CMOS Single-chip 8-bit MCU with 24CH Touch Switch

Main features

8-bit Microcontroller with high performance M8051 CPU

Basic MCU Function
- 32K bytes Flash Code Memory
- Code Area Protection
- 256 bytes SRAM Data Memory
- 1,792 bytes XRAM

Built-in Analog Function
- Power-On Reset and Low Voltage Indicator Reset
- Internal 16 MHz RC Oscillator

24CH Self Capactitive Touch Switch
- 16-bit Sensing Resolution
- Fast Initial Self-Calibration
- Key Detection Mode : Single/Multi-Mode
- The Improvement of SNR by Bias-Calibration in Analog Sensing Block

LED Driver
- 8COM X 20SEG

8CH 12-bit AD Converter

Peripherals
- 32-bit Free Run Counter
- Timer/Counter : 16-bit X 3CH
- PWM : 16-bit X 3CH
- 2CH Buzzer Driver
- I2C / 2CH USART
- 32.768kHz Crystal for Sub Clock

I/O and Packages
- Up to 34 Programmable I/O Lines with 40QFN
- Up to 30 Programmable I/O Lines with 32SOP/32LQFP
- Up to 26 Programmable I/O Lines with 28SOP
- Up to 18 Programmable I/O Lines with 20SOP

Operating Voltage
- 2.0V ~ 5.5V (@16MHz)
- 2.7V ~ 5.5V (with Touch & LED Driver)

Operating Conditions
- -40°C to 85°C temperature range

Application
- Touch Key Application with LED Driver

A96T218

User's Manual

V 1.08

Revised 09 May, 2023

Revision history

Version	Date	Revision list	Page
1.0	2016.10.27	First Release	**-**
1.1	2016.11.15	Format Standardization. Features & Pin Assignment Modified.	-
1.2	2016.12.22	DC Characteristics Modified (IDD1, IDD2)	27
1.3	2017.04.21	Development Tools Update Package Update WT Contents Modified	5 ~ 9 15 ~ 18 69, 71
1.4	2017.05.31	PSR1 Description Modified (P27/P26)	39
1.5	2017.09.24	Table 8-1 modified, DISPSEL bit is removed. LEDCON & LEDSR modified Table 11-14 modified (COM1~8 to COM0~7) Figure 11-40 modified (Dimming Range, Overlap Time)	35, 137, 138, 141 ~ 143
1.6	2017.12.07	Note added Figure 3-1 to 3-4 and Table 5-1 modified (XCK1 to ACK1) VOH3 removed	11 ~ 14, 19, 28, 123, 137
1.7	2019.07.09	Package 20SOP added	3, 4, 14, 19, 20
1.07	2023.05.09	Changed the format of the revision number to “X.YY” according to internal policy.	
1.08	2024.12.02	Updated the disclaimer.	

1. Overview

1.1 Description

The A96T218 is an advanced CMOS 8-bit microcontroller with 32K bytes of FLASH. This is powerful microcontroller which provides a highly flexible and cost effective solution to many embedded control applications. This provides the following features : 24Ch Self Capacitive Touch switch, 32K bytes of FLASH, 256 bytes of SRAM, 1792 bytes of XRAM, 16-bit timer/counter, Buzzer, Free Run Counter, Watchdog timer, I^2C, USART, on-chip POR, BOD, LVI and 16-bit PWM output, on-chip oscillator, clock circuitry and 8COM X 20SEG LED Driver. The A96T218 also supports power saving modes to reduce power consumption.

Table 1-1 Ordering Information of A96T218

Device name	ROM size	SRAM size	Package
A96T218OUN	32Kbytes FLASH	I:256 bytes X:1792 bytes	40 QFN
A96T218KLN			32 LQFP
A96T218KDN			32 SOP
A96T218GDN			28 SOP
A96T218FDN			20 SOP

1.2 Features

- **CPU**
  - 8 Bit CISC Core (8051 Compatible, 2 Clocks per Cycle)
- **32K Bytes On-chip FLASH**
  - Endurance : 10,000 times
  - Endurance : 100,000 times (1K Data Area, h7C00~h7FFF)
  - Retention : 10 years
- **256 Bytes SRAM**
- **1,792 Bytes XRAM**
- **General Purpose I/O**
  - 34 Ports (P0[7:0], P1[7:0], P2[7:0], P3[7:0], P4[1:0]) : 40-pin Package
  - 30 Ports (P0[7:0], P1[5:0], P2[7:0], P3[7:0]) : 32-pin Package
  - 26 Ports (P0[7:0], P1[5:0], P2[7:0], P3[3:0]) : 28-pin Package
  - 18 Ports (P0[3:0], P1[5:0], P2[7:2], P3[3:2]) : 20-pin Package
- **24-Ch Self Capacitive Touch Switch**
  - 16-bit Sensing Resolutions
  - Fast Initial Self-Calibration.
  - Key Detection Mode : Single/Multi-Mode
  - The Improvement of the SNR by Bias-Calibration in analog sensing block
  - 24CH(40-pin/32-pin)/20CH(28-pin)
  - 12CH(20-pin)
- **LED Driver 8COM X 20SEG**
- **32-bit Free Run Counter**
- **Basic Interval Timer**
- **Timer / Counter**
  - 16 Bits × 3 Channels
- **3-ch 16-bit PWM (using Timer0, 1, 2)**
- **Watch Dog Timer**
- **2-ch Buzzer Driver**
- **I^2C**
- **2-ch USART**
- **12-bit AD Converter**
  - 8 Input Channel
- **Interrupt Sources**
  - External (8)
  - USART(4)
  - WT (1)
  - TOUCH (1)
  - LED (1)
  - ADC(1)
  - Free Run Counter (1)
  - I^2C (1)
  - Timer (3)
  - WDT (1)
  - BIT (1)
  - LVI (1)
- **On-Chip RC-Oscillator**
  - 16MHz (±3%@-40~+85℃)
- **On-Chip WDT-Oscillator**
  - 256kHz OSC (±10%@-40~+85℃)
- **32.768kHz crystal for Sub clock**
- **Power On Reset & Brown-Out Detector**
  - 1.4V (POR)
  - 1.6V (BOD)
- **Programmable Low-Voltage Indicator**
  - 3-Level (2.5V / 3.6V / 4.2V)
- **Minimum Instruction Execution Time**
  - 200ns (@10MHz, NOP Instruction)
- **Power down mode**
  - IDLE, STOP1, STOP2 mode
- **Operating Frequency**
  - 2, 4, 8, 16MHz (internal RC oscillator)
- **Operating Voltage**
  - 2.0V ~ 5.5V (@ 16MHz)
  - 2.7V ~ 5.5V (Touch & LED Driver)
- **Operating Temperature : -40 ~ +85℃**
- **Package Type (Pb Free)**
  - 40 QFN
  - 32 LQFP
  - 32, 28, 20 SOP

1.3 Development tools

1.3.1 Compiler

ABOV semiconductor does not provide any compiler for the A96T218. But the CPU core of A96T218 is M8051 core, you can use all kinds of third party's standard 8051 compiler like Keil C Compiler, Open Source SDCC (Small Device C Compiler). These compilers' output debug information can be integrated with our OCD1 emulator and debugger. Refer to OCD1 manual for more details.

1.3.2 OCD1 Emulator and Debugger

The OCD1 emulator supports ABOV Semiconductor's 8051 series MCU emulation.

The OCD1 interface uses two wires interfacing between PC and MCU which is attached to user's system. The OCD1 can read or change the value of MCU internal memory and I/O peripherals. And also the OCD1 controls MCU internal debugging logic, it means OCD1 controls emulation, step run, monitoring etc.

The OCD1 Debugger program works on Microsoft-Windows NT, 2000, XP, Vista (32-bit), Windows 8, Windows 10 operating system.

If you want to see more details, please refer OCD1 debugger manual. You can download debugger S/W and manual from our web-site.

- P11 (A96T218 DSCL pin)
- P10 (A96T218 DSDA pin)

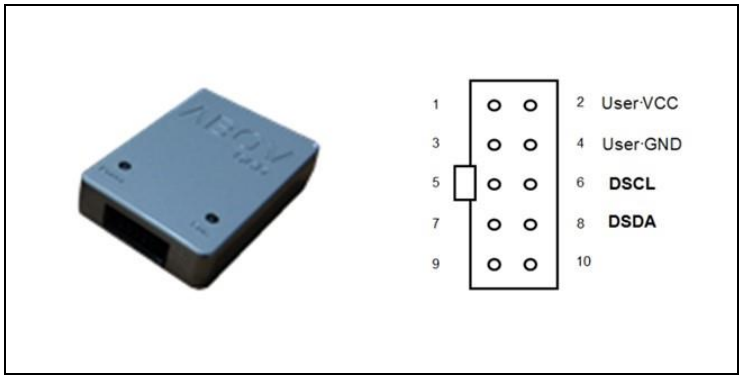


Figure 1.1 On Chip Debugger 1 and Pin description

OCD2 (On Chip Debug) Emulator

- MCU emulation control via 2pin or 3pin OCD interface.
- 2pin interface : OCD2 clock & data.
- 1pin option interface
  - Support device OCD2 mode entry during user S/W is running.
  - Support exact emulation time measurement.
- Higher interface speed than OCD dongle.
- Support newly added debugging specifications.
  - Data access break (1, 2, 4bytes),
  - internal OSC Frequency measurement and trimming, etc.
- Compact size.
- Cost effective emulator.
- Emulation & debugging on the target system directly.
- Real time emulation.
- PC interface : USB.

Debugger

- Operates with OCD and OCD2 emulator H/W.
- Integrated Development Environment (IDE).
  Support docking windows and menus.
- Support Free run, Step run, auto step run.
- Support Symbolic debugging.
- Support Source level debugging.

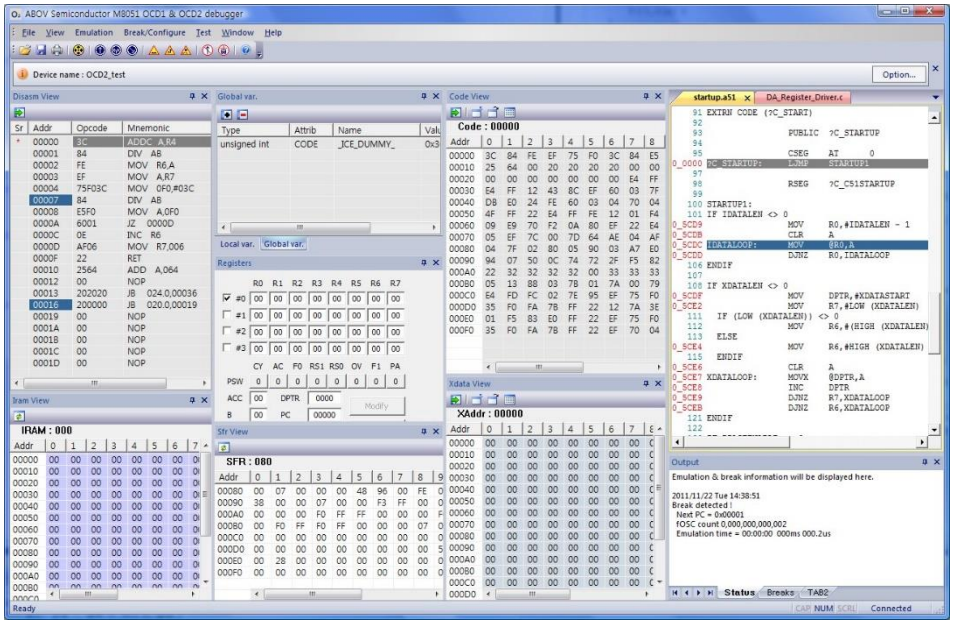


Figure 1.2 OCD Debugger

Support Devices

- MC95xxxx
- MC96xxxx / A96xxxx
- MC97xxxx

1.3.3 Programmer

E-PGM +

- Support ABOV / ADAM devices
- 2~5 times faster than S-PGM+
- Main controller : 32 bit MCU @ 72MHz
- Buffer memory : 1 MByte

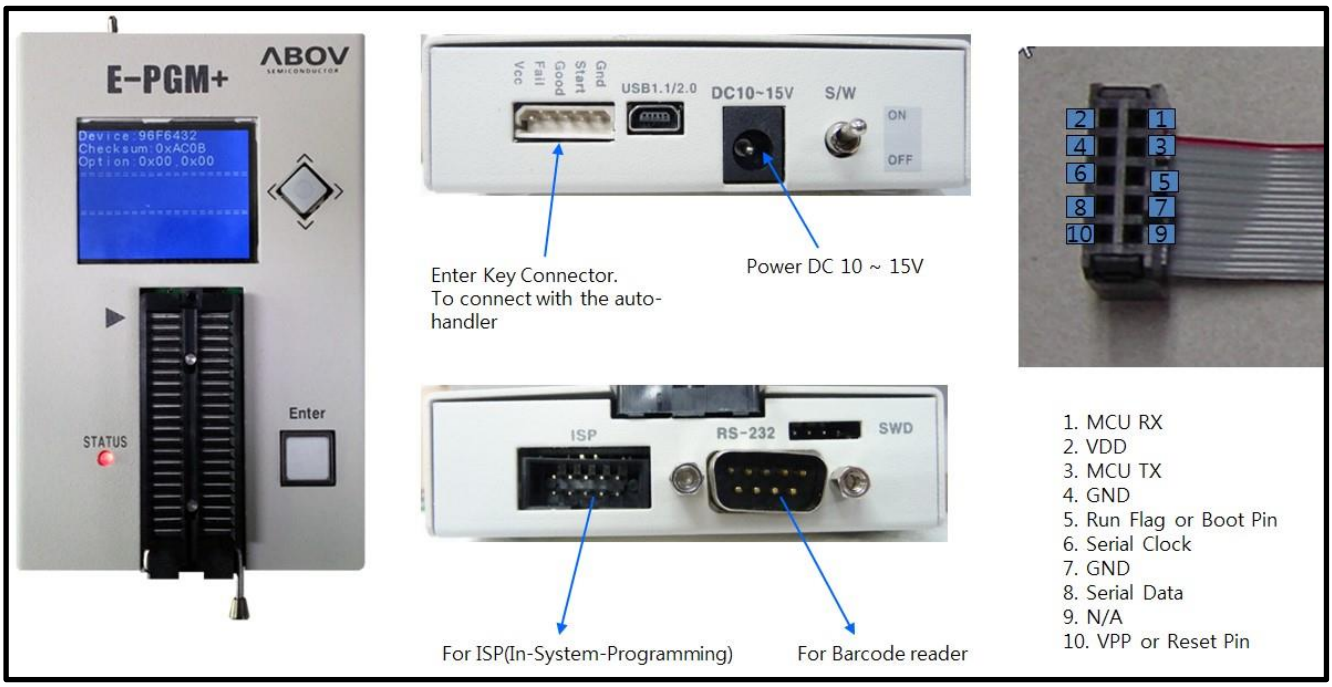


Figure 1.3 PGMplus USB

PGMPlusLC 2

Description

PGMPlusLC2 is for ISP (In System Programming). It is used to write into the MCU Which is already mounted on target board using 10pin cable.

Features

- PGMplusLC2 is low cost writing Tool.
- USB interface is supported.
- Not need USB driver installation.
- Connect the external power adaptor (5v@2A).
- Fast 32-bit Cortex-M3 MCU is used.
- Supported high voltage Max 18V.
- PGMplusLC2 is based on PC environment.
- PGMplusLC2 is faster than PGMplusLC.
- Transmission speed is 64Kbyte/s

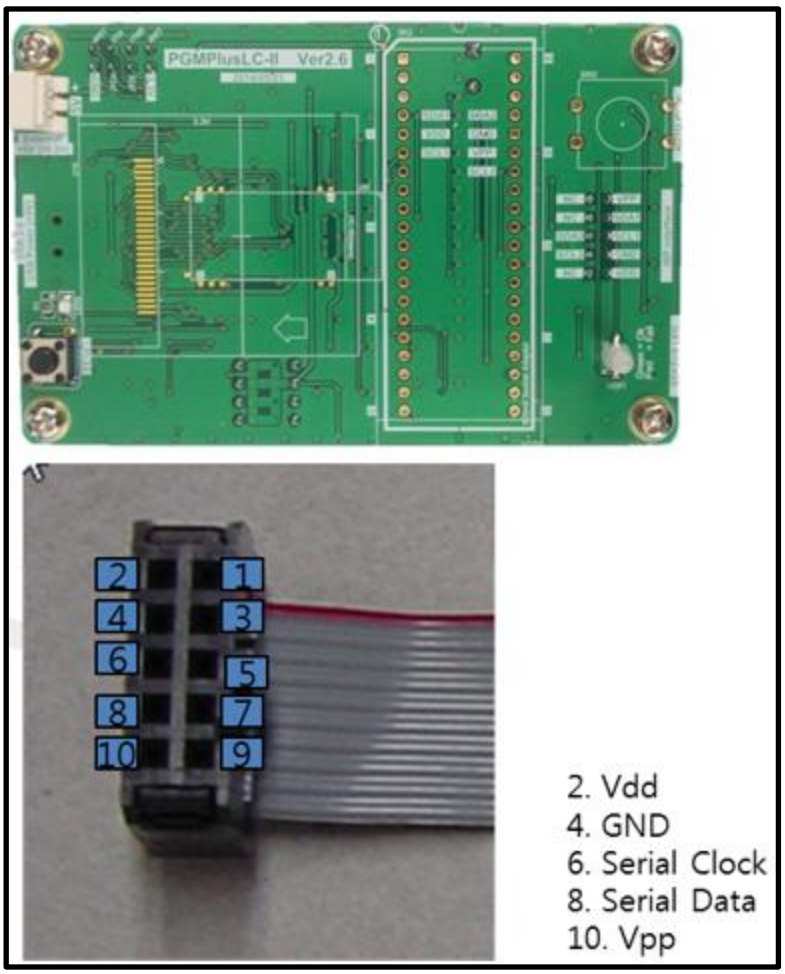


Figure 1.4 PGMplusLC Writer

E-PGM+ Gang4/6

- Product name : **E-PGM+ GANG 4**
- Dimension(x , y, h) : 33.5 x 22.5 x35mm
- Weight : 2.0kg
- Input Voltage : DC Adaptor 15V/2A
- Power Consumption :
- Operating Temp : -10 ~ 40℃
- Storage Temp : -30 ~ 80℃
- Water Proof : No

- Product name : **E-PGM+ GANG 6**
- Dimension(x , y, h) : 148.2 x 22.5 x35mm
- Weight : 2.8kg
- Input Voltage : DC Adaptor 15V/2A
- Power Consumption :
- Operating Temp : -10 ~ 40℃
- Storage Temp : -30 ~ 80℃
- Water Proof : No

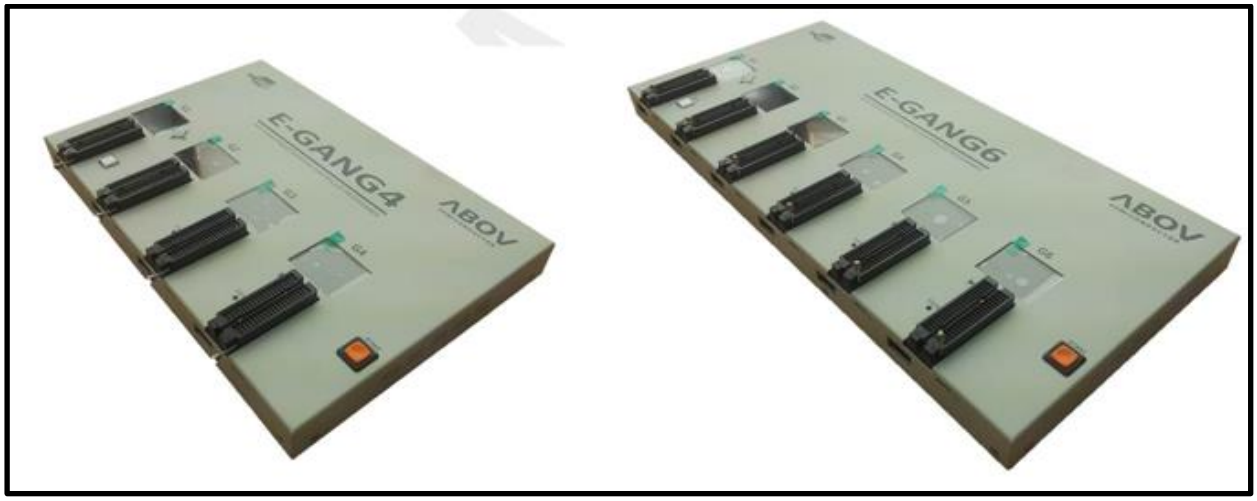


Figure 1.5 Gang Programmer

2. Block Diagram

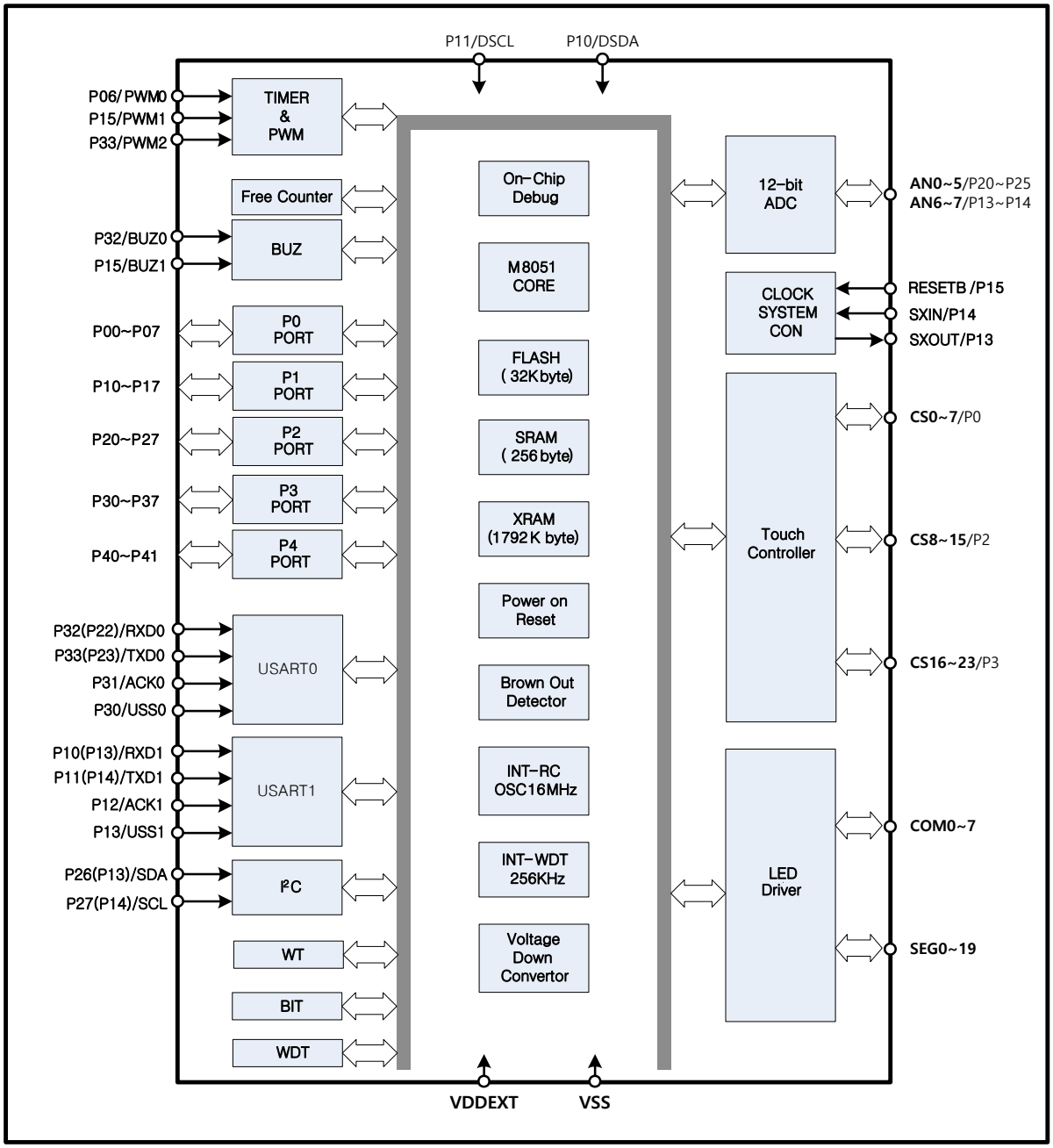


Figure 2-1 A96T218 Block Diagram

3. Pin Assignment

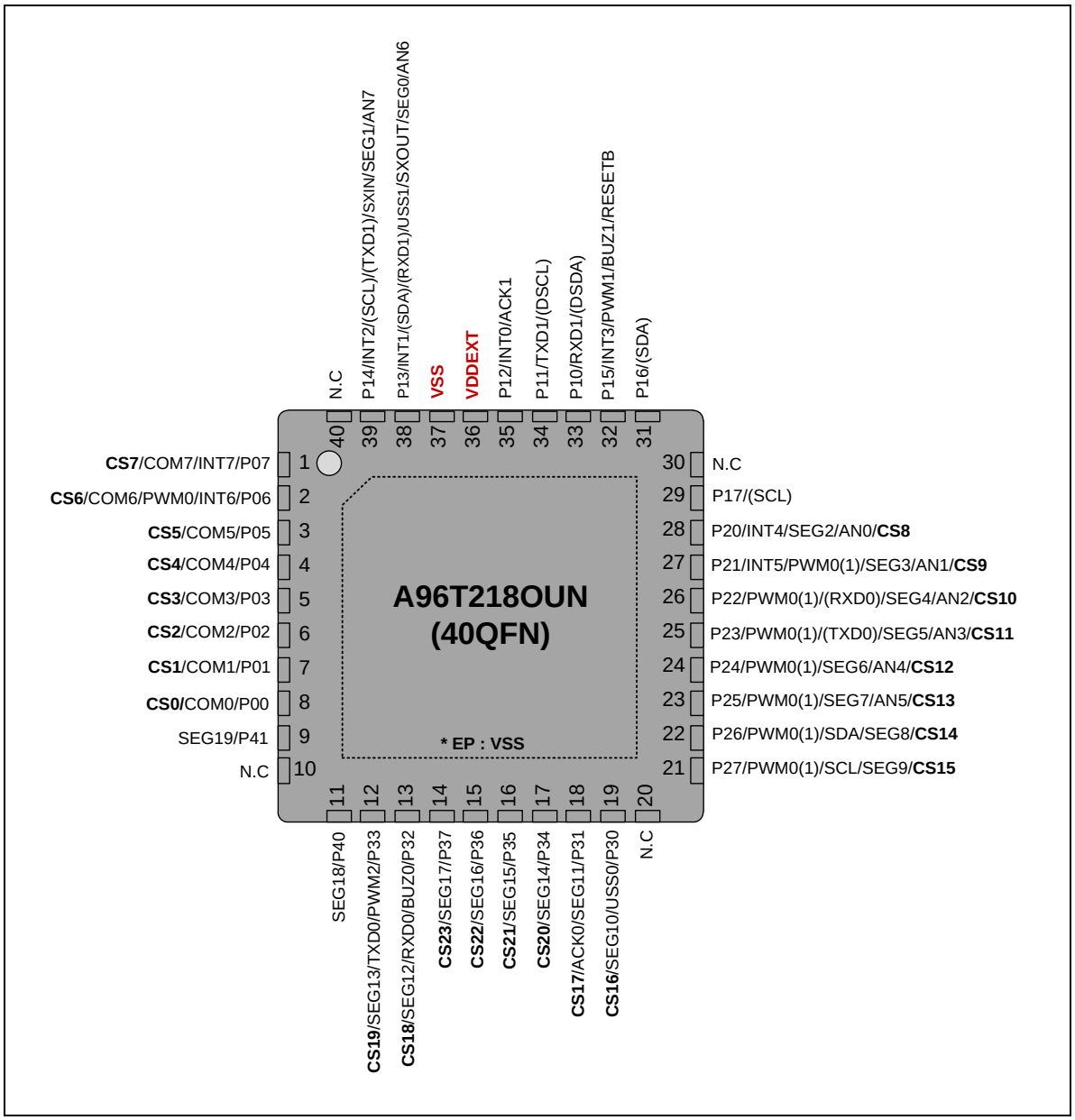


Figure 3-1 A96T218 40 QFN Pin Assignment

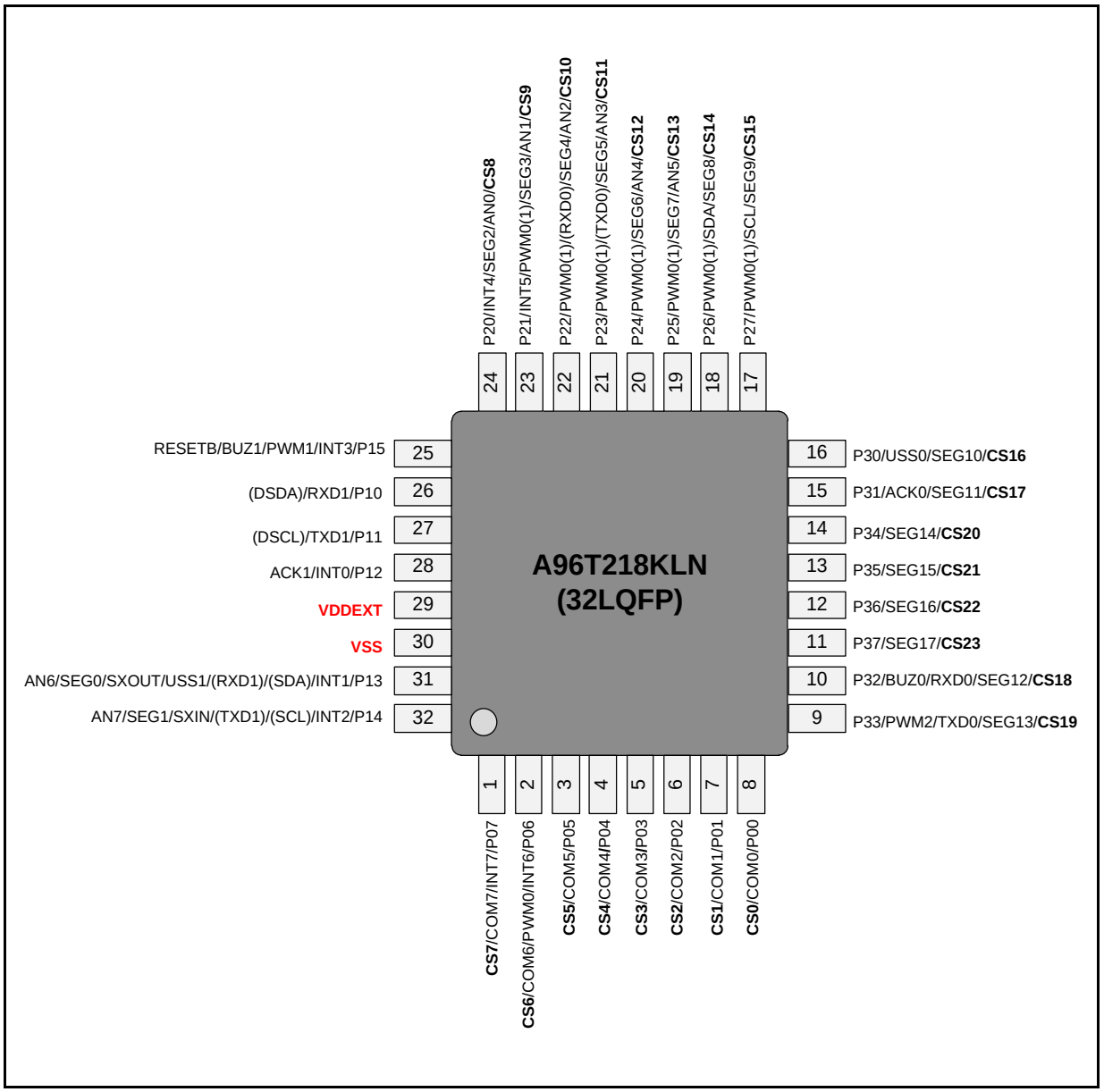


Figure 3-2 A96T218 32 LQFP Pin Assignment

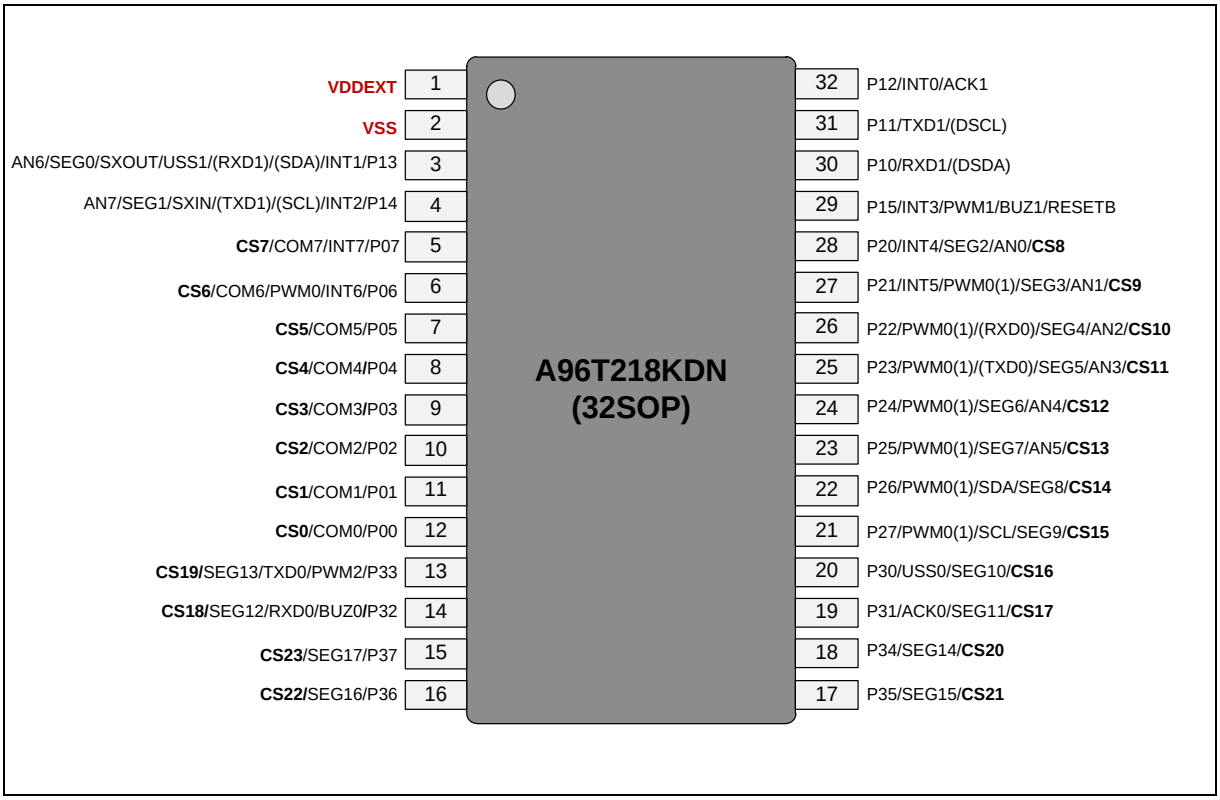


Figure 3-3 A96T218 32 SOP Pin Assignment

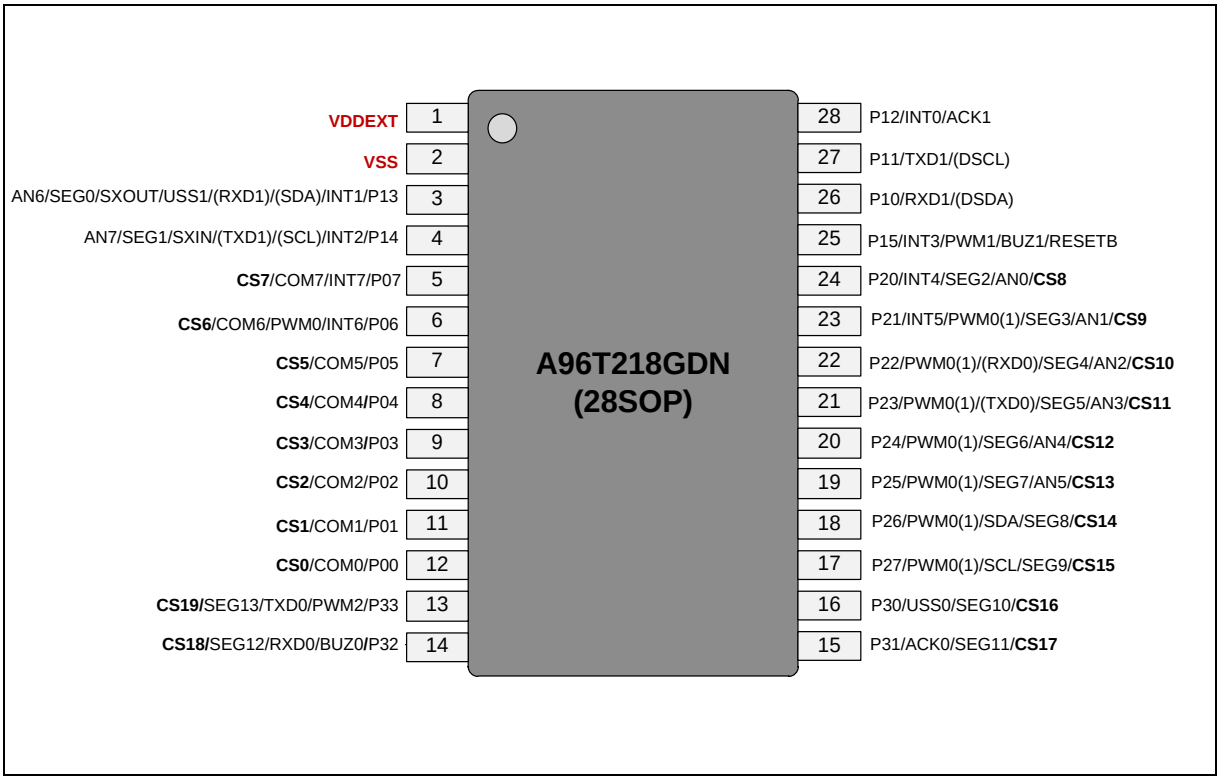


Figure 3-4 A96T218 28 SOP Pin Assignment

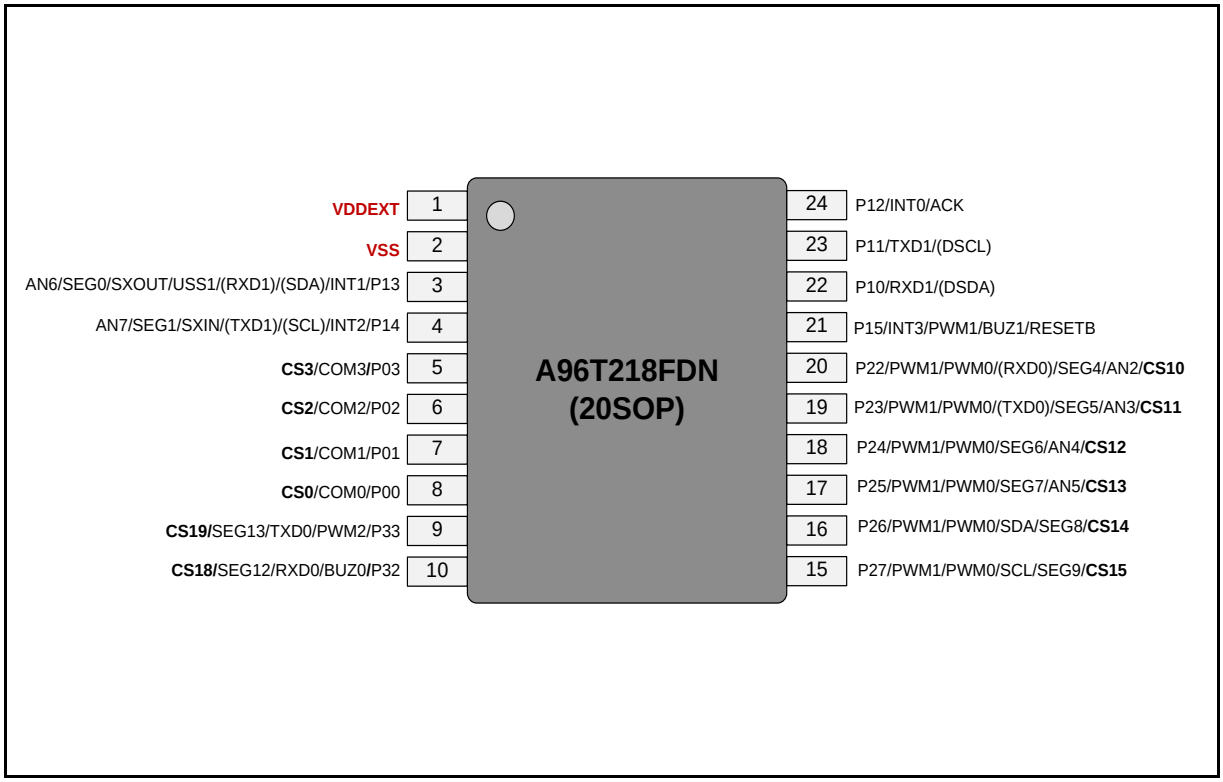


Figure 3-5 A96T218 20SOP assignment

4. Package Diagram

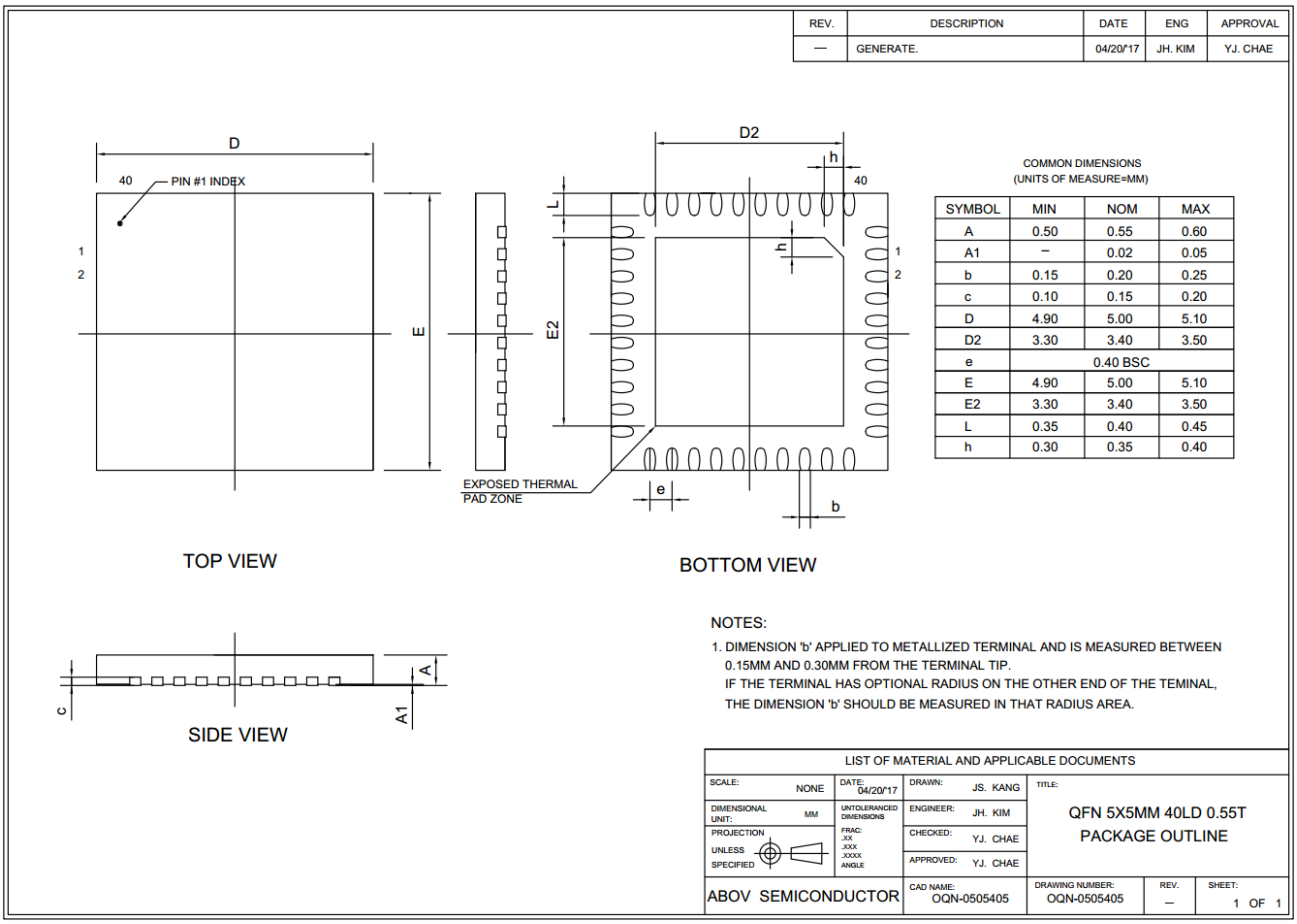
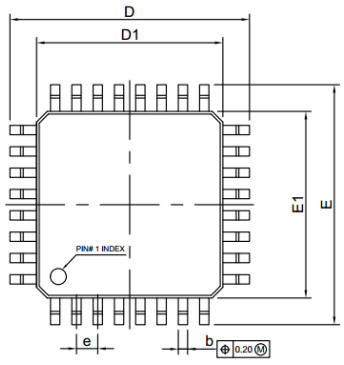
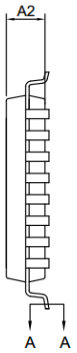


Figure 4-1 40 pin QFN Package

REV.	DESCRIPTION	DATE	ENG	APPROVAL
—	GENERATE.	04/20/'17	JH. KIM	YJ. CHAE



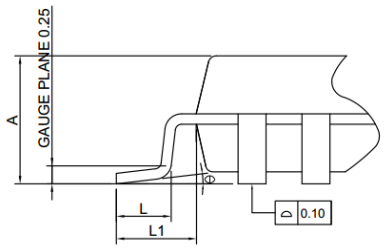


COMMON DIMENSIONS
(UNITS OF MEASURE=MILLIMETER)

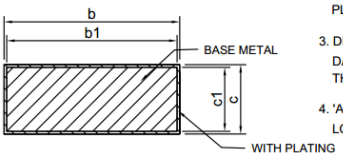
SYMBOL	MIN	NOM	MAX
A	—	—	1.60
A1	0.05	—	0.20
A2	1.35	1.40	1.45
b	0.30	0.37	0.42
b1	0.30	0.35	0.40
c	0.09	—	0.20
c1	0.09	—	0.16
D	8.80	9.00	9.20
D1	6.80	7.00	7.20
E	8.80	9.00	9.20
E1	6.80	7.00	7.20
e		0.80 BSC	
L	0.45	0.60	0.75
L1		1.00 REF	
θ	0°	3.5°	7°

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.



LEAD FORM PART



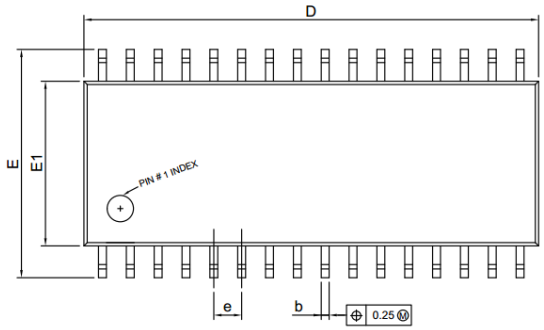
SECTION A - A

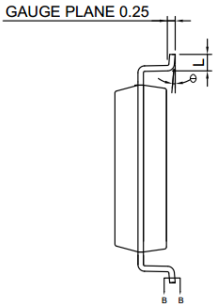
LIST OF MATERIAL AND APPLICABLE DOCUMENTS

SCALE: NONE	DATE: 04/20/'17	DRAWN: JS. KANG	TITLE: LQFP 7X7MM 32LD PACKAGE OUTLINE
DIMENSIONAL UNIT: MM	UNTOLERANCED DIMENSIONS	ENGINEER: JH. KIM	
PROJECTION UNLESS SPECIFIED	FRAC: .XX .XXX .XXXX ANGLE	CHECKED: YJ. CHAE	
		APPROVED: YJ. CHAE	
ABOV SEMICONDUCTOR	CAD NAME: OLQ-0707032	DRAWING NUMBER: OLQ-0707032	REV. — / SHEET: 1 OF 1

Figure 4-2 32 pin LQFP Package

REV.	DESCRIPTION	DATE	ENG	APPROVAL
—	GENERATE.	04/20/'17	JH. KIM	YJ. CHAE



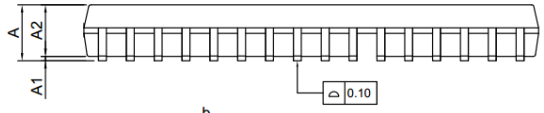


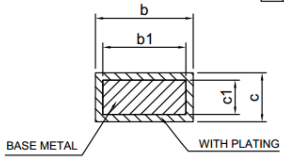
COMMON DIMENSIONS
(UNITS OF MEASURE=MILLIMETER)

SYMBOL	MIN	NOM	MAX
A	—	—	2.60
A1	0.05	—	0.25
A2	2.14	—	2.34
b	0.30	—	0.50
b1	0.30	—	0.48
c	0.10	—	0.33
c1	0.10	—	0.30
D	20.60	—	20.90
E	10.20	—	10.60
E1	7.40	—	7.60
e	1.27 BSC		
L	0.40		1.27
θ	0°	—	8°

NOTES:

1. DIMENSION 'D' DOES NOT INCLUDE MOLD FLASH, PROTRUSIONS OR GATE BURRS. MOLD FLASH, PROTRUSIONS OR GATE BURRS SHALL NOT EXCEED 0.15MM PER END. DIMENSION 'E1' DOES NOT INCLUDE INTERLEAD FLASH OR PROTRUSION. INTERLEAD FLASH OR PROTRUSION SHALL NOT EXCEED 0.25MM PER SIDE.
2. DIMENSION 'b' DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL BE 0.10MM TOTAL IN EXCESS OF THE 'b' DIMENSION MAXIMUM MATERIAL CONDITION.
3. 'A1' IS DEFINED AS THE VERTICAL DISTANCE FROM THE SEATING PLANE TO THE LOWEST POINT ON THE PACKAGE BODY.





SECTION B-B

LIST OF MATERIAL AND APPLICABLE DOCUMENTS

SCALE: NONE	DATE 04/20/'17	DRAWN: JS. KANG	TITLE: SOP 300MIL 32LD 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: OSO-3000032	DRAWING NUMBER: OSO-3000032	REV. — / SHEET: 1 OF 1

Figure 4-3 32 pin SOP Package

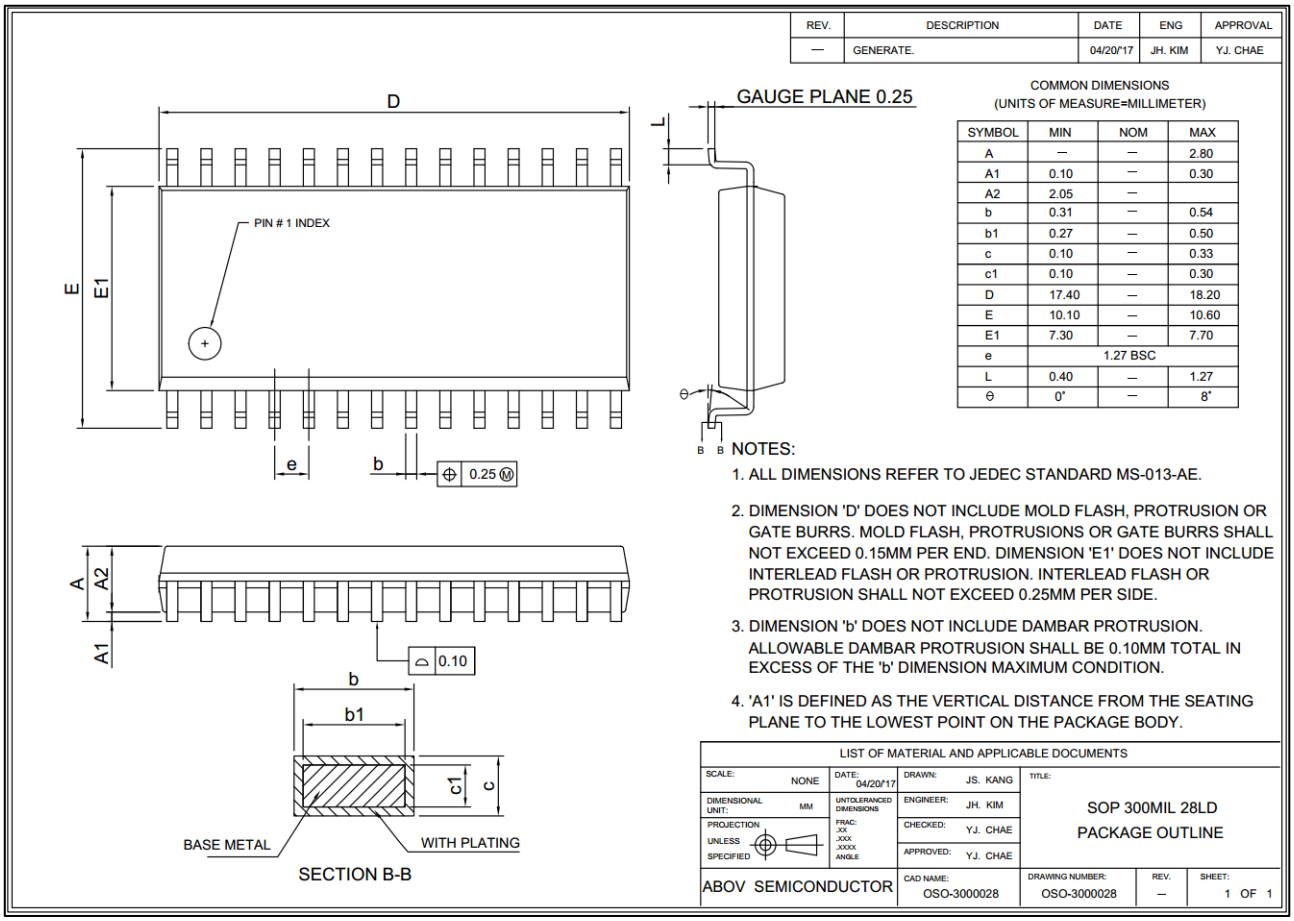
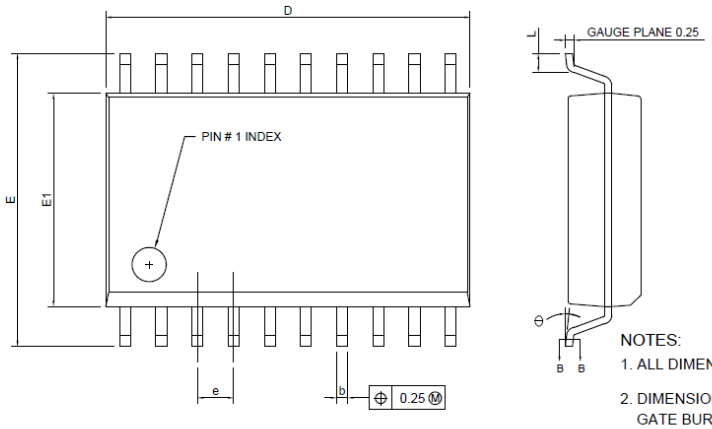


Figure 4-4 28 pin SOP Package

REV.	DESCRIPTION	DATE	ENG	APPROVAL
—	GENERATE.	08/13/'17	YJ. CHAE	YJ. CHAE

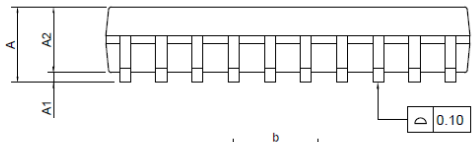


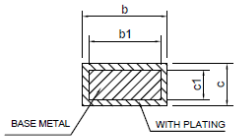
COMMON DIMENSIONS
(UNITS OF MEASURE=MILLIMETER)

SYMBOL	MIN	NOM	MAX
A	—	—	2.65
A1	0.10	—	0.30
A2	2.05	—	—
b	0.31	—	0.51
b1	0.27	—	0.48
c	0.10	—	0.33
c1	0.10	—	0.30
D	12.60	—	13.00
E	10.00	—	10.60
E1	7.30	—	7.70
e	1.27 BSC		
L	0.40	—	1.27
θ	0°	—	8°

NOTES:

1. ALL DIMENSIONS REFER TO JEDEC STANDARD MS-013-AC.
2. DIMENSION 'D' DOES NOT INCLUDE MOLD FLASH, PROTRUSION OR GATE BURRS. MOLD FLASH, PROTRUSIONS OR GATE BURRS SHALL NOT EXCEED 0.15MM PER END. DIMENSION 'E1' DOES NOT INCLUDE INTERLEAD FLASH OR PROTRUSION. INTERLEAD FLASH OR PROTRUSION SHALL NOT EXCEED 0.25MM PER SIDE.
3. DIMENSION 'b' DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL BE 0.10MM TOTAL IN EXCESS OF THE 'b' DIMENSION MAXIMUM CONDITION.
4. 'A1' IS DEFINED AS THE VERTICAL DISTANCE FROM THE SEATING PLANE TO THE LOWEST POINT ON THE PACKAGE BODY.





SECTION B-B

LIST OF MATERIAL AND APPLICABLE DOCUMENTS

SCALE: NONE	DATE: 08/13/'17	DRAWN: JS. KANG	TITLE: SOP 300MIL 20LD 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: OSO-3000020	DRAWING NUMBER: OSO-3000020	REV. — / SHEET: 1 OF 1

Figure 4-5 20 SOP Package

5. Pin Description

Table 5-1 Normal Pin Description

PIN Name				I/O	Function	@RESET	Shared with
20PIN	28PIN	32PIN	40PIN				
P00	P00	P00	P00	I/O	Port P0 8-Bit I/O Port Can be set in input or output mode in 1-bit units Internal pull-up register can be used via software when this port is used as input port Open Drain enable register can be used via software when this port is used as output port	Input	COM0/CS0
P01	P01	P01	P01				COM1/CS1
P02	P02	P02	P02				COM2/CS2
P03	P03	P03	P03				COM3/CS3
-	P04	P04	P04				COM4/CS4
-	P05	P05	P05				COM5/CS5
-	P06	P06	P06				INT6/PWM0/COM6/CS6
-	P07	P07	P07				INT7/COM7/CS7
P10	P10	P10	P10	I/O	Port P1 6-Bit I/O Port Can be set in input or output mode in 1-bit units Internal pull-up register can be used via software when this port is used as input port Open Drain enable register can be used via software when this port is used as output port (The output type of the P12, P16, and P17 pins is open-drain.	Input	RXD1/(DSDA)
P11	P11	P11	P11				TXD1/(DSCL)
P12	P12	P12	P12				ACK1
P13	P13	P13	P13				INT1/(SDA)/(RXD1)/USS1/SXOUT/ SEG0/ AN6
P14	P14	P14	P14				INT2/(SCL)/(TXD1)/SXIN/SEG1/AN7
P15	P15	P15	P15				INT3/PWM1/BUZ1/RESETB
-	-	-	P16				(SDA)
-	-	-	P17				(SCL)
-	P20	P20	P20	I/O	Port P2 8-Bit I/O Port Can be set in input or output mode in 1-bit units Internal pull-up register can be used via software when this port is used as input port Open Drain enable register can be used via software when this port is used as output port	Input	INT4/SEG2/AN0/CS8
-	P21	P21	P21				INT5/PWM1/PWM0/SEG3/AN1/CS9
P22	P22	P22	P22				PWM1/PWM0/(RXD0)/SEG4/AN2/ CS10
P23	P23	P23	P23				PWM1/PWM0/(TXD0)/SEG5/AN3/CS11
P24	P24	P24	P24				PWM1/PWM0/SEG6/AN4/CS12
P25	P25	P25	P25				PWM1/PWM0/SEG7/AN5/CS13
P26	P26	P26	P26				PWM1/PWM0/SDA/SEG8/CS14
P27	P27	P27	P27				PWM1/PWM0/SCL/SEG9/CS15
-	P30	P30	P30	I/O	Port P3 8-Bit I/O Port Can be set in input or output mode in 1-bit units Internal pull-up register can be used via software when this port is used as input port Open Drain enable register can be used via software when this port is used as output port	Input	USS0/SEG10/CS16
-	P31	P31	P31				ACK0/SEG11/CS17
P32	P32	P32	P32				BUZ0/RXD0/SEG12/CS18
P33	P33	P33	P33				PWM2/TXD0/SEG13/CS19
-	-	P34	P34				SEG14/CS20
-	-	P35	P35				SEG15/CS21
-	-	P36	P36				SEG16/CS22
-	-	P37	P37				SEG17/CS23
-	-	-	P40		Port P4 2-Bit I/O Port	Input	SEG18
-	-	-	P41				SEG19

6. Port Structures

6.1 General Purpose I/O Port

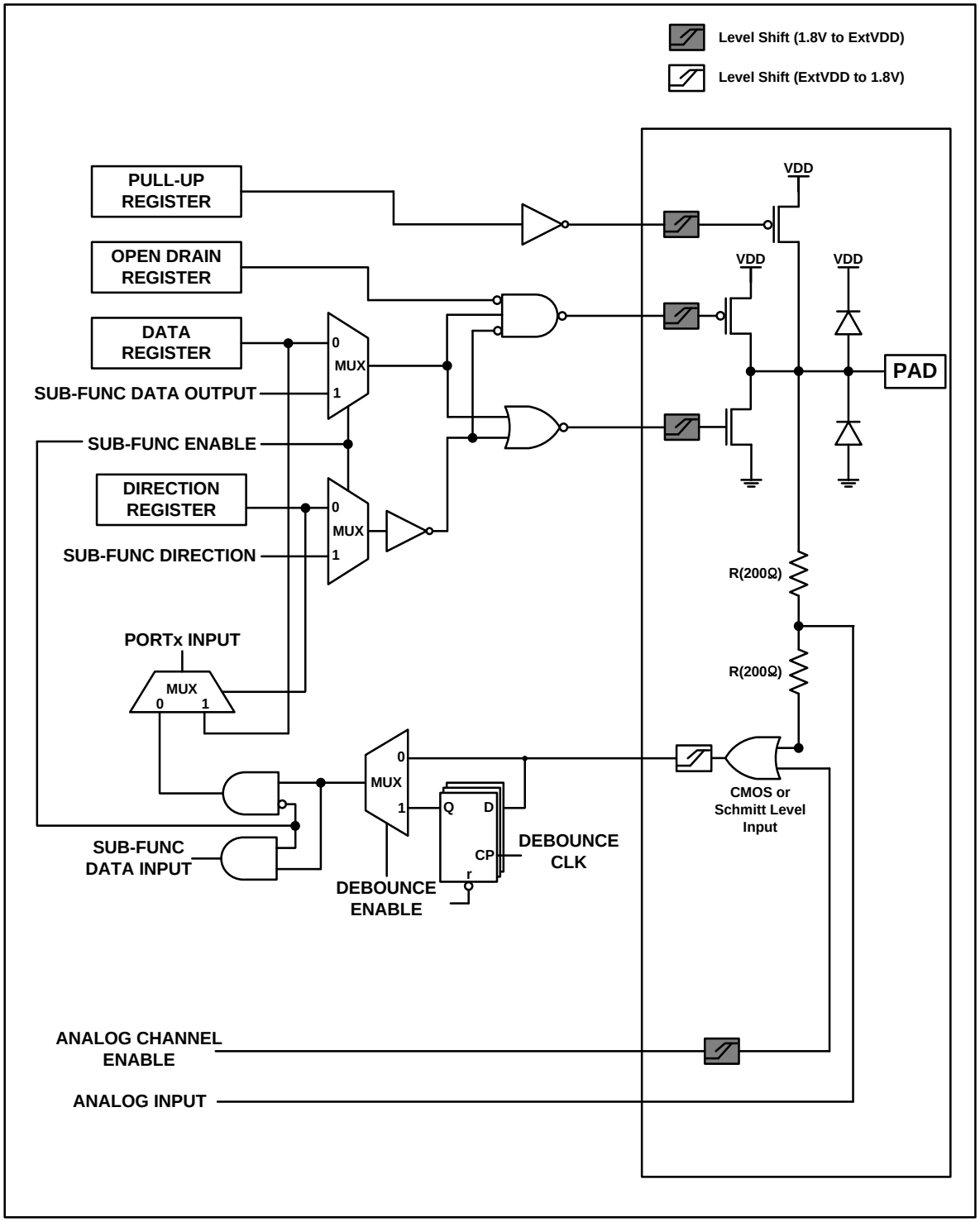


Figure 6-1 General Purpose I/O Port

6.2 General Purpose I/O Port with 1.8V Interface (P10, P11)

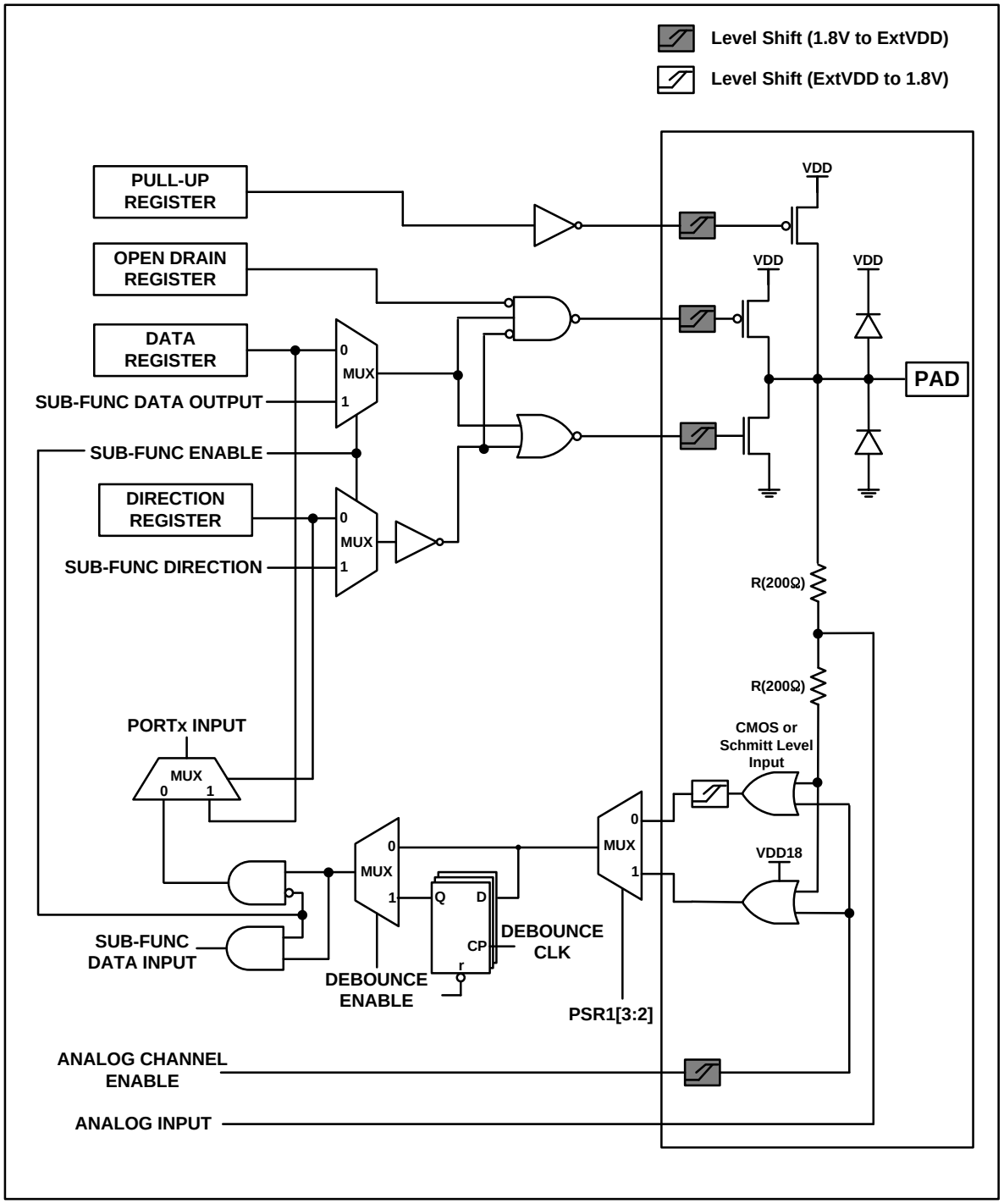


Figure 6-2 General Purpose I/O Port with 1.8V Interface

6.3 Open-Drain I/O Port with 1.8V Interface (P12, P16, P17)

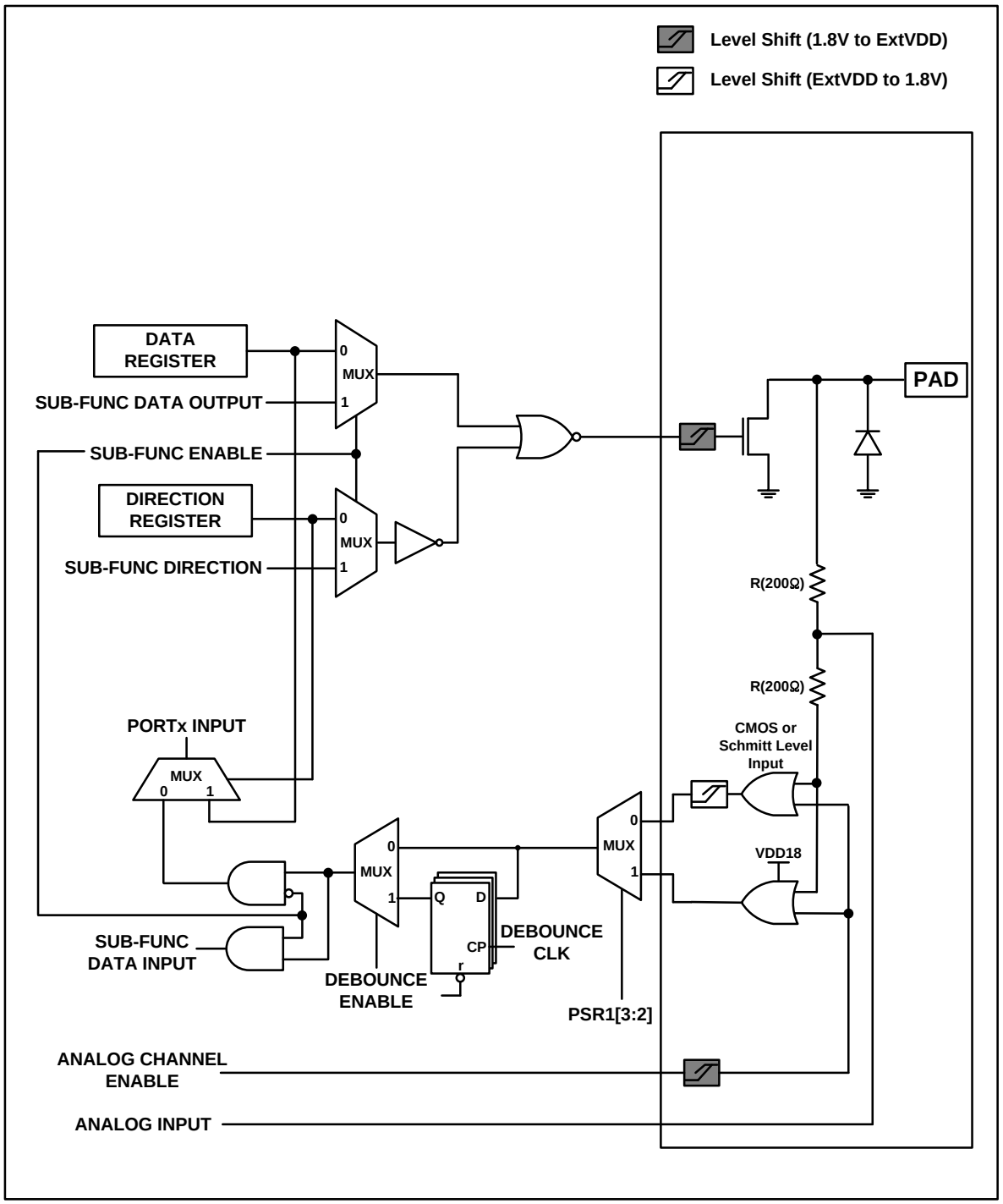


Figure 6-3 Open-Drain I/O Port with 1.8V Interface

7. Electrical Characteristics

7.1 Absolute Maximum Ratings

Table 7-1 Absolute Maximum Ratings

Parameter	Symbol	Rating	Unit
Supply Voltage	VDD	-0.3~+6.5	V
	VSS	-0.3~+0.3	V
Normal Voltage Pin	VI	-0.3~VDD+0.3	V
	VO	-0.3~VDD+0.3	V
	IOH	10	mA
	∑IOH	80	mA
	IOL	20	mA
	∑IOL	160	mA
Total Power Dissipation	PT	600	mW
Storage Temperature	TSTG	-65~+150	℃

Note) Stresses beyond those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device at any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

7.2 Recommended Operating Conditions

Table 7-2 Recommended Operation Conditions

Parameter	Symbol	Condition	MIN	TYP	MAX	Unit
Supply Voltage	VDD	Internal RC-OSC 16MHz	2.0	-	5.5	V
		Sub X-tal 32~38kHz	2.0	-	5.5	V
		Touch & LED Driver	2.7	-	5.5	V
Operating Temperature	TOPR	VDD=2.0~5.5V	-40	-	85	℃
Operating Frequency	FIRC	Internal RC-OSC	-	16	-	MHz
	FWDT	Internal WDT Ring-OSC	-	256		kHz

7.3 Voltage Dropout Converter Characteristics

Table 7-3 Voltage Dropout Converter Characteristics

Parameter	Symbol	Condition	MIN	TYP	MAX	Unit
Operating Voltage		-	2.0	-	5.5	V
Operating Temperature		-	-40	-	+85	℃
Regulation Voltage		-	-	1.9	-	V
Drop-out Voltage		-	-	0.02	-	V
Current Drivability		RUN/IDLE	-	20	-	mA
		SUB-ACTIVE	-	1	-	mA
		STOP1	-	50	-	uA
		STOP2	-	10	-	uA
Operating Current	IDD1	RUN/IDLE	-	0.5	-	mA
	IDD2	SUB-ACTIVE	-	0.1	-	mA
	SIDD1	STOP1	-	5	-	uA
	SIDD2	STOP2	-	0.1	-	uA
Drivability Transition Time	TRAN1	SUB to RUN	-	1	-	us
	TRAN2	STOP to RUN	-	200	-	us

Note) STOP1: WDTRC OSC run, STOP2: WDTRC OSC stop.

7.4 Power-On Reset Characteristics

Table 7-4 Power-On Reset Characteristics

Parameter	Symbol	Condition	MIN	TYP	MAX	Unit
Operating Voltage		-	VSS	-	5.5	V
Operating Temperature		-	-40	-	+85	℃
RESET Release Level	POR	-	1.3	1.4	1.5	V
Operating Current	IDD	-	-	1	-	uA
VDD Rise Rate	VRR		0.05		50	V/ms

7.5 Brown Out Detector & Low Voltage Indicator Characteristics

Table 7-5 Brown Out Detector & Low Voltage Indicator Characteristics

Parameter	Symbol	Condition	MIN	TYP	MAX	Unit
Operating Voltage		-	VSS	-	5.5	V
Operating Temperature		-	-40	-	+85	℃
Detection Level	4.2V	LVI	3.8	-	4.6	V
	3.6V	LVI	3.2	-	3.8	V
	2.5V	LVI	2.25	-	2.75	V
	1.6V	BOD	1.5	-	1.8	V
Hysteresis		-	-	40	-	mV
Operating Current	IDD	-	-	30	-	uA

7.6 Internal RC Oscillator Characteristics

Table 7-6 Internal RC Oscillator Characteristics

Parameter	Symbol	Condition	MIN	TYP	MAX	Unit
Operating Voltage		-	2.0	-	5.5	V
Operating Temperature		-	-40	25	+85	℃
Frequency		-	15.52	16	16.48	MHz
Stabilization Time		-	-	1	-	ms
Operating Current	IDD	-	-	400	-	uA

7.7 Ring-Oscillator Characteristics

Table 7-7 Ring-Oscillator Characteristics

Parameter	Symbol	Condition	MIN	TYP	MAX	Unit
Operating Voltage		-	2.0	-	5.5	V
Operating Temperature		-	-40	-	+85	℃
Frequency		-	230.4	256	281.6	kHz
Stabilization Time		-	-	1	-	ms
Operating Current	IDD	-	-	2	-	uA

7.8 Sub-Oscillator Characteristics

Table 7-8 Sub-Oscillator Characteristics

Oscillator	Parameter	Condition	MIN	TYP	MAX	Unit
Crystal	Sub Oscillation Frequency	2.0V-5.5V	32	32.768	38	kHz
External Clock	SXIN Input Frequency		32	-	100	kHz

7.9 Touch Switch Characteristics

Table 7-9 Touch Switch Characteristics

Parameter	Symbol	Condition	MIN	TYP	MAX	Unit
Operating Voltage	V_{DD}	-	2.7	-	5.5	V
	V_{DDA}	-	2.7	-	5.5	V
VDC Voltage	V_{CCL}	From MCU	-	1.9	-	V
SNR (Signal-to-Noise Ratio)	SNR		-	20	-	dB
Self-Calibration Time	T_{CAL}	-	-	10	-	ms
Scan Speed	T_{SCAN}		-	10	-	ms
Supply Current	I_{VDD}		-	T.B.D	-	mA
Operation Temperature	T_{OPER}	-	40	-	85	℃

7.10 A/D Converter Characteristics

Table 7-10 A/D Converter Characteristics

(TA=-40℃ ~ +85℃, VDD=2.7V ~ 5.5V, VSS=0V)

Parameter	Symbol	Condition	MIN	TYP	MAX	Unit
A/D converting Resolution	-	-	-	12	-	bits
Integral Linearity Error	ILE	VDD=5.12V, Vss=0V, TA=+25℃	-	-	±6	LSB
Differential Linearity Error	DLE		-	-	±3	
Offset Error of Top	EOT		-	-±1	±6	
Zero Offset Error	EOB		-	±1	±6	
Overall Accuracy	-		-	±3	-	
Conversion Time	tCONV	-	-	60	-	Cycle
Analog Input Voltage	VAIN	-	VSS	-	Vref	V
Analog Input Current	IAIN	VDD=5V	-	-	10	uA
Analog Block Current	IAVDD	VDD=5V VDD=3V	-	1 0.5	3 1.5	mA
		VDD=5V (Power down mode)	-	1	5	uA

7.11 DC Characteristics

Table 7-11 DC Characteristics

(VDD =2.0~5.5V, VSS =0V, f_{XIN}=10.0MHz, TA=-40~+85℃)

Parameter	Symbol	Condition	MIN	TYP	MAX	Unit
Input Low Voltage	VIL0	ALL I/O	-0.5	-	0.3*VDD	V
	VIL1	P10, P11, P12, P16, P17 are input 1.8V level	-0.5	-	0.3*1.8V	
Input High Voltage	VIH0	ALL I/O	0.7*VDD	-	VDD	V
	VIH1	P10, P11, P12, P16, P17 are input 1.8V level	0.7*1.8V	-	VDD	
Output Low Voltage	VOL1	ALL I/O (IOL=20mA, VDD=4.5V)	-	-	1	V
Output High Voltage (P10, P11)	VOH1	IOH=-8.57mA, VDD=4.5V	3.5	-	-	V
SEG Output High Voltage	VOH2	IOH=14mA, VDD=5V, SEG Port	VDD-3V	-	-	V
High sink current drive capability	IOL2	COM port(P00~P07), VDD=5V	88	110	-	mA
Input High Leakage Current	IIH	ALL PAD	-	-	1	uA
Input Low Leakage Current	IIL	ALL PAD	-1	-	-	uA
Pull-Up Resister	RPU1	ALL PAD @5V	30	-	70	kΩ
	RPU2	ALL PAD @3V	70	-	150	kΩ
Power Supply Current	IDD1	Run Mode, 16MHz @5V	-	3	5	mA
	IDD2	Sleep Mode, 16MHz @5V	-	2.5	3.5	mA
	IDD3	STOP1 Mode, WDT Active @5V (BOD enable)	-	40	-	uA
	IDD4	STOP1 Mode, WDT Active @5V (BOD disable)	-	10	-	uA
	IDD5	STOP2 Mode, WDT Disable @5V (BOD enable)	-	30	-	uA
	IDD6	STOP2 Mode, WDT Disable @5V (BOD disable)	-	1	-	uA
	IDD7	fsub=32.768kHz @5V		10		uA

Note) STOP1: WDT only running, STOP2: All function disable.

7.12 AC Characteristics

Table 7-12 AC Characteristics

(VDD=5.0V±10%, VSS=0V, TA=-40~+85℃)

Parameter	Symbol	PIN	MIN	TYP	MAX	Unit
Operating Frequency	fMCP	-	2	-	16	MHz
System Clock Cycle Time	tSYS	-	62.5	-	500	ns
Oscillation Stabilization Time (16MHz)	tMST1	-	-	-	10	ms
External Counter Input “H” or “L” Pulse Width	tECW	EC0~ECx	2	-	-	tSYS
Event Counter Transition Time	tREC,tFEC	EC0~ECx	-	-	20	ns
nRESET Input Pulse “L” Width	tRST	nRESET	8	-	-	tSYS

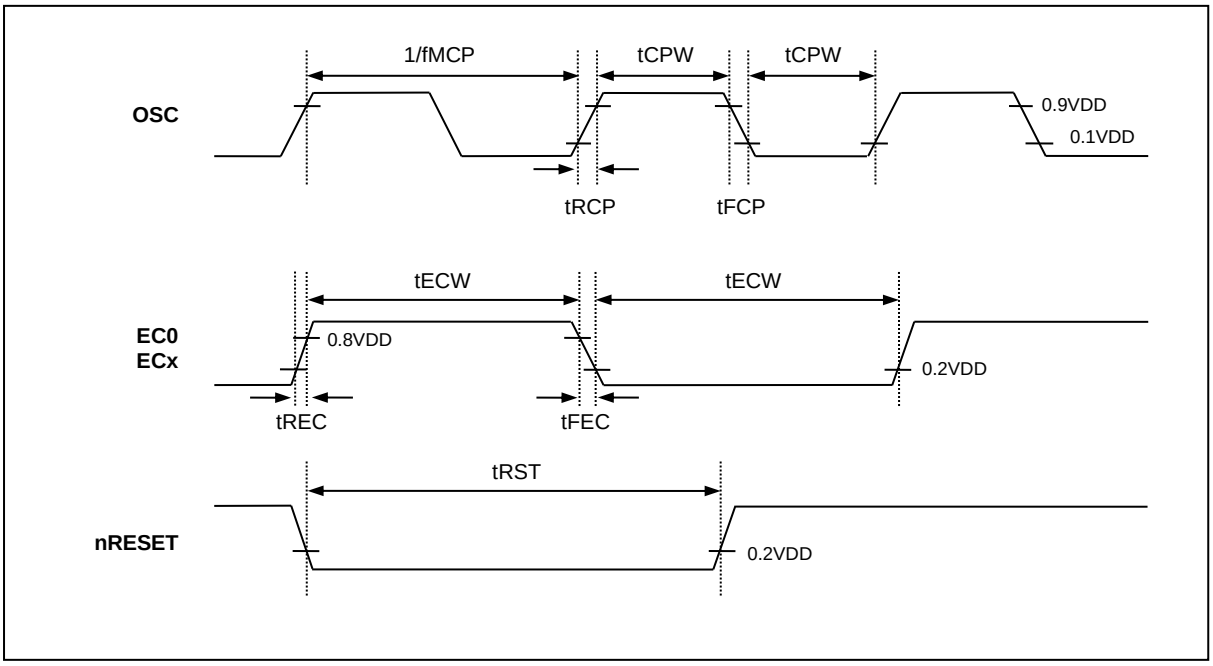


Figure 7-1 AC Timing

7.13 Typical Characteristics

These graphs and tables provided in this section are for design guidance only and are not tested or guaranteed. In some graphs or tables the data presented are outside specified operating range (e.g. outside specified VDD range). This is for information only and devices are guaranteed to operate properly only within the specified range.

The data presented in this section is a statistical summary of data collected on units from different lots over a period of time. “Typical” represents the mean of the distribution while “max” or “min” represents (mean + 3σ) and (mean - 3σ) respectively where σ is standard deviation.

8. Memory

The A96T218 addresses two separate address memory stores: Program memory and Data memory. The logical separation of Program and Data memory allows Data memory to be assessed by 8-bit addresses, which can be more quickly stored and manipulated by 8-bit CPU. Nevertheless, 16-bit Data memory addresses can also be generated through the DPTR register.

Program memory can only be read, not written to. There can be up to 64K bytes of Program memory in a bank. In the A96T218 FLASH version of these devices the 32K bytes of Program memory are provided on-chip. Data memory can be read and written to up to 256 bytes internal memory (DATA) including the stack area.

8.1 Program Memory

A 16-bit program counter is capable of addressing up to 64K bytes for one bank of memory space, but this device has 32K bytes program memory space.

Figure 8-1 shows a map of the lower part of the program memory. After reset, the CPU begins execution from location 0000H. Each interrupt is assigned a fixed location in program memory. The interrupt causes the CPU to jump to that location, where it commences execution of the service routine. External interrupt 0, for example, is assigned to location 0003H. If external interrupt 0 is going to be used, its service routine must begin at location 0003H. If the interrupt is not going to be used, its service location is available as general purpose program memory. If an interrupt service routine is short enough (as is often the case in control applications), it can reside entirely within that 8 byte interval. Longer service routines can use a jump instruction to skip over subsequent interrupt locations, if other interrupts are in use.

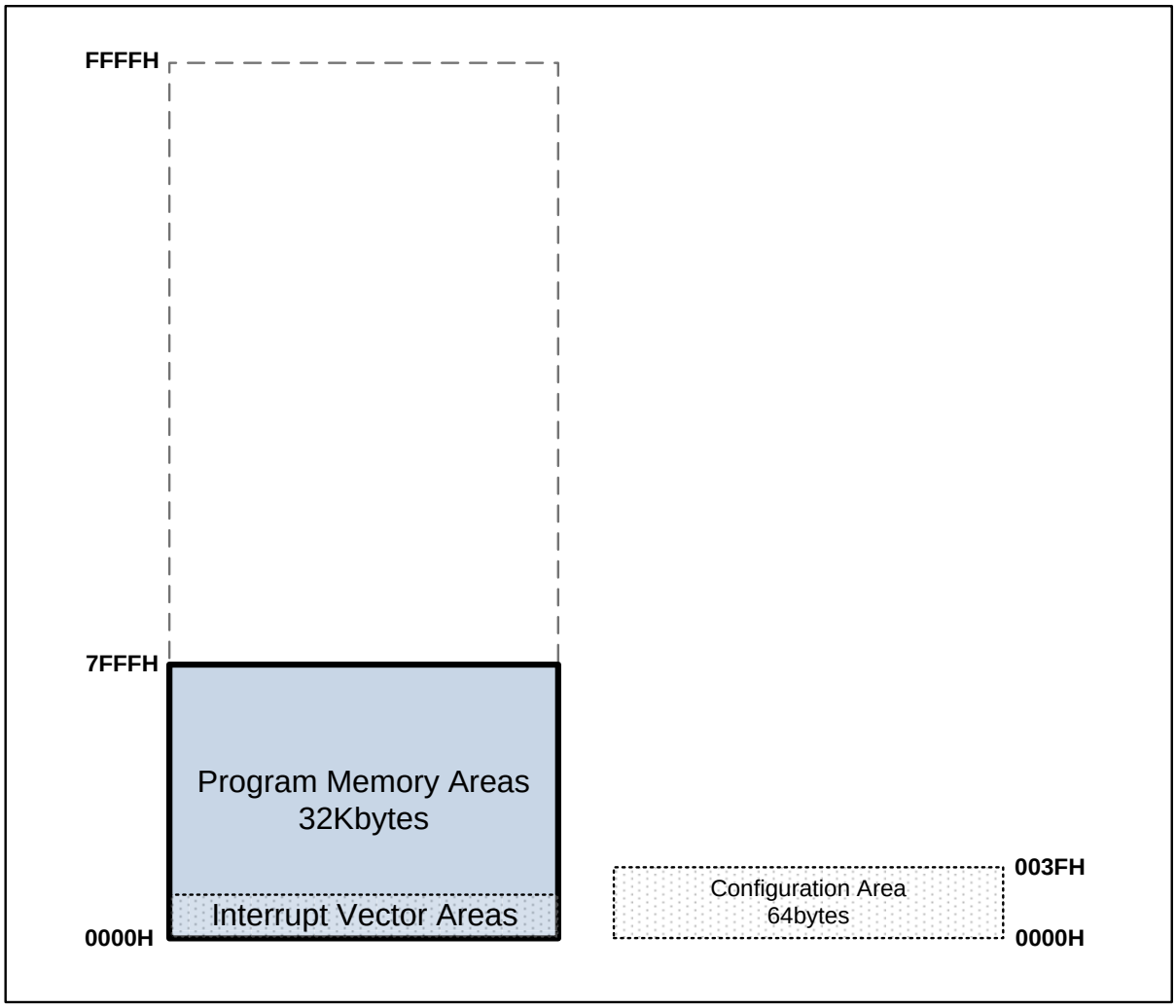


Figure 8-1 Program Memory

- User Function Mode: 32K Bytes Included Interrupt Vector Region
- Non-volatile and reprogramming memory: Flash memory based on EEPROM cell

8.2 Data Memory

Figure 8-2 shows the internal Data memory space available.

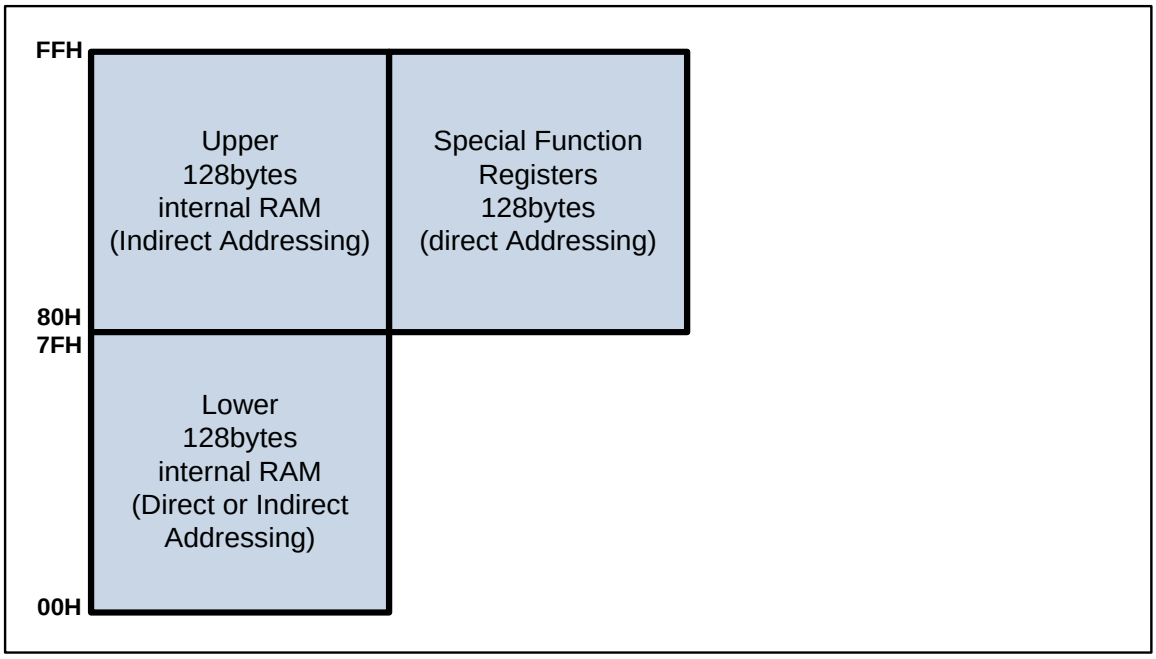


Figure 8-2 Data Memory Map

The internal memory space is divided into three blocks, which are generally referred to as the lower 128, upper 128, and SFR space.

Internal Data memory addresses are always one byte wide, which implies an address space of only 256 bytes. However, the addressing modes for internal RAM can in fact accommodate 384 bytes, using a simple trick. Direct addresses higher than 7FH access one memory space and indirect addresses higher than 7FH access a different memory space. Thus Figure 8-2 shows the upper 128 and SFR space occupying the same block of addresses, 80H through FFH, although they are physically separate entities.

The lower 128 bytes of RAM are present in all 8051 devices as mapped in figure 8-3. The lowest 32 bytes are grouped into 4 banks of 8 registers. Program instructions call out these registers as R0 through R7. Two bits in the Program Status Word select which register bank is in use. This allows more efficient used of code space, since register instructions are shorter than instructions that use direct addressing.

The next 16 bytes above the register banks form a block of bit-addressable memory space. The 8051 instruction set includes a wide selection of single-bit instructions, and the 128 bits in this area can be directly addressed by these instructions. The bit addresses in this area are 00H through 7FH.

All of the bytes in the lower 128 can be accessed by either direct or indirect addressing. The upper 128 bytes RAM can only be accessed by indirect addressing. These spaces are used for user RAM and stack pointer.

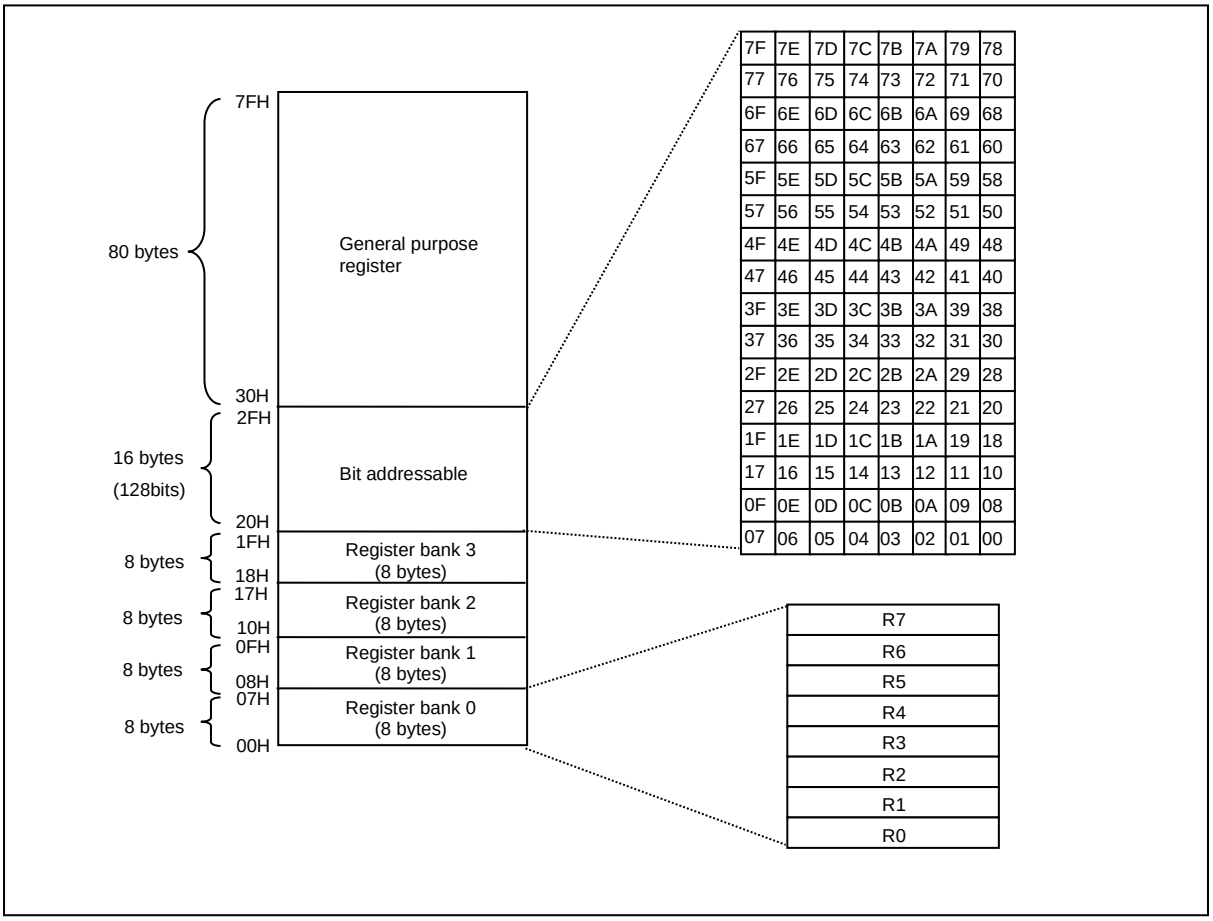


Figure 8-3 Low 128 bytes RAM

8.3 XSFR

A96T218 has 1792Bytes XSRAM. This area has no relation with RAM/FLASH. It can read and write through SFR with 8-bit unit.

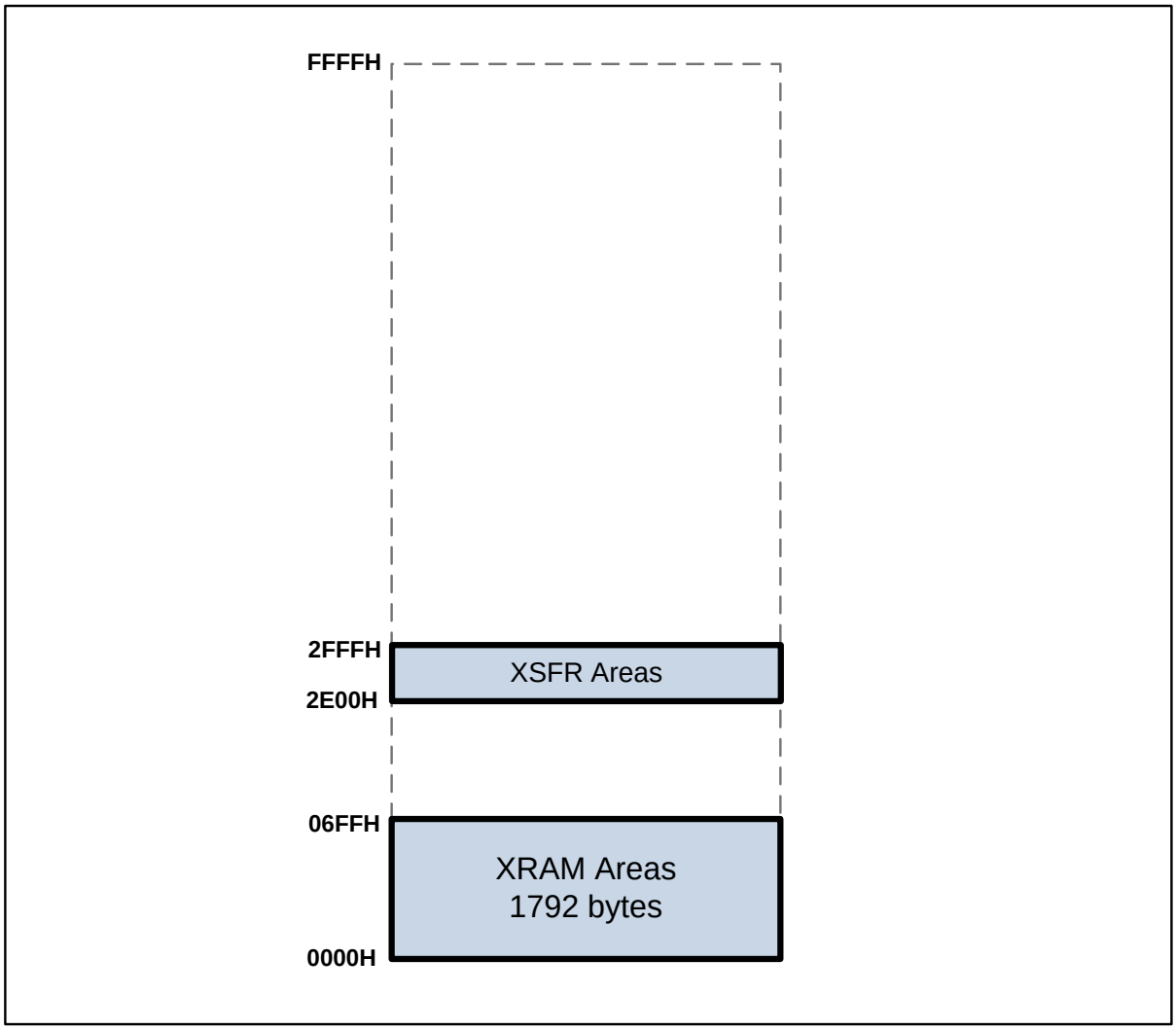


Figure 8-4 XDATA Memory Area

8.4 SFR Map

8.4.1 SFR Map Summary

Table 8-1 SFR Map Summary

	0H/8H(1)	1H/9H	2H/AH	3H/BH	4H/CH	5H/DH	6H/EH	7H/FH
2F58H	FUSE_PKGx	-	-	-	-	-	TEST_REGB	TEST_REGA
2F50H	FUSE_CONF	FUSE_CAL0	FUSE_CAL1	FUSE_RING	FUSE_BGR3	FUSE_FLS	FUSE_TOUC	FUSE_XTAL
2F40H	FCHH	FCHL	FCLH	FCLL	FCMR	-	-	-
2ED0H	CMPCTL	-	-	-	-	-	-	-
2EC8H	FPCAL	VCAL	CCAL	TSCTL0	TSCTL1	P0FT	P2FT	P3FT
2EC0H	TTMR	TTDLL	TTDLH	TTDHL	TTDHH	KENR	KSMR	NFS
2E50H	RSD20	RSD21	RSD22	RSD23	RSD24	RSD25	RSD26	RSD27
2E48H	RSD10	RSD11	RSD12	RSD13	RSD14	RSD15	RSD16	RSD17
2E40H	RSD00	RSD01	RSD02	RSD03	RSD04	RSD05	RSD06	RSD07
2E28H	KS24H	KS24L	KS25H	KS25L	KS26H	KS26L	KS27H	KS27L
2E20H	KS20H	KS20L	KS21H	KS21L	KS22H	KS22L	KS23H	KS23L
2E18H	KS14H	KS14L	KS15H	KS15L	KS16H	KS16L	KS17H	KS17L
2E10H	KS10H	KS10L	KS11H	KS11L	KS12H	KS12L	KS13H	KS13L
2E08H	KS04H	KS04L	KS05H	KS05L	KS06H	KS06L	KS07H	KS07L
2E00H	KS00H	KS00L	KS01H	KS01L	KS02H	KS02L	KS03H	KS03L
F8H	IP1	P4DB	UCTRL11	UCTRL12	UCTRL13	USTAT1	UBAUD1	UDATA1
F0H	B	-	FEARL	FEARM	FEARH	FEDR	FETR	-
E8H	P4PU	P4OD	FEMR	FECR	FESR	FETCR	-	-
E0H	ACC	-	TSCR	TSTR	-	-	-	-
D8H	P4	P4IO	I2CMR	I2CSR	I2CSCLLR	I2CSCLHR	I2CSDHR	I2CDR
D0H	PSW	ADCM	ADCM2	ADCRL	ADCRH	TMISR	I2CSAR1	I2CSAR
C8H	WTMR	WTR /WTCR	UCTRL01	UCTRL02	UCTRL03	USTAT0	UBAUD0	UDATA0
C0H	BUZCR1	BUZDR1	T2CR	T2CR1	PWM2DRL CDR2L / T2L	PWM2DRH CDR2H / T2H	PWM2PRL T2DRL	PWM2PRH T2DRH
B8H	IP	PSR0	T1CR	T1CR1	PWM1DRL CDR1L / T1L	PWM1DRH CDR1H / T1H	PWM1PRL T1DRL	PWM1PRH T1DRH
B0H	BUZCR0	BUZDR0	T0CR	T0CR1	PWM0DRL CDR0L / T0L	PWM0DRH CDR0H / T0H	PWM0PRL T0DRL	PWM0PRH T0DRH
A8H	IE	IE1	IE2	IE3	PSR1	PSR2	PSR3	PSRPWM
A0H	PSR4	P3IO	EO	EIENAB	EIFLAG	EIEDGE	EIPOLA	BothEdgeEN
98H	P3	P2IO	P2PU	P2OD	P2DB	P3PU	P3OD	P3DB
90H	P2	P1IO	P1PU	P1OD	P1DB	P0PU	P0OD	P0DB
88H	P1	P0IO	SCCR	BCCR	BITR	WDTMR	WDTR /WDTCR	RSFR
80H	P0	SP	DPL	DPH	DPL1	DPH1	BODR	PCON

Note: 1) These registers are bit-addressable

8.4.2 Compiler Compatible SFR

ACC (Accumulator) : E0H

7	6	5	4	3	2	1	0
ACC							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

ACC Accumulator

B (B Register) : F0H

7	6	5	4	3	2	1	0
B							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

B B Register

SP (Stack Pointer) : 81H

7	6	5	4	3	2	1	0
SP							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 07H

SP Stack Pointer

DPL (Data Pointer Low Byte) : 82H

7	6	5	4	3	2	1	0
DPL							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

DPL Data Pointer Low Byte

DPH (Data Pointer High Byte) : 83H

7	6	5	4	3	2	1	0
DPH							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

DPH Data Pointer High Byte

DPL1 (Data Pointer Low Byte) : 84H

7	6	5	4	3	2	1	0
DPL							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

DPL Data Pointer Low Byte

DPH1 (Data Pointer High Byte) : 85H

7	6	5	4	3	2	1	0
DPH							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

DPH Data Pointer High Byte

PSW (Program Status Word) : D0H

7	6	5	4	3	2	1	0
CY	AC	F0	RS1	RS0	OV	F1	P
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

CY Carry Flag
AC Auxiliary Carry Flag
F0 General Purpose User-Definable Flag
RS1 Register Bank Select bit 1
RS0 Register Bank Select bit 0
OV Overflow Flag
F1 User-Definable Flag
P Parity Flag. Set/cleared by hardware each instruction cycle to indicate an odd/even number of ‘1’ bits in the accumulator

EO (Extended Operation Register) : A2H

7	6	5	4	3	2	1	0
-	-	-	TRAP_EN	-	-	-	DPSEL.0
R	R	R	R/W	R	R	R	R/W

Initial value : 00H

TRAP_EN Select the instruction
0 Select MOVC @(DPTR++), A
1 Select Software TRAP instruction

DPSEL Select Banked Data Point Register
0 DPTR0
1 DPTR1

9. I/O Ports

9.1 I/O Ports

The A96T218 has 34 I/O ports (P0 ~ P4). Each port can be easily configured by software as I/O pin, internal pull up and open drain pin to meet various system configurations and design requirements.

9.2 Port Register

9.2.1 Data Register (P0~P4)

Data Register is a bidirectional I/O port. If ports are configured as output ports, data can be written to the corresponding bit of the Px. If ports are configured as input ports, the data can be read from the corresponding bit of the Px.

9.2.2 Direction Register (P0IO~P4IO)

Each I/O pin can independently used as an input or an output through the PxIO register. Bits cleared in this read/write register will select the corresponding pin in Px to become an input, setting a bit sets the pin to output. All bits are cleared by a system reset.

9.2.3 Pull-up Resistor Selection Register (P0PU~P4PU)

The on-chip pull-up resistor can be connected to them in 1-bit units with a pull-up resistor selection register (PxPU). The pull-up register selection controls the pull-up resister enable/disable of each port. When the corresponding bit is 1, the pull-up resister of the pin is enabled. When 0, the pull-up resister is disabled. All bits are cleared by a system reset.

9.2.4 Open-drain Selection Register (P0OD~P4OD)

There is internally open-drain selection register (PxOD) in P0 ~ P4. The open-drain selection register controls the open-drain enable/disable of each port. Ports become push-pull by a system reset. You should connect an internal resistor or an external resistor in open-drain output mode.

9.2.5 De-bounce Enable Register (P0DB~P4DB)

P0 ~ P4 support de-bounce function. De-bounce time of each ports has 16us

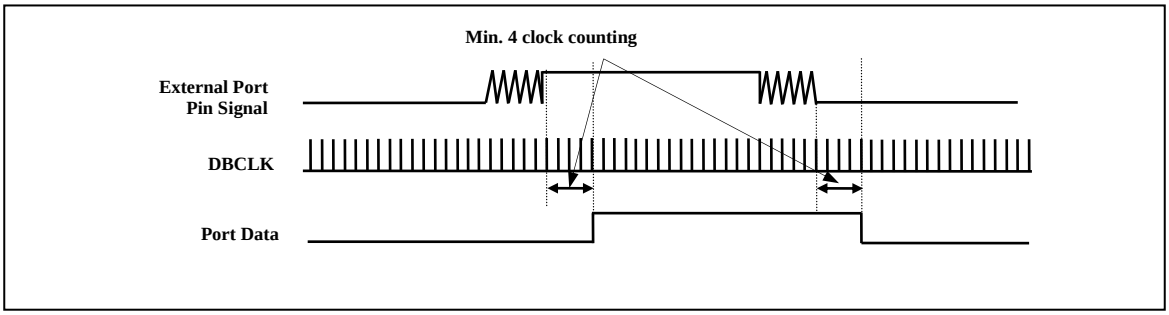


Figure 9-1 Debounce Function

9.2.6 Port Selection Register PSR0,1,2,3,4, PSRPWM

9.2.6.1 Port Selection Register (PSR0,2,3,4)

PSR0,2,3,4 registers prevent the input leakage current when ports are connected to analog inputs. If the bit of PSRx is ‘1’, the dynamic current path of the schmitt OR gate of the port is cut off and the digital input of the corresponding port is always ‘1’.

9.2.6.2 Special function Port Selection Register (PSR1)

The bit of PSR1[0] is ‘0’, I^2C SCL/SDA use P27/P26
The bit of PSR1[0] is ‘1’, I^2C SCL/SDA use P14/P13

The bit of PSR1[1] is ‘0’, USART1 RXD/TXD use P10/P11
The bit of PSR1[1] is ‘1’, USART1 RXD/TXD use P13/P14

The bit of PSR1[2] is ‘0’, USART0 RXD/TXD use P32/P33
The bit of PSR1[2] is ‘1’, USART0 RXD/TXD use P22/P23

The bit of PSR1[3] is ‘0’, Reserved (Refer to PSR1[0])
The bit of PSR1[3] is ‘1’, I^2C SCL/SDA use P17/P16

The bit of PSR1[4] is ‘0’, VDD/VSS input level use P12
The bit of PSR1[4] is ‘1’, 1.8V/VSS input level use P12

The bit of PSR1[5] is ‘0’, VDD/VSS input level use P17/P16
The bit of PSR1[5] is ‘1’, 1.8V/VSS input level use P17/P16

The bit of PSR1[6] is ‘0’, VDD/VSS input level use P11/P10
The bit of PSR1[6] is ‘1’, 1.8V/VSS input level use P11/P10

9.2.6.3 Special function Port Selection Register (PSRPWM)

The bit of PSRPWM[2:0] is 0x00 , PWM0 Out P06 (Default)
The bit of PSRPWM[2:0] is 0x01 , PWM0 Out P21
The bit of PSRPWM[2:0] is 0x02 , PWM0 Out P22
The bit of PSRPWM[2:0] is 0x03 , PWM0 Out P23
The bit of PSRPWM[2:0] is 0x04 , PWM0 Out P24
The bit of PSRPWM[2:0] is 0x02 , PWM0 Out P25
The bit of PSRPWM[2:0] is 0x03 , PWM0 Out P26
The bit of PSRPWM[2:0] is 0x04 , PWM0 Out P27

The bit of PSRPWM[5:3] is 0x00 , PWM1 Out P15 (Default)
The bit of PSRPWM[5:3] is 0x01 , PWM1 Out P21

The bit of PSRPWM[5:3] is 0x02 , PWM1 Out P22
The bit of PSRPWM[5:3] is 0x03 , PWM1 Out P23
The bit of PSRPWM[5:3] is 0x04 , PWM1 Out P24
The bit of PSRPWM[5:3] is 0x02 , PWM1 Out P25
The bit of PSRPWM[5:3] is 0x03 , PWM1 Out P26
The bit of PSRPWM[5:3] is 0x04 , PWM1 Out P27

9.2.7 Register Map

Table 9-1 Register Map

Name	Address	Dir	Default	Description
P0	80H	R/W	00H	P0 Data Register
P0IO	89H	R/W	00H	P0 Direction Register
P0PU	95H	R/W	00H	P0 Pull-up Resistor Selection Register
P0OD	96H	R/W	00H	P0 Open-drain Selection Register
P0DB	97H	R/W	00H	P0 Debounce Enable Register
P1	88H	R/W	00H	P1 Data Register
P1IO	91H	R/W	00H	P1 Direction Register
P1PU	92H	R/W	00H	P1 Pull-up Resistor Selection Register
P1OD	93H	R/W	00H	P1 Open-drain Selection Register
P1DB	94H	R/W	00H	P1 Debounce Enable Register
P2	90H	R/W	00H	P2 Data Register
P2IO	99H	R/W	00H	P2 Direction Register
P2PU	9AH	R/W	00H	P2 Pull-up Resistor Selection Register
P2OD	9BH	R/W	00H	P2 Open-drain Selection Register
P2DB	9CH	R/W	00H	P2 Debounce Enable Register
P3	98H	R/W	00H	P3 Data Register
P3IO	A1H	R/W	00H	P3 Direction Register
P3PU	9DH	R/W	00H	P3 Pull-up Resistor Selection Register
P3OD	9EH	R/W	00H	P3 Open-drain Selection Register
P3DB	9FH	R/W	00H	P3 Debounce Enable Register
P4	D8H	R/W	00H	P4 Data Register
P4IO	D9H	R/W	00H	P4 Direction Register
P4PU	E8H	R/W	00H	P4 Pull-up Resistor Selection Register
P4OD	E9H	R/W	00H	P4 Open-drain Selection Register
P4DB	F9H	R/W	00H	P4 Debounce Enable Register
PSR0	ABH	R/W	00H	P0 Port Select Register
PSR1	ACH	R/W	80H	I^2C, USART Port Selection Register
PSR2	ADH	R/W	00H	P2 Port Select Register
PSR3	AEH	R/W	00H	P3 Port Select Register
PSR4	A0H	R/W	00H	P4 Port Select Register
PSRPWM	AFH	R/W	00H	PWM Port Select Register

9.3 P0, P1, P2, P3, P4 Port

9.3.1 Px Port Description

Px is 8-bit I/O port. Px control registers consist of Data register (Px), direction register (PxIO), debounce enable register (PxDB), pull-up register selection register (PxPU), open-drain selection register (PxOD).

9.3.2 Register Description for Px

P0, P1, P2, P3, P4 (Px Data Register) : 80H, 88H, 90H, 98H, D8H

7	6	5	4	3	2	1	0
Px7	Px6	Px5	Px4	Px3	Px2	Px1	Px0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

Px[7:0] I/O Data

P0IO, P1IO, P2IO, P3IO, P4IO (Px Direction Register) : 89H, 91H, 99H, A1H, D9H

7	6	5	4	3	2	1	0
Px7IO	Px6IO	Px5IO	Px4IO	Px3IO	Px2IO	Px1IO	Px0IO
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

PxIO[7:0] Px data I/O direction.

0 Input

1 Output

P0PU, P1PU, P2PU, P3PU, P4PU (Px Pull-up Resistor Selection Register) : 95H, 92H, 9AH, 9DH, E8H

7	6	5	4	3	2	1	0
Px7PU	Px6PU	Px5PU	Px4PU	Px3PU	Px2PU	Px1PU	Px0PU
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

PxPU[7:0] Configure pull-up resistor of Px port

0 Disable

1 Enable

P0OD, P1OD, P2OD, P3OD, P4OD (Px Open-drain Selection Register) : 96H, 93H, 9BH, 9EH, E9H

7	6	5	4	3	2	1	0
Px7OD	Px6OD	Px5OD	Px4OD	Px3OD	Px2OD	Px1OD	Px0OD
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

PxOD[7:0] Configure open-drain of Px port

0 Disable

1 Enable

P0DB, P1DB, P2DB, P3DB, P4DB (Px Debounce Enable Register) : 97H, 94H, 9CH, 9FH, F9H

7	6	5	4	3	2	1	0
Px7DB	Px6DB	Px5DB	Px4DB	Px3DB	Px2DB	Px1DB	Px0DB
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

PxDB[7:0] Configure debounce of Px port

0 Disable

1 Enable

PSR0,2,3,4 registers prevent the input leakage current when ports are connected to analog inputs (ADC input) or outputs (TOUCH scan wave output). If the bit of PSR0,2,3,4 is '1', the dynamic current path of the schmitt OR gate of the port is cut off and the digital input of the corresponding port is always '1'.

PSR0 (P0 Ports leakage prevent Register) : ABH

7	6	5	4	3	2	1	0
PSR07	PSR06	PSR05	PSR04	PSR03	PSR02	PSR01	PSR00
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

PSR0[7:0] P07~P00 ports leakage prevent register

0 Disable leakage prevent function (default)

1 Enable leakage prevent function for TOUCH

PSR2 (P2 Ports leakage prevent Register) : ADH

7	6	5	4	3	2	1	0
PSR27	PSR26	PSR25	PSR24	PSR23	PSR22	PSR21	PSR20
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

PSR2[7:0] P27~P20 port leakage prevent register

0 Disable leakage prevent function (default)

1 Enable leakage prevent function for TOUCH or ADC[5:0]

PSR3 (P3 Ports leakage prevent Register) : AEH

7	6	5	4	3	2	1	0
PSR37	PSR36	PSR35	PSR34	PSR33	PSR32	PSR31	PSR30
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

PSR3[7:0] P37~P30 port leakage prevent register

0 Disable leakage prevent function (default)

1 Enable leakage prevent function for TOUCH

PSR4 (P4 Ports leakage prevent Register) : A0H

7	6	5	4	3	2	1	0
-	-	-	-	PSR43	PSR42	PSR41	PSR40
-	-	-	-	R/W	R/W	R/W	R/W

Initial value : 00H

PSR4[3:2] P41~P40 port leakage prevent register
- 0 Disable leakage prevent function (default)
- 1 Enable leakage prevent function

PSR4[1:0] P14~P13 port leakage prevent register
- 0 Disable leakage prevent function (default)
- 1 Enable leakage prevent function for ADC[7:6]

PSR1 (I^2C, USART Ports and VIH level Selection Register) : ACH

7	6	5	4	3	2	1	0
PSR17	PSR16	PSR15	PSR14	PSR13	PSR12	PSR11	PSR10
-	-	-	-	R/W	R/W	R/W	R/W

Initial value : 80H

PSR1[7] I2C state machine clear register at stop mode (TEST Only)
- 0 Un-clear at enter stop mode
- 1 Clear at enter stop mode (default)

PSR1[6] VIH Level Selector for P11/P10 Input
- 0 VDD Level (default)
- 1 1.8V Level

PSR1[5] VIH Level Selector for P17/P16 Input
- 0 VDD Level (default)
- 1 1.8V Level

PSR1[4] VIH Level Selector for P12 Input
- 0 VDD Level (default)
- 1 1.8V Level

PSR1[3] I^2C ports selection register
- 0 Reserved (default)
- 1 P16, P17 for I^2C (In this case P2[7:6] is not support GPIO and PWM function)

PSR1[2] USART ports selection register
- 0 P32, P33 for USART0 (default)
- 1 P22, P23 for USART0

PSR1[1] USART ports selection register
- 0 P10, P11 for USART1 (default)
- 1 P13, P14 for USART1

PSR1[0] I^2C ports selection register
- 0 P26, P27 for I^2C (default)
- 1 P13, P14 for I^2C

PSRPWM (PWM Ports Selection Register) : AFH

7	6	5	4	3	2	1	0
-	-	PSRpwm5	PSRpwm4	PSRpwm3	PSRpwm2	PSRpwm1	PSRpwm0
-	-	R/W	R/W	R/W	R/W	R/W	R/W

Initial value: 00H

PSRpwm[5:3] PWM1 output port select register

0	0	0	Select P1[5] (default)
0	0	1	Select P2[1]
0	1	0	Select P2[2]

	0	1	1	Select P2[3]
	1	0	0	Select P2[4]
	1	0	1	Select P2[5]
	1	1	0	Select P2[6]
	1	1	1	Select P2[7]

PSRpwm[2:0] PWM0 output port select register

0	0	0	Select P0[6] (Default)
0	0	1	Select P2[1]
0	1	0	Select P2[2]
0	1	1	Select P2[3]
1	0	0	Select P2[4]
1	0	1	Select P2[5]
1	1	0	Select P2[6]
1	1	1	Select P2[7]

Note : PWM2 output port is only P3[3]

10. Interrupt Controller

10.1 Overview

The A96T218 supports up to 24 interrupt sources. The interrupts have separate enable register bits associated with them, allowing software control. They can also have four levels of priority assigned to them. The interrupt controller has following features:

- receive the request from 24 interrupt source
- 6 group priority
- 4 priority levels
- Multi Interrupt possibility
- If the requests of different priority levels are received simultaneously, the request of higher priority level is serviced
- Each interrupt source can control by EA bit and each IEx bit
- Interrupt latency: 5~8 machine cycles in single interrupt system

The maskable interrupts are enabled through five pair of interrupt enable registers (IE, IE1, IE2, IE3). Bits of IE, IE1, IE2, IE3 register each individually enable/disable a particular interrupt source. Overall control is provided by bit 7 of IE (EA). When EA is set to '0', all interrupts are disabled: when EA is set to '1', interrupts are individually enabled or disabled through the other bits of the interrupt enable registers. The A96T218 supports a 4-level priority scheme. Each maskable interrupt is individually assigned to one of four priority levels by writing to IP or IP1.

Priority sets two bit which is to IP and IP1 register about group. If two requests of different priority levels are received simultaneously, the request of higher priority level is serviced. If the request of same or lower priority level is received, that request is not serviced.

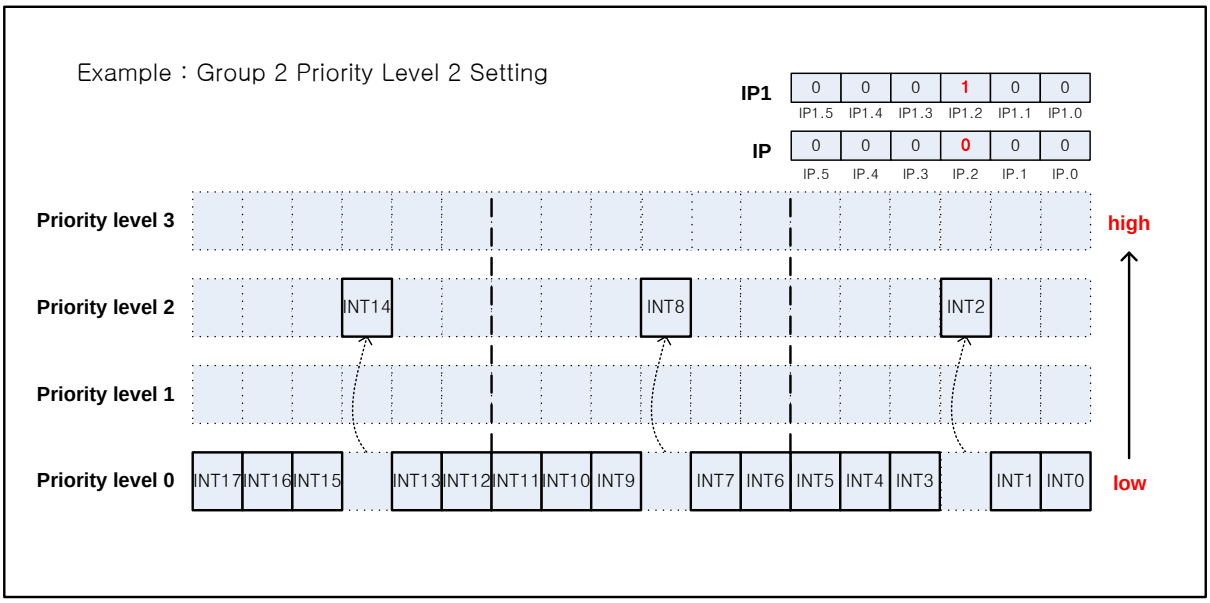


Figure 10-1 Interrupt Group Priority Level

10.2 Block Diagram

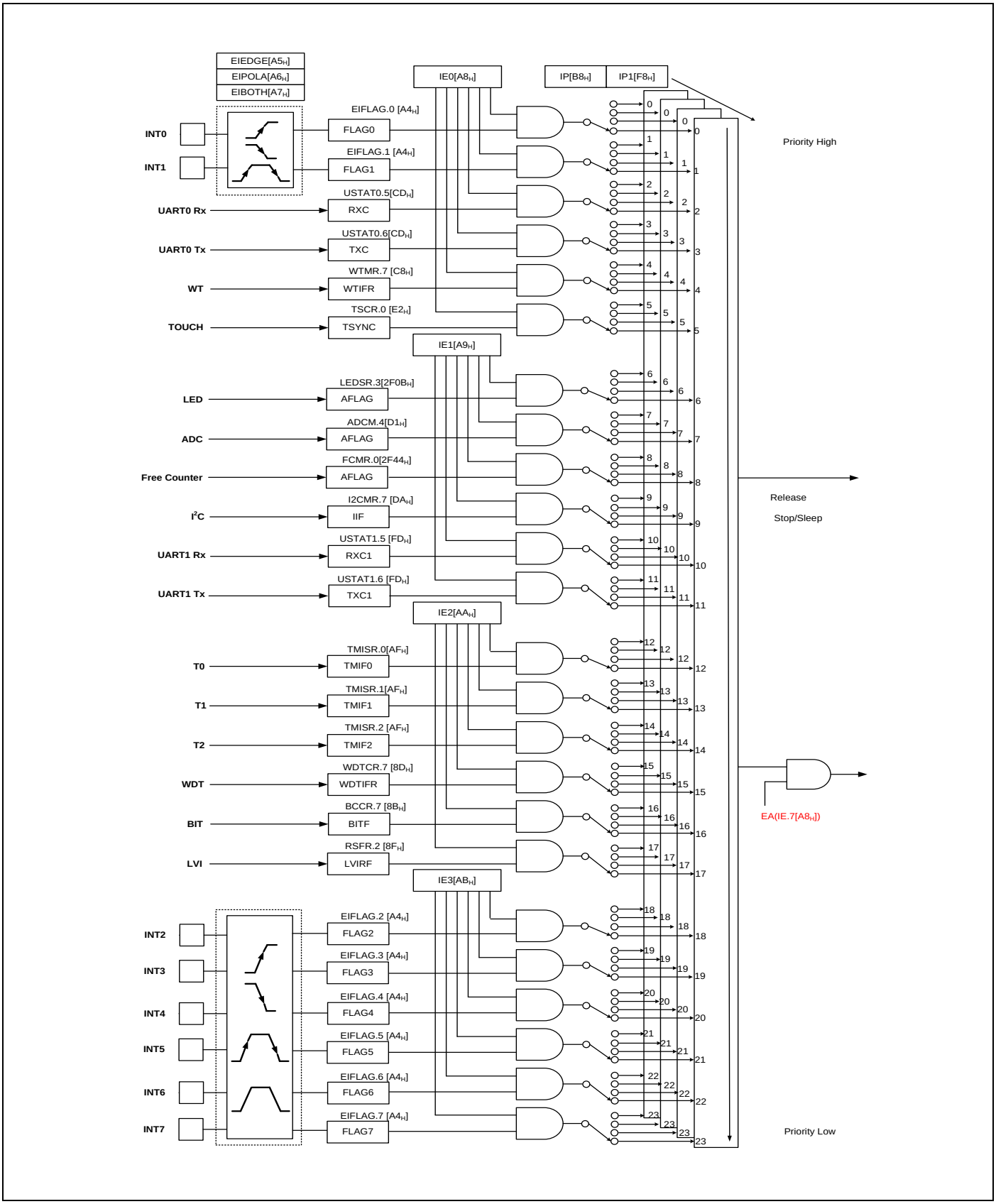


Figure 10-2 Block Diagram of Interrupt

10.3 Interrupt Vector Table

The interrupt controller supports 24 interrupt sources as shown in the Table 10-1 below. When interrupt becomes service, long call instruction (LCALL) is executed in the vector address. Interrupt request 24 has a decided priority order.

Table 10-1 Interrupt Vector Address Table

Interrupt Source	Symbol	Interrupt Enable Bit	Mask	Vector Address
External Int0	INT0	IE0.0	Maskable	0003H
External Int1	INT1	IE0.1	Maskable	000BH
USART R0	INT2	IE0.2	Maskable	0013H
USART T0	INT3	IE0.3	Maskable	001BH
WT	INT4	IE0.4	Maskable	0023H
TSYNC	INT5	IE0.5	Maskable	002BH
LED	INT6	IE1.0	Maskable	0033H
ADC	INT7	IE1.1	Maskable	003BH
Free Count	INT8	IE1.2	Maskable	0043H
I2C	INT9	IE1.3	Maskable	004BH
USART R1	INT10	IE1.4	Maskable	0053H
USART T1	INT11	IE1.5	Maskable	005BH
T0	INT12	IE2.0	Maskable	0063H
T1	INT13	IE2.1	Maskable	006BH
T2	INT14	IE2.2	Maskable	0073H
WDT	INT15	IE2.3	Maskable	007BH
BIT	INT16	IE2.4	Maskable	0083H
LVI	INT17	IE2.5	Maskable	008BH
External Int2	INT18	IE3.0	Maskable	0093H
External Int3	INT19	IE3.1	Maskable	009BH
External Int4	INT20	IE3.2	Maskable	00A3H
External Int5	INT21	IE3.3	Maskable	00ABH
External Int6	INT22	IE3.4	Maskable	00B3H
External Int7	INT23	IE3.5	Maskable	00BBH

For mask-able interrupt execution, first EA bit must set ‘1’ and specific interrupt source must set ‘1’ by writing a ‘1’ to associated bit in the IEx. If interrupt request is received, specific interrupt request flag set ‘1’. And it remains ‘1’ until CPU accepts interrupt. After that, interrupt request flag will be cleared automatically.

10.4 Interrupt Sequence

An interrupt request is held until the interrupt is accepted or the interrupt latch is cleared to ‘0’ by a reset or an instruction. Interrupt acceptance always generates at last cycle of the instruction. So instead of fetching the current instruction, CPU executes internally LCALL instruction and saves the PC stack. For the interrupt service routine, the interrupt controller gives the address of LJMP instruction to CPU. After finishing the current instruction, at the next instruction to go interrupt service routine needs 5~8 machine cycle and the interrupt service task is terminated upon execution of an interrupt return instruction [RETI]. After generating interrupt, to go to interrupt service routine, the following process is progressed

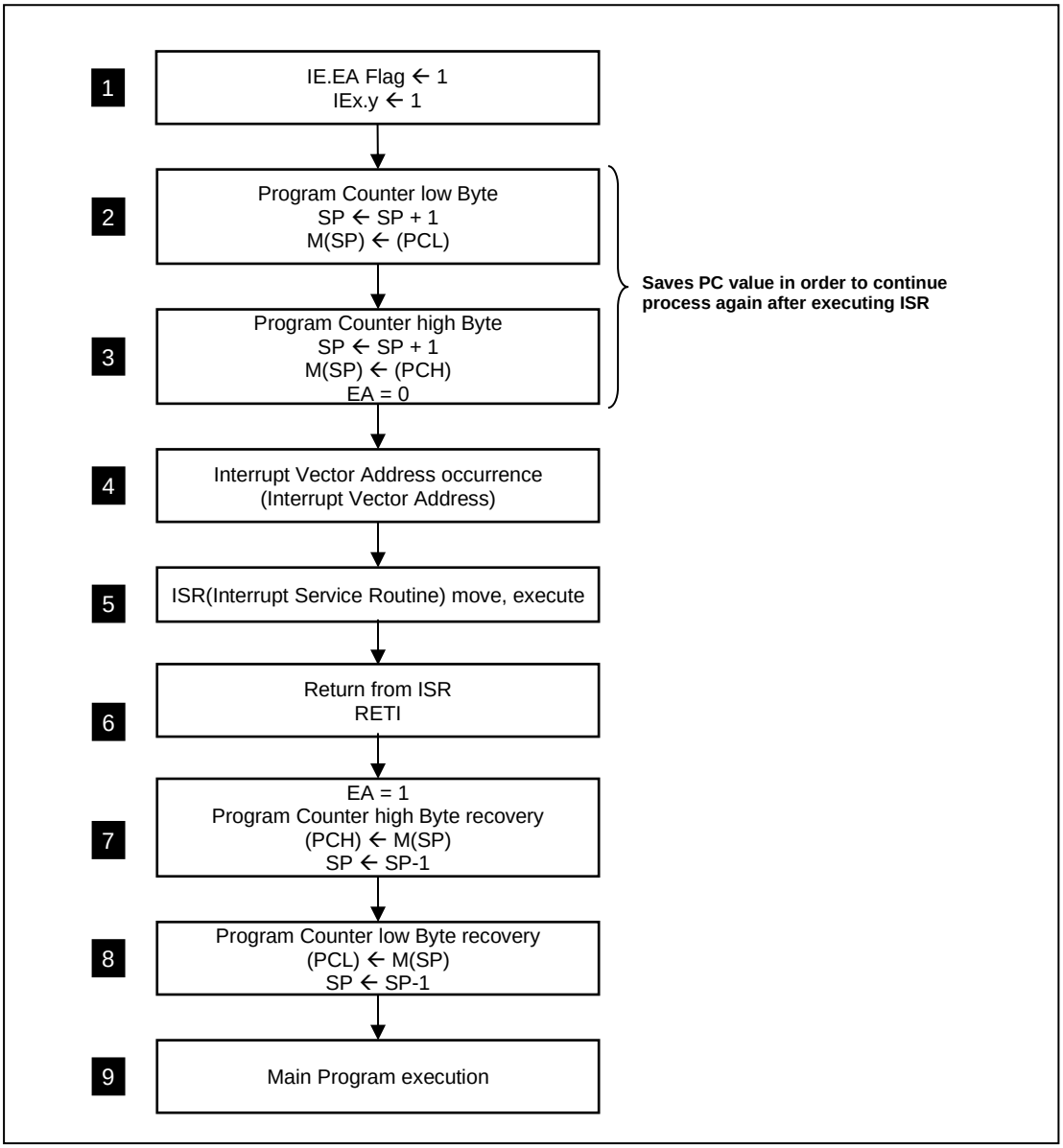


Figure 10-3 Interrupt Sequence Flow

10.5 Effective Timing after Controlling Interrupt bit

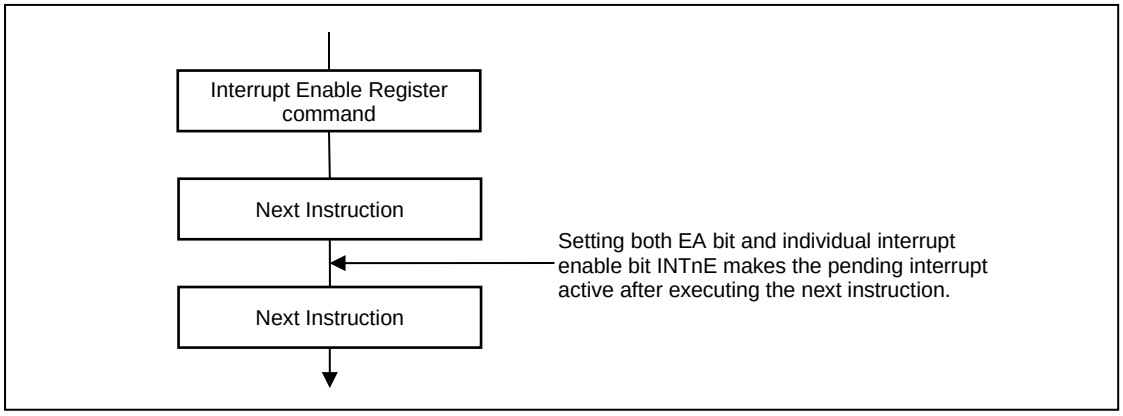


Figure 10-4 Interrupt Enable Register Effective Timing

10.6 Multi Interrupt

If two requests of different priority levels are received simultaneously, the request of higher priority level is serviced. If requests of the interrupt are received at the same time simultaneously, an interrupt polling sequence determines by hardware which request is serviced. However, multiple processing through software for special features is possible.

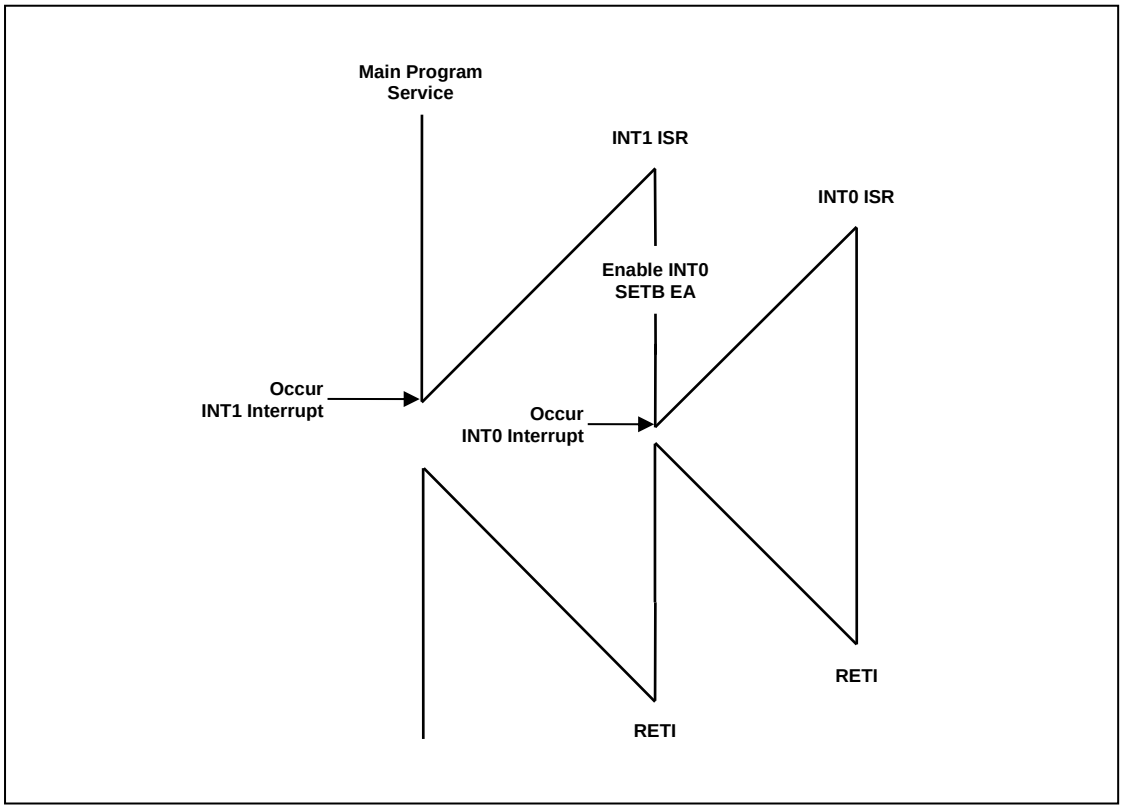


Figure 10-5 Execution of Multi Interrupt

Following example is shown to service INT0 routine during INT1 routine in Figure 10-5. In this example, INT0 interrupt priority is higher than INT1 interrupt priority. If some interrupt is lower than INT1 priority, it can't service its interrupt routine.

Example) Software Multi Interrupt:

```
INT1:   MOV     IE, #81H      ; Enable INT0 only
        MOV     IE1, #00H     ; Disable others
        SETB    EA            ; Enable global interrupt (necessary for multi interrupt)
        :
```

10.7 Interrupt Enable Accept Timing

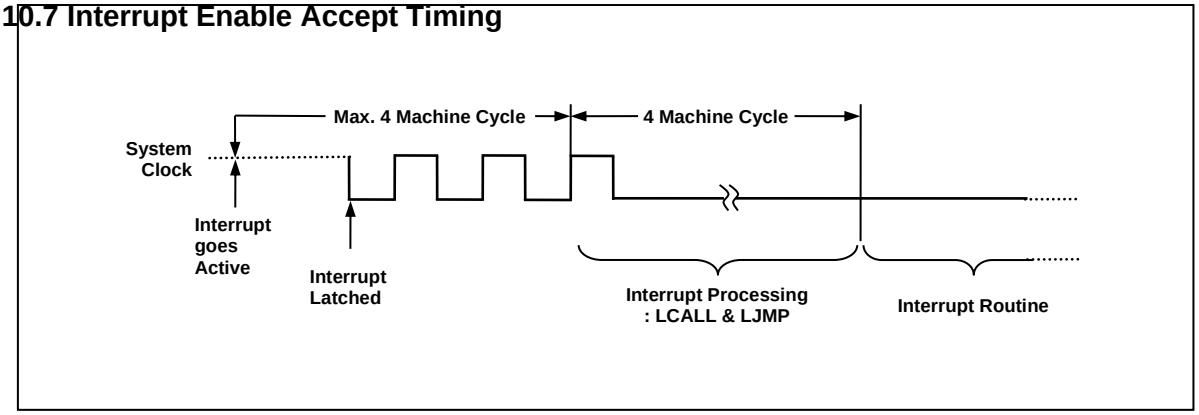


Figure 10-6 Interrupt Response Timing Diagram

10.8 Interrupt Service Routine Address

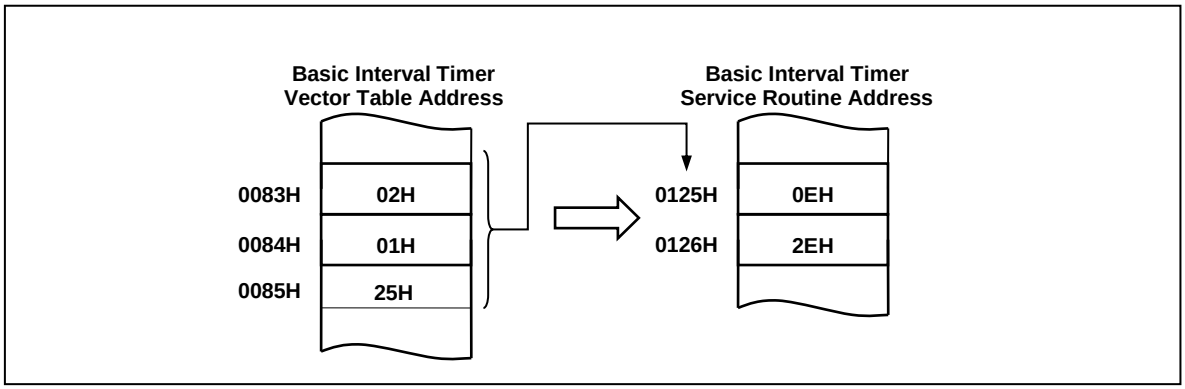


Figure 10-7 Correspondence between Vector Table Address and the Entry Address of ISR

10.9 Saving/Restore General-Purpose Registers

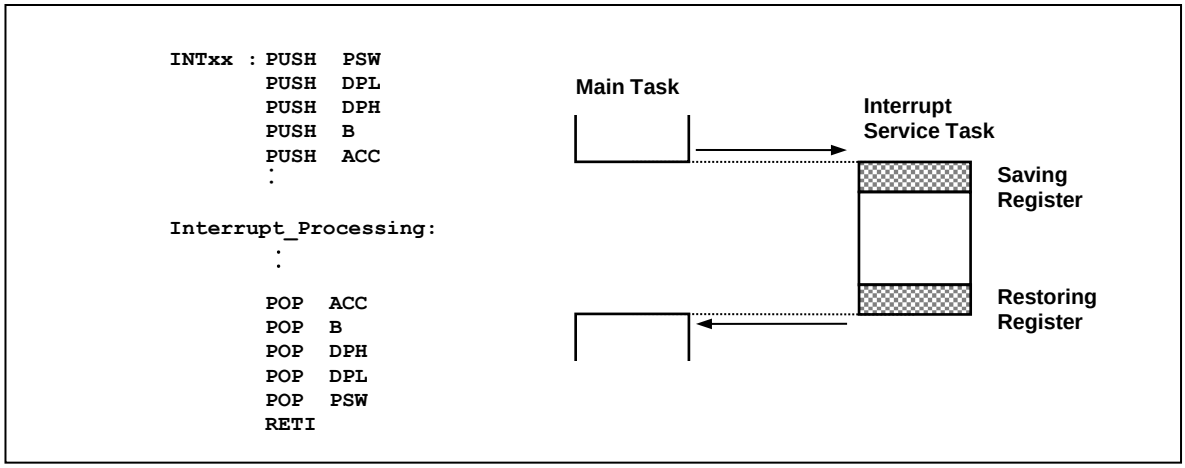


Figure 10-8 Saving/Restore Process Diagram & Sample Source

10.10 Interrupt Timing

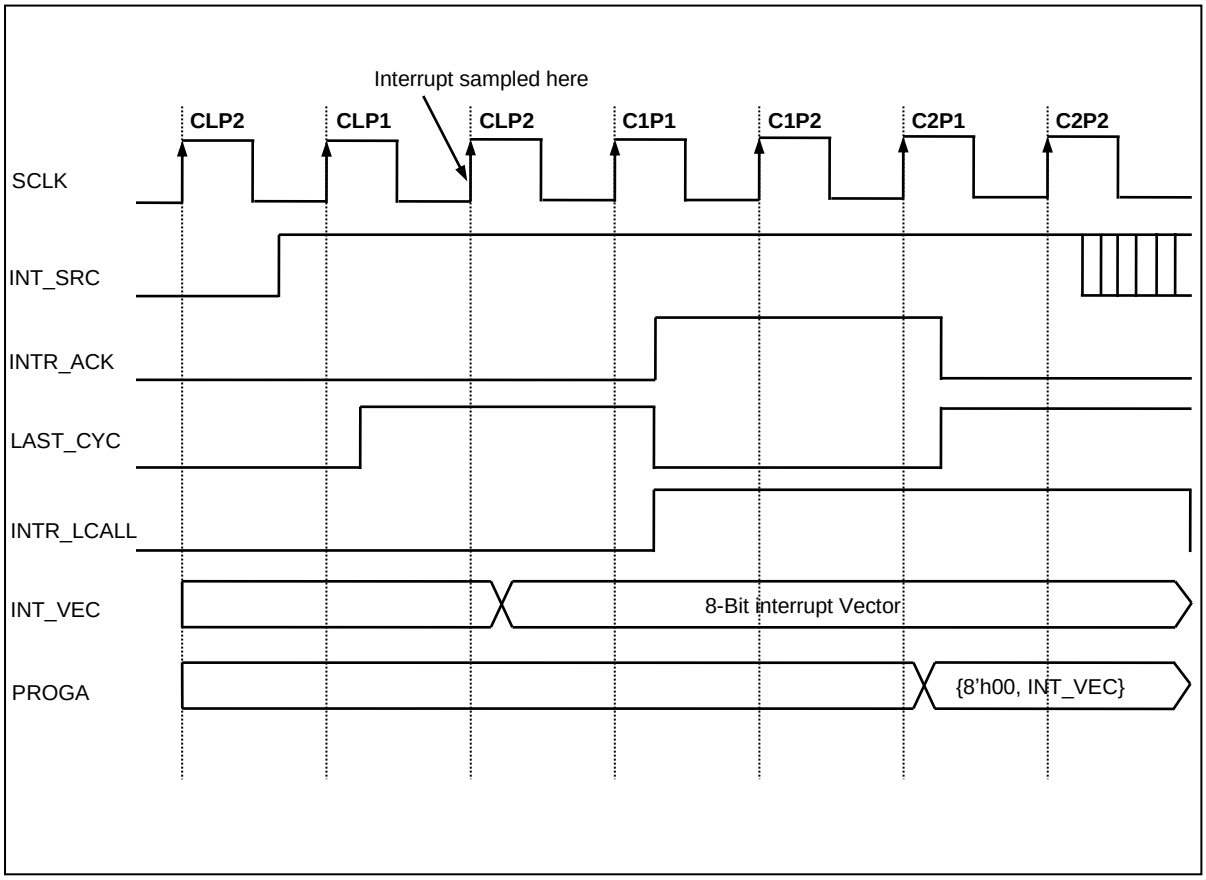


Figure 10-9 Timing chart of Interrupt Acceptance and Interrupt Return Instruction

Interrupt source sampled at last cycle of the command. When sampling interrupt source, it is decided to low 8-bit of interrupt vector. M8051W core makes interrupt acknowledge at first cycle of command, executes long call to jump interrupt routine as INT_VEC.

Note) command cycle CxPx : L=Last cycle, 1=1st cycle or 1st phase, 2=2nd cycle or 2nd phase

10.11 Interrupt Register Overview

10.11.1 Interrupt Enable Register (IE, IE1, IE2, IE3)

Interrupt enable register consists of Global interrupt control bit (EA) and peripheral interrupt control bits. Totally 24 peripheral are able to control interrupt.

10.11.2 Interrupt Priority Register (IP, IP1)

The 24 interrupt divides 6 groups which have each 4 interrupt sources. A group can decide 4 levels interrupt priority using interrupt priority register. Level 3 is the high priority, while level 0 is the low priority. Initially, IP, IP1 reset value is '0'. At that initialization, low interrupt number has a higher priority than high interrupt number. If decided the priority, low interrupt number has a higher priority than high interrupt number in that group.

10.11.3 Register Map

Table 10-2 Register Map

Name	Address	Dir	Default	Description
IE	A8H	R/W	00H	Interrupt Enable Register
IE1	A9H	R/W	00H	Interrupt Enable Register 1
IE2	AAH	R/W	00H	Interrupt Enable Register 2
IE3	ABH	R/W	00H	Interrupt Enable Register 3
IP	B8H	R/W	00H	Interrupt Priority Register
IP1	F8H	R/W	00H	Interrupt Priority Register 1
EIENAB	A3H	R/W	00H	Interrupt Enable Register
EIFLAG	A4H	R/W	00H	Interrupt Flag Register
EIEDGE	A5H	R/W	00H	Interrupt Edge Register
EIPOLA	A6H	R/W	00H	Interrupt Polarity Register
EIBOTH	A7H	R/W	00H	Interrupt Both Edge Register

10.12 Interrupt Register Description

The Interrupt Register is used for controlling interrupt functions. Also it has pin change interrupt control registers. The interrupt register consists of Interrupt Enable Register (IE), Interrupt Enable Register 1 (IE1), Interrupt Enable Register 2 (IE2), Interrupt Enable Register 3 (IE3), Interrupt Priority Register (IP), Interrupt Priority Register 1 (IP1). The pin change interrupt on P0 and P1 ports receive the bot edge (posedge and negedge) interrupt request.

10.12.1 Register Description for Interrupt

IE (Interrupt Enable Register) : A8H

7	6	5	4	3	2	1	0
EA	-	INT5E	INT4E	INT3E	INT2E	INT1E	INT0E
R/W	-	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

EA Enable or disable all interrupt bits
- 0 All Interrupt disable
- 1 All Interrupt enable

INT5E Enable or disable Touch Raw Data Sync Interrupt
- 0 Disable
- 1 Enable

INT4E Enable or disable WT Interrupt
- 0 Disable
- 1 Enable

INT3E Enable or disable UART TX0 Interrupt
- 0 Disable
- 1 Enable

INT2E Enable or disable UART RX0 Interrupt
- 0 Disable
- 1 Enable

INT1E Enable or disable External Interrupt1
- 0 Disable
- 1 Enable

INT0E Enable or disable External Interrupt0
0 Disable
1 Enable

IE1 (Interrupt Enable Register 1) : A9H

7	6	5	4	3	2	1	0
-	-	INT11E	INT10E	INT9E	INT8E	INT7E	INT6E
-	-	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

INT11E Enable or disable UART TX1 Interrupt
0 Disable
1 Enable

INT10E Enable or disable UART RX1 Interrupt
0 Disable
1 Enable

INT9E Enable or disable I^2C Interrupt
0 Disable
1 Enable

INT8E Enable or disable Free Run Counter
0 Disable
1 Enable

INT7E Enable or disable ADC Interrupt
0 Disable
1 Enable

INT6E Enable or disable LED Interrupt
0 Disable
1 Enable

IE2 (Interrupt Enable Register 2) : AAH

7	6	5	4	3	2	1	0
-	-	INT17E	INT16E	INT15E	INT14E	INT13E	INT12E
-	-	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

INT17E Enable or disable LVI Interrupt
0 Disable
1 Enable

INT16E Enable or disable BIT Interrupt
0 Disable
1 Enable

INT15E Enable or disable Watch Dog Timer Interrupt
0 Disable
1 Enable

INT14E Enable or disable Timer 2 Interrupt
0 Disable
1 Enable

INT13E Enable or disable Timer 1 Interrupt
0 Disable

	1	Enable
INT12E	Enable or disable Timer 0 Interrupt	
	0	Disable
	1	Enable

IE3 (Interrupt Enable Register 3) : ABH

7	6	5	4	3	2	1	0
-	-	INT23E	INT22E	INT21E	INT20E	INT19E	INT18E
-	-	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

INT23E	Enable or disable External Interrupt7	
	0	Disable
	1	Enable
INT22E	Enable or disable External Interrupt6	
	0	Disable
	1	Enable
INT21E	Enable or disable External Interrupt5	
	0	Disable
	1	Enable
INT20E	Enable or disable External Interrupt4	
	0	Disable
	1	Enable
INT19E	Enable or disable External Interrupt3	
	0	Disable
	1	Enable
INT18E	Enable or disable External Interrupt2	
	0	Disable
	1	Enable

IP (Interrupt Priority Register) : B8H

7	6	5	4	3	2	1	0
-	-	IP5	IP4	IP3	IP2	IP1	IP0
-	-	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

IP1 (Interrupt Priority Register 1) : F8H

7	6	5	4	3	2	1	0
-	-	IP15	IP14	IP13	IP12	IP11	IP10
-	-	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

IP1[5:0], IP[5:0]	Select Interrupt Group Priority		
	IP1	IP	Description
	0	0	level 0 (default : lowest)
	0	1	level 1
	1	0	level 2
	1	1	level 3 (highest)

EIENAB (External Interrupt Enable Register) : A3H

7	6	5	4	3	2	1	0
ENAB7	ENAB6	ENAB5	ENAB4	ENAB3	ENAB2	ENAB1	ENAB0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

ENAB7 Enable or Disable External Interrupt 7
- 0 Disable External Interrupt 7(default)
- 1 Enable External Interrupt 7

ENAB6 Enable or Disable External Interrupt 6
- 0 Disable External Interrupt 6(default)
- 1 Enable External Interrupt 6

ENAB5 Enable or Disable External Interrupt 5
- 0 Disable External Interrupt 5(default)
- 1 Enable External Interrupt 5

ENAB4 Enable or Disable External Interrupt 4
- 0 Disable External Interrupt 4(default)
- 1 Enable External Interrupt 4

ENAB3 Enable or Disable External Interrupt 3
- 0 Disable External Interrupt 3(default)
- 1 Enable External Interrupt 3

ENAB2 Enable or Disable External Interrupt 2
- 0 Disable External Interrupt 2(default)
- 1 Enable External Interrupt 2

ENAB1 Enable or Disable External Interrupt 1
- 0 Disable External Interrupt 1(default)
- 1 Enable External Interrupt 1

ENAB0 Enable or Disable External Interrupt 0
- 0 Disable External Interrupt 0(default)
- 1 Enable External Interrupt 0

EIFLAG (External Interrupt Flag Register) : A4H

7	6	5	4	3	2	1	0
FLAG7	FLAG6	FLAG5	FLAG4	FLAG3	FLAG2	FLAG1	FLAG0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

If External Interrupt is occurred, the flag becomes ‘1’. The flag can be cleared by writing a ‘0’ to bit. It is also cleared automatically after interrupt service routine is served.

FLAG7 When External Interrupt 7 is occurred this bit is set.
- 0 External Interrupt 7 is not occurred
- 1 External Interrupt 7 is occurred

FLAG6 When External Interrupt 6 is occurred this bit is set.
- 0 External Interrupt 6 is not occurred
- 1 External Interrupt 6 is occurred

FLAG5 When External Interrupt 5 is occurred this bit is set.
- 0 External Interrupt 5 is not occurred
- 1 External Interrupt 5 is occurred

FLAG4 When External Interrupt 4 is occurred this bit is set.
0 External Interrupt 4 is not occurred
1 External Interrupt 4 is occurred

FLAG3 When External Interrupt 3 is occurred this bit is set.
0 External Interrupt 3 is not occurred
1 External Interrupt 3 is occurred

FLAG2 When External Interrupt 2 is occurred this bit is set.
0 External Interrupt 2 is not occurred
1 External Interrupt 2 is occurred

FLAG1 When External Interrupt 1 is occurred this bit is set.
0 External Interrupt 1 is not occurred
1 External Interrupt 1 is occurred

FLAG0 When External Interrupt 0 is occurred this bit is set.
0 External Interrupt 0 is not occurred
1 External Interrupt 0 is occurred

EIEDGE (External Interrupt Edge Register) : A5H

7	6	5	4	3	2	1	0
EDGE7	EDGE6	EDGE5	EDGE4	EDGE3	EDGE2	EDGE1	EDGE0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

EDGE7 Determines the type of External interrupt 7, edge or level sensitive.
0 Level (default)
1 Edge

EDGE6 Determines the type of External interrupt 6, edge or level sensitive.
0 Level (default)
1 Edge

EDGE5 Determines the type of External interrupt 5, edge or level sensitive.
0 Level (default)
1 Edge

EDGE4 Determines the type of External interrupt 4, edge or level sensitive.
0 Level (default)
1 Edge

EDGE3 Determines the type of External interrupt 3, edge or level sensitive.
0 Level (default)
1 Edge

EDGE2 Determines the type of External interrupt 2, edge or level sensitive.
0 Level (default)
1 Edge

EDGE1 Determines the type of External interrupt 1, edge or level sensitive.
0 Level (default)
1 Edge

EDGE0 Determines the type of External interrupt 0, edge or level sensitive.
0 Level (default)
1 Edge

EIPOLA (External Interrupt Polarity Register) : A6H

7	6	5	4	3	2	1	0
POLA7	POLA6	POLA5	POLA4	POLA3	POLA2	POLA1	POLA0

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

According to EIEDGE, this register acts differently. If EIEDGE is level type, external interrupt polarity have level value. If EIEDGE is edge type, external interrupt polarity have edge value.

POLA7 Determine the polarity of External Interrupt 7
- 0 When High level or rising edge, Interrupt occur(default)
- 1 When Low level or falling edge, Interrupt occur

POLA6 Determine the polarity of External Interrupt 6
- 0 When High level or rising edge, Interrupt occur(default)
- 1 When Low level or falling edge, Interrupt occur

POLA5 Determine the polarity of External Interrupt 5
- 0 When High level or rising edge, Interrupt occur(default)
- 1 When Low level or falling edge, Interrupt occur

POLA4 Determine the polarity of External Interrupt 4
- 0 When High level or rising edge, Interrupt occur(default)
- 1 When Low level or falling edge, Interrupt occur

POLA3 Determine the polarity of External Interrupt 3
- 0 When High level or rising edge, Interrupt occur(default)
- 1 When Low level or falling edge, Interrupt occur

POLA2 Determine the polarity of External Interrupt 2
- 0 When High level or rising edge, Interrupt occur(default)
- 1 When Low level or falling edge, Interrupt occur

POLA1 Determine the polarity of External Interrupt 1
- 0 When High level or rising edge, Interrupt occur(default)
- 1 When Low level or falling edge, Interrupt occur

POLA0 Determine the polarity of External Interrupt 0
- 0 When High level or rising edge, Interrupt occur(default)
- 1 When Low level or falling edge, Interrupt occur

EIBOTH (External Interrupt Both Edge Enable Register) : A7H

7	6	5	4	3	2	1	0
BOTH7	BOTH6	BOTH5	BOTH4	BOTH3	BOTH2	BOTH1	BOTH0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

If BOTHx is written to ‘1’, the corresponding external pin interrupt is enabled by both edges(no level).
And EIEDGE and EIPOLA register value are ignored.

BOTH7 Determine the type of External Interrupt 7
- 0 Both edge detection Disable (default)
- 1 Both edge detection Enable

BOTH6 Determine the type of External Interrupt 6
- 0 Both edge detection Disable (default)
- 1 Both edge detection Enable

BOTH5 Determine the type of External Interrupt 5
- 0 Both edge detection Disable (default)
- 1 Both edge detection Enable

BOTH4 Determine the type of External Interrupt 4
- 0 Both edge detection Disable (default)

	1	Both edge detection Enable
BOTH3	Determine the type of External Interrupt 3	
	0	Both edge detection Disable (default)
	1	Both edge detection Enable
BOTH2	Determine the type of External Interrupt 2	
	0	Both edge detection Disable (default)
	1	Both edge detection Enable
BOTH1	Determine the type of External Interrupt 1	
	0	Both edge detection Disable (default)
	1	Both edge detection Enable
BOTH0	Determine the type of External Interrupt 0	
	0	Both edge detection Disable (default)
	1	Both edge detection Enable

11. Peripheral Hardware

11.1 Clock Generator

11.1.1 Overview

As shown in Figure 11-1, the clock generator produces the basic clock pulses which provide the system clock to be supplied to the CPU and the peripheral hardware. It contains main-frequency clock oscillator. The default system clock is INT-RC Oscillator and the default division rate is one. In order to stabilize system internally, use 256kHz RING oscillator for BIT, WDT and ports de-bounce.

- Calibrated Internal RC Oscillator (16 MHz)
  . INT-RC OSC/1 (16 MHz)
  . INT-RC OSC/2 (8 MHz)
  . INT-RC OSC/4 (4 MHz)
  . INT-RC OSC/8 (2 MHz)

11.1.2 Block Diagram

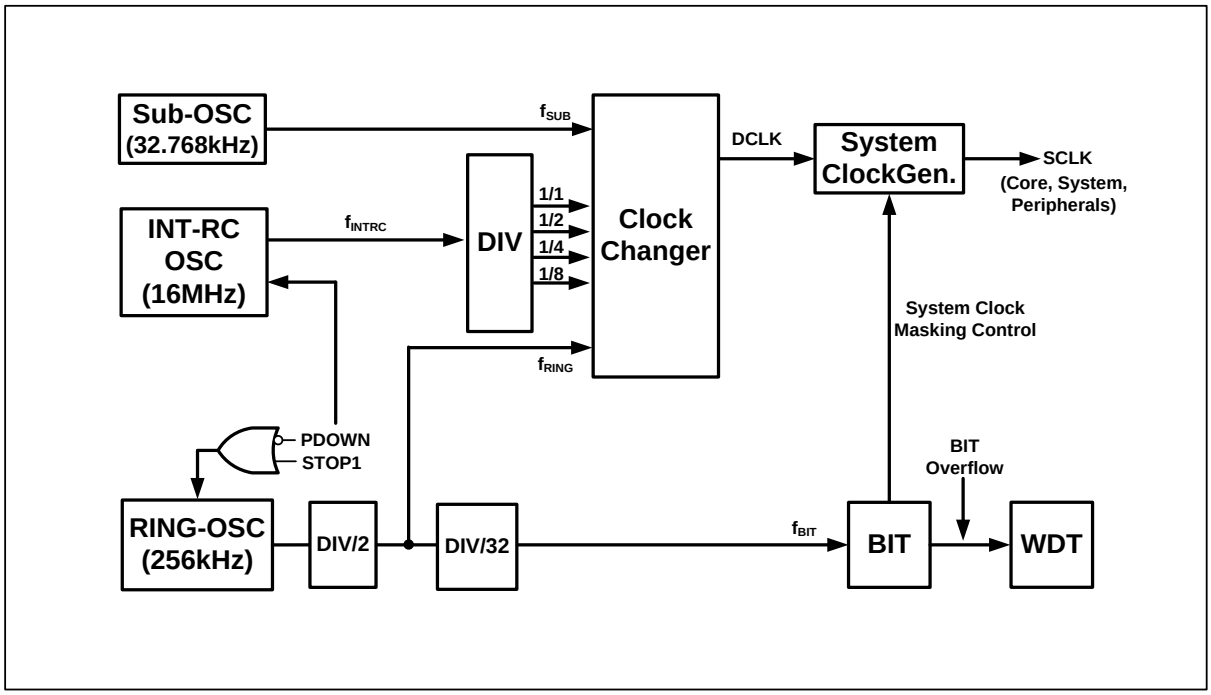


Figure 11-1 Clock Generator Block Diagram

11.1.3 Register Map

Table 11-1 Register Map

Name	Address	Dir	Default	Description
SCCR	8AH	R/W	00H	System and Clock Control Register

11.1.4 Clock Generator Register Description

The Clock Generation Register uses clock control for system operation. The clock generation consists of System and Clock register.

11.1.5 Register Description for Clock Generator

SCCR (System and Clock Control Register) : 8AH

7	6	5	4	3	2	1	0
STOP1	DIV1	DIV0	CBYS	ISTOP	REDUCEC	CS1	CS0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value :00H

STOP1 Control the STOP Mode

Note) When PCON=0x03, This bit is applied. When PCON=0x01, This bit is not applied .

0 STOP2 Mode (at PCON=0x03) (default)

1 STOP1 Mode (at PCON=0x03)

DIV[1:0] When using f_{INTRC} as system clock, determine division rate. Note) when using f_{INTRC} as system clock, only division rate come into effect.

Note) To change by software, CBYS set to ‘1’

DIV1	DIV0	description
0	0	$f_{INTRC/1}$ (16MHz, default)
0	1	$f_{INTRC/2}$ (8MHz)
1	0	$f_{INTRC/4}$ (4MHz)
1	1	$f_{INTRC/8}$ (2MHz)

CBYS Control the scheme of clock change. If this bit set to ‘0’, clock change is controlled by hardware. But if this set to ‘1’, clock change is controlled by software. Ex) when setting CS, if CBYS bit set to ‘0’, it is not changed right now, CPU goes to STOP mode and then when wake-up, it applies to clock change.

Note) when clear this bit, keep other bits in SCCR.

0 Clock changed by hardware during stop mode (default)

1 Clock changed by software

ISTOP Control the operation of INT-RC Oscillation

Note) when CBYS=’1’, It is applied

0 RC-Oscillation enable (default)

1 RC-Oscillation disable

REDUCEC When STOP1 Mode is set, This bit is applied for reduce current

0 Normal current (default)

1 Reduce current

CS[1:0] Determine System Clock

Note) by CBYS bit, reflection point is decided

CS1	CS0	Description
0	0	f_{INTRC} INTRC (16 MHz, default)
0	1	f_{INTRC} INTRC (16 MHz)
1	0	f_{SUBOSC} (32.768 kHz)
1	1	f_{RING} (128 kHz)

11.2 BIT

11.2.1 Overview

The 8-bit Basic Interval Timer that is free-run and can't stop. Block diagram is shown in Figure 11-2. In addition, the Basic Interval Timer generates the time base for watchdog timer counting. It also provides a Basic interval timer interrupt (BITF).

The Basic Interval Timer (BIT) features:

- During Power On, BIT gives a stable clock generation time
- On exiting Stop mode, BIT gives a stable clock generation time
- As clock function, time interrupt occurrence

11.2.2 Block Diagram

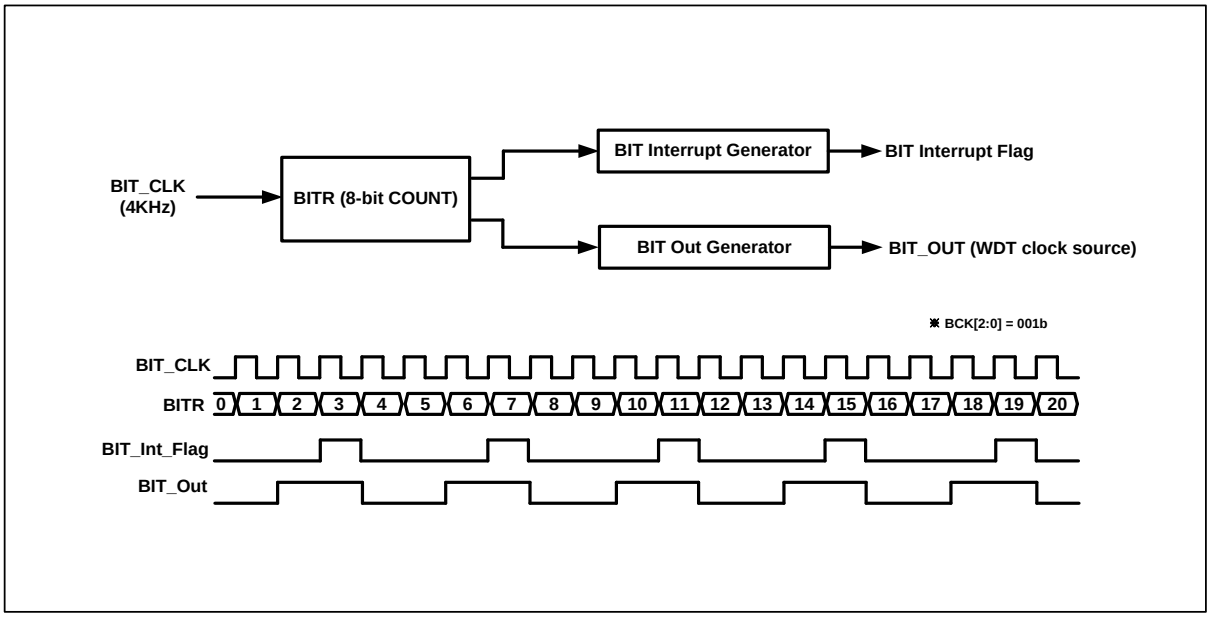


Figure 11-2 BIT Block Diagram

11.2.3 Register Map

Table 11-2 Register Map

Name	Address	Dir	Default	Description
BCCR	8BH	R/W	05H	BIT Clock Control Register
BITR	8CH	R	00H	Basic Interval Timer Register

11.2.4 Bit Interval Timer Register Description

The Bit Interval Timer Register consists of BIT Clock control register (BCCR) and Basic Interval Timer register (BITR). If BCLR bit set to '1', BITR becomes '0' and then counts up. After 1 machine cycle, BCLR bit is cleared as '0' automatically.

11.2.5 Register Description for Bit Interval Timer

BCCR (BIT Clock Control Register) : 8BH

7	6	5	4	3	2	1	0
BITF	-	-	-	BCLR	BCK2	BCK1	BCK0
R/W	R	R	R	R/W	R/W	R/W	R/W

Initial value : 05H

BITF When BIT Interrupt occurs, this bit becomes ‘1’. For clearing bit, write ‘0’ to this bit.

0 no generation

1 generation

BCLR If BCLR Bit is written to ‘1’, BIT Counter is cleared as ‘0’

0 Free Running

1 Clear Counter

BCK[2:0] Select BIT overflow period (BIT Clock 4KHz)

BCK2	BCK1	BCK0	
0	0	0	0.5msec (BIT Clock * 2)
0	0	1	1msec
0	1	0	2msec
0	1	1	4msec
1	0	0	8msec
1	0	1	16msec (default)
1	1	0	32msec
1	1	1	64msec

BITR (Basic Interval Timer Register) : 8CH

7	6	5	4	3	2	1	0
BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R	R	R	R	R	R	R	R

Initial value : 00H

BIT[7:0] BIT Counter

11.3 WDT

11.3.1 Overview

The watchdog timer rapidly detects the CPU malfunction such as endless looping caused by noise or the like, and resumes the CPU to the normal state. The watchdog timer signal for detecting malfunction can be selected either a reset CPU or an interrupt request. When the watchdog timer is not being used for malfunction detection, it can be used as a timer to generate an interrupt at fixed intervals. It is possible to use free running 8-bit timer mode (WDTRSON='0') or watch dog timer mode (WDTRSON='1') as setting WDTMR[6] bit. If writing WDTMR[5] to '1', WDT counter value is cleared and counts up. After 1 machine cycle, this bit has '0' automatically. The watchdog timer consists of 8-bit binary counter and the watchdog timer data register. When the value of 8-bit binary counter is equal to the 8 bits of WDTR, the interrupt request flag is generated. This can be used as Watchdog timer interrupt or reset the CPU in accordance with the bit WDTRSON.

The clock source of Watch Dog Timer is BIT overflow output. The interval of watchdog timer interrupt is decided by BIT overflow period and WDTR set value. The equation is as below

WDT Interrupt Interval = (BIT Interrupt Interval) X (WDTR Value+1)

11.3.2 Block Diagram

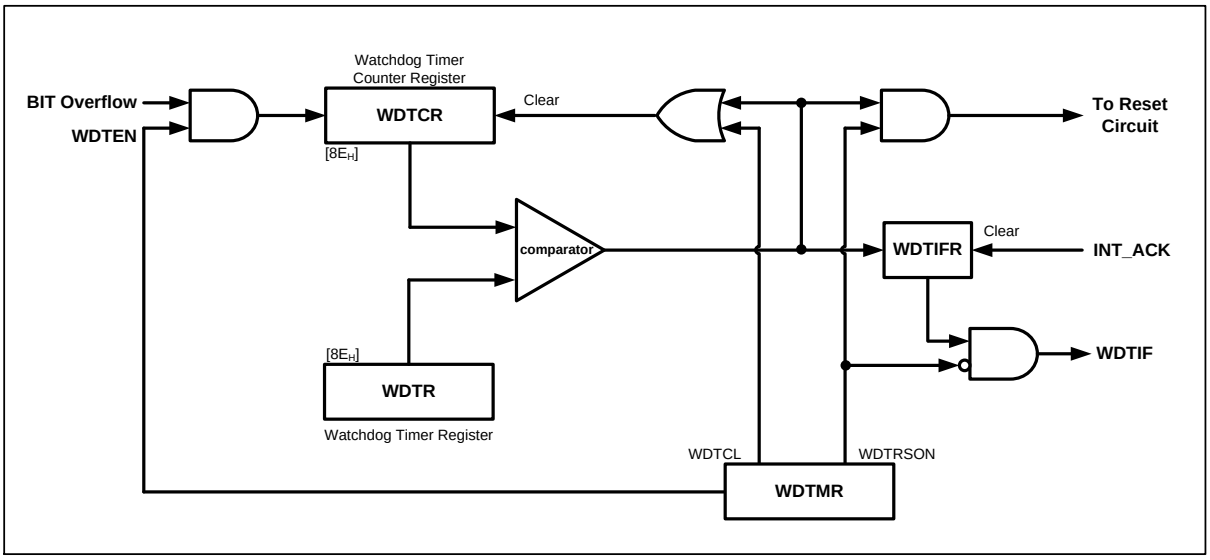


Figure 11-3 WDT Block Diagram

11.3.3 Register Map

Table 11-3 Register Map

Name	Address	Dir	Default	Description
WDTR	8EH	W	FFH	Watch Dog Timer Register
WDTCR	8EH	R	00H	Watch Dog Timer Counter Register
WDTMR	8DH	R/W	00H	Watch Dog Timer Mode Register

11.3.4 Watch Dog Timer Register Description

The Watch dog timer (WDT) Register consists of Watch Dog Timer Register (WDTR), Watch Dog Timer Counter Register (WDTCR) and Watch Dog Timer Mode Register (WDTMR).

11.3.5 Register Description for Watch Dog Timer

WDTR (Watch Dog Timer Register: Write Case) : 8EH

7	6	5	4	3	2	1	0
WDTR7	WDTR6	WDTR5	WDTR4	WDTR3	WDTR2	WDTR1	WDTR0
W	W	W	W	W	W	W	W

Initial value : FFH

WDTR[7:0] Set a period
WDT Interrupt Interval=(BIT Interrupt Interval) x(WDTR Value+1)

Note) To guarantee proper operation, the data should be greater than 01H.

WDTCR (Watch Dog Timer Counter Register: Read Case) : 8EH

7	6	5	4	3	2	1	0
WDTCR7	WDTCR6	WDTCR5	WDTCR4	WDTCR3	WDTCR2	WDTCR1	WDTCR0
R	R	R	R	R	R	R	R

Initial value : 00H

WDTCR[7:0] WDT Counter

WDTMR (Watch Dog Timer Mode Register) : 8DH

7	6	5	4	3	2	1	0
WDTEN	WDTRSON	WDTCL	-	-	-	-	WDTIFR
R/W	R/W	R/W	-	-	-	-	R/W

Initial value : 00H

WDTEN Control WDT operation
0 disable
1 enable

WDTRSON Control WDT Reset operation
0 Free Running 8-bit timer
1 Watch Dog Timer Reset ON

WDTCL Clear WDT Counter
0 Free Run
1 Clear WDT Counter (auto clear after 1 Cycle)
When WDT Interrupt occurs, this bit becomes ‘1’. For clearing bit, write

WDTIFR	‘0’ to this bit or auto clear by INT_ACK signal.	
	0	WDT Interrupt no generation
	1	WDT Interrupt generation

11.3.6 WDT Interrupt Timing Waveform

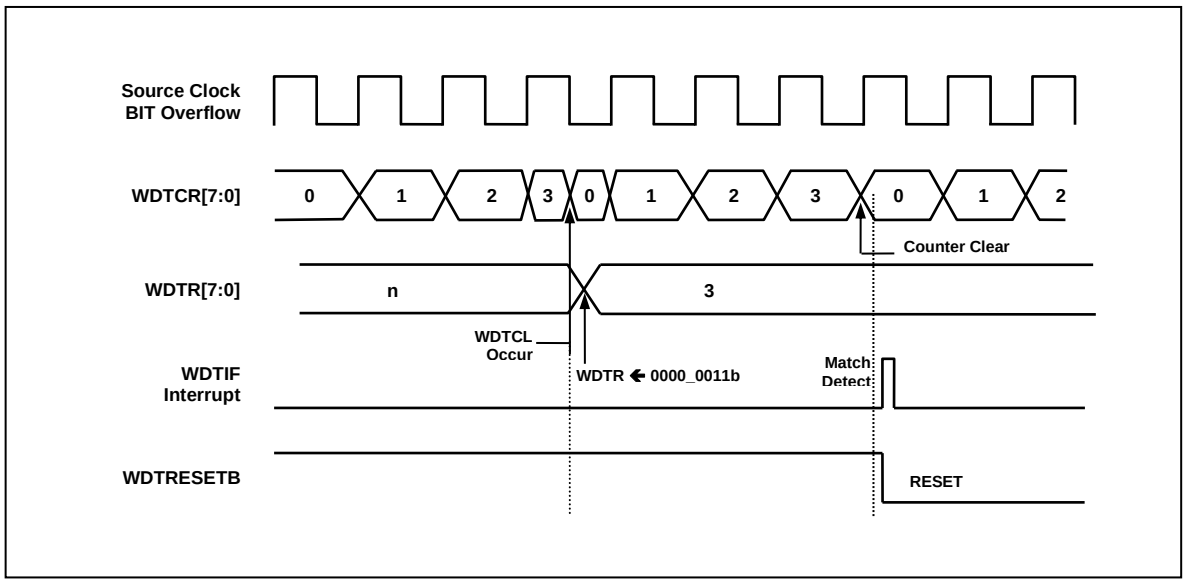


Figure 11-4 WDT Interrupt Timing Waveform

11.4 WT

11.4.1 Overview

The watch timer has the function for RTC (Real Time Clock) operation. It is generally used for RTC design. The internal structure of the watch timer consists of the clock source select circuit, timer counter circuit, output select circuit and watch timer mode register. To operate the watch timer, determine the input clock source, output interval and set WTEN to ‘1’ in watch timer mode register (WTMR). It is able to execute simultaneously or individually. To stop or reset WT, clear the WTEN bit in WTMR register. Even if CPU is STOP mode, sub clock is able to be alive so WT can continue the operation. The watch timer counter circuits may be composed of 21-bit counter which is low 14-bit with binary counter and high 7-bit with auto reload counter in order to raise resolution. In WTR, it can control WT clear and set Interval value at write time, and it can read 7-bit WT counter value at read time.

11.4.2 Block Diagram

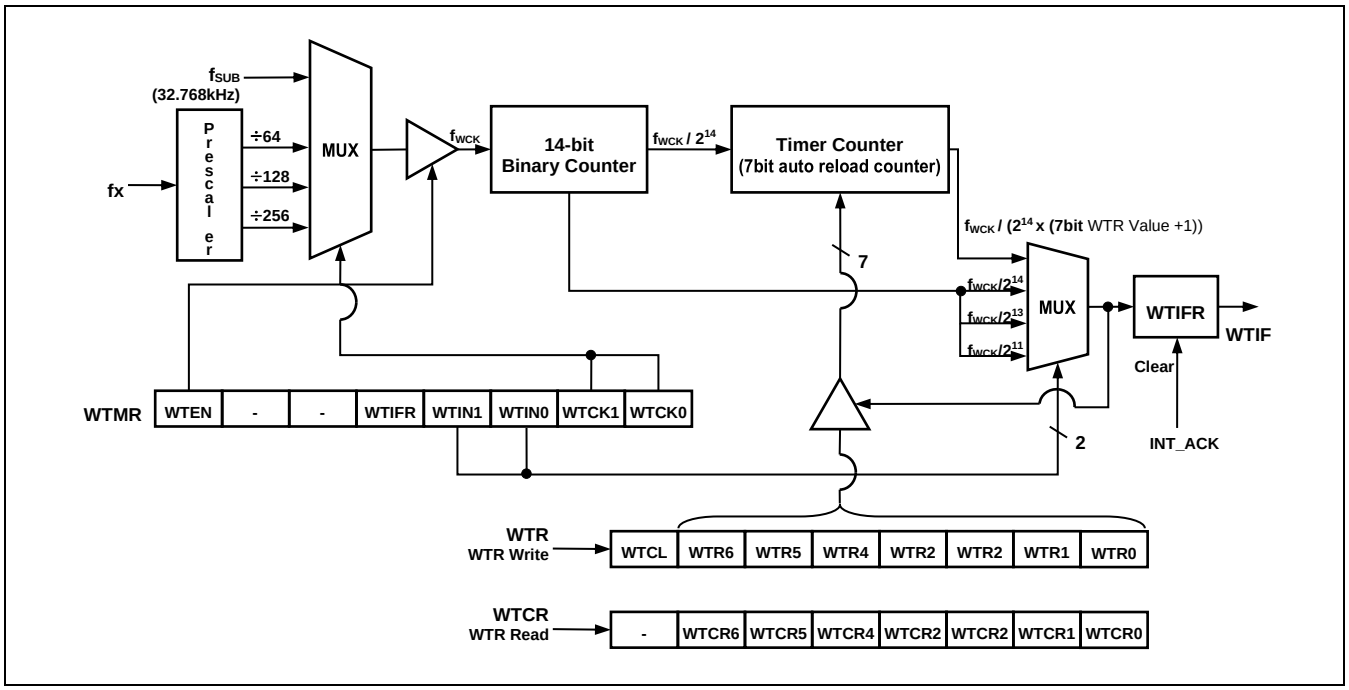


Figure 11-5 WT Block Diagram

11.4.3 Register Map

Table 11-4 Register Map

Name	Address	Dir	Default	Description
WTMR	C8H	R/W	00H	Watch Timer Mode Register
WTR	C9H	W	7FH	Watch Timer Register
WTCR	C9H	R	00H	Watch Timer Counter Register

11.4.4 Watch Dog Timer Register Description

The watch timer register (WT) consists of Watch Timer Mode Register (WTMR), Watch Timer Counter Register (WTCR) and Watch Timer Register (WTR). As WTMR is 6-bit writable/readable register, WTMR can control the clock source (WTCK), interrupt interval (WTIN) and function enable/disable (WTEN). Also there is WT interrupt flag bit (WTIFR).

11.4.5 Register Description for Watch Dog Timer

WTMR (Watch Timer Mode Register) : C8H

7	6	5	4	3	2	1	0
WTEN	-	-	WTIFR	WTIN1	WTIN0	WTCK1	WTCK0
R/W	-	-	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

WTEN Control Watch Timer

0 disable

1 enable

WTIFR When WT Interrupt occurs, this bit becomes ‘1’. For clearing bit, write ‘0’ to this bit or auto clear by INT_ACK signal.

0 WT Interrupt no generation

1 WT Interrupt generation

WTIN[1:0] Determine interrupt interval

WTIN1	WTIN0	description
0	0	$fwck/2^{11}$
0	1	$fwck/2^{13}$
1	0	$fwck/2^{14}$
1	1	$fwck/(2^{14} \times (7bit\ WT\ Value + 1))$

WTCK[1:0] Determine Source Clock

WTCK1	WTCK0	description
0	0	fsub
0	1	fx/256
1	0	fx/128
1	1	fx/64

NOTE)

1. Remark: fx– Main system clock oscillation frequency
2. fsub- Sub clock oscillation frequency
3. fwck- selected Watch Timer clock

WTR (Watch Timer Register: Write Case) : C9H

7	6	5	4	3	2	1	0
WTCL	WTR6	WTR5	WTR4	WTR3	WTR2	WTR1	WTR0
W	W	W	W	W	W	W	W

Initial value : 7FH

WTCL Clear WT Counter
0 Free Run
1 Clear WT Counter (auto clear after 1 Cycle)

WTR[6:0] Set WT period
WT Interrupt Interval=fwck/(2^{14} x(7bit WT Value+1))

NOTE)

1. To guarantee proper operation, it is greater than 01H to write WTR.

WTCR (Watch Timer Counter Register: Read Case) : C9H

7	6	5	4	3	2	1	0
	WTCR6	WTCR5	WTCR4	WTCR3	WTCR2	WTCR1	WTCR0
-	R	R	R	R	R	R	R

Initial value : 00H

WTCR[6:0] WT Counter

11.5 Free Run Counter

11.5.1 Overview

The clock source of the free run counter is BIT overflow output. The interval of free run counter interrupt is decided by BIT overflow period and 32 bits count value of the four registers. The counter value is the 32 bits value (unsigned long type) and stored to the four registers to easily read. The four registers are (MSB) FCHH, FCHL, FCLH, & FCLL (LSB). If the free run counter starts counting, counter value will be increased with 1 from 0x00000000 to 0xFFFFFFFF. If the counter value is 0xFFFFFFFF, it will be reset to 0x0000000 and continue the counting eternally. If the count period is about 1ms, the full counting time will be about 49 days.

When the count value becomes 0xFFFFFFFF, the interrupt request flag is generated (Option). If the free run counter is not needed, reset the FCEN bit of the Free Run Counter Enable Register (FCMR) to be disable the free run counter.

11.5.2 Block Diagram

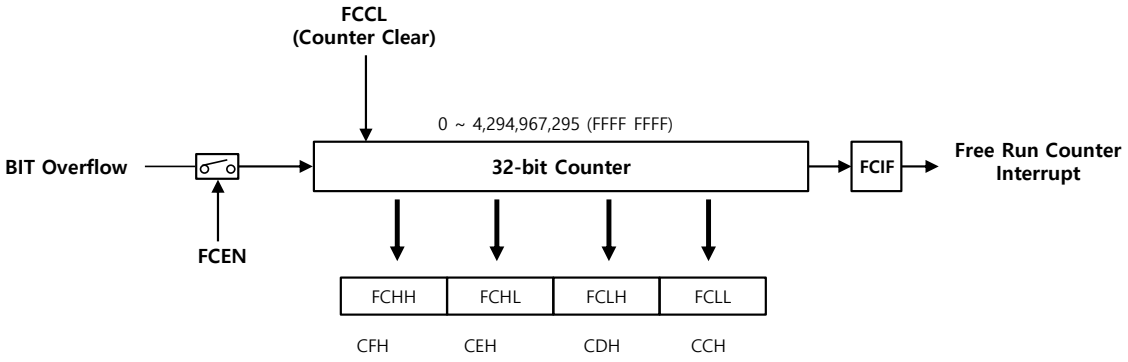


Figure 11-6 Free Run Counter Block Diagram

11.5.3 Register Map

Table 11-5 Register Map

이름	어드레스	방향	Initial value	설명
FCMR	2F44H	R/W	00H	Free Run Counter Enable Register
FCLL	2F43H	R	00H	Free Run Counter Low L
FCLH	2F42H	R	00H	Free Run Counter Low H
FCHL	2F41H	R	00H	Free Run Counter High L
FCHH	2F40H	R	00H	Free Run Counter High H

11.5.4 Free Run Counter Register Description

Free Run Counter consists of Free Run Counter Enable Register (FCMR), Free Run Counter Register (FCHH, FCHL, FCLH, FCLL).

11.5.5 Register Description for Free Run Counter

FCMR (Free Run Counter Enable Register) : 2F44H

7	6	5	4	3	2	1	0
FCEN	FCCL	-	-	-	-	-	FCIF
W/R	W						W/R

Initial value : 00H

FCEN Free Run Counter Operation Control
- 0 Free Run Counter Disable (Default)
- 1 Free Run Counter Enable

FCCL Free Run Counter Clear

FCIF Free Run Counter Interrupt Flag
- 0 No Free Run Counter Interrupt
- 1 Free Run Counter inrerrupt occurred, write”0” to clear interrupt flag

FCLL (Free Run Counter Low L) : 2F43H

7	6	5	4	3	2	1	0
FCLL7	FCLL6	FCLL5	FCLL4	FCLL3	FCLL2	FCLL1	FCLL0
R	R	R	R	R	R	R	R

Initial value : 00H

FCLL[7:0] Free Run Counter Low L

FCLH (Free Run Counter Low H) : 2F42H

7	6	5	4	3	2	1	0
FCLH7	FCLH6	FCLH5	FCLH4	FCLH3	FCLH2	FCLH1	FCLH0
R	R	R	R	R	R	R	R

Initial value : 00H

FCLH[7:0] Free Run Counter Low H

FCHL (Free Run Counter High L) : 2F41H

7	6	5	4	3	2	1	0
FCHL7	FCHL6	FCHL5	FCHL4	FCHL3	FCHL2	FCHL1	FCHL0
R	R	R	R	R	R	R	R

Initial value : 00H

FCHL[7:0] Free Run Counter High L

FCHH (Free Run Counter High H) : 2F40H

7	6	5	4	3	2	1	0
FCHH7	FCHH6	FCHH5	FCHH4	FCHH3	FCHH2	FCHH1	FCHH0
R	R	R	R	R	R	R	R

Initial value : 00H

FCHH[7:0] Free Run Counter High H

11.6 Buzzer Driver

11.6.1 Overview

The Buzzer consists of 8-bit Counter and Buzzer Data Registers (BUZDR0, BUZDR1), Buzzer Control Register (BUZCR0, BUZCR1). The Square Wave (122.07Hz~250KHz @16MHz or 61.035Hz~125KHz, @8MHz) gets out of P06/BUZ0 or P15/BUZ1 pin. BUZDRx (Buzzer Data Registers) controls the Buzzer frequency (look at the following expression). In the BUZCRx (Buzzer Control Register), BUCK[1:0] selects source clock divided from prescaler.

$$f_{BUZ}(Hz) = \frac{\text{Oscillator Frequency}}{2 \times \text{Prescaler Ratio} \times (\text{BUZDR} + 1)}$$

Table 11-6 Buzzer Frequency at 16MHz

BUZDRx[7:0]	Buzzer Frequency (kHz)			
	BUZCRx[2:1]=00	BUZCRx[2:1]=01	BUZCRx[2:1]=10	BUZCRx[2:1]=11
0000_0000	250kHz	125.00kHz	62.50kHz	31.250kHz
0000_0001	125kHz	62.50kHz	31.25kHz	15.625kHz
…	…	…	…	…
1111_1101	984.252Hz	492.126Hz	246.063Hz	123.031Hz
1111_1110	980.392Hz	490.196Hz	245.098Hz	122.549Hz
1111_1111	976.563Hz	488.281Hz	244.141Hz	122.070Hz

11.6.2 Block Diagram

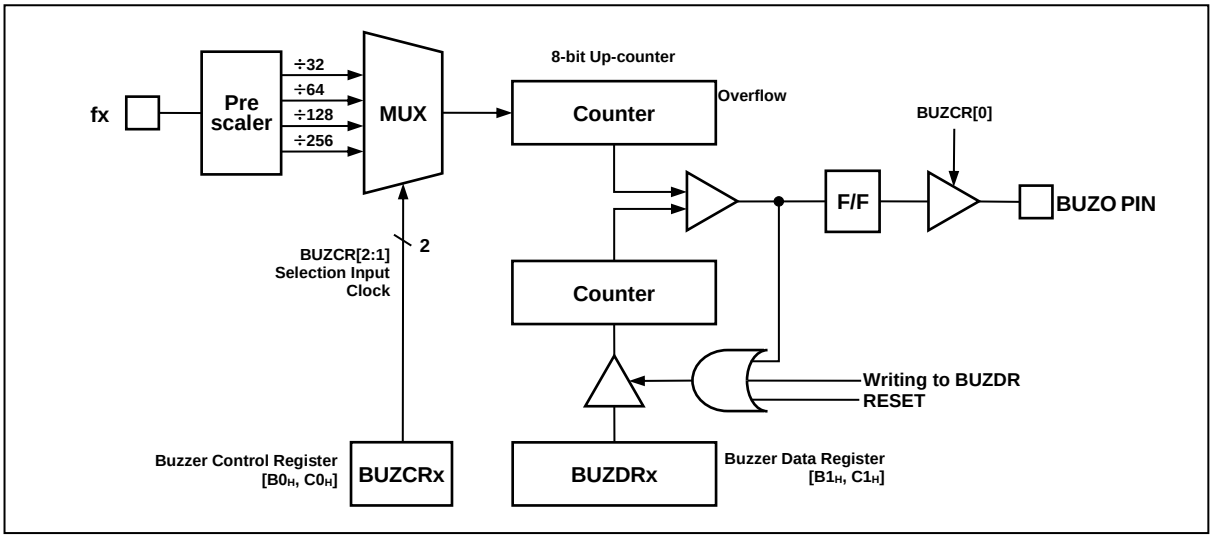


Figure 11-7 Buzzer Driver Block Diagram

11.6.3 Register Map

Table 11-7 Register Map

Name	Address	Dir	Default	Description
BUZCR0	B0H	R/W	00H	Buzzer Control Register 0
BUZDR0	B1H	R/W	FFH	Buzzer Data Register 0
BUZCR1	C0H	R/W	00H	Buzzer Control Register 1
BUZDR1	C1H	R/W	FFH	Buuzer Data Register 1

11.6.4 Buzzer Driver x Register Description

Buzzer Driver consists of Buzzer Data x Register (BUZDR0, BUZDR1), Buzzer Control x Register (BUZCR0, BUZCR1).

11.6.5 Register Description for Buzzer Driver

BUZCR0, BUZCR1 (Buzzer Control 0~1 Register) : B0H, B1H

7	6	5	4	3	2	1	0
-	-	-	-	-	BUCK1	BUCK0	BUZEN
-	-	-	-	-	R/W	R/W	R/W

Initial value : 00H

BUCK[1:0] Buzzer Driver Source Clock Selection

BUCK1	BUCK0	Source Clock
0	0	fx/32
0	1	fx/64
1	0	fx/128
1	1	fx/256

BUZEN Buzzer Driver Operation Control

0 Buzzer Driver disable

1 Buzzer Driver enable

Note) fx: Main system clock oscillation frequency

BUZDR0, BUZDR1 (Buzzer Data 0~1 Register) : B1H, C1H

7	6	5	4	3	2	1	0
BUZDR7	BUZDR6	BUZDR5	BUZDR4	BUZDR3	BUZDR2	BUZDR1	BUZDR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : FFH

BUZDR[7:0] This bits control the Buzzer frequency
Its resolution is 00H ~ FFH

11.7 Timer/PWM

11.7.1 16-bit Timer/Event Counter 0, 1, 2

11.7.1.1 Overview

The 16-bit timer x consists of Multiplexer, Timer Data Register High/Low, Timer Register High/Low, Timer Mode Control Register, PWM Duty High/Low, PWM Period High/Low Register It is able to use internal 16-bit timer/counter without a port output function.

The 16-bit timer x is able to use the divided clock of the main clock selected from prescaler output.

11.7.1.2 16-Bit Timer/Counter Mode

In the 16-bit Timer/Counter Mode, If the TxH + TxL value and the TxDRH + TxDRL value are matched, Tx/PWMx port outputs. The output is 50:50 of duty square wave, the frequency is following

$$f_{COMP} = \frac{\text{Timer Clock Frequency}}{2 \times \text{PrescalerValue} \times (TxDR + 1)}$$

f_{COMP} is timer output frequency and TxDR is the 16 bits value of TxDRH and TxDRL.

To export the compare output as Tx/PWMx, the Tx_PE bit in the TxCR1 register must set to ‘1’.

The 16-bit Timer/Counter Mode is selected by control registers as shown in Figure 11-8.

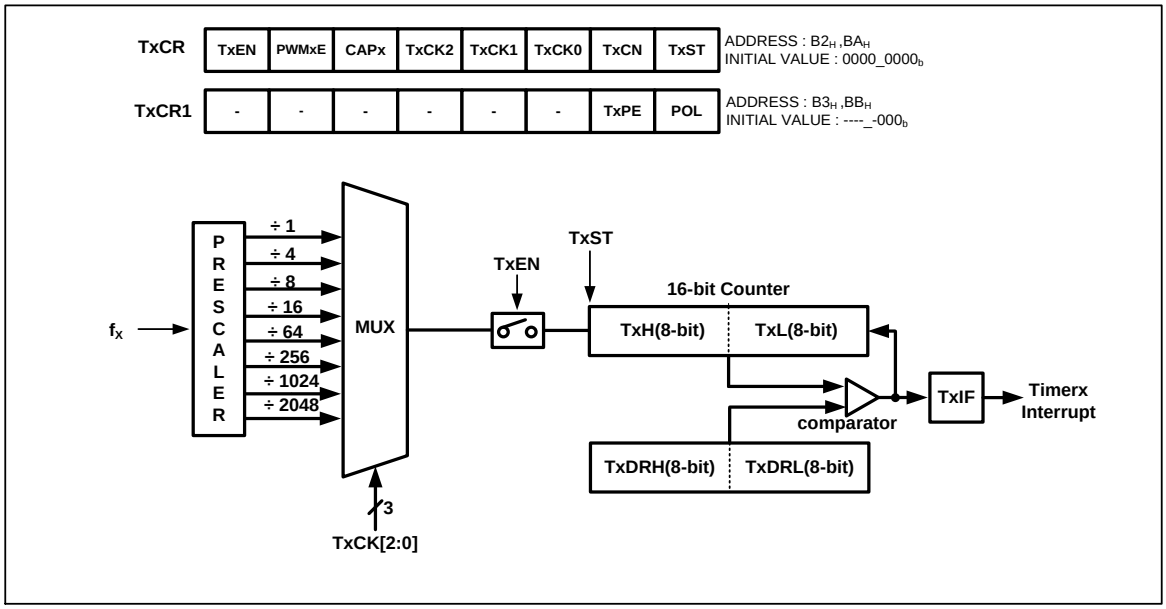


Figure 11-8 Timer x 16-bit Mode Block Diagram

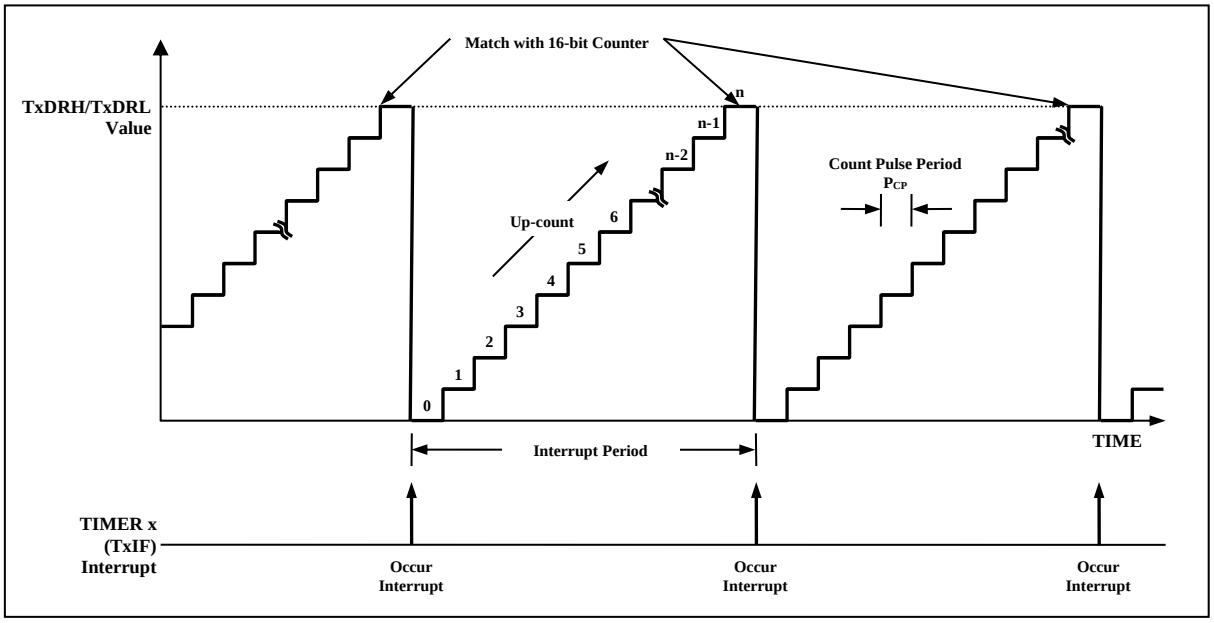


Figure 11-9 Interrupt of Timer/Counter Mode

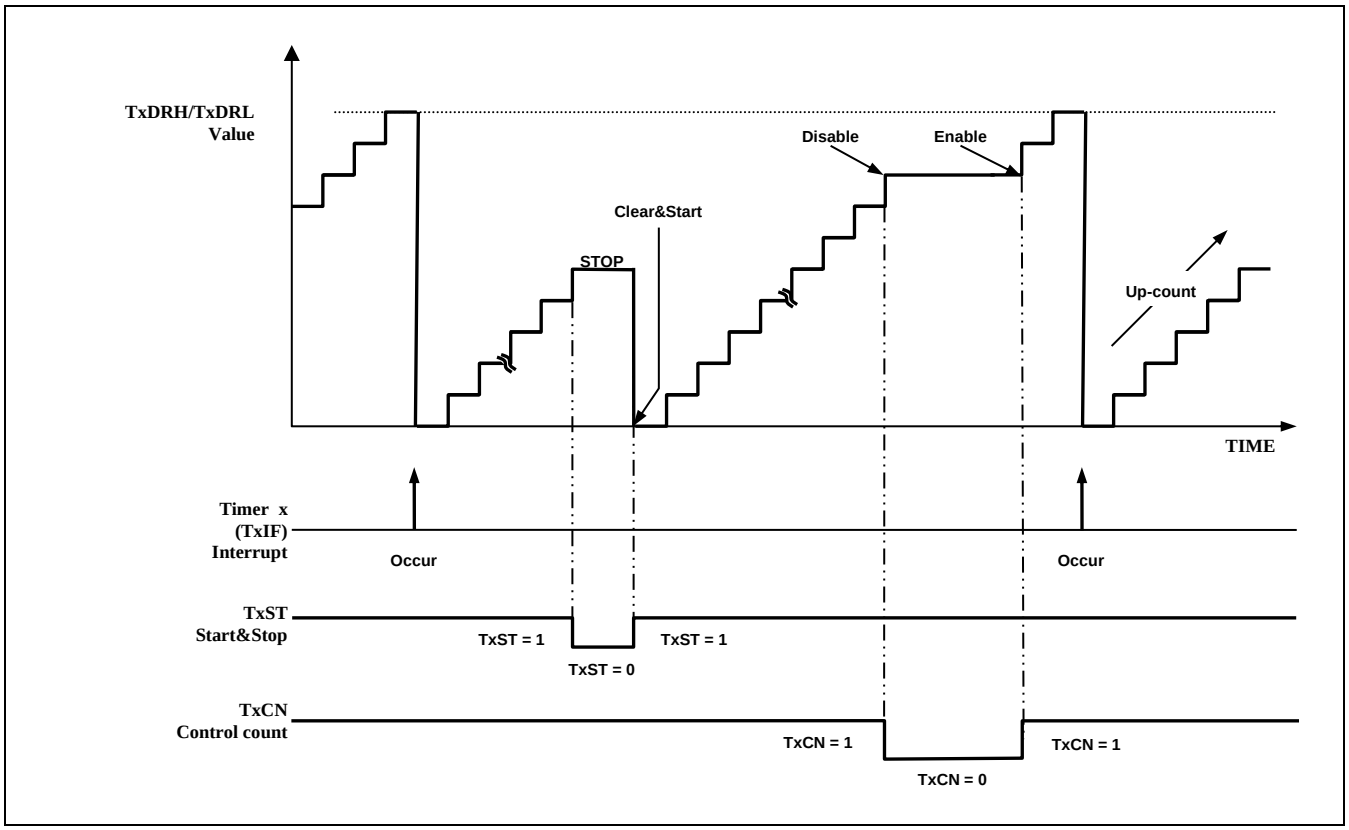


Figure 11-10 Operation Example of Timer/Event Counter Mode

11.7.1.3 16-Bit Capture Mode

The timer x capture mode is set by CAPx as '1' in TxCR register. The clock is same source as Output Compare mode. The interrupt occurs at TxH, TxL and TxDRH, TxDRL matching time. The capture result is loaded into CDRxH, CDRxL. The TxH, TxL value is automatically cleared(0000_H) by hardware and restarts counter.

This timer interrupt in capture mode is very useful when the pulse width of captured signal is wider than the maximum period of timer. As the EIEDGE and EIPOLA and EIBOTH register setting, the external interrupt INTx function is chosen.

The CDRxH, PWMxDRH and TxH are in same address. In the capture mode, reading operation is read the CDRxH, not TxH because path is opened to the CDRxH. PWMxDRH will be changed in writing operation. The PWMxDRL, TxL, CDRxL has the same function.

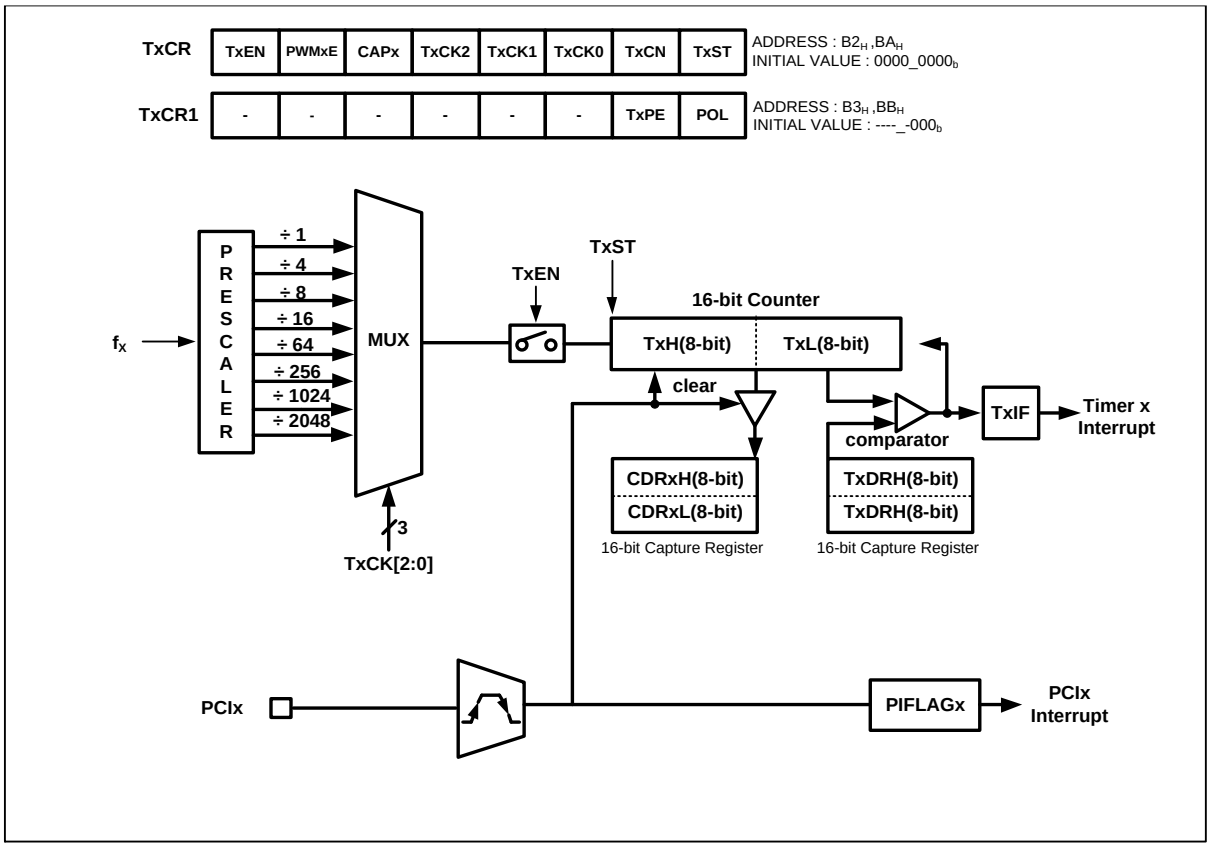


Figure 11-11 Timer x 16bit Capture Mode

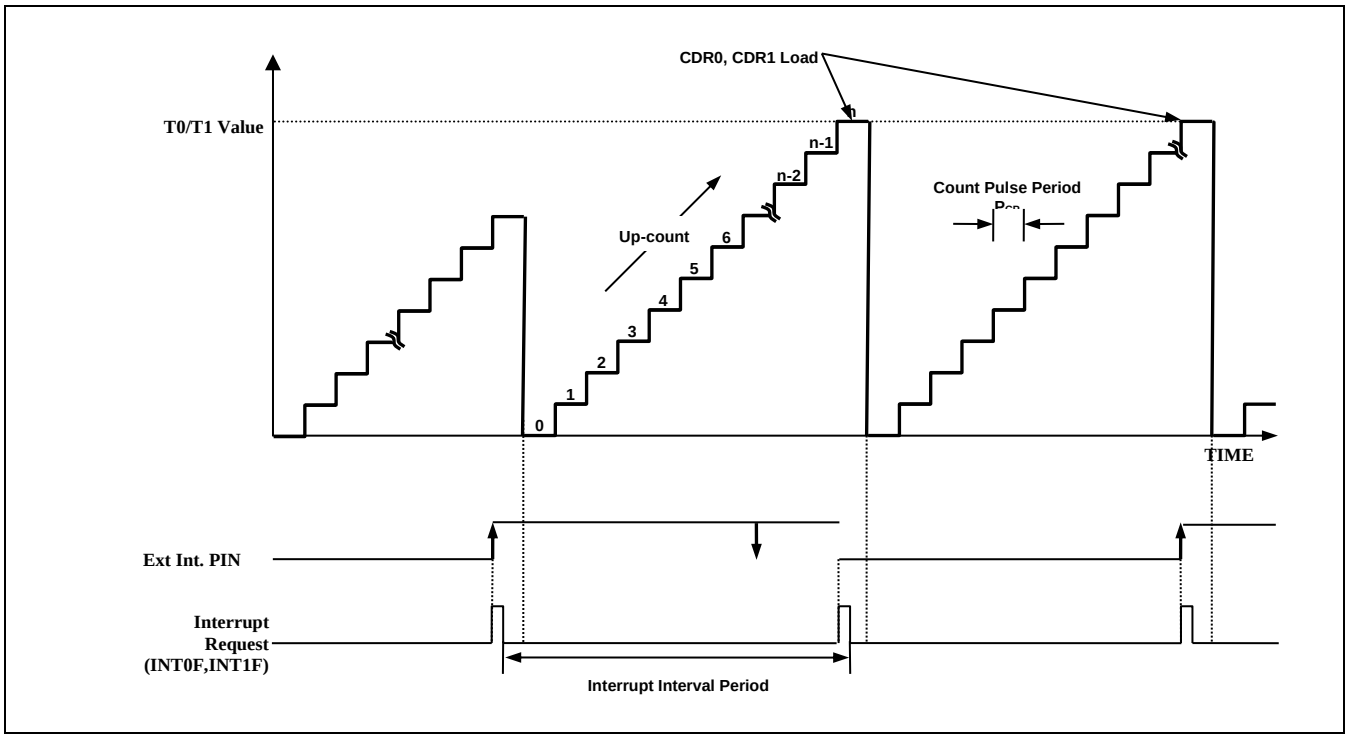


Figure 11-12 Input Capture Mode Operation of Timer 0, 1

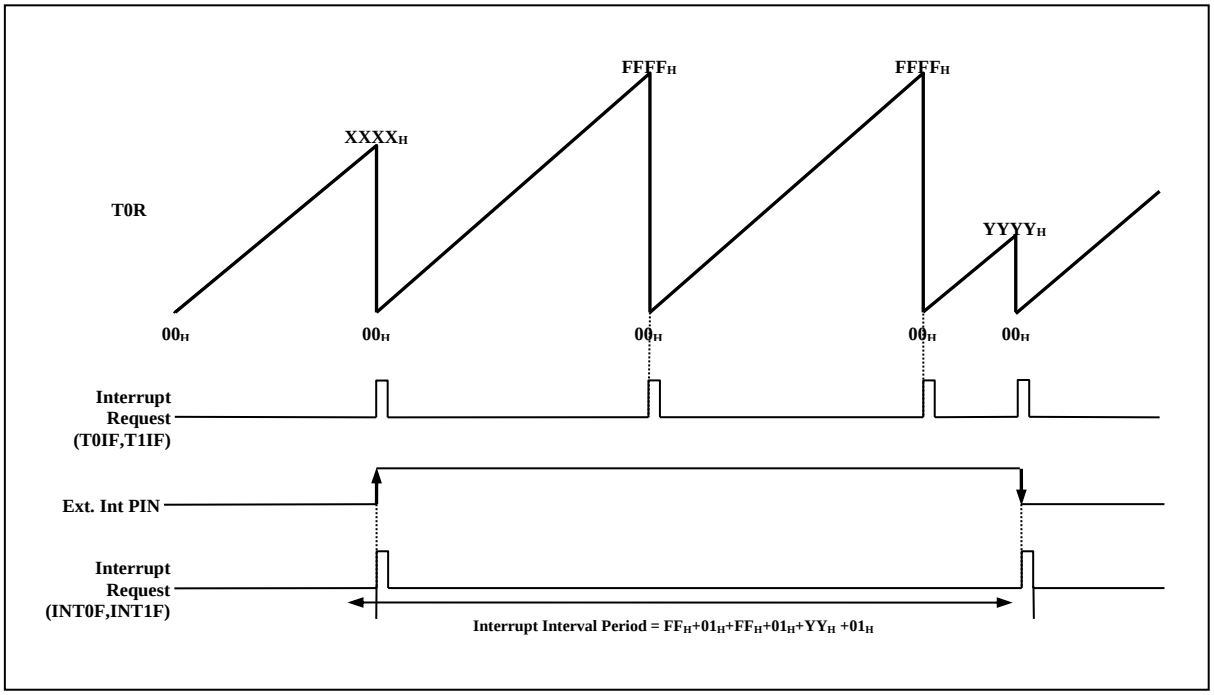


Figure 11-13 Express Timer Overflow in Capture Mode

11.7.1.4 PWM Mode

The timer x has a PWM (pulse Width Modulation) function. In PWM mode, the Tx/PWMx output pin outputs up to 16-bit resolution PWM output. This pin should be configured as a PWM output by set TX_PE to ‘1’. The PWM output mode is determined by the PWMxPRH, PWMxPRL, PWMxDRH and PWMxDRL. And you should configure PWMxE bit to “1” in TxCR register. PWM Period and Duty same output shown in Figure 11-16.

PWM Period = [PWMxPRH, PWMxPRL] X Timer x Clock Period

PWM Duty = [PWMxDRH, PWMxDRL] X Timer x Clock Period

Table 11-8 PWM Frequency vs. Resolution at 16 MHz

Resolution	Frequency		
	TxCK[2:0]=000 (62.5ns)	TxCK[2:0]=001(250ns)	TxCK[2:0]=010(500ns)
16-bit	244.141Hz	61.035Hz	30.518Hz
15-bit	488.28Hz	122.070Hz	60.035Hz
10-bit	15.625KHz	3.906KHz	1.953KHz
9-bit	31.250KHz	7.813KHz	3.906KHz
8-bit	62.500KHz	15.625KHz	7.813KHz

The POL bit of TxCR register decides the polarity of duty cycle.
The PWM function is not supported in these cases : When the duty value is “00H” or set same to the period value.

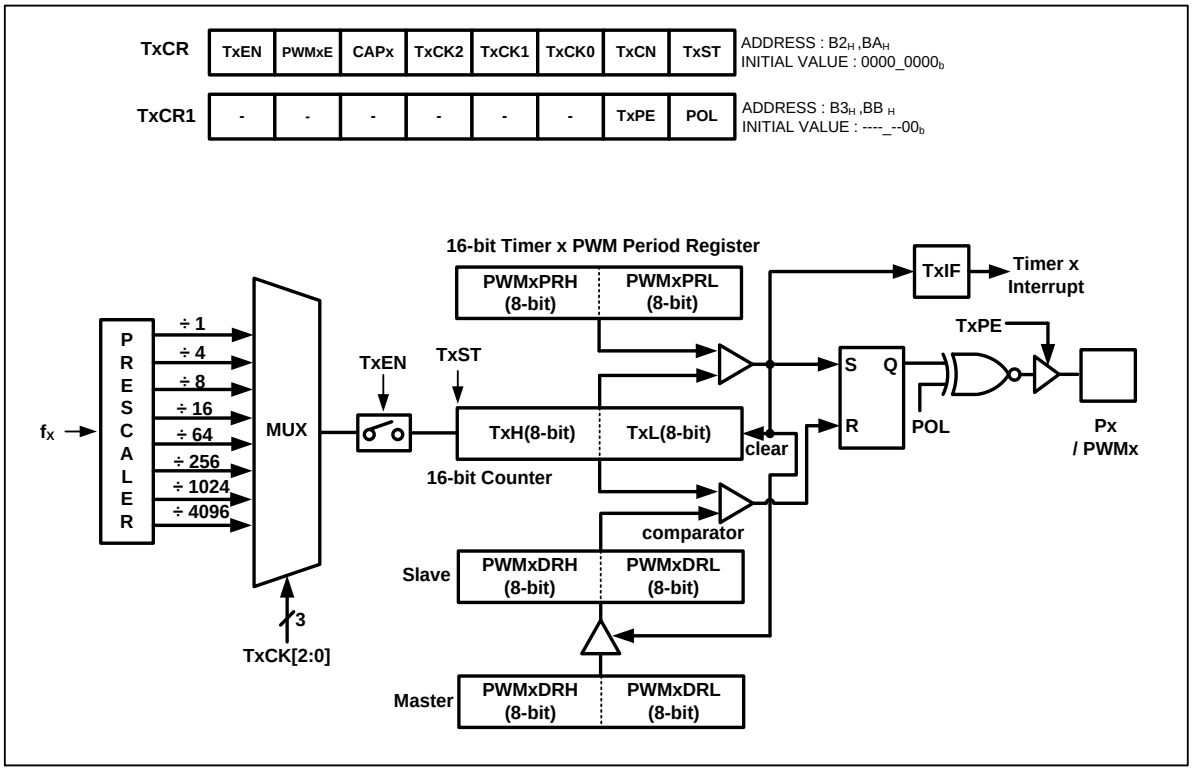


Figure 11-14 PWM Mode

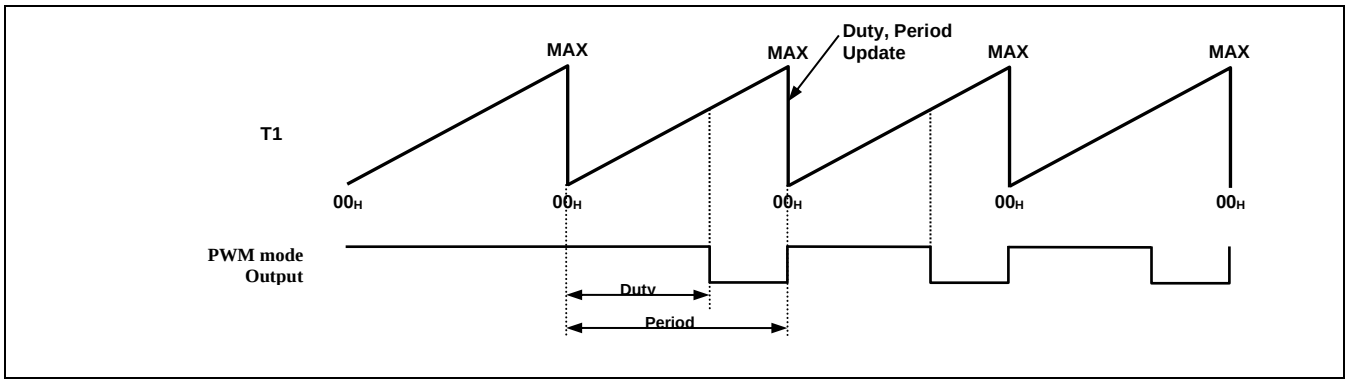


Figure 11-15 Example of PWM Output Waveform

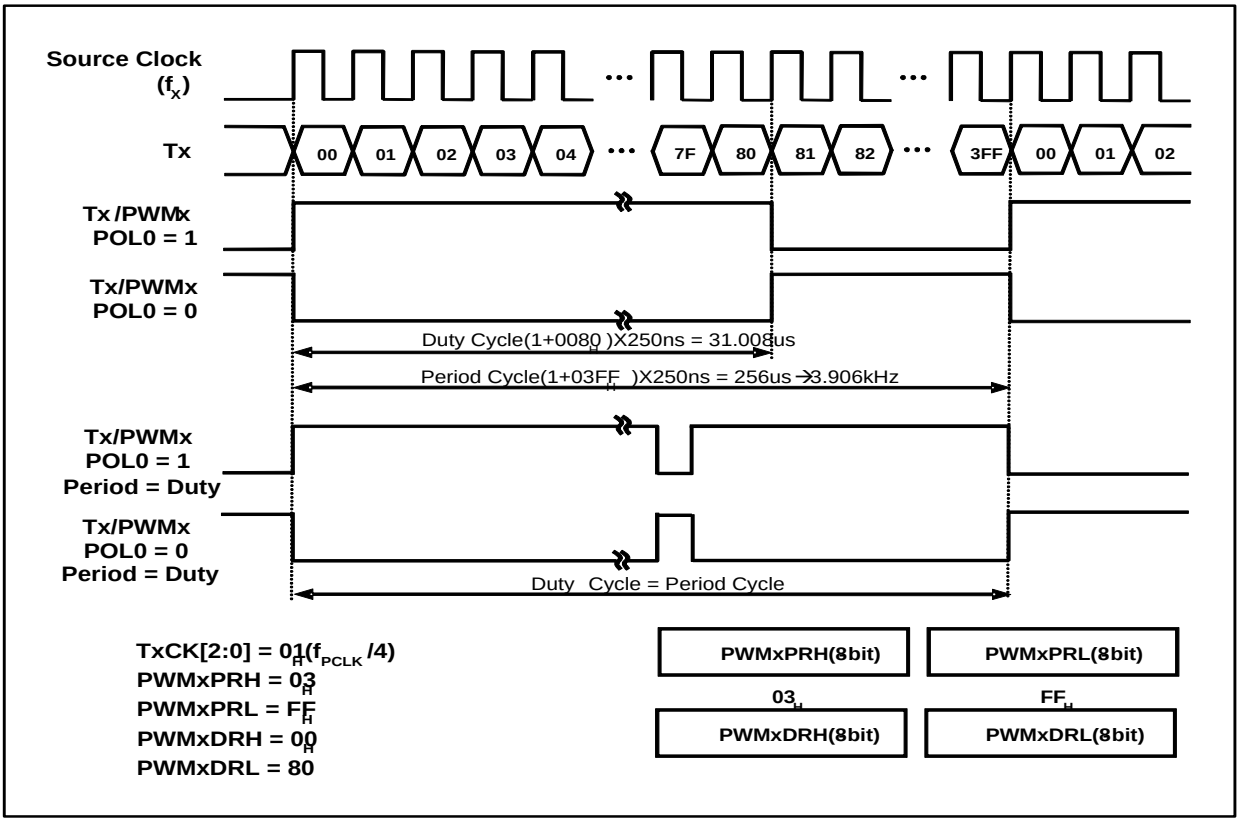


Figure 11-16 Example of PWM (16MHz)

11.7.1.5 Register Map

Table 11-9 Register Map

Name	Address	Dir	Default	Description
T0CR	$B2_H$	R/W	00_H	Timer 0 Mode Control Register
T0CR1	$B3_H$	R/W	00_H	Timer 0 Mode Control Register 1
T0L	$B4_H$	R	00_H	Timer 0 Low Register
PWM0DRL	$B4_H$	R/W	00_H	PWM 0 Duty Register Low
CDR0L	$B4_H$	R	00_H	Timer 0 Capture Data Register Low
T0H	$B5_H$	R	00_H	Timer 0 Register High
PWM0DRH	$B5_H$	R/W	00_H	PWM 0 Duty Register High
CDR0H	$B5_H$	R	00_H	Timer 0 Capture Data Register High
T0DRL	$B6_H$	W	FF_H	Timer 0 Data Register Low
PWM0PRL	$B6_H$	W	FF_H	PWM 0 Period Register Low
T0DRH	$B7_H$	W	FF_H	Timer 0 Data Register High
PWM0PRH	$B7_H$	W	FF_H	PWM 0 Period Register High
T1CR	BA_H	R/W	00_H	Timer 1 Mode Control Register
T1CR1	BB_H	R/W	00_H	Timer 1 Mode Control Register 1
T1L	BC_H	R	00_H	Timer 1 Register Low
PWM1DRL	BC_H	R/W	00_H	PWM 1 Duty Register Low
CDR1L	BC_H	R	00_H	Timer 1 Capture Data Register Low
T1H	BD_H	R	00_H	Timer 1 Register High
PWM1DRH	BD_H	R/W	00_H	PWM 1 Duty Register High
CDR1H	BD_H	R	00_H	Timer 1 Capture Data Register High
T1DRL	BE_H	W	FF_H	Timer 1 Data Register Low
PWM1PRL	BE_H	W	FF_H	PWM 1 Period Register Low
T1DRH	BF_H	W	FF_H	Timer 1 Data Register High
PWM1PRH	BF_H	W	FF_H	PWM 1 Period Register High
T2CR	$C2_H$	R/W	00_H	Timer 2 Mode Control Register
T2CR1	$C3_H$	R/W	00_H	Timer 2 Mode Control Register 1
T2L	$C4_H$	R	00_H	Timer 2 Low Register
PWM2DRL	$C4_H$	R/W	00_H	PWM 2 Duty Register Low
CDR2L	$C4_H$	R	00_H	Timer 2 Capture Data Register Low
T2H	$C5_H$	R	00_H	Timer 2 Register High
PWM2DRH	$C5_H$	R/W	00_H	PWM 2 Duty Register High
CDR2H	$C5_H$	R	00_H	Timer 2 Capture Data Register High
T2DRL	$C6_H$	W	FF_H	Timer 2 Data Register Low
PWM2PRL	$C6_H$	W	FF_H	PWM 2 Period Register Low
T2DRH	$C7_H$	W	FF_H	Timer 2 Data Register High
PWM2PRH	$C7_H$	W	FF_H	PWM 2 Period Register High

11.7.1.6 Timer/Counter x Register Description

The Timer 0~x Register consists of Timer 0~x Mode Control Register (T0CR), (T1CR), Timer 0~x Mode Control Register 1 (T0CR1), (T1CR1), Timer 0~x Low Register (T0L), (T1L), Timer 0~x Data Register Low (T0DRL), (T1DRL),

Timer 0~x High Register (T0H), (T1H) Timer 0~x Data Register High (T0DRH), (T1DRH), Timer 0~x Capture Data Low Register (CDR0L), (CDR1L), Timer 0~x Capture Data High Register (CDR0H), (CDR1H), PWM0~x Low Duty Register (PWM0DRL), (PWM1DRL), PWM0~x High Duty Register (PWM0DRH), (PWM1DRH), PWM0~x Low Period Register (PWM0PRL), (PWM1PRL), PWM0~x High Period Register (PWM0PRH), (PWM1PRH).

11.7.1.7 Register Description for Timer/Counter x

T0CR, T1CR, T2CR (Timer 0~2 Mode Control Register): B2H, BAH, C2H

7	6	5	4	3	2	1	0
TxEN	PWMxE	CAPx	TxCK2	TxCK1	TxCK0	TxCN	TxST
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00_H

TxEN Control Timer X
- 0 0
- 1 Timer X enable

PWMxE Control PWM X enable
- 0 PWM X disable
- 1 PWM X enable

CAPx Control Timer X capture mode.
- 0 Timer mode
- 1 Capture mode

TxCK[2:0] Select clock source of Timer X. Fx is the frequency of main system

TxCK2	TxCK1	TxCK0	description
0	0	0	f_X
0	0	1	f_X /4
0	1	0	f_X /8
0	1	1	f_X /16
1	0	0	f_X /64
1	0	1	f_X /256
1	1	0	f_X /1024
1	1	1	f_X /2048

TxCN Control Timer X Count pause/continue.
- 0 Temporary count stop
- 1 Continue count

TxST Control Timer x start/stop
- 0 Counter stop
- 1 Clear counter and start

Note) set TxST bit after write to Tx, PWMx, CDRx registers.

T0CR1, T1CR1, T2CR1 (Timer 0~2 Mode Control Register 1) : B3H, , BBH, C3H

7	6	5	4	3	2	1	0
-	-	-	-	-	-	Tx_PE	POL
-	-	-	-	-	-	R/W	R/W

Initial value : 00_H

T0_PE Control Timer X Output port

0 Timer X Output disable
1 Timer X Output enable

POL Configure PWM polarity
0 Negative (Duty Match: Set)
1 Positive (Duty Match: Clear)

T0L, T1L, T2L (Timer 0~2 Register Low, Read Case) : B4H, BCH, C4H

7	6	5	4	3	2	1	0
TxL7	TxL6	TxL5	TxL4	TxL3	TxL2	TxL1	TxL0
R	R	R	R	R	R	R	R

Initial value : 00_H

TxL[7:0] TxL Counter Period Low data.

CDR0L, CDR1L, CDR2L (Capture 0~2 Data Register Low, Read Case) : B4H, BCH, C4H

7	6	5	4	3	2	1	0
CDRxL07	CDRxL06	CDRxL05	CDRxL04	CDRxL03	CDRxL02	CDRxL01	CDRxL00
R	R	R	R	R	R	R	R

Initial value : 00_H

CDRxL[7:0] Tx Capture Low data.

PWM0DRL, PWM1DRL (PWM 0~2 Duty Register Low, Write Case) : B4H, BCH, C4H

7	6	5	4	3	2	1	0
PWMxLD7	PWMxLD6	PWMxLD5	PWMxLD4	PWMxLD3	PWMxLD2	PWMxLD1	PWMxLD0
W	W	W	W	W	W	W	W

Initial value : 00_H

PWMxLD[7:0] Tx PWM Duty Low data
Note) Writing is effective only when PWMxE = 1 and T0ST = 0

T0H, T1H, T2H (Timer 0~2 Register High, Read Case) : B5H, BDH, C5H

7	6	5	4	3	2	1	0
TxH7	TxH6	TxH5	TxH4	TxH3	TxH2	TxH1	TxH0
R	R	R	R	R	R	R	R

Initial value : 00_H

TxH[7:0] TxH Counter Period High data.

CDR0H, CDR1H, CDR2H (Capture 0~2 Data High Register, Read Case) : B5H, BDH, C5H

7	6	5	4	3	2	1	0
CDRxH07	CDRxH06	CDRxH05	CDRxH04	CDRxH03	CDRxH02	CDRxH01	CDRxH00
R	R	R	R	R	R	R	R

Initial value : 00_H

CDRxH[7:0] Tx Capture High data

PWM0DRH, PWM1DRH, PWM2DRH (PWM 0~2 Duty Register High,Write Case) : B5H, BDH, C5H

7	6	5	4	3	2	1	0
PWMxHD7	PWMxHD6	PWMxHD5	PWMxHD4	PWMxHD3	PWMxHD2	PWMxHD1	PWMxHD0
W	W	W	W	W	W	W	W

Initial value : 00_H

PWMxHD[7:0] Tx PWM Duty High data
Note) Writing is effective only when PWMxE = 1 and T0ST = 0

T0DRL, T1DRL, T2DRL (Timer 0~2 Data Register Low, Write Case) : B6H, BEH, C6H

7	6	5	4	3	2	1	0
TxLD7	TxLD6	TxLD5	TxLD4	TxLD3	TxLD2	TxLD1	TxLD0
W	W	W	W	W	W	W	W

Initial value : FF_H

TxLD[7:0] TxL Compare Low data

Note) Be sure to clear PWMxE before loading this register.

PWM0PRL, PWM1PRL, PWM2PRL (PWM 0~2 Period Register Low,Write Case) : B6H, BEH, C6H

7	6	5	4	3	2	1	0
PWM0LP7	PWM0LP6	PWM0LP5	PWM0LP4	PWM0LP3	PWM0LP2	PWM0LP1	PWM0LP0
W	W	W	W	W	W	W	W

Initial value : FF_H

PWM0LP[7:0] T0 PWM Duty Low data
Note) Writing is effective only when PWM0E = 1 and T0ST = 0

T0DRH, T1DRH, T2DRH (Timer 0~2 Data Register High, Write Case) : B7H, BFH, C7H

7	6	5	4	3	2	1	0
TxHD7	TxHD6	TxHD5	TxHD4	TxHD3	TxHD2	TxHD1	TxHD0
W	W	W	W	W	W	W	W

Initial value : FF_H

TxHD[7:0] TxH Compare High data

Note) Be sure to clear PWMxE before loading this register.

PWM0PRH, PWM1PRH, PWM2PRH (PWM 0~2 Period Register High,Write Case) : B7H,BFH,C7H

7	6	5	4	3	2	1	0
PWMxHP7	PWMxHP6	PWMxHP5	PWMxHP4	PWMxHP3	PWMxHP2	PWMxHP1	PWMxHP0
R/W	W	W	W	W	W	W	W

Initial value : FF_H

PWMxHP[7:0] Tx PWM Duty High data
Note) Writing is effective only when PWMxE = 1 and T0ST = 0

11.7.2 Timer Interrupt Status Register (TMISR)

11.7.2.1 Register Description for TMISR

TMISR (Timer Interrupt Status Register) : D5H

7	6	5	4	3	2	1	0
-	-	-	-	-	TMIF2	TMIF1	TMIF0
-	-	-	-	-	R/W	R/W	R/W

Initial value : 00H

TMIF2 Timer 2 Interrupt Flag
- 0 No Timer 2 interrupt
- 1 Timer 2 interrupt occurred, write “1” to clear interrupt flag

TMIF1 Timer 1 Interrupt Flag
- 0 No Timer 1 interrupt
- 1 Timer 1 interrupt occurred, write “1” to clear interrupt flag

TMIF0 Timer 0 Interrupt Flag
- 0 No Timer 0 interrupt
- 1 Timer 0 interrupt occurred, write “1” to clear interrupt flag

Note) The Timer Interrupt Status Register contains interrupt information of each timers. Even if user disabled timer interrupt at IE2, user could check timer interrupt condition from this register.

11.8 USART

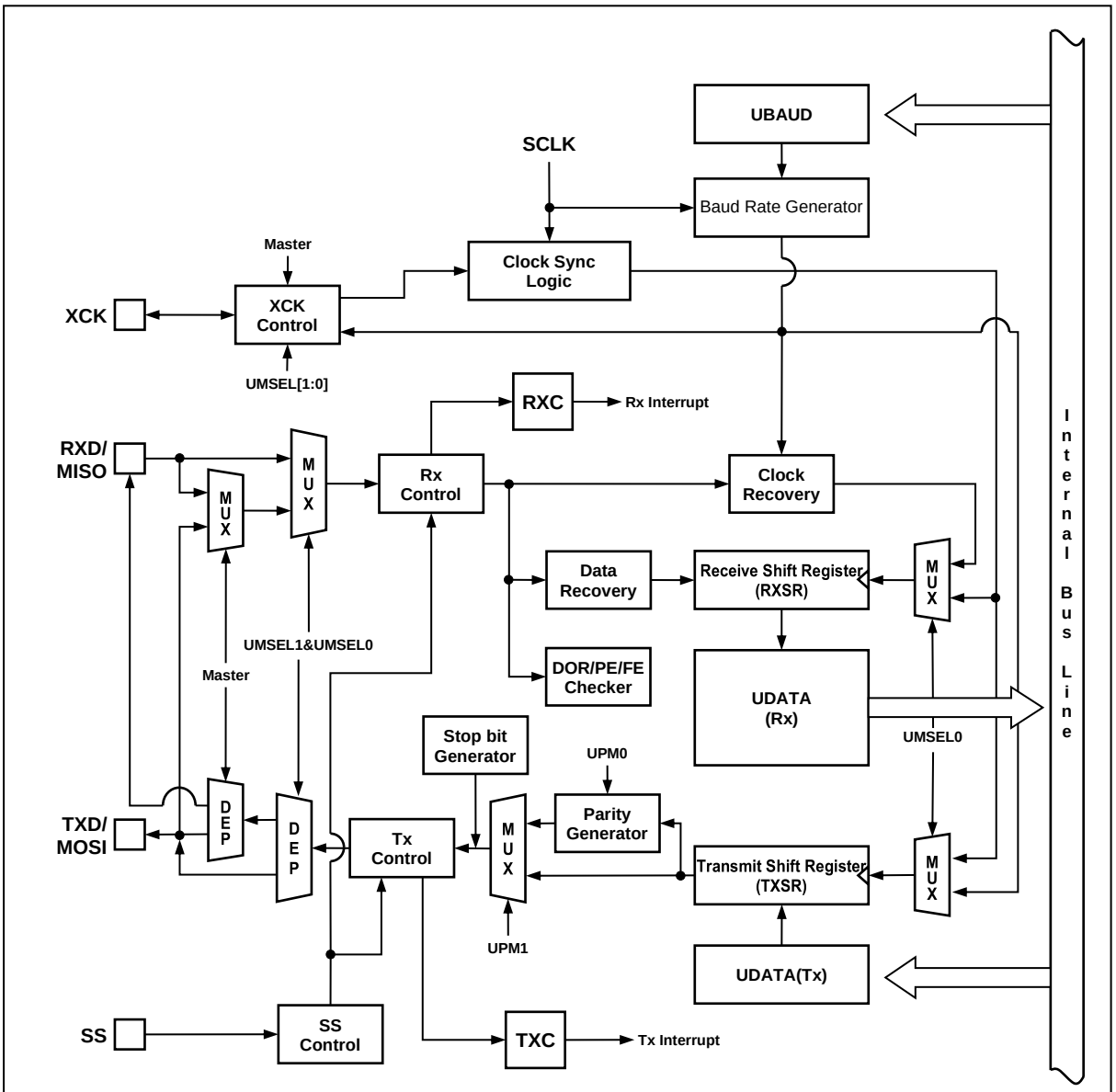
11.8.1 Overview

The Universal Synchronous and Asynchronous serial Receiver and Transmitter (USART) is a highly flexible serial communication device. The main features are listed below.

- Full Duplex Operation (Independent Serial Receive and Transmit Registers)
- Asynchronous or Synchronous Operation
- Master or Slave Clocked Synchronous and SPI Operation
- Supports all four SPI Modes of Operation (Mode 0, 1, 2, 3)
- LSB First or MSB First Data Transfer @SPI mode
- High Resolution 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
- Data OverRun Detection
- Framing Error Detection
- Digital Low Pass Filter
- Three Separate Interrupts on TX Complete, TX Data Register Empty and RX Complete
- Double Speed Asynchronous Communication Mode

USART has three main parts of Clock Generator, Transmitter and Receiver. The Clock Generation logic consists of synchronization logic for external clock input used by synchronous or SPI slave operation, and the baud rate generator for asynchronous or master (synchronous or SPI) operation. The Transmitter consists of a single write buffer, a serial shift register, parity generator and control logic for handling different serial frame formats. The write buffer allows a continuous transfer of data without any delay 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 to the recovery unit, the Receiver includes a parity checker, a shift register, a two level receive FIFO(UDATA) and control logic. The Receiver supports the same frame formats as the Transmitter and can detect Frame Error, Data OverRun and Parity Errors.

11.8.2 Block Diagram



UCTRL1	UMSEL1	UMSEL0	UPM1	UPM0	USIZE2 UDORD	USIZE2 UCPHA	USIZE0	UCPOL
UCTRL2	UDRIE	TXCIE	RXCIE	WAKEIE	TXE	RXE	USARTEN	U2X
UCTRL3	MASTER	LOOPS	DISXCK	SPISS	-	USBS	TX8	RX8
USTAT	UDRE	TXC	RXC	WAKE	SOFTRST	DOR	FE	PE

ADDRESS : FA_H
INITIAL VALUE : 0000_0000_B

ADDRESS : FB_H
INITIAL VALUE : 0000_0000_B

ADDRESS : FC_H
INITIAL VALUE : 0000_-000_B

ADDRESS : FD_H
INITIAL VALUE : 1000_0000_B

Figure 11-17 USART Block Diagram

11.8.3 Clock Generation

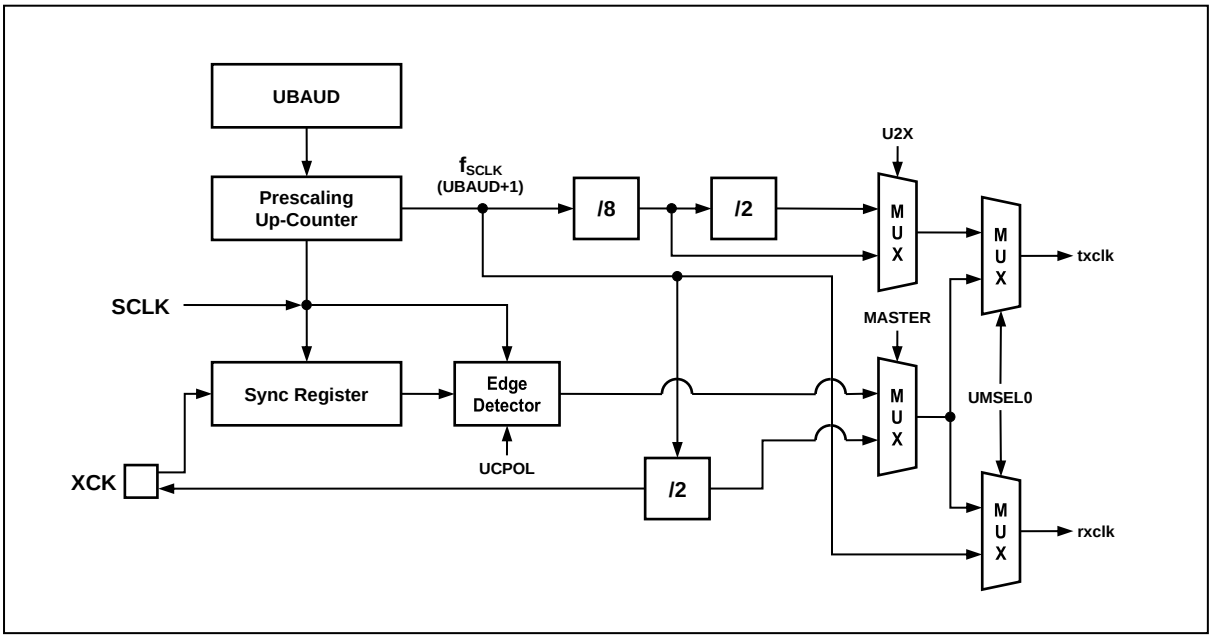


Figure 11-18 Clock Generation Block Diagram

The Clock generation logic generates the base clock for the Transmitter and Receiver. The USART supports four modes of clock operation and those are Normal Asynchronous, Double Speed Asynchronous, Master Synchronous and Slave Synchronous. The clock generation scheme for Master SPI and Slave SPI mode is the same as Master Synchronous and Slave Synchronous operation mode. The UMSELn bit in UCTRL1 register selects between asynchronous and synchronous operation. Asynchronous Double Speed mode is controlled by the U2X bit in the UCTRL2 register. The MASTER bit in UCTRL2 register controls whether the clock source is internal (Master mode, output port) or external (Slave mode, input port). The XCK pin is only active when the USART operates in Synchronous or SPI mode.

Table below contains equations for calculating the baud rate (in bps).

Table 11-10 Equations for Calculating Baud Rate Register Setting

Operating Mode	Equation for Calculating Baud Rate
Asynchronous Normal Mode (U2X=0)	$\text{Baud Rate} = \frac{\text{fSCLK}}{16(\text{UBAUDx} + 1)}$
Asynchronous Double Speed Mode (U2X=1)	$\text{Baud Rate} = \frac{\text{fSCLK}}{8(\text{UBAUDx} + 1)}$
Synchronous or SPI Master Mode	$\text{Baud Rate} = \frac{\text{fSCLK}}{2(\text{UBAUDx} + 1)}$

11.8.4 External Clock (XCK)

External clocking is used by the synchronous or SPI slave modes of operation.

External clock input from the XCK pin is sampled by a synchronization logic to remove meta-stability. The output from the synchronization logic must then pass through an edge detector before it can be used by the Transmitter and Receiver. This process introduces a two CPU clock period delay and therefore the maximum frequency of the external XCK pin is limited by the following equation.

$$\mathrm{fXCK} = \frac{\mathrm{fSCLK}}{4}$$

where fXCK is the frequency of XCK and fSCLK is the frequency of main system clock (SCLK).

11.8.5 Synchronous Mode Operation

When synchronous or spi mode is used, the XCK pin will be used as either clock input (slave) or clock output (master). The dependency between the clock edges and data sampling or data change is the same. The basic principle is that data input on RXD (MISO in spi mode) pin is sampled at the opposite XCK clock edge of the edge in the data output on TXD (MOSI in spi mode) pin is changed.

The UCPOL bit in UCTRL1 register selects which XCK clock edge is used for data sampling and which is used for data change. As shown in the figure below, when UCPOL is zero the data will be changed at rising XCK edge and sampled at falling XCK edge.

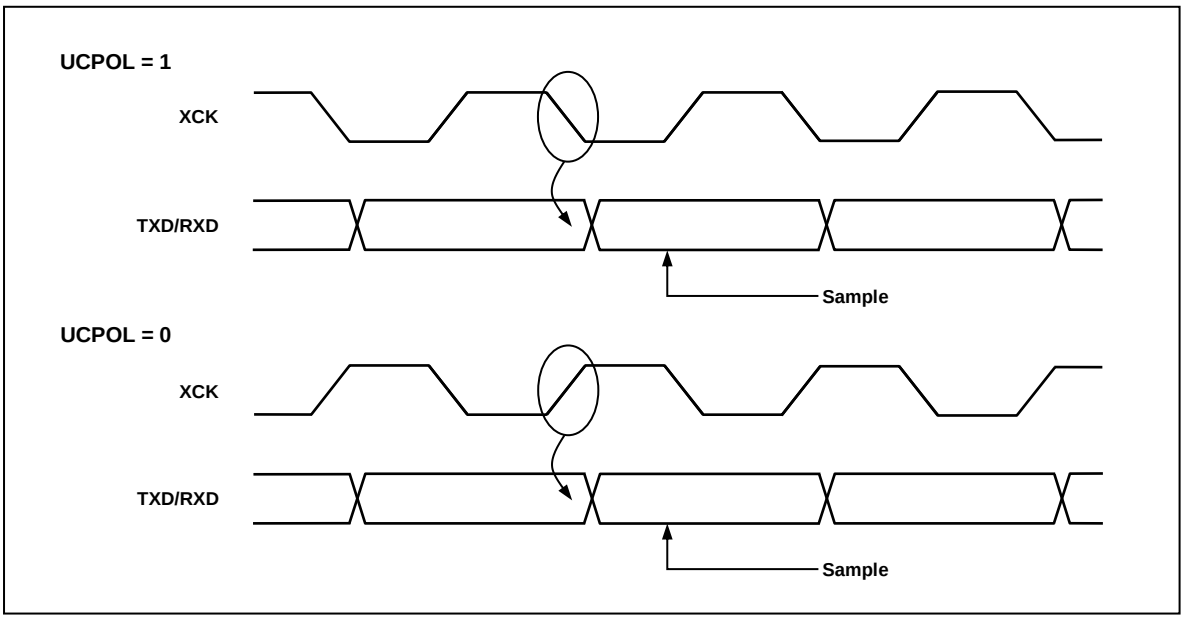


Figure 11-19 Synchronous Mode XCKn Timing

11.8.6 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 checking.

The USART supports all 30 combinations of the following 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 a total of nine, are succeeding, ending with the most significant bit (MSB). If enabled the parity bit is inserted after the data bits, before the stop bits. 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 means high state of data pin. The next figure shows the possible combinations of the frame formats. Bits inside brackets are optional.

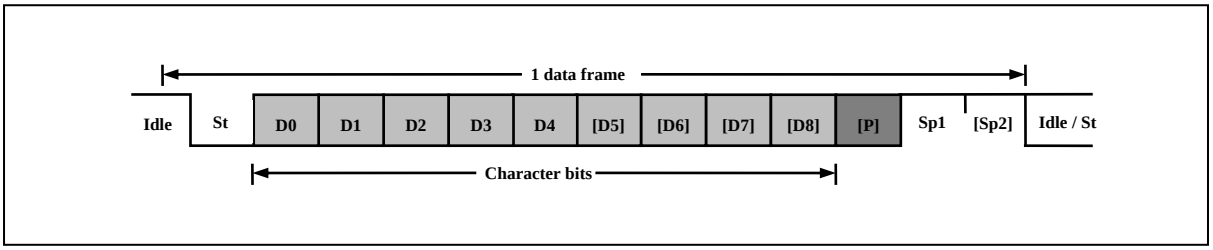


Figure 11-20 Frame Format

1 data frame consists of the following bits

- Idle No communication on communication line (TxD/RxD)
- St Start bit (Low)
- Dn Data bits (0~8)
- Parity bit ---------- Even parity, Odd parity, No parity
- Stop bit(s) ---------- 1 bit or 2 bits

The frame format used by the USART is set by the USIZE[2:0], UPM[1:0] and USBS bits in UCTRL1 register. The Transmitter and Receiver use the same setting.

11.8.7 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 MSB and first stop bit of a serial frame.

$P_{even} = D_{n-1} \wedge \ldots \wedge D_3 \wedge D_2 \wedge D_1 \wedge D_0 \wedge 0$

$P_{odd} = D_{n-1} \wedge \ldots \wedge D_3 \wedge D_2 \wedge D_1 \wedge D_0 \wedge 1$

P_{even} : Parity bit using even parity

P_{odd} : Parity bit using odd parity

D_n : Data bit n of the character

11.8.8 USART Transmitter

The USART Transmitter is enabled by setting the TXE bit in UCTRL2 register. When the Transmitter is enabled, the normal port operation of the TXD pin is overridden by the serial output pin of USART. The baud-rate, operation mode and frame format must be setup once before doing any transmissions. If synchronous or spi operation is used, the clock on the XCK pin will be overridden and used as transmission clock. If USART operates in spi mode, SS pin is used as SS input pin in slave mode or can be configured as SS output pin in master mode. This can be done by setting SPISS bit in UCTRL3 register.

11.8.8.1 Sending Tx data

A data transmission is initiated by loading the transmit buffer (UDATA register I/O location) with the data to be transmitted. The data written in transmit buffer 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 with new data, it will transfer one complete frame at the settings of control registers. If the 9-bit characters are used in asynchronous or synchronous operation mode (USIZE[2:0]=7), the ninth bit must be written to the TX8 bit in UCTRL3 register before loading transmit buffer (UDATA register).

11.8.8.2 Transmitter flag and interrupt

The USART Transmitter has 2 flags which indicate its state. One is USART Data Register Empty (UDRE) and the other is Transmit Complete (TXC). Both flags can be interrupt sources.

UDRE flag indicates whether the transmit buffer is ready to be loaded with new data. This bit is set when the transmit buffer is empty and cleared when the transmit buffer contains data to be transmitted that has not yet been moved into the shift register. And also this flag can be cleared by writing ‘0’ to this bit position. Writing ‘1’ to this bit position is prevented.

When the Data Register Empty Interrupt Enable (UDRIE) bit in UCTRL2 register is set and the Global Interrupt is enabled, USART Data Register Empty Interrupt is generated while UDRE flag is set.

The Transmit Complete (TXC) flag bit is set when the entire frame in the transmit shift register has been shifted out and there are no more data in the transmit buffer. The TXC flag is automatically cleared when the Transmit Complete Interrupt service routine is executed, or it can be cleared by writing ‘0’ to TXC bit in USTAT register.

When the Transmit Complete Interrupt Enable (TXCIE) bit in UCTRL2 register is set and the Global Interrupt is enabled, USART Transmit Complete Interrupt is generated while TXC flag is set.

11.8.8.3 Parity Generator

The Parity Generator calculates the parity bit for the sending serial frame data. When parity bit is enabled (UPM[1]=1), the transmitter control logic inserts the parity bit between the MSB and the first stop bit of the sending frame.

11.8.8.4 Disabling Transmitter

Disabling the Transmitter by clearing the TXE bit will not become effective until ongoing transmission is completed. When the Transmitter is disabled, the TXD pin is used as normal General Purpose I/O (GPIO) or primary function pin.

11.8.9 USART Receiver

The USART Receiver is enabled by setting the RXE bit in the UCTRL1 register. When the Receiver is enabled, the normal pin operation of the RXD pin is overridden by the USART as the serial input pin of the Receiver. The baud-rate, mode of operation and frame format must be set before serial reception. If synchronous or spi operation is used, the clock on the XCK pin will be used as transfer clock. If USART operates in spi mode, SS pin is used as SS input pin in slave mode or can be configured as SS output pin in master mode. This can be done by setting SPISS bit in UCTRL3 register.

11.8.9.1 Receiving Rx data

When USART is in synchronous or asynchronous operation mode, the Receiver starts data reception when it detects a valid start bit (LOW) on RXD pin. Each bit after start bit is sampled at pre-defined baud-rate (asynchronous) or sampling edge of XCK (synchronous), and shifted into the receive shift register until the first stop bit of a frame is received. Even if there's 2nd stop bit in the frame, the 2nd stop bit is ignored by the Receiver. That is, receiving the first stop bit means that a complete serial frame is present in the receiver shift register and contents of the shift register are to be moved into the receive buffer. The receive buffer is read by reading the UDATA register.

If 9-bit characters are used (USIZE[2:0] = 7) the ninth bit is stored in the RX8 bit position in the UCTRL3 register. The 9th bit must be read from the RX8 bit before reading the low 8 bits from the UDATA register. Likewise, the error flags FE, DOR, PE must be read before reading the data from UDATA register. This is because the error flags are stored in the same FIFO position of the receive buffer.

11.8.9.2 Receiver flag and interrupt

The USART Receiver has one flag that indicates the Receiver state.

The Receive Complete (RXC) flag indicates whether there are unread data present in the receive buffer. This flag is set when there are unread data in the receive buffer and cleared when the receive buffer is empty. If the Receiver is disabled (RXE=0), the receiver buffer is flushed and the RXC flag is cleared.

When the Receive Complete Interrupt Enable (RXCIE) bit in the UCTRL2 register is set and Global Interrupt is enabled, the USART Receiver Complete Interrupt is generated while RXC flag is set.

The USART Receiver has three error flags which are Frame Error (FE), Data OverRun (DOR) and Parity Error (PE). These error flags can be read from the USTAT register. As data received are stored in the 2-level receive buffer, these error flags are also stored in the same position of receive buffer. So, before reading received data from UDATA register, read the USTAT register first which contains error flags.

The Frame Error (FE) flag indicates the state of the first stop bit. The FE flag is zero when the stop bit was correctly detected as one, and the FE flag is one when the stop bit was incorrect, ie detected as zero. This flag can be used for detecting out-of-sync conditions between data frames.

The Data OverRun (DOR) flag indicates data loss due to a receive buffer full condition. A DOR occurs when the receive buffer is full, and another new data is present in the receive shift register which are to be stored into the receive buffer. After the DOR flag is set, all the incoming data are lost. To prevent data loss or clear this flag, read the receive buffer.

The Parity Error (PE) flag indicates that the frame in the receive buffer had a Parity Error when received. If Parity Check function is not enabled (UPM[1]=0), the PE bit is always read zero.

Note) The error flags related to receive operation are not used when USART is in SPI mode.

11.8.9.3 Parity Checker

If Parity Bit is enabled (UPM[1]=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.

11.8.9.4 Disabling Receiver

In contrast to Transmitter, disabling the Receiver by clearing RXE bit makes the Receiver inactive immediately. When the Receiver is disabled the Receiver flushes the receive buffer and the remaining data in the buffer is all reset. The RXD pin is not overridden the function of USART, so RXD pin becomes normal GPIO or primary function pin.

11.8.9.5 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 RXD pin.

The Data recovery logic samples and low pass filters the incoming bits, and this removes the noise of RXD pin.

The next figure illustrates the sampling process of the start bit of an incoming frame. The sampling rate is 16 times the baud-rate for normal mode, and 8 times the baud rate for Double Speed mode (U2X=1). The horizontal arrows show the synchronization variation due to the asynchronous sampling process. Note that larger time variation is shown when using the Double Speed mode.

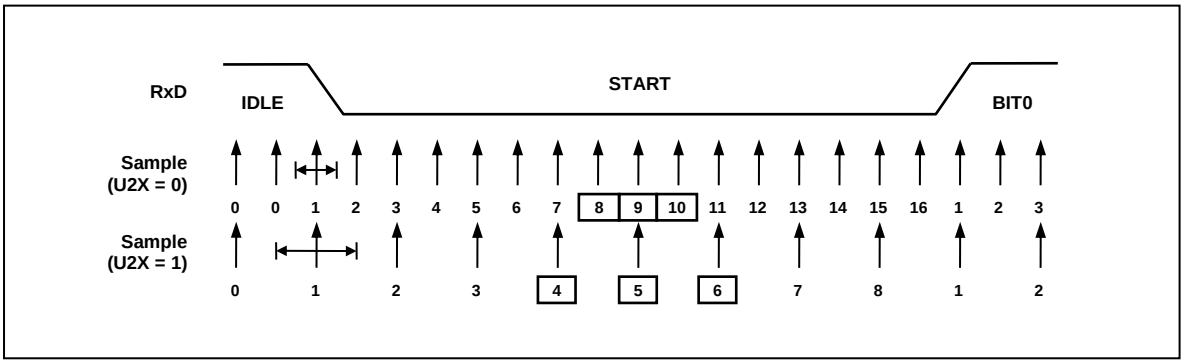


Figure 11-21 Start Bit Sampling

When the Receiver is enabled (RXE=1), the clock recovery logic tries to find a high to low transition on the RXD line, the start bit condition. After detecting high to low transition on RXD line, the clock recovery logic uses samples 8,9, and

10 for Normal mode, and samples 4, 5, and 6 for Double Speed mode to decide if a 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. And 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 similar to the clock recovery process. The data recovery logic samples 16 times for each incoming bits for Normal mode and 8 times for Double Speed mode. And uses sample 8, 9, and 10 to decide data value for Normal mode, samples 4, 5, and 6 for Double Speed mode. If more than 2 samples have low levels, the received bit is considered to a logic 0 and more than 2 samples have high levels, the received bit is considered to 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 waiting to find start bit.

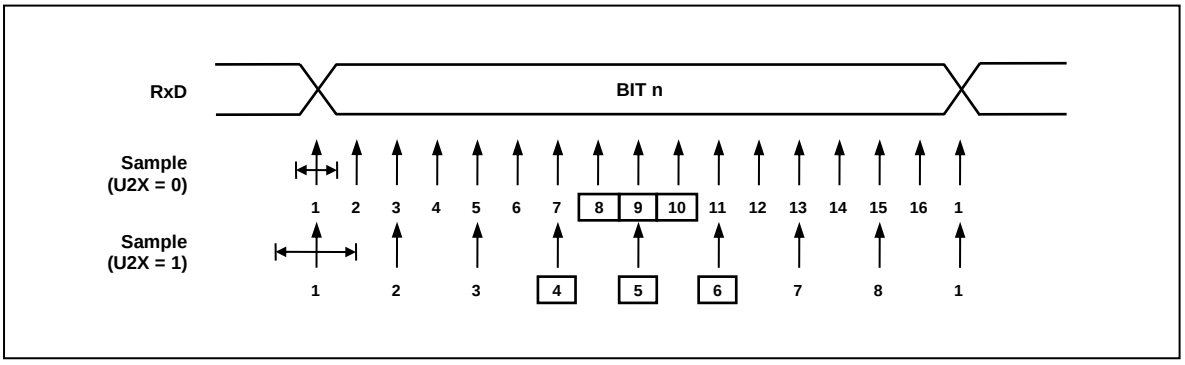


Figure 11-22 Sampling of Data and Parity Bit

The process for detecting stop bit is like clock and data recovery process. That is, if 2 or more samples of 3 center values have high level, correct stop bit is detected, else a Frame Error flag is set. After deciding first stop bit whether a valid stop bit is received or not, the Receiver goes idle state and monitors the RXD line to check a valid high to low transition is detected (start bit detection).

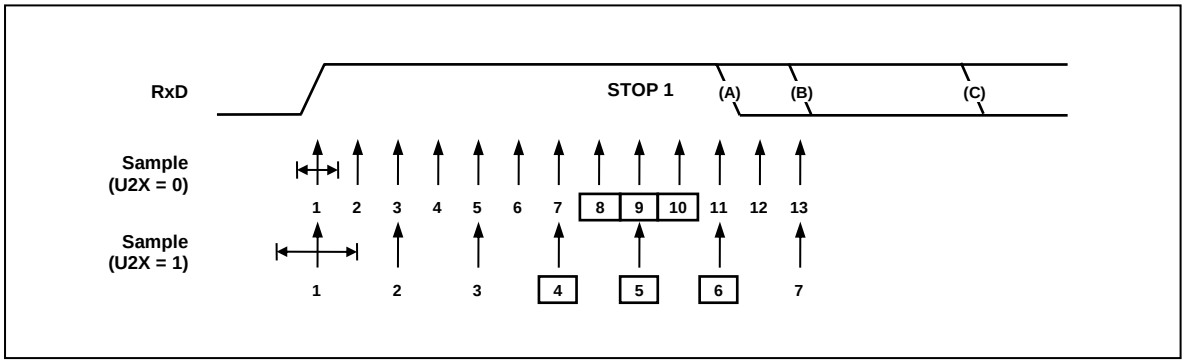


Figure 11-23 Stop Bit Sampling and Next Start Bit Sampling

11.8.10 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
- Master or Slave operation
- Supports all four SPI modes of operation (mode0, 1, 2, and 3)
- Selectable LSB first or MSB first data transfer
- Double buffered transmit and receive
- Programmable transmit bit rate

When SPI mode is enabled (UMSEL[1:0]=3), the Slave Select (SS) pin becomes active low input in slave mode operation, or can be output in master mode operation if SPISS bit is set.

Note that during SPI mode of operation, the pin RXD is renamed as MISO and TXD is renamed as MOSI for compatibility to other SPI devices.

11.8.10.1 SPI Clock formats and timing

To accommodate a wide variety of synchronous serial peripherals from different manufacturers, the USART has a clock polarity bit (UCPOL) and a clock phase control bit (UCPHA) to select one of four clock formats for data transfers. UCPOL selectively insert an inverter in series with the clock. UCPHA chooses between two different clock phase relationships between the clock and data. Note that UCPHA and UCPOL bits in UCTRL1 register have different meanings according to the UMSEL[1:0] bits which decides the operating mode of USART.

Table below shows four combinations of UCPOL and UCPHA for SPI mode 0, 1, 2, and 3.

Table 11-11 CPOL Funtionality

SPI Mode	UCPOL	UCPHA	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)

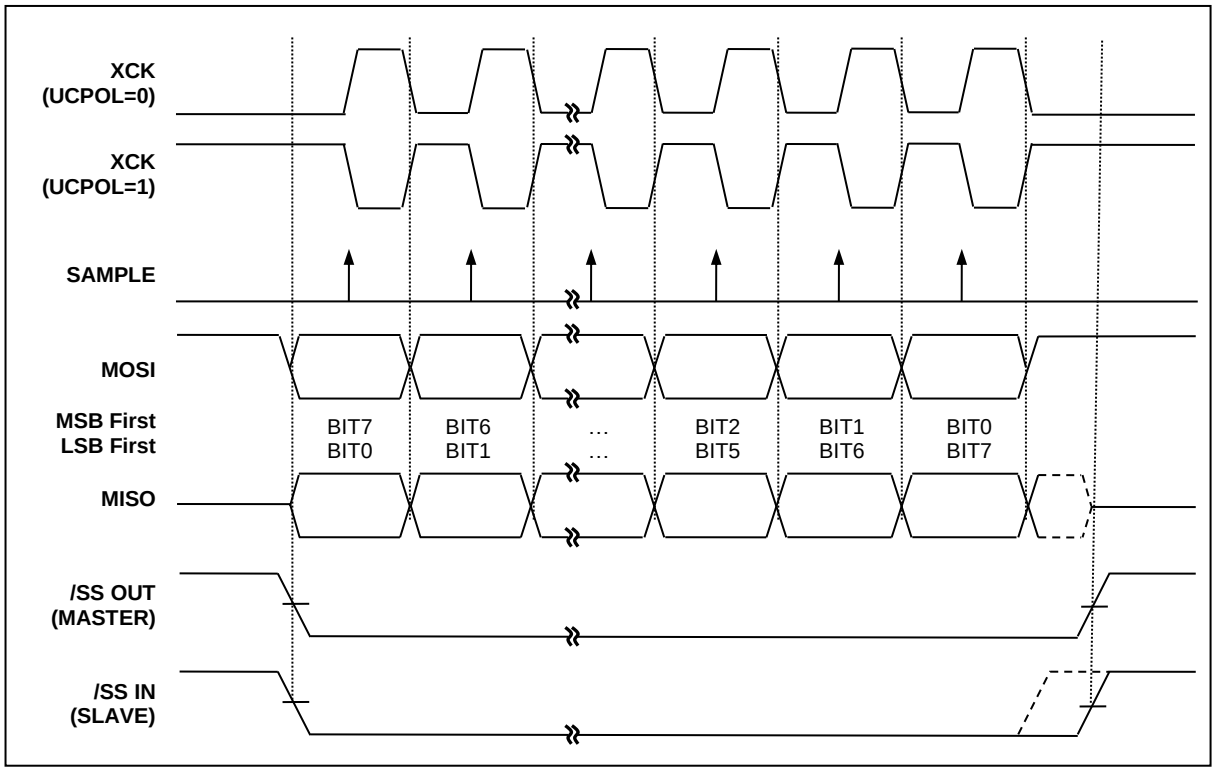


Figure 11-24 SPI Clock Formats when UCPHA=0

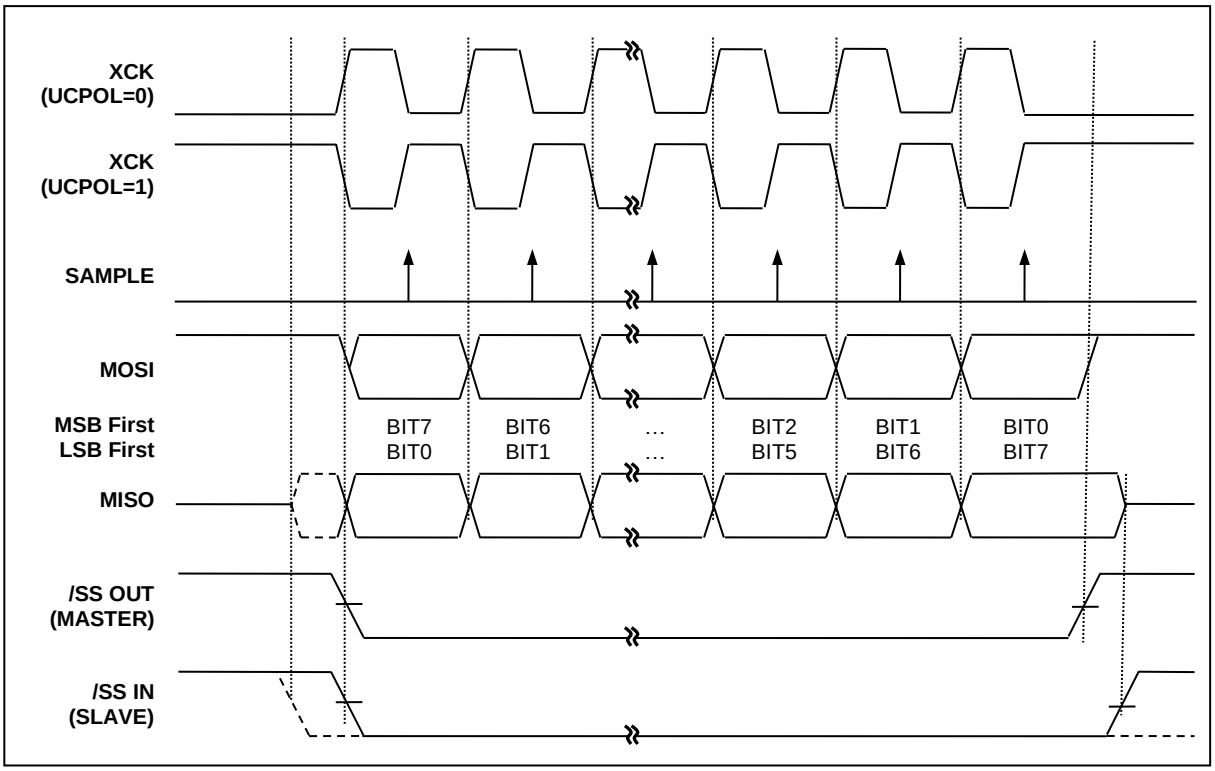


Figure 11-25 SPI Clock Formats when UCPHA=1

When UCPHA=0, the slave begins to drive its MISO output with the first data bit value when SS goes to active low. The first XCK edge causes both the master and the slave to sample the data bit value on their MISO and MOSI inputs, respectively. At the second XCK edge, the USART shifts the second data bit value out to the MOSI and MISO outputs of the master and slave, respectively. Unlike the case of UCPHA=1, when UCPHA=0, the slave's SS input must go to its inactive high level between transfers. This is because the slave can prepare the first data bit when it detects falling edge of SS input.

When UCPHA=1, the slave begins to drive its MISO output when SS goes active low, but the data is not defined until the first XCK edge. The first XCK edge shifts the first bit of data from the shifter onto the MOSI output of the master and the MISO output of the slave. The next XCK edge causes both the master and slave to sample the data bit value on their MISO and MOSI inputs, respectively. At the third XCK edge, the USART shifts the second data bit value out to the MOSI and MISO output of the master and slave respectively. When UCPHA=1, the slave's SS input is not required to go to its inactive high level between transfers.

Because the SPI logic reuses the USART resources, SPI mode of operation is similar to that of synchronous or asynchronous operation. An SPI transfer is initiated by checking for the USART Data Register Empty flag (UDRE=1) and then writing a byte of data to the UDATA Register. In master mode of operation, even if transmission is not enabled (TXE=0), writing data to the UDATA register is necessary because the clock XCK is generated from transmitter block.

11.8.11 Register Map

Table 11-12 Register Map

Name	Address	Dir	Default	Description
UCTRL01	CAH	R/W	00H	USART0 Control 1 Register
UCTRL02	CBH	R/W	00H	USART0 Control 2 Register
UCTRL03	CCH	R/W	00H	USART0 Control 3 Register
USTAT0	CDH	R	80H	USART0 Status Register
UBAUD0	CEH	R/W	FFH	USART0 Baud Rate Generation Register
UDATA0	CFH	R/W	FFH	USART0 Data Register
UCTRL11	FAH	R/W	00H	USART1 Control 1 Register
UCTRL12	FBH	R/W	00H	USART1 Control 2 Register
UCTRL13	FCH	R/W	00H	USART1 Control 3 Register
USTAT1	FDH	R	80H	USART1 Status Register
UBAUD1	FEH	R/W	FFH	USART1 Baud Rate Generation Register
UDATA1	FFH	R/W	FFH	USART1 Data Register

11.8.12 USART Register Description

USART module consists of USART Control 1 Register (UCTRL1), USART Control 2 Register (UCTRL2), USART Control 3 Register (UCTRL3), USART Status Register (USTAT), USART Data Register (UDATA), and USART Baud Rate Generation Register (UBAUD).

11.8.13 Register Description for USART

UCTRL01, UCTRL11 (USART Control 1 Register) : CAH, FAH

7	6	5	4	3	2	1	0
UMSEL1	UMSEL0	UPM1	UPM0	USIZE2	USIZE1 UDORD	USIZE0 UCPHA	UCPOL
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00_H

UMSEL[1:0] Selects operation mode of USART

UMSEL1	UMSEL0	Operating Mode
0	0	Asynchronous Mode (Normal Uart)
0	1	Synchronous Mode (Synchronous Uart)
1	0	Reserved
1	1	SPI Mode

UPM[1:0] Selects Parity Generation and Check methods

UPM1	UPM0	Parity mode
0	0	No Parity
0	1	Reserved
1	0	Even Parity
1	1	Odd Parity

USIZE[2:0] When in asynchronous or synchronous mode of operation, selects the length of data bits in frame.

USIZE2	USIZE1	USIZE0	Data length
0	0	0	5 bit

0	0	1	6 bit
0	1	0	7 bit
0	1	1	8 bit
1	0	0	Reserved
1	0	1	Reserved
1	1	0	Reserved
1	1	1	9 bit

UDORD This bit is in the same bit position with USIZE1. In SPI mode, when set to one the MSB of the data byte is transmitted first. When set to zero the LSB of the data byte is transmitted first.

0 LSB First

1 MSB First

UCPOL Selects polarity of XCK in synchronous or spi mode

0 TXD change @Rising Edge, RXD change @Falling Edge

1 TXD change @ Falling Edge, RXD change @ Rising Edge

UCPHA This bit is in the same bit position with USIZE0. In SPI mode, along with UCPOL bit, selects one of two clock formats for different kinds of synchronous serial peripherals. Leading edge means first XCK edge and trailing edge means 2nd or last clock edge of XCK in one XCK pulse. And Sample means detecting of incoming receive bit, Setup means preparing transmit data.

UCPOL	UCPHA	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)

UCTRL02, UCTRL12 (USART Control 2 Register) : CBH, FBH

7	6	5	4	3	2	1	0
UDRIE	TXCIE	RXCIE	WAKEIE	TXE	RXE	USARTEN	U2X
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00_H

UDRIE Interrupt enable bit for USART Data Register Empty.

0 Interrupt from UDRE is inhibited (use polling)

1 When UDRE is set, request an interrupt

TXCIE Interrupt enable bit for Transmit Complete.

0 Interrupt from TXC is inhibited (use polling)

1 When TXC is set, request an interrupt

RXCIE Interrupt enable bit for Receive Complete

0 Interrupt from RXC is inhibited (use polling)

1 When RXC is set, request an interrupt

WAKEIE Interrupt enable bit for Asynchronous Wake in STOP mode. When device is in stop mode, if RXD goes to LOW level an interrupt can be requested to wake-up system.

0 Interrupt from Wake is inhibited

1 When WAKE is set, request an interrupt

TXE Enables the transmitter unit.

0 Transmitter is disabled

1 Transmitter is enabled

RXE Enables the receiver unit.

0 Receiver is disabled

1 Receiver is enabled

USARTEN Activate USART module by supplying clock.

0 USART is disabled (clock is halted)

1 USART is enabled

U2X This bit only has effect for the asynchronous operation and selects receiver sampling rate.

0 Normal asynchronous operation

1 Double Speed asynchronous operation

UCTRL03, UCTRL13 (USART Control 3 Register) : CCH, FCH

7	6	5	4	3	2	1	0
MASTER	LOOPS	DISXCK	SPISS	-	USBS	TX8	RX8
R/W	R/W	R/W	R/W	-	R/W	R/W	R/W

Initial value : 00_H

MASTER Selects master or slave in SPI or Synchronous mode operation and controls the direction of XCK pin.

0 Slave mode operation and XCK is input pin.

1 Master mode operation and XCK is output pin

LOOPS Controls the Loop Back mode of USART, for test mode

0 Normal operation

1 Loop Back mode

DISXCK In Synchronous mode of operation, selects the waveform of XCK output.

0 XCK is free-running while USART is enabled in synchronous master mode.

1 XCK is active while any frame is on transferring.

SPISS Controls the functionality of SS pin in master SPI mode.

0 SS pin is normal GPIO or other primary function

1 SS output to other slave device

USBS Selects the length of stop bit in Asynchronous or Synchronous mode of operation.

0 1 Stop Bit

1 2 Stop Bit

TX8 The ninth bit of data frame in Asynchronous or Synchronous mode of operation. Write this bit first before loading the UDATA register.

0 MSB (9th bit) to be transmitted is ‘0’

1 MSB (9th bit) to be transmitted is ‘1’

RX8 The ninth bit of data frame in Asynchronous or Synchronous mode of operation. Read this bit first before reading the receive buffer.

0 MSB (9th bit) received is ‘0’

1 MSB (9th bit) received is ‘1’

USTAT0, USTAT1 (USART Status Register) : CDH, FDH

7	6	5	4	3	2	1	0
UDRE	TXC	RXC	WAKE	SOFTRST	DOR	FE	PE
R/W	R/W	R/W	R/W	R/W	R	R	R

Initial value : 80_H

UDRE The UDRE flag indicates if the transmit buffer (UDATA) is ready to be loaded with new data. If UDRE is '1', it means the transmit buffer is empty and can hold one or two new data. This flag can generate an UDRE interrupt. Writing '0' to this bit position will clear UDRE flag.

0 Transmit buffer is not empty.

1 Transmit buffer is empty.

TXC This flag is set when the entire frame in the transmit shift register has been shifted out and there is no new data currently present in the transmit buffer. This flag is automatically cleared when the interrupt service routine of a TXC interrupt is executed. It is also cleared by writing '0' to this bit position. This flag can generate a TXC interrupt.

0 Transmission is ongoing.

1 Transmit buffer is empty and the data in transmit shift register are shifted out completely.

RXC This flag is set when there are unread data in the receive buffer and cleared when all the data in the receive buffer are read. The RXC flag can be used to generate a RXC interrupt.

0 There is no data unread in the receive buffer

1 There are more than 1 data in the receive buffer

WAKE This flag is set when the RX pin is detected low while the CPU is in stop mode. This flag can be used to generate a WAKE interrupt. This bit is set only when in asynchronous mode of operation. NOTE

0 No WAKE interrupt is generated.

1 WAKE interrupt is generated.

SOFTRST This is an internal reset and only has effect on USART. Writing '1' to this bit initializes the internal logic of USART and is auto cleared.

0 No operation

1 Reset USART

DOR This bit is set if a Data OverRun occurs. While this bit is set, the incoming data frame is ignored. This flag is valid until the receive buffer is read.

0 No Data OverRun

1 Data OverRun detected

FE This bit is set if the first stop bit of next character in the receive buffer is detected as '0'. This bit is valid until the receive buffer is read.

0 No Frame Error

1 Frame Error detected

PE This bit is set if the next character in the receive buffer has a Parity Error when received while Parity Checking is enabled. This bit is valid until the receive buffer is read.

0 No Parity Error

1 Parity Error detected

NOTE When the WAKE function of USART is used as a release source from STOP mode, it is required to clear this bit in the RX interrupt service routine. Else the device will not wake-up from STOP mode again by the change of RX pin.

UBAUD0, UBAUD1 (USART Baud-Rate Generation Register) : CEH, FEH

7	6	5	4	3	2	1	0
UBAUD7	UBAUD6	UBAUD5	UBAUD4	UBAUD3	UBAUD2	UBAUD1	UBAUD0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : FF_H

UBAUD [7:0] The value in this register is used to generate internal baud rate in asynchronous mode or to generate XCK clock in synchronous or spi mode. To prevent malfunction, do not write ‘0’ in asynchronous mode, and do not write ‘0’ or ‘1’ in synchronous or spi mode.

UDATA0, UDATA1 (USART Data Register) : CFH, FFH

7	6	5	4	3	2	1	0
UDATA7	UDATA6	UDATA5	UDATA4	UDATA3	UDATA2	UDATA1	UDATA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : FF_H

UDATA [7:0] The USART 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 UDATA register. Reading the UDATA register returns the contents of the Receive Buffer.

Write this register only when the UDRE flag is set. In spi or synchronous master mode, write this register even if TX is not enabled to generate clock, XCK.

11.8.14 Baud Rate Setting (example)

Table 11-13 Examples of UBAUD Settings for Commonly Used Oscillator Frequencies

(a) fOSC = Multiples of 1MHz

Baud Rate	fOSC =1.00MHz				fOSC =2.00MHz				fOSC =4.00MHz			
	U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1	
	UBAUD	ERROR	UBAUD	ERROR	UBAUD	ERROR	UBAUD	ERROR	UBAUD	ERROR	UBAUD	ERROR
2400	25	0.2%	51	0.2%	51	0.2%	103	0.2%	103	0.2%	207	0.2%
4800	12	0.2%	25	0.2%	25	0.2%	51	0.2%	51	0.2%	103	0.2%
9600	6	-7.0%	12	0.2%	12	0.2%	25	0.2%	25	0.2%	51	0.2%
14.4K	3	8.5%	8	-3.5%	8	-3.5%	16	2.1%	16	2.1%	34	-0.8%
19.2K	2	8.5%	6	-7.0%	6	-7.0%	12	0.2%	12	0.2%	25	0.2%
28.8K	1	8.5%	3	8.5%	3	8.5%	8	-3.5%	8	-3.5%	16	2.1%
38.4K	1	-18.6%	2	8.5%	2	8.5%	6	-7.0%	6	-7.0%	12	0.2%
57.6K	-	-	1	8.5%	1	8.5%	3	8.5%	3	8.5%	8	-3.5%
76.8K	-	-	1	-18.6%	1	-18.6%	2	8.5%	2	8.5%	6	-7.0%
115.2K	-	-	-	-	-	-	1	8.5%	1	8.5%	3	8.5%
230.4K	-	-	-	-	-	-	-	-	-	-	1	8.5%
250K	-	-	-	-	-	-	-	-	-	-	1	0.0%
0.5M	-	-	-	-	-	-	-	-	-	-	-	-

Baud Rate	fOSC =8.00MHz				fOSC =16.00MHz			
	U2X=0		U2X=1		U2X=0		U2X=1	
	UBAUD	ERROR	UBAUD	ERROR	UBAUD	ERROR	UBAUD	ERROR
2400	207	0.2%	-	-	-	-	-	-
4800	103	0.2%	207	0.2%	207	0.2%	-	-
9600	51	0.2%	103	0.2%	103	0.2%	207	0.2%
14.4K	34	-0.8%	68	0.6%	68	0.6%	138	-0.1%
19.2K	25	0.2%	51	0.2%	51	0.2%	103	0.2%
28.8K	16	2.1%	34	-0.8%	34	-0.8%	68	0.6%
38.4K	12	0.2%	25	0.2%	25	0.2%	51	0.2%
57.6K	8	-3.5%	16	2.1%	16	2.1%	34	-0.8%
76.8K	6	-7.0%	12	0.2%	12	0.2%	25	0.2%
115.2K	3	8.5%	8	-3.5%	8	-3.5%	16	2.1%
230.4K	1	8.5%	3	8.5%	3	8.5%	8	-3.5%
250K	1	0.0%	3	0.0%	3	0.0%	7	0.0%
0.5M	-	-	1	0.0%	1	0.0%	3	0.0%

(b) fOSC = Multiples of 1.8432MHz

Baud Rate	fOSC =1.8432MHz (x1)				fOSC =3.6864MHz (x2)				fOSC =7.3728MHz (x4)			
	U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1	
	UBAUD	ERROR	UBAUD	ERROR	UBAUD	ERROR	UBAUD	ERROR	UBAUD	ERROR	UBAUD	ERROR
2400	47	0.0%	95	0.0%	95	0.0%	191	0.0%	191	0.0%	-	-
4800	23	0.0%	47	0.0%	47	0.0%	95	0.0%	95	0.0%	191	0.0%
9600	11	0.0%	23	0.0%	23	0.0%	47	0.0%	47	0.0%	95	0.0%
14.4K	7	0.0%	15	0.0%	15	0.0%	31	0.0%	31	0.0%	63	0.0%
19.2K	5	0.0%	11	0.0%	11	0.0%	23	0.0%	23	0.0%	47	0.0%
28.8K	3	0.0%	7	0.0%	7	0.0%	15	0.0%	15	0.0%	31	0.0%
38.4K	2	0.0%	5	0.0%	5	0.0%	11	0.0%	11	0.0%	23	0.0%
57.6K	1	-25.0%	3	0.0%	3	0.0%	7	0.0%	7	0.0%	15	0.0%
76.8K	1	0.0%	2	0.0%	2	0.0%	5	0.0%	5	0.0%	11	0.0%
115.2K	-	-	1	0.0%	1	0.0%	3	0.0%	3	0.0%	7	0.0%
230.4K	-	-	-	-	-	-	1	0.0%	1	0.0%	3	0.0%
250K	-	-	-	-	-	-	1	-7.8%	1	-7.8%	3	-7.8%
0.5M	-	-	-	-	-	-	-	-	-	-	1	-7.8%

Baud Rate	fOSC =11.0592MHz (x6)				fOSC =14.7456MHz (x8)			
	U2X=0		U2X=1		U2X=0		U2X=1	
	UBAUD	ERROR	UBAUD	ERROR	UBAUD	ERROR	UBAUD	ERROR
2400	-	-	-	-	-	-	-	-
4800	143	0.0%	-	-	191	0.0%	-	-
9600	71	0.0%	143	0.0%	95	0.0%	191	0.0%
14.4K	47	0.0%	95	0.0%	63	0.0%	127	0.0%
19.2K	35	0.0%	71	0.0%	47	0.0%	95	0.0%
28.8K	23	0.0%	47	0.0%	31	0.0%	63	0.0%
38.4K	17	0.0%	35	0.0%	23	0.0%	47	0.0%
57.6K	11	0.0%	23	0.0%	15	0.0%	31	0.0%
76.8K	8	0.0%	17	0.0%	11	0.0%	23	0.0%
115.2K	5	0.0%	11	0.0%	7	0.0%	15	0.0%
230.4K	2	0.0%	5	0.0%	3	0.0%	7	0.0%
250K	2	-7.8%	5	-7.8%	3	-7.8%	6	5.3%
0.5M	-	-	2	-7.8%	1	-7.8%	3	-7.8%

11.9 I²C

11.9.1 Overview

The I²C is one of industrial standard serial communication protocols, and which uses 2 bus lines Serial Data Line (SDA) and Serial Clock Line (SCL) to exchange data. Because both SDA and SCL lines are open-drain output, each line needs pull-up resistor. The features are as shown below.

- Compatible with I²C bus standard
- Multi-master operation
- Up to 400 KHz data transfer speed
- 7 bit address
- Support 2 slave addresses
- Both master and slave operation
- Bus busy detection

11.9.2 Block Diagram

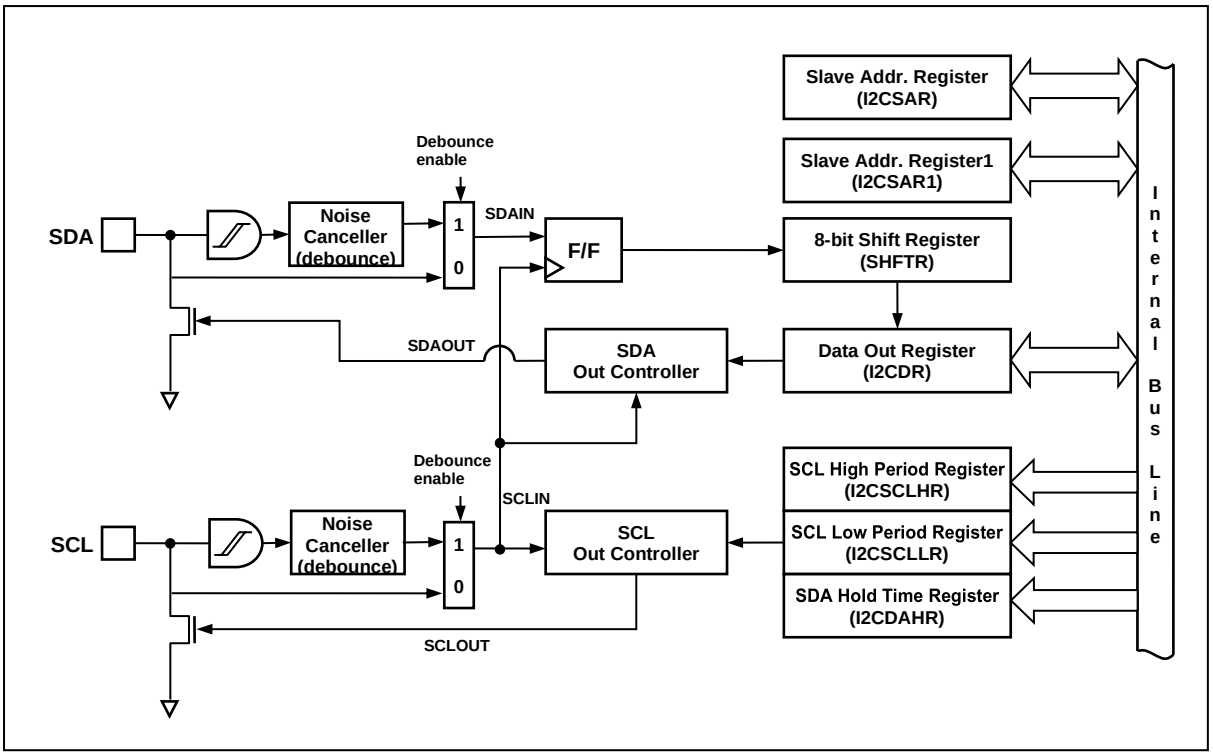


Figure 11-26 I²C Block Diagram

11.9.3 I²C Bit Transfer

The data on the SDA line must be stable during HIGH period of the clock, SCL. The HIGH or LOW state of the data line can only change when the clock signal on the SCL line is LOW. The exceptions are START(S), repeated START(Sr) and STOP(P) condition where data line changes when clock line is high.

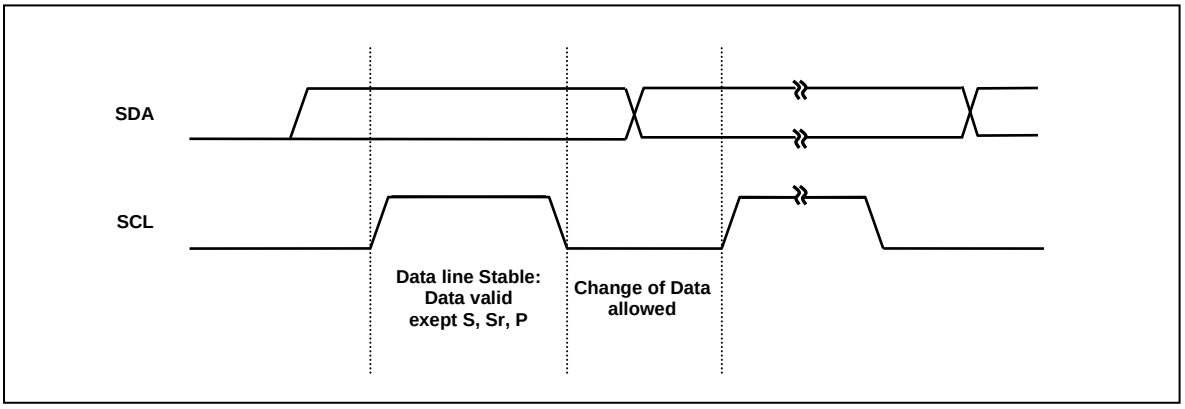


Figure 11-27 Bit Transfer on the I²C-Bus

11.9.4 Start / Repeated Start / Stop

One master can issue a START (S) condition to notice other devices connected to the SCL, SDA lines that it 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 SDA line while SCL is high defines a START (S) condition.

A low to high transition on the SDA line while SCL 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, ie, the bus is busy between START and STOP condition. If a repeated START condition (Sr) is generated instead of STOP condition, the bus stays busy. So, the START and repeated START conditions are functionally identical.

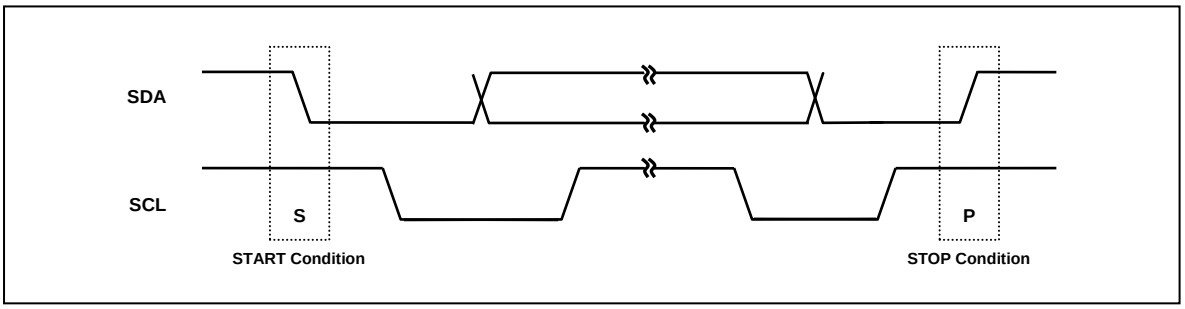


Figure 11-28 START and STOP Condition

11.9.5 Data Transfer

Every byte put on the SDA line must be 8-bits long. 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 can’t receive or transmit another complete byte of data until it has performed some other function, it can hold the clock line SCL 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 SCL.

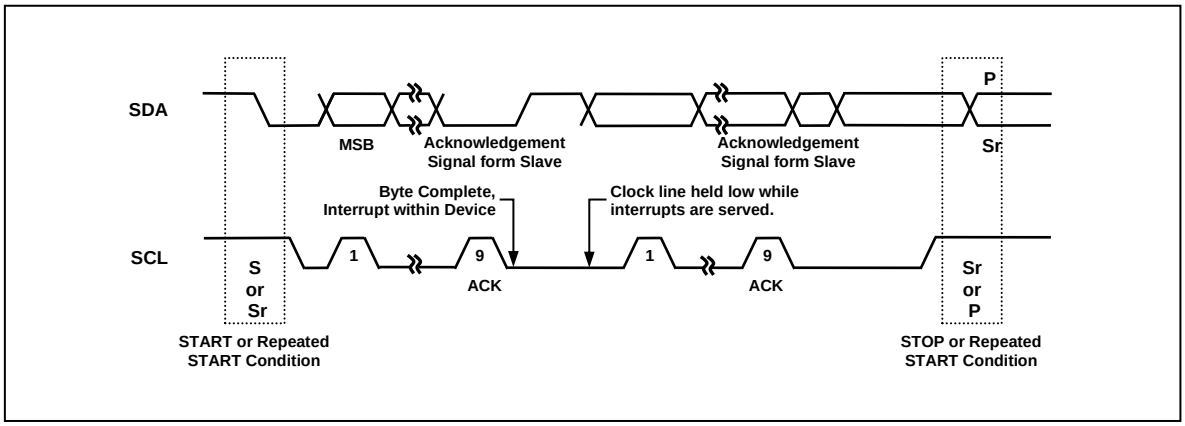


Figure 11-29 Data Transfer on the I²C-Bus

11.9.6 Acknowledge

The acknowledge related clock pulse is generated by the master. The transmitter releases the SDA line (HIGH) during the acknowledge clock pulse. The receiver must pull down the SDA line during the acknowledge clock pulse so that it remains stable 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's performing some real time function, the data line must be left HIGH by the slave. And also, when a slave addressed by a master is unable to receive more data bits, the slave receiver must release the SDA 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.

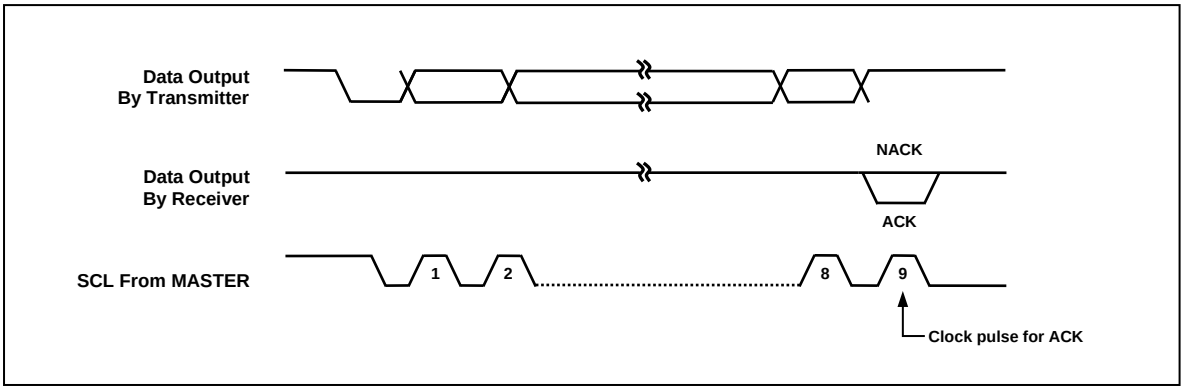


Figure 11-30 Acknowledge on the I²C-Bus

11.9.7 Synchronization / Arbitration

Clock synchronization is performed using the wired-AND connection of I²C interfaces to the SCL line. This means that a HIGH to LOW transition on the SCL line will cause the devices concerned to start counting off their LOW period and it will hold the SCL 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 SCL line if another clock is still within its LOW period. In this way, a synchronized SCL 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 SDA line, while the SCL line is at the HIGH level, in such a way that the master which transmits a

HIGH level, while another master is transmitting a LOW level will switch off its DATA output state because the level on the bus doesn't 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.

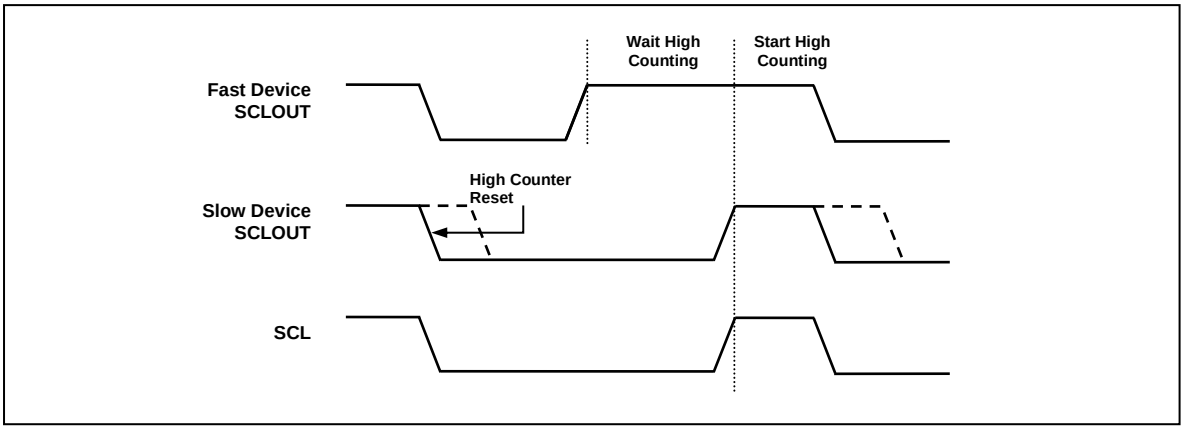


Figure 11-31 Clock Synchronization during Arbitration Procedure

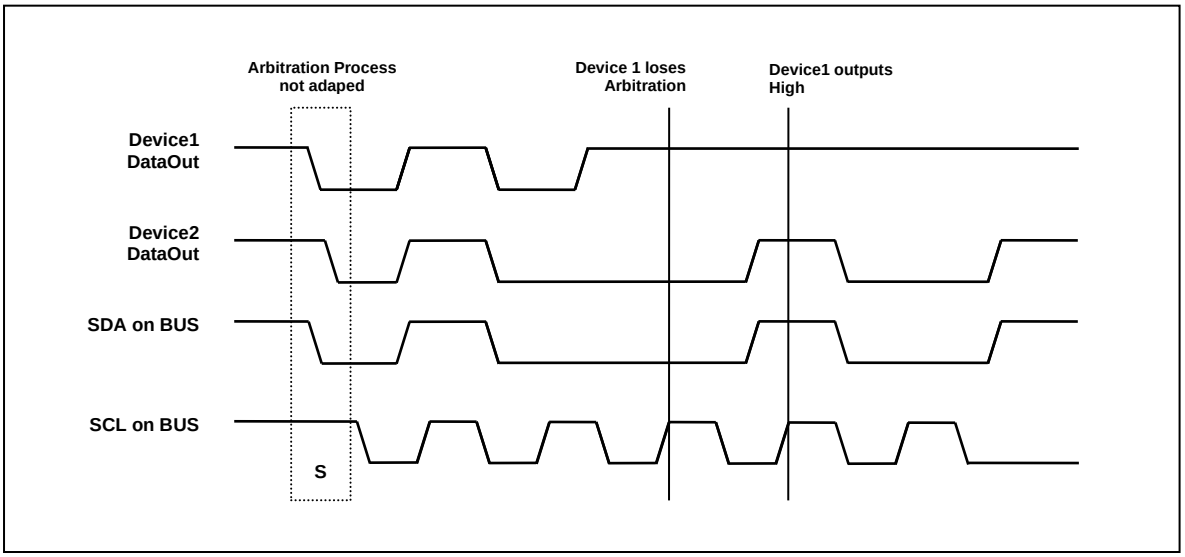


Figure 11-32 Arbitration Procedure of Two Masters

11.9.8 I2C Block Operation

The I2C block as peripheral design is independently operating with main CPU operation. The operation of I2C block does a byte unit of I2C frame. After finishing a byte operation (transmit/receive data and clock) on I2C bus system, I2C block generate I2C interrupt for next byte operation. The I2C Interrupt service manage I2C block with the SFR registers, data load/read register (I2CDR) from/to I2C bus system, block control register (I2CMR), the state register (I2CSR) contained operation result. An operation unit of I2C H/W block generates/ receive 9 SCL clock that are for 8 bits data and an ACK. I2C block send / receive ACK signal at 9th clock of SCL according to I2C specification.

The I2C application software initialize I2C block condition depended on clock system, I2C devices condition after system power on.

An application S/W prepares I2C bus communication resource on RAM buffers. If it is to set the start flag in I2CMR register. I2C block start to generate start signal and send a Slave address to slave device. All steps of I2C

communication service except start signal and slave address is done by H/W block and I2C Interrupt service. Therefore main application software can reduce time resource while I2C Data write/read operation.

I2C block design supports both functions of master/ Slave on the same block. In case of Masker device it generate SCL clock to slave device and the case of slave mode receive SCL clock from master device.

I2C block decide SDA data direction with the data direction bit (R/W) of device address in both cases of master and slave mode(TMODE bit 0-> Receive, 1-> Transmit)

Note) When an I^2C interrupt is generated by I2C block, IIF flag in I2CMR register is set and it is cleared by writing any value to I2CSR. When I^2C interrupt occurs, the SCL line is hold LOW for reading/writing I2CDR register and control I2CMR until writing any value to I2CSR. When the IIF flag is set, the I2CSR contains a value for the state of the I2C bus. According to the value in I2CSR, software can decide what to do next.

I^2C can operate in 4 modes by configuring master/slave, transmitter/receiver.

I2C block initialization process

After power ON, it is necessary to have to initialize I2C block for that I2C Block provide I2C Slave device service

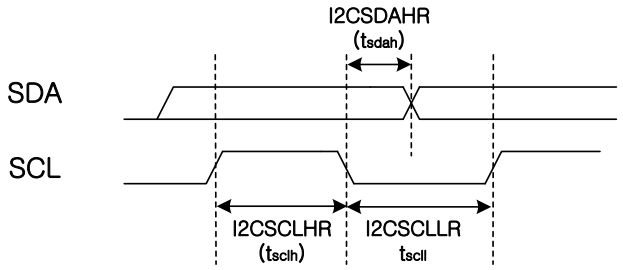
① I2C block will start operation (operation clock active) by setting IICEN bit on I2CMR register.

I2CMR = IICEN; // I2C block enable

② Reset I2C block by setting RESET bit on I2CMR register.

I2CMR = RESET; // Reset I2C block by S/W

③ Depended on I2C devices, it shall define I2C SCL max clock and write the value of SCL Low /high time and SDA hold time on I2CSCLLR, I2CSCLHR, I2CSDAHR as following diagram



The timing values are calculated as the follow formula

t_{scll} = t_{sysclk} (4xI2CSCLLR +1) → SCL clock low time

t_{sclh} = t_{sysclk} (4xI2CSCLHR +3) → SCL clock High time

t_{sdah} = t_{sysclk} (I2CSDAHR +1) → SDA data hold time after falling edge of SCL

* t_{sysclk} = system clock timing

Ex)) In case of I2C clock (100KHz) and system clock(4MHz), each of t_{scll}, t_{sclh} times is 5uSec and tsdah is 2.5 uSec.

I2CSCLLR = 5; I2CSCLHR = 4; I2CSDAHR = 9;

④ It is to decide I2C Slave device address and write the address to I2CSAR

I2CSAR = SELF_ADDRESS;

⑤ Finally be ready to get I2C data from I2C bus system as slave device by setting I2C interrupt enable, I2C block enable, ACK enable bits on I2CMR register

I2CMR = IICEN+INTEN+ACKEN; // I2C interrupt enable

I2C interrupt Service

I2C Interrupt service will use for next management action and data load/read from I2C block after I2C H/W block operation (as I2C Master/ Slave device). Because I2C block acts I2C data receiving/writing as a byte unit, I2C block make I2C interrupt for next action of I2C block. While the interrupt happen, I2C block serve the state of I2C bus condition and operation result to I2CSR register. Interrupt service look both registers of I2CMR and I2CSR and do next steps (Save a data from I2CDR, load a data to I2CDR, make STOP condition or Re-start so on).

I2C Interrupt occur at after the following cases

1) As I2C Master Device

- sending a byte on I2CDR register after setting Start bit. (GCALL interrupt)
- sending a byte on I2CDR register after write to I2CSR.(TEND interrupt)
- receiving a byte on I2CDR after write to I2CSR (TEND interrupt)
- Occurring an arbitration loss (MLOST interrupt)
- detecting Stop condition (STOP interrupt)

2) As I2C Slave device

- getting start condition and same device address from a Master (SSEL interrupt)
- sending a byte on I2CDR register after write to I2CSR.(TEND interrupt)
- receiving a byte on I2CDR after write to I2CSR (TEND interrupt)
- detecting Stop condition (STOP interrupt)

Depended on above results I2C service provide services to read/write data from/to I2CDR, generate STOP condition, make next I2C Block action by writing a data to I2CSR register.

Bus arbitration of I2C block processes from I2C bus start condition to last data of I2C data frame. If getting an arbitration loss (MLOST interrupt), I2C interrupt service make I2C block Reset for bus free.

Master transmitter

Main software is to have write/read data to/from slave I2C device. The software has to be ready to get number of data with internal RAM or sending data on internal RAM according to I2C bus protocol type of Slave device. It writes Salve Address to I2CDR register in I2C Block and then if it set START bit on I2CMR register I2C block send slave address with SCL clock to slave device. I2C Block takes master mode (MASTER bit -> 1) and take the read/write state (TMODE bit , read(0), write(1)) according to the data direction bit (R/W) of device address.

The following is examples software for the case of master mode

Master write

```
I2CMR  = IICEN+INTEN;              // set I2C block( enable IIC block, I2C interrupt)
I2CDR  =  Slave Address + Write mode;  // load target Salve Address
I2CMR  |= SRT;                     // generate start condition and send slave address
```

I2C Interrupt Service

```
If(Master Mode) and (TMODE)
    If( ACK and GCALL or ACK and TEND )
     If ( Not End of Data )
        I2CDR  =  NEXT DATA;       // load target Salve Address
        I2CSR = 0xFF;              // Byte transmit start
   ELSE
    I2CMR = IICEN+INTEN+STP;      // STOP generation
    ELSE
     Initialize I2C block          // if have ACK error, any error
```

```
End of I2C interrupt service
```

Master Read **(without sub address of Slave device)**

```
I2CMR  =  IICEN+INTEN;        // start generate
I2CDR  =  Slave Address + Read mode;          // load target Salve Address
I2CMR  |= SRT;               // generate start condition
```

I2C Interrupt Service

```
If (Master mode) and ( /TMODE)
   If( ACK and GCALL )
  I2CMR |= ACKEN                // After receive data, generate ACK
    I2CSR = 0xFF;                 // Byte transmit start
   ELSE
    if ACK and TEND )
     If ( Not End of Data )
            If(LAST Data)
           I2CMR &= ~ACKEN        // After receive data, generate ACK
       I2C_buffer = I2CDR           // read
       I2CSR = 0xFF;              // Byte transmit start
    ELSE
     If( ~ACK and TEND)
    I2CMR = IICEN+INTEN+STP;    // STOP generation
      I2CSR = 0xFF;              // Byte transmit start
     ELSE
      Initialize I2C block              // if have ACK error, any error
End of I2C interrupt service
```

Slave Receiver

I2C Block that is under IIC enable and INTEN enable on I2CMR is monitoring I2C bus lines for being a start condition and self-address with I2CSAD. To have both signals of start signal and getting self-address, I2C block generate I2C interrupt with the status bits (SSEL, BUSY RXACK, SLAVE mode ..) after sending ACK signal. At the time **I2C block control SCL line to low state** for ready to get/handle next i2c data. If I2C block by I2C interrupt service is ready for next step, it is to release the SCL line to high state for getting next SCL clock from the master. I2C Block decide bus direction (data receive/transmission) by data direction (R/W) bit in Slave address from master. The state of bus direction is on TMOD bit on I2CSR register. If the master generate Stop condition I2C block receive STOP condition and generate I2C interrupt. I2C interrupt service write any data to I2CSR and finish Slave operation. I2C interrupt service and state register condition is diagrammed in Figure xxxx.

I2C Interrupt service

```
I2C Slave service
        if((Getting SSEL and send ACK)    // received Self-address form master
           if(TMODE)                    // data direction (R/W)
            I2CDR=I2C_TXData           // Transmission mode, Load data
          else
  I2C_RXData =I2CDR
        else
          if (Get STOP condition)
          else
               if (TMODE)               // data direction (R/W)
                I2CDR= I2C_TXData     // Transmission mode, Load data
              else
```

```
    I2C_RXData =I2CDR     // Save received Data
I2CSR=0xff;
```

11.9.9 Register Map

Name	Address	Dir	Default	Description
I2CMR	DAH	R/W	00H	I^2C Mode Control Register
I2CSR	DBH	R	00H	I^2C Status Register
I2CSCLLR	DCH	R/W	3FH	SCL Low Period Register
I2CSCLHR	DDH	R/W	3FH	SCL High Period Register
I2CSDAHR	DEH	R/W	01H	SDA Hold Time Register
I2CDR	DFH	R/W	FFH	I^2C Data Register
I2CSAR	D7H	R/W	00H	I^2C Slave Address Register
I2CSAR1	D6H	R/W	00H	I^2C Slave Address Register 1

11.9.10 I^2C Register Description

I^2C Registers are composed of I^2C Mode Control Register (I2CMR), I^2C Status Register (I2CSR), SCL Low Period Register (I2CSCLLR), SCL High Period Register (I2CSCLHR), SDA Hold Time Register (I2CSDAHR), I^2C Data Register (I2CDR), and I^2C Slave Address Register (I2CSAR).

11.9.11 Register Description for I^2C

I2CMR (I^2C Mode Control Register) : DAH

7	6	5	4	3	2	1	0
IIF	IICEN	RESET	INTEN	ACKEN	MASTER	STOP	START
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W

Initial value : 00H

IIF This is interrupt flag bit.
- 0 No interrupt is generated or interrupt is cleared
- 1 An interrupt is generated

IICEN Enable I^2C Function Block (by providing clock)
- 0 I^2C is inactive
- 1 I^2C is active

RESET Initialize internal registers of I^2C.
- 0 No operation
- 1 Initialize I^2C, auto cleared

INTEN Enable interrupt generation of I^2C.
- 0 Disable interrupt, operates in polling mode
- 1 Enable interrupt

ACKEN Controls ACK signal generation at ninth SCL period.
Note) ACK signal is output (SDA=0) for the following 3 cases.
When received address packet equals to SLA bits in I2CSAR
When received address packet equals to value 0x00 with GCALL enabled
When I^2C operates as a receiver (master or slave)
- 0 No ACK signal is generated (SDA=1)
- 1 ACK signal is generated (SDA=0)

MASTER Represent operating mode of I^2C
- 0 I^2C is in slave mode
- 1 I^2C is in master mode

STOP When I^2C is master, generates STOP condition.

0 No operation

1 STOP condition is to be generated

START When I^2C is master, generates START condition.

0 No operation

1 START or repeated START condition is to be generated

I2CSR (I^2C Status Register) : DBH

7	6	5	4	3	2	1	0
GCALL	TEND	STOP	SSEL	MLOST	BUSY	TMODE	RXACK
R	R	R	R	R	R	R	R

Initial value : 00H

GCALL This bit has different meaning depending on whether I^2C is master or slave. Note 1)

When I^2C is a master, this bit represents whether it received AACK (Address ACK) from slave.

When I^2C is a slave, this bit is used to indicate general call.

0 No AACK is received (Master mode)

1 AACK is received (Master mode)

0 Received address is not general call address (Slave mode)

1 General call address is detected (Slave mode)

TEND This bit is set when 1-Byte of data is transferred completely. Note 1)

0 1 byte of data is not completely transferred

1 1 byte of data is completely transferred

STOP This bit is set when STOP condition is detected. Note 1)

0 No STOP condition is detected

1 STOP condition is detected

SSEL This bit is set when I^2C is addressed by other master. Note 1)

0 I^2C is not selected as slave

1 I^2C is addressed by other master and acts as a slave

MLOST This bit represents the result of bus arbitration in master mode. Note 1)

0 I^2C maintains bus mastership

1 I^2C has lost bus mastership during arbitration process

BUSY This bit reflects bus status.

0 I^2C bus is idle, so any master can issue a START condition

1 I^2C bus is busy

TMODE This bit is used to indicate whether I^2C is transmitter or receiver.

0 I^2C is a receiver

1 I^2C is a transmitter

RXACK This bit shows the state of ACK signal.

0 No ACK is received

1 ACK is generated at ninth SCL period

Note 1) These bits can be source of interrupt.

When an I^2C interrupt occurs except for STOP interrupt, the SCL line is hold LOW. To release SCL, write arbitrary value to I2CSR. When I2CSR is written, the TEND, STOP, SSEL, LOST, RXACK bits are cleared.

I2CSCLLR (SCL Low Period Register) : DCH

7	6	5	4	3	2	1	0
SCLL7	SCLL6	SCLL5	SCLL4	SCLL3	SCLL2	SCLL1	SCLL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 3FH

SCLL[7:0] This register defines the LOW period of SCL when I^{2}C operates in master mode. The base clock is SCLK, the system clock, and the period is calculated by the formula : $t_{SCLK} \times (4 \times SCLL + 1)$ where t_{SCLK} is the period of SCLK.

I2CSCLHR (SCL High Period Register) : DDH

7	6	5	4	3	2	1	0
SCLH7	SCLH6	SCLH5	SCLH4	SCLH3	SCLH2	SCLH1	SCLH0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 3FH

SCLH[7:0] This register defines the HIGH period of SCL when I^{2}C operates in master mode. The base clock is SCLK, the system clock, and the period is calculated by the formula : $t_{SCLK} \times (4 \times SCLH + 3)$ where t_{SCLK} is the period of SCLK.

So, the operating frequency of I^{2}C in master mode (fI2C) is calculated by the following equation.

$$\mathrm{fI2C} = \frac{1}{\mathrm{tSCLK} \times (4\,(\mathrm{SCLL} + \mathrm{SCLH}) + 4)}$$

I2CSDAHR (SDA Hold Time Register) : DEH

7	6	5	4	3	2	1	0
SDAH7	SDAH6	SDAH5	SDAH4	SDAH3	SDAH2	SDAH1	SDAH0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 01H

SDAH[7:0] This register is used to control SDA output timing from the falling edge of SCL. Note that SDA is changed after $t_{SCLK} \times SDAH$. In master mode, load half the value of SCLL to this register to make SDA change in the middle of SCL. In slave mode, configure this register regarding the frequency of SCL from master. The SDA is changed after $t_{SCLK} \times (SDAH + 1)$. So, to insure normal operation in slave mode, the value $t_{SCLK} \times (SDAH + 1)$ must be smaller than the period of SCL.

I2CDR (I^{2}C Data Register) : DFH

7	6	5	4	3	2	1	0
ICD7	ICD6	ICD5	ICD4	ICD3	ICD2	ICD1	ICD0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : FFH

ICD[7:0] When I^{2}C is configured as a transmitter, load this register with data to be transmitted. When I^{2}C is a receiver, the received data is stored into this register.

I2CSAR (I^{2}C Slave Address Register) : D7H

7	6	5	4	3	2	1	0
SLA7	SLA6	SLA5	SLA4	SLA3	SLA2	SLA1	GCALLEN
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

SLA[7:1] These bits configure the slave address of this I^{2}C module when I^{2}C operates in slave mode.

GCALLEN This bit decides whether I^{2}C allows general call address or not when I^{2}C operates in slave mode.

0 Ignore general call address

1 Allow general call address

I2CSAR1 (I^{2}C Slave Address Register 1) : D6H

7	6	5	4	3	2	1	0
SLA7	SLA6	SLA5	SLA4	SLA3	SLA2	SLA1	GCALLEN
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

SLA[7:1] These bits configure the slave address of this I^{2}C module when I^{2}C operates in slave mode.

GCALLEN This bit decides whether I^{2}C allows general call address or not when I^{2}C operates in slave mode.

0 Ignore general call address

1 Allow general call address

11.10 12-Bit A/D Converter

11.10.1 Overview

The analog-to-digital converter (A/D) allows conversion of an analog input signal to a corresponding 12-bit digital value. The A/D module has 8 analog inputs. The output of the multiplex is the input into the converter, which generates the result via successive approximation. The A/D module has four registers which are the control register ADCM (A/D Converter Mode Register), ADCM2 (A/D Converter Mode Register 2) and A/D result register ADCHR (A/D Converter Result High Register) and ADCLR (A/D Converter Result Low Register). It is selected for the corresponding channel to be converted by setting ADSEL[3:0]. To executing A/D conversion, ADST bit sets to '1'. The register ADCHR and ADCLR contains the results of the A/D conversion. When the conversion is completed, the result is loaded into the ADCHR and ADCLR, the A/D conversion status bit AFLAG is set to '1', and the A/D interrupt is set. For processing A/D conversion, AFLAG bit is read as '0'. If using STBY (power down) bit, the ADC is disabled. Also internal timer, external generating event, comparator, the trigger of timer1pwm and etc. can start ADC regardless of interrupt occurrence.

ADC Conversion Time = ADCLK * 60 cycles

After STBY bit is reset (ADC power enable) and it is restarted, during some cycle, ADC conversion value may have an inaccurate value.

11.10.2 Block Diagram

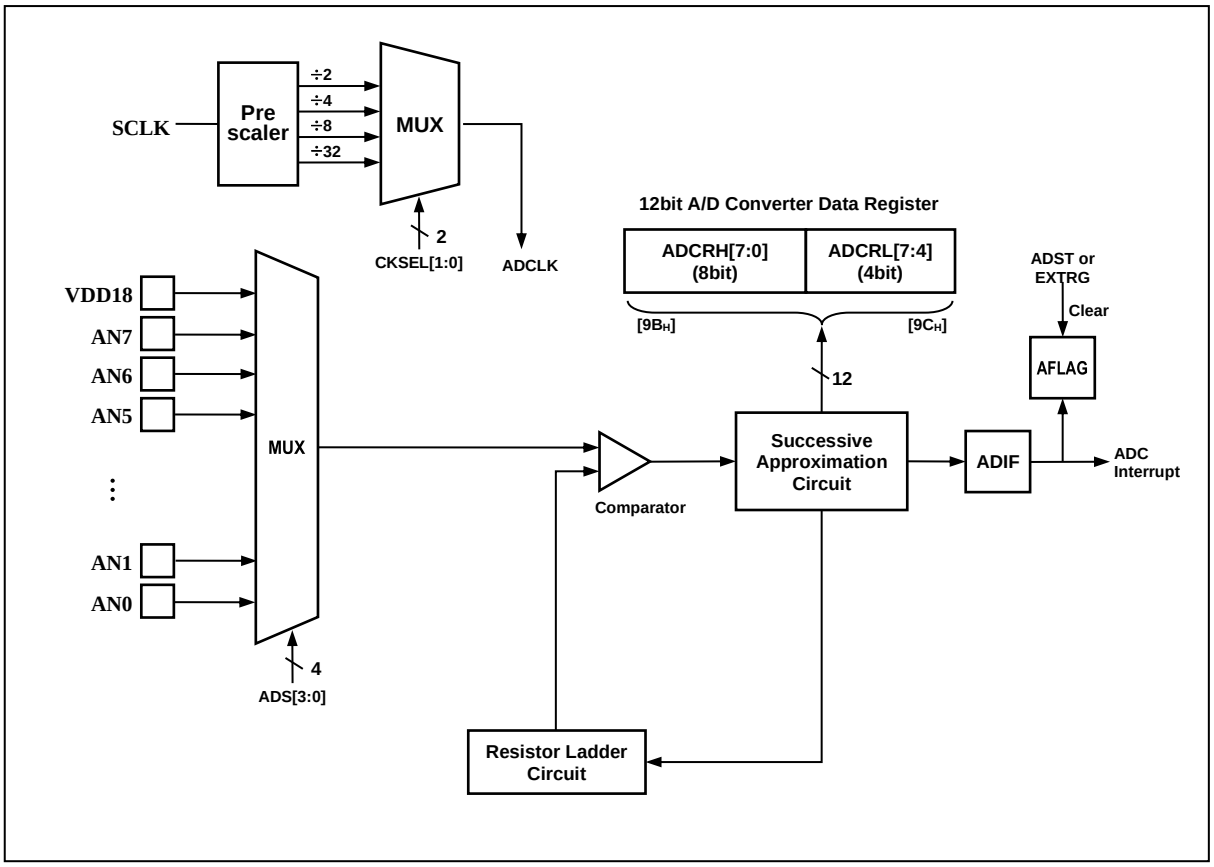


Figure 11-33 ADC Block Diagram

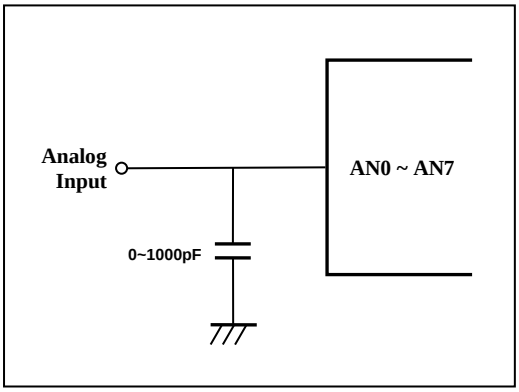


Figure 11-34 A/D Analog Input Pin Connecting Capacitor

11.10.3 ADC Operation

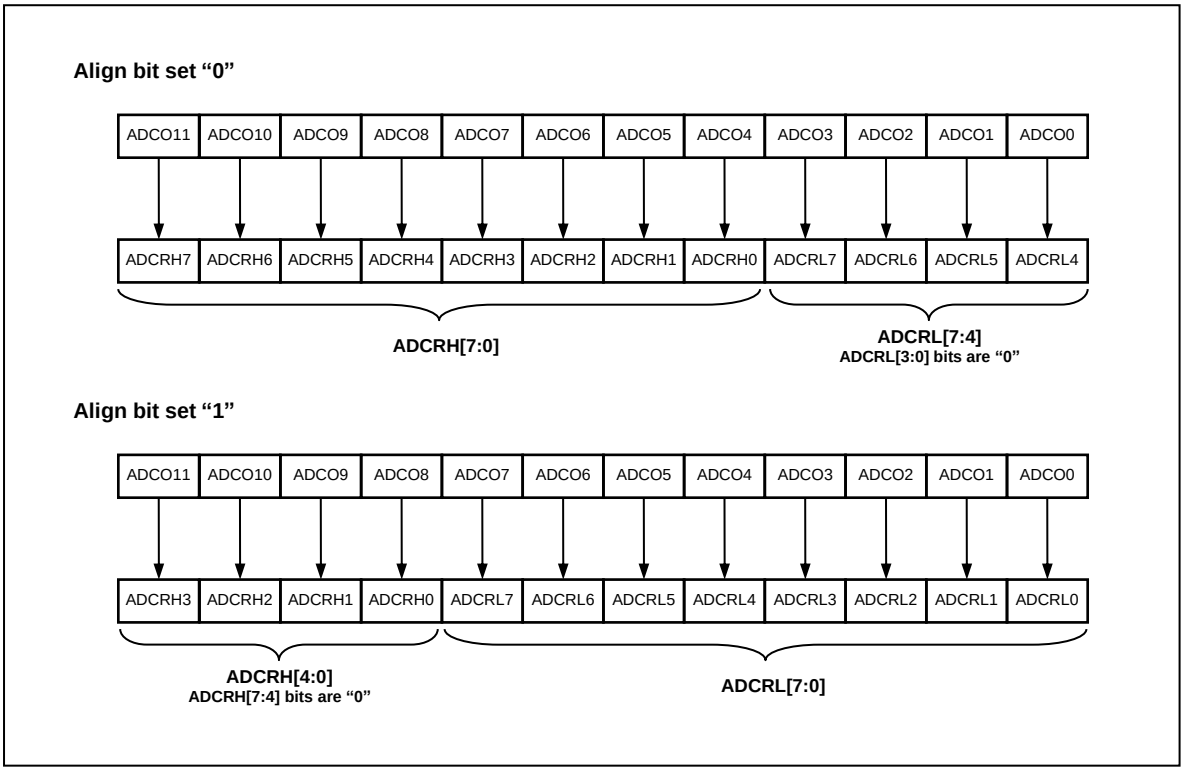


Figure 11-35 ADC Operation for Align Bit

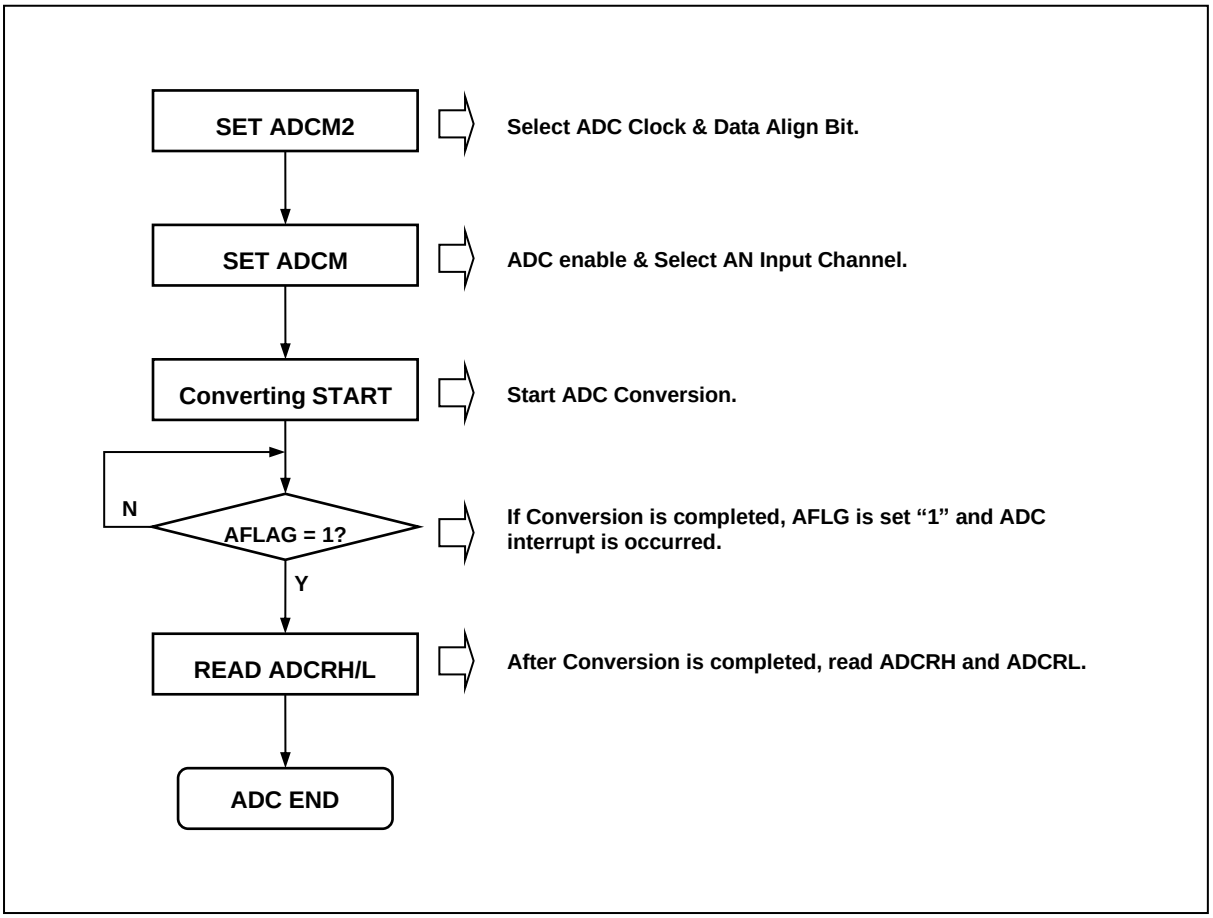


Figure 11-36 Converter Operation Flow

11.10.4 Register Map

Name	Address	Dir	Default	Description
ADCM	D1H	R/W	8FH	A/D Converter Mode Register
ADCM2	D2H	R/W	02H	A/D Converter Mode 2 Register
ADCRL	D3H	R	-	A/D Converter Result Low Register
ADCRH	D4H	R	-	A/D Converter Result High Register

11.10.5 ADC Register Description

The ADC Register consists of A/D Converter Mode Register (ADCM), A/D Converter Result High Register (ADCRH), A/D Converter Result Low Register (ADCRL), A/D Converter Mode 2 Register (ADCM2).

11.10.6 Register Description for ADC

ADCM (A/D Converter Mode Register) : D1H

7	6	5	4	3	2	1	0
STBY	ADST	REFSEL	AFLAG	ADSEL3	ADSEL2	ADSEL1	ADSEL0
R/W	R/W	R/W	R	R/W	R/W	R/W	R/W

Initial value : 8FH

STBY Control operation of A/D standby (power down)

0 ADC module enable

1 ADC module disable (power down)

ADST Control A/D Conversion stop/start.

0 ADC Conversion Stop

1 ADC Conversion Start

REFSEL A/D Converter reference selection

0 Internal Reference (Vref=VDD)

1 External Reference(AVREF, AN0 disable)

AFLAG A/D Converter operation state

0 During A/D Conversion

1 A/D Conversion finished

ADSEL[3:0] A/D Converter input selection

ADSEL3	ADSEL 2	ADSEL 1	ADSEL 0	Description
0	0	0	0	Channel0(AN0)
0	0	0	1	Channel1(AN1)
0	0	1	0	Channel2(AN2)
0	0	1	1	Channel3(AN3)
0	1	0	0	Channel4(AN4)
0	1	0	1	Channel5(AN5)
0	1	1	0	Channel6(AN6)
0	1	1	1	Channel7(AN7)
1	0	0	0	Reserved
1	0	0	1	Reserved
1	0	1	0	Reserved
1	0	1	1	Reserved
1	1	0	0	Reserved
1	1	0	1	Reserved
1	1	1	0	Reserved
1	1	1	1	VDD18 (Test Only)

ADCM2 (A/D Converter Mode 2 Register) : D2H

7	6	5	4	3	2	1	0
EXTRG	TSEL2	TSEL1	TSEL0	ALIGN	CKSEL2	CKSEL1	CKSEL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 02H

EXTRG A/D External Trigger

A/D conversion Start by external Trigger, and Stop by clearing this bit

0 A/D conversion Stop and External Trigger disable (default)

1 External Trigger enable

TSEL[2:0] A/D Trigger Source selection

TSEL2	TSEL1	TSEL0	Description
0	0	0	Ext. Interrupt0
0	0	1	Ext. Interrupt1
0	1	0	I2C interrupt
0	1	1	USART RX interrupt
1	0	0	TIMER0 interrupt

1	0	1	TIMER1 interrupt
1	1	0	WDT interrupt
1	1	1	TOUCH interrupt

ALIGN A/D Converter data align selection.

0 MSB align (ADCRH[7:0], ADCRL[7:4])

1 LSB align (ADCRH[3:0], ADCRL[7:0])

CKSEL[2:0] A/D Converter Clock selection

CKSEL2	CKSEL1	CKSEL0	ADC Clock	ADC VDD
0	0	0	fx/2	Test Only
0	0	1	fx/4	3V~5V
0	1	0	fx/8 (default)	3V~5V
0	1	1	fx/16	3V~5V
1	0	0	fx/32	2.7V~3V
1	0	1	fx/64	2.4V~2.7V
1	1	0	fx/64	2.4V~2.7V
1	1	1	fx/64	2.4V~2.7V

Note) 1. fx : system clock

2. ADC clock must be used Max. 3MHz greater than 4V.

3. ADC clock must be used Max. 2MHz less than 4V.

ADCRL (A/D Converter Result Low Register) : D3H

7	6	5	4	3	2	1	0
ADDM3 ADDL7	ADDM2 ADDL6	ADDM1 ADDL5	ADDM0 ADDL4	ADDL3	ADDL2	ADDL1	ADDL0
R	R	R	R	R	R	R	R

Initial value : xxH

ADDM[3:0] MSB align, A/D Converter Low result (4-bit)

ADDL[7:0] LSB align, A/D Converter Low result (8-bit)

ADCRH (A/D Converter Result High Register) : D4H

7	6	5	4	3	2	1	0
ADDM11	ADDM10	ADDM9	ADDM8	ADDM7 ADDL11	ADDM6 ADDL10	ADDM5 ADDL9	ADDM4 ADDL8
R	R	R	R	R	R	R	R

Initial value : xxH

ADDM[11:4] MSB align, A/D Converter High result (8-bit)

ADDL[11:8] LSB align, A/D Converter High result (4-bit)

11.11 24-Channel Touch Switch

11.11.1 Features

. Self Capacitive Touch Key Sensor.
. Total 24-channel Touch Key Support.
. 16-bits Sensing Resolutions.
. Fast Initial Self Calibration.
. Key Detection Mode : Single/Multi-Mode.
. Clock Frequency during Sensing Operation : 16MHz.
. The Improvement of the SNR by Bias-Calibration in Analog Sensing Block..
. Touch Circuit Noise Filtering Reinforcement
. VDD Operating Voltage : 2.7V ~ 5.5V.
. Current Consumption@STOPmode :< 1uA.
. Operation Temperature : -40℃ ~ 85℃.

Note) When using as touch channel, be sure to set the corresponding port to input(PxIO.bit=0) and set PSR.bit to 1.

11.11.2 Block Diagram

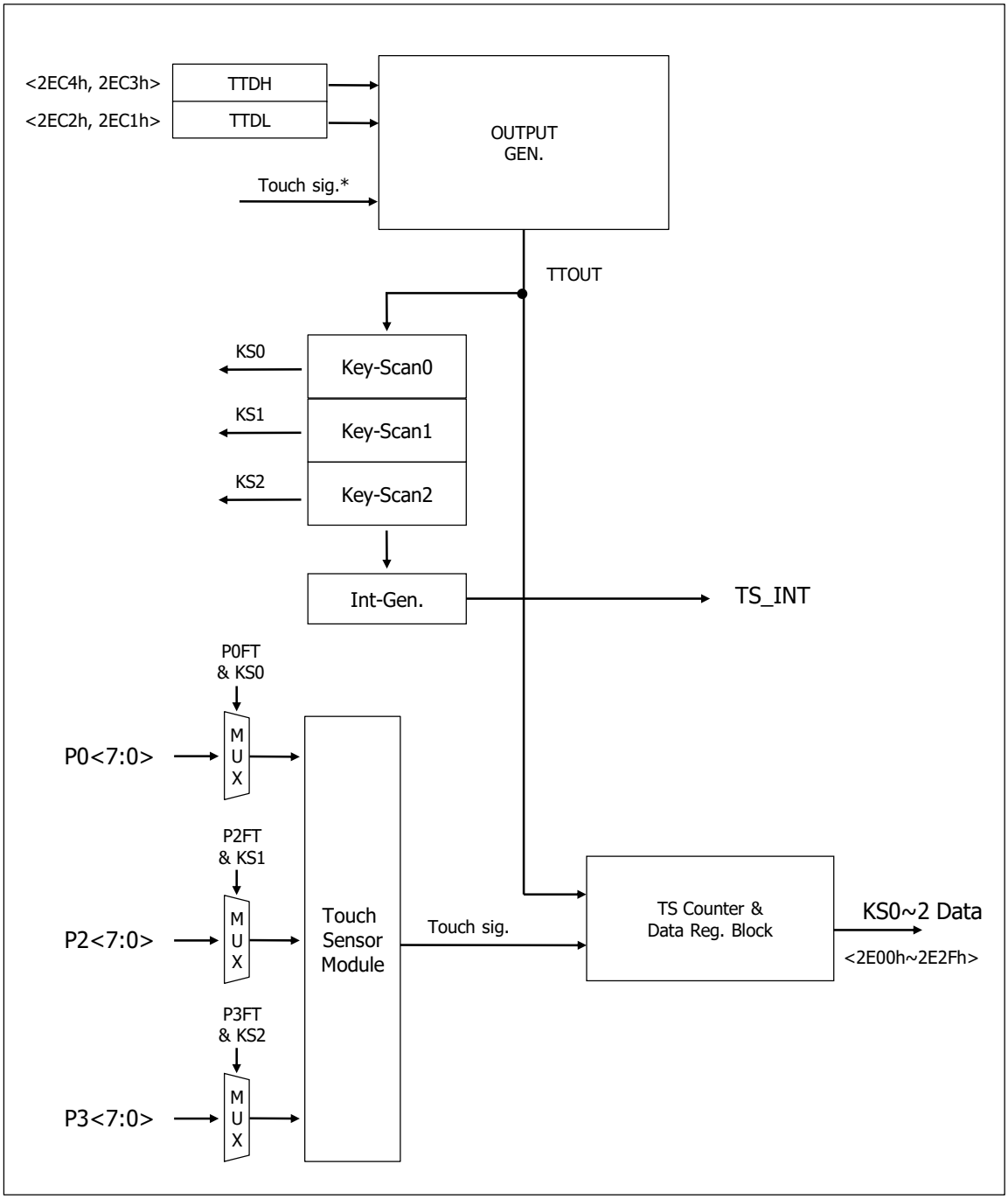


Figure 11-37 24 Channels Touch Switch Block Diagram

11.11.3 Register Map

Name	Address	Dir	Default	Description
TSCR	E2H	R/W	00H	Touch Sensing Interrupt Flag Register
TSTR	E3H	R/W	00H	Touch Sensing Test Register
KS00H	2E00H	R		Touch Sensing Data High Register
KS00L	2E01H	R		Touch Sensing Data Low Register
…				
KS27H	2E2EH	R		Touch Sensing Data High Register
KS27L	2E2FH	R		Touch Sensing Data Low Register
RSD00	2E40H	W	-	Resistor Selection Data Register
RSD01	2E41H	W	-	Resistor Selection Data Register
…				
RSD26	2E5EH	W	-	Resistor Selection Data Register
RSD27	2E5FH	W	-	Resistor Selection Data Register
TTMR	2EC0H	R/W	00H	Touch-Timer Mode Register
TTDLL	2EC1H	W	-	Touch Timer Data Low Register Low-Byte
TTDLH	2EC2H	W	-	Touch Timer Data Low Register High-Byte
TTDHL	2EC3H	W	-	Touch Timer Data High Register Low-Byte
TTDHH	2EC4H	W	-	Touch Timer Data High Register High-Byte
KENR	2EC5H	R/W	00H	Touch Key Enable Register
KSMR	2EC6H	R/W	00H	Touch Key Scan Register
FPCAL	2EC8H	R/W	00H	Functional Power Calibration Register
VCAL	2EC9H	R/W	00H	Voltage Calibration Register
CCAL	2ECAH	R/W	00H	Current Calibration Register for Frequency Control
TSCTL0	2ECBH	R/W	00H	Touch Sensor Control Register 0
TSCTL1	2ECCH	R/W	00H	Touch Sensor Control Register 1
P0FT	2ECDH	R/W	00H	P0 Port Touch Function Register
P2FT	2ECEH	R/W	00H	P2 Port Touch Function Register
P3FT	2ECFH	R/W	00H	P3 Port Touch Function Register

11.11.4 Register Description for Touch Sensing

TSCR (Touch Sensing Interrupt Flag Register) : E2H

7	6	5	4	3	2	1	0
TSIFE-	-	-	-	-	-	-	TSIFR
R/W	-	-	-	-	-	-	R/W

Initial value : 00H

TSIFE Touch Sensor Interrupt Enable
- 0 Touch Sensor INT Disable (Default)
- 1 Touch Sensor INT Enable

TSIFR Touch Sensor Interrupt Flag
- 0 Not ready for RAW data (Default)
- 1 Ready for RAW data. Can read the RAW data.

TSTR (Touch Sensing Test Register) : E3H (TEST only)

7	6	5	4	3	2	1	0
TESTEN1	TESTEN0	NCEN1	NCEN0	-	-	-	-
R/W	R/W	R/W	R/W	-	-	-	-

Initial value : 00H

TESTEN1 Touch Sensor digital test enable
- 0 Test output disable (Default)
- 1 Test output enable at P1[1:0]

TESTEN0 Touch Sensor analog test enable
- 0 Test output disable (Default)
- 1 TEST_TSV signal output enable at P1[5]

NCEN[1:0] Select time of noise canceller
- 11 Cancel under 25 ns
- 10 Cancel under 20 ns
- 01 Cancel under 15 ns
- 00 Cancel under 10 ns (default)

RSD00 (Resistor Selection Data Register 0) : 2E40H

7	6	5	4	3	2	1	0
RSD7	RSD6	RSD5	RSD4	RSD3	RSD2	RSD1	RSD0
W	W	W	W	W	W	W	W
6K	5K	4.5K	4K	3.5K	3K	2.5K	2K

Initial value : xxH

The initial value for the all RSDxx registers (RSD00 ~ RSD27) is unknowned.
User must define the initial value for the all RSD registers except of zero value (00H).
If user does'nt defines all RSD registers but some RSD registers or zero value for RSD, the touch sensing operation will be unstable.

RSD7 Resistor 7 of Touch Signal
- 0 Resistor 7 Disable (Default)
- 1 Resistor 7 Enable

RSD6 Resistor 6 of Touch Signal
- 0 Resistor 6 Disable (Default)
- 1 Resistor 6 Enable

RSD5	Resistor 5 of Touch Signal	
	0	Resistor 5 Disable (Default)
	1	Resistor 5 Enable
RSD4	Resistor 4 of Touch Signal	
	0	Resistor 4 Disable (Default)
	1	Resistor 4 Enable
RSD3	Resistor 3 of Touch Signal	
	0	Resistor 3 Disable (Default)
	1	Resistor 3 Enable
RSD2	Resistor 2 of Touch Signal	
	0	Resistor 2 Disable (Default)
	1	Resistor 2 Enable
RSD1	Resistor 1 of Touch Signal	
	0	Resistor 1 Disable (Default)
	1	Resistor 1 Enable
RSD0	Resistor 0 of Touch Signal	
	0	Resistor 0 Disable (Default)
	1	Resistor 0 Enable

RSD01 (Resistor Selection Data Register 1) : 2E41H
RSD02 (Resistor Selection Data Register 2) : 2E42H
RSD03 (Resistor Selection Data Register 3) : 2E43H
RSD04 (Resistor Selection Data Register 4) : 2E44H
RSD05 (Resistor Selection Data Register 5) : 2E45H
RSD06 (Resistor Selection Data Register 6) : 2E46H
RSD07 (Resistor Selection Data Register 7) : 2E47H

RSD10 (Resistor Selection Data Register 10) : 2E48H
RSD11 (Resistor Selection Data Register 11) : 2E49H
RSD12 (Resistor Selection Data Register 12) : 2E4AH
RSD13 (Resistor Selection Data Register 13) : 2E4BH
RSD14 (Resistor Selection Data Register 14) : 2E4CH
RSD15 (Resistor Selection Data Register 15) : 2E4DH
RSD16 (Resistor Selection Data Register 16) : 2E4EH
RSD17 (Resistor Selection Data Register 17) : 2E4FH

RSD20 (Resistor Selection Data Register 20) : 2E50H
RSD21 (Resistor Selection Data Register 21) : 2E51H
RSD22 (Resistor Selection Data Register 22) : 2E52H
RSD23 (Resistor Selection Data Register 23) : 2E53H
RSD24 (Resistor Selection Data Register 24) : 2E54H
RSD25 (Resistor Selection Data Register 25) : 2E55H
RSD26 (Resistor Selection Data Register 26) : 2E56H
RSD27 (Resistor Selection Data Register 27) : 2E57H

TTMR (Touch-Timer Mode Register) : 2EC0H

7	6	5	4	3	2	1	0
TTCS	CCKS	0	0	0	0	TTCK1	TTCK0
R/W	R/W		-	-	-	R/W	R/W

Initial value : 00H

TTCS Touch Timer Clear and Start
- 0 Touch-Timer Stop (Default)
- 1 Touch-Timer Clear and Start

CCKS Counter CK Selection
- 0 Counter CK = Touch Signal (Default)
- 1 Counter CK = SCK
  * Remarks : with VCAL.6

TTCK[1:0] Reference Clock Selection
- 00 SCK (Default)
- 01 TSIG0
- 10 -
- 11 SCK or IRC16M
  * Remarks : with VCAL.6

TTDLL (Touch timer Data Low Register Low-Byte) : 2EC1H

7	6	5	4	3	2	1	0
TTDLL							
W	W	W	W	W	W	W	W

Initial value : -

TTDLH (Touch timer Data Low Register High-Byte) : 2EC2H

7	6	5	4	3	2	1	0
TTDLH							
W	W	W	W	W	W	W	W

Initial value : -

TTDHL (Touch timer Data High Register Low-Byte) : 2EC3H

7	6	5	4	3	2	1	0
TTDLL							
W	W	W	W	W	W	W	W

Initial value : -

TTDHH (Touch timer Data High Register High-Byte) : 2EC4H

7	6	5	4	3	2	1	0
TTDLH							
W	W	W	W	W	W	W	W

Initial value : -

KENR (Touch Key Enable Register) : 2EC5H

7	6	5	4	3	2	1	0
KEN7	KEN6	KEN5	KEN4	KEN3	KEN2	KEN1	KEN0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

KEN7 Key-7 Enable
- 0 Disable (Default)
- 1 Enable Touch Key by P0FT7/P2FT7/P3FT7

KEN6 Key-6 Enable
- 0 Disable (Default)
- 1 Enable Touch Key by P0FT6/P2FT6/P3FT6

KEN5 Key-5 Enable
- 0 Disable (Default)
- 1 Enable Touch Key by P0FT5/P2FT5/P3FT5

KEN4 Key-4 Enable
- 0 Disable (Default)
- 1 Enable Touch Key by P0FT4/P2FT4/P3FT4

KEN3 Key-3 Enable
- 0 Disable (Default)
- 1 Enable Touch Key by P0FT3/P2FT3/P3FT3

KEN2 Key-2 Enable
- 0 Disable (Default)
- 1 Enable Touch Key by P0FT2/P2FT2/P3FT2

KEN1 Key-1 Enable
- 0 Disable (Default)
- 1 Enable Touch Key by P0FT1/P2FT1/P3FT1

KEN0 Key-0 Enable
- 0 Disable (Default)
- 1 Enable Touch Key by P0FT0/P2FT0/P3FT0

KSMR (Touch Key Scan Mode Register) : 2EC6H

7	6	5	4	3	2	1	0
CBSH	CBS2	CBS1	CBS0	MSCAN	KSEN2	KSEN1	KSEN0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

CBSH Internal Resistor Off
- 0 Resistor On(Default)
- 1 Resistor Off

CBS2 Comparator Bias2 Control
- 0 Disable (Default)
- 1 Enable

CBS1 Comparator Bias1 Control
- 0 Disable (Default)
- 1 Enable

CBS0 Comparator Bias0 Control
- 0 Disable (Default)
- 1 Enable

MSCAN Manual Key Scan Enable
- 0 Auto key Scan Mode (Default)
- 1 Manual Key Scan Mode

KSEN2 Key-Scan Module 2 Enable
- 0 Disable TSCOUNT2, Key-Scan Module 2 (Default)
- 1 Enable TSCOUNT2, Key-Scan Module 2
  * Remarks : P3 (CS16 ~ 23)

KSEN1 Key-Scan Module 1 Enable
- 0 Disable TSCOUNT1, Key-Scan Module1 (Default)
- 1 Enable TSCOUNT1, Key-Scan Module1
  * Remarks : P2 (CS8 ~ 15)

KSEN0 Key-Scan Module 0 Enable
- 0 Disable TSCOUNT0, Key-Scan Module0 (Default)
- 1 Enable TSCOUNT0, Key-Scan Module0
  * Remarks : P0 (CS0 ~ 7)

FPCAL (Functional Power Calibration Register) : 2EC8H

7	6	5	4	3	2	1	0
-	-	PCALF1	PCALF0	BCALF3	BCALF2	BCALF1	BCALF0
		R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

PCALF[1:0] Touch Sensor CBS Power Control
- 00 Maximum Voltage (Default)
- 01
- 10
- 11 Minimum Voltage

BCALF[3:0] Touch Sensor Bias Control
- 1000 Maximum Voltage
- 1001
- 1010
- ~
- 1111
- 0000 (Default)
- 0001
- 0010
- ~
- 0111 Minimum Voltage

VCAL (Voltage Calibration Register) : 2EC9H

7	6	5	4	3	2	1	0
DPEN	CCKM	VCAL5	VCAL4	VCAL3	VCAL2	VCAL1	VCAL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

DPEN Port Direct Pull-up Enable

0 Disable Direct Pull-up (Default)
1 Enable Direct Pull-up
* Remakrs : Main Chip Only

CCKM Counter CK MUX

0 CCK = SCK
1 CCK = IRC16M
* Remarks : with TTMR.6

VCAL[5:3] Touch Sensing Signal High Voltage Control

111 0.88 * VREF
110 0.84 * VREF
101 0.80 * VREF
100 0.75 * VREF
011 0.70 * VREF
010 0.65 * VREF
001 0.60 * VREF
000 0.55 * VREF (Default)

VCAL[2:0] Touch Sensing Signal Low Voltage Control

111 0.84 * VREF
110 0.80 * VREF
101 0.75 * VREF
100 0.70 * VREF
011 0.65 * VREF
010 0.60 * VREF
001 0.55 * VREF
000 0.40 * VREF (Default)

CCAL (Current Calubration Register for Frequency Control) : 2ECAH

7	6	5	4	3	2	1	0
CNV	CPV	CCAL5	CCAL4	CCAL3	CCAL2	CCAL1	CCAL0
R/W	R.W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

CNV N_Cal Fix

0 Enable (Default)
1 Disable

CPV P_Cal Fix

0 Enable (Default)
1 Disable

CCAL[5:4] Touch Sensing Signal Current Control (Freq.)

11 Maximum Frequency
10
01

	00	Minimum Frequency (Default)
CCAL[3:0]	Touch Sensing Signal Current Control(Freq.)	
	1111	Maximum Frequency
	1110	
	1101	
	~	
	1000	
	0000	(Default)
	0001	
	0010	
	~	
	0111	Minum Frequency

TSCTL0 (Touch Sensor Control Register 0) : 2ECBH

7	6	5	4	3	2	1	0
TSBS	TSEN	SPUEN	FBIASEN	NFEN	SIN2	0	SIN0
R/W	R/W	R/W	R/W	R/W	R/W		R/W

Initial value : 00H

TSBS	Touch Sensor Bias Set	
	0	Touch Sensor Bias OFF(Default)
	1	Touch Sensor Bias ON
TSEN	Touch Sensing Enable	
	0	Touch Sensing Disable (Default)
	1	Touch Sensing Enable
SPUEN	Signal Pull-up Enable (Internal High Drive Control)	
	0	Touch Signal Pull-up Disable (Default)
	1	Touch Signal Pull-up Enable
FBIASEN	Functional BIAS Enable	
	0	Config. Reg. Enable (Default)
	1	FPCAL Ref. Enable
NFEN	Noise Filter Enable	
	0	Noiste Filter Disable (Default)
	1	Noise Filter Enable
SIN2	Touch Sensing Port Driver2 Control	
	0	Driver2 Disable (Default)
	1	Driver2 Enable
SIN0	Touch Sensing Port Driver0 Control	
	0	Driver0 Disable (Default)
	1	Driver0 Enable

TSCTL1 (Touch Sensor Control Register 1) : 2ECCH

7	6	5	4	3	2	1	0
DS1	DS0	CIEN	RIEN	LSDIS	TBIAS2	TBIAS1	TBIAS0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

DS1[1:0] Delay Selection
- 00 Normal (Default)
- 01 Fast
- 10 Slow
- 11 Forbidden

CIEN NF Capacitor Enable
- 0 NF Cap. Input Disable (Default)
- 1 NF Cap. Input Enable

RIEN NF Resistor Enable
- 0 NF Resistor. Input Disable (Default)
- 1 NF Resistor Input Enable

LSDIS Low Sensing Disable
- 0 Low Sensing Enable (Default)
- 1 Low Sensing Disable

TBIAS[2:0] Touch Sensor Bias Test (* Remarks : P15)
- 000 Floating (Default)
- 001 Touch Bias Output
- 010 Touch Sensor Reference Power Output
- 011 Forbidden
- 100 Floating
- 101 Touch Signal Low-Voltage Output
- 110 Touch Signal High-Voltage Output
- 1111 Forbidden

P0FT (P0 Port Touch Function Register) : 2ECDH

7	6	5	4	3	2	1	0
P0FT7	P0FT6	P0FT5	P0FT4	P0FT3	P0FT2	P0FT1	P0FT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

P0FT7 P07 Function Selection
- 0 P07 Port-Function Selection (Default)
- 1 P07 Touch-Key Function Selection (= CS7)

P0FT 6 P06 Function Selection
- 0 P06 Port-Function Selection (Default)
- 1 P06 Touch-Key Function Selection (= CS6)

P0FT 5 P05 Function Selection
- 0 P05 Port-Function Selection (Default)
- 1 P05 Touch-Key Function Selection (= CS5)

P0FT 4 P04 Function Selection
- 0 P04 Port-Function Selection (Default)
- 1 P04 Touch-Key Function Selection (= CS4)

P0FT 3 P03 Function Selection

0 P03 Port-Function Selection (Default)
1 P03 Touch-Key Function Selection (= CS3)

P0FT 2 P02 Function Selection
0 P02 Port-Function Selection (Default)
1 P02 Touch-Key Function Selection (= CS2)

P0FT 1 P01 Function Selection
0 P01 Port-Function Selection (Default)
1 P01 Touch-Key Function Selection (= CS1)

P0FT 0 P00 Function Selection
0 P00 Port-Function Selection (Default)
1 P00 Touch-Key Function Selection (= CS0)

P2FT (P2 Port Touch Function Register) : 2ECEH

7	6	5	4	3	2	1	0
P2FT7	P2FT6	P2FT5	P2FT4	P2FT3	P2FT2	P2FT1	P2FT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

P2FT7 P27 Function Selection
0 P27 Port-Function Selection (Default)
1 P27 Touch-Key Function Selection (= CS15)

P2FT 6 P26 Function Selection
0 P26 Port-Function Selection (Default)
1 P26 Touch-Key Function Selection (= CS14)

P2FT 5 P25 Function Selection
0 P25 Port-Function Selection (Default)
1 P25 Touch-Key Function Selection (= CS13)

P2FT 4 P24 Function Selection
0 P24 Port-Function Selection (Default)
1 P24 Touch-Key Function Selection (= CS12)

P2FT 3 P23 Function Selection
0 P23 Port-Function Selection (Default)
1 P23 Touch-Key Function Selection (= CS11)

P2FT 2 P22 Function Selection
0 P22 Port-Function Selection (Default)
1 P22 Touch-Key Function Selection (= CS10)

P2FT 1 P21 Function Selection
0 P21 Port-Function Selection (Default)
1 P21 Touch-Key Function Selection (= CS9)

P2FT 0 P20 Function Selection
0 P20 Port-Function Selection (Default)
1 P20 Touch-Key Function Selection (= CS8)

P3FT (P3 Port Touch Function Register) : 2ECFH

7	6	5	4	3	2	1	0
P3FT7	P3FT6	P3FT5	P3FT4	P3FT3	P3FT2	P3FT1	P3FT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

P3FT7 P37 Function Selection
- 0 P37 Port-Function Selection (Default)
- 1 P37 Touch-Key Function Selection (= CS23)

P3FT 6 P36 Function Selection
- 0 P36 Port-Function Selection (Default)
- 1 P36 Touch-Key Function Selection (= CS22)

P3FT 5 P35 Function Selection
- 0 P35 Port-Function Selection (Default)
- 1 P35 Touch-Key Function Selection (= CS21)

P3FT 4 P34 Function Selection
- 0 P34 Port-Function Selection (Default)
- 1 P34 Touch-Key Function Selection (= CS20)

P3FT 3 P33 Function Selection
- 0 P33 Port-Function Selection (Default)
- 1 P33 Touch-Key Function Selection (= CS19)

P3FT 2 P32 Function Selection
- 0 P32 Port-Function Selection (Default)
- 1 P32 Touch-Key Function Selection (= CS18)

P3FT 1 P31 Function Selection
- 0 P31 Port-Function Selection (Default)
- 1 P31 Touch-Key Function Selection (= CS17)

P3FT 0 P30 Function Selection
- 0 P30 Port-Function Selection (Default)
- 1 P30 Touch-Key Function Selection (= CS16)

11.11.5 User Programming Procedure

Start

```
TSCTL0 = 0xF7;   // Touch Bias, Sensing On, Pull-up En, Config. En, SIN2/1/0 En.
TSCTL1 = 0x08;   // Bias Test Disable
FPCAL = 0x10;    // CBS=1
VCAL = 0x2A;     // Min. Voltage
CCAL = 0x3F;     // Fastest FREQ
RSD00 = 0x02;    // Resistor7~0 Set. For CS00
RSD01 = 0x02;    // Resistor7~0 Set. For CS01
RSD02 = 0x02;    // Resistor7~0 Set. For CS02
RSD03 = 0x02;    // Resistor7~0 Set. For CS03
RSD04 = 0x02;    // Resistor7~0 Set. For CS04
RSD05 = 0x02;    // Resistor7~0 Set. For CS05
RSD06 = 0x02;    // Resistor7~0 Set. For CS06
RSD07 = 0x02;    // Resistor7~0 Set. For CS07
RSD10 = 0x02;    // Resistor7~0 Set. For CS08
RSD11 = 0x02;    // Resistor7~0 Set. For CS09
RSD12 = 0x02;    // Resistor7~0 Set. For CS10
RSD13 = 0x02;    // Resistor7~0 Set. For CS11
RSD14 = 0x02;    // Resistor7~0 Set. For CS12
RSD15 = 0x02;    // Resistor7~0 Set. For CS13
RSD16 = 0x02;    // Resistor7~0 Set. For CS14
RSD17 = 0x02;    // Resistor7~0 Set. For CS15
RSD20 = 0x02;    // Resistor7~0 Set. For CS16
RSD21 = 0x02;    // Resistor7~0 Set. For CS17
RSD22 = 0x02;    // Resistor7~0 Set. For CS18
RSD23 = 0x02;    // Resistor7~0 Set. For CS19
RSD24 = 0x02;    // Resistor7~0 Set. For CS20
RSD25 = 0x02;    // Resistor7~0 Set. For CS21
RSD26 = 0x02;    // Resistor7~0 Set. For CS22
RSD27 = 0x02;    // Resistor7~0 Set. For CS23

KENR = 0xFF;     // Key Scan 0~7 Enable,
                 // capable of writing during TTCS=0 only
KSMR = 0x75;     // Auto Scan, Pull-down, KSEN0 Enable
P0FT = 0xFF;     // P0 Touch Port All Enable but, depend on KENR, KSMR
P2FT = 0x00;     // P2 Touch Port All Disable
P3FT = 0x00;     // P3 Touch Port All Disable
TTDLL = 0x90;    // 'h0190 = 50us @8M
TTDLH = 0x01;    //
TTDHL = 0x20;    // 'h0320 = 100us @8M
TTDHH = 0x03;    //
TTMR = 0x80;     // TTCS Start, Touch Signal Count
...
// KS00H/L, KS01H/L, KS02H/L, … read after TTMR.7=0
```

Figure 11-38 User Programming Procedure

11.12 LED Driver

11.12.1 Overview

LED drive contains a 8 COM X 20 SEG output pin . Support 1 / 4 to 1 /8 duty cycle voltage drive. The controller consists of display data RAM memory and a duty generator . SEG0-SEG19 pin can also be used as I / O pins. P0SS and P2SS register setting is invalid , SEG1S, SEG2S, SEG3S and COMS registers are used to control SEG0-19, COM0 - COM7 and select I / O port mode .

During the power-on reset , reset pin , low voltage reset or watchdog reset , LED is turned off.

Note) When using as LED Driver port, be sure to set the corresponding port to input(PxIO.bit=0) and set PSR.bit to 1.

11.12.2 Block Diagram

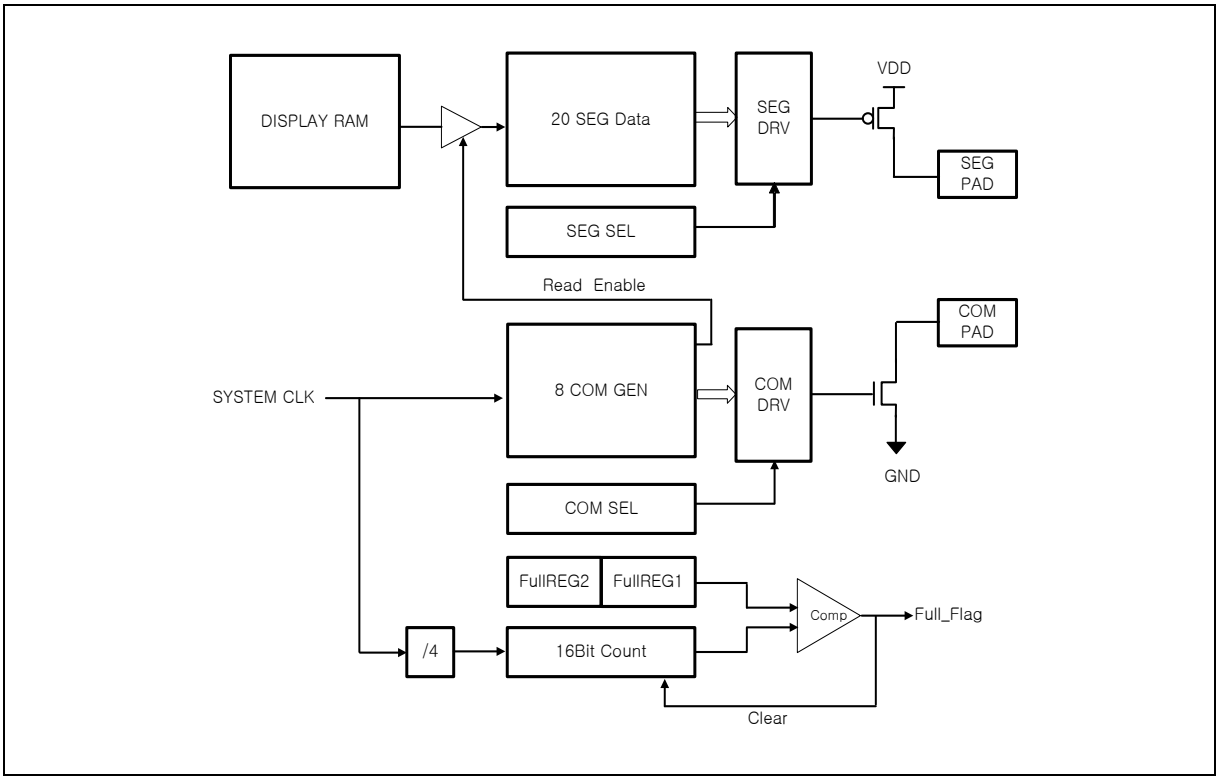


Figure 11-39 LED Driver Block Diagram

11.12.3 Register Description for LED Driver

LEDCON (LED Control Register) : 2F00H

7	6	5	4	3	2	1	0
OVERLAP	-	FULLON	LEDON	-	DUTY2	DUTY1	DUTY0
R/W	-	R/W	R/W	-	R/W	R/W	R/W

Initial value : 00H

OVERLAP OVERLAP Time Select(in association with OVRTS of LEDSR register)
0 Select (Ref.OVRTS of LEDSR register)
1 Non-select (OVERLAP TIME is System Clock)

FULLON Full Count Enable
0 Disable
1 Enable

LEDON LED Enable
0 Disable
1 Enable

DUTY[2:0] Duty Selection
000 1/2 Duty
001 1/3 Duty
010 1/4 Duty
011 1/5 Duty
100 1/6 Duty
101 1/7 Duty
110 1/8 Duty
111 1/8 Duty

LEDCCR (LED Clock Control Register) : 2F01H

7	6	5	4	3	2	1	0
LEDCK7	LEDCK6	LEDCK5	LEDCK4	LEDCK3	LEDCK2	LEDCK1	LEDCK0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

LEDCCR[7:0] LED Clock select
LED Clock frequency = System Clock frequency / 8 / LEDCK

LEDCOM (COM Scan Width Control Register) : 2F02H

7	6	5	4	3	2	1	0
LEDCOM7	LEDCOM6	LEDCOM5	LEDCOM4	LEDCOM3	LEDCOM2	LEDCOM1	LEDCOM0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

LEDCOM[7:0] LED single COM scan width select
Single-COM scan width = LED clock frequency / LEDCOM

SEG1S (SEG[7:0] Select Register) : 2F03H

7	6	5	4	3	2	1	0
SEGR7	SEGR6	SEGR5	SEGR4	SEGR3	SEGR2	SEGR1	SEGR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

SEG1S[7:0] Port Mode Select
- 0 Normal I/O Port select
- 1 SEG[7:0] select

SEG2S (SEG[15:8] Select Register) : 2F04H

7	6	5	4	3	2	1	0
SEGR15	SEGR14	SEGR13	SEGR12	SEGR11	SEGR10	SEGR9	SEGR8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

SEG2S[15:8] Port Mode Select
- 0 Normal I/O Port select
- 1 SEG[15:8] select

SEG3S (SEG[20:16] Select Register) : 2F05H

7	6	5	4	3	2	1	0
-	-	-	SEGR20	SEGR19	SEGR18	SEGR17	SEGR16
-	-	-	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

SEG3S[20:16] Port Mode Select
- 0 Normal I/O Port select
- 1 SEG[20:16] select

COMR (COM Select Register) : 2F06H

7	6	5	4	3	2	1	0
COMR7	COMR6	COMR5	COMR4	COMR3	COMR2	COMR1	COMR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

COMR[7:0] Port Mode Select
- 0 Normal I/O Port select
- 1 COM[7:0] select

DIMM (Dimming Select Register) : 2F07H

7	6	5	4	3	2	1	0
DIMM7	DIMM6	DIMM5	DIMM4	DIMM3	DIMM2	DIMM1	DIMM0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

***DIMM[7:0]** Dimming Selection Register

8'h00 COM pulse width 100%

8'hff COM pulse width 100%

else COM pulse width is LEDCLK Freq * DIMM[7:0]

OVERLAP Time < *DIMM Value < LEDCOM value

FULLREG1 (Full Count Low data Register) : 2F08H

7	6	5	4	3	2	1	0
FULL7	FULL6	FULL5	FULL4	FULL3	FULL2	FULL1	FULL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

FULL[7:0] Full Count Low data Register

FULLREG2 (Full Count High data Register) : 2F09H

7	6	5	4	3	2	1	0
FULL15	FULL14	FULL13	FULL12	FULL11	FULL10	FULL9	FULL8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

FULL[15:8] Full Count high data Register

LEDSR (LED STATUS Register) : 2F0BH

7	6	5	4	3	2	1	0
LEDINTE	OVERTS1	OVERTS0	-	END_IF	FULL_FLAG	LED_ON	-
R/W	R/W	R/W	-	R/W-	R/W	R	-

Initial value : 01H

LEDINTE LED Interrupt Enable

0 Disable

1 Enable

OVERTS[1:0] Overlap Time Select (under OVERLAP = 0)

00 1/(System Clock) X 512 (about 32us @16MHz)

01 1/(System Clock) X 256 (about 16us @16MHz)

10 1/(System Clock) X 128 (about 8us @16MHz)

11 1/(System Clock) X 64 (about 4us @16MHz)

END_IF LED Interrupt Source

0 not generated

1 generate

FULL_FLAG Whether or not the same Full Data and Count

0 Not equal

1 equal

LED_ON Whether or not LED Start

0 LED not start

1 LED start

11.12.4 LED RAM

Table 11-14 LED RAM

Address		7	6	5	4	3	2	1	0
2F10H	COM0L	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
2F11H	COM0M	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8
2F12H	COM0H	-	-	SEG21	SEG20	SEG19	SEG18	SEG17	SEG16
2F13H	COM1L	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
2F14H	COM1M	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8
2F15H	COM1H	-	-	SEG21	SEG20	SEG19	SEG18	SEG17	SEG16
2F16H	COM2L	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
2F17H	COM2M	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8
2F18H	COM2H	-	-	SEG21	SEG20	SEG19	SEG18	SEG17	SEG16
2F19H	COM3L	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
2F1AH	COM3M	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8
2F1BH	COM3H	-	-	SEG21	SEG20	SEG19	SEG18	SEG17	SEG16
2F1CH	COM4L	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
2F1DH	COM4M	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8
2F1EH	COM4H	-	-	SEG21	SEG20	SEG19	SEG18	SEG17	SEG16
2F1FH	COM5L	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
2F20H	COM5M	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8
2F21H	COM5H	-	-	SEG21	SEG20	SEG19	SEG18	SEG17	SEG16
2F22H	COM6L	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
2F23H	COM6M	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8
2F24H	COM6H	-	-	SEG21	SEG20	SEG19	SEG18	SEG17	SEG16
2F25H	COM7L	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
2F26H	COM7M	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8
2F27H	COM7H	-	-	SEG21	SEG20	SEG19	SEG18	SEG17	SEG16

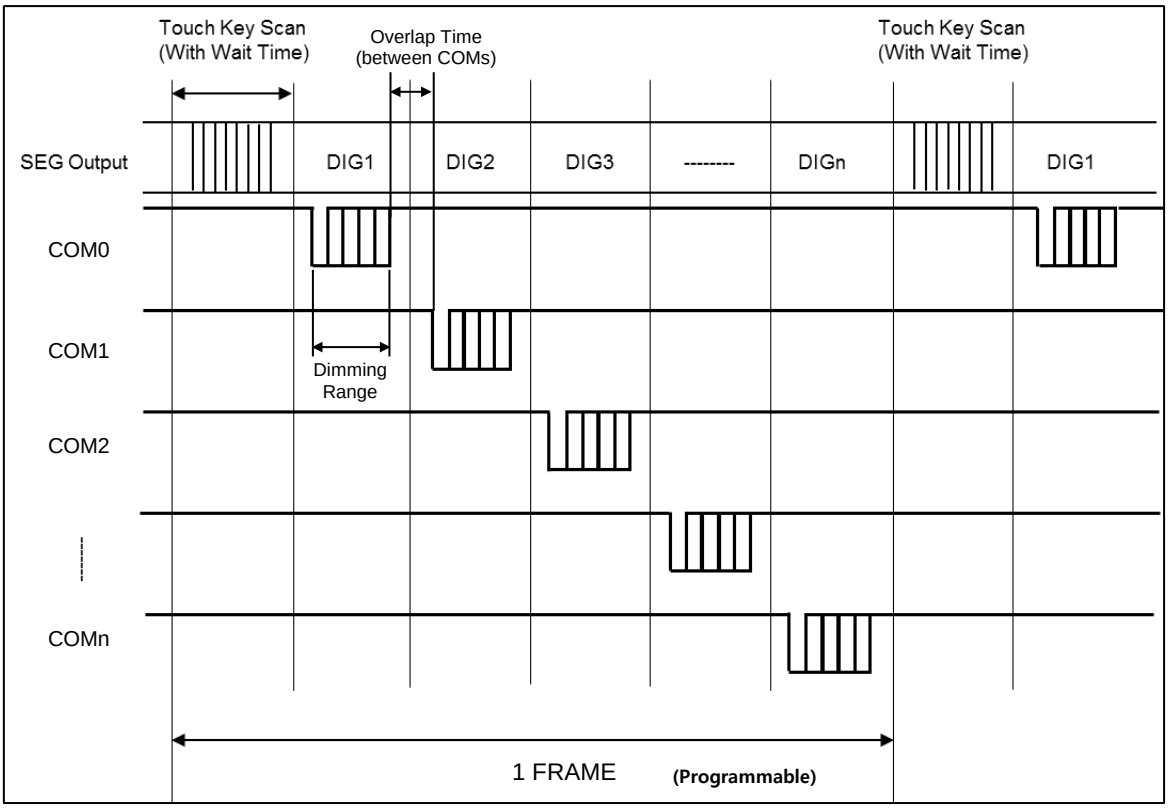


Figure 11-40 SHARE Touch Buttons

12. Power Down Operation

12.1 Overview

The A96T218 has three power-down modes to minimize the power consumption of the device. In power down mode, power consumption is reduced considerably. The device provides three kinds of power saving functions, IDLE, STOP1 and STOP2 mode. In three modes, program is stopped.

12.2 Peripheral Operation in IDLE/STOP Mode

Table 12-1 Peripheral Operation during Power Down Mode.

Peripheral	IDLE Mode	STOP1 Mode	STOP2 Mode
CPU	ALL CPU Operation are Disable	ALL CPU Operation are Disable	ALL CPU Operation are Disable
RAM	Retain	Retain	Retain
Basic Interval Timer	Operates Continuously	Operates Continuously	Stop
Watch Dog Timer	Operates Continuously	Operates Continuously	Stop
Timer	Operates Continuously	Halted	Halted
I^2C	Operates Continuously	Stop	Stop
Internal OSC (16MHz)	Oscillation	Stop	Stop
Internal RC-OSC (1MHz)	Oscillation	Oscillation	Stop
I/O Port	Retain	Retain	Retain
Control Register	Retain	Retain	Retain
Address Data Bus	Retain	Retain	Retain
Release Method	By RESET, all Interrupts	By RESET, External Interrupt, I^2C (slave mode), WDT, BIT, USART	By RESET, External Interrupt, I^2C (slave mode), USART

12.3 IDLE mode

The power control register is set to ‘01h’ to enter the IDLE Mode. In this mode, the internal oscillation circuits remain active. Oscillation continues and peripherals are operated normally but CPU stops. It is released by reset or interrupt. To be released by interrupt, interrupt should be enabled before IDLE mode. If using reset, because the device becomes initialized state, the registers have reset value.

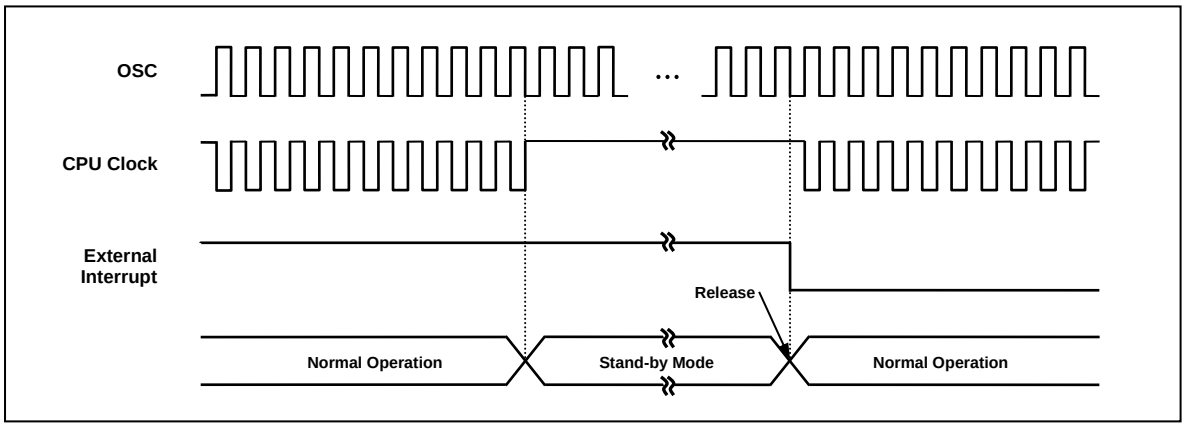


Figure 12-1 IDLE Mode Release Timing by External Interrupt

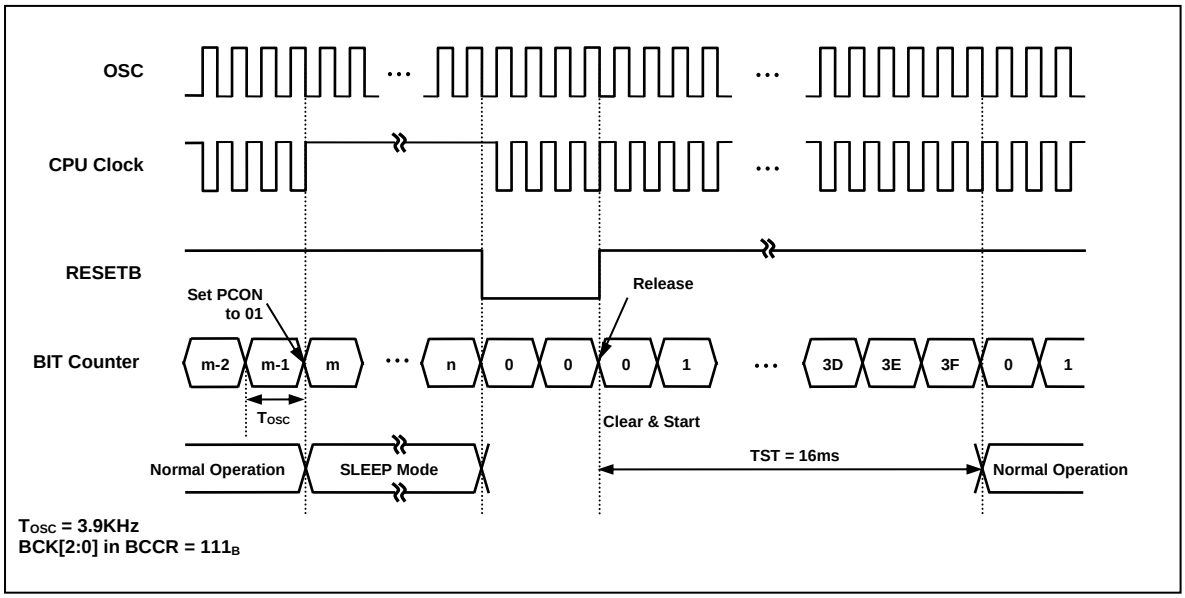


Figure 12-2 IDLE Mode Release Timing by RESETB

(Ex) MOV PCON, #0000_0001b ; setting of IDLE mode : set the bit of STOP and IDLE Control register (PCON)

12.4 STOP mode

The power control register is set to '03h' to enter the STOP Mode. In the stop mode, the main oscillator, system clock, and peripheral clock are stopped, but watch dog timer continue to operate. With the clock frozen, all functions are stopped, but the on-chip RAM and control registers are held.

The source for exit from STOP mode is hardware reset and interrupts. The reset re-defines all the control registers.

When exit from STOP mode, enough oscillation stabilization time is required to normal operation. Figure 12-3 shows the timing diagram. When released from STOP mode, the basic interval timer is activated on wake-up. Therefore, before STOP instruction, user must be set its relevant prescaler divide ratio to have long enough time (more than 1msec). this guarantees that oscillator has started and stabilized.

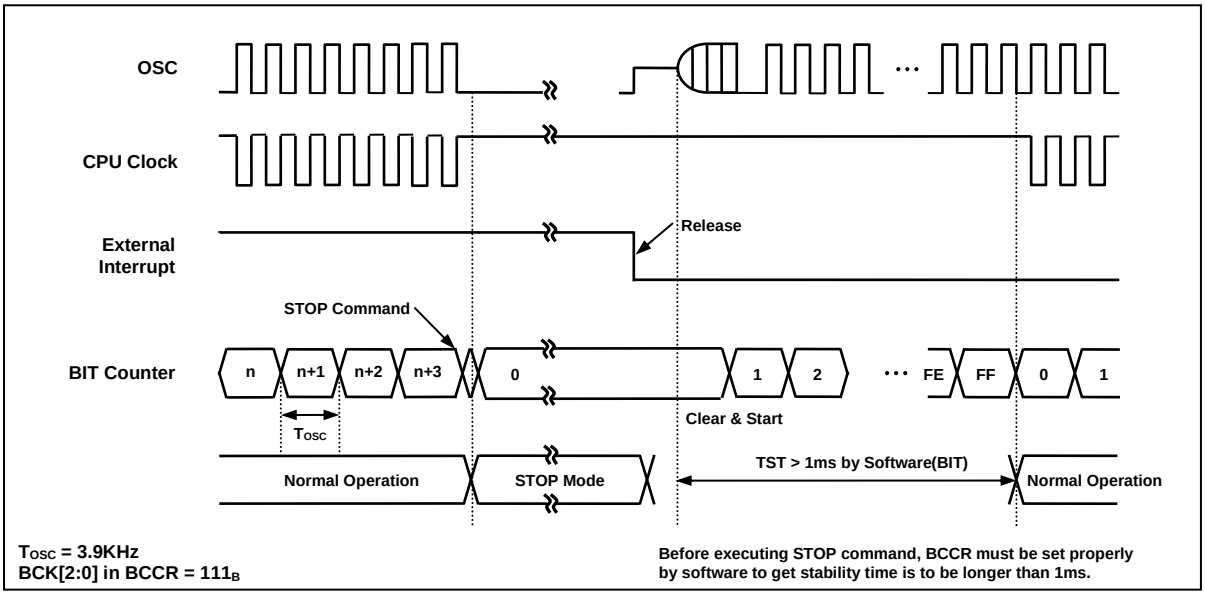


Figure 12-3 STOP Mode Release Timing by Exteranl Interrupt

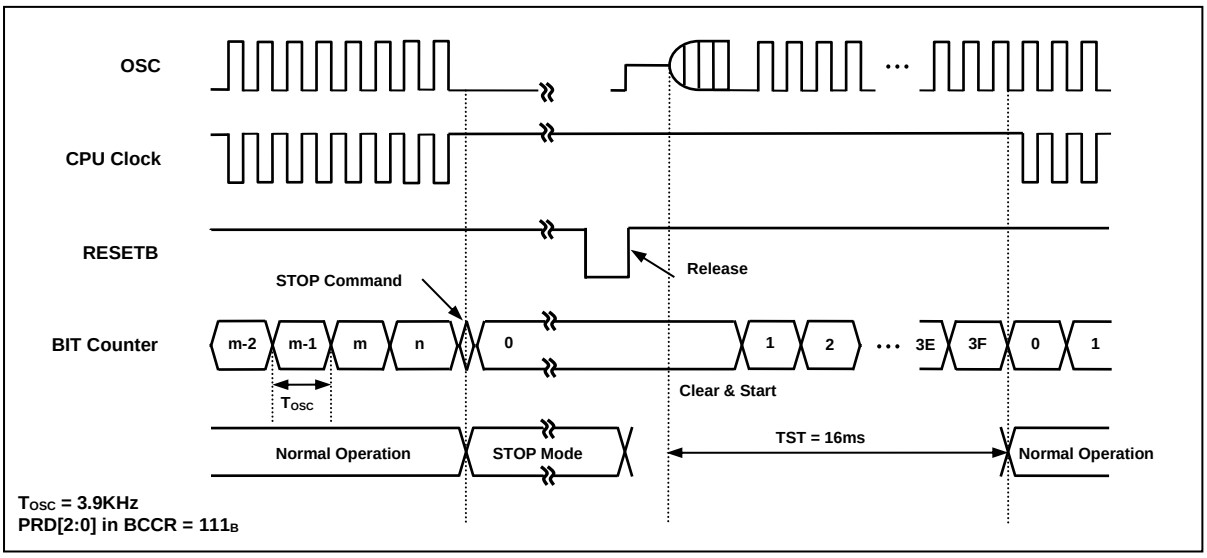


Figure 12-4 STOP Mode Release Timing by RESETB

12.5 Release Operation of STOP1, 2 Mode

After STOP1, 2 mode is released, the operation begins according to content of related interrupt register just before STOP1, 2 mode start (Figure 12-5). Interrupt Enable Flag of All (EA) of IE should be set to `1`. Released by only interrupt which each interrupt enable flag = `1`, and jump to the relevant interrupt service routine.

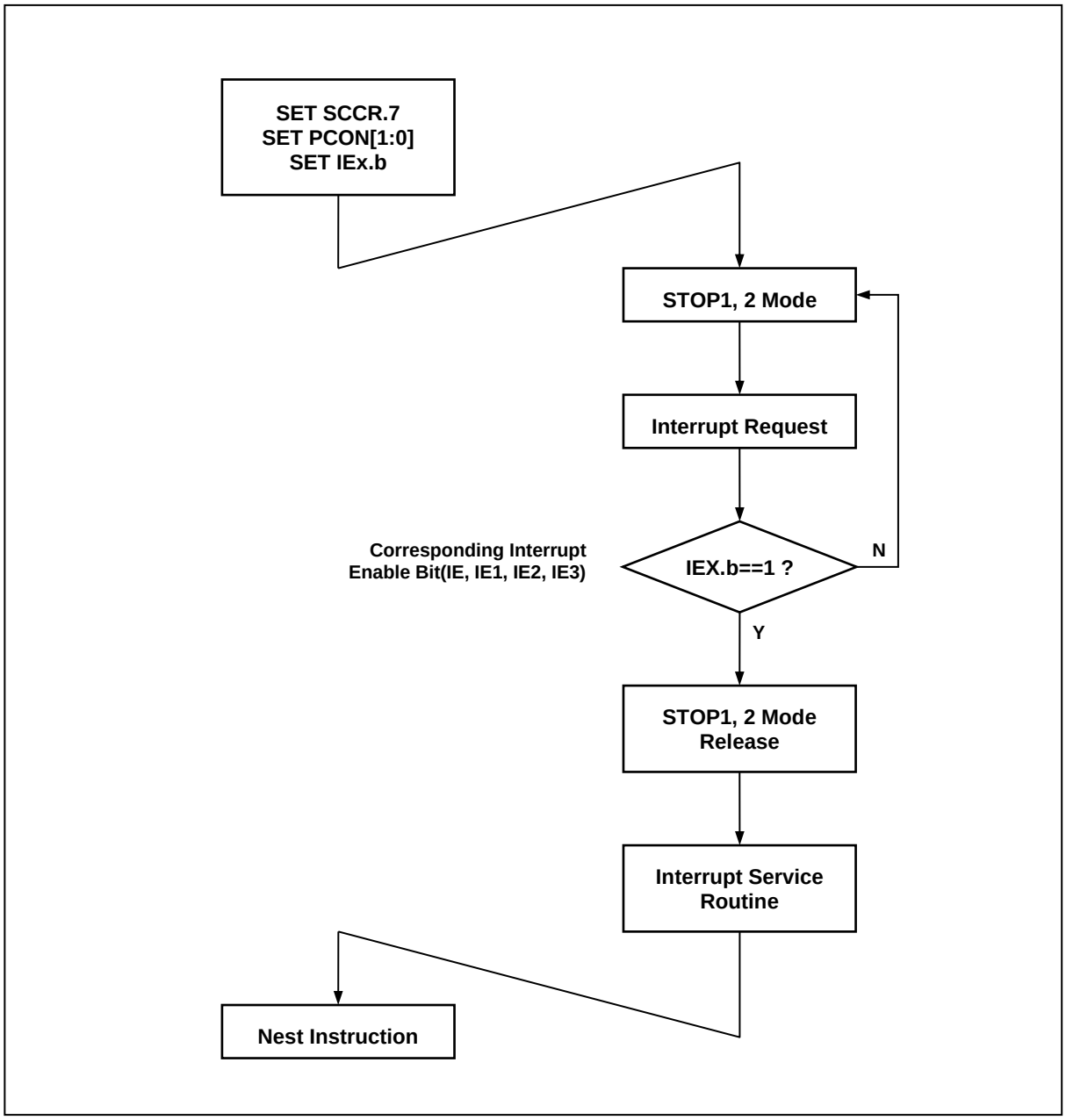


Figure 12-5 STOP1, 2 Mode Release Flow

12.5.1 Register Map

Table 12-2 Register Map

Name	Address	Dir	Default	Description
PCON	87H	R/W	00H	Power Control Register

12.5.2 Power Down Operation Register Description

The Power Down Operation Register consists of the Power Control Register (PCON).

12.5.3 Register Description for Power Down Operation

PCON (Power Control Register) : 87H

7	6	5	4	3	2	1	0
bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

IDLE Mode

01H IDLE mode enable

STOP1, 2 Mode

03H STOP1, 2 mode enable

Note)

1. To enter IDLE mode, PCON must be set to ‘01H’.
2. To STOP1,2 mode, PCON must be set to ‘03H’.
   (In STOP1,2 mode, PCON register is cleared automatically by interrupt or reset)
3. When PCON is set to ‘03H’, if SCCR[7] is set to ‘1’, it enters the STOP1 mode. if SCCR[7] is cleared to ‘0’, it enters the STOP2 mode
4. The different thing in STOP 1,2 is only clock operation of internal 125kHz-OSC during STOP mode operating.

13. RESET

13.1 Overview

The A96T218 has reset by external RESETB pin. The following is the hardware setting value.

Table 13-1 Reset state

On Chip Hardware	Initial Value
Program Counter (PC)	0000h
Accumulator	00h
Stack Pointer (SP)	07h
Peripheral Clock	On
Control Register	Peripheral Registers refer
Brown-Out Detector	Enable

13.2 Reset Source

The A96T218 has five types of reset generation procedures. The following is the reset sources.

- External RESETB
- Power ON RESET (POR)
- WDT Overflow Reset (In the case of WDTEN = `1`)
- BOD Reset (In the case of BODEN = `1 `)
- LVI Reset (In the case of LVI_S_I[1:0] ≠ `0 `)
- OCD Reset

13.3 Block Diagram

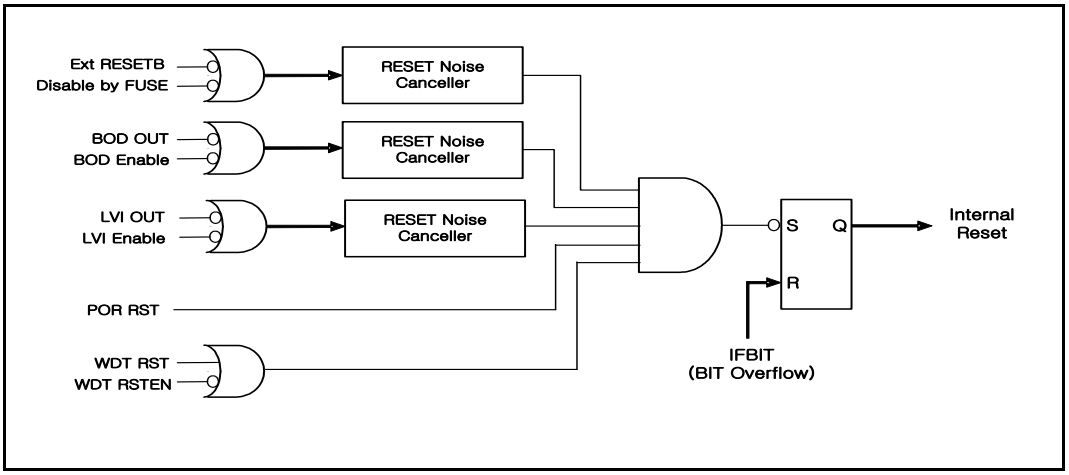


Figure 13-1 RESET Block Diagram

13.4 RESET Noise Canceller

The Figure 13-2 is the Noise canceller diagram for Noise cancel of RESET. It has the Noise cancel value of about 32us (@V_{DD}=5V) to the low input of System Reset.

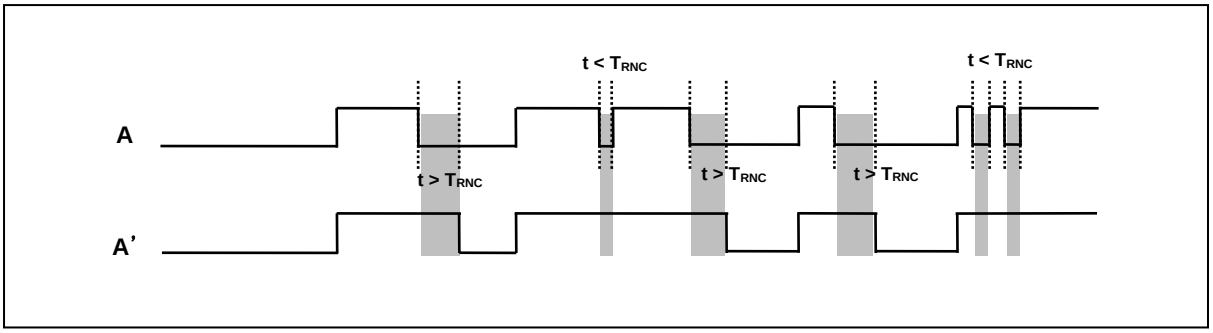


Figure 13-2 Reset Noise Canceller Time Diagram

13.5 Power ON Reset

When rising device power, the POR (Power ON Reset) have a function to reset the device. If using POR, it executes the device RESET function instead of the RESET IC or the RESET circuits. And External RESET PIN is able to use as Normal input pin.

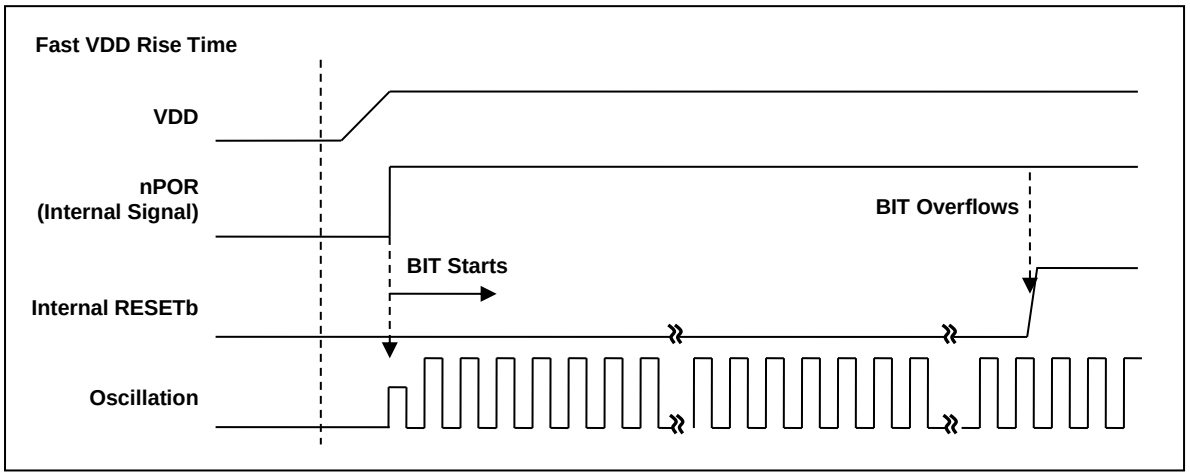


Figure 13-3 Fast VDD Rising Time

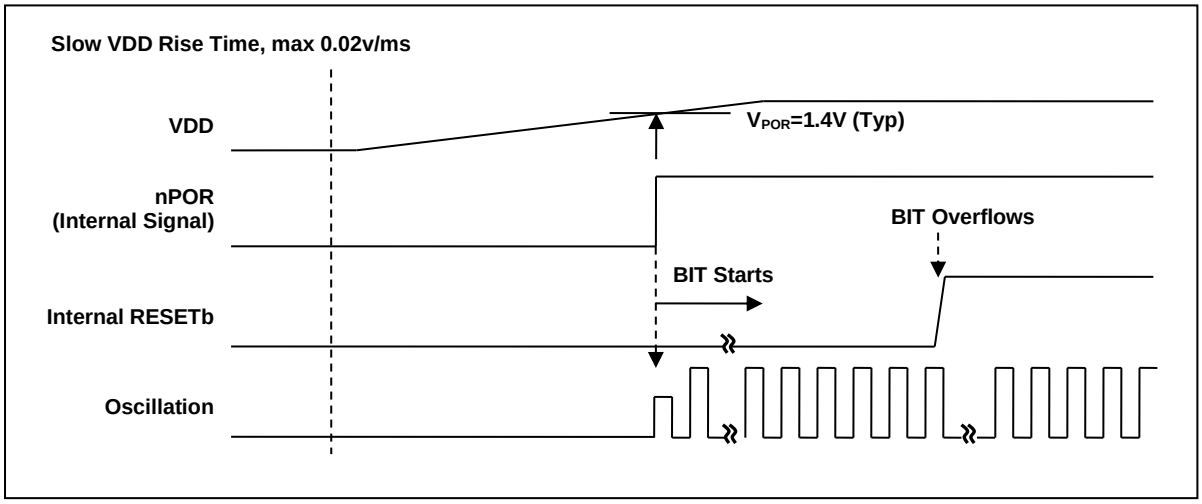


Figure 13-4 Internal RESET Release Timing on Power-Up

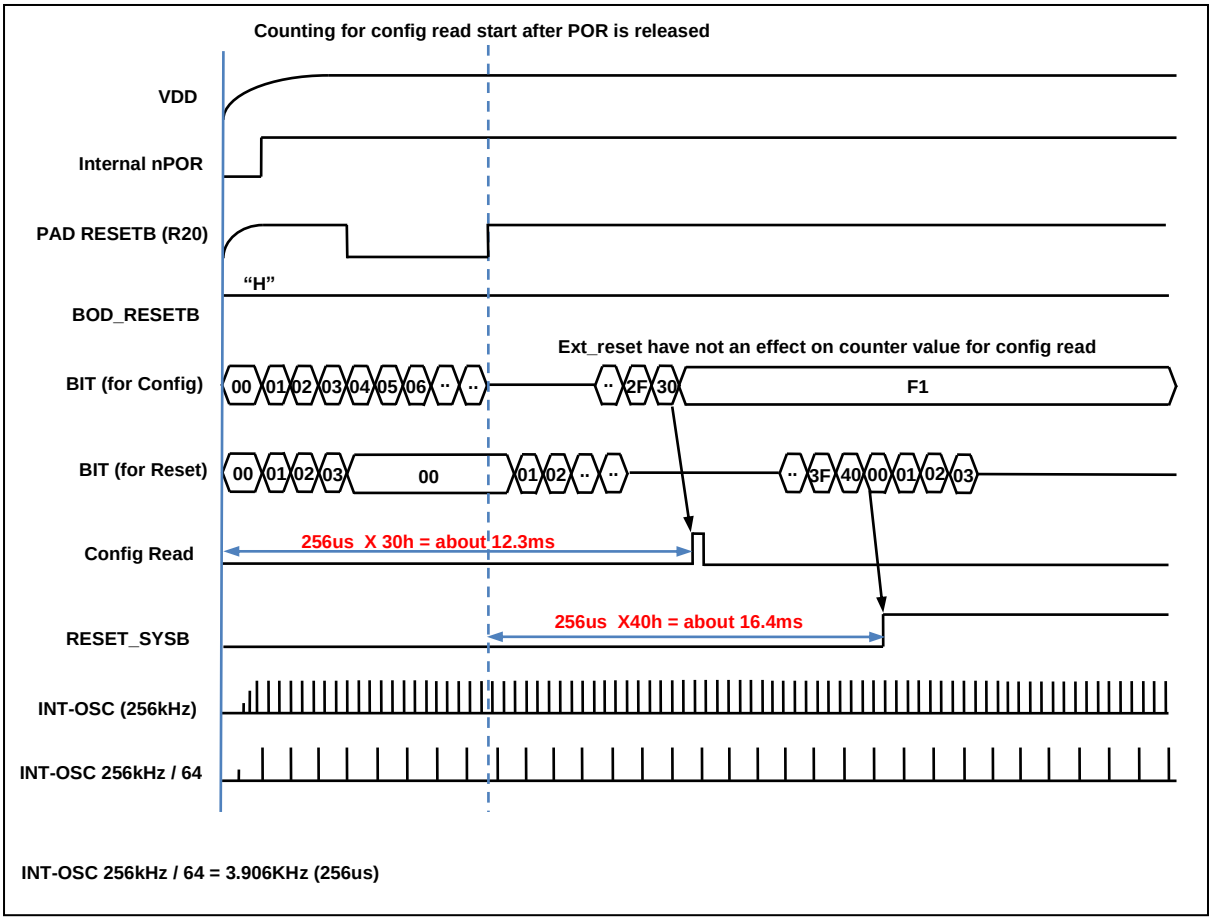


Figure 13-5 Configuration Timing when Power-on

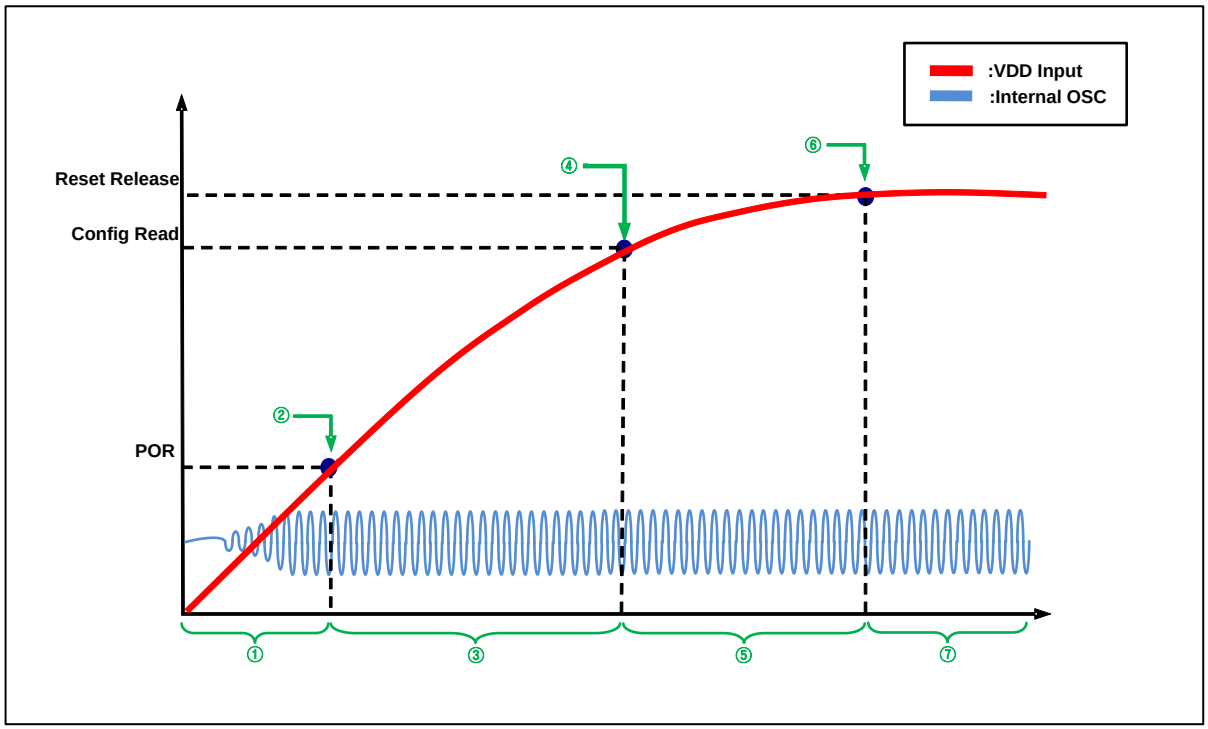


Figure 13-6 Boot Process Waveform

Table 13-2 Boot Process Description

Process	Description	Remarks
①	No operation Internal OSC (256kHz) ON	-about 0.8V
②	1st POR level Detection	-about 1.4V ~ 1.5V
③	(INT-OSC 256kHz/64)×30h Delay section (=12ms) VDD input voltage must rise over than flash operating voltage for Config read	-Slew Rate >= 0.025V/ms
④	Config read point	-about 1.5V ~ 1.6V -Config Value is determined by Writing Option
⑤	Rising section to Reset Release Level	-16ms point after POR or Ext_reset release
⑥	Reset Release section (BIT overflow) i) after16ms, after External Reset Release (External reset) ii) 16ms point after POR (POR only)	-BIT is used for Peripheral stability
⑦	Normal operation	

13.6 External RESETB Input

The External RESETB is the input to a Schmitt trigger. A reset in accomplished by holding the reset pin low for at least 32us over, within the operating voltage range and oscillation stable, it is applied, and the internal state is initialized. After reset state becomes ‘1’, it needs the stabilization time with 16ms and after the stable state, the internal RESET becomes ‘1’. The Reset process step needs 5 oscillator clocks. And the program execution starts at the vector address stored at address 0000H.

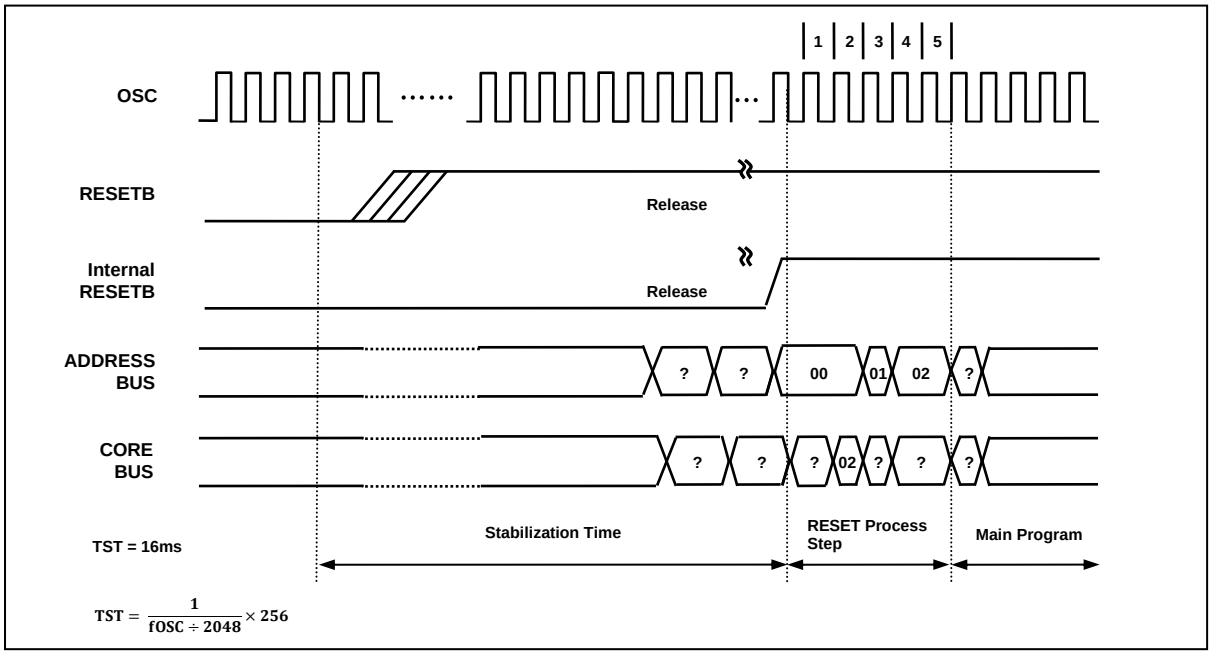


Figure 13-7 Timing Diagram after RESET

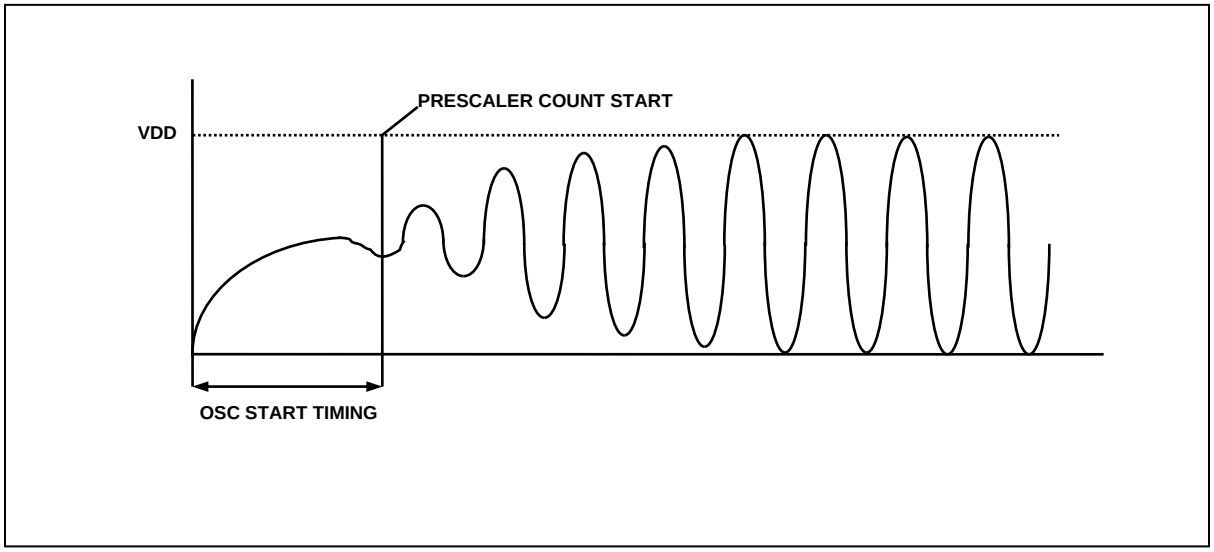


Figure 13-8 Oscillator Generating Waveform Example

Note) as shown Figure 13-8, the stable generating time is not included in the start-up time.

13.7 Brown Out Detector Processor

The A96T218 has an On-chip Brown-out detection and LVI circuit for monitoring the VDD level during operation by comparing it to a fixed trigger level. The trigger level for the BOD and LVI can be selected by BODEN and LVILS[1:0] bit to be 1.6V, 2.5V, 3.6V or 4.2V. In the STOP mode, this will contribute significantly to the total current consumption. So to minimize the current consumption, the BODEN bit is set to off by software.

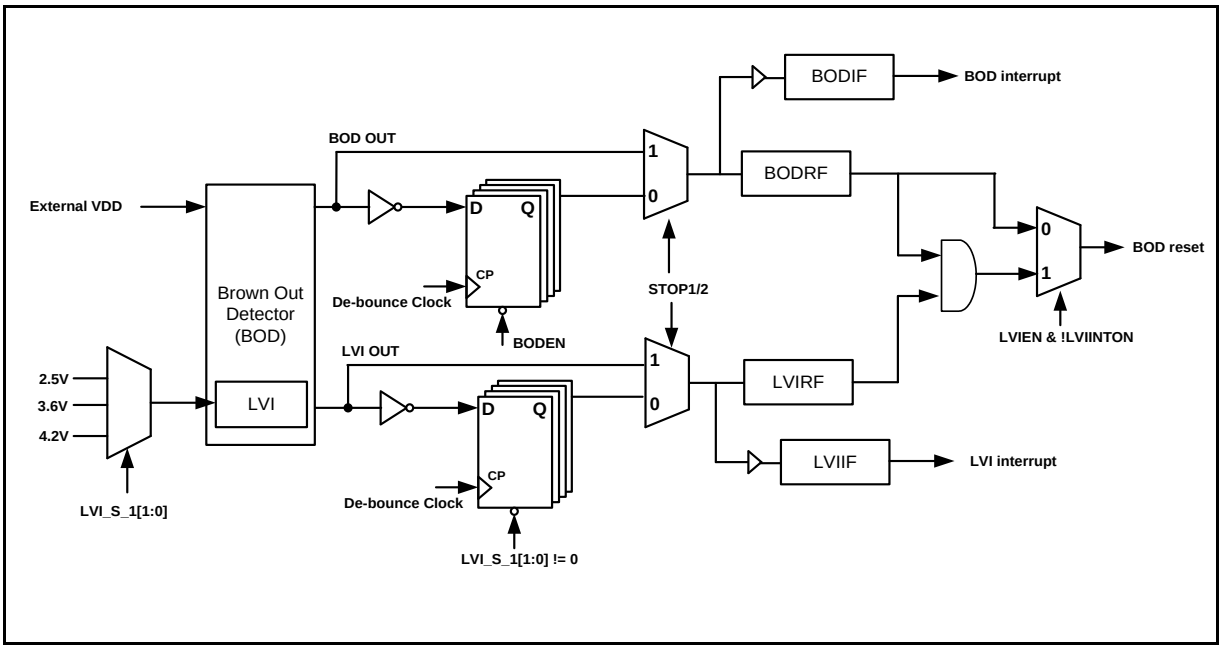


Figure 13-9 Block Diagram of BOD

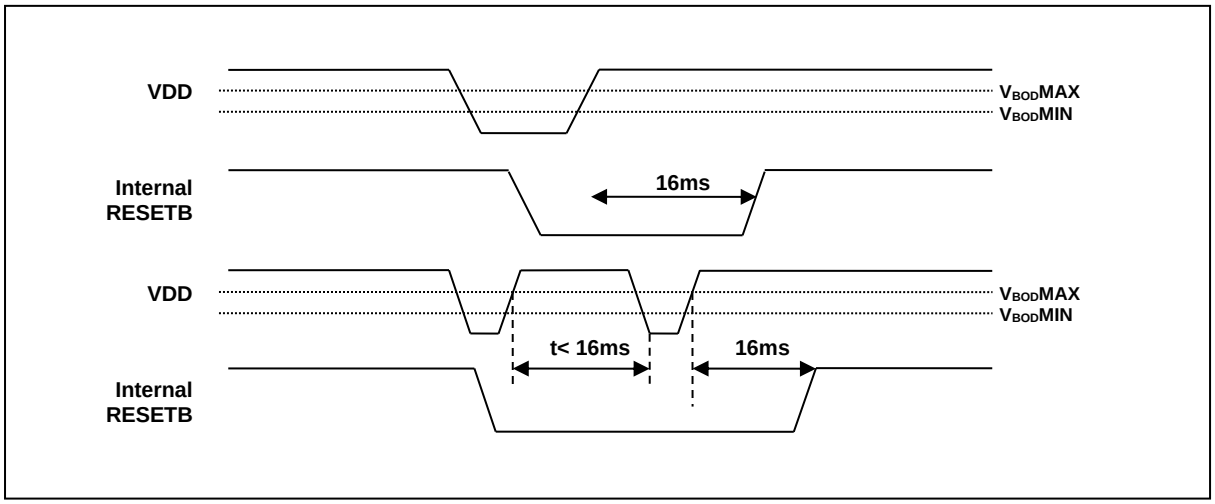


Figure 13-10 Internal Reset at the Power Fail Situation

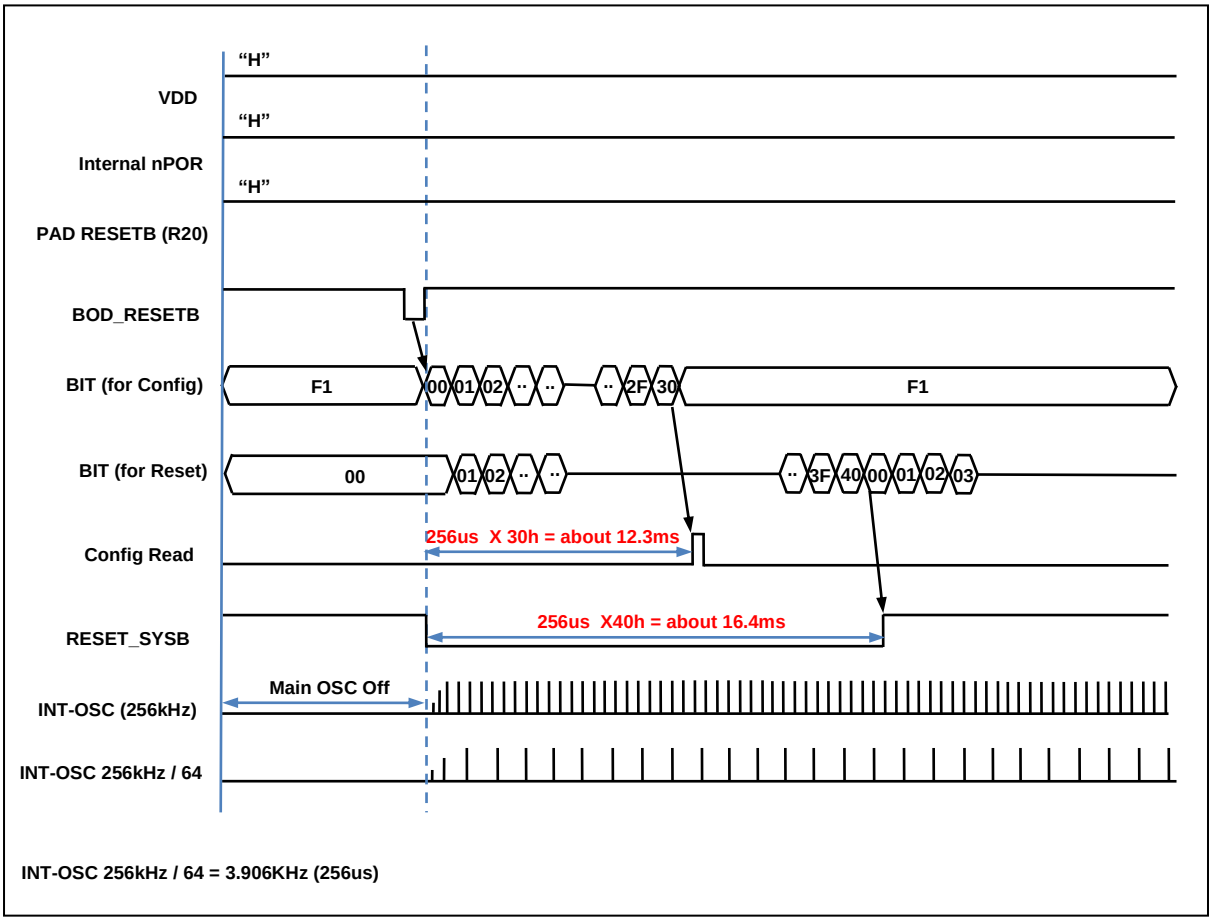


Figure 13-11 Configuration Timing when BOD RESET

13.7.1 Register Map

Table 13-3 Register Map

Name	Address	Dir	Default	Description
BODR	86H	R/W	09H	BOD Control Register
RSFR	8FH	R/W	80H	RESET Source FLAG Register

13.7.2 Reset Operation Register Description

Reset and Interrupt control Register consists of the BOD Control Register (BODR).

13.7.3 Register Description for Reset Operation

BODR (BOD Control Register) : 86H

7	6	5	4	3	2	1	0
-	-	-	-	LVIINTON	LVILS[1]	LVILS[0]	BODEN
-	-	-	-	R/W	R/W	R/W	R/W

Initial value : 09H

LVIINTON Select Interrupt or Reset

0 Reset

1 Interrupt (default)

LVILS[1:0] LVI level Voltage

LVILS[1]	LVILS[0]	Description
0	0	LVI disable (default)
0	1	2.5V
1	0	3.6V
1	1	4.2V

BODEN 1.6V BOD operation

0 BOD disable

1 BOD enable (default)

RSFR (RESET Source FLAG Register) : 8FH

7	6	5	4	3	2	1	0
PORF	EXTRF	WDTRF	OCDRF	BODRF	LVIRF	-	-
R/W	R/W	R/W	R/W	R/W	R/W	-	-

Initial value : 80H

PORF Power-On Reset flag bit. The bit is reset by writing ‘0’ to this bit.

0 No detection

1 Detection

EXTRF External Reset flag bit. The bit is reset by writing ‘0’ to this bit or by Power ON reset.

0 No detection

1 Detection

WDTRF Watch Dog Reset flag bit. The bit is reset by writing ‘0’ to this bit or by Power ON reset.

0 No detection

1 Detection

OCDRF On-Chip Debug Reset flag bit. The bit is reset by writing ‘0’ to this bit or by Power ON reset.

0 No detection

1 Detection

BODRF Brown-Out Reset flag bit. The bit is reset by writing ‘0’ to this bit or by Power ON reset.

0 No detection

1 Detection

LVIRF LVI Reset flag bit. The bit is reset by writing ‘0’ to this bit or by Power ON reset.

0 No detection

1 Detection

14. On-chip Debug System

14.1 Overview

14.1.1 Description

On-chip debug System (OCD) of A96T218 can be used for programming the non-volatile memories and on-chip debugging. Detailed descriptions for programming via the OCD interface can be found in the following chapter.

Figure 14-1shows a block diagram of the OCD interface and the On-chip Debug system.

14.1.2 Feature

- Two-wire external interface: 1-wire serial clock input, 1-wire bi-directional serial data bus
- Debugger Access to:
  - All Internal Peripheral Units
  - Internal data RAM
  - Program Counter
  - Flash Memory
- Extensive On-chip Debug Support for Break Conditions, Including
  - Break Instruction
  - Single Step Break
  - Program Memory Break Points on Single Address
  - Programming of Flash, Fuses, and Lock Bits through the two-wire Interface
  - On-chip Debugging Supported by Dr.Choice®
- Operating frequency

  Supports the maximum frequency of the target MCU

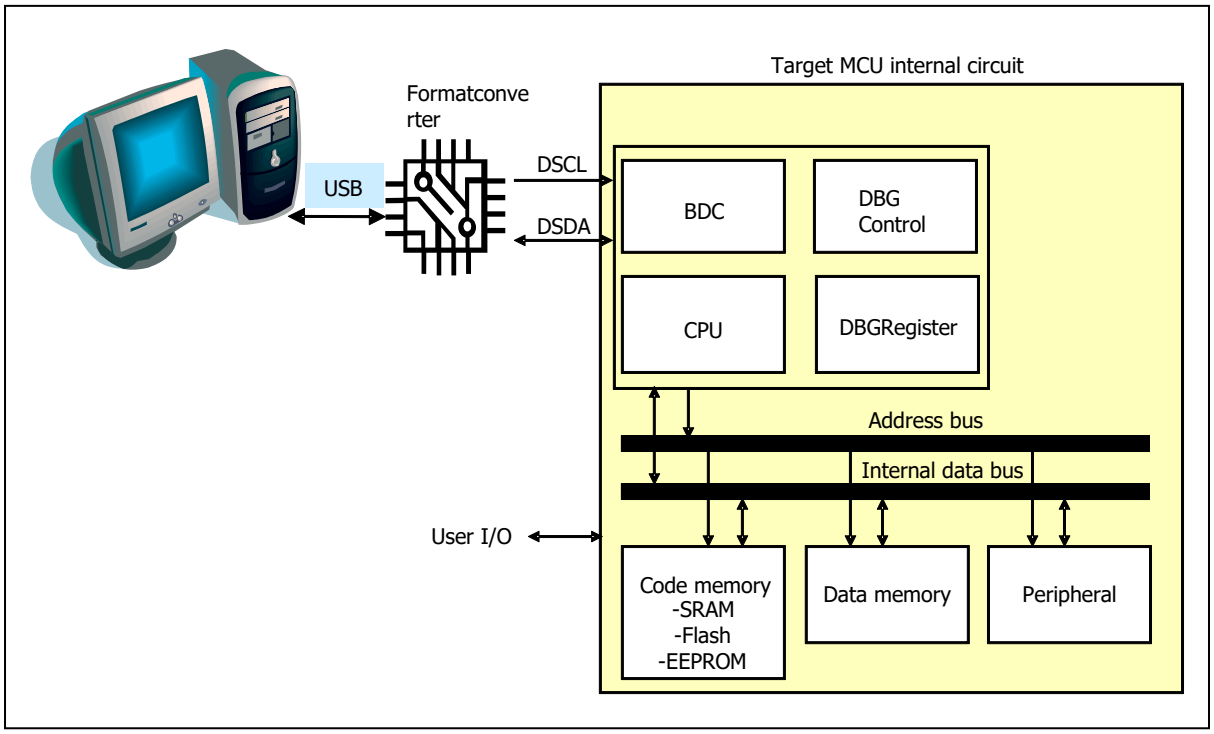


Figure 14-1 Block Diagram of On-chip Debug System

14.2 Two-pin External Interface

14.2.1 Basic Transmission Packet

- 10-bit packet transmission using two-pin interface.
- 1-packet consists of 8-bit data, 1-bit parity and 1-bit acknowledge.
- Parity is even of ‘1’ for 8-bit data in transmitter.
- Receiver generates acknowledge bit as ‘0’ when transmission for 8-bit data and its parity has no error.
- When transmitter has no acknowledge (Acknowledge bit is ‘1’ at tenth clock), error process is executed in transmitter.
- When acknowledge error is generated, host PC makes stop condition and transmits command which has error again.
- Background debugger command is composed of a bundle of packet.
- Star condition and stop condition notify the start and the stop of background debugger command respectively.

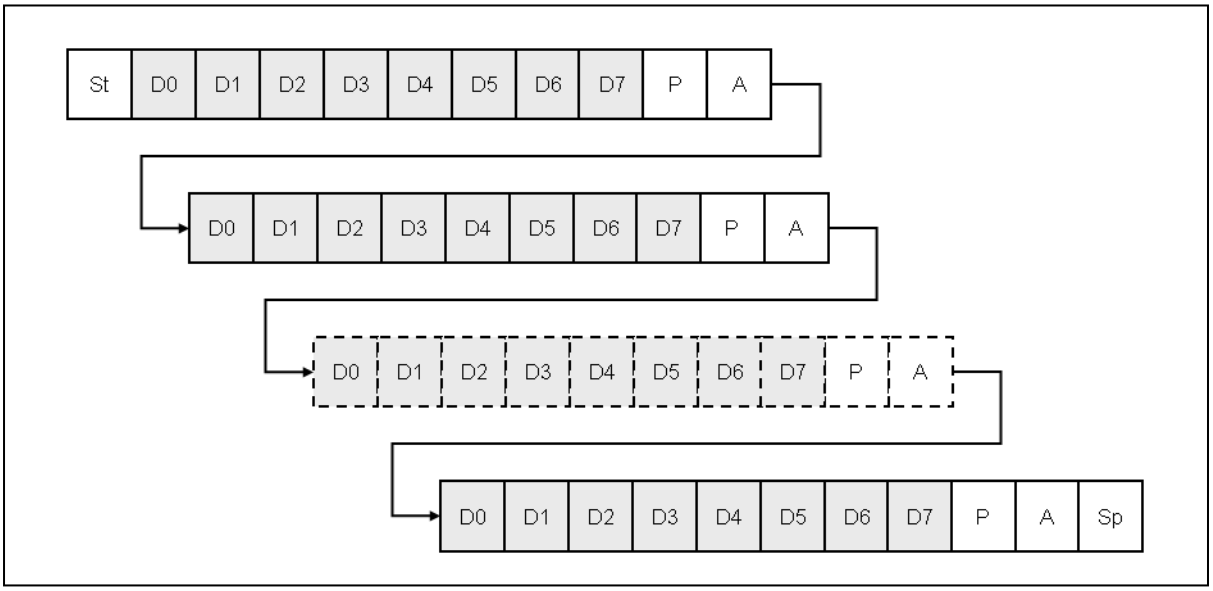


Figure 14-2 10-bit Transmission Packet

14.2.2 Packet Transmission Timing

14.2.2.1 Data transfer

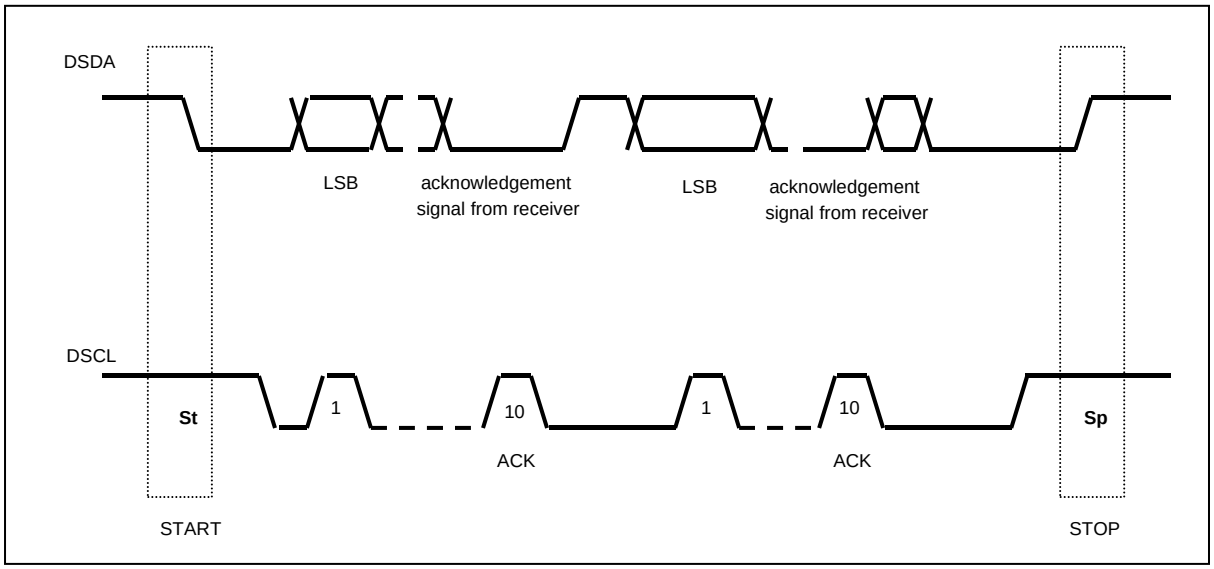


Figure 14-3 Data Transfer on the Twin Bus

14.2.2.2 Bit transfer

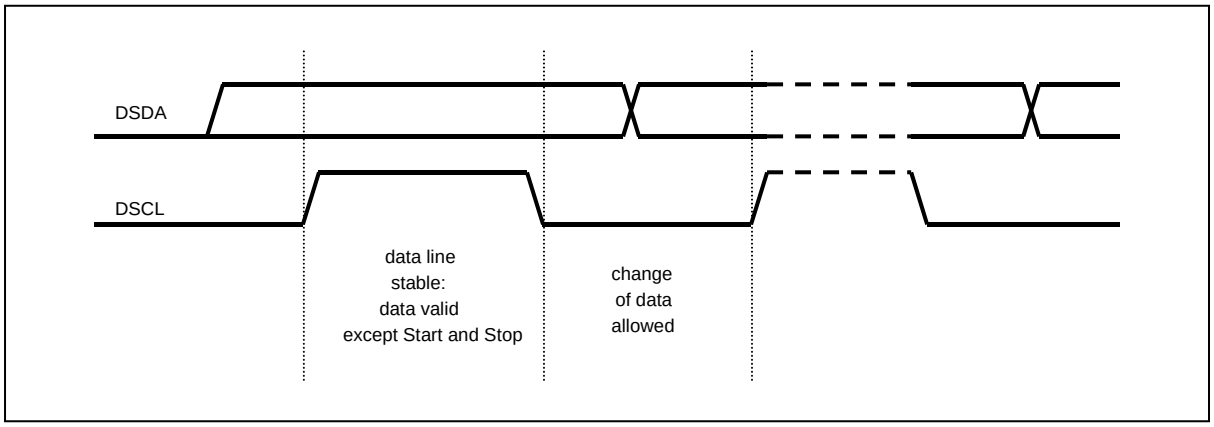


Figure 14-4 Bit Transfer on the Serial Bus

14.2.2.3 Start and stop condition

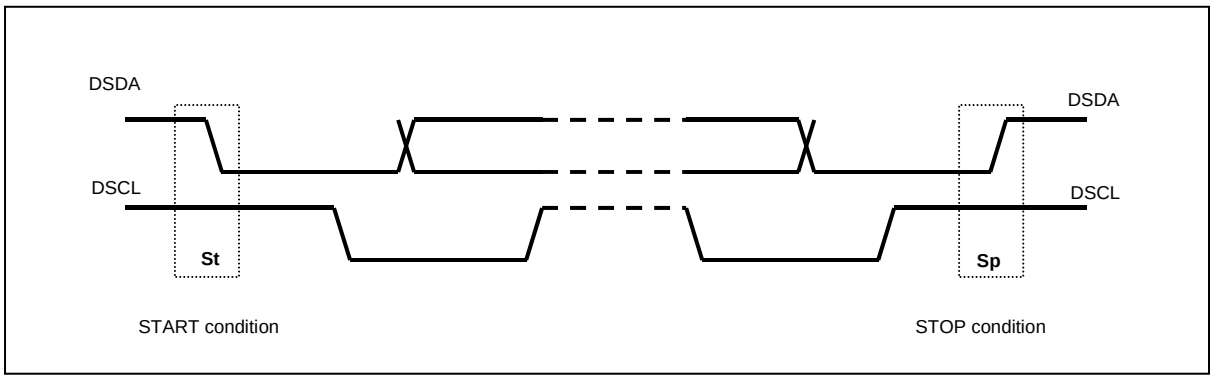


Figure 14-5 Start and Stop Condition

14.2.2.4 Acknowledge bit

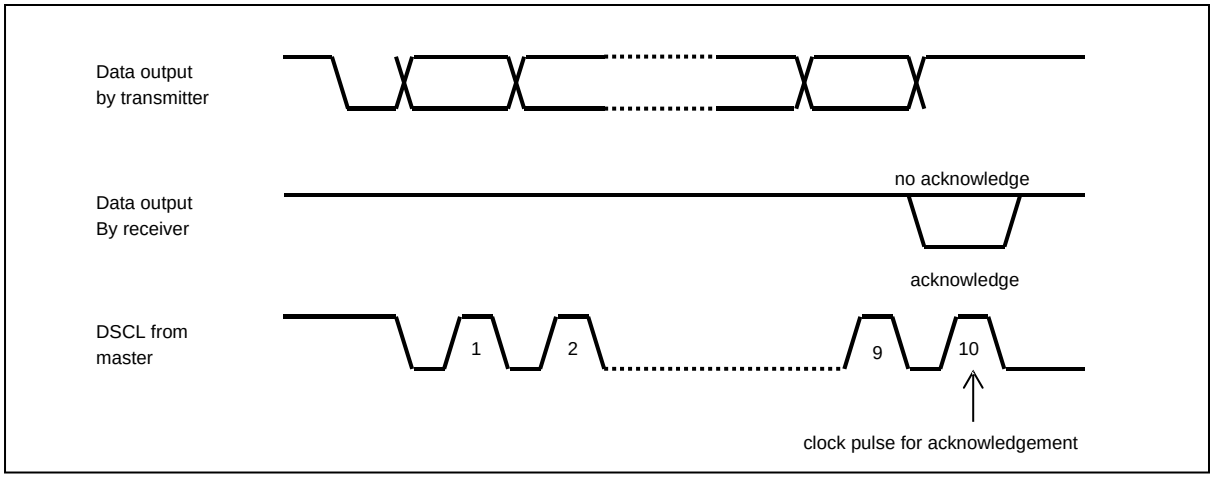


Figure 14-6 Acknowledge on the Serial Bus

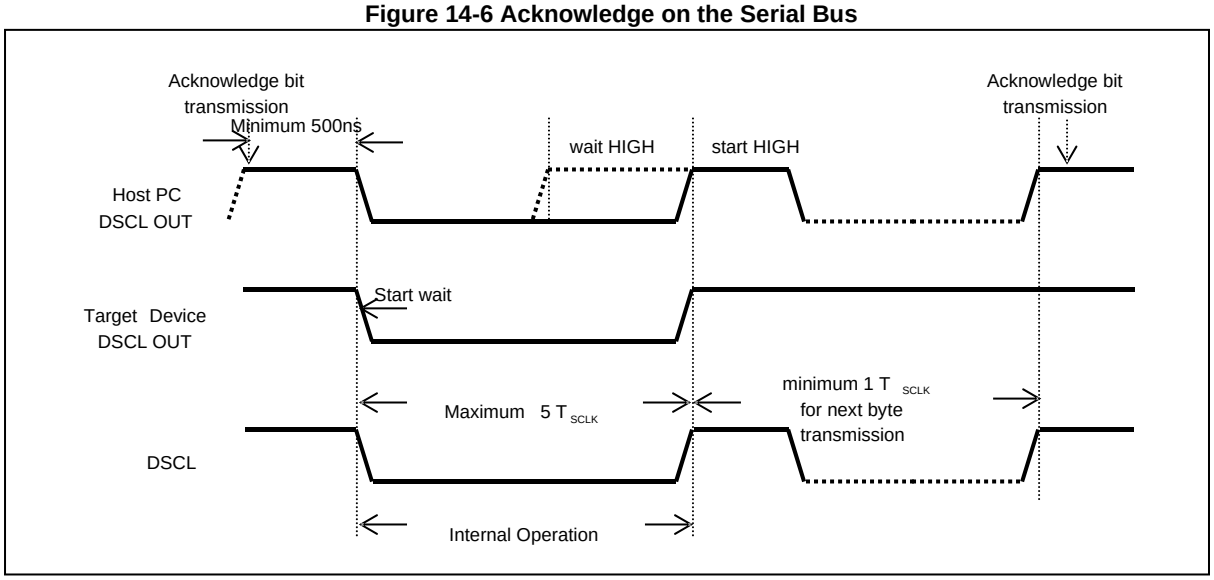


Figure 14-7 Clock Synchronization during Wait Procedure

14.2.3 Connection of Transmission

Two-pin interface connection uses open-drain (wire-AND bidirectional I/O).

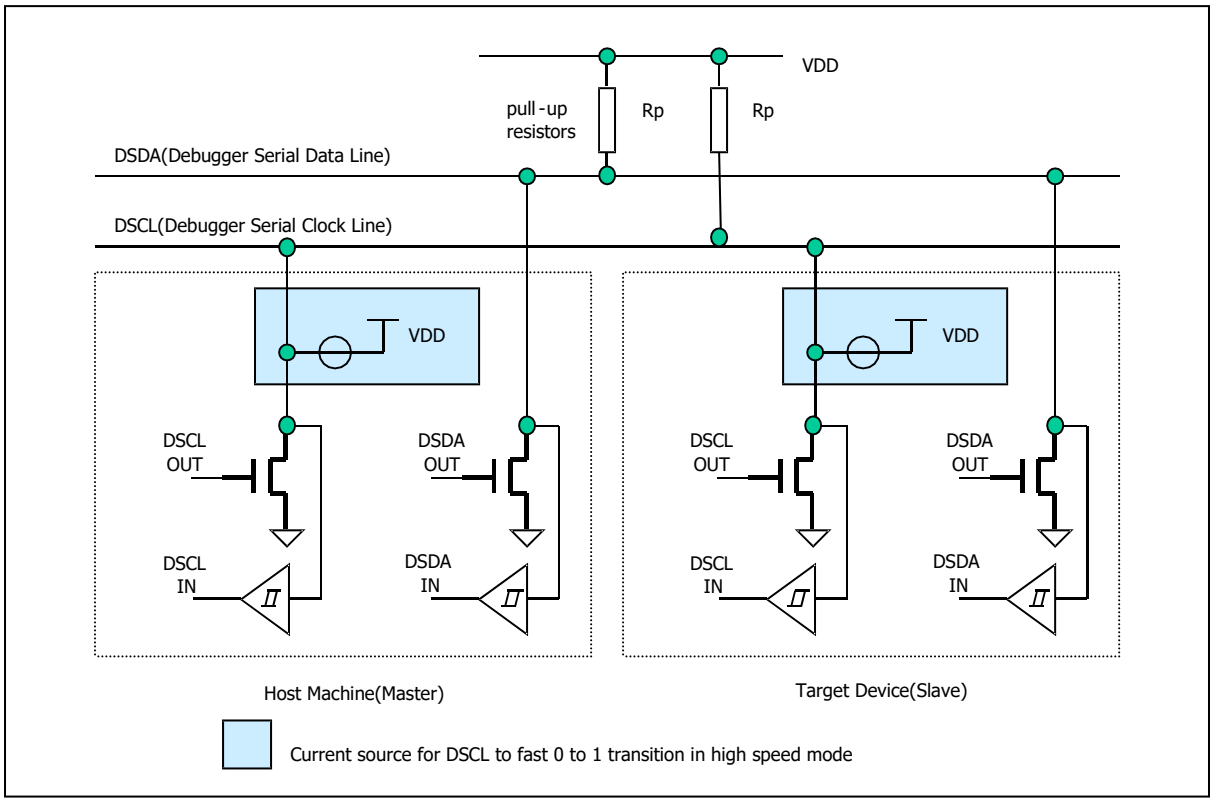


Figure 14-8 Connection of Transmission

15. Memory Programming

15.1 Overview

15.1.1 Description

A96T218 has flash memory to which a program can be written, erased, and overwritten while mounted on the board. Serial ISP mode is supported.

15.1.2 Features

- Flash Size : 32Kbytes
- Single power supply program and erase
- Command interface for fast program and erase operation
- Up to 10,000 program/erase cycles at typical voltage and temperature for flash memory
- Security feature

15.2 Flash Control and Status Register

Registers to control Flash and Data EEPROM are Mode Register (FEMR), Control Register (FECR), Status Register (FESR), Time Control Register (FETCR), Address Low Register x (FEARLx), Address Middle Register x (FEARMx), address High Register (FEARH). They are mapped to SFR area and can be accessed only in programming mode.

15.2.1 Register Map

Table 15-1 Register Map

Name	Address	Dir	Default	Description
FEMR	EAH	R/W	00H	Flash Mode Register
FECR	EBH	R/W	03H	Flash Control Register
FESR	ECH	R/W	80H	Flash Status Register
FETCR	EDH	R/W	00H	Flash Time Control Register
FEARL1	EEH	R/W	00H	Flash Address Low Register 1
FEARM1	EFH	R/W	00H	Flash Address Middle Register 1
FEARL	F2H	R/W	00H	Flash Address Low Register
FEARM	F3H	R/W	00H	Flash Address Middle Register
FEARH	F4H	R/W	00H	Flash Address High Register

15.2.2 Register Description for Flash

FEMR (Flash Mode Register) : EAH

7	6	5	4	3	2	1	0
FSEL	-	PGM	ERASE	PBUFF	OTPE	VFY	FEEN
R/W	-	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

FSEL Select flash memory.
- 0 Deselect flash memory
- 1 Select flash memory

PGM Enable program or program verify mode with VFY
- 0 Disable program or program verify mode
- 1 Enable program or program verify mode

ERASE Enable erase or erase verify mode with VFY
- 0 Disable erase or erase verify mode
- 1 Enable erase or erase verify mode

PBUFF Select page buffer
- 0 Deselect page buffer
- 1 Select page buffer

OTPE Select OTP area instead of program memory
- 0 Deselect OTP area
- 1 Select OTP area

VFY Set program or erase verify mode with PGM or ERASE

Program Verify: PGM=1, VFY=1

Erase Verify: ERASE=1, VFY=1

FEEN Enable program and erase of Flash. When inactive, it is possible to read as normal mode
- 0 Disable program and erase
- 1 Enable program and erase

FECR (Flash Control Register) : EBH

7	6	5	4	3	2	1	0
AEF	-	EXIT1	EXIT0	WRITE	READ	nFERST	nPBRST
R/W	-	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 03H

AEF Enable flash bulk erase mode
- 0 Disable bulk erase mode of Flash memory
- 1 Enable bulk erase mode of Flash memory

EXIT[1:0] Exit from program mode. It is cleared automatically after 1 clock

EXIT1	EXIT0	Description
0	0	Don’t exit from program mode
0	1	Don’t exit from program mode
1	0	Don’t exit from program mode
1	1	Exit from program mode

WRITE Start to program or erase of Flash. It is cleared automatically after 1 clock
- 0 No operation
- 1 Start to program or erase of Flash

READ Start auto-verify of Flash. It is cleared automatically after 1 clock

0 No operation

1 Start auto-verify of Flash (Checksum or CRC16)

nFERST Reset Flash control logic. It is set automatically after 1 clock

0 Reset Flash control logic

1 No operation (default)

nPBRST Reset page buffer with PBUFF. It is set automatically after 1 clock

PBUFF	nPBRST	Description
0	0	Page buffer reset
1	0	Page buffer select register reset
X	1	No operation (default)

WRITE and READ bits can be used in program, erase and verify mode with FEAR registers. Read or writes for memory cell or page buffer uses read and write enable signals from memory controller. Indirect address mode with FEAR is only allowed to program, erase and verify

FESR (Flash Status Register) : ECH

7	6	5	4	3	2	1	0
PEVBSY	-	-	-	ROMINT	WMODE	EMODE	VMODE
R	R	R	R	R/W	R	R	R

Initial value : 80H

PEVBSY Operation status flag. It is cleared automatically when operation starts. Operations are program, erase or verification

0 Busy (Operation processing)

1 Complete Operation

ROMINT Flash interrupt request flag. Auto-cleared when program/erase/verify starts. Active in program/erase/verify completion

0 No interrupt request.

1 Interrupt request.

WMODE Write mode flag

EMODE Erase mode flag

VMODE Verify mode flag

FEARL1 (Flash address low Register 1) : EEH

7	6	5	4	3	2	1	0
ARL17	ARL16	ARL15	ARL14	ARL13	ARL12	ARL11	ARL10
W	W	W	W	W	W	W	W

Initial value : 00H

ARL1[7:0] Flash address low 1

FEARM1 (Flash address middle Register 1) : EFH

7	6	5	4	3	2	1	0
ARM17	ARM16	ARM15	ARM14	ARM13	ARM12	ARM11	ARM10
W	W	W	W	W	W	W	W

Initial value : 00H

ARM1[7:0] Flash address middle 1

FEARL (Flash address low Register) : F2H

7	6	5	4	3	2	1	0
ARL7	ARL6	ARL5	ARL4	ARL3	ARL2	ARL1	ARL0
W	W	W	W	W	W	W	W

Initial value : 00H

ARL[7:0] Flash address low

FEARM (Flash address middle Register) : F3H

7	6	5	4	3	2	1	0
ARM7	ARM6	ARM5	ARM4	ARM3	ARM2	ARM1	ARM0
W	W	W	W	W	W	W	W

Initial value : 00H

ARM[7:0] Flash address middle

FEARH (Flash address high Register) : F4H

7	6	5	4	3	2	1	0
ARH7	ARH6	ARH5	ARH4	ARH3	ARH2	ARH1	ARH0
W	W	W	W	W	W	W	W

Initial value : 00H

ARH[7:0] Flash address high

FEAR registers are used for program, erase and auto-verify. In program and erase mode, it is page address and ignored the same least significant bits as the number of bits of page address. In auto-verify mode, address increases automatically by one.

FEARs are write-only register. Reading these registers returns 24-bit checksum result.

This device can support internal CheckSum calculation, device verification time will be decreased dramatically.

CheckSum cannot detect error address or error bit, but it is quite good feature in mass product programming.

Device data read out time takes few seconds. However, you can read out device CheckSum within 10’s ~ 100’s of milliseconds. It is 100’s times faster than normal data read.

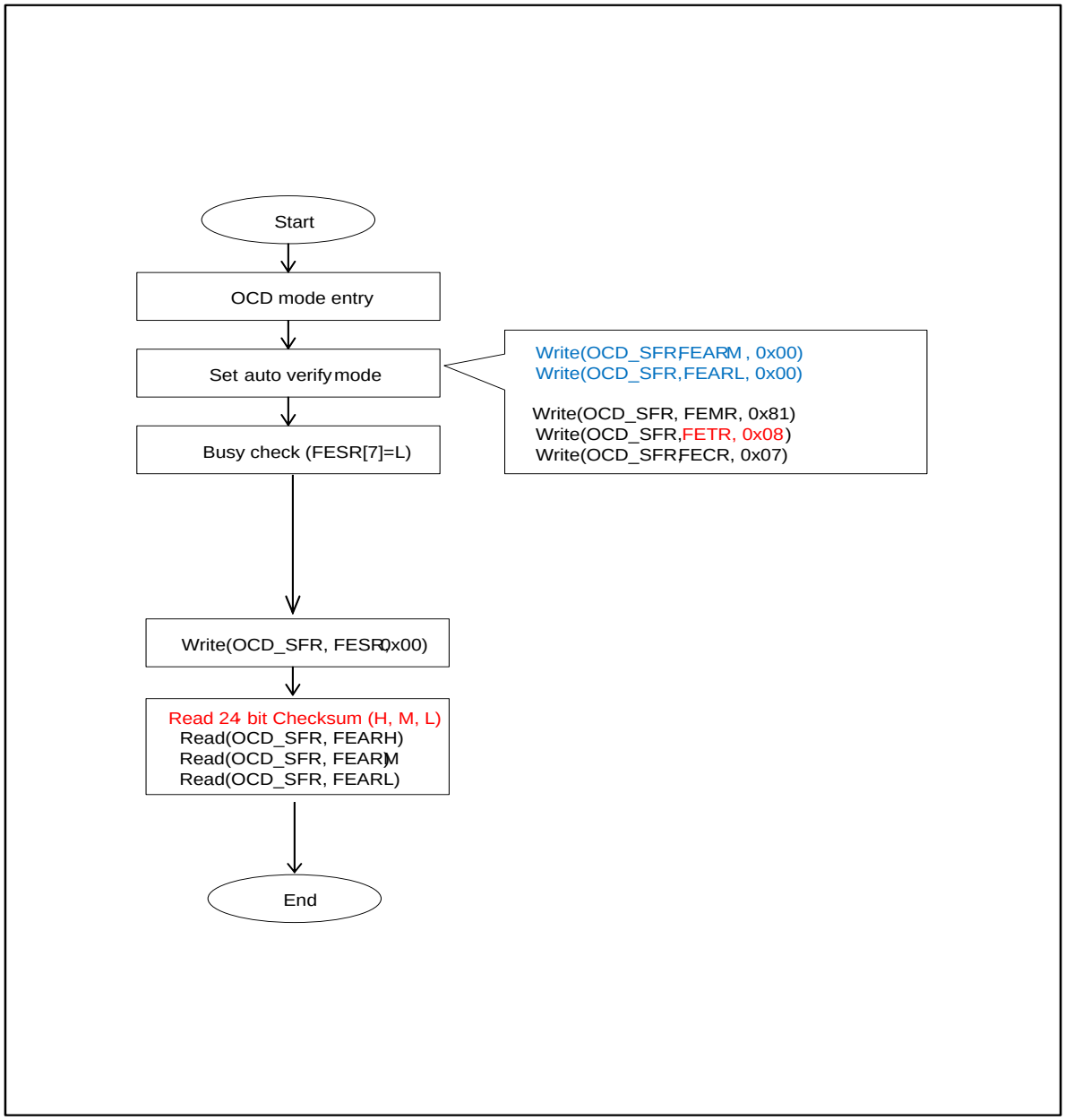


Figure 15-1 Read device internal CkeckSum

FETCR (Flash Time control Register) : EDH

7	6	5	4	3	2	1	0
TCR7	TCR6	TCR5	TCR4	TCR3	TCR2	TCR1	TCR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value : 00H

TCR[7:0] Flash Time control

Program and erase time is controlled by setting FETCR register. Program and erase timer uses 10-bit counter. It increases by one at each INTRC/256 clock frequency ($f_{INTRC/256}$=62.5kHz). It is cleared when program or erase starts. Timer stops when 10-bit counter is same to FETCR. PEVBSY is cleared when program, erase or verify starts and set when program, erase or verify stops.

Max program/erase time at INTRC/256 clock : (255+1) * 2 * (16.0us) = 8.192ms

In the case of ±3% of error rate of counter source clock, program or erase time is 7.94~8.44ms

* Program/erase time calculation

for page write or erase, Tpe = (TCON+1) * 2 * ($f_{INTRC/256}$)

Recommended bulk erase time : FETCR = 80h

Recommended program / page erase time : FETCR = 50h

Table 15-2 Program/erase Time

	Min	Typ	Max	Unit
program/erase Time	2.4	2.5	2.6	ms

15.3 Memory Map

15.3.1 Flash Memory Map

Program memory uses 32K bytes of Flash memory. It is read by byte and written by byte or page. One page is 64-bytes

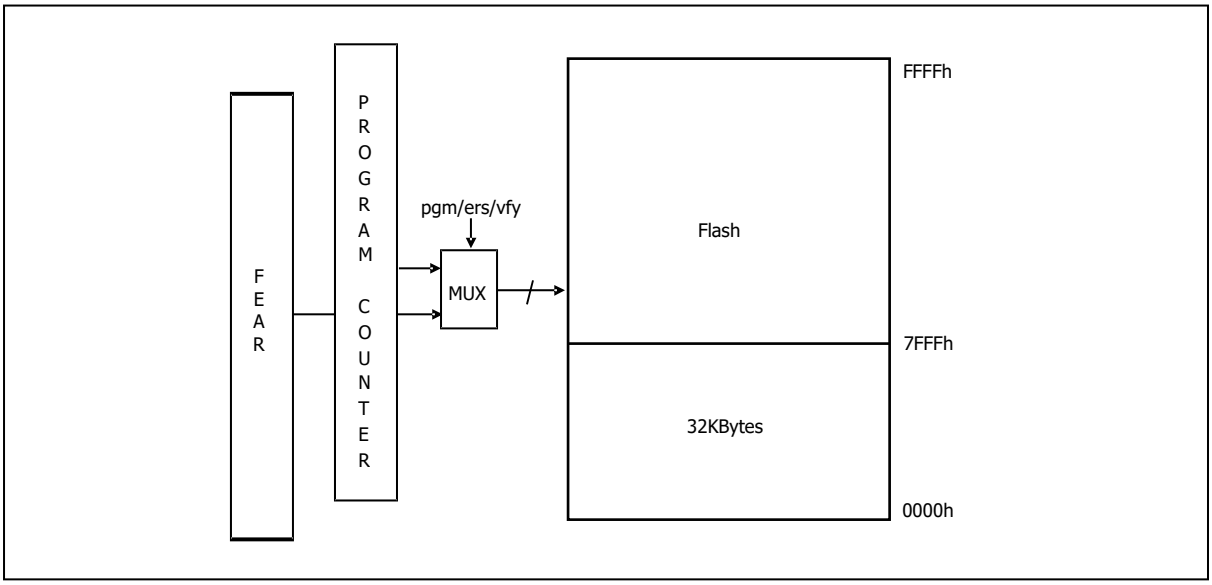


Figure 15-2 Flash Memory Map

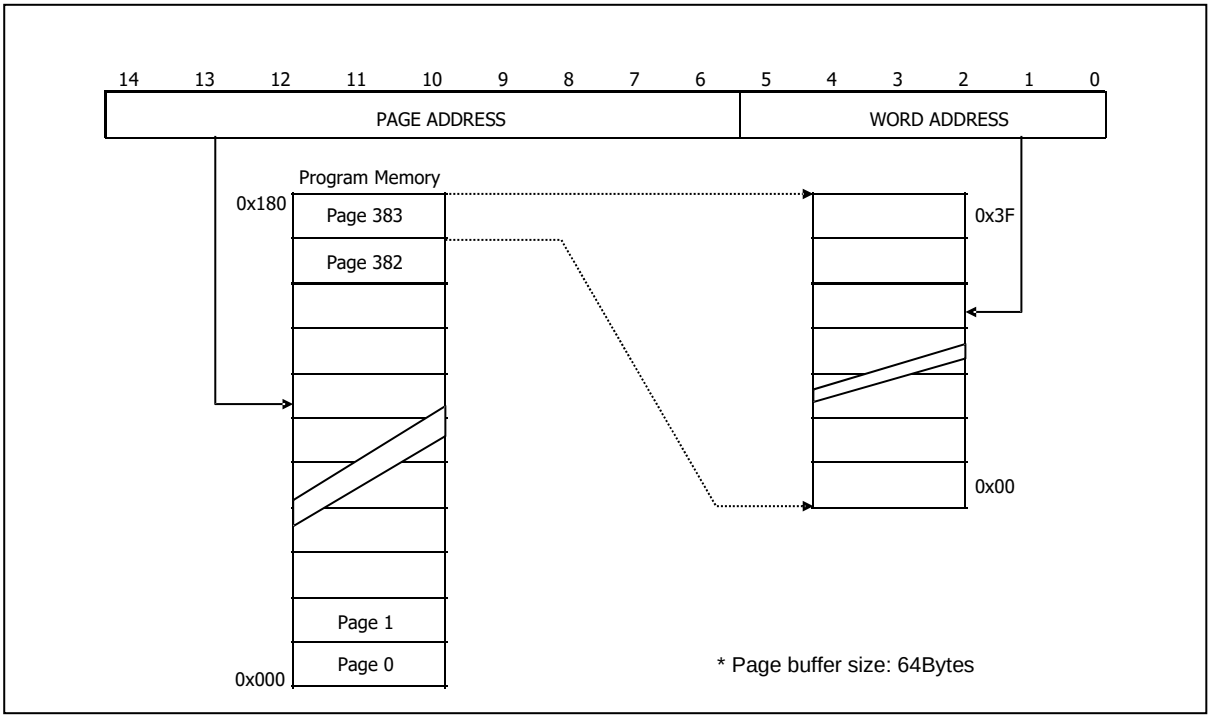


Figure 15-3 Address Configuration of Flash Memory

15.4 Serial In-System Program Mode

Serial in-system program uses the interface of debugger which uses two wires. Refer to chapter 14 in details about debugger

15.4.1 Flash Operation

Configuration(This Configuration is just used for follow description)

7	6	5	4	3	2	1	0
-	FEMR[4]&[1]	FEMR[5]&[1]	-	-	FEMR[2]	FECR[6]	FECR[7]
-	ERASE&VFY	PGM&VFY	-	-	OTPE	AEE	AEF

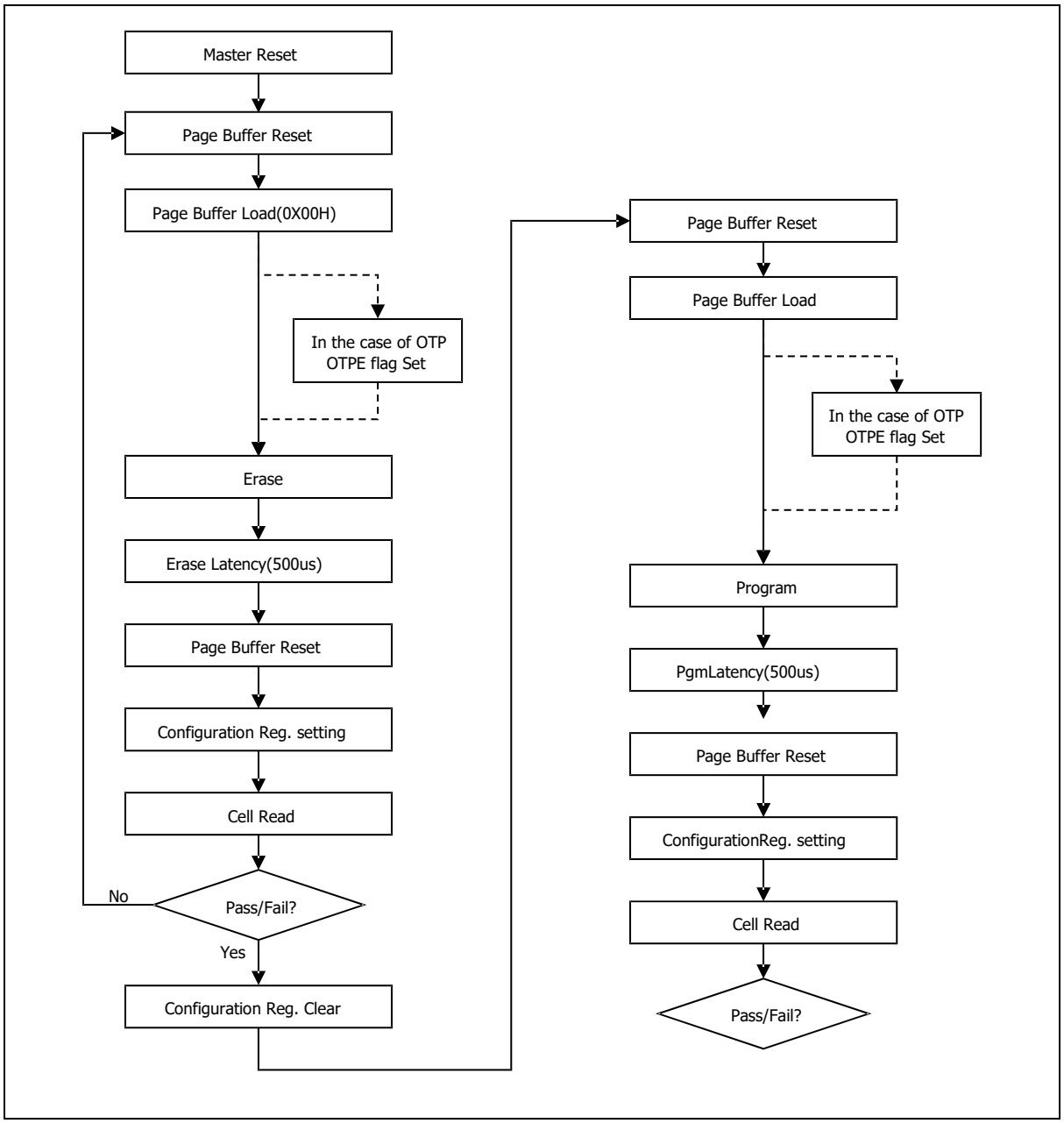


Figure 15-4 The Sequence of Page Program and Erase of Flash Memory

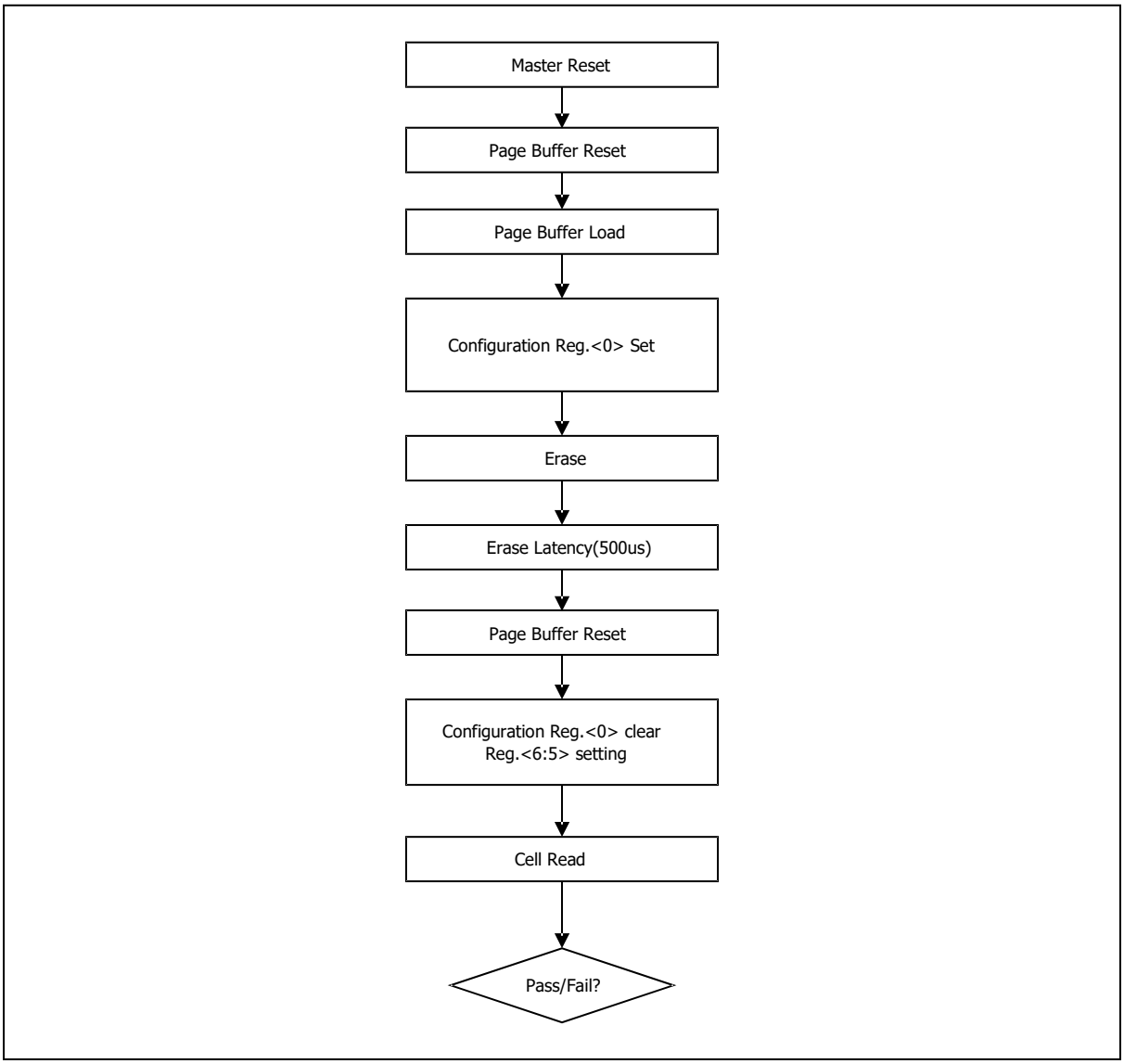


Figure 15-5 The Sequence of Bulk Erase of Flash Memory

15.4.1.1 Flash Read

Step 1. Enter OCD(=ISP) mode.
Step 2. Set ENBDM bit of BCR.
Step 3. Enable debug and Request debug mode.
Step 4. Read data from Flash.

15.4.1.2 Enable program mode

Step 1. Enter OCD(=ISP) mode.[1]
Step 2. Set ENBDM bit of BCR.
Step 3. Enable debug and Request debug mode.
Step 4. Enter program/erase mode sequence.[2]
(1) Write 0xAA to 0xF555.
(2) Write 0x55 to 0xFAAA.

(3) Write 0xA5 to 0xF555.

[1] Refer to how to enter ISP mode..

[2] Command sequence to activate Flash write/erase mode. It is composed of sequentially writing data of Flash memory.

15.4.1.3 Flash write mode

Step 1. Enable program mode.
Step 2. Reset page buffer. FEMR: 1000_0001 FECR:0000_0010
Step 3. Select page buffer. FEMR:1000_1001
Step 4. Write data to page buffer.(Address automatically increases by twin.)
Step 5. Set write mode. FEMR:1010_0001
Step 6. Set page address. FEARH:FEARM:FEARL=20'hx_xxxx
Step 7. Set FETCR.
Step 8. Start program. FECR:0000_1011
Step 9. Insert one NOP operation
Step 10. Read FESR until PEVBSY is 1.
Step 11. Repeat step2 to step 8 until all pages are written.

15.4.1.4 Flash page erase mode

Step 1. Enable program mode.
Step 2. Reset page buffer. FEMR: 1000_0001 FECR:0000_0010
Step 3. Select page buffer. FEMR:1000_1001
Step 4. Write 'h00 to page buffer. (Data value is not important.)
Step 5. Set erase mode. FEMR:1001_0001
Step 6. Set page address. FEARH:FEARM:FEARL=20'hx_xxxx
Step 7. Set FETCR.
Step 8.Start erase. FECR:0000_1011
Step 9. Insert one NOP operation
Step 10. Read FESR until PEVBSY is 1.
Step 11. Repeat step2 to step 8 until all pages are erased.

15.4.1.5 Flash bulk erase mode

Step 1. Enable program mode.
Step 2. Reset page buffer. FEMR: 1000_0001 FECR:0000_0010
Step 3. Select page buffer. FEMR:1000_1001
Step 4. Write 'h00 to page buffer. (Data value is not important.)
Step 5. Set erase mode. FEMR:1001_0001.
(Only main cell area is erased. For bulk erase including OTP area, select OTP area.(set FEMR to 1000_1101.)
Step 6. Set FETCR
Step 7. Start bulk erase. FECR:1000_1011
Step 8. Insert one NOP operation

Step 9. Read FESR until PEVBSY is 1.

15.4.1.6 Flash OTP area read mode

Step 1. Enter OCD(=ISP) mode.
Step 2. Set ENBDM bit of BCR.
Step 3. Enable debug and Request debug mode.
Step 4. Select OTP area. FEMR:1000_0101
Step 5. Read data from Flash.

15.4.1.7 Flash OTP area write mode

Step 1. Enable program mode.
Step 2. Reset page buffer. FEMR: 1000_0001 FECR:0000_0010
Step 3. Select page buffer. FEMR:1000_1001
Step 4. Write data to page buffer.(Address automatically increases by twin.)
Step 5. Set write mode and select OTP area. FEMR:1010_0101
Step 6. Set page address. FEARH:FEARM:FEARL=20’hx_xxxx
Step 7. Set FETCR.
Step 8. Start program. FECR:0000_1011
Step 9. Insert one NOP operation
Step 10. Read FESR until PEVBSY is 1.

15.4.1.8 Flash OTP area erase mode

Step 1. Enable program mode.
Step 2. Reset page buffer. FEMR: 1000_0001 FECR:0000_0010
Step 3. Select page buffer. FEMR:1000_1001
Step 4. Write ‘h00 to page buffer. (Data value is not important.)
Step 5. Set erase mode and select OTP area. FEMR:1001_0101
Step 6. Set page address. FEARH:FEARM:FEARL=20’hx_xxxx
Step 7. Set FETCR.
Step 8.Start erase. FECR:0000_1011
Step 9. Insert one NOP operation
Step 10. Read FESR until PEVBSY is 1.

15.4.1.9 Flash program verify mode

Step 1. Enable program mode.
Step 2. Set program verify mode. FEMR:1010_0011
Step 3. Read data from Flash.

15.4.1.10 OTP program verify mode

Step 1. Enable program mode.
Step 2. Set program verify mode. FEMR:1010_0111
Step 3. Read data from Flash.

15.4.1.11 Flash erase verify mode

Step 1. Enable program mode.
Step 2. Set erase verify mode. FEMR:1001_0011
Step 3. Read data from Flash.

15.4.1.12 Flash page buffer read

Step 1. Enable program mode.
Step 2. Select page buffer. FEMR:1000_1001
Step 3. Read data from Flash.

15.4.2 Summary of Flash Program/Erase Mode

Table 15-3 Operation Mode

Operation mode		Description
FLASH	Flash read	Read cell by byte.
	Flash write	Write cell by bytes or page.
	Flash page erase	Erase cell by page.
	Flash bulk erase	Erase the whole cells.
	Flash program verify	Read cell in verify mode after programming.
	Flash erase verify	Read cell in verify mode after erase.
	Flash page buffer load	Load data to page buffer.

15.5 Mode Entrance Method of ISP Mode

15.5.1 Mode Entrance Method for ISP

TARGET MODE	DSDA	DSCL	DSDA
OCD(ISP)	‘hC	‘hC	‘hC

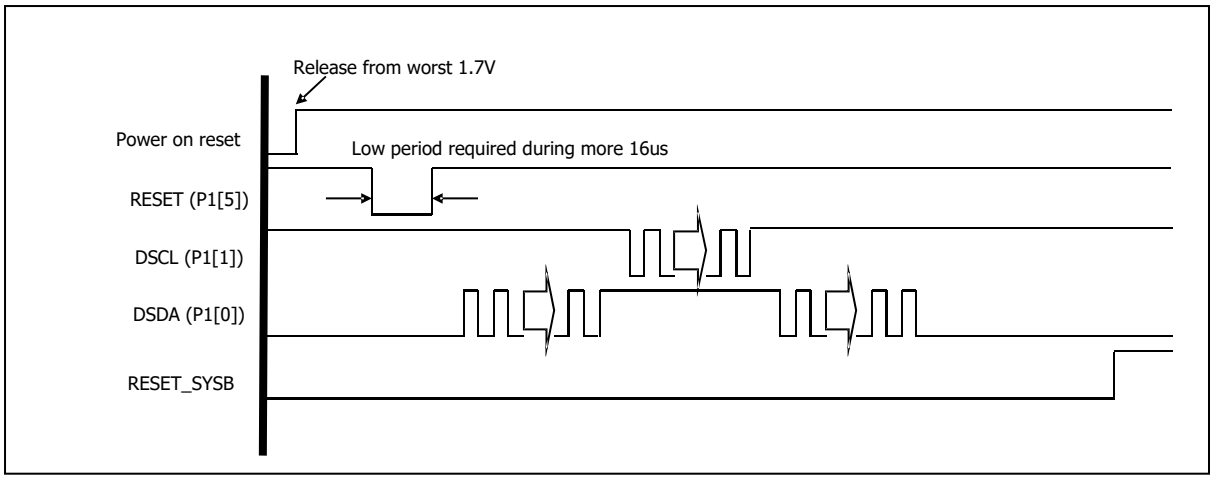


Figure 15-6 ISP Mode

15.6 Security

A96T218 provides Lock bits which can be left un-programmed (“0”) or can be programmed (“1”) to obtain the additional features listed in Table 15-4. The Lock bit can only be erased to “0” with the bulk erase command and a value of more than 0x40 at FETCR.

Table 15-4 Security policy using lock-bits

LOCK MODE	USER MODE								ISP MODE							
	FLASH				OTP				FLASH				OTP			
LOCKF	R	W	PE	BE	R	W	PE	BE	R	W	PE	BE	R	W	PE	BE
0	O	O	O	X	X	X	X	X	O	O	O	O	O	O	O	O
1	O	O	O	X	X	X	X	X	X	X	X	O	O	X	X	O

- LOCKF: Lock bit of Flash memory
- R: Read
- W: Write
- PE: Page erase
- BE: Bulk Erase
- O: Operation is possible.
- X: Operation is impossible.

16. Configure option

16.1 Configure Option Control Register

FUSE_CONF (Pseudo-Configure Data) : 2F50H

7	6	5	4	3	2	1	0
BSIZE1	BSIZE0	SXINEN	-	GPIOSEL	LOCKB	-	LOCKF
R	R	R	-	R	R	-	R

Initial value : 00H

BSIZE Select Specific Area for Write Protection
Note) When LOCKB is set, it is applied.

00 0000h~03FFh (1KB)

01 0000h~07FFh (2KB)

10 0000h~0FFFh (4KB)

11 0000h~7BFFh (31KB : If you use data eeprom, must be selected this option)

SXINEN Enable Sub-Oscillator (32.768kHz)

0 Sub-Xtal Disable (default)

1 Sub-Xtal Enable

GPIOSEL Select GPIO or External Reset

0 Select External Reset (default)

1 Select GPIO (P15)

LOCKB Enable Specific Area (Boot Area) Write Protection

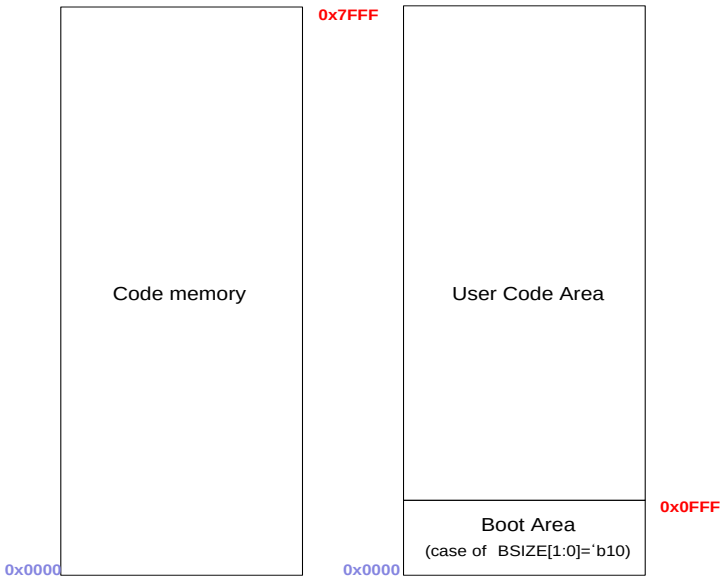
0 LOCK Disable

1 LOCK Enable

LOCKF CODE Read Protection

0 LOCK Disable

1 LOCK Enable



* BSIZE : Select protected boot area (base address = 0x0000)
(11 = 0~31KB, 10 = 0~4KB, 01 = 0~2KB, 00 = 0~1KB)

Figure 16-1 Boot area write protection by LOCKB (from bottom to BSIZE)

17. APPENDIX

A. Instruction Table

Instructions are either 1, 2 or 3 bytes long as listed in the 'Bytes' column below.

Each instruction takes either 1, 2 or 4 machine cycles to execute as listed in the following table. 1 machine cycle comprises 2 system clock cycles.

ARITHMETIC				
Mnemonic	**Description**	**Bytes**	**Cycles**	**Hex code**
ADD A,Rn	Add register to A	1	1	28-2F
ADD A,dir	Add direct byte to A	2	1	25
ADD A,@Ri	Add indirect memory to A	1	1	26-27
ADD A,#data	Add immediate to A	2	1	24
ADDC A,Rn	Add register to A with carry	1	1	38-3F
ADDC A,dir	Add direct byte to A with carry	2	1	35
ADDC A,@Ri	Add indirect memory to A with carry	1	1	36-37
ADDC A,#data	Add immediate to A with carry	2	1	34
SUBB A,Rn	Subtract register from A with borrow	1	1	98-9F
SUBB A,dir	Subtract direct byte from A with borrow	2	1	95
SUBB A,@Ri	Subtract indirect memory from A with borrow	1	1	96-97
SUBB A,#data	Subtract immediate from A with borrow	2	1	94
INC A	Increment A	1	1	04
INC Rn	Increment register	1	1	08-0F
INC dir	Increment direct byte	2	1	05
INC @Ri	Increment indirect memory	1	1	06-07
DEC A	Decrement A	1	1	14
DEC Rn	Decrement register	1	1	18-1F
DEC dir	Decrement direct byte	2	1	15
DEC @Ri	Decrement indirect memory	1	1	16-17
INC DPTR	Increment data pointer	1	2	A3
MUL AB	Multiply A by B	1	4	A4
DIV AB	Divide A by B	1	4	84
DA A	Decimal Adjust A	1	1	D4

LOGICAL				
Mnemonic	**Description**	**Bytes**	**Cycles**	**Hex code**
ANL A,Rn	AND register to A	1	1	58-5F
ANL A,dir	AND direct byte to A	2	1	55
ANL A,@Ri	AND indirect memory to A	1	1	56-57
ANL A,#data	AND immediate to A	2	1	54
ANL dir,A	AND A to direct byte	2	1	52
ANL dir,#data	AND immediate to direct byte	3	2	53
ORL A,Rn	OR register to A	1	1	48-4F
ORL A,dir	OR direct byte to A	2	1	45
ORL A,@Ri	OR indirect memory to A	1	1	46-47
ORL A,#data	OR immediate to A	2	1	44
ORL dir,A	OR A to direct byte	2	1	42
ORL dir,#data	OR immediate to direct byte	3	2	43
XRL A,Rn	Exclusive-OR register to A	1	1	68-6F
XRL A,dir	Exclusive-OR direct byte to A	2	1	65
XRL A, @Ri	Exclusive-OR indirect memory to A	1	1	66-67

XRL A,#data	Exclusive-OR immediate to A	2	1	64
XRL dir,A	Exclusive-OR A to direct byte	2	1	62
XRL dir,#data	Exclusive-OR immediate to direct byte	3	2	63
CLR A	Clear A	1	1	E4
CPL A	Complement A	1	1	F4
SWAP A	Swap Nibbles of A	1	1	C4
RL A	Rotate A left	1	1	23
RLC A	Rotate A left through carry	1	1	33
RR A	Rotate A right	1	1	03
RRC A	Rotate A right through carry	1	1	13

DATA TRANSFER				
Mnemonic	**Description**	**Bytes**	**Cycles**	**Hex code**
MOV A,Rn	Move register to A	1	1	E8-EF
MOV A,dir	Move direct byte to A	2	1	E5
MOV A,@Ri	Move indirect memory to A	1	1	E6-E7
MOV A,#data	Move immediate to A	2	1	74
MOV Rn,A	Move A to register	1	1	F8-FF
MOV Rn,dir	Move direct byte to register	2	2	A8-AF
MOV Rn,#data	Move immediate to register	2	1	78-7F
MOV dir,A	Move A to direct byte	2	1	F5
MOV dir,Rn	Move register to direct byte	2	2	88-8F
MOV dir,dir	Move direct byte to direct byte	3	2	85
MOV dir,@Ri	Move indirect memory to direct byte	2	2	86-87
MOV dir,#data	Move immediate to direct byte	3	2	75
MOV @Ri,A	Move A to indirect memory	1	1	F6-F7
MOV @Ri,dir	Move direct byte to indirect memory	2	2	A6-A7
MOV @Ri,#data	Move immediate to indirect memory	2	1	76-77
MOV DPTR,#data	Move immediate to data pointer	3	2	90
MOVC A,@A+DPTR	Move code byte relative DPTR to A	1	2	93
MOVC A,@A+PC	Move code byte relative PC to A	1	2	83
MOVX A,@Ri	Move external data(A8) to A	1	2	E2-E3
MOVX A,@DPTR	Move external data(A16) to A	1	2	E0
MOVX @Ri,A	Move A to external data(A8)	1	2	F2-F3
MOVX @DPTR,A	Move A to external data(A16)	1	2	F0
PUSH dir	Push direct byte onto stack	2	2	C0
POP dir	Pop direct byte from stack	2	2	D0
XCH A,Rn	Exchange A and register	1	1	C8-CF
XCH A,dir	Exchange A and direct byte	2	1	C5
XCH A,@Ri	Exchange A and indirect memory	1	1	C6-C7
XCHD A,@Ri	Exchange A and indirect memory nibble	1	1	D6-D7

BOOLEAN				
Mnemonic	**Description**	**Bytes**	**Cycles**	**Hex code**
CLR C	Clear carry	1	1	C3
CLR bit	Clear direct bit	2	1	C2
SETB C	Set carry	1	1	D3
SETB bit	Set direct bit	2	1	D2
CPL C	Complement carry	1	1	B3
CPL bit	Complement direct bit	2	1	B2
ANL C,bit	AND direct bit to carry	2	2	82
ANL C,/bit	AND direct bit inverse to carry	2	2	B0

ORL C,bit	OR direct bit to carry	2	2	72
ORL C,/bit	OR direct bit inverse to carry	2	2	A0
MOV C,bit	Move direct bit to carry	2	1	A2
MOV bit,C	Move carry to direct bit	2	2	92

BRANCHING				
Mnemonic	**Description**	**Bytes**	**Cycles**	**Hex code**
ACALL addr 11	Absolute jump to subroutine	2	2	11→F1
LCALL addr 16	Long jump to subroutine	3	2	12
RET	Return from subroutine	1	2	22
RETI	Return from interrupt	1	2	32
AJMP addr 11	Absolute jump unconditional	2	2	01→E1
LJMP addr 16	Long jump unconditional	3	2	02
SJMP rel	Short jump (relative address)	2	2	80
JC rel	Jump on carry = 1	2	2	40
JNC rel	Jump on carry = 0	2	2	50
JB bit,rel	Jump on direct bit = 1	3	2	20
JNB bit,rel	Jump on direct bit = 0	3	2	30
JBC bit,rel	Jump on direct bit = 1 and clear	3	2	10
JMP @A+DPTR	Jump indirect relative DPTR	1	2	73
JZ rel	Jump on accumulator = 0	2	2	60
JNZ rel	Jump on accumulator ≠ 0	2	2	70
CJNE A,dir,rel	Compare A,direct jne relative	3	2	B5
CJNE A,#d,rel	Compare A,immediate jne relative	3	2	B4
CJNE Rn,#d,rel	Compare register, immediate jne relative	3	2	B8-BF
CJNE @Ri,#d,rel	Compare indirect, immediate jne relative	3	2	B6-B7
DJNZ Rn,rel	Decrement register, jnz relative	2	2	D8-DF
DJNZ dir,rel	Decrement direct byte, jnz relative	3	2	D5

MISCELLANEOUS				
Mnemonic	**Description**	**Bytes**	**Cycles**	**Hex code**
NOP	No operation	1	1	00

ADDITIONAL INSTRUCTIONS (selected through EO[7:4])				
Mnemonic	**Description**	**Bytes**	**Cycles**	**Hex code**
MOVC @(DPTR++),A	M8051W/M8051EW-specific instruction supporting software download into program memory	1	2	A5
TRAP	Software break command	1	1	A5

In the above table, an entry such as E8-EF indicates a continuous block of hex op-codes used for 8 different registers, the register numbers of which are defined by the lowest three bits of the corresponding code. Non-continuous blocks of codes, shown as 11→F1 (for example), are used for absolute jumps and calls, with the top 3 bits of the code being used to store the top three bits of the destination address.

The CJNE instructions use the abbreviation #d for immediate data; other instructions use #data.

Table of Contents

Revision history 2
1. Overview 3
1.1 Description 3
1.2 Features 4
1.3 Development tools 5
2. Block Diagram 10
3. Pin Assignment 11
4. Package Diagram 15
5. Pin Description 20
6. Port Structures 21
6.1 General Purpose I/O Port 21
6.2 General Purpose I/O Port with 1.8V Interface (P10, P11) 22
6.3 Open-Drain I/O Port with 1.8V Interface (P12, P16, P17) 23
7. Electrical Characteristics 24
7.1 Absolute Maximum Ratings 24
7.2 Recommended Operating Conditions 24
7.3 Voltage Dropout Converter Characteristics 25
7.4 Power-On Reset Characteristics 25
7.5 Brown Out Detector & Low Voltage Indicator Characteristics 25
7.6 Internal RC Oscillator Characteristics 26
7.7 Ring-Oscillator Characteristics 26
7.8 Sub-Oscillator Characteristics 26
7.9 Touch Switch Characteristics 26
7.10 A/D Converter Characteristics 27
7.11 DC Characteristics 28
7.12 AC Characteristics 29
7.13 Typical Characteristics 30
8. Memory 31
8.1 Program Memory 31
8.2 Data Memory 33
8.3 XSFR 35
8.4 SFR Map 36
9. I/O Ports 39
9.1 I/O Ports 39
9.2 Port Register 39
9.3 P0, P1, P2, P3, P4 Port 43
10. Interrupt Controller 47
10.1 Overview 47
10.2 Block Diagram 48
10.3 Interrupt Vector Table 49
10.4 Interrupt Sequence 50
10.5 Effective Timing after Controlling Interrupt bit 51
10.6 Multi Interrupt 51
10.7 Interrupt Enable Accept Timing 53
10.8 Interrupt Service Routine Address 53

10.9 Saving/Restore General-Purpose Registers 53
10.10 Interrupt Timing 54
10.11 Interrupt Register Overview 54
10.12 Interrupt Register Description 55
11. Peripheral Hardware 62
11.1 Clock Generator 62
11.2 BIT 65
11.3 WDT 67
11.4 WT 70
11.5 Free Run Counter 73
11.6 Buzzer Driver 75
11.7 Timer/PWM 77
11.8 USART 88
11.9 I^2C 107
11.10 12-Bit A/D Converter 119
11.11 24-Channel Touch Switch 124
11.12 LED Driver 138
12. Power Down Operation 145
12.1 Overview 145
12.2 Peripheral Operation in IDLE/STOP Mode 145
12.3 IDLE mode 146
12.4 STOP mode 147
12.5 Release Operation of STOP1, 2 Mode 148
13. RESET 150
13.1 Overview 150
13.2 Reset Source 150
13.3 Block Diagram 150
13.4 RESET Noise Canceller 151
13.5 Power ON Reset 151
13.6 External RESETB Input 154
13.7 Brown Out Detector Processor 155
14. On-chip Debug System 158
14.1 Overview 158
14.2 Two-pin External Interface 159
15. Memory Programming 163
15.1 Overview 163
15.2 Flash Control and Status Register 163
15.3 Memory Map 169
15.4 Serial In-System Program Mode 170
15.5 Mode Entrance Method of ISP Mode 175
15.6 Security 176
16. Configure option 177
16.1 Configure Option Control Register 177
17. APPENDIX 179
Table of Contents 182

List of Figures

Figure 2-1 A96T218 Block Diagram 10
Figure 3-1 A96T218 40 QFN Pin Assignment 11
Figure 3-2 A96T218 32 LQFP Pin Assignment 12
Figure 3-3 A96T218 32 SOP Pin Assignment 13
Figure 3-4 A96T218 28 SOP Pin Assignment 14
Figure 4-1 40 pin QFN Package 15
Figure 4-2 32 pin LQFP Package 16
Figure 4-3 32 pin SOP Package 17
Figure 4-4 28 pin SOP Package 18
Figure 6-1 General Purpose I/O Port 21
Figure 6-2 General Purpose I/O Port with 1.8V Interface 22
Figure 6-3 Open-Drain I/O Port with 1.8V Interface 23
Figure 7-1 AC Timing 29
Figure 8-1 Program Memory 32
Figure 8-2 Data Memory Map 33
Figure 8-3 Low 128 bytes RAM 34
Figure 8-4 XDATA Memory Area 35
Figure 9-1 Debounce Function 39
Figure 10-1 Interrupt Group Priority Level 47
Figure 10-2 Block Diagram of Interrupt 48
Figure 10-3 Interrupt Sequence Flow 50
Figure 10-4 Interrupt Enable Register Effective Timing 51
Figure 10-5 Execution of Multi Interrupt 51
Figure 10-6 Interrupt Response Timing Diagram 53
Figure 10-7 Correspondence between Vector Table Address and the Entry Address of ISR 53
Figure 10-8 Saving/Restore Process Diagram & Sample Source 53
Figure 10-9 Timing chart of Interrupt Acceptance and Interrupt Return Instruction 54
Figure 11-1 Clock Generator Block Diagram 62
Figure 11-2 BIT Block Diagram 65
Figure 11-3 WDT Block Diagram 67
Figure 11-4 WDT Interrupt Timing Waveform 69
Figure 11-5 WT Block Diagram 70
Figure 11-6 Free Run Counter Block Diagram 73
Figure 11-7 Buzzer Driver Block Diagram 75
Figure 11-8 Timer x 16-bit Mode Block Diagram 77
Figure 11-9 Interrupt of Timer/Counter Mode 78
Figure 11-10 Operation Example of Timer/Event Counter Mode 78
Figure 11-11 Timer x 16bit Capture Mode 79
Figure 11-12 Input Capture Mode Operation of Timer 0, 1 80
Figure 11-13 Express Timer Overflow in Capture Mode 80
Figure 11-14 PWM Mode 81
Figure 11-15 Example of PWM Output Waveform 82
Figure 11-16 Example of PWM (16MHz) 82

Figure 11-17 USART Block Diagram 89
Figure 11-18 Clock Generation Block Diagram 90
Figure 11-19 Synchronous Mode XCKn Timing 91
Figure 11-20 Frame Format 92
Figure 11-21 Start Bit Sampling 95
Figure 11-22 Sampling of Data and Parity Bit 96
Figure 11-23 Stop Bit Sampling and Next Start Bit Sampling 96
Figure 11-24 SPI Clock Formats when UCPHA=0 98
Figure 11-25 SPI Clock Formats when UCPHA=1 98
Figure 11-26 I^2C Block Diagram 107
Figure 11-27 Bit Transfer on the I^2C-Bus 108
Figure 11-28 START and STOP Condition 108
Figure 11-29 Data Transfer on the I^2C-Bus 109
Figure 11-30 Acknowledge on the I^2C-Bus 109
Figure 11-31 Clock Synchronization during Arbitration Procedure 110
Figure 11-32 Arbitration Procedure of Two Masters 110
Figure 11-33 ADC Block Diagram 119
Figure 11-34 A/D Analog Input Pin Connecting Capacitor 120
Figure 11-35 ADC Operation for Align Bit 120
Figure 11-36 Converter Operation Flow 121
Figure 11-37 24 Channels Touch Switch Block Diagram 125
Figure 11-38 User Programming Procedure 137
Figure 11-39 LED Driver Block Diagram 138
Figure 11-40 SHARE Touch Buttons 144
Figure 12-1 IDLE Mode Release Timing by External Interrupt 146
Figure 12-2 IDLE Mode Release Timing by RESETB 146
Figure 12-3 STOP Mode Release Timing by Exteranl Interrupt 147
Figure 12-4 STOP Mode Release Timing by RESETB 147
Figure 12-5 STOP1, 2 Mode Release Flow 148
Figure 13-1 RESET Block Diagram 150
Figure 13-2 Reset Noise Canceller Time Diagram 151
Figure 13-3 Fast VDD Rising Time 151
Figure 13-4 Internal RESET Release Timing on Power-Up 152
Figure 13-5 Configuration Timing when Power-on 152
Figure 13-6 Boot Process Waveform 153
Figure 13-7 Timing Diagram after RESET 154
Figure 13-8 Oscillator Generating Waveform Example 154
Figure 13-9 Block Diagram of BOD 155
Figure 13-10 Internal Reset at the Power Fail Situation 155
Figure 13-11 Configuration Timing when BOD RESET 156
Figure 14-1 Block Diagram of On-chip Debug System 158
Figure 14-2 10-bit Transmission Packet 159
Figure 14-3 Data Transfer on the Twin Bus 160
Figure 14-4 Bit Transfer on the Serial Bus 160
Figure 14-5 Start and Stop Condition 161

Figure 14-6 Acknowledge on the Serial Bus 161
Figure 14-7 Clock Synchronization during Wait Procedure 161
Figure 14-8 Connection of Transmission 162
Figure 15-1 Read device internal CkeckSum 167
Figure 15-2 Flash Memory Map 169
Figure 15-3 Address Configuration of Flash Memory 169
Figure 15-4 The Sequence of Page Program and Erase of Flash Memory 170
Figure 15-5 The Sequence of Bulk Erase of Flash Memory 171
Figure 15-6 ISP Mode 175
Figure 16-1 Boot area write protection by LOCKB (from bottom to BSIZE) 178

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 DISCLAIMES ALL EXPRESS AND IMPLIED WARRANTIES AND SHALL NOT BE RESPONSIBLE OR LIABLE FOR ANY INJUIRES OR DAMAGES RELATED TO USE OF ABOV PRODUCTS IN SUCH UNAUTHORIZED APPLICATIONS.** ABOV and the ABOV logo are trademarks of ABOV. For additional information about ABOV trademarks, please refer to https://www.abov.co.kr/en/about/corporate_identity.php. 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.

© 2016 ABOV Semiconductor – All rights reserved