPS	Reserved						BCMP8	BCMP7	BCMP6	BCMP5	BCMP4	BCMP3	BCMP2	BCMP1	BCMP0
0x0000		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
–		RW	-	-	-	-	-	-	RW	RW	RW	RW	RW	RW	RW	RW	RW

15	BCMPS	Baud Rate Compensation Sign.	
		0	Plus 1 clock for compensation.
		1	Minus 1 clock for compensation.
x	BCMPx	Baud Rate Compensation bits. x: 0 to 8.	
		0	No compensation.
		1	1 clock compensation with sign bit (BCMPS).

18.3.9 LPUARTn_RTODR: LPUARTn receive time out data register

LPUARTn_RTODR register is 32-bit size and accessible in 32/16/8-bit. (n = 0 and 1)

LPUART0_RTODR=0x4000_5C28, LPUART1_RTODR=0x4000_5CA8

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="8">Reserved</td><td colspan="24">RTOD</td></tr>
<tr><td colspan="8">0x00</td><td colspan="24">0x00FFFF</td></tr>
<tr><td colspan="8">-</td><td colspan="24">RW</td></tr>
</table>

23 0	RTOD	LPUARTn Receive Time Out Data

18.3.10 LPUARTn_RCDR: LPUARTn receive character detection data register

LPUARTn_RCDR register is 32-bit size and accessible in 32/16/8-bit. (n = 0 and 1)

LPUART0_RCDR=0x4000_5C2C, LPUART1_RCDR=0x4000_5CAC

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="24">Reserved</td><td colspan="8">RCDD</td></tr>
<tr><td colspan="24">0x000000</td><td colspan="8">0x00</td></tr>
<tr><td colspan="24">-</td><td colspan="8">RW</td></tr>
</table>

7 0	RCDD	LPUARTn Receive Character Detection Data.

18.3.11 LPUARTn_DLYDR: LPUARTn Tx delay time data register

LPUARTn_DLYDR register is 32-bit size and accessible in 32/16/8-bit. (n = 0 and 1)

LPUART0_DLYDR=0x4000_5C30, LPUART1_DLYDR=0x4000_5CB0

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="24">Reserved</td><td colspan="8">DLYD</td></tr>
<tr><td colspan="24">0x000000</td><td colspan="8">0x00</td></tr>
<tr><td colspan="24">-</td><td colspan="8">RW</td></tr>
</table>

7 0	DLYD	LPUARTn Tx Delay Data. This register is used for transmit delay time between the last stop bit and the next start bit with baud rate unit. The data in the LPUARTn_TDR register will be transferred to the transmit shift register after delay time. Delay time: DLYD[7:0] x “baud rate clock period”. No delay on DLYD[7:0] = 0.

18.4 Functional description

The LPUARTn block comprises a clock generator, a transmitter and a receiver.

The clock generation logic consists of a baud rate generator.

The transmitter consists of a write buffer, a serial shift register, a parity generator, and a control logic. Using DMA allows continuous transfer of data without any s/w involvement between frames.

The receiver is the most complex part of the low power UARTn module due to its clock and data recovery units. The recovery unit is used for asynchronous data reception. In addition, the receiver has a parity checker, a shift register, and a control logic. The receiver supports the same frame formats as the transmitter and can detect frame error, data overrun and parity errors.

18.4.1 LPUARTn clock generation

The clock generation logic generates clocks for the transmitter and the receiver. The LPUARTn baud rate generator supports three modes of clock operation, which are 16 oversampling mode, 8 oversampling mode, and only 1 sampling mode. The only 1 sampling mode can be used with XSOSC (32.768kHz).

Table 82 shows equations for baud rate calculation (in bps).

Table 82. Equations for Calculating Baud Rate Register Settings

Oversampling	Equation for calculating baud rate
16 oversampling mode (OVRS = 00b)	Baud Rate = f_{LPUART}/(16(LPUARTn_BDR+1))
8 oversampling mode (OVRS = 01b)	Baud Rate = f_{LPUART}/(8(LPUARTn_BDR+1))
Only 1 sampling mode (OVRS = 10b)	Baud Rate = f_{LPUART}/(LPUARTn_BDR+1)

18.4.2 LPUARTn baud rate compensation

The baud rate compensation is used to optimize the precision in each bit. There is a sign (BCMPS bit of LPUARTn_BCMP register) bit to define the positive or negative compensation in each bit. If the sign bit is “0b”, one clock of fLPUARTn will be appended to the compensated bit. If the sign bit is “1b”, one clock of fLPUARTn will be taken out from the compensated bit.

There are nine bits to define whether the relative compensation is required for each bit. The bits are BCMP[7:0] for data and BCMP8 for parity.

Example

1. fLPUARTn = 32.768kHz, No oversampling, Baud rate = 9600 bps

   32.768kHz/(1 x 9600) = 3.413, LPUARTn_BDR = 3 - 1 = 2, and “Baud rate clock”/bit = 3 x 1

   So, “Clock error”/bit: 3.413 x 1 - 3 x 1 = 0.413 clock → “1 clock compensation”/bit if a BCMPx bit is “1b”.

   The result is that the sign bit, BCMPS, is “0b” for positive compensation and the baud rate compensation bits, BCMP[8:0], are “010100101b”. (CEPB: “clock error”/bit)

Table 83. Baud Rate Compensation Example 1

Rx/Tx bit	BCMPx bit	Clock Error	Compensation bit	Final clock error
Start bit	–	-0.413 (CEPB)	x	-0.413
D0	bit 0	-0.827 (CEPB+ before compensation)	1	0.173
D1	bit 1	-0.240 (CEPB+ before compensation)	0	-0.240
D2	bit 2	-0.653 (CEPB+ before compensation)	1	0.347
D3	bit 3	-0.067 (CEPB+ before compensation)	0	-0.067
D4	bit 4	-0.480 (CEPB+ before compensation)	0	-0.480
D5	bit 5	-0.893 (CEPB+ before compensation)	1	0.107
D6	bit 6	-0.307 (CEPB+ before compensation)	0	-0.307
D7	bit 7	-0.720 (CEPB+ before compensation)	1	0.280
Parity bit	bit 8	-0.133 (CEPB+ before compensation)	0	-0.133

2. fLPUARTn = 32.768kHz, No oversampling, Baud rate = 2400 bps

   32.768kHz/(1 x 2400) = 13.653, LPUARTn_BDR = 14 - 1 = 13, and “Baud rate clock”/bit = 14 x 1

   So, “Clock error”/bit: 13.653 x 1 - 14 x 1 = - 0.347 clock → “1 clock compensation”/bit if a BCMPx bit is “1b”.

   The result is that the sign bit, BCMPS, is “1b” for negative compensation and the baud rate compensation bits, BCMP[8:0], are “001001001b”. (CEPB: “clock error”/bit)

Table 84. Baud Rate Compensation Example 2

Rx/Tx bit	BCMPx bit	Clock Error	Compensation bit	Final Clock Error
Start bit	–	+0.347 (CEPB)	x	0.347
D0	bit 0	0.693 (CEPB+ before compensation)	1	-0.307
D1	bit 1	0.040 (CEPB+ before compensation)	0	0.040
D2	bit 2	0.387 (CEPB+ before compensation)	0	0.387
D3	bit 3	0.733 (CEPB+ before compensation)	1	-0.267
D4	bit 4	0.080 (CEPB+ before compensation)	0	0.080
D5	bit 5	0.427 (CEPB+ before compensation)	0	0.427
D6	bit 6	0.773 (CEPB+ before compensation)	1	-0.227
D7	bit 7	0.120 (CEPB+ before compensation)	0	0.120
Parity bit	bit 8	0.467 (CEPB+ before compensation)	0	0.467

18.4.3 LPUARTn interface data format

A serial frame is defined to be composed of one character of data bits with synchronization bits (start and stop bits) and an optional parity bit for error detection.

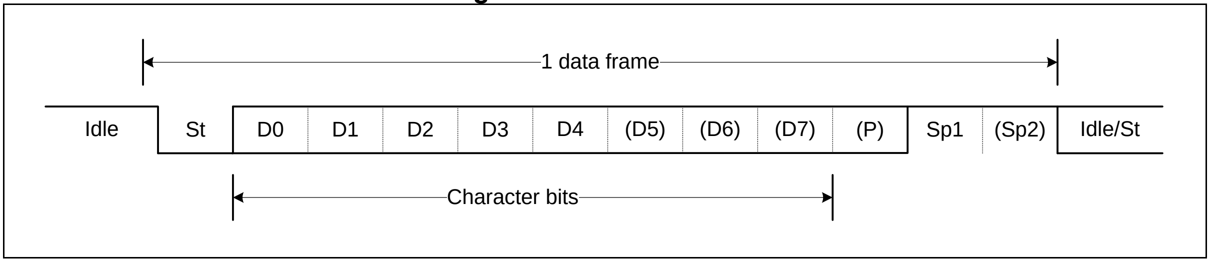
The LPUARTn supports all 24 combinations of the followings as valid frame formats.

- 1 start bit
- 5, 6, 7, or 8 data bits
- No, even, or odd parity bit.
- 1 or 2 stop bits.

A frame starts with the start bit followed by the least significant data bit (LSB). Then the next data bits, up to eight, follow, ending with the most significant bit (MSB). If parity function is enabled, the parity bit is inserted between the last data bit and the stop bit.

A high-to-low transition on data pin is considered as start bit. When a complete frame is transmitted, it can be directly followed by a new frame, or the communication line can be set to an idle state. The idle state means high state of data pin. The following figure shows the possible combinations of the frame formats. Bits inside round brackets are optional.

Figure 94. Frame Format



1 data frame consists of the following bits:

- Idle: No communication on communication line (LPTXDn/LPRXDn)
- St: Start bit (Low)
- Dm: Data bits (0 ~ 7)
- P: Parity bit (even parity, odd parity, no parity)
- Sp: Stop bit (1 bit or 2 bits)

The frame format is set by configuring DLEN[1:0], PSEL, PEN, and STOPB bits in the LPUARTn_CR1 register. The transmitter and the receiver use the same values.

18.4.4 LPUARTn interface parity bit

The parity bit is calculated by doing an exclusive-OR of all data bits. If odd parity is used, the result of the exclusive-OR is inverted. The parity bit is located between the last data bit and first stop bit of a serial frame.

- Peven = Dm-1 ^ … ^ D3 ^ D2 ^ D1 ^ D0 ^ 0
- Podd = Dm-1 ^ … ^ D3 ^ D2 ^ D1 ^ D0 ^ 1
- Peven: Parity bit using even parity
- Podd: Parity bit using odd parity
- Dm: Data bit n of the character

18.4.5 LPUARTn transmitter

The LPUARTn transmitter is enabled by setting the TXE bit in LPUARTn_CR1 register. When the transmitter is enabled, the LPTXDn pin should be set to LPTXDn function for the serial output pin by the GPIO registers.

Baud-rate, operation mode and frame format must be set up before doing any transmission.

18.4.5.1 LPUARTn sending TX data

A data transmission is initiated by loading data to the transmit data register (LPUARTn_TDR register). The data to be written in transmit data register is moved to the shift register when the shift register is ready to send a new frame.

The shift register is loaded with the new data if it is in idle state or immediately after the last stop bit of the previous frame is transmitted. When the shift register is loaded to the new data, it will transfer one complete frame according to the settings of control registers. (n = 0 and 1)

18.4.5.2 LPUARTn parity generator

The parity generator calculates parity bit for the serial frame data to be sent. When the parity bit is enabled (PENn = 1), the transmitter control logic inserts the parity bit between the last data bit and the first stop bit of the frame to be sent.

18.4.6 LPUARTn receiver

The LPUARTn receiver is enabled by setting the RXE bit in the LPUARTn_CR1 register. When the receiver is enabled, the LPRXDn pin should be set to LPRXDn function for the serial input pin by the GPIO registers.

Baud-rate, operation mode, and frame format must be set before the serial reception.

18.4.6.1 LPUARTn receiving RX data

The receiver starts data reception when it detects a valid start bit (LOW) on LPRXDn pin. Each bit after start bit is sampled at predefined baud-rate, and shifted into the receive shift register until the first stop bit of a frame is received.

Even if there is the second stop bit in the frame, the second stop bit is ignored by the receiver. That is, receiving the first stop bit means that a complete serial frame is presented in the receiver shift register and contents of the shift register are to be moved into the receive data register.

18.4.6.2 LPUARTn parity checker

If the parity bit is enabled (PEN = 1), the Parity Checker calculates the parity of the data bits in incoming frame and compares the result with the parity bit from the received serial frame.

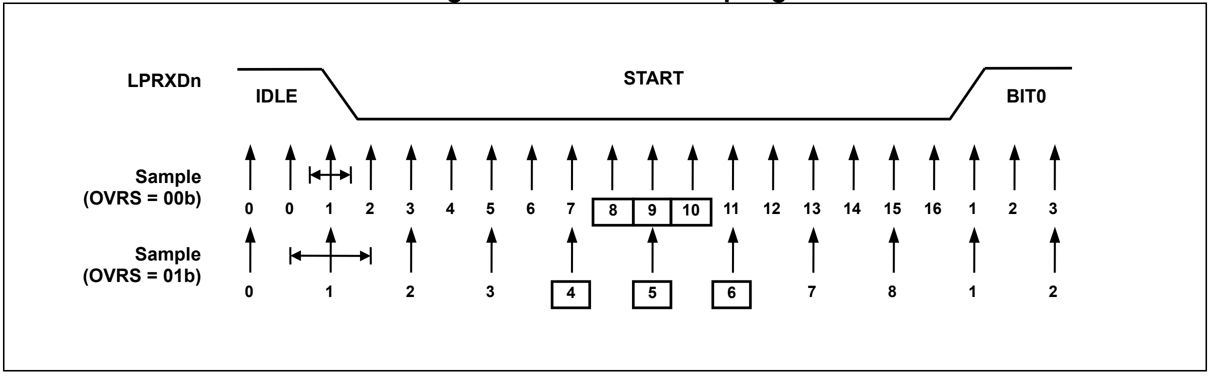
18.4.6.3 LPUARTn data reception

To receive data frame, the receiver includes a clock and data recovery unit. The clock recovery logic is used for synchronizing the internally generated baud-rate clock to the incoming asynchronous serial frame on the LPRXDn pin.

The data recovery logic samples and filters the incoming bits with a low pass filter, and removes the noise of receive pin.

Figure 95 illustrates the sampling process of a start bit of an incoming frame. The sampling rate is 16 times the baud rate in 16 oversampling mode and 8 times the baud rate for 8 oversampling mode. The horizontal arrows show the synchronization variation due to the asynchronous sampling process. Note that larger time variation is seen when using 8 oversampling mode.

Figure 95. Start Bit Sampling



When the receiver is enabled (RXEn=1), the clock recovery logic tries to find a high-to-low transition on the LPRXDn line, the start bit condition. After detecting high to low transition on the line, the clock recovery logic uses samples 8, 9 and 10 for 16 oversampling mode to detect whether valid start bit is received.

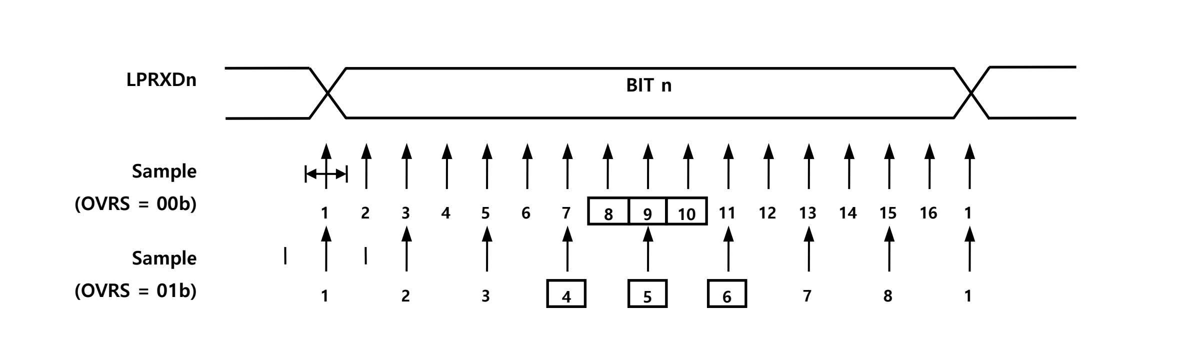
If more than 2 samples have logical low level, it is considered that a valid start bit is detected and the internally generated clock is synchronized to the incoming data frame. Then the data recovery can begin. The synchronization process is repeated for each start bit.

As described above, when the receiver clock is synchronized to the start bit, the data recovery can begin. Data recovery process is almost the same as clock recovery process. The data recovery logic samples each incoming bit 16 times for 16 oversampling mode and 8 times for 8 oversampling mode, and uses sample 8, 9 and 10 to decide data value. If more than 2 samples have low levels, the received bit is considered as a logic '0' and if more than 2 samples have high levels, the received bit is considered as a logic '1'.

The data recovery process is then repeated until a complete frame is received, including the first stop bit. The decided bit value is stored in the receive shift register in order.

Note that the receiver only uses the first stop bit of a frame. Internally, after receiving the first stop bit, the receiver is in idle state and waits to find the start bit.

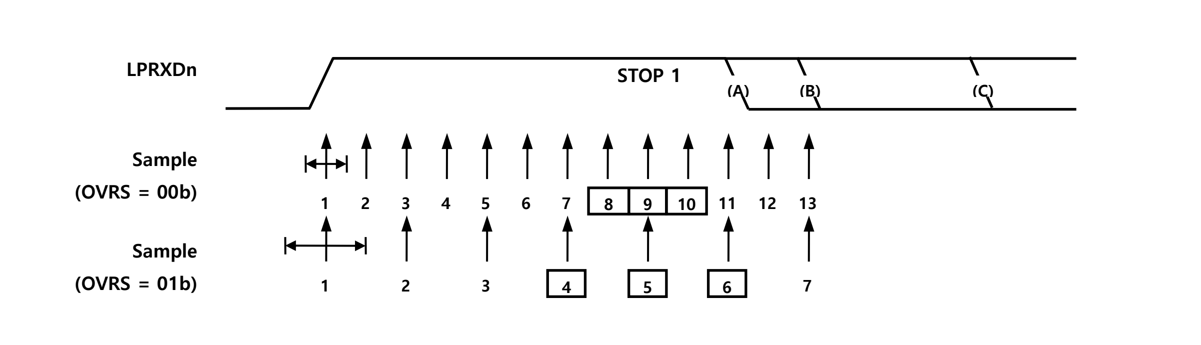
Figure 96. Sampling of Data and Parity Bit



The process for detecting stop bit is the same as clock and data recovery process. That is, if 2 or more samples of 3 center values have high level, correct stop bit is detected, or else a frame error (FE) flag is set.

After deciding whether the first stop bit is valid or not, the receiver goes to idle state and monitors the LPRXDn line to check whether a valid high to low transition is detected (start bit detection).

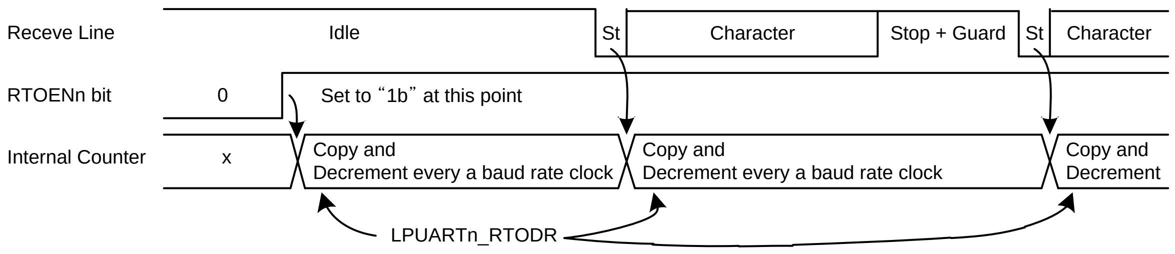
Figure 97. Stop Bit Sampling and Next Start Bit Sampling



18.4.6.4 LPUARTn receive time out function

The receive time out function is used for checking a frame finish. This function is to count time with baud rate unit between the last start bit and a new start bit, and between setting the RTOEN bit of the LPUARTn_CR register and a new start bit. The LPUARTn_RTODR register should have duration time value before using the receive time out function.

Figure 98. Receive Time Out Function



NOTES:

1. The value of LPUARTn_RTODR register is transferred to an internal counter when the RTOENn bit is set to "1b" or when a start bit is received to.
2. If the RTOEN bit is set to "1b", the internal counter counts down until underflow.
3. The RTOnIFLAG bit of LPUARTn_IFSR register is set to "1b" on the internal counter underflow and the receive time out function is finished. So, to re-use the receive time out function, the RTOEN bit must be cleared to "0b" and then set to "1b" again.

18.4.6.5 1-wire half-duplex communication

1-wire half-duplex mode is selected by configuring HDCOM bit in the LPUARTn_CR1 register. The TXDn and the RXDn lines are internally connected, the RXDn pin is not used, and the TXDn pin is always an input when no transmitted. So, the TXDn pin must be configured to open-drain with an external pull-up resistor.

19 I2C 0/1/2 Interface

I2C is one of industrial standard serial communication protocols, which uses 2 bus lines, Serial Data Line (SDAn) and a Serial Clock Line (SCLn). These are used to exchange data.

Because both of the SDAn and SCLn lines are open-drain outputs, each line needs a pull-up resistor (n = 0, 1, and 2).

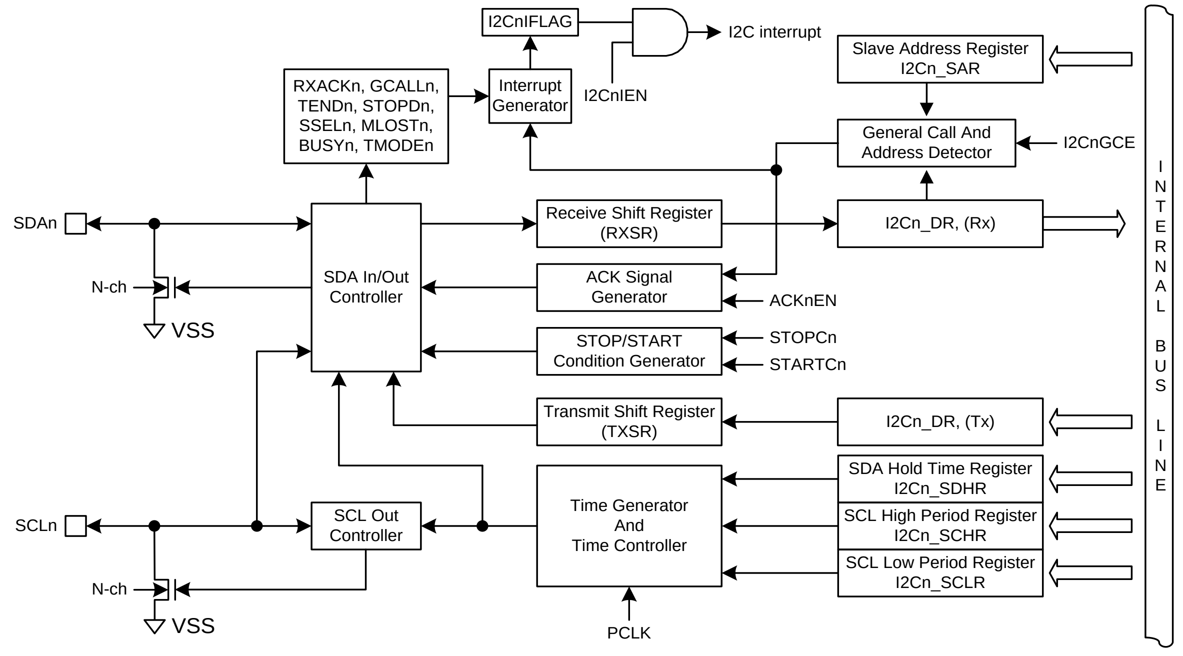
The I2C interface 0/1/2 of the A31L21x series features the followings:

- Compatible with I2C bus standard
- Multi-master operation
- Up to 1MHz data transfer read speed
- 7-bit address
- Two slave addresses supported
- Master and slave operations
- Bus busy detection

19.1 I2C 0/1/2 block diagram

Figure 99 shows a block diagram of the I2C block.

Figure 99. I2C Block Diagram (n = 0, 1, and 2)



NOTE: When the corresponding port of I2C is a sub-function for SCLn/SDAn pin, the SCLn/SDAn pins are automatically set as N-channel open-drain outputs and the input latch is read when reading the pins. The corresponding pull-up resistor is determined by the control register.

19.2 Pin description for I2C 0/1/2

Table 85. Pins and External Signals for I2C (n = 0, 1, and 2)

Pin name	Type	Description
SCLn	I/O	I2C channel n Serial clock bus line (open-drain)
SDAn	I/O	I2C channel n Serial data bus line (open-drain)

19.3 Registers

Base address and register map of the I2C 0/1/2 are shown in Table 86 and Table 87.

Table 86. Base Address of I2C Interface

Name	Base address	Size	Description
I2C0	0x4000_4800	256	I2C0 Block
I2C1	0x4000_4900	256	I2C1 Block
I2C2	0x4000_4A00	256	I2C2 Block

Table 87. I2C Register Map (n = 0, 1, and 2)

Name	Offset	Type	Description	Reset Value
I2Cn_CR	0x00	RW	I2Cn Control Register	0x00000000
I2Cn_ST	0x04	RW	I2Cn Status Register	0x00000000
I2Cn_SAR1	0x08	RW	I2Cn Slave Address Register 1	0x00000000
I2Cn_SAR2	0x0C	RW	I2Cn Slave Address Register 2	0x00000000
I2Cn_DR	0x10	RW	I2Cn Data Register	0x00000000
I2Cn_SDHR	0x14	RW	I2Cn SDA Hold Time Register	0x00000001
I2Cn_SCLR	0x18	RW	I2Cn SCL Low Period Register	0x0000003F
I2Cn_SCHR	0x1C	RW	I2Cn SCL High Period Register	0x0000003F

19.3.1 I2Cn_CR: I2Cn control register

The register can be set to configure I2C operation mode activate I2C transactions.

I2Cn_CR register is 32-bit size and accessible in 32/16/8-bit. (n = 0, 1, and 2)

I2C0_CR=0x4000_4800, I2C1_CR=0x4000_4900, I2C2_CR=0x4000_4A00

31 30 29 28 27 26 25 24 | 23 22 21 20 19 18 17 16 | 15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
Reserved	I2CnEN	TXDLYENBn	I2CnIEN	I2CnIFLAG	ACKnEN	IMASTERn	STOPCn	STARTCn
0x000000	0	0	0	0	0	0	0	0
-	RW	RW	RW	RO	RW	RO	RW	RW

7	I2CnEN	Activate I2Cn Block.
		0 Disable I2Cn block.
		1 Enable I2Cn block.
6	TXDLYENBn	I2Cn_SDHR Register Control.
		0 Enable I2Cn_SDHR register.
		1 Disable I2Cn_SDHR register.
5	I2CnIEN	I2Cn Interrupt Enable.
		0 Disable I2Cn interrupt.
		1 Enable I2Cn interrupt.
4	I2CnIFLAG	I2Cn Interrupt Flag. This bit is cleared when all interrupt source bits in the I2Cn_ST register are cleared to ‘0’.
		0 No request occurred.
		1 Request occurred.
3	ACKnEN	Controls ACK signal generation at ninth SCL period.
		0 No ACK signal is generated. (SDA = 1)
		1 ACK signal is generated. (SDA = 0)
		NOTES: ACK signal is output (SDA = 0) for the following 3 cases. — When received address packet is equal to SLAn[6:0] bits in I2Cn_SAR1/I2Cn_SAR2 register. — When received address packet is equal to value 0x00 with GCALLn enabled. — When I2Cn operates as a receiver (master or slave)
2	IMASTERn	Represents Operation Mode of I2Cn. This bit is cleared to ‘0’ on STOP condition.
		0 I2Cn is in slave mode.
		1 I2Cn is in master mode.
1	STOPCn	STOP Condition Generation When I2Cn is master.
		0 No effect.
		1 Generate STOP condition.
0	STARTCn	START Condition Generation When I2Cn is master.
		0 No effect.
		1 Generate START or Repeated START condition.

19.3.2 I2Cn_ST: I2Cn status register

I2Cn_ST register is 32-bit size and accessible in 32/16/8-bit. (n = 0, 1, and 2)

I2C0_ST=0x4000_4804, I2C1_ST=0x4000_4904, I2C2_ST=0x4000_4A04

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
Reserved			**GCALLn**	**TENDn**	**STOPDn**	**SSELn**	**MLOSTn**	**BUSYn**	**TMODEn**	**RXACKn**
0x000000			**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
-			**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RO**	**RW**

7	GCALLn	This bit has different meaning depending on whether I2C is master or slave. When I2C is a master, this bit represents whether it received AACK (address ACK) from slave.	
		0	No AACK is received. (Master mode)
		1	AACK is received (Master mode). It may be set to ‘1’ after address transmission.
		When I2C is a slave, this bit is used to indicate general call.	
		0	General call address is not detected. (Slave mode)
		1	General call address is detected. (Slave mode)
6	TENDn	This bit is set when 1-byte of data is transferred completely.	
		0	1 byte of data is not completely transferred.
		1	1 byte of data is completely transferred.
5	STOPDn	This bit is set when a STOP condition is detected.	
		0	A STOP condition is not detected.
		1	A STOP condition is detected.
4	SSELn	This bit is set when I2C is addressed by other master.	
		0	I2C is not selected as a slave.
		1	I2C is addressed by other master and acts as a slave.
3	MLOSTn	This bit represents the result of bus arbitration in master mode.	
		0	I2C maintains bus mastership.
		1	I2C has lost bus mastership during arbitration process.
2	BUSYn	This bit reflects bus status.	
		0	I2C bus is idle, so a master can issue a START condition.
		1	I2C bus is busy.
1	TMODEn	This bit is used to indicate whether I2C is transmitter or receiver.	
		0	I2C is a receiver.
		1	I2C is a transmitter.
0	RXACKn	This bit shows the state of ACK signal.	
		0	No ACK is received.
		1	ACK is received at ninth SCL period.

NOTES:

1. The GCALLn, TENDn, STOPDn, SSELn, and MLOSTn bits can be source of interrupt.
2. When an I2C interrupt occurs except for DEEP SLEEP mode, the SCL line is held low. To release SCL, Clear to “0b” all interrupt source bits in I2Cn_ST register.
3. The GCALLn, TENDn, STOPDn, SSELn, MLOSTn, and RXACKn bits are cleared when ‘1’ is written to the corresponding bit.

19.3.3 I2Cn_SAR1: I2Cn slave address register 1

I2Cn_SAR1 register is 32-bit size and accessible in 32/16/8-bit. (n = 0, 1, and 2)

I2C0_SAR1=0x4000_4808, I2C1_SAR1=0x4000_4908, I2C2_SAR1=0x4000_4A08

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8	7 6 5 4 3 2 1	0
Reserved	**SLAn**	**GCALLnEN**
0x000000	**0000000**	**0**
-	**RW**	**RW**

7 1	SLAn	These bits configure the slave address 1 in slave mode.
0	GCALLnEN	This bit decides whether I2Cn allows general call address 1 or not in I2Cn slave mode. 0 Ignore general call address 1. 1 Allow general call address 1.

19.3.4 I2Cn_SAR2: I2Cn slave address register 2

I2Cn_SAR2 register is 32-bit size and accessible in 32/16/8-bit. (n = 0, 1, and 2)

I2C0_SAR2=0x4000_480C, I2C1_SAR2=0x4000_490C, I2C2_SAR2=0x4000_4A0C

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8	7 6 5 4 3 2 1	0
Reserved	**SLAn**	**GCALLnEN**
0x000000	**0000000**	**0**
-	**RW**	**RW**

7 1	SLAn	These bits configure the slave address 2 in slave mode.
0	GCALLnEN	This bit decides whether I2Cn allows general call address 2 or not in I2Cn slave mode. 0 Ignore general call address 2. 1 Allow general call address 2.

19.3.5 I2Cn_DR: I2Cn data register

I2Cn_DR register is 32-bit size and accessible in 32/16/8-bit. (n = 0, 1, and 2)

I2C0_DR=0x4000_4810, I2C1_DR=0x4000_4910, I2C2_DR=0x4000_4A10

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																								**DATA**							
0x000000																								**0x00**							
-																								**RW**							

7 0	DATA	The I2Cn_DR Transmit buffer and Receive buffer share the same I/O address with this DATA register. The Transmit Data Buffer is the destination for data written to the I2Cn_DR register. Reading the I2Cn_DR register returns the contents of the Receive Buffer.

19.3.6 I2Cn_SDHR: I2Cn SDA hold time register

I2Cn_SDHR register is 32-bit size and accessible in 32/16/8-bit. (n = 0, 1, and 2)

I2C0_SDHR=0x4000_4814, I2C1_SDHR=0x4000_4914, I2C2_SDHR=0x4000_4A14

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																				**HLDT**											
0x00000																				**0x001**											
-																				**RW**											

11 0	HLDT	This register is used to control SDA output timing from the falling edge of SCL. Note that SDA is changed after tPCLK X (I2Cn_SDHR+2). In master mode, load half the value of I2Cn_SCLR to this register to make SDA switch in the middle of SCL. In slave mode, configure this register regarding the frequency of SCL from master. The SDA is changed after tPCLK X (I2Cn_SDHR+2) in master mode. So, to ensure proper operation in slave mode, the value tPCLK X (I2Cn_SDHR + 2) must be smaller than the period of SCL.

19.3.7 I2Cn_SCLR: I2Cn SCL low period register

I2Cn_SCLR register is 32-bit size and accessible in 32/16/8-bit. (n = 0, 1, and 2)

I2C0_SCLR=0x4000_4818, I2C1_SCLR=0x4000_4918, I2C2_SCLR=0x4000_4A18

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="20">Reserved</td><td colspan="12">SCLL</td></tr>
<tr><td colspan="20">0x00000</td><td colspan="12">0x03F</td></tr>
<tr><td colspan="20">-</td><td colspan="12">RW</td></tr>
</table>

11 0	SCLL	This register defines the low period of SCL in master mode. The base clock is PCLK and the period is calculated by the formula: tPCLK X (4 X I2Cn_SCLR + 3) where tPCLK is the period of PCLK.

19.3.8 I2Cn_SCHR: I2Cn SCL high period register

I2Cn_SCHR register is 32-bit size and accessible in 32/16/8-bit. (n = 0, 1, and 2)

I2C0_SCHR=0x4000_481C, I2C1_SCHR=0x4000_491C, I2C2_SCHR=0x4000_4A1C

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="20">Reserved</td><td colspan="12">SCLH</td></tr>
<tr><td colspan="20">0x00000</td><td colspan="12">0x03F</td></tr>
<tr><td colspan="20">-</td><td colspan="12">RW</td></tr>
</table>

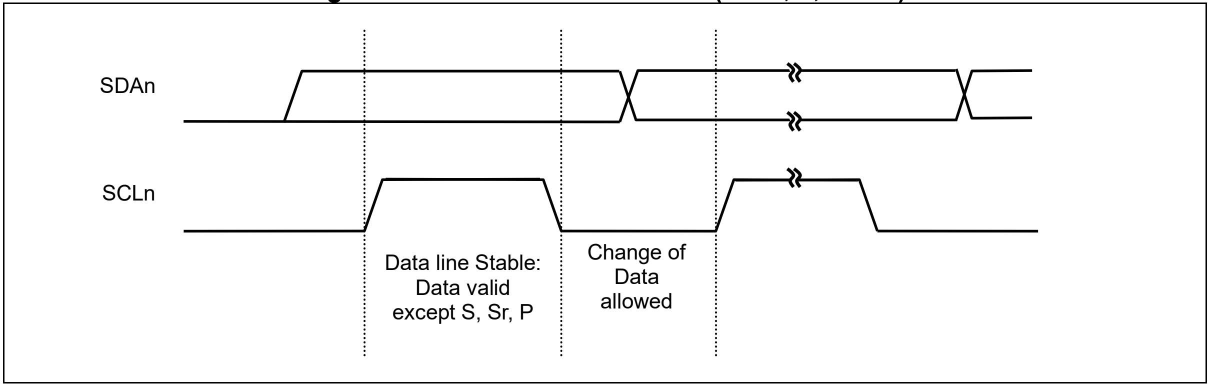
11 0	SCLH	This register defines the high period of SCL in master mode. The base clock is PCLK and the period is calculated by the formula: tPCLK X (4 X I2Cn_SCHR + 3) where tPCLK is the period of PCLK.

19.4 Functional description

19.4.1 I2C bit transfer

The data on the SDAn line must be stable during HIGH period of the clock, SCLn. The HIGH or LOW state of the data line can only change when the clock signal on the SCLn line is LOW. The exceptions are START(S), repeated START(Sr), and STOP(P) condition, where data line changes when clock line is high.

Figure 100. I2C Bus Bit Transfer (n = 0, 1, and 2)



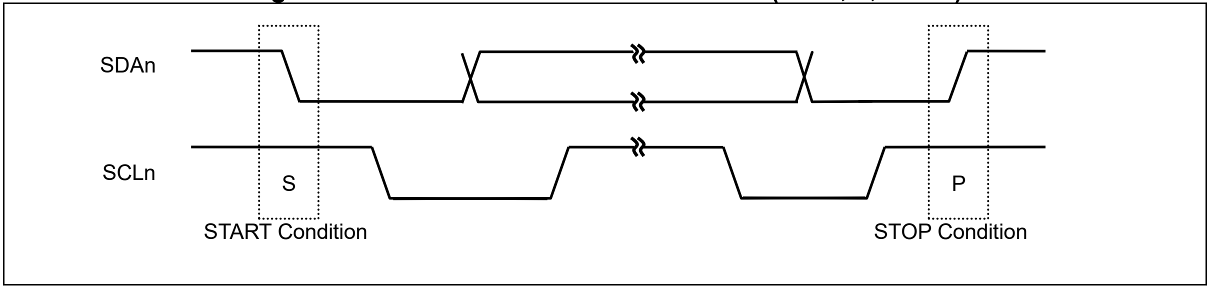
19.4.2 START/Repeated START/STOP

One master can issue a START (S) condition to detect other devices connected to the SCLn, SDAn lines that will use the bus. A STOP (P) condition is generated by the master to release the bus lines so that other devices can use it.

- A high to low transition on the SDAn line while SCLn is high defines a START (S) condition.
- A low to high transition on the SDAn line while SCLn is high defines a STOP (P) condition.

START and STOP conditions are always generated by the master. The bus is considered to be busy after START condition. The bus is considered to be free again after STOP condition, i.e., the bus is busy between START and STOP condition. If a repeated START condition (Sr) is generated instead of STOP condition, the bus stays in busy mode. So, the START and repeated START conditions are functionally identical.

Figure 101. START and STOP Conditions (n = 0, 1, and 2)



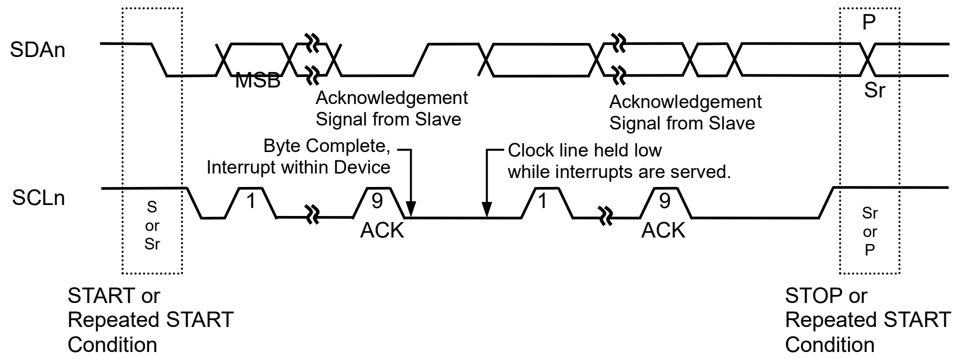
19.4.3 Data transfer

Every byte on the SDAn line must be 8-bits long, but the number of bytes that can be transmitted per transfer is unlimited.

Each byte has to be followed by an acknowledge bit. Data is transferred with the most significant bit (MSB) first. If a slave cannot receive or transmit another complete byte of data until it has performed some other function, it can hold the clock line SCLn LOW to force the master into a wait state.

Data transfer then continues when the slave is ready for another byte of data and releases clock line SCLn.

Figure 102. I2C Bus Data Transfer (n = 0, 1, and 2)


SDAn
MSB
Acknowledgement Signal from Slave
Acknowledgement Signal from Slave
P
Sr
Byte Complete, Interrupt within Device
Clock line held low while interrupts are served.
SCLn
S or Sr
1
9
ACK
1
9
ACK
Sr or P
START or Repeated START Condition
STOP or Repeated START Condition

19.4.4 Acknowledge

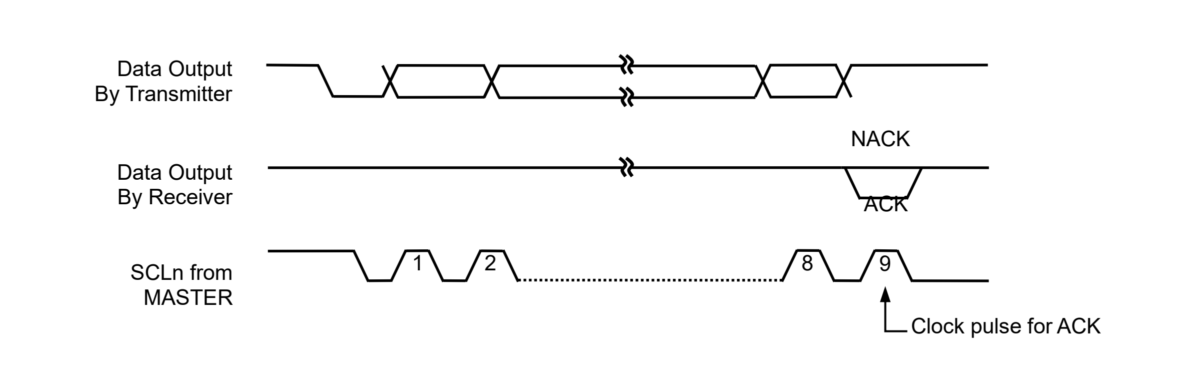
An acknowledge clock pulse is generated by the master. The transmitter releases the SDAn line (HIGH) during the acknowledge clock pulse. The receiver must pull down the SDAn line during the acknowledge clock pulse so that it remains stable at LOW during the HIGH period of this clock pulse.

When a slave is addressed by a master (Address Packet), and if it is unable to receive or transmit because it is performing some real time function, the data line must be left HIGH by the slave.

In addition, when a slave addressed by a master is unable to receive more data bits, the slave receiver must release the SDAn line (Data Packet). The master can then generate either a STOP condition to abort the transfer, or a repeated START condition to start a new transfer.

If a master receiver is involved in a transfer, it must signal the end of data to the slave transmitter by not generating an acknowledge on the last byte that was clocked out of the slave. The slave transmitter must release the data line to allow the master to generate a STOP or repeated START condition.

Figure 103. I2C Bus Acknowledge (n = 0, 1, and 2)



19.4.5 Synchronization/arbitration

Clock synchronization is performed using the wired-AND connection of I2C interfaces to the SCLn line. This means that a HIGH to LOW transition on the SCLn line will cause the devices concerned to start counting off their LOW period and it will hold the SCLn line in that state until the clock HIGH state is reached.

However the LOW to HIGH transition of this clock may not change the state of the SCLn line if another clock is still within its LOW period. In this way, a synchronized SCLn clock is generated with its LOW period determined by the device with the longest clock LOW period, and its HIGH period determined by the one with the shortest clock HIGH period. A master may start a transfer only if the bus is free. Two or more masters may generate a START condition.

Arbitration takes place on the SDAn line, while the SCLn line is at the HIGH level, in such a way that a master that transmits a HIGH level, while another master that transmits a LOW level, will switch off its DATA output state because the level on the bus does not correspond to its own level.

Arbitration continues for many bits until a winning master gets the ownership of I2C bus. Its first stage is comparison of the address bits.

Figure 104. Clock Synchronization during Arbitration Procedure (n = 0, 1, and 2)

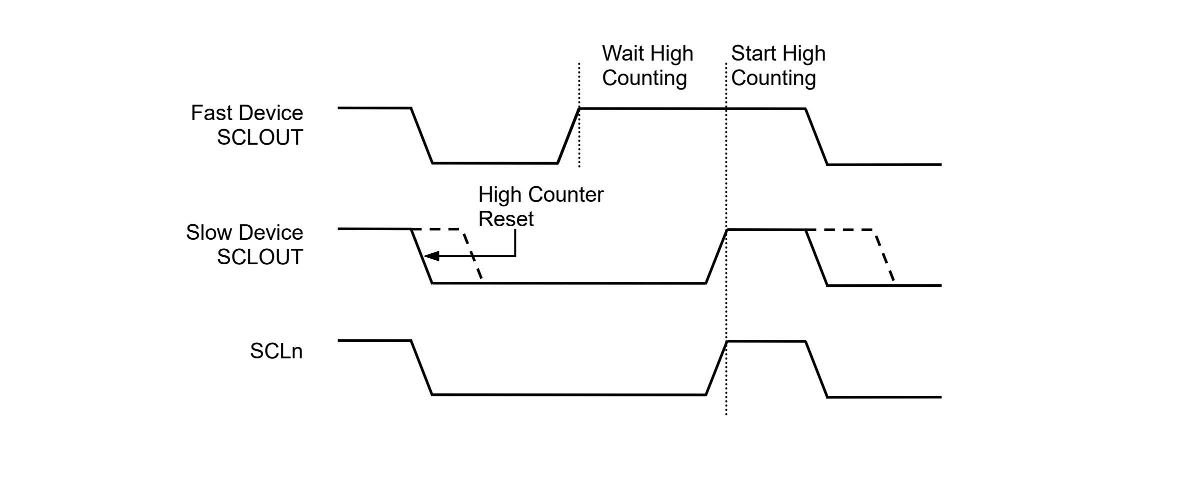
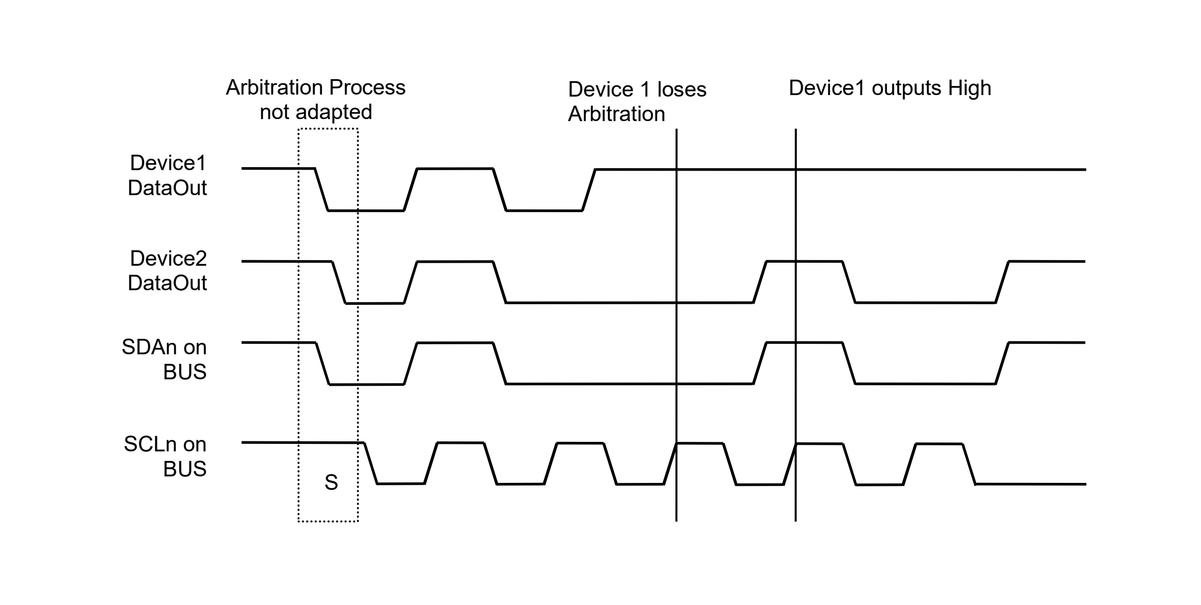


Figure 105. Arbitration Procedure between Two Masters (n = 0, 1, and 2)



19.5 I2C operation

The I2C is byte-oriented and interrupt-based. Interrupts are issued after all bus events except for the transmission of a START condition. Since I2C is interrupt based, the application software is free to carry on with other operations during an I2C byte transfer.

Note that when an I2C interrupt is generated, I2CnIFLAG flag in I2Cn_CR register is set, and it is cleared when all interrupt source bits in the I2Cn_ST register are cleared to ‘0’. When I2C interrupt occurs, the SCLn line is held at LOW until all interrupt source bits in I2Cn_ST register are cleared to ‘0’. When the I2CnIFLAG flag is set, the I2Cn_ST contains a value indicating the current state of the I2C bus. According to the value in I2Cn_ST, software can decide what to do next.

I2C can operate in 4 modes: master/slave, transmitter/receiver. The operating mode is configured by a winning master.

A more detailed explanation follows below. (n = 0, 1, and 2)

19.5.1 Master transmitter

To operate I2C as a master transmitter, follow the recommended steps below.

1. Enable I2C by setting I2CnEN bit in I2Cn_CR. This provides main clock to the peripheral.

2. Load SLA+W into the I2Cn_DR, where SLA is the address of slave device and W is the transfer direction from the viewpoint of master. For master transmitter, W is ‘0’. Note that I2Cn_DR is used for both address and data.

3. Configure baud rate by writing desired value to both I2Cn_SCLR and I2Cn_SCHR for the Low and High period of SCLn line.

4. Configure the I2Cn_SDHR to decide when SDAn changes value from falling edge of SCLn. If SDA should change in the middle of SCLn LOW period, load half the value of I2Cn_SCLR to the I2Cn_SDHR.

5. Set the STARTCn bit in I2Cn_CR. This transmits a START condition. Also, configure how to handle interrupt and ACK signal. When the STARTCn bit is set, 8-bit data in I2Cn_DR is transmitted out according to the baud-rate.

6. This is ACK signal processing stage for address packet transmitted by master. When 7-bit address and 1-bit transfer direction is transmitted to target slave device, the master can know whether the slave acknowledged or not in the 9th high period of SCLn. If the master gains bus mastership, I2C generates GCALL interrupt regardless of receiving ACK from the slave device. When I2C loses bus mastership during arbitration process, the MLOSTn bit in I2Cn_ST is set, and I2C waits in idle state or can be operated as an addressed slave.

To operate as a slave when the MLOSTn bit in I2Cn_ST is set, the ACKnEN bit in I2Cn_CR must be set and the received 7-bit address must match the SLAn bits in I2Cn_SAR1/2. In this case, I2C operates as a slave transmitter or a slave receiver (go to appropriate section). In this stage, I2C holds the SCLn LOW. The reason for this is to decide whether I2C should continue serial transfer or stop communication. The following steps continue, assuming that I2C does not lose mastership during the first data transfer.

I2C (Master) can choose one of the following cases regardless of receiving ACK signal from slave.

A. Master receives ACK signal from slave, so continues data transfer since slave can receive more data from master. In this case, load data to transmit to I2Cn_DR.

B. Master stops data transfer even if it receives ACK signal from slave. In this case, set the STOPC bit in I2Cn_CR.

C. Master transmits repeated START condition without checking ACK signal. In this case, load SLA+R/W into the I2Cn_DR and set STARTCn bit in I2Cn_CR.

After doing any of the actions above, clear all interrupt source bits in I2Cn_ST to ‘0’ to release SCLn line. In case of A, move to step 7. In case of B, move to step 9 to handle STOP interrupt. In case of C, move to step 6 after transmitting the data in I2Cn_DR, and if transfer direction bit is ‘1’, go to master receiver section.

7. 1-Byte of data is transmitted. During data transfer, bus arbitration continues.

8. This is ACK signal processing stage for data packet transmitted by master. I2C holds the SCLn LOW. When I2C loses bus mastership while transmitting data to arbitrate other masters, the MLOSTn bit in I2Cn_ST is set. If then, I2C waits in idle state. When the data in I2Cn_DR is transmitted completely, I2C generates TENDn interrupt.

   I2C can choose one of the following cases regardless of receiving ACK signal from slave.

   A. Master receives ACK signal from slave, so continues data transfer since slave can receive more data from master. In this case, load data to transmit to I2Cn_DR.

   B. Master stops data transfer even if it receives ACK signal from slave. In this case, set the STOPCn bit in I2Cn_CR.

   C. Master transmits repeated START condition without checking ACK signal. In this case, load SLA+R/W into the I2Cn_DR and set the STARTCn bit in I2Cn_CR.

After doing any of the actions above, clear all interrupt source bits in I2Cn_ST to ‘0’ to release SCL line. In case of A, move to step 7. In case of B, move to step 9 to handle STOP interrupt. In case of C, move to step 6 after transmitting the data in I2CDR, and if transfer direction bit is ‘1’, go to master receiver section.

9. This is the final step for master transmitter function of I2C, handling STOP interrupt. The STOP bit indicates that data transfer between master and slave is over. To clear I2Cn_ST, write “0xff” to I2Cn_ST. After this, I2C enters idle state.

19.5.2 Master receiver

To operate I2C in master receiver, follow the recommended steps below.

1. Enable I2C by setting I2CnEN bit in I2Cn_CR. This provides main clock to the peripheral.
2. Load SLA+R into the I2Cn_DR, where SLA is address of slave device and R is transfer direction from the viewpoint of the master. For master receiver, R is ‘1’. Note that I2Cn_DR is used for both address and data.
3. Configure baud rate by writing desired value to both I2Cn_SCLR and I2Cn_SCHR for the Low and High period of SCLn line.
4. Configure the I2Cn_SDHR to decide when SDAn changes value from falling edge of SCLn. If SDAn should change in the middle of SCLn LOW period, load half the value of I2Cn_SCLR to the I2Cn_SDHR.
5. Set the STARTCn bit in I2Cn_CR. This transmits a START condition. Also, configure how to handle interrupt and ACK signal. When the STARTCn bit is set, 8-bit data in I2Cn_DR is transmitted out according to the baud-rate.
6. This is ACK signal processing stage for address packet transmitted by master. When 7-bit address and 1-bit transfer direction is transmitted to target slave device, the master can know whether the slave acknowledged or not in the 9th high period of SCLn. If the master gains bus mastership, I2C generates GCALL interrupt regardless of receiving ACK from the slave device. When I2C loses bus mastership during arbitration process, the MLOSTn bit in I2Cn_ST is set, and I2C waits in idle state or can be operated as an addressed slave.

   To operate as a slave when the MLOSTn bit in I2Cn_ST is set, the ACKnEN bit in I2Cn_CR must be set, and the received 7-bit address must equal to the SLAn bits in I2Cn_SAR1/2. In this case, I2C operates as a slave transmitter or a slave receiver (go to appropriate section). In this stage, I2C holds the SCLn LOW. The reason for this is to decide whether I2C should continue serial transfer or stop communication. The following steps continue, assuming that I2C does not lose mastership during the first data transfer.

I2C (Master) can choose one of the following cases according to the reception of ACK signal from slave.

A. Master receives ACK signal from slave, so continues data transfer since slave can prepare and transmit more data to master. Configure ACKnEN bit in I2Cn_CR to decide whether I2C should Acknowledges the next data to be received or not.

B. Master stops data transfer since it receives no ACK signal from slave. In this case, set the STOPCn bit in I2Cn_CR.

C. Master transmits repeated START condition due to lack of ACK signal from slave. In this case, load SLA+R/W into the I2Cn_DR and set STARTCn bit in I2Cn_CR.

After doing any of the actions above, clear all interrupt source bits in I2Cn_ST to '0' to release SCLn line. In case of A, move to step 7. In case of B, move to step 9 to handle STOP interrupt. In case of C, move to step 6 after transmitting the data in I2Cn_DR, and if transfer direction bit is '0', go to master transmitter section.

7. 1-Byte of data is received.

8. This is ACK signal processing stage for data packet transmitted by slave. I2C holds the SCLn LOW. When 1-Byte of data is received completely, I2C generates TENDn interrupt.

   I2C can choose one of the following cases according to the RXACKn flag in I2Cn_ST.

   A. Master continues receiving data from slave. To do this, set ACKnEN bit in I2Cn_CR to acknowledge the next data to be received.

   B. Master wants to terminate data transfer when it receives next data by not generating ACK signal. This can be done by clearing ACKnEN bit in I2Cn_CR.

   C. Since no ACK signal is detected, master terminates data transfer. In this case, set the STOPCn bit in I2Cn_CR.

   D. No ACK signal is detected, and master transmits repeated START condition. In this case, load SLA+R/W into the I2Cn_DR and set the STARTCn bit in I2Cn_CR.

   After doing any of the actions above, clear all interrupt source bits in I2Cn_ST to '0' to release SCLn line. In case of A and B, move to step 7. In case of C, move to step 9 to handle STOP interrupt. In case of D, move to step 6 after transmitting the data in I2Cn_DR, and if transfer direction bit is '0', go to master transmitter section.

9. This is the final step for master receiver function of I2C, handling STOP interrupt. The STOP bit indicates that data transfer between master and slave is over. To clear I2Cn_ST, write "0xff" value to I2Cn_ST. After this, I2C enters idle state.

19.5.3 Slave transmitter

To operate I2C in slave transmitter, follow the recommended steps below.

1. If the operating clock (HCLK) of the system is slower than that of SCLn, load value 0x00 into I2Cn_SDHR to make SDAn change within one system clock period from the falling edge of SCLn. Note that the hold time of SDAn is calculated by SDAH x period of HCLK where SDAH is multiple of number of HCLK coming from I2Cn_SDHR. When the hold time of SDAn is longer than the period of HCLK, I2C (slave) cannot transmit serial data properly.

2. Enable I2C by setting I2CnIEN bit and I2CnEN bit in I2Cn_CR. This provides main clock to the peripheral.

3. When a START condition is detected, I2C receives one byte of data and compares it with SLAn bits in I2Cn_SAR1/2. If the GCALLnEN bit in I2Cn_SAR1/2 is enabled, I2C compares the received data with value 0x00, the general call address.

4. If the received address does not match SLAn bits in I2CnSAR, I2C enters idle state, i.e., waits for another START condition. Otherwise, if the address equals to SLAn bits and the ACKnEN bit is enabled, I2C generates SSELn interrupt and the SCLn line is held LOW. Note that even if the address matches SLAn bits, when the ACKnEN bit is disabled, I2C enters idle state. When SSELn interrupt occurs, load transmit data to I2Cn_DR and clear all interrupt source bits in I2Cn_ST to ‘0’ to release SCLn line.

5. 1-Byte of data is transmitted.

6. In this step, I2C generates TENDn interrupt and holds the SCLn line LOW regardless of receiving ACK signal from master. Slave can select one of the following cases.

   A. No ACK signal is detected and I2C waits STOP or repeated START condition.

   B. ACK signal from master is detected. Load data to transmit into I2Cn_DR.

   After doing any of the actions above, clear all interrupt source bits in I2Cn_ST to ‘0’ to release SCLn line. In case of A, move to step 7 to terminate communication. In case of B, move to step 5. In either case, a repeated START condition can be detected. For that case, move step 4.

7. This is the final step for slave transmitter function of I2C, handling STOP interrupt. The STOPCn bit indicates that data transfer between master and slave is over. To clear I2Cn_ST, write “0xff” to I2Cn_ST. After this, I2C enters idle state.

19.5.4 Slave receiver

To operate I2C in slave receiver, follow the recommended steps below.

1. If the operating clock (HCLK) of the system is slower than that of SCLn, load value 0x00 into I2Cn_SDHR to make SDAn change within one system clock period from the falling edge of SCLn. Note that the hold time of SDAn is calculated by SDAH x period of HCLK where SDAH is multiple of number of HCLK coming from I2Cn_SDHR. When the hold time of SDAn is longer than the period of HCLK, I2C (slave) cannot transmit serial data properly.

2. Enable I2C by setting I2CnIEN bit in I2Cn_CR. This provides main clock to the peripheral.

3. When a START condition is detected, I2C receives one byte of data and compares it with SLA bits in I2CSAR. If the GCALLnEN bit in I2Cn_SAR1/2 is enabled, I2C compares the received data with value 0x00, the general call address.

4. If the received address does not match SLAn bits in I2Cn_SAR1/2, I2C enters idle state i.e., waits for another START condition. Otherwise, if the address match SLAn bits and the ACKnEN bit is enabled, I2C generates SSELn interrupt and the SCLn line is held LOW. Note that even if the address equals to SLA bits, when the ACKnEN bit is disabled, I2C enters idle state. When SSELn interrupt occurs and I2C is ready to receive data, clear all interrupt source bits in I2Cn_ST to ‘0’ to release SCLn line.

5. 1-Byte of data is received.

6. In this step, I2C generates TENDn interrupt and holds the SCLn line LOW regardless of receiving ACK signal from master. Slave can select one of the following cases.

   A. No ACK signal is detected (ACKnEN=0) and I2C waits STOP or repeated START condition.

   B. ACK signal is detected (ACKnEN=1) and I2C can continue to receive data from master.

   After doing any of the actions above, clear all interrupt source bits in I2Cn_ST to ‘0’ to release SCLn line. In case of A, move to step 7 to terminate communication. In case of B, move to step 5. In either case, a repeated START condition can be detected. For that case, move to step 4.

7. This is the final step for slave receiver function of I2C, handling STOP interrupt. The STOPCn bit indicates that data transfer between master and slave is over. To clear I2Cn_ST, write “0xff” to I2Cn_ST. After this, I2C enters idle state.

20 SPI 0/1/2/3 Interface

SPI interface enables synchronous serial data transfer between external serial devices. It allows full-duplex communication using 4-wires (MOSIn, MISOn, SCKn, SSn).

It supports master and slave modes, and selects serial clock (SCKn) polarity. In addition, for the data transmission, it selects whether to transfer LSB first or MSB first.

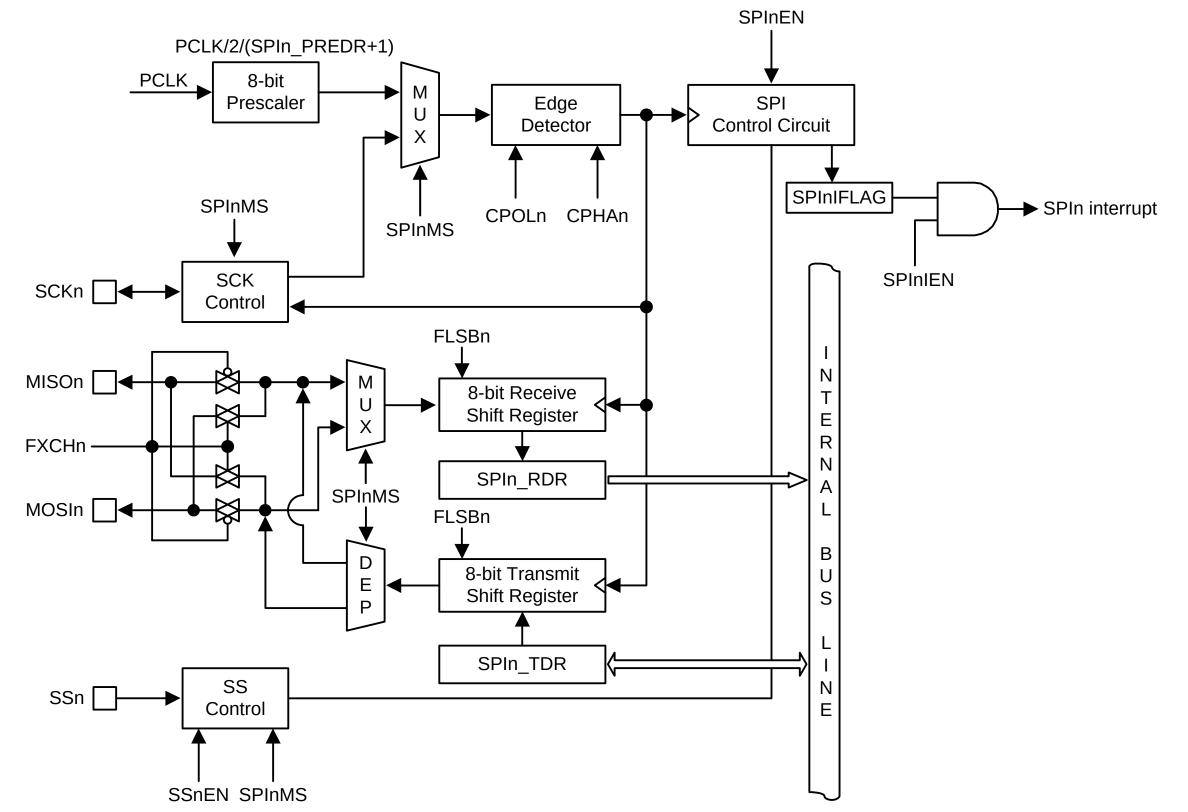
The SPI 0/1/2/3 of the A31L21x series features the followings:

- Master and slave modes supported
- Clock polarity selection
- Up to 16MHz data transmission
- Exchangeable MOSIn and MISOn functions

20.1 SPI 0/1/2/3 block diagram

Figure 106 shows a block diagram of the SPI block.

Figure 106. SPI Block Diagram (n = 0, 1, 2, and 3)



NOTES:

1. The clock frequency of the SCKn should be less than or equal to the PCLK/2.
2. Interrupts of the SPIn occur only when one byte transmission/reception finishes. So, the status (SPInMS and SSnHIGH) bits should be checked by s/w.
3. When DMA is used, at least 32 clocks of PCLK are required for the interval between data-receptions.

20.2 Pin description for SPI 0/1/2/3

Table 88. Pins and External Signals for SPI (n = 0, 1, 2, and 3)

Pin name	Type	Description
SSn	I/O	SPIn Slave select input/output
SCKn	I/O	SPIn Serial clock input/output
MOSIn	I/O	SPIn Serial data (Master output, Slave input)
MISOn	I/O	SPIn Serial data (Master input, Slave output)

20.3 Registers

Base address and register map of the SPI 0/1/2/3 are shown in Table 89 and Table 90.

Table 89. Base Address of SPI Interface

Name	Base address	Size	Description
SPI0	0x4000_5800	128	SPI0 Block
SPI1	0x4000_5880	128	SPI1 Block
SPI2	0x4000_5900	128	SPI2 Block
SPI3	0x4000_5980	128	SPI3 Block

Table 90. SPI Register Map (n = 0, 1, 2, and 3)

Name	Offset	Type	Description	Reset value
SPIn_CR	0x00	RW	SPIn Control Register	0x00000000
SPIn_SR	0x04	RW	SPIn Status Register	0x00000000
SPIn_RDR	0x08	RO	SPIn Receive Data Register	0x00000000
SPIn_TDR	0x0C	RW	SPIn Transmit Data Register	0x00000000
SPIn_PREDR	0x10	RW	SPIn Prescaler Data Register	0x000003FF

20.3.1 SPIn_CR: SPIn control register

SPI module should be configured properly before running.

SPIn_CR register is 32-bit size and accessible in 32/16/8-bit. (n = 0, 1, 2, and 3)

SPI0_CR=0x4000_5800, SPI1_CR=0x4000_5880
SPI2_CR=0x4000_5900, SPI3_CR=0x4000_5980

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
Reserved	**SPInEN**	**FLSBn**	**SPInMS**	**Reserved**	**SPInIEN**	**Reserved**	**CPOLn**	**CPHAn**
0x000000	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
-	**RW**	**RW**	**RW**	**-**	**RW**	**-**	**RW**	**RW**

7	SPInEN	SPIn Operation Control. **NOTE**: This bit should be set to “1b” after setting the related registers.	
		0	Disable SPIn operation.
		1	Enable SPIn operation.
6	FLSBn	Data Transmission sequence selection.	
		0	MSB first.
		1	LSB first.
5	SPInMS	Master/Slave Selection.	
		0	Slave mode.
		1	Master mode.
3	SPInIEN	SPIn Interrupt Enable.	
		0	Disable SPIn interrupt.
		1	Enable SPIn interrupt.
1	CPOLn	Selects the clock polarity of SCK.	
		0	SCK to 0 when idle.
		1	SCK to 1 when idle.
0	CPHAn	The CPOLn and this bit determine if data are sampled on the leading or the trailing edge of SCK.	

CPOLn	CPHAn	Leading edge	Trailing edge
0	0	Sample (Rising)	Setup (Falling)
0	1	Setup (Rising)	Sample (Falling)
1	0	Sample (Falling)	Setup (Rising)
1	1	Setup (Falling)	Sample (Rising)

20.3.2 SPIn_SR: SPIn status register

SPIn_SR register is 32-bit size and accessible in 32/16/8-bit. (n = 0, 1, 2, and 3)

SPI0_SR=0x4000_5804, SPI1_SR=0x4000_5884
SPI2_SR=0x4000_5904, SPI3_SR=0x4000_5984

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
Reserved			SPInIFLAG	Reserved		SSnHIGH	Reserved		FXCHn	SSnEN
0x000000			0	0	0	0	0	0	0	0
-			RW	-	-	RW	-	-	RW	RW

7	SPInIFLAG	SPIn Interrupt Flag.	
		0	No request occurred.
		1	Request occurred. This bit is cleared to '0' when write '1'.
4	SSnHIGH	This bit is set when the SSn pin goes high level during the pin is the corresponding function.	
		0	No effect when '0' is written.
		1	The SSn pin has gone from low level to high. This bit is cleared to '0' when write '1'.
1	FXCHn	SPIn Pin Function Exchange Control.	
		0	No effect.
		1	Exchange MOSIn and MISOn function.
0	SSnEN	SSn Pin Operation Control.	
		0	Disable SSn pin operation.
		1	Enable SSn pin operation. The corresponding SSn Pin should be configured to the alternative function.

20.3.3 SPIn_RDR: SPIn receive data register

SPIn_RDR register is 32-bit size and accessible in 32/16/8-bit. (n = 0, 1, 2, and 3)

SPI0_RDR=0x4000_5808, SPI1_RDR=0x4000_5888
SPI0_RDR=0x4000_5908, SPI1_RDR=0x4000_5988

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved			RDATA
0x000000			0x00
-			RO

7 0	RDATA	SPIn Receive Data.

20.3.4 SPIn_TDR: SPIn transmit data register

SPIn_TDR register is 32-bit size and accessible in 32/16/8-bit. (n = 0, 1, 2, and 3)

SPI0_TDR=0x4000_580C, SPI1_TDR=0x4000_588C
SPI2_TDR=0x4000_590C, SPI3_TDR=0x4000_598C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																								TDATA							
0x000000																								0x00							
-																								RW							

7 0	TDATA	SPIn Transmit Data. When it is written a byte to this data register, the SPIn will start. **NOTE**: The data to be transmitted should be written after all control registers are set.

20.3.5 SPIn_PREDR: SPIn prescaler data register

SPIn_PREDR register is 32-bit size and accessible in 32/16/8-bit. (n = 0, 1, 2, and 3)

SPI0_PREDR=0x4000_5810, SPI1_PREDR=0x4000_5890
SPI2_PREDR=0x4000_5910, SPI3_PREDR=0x4000_5990

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																						PRED									
0x000000																						0x3FF									
-																						RW									

9 0	PRED	The value in this register is used to generate an SCK clock. SCKn clock: PCLK/2/(PRED[9:0] +1). The SCKn clock must be less than or equal to 16MHz. The range is 0x00 to 0x3FF.

20.4 Functional description

When SPIn block is enabled (SPInEN = ’1’), the slave select (SSn) pin becomes active LOW input in slave mode operation if SSnEN bit is set to ‘1’. The SSn function is not automatically controlled in master mode operation even if SSnEN bit is set to ‘1’. (n = 0, 1, 2, and 3)

20.4.1 SPI clock formats and timing

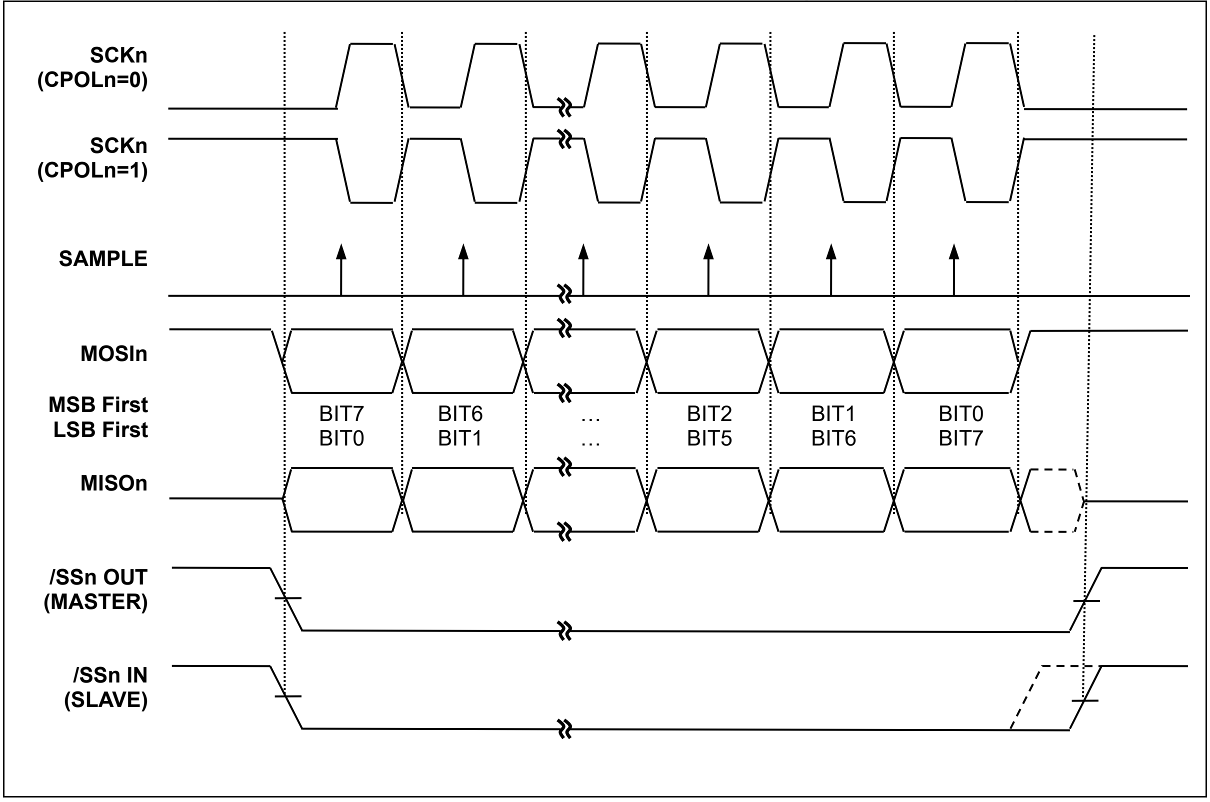
To accommodate a wide variety of synchronous serial peripherals from different manufacturers, the SPIn has a clock polarity bit (CPOLn) and a clock phase control bit (CPHAn) to select one of four clock formats for data transfers. CPOLn selectively inserts an inverter in series with the clock. CPHAn chooses between two different clock phase relationships between the clock and data.

Table 91 shows the four combinations of CPOLn and CPHAn for SPIn. (n = 0, 1, 2, and 3)

Table 91. CPOL Functionality (n = 0, 1, 2, and 3)

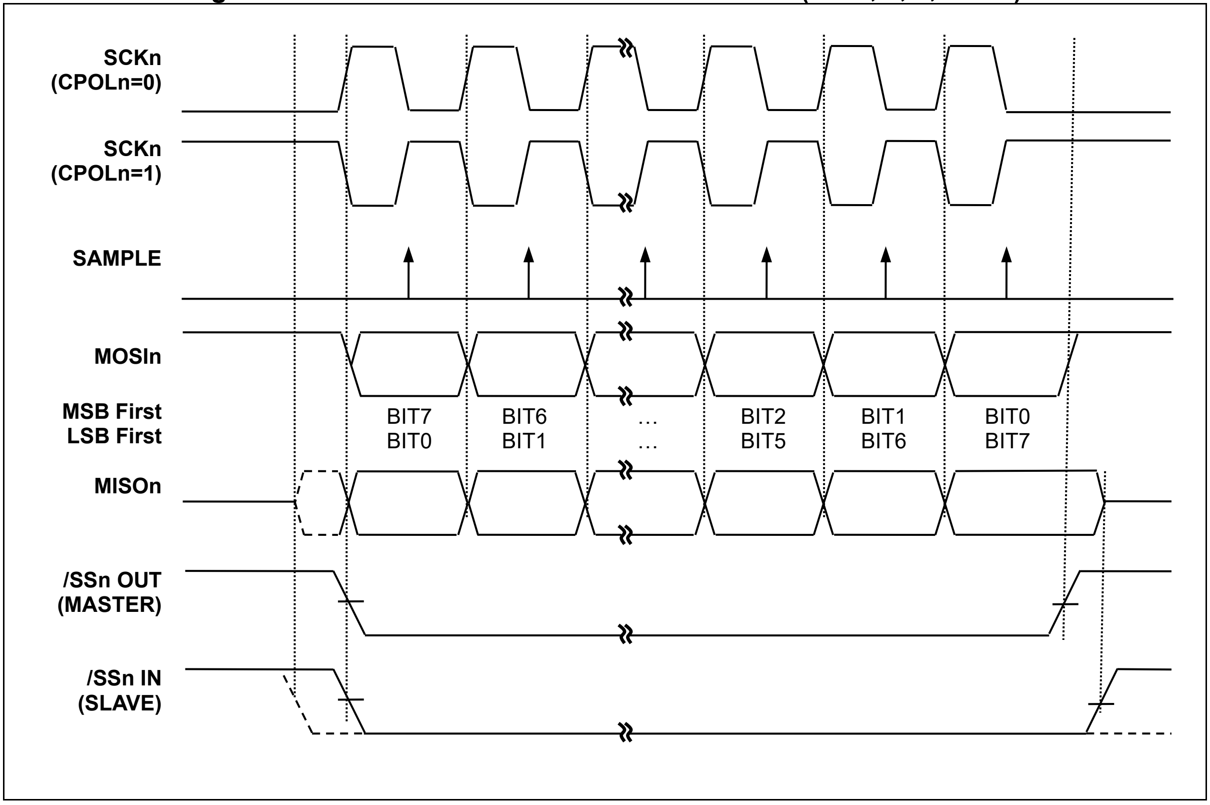
SPIn Mode	CPOLn	CPHAn	Leading Edge	Trailing Edge
0	0	0	Sample (Rising)	Setup (Falling)
1	0	1	Setup (Rising)	Sample (Falling)
2	1	0	Sample (Falling)	Setup (Rising)
3	1	1	Setup (Falling)	Sample (Rising)

Figure 107. SPIn Clock Formats when CPHAn=0 (n = 0, 1, 2, and 3)



When CPHAn=0, the slave begins to drive its MISOn output with the first data bit value when SSn goes to active low. The first SCKn edge causes both the master and the slave to sample the data bit value on their MISOn and MOSIn inputs, respectively. At the second SCKn edge, the SPIn shifts the second data bit value out to the MOSIn and MISOn outputs of the master and slave, respectively. (n = 0, 1, 2, and 3)

Figure 108. SPIn Clock Formats when CPHAn=1 (n = 0, 1, 2, and 3)



When CPHAn=1, the slave begins to drive its MISOn output when SSn goes active low, but the data is not defined until the first SCKn edge.

The first SCKn edge shifts the first bit of data from the shifter onto the MOSIn output of the master and the MISOn output of the slave. The next SCKn edge causes both the master and slave to sample the data bit value on their MISOn and MOSIn inputs, respectively. At the third SCKn edge, the USART shifts the second data bit value out to the MOSIn and MISOn output of the master and slave respectively.

When CPHAn=1, the slave's SSn input is not required to go to its inactive high level between transfers.

21 Smartcard Interface 0

A smartcard interface block of the A31L21x series is based on ISO/IEC 7816-3 standard. It supports UART mode to communicate with others.

This smartcard interface block has thirteen registers such as control registers (SCn_CR1, SCn_CR2, SCn_CR3), receive data register (SCn_RDR), transmit data register (SCn_TDR), baud-rate data register (SCn_BDR), and so on.

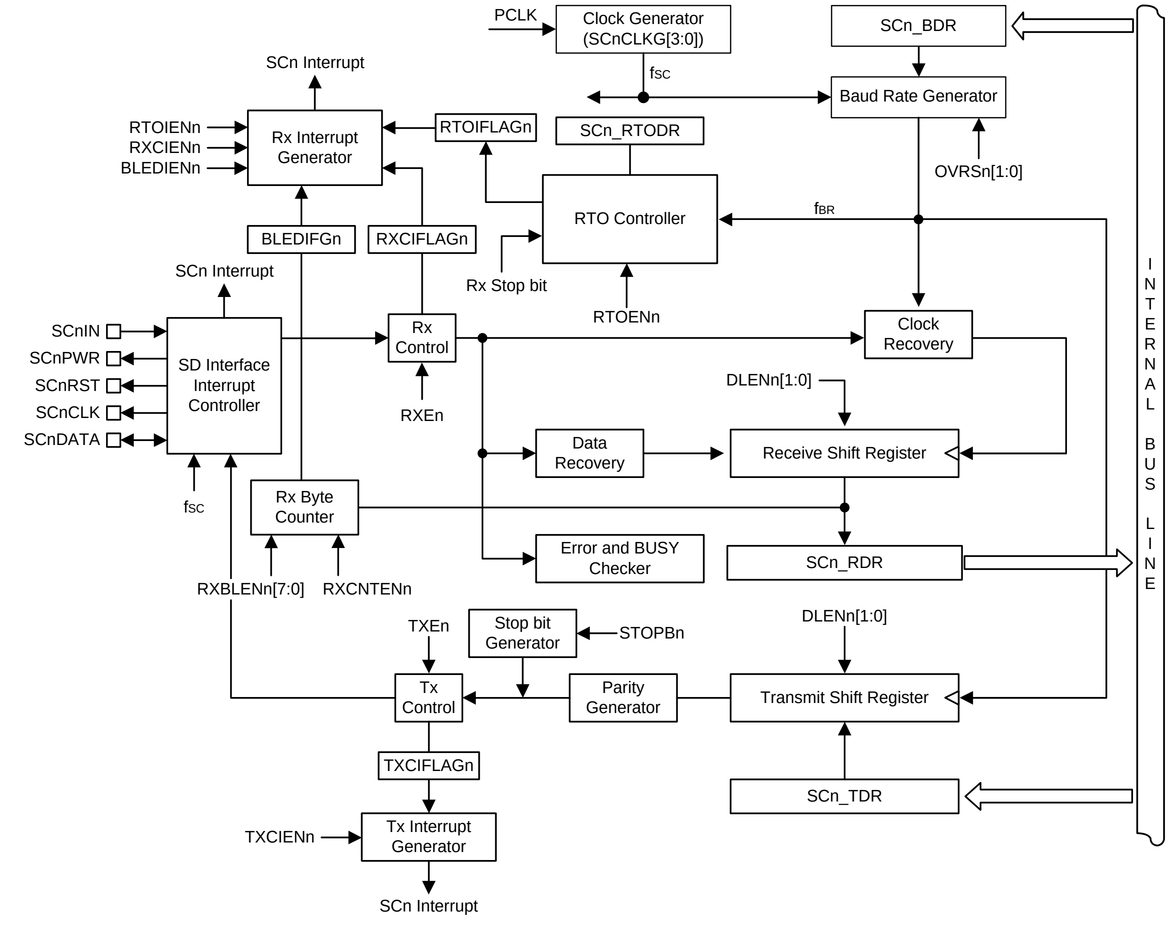
The smartcard interface 0 of the A31L21x series features the followings:

- ISO-7816-3 T = 0, T = 1 compatible
- DMA transfer supported
- Programmable guard time
- Auto activation sequence supported
- Auto warm reset sequence supported
- Auto deactivation sequence supported
- Auto convention detection sequence supported
- Baud rate compensation function
- UART mode selection
- Full duplex asynchronous operation
- Programmable baud-rate generation
- Even, odd, or no parity bit generation and detection selection
- 1 or 2 Stop bit generation selection
- Programmable data delay time after stop bit

21.1 Smartcard interface 0 block diagram

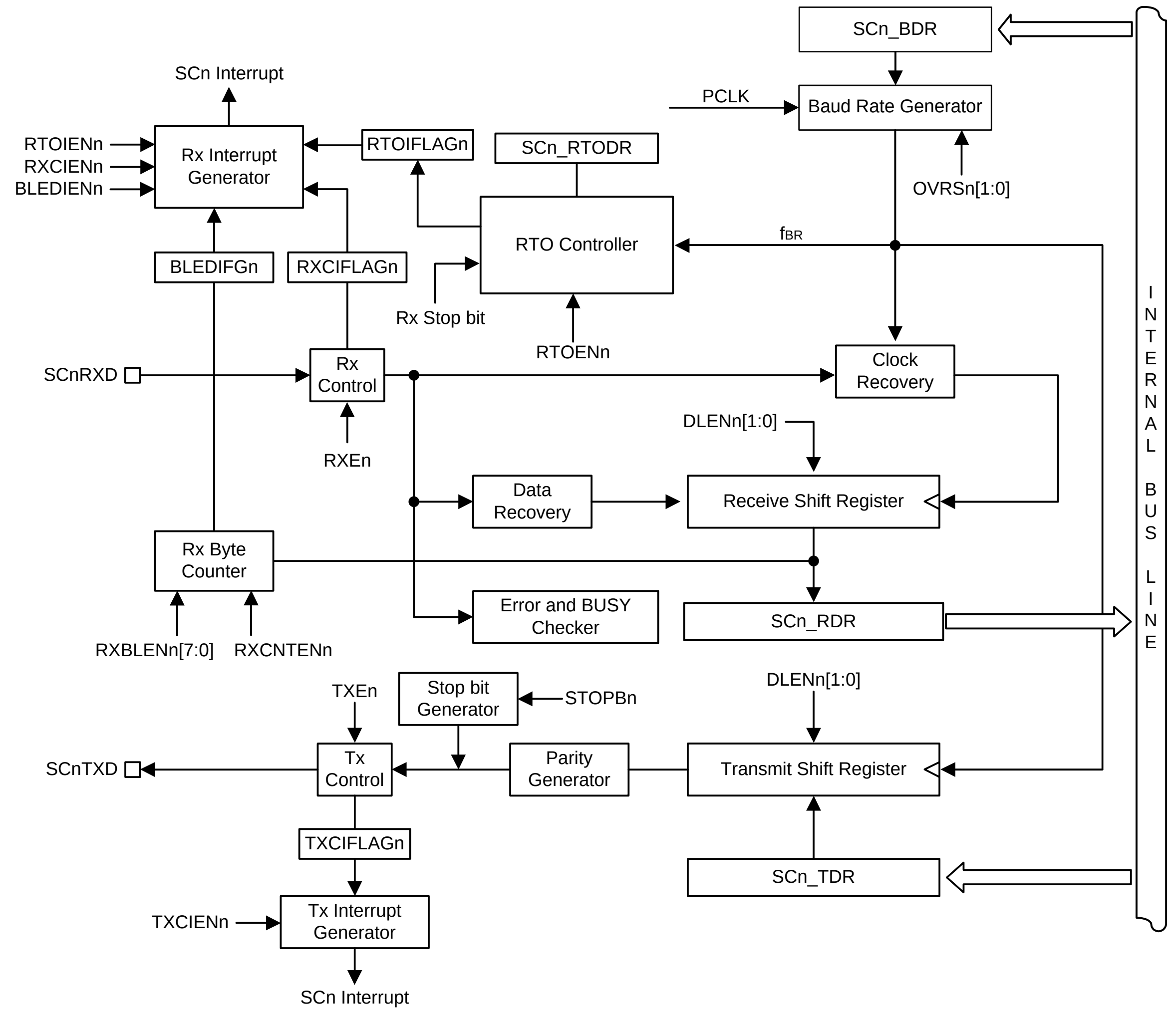
Figure 109 and Figure 110 shows a block diagram of the Smartcard interface block.

Figure 109. Smartcard Interface Mode Block Diagram (n = 0)



NOTE: After checking that the TXCIFLAGn bit is set to '1', write the data to be sent to the SCn_TDR register.

Figure 110. UART Mode Block Diagram (n = 0)



NOTE: After checking that the TXCIFLAGn bit is set to '1', write the data to be sent to the SCn_TDR register.

21.2 Pin description for Smartcard interface 0

Table 92. Pins and External Signals for Smartcard Interface (n = 0)

Pin name	Type	Description
SCnPWR	O	Smartcard power control output
SCnDATA	I/O	Smartcard data input/output
SCnRST	O	Smartcard reset output
SCnCLK	O	Smartcard clock output
SCnIN	I	Smartcard detection input
SCnTXD	O	SCn’s UART data output
SCnRXD	I	SCn’s UART data input

21.3 Registers

Base address and register map of the smartcard interface block are shown in Table 93 and Table 94.

Table 93. Base Address of Smartcard Interface Blocks

Name	Base address	Size	Description
SC0	0x4000_5300	128	SC0 Block (Smartcard Interface + UART mode)

Table 94. Smartcard Interface Register Map (n = 0)

Name	Offset	Type	Description	Reset Value
SCn_CR1	0x00	RW	SCn Control Register 1	0x00000000
SCn_CR2	0x04	RW	SCn Control Register 2	0x00000000
SCn_CR3	0x08	RW	SCn Control Register 3	0x00000000
–	–	–	Reserved	–
SCn_IER	0x10	RW	SCn Interrupt Enable Register	0x00000000
SCn_IFSR	0x14	RW	SCn Interrupt Flag and Status Register	0x00000004
SCn_RDR	0x18	RO	SCn Receive Data Register	0x00000000
SCn_TDR	0x1C	RW	SCn Transmit Data Register	0x00000000
SCn_BDR	0x20	RW	SCn Baud Rate Data Register	0x0000FFFF
SCn_BCMP	0x24	RW	SCn Baud Rate Compensation Register	0x00000000
SCn_RTODR	0x28	RW	SCn Receive Time Out Data Register	0x0000FFFF
SCn_EGTR	0x2C	RW	SCn Tx Extra Guard Time Register	0x00000000
SCn_T3DR	0x30	RW	SCn T3 Duration Data Register	0x00000000
SCn_T4DR	0x34	RW	SCn T4 Duration Data Register	0x00000000

21.3.1 SCn_CR1: SCn control register 1

Smartcard module should be configured properly before running.

SCn_CR1 register is 32-bit size and accessible in 32/16/8-bit. (n = 0)

SC0_CR1=0x4000_5300

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																SCnMD	PENn	PSELn	Reserved		DLENn		Reserved	STOPBn	Reserved		OVRSn	TXEn	RXEn	RTOENn	SCInEN
0x0000																0	0	0	00		00		0	0	00		0	0	0	0	0
-																RW	RW	RW	-		RW		RW	RW	-		RW	RW	RW	RW	RW

15	SCnMD	Smartcard Interface Mode Selection bit	
		0	Smartcard interface mode (SCnPWR/RST/CLK/DATA/IN)
		1	UART mode (SCnRXD/TXD)
14	PENn	Parity Enable bit	
		0	Disable parity bit generation and detection
		1	Enable parity bit generation and detection
13	PSELn	Parity Selection bit	
		0	Odd parity (Odd number of logic ‘1’)
		1	Even parity (Even number of logic ‘1’)
10 9	DLENn	Data Length Selection bits	
		00	5bit. Start,D0,D1,D2,D3,D4,(Parity),Stop1,(Stop2)
		01	6bit. Start,D0,D1,D2,D3,D4,D5,(Parity),Stop1,(Stop2)
		10	7bit. Start,D0,D1,D2,D3,D4,D5,D6,(Parity),Stop1,(Stop2)
		11	8bit. Start,D0,D1,D2,D3,D4,D5,D6,D7,(Parity),Stop1,(Stop2)
7	STOPBn	Stop bit	
		0	1 Stop bit
		1	2 Stop bits
4	OVRSn	Oversampling Selection bit	
		0	16 oversampling
		1	8 oversampling
3	TXEn	Enable the Transmitter unit	
		0	Transmitter is disabled
		1	Transmitter is enabled
2	RXEn	Enable the Receiver unit	
		0	Receiver is disabled
		1	Receiver is enabled
		NOTE: Write “0b” to this RXEn bit to disable Rx before Tx start. Otherwise, an Rx interrupt can occur.	
1	RTOENn	Receive Time Out Function Enable bit. This function is to count time with baud rate unit from the leading edge of a start bit to a new start bit. The receive time out controller counts down from the value of SCn_RTODR register every start bit and set this bit. The RTOIFLAG bit is set to “1b” at the counter underflow.	
		0	Disable receive time out function
		1	Enable receive time out function
0	SCInEN	Smartcard Interface Block Enable bit. This bit can be cleared to “0b” during the corresponding TXEn and RXEn bits are all “0b”.	
		0	Disable SCn block
		1	Enable SCn block
		NOTE: If this bit is cleared to ‘0’, the current operations are discarded, the configuration is kept, and all the status flags are set to reset values.	

21.3.2 SCn_CR2: SCn control register 2

Smartcard module should be configured properly before running. This register is used only for smartcard interface mode. SCn_CR2 register is 32-bit size and accessible in 32/16/8-bit. (n = 0)

SC0_CR2=0x4000_5304

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15	14	13	12 11 10	9	8	7	6	5	4	3 2 1 0
Reserved	**ACTENn**	**WRENn**	**DACTENn**	**Reserved**	**SCnINST**	**SCnPWRLV**	**SCnRSTLV**	**SCnDATALV**	**SCnCLKLV**	**SCnCLKEN**	**SCnCLKG**
0x00000	**0**	**0**	**0**	**000**	**0**	**0**	**0**	**0**	**0**	**0**	**0000**
-	**RW**	**RW**	**RW**	-	**RO**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**

15	ACTENn	Auto Activation Enable bit	
		0	No effect
		1	Enable activation and cold reset (This bit is automatically cleared after operation)
14	WRENn	Auto Warm Reset Enable bit	
		0	No effect
		1	Enable warm reset (This bit is automatically cleared after operation)
13	DACTENn	Auto Deactivation Enable bit	
		0	No effect
		1	Enable deactivation (This bit is automatically cleared after operation)
9	SCnINST	SCnIN Pin Status bit	
		0	SCnIN pin state at low level
		1	SCnIN pin state at high level
8	SCnPWRLV	SCnPWR Pin Level Setting bit	
		0	SCnPWR pin to low level
		1	SCnPWR pin to high level
		NOTE:	When Activation/"Warm Reset"/Deactivation auto mode, this bit is automatically configured. So don't fill this bit on auto mode.
7	SCnRSTLV	SCnRST Pin Level Setting bit	
		0	SCnRST pin to low level
		1	SCnRST pin to high level
		NOTE:	When Activation/"Warm Reset"/Deactivation auto mode, this bit is automatically configured. So don't fill this bit on auto mode.
6	SCnDATALV	SCnDATA Pin Level Setting bit. The SCDATA pin must be configured to open-drain with an external pull-up resistor for a bidirectional line.	
		0	SCnDATA pin to low level
		1	The SCnDATA pin is high level with an external pull-up resistor and reception mode.
		NOTE:	When Activation/"Warm Reset"/Deactivation auto mode, this bit is automatically configured. So don't fill this bit on auto mode.
5	SCnCLKLV	SCnCLK Pin Level Setting bit	
		0	SCnCLK pin to low level on clock generation disable
		1	SCnCLK pin to high level on clock generation disable
4	SCnCLKEN	Smartcard Clock Generation Enable bit. The generated clock output also maintains a 50:50 duty cycle even when enable/disable.	
		0	Disable smartcard clock generation
		1	Enable smartcard clock generation
		NOTE:	When Activation/"Warm Reset"/Deactivation auto mode, this bit is automatically configured. So don't fill this bit on auto mode.
3 0	SCnCLKG	This bit-field is used to generate smartcard clock. When SCnCLKG[3:0] = 0, Smartcard clock (fsc): PCLK/1 with 1/2 duty When SCnCLKG[3:0] = 1 to 15, Smartcard clock (fsc): PCLK/(value x 2) with 1/2 duty	

NOTE: The ACTENn, WRENn, and DACTENn bits won't be set to 2 bits or more at the same time.

21.3.3 SCn_CR3: SCn control register 3

Smartcard module should be configured properly before running. This register is used only for smartcard interface mode.

SCn_CR3 register is 32-bit size and accessible in 32/16/8-bit. (n = 0)

SC0_CR3=0x4000_5308

31 30 29 28 27 26 25 24	23	22	21	20	19 18 17	16	15	14 13	12 11 10 9	8	7 6 5 4 3 2 1 0
Reserved	**ACONDETn**	**CONSELn**	**Reserved**	**RETRYENn**	**RETRYn**	**DLYRETRYn**	**Reserved**	**SCnINPOL**	**Reserved**	**RXCNTENn**	**RXBLENn**
0x000000	**0**	**0**	**0**	**0**	**000**	**0**	**0**	**00**	**0000**	**0**	**0x00**
-	**RW**	**RW**	-	**RW**	**RW**	**RW**	-	**RW**	-	**RW**	**RW**

23	ACONDETn	Auto Convention Detection bit 0 No effect 1 Auto convention detection (This bit is automatically cleared after operation) **NOTES**: 1. This bit should be set before ATR (Answer to Reset) 2. The even/enable parity (PENn = 1, PSELn = 1) and 8-bits data length should be configured before setting this bit. 3. If the received value is invalid for TS, a CONERIFGn bit will be set. 4. If the received value is valid for TS, a CONEDIFGn bit will be set. 5. This bit is effective only during the smartcard interface mode.
22	CONSELn	Convention Selection bit 0 Direct convention (LSB-1st shift out, 0: Low level, 1: High level) 1 Inverse convention (MSB-1st shift out, 0: High level, 1: low level). **NOTE**: When auto convention detection mode, this bit is automatically configured after TS receive.
20	RETRYENn	Tx/Rx Error Signal Generation/Detection Retry Enable bit 0 Disable error signal generation/detection and retry Rx/Tx 1 Enable error signal generation/detection and retry Rx/Tx **NOTES**: 1. If the error signal repeats as the set number of retries, a TRYERIFGn bit is set, this bit is cleared to stop error signal generation/detection, it stops transmit (Tx), and the last received byte (Rx) is saved to the SCn_RDR register. 2. When receive mode, the received byte with error isn’t saved except the last. 3. The TXCIFLAGn or RXCIFLAGn bit may be set.
19 17	RETRYn[2:0]	The number of retry. RETRYn[2:0]+0
16	DLYRETRYn	Delay Time Before Retry Selection bit 0 2.5 etu delay before re-transmit byte 1 2.5 etu + “extra guard time” delay before re-transmit byte
14 13	SCnINPOL	SCnIN Pin Input Polarity Selection bits. 00 Smartcard Insert/Removal on falling edge 01 Smartcard Insert/Removal on rising edge 10 Smartcard Insert/Removal on both of falling and rising edge 11 Reserved **NOTE**: This bit is effective only during the smartcard interface mode.
8	RXCNTENn	Received byte Count Enable bit. This bit should be set before the start of a block reception. 0 No effect 1 Received block length counts every Rx. (This bit is automatically cleared after the BLEDIFGn bit is set.)
7 0	RXBLENn	Received Block Length bits. This bit-field can be used for the block length in smartcard T=1. It is “information characters” + “Epilogue field (1-LEC/2-CRC)” – 1. These bits

should be written by s/w before the RXCNTEN bit is set.
0 = 0 information characters + LEC
1 = 0 information characters + CRC
N except 0 and 1 = (N-1) information characters + CRC, total (N+1) characters.

NOTE: A BLEDIFGn bit will be set when 4th byte has received after the RXCNTENn bit is set to “1b”. The contents in the 3rd received byte should be moved to this bit-field by S/W. The BLEDIFGn bit is set when the number of received bytes from the start of the block is equal or greater than RXBLENn[7:0] + 4.

21.3.4 SCn_IER: SCn interrupt enable register

SCn_IER register is 32-bit size and accessible in 32/16/8-bit. (n = 0)

SC0_IER=0x4000_5310

31 30 29 28 27 26 25 24	23	22	21	20	19	18	17	16	15 14 13 12 11 10 9 8 7	6	5 4 3	2	1	0
Reserved	RSTAIENn	SERIENn	SEDIENn	CONERIENn	CONEDIENn	TRYERIENn	SCINIENn	BLEDIENn	Reserved	RTOIENn	Reserved	TXCIENn	Reserved	RXCIENn
0x00	0	0	0	0	0	0	0	0	0x000	0	000	0	0	0
-	RW	RW	RW	RW	RW	RW	RW	RW	-	RW	-	RW	-	RW

23	RSTAIENn	Reset Assertion Interrupt Enable bit	
		0	Disable reset assertion interrupt
		1	Enable reset assertion interrupt
22	SERIENn	Sequence Error Interrupt Enable bit	
		0	Disable sequence error interrupt
		1	Enable sequence error interrupt
21	SEDIENn	Sequence End Interrupt Enable bit	
		0	Disable sequence end interrupt
		1	Enable sequence end interrupt
20	CONERIENn	Convention Detection Error Interrupt Enable bit	
		0	Disable convention detection error interrupt
		1	Enable convention detection error interrupt
19	CONEDIENn	Convention Detection End Interrupt Enable bit	
		0	Disable convention detection end interrupt
		1	Enable convention detection end interrupt
18	TRYERIENn	Transmit Retry Error Interrupt Enable bit	
		0	Disable transmit retry error interrupt
		1	Enable transmit retry error interrupt
17	SCINIENn	SCnIN Pin Valid Edge Interrupt Enable bit	
		0	Disable SCnIN pin valid edge interrupt
		1	Enable SCnIN pin valid edge interrupt
16	BLEDIENn	Block Length Count End Interrupt Enable bit	
		0	Disable block length count end interrupt
		1	Enable block length count end interrupt
6	RTOIENn	Receive Time Out Interrupt Enable bit	
		0	Disable receive time out interrupt
		1	Enable receive time out interrupt
2	TXCIENn	Transmit Complete Interrupt Enable bit	
		0	Disable transmit complete interrupt
		1	Enable transmit complete interrupt
0	RXCIENn	Receive Data Register Not Empty Interrupt Enable bit.	
		0	Disable receive data not empty interrupt
		1	Enable receive data not empty interrupt

21.3.5 SCn_IFSR: SCn interrupt status register

SCn_IFSR register is 32-bit size and accessible in 32/16/8-bit. (n = 0)

SC0_IFSR=0x4000_5314

31 30 29 28 27 26 25 24	23	22	21	20	19	18	17	16	15	14	13	12	11 10 9 8 7	6	5 4 3	2	1	0
Reserved	**RSTAIFGn**	**SERIFGn**	**SEDIFGn**	**CONERIFGn**	**CONEDIFGn**	**TRYERIFGn**	**SCINIFGn**	**BLEDIFGn**	**DORn**	**FEn**	**PEn**	**RXBUSYn**	**Reserved**	**RTOIFLAGn**	**Reserved**	**TXCIFLAGn**	**Reserved**	**RXCIFLAGn**
0x00	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0x00**	**0**	**000**	**1**	**0**	**0**
-	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RO**	-	**RW**	-	**RW**	-	**RW**

23	RSTAIFGn	Reset Assertion Interrupt Flag. This bit is set to “1b” when the reset signal goes high level during “auto activation and cold reset” and “auto warm reset”.	
		0	No request occurred
		1	Request occurred. This bit is cleared to ‘0’ when write ‘1’.
22	SERIFGn	Sequence Error Interrupt Flag. This bit is set to “1b” when the sequence is invalid during “auto activation and cold reset”, “auto warm reset”, and “auto deactivation”.	
		0	No request occurred
		1	Request occurred. This bit is cleared to ‘0’ when write ‘1’.
21	SEDIFGn	Sequence End Interrupt Flag. This bit is set to “1b” when the sequence is finish and valid during “auto activation and cold reset”, “auto warm reset”, and “auto deactivation”.	
		0	No request occurred
		1	Request occurred. This bit is cleared to ‘0’ when write ‘1’.
20	CONERIFGn	Convention Detection Error Interrupt Flag. This bit is set to “1b” when the received value on ATR is invalid as an initial character TS.	
		0	No request occurred
		1	Request occurred. This bit is cleared to ‘0’ when write ‘1’.
19	CONEDIFGn	Convention Detection End Interrupt Flag. This bit is set to “1b” when the received value on ATR is valid as an initial character TS.	
		0	No request occurred
		1	Request occurred. This bit is cleared to ‘0’ when write ‘1’.
18	TRYERIFGn	Transmit Retry Error Interrupt Flag. This bit is set to “1b” when the parity error signal on Tx/Rx repeats as the set number of retries by RETRYn[2:0] bits of SCn_CR3 register.	
		0	No request occurred
		1	Request occurred, This bit is cleared to ‘0’ when write ‘1’.
17	SCINIFGn	SCnIN Pin Valid Edge Interrupt Flag. This bit is set to “1b” when it occurs valid edge set by SCnINPOL[1:0] bits of the SCn_CR3 register.	
		0	No request occurred
		1	Request occurred. This bit is cleared to ‘0’ when write ‘1’.
16	BLEDIFGn	Block Length Count End Interrupt Flag. This bit is set to “1b” when the number of received bytes from the start of the block is equal or greater than RXBLENn[7:0] + 4.	
		0	No request occurred
		1	Request occurred. This bit is cleared to ‘0’ when write ‘1’.
15	DORn	Data Overrun bit. This bit is set when the receive shift register is transferred to the SCn_RDR register while the RXCIFLAGn=1. The data of the shift register are ignored.	
		0	No data overrun
		1	Data overrun detected. This bit is cleared to ‘0’ when write ‘1’.
14	FEn	Frame Error bit. This bit is set when the received data have not a valid stop bit (That is, the stop bit following the last data bit or parity bit is detected as “0b”).	
		0	No frame error
		1	Frame error detected
13	PEn	Parity Error bit. This bit is set when the received data has a parity error on parity enable.	
		0	No parity error
		1	Parity error detected

12	RXBUSYn		SCnRXD Line Busy bit. This bit is set at a start bit and reset at the end of the reception.
		0	Receive line (SCnRXD) is not busy
		1	Reception on going
6	RTOIFLAGn		Receive Time Out Interrupt Flag. This bit is set to “1b” at the counter underflow of the receive time out controller.
		0	No request occurred
		1	Request occurred, This bit is cleared to ‘0’ when write ‘1’.
2	TXCIFLAGn		Transmit Complete Interrupt Flag. This flag is set to “1b” when the data in the transmit shift register has been shifted out.
		0	No request occurred
		1	The data in the transmit shift register are shifted out completely. This bit is cleared to ‘0’ when write ‘1’.
0	RXCIFLAGn		Receive Data Register Not Empty Interrupt Flag. This bit is set to “1b” when the data in the receive shift register has been transferred to the SCn_RDR register. The bit is cleared by a read to the SCn_RDR register.
		0	No request occurred
		1	There is data in the receive data register. This bit is cleared to ‘0’ when write ‘1’.

21.3.6 SCn_RDR: SCn receive data register

SCn_RDR register is 32-bit size and accessible in 32/16/8-bit. (n = 0)

SC0_RDR=0x4000_5318

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9	8	7 6 5 4 3 2 1 0
Reserved	**PARB**	**RDATA**
0x000000	**0**	**0x00**
-	**RW**	**RO**

8	PARB	Parity bit. This is a received parity bit.
7 0	RDATA	Receive Data bits. A receive shift register is moved to this register after stop bit.

21.3.7 SCn_TDR: SCn transmit data register

SCn_TDR register is 32-bit size and accessible in 32/16/8-bit. (n = 0)

SC0_TDR=0x4000_531C

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved	**TDATA**
0x000000	**0x00**
-	**RW**

7 0	TDATA	Transmit Data bits. This register is moved to the transmit shift register after a previous character is completely shifted out.

21.3.8 SCn_BDR: SCn baud rate generation register

SCn_BDR register is 32-bit size and accessible in 32/16/8-bit. (n = 0)

SC0_BDR=0x4000_5320

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		**BDATA**	
0x0000		**0xFFFF**	
-		**RW**	

15 0	BDATA	These bits are used to generate baud rate. BDATA[15:0] range: 0x0 to 0xFFFF 16 oversampling: — Baud Rate = f_{SC}/{16 x (BDATA[15:0] +1)}, On smart card interface mode — Baud Rate = f_{PCLK}/{16 x (BDATA[15:0] +1)}, On UART mode 8 oversampling: — Baud Rate = f_{SC}/{8 x (BDATA[15:0] +1)}, On smart card interface mode — Baud Rate = f_{PCLK}/{8 x (BDATA[15:0] +1)}, On uart mode

21.3.9 SCn_BCMP: SCn baud rate compensation register

SCn_BCMP register is 32-bit size and accessible in 32/16/8-bit. (n = 0)

SC0_BCMP=0x4000_5324

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15	14 13 12 11 10 9	8	7	6	5	4	3	2	1	0
Reserved		**BCMPS**	**Reserved**	**BCMP8**	**BCMP7**	**BCMP6**	**BCMP5**	**BCMP4**	**BCMP3**	**BCMP2**	**BCMP1**	**BCMP0**
0x0000		**0**	**0x00**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
-		**RW**	-	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**	**RW**

15	BCMPS	Baud Rate Compensation Sign bit
		0Plus 1 clock for compensation
		1Minus 1 clock for compensation
8 0	BCMPx	Baud Rate Compensation bits. x: 0 to 8
		0No compensation
		11 clock compensation every sampling with sign bit (BCMPS)
		NOTE: The BCMP8 bit is for parity bit and the BCMP[7:0] are for Data[7:0].

21.3.10 SCn_RTODR: SCn receive time out data register

SCn_RTODR register is 32-bit size and accessible in 32/16/8-bit. (n = 0)

SC0_RTODR=0x4000_5328

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved	**RTOD**		
0x00	**0x00FFFF**		
-	**RW**		

23 0	RTOD	SCn Receive Time Out Data bits

21.3.11 SCn_EGTR: SCn transmit extra guard time register

SCn_EGTR register is 32-bit size and accessible in 32/16/8-bit. (n = 0)

SC0_EGTR=0x4000_532C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved			**EGT**
0x000000			**0x00**
-			**RW**

7 0	EGT	SCn Tx Extra Guard Time bits. This register is used for transmit extra guard time between the last stop bit and the next start bit with baud rate units. The data in the SCn_TDR register will be transferred to transmit shift register after extra guard time. Extra Guard time: EGT[7:0] x “baud rate clock period”. No extra guard on EGT[7:0]=0.

21.3.12 SCn_T3DR: SCn T3 duration data register

SCn_T3DR register is 32-bit size and accessible in 32/16/8-bit. (n = 0)

SC0_T3DR=0x4000_5330

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Reserved	**T3D**
0x0000	**0x0000**
-	**RW**

15 0	T3D	T3 Duration Data bits. T3D[15:0] + 1.

21.3.13 SCn_T4DR: SCn T4 duration data register

SCn_T4DR register is 32-bit size and accessible in 32/16/8-bit. (n = 0)

SC0_T4DR=0x4000_5334

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Reserved	**T4D**
0x0000	**0x0000**
-	**RW**

15 0	T4D	T4 Duration Data bits. T4D[15:0] + 1

21.4 Functional description

The smartcard interface block comprises clock generator, transmitter, and receiver. The clock generation logic consists of the baud rate generator and clock output logic for smartcard interface mode.

The Transmitter consists of a write buffer, a serial shift register, parity generator, and control. Using DMA allows continuous transfer of data without any s/w involvement between frames. The receiver is the most complex part of the smartcard interface module due to its clock and data recovery units. The recovery unit is used for asynchronous data reception.

In addition to the recovery unit, the receiver includes a parity checker, a shift register, and control logic. The receiver supports the same frame formats as the transmitter and can detect frame error, data overrun and parity errors.

21.4.1 Smartcard interface clock generation

The clock generation logic generates the clock for the transmitter and receiver, and SCnCLK clock for smartcard interface mode. The SCn baud rate generator supports two modes of clock operation, which are 16 oversampling and 8 oversampling mode.

The SCnCLK output clock (fSC) for smartcard interface mode is controlled by SCnCLKEN bit of SCn_CR2 register and the frequency is determined by SCnCLKG[3:0] bits of SCn_CR2 register. The baud rate generator uses the PCLK for UART mode and the fSC for smartcard interface mode. (n = 0)

Table 95 shows the equations for calculating the baud rate (in bps).

Table 95. Equations for Calculating Baud Rate Register Settings (n = 0)

Oversampling	Equation for UART mode	Equation for smartcard interface mode
16 oversampling mode (OVRSn = 0)	Baud Rate = PCLK/(16(SCn_BDR+1))	Baud Rate = f_{SC}/(16(SCn_BDR+1))
8 oversampling mode (OVRSn = 1)	Baud Rate = PCLK/(8(SCn_BDR+1))	Baud Rate = f_{SC}/(8(SCn_BDR+1))

21.4.2 Smartcard interface baud rate compensation

The baud rate compensation is used to optimize the precision in each bit. There is a sign (BCMPS bit of SCn_BCMP register) bit to define the positive or negative compensation in each bit.

If the sign bit is “0b”, one clock of f_{SC} in smartcard interface mode or one clock of PCLK in UART mode will be appended in the compensated bit. If the sign bit is “1b”, one clock of f_{SC} in smartcard interface mode or one clock of PCLK in UART mode will be decreased in the compensated bit.

There are nine bits to define whether the relative compensation is required for each bit. The bits are BCMP[7:0] for data and BCMP8 for parity. (n = 0)

Example

1. PCLK = 32MHz, 16 oversampling, Baud rate = 9600 bps

   32MHz/(16 x 9600) = 208.3333, SCn_BDR = 208 - 1 = 207, and “Baud rate clock”/bit = 208 x 16

   So, “Clock error”/bit: 208.3333 x 16 - 208 x 16 = 5.3328 clock → “16 clock compensation”/bit if a BCMPx bit is “1b”.

   The result is that the sign bit, BCMPS, is “0b” for positive compensation and the baud rate compensation bits, BCMP[8:0], are “001001001b”. (CEPB: “clock error”/bit)

Table 96. Baud Rate Compensation Example 1

Rx/Tx bit	BCMPx bit	Clock error	Compensation bit	Final Clock Error
Start bit	–	-5.333 (CEPB)	x	-5.333
D0	bit 0	-10.667 (CEPB+ Before compensation)	1	5.333
D1	bit 1	0.000 (CEPB+ Before compensation)	0	0.000
D2	bit 2	-5.333 (CEPB+ Before compensation)	0	-5.333
D3	bit 3	-10.667 (CEPB+ Before compensation)	1	5.333
D4	bit 4	0.000 (CEPB+ Before compensation)	0	0.000
D5	bit 5	-5.333 (CEPB+ Before compensation)	0	-5.333
D6	bit 6	-10.667 (CEPB+ Before compensation)	1	5.333
D7	bit 7	0.000 (CEPB+ Before compensation)	0	0.000
Parity bit	bit 8	-5.333 (CEPB+ Before compensation)	0	-5.333

2. f_{SC} = 3.2MHz, 16 oversampling, Baud rate = 4800 bps

   3.2MHz/(16 x 4800) = 41.6667, SCn_BDR = 42 - 1 = 41, and “Baud rate clock”/bit = 42 x 16

   So, “Clock error”/bit: 41.6667 x 16 - 42 x 16 = -5.3333 clock → “16 clock compensation”/bit if a BCMPx bit is “1b”.

   The result is that the sign bit, BCMPS, is “1b” for negative compensation and the baud rate compensation bits, BCMP[8:0], are “001001001b”. (CEPB: “clock error”/bit)

Table 97. Baud Rate Compensation Example 2

Rx/Tx bit	BCMPx bit	Clock Error	Compensation bit	Final Clock Error
Start bit	–	+5.333 (CEPB)	x	+5.333
D0	bit 0	+10.667 (CEPB+ Before compensation)	1	-5.333
D1	bit 1	0.000 (CEPB+ Before compensation)	0	0.000
D2	bit 2	+5.333 (CEPB+ Before compensation)	0	+5.333
D3	bit 3	+10.667 (CEPB+ Before compensation)	1	-5.333
D4	bit 4	0.000 (CEPB+ Before compensation)	0	0.000
D5	bit 5	+5.333 (CEPB+ Before compensation)	0	+5.333
D6	bit 6	+10.667 (CEPB+ Before compensation)	1	-5.333
D7	bit 7	0.000 (CEPB+ Before compensation)	0	0.000
Parity bit	bit 8	+5.333 (CEPB+ Before compensation)	0	+5.333

21.4.3 Smartcard interface data format

A serial frame is defined to be one character of data bits with synchronization bits (start and stop bits), and optionally a parity bit for error detection. The SCn supports all 24 combinations of the followings as valid frame formats.

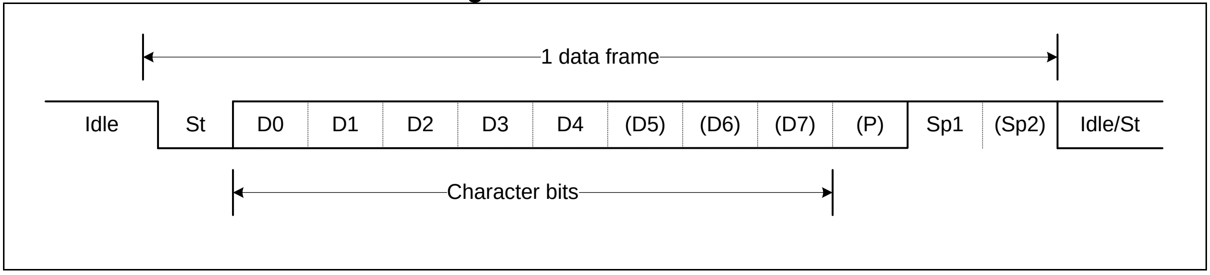
- 1 start bit
- 5, 6, 7, or 8 data bits
- No, even, or odd parity bit.
- 1 or 2 stop bits.

A frame starts with the start bit followed by the least significant data bit (LSB) or the most significant data bit (MSB) on the inverse convention of smartcard interface mode. Then the next data bits, up to eight, follow, ending with the most or least significant bit (MSB or LSB). If parity function is enabled, the parity bit is inserted between the last data bit and the stop bit.

A high-to-low transition on data pin is considered as start bit. When a complete frame is transmitted, it can be directly followed by a new frame, or the communication line can be set to an idle state. The idle state means high state of data pin.

The following figure shows the possible combinations of the frame formats. Bits inside round brackets are optional.

Figure 111. Frame Format



1 data frame consists of the following bits:

- Idle: No communication on communication line (SCnTXD/SCnRXD or SCnDATA)
- St : Start bit (low)
- Dm: Data bits (0 to 7)
- P: Parity bit (even parity, odd parity, no parity)
- Sp: Stop bit (1 bit or 2 bits)

The frame format is set by the DLENn[1:0], PSELn, PENn, and STOPBn bits in SCn_CR1 register. The transmitter and receiver use the same figures.

On the smartcard interface mode, the PENn and PSELn bits are set to “1b” for even parity and the DLENn[1:0] bits are set to “11b” for 8-bits data length. (n = 0)

21.4.4 Smartcard interface parity bit

The parity bit is calculated by doing an exclusive-OR of all the data bits. If odd parity is used, the result of the exclusive-OR is inverted. The parity bit is located between the last data bit and first stop bit of a serial frame.

- Peven = Dm-1 ^ … ^ D3 ^ D2 ^ D1 ^ D0 ^ 0
- Podd = Dm-1 ^ … ^ D3 ^ D2 ^ D1 ^ D0 ^ 1
- Peven: Parity bit using even parity
- Podd: Parity bit using odd parity
- Dm: Data bit n of the character

21.4.5 Smartcard interface transmitter

The transmitter is enabled by setting the TXEn bit in SCn_CR1 register. When the transmitter is enabled, the SCnDATA/SCnTXD pin should be set to SCnDATA function for the serial data input/output in smartcard interface mode or SCnTXD function for the serial output pin in UART mode by the GPIO registers.

Baud-rate, operation mode and frame format must be set up before doing any transmission.

In smartcard interface mode, the SCnCLK pin is used as the clock of a smartcard, so it should be selected to do SCnCLK function by the GPIO registers. (n = 0)

21.4.5.1 Smartcard interface sending TX data

A data transmission is initiated by loading the transmit data register (SCn_TDR register) to the data to be transmitted. The data to be written in transmit data register is moved to the shift register when the shift register is ready to send a new frame.

The shift register is loaded with the new data if it is in idle state or immediately after the last stop bit of the previous frame is transmitted. When the shift register is loaded to the new data, it will transfer one complete frame according to the settings of control registers. (n = 0)

21.4.5.2 Smartcard interface parity generator

The parity generator calculates the parity bit for the serial frame data to be sent. When parity bit is enabled (PENn = 1), the transmitter control logic inserts the parity bit between the last data bit and the first stop bit of the frame to be sent. (n = 0)

21.4.6 Smartcard interface receiver

The receiver is enabled by setting the RXEn bit in the SCn_CR1 register. When the receiver is enabled, the SCnDATA/SCnTXD pin should be set to SCnDATA function for the serial data input/output in smartcard interface mode or SCnRXD function for the serial input pin in UART mode by the GPIO registers.

Baud-rate, mode of operation and frame format must be set before serial reception.

In smartcard interface mode, the SCnCLK pin is used as the clock of a smartcard, so it should be selected to do SCnCLK function by the GPIO registers. (n = 0)

21.4.6.1 Smartcard interface receiving RX data

The receiver starts data reception when it detects a valid start bit (LOW) on SCnRXD pin during UART mode or on SCnDATA pin during smartcard interface mode. Each bit after start bit is sampled at predefined baud-rate and shifted into the receive shift register until the first stop bit of a frame is received.

Even if there is the second stop bit in the frame, the second stop bit is ignored by the receiver. That is, receiving the first stop bit means that a complete serial frame is presented in the receiver shift register and contents of the shift register are to be moved into the receive data register. (n = 0)

21.4.6.2 Smartcard interface parity checker

If parity bit is enabled (PENn = 1), the Parity Checker calculates the parity of the data bits in incoming frame and compares the result with the parity bit from the received serial frame. (n = 0)

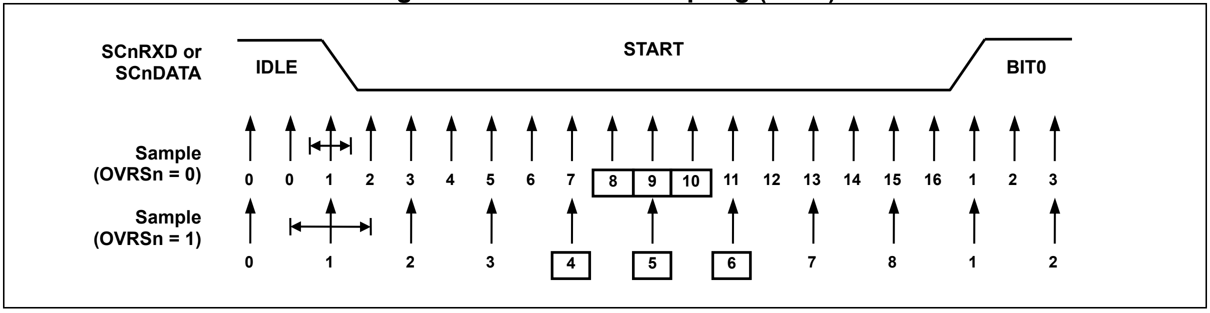
21.4.6.3 Smartcard interface data reception

To receive data frame, the receiver includes a clock and data recovery unit. The clock recovery logic is used for synchronizing the internally generated baud-rate clock to the incoming asynchronous serial frame on the SCnRXD pin in UART mode or on the SCnDATA pin in smartcard interface mode. The data recovery logic samples and filters the incoming bits with a low pass filter, and removes the noise of receive pin.

The following figure illustrates the sampling process of the start bit of an incoming frame. The sampling rate is 16 times the baud-rate in 16 oversampling mode and 8 times the baud-rate for 8 oversampling mode (DBLSn=1).

The horizontal arrows show the synchronization variation due to the asynchronous sampling process. Note that larger time variation is seen using the 8 oversampling mode. (n = 0)

Figure 112. Start Bit Sampling (n = 0)

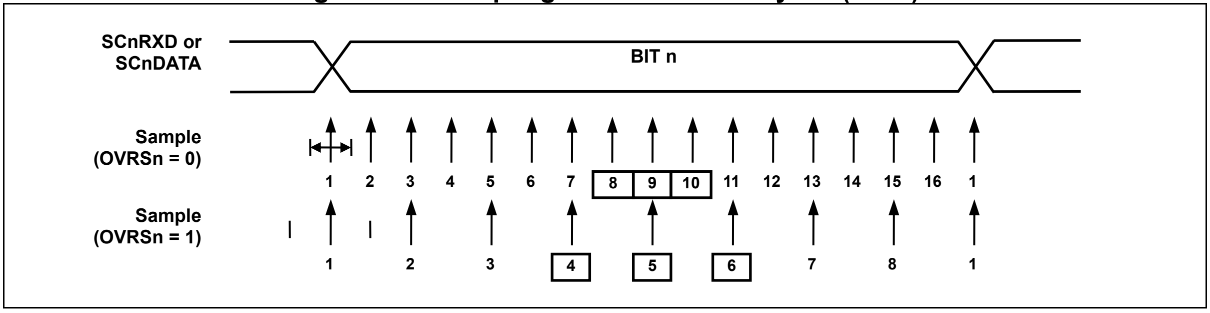


When the receiver is enabled (RXEn=1), the clock recovery logic tries to find a high-to-low transition on the SCnRXD line or SCnDATA line, the start bit condition. After detecting high to low transition on the line, the clock recovery logic uses samples 8, 9 and 10 for 16 oversampling mode to detect whether valid start bit is received. If more than 2 samples have logical low level, it is considered that a valid start bit is detected and the internally generated clock is synchronized to the incoming data frame. Then the data recovery can begin. The synchronization process is repeated for each start bit.

As described above, when the receiver clock is synchronized to the start bit, the data recovery can begin. Data recovery process is almost the same as clock recovery process. The data recovery logic samples each incoming bit 16 times for 16 oversampling mode and 8 times for 8 oversampling mode, and uses sample 8, 9 and 10 to decide data value. If more than 2 samples have low levels, the received bit is considered as a logic ‘0’ and if more than 2 samples have high levels, the received bit is considered as a logic ‘1’. The data recovery process is then repeated until a complete frame is received, including the first stop bit. The decided bit value is stored in the receive shift register in order.

Note that the receiver only uses the first stop bit of a frame. Internally, after receiving the first stop bit, the receiver is in idle state and waits to find the start bit. (n = 0)

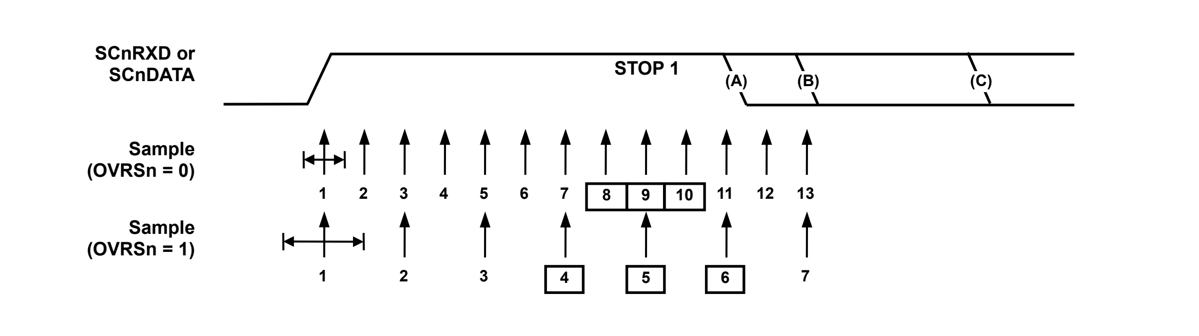
Figure 113. Sampling of Data and Parity Bit (n = 0)



The process for detecting stop bit is the same as clock and data recovery process. That is, if 2 or more samples of 3 center values have high level, correct stop bit is detected, or else a frame error (FEn) flag is set.

After deciding whether the first stop bit is valid or not, the receiver goes to idle state and monitors the SCnRXD or SCnDATA line to check whether a valid high to low transition is detected (start bit detection). (n = 0)

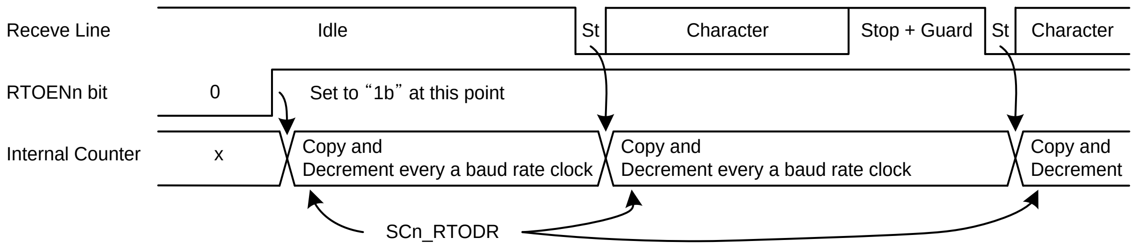
Figure 114. Stop Bit Sampling and Next Start Bit Sampling (n = 0)



21.4.6.4 Smartcard interface receive time out function

The receive time out function is used for checking a frame finish. This function is to count time with baud rate unit between the last start bit and a new start bit, and between setting the RTOENn bit of SCn_CR1 register and a new start bit. The SCn_RTODR register should have duration time value before using the receive time out function. (n = 0)

Figure 115. Receive Time Out Function



NOTES:

1. The value of SCn_RTODR register is transferred to an internal counter when the RTOENn bit is set to “1b” or when a start bit is received to.
2. If the RTOENn bit is set to “1b”, the internal counter counts down until underflow.
3. The RTOIFLAGn bit of SCn_IFSR register is set to “1b” on the internal counter underflow and the receive time out function is finished. So, to re-use the receive time out function, the RTOENn bit must be cleared to “0b” and then set to “1b” again.

21.4.7 Smartcard interface mode

Smartcard interface mode is selected by setting the SCnMD bit of SCn_CR1 register. When the smartcard interface mode, the PENn and PSELn bits should be set to “1b” and the DLENn[1:0] bits should be set to “11b” for 8-bit data length.

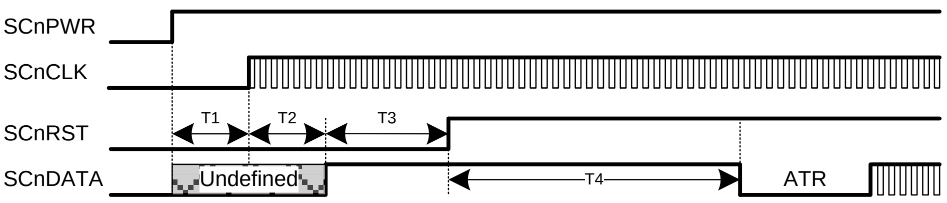
The smartcard interface mode uses five pins, SCnPWR, SCnRST, SCnCLK, SCnDATA, and SCnIN. So, the pins are configured as alternative function of smartcard interface mode. The smartcard interface mode acts as a half-duplex asynchronous communication using the SCnDATA pin. (n = 0)

21.4.7.1 Activation

The smartcard interface mode supports auto activation, but the activation can be controlled by software. The activation sequence is as follows. (n = 0)

- All the related pins are low level.
- The SCnPWR pin is set to high level.
- The SCnCLK pin is enabled for smartcard interface clock in reception mode.
- The SCnRST pin asserts to high level.

Figure 116. Activation Sequence



NOTES:
1. The RSTAIFGn bit of SCn_IFSR register will be set immediately after T3.
2. If the sequence is invalid or the ATR is not in T4, a SERIFGn bit of SCn_IFSR register will be set.
3. If the sequence is valid, a SEDIFGn bit of SCn_IFSR register will be set.

Table 98. Auto Activation Timing Table

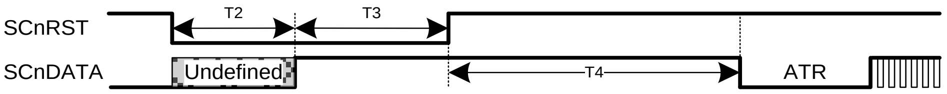
Time	Comment	No. of SCn Clock (fsc)
T1	SCnPWR High to SCnCLK Start	128, In Reception Mode
T2	SCnCLK Start to Rx Enable	128, In Reception Mode
T3	Rx Enable to SCnRST Assert	300 ≤ T3 ≤ 65536, SCnT3DR[15:0]+1
T4	SCnRST Assert to ATR Appear	400 ≤ T4 ≤ 40000, SCnT4DR[15:0]+1

21.4.7.2 Warm reset

The smartcard interface mode supports auto warm reset, but the warm reset can be controlled by software. The warm reset sequence is as follows. (n = 0)

- The SCnRST pin de-asserts to low level.
- The SCnDATA pin is in reception mode after T2 time.
- The SCnRST pin re-asserts to high level after T3 time.

Figure 117. Warm Reset Sequence



NOTES:

1. The RSTAIFGn bit of SCn_IFSR register will be set immediately after T3.
2. If the sequence is invalid or the ATR is not in T4, a SERIFGn bit of SCn_IFSR register will be set.
3. If the sequence is valid, a SEDIFGn bit of SCn_IFSR register will be set.

Table 99. Auto Warm Reset Timing Table

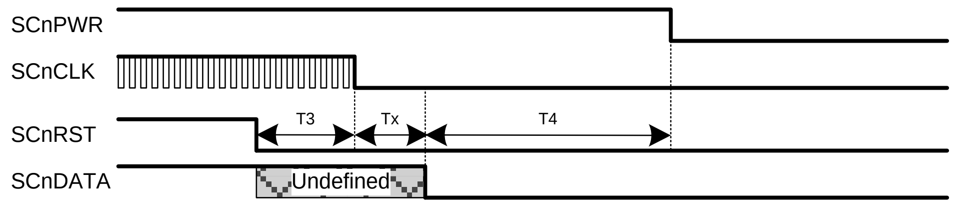
Time	Comment	No. of SCn Clock (fsc)
T2	SCnRST Low to Rx enable	128, In Reception Mode
T3	Rx Enable to SCnRST Assert	300 ≤ T3 ≤ 65536, SCnT3DR[15:0]+1
T4	SCnRST Assert to ATR Appear	400 ≤ T4 ≤ 40000, SCnT4DR[15:0]+1

21.4.7.3 Deactivation

The smartcard interface mode supports auto deactivation, but the deactivation can be controlled by software. The deactivation sequence is as follows. (n = 0)

- The SCnRST pin de-asserts to low level.
- The SCnCLK pin is disabled and set to low level.
- The SCnDATA pin is set to low level.
- The SCnPWR pin is set to low level.

Figure 118. Deactivation Sequence



NOTE: If the sequence is valid, a SEDIFGn bit of SCn_IFSR register will be set.

Table 100. Auto Deactivation Timing Table

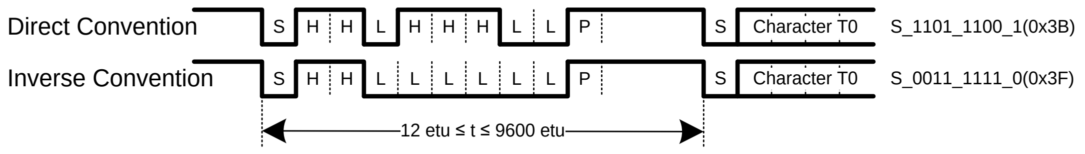
Time	Comment	No. of SCn Clock (fsc)
T3	SCnRST Low to SCnCLK Disable/Low	50 ≤ T3 ≤ 200, SCnT3DR[15:0]+1
Tx	SCnCLK Disable to SCnDATA Low	128, In Reception Mode
T4	SCnDATA Low to SCnPWR Low	50 ≤ T4 ≤ 200, SCnT4DR[15:0]+1

21.4.7.4 Initial character TS

The initial character TS of Answer to Reset (ATR) has two conventions according to 7816-3. That is direct and inverse conventions. The direct convention is LSB-first and bit value “1b” corresponding to state H.

The inverse convention is MSB-first and bit value “1b” corresponding to stathe L. The smartcard interface mode supports auto convention detection by setting the ACONDETn bit of SCn_CR3 register, but the convention detection can be controlled by software. If auto convention is used, the ACONDETn bit should be set before ATR. (n = 0)

Figure 119. Initial Character TS



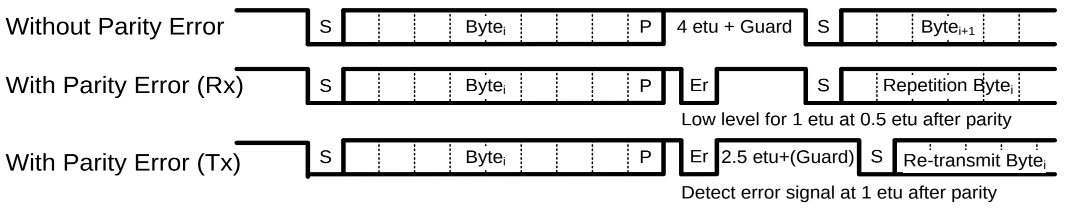
NOTE: Where etu, elementary time unit, is baud-rate bit unit.

21.4.7.5 Error signal and character repetition

In smartcard interface mode, when character parity is incorrect, the receiver shall transmit an error signal to transmitter to inform parity error by pulling down the SCnDATA pin to low during one to two etu. Then the transmitter will retransmit the character. The smartcard interface mode supports an error detection function in the receiver and a re-transmit function in the transmitter.

The error detection and retransmit function are enabled by setting the RETRYENn bit of SCn_CR3 register and the retry number is defined by the RETRYn[2:0] bits of SCn_CR3 register. The retry limitation maximum is seven when the RETRYn[2:0] bits are 0x7. If the error signal repeats as the set number of retries, the RETRYENn bit is cleared to “0b” to stop error signal generation/detection and the TRYERIFGn bit is set to “1b”. (n = 0)

Figure 120. Smartcard Interface Error Signal When the RETRYENn Bit is Set to “1b”



22 LCD Driver

LCD driver of the A31L21x series includes an LCD control register (LCD_CR) and an LCD contrast control register (LCD_CCR). LCLK[1:0] of the LCD_CR determines frequency of COM signal scanning each segment output. A RESET clears the LCD_CR, and sets the LCD_CCR to logic ‘0’.

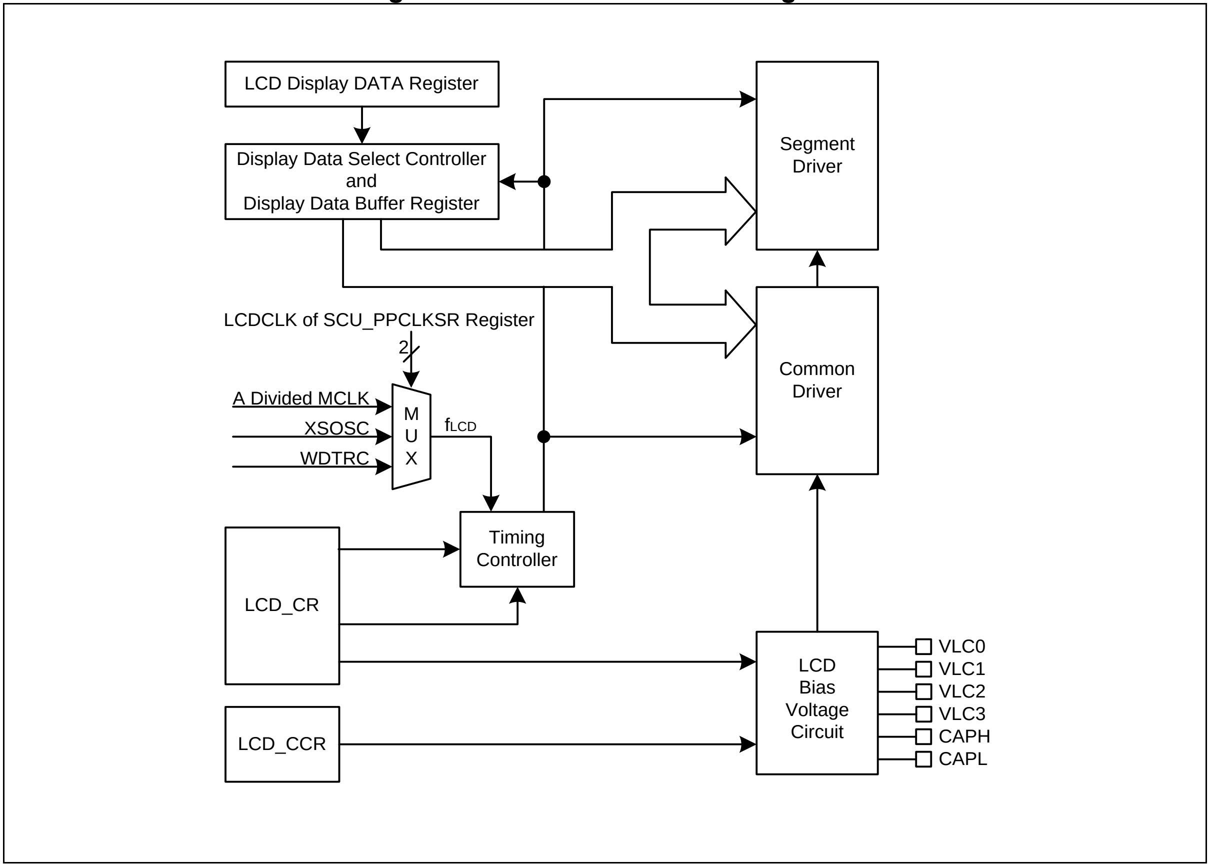
LCD display can continue its operation even during SLEEP mode and DEEP SLEEP mode if it uses a selected clock for LCD driver.

A clock and duty of the LCD driver is initialized by hardware whenever a value is written to the control register. So, it is recommended not to rewrite the LCD_CR frequently.

22.1 LCD driver block diagram

Figure 121 shows a block diagram of the LCD driver block.

Figure 121. LCD Driver Block Diagram



22.2 Pin description for LCD driver

Table 101. Pins and External Signals for LCD Driver

Pin name	Type	Description
COM0 to COM7	O	LCD common signal outputs
SEG0 to SEG44	O	LCD segment signal outputs
VLC0/1/2/3	I/O	LCD bias voltage input/output
CAPH/L	O	Capacitor terminal for voltage booster

22.3 Registers

Base address and register map of the LCD driver are shown in Table 102 and Table 103.

Table 102. Base Address of LCD Driver

Name	Base address
LCD	0x4000_5000

Table 103. LCD Driver Register Map

Name	Offset	Type	Description	Reset value
LCD_CR	0x0000	RW	LCD driver control register	0x00000000
LCD_CCR	0x0004	RW	LCD contrast control register	0x00000000
LCD_DR0 to LCD_DR44	0x0010 to 0x003C	RW	LCD display data register 0 to 44	Unknown

22.3.1 LCD_CR: LCD driver control register

LCD_CR register is 32-bit size and accessible in 32/16/8-bit.

LCD_CR=0x4000_5000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																				**Reserved**		**BTYPE**		**IRSEL**		**DBS**			**LCLK**		**DISP**
0x00000																				**00**		**00**		**00**		**000**			**00**		**0**
-																				-		**RW**		**RW**		**RW**			**RW**		**RW**

9	BTYPE	LCD Bias Type Selection.
8		00 Internal resistor bias.
		01 External resistor bias.
		10 Capacitor bias (Voltage booster).
		11 Not available.
		NOTES: 1. All the VLC0 – VLC3, CAPH, and CAPL pins must be used as bias functions when the capacitor bias is selected for the LCD bias type. 2. 2. Refer to the PB_MOD and PB_AFSR1 registers for pin function.
7	IRSEL	Internal LCD Bias Dividing Resistor Selection.
6		00 RLCD2: 60[kΩ]
		01 RLCD1: 30[kΩ]
		10 RLCD3: 120[kΩ]
		11 Not available.
5	DBS	LCD Duty and Bias Selection.
3		000 1/8 duty, 1/4 bias.
		001 1/6 duty, 1/4 bias.
		010 1/5 duty, 1/3 bias.
		011 1/4 duty, 1/3 bias.
		100 1/3 duty, 1/3 bias.
		101 1/3 duty, 1/2 bias
		110 1/2 duty, 1/2 bias
		Others reserved.
2	LCLK	LCD Clock Selection (When fLCD = 32.768kHz).
1		00 128Hz.
		01 256Hz.
		10 512Hz.
		11 1024Hz.
0	DISP	LCD Display Control.
		0 Display off (The LCD block and voltage booster off).
		1 Normal display on (When the BTYPE[1:0] = “10b”, the voltage booster is turn on).

22.3.2 LCD_CCR: LCD contrast control register

LCD_CCR register is 32-bit size and accessible in 32/16/8-bit.

LCD_CCR=0x4000_5004

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="28">Reserved</td><td colspan="4">VLCD</td></tr>
<tr><td colspan="28">0x0000000</td><td colspan="4">0000</td></tr>
<tr><td colspan="28">-</td><td colspan="4">RW</td></tr>
</table>

3
0 VLCD VLC3 Voltage Control when the capacitor bias is enabled by the BTYPE[1:0] bits.

	1/3 Bias	1/2 and 1/4 Bias
0000	0.725V	0.725V
0001	0.750V	0.750V
0010	0.775V	0.775V
0011	0.800V	0.800V
0100	0.825V	0.825V
0101	0.850V	Not available
0110	0.875V	Not available
0111	0.900V	Not available
1000	0.925V	Not available
1001	0.950V	Not available
1010	0.975V	Not available
1011	1.000V	Not available
1100	1.025V	Not available
1101	1.050V	Not available
1110	1.075V	Not available
1111	1.100V	Not available

NOTE: The VLC0 voltage can be calculated by the below formulas.

— VLC0 = VLC3 x 3 when 1/3 bias.

— VLC0 = VLC3 x 4 when 1/2 and 1/4 bias

22.3.3 LCD_DRx: LCD display data register x (x = 0 to 44)

LCD_DRx register is 8-bit size and accessible in 32/16/8-bit. (n = 0 to 11)

LCD_DRx=0x4000_5010 to 0x4000_503C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
LCD_DR(4xn+3)	LCD_DR(4xn+2)	LCD_DR(4xn+1)	LCD_DR(4xn+0)
0xXX	0xXX	0xXX	0xXX
RW	RW	RW	RW

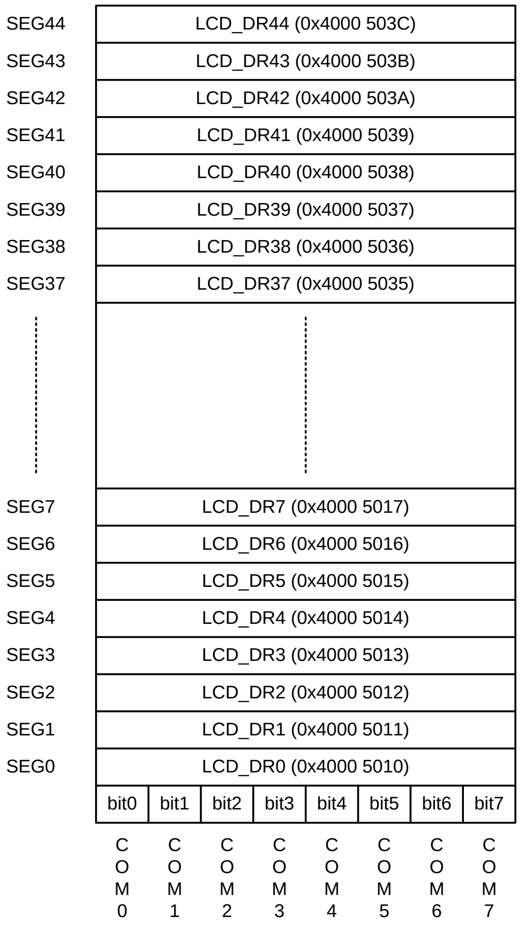
31 24	LCD_DR(4xn+3)	LCD Display Data.
23 16	LCD_DR(4xn+2)	LCD Display Data.
15 8	LCD_DR(4xn+1)	LCD Display Data.
7 0	LCD_DR(4xn+0)	LCD Display Data.

22.4 LCD display RAM organization

Display data are stored in display data area. The display data stored to the display data area (address 0x4000_5010-0x4000_503C) are read automatically and are sent to the LCD driver by hardware. The LCD driver generates the segment and common signals in accordance with the display data and driving method. Therefore, display patterns can be changed by simply overwriting the contents of the display data area with a program.

Figure 122 shows the correspondence between the display data area and the COM/SEG pins. The LCD is turned on when the display data is ‘1’ and turned off when it is ‘0’.

Figure 122. LCD Display RAM Organization



SEG44	LCD_DR44 (0x4000 503C)							
SEG43	LCD_DR43 (0x4000 503B)							
SEG42	LCD_DR42 (0x4000 503A)							
SEG41	LCD_DR41 (0x4000 5039)							
SEG40	LCD_DR40 (0x4000 5038)							
SEG39	LCD_DR39 (0x4000 5037)							
SEG38	LCD_DR38 (0x4000 5036)							
SEG37	LCD_DR37 (0x4000 5035)							
SEG7	LCD_DR7 (0x4000 5017)							
SEG6	LCD_DR6 (0x4000 5016)							
SEG5	LCD_DR5 (0x4000 5015)							
SEG4	LCD_DR4 (0x4000 5014)							
SEG3	LCD_DR3 (0x4000 5013)							
SEG2	LCD_DR2 (0x4000 5012)							
SEG1	LCD_DR1 (0x4000 5011)							
SEG0	LCD_DR0 (0x4000 5010)							
	bit0	bit1	bit2	bit3	bit4	bit5	bit6	bit7
	COM0	COM1	COM2	COM3	COM4	COM5	COM6	COM7

22.5 LCD signal waveform

Figure 123. LCD Signal Waveforms (1/2 Duty, 1/2 Bias)

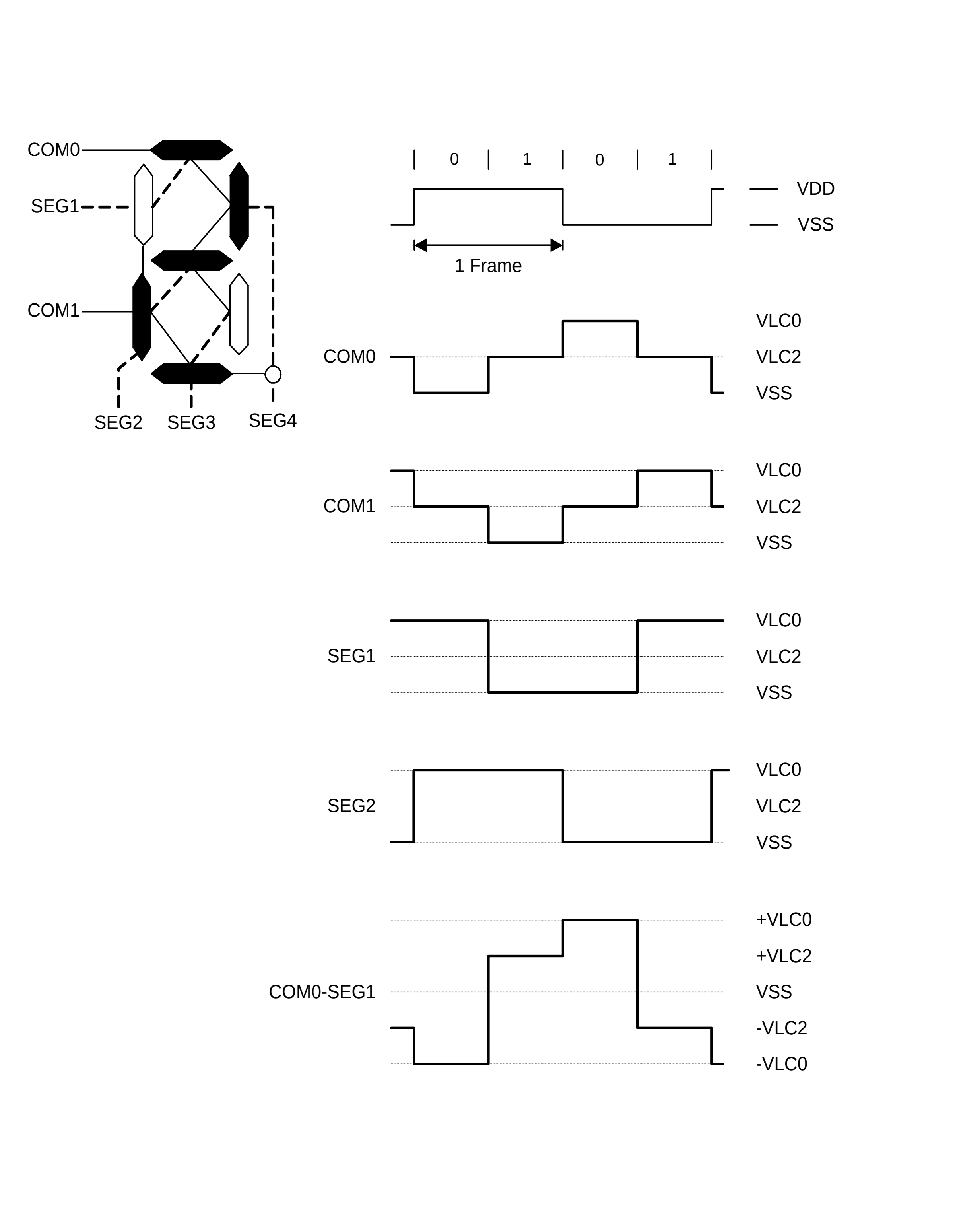


Figure 124. LCD Signal Waveforms (1/3 Duty, 1/3 Bias)

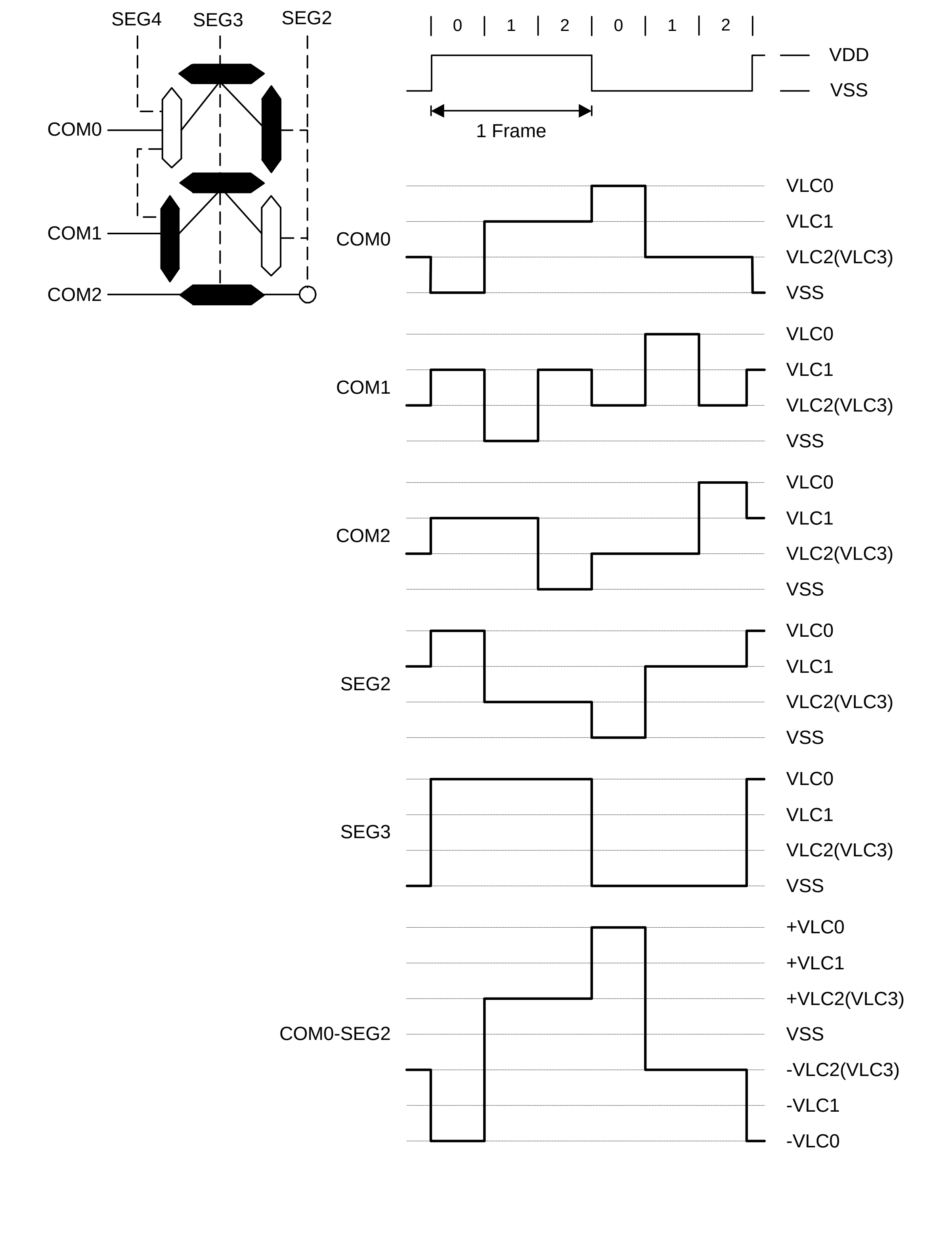


Figure 125. LCD Signal Waveforms (1/4 Duty, 1/3 Bias)

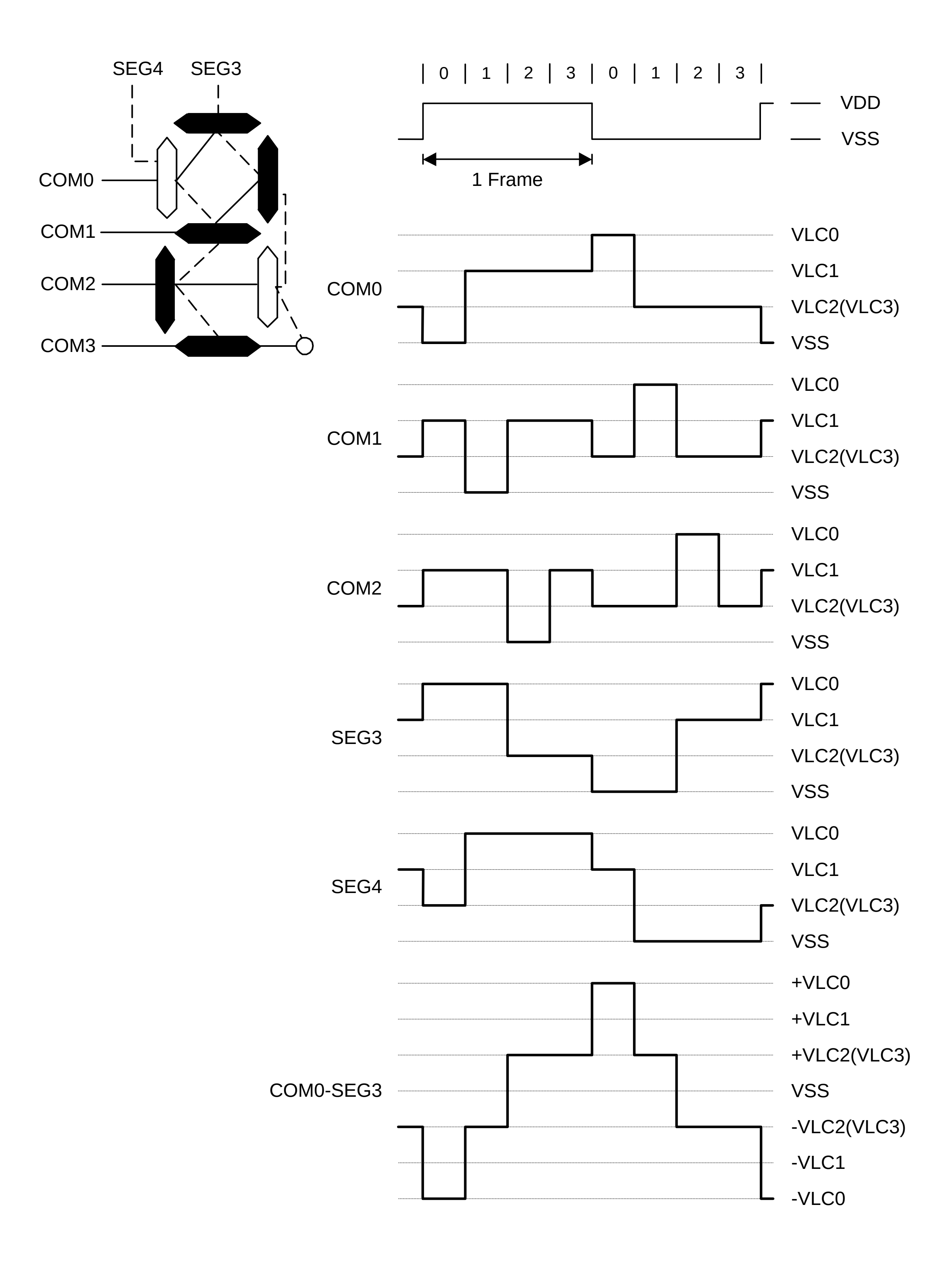
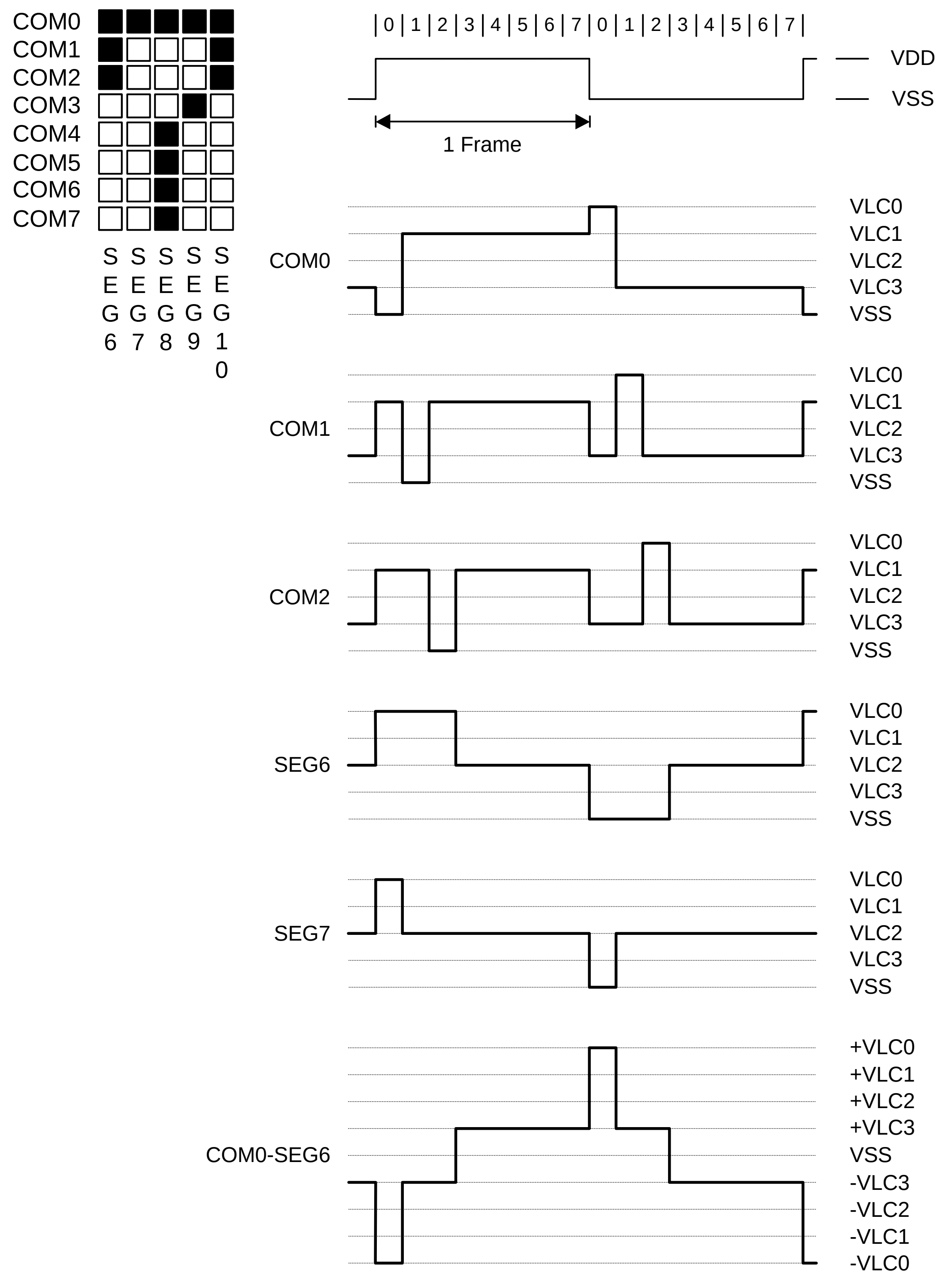
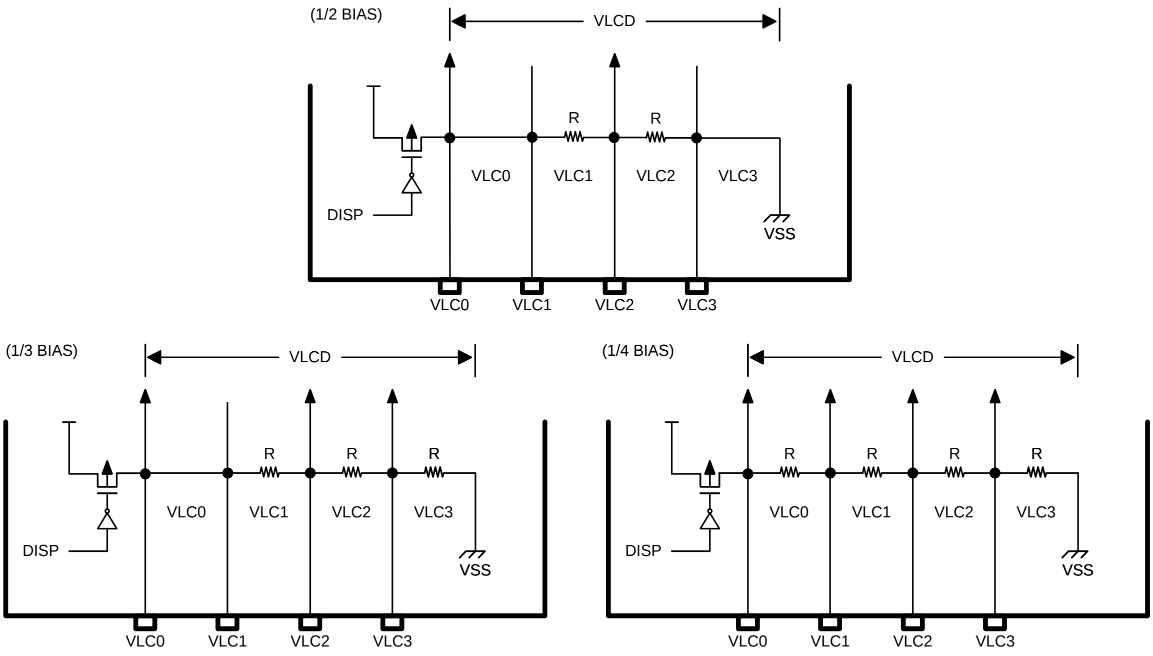


Figure 126. LCD Signal Waveforms (1/8 Duty, 1/4 Bias)



22.6 Internal resistor bias connection

Figure 127. Internal Resistor Bias Connection

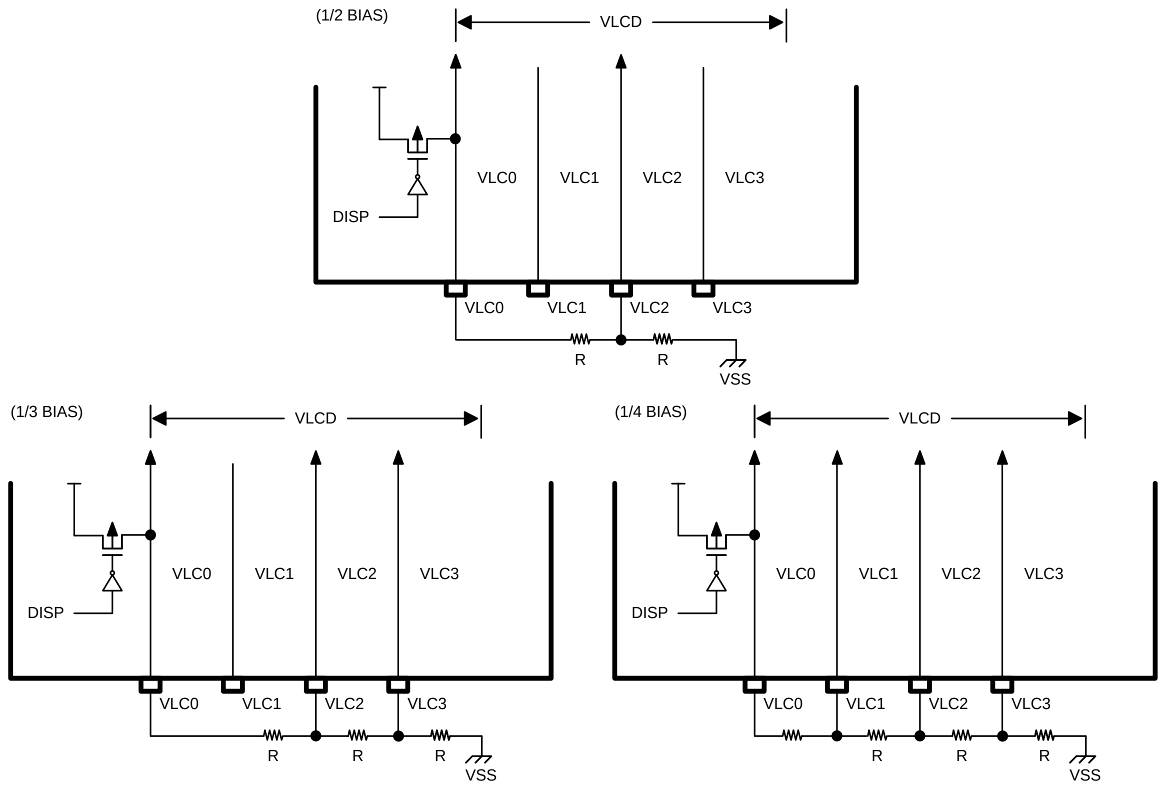


NOTES:

1. When 1/2 bias are selected, the VLC0/VLC1 and VLC3/VSS are internally connected like the above figure.
2. When 1/3 bias are selected, the VLC0 and VLC1 are internally connected.
3. When the internal resistors are selected, all the VLC0, VLC1, VLC2, VLC3, CAPH, and CAPL pins can be used for normal I/O.

22.7 External resistor bias connection

Figure 128. External Resistor Bias Connection

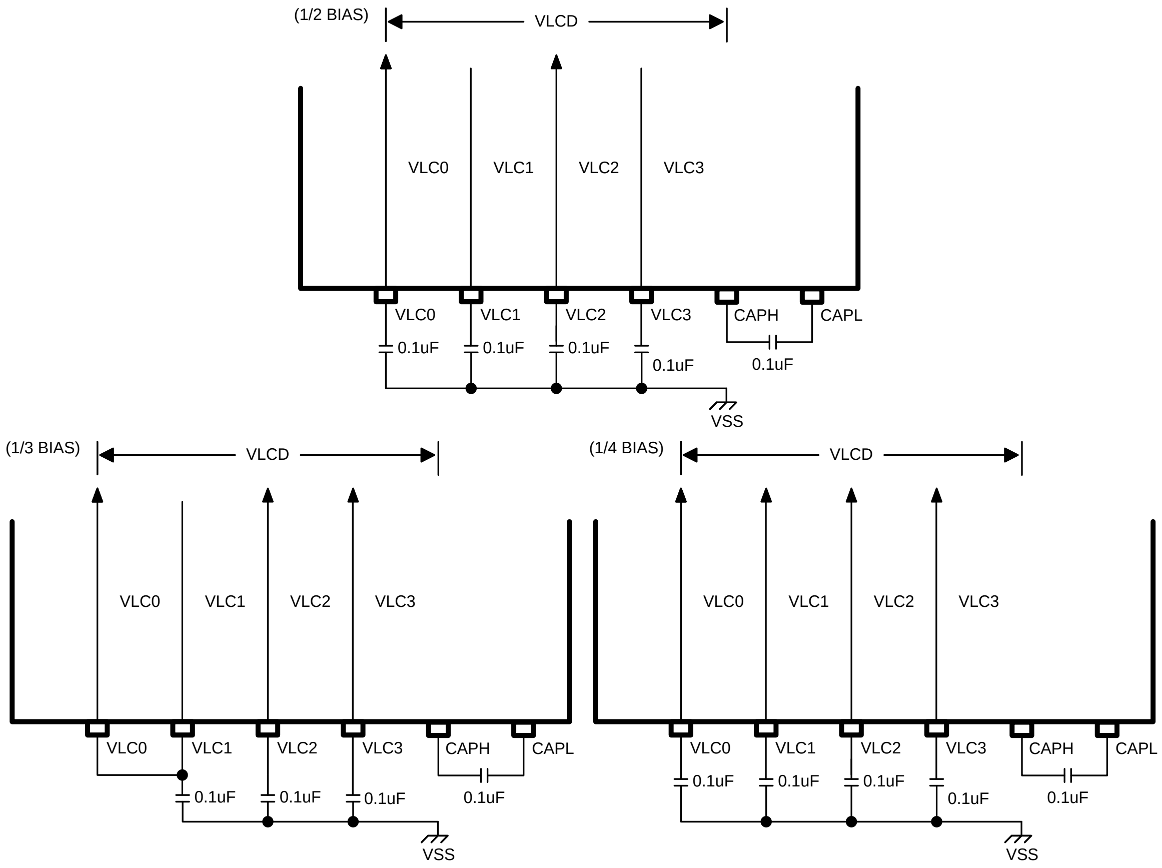


NOTES:

1. When the external resistor bias is selected, the internal resistors for bias are disconnected.
2. When the external resistor bias is selected, the dividing resistors should be connected like the above figure and the needed bias pins should be selected as the LCD bias function.
   - When it is 1/2 bias, the VLC0 and VLC2 pins should be selected as the LCD bias function. The other pins can be used for normal I/O.
   - When it is 1/3 bias, the VLC0, VLC2, and VLC3 pins should be selected as the LCD bias function.
     The other pins can be used for normal I/O.
   - When it is 1/4 bias, the VLC0, VLC1, VLC2, and VLC3 pins should be selected as the LCD bias function. The other pins can be used for normal I/O.

22.8 Capacitor bias connection

Figure 129. Capacitor Bias Connection



NOTE: When the capacitor bias is selected for the VLCD voltage, all the VLC0, VLC1, VLC2, VLC3, CAPH, and CAPL pins should be selected for LCD bias function and should be connected like the above figure.

23 Cyclic Redundancy Check (CRC) and checksum

Cyclic Redundancy Check (CRC) generator is used to obtain 8/16/32-bit CRC code of Flash ROM and any data stream.

Among other applications, CRC-based techniques are used to verify data transmission or storage integrity. In the scope of functional safety standards, they offer means of verifying Flash memory's integrity.

The CRC generator helps computing the signature of the software during runtime, comparing with a reference signature.

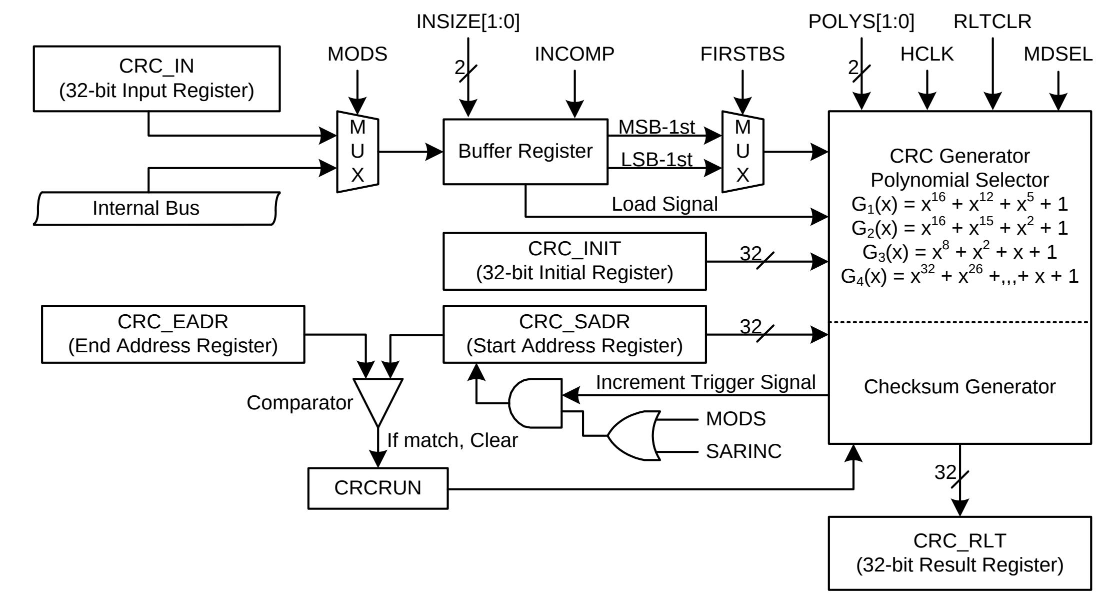
A CRC generator of the A31L21x series has following features:

- Auto CRC and User CRC Mode
- Supports CRC-CCITT ($G_1(x) = x^{16} + x^{12} + x^5 + 1$)
- Supports CRC-16 ($G_2(x) = x^{16} + x^{15} + x^2 + 1$)
- Supports CRC-8 ($G_3(x) = x^8 + x^2 + x + 1$)
- Supports CRC-32 ($G_4(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$)
- CRC and Checksum mode
- CRC/Checksum Start Address Auto Increment (User mode only)

23.1 CRC and checksum block diagram

Figure 130 shows a block diagram of the CRC and checksum interface block.

Figure 130. CRC and Checksum Block Diagram



NOTES:

1. The operation is finished after calculating from the address specified by the CRC_SADR to the address specified by the CRC_EADR, in auto mode or when SARINC=1.
2. The CRC_SADR and CRC_EADR have the same value after finishing the operation.
3. The end address in the CRC_EADR must be greater than the start address in CRC_SADR.
4. The end address must be at least 0x7C from the start address in “Auto mode”.
5. The CRC_SADR/CRC_EADR in “Auto mode” should have any value in Flash memory area (0x10000000 to 0x1001FFFF).
6. The CPU will be held at “Auto mode” if the CPU is in the Flash memory, and global interrupts should be disabled by software.
7. Users must set the HCLK frequency to be less than or equal to 20MHz in “Auto mode” by configuring the MODS bit of the CRC_CR register.
8. In Auto mode, the CRC_CR can’t be written and the CRC_RLT can’t be read.

23.2 Registers

Base address and register map of the CRC and checksum block are shown in Table 104 and Table 105.

Table 104. Base Address of CRC

Name	Base address
CRC	0x3000_1000

Table 105. CRC Register Map

Name	Offset	Type	Description	Reset value
CRC_CR	0x0000	RW	CRC/Checksum Control Register	0x00000000
CRC_IN	0x0004	RW	CRC/Checksum Input Data Register	0x00000000
CRC_RLT	0x0008	RO	CRC/Checksum Result Data Register	0xFFFFFFFF
CRC_INIT	0x000C	RW	CRC/Checksum Initial Data Register	0x00000000
CRC_SADR	0x0010	RW	CRC/Checksum Start Address Register	0x10000000
CRC_EADR	0x0014	RW	CRC/Checksum End Address Register	0x1001FFFC

23.2.1 CRC_CR: CRC control register

CRC_CR register is 32-bit size and accessible in 32/16/8-bit.

CRC_CR=0x3000_1000

<table>
<tr><td colspan="16">31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="16">Reserved</td><td colspan="2">INSIZE</td><td colspan="3">Reserved</td><td>INCOMP</td><td colspan="2">Reserved</td><td>MODS</td><td>RLTCLR</td><td>MDSEL</td><td colspan="2">POLYS</td><td>SARINC</td><td>FIRSTBS</td><td>CRCRUN</td></tr>
<tr><td colspan="16">0x0000</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td colspan="16">-</td><td>RW</td><td>RW</td><td>-</td><td>-</td><td>-</td><td>RW</td><td>-</td><td>-</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td></tr>
</table>

15 14	INSIZE	Input Data Size Selection.	
		00	32-bit is the input data size.
		01	16-bit is the input data size.
		10	8-bit is the input data size.
		11	Reserved.
10	INCOMP	Input Data Complement.	
		0	No effect
		1	1’s complement of input data. Ex) If 0x3AB7, the complement data are 0xC548.
7	MODS	User/Auto Mode Selection.	
		0	User mode. (Calculate every data written to the CRC_IN register)
		1	Auto mode. (Calculate till CRC_SADR == CRC_EADR)
6	RLTCLR	CRC/Checksum Result Data Register (CRC_RLT) Initialization.	
		0	No effect.
		1	Initialize the CRC_RLT register with the value of CRC_INIT (This bit is automatically cleared to ‘0’ after operation)
5	MDSEL	CRC/Checksum Selection.	
		0	Select CRC.
		1	Select checksum.
4 3	POLYS	Polynomial Selection. (CRC only)	
		00	CRC-CCITT ($G_1(x) = x^{16} + x^{12} + x^5 + 1$)
		01	CRC-16 ($G_2(x) = x^{16} + x^{15} + x^2 + 1$)
		10	CRC-8 ($G_3(x) = x^8 + x^2 + x + 1$)
		11	CRC-32 ($G_4(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} +x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$)
2	SARINC	CRC/Checksum Start Address Auto Increment Control. (User mode only)	
		0	No effect.
		1	The CRC/Checksum start address register is incremented as the selected input size every writing to the CRC_IN register.
1	FIRSTBS	First Shifted-in Selection. (CRC only)	
		0	MSB-1st.
		1	LSB-1st.
0	CRCRUN	CRC/Checksum Start Control and Busy.	
		0	Not busy. The CRC operation can be finished by writing ‘0’ to this bit while running.
		1	Start CRC operation. This bit is automatically cleared to ‘0’ when the value of CRC_SADR register reaches the value of CRC_EADR register.
		NOTE:	The 5 “NOP instruction” should be executed immediately after this bit is set to ‘1’.

NOTES:

1. The CRC_RLT register and the CRC/Checksum block should be initialized by writing ‘1’ to the RLTCLR bit before a new CRC/Checksum calculation.
2. The CRCRUN bit should be set to ‘1’ last time after setting appropriate values to the registers.
3. On the user mode, it will be calculated every writing data to the CRC_IN register during CRCRUN==1.

4. On the user mode with SARINC==0, the block is finished by writing ‘0’ to the CRCRUN bit.
5. It is prohibited writing any data to the CRC_IN register during CRCRUN==0.
6. The checksum is calculated by a selected input data size unit.
   — Ex1) On 8-bit size, CRC_RLT = 8-bit byte + 8-bit byte + 8-bit byte + -----.
   — Ex2) On 16-bit size, CRC_RLT = 16-bit word + 16-bit word + 16-bit word + -----.
   — Ex3) On 32-bit size, CRC_RLT = 32-bit word + 32-bit word + 32-bit word + -----.
7. The 5 “NOP Instruction” should follow immediately after CRCRUN bit is set to ‘1’.

23.2.2 CRC_IN: CRC input data register

CRC_IN register is 32-bit size.

CRC_IN=0x3000_1004

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INDATA																															
0x00000000 **RW**																															

31 0	INDATA	CRC Input Data.

NOTE: The CRC_IN register can be written by 1-byte (8-bits), half-word (16-bits), and 1-word (32-bits).

23.2.3 CRC_RLT: CRC result data register

CRC_RLT register is 32-bit size and accessible in 32/16/8-bit.

CRC_RLT=0x3000_1008

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RLTDATA																															
0xFFFFFFFF **RO**																															

31 0	RLTDATA	CRC Result Data.

23.2.4 CRC_INIT: CRC initial data register

CRC_INIT register is 32-bit size and accessible in 32/16/8-bit.

CRC_INIT=0x3000_100C

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="32">INIDATA</td></tr>
<tr><td colspan="32">0x00000000</td></tr>
<tr><td colspan="32">RW</td></tr>
</table>

31 0	INIDATA	CRC Initial Data.

23.2.5 CRC_SADR: CRC start address register

CRC_SADR register is 32-bit size and accessible in 32/16/8-bit.

CRC_SADR=0x3000_1010

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="32">SADR</td></tr>
<tr><td colspan="32">0x10000000</td></tr>
<tr><td colspan="32">RW</td></tr>
</table>

31 0	SADR	CRC Start Address

NOTES:

1. The LSB-1bit of the start address should be “0b” on the 16-bits input data size.
2. The LSB-2bits of the start address should be “00b” on the 32-bits input data size.

23.2.6 CRC_EADR: CRC end address register

CRC_EADR register is 32-bit size and accessible in 32/16/8-bit.

CRC_EADR=0x3000_1014

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EADR																															
0x1001FFFC																															
RW																															

31 0	EADR	CRC End Address.

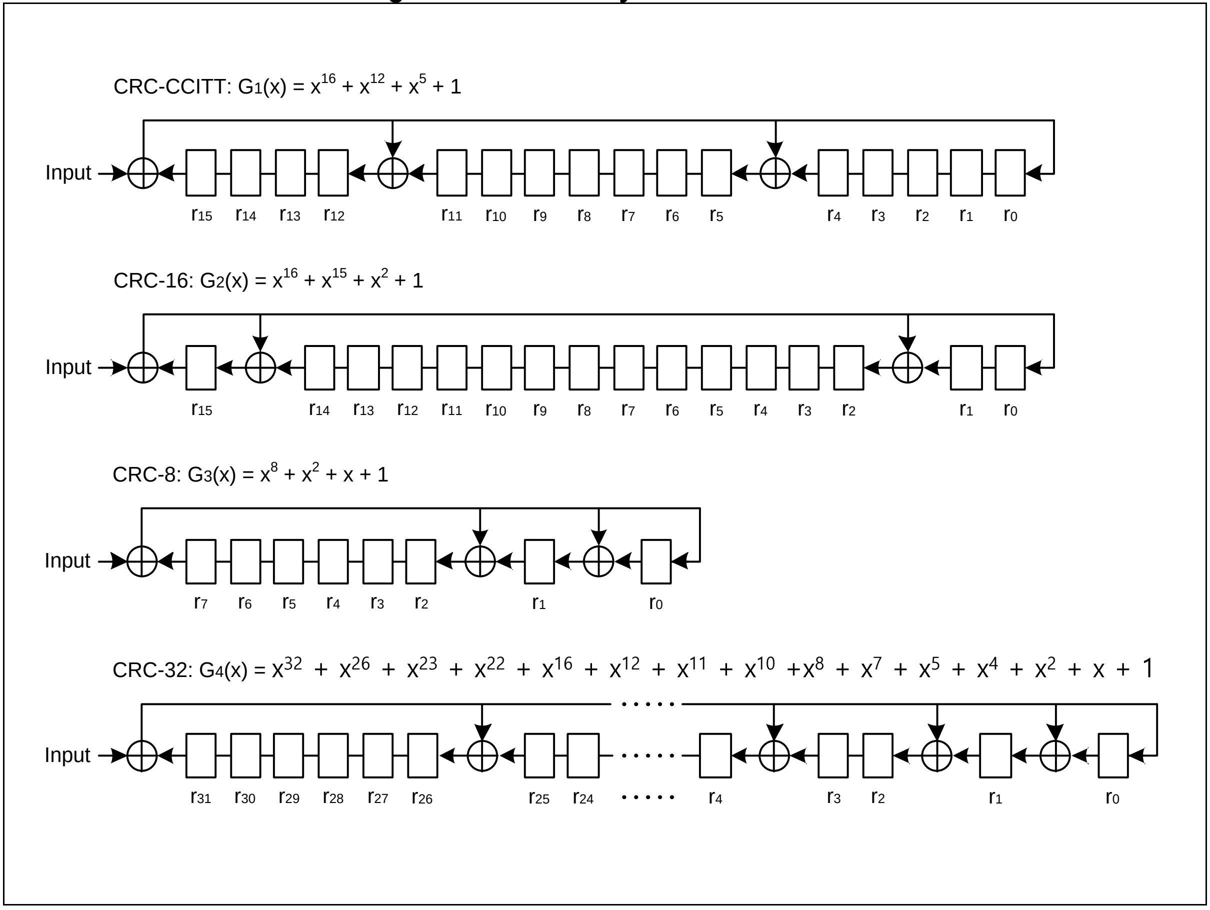
NOTES:

1. The LSB-1bit of the end address should be “0b” on the 16-bits input data size.
2. The LSB-2bits of the end address should be “00b” on the 32-bits input data size.

23.3 Functional description

23.3.1 CRC polynomial structure

Figure 131. CRC Polynomial Structure


CRC-CCITT: G1(x) = x16 + x12 + x5 + 1
Input
r15 r14 r13 r12 r11 r10 r9 r8 r7 r6 r5 r4 r3 r2 r1 r0
CRC-16: G2(x) = x16 + x15 + x2 + 1
Input
r15 r14 r13 r12 r11 r10 r9 r8 r7 r6 r5 r4 r3 r2 r1 r0
CRC-8: G3(x) = x8 + x2 + x + 1
Input
r7 r6 r5 r4 r3 r2 r1 r0
CRC-32: G4(x) = x32 + x26 + x23 + x22 + x16 + x12 + x11 + x10 +x8 + x7 + x5 + x4 + x2 + x + 1
Input
r31 r30 r29 r28 r27 r26 r25 r24 ····· r4 r3 r2 r1 r0

23.3.2 The CRC operation procedure in auto CRC/checksum mode

1. CRC/Checksum Clock Enable
2. Set CRC start address register. (CRC_SADR)
3. Set CRC end address register. (CRC_EADR)
4. Set CRC initial data register. (CRC_INIT)
5. Global interrupt Disable.
6. Select CRC(HCLK) Clock. (HCLK should be less than or equal to 20MHz during CRC/Checksum auto mode)
7. Select Auto CRC/Checksum Mode and CRC.
8. CRC operation starts. (CRCRUN = 1)
9. Read the CRC result.
10. Global interrupt Enable.

23.3.3 The CRC operation procedure in user CRC/checksum mode

1. CRC/Checksum Clock Enable
2. Set CRC start address register. (CRC_SADR)
3. Set CRC end address register. (CRC_EADR)
4. Set CRC initial data register. (CRC_INIT)
5. Select User CRC/Checksum Mode and CRC
6. CRC operation starts. (CRCRUN = 1)
7. Input CRC Data at CRC_IN.
8. Check CRC is finished on Start Address Auto Increment or Compare Start address and End address in order to check CRC end point.
9. Repeat 8 and 9 until CRC end point.
10. CRC Stop and read CRC result.

24 Advanced Encryption Standard (AES-128)

AES-128 can be used to encrypt and decrypt data using the AES algorithm. It is compliant implementation of “The advanced encryption standard as defined by Federal Information Processing Standards Publication” (FIPS PUB 179, 2001 November 26).

The AES-128 encrypts and decrypts 128-bit blocks using 128-bit key length. It supports key expansion function for decryption and DMA transfer for incoming and for outcoming data.

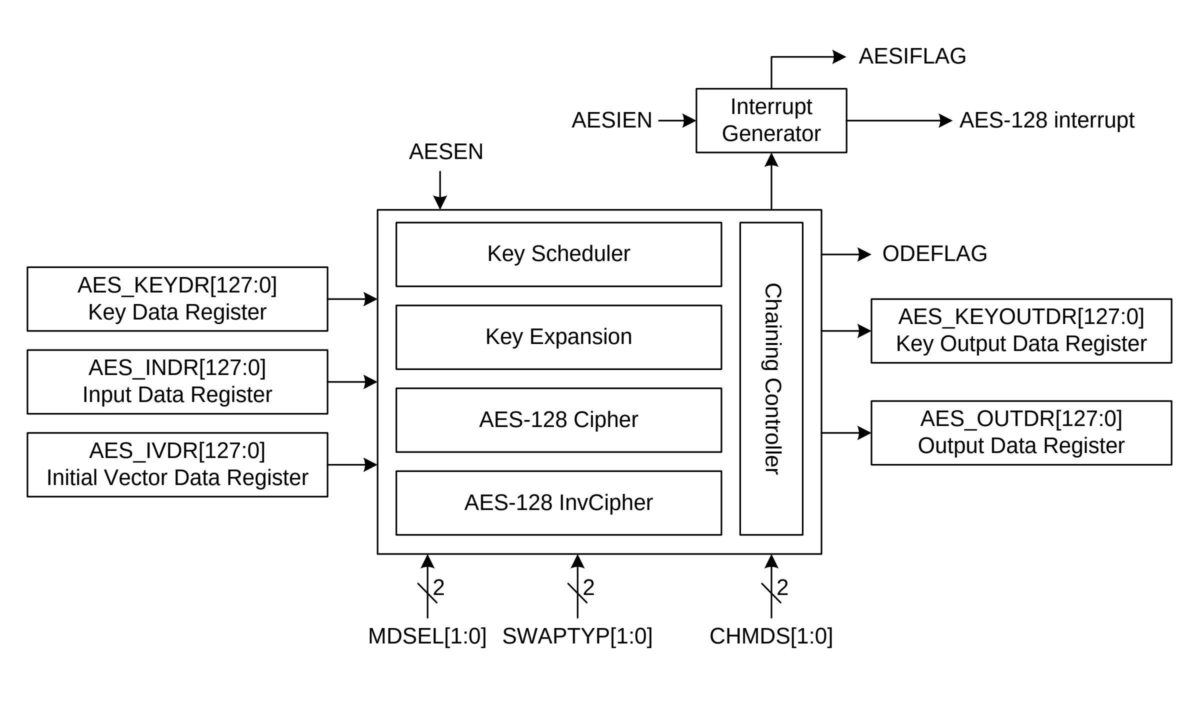
The AES-128 of the A31L21x series has following features:

- Encryption and decryption using AES Rijndael block cipher algorithm
- NIST FIPS 197 compliant implementation of AES encryption/decryption algorithm
- Key scheduler and key expansion for decryption
- 128-bit data block processing
- 128-bit key length
- 211 cycles of PCLK to encrypt/decrypt one 128-bit block
- Electronic codebook (ECB), cipher block chaining (CBC), and counter mode (CTR) supported
- 32-bit input and output buffers
- Automatic data flow control with support of DMA using 2 channels

24.1 AES-128 block diagram

Figure 132 shows a block diagram of the AES-128 block.

Figure 132. AES-128 Block Diagram


AESIFLAG
AESIEN
Interrupt Generator
AES-128 interrupt
AESEN
Key Scheduler
Key Expansion
AES-128 Cipher
AES-128 InvCipher
Chaining Controller
ODEFLAG
AES_KEYDR[127:0] Key Data Register
AES_INDR[127:0] Input Data Register
AES_IVDR[127:0] Initial Vector Data Register
AES_KEYOUTDR[127:0] Key Output Data Register
AES_OUTDR[127:0] Output Data Register
2
2
2
MDSEL[1:0]
SWAPTYP[1:0]
CHMDS[1:0]

24.2 Registers

Base address and register map of the AES-128 block are shown in Table 106 and Table 107.

Table 106. Base Address of AES-128

Name	Base address
AES	0x4000_5E00

Table 107. AES-128 Register Map

Name	Offset	Type	Description	Reset Value
AES_CR	0x0000	RW	AES Control Register	0x00000200
AES_INDR	0x0008	WO	AES Input Data Register	Unknown
AES_OUTDR	0x000C	RO	AES Output Data Register	Unknown
AES_KEYDR1	0x0010	RW	AES Key Data Register 1	Unknown
AES_KEYDR2	0x0014	RW	AES Key Data Register 2	Unknown
AES_KEYDR3	0x0018	RW	AES Key Data Register 3	Unknown
AES_KEYDR4	0x001C	RW	AES Key Data Register 4	Unknown
AES_IVDR1	0x0020	RW	AES Initial Vector Data Register 1	Unknown
AES_IVDR2	0x0024	RW	AES Initial Vector Data Register 2	Unknown
AES_IVDR3	0x0028	RW	AES Initial Vector Data Register 3	Unknown
AES_IVDR4	0x002C	RW	AES Initial Vector Data Register 4	Unknown
AES_INRDR1	0x0030	RO	AES Input Read Data Register 1	Unknown
AES_INRDR2	0x0034	RO	AES Input Read Data Register 2	Unknown
AES_INRDR3	0x0038	RO	AES Input Read Data Register 3	Unknown
AES_INRDR4	0x003C	RO	AES Input Read Data Register 4	Unknown
AES_KEYOUTDR1	0x0040	RO	AES Key Output Data Register 1	Unknown
AES_KEYOUTDR2	0x0044	RO	AES Key Output Data Register 2	Unknown
AES_KEYOUTDR3	0x0048	RO	AES Key Output Data Register 3	Unknown
AES_KEYOUTDR4	0x004C	RO	AES Key Output Data Register 4	Unknown

24.2.1 AES_CR: AES-128 control register

AES_CR register is 32-bit size and accessible in 32/16/8-bit.

AES_CR=0x4000_5E00

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13	12	11 10	9	8	7 6	5 4	3 2	1	0
Reserved	**Reserved**	**AESIEN**	**Reserved**	**ODEFLAG**	**AESIFLAG**	**CHMDS**	**SWAPTYP**	**MDSEL**	**Reserved**	**AESEN**
0x0000	**000**	**0**	**00**	**1**	**0**	**00**	**00**	**00**	**0**	**0**
-	- - -	**RW**	- -	**RO**	**RW**	**RW**	**RW**	**RW**	-	**RW**

12	AESIEN	AES-128 Computation Finish Interrupt Enable bit	
		0	Disable AES-128 interrupt
		1	Enable AES-128 interrupt
9	ODEFLAG	Output Data Empty flag bit	
		0	There is output data to read. The output data register must be read four times before writing to the input data register for new encryption or decryption.
		1	There is no output data to read. That is, the output data register is empty.
8	AESIFLAG	AES-128 Computation Finish Interrupt flag bit	
		0	No request occurred
		1	Request occurred. This bit is cleared to '0' when write '1'.
7 6	CHMDS	Chaining Mode Selection bits	
		00	Electronic code book (ECB)
		01	Cipher block chaining (CBC)
		10	Counter mode (CTR)
		11	Reserved
5 4	SWAPTYP	Input/Output Data Swap Type Selection bits	
		00	No swap
		01	Half word swap
		10	Byte swap
		11	Bit swap
3 2	MDSEL	AES-128 Operation Mode Selection bits	
		00	Select cipher (Encryption) mode
		01	Select inverse cipher (Decryption) mode
		10	Select key expansion function
		11	Reserved
0	AESEN	AES-128 Operation Enable bit	
		0	Disable AES-128
		1	Enable AES-128 (The AES block will be initialized for a new block processing)

NOTES:

1. The corresponding data should be written to the AES_KEYDR and AES_IVDR registers before this bit is set. The AES_IVDR register is not used for ECB mode.
2. The AES-128 operation starts after writing 4-times (MSB first) to the AES_INDR register with 32-bit word unit during this bit is "1b".
3. This bit is cleared to "0b" when a computation is finished on key expansion function.
4. The out data can get by reading the AES_OUTDR register 4-times with word unit.
5. On the key expansion function, the AES-128 operation starts if this bit is set.

24.2.2 AES_INDR: AES-128 input data register

AES_INDR register is 32-bit size and accessible in 32-bit.

AES_INDR=0x4000_5E08

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INDATA																															
0xXXXXXXXX																															
WO																															

31 0	INDATA	AES-128 Input Data (Plain or Cipher text) bits. The input data should be written 4-times (MSB first) with 32-bit word unit.

24.2.3 AES_OUTDR: AES-128 output data register

AES_OUTDR register is 32-bit size and accessible in 32-bit.

AES_OUTDR=0x4000_5E0C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OUTDATA																															
0xXXXXXXXX																															
RO																															

31 0	OUTDATA	AES-128 Output Data (Cipher or Plain text) bits. The output data should be read 4-times (MSB first) with 32-bit word unit.

24.2.4 AES_KEYDR1: AES-128 key data register 1, LSB

AES_KEYDR1 register is 32-bit size and accessible in 32-bit.

AES_KEYDR1=0x4000_5E10

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

31 0	KDATA[31:0]	AES-128 Key Data bits

24.2.5 AES_KEYDR2: AES-128 key data register 2

AES_KEYDR2 register is 32-bit size and accessible in 32-bit.

AES_KEYDR2=0x4000_5E14

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KDATA[63:32]																															
0xXXXXXXXX																															
RW																															

31 0	KDATA[63:32]	AES-128 Key Data bits

24.2.6 AES_KEYDR3: AES-128 key data register 3

AES_KEYDR3 register is 32-bit size and accessible in 32-bit.

AES_KEYDR3=0x4000_5E18

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KDATA[95:64]																															
0xXXXXXXXX																															
RW																															

31 0	KDATA[95:64]	AES-128 Key Data bits

24.2.7 AES_KEYDR4: AES-128 key data register 4, MSB

AES_KEYDR4 register is 32-bit size and accessible in 32-bit.

AES_KEYDR4=0x4000_5E1C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KDATA[127:96]																															
0xXXXXXXXX																															
RW																															

31 0	KDATA[127:96]	AES-128 Key Data bits

24.2.8 AES_IVDR1: AES-128 initial vector data register 1, LSB

AES_IVDR1 register is 32-bit size and accessible in 32-bit.

AES_IVDR1=0x4000_5E20

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

31 0	IVDATA[31:0]	AES-128 Initial Vector Data bits

24.2.9 AES_IVDR2: AES-128 initial vector data register 2

AES_IVDR2 register is 32-bit size and accessible in 32-bit.

AES_IVDR2=0x4000_5E24

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IVDATA[63:32]																															
0xXXXXXXXX																															
RW																															

31 0	IVDATA[63:32]	AES-128 Initial Vector Data bits

24.2.10 AES_IVDR3: AES-128 initial vector data register 3

AES_IVDR3 register is 32-bit size and accessible in 32-bit.

AES_IVDR3=0x4000_5E28

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IVDATA[95:64]																															
0xXXXXXXXX																															
RW																															

31 0	IVDATA[95:64]	AES-128 Initial Vector Data bits

24.2.11 AES_IVDR4: AES-128 initial vector data register 4, MSB

AES_IVDR4 register is 32-bit size and accessible in 32-bit.

AES_IVDR4=0x4000_5E2C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IVDATA[127:96]																															
0xXXXXXXXX																															
RW																															

31 0	IVDATA[127:96]	AES-128 Initial Vector Data bits

24.2.12 AES_INRDR1: AES-128 input read data register 1, LSB

AES_INRDR1 register is 32-bit size and accessible in 32-bit.

AES_INRDR1=0x4000_5E30

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

31 0	INRD[31:0]	AES-128 Input Read Data bits

24.2.13 AES_INRDR2: AES-128 input read data register 2

AES_INRDR2 register is 32-bit size and accessible in 32-bit.

AES_INRDR2=0x4000_5E34

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INRD[63:32]																															
0xXXXXXXXX **RO**																															

31 0	INRD[63:32]	AES-128 Input Read Data bits

24.2.14 AES_INRDR3: AES-128 input read data register 3

AES_INRDR3 register is 32-bit size and accessible in 32-bit.

AES_INRDR3=0x4000_5E38

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INRD[95:64]																															
0xXXXXXXXX																															
RO																															

31 0	INRD[95:64]	AES-128 Input Read Data bits

24.2.15 AES_INRDR4: AES-128 input read data register 4, MSB

AES_INRDR4 register is 32-bit size and accessible in 32-bit.

AES_INRDR4=0x4000_5E3C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INRD[127:96]																															
0xXXXXXXXX																															
RO																															

31 0	INRD[127:96]	AES-128 Input Read Data bits

24.2.16 AES_KEYOUTDR1: AES-128 key output data register 1, LSB

AES_KEYOUTDR1 register is 32-bit size and accessible in 32-bit.

AES_KEYOUTDR1=0x4000_5E40

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="32">KEYOD[31:0]</td></tr>
<tr><td colspan="32">0xXXXXXXXX
RO</td></tr>
</table>

31 0	KEYOD[31:0]	AES-128 Key Output Data bits

24.2.17 AES_KEYOUTDR2: AES-128 key output data register 2

AES_KEYOUTDR2 register is 32-bit size and accessible in 32-bit.

AES_KEYOUTDR2=0x4000_5E44

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="32">KEYOD[63:32]</td></tr>
<tr><td colspan="32">0xXXXXXXXX
RO</td></tr>
</table>

31 0	KEYOD[63:32]	AES-128 Key Output Data bits

24.2.18 AES_KEYOUTDR3: AES-128 key output data register 3

AES_KEYOUTDR3 register is 32-bit size and accessible in 32-bit.

AES_KEYOUTDR3=0x4000_5E48

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="32">KEYOD[95:64]</td></tr>
<tr><td colspan="32">0xXXXXXXXX
RO</td></tr>
</table>

31 0	KEYOD[95:64]	AES-128 Key Output Data bits

24.2.19 AES_KEYOUTDR4: AES-128 key output data register 4, MSB

AES_KEYOUTDR4 register is 32-bit size and accessible in 32-bit.

AES_KEYOUTDR4=0x4000_5E4C

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="32">KEYOD[127:96]</td></tr>
<tr><td colspan="32">0xXXXXXXXX
RO</td></tr>
</table>

31 0	KEYOD[127:96]	AES-128 Key Output Data bits

24.3 Functional description

The AES-128 processes a data block of 128-bits using a 128-bit long key. It supports Electronic CodeBook (ECB), Cipher Block Chaining (CBC), and Counter mode (CTR) by hardware.

The AES-128 provides 3 operating modes: encryption, decryption, and key expansion function. The operating mode, chaining mode, and data swap type can be changed only when the AES-128 is disabled (the AESEN bit of AES_CR register is “0b”).

The AES_KEYDR and AES_IVDR registers for the AES-128 should be set before the AES-128 is enabled if necessary.

24.3.1 AES chaining algorithms

There are three chaining algorithms performed by AES hardware, as follows:

- Electronic CodeBook (ECB)
- Cipher Block Chaining (CBC)
- Counter Mode (CTR)

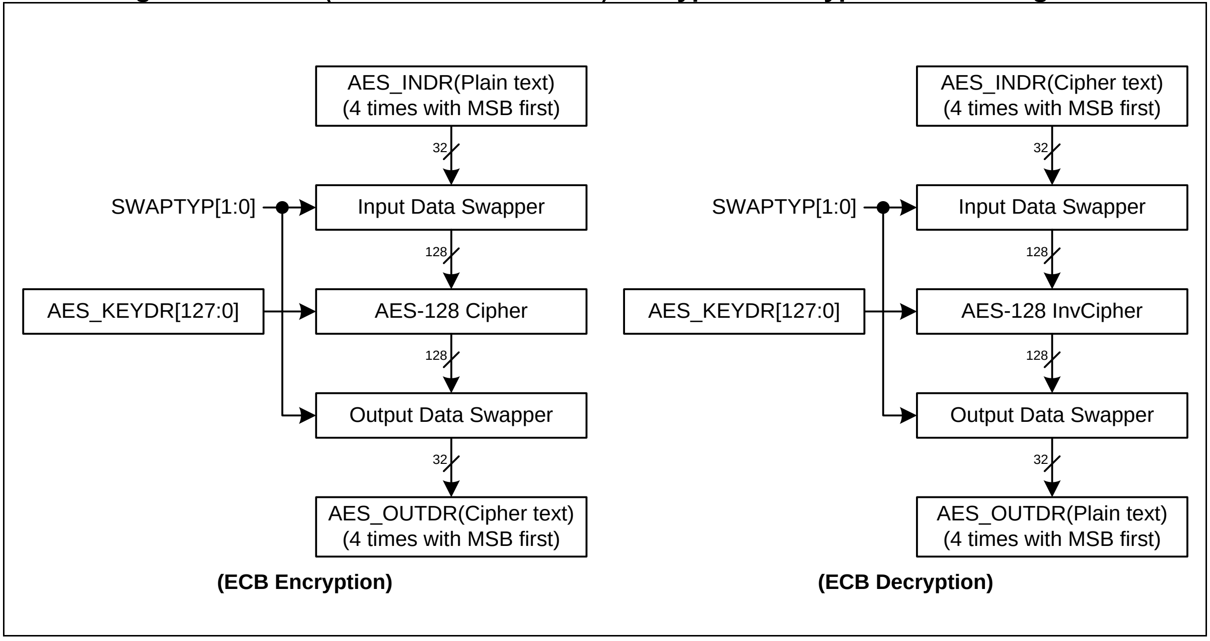
Each algorithm can be selected by setting the CHMDS[1:0] bits of the AES_CR register when the AES-128 is disabled.

24.3.1.1 Electronic CodeBook (ECB)

The ECB does not use the AES_IVDR register.

Figure 133 describes the principle of the ECB algorithm for encryption and decryption.

Figure 133. ECB (Electronic CodeBook) Encryption/Decryption Block Diagram

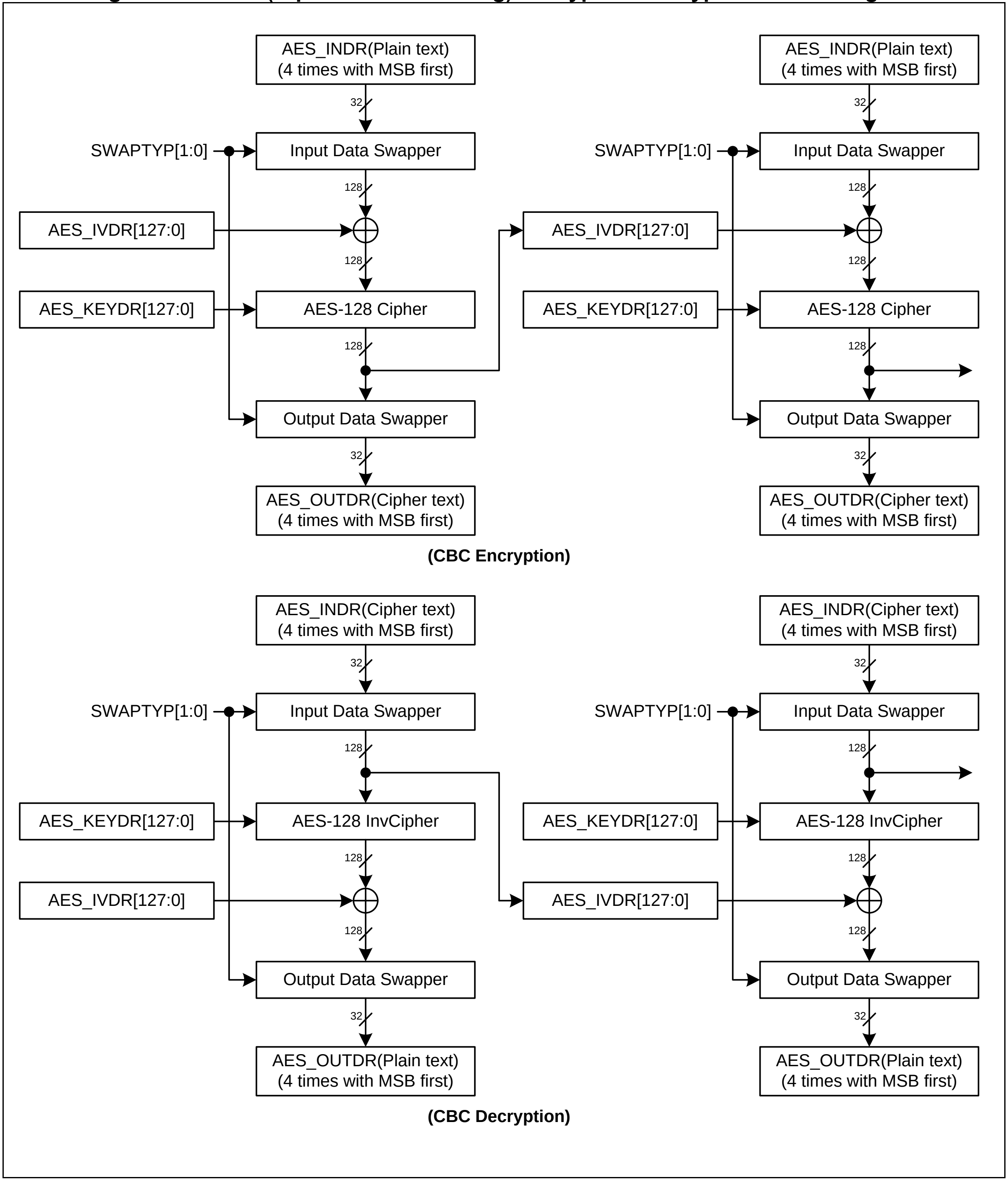


24.3.1.2 Cipher block chaining (CBC)

Each plain text is XORed with the previous cipher text before encryption and the first block processing uses an initialization vector (AES_IVDR register).

Figure 134 describes the principle of the CBC algorithm for encryption and decryption.

Figure 134. CBC (Cipher block chaining) Encryption/Decryption Block Diagram



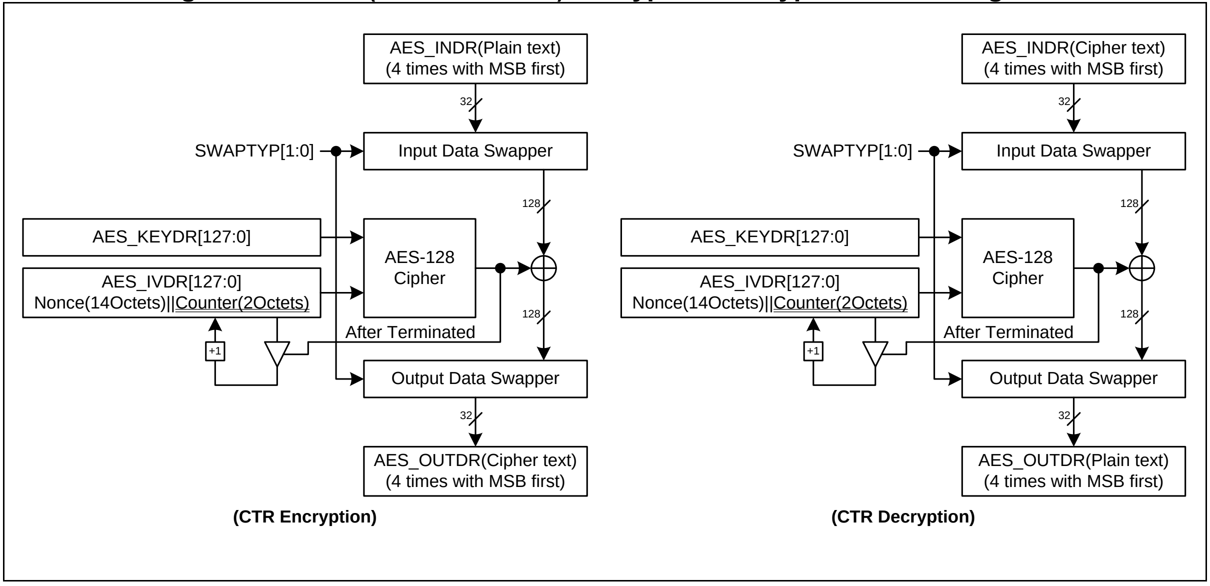
24.3.1.3 Counter mode (CTR)

In counter mode, a 16-bit counter in addition to a nonce is used for the XOR operation with the plain text or cipher text.

Figure 135 describes the principle of the CBC algorithm for encryption and decryption. In the figure, the nonce value and 16-bit counter are applied to the AES_IVDR register and organized like nonce[127:16] + counter[15:0].

The counter is automatically incremented from the initial value for each block to be processed. The key value for encryption and decryption is the same in the CTR mode. Therefore, the key expansion function is not necessary.

Figure 135. CTR (Counter mode) Encryption/Decryption Block Diagram



24.3.2 Data swap

The system memory organization of the A31L21x device is little-endian. That is, the less-significant data occupies the lowest address location. The data swap function will help in handling input and output data.

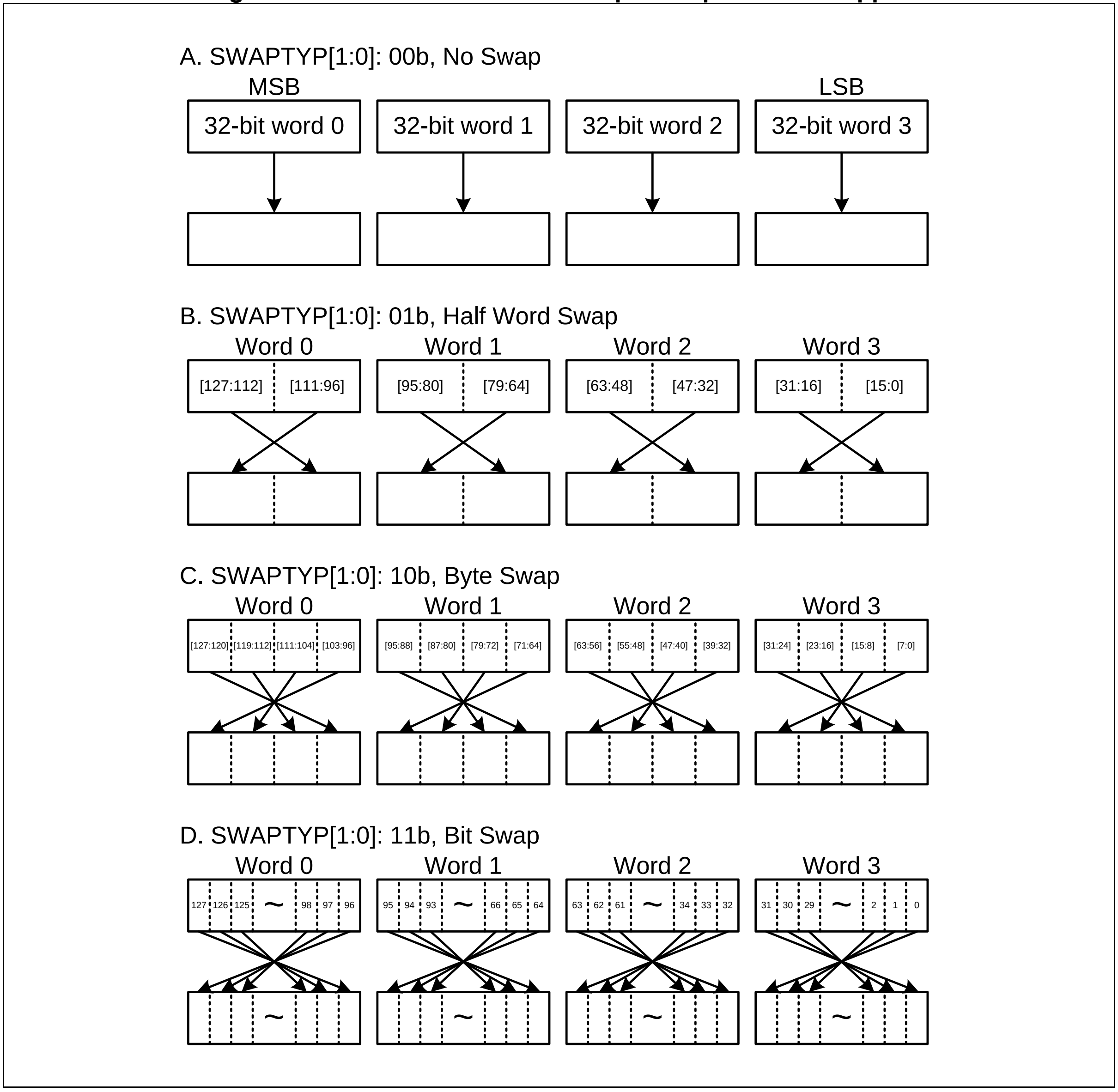
There are three types of data swap operations:

- Half-word swap
- Byte swap
- Bit swap

The data swap operation is applied to the AES_INDR and AES_OUTDR registers only.

Figure 136 shows the 3 types data swapping.

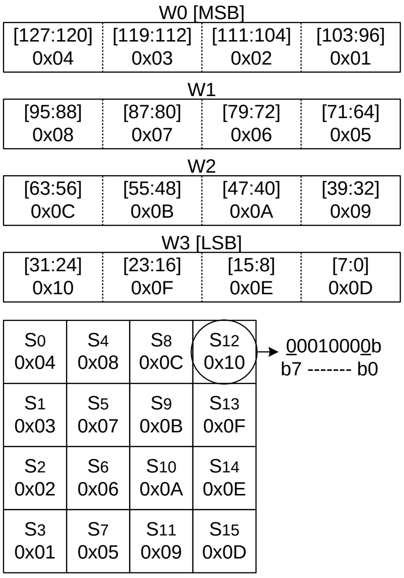
Figure 136. Process Structure of Input/Output Data Swapper



Since the system memory is little-endian, special attention is required when writing data to registers or reading from registers.

Figure 137 shows how data is written to the matrix.

Figure 137. Data Written in the Matrix



W0 [MSB]

[127:120]	[119:112]	[111:104]	[103:96]
0x04	0x03	0x02	0x01

W1

[95:88]	[87:80]	[79:72]	[71:64]
0x08	0x07	0x06	0x05

W2

[63:56]	[55:48]	[47:40]	[39:32]
0x0C	0x0B	0x0A	0x09

W3 [LSB]

[31:24]	[23:16]	[15:8]	[7:0]
0x10	0x0F	0x0E	0x0D

S0 0x04	S4 0x08	S8 0x0C	S12 0x10
S1 0x03	S5 0x07	S9 0x0B	S13 0x0F
S2 0x02	S6 0x06	S10 0x0A	S14 0x0E
S3 0x01	S7 0x05	S11 0x09	S15 0x0D

24.3.3 Operation modes

There are three operation modes for AES-128 as follows:

- Cipher (Encryption) mode
- Inverse cipher (Decryption) mode
- Key expansion function

Each operation mode can be selected by setting the MDSEL[1:0] bits of the AES_CR register, when the AES-128 is disabled.

24.3.3.1 Cipher mode (Encryption)

1. Disable the AES-128 by resetting the AESEN bit of AES_CR register.
2. Select cipher mode by writing “00b” to the MDSEL[1:0] bits of AES_CR register and select the chaining mode to be applied using the CHMDS[1:0] bits of AES_CR register.
3. Write 128-bit encryption key to the AES_KEYDR register and an initial vector data to the AES_IVDR register if needed. The AES_IVDR register is not used for ECB mode.
4. Enable the AES-128 by setting the AESEN bit of AES_CR register.
5. Write the plain text (MSB first) to the AES_INDR register 4 times.
6. Wait until the AESIFLAG bit is set in the AES_CR register.
7. Reads the AES_OUTDR register 4 times to get the cipher text (MSB first).
8. Repeat steps 5, 6, and 7 to process all the blocks with the same encryption key.

24.3.3.2 Inverse cipher mode (Decryption)

1. Disable the AES-128 by resetting the AESEN bit of AES_CR register.
2. Select inverse cipher mode by writing “01b” to the MDSEL[1:0] bits of AES_CR register and select the chaining mode to be applied using the CHMDS[1:0] bits of AES_CR register.
3. Write 128-bit decryption key to the AES_KEYDR register and an initial vector data to the AES_IVDR register if needed. The AES_IVDR register is not used for ECB mode.
4. Enable the AES-128 by setting the AESEN bit of AES_CR register.
5. Write the cipher text (MSB first) to the AES_INDR register 4 times.
6. Wait until the AESIFLAG bit is set in the AES_CR register.
7. Reads the AES_OUTDR register 4 times to get the plain text (MSB first).
8. Repeat steps 5, 6, and 7 to process all the blocks with the same decryption key.

24.3.3.3 Key expansion function

1. Disable the AES-128 by resetting the AESEN bit of AES_CR register.
2. Select key expansion function by writing “10b” to the MDSEL[1:0] bits of AES_CR register. The CHMDS[1:0] bits of AES_CR register are not used in this function.
3. Write 128-bit encryption key to the AES_KEYDR register to obtain the decryption key. The AES_IVDR register is not used for this function.
4. Enable the AES-128 by setting the AESEN bit of AES_CR register.
5. Write the cipher text (MSB first) to the AES_INDR register 4 times.
6. Wait until the AESIFLAG bit is set in the AES_CR register. The AESEN bit is cleared to “0b” after calculation by hardware.
7. Read the decryption key from the AES_KEYOUTDR register.

25 True Random Number Generator (TRNG)

True Random Number Generator (TRNG) in the A31L21x series continuously provides 32-bit entropy samples based on analog noise sources. It creates non-deterministic randomness that cannot be predicted by any physical source.

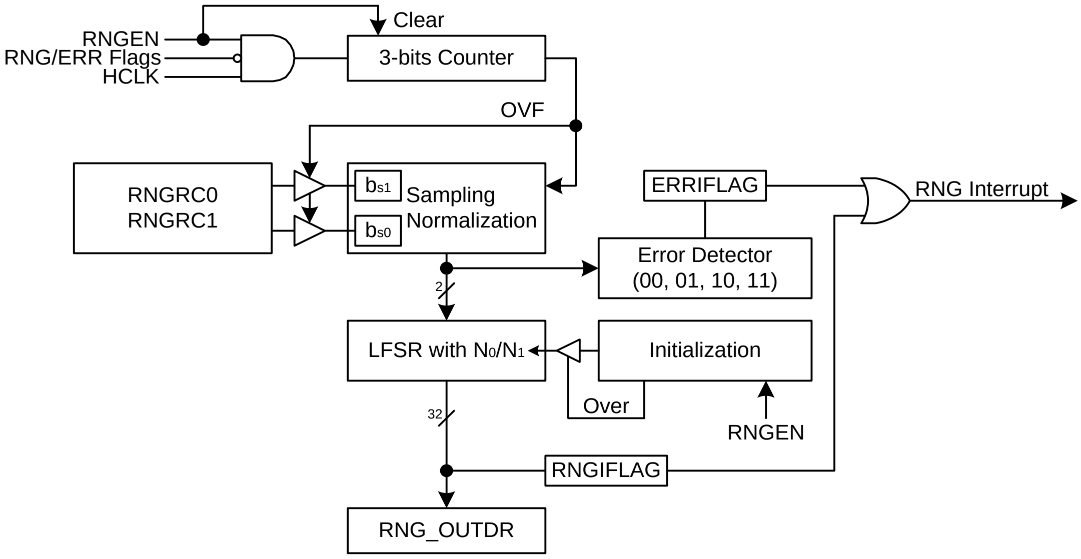
The TRNG in the A31L21x series has following features:

- 32-bit non-deterministic random numbers produced by an analog entropy source post-processed with the linear-feedback shift registers (LFSR)
- Generation of a 32-bit random sample every 20 – 50 RNG clock cycle
- An error detector checking patterns that are repeated more than 16 times in a row
- A disable function to reduce power consumption

25.1 RNG block diagram

Figure 138 shows a block diagram of the RNG block.

Figure 138. RNG Block Diagram



NOTE: The error detector checks whether the same pattern (00b, 01b, 10b, or 11b) is repeated 16 times or more continuously.

25.2 Registers

Base address and register map of the RNG block are shown in Table 108 and Table 109.

Table 108. Base Address of RNG

Name	Base address
RNG	0x4000_5F00

Table 109. RNG Register Map

Name	Offset	Type	Description	Reset value
RNG_CR	0x0000	RW	RNG Control Register	0x00000000
RNG_OUTDR	0x0004	RO	RNG Output Data Register	Unknown

25.2.1 RNG_CR: random number generator control register

RNG_CR register is 32-bit size and accessible in 32/16/8-bit.

RNG_CR=0x4000_5F00

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8	7	6	5	4	3	2 1	0
Reserved	RNGIEN	ERRIEN	Reserved	RNGIFLAG	ERRIFLAG	Reserved	RNGEN
0x000000	0	0	0	0	0	00	0
-	RW	RW	-	RW	RW	-	RW

7	RNGIEN	Random Data Ready Interrupt Enable bit	
		0	Disable random number generator interrupt
		1	Enable random number generator interrupt
6	ERRIEN	Error Detection Interrupt Enable bit	
		0	Disable error detection interrupt
		1	Enable error detection interrupt
4	RNGIFLAG	Random Data Ready Interrupt flag bit	
		0	No request occurred
		1	Request occurred. This bit is cleared to ‘0’ when write ‘1’.
		NOTES: 1. This bit will be set by hardware when new random data are ready. 2. When this bit is ‘1’, the RNG block stops temporarily operation.	
3	ERRIFLAG	Error Detection Interrupt flag bit	
		0	No request occurred (No effect when write ‘0’ or ‘1’)
		1	Request occurred. This bit is cleared to ‘0’ when write ‘1’.
		NOTE: If an error occurs, the RNG_OUTDR register must not be used, s/w should clear this bit to recover, and the RNGEN bit must be cleared and set to reinitialize and restart the RNG block.	
0	RNGEN	Random Number Generator Enable bit	
		0	Disable random number generator
		1	Enable random number generator (Clear 3-bits counter and start)

25.2.2 RNG_OUTDR: random number generator output data register

RNG_OUTDR register is 32-bit size and accessible in 32/16/8-bit.

RNG_OUTDR=0x4000_5F04

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="32">OUTDATA</td></tr>
<tr><td colspan="32">0xXXXXXXXX</td></tr>
<tr><td colspan="32">RO</td></tr>
</table>

31 0	OUTDATA	Random Number Output Data bits.

25.3 Functional description

The random number generator produces non-deterministic random numbers almost at regular intervals using internal RC oscillators as noise source. The non-deterministic random numbers can be used to generate true random numbers with CRC function.

It has free-running noise sources, normalization block, linear-feedback shift registers (LFSR), and error detection logic.

25.3.1 Noise source

The RNG block uses free-running internal RC oscillators as noise source. The RC oscillators are independent to the system clock frequency. When the RNGEN bit of the RNG_CR register is cleared to disable the RNG block, the oscillators are automatically stopped to reduce current consumption.

25.3.2 Post processing

The sampled bits from noise sources are passed to the normalization block. The normalization block filters to reduce the bias to an acceptable level. Finally, a linear feedback shift register (LFSR) performs a whitening process to produce 32-bit bit-strings. The 32-bit bit-strings may be used as the input of CRC function to generate true random number. When random data is generated, the RNGIFLAG bit of the RNG_CR register is set to “1b” and the RNG block is temporarily stopped. So, the RNGIFLAG bit must be cleared to create new data.

25.3.3 Error detection

The error detection logic always checks the output bit-strings to see if the same pattern (00b, 01b, 10b, and 11b) is repeated more than 16 times. If error is detected, the ERRIFLAG bit of RNG_CR register is set to “1b”. In this case, after clearing the ERRIFLAG and RNGEN bits of the RNG_CR register, the RNGEN bit must be set to 1 again for reinitialization and restart.

26 Temperature Sensor (TS)

The Temperature Sensor (TS) is a ring-oscillator type, and can be used to measure the junction temperature of the device. The nominal frequency at 30℃ is about 0.9MHz and it varies from 0.6MHz to 1.25MHz as the temperature changes from -20℃ to +105℃.

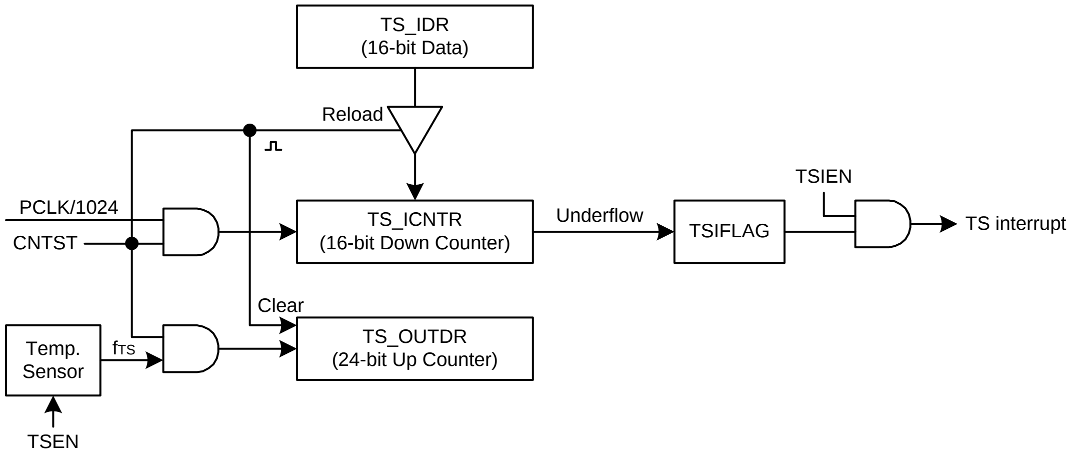
The TS of the A31L21x series has following features:

- -20℃ to +105℃ wide range of operating temperature
- A down counter at 16-bit intervals to count the frequency of the TS
- A 24-bit data register to store the count value of the temperature sensor frequency

26.1 TS block diagram

Figure 139 shows a block diagram of the temperature sensor block.

Figure 139. TS Block Diagram



NOTE: The CNTST bit will be cleared to “0b” after the underflow of TS_ICNTR.

26.2 Registers

Base address and register map of the temperature sensor block are shown in Table 110 and Table 111.

Table 110. Base Address of TS

Name	Base address
TS	0x4000_5F80

Table 111. TS Register Map

Name	Offset	Type	Description	Reset value
TS_CR	0x0000	RW	Temperature Sensor Control Register	0x00000000
TS_IDR	0x0004	RW	Temperature Sensor Interval Data Register	0x0000FFFF
TS_ICNTR	0x0008	RO	Temperature Sensor Interval Counter Register	0x0000FFFF
TS_OUTDR	0x000C	RO	Temperature Sensor Output Data Register	0x00000000

26.2.1 TS_CR: Temperature Sensor control register

TS_CR register is 32-bit size and accessible in 32/16/8-bit.

TS_CR=0x4000_5F80

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
Reserved	TSIEN	Reserved		TSIFLAG	Reserved		CNTST	TSEN
0x000000	0	00		0	00		0	0
-	RW	-	-	RW	-	-	RW	RW

7	TSIEN	Temperature Sensor Interrupt Enable
		0 Disable interrupt
		1 Enable interrupt
4	TSIFLAG	Temperature Sensor Interrupt flag
		0 No request occurred
		1 Request occurred. This bit is cleared to ‘0’ when write ‘1’.
1	CNTST	Counting Start
		0 Stop counting
		1 Counting start after “reload data to TS_ICNTR and clear TS_OUTDR”
		NOTE: This bit will be cleared to “0b” after the underflow of TS_ICNTR register.
0	TSEN	Temperature Sensor Enable
		0 Disable temperature sensor
		1 Enable temperature sensor

26.2.2 TS_IDR: Temperature Sensor interval data register

TS_IDR register is 32-bit size and accessible in 32/16/8-bit.

TS_IDR=0x4000_5F84

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																IDATA															
0x0000																0xFFFF															
-																RW															

15 0	IDATA	Temperature Sensor Interval Data bits.

26.2.3 TS_ICNTR: Temperature Sensor interval counter register

TS_ICNTR register is 32-bit size and accessible in 32/16/8-bit.

TS_ICNTR=0x4000_5F88

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																ICNT															
0x0000																0xFFFF															
-																RO															

15 0	ICNT	Temperature Sensor Interval Counter bits.

26.2.4 TS_OUTDR: Temperature Sensor output data register

TS_OUTDR register is 32-bit size and accessible in 32/16/8-bit.

TS_OUTDR=0x4000_5F8C

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="8">Reserved</td><td colspan="24">OUTDATA</td></tr>
<tr><td colspan="8">0x0000</td><td colspan="24">0x000000</td></tr>
<tr><td colspan="8">-</td><td colspan="24">RO</td></tr>
</table>

23 0	OUTDATA	Temperature Sensor Output Data bits.

26.3 Functional description

26.3.1 Ring-oscillator of Temperature Sensor

The ring-oscillator of the TS needs a maximum stabilization time of about 500usec. Its frequency ranges from about 0.6MHz to 1.25MHz depending on temperature. The frequency variation is about 3.2kHz per Celsius degree and its frequency is directly proportional to temperature.

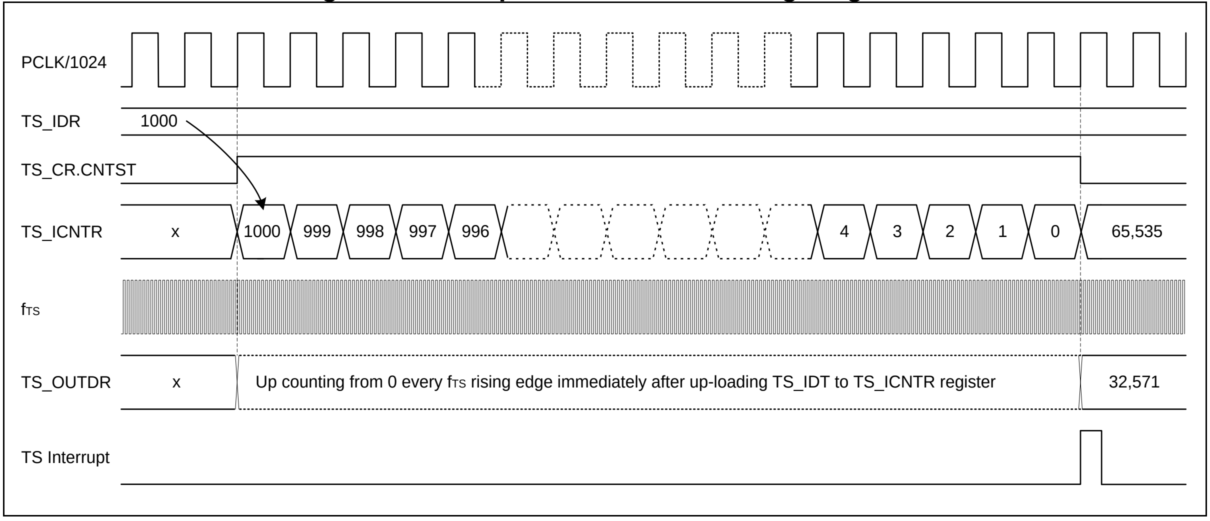
26.3.2 Frequency counting

The frequency of the TS can be measured indirectly by using the system clock. The TS block provides the 16-bit interval data and 24-bit output data registers to count the number of temperature sensor clocks during a specific period.

The 16-bit interval data register, TS_IDR, sets the measurement period.

Figure 140 shows temperature sensor timing diagram.

Figure 140. Temperature Sensor Timing Diagram


PCLK/1024
TS_IDR
1000
TS_CR.CNTST
TS_ICNTR
x
1000
999
998
997
996
4
3
2
1
0
65,535
f_{TS}
TS_OUTDR
x
Up counting from 0 every f_{TS} rising edge immediately after up-loading TS_IDT to TS_ICNTR register
32,571
TS Interrupt

26.3.3 Temperature calculation

The Configure Option Page 0 provides the frequency values of the TS measured by the manufacturer at 30[℃], 85[℃] and 105[℃], respectively.

Table 112 shows the corresponding registers in the Configure Option Page 0.

Table 112. TS Output Frequency Registers in Configure Option Page 0

Name	Address	Type	Description
TS_FREQ_T30	0x1FFFF048	RO	Temperature Sensor Output Frequency acquired at 30[℃] [Hz]
TS_FREQ_T85	0x1FFFF04C	RO	Temperature Sensor Output Frequency acquired at 85[℃] [Hz] (Commercial grade)
TS_FREQ_T105	0x1FFFF06C	RO	Temperature Sensor Output Frequency acquired at 105[℃] [Hz] (Industrial grade)

The temperature can be calculated using the formula below:

$$Temperature = \frac{F(T) - F(30)}{\Delta F} + 30\ [℃]$$

Where,

$$\Delta F = \frac{F(T2) - F(T1)}{T2 - T1}$$

T1 = 30℃, T2 = 85℃(Commercial grade) or 105℃(Industrial grade)

F(T1) [kHz] is the temperature sensor output frequency acquired at 30℃

F(T2) [kHz] is the temperature sensor output frequency acquired at 85℃(Commercial grade) or 105℃(Industrial grade)

F(T) [kHz] is the temperature sensor output frequency acquired at an arbitrary temperature.

Example

- TS_FREQ_T30 of Configure Option Page 0 = 880,000 (880 kHz)
- TS_FREQ_T105 of Configure Option Page 0= 1,130,000 (1,130 kHz)
- If the acquired frequency at arbitrary temperature is about 730,000 (730 kHz)
- ∆F = (1,130 – 880)/(105 – 30) = 250/75 = 3.3 [kHz]
- Measured temperature = (730.000 – 880)/3.3 + 30 = -45.4 +30 ≈ -15.4 [℃]
- Therefore, the result is about -15.4℃.

27 Direct Memory Access (DMA) Controller

Direct Memory Access (DMA) controller transfers data without s/w assert. The DMA has 7 channels, and the DMA controller has four registers such as a control register (DMACHn_CR), a peripheral address register (DMACHn_PAR), a memory address register (DMACHn_MAR), and an interrupt enable and status register (DMACHn_IESR).

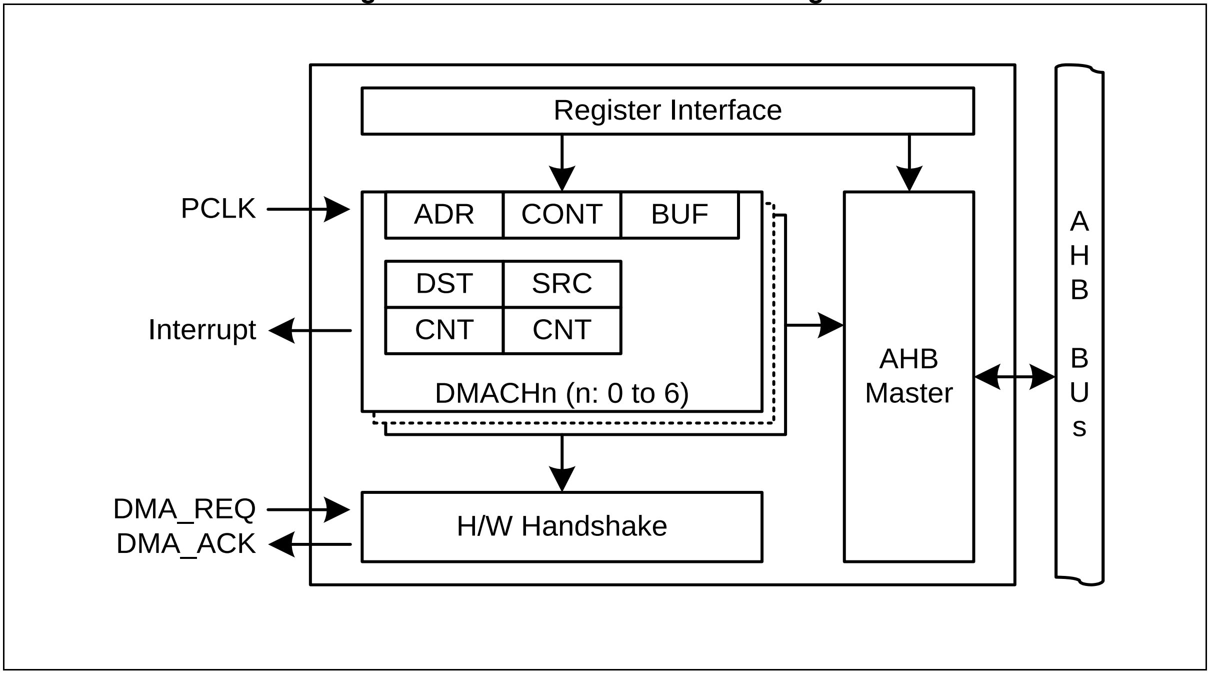
The DMA controller of the A31L21x series features the followings:

- 7 channels supported
- 8/16/32-bit data size supported
- Transfer memory-to-peripheral
- Transfer peripheral-to-memory

27.1 DMA controller block diagram

Figure 141 shows a block diagram of the DMA controller block.

Figure 141. DMA Controller Block Diagram



27.2 Registers

Base address and register map of the DMA channel 0/1/2/3/4/5/6 are shown in Table 113 and Table 114.

Table 113. Base Address of DMA Channel 0/1/2/3/4/5/6

Name	Base address	Size	Description
DMACH0	0x4000_5D00	32	DMA Channel 0
DMACH1	0x4000_5D20	32	DMA Channel 1
DMACH2	0x4000_5D40	32	DMA Channel 2
DMACH3	0x4000_5D60	32	DMA Channel 3
DMACH4	0x4000_5D80	32	DMA Channel 4
DMACH5	0x4000_5DA0	32	DMA Channel 5
DMACH6	0x4000_5DC0	32	DMA Channel 6

Table 114. DMA Channel n Register Map (n = 0, 1, 2, 3, 4, 5, and 6)

Name	Offset	Type	Description	Reset value
DMACHn_CR	0x0000	RW	DMA Channel n Control Register	0x00000000
DMACHn_IESR	0x0004	RW	DMA Channel n Interrupt Enable and Status Register	0x00000000
DMACHn_PAR	0x0008	RW	DMA Channel n Peripheral Address Register	0x40000000
DMACHn_MAR	0x000C	RW	DMA Channel n Memory Address Register	0x20000000

27.2.1 DMACHn_CR: DMA channel n control register

DMACHn_CR register is 32-bit size and accessible in 32/16/8-bit. (n = 0, 1, 2, 3, 4, 5, and 6)

DMACH0_CR=0x4000_5D00, DMACH1_CR=0x4000_5D20, DMACH2_CR=0x4000_5D40
DMACH3_CR=0x4000_5D60, DMACH4_CR=0x4000_5D80
DMACH5_CR=0x4000_5DA0, DMACH6_CR=0x4000_5DC0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				TRANSCNT												ERFGSTP	Reserved	PERSEL						Reserved				SIZE		DIR	CHnEN
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-	-	-	-	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	-	RW	RW	RW	RW	RW	RW	-	-	-	-	RW	RW	RW	RW

27 16	TRANSCNT	The number of times to transfer. These bits should be written out except ‘0’ before the DMA channel is activated.	
		0x000	All transmissions have been completed.
		Others	The number of times left to transfer. The value is decremented by ‘1’ every transfer and the transfer ends when it reaches zero.
15	ERFGSTP	Error Flag Stop bit. This bit is used to stop the DMA transfer when an error flag of a selected peripheral is set to ‘1’.	
		0	Disable DMA stop function by an error of peripheral
		1	Enable DMA stop function by an error of peripheral
13 8	PERSEL	Peripheral Selection.	
		000000	Channel idle
		000001	ADC
		000010	SPI0 Rx
		000011	SPI0 Tx
		000100	SPI1 Rx
		000101	SPI1 Tx
		000110	USART10 Rx
		000111	USART10 Tx
		001000	I2C0 Rx
		001001	I2C0 Tx
		001010	I2C1 Rx
		001011	I2C1 Tx
		001100	UART0 Rx
		001101	UART0 Tx
		010000	LPUART0 Rx
		010001	LPUART0 Tx
		010010	SC0 Rx
		010011	SC0 Tx
		010110	SPI2 Rx
		010111	SPI2 Tx
		011000	SPI3 Rx
		011001	SPI3 Tx
		011010	USART11 Rx
		011011	USART11 Tx
		011100	I2C2 Rx
		011101	I2C2 Tx
		011110	Read data from AES_OUTDR of AES 128-bit
		011111	Write data to AES_INDR of AES 128-bit
		100000	LPUART1 Rx
		100001	LPUART1 Tx
		Others	Reserved
3	SIZE	Transfer Size Selection.	

2		00	8-bits.
		01	16-bits.
		10	32-bits.
		11	Not used.
1	DIR	Transfer Direction.	
		0	Transfer is from memory to peripheral.
		1	Transfer is from peripheral to memory.
0	CHnEN	DMA Channel Enable. This bit is automatically cleared to '0' immediately after transfer completion or error.	
		0	Disable channel n.
		1	Enable channel n.
		NOTE: All DMA channels must be disabled by S/W before entering SLEEP and DEEP SLEEP mode.	

NOTES:

1. When ADC is DMA transfer.
   — The EOCIFLAG bit of ADC_IESR register is the request signal of DMA transfer.
   — If the OVRUNIFLAG bit of ADC_IESR register is set during transfer on the ERFGSTP = 1, the corresponding transfer error interrupt flag bit is set and the transfer will be stopped.
2. When SPIn is DMA transfer. Where n = 0, 1, 2, and 3
   — The SPInIFLAG bit of SPIn_SR register is the request signal of DMA transfer.
3. When USARTn is DMA transfer. Where n = 10 and 11
   — The DREn and RXCn bits of USARTn_ST register are the request signal for Tx and Rx of DMA transfer.
   — If the DORn, FEn, and PEn bits of USARTn_ST register are set during transfer on the ERFGSTP = 1, the corresponding transfer error interrupt flag bit is set and the transfer will be stopped.
4. When I2Cn is DMA transfer. Where n = 0, 1, and 2
   — The values of I2Cn_ST register are the request signal for Tx and Rx of DMA transfer.
   — Abbreviations
   > "SnDA": Start and Device address.
   > "rSnDA": Restart and Device address.
   > "CSnSP": Clear status and stop.
   > "CS": Clear status.
   > N is the number of bytes to be received or transmitted.
   > On the master Tx: "SnDA" by S/W + Transmit(N) by DMA + "CSnSP" by S/W
   > On the master Rx: Up to "rSnDA" by S/W + Receive (N-1) by DMA + Receive(1+NACK) and "CSnSP" by S/W
   > On the slave Tx: Up to "rSnDA" by S/W + Transmit (N) by DMA + "CS" by S/W
   > On the slave Rx: "SnDA" by S/W + Receive(N) by DMA + "CS" by S/W
   > The corresponding DMA channel should be enabled immediately before "SnDA" on a master Tx or slave Rx.
   > The corresponding DMA channel should be enabled immediately before "rSnDA" on a master Rx or slave Tx.
5. When UARTn is DMA transfer. Where n: 0
   — The flags of UARTn_LSR register are the request signal for Tx and Rx of DMA transfer.
   — If an error occurs during transfer on the ERFGSTP = 1, the corresponding transfer error interrupt flag bit is set and the transfer will be stopped.
6. When LPUARTn is DMA transfer. Where n: 0 and 1
   — The TXCIFLAGn and RXCIFLAGn bits of LPUARTn_IFSR register are the request signal for Tx and Rx of DMA.
   — If the DORn, FEn, and PEn bits of LPUARTn_IFSR register are set during transfer on the ERFGSTP = 1, the corresponding transfer error interrupt flag bit is set and the transfer will be stopped.
7. When SCn is DMA transfer. Where n: 0
   — The TXCIFLAGn and RXCIFLAGn bits of SCn_IFSR register are the request signal for Tx and Rx of DMA.
   — If the DORn, FEn, PEn, and TRYERIFGn bits of SCn_IFSR register are set during transfer on the ERFGSTP = 1, the corresponding transfer error interrupt flag bit is set and the transfer will be stopped.
8. When AES 128-bit is DMA transfer.
   — The AESIFLAG and ODEFLAG bits of AES_CR register are the request signal of DMA transfer.

27.2.2 DMACHn_IESR: DMA channel n interrupt enable and status register

DMACHn_IESR register is 32-bit size and accessible in 32/16/8-bit. (n = 0, 1, 2, 3, 4, 5, and 6)

DMACH0_IESR=0x4000_5D04, DMACH1_IESR=0x4000_5D24, DMACH2_IESR=0x4000_5D44
DMACH3_IESR=0x4000_5D64, DMACH4_IESR=0x4000_5D84
DMACH5_IESR=0x4000_5DA4, DMACH6_IESR=0x4000_5DC4

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																								Reserved		TRERIENn	TRCIENn	Reserved		TRERIFGn	TRCIFGn
0x000000																								0	0	0	0	0	0	0	0
–																								-	-	RW	RW	-	-	RW	RW

5	TRERIENn	Transfer Error Interrupt Enable.	
		0	Disable transfer error interrupt.
		1	Enable transfer error interrupt.
4	TRCIENn	Transfer Complete Interrupt Enable.	
		0	Disable transfer complete interrupt.
		1	Enable transfer complete interrupt.
1	TRERIFGn	Transfer Error Interrupt Flag bit. This bit is set to “1b” when an error occurs on the transfer.	
		0	No request occurred.
		1	Request occurred, This bit is cleared to ‘0’ when write ‘1’.
		NOTE:	This bit will be set by an error occur during the ERFGSTP bit of DMACHn_CR register is set to 1.
0	TRCIFGn	Transfer Complete Interrupt Flag. This bit is set to “1b” when the transfer is finished.	
		0	No request occurred.
		1	Request occurred, This bit is cleared to ‘0’ when write ‘1’.
		NOTE:	On the DIR bit of DMACHn_CR is “0b” (Tx of an interface), this bit is set when the DMA memory to transmit is empty. That is, since the last data is being transmitted, the next Tx should be started after the corresponding Tx is completed. Check the corresponding flag of Tx interrupt to see if the last Tx is complete.

27.2.3 DMACHn_PAR: DMA channel n peripheral address register

DMACHn_PAR register is 32-bit size and accessible in 32/16/8-bit. (n = 0, 1, 2, 3, 4, 5, and 6)

DMACH0_PAR=0x4000_5D08, DMACH1_PAR=0x4000_5D28, DMACH2_PAR=0x4000_5D48
DMACH3_PAR=0x4000_5D68, DMACH4_PAR=0x4000_5D88
DMACH5_PAR=0x4000_5DA8, DMACH6_PAR=0x4000_5DC8

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
PBADR	POADR
0x4000	0x0000
RO	RW

31 16	PBADR	Peripheral Base Address. This is fixed at 0x4000 for APB peripherals.
15 0	POADR	Peripheral Offset Address. If the DIR bit is “0b”, this is the destination offset address of data transfer. If the DIR bit is “1b”, this is the source offset address of data transfer.

27.2.4 DMACHn_MAR: DMA channel n memory address register

DMACHn_MAR register is 32-bit size and accessible in 32/16/8-bit. (n = 0, 1, 2, 3, 4, 5, and 6)

DMACH0_MAR=0x4000_5D0C, DMACH1_MAR=0x4000_5D2C, DMACH2_MAR=0x4000_5D4C
DMACH3_MAR=0x4000_5D6C, DMACH4_MAR=0x4000_5D8C
DMACH5_MAR=0x4000_5DAC, DMACH6_MAR=0x4000_5DCC

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15	14 13 12	11 10 9 8 7 6 5 4 3 2 1 0
MBADR	MAR15	MOADR	
0x2000	0	000	0x000
RO	RO	RW	RW

31 16	MBADR	Memory Base Address. These bits are fixed with 0x2000.
15	MAR15	This bit is fixed with “0b”
14 0	MOADR	Memory Offset Address. This is the address of the memory area from/to which the data will be read/written. This register will be incremented by $2^{SIZE[1:0]}$ for every transfer. When SIZE[1:0] is “01b”, the MOADR[0] bit is ignored. When SIZE[1:0] is “10b”, the MOADR[1:0] bits are ignored. If the DIR bit is “0b”, this is the source memory address of data transfer. If the DIR bit is “1b”, this is the destination memory address of data transfer.

27.3 Functional description

The DMA controller performs direct memory transfer by sharing the system bus with CPU core. The system bus is shared by 2 AHB masters following the round-robin priority strategy. So the DMA controller can share the half of system bandwidth.

The DMA controller can be triggered only peripheral request. When a peripheral request the transfer to the DMA controller, related channel is activate and access the bus to transfer requested data from memory to peripheral data buffer or from peripheral data buffer to memory space.

- User set both of peripheral address and memory address
- User configure DMA operation mode and transfer count.
- User enable DMA channel
- DMA request is occurred from peripheral.
- DMA activate channel which was requested
- DMA read data from source address and save it internal buffer.
- DMA write the buffered data to destination address.
- Transfer count number is decreased by 1.
- When Transfer count is 0, EOT flag is set and notice to peripheral to issue the interrupt

Figure 142. DMA Controller and Operation Channel

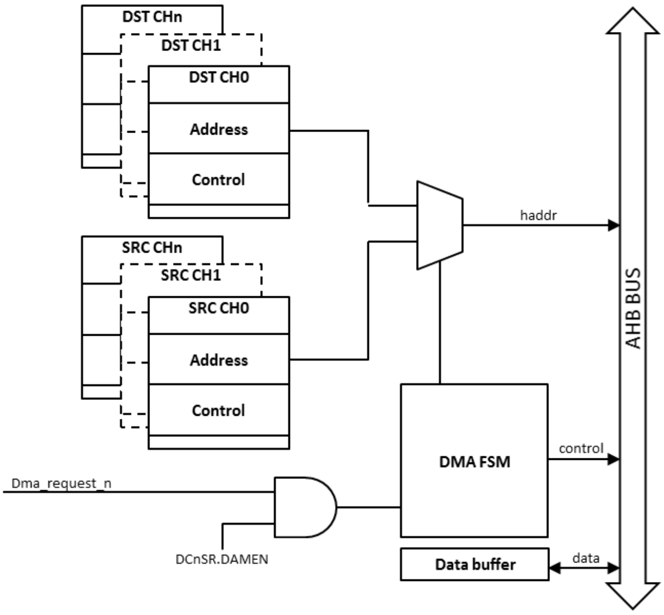
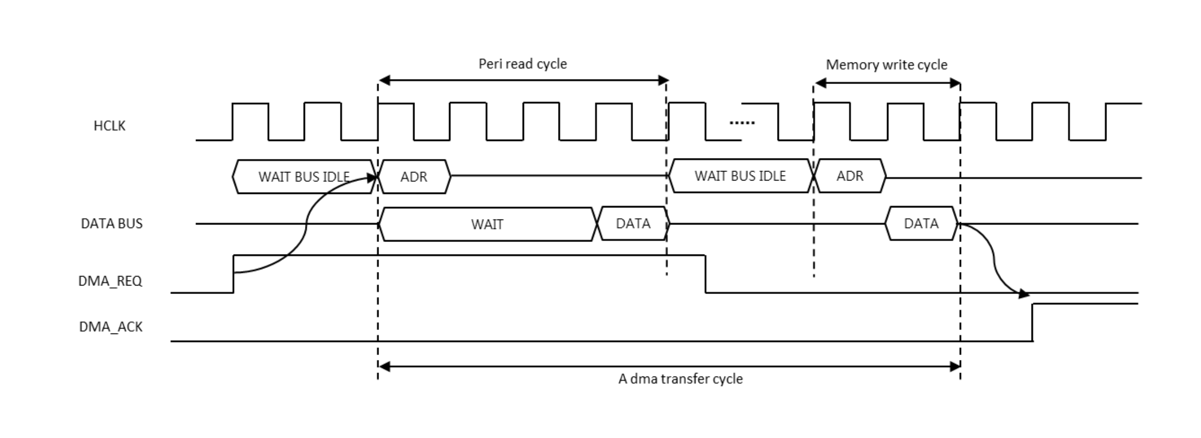


Figure 143 shows the functional timing diagram of DMA controller. The transfer request from peripheral is pended internally and it will invoke source data read transfer on the AHB bus. The read data from the source address is stored in the internal buffer. Then this data will be transferred to the destination address when the AHB bus is available.

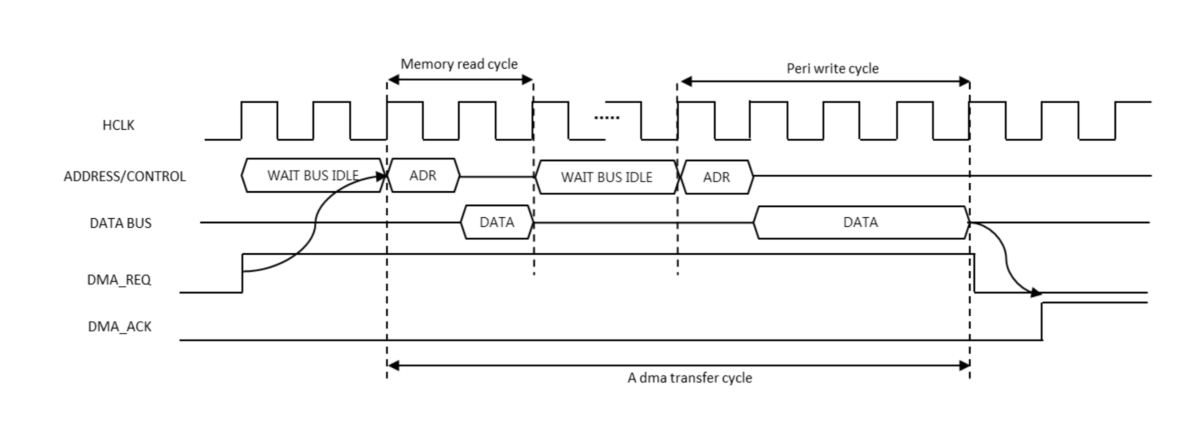
The timing diagram for a DMA transfer from peripheral to memory is shown in below figure. 4-clock cycle latency exists during accessing the peripheral. If the bus is occupied by different bus master, there are amount of bus waiting cycles.

Figure 143. DMA Transfer from Peripheral to Memory



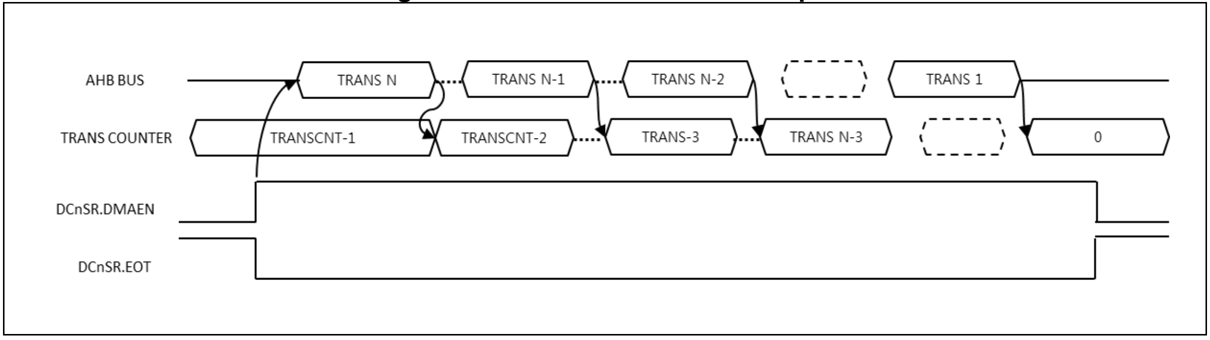
The timing diagram for a DMA transfer from memory to peripheral is shown in Figure 144. 4-clock cycle latency exists during accessing the peripheral. If the bus is occupied by different bus master, there are amount of bus waiting cycles.

Figure 144. DMA Transfer from Memory to Peripheral



The figure below is an example N data transfers with the DMA. The DMA transfer is started when DCnSR.DMAEN is set and will be cleared when all the number of transfer is completed.

Figure 145. N DMA Transfer Example



28 Electrical Characteristics

Unless otherwise specified, test conditions for DC characteristics are as shown in the followings:

- T_A = - 40 °C to +85°C(Commercial grade) or T_A = - 40 °C to + 105°C(Industrial grade)
- VDD = 1.71V to 3.6V

NOTE: Refer to **Figure 169. A31L21x Series Numbering Nomenclature** for device part number by Commercial and Industrial grade.

28.1 Absolute maximum ratings

Absolute maximum ratings are limiting values of operating and environmental conditions, which should not be exceeded under the worst possible conditions.

Table 115. Absolute Maximum Ratings

Parameter	Symbol	Ratings	Unit	Remark
Supply voltage	VDD	-0.3 to +4.0	V	–
Normal pin	V_I	-0.3 to VDD +0.3	V	Voltage on any pin with respect to VSS
	V_O	-0.3 to VDD +0.3	V	
	I_{OH}	-15	mA	Maximum current output sourced by (I_{OH} per I/O pin)
	ΣI_{OH}	-60	mA	Maximum current (ΣI_{OH})
	I_{OL}	20	mA	Maximum current sunk by (I_{OL} per I/O pin)
	ΣI_{OL}	160	mA	Maximum current (ΣI_{OL})
5V tolerant pin	VI	-0.3 to +6.0	V	Voltage on any pin with respect to VSS
Total power dissipation	P_T	600	mW	–
Storage temperature	T_{STG}	-65 to +150	°C	–

28.2 Recommended operating conditions

Table 116. Recommended Operating Conditions

Parameter	Symbol	Conditions			Min	Max	Units
Operating voltage	VDD	fx = 32 to 38kHz	Sub clock		1.71	3.6	V
		fx = 2.0 to 4.2MHz	Main clock	Ceramic	1.8	3.6	
		fx = 2.0 to 16MHz		Crystal	2.7	3.6	
		fx = 2.0 to 32MHz	External clock		3.0	3.6	
		fx = 40kHz	Internal RC		1.71	3.6	
		fx = 2.5 to 32MHz			1.71	3.6	
Input voltage	V_{IN}	Normal Pin			-0.3	VDD+0.3	V
		5V tolerance Pins, PD[4:0]	2.0V ≤ VDD ≤ 3.6V		-0.3	5.5	
			1.71V ≤ VDD < 2.0V		-0.3	5.0	
Operating temperature	T_{OPR}	VDD = 1.71 to 3.6V (Commercial grade)			-40	85	°C
		VDD = 1.71 to 3.6V (Industrial grade)			-40	105	

28.3 ADC characteristics

Table 117. ADC Characteristics

(TA = 25℃)

Parameter	Symbol	Conditions	Min	Typ	Max	Units
Resolution	–	–	–	12	–	bit
Integral non-linearity	INL	AVDD=1.8V – 3.6V	–	–	±6	LSB
Differential non-linearity	DNL		–	±1	±2	
Zero offset error	ZOE		–	–	±5	
Full scale error	FSE		–	–	±5	
Integral non-linearity	INL	AVDD=1.71V – 3.6V	–	–	±6	LSB
Differential non-linearity	DNL		–	±1	±2.5	
Conversion time	t_{CONV}	AVDD=2.7V – 3.6V	1	–	–	µs
		AVDD=1.71V – 3.6V	2	–	–	
Analog input voltage	V_{AN}	–	VSS	–	AVDD	V
Analog voltage	AVDD	–	VDD-0.3	VDD	VDD+0.3	V
ADC stabilization time	t_{STAB}	–	–	–	16	1/f_{ADC}
Band gap reference buffer voltage	V_{ADCBUF}	Conversion time: 8us	890	940	990	mV
ADC input leakage current	I_{AN}	AVDD=3.0V	–	–	2	µA
ADC current	I_{ADC}	Enable AVDD=3.0V, f_{ADC}=16MHz	–	400	800	µA
		Disable	–	–	10	nA

NOTES:

1. Zero offset error is a difference between 0x000 and the converted output for zero input voltage (VSS).
2. Full scale error is a difference between 0xFFF and the converted output for top input voltage (VDD).

28.4 Power-on Reset characteristics

Table 118. Power-on Reset Characteristics

Parameter	Symbol	Conditions	Min	Typ	Max	Units
Reset release level	V_{POR}	–	–	1.2	–	V
Hysteresis	△V	–	–	0.1	–	V
VDD voltage rising time	t_R	0.2V to 2.0V	0.05	–	100	V/ms
POR current	I_{POR}	–	–	21	40	nA

28.5 Comparator characteristics

Table 119. Comparator Characteristics

(TA = 25℃)

Parameter	Symbol	Conditions		Min	Typ	Max	Units
Input offset voltage	VOF	VDD=3.0V, VIN=1/2VDD		–	±4	±20	mV
Operating voltage	VDD	All comparator pins except below		1.71	–	3.6	V
		CP1N3, CP1P2, CP1P3, CP1P4, CP1P5		2.0	–	3.6	
Startup time	t_{START}	Fast speed		–	15	20	µs
		Slow speed		–	20	25	
Propagation delay	t_{DELAY}	1.71V ≤ VDD ≤ 2.7V	Fast Speed	–	1.2	4	µs
		2.7V ≤ VDD ≤ 3.6V		–	0.8	2	
		1.71V ≤ VDD ≤ 2.7V	Slow Speed	–	2.5	6	
		2.7V ≤ VDD ≤ 3.6V		–	1.8	3.5	
Hysteresis	△V+	VDD=3.0V, VIN- = 1/2VDD, HYSnEN=1		5	10	20	mV
	△V-			-20	-10	-5	
Minimum input level	V_{INMIN}	HYSnEN=1		50	–	–	mVp-p
Reference resistors	R_{REF}	VDD=3.0V		21	30	39	kΩ
Comparator current	ICMP	Enable, fast speed	VDD=3.0V	–	3.5	5	µA
		Enable, slow speed		–	1.0	2	
		Disable		–	–	0.02	

28.6 Temperature Sensor characteristics

Table 120. Temperature Sensor Characteristics

Parameter	Symbol	Conditions		Min	Typ	Max	Units
Temp. linearity	T_{LIN}	Down to -20℃		–	±4	–	℃
Frequency variation	△F	(F(T2) – F(T1)) ÷ (T2 – T1)		1.8	3.2	5.7	kHz/℃
Frequency deviation	–	△F ÷ F(30)		0.25	0.35	0.45	%
Sensor current	I_{TS}	Enable	VDD = 3.0V	–	10	20	uA
		Disable		–	–	10	nA
Startup time	t_{START}	–		–	–	500	µs

NOTES:

1. Temperature = {(F(T) – F(30)) ÷ △F} + 30 [℃], Where: T1 = 30℃, T2 = 85℃(Commercial grade) or 105℃(Industrial grade)
2. F(T1) [kHz] is the temperature sensor output frequency acquired at 30℃.
3. F(T2) [kHz] is the temperature sensor output frequency acquired at 85℃(Commercial grade) or 105℃(Industrial grade).
4. F(T) [kHz] is the temperature sensor output frequency acquired at an arbitrary temperature.

28.7 Low Voltage Reset/Indicator characteristics

Table 121. Low Voltage Reset/Indicator Characteristics

(TA = 25℃)

Parameter	Symbol	Conditions		Min	Typ	Max	Units
Detection level	V_{LVR} V_{LVI}	• LVR: All levels, • LVI: Other levels except 1.50V, • 1.50V level: Rising edge voltage, • Other levels: Falling edge voltage		–	1.50	1.70	V
				1.72	1.87	2.02	
				1.87	2.02	2.17	
				2.02	2.17	2.32	
				2.17	2.32	2.47	
				2.27	2.47	2.67	
				2.44	2.64	2.84	
				2.58	2.78	2.98	
Hysteresis	$\triangle V$	–		–	40	150	mV
Minimum pulse width	t_{LVRW} t_{LVIW}	–		100	–	–	µs
LVR/LVI current	$I_{LVR/LVI}$	Enable, one of two	VDD = 3V	–	200	400	nA
		Enable, both		–	250	500	
		Disable		–	–	10	

28.8 High frequency internal RC oscillator characteristics

Table 122. High Frequency Internal RC Oscillator Characteristics

Parameter	Symbol	Conditions	Min	Typ	Max	Units
Frequency	f_{HIRC}	VDD = 1.71V to 3.6V	–	32	–	MHz
Accuracy	–	T_A = -40 °C to +85 °C (commercial grade)	–	–	±2.0	%
		T_A = -40 °C to +105 °C (industrial grade)	–	–	±3.0	
Clock duty ratio	T_{OD}	–	40	50	60	%
Stabilization time	t_{HFS}	–	–	–	2	µs
IRC current	I_{HIRC}	Enable	–	300	450	µA
		Disable	–	–	10	nA

28.9 Internal Watchdog Timer RC oscillator characteristics

Table 123. Internal Watchdog Timer RC Oscillator Characteristics

Parameter	Symbol	Conditions	Min	Typ	Max	Units
Frequency	f_{WDTRC}	–	34	40	46	kHz
Stabilization time	t_{WDTS}	–	–	–	100	µs
WDTRC current	I_{WDTRC}	Enable	–	450	650	nA
		Disable	–	–	10	

28.10 Timer 60 RC oscillator characteristics

Table 124. Timer 60 RC Oscillator Characteristics

Parameter	Symbol	Conditions	Min	Typ	Max	Units
Frequency	f_{T60RC}	–	50	100	200	Hz
Stabilization time	t_{T60S}	–	–	–	100	µs
T60 current	I_{T60RC}	Enable	–	200	350	nA
		Disable	–	–	10	

28.11 LCD voltage characteristics

Table 125. LCD Voltage Characteristics

Parameter	Symbol	Conditions	Min	Typ	Max	Units
LCD voltage	VLC3	• Voltage booster enabled • 1/3 bias: All levels • 1/2 and 1/4 bias: Up to 0.825V	Typx0.93	0.725	Typx1.07	V
				0.750		
				0.775		
				0.800		
				0.825		
				0.850		
				0.875		
				0.900		
				0.925		
				0.950		
				0.975		
				1.000		
				1.025		
				1.050		
				1.075		
				1.100		
LCD mid bias voltage	VLC0/1	• Voltage booster enabled • 1/3 bias, No panel load • VDD=3.0V	Typ x 0.9	3 x VLC3	Typ x 1.1	V
	VLC2		Typ x 0.9	2 x VLC3	Typ x 1.1	
	VLC0	• Voltage booster enabled • 1/2 and 1/4 bias, No panel load • VDD=3.0V	Typ x 0.9	4 x VLC3	Typ x 1.1	
	VLC1		Typ x 0.9	3 x VLC3	Typ x 1.1	
	VLC2		Typ x 0.9	2 x VLC3	Typ x 1.1	
	VLC1	• LCD dividing register • 1/4 bias, No panel load • VDD=2.7V to 3.6V, VLC0=VDD	Typ - 0.2	3/4 x VLC3	Typ + 0.2	V
	VLC2		Typ - 0.2	2/4 x VLC3	Typ + 0.2	
	VLC3		Typ - 0.2	1/4 x VLC3	Typ + 0.2	
LCD driver output impedance	R_{LO}	VLCD=3V, ILOAD = ±10uA	–	5	10	kΩ
LCD bias dividing resistor	RLCD1	• Internal resistor mode • T_A = 25℃	20	30	40	kΩ
	RLCD2		40	60	80	
	RLCD3		80	120	160	
LCD block current	I_{LCD}	• Voltage booster mode • VDD=3V, VLCD=3.15V, 1/3bias	–	3	6	uA

28.12 DC electrical characteristics

Table 126. DC Electrical Characteristics

Parameter	Symbol	Conditions	Min	Typ	Max	Units
Input High Voltage	V_{IH}	All input pins, nRESET	0.8VDD	–	VDD	V
Input Low Voltage	V_{IL}	All input pins, nRESET	–	–	0.2VDD	V
Input hysteresis	△V	All input pins, nRESET, VDD=3V	100	200	–	mV
Output High Voltage	V_{OH}	VDD=3V, I_{OH} = – 10mA, T_A=25°C	VDD-1.0	–	–	V
Output Low Voltage	V_{OL1}	VDD=3V, I_{OL1} = 10mA, T_A=25°C All output ports except V_{OL2}	–	–	1.0	V
	V_{OL2}	VDD=3V, I_{OL2} = 15mA, T_A=25°C, PB[7:0]	–	–	1.0	
Input high leakage current	I_{IH}	All Input ports	–	–	1	µA
Input low leakage current	I_{IL}	All Input ports	– 1	–	–	µA
Pull-up resistor	R_{PU}	V_I=0V, T_A=25°C, VDD=3V All Input ports	25	50	100	kΩ
		V_I=0V, T_A=25°C, VDD=3V RESETB	150	250	400	
Pull-down resistor	R_{PD}	V_I=VDD, T_A=25°C, VDD=3V All Input ports	25	50	100	kΩ
OSC feedback resistor	R_{X1}	XIN=VDD, XOUT=VSS, T_A=25°C, VDD=3V	0.6	1.2	2.0	MΩ
	R_{X2}	T_A=25°C, VDD=3V	4.0	7.0	14.0	MΩ

28.13 Supply current characteristics

Table 127. Supply Current Characteristics

Parameter	Symbol	Conditions			Typ	Max	Units
Supply current	I_{DD1} (main run)	f_{HIRC} = 32MHz		VDD=3V, Code executed from Flash	2.9	4.0	mA
		f_{HIRC} = 16MHz			1.9	2.7	
		f_{XIN} = 16MHz			1.8	2.5	
		f_{HIRC} = 32MHz		VDD=3V, Code executed from RAM, Flash power off	2.6	3.6	mA
		f_{HIRC} = 16MHz			1.8	2.5	
		f_{XIN} = 16MHz			1.7	2.4	
	I_{DD2} (main sleep)	f_{HIRC} = 32MHz		VDD=3V, SLEEP in Flash	1.4	2.0	mA
		f_{HIRC} = 16MHz			0.9	1.3	
		f_{XIN} = 16MHz			0.9	1.3	
		f_{HIRC} = 32MHz		VDD=3V, SLEEP in RAM, Flash power off	1.3	1.8	mA
		f_{HIRC} = 16MHz			0.8	1.1	
		f_{XIN} = 16MHz			0.8	1.1	
	I_{DD3} (sub run)	f_{SUB} = 32.768kHz (C_L: 7pF), or f_{WDTRC} = 40kHz	T_A=25°C	VDD=3V Code executed from Flash	13.0	23.0	uA
			T_A=85°C		21.0	32.0	
			T_A=105°C		30.0	45.0	
			T_A=25°C	VDD=3V, Code executed from RAM, Flash power off	10.0	19.0	uA
			T_A=85°C		19.0	35.0	
			T_A=105°C		28.0	45.0	
	I_{DD4} (sub sleep)	f_{SUB} = 32.768kHz (C_L: 7pF), or f_{WDTRC} = 40kHz	T_A=25°C	VDD=3V, SLEEP in Flash	2.2	5.6	uA
			T_A=85°C		6.0	21.0	
			T_A=105°C		12.0	28.0	
	I_{DD5}	VDD=3V DEEP SLEEP mode 0	T_A=25°C	RTCC/f_{SUB} Off	0.39	0.99	uA
			T_A=85°C		2.24	7.9	
			T_A=105°C		5.0	18.9	
			T_A=25°C	RTCC/f_{SUB} On	0.99	1.6	uA
			T_A=85°C		2.9	9.9	
			T_A=105°C		6.4	24.0	
	I_{DD6}	VDD=3V DEEP SLEEP mode 1	T_A=25°C	RTCC/f_{SUB} Off	0.35	0.9	uA
			T_A=85°C		1.5	6.9	
			T_A=105°C		3.5	15.9	
	I_{DD7}	VDD=3V DEEP SLEEP mode 2	T_A=25°C	RTCC/f_{SUB} Off	0.32	0.7	uA
			T_A=85°C		0.8	2.9	
			T_A=105°C		1.7	6.8	
	I_{DD8}	VDD=3V DEEP SLEEP mode 3 (Shutdown)	T_A=25°C	All Off	45	90	nA
			T_A=85°C		0.35	2.1	uA
			T_A=105°C		0.95	3.1	

NOTES:

1. Where the f_{XIN} is an external main oscillator, the f_{SUB} is an external sub oscillator (ISET_I[2:0] = 0x5), and the f_{HIRC} is a high frequency internal RC oscillator.
2. All supply current items don't include the current of WDTRC oscillator and a peripheral block except when explicitly mentioned. However, it does include the current of the power-on reset (POR) block.
3. There is thing to watch out for before entering a deep sleep mode 1. Refer to "functional table on current mode" in the PMU block.

Figure 146. IDD4 (SLEEP mode, fSUB = 32.768 kHz) at VDD = 3V

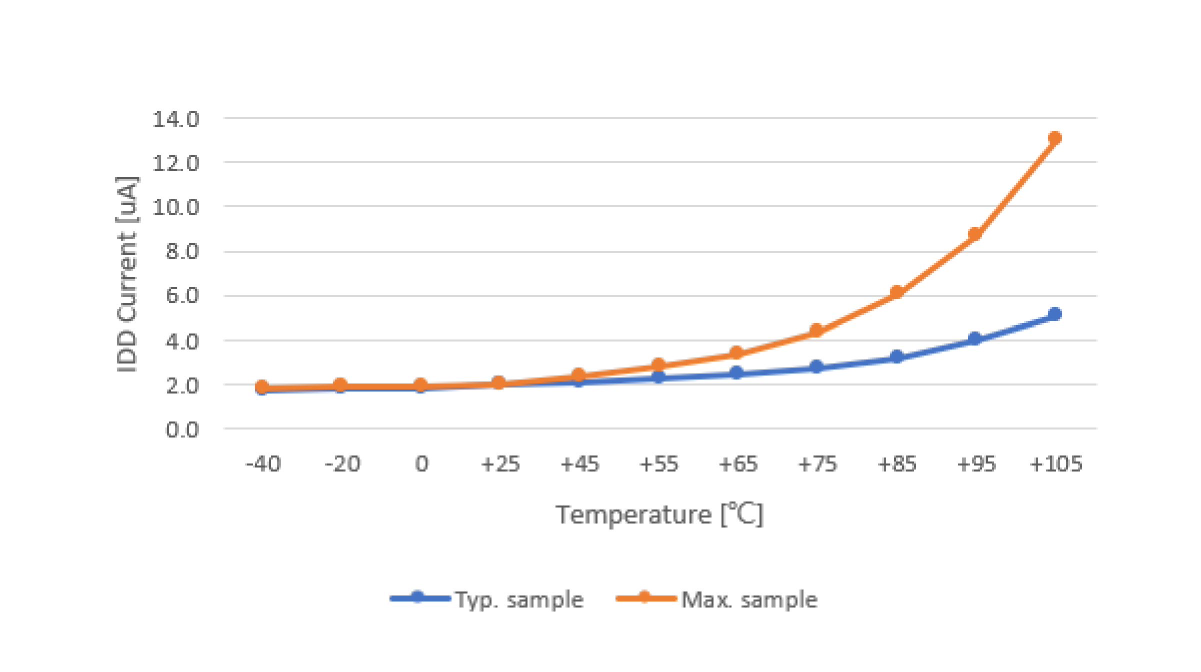


Figure 147. IDD4 (SLEEP mode, fWDTRC = 40 kHz) at VDD = 3V

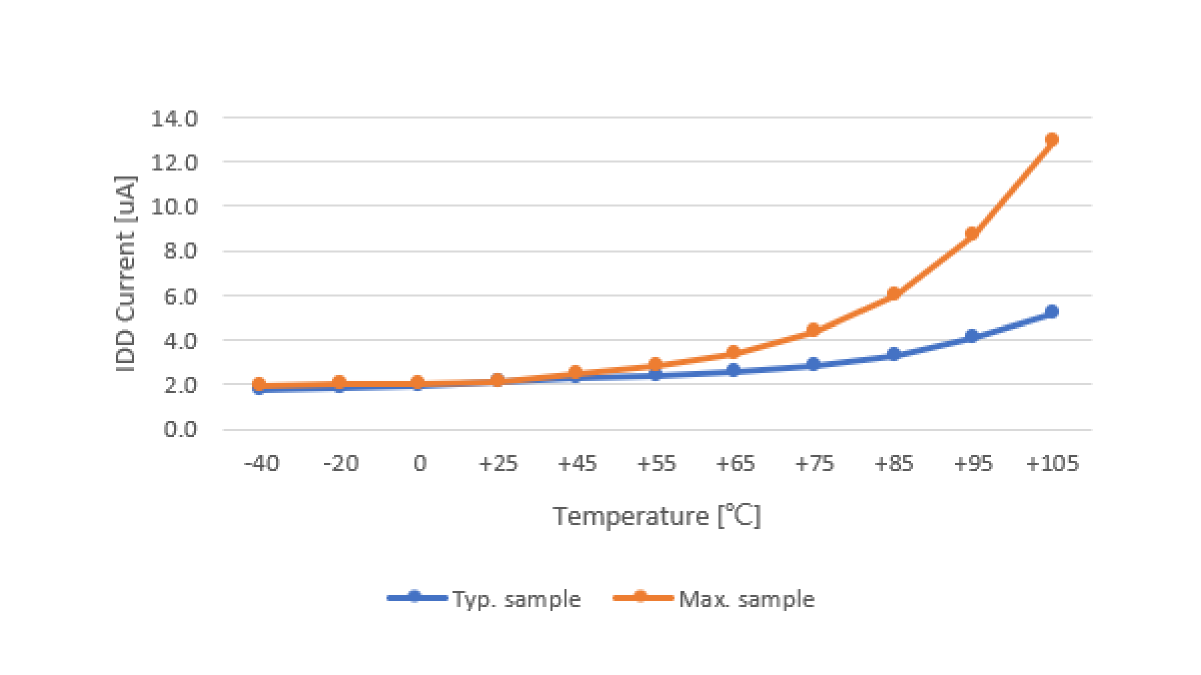


Figure 148. IDD5 (DEEP SLEEP mode 0, RTCC/fSUB Off) at VDD = 3V

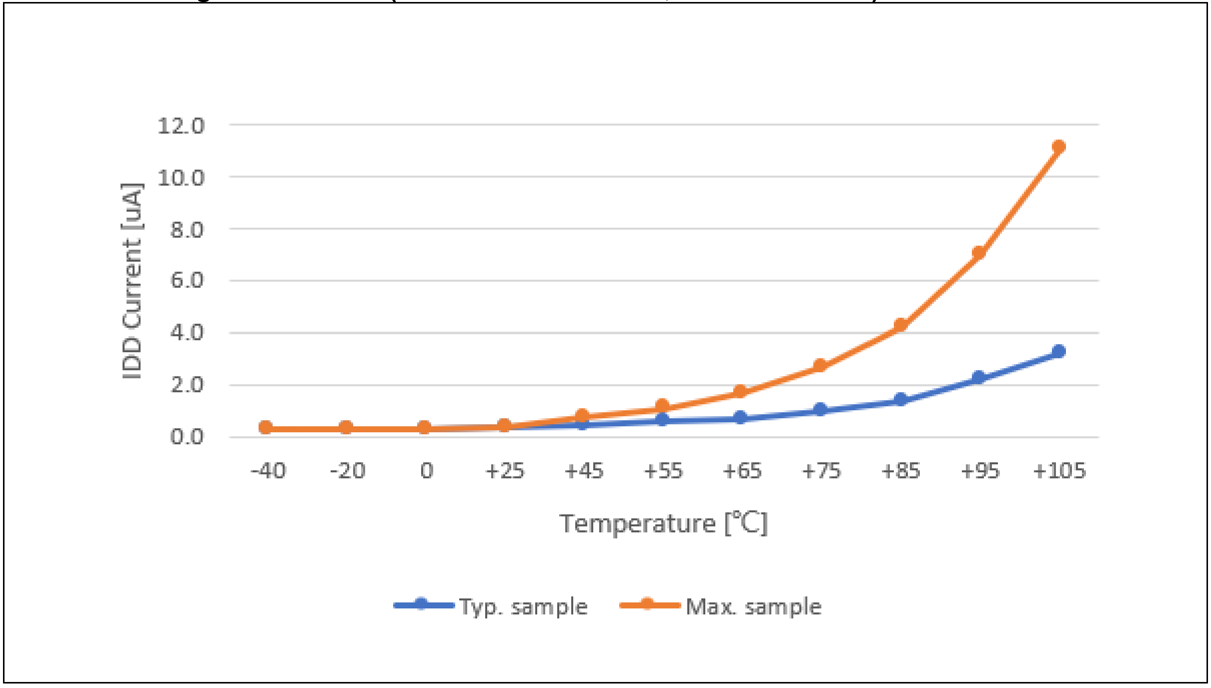


Figure 149. IDD5 (DEEP SLEEP mode 0, RTCC/fSUB On) at VDD = 3V

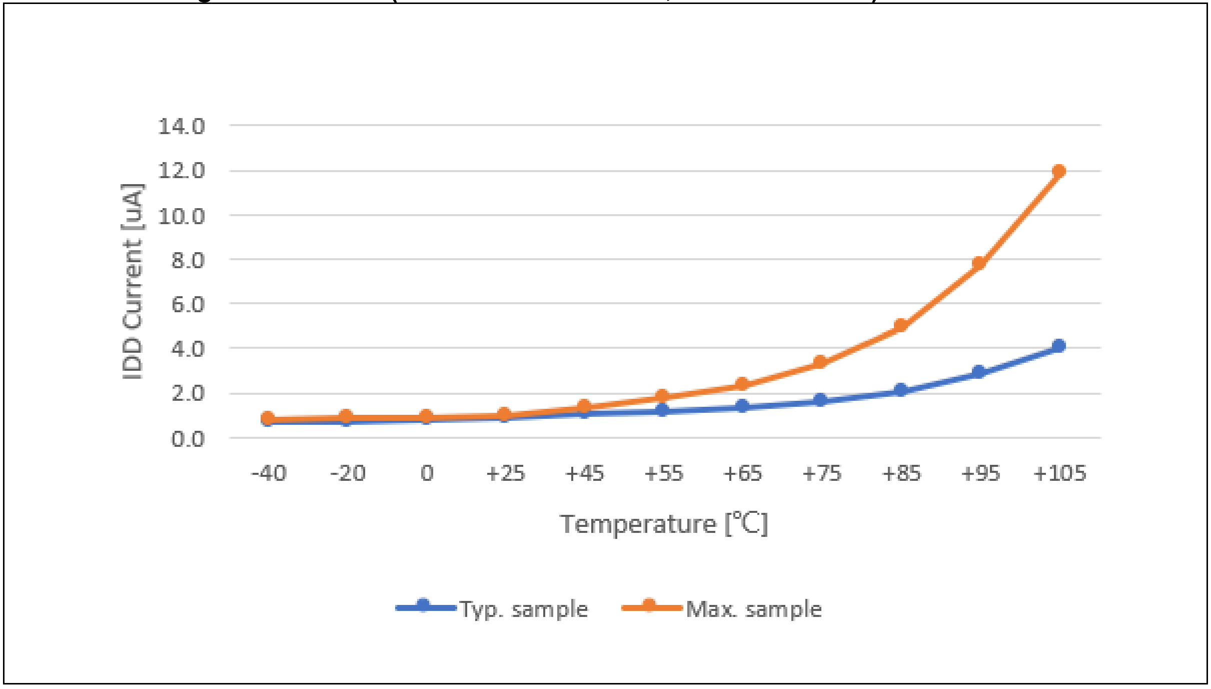


Figure 150. IDD6 (DEEP SLEEP mode 1) at VDD = 3V

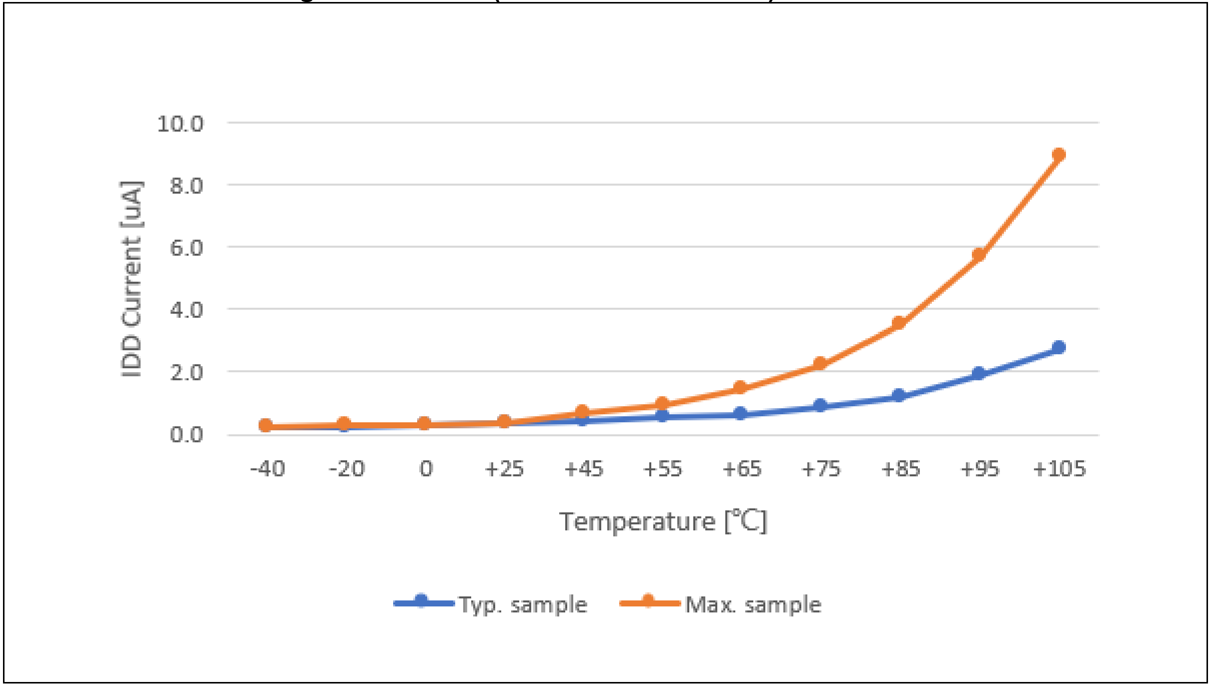
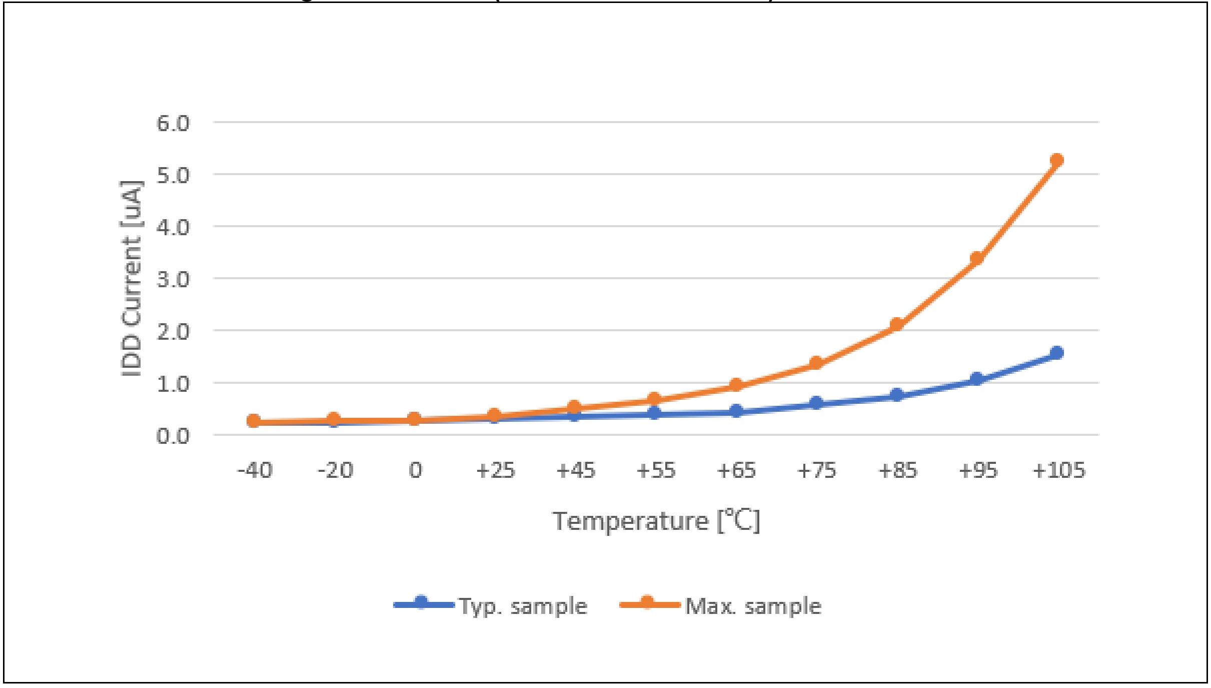


Figure 151. IDD7 (DEEP SLEEP mode 2) at VDD = 3V

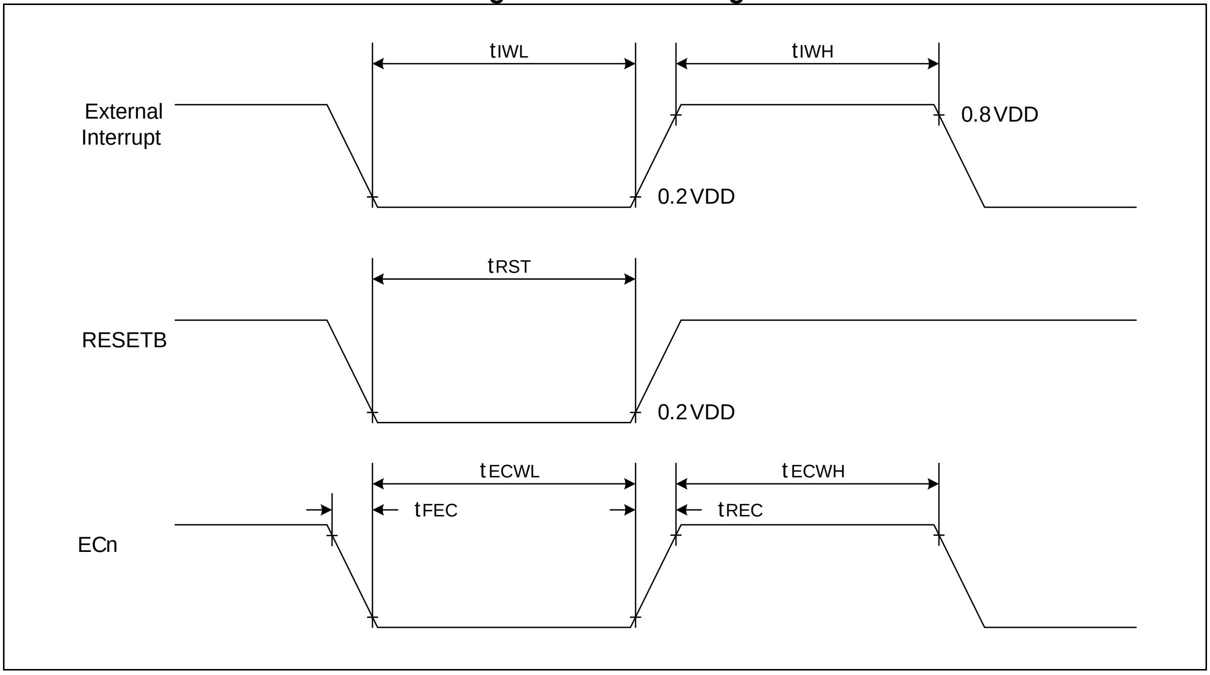


28.14 AC characteristics

Table 128. AC Characteristics

Parameter	Symbol	Conditions	Min	Typ	Max	Units
RESETB input low width	t_{RST}	VDD = 3 V	20	–	–	μs
Interrupt input high, low width	t_{IWH}, t_{IWL}	All interrupts, VDD = 3 V	50	–	–	ns
External counter input high, low pulse width	t_{ECWH}, t_{ECWL}	VDD = 3 V All external counter input	1	–	–	$1/f_{PCLK}$
External counter transition time	t_{REC}, t_{FEC}	ECn, VDD = 3 V All external counter input	–	–	10	ns
I/O frequency	f_{IO1}	VDD = 3.0V, C_L = 30pF, All except f_{IO2}	–	–	10	MHz
	f_{IO2}	VDD = 2.7V, C_L = 30pF, SPI pins	–	–	16	

Figure 152. AC Timing

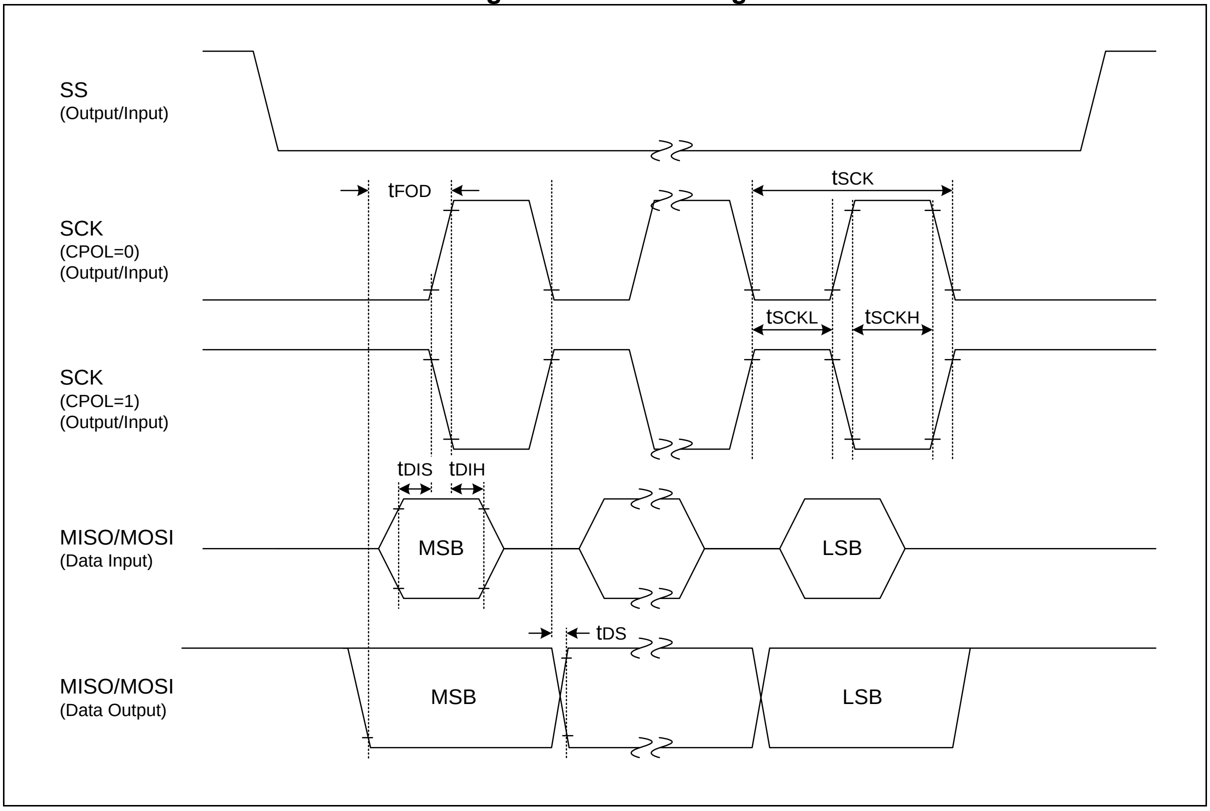


28.15 SPI characteristics

Table 129. SPI Characteristics

Parameter	Symbol	Conditions		Min	Typ	Max	Units
SPI clock frequency	f_{SCK}	VDD ≥ 2.7V	Internal SCK source	–	–	16	MHz
			External SCK source				
	f_{SCK}	VDD ≥ 1.71V	Internal SCK source	–	–	12	
			External SCK source				
Input/output clock high, low pulse width	t_{SCKH}, t_{SCKL}	Internal/External SCK source		0.8*Typ	t_{SCK}/2	1.2*Typ	ns
First output clock delay time	t_{FOD}	Internal/External SCK source, CPHA = 0		0.4*t_{SCK}	–	–	
Output clock delay time	t_{DS}	–		–	–	18	
Input setup time	t_{DIS}			13	–	–	
Input hold time	t_{DIH}			15	–	–	

Figure 153. SPI Timing

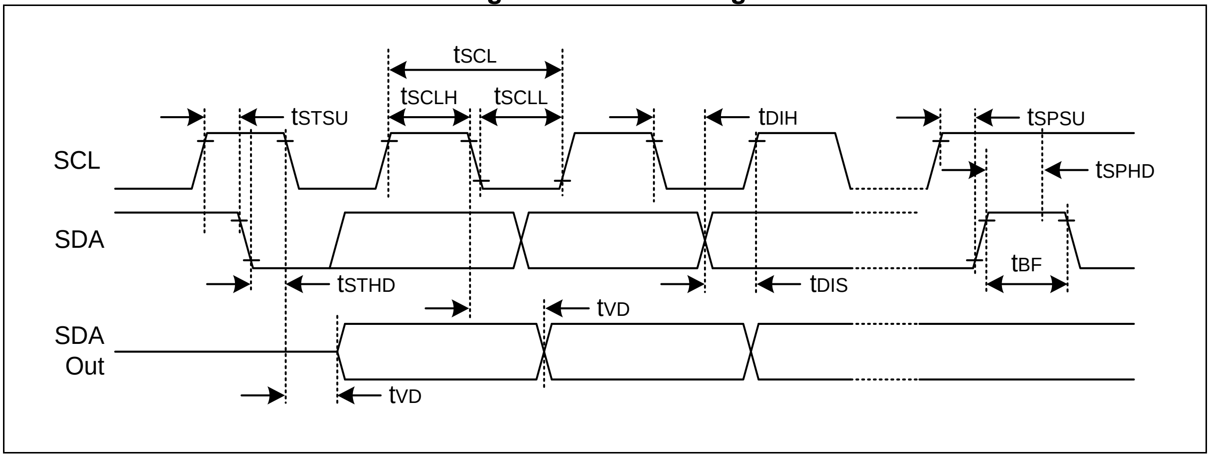


28.16 I2C characteristics

Table 130. I2C Characteristics

Parameter	Symbol	Standard		Fast		Fast Plus		Units
		Min	Max	Min	Max	Min	Max	
I2C operating voltage	–	VDD ≥ 1.71V		VDD ≥ 2V		VDD ≥ 2.7V		–
Clock frequency	t_{SCL}	0	100	0	400	0	1000	kHz
Clock high pulse width	t_{SCLH}	4.0	–	0.6	–	0.26	–	µs
Clock low pulse width	t_{SCLL}	4.7	–	1.3	–	0.5	–	
Bus free time	t_{BF}	4.7	–	1.3	–	0.5	–	
Start condition setup time	t_{STSU}	4.7	–	0.6	–	0.26	–	
Start condition hold time	t_{STHD}	4.0	–	0.6	–	0.26	–	
Stop condition setup time	t_{SPSU}	4.0	–	0.6	–	0.26	–	
Stop condition hold time	t_{SPHD}	4.0	–	0.6	–	0.26	–	
Output Valid from Clock	t_{VD}	0	–	0	–	0	–	
Data input hold time	t_{DIH}	0	–	0	1.0	0	0.45	
Data input setup time	t_{DIS}	250	–	100	–	50	–	ns

Figure 154. I2C Timing



28.17 UART timing characteristics

Table 131. UART Timing Characteristics (PCLK=32MHz)

Parameter	Symbol	Min	Typ	Max	Units
Serial port clock cycle time	t_{SCK}	–	–	2000	kHz
Output data setup to clock rising edge	t_{S1}	t_{SCK} x 12/16	–	–	ns
Clock rising edge to input data valid	t_{S2}	–	–	t_{SCK} x 13/16	
Output data hold after clock rising edge	t_{H1}	–	–	50	
Input data hold after clock rising edge	t_{H2}	0	–	–	
Serial port clock High, Low level width	t_{HIGH}, t_{LOW}	t_{SCK} x 6/16	t_{SCK} x 8/16	t_{SCK} x 10/16	

Figure 155. UART Timing Characteristics

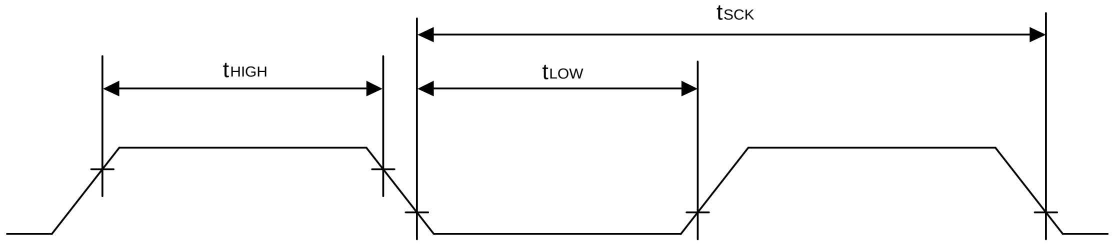
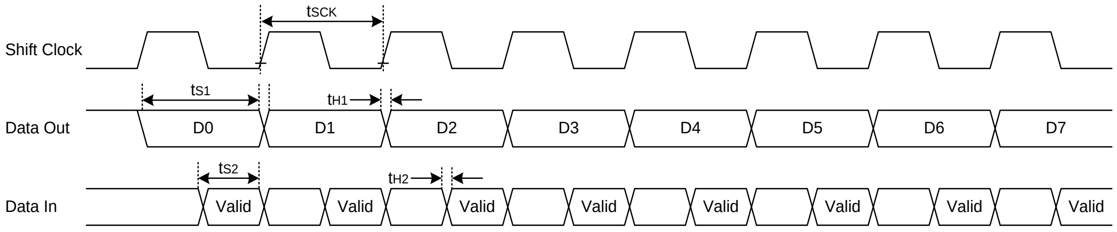


Figure 156. Timing Waveform of UART Module



28.18 Data retention voltage in DEEP SLEEP mode 0/1

Table 132. Data Retention Voltage in DEEP SLEEP mode 0/1

Parameter	Symbol	Conditions	Min	Typ	Max	Units
Data retention supply voltage	V_{DDDR}	–	1.71	–	3.6	V
Data retention supply current	I_{DDDR}	• V_{DDDR} = 1.71V (T_A=25°C) • DEEP SLEEP mode 0/1	–	–	1	µA

28.19 Internal Flash memory and Data Flash memory characteristics

Table 133. Internal Flash Memory and Data Flash Memory Characteristics

Parameter	Symbol	Conditions		Min	Typ	Max	Units
Page write time	t_{FSW}	–		–	3.0	3.5	ms
Page erase time	t_{FSE}	–		–	3.0	3.5	
Chip erase time	t_{FCE}	–		–	3.0	3.5	
Program voltage	V_{PGM}	On erase/write		2.0	–	3.6	V
System clock frequency	f_{HCLK}	–		2.0	–	–	MHz
Flash Memory Endurance of Write/Erase	NF_{FWE}	• Page 0 to 511 • Configure Option Page 1	T_A=25 °C, Page unit	10,000	–	–	Cycles
		Configure Option Page 2/3		100,000			
Data Flash Memory Endurance of Write/Erase	NF_{DFWE}	Page 0 to 127	T_A=25 °C, Page unit	100,000	–	–	Cycles
Retention time	t_{FRT}			10	–	–	Years

28.20 Input/output capacitance

Table 134. Input/Output Capacitance

(VDD = 0V)

Parameter	Symbol	Conditions	Min	Typ	Max	Units
Input capacitance	C_{IN}	• f=1MHz • Unmeasured pins are connected VSS	–	–	10	pF
Output capacitance	C_{OUT}					
I/O capacitance	C_{IO}					

28.21 Main oscillator characteristics

Table 135. Main Oscillator Characteristics

(VDD = 1.8V to 3.6V)

Oscillator	Parameter	Conditions	Min	Typ	Max	Units
Crystal	Main oscillation frequency	2.7 V to 3.6 V	2.0	–	16.0	MHz
Ceramic Oscillator	Main oscillation frequency	1.8 V to 3.6 V	2.0	–	4.2	
		2.7 V to 3.6 V	2.0	–	16.0	
External Clock	XIN input frequency	3.0 V to 3.6 V	2.0	–	32.0	MHz
	External Clock Duty Ratio	–	45	50	55	%

Figure 157. Crystal/Ceramic Oscillator

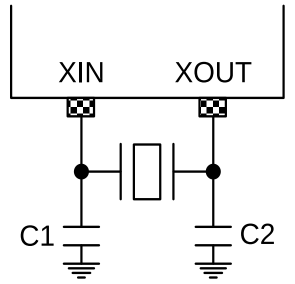
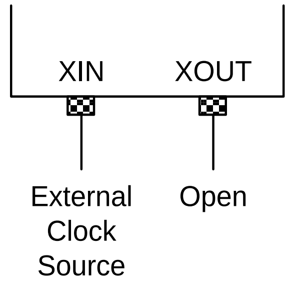


Figure 158. External Clock

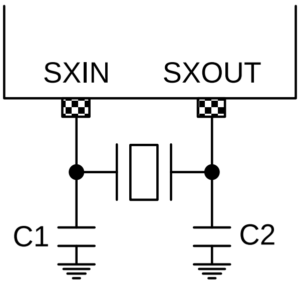


28.22 Sub-oscillator characteristics

Table 136. Sub-oscillator Characteristics

Oscillator	Parameter	Conditions	Min	Typ	Max	Units
Crystal	Sub oscillation frequency	1.71 V to 3.6 V	32	32.768	38	kHz

Figure 159. Crystal Oscillator



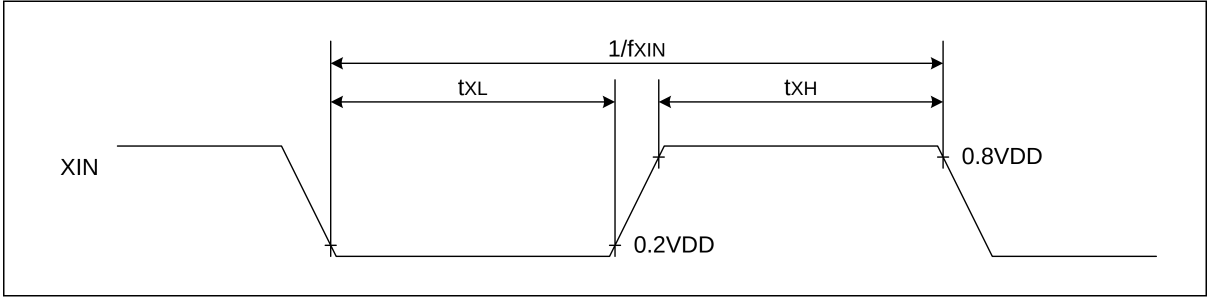
28.23 Main oscillation stabilization time

Table 137. Main Oscillation Stabilization Time

(VDD = 1.8V to 3.6V)

Oscillator	Conditions		Min	Typ	Max	Units
Crystal	• f_{XIN} ≥ 2MHz • Oscillation stabilization occurs when VDD is equal to the minimum oscillator voltage range.	VDD = 2.7V to 3.6V	–	–	60	ms
Ceramic		VDD = 1.8V to 3.6V	–	–	10	
External clock	• f_{XIN} = 2.0 to 32MHz • XIN input high and low width (t_{XL}, t_{XH})		15.6	–	250	ns

Figure 160. Clock Timing Measurement at XIN



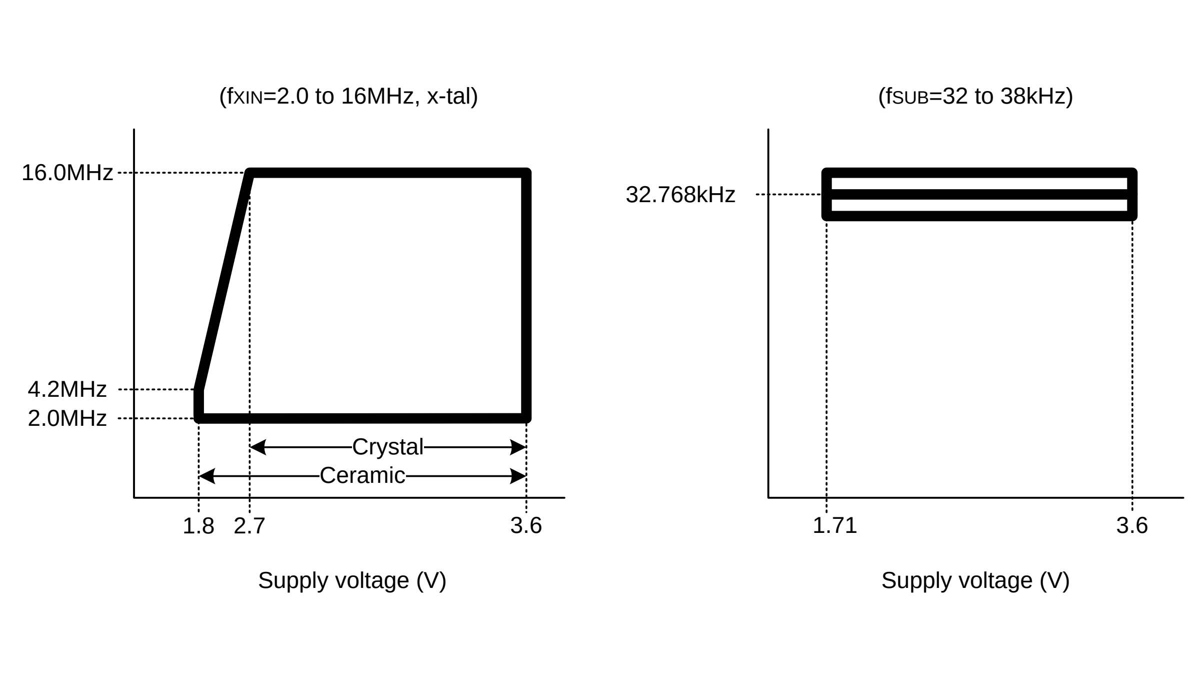
28.24 Sub-oscillation stabilization time

Table 138. Sub-oscillation Stabilization Time

Oscillator	Conditions	Min	Typ.	Max	Unit
Crystal	–	–	–	10	sec
	VDD=3V, T_A=25 °C, ISET_I[2:0] = 0x7	–	0.7	1.5	

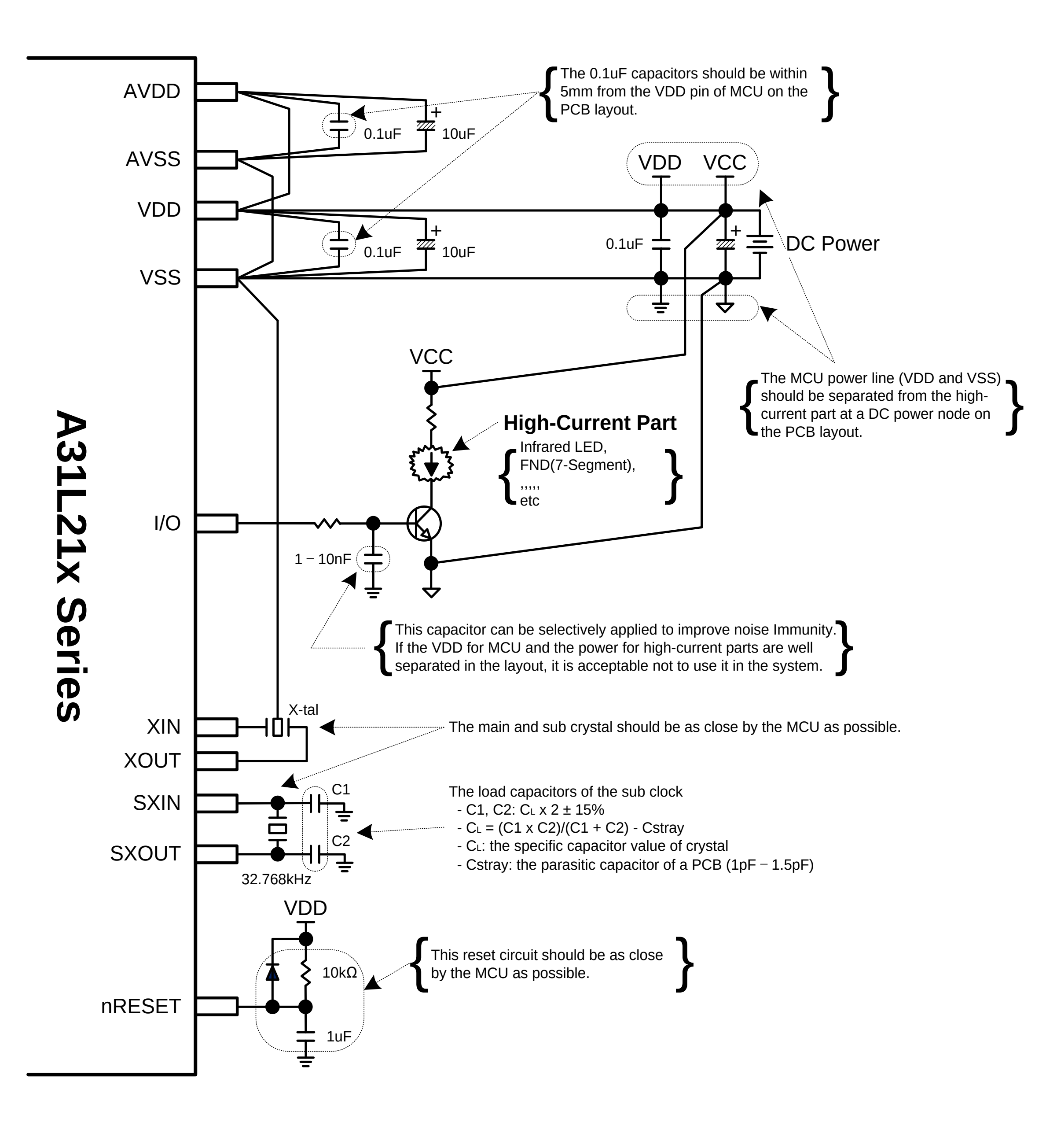
28.25 Operating voltage range

Figure 161. Operating Voltage Range



28.26 Recommended circuit and layout

Figure 162. Recommended Circuit and Layout



The 0.1uF capacitors should be within 5mm from the VDD pin of MCU on the PCB layout.

The MCU power line (VDD and VSS) should be separated from the high-current part at a DC power node on the PCB layout.

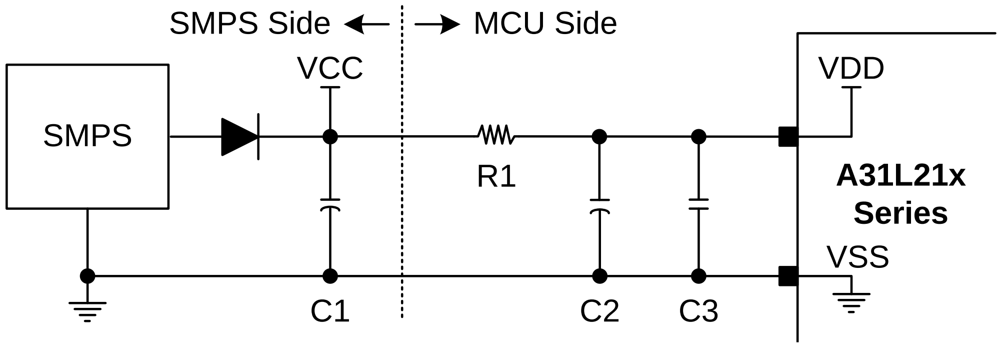
This capacitor can be selectively applied to improve noise Immunity. If the VDD for MCU and the power for high-current parts are well separated in the layout, it is acceptable not to use it in the system.

The main and sub crystal should be as close by the MCU as possible.

The load capacitors of the sub clock
- C1, C2: C_L x 2 ± 15%
- C_L = (C1 x C2)/(C1 + C2) - Cstray
- C_L: the specific capacitor value of crystal
- Cstray: the parasitic capacitor of a PCB (1pF ~ 1.5pF)

This reset circuit should be as close by the MCU as possible.

Figure 163. Recommended Circuit and Layout with SMPS Power



NOTES:

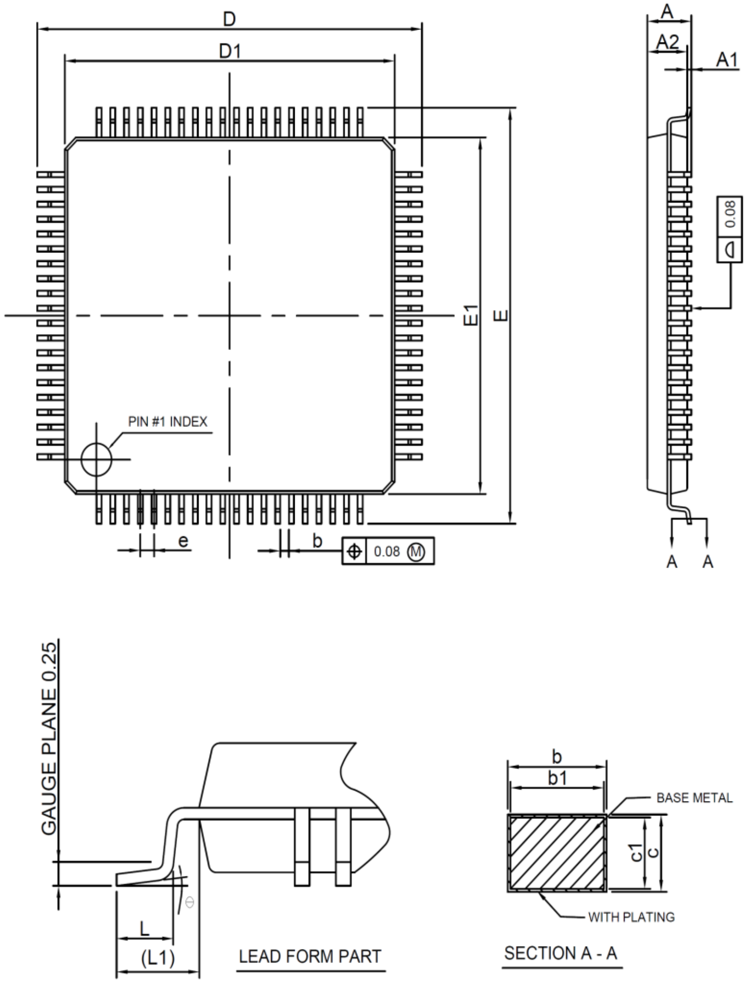
1. Capacitor C1 is to flatten out the voltage of the SMPS power, VCC.
   — Recommended C1: 470uF/25V more
2. Resistor R1 and capacitor C2 are the RC filter for VDD and suppress the ripple of VCC.
   — Recommended R1: 10Ω - 20Ω
   — Recommended C2: 47uF/25V more
   — The R1 and C2 should be as close by the C3 as possible.
3. Capacitor C3 is used for temperature compensation because an electrolytic capacitor becomes worse characteristics at low temperature.
   — Recommended C3: ceramic capacitor 2.2uF more.
   — C3 should be within 1cm from VDD pin of MCU on the PCB layout.
4. The above circuit is recommended to improve noise immunity (EFT, Surge, ESD, etc.) when the SMPS supplies the VDD of MCU.

29 Package Information

29.1 80 LQFP package information

Figure 164. 80 LQFP 12 x 12 Package Outline

REV.	DESCRIPTION	DATE	ENG	APPROVAL
—	GENERATE.	06/12/'17	JH. KIM	YJ. CHAE



COMMON DIMENSIONS
(UNITS OF MEASURE=MILLIMETER)

SYMBOL	MIN	NOM	MAX
A	—	—	1.60
A1	0.05	—	0.15
A2	1.35	1.40	1.45
b	0.17	0.22	0.27
b1	0.17	0.20	0.23
c	0.09	—	0.20
c1	0.09	—	0.16
D	13.80	14.00	14.20
D1	11.80	12.00	12.20
E	13.80	14.00	14.20
E1	11.80	12.00	12.20
e	0.50 BSC		
L	0.45	0.60	0.75
L1	1.00 REF		
θ	0°	3.5°	7°

NOTES:

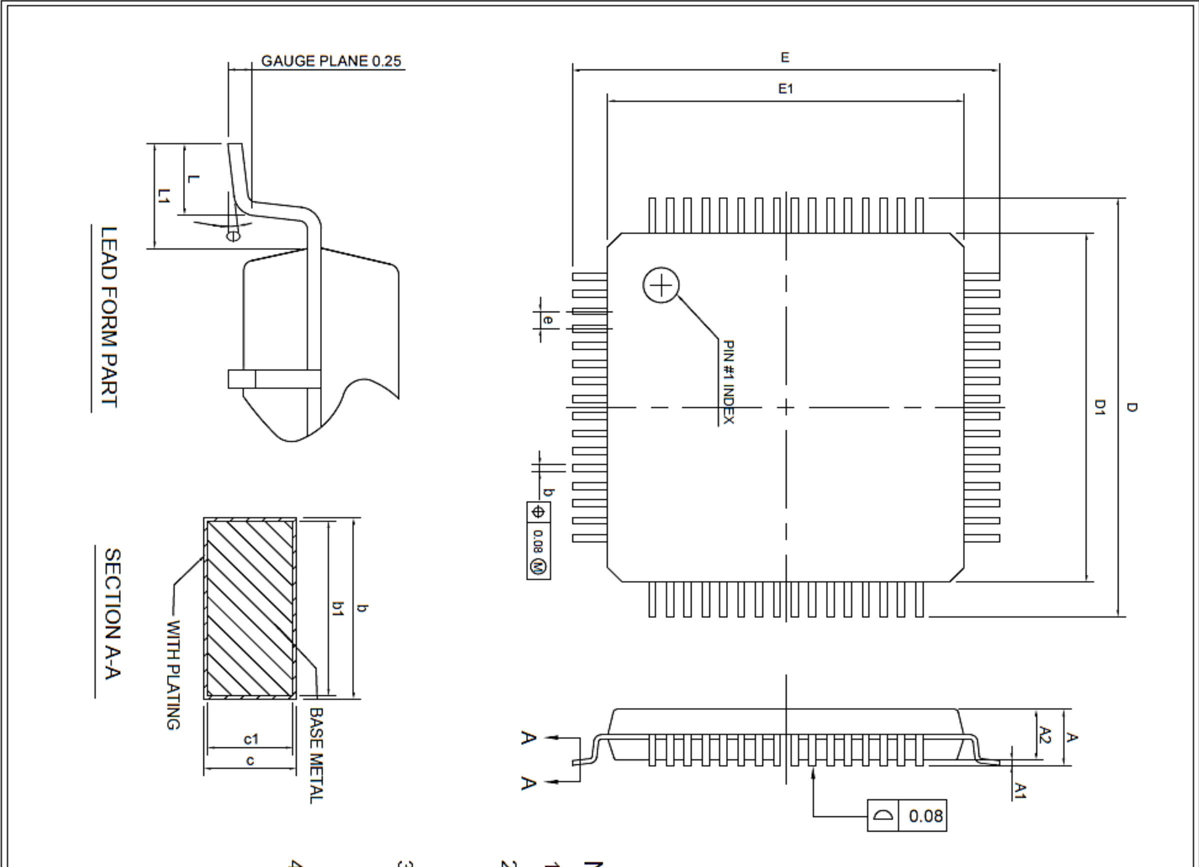
1. ALL DIMENSIONS REFER TO JEDEC STANDARD MS-026-BDD.
2. DIMENSION 'D1' AND 'E1' DO NOT INCLUDE MOLD PROTRUSIONS. ALLOWABLE PROTRUSION IS 0.25MM PER SIDE. 'D1' AND 'E1' ARE MAXIMUM PLASTIC BODY SIZE DIMENSIONS INCLUDING MOLD MISMATCH.
3. DIMENSION 'b' DOES NOT INLCUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL NOT CAUSE THE LEAD WIDTH TO EXCEED THE MAXIMUM 'b' DIMENSION BY MORE THAN 0.08MM.
4. 'A1' IS DEFINED AS THE DISTANCE FROM THE SEATING PLANE TO THE LOWEST POINT ON THE PACKAGE BODY.

LIST OF MATERIAL AND APPLICABLE DOCUMENTS

SCALE: NONE	DATE: 06/12/'17	DRAWN: JS. KANG	TITLE: LQFP 12X12MM 80LD PACKAGE OUTLINE
DIMENSIONAL UNIT: MM	UNTOLERANCED DIMENSIONS	ENGINEER: JH. KIM	
PROJECTION	FRAC: .XX .XXX .XXXX ANGLE	CHECKED: YJ. CHAE	
UNLESS SPECIFIED		APPROVED: YJ. CHAE	
ABOV SEMICONDUCTOR		CAD NAME: OLQ-1212080	DRAWING NUMBER: OLQ-1212080
			REV.: —
			SHEET: 1 OF 1

29.2 64 LQFP package information

Figure 165. 64 LQFP 10 x 10 Package Outline



NOTES:

1. ALL DIMENSION REFER TO JEDEC STANDARD MS-026-BCD.
2. DIMENSIONS 'D1' AND 'E1' DO NOT INCLUDE MOLD PROTRUSION. ALLOWABLE PROTRUSION IS 0.25MM PER SIDE. 'D1' AND 'E1' ARE MAXIMUM PLASTIC BODY SIZE DIMENSIONS INCLUDING MOLD MISMATCH.
3. DIMENSION 'b' DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL NOT CAUSE THE LEAD WIDTH TO EXCEED THE MAXIMUM 'b' DIMENSION BY MORE THAN 0.08MM.
4. 'A1' IS DEFINED AS THE DISTANCE FROMTHE SEATING PLANE TO THE LOWEST POINT ON THE PACKAGE BODY.

REV.	DESCRIPTION	DATE	ENG	APPROVAL
—	GENERATE.	04/18/'17	JH. KIM	YJ. CHAE

COMMON DIMENSIONS
(UNITS OF MEASURE=MILLIMETER)

SYMBOL	MIN	NOM	MAX
A	—	—	1.60
A1	0.05	—	0.15
A2	1.35	1.40	1.45
b	0.17	0.22	0.27
b1	0.17	0.20	0.23
c	0.09	—	0.20
c1	0.09	—	0.16
D	11.80	12.00	12.20
D1	9.80	10.00	10.20
E	11.80	12.00	12.20
E1	9.80	10.00	10.20
e		0.50 BSC	
L	0.45	0.60	0.75
L1		1.00 REF	
θ	0°	3.5°	7°

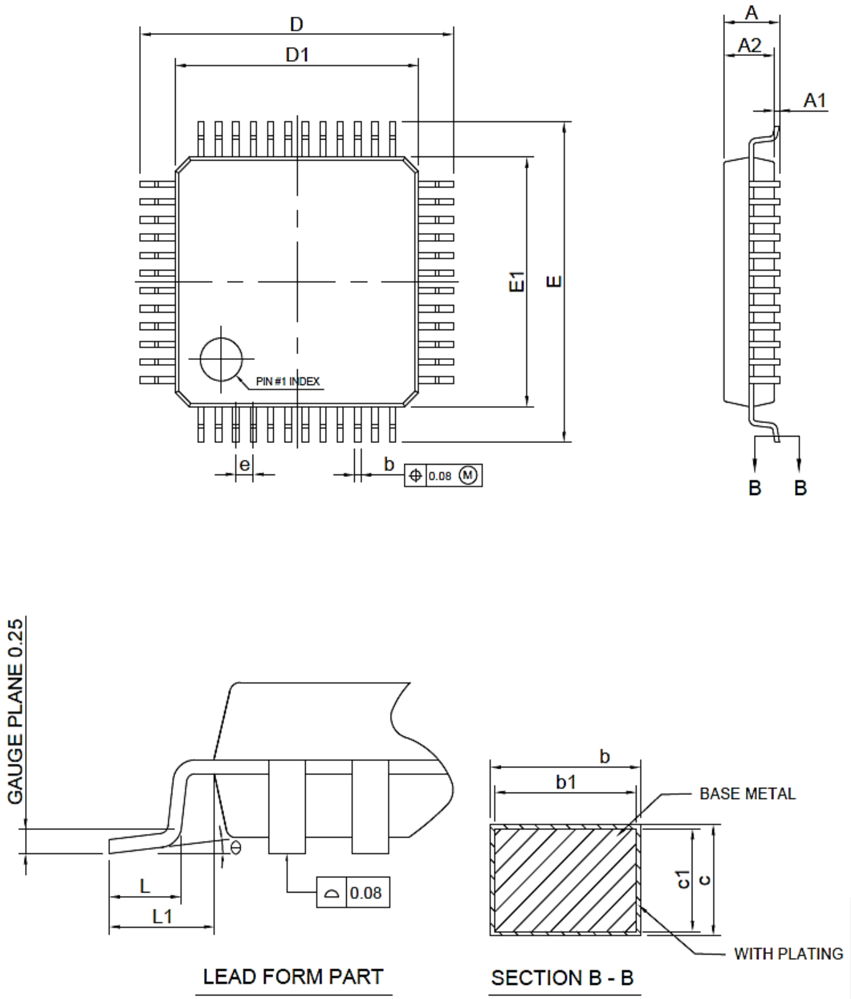
LIST OF MATERIAL AND APPLICABLE DOCUMENTS

SCALE: NONE	DATE: 04/18/'17	DRAWN: JS. KANG	TITLE: LQFP 10X10MM 64LD PACKAGE OUTLINE
DIMENSIONAL UNIT: MM	UNTOLERANCED DIMENSIONS FRAC: .XX .XXX .XXXX ANGLE	ENGINEER: JH. KIM	
PROJECTION UNLESS SPECIFIED		CHECKED: YJ. CHAE	
		APPROVED: YJ. CHAE	
ABOV SEMICONDUCTOR		CAD NAME: OLQ-1010064	DRAWING NUMBER: OLQ-1010064 / REV.: — / SHEET: 1 OF 1

29.3 48 LQFP package information

Figure 166. 48 LQFP 07 x 07 Package Outline

REV.	DESCRIPTION	DATE	ENG	APPROVAL
—	GENERATE.	04/18/'17	JH. KIM	YJ. CHAE



COMMON DIMENSIONS
(UNITS OF MEASURE=MILLIMETER)

SYMBOL	MIN	NOM	MAX
A	—	—	1.60
A1	0.05	—	0.15
A2	1.35	1.40	1.45
b	0.17	0.22	0.27
b1	0.17	0.20	0.23
c	0.09	—	0.20
c1	0.09	—	0.16
D	8.80	9.00	9.20
D1	6.80	7.00	7.20
E	8.80	9.00	9.20
E1	6.80	7.00	7.20
e		0.50 BSC	
L	0.45	0.60	0.75
L1		1.00 REF	
θ	0°	3.5°	7°

NOTES:

1. ALL DIMENSIONS REFER TO JEDEC STANDARD MS-026-BBC.
2. DIMENSION 'D1' AND 'E1' DO NOT INCLDUE MOLD PROTRUSIONS. ALLOWABLE PROTRUSION IS 0.25MM PER SIDE. 'D1' AND 'E1' ARE MAXIMUM PLASTIC BODY DIMENSIONS INCLUDING MOLD MISMATCH.
3. DIMENSION 'b' DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMABR PROTRUSION SHALL NOT CAUSE THE LEAD WIDTH TO EXCEED THE MAXIMUM 'b' DIMENSION BY MORE THAN 0.08MM.
4. 'A1' IS DEFINED AS THE DISTANCE FROM THE SEATING PLANE TO THE LOWEST POINT ON THE PACKAGE BODY.

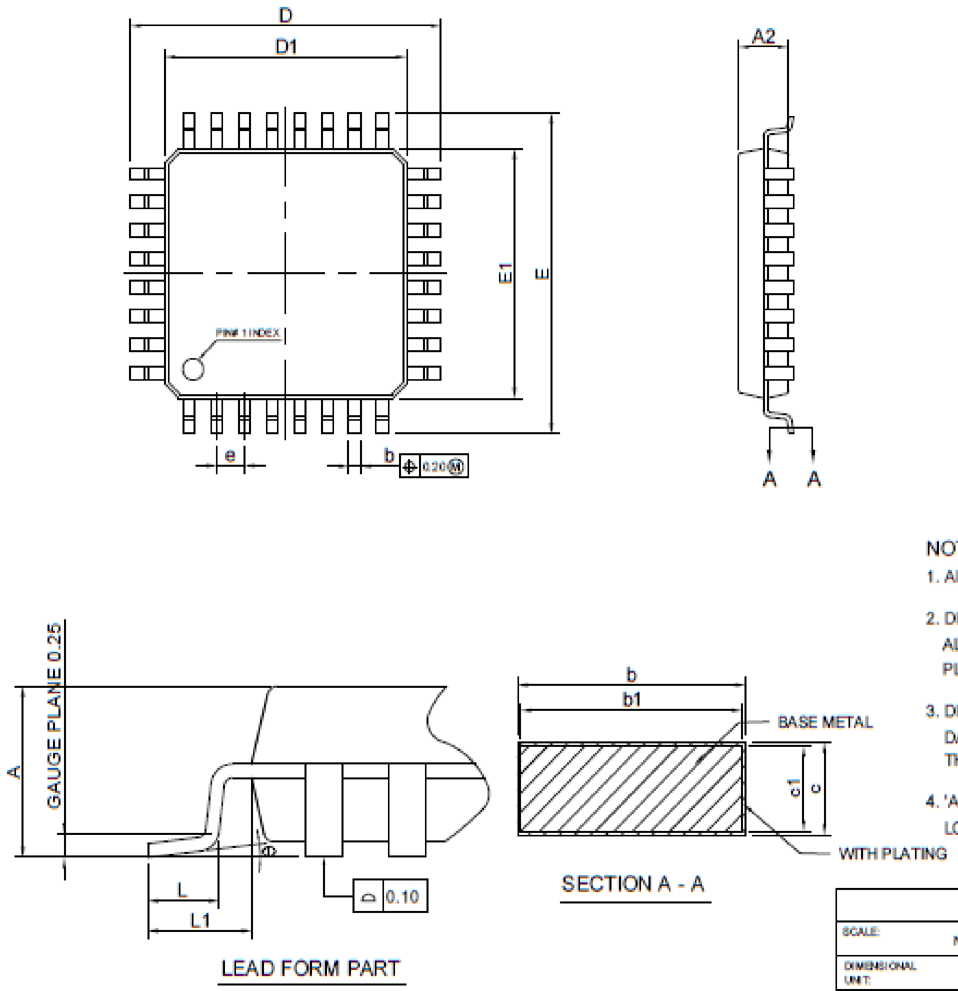
LIST OF MATERIAL AND APPLICABLE DOCUMENTS

SCALE: NONE	DATE:	DRAWN: JS. KANG	TITLE: LQFP 7X7MM 48LD PACKAGE OUTLINE
DIMENSIONAL UNIT: MM	UNTOLERANCED DIMENSIONS	ENGINEER: YJ. CHAE	
PROJECTION UNLESS SPECIFIED	FRAC: .XX .XXX .XXXX ANGLE	CHECKED: YJ. CHAE	
		APPROVED: JH. KIM	
ABOV SEMICONDUCTOR		CAD NAME: OLQ-0707048	DRAWING NUMBER: OLQ-0707048 / REV. — / SHEET: 1 OF 1

29.4 32 LQFP package information

Figure 167. 32 LQFP 07 x 07 Package Outline

REV.	DESCRIPTION	DATE	ENG	APPROVAL
—	GENERATE.	04/20/'17	JH. KIM	YJ. CHAE



COMMON DIMENSIONS
(UNITS OF MEASURE=MILLIMETER)

SYMBOL	MIN	NOM	MAX
A	—	—	1.60
A1	0.05	—	0.20
A2	1.35	1.40	1.45
b	0.30	0.37	0.42
b1	0.30	0.35	0.40
c	0.09	—	0.20
c1	0.09	—	0.16
D	8.80	9.00	9.20
D1	6.80	7.00	7.20
E	8.80	9.00	9.20
E1	6.80	7.00	7.20
e		0.80 BSC	
L	0.45	0.60	0.75
L1	1.00 REF		
θ	0°	3.5°	7°

NOTES:

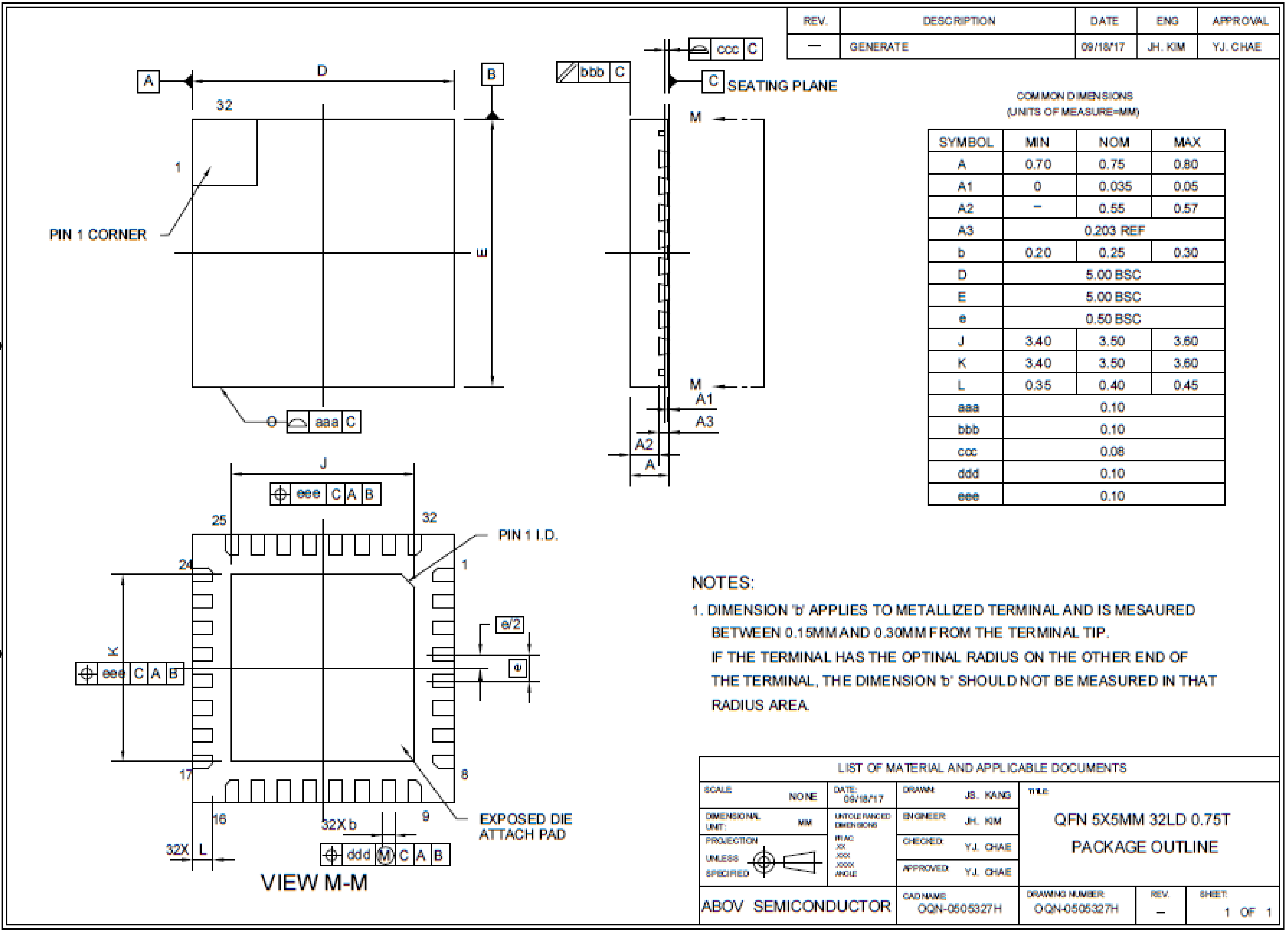
1. ALL DIMENSIONS REFER TO JEDEC STANDARD MS-026-BBA.
2. DIMENSIONS 'D1' AND 'E1' DO NOT INCLUDE MOLD PROTRUSIONS. ALLOWABLE PROTRUSION IS 0.25MM PER SIDE. 'D1' AND 'E1' ARE MAXIMUM PLACTIS BODY SIZE DIMENSIONS INCLUDING MOLD MISMATCH.
3. DIMENSION 'b' DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL NOT CAUSE THE LEAD WIDTH TO EXCEED THE MAXIMUM 'b' DIMENSION BY MORE THAN 0.08MM.
4. 'A1' IS DEFINED AS THE DISTANCE FROM THE SEATING PLANE TO THE LOWEST POINT ON THE PACKAGE BODY.

LIST OF MATERIAL AND APPLICABLE DOCUMENTS

SCALE: NONE	DATE: 04/20/'17	DRAWN: JS. KANG	TITLE: LQFP 7X7MM 32LD PACKAGE OUTLINE
DIMENSIONAL UNIT: MM	UNTOLERANCED DIMENSIONS	ENGINEER: JH. KIM	
PROJECTION	FRAC: .XX .XXX .XXXX ANGLE	CHECKED: YJ. CHAE	
UNLESS SPECIFIED		APPROVED: YJ. CHAE	
ABOV SEMICONDUCTOR		CAD NAME: OLQ-0707032	DRAWING NUMBER: OLQ-0707032 / REV: — / SHEET: 1 OF 1

29.5 32 QFN package information

Figure 168. 32 QFN 05 x 05 Package Outline



REV.	DESCRIPTION	DATE	ENG	APPROVAL
–	GENERATE	09/18/17	JH. KIM	YJ. CHAE

COMMON DIMENSIONS
(UNITS OF MEASURE=MM)

SYMBOL	MIN	NOM	MAX
A	0.70	0.75	0.80
A1	0	0.035	0.05
A2	–	0.55	0.57
A3		0.203 REF	
b	0.20	0.25	0.30
D		5.00 BSC	
E		5.00 BSC	
e		0.50 BSC	
J	3.40	3.50	3.60
K	3.40	3.50	3.60
L	0.35	0.40	0.45
aaa		0.10	
bbb		0.10	
ccc		0.08	
ddd		0.10	
eee		0.10	

NOTES:

1. DIMENSION 'b' APPLIES TO METALLIZED TERMINAL AND IS MESAURED BETWEEN 0.15MM AND 0.30MM FROM THE TERMINAL TIP.
   IF THE TERMINAL HAS THE OPTINAL RADIUS ON THE OTHER END OF THE TERMINAL, THE DIMENSION 'b' SHOULD NOT BE MEASURED IN THAT RADIUS AREA.

LIST OF MATERIAL AND APPLICABLE DOCUMENTS

SCALE NONE	DATE: 09/18/17	DRAWN JS. KANG	TITLE QFN 5X5MM 32LD 0.75T PACKAGE OUTLINE
DIMENSIONAL UNIT: MM	UNTOLERANCED DIMENSIONS	ENGINEER JH. KIM	
PROJECTION UNLESS SPECIFIED	FRAC .XX .XXX .XXXX ANGLE	CHECKED YJ. CHAE	
		APPROVED YJ. CHAE	
ABOV SEMICONDUCTOR	CAD NAME OQN-0505327H	DRAWING NUMBER OQN-0505327H	REV. – SHEET 1 OF 1

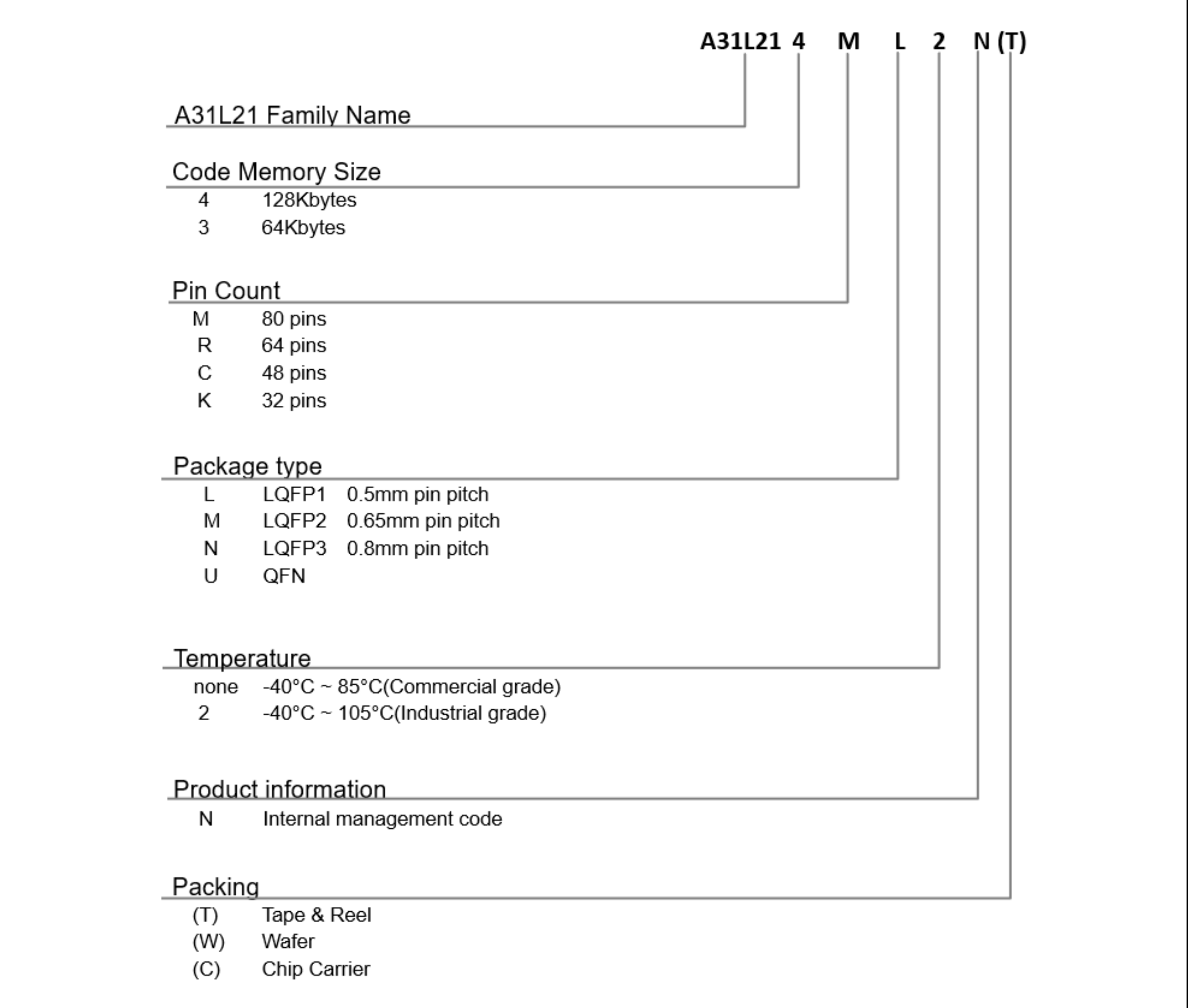
30 Ordering information

Table 139. A31L21x Series Ordering Information

Part Number	Flash	SRAM	USART	UART	LPUART	SC	I2C	SPI	TIMER	LCD	ADC	I/O	Package
A31L214ML	128KB	20KB	2	1	2	1	3	4	9	39x8	16ch	73	80LQFP-1212
A31L214RL*	128KB	20KB	2	1	2	1	3	4	9	27x8	16ch	57	64LQFP-1010
A31L214CL*	128KB	20KB	2	1	2	1	3	3	8	15x4	8ch	41	48LQFP-0707
A31L214KN*	128KB	20KB	2	1	2	1	3	1	3	–	8ch	28	32LQFP-0707
A31L214KU*	128KB	20KB	2	1	2	1	3	1	4	–	8ch	28	32QFN-0505
A31L213ML*	64KB	20KB	2	1	2	1	3	4	9	39x8	16ch	73	80LQFP-1212
A31L213RL*	64KB	20KB	2	1	2	1	3	4	9	27x8	16ch	57	64LQFP-1010
A31L213CL*	64KB	20KB	2	1	2	1	3	3	8	15x4	8ch	41	48LQFP-0707
A31L213KN*	64KB	20KB	2	1	2	1	3	1	3	–	8ch	28	32LQFP-0707
A31L213KU*	64KB	20KB	2	1	2	1	3	1	4	–	8ch	28	32QFN-0505

* For available options or further information on the devices marked with “*”, please contact the ABOV sales office.

Figure 169. A31L21x Series Numbering Nomenclature



A31L21 4 M L 2 N (T)

A31L21 Family Name

Code Memory Size

4	128Kbytes
3	64Kbytes

Pin Count

M	80 pins
R	64 pins
C	48 pins
K	32 pins

Package type

L	LQFP1	0.5mm pin pitch
M	LQFP2	0.65mm pin pitch
N	LQFP3	0.8mm pin pitch
U	QFN	

Temperature

none	-40°C ~ 85°C(Commercial grade)
2	-40°C ~ 105°C(Industrial grade)

Product information

N	Internal management code

Packing

(T)	Tape & Reel
(W)	Wafer
(C)	Chip Carrier

NOTE: For more information on any aspect of this device, please contact your nearest distributor or the ABOV sales office.

31 Development Tools

This chapter introduces various development tools for the A31L21x series. ABOV offers software tools, debuggers, and programmers to help users in generating right results to match target applications. ABOV supports the entire development ecosystem for the customers.

31.1 Compiler

ABOV semiconductor does not provide any compiler for the A31L21x series. However, since the A31L21x series has the ARM's high-speed 32-bit Cortex-M0+ Core as a CPU, users can use all kinds of third party's standard compiler such as Keil C Compiler.

These compilers' output debug information can be integrated with our A-Link and A-Link Pro. For more information regarding the A-Link and A-Link Pro, please visit ABOV website www.abovsemi.com.

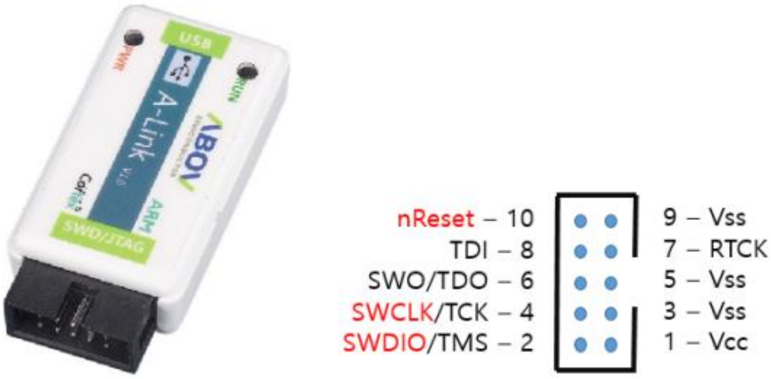
31.2 Debugger

The A-Link and A-Link Pro support ABOV Semiconductor's A31L21x MCU emulation in SWD Interface. The A-Link and A-Link Pro use two wires interfacing between PC and MCU, which is attached to user's system. The A-Link and A-Link Pro can read or change the value of MCU's internal memory and I/O peripherals. In addition, the A-Link and A-Link Pro control MCU's internal debugging logic. This means A-Link and A-Link Pro control emulation, step run, monitoring and many more functions regarding debugging.

The A-Link and A-Link Pro run underneath MS operating system such as MS-Windows NT/2000/XP/Vista/7/8/8.1/10 (32-bit, 64-bit).

Programming information using the A-Link and A-Link Pro are provided in Figure 170. More detailed information about the A-Link and A-Link Pro, please visit our website www.abovsemi.com and download the debugger S/W and documents.

Figure 170. A-Link and Pin Descriptions



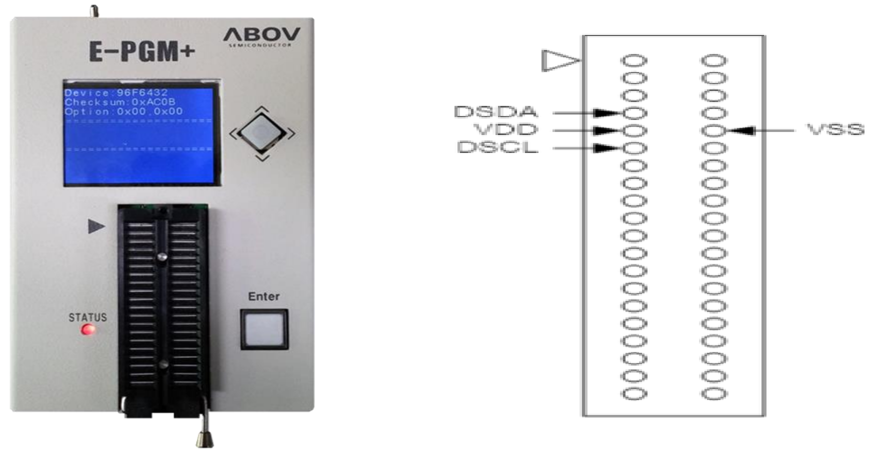
31.3 Programmer

31.3.1 E-PGM+

E-PGM+ is a single programmer, and allows a user to program on the device directly.

- ABOV devices supported
- 2 to 5 times faster than S-PGM+
- Main controller: 32-bit MCU @ 72MHz
- Buffer memory: 1MB

Figure 171. E-PGM+ (Single Writer) and Pin Descriptions



31.3.2 Gang programmer

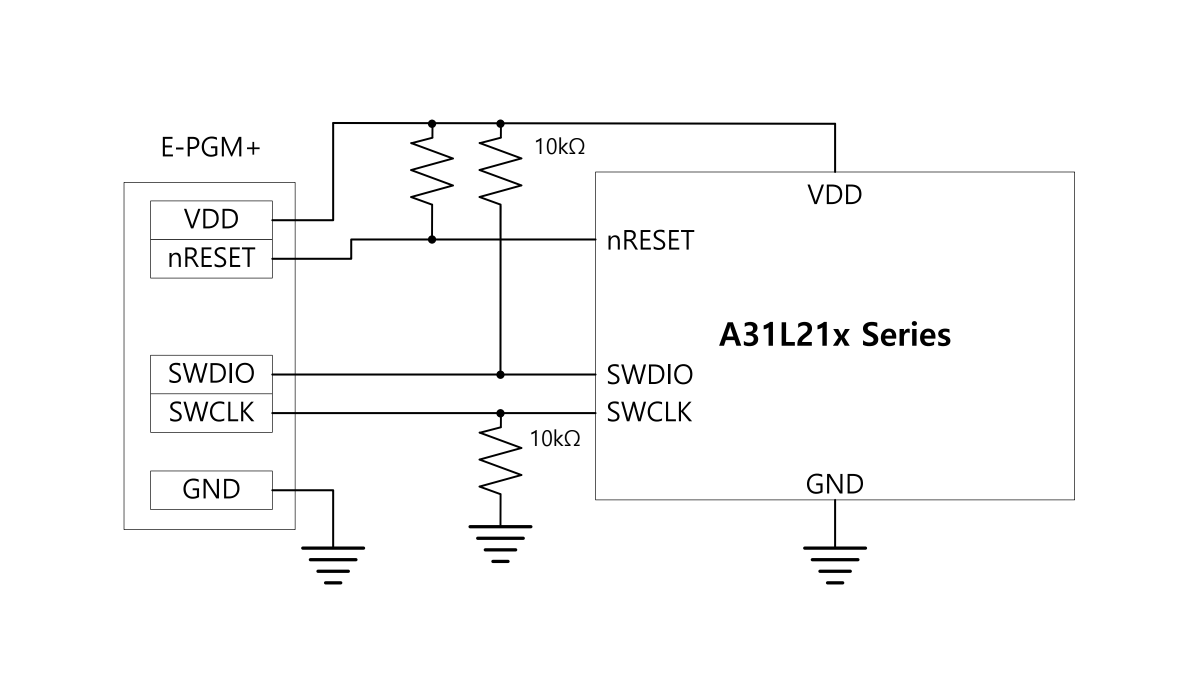
E-Gang4 and E-Gang6 allows a user to program on multiple devices at a time. They run not only in PC controlled mode but also in standalone mode without PC control. USB interface is available and it is easy to connect to the handler.

Figure 172. E-Gang4 and E-Gang6 (for Mass Production)

31.4 SWD debug mode and E-PGM+ connection

Connections for SWD debugger interface or E-PGM+ is described in Figure 173.

Figure 173. Connection between A31L21x Series and E-PGM+ using SWD Debugger Interface



Revision History

Revision	Date	Notes
1.00	Feb. 27, 2023	Initial release

Korea

Regional Office, Seoul
R&D, Marketing & Sales
8th Fl., 330, Yeongdong-daero,
Gangnam-gu, Seoul,
06177, Korea

Tel: +82-2-2193-2200
Fax: +82-2-508-6903
www.abovsemi.com

HQ, Ochang
R&D, QA, and Test Center
93, Gangni 1-gil, Ochang-eup,
Cheongwon-gun,
Chungcheongbuk-do,
28126, Korea

Tel: +82-43-219-5200
Fax: +82-43-217-3534
www.abovsemi.com

Domestic Sales Manager
Tel: +82-2-2193-2206
Fax: +82-2-508-6903
Email: sales_kr@abov.co.kr

Global Sales Manager
Tel: +82-2-2193-2281
Fax: +82-2-508-6903
Email: sales_gl@abov.co.kr

China Sales Manager
Tel: +86-755-8287-2205
Fax: +86-755-8287-2204
Email: sales_cn@abov.co.kr

ABOV Disclaimer
IMPORTANT NOTICE – PLEASE READ CAREFULLY

ABOV Semiconductor ("ABOV") reserves the right to make changes, corrections, enhancements, modifications, and improvements to ABOV products and/or to this document at any time without notice. ABOV does not give warranties as to the accuracy or completeness of the information included herein. Purchasers should obtain the latest relevant information of ABOV products before placing orders. Purchasers are entirely responsible for the choice, selection, and use of ABOV products and ABOV assumes no liability for application assistance or the design of purchasers' products. No license, express or implied, to any intellectual property rights is granted by ABOV herein. ABOV disclaims all express and implied warranties and shall not be responsible or liable for any injuries or damages related to use of ABOV products in such unauthorized applications. ABOV and the ABOV logo are trademarks of ABOV. All other product or service names are the property of their respective owners. Information in this document supersedes and replaces the information previously supplied in any former versions of this document.