AT8BE62D

14 I/O + 13-ch ADC 8-bit MTP-Based MCU with EEPROM

Version 1.5

Nov. 28, 2025

ATW-Tech

Revision History

Version	Date	Description	Modified Page
1.0	2021/09/30	Formal release.	-
1.1	2022/05/31	Modified OSCCAL initial value description.	36
1.2	2022/08/31	Add EEPROM endurance cycles and Characteristics.	9, 12, 125
1.3	2023/02/28	Modified error.	100
1.4	2024/11/29	Update CMP Table.	87
1.5	2025/11/28	1. Updated Wide operating voltage range. 2. Update Output high current & Output Low current. 3. LVD voltage error. 4. Recommended Operating Voltage.	9, 12 123 124 131

Table of Contents

1. 概述

AT8BE62D 是以MTP作為程式記憶體，並以EEPROM作為非揮發性資料記憶體的 8 位元微控制器，特別適合斷電後還需保持資料內容的應用，例如遙控器、風扇/燈光控制或是遊樂器周邊等等。除此之外內建類比數位轉換器也適用於家電或量測等等的應用設計，除此之外更針對霧化器產品做優化調適。採用CMOS製程並同時提供客戶低成本、高性能、及高抗電磁干擾等顯著優勢。而MTP作為程式記憶體能讓使用者更方便且有效率的開發產品。AT8BE62D 核心建立在精簡指令集架構，可以很容易地做編輯和控制，共有 53 條指令。除了少數指令需要 2 個時序，大多數指令都是 1 個時序即能完成，可以讓使用者輕鬆地以程式控制完成不同的應用。因此非常適合各種中低記憶容量但又複雜的應用。

AT8BE62D內建除錯仿真電路，利用兩個腳位與很少的外接硬體，就能實現在線仿真器的大多數功能，例如設定程式執行條件與中斷條件，片上單步執行，以及查看及設定各種暫存器的內容。仿真的執行效果將比一般的仿真器更接近實際IC運作。

AT8BE62D內建 12 位元高精度 11+2 通道類比數位轉換器與高精度電壓比較器，足以應付各種類比介面的偵測與量測。

AT8BE62D內建的高頻振盪器，能彈性選擇輸出頻率區段，以適用於多樣的霧化器元件，更可以程式細調輸出頻率， 細度足以應付霧化器的元件偏差。

在I/O的資源方面，AT8BE62D 有 14 根彈性的雙向I/O腳，每個I/O腳都有單獨的暫存器控制為輸入或輸出腳。而且每一個I/O腳位都有附加的程式控制功能如上拉或下拉電阻或開漏極(Open-Drain) 輸出。此外針對紅外線搖控的產品方面，AT8BE62D內建了可選擇頻率的紅外載波發射口。

AT8BE62D 有四組計時器，可用系統頻率當作一般的計時的應用或者從外部訊號觸發來計數。另外AT8BE62D 提供 5 組 10 位元解析度的PWM輸出，1 組蜂鳴器輸出可用來驅動馬達、LED、或蜂鳴器等等。AT8BE62D的計時器同時具有增強型捕捉/比較/可編程死區時間的PWM模塊(ECCP).

AT8BE62D 採用雙時鐘機制，高速振盪或者低速振盪都可以分別選擇內部RC振盪或外部晶振輸入。在雙時鐘機制下，AT8BE62D 可選擇多種工作模式如正常模式(Normal)、慢速模式(Slow mode)、待機模式(Standby mode) 與睡眠模式(Halt mode)可節省電力消耗延長電池壽命。並且微控制器在使用內部RC高速振盪時，低速振盪可以同時使用外部精準的晶振計時，可以維持高速處理同時又能精準計算真實時間。

在省電的模式下如待機模式(Standby mode) 與睡眠模式(Halt mode)中，有多種事件可以觸發中斷喚醒AT8BE62D 進入正常操作模式(Normal) 或 慢速模式(Slow mode) 來處理突發事件。

AT8BE62D能通過初始設定實現與AT8B60E相容的功能，可使用AT8BE62D內建除錯仿真電路來模擬實際IC動作。

1.1 功能

- 系統主頻提供 20.8M/19.2M/16M/14.4M/13.6M 五種選擇。
- Timer 1/4/5 可選擇系統主頻直接輸入。
- 高頻率振盪器(F_{HOSC})可微調，微調精度至 ±0.1%。
- 內建二線控制的除錯仿真電路(On Chip Debug)。
- 雜訊過濾功能(Noise Filter)打開時可容忍超過±4KV的EFT。(操作電壓@5V)
- 寬廣的工作電壓：
  - 3.3V ~ 5.5V @20.8MHz/2T.
  - 1.6V ~ 5.5V @4MHz/4T.
- 寬廣的工作溫度：-20°C ~ 85°C。
- 2Kx14 bits MTP。
- 256 bytes EEPROM。
- 128 bytes SRAM。
- 14 根可分別單獨控制輸入輸出方向的I/O腳(GPIO)、PA[7:0]、PB[5:0] 。
- PA[5:0]及PB[3:0]可選擇輸入時使用內建下拉電阻。
- PA[7:0]及PB[5:0]可選擇輸入時使用上拉電阻，上拉電阻為 100KΩ。
- PB[5:0]可選擇開漏極輸出(Open-Drain)。
- 所有I/O腳輸出灌電流(Sink Current)有三檔可選: 小(Small)、一般(Normal)或大(Large)。
- 所有I/O腳輸出推電流(Drive Current)可選擇小(Small)或一般(Normal)。
- 8 層程式堆棧(Stack)。
- 存取資料有直接或間接定址模式。
- 一組 8 位元上數計時器(Timer0)包含可程式化的頻率預除線路。
- 三組 10 位元下數計時器(Timer1, 4, 5)可選重複載入或連續下數計時。
- 五個 10 位元脈衝寬度調變(PWM1, 2, 3, 4, 5)。
- Timer4, 5 可組成 16 位元捕捉/比較功能, Timer5/PWM5 可轉變成具死區控制的PWM輸出。
- 一個蜂鳴器輸出(BZ1)。
- 38/57KHz紅外線載波頻率可供選擇，同時載波之極性也可以根據數據作選擇。
- 內建準確的低電壓偵測電路(LVD)。
- 內建 11+2 通道 12 位元類比數位轉換器(Analog to Digital Converter)。
- 內建準確的電壓比較器(Voltage Comparator)。
- 內建上電復位電路(POR)。

- 內建低壓復位功能(LVR)。
- 內建看門狗計時(WDT)，可由程式韌體控制開關。
- 內建電阻頻率轉換器(RFC)功能.
- 雙時鐘機制，系統可以隨時切換高速振盪或者低速振盪。
  - 高速振盪：E_HXT (超過 6MHz外部高速石英振盪)
    E_XT (455K~6MHz外部石英振盪)
    I_HRC (1~20.8MHz內部高速RC振盪)
  - 低速振盪：E_LXT (32KHz外部低速石英振盪)
    I_LRC (內部 32KHz低速RC振盪)
- 四種工作模式可隨系統需求調整電流消耗：正常模式(Normal)、慢速模式(Slow mode)、待機模式(Standby mode) 與 睡眠模式(Halt mode)。
- 十二種硬體中斷：
  - Timer0 溢位中斷。
  - Timer1 借位中斷。
  - Timer4 借位中斷。
  - Timer5 借位中斷 or CCP中斷。
  - WDT 中斷。
  - PA/PB 輸入狀態改變中斷。
  - 三組外部中斷輸入。
  - 低電壓偵測/比較器輸出轉態中斷。
  - 類比數位轉換完成中斷。
  - EEPROM寫入完成中斷。
- AT8BE62D在待機模式(Standby mode)下的十一種喚醒中斷：
  - Timer0 溢位中斷。
  - Timer1 借位中斷。
  - Timer4 借位中斷。
  - Timer5 借位中斷 or CCP中斷。
  - WDT 中斷。
  - PA/PB 輸入狀態改變中斷。
  - 三組外部中斷輸入。
  - 低電壓偵測/比較器輸出轉態中斷。
  - 類比數位轉換完成中斷。
- AT8BE62D在睡眠模式(Halt mode)下的五種喚醒中斷：
  - WDT 中斷。
  - PA/PB 輸入狀態改變中斷。
  - 三組外部中斷輸入。

1. General Description

AT8BE62D is a MTP based 8-bit MCU with EEPROM data memory built-in, tailored for applications which retaining data are needed after power is shut-down. The feature of re-programmable of MTP memory makes product development more flexible and efficient. AT8BE62D adopts advanced CMOS technology to provide customers remarkable solution with low cost, high performance and high noise immunity benefits. RISC architecture is applied to AT8BE62D and it provides 53 instructions. All instructions are executed in single instruction cycle except program branch and skip instructions which will take two instruction cycles. Therefore, AT8BE62D is very suitable for those applications that are sophisticated but compact program size is required.

AT8BE62D provide on-chip debug (OCD) facilities as a low cost alternative to ICE. With OCD and a minimum of extra hardware, AT8BE62D can do ICE tasks such as free running, single stepping, break point setting and internal ram/register accessing.

AT8BE62D provides 11+2 channel high-precision 12-bit analog-to-digital converter (ADC), and high-precision Low Dropout Regulator and analog voltage comparator. They are suitable for any analog interface detection and measurement applications.

The AT8BE62D built-in high speed oscillator can be optioned for different center frequency. users can adjust this frequency in the program in a very small frequency step. With this feature ensures that AT8BE62D will find excellent use in different ultrasonic nebulizer applications.

As AT8BE62D address I/O type applications, it can provide 14 I/O pins for applications which require abundant input and output functionality. Moreover, each I/O pin may have additional features, like Pull-High/Pull-Low resistor and open-drain output type through programming. Moreover, AT8BE62D has built-in large infrared (IR) carrier generator (340mA@3V) with selectable IR carrier frequency and polarity for applications which demand remote control feature.

AT8BE62D also provides 4 sets of timers which can be used as regular timer based on system oscillation or event counter with external trigger clock. Moreover, AT8BE62D provides 5 sets of 10-bit resolution Pulse Width Modulation (PWM) output and 1 sets of buzzer output in order to drive motor/LED and buzzer. Moreover, AT8BE62D timers provide powerful enhance capture/compare/PWM (ECCP) functions.

AT8BE62D employs dual-clock oscillation mechanism, either high oscillation or low oscillation can be derived from internal resistor/capacitor oscillator or external crystal oscillator. Moreover, based on dual-clock mechanism, AT8BE62D provides kinds of operation mode like Normal mode, Slow mode, Standby mode and Halt mode in order to save power consumption and lengthen battery operation life. Moreover, it is possible to use internal high-frequency oscillator as CPU operating clock source and external 32KHz crystal oscillator as timer clock input, so as to accurate count real time and maintain CPU working power.

While AT8BE62D operates in Standby mode and Halt mode, kinds of event will issue interrupt requests and can wake-up AT8BE62D to enter Normal mode and Slow mode in order to process urgent events.

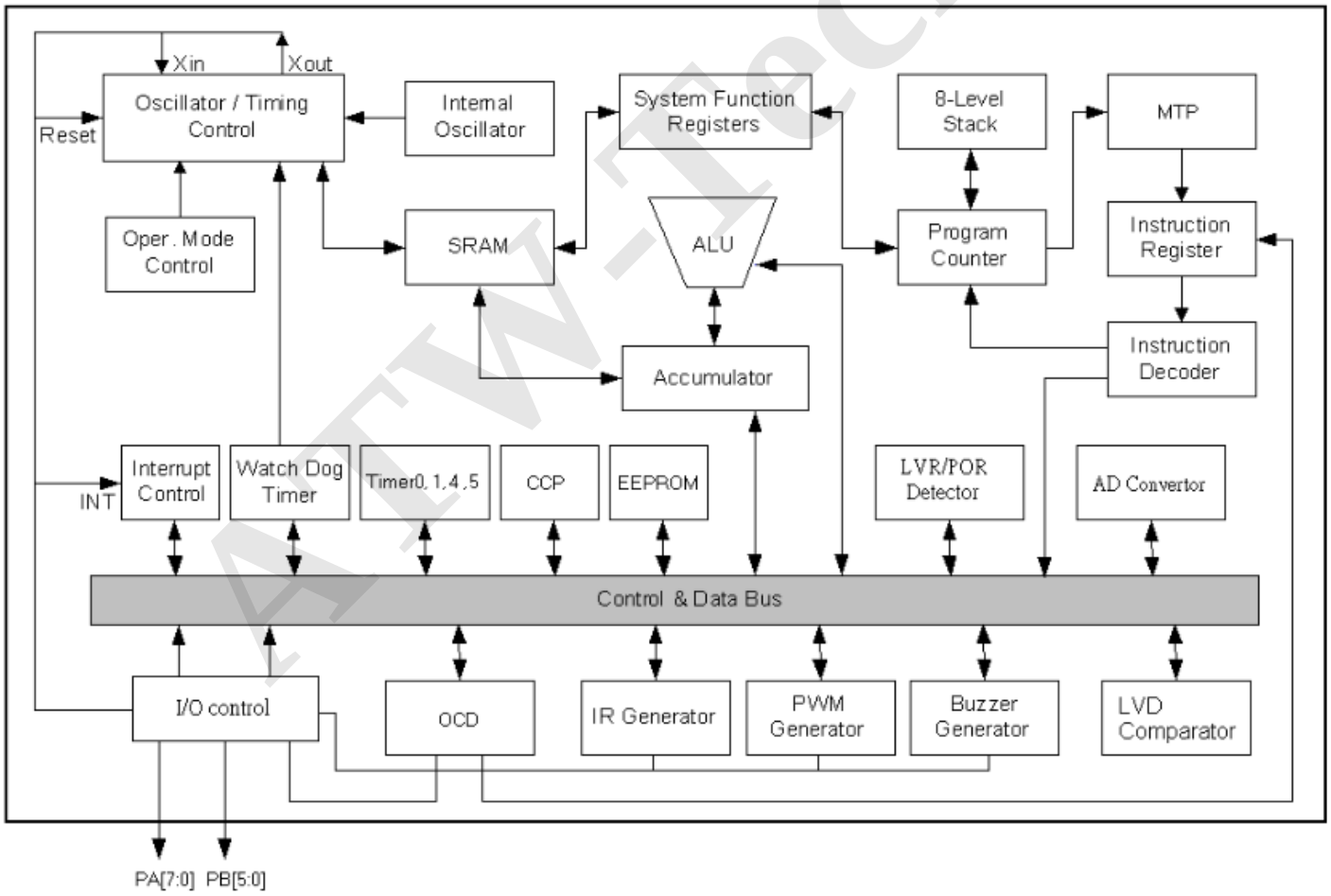
AT8BE62D can be configured as AT8B60E compatible functions. AT8B60E shares the benefit of AT8BE62D OCD.

1.1 Features

- Wide operating voltage range:
  - 3.3V ~ 5.5V @20.8MHz/2T.
  - 1.6V ~ 5.5V @4MHz/4T.
- Wide operating temperature: -20°C ~ 85°C.
- On-chip debugging (OCD) via two pins.
- 2K x 14 bits MTP.
- 256 bytes EEPROM.
- 128 bytes SRAM.
- 14 general purpose I/O pins (GPIO), PA[7:0], PB[5:0] with independent direction control.
- PA[5:0] , PB[3:0] have features of Pull-Low resistor for input pin.
- PA[7:0] and PB[5:0] have features of Pull-High resistor, the value of Pull-High resistor 100KΩ .
- PB[5:0] has features of Open-Drain output.
- I/O ports output current mode can be small sink, normal sink or large sink.
- I/O ports output current mode can be small drive or normal drive.
- 8-level hardware Stack.
- Direct and indirect addressing modes for data access.
- One 8-bit up-count timer (Timer0) with programmable prescaler.
- Three 10-bit reload or continuous down-count timers (Timer1, 4, 5).
- Five 10-bit resolution PWM (PWM1, 2, 3, 4, 5) output.
- In a special mode, Timer input clock is provided from I_HRC directly.
- One buzzer (BZ1) output.
- Selectable 38/57KHz IR carrier frequency and high/low polarity according to data value.
- IR carrier sink current can be normal sink current or 340mA large sink current.
- Built-in high-precision Low-Voltage Detector (LVD).
- Built-in 11+2 channel high-precision 12-bit ADC.
- Built-in high-precision Voltage Comparator.
- Built-in Power-On Reset (POR).
- Built-in Low-Voltage Reset (LVR).
- Built-in Watch-Dog Timer (WDT) enabled/disabled by firmware control.

- Built-in Resistance to Frequency Converter (RFC) function.
- Dual-clock oscillation: System clock can switch between high oscillation and low oscillation.
  - High oscillation: E_HXT (External High Crystal Oscillator, above 6MHz)
    E_XT (External Crystal Oscillator, 455K~6MHz)
    I_HRC (Internal High Resistor/Capacitor Oscillator ranging from 1M~20.8MHz)
  - Low oscillation: E_LXT (External Low Crystal Oscillator, about 32KHz)
    I_LRC (Internal 32KHz oscillator)
- Four kinds of operation mode to reduce system power consumption:
  - Normal mode, Slow mode, Standby mode and Halt mode.
- Twelve hardware interrupt events:
  - Timer0 overflow interrupt.
  - Timer1 underflow interrupt.
  - Timer4 underflow interrupt.
  - Timer5 underflow interrupt or CCP interrupt.
  - WDT timeout interrupt.
  - PA/PB input change interrupt.
  - 3 set External interrupt.
  - LVD / Comparator ouput status change interrupt.
  - ADC end-of-convert interrupt.
  - End of EEPROM write interrupt.
- Eleven interrupt events to wake-up AT8BE62D from Standby mode:
  - Timer0 overflow interrupt.
  - Timer1 underflow interrupt.
  - Timer4 underflow interrupt.
  - Timer5 underflow interrupt or CCP interrupt.
  - WDT timeout interrupt.
  - PA/PB input change interrupt.
  - 3 set External interrupt.
  - LVD/Comparator output status change interrupt.
  - ADC end-of-convert interrupt.
- Five interrupt events to wake-up AT8BE62D from Halt mode:
  - WDT timeout interrupt.
  - PA/PB input change interrupt.
  - 3 set External interrupt.

1.2 Block Diagram



1.3 Pin Assignment

AT8BE62D provides 3 kinds of package types which are SOP16, SOP14 and SOP8.

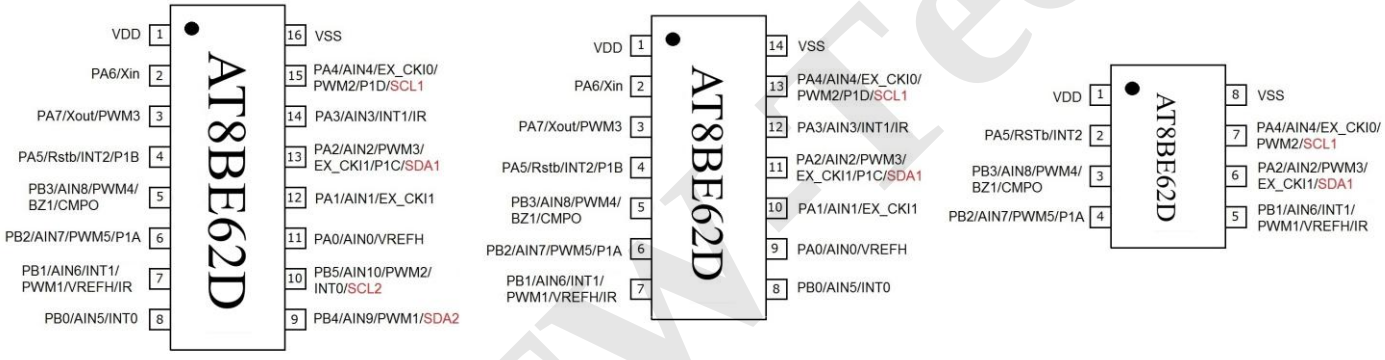


Figure 1 Package pin assignment

1.4 Pin Description

Pin Name	I/O	Description
PA0 AIN0 VREFH	I/O	PA0 is bidirectional I/O pin, and can be comparator input pin. PA0 can be ADC analog input pin. PA0 can be ADC external high reference voltage source.
PA1 AIN1 EX_CKI1	I/O	PA1 is bidirectional I/O pin, and can be comparator input pin. PA1 can be ADC analog input pin. PA1 can be Timer4/5 clock source EX_CKI1.
PA2 AIN2 PWM3 EX_CKI1 P1C SDA1	I/O	PA2 is bidirectional I/O pin, and can be comparator input pin. PA2 can be ADC analog input pin. PA2 can be output of PWM3. PA2 can be Timer4/5 clock source EX_CKI1. PA2 can be P1C output in CCP mode PA2 can be programming pad SDI1.
PA3 AIN3 INT1 IR	I/O	PA3 is bidirectional I/O pin, and can be comparator input pin. PA3 can be ADC analog input pin. PA3 can be input pin of external interrupt INT1 PA3 can be IR carrier output pin.
PA4 AIN4 PWM2 EX_CKI0 P1D SCL1	I/O	PA4 is bidirectional I/O pin. PA4 can be ADC analog input pin. PA4 can be output of PWM2. PA4 can be Timer0/1 clock source EX_CKI0. PA4 can be P1D output in CCP mode PA4 can be programming pad SCL1.
PA5 RSTb INT2 P1B	I/O	PA5 is bidirectional I/O pin. PA5 can be the reset pin RSTb. PA5 can be input pin of external interrupt INT2. PA5 can be P1B output in CCP mode
PA6 Xin	I/O	PA6 is bidirectional I/O pin. PA6 can be the input pin of crystal oscillator Xin.
PA7 Xout PWM3	I/O	PA7 is bidirectional I/O pin. PA7 can be the output pin of crystal oscillator Xout. PA7 can be output of instruction clock.
PB0 AIN5 INT0	I/O	PB0 is bidirectional I/O pin. PB0 can be ADC analog input pin. PB0 can be input pin of external interrupt INT0.
PB1 AIN6 INT1 PWM1 VREFH IR	I/O	PB1 is bidirectional I/O pin. PB1 can be ADC analog input pin. PB1 can be input pin of external interrupt INT1. PB1 can be output of PWM1. PB1 can be ADC external high reference voltage source. PB1 can be IR carrier output pin.

Pin Name	I/O	Description
PB2 AIN7 PWM5 P1A	I/O	PB2 is bidirectional I/O pin. PB2 can be ADC analog input pin. PB2 can be output of PWM5. PB2 can be P1A output in CCP mode.
PB3 AIN8 PWM4 BZ1 CMPO	I/O	PB3 is bidirectional I/O pin. PB3 can be ADC analog input pin. PB3 can be output of PWM4. PB3 can be output of Buzzer1. PB3 can be comparator output.
PB4 AIN8 PWM1 SDA2	I/O	PB4 is bidirectional I/O pin. PB4 can be ADC analog input pin. PB4 can be output of PWM1. PB4 can be programming pad SDA2.
PB5 AIN10 PWM2 INT0 SCL2	I/O	PB5 is bidirectional I/O pin. PB5 can be ADC analog input pin. PB5 can be output of PWM1. PB5 can be input pin of external interrupt INT0. PB5 can be programming pad SCL2.
VDD	-	Positive power supply
VSS	-	Ground

2. Memory Organization

AT8BE62D memory is divided into two categories: one is program memory and the other is data memory. Data memory is subdivided into Register Memory and EEPROM Memory.

2.1 Program Memory

The program memory space of AT8BE62D is 2K words. Therefore, the Program Counter (PC) is 11-bit wide in order to address any location of program memory.

Some locations of program memory are reserved as interrupt entrance. Power-On Reset vector is located at 0x000. Software interrupt vector is located at 0x001. Internal and external hardware interrupt vector is located at 0x008.

AT8BE62D provides instruction GOTOA, CALLA to address 256 location of program space. AT8BE62D provides instructions LCALL and LGOTO to address any location of program space.

When a call or interrupt is happening, next ROM address is written to top of the stack, when RET, RETIA or RETIE instruction is executed, the top of stack data is read and loaded to PC.

AT8BE62D program ROM address 0x7FE~0x7FF are reserved space, if user tries to write code in these addresses will get unexpected false functions.

AT8BE62D program ROM address 0x00E~0x00F are preset rolling code and can be released and used as normal program space.

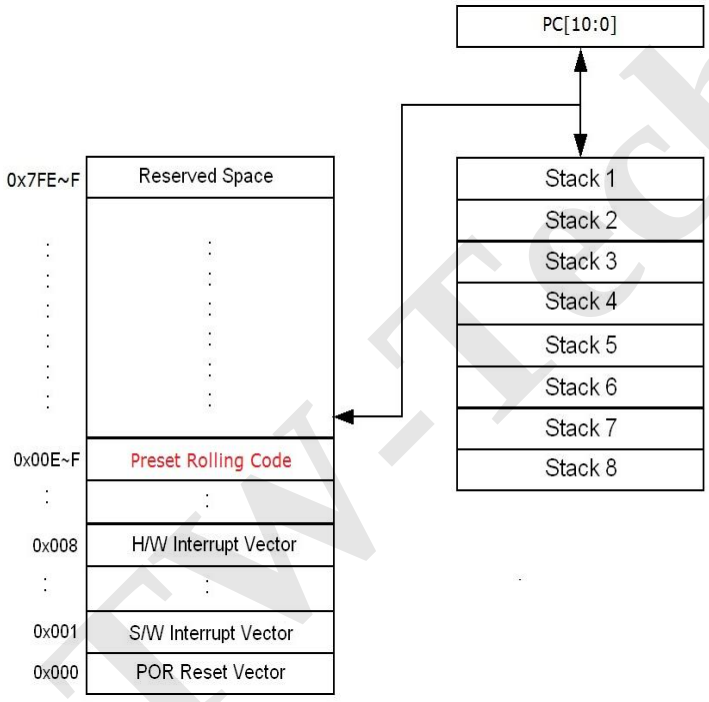


Figure 2 Program Memory Address Mapping

2.2 Register Memory

According to instructions used to access register, this memory can be divided into four kinds of categories: that is, R-page, F-page, S-page and T-page. R-page register consists of Special-function register (SFR) and General Purpose Register (GPR), GPR are made of SRAM and user can use them to store variables or intermediate results.

R-page data memory is divided into 4 banks and can be accessed directly or indirectly through a SFR register which is File Select Register (FSR). STATUS [7:6] are used as Bank register BK[1:0] to select one bank out of the 4 banks.

There are 2 addressing mode for R-page register accessing: direct addressing mode and indirect addressing mode.

The indirect addressing mode of data memory access is described in the following graph. This indirect addressing mode is implied by accessing register INDF. The bank selection is determined by STATUS[7:6] and the location selection is from FSR[6:0].

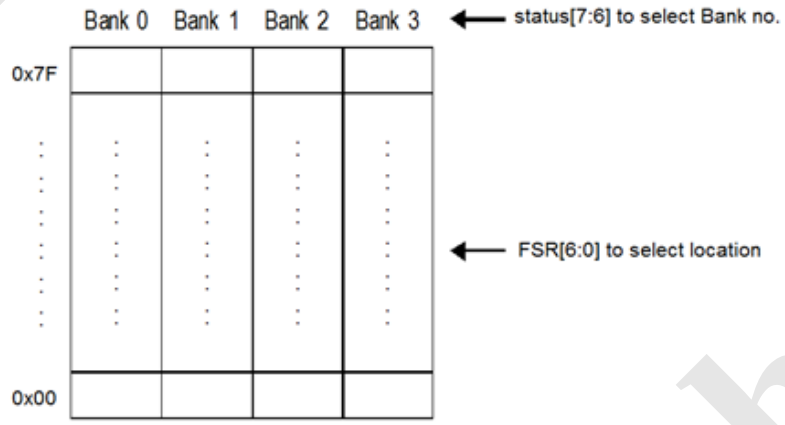


Figure 3 Indirect Addressing Mode of Data Memory Access

The direct addressing mode of data memory access is described below. The bank selection is determined by STATUS [7:6] and the location selection is from instruction op-code[6:0] immediately.

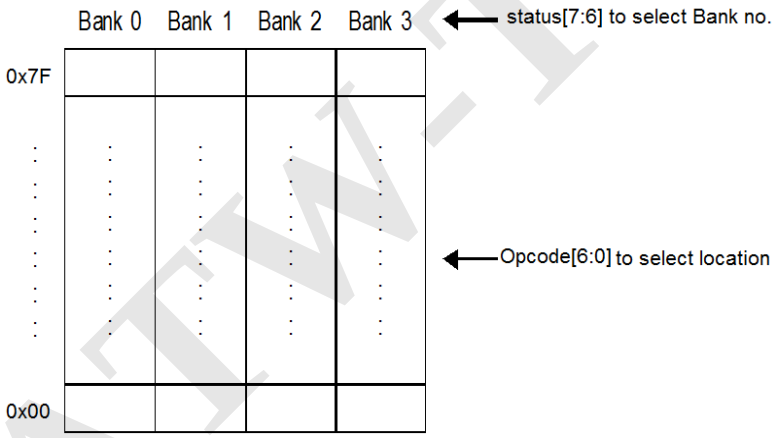


Figure 4 Direct Addressing Mode of Data Memory Access

R-page SFR can be accessed by general instructions like arithmetic instructions and data movement instructions. The R-page SFR occupies address from 0x0 to 0x1F of Bank 0. However, the same address range of Bank 1, Bank 2 and Bank 3 are mapped back to Bank 0. In other words, R-page SFR physically existed at Bank 0. The GPR physically occupy address from 0x20 to 0x7F of Bank 0 and 0x20 to 0x3F of Bank 1. Other bank in address from 0x20 to 0x7F are mapped back as the Table 1 shows.

The AT8BE62D register name and address mapping of R-page SFR are described in the following table.

Status [7:6] Address	**00 (Bank 0)**	**01 (Bank 1)**	**10 (Bank 2)**	**11 (Bank 3)**
0x0	INDF	*The same mapping as Bank 0*		
0x1	TMR0			
0x2	PCL			
0x3	STATUS			
0x4	FSR			
0x5	PORTA			
0x6	PORTB			
0x7	-			
0x8	PCON			
0x9	BWUCON			
0xA	PCHBUF			
0xB	ABPLCON			
0xC	BPHCON			
0xD	-			
0xE	INTE			
0xF	INTF			
0x10	ADMD			
0x11	ADR			
0x12	ADD			
0x13	ADVREFH			
0x14	ADCR			
0x15	AWUCON			
0x16	PACON			
0x17	ADJMD			
0x18	INTEDG			
0x19	TMRH			
0x1A	ANAEN			
0x1B	RFC			
0x1C	TM4RH			
0x1D	OSCCALH			
0x1E	OSCCALL			
0x1F	INTE2			

Status [7:6] Address	00 (Bank 0)	01 (Bank 1)	10 (Bank 2)	11 (Bank 3)
0x20 ~ 0x3F	General Purpose Register	General Purpose Register	*Mapped to bank0*	*Mapped to Bank1*
0x40 ~ 0x7F	General Purpose Register	*Mapped to bank0*	*Mapped to bank0*	*Mapped to bank0*

Table 1 R-page SFR Address Mapping

F-page SFR can be accessed only by instructions IOST and IOSTR. S-page SFR can be accessed only by instructions SFUN and SFUNR. T-page SFR can be accessed only by instructions TFUN and TFUNR. STATUS[7:6] bank select bits are ignored while F-page, S-page register and T-page register is accessed. The register name and address mapping of F-page , S-page and T-page are depicted in the following table.

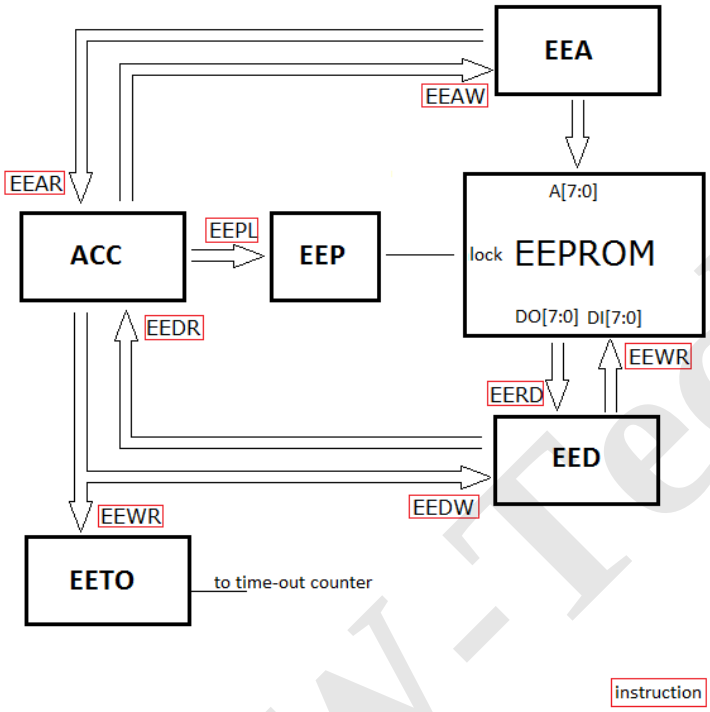
SFR Category Address	F-page SFR	S-page SFR	T-page SFR
0x0	-	TMR1	
0x1	-	T1CR1	
0x2	-	T1CR2	
0x3	-	PWM1DUTY	
0x4	-	PS1CV	
0x5	IOSTA	BZ1CR	
0x6	IOSTB	IRCR	
0x7	-	TBHP	INTE3
0x8	-	TBHD	INTF3
0x9	APHCON	-	
0xA	PS0CV	P2CR1	
0xB	-	-	
0xC	BODCON	PWM2DUTY	
0xD	-	-	
0xE	CMPCR	-	
0xF	PCON1	OSCCR	
0X10	-	-	
0X11	-	P3CR1	
0X12	-	-	
0X13	-	PWM3DUTY	
0X14	-	-	
0X15	-	TMR4	
0X16		T4CR1	
0X17		T4CR2	
0X18		PWM4DUTY	
0X19		PS4CV	
0X1A		TMR5	

SFR Category Address	F-page SFR	S-page SFR	T-page SFR
0X1B		T5CR1	
0X1C		T5CR2	
0X1D		PWM5DUTY	
0X1E		PS5CV	CCPCON
0X1F		TM5RH	PWMDB

Table 2 F-page ,S-page and T-page SFR Address Mapping

2.3 EEPROM Memory

Read and write access to EEPROM memory take place indirectly through 4 special function registers, namely, **EEA, EED, EEP** and **EETO**. EEA register holds the EEPROM address to be access. EED register holds the data to be written, or the data read at the address in EEA. EEP holds the unlock key to write EEPROM data. Another register called EETO holds the EEPROM write time-out period information when EEPROM write time out configuration EEWR_TO is enabled. The following figure shows the block diagram how EEPROM



operates.

7 instructions is provided to control the data flow between these 4 special register and 256-byte EEPROM. EEAR/EEAW are instructions for reading and writing the EEA register. EEDR EEDW are instructions for reading and writing the EED register. EERD/EEWR/EEPL instructions, on the other hand, control the data flow between EEPROM and EED register, according to the EEPROM address provided by EEA. Moreover, The

EETO register will be set when EEWR instruction is issued. If EEPROM write time is longer then the time out period indicated by EETO, the EEPROM write operation will be halt, and a time-out flag will be set (PCON[1]).

. The following table describes the EEPROM instructions.

Mnemonic Operands	Description	Status affected
EEAR	Write ACC from EEA.	-
EEAW	Write EEA from ACC	-
EEDR	Write ACC from EED.	-
EEDW	Write EED from ACC	-
EERD	Read EEPROM with address EEA, and write to EED register	-
EEWR	Write EEPROM with address EEA and data EED. Write EETO from ACC	-
EEPL	Write serial codes to unlock/lock EE write protect	-

Before writing EEPROM data with the EEWR instruction, 3 consecutive code must be written to the EEP register with the EEPL instruction. These 3 code are C9H, 3AH and D3H. After these 3 code are written to EEP register, the EEPROM write is unlocked or locked. The EEPROM write is locked after power-on. Configuration word EEPL_MODE select EEPL instruction is once-type or toggle-type. Only when EEPROM write is unlock will the EEPROM be written by applying the EEWR instruction. The following are example code of EEPROM write unlock process.

```
MOVIA   0xC9
EEPL
MOVIA   0x3A
EEPL
MOVIA   0xD3
EEPL            ; unlock process complete
```

When EEWR is successfully executed and completed, an EEWIF interrupt will be launched if the EEWIE is set to 1 and the global interrupt GIE is enabled.

Note:

1. ***To prevent unwanted halt in a running program if EEPROM writes fail, enabling the watchdog reset or enabling EEPROM write time-out functions is suggested.***
2. ***When user writes data to EEPROM, it is suggested to turn on LVD =2.4V function to monitor VDD.***
3. ***When VDD is under 2.4v, data cannot be written into EEPROM. User can use LVD & LVR function to prevent writing data to EEPROM fail.***

2.3.1 EEA (EEPROM Address Register)

Name	SFR Type	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
EEA	EE	X	EEA[7:0]						
R/W Property		X	R/W						
Initial Value		X	XXXXXXX						

EEA[6:0]: Point to EEPROM address. EEA[6] must be 0. Read/Write this register by instruction EEAR/EEAW.

2.3.2 EED (EEPROM Data Register)

Name	SFR Type	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
EED	EE	EED[7:0]							
R/W Property		R/W							
Initial Value		xxxxxxxx							

EED[7:0]: EEPROM data register . Read/Write this register by instruction EEDR/EEDW.

2.3.3 EEPL (EEPROM write protect Register)

Name	SFR Type	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
EEPL	EE	PL[7:0]							
R/W Property		W							
Initial Value		xxxxxxxx							

PL[7:0]: EEPROM lock/unlock code. Written by instruction EEPL.

2.3.4 EETO (EEPROM Time-Out Register)

Name	SFR Type	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
EETO	EE	-	-	-	-	-	TO2	TO1	TO0
R/W Property		-	-	-	-	-	W	W	W
Initial Value		X	X	X	X	X	X	X	X

TO[2:0]: EEPROM write time out period register. Written by EEWR instruction.

The time-out period is shown in the following table:

TO[2:0]	Time-out period
000	1ms
001	4ms
010	2ms
011	8ms
100	16ms
101	32ms

TO[2:0]	Time-out period
110	16ms
111	32ms

Ex:

```
MOVIA   0x01        ; setting write time-out period as 4ms
EEWR                ; write EEPROM
```

Note:

1. ***Before writing EEPROM, be sure to set initial data into EETO[2:0] at first.***
2. ***EEPROM time-out reference:***

   Time-out period set 4ms, if V_{OPL} = 2.1V ~2.2V

   Time-out period set 2ms, if V_{OPL} = 2.3V ~2.6V

   Time-out period set 1ms, if V_{OPL} ≧ 2.7V

3. Function Description

This chapter will describe the detailed operations of AT8BE62D.

3.1 R-page Special Function Register

3.1.1 INDF (Indirect Addressing Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
INDF	R	0x0	INDF[7:0]							
R/W Property			R/W							
Initial Value			xxxxxxxx							

The register INDF is not physically existed and it is used as indirect addressing mode. Any instruction accessing INDF actually accesses the register pointed by register FSR

3.1.2 TMR0 (Timer0 Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
TMR0	R	0x1	TMR0[7:0]							
R/W Property			R/W							
Initial Value			xxxxxxxx							

When read the register TMR0, it actually read the current running value of Timer0.

Write the register TMR0 will change the current value of Timer0.

Timer0 clock source can be from instruction clock F_{INST}, or from external pin EX_CKI0, or from Low Oscillator Frequency according to T0MD and configuration word setting.

3.1.3 PCL (Low Byte of PC[10:0])

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PCL	R	0x2	PCL[7:0]							
R/W Property			R/W							
Initial Value			0x00							

The register PCL is the least significant byte (LSB) of 11-bit PC. PCL will be increased by one after one instruction is executed except some instructions which will change PC directly. The high byte of PC, i.e. PC[10:8], is not directly accessible. Update of PC[10:8] must be done through register PCHBUF.

For LGOTO instruction, PC[10:0] is from instruction word.

For LCALL instruction, PC[10:0] is from instruction word. Moreover the next PC address, i.e. PC+1, will push onto top of Stack.

3.1.4 STATUS (Status Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
STATUS	R	0x3	BK[1]	BK[0]	GP5	/TO	/PD	Z	DC	C
R/W Property			R/W	R/W	R/W	R/W(*2)	R/W(*1)	R/W	R/W	R/W
Initial Value			0	0	0	1	1	X	X	X

The register STATUS contains result of arithmetic instructions and reasons to cause reset.

C: Carry/Borrow bit

C=1, carry is occurred for addition instruction or borrow is not occurred for subtraction instruction.

C=0, carry is not occurred for addition instruction or borrow is occurred for subtraction instruction.

DC: Half Carry/half Borrow bit

DC=1, carry from the 4th LSB is occurred for addition instruction or borrow from the 4th LSB is not occurred for subtraction instruction.

DC=0, carry from the 4th LSB is not occurred for addition instruction or borrow from the 4th LSB is occurred for subtraction instruction.

Z: Zero bit

Z=1, result of logical operation is zero.

Z=0, result of logical operation is not zero.

/PD: Power down flag bit

/PD=1, after power-up or after instruction CLRWDT is executed.

/PD=0, after instruction SLEEP is executed.

/TO: Time overflow flag bit

/TO=1, after power-up or after instruction CLRWDT or SLEEP is executed.

/TO=0, WDT timeout is occurred.

GP5: General purpose read/write register bit.

BK[1:0]: Bank register is used to select one specific bank of data memory. BK[1:0]=00b, Bank 0 is selected. BK[1:0]=01b, Bank 1 is selected. BK[1:0]=10b, Bank 2 is selected. BK[1:0]=11b, Bank 3 is selected.

(*1) can be cleared by sleep instruction.

(*2) can be set by clrwdt instruction.

3.1.5 FSR (Register File Selection Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
FSR	R	0x4	GP7	FSR[6:0]						
R/W Property			R/W							
Initial Value			0	X	X	X	X	X	X	X

FSR[6:0]: Select one register out of 128 registers of specific Bank.

GP7: general register.

3.1.6 PortA (PortA Data Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PortA	R	0x5	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
R/W Property			R/W							
Initial Value			Data latch value is xxxxxxxx, read value is xxxxxxxx port value(PA7~PA0)							

While reading PortA, it will get the status of the specific pin if that pin is configured as input pin. However, if that pin is configured as output pin, whether it will get the status of the pin or the value of the corresponding output data latch is depend on the configuration option RD_OPT. While writing to PortA, data is written to PA's output data latch.

3.1.7 PortB (PortB Data Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PortB	R	0x6	-	-	PB5	PB4	PB3	PB2	PB1	PB0
R/W Property			-	-	R/W					
Initial Value			Data latch value is xxxxxxxx, read value is xxxxxxxx port value(PB5~PB0)							

While reading PortB, it will get the status of the specific pin if that pin is configured as input pin. However, if that pin is configured as output pin, whether it will get the status of the pin or the value of the corresponding output data latch is depending on the configuration option RD_OPT. While writing to PortB, data is written to PB's output data latch.

3.1.8 PCON (Power Control Register)

Name	SFR Type	Addr	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PCON	R	0x8	WDTEN	/PLPA4	LVDEN	/PHPA5	LVREN	GP2	EEW_ERR	EELOCK
R/W Property			R/W						R	R
Initial Value			1	1	0	1	1	0	0	1

GP2: General read/write register bits.

EELOCK: EEPROM write LOCK flag.

EELOCK=1, the EEPROM write is in lock state.

EELOCK=0, the EEPROM write is in unlock state.

EEW_ERR: EEPROM write Time-out flag. (Note: EEWR_TO configuration must be enabled)

EEW_ERR=1, the EEPROM write is time-out.

EEW_ERR=0, the EEPROM write is not time-out.

LVREN: Enable/disable LVR.

LVREN=1, enable LVR.

LVREN=0, disable LVR.

/PHPA5: Disable/enable PA5 Pull-High resistor.

/PHPA5=1, disable PA5 Pull-High resistor.

/PHPA5=0, enable PA5 Pull-High resistor.

LVDEN: Enable/disable LVD.

LVDEN=1, enable LVD.

LVDEN=0, disable LVD.

/PLPA4: Disable/enable PA4 Pull-Low resistor.

/PLPA4=1, disable PA4 Pull-Low resistor.

/PLPA4=0, enable PA4 Pull-Low resistor.

WDTEN: Enable/disable WDT.

WDTEN=1, enable WDT.

WDTEN=0, disable WDT.

3.1.9 BWUCON (PortB Wake-up Control Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
BWUCON	R	0x9	-	-	WUPB5	WUPB4	WUPB3	WUPB2	WUPB1	WUPB0
R/W Property			-	-	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value			X	X	0	0	0	0	0	0

WUPBx: Enable/disable PBx wake-up function, 0 ≤ x ≤ 5.

WUPBx=1, enable PBx wake-up function.

WUPBx=0, disable PBx wake-up function.

3.1.10 PCHBUF (High Byte of PC)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PCHBUF	R	0xA	-	XSPD_STP	-	-	-	PCHBUF[2:0]		
R/W Property			-	W	-	-	-	W		
Initial Value			X	0	X	X	X	0		

PCHBUF[2:0]: Buffer of the 10^{th} ~ 8^{th} bit of PC.

XSPD_STP: Write 1 to stop crystal 32.768K speed-up function, write-only.

3.1.11 ABPLCON (PortA/PortB Pull-Low Resistor Control Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ABPLCON	R	0xB	/PLPB3	/PLPB2	/PLPB1	/PLPB0	/PLPA3	/PLPA2	/PLPA1	/PLPA0
R/W Property			R/W							
Initial Value			1	1	1	1	1	1	1	1

/PLPAx: Disable/enable PAx Pull-Low resistor, 0 ≤ x ≤ 3.

/PLPAx=1, disable PAx Pull-Low resistor.

/PLPAx=0, enable PAx Pull-Low resistor.

/PLPBx: Disable/enable PBx Pull-Low resistor, 0 ≤ x ≤ 3.

/PLPBx=1, disable PBx Pull-Low resistor.

/PLPBx=0, enable PBx Pull-Low resistor.

3.1.12 BPHCON (PortB Pull-High Resistor Control Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
BPHCON	R	0xC	-	-	/PHPB5	/PHPB4	/PHPB3	/PHPB2	/PHPB1	/PHPB0
R/W Property			-	-	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value			X	X	1	1	1	1	1	1

/PHPBx: Disable/enable PBx Pull-High resistor, 0 ≤ x ≤ 5.

/PHPBx=1, disable PBx Pull-High resistor.

/PHPBx=0, enable PBx Pull-High resistor.

3.1.13 INTE (Interrupt Enable Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
INTE	R	0xE	INT1IE	WDTIE	-	LVDIE	T1IE	INT0IE	PABIE	T0IE
R/W Property			R/W	R/W	-	R/W	R/W	R/W	R/W	R/W
Initial Value			0	0	X	0	0	0	0	0

T0IE: Timer0 overflow interrupt enable bit.

T0IE=1, enable Timer0 overflow interrupt.

T0IE=0, disable Timer0 overflow interrupt.

PABIE: PortA/PortB input change interrupt enable bit.

PABIE=1, enable PortA/PortB input change interrupt.

PABIE=0, disable PortA/PortB input change interrupt.

INT0IE: External interrupt 0 enable bit.

INT0IE=1, enable external interrupt 0.

INT0IE=0, disable external interrupt 0.

T1IE: Timer1 underflow interrupt enable bit.

T1IE=1, enable Timer1 underflow interrupt.

T1IE=0, disable Timer1 underflow interrupt.

LVDCMPIE: Low-voltage detector interrupt enable bit.

LVDCMPIE=1, enable low-voltage detector interrupt.

LVDCMPIE=0, disable low-voltage detector interrupt.

WDTIE: WDT timeout interrupt enable bit.

WDTIE=1, enable WDT timeout interrupt.

WDTIE=0, disable WDT timeout interrupt.

INT1IE: External interrupt 1 enable bit.

INT1IE=1, enable external interrupt 1.

INT1IE=0, disable external interrupt 1.

3.1.14 INTF (Interrupt Flag Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
INTF	R	0xF	INT1IF	WDTIF	-	LVDIF	T1IF	INT0IF	PABIF	T0IF
R/W Property			R/W	R/W	-	R/W	R/W	R/W	R/W	R/W
Initial Value(note*)			0	0	X	0	0	0	0	0

T0IF: Timer0 overflow interrupt flag bit.

T0IF=1, Timer0 overflow interrupt is occurred.

T0IF must be clear by firmware.

PABIF: PortA/PortB input change interrupt flag bit.

PABIF=1, PortA/PortB input change interrupt is occurred.

PABIF must be clear by firmware.

INT0IF: External interrupt 0 flag bit.

INT0IF=1, external interrupt 0 is occurred.

INT0IF must be clear by firmware.

T1IF: Timer1 underflow interrupt flag bit.

T1IF=1, Timer1 underflow interrupt is occurred.

T1IF must be clear by firmware.

LVDIF: Low-voltage detector interrupt flag bit.

LVDIF=1, Low-voltage detector interrupt is occurred.

LVDIF must be clear by firmware.

WDTIF: WDT timeout interrupt flag bit.

WDTIF=1, WDT timeout interrupt is occurred.

WDTIF must be clear by firmware.

INT1IF: External interrupt 1 flag bit.

INT1IF=1, external interrupt 1 is occurred.

INT1IF must be clear by firmware.

Note: When corresponding INTE bit is not enabled, the read interrupt flag is 0.

3.1.15 ADMD (ADC mode Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADMD	R	0x10	ADEN	START	EOC	GCHS	CHS3	CHS2	CHS1	CHS0
R/W Property			R/W	W	R	R/W	R/W	R/W	R/W	R/W
Initial Value			0	0	1	0	0	0	0	0

ADEN: ADC enable bit.

ADEN=1, ADC is enabled.

START: Start an ADC conversion session.

When write 1 to this bit, start to execute ADC converting. This bit is write-only. Read this bit will get 0.

EOC: ADC status bit, read-only.

EOC=1 : ADC is end-of-convert, the ADC data present in ADR and ADD is available.

EOC=0 : ADC is in procession.

GCHS: ADC global channel select bit.

GCHS=0 : disable all ADC input channel.

GCHS=1 : enable ADC input channel.

CHS3~0: ADC input channel select bits.

0000=select PA0 pad as ADC input,

0001=select PA1 pad as ADC input,

0010=select PA2 pad as ADC input,

0011=select PA3 pad as ADC input,

0100=select PA4 pad as ADC input,

0101=select PB0 pad as ADC input,

0110=select PB1 pad as ADC input,

0111=select PB2 pad as ADC input,

1000=select PB3 pad as ADC input,

1001=select PB4 pad as ADC input,

1010=select PB5 pad as ADC input.

1011=select 1/4 VDD as ADC input.

1100=select VSS as ADC input.

3.1.16 ADR (ADC clock, ADC interrupt flag and ADC LSB output Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADR	R	0x11	ADIF	ADIE	ADCK1	ADCK0	AD3	AD2	AD1	AD0
R/W Property			R/W	R/W	R/W	R/W	R	R	R	R
Initial Value			0	0	0	0	X	X	X	X

ADIF: ADC interrupt flag bit.

ADIF=1, ADC end-of-convert interrupt is occurred.

ADIF must be clear by firmware.

ADIE: ADC end-of-convert interrupt enable bit.

ADIE=1 : enable ADC interrupt.

ADIE=0 : disable ADC interrupt.

ADCK1~0: ADC clock select.

00: ADC clock= F_{INST}/16, 01: ADC clock= F_{INST}/8, 10: ADC clock= F_{INST}/1, 11: ADC clock= F_{INST}/2.

AD3~0: 12-bit low-nibble ADC data buffer.

3.1.17 ADD (ADC output data Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADD	R	0x12	AD11	AD10	AD9	AD8	AD7	AD6	AD5	AD4
R/W Property			R	R	R	R	R	R	R	R
Initial Value			1	0	0	0	0	0	0	0

AD11~4: High-byte ADC data buffer.

3.1.18 ADVREFH (ADC high reference voltage Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADVREFH	R	0x13	EVHENB	-	-	-	-	-	VHS1	VHS0
R/W Property			R/W	-	-	-	-	-	R/W	R/W
Initial Value			0	X	X	X	X	X	1	1

EVHENB: ADC reference high voltage (VREFH) select control bit.

EVHENB=0: ADC reference high voltage is internal generated, the voltage selected depends on VHS1~0.

EVHENB=1: ADC reference high voltage is supplied by external pin PA0.

VHS1~0: ADC internal reference high voltage select bits.

11: VREFH=VDD, 10: VREFH=4V, 01: VREFH=3V, 00: VREFH=2V.

3.1.19 ADCR (Sampling pulse and ADC bit Register)

Name	SFR Type	Addr	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADCR	R	0x14	-	PBCON5	PBCON4	PBCON3	SHCK1	SHCK0	ADCR1	ADCR0
R/W Property			-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value			X	0	0	0	1	0	1	0

SHCK1~0: Sampling pulse width select.

00: 1 ADC clock, 01: 2 ADC clock, 10: 4 ADC clock, 11: 8 ADC clock.

ADCR1~0: ADC conversion bit no. select.

00: 8-bit ADC, 01: 10-bit ADC, 1x: 12-bit ADC.

PBCONx: PB analog pin select, 3 ≤ x ≤ 5.

0=PBx can be analog ADC input or digital I/O pin.

1=PBx is pure analog ADC input pin for power-saving.

3.1.20 AWUCON (PortA Wake-up Control Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
AWUCON	R	0x15	WUPA7	WUPA6	WUPA5	WUPA4	WUPA3	WUPA2	WUPA1	WUPA0
R/W Property			R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value			0	0	0	0	0	0	0	0

WUPAx: Enable/disable PAx wake-up function, 0 ≤ x ≤ 7.

WUPAx=1, enable PAx wake-up function.

WUPAx=0, disable PAx wake-up function.

3.1.21 PACON (ADC analog pin Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PACON	R	0x16	PBCON2	PBCON1	PBCON0	PACON4	PACON3	PACON2	PACON1	PACON0
R/W Property			R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value			0	0	0	0	0	0	0	0

PACONx: PA analog pin select, 0 ≤ x ≤ 4.

0=PAx can be analog ADC input or digital I/O pin.

1=PAx is pure analog ADC input pin for power-saving.

PBCONx: PB analog pin select, 0 ≤ x ≤ 2.

0=PBx can be analog ADC input or digital I/O pin.

1=PBx is pure analog ADC input pin for power-saving.

3.1.22 ADJMD (ADC offset Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADJMD	R	0x17	-	-	ADJ_SIGN	ADJ[4]	ADJ[3]	ADJ[2]	ADJ[1]	ADJ[0]
R/W Property			-	-	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value			X	X	0	0	0	0	0	0

ADJ[x]: adjustment bit select, 0 ≤ x ≤ 4.

00000 = offset 0mV

11111 = offset 11mV.

ADJ_SIGN: adjustment sign bit

0 = adc data decrease

1 = adc data increase

Note: For application, please refer to NYIDE example code "ADC_Interrupt _AutoK".

.

3.1.23 INTEDG (Interrupt Edge Register)

Name	SFR Type	Addr	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
INTEDG	R	0x18	INT2DEG	EIS2	EIS1	EIS0	INT1G1	INT1G0	INT0G1	INT0G0
R/W Property			R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value			0	0	0	0	0	1	0	1

EIS2: External interrupt 2 select bit

EIS2=1, PA5 is external interrupt 2.

EIS2=0, PA5 is GPIO.

EIS1: External interrupt 1 select bit

EIS1=1, PB1/PA3 is external interrupt 1.

EIS1=0, PB1/PA3 is GPIO.

EIS0: External interrupt 0 select bit

EIS0=1, PB0/PB5 is external interrupt 0.

EIS0=0, PB0/PB5 is GPIO.

INT1G1~0: INT1 edge trigger select bit.

00: reserved, 01: rising edge, 10: falling edge, 11: rising/falling edge.

INT0G1~0: INT0 edge trigger select bit.

00: reserved, 01: rising edge, 10: falling edge, 11: rising/falling edge.

INT2DEG: INT2 edge trigger select bit.

0: falling edge, 1: rising edge.

3.1.24 TMRH (Timer High Byte Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
TMRH	R	0x19	-	-	TMR19	TMR18	PWM2 DUTY9	PWM2 DUTY8	PWM1 DUTY9	PWM1 DUTY8
R/W Property			-	-	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value			X	X	X	X	X	X	X	X

TMR19~8: Timer1 MSB 2 bits. Write these 2 bits will overwrite the 10-bit Timer1 load value of bit 9 and 8. Read these 2 bits will get the Timer1 bit9-8 current value.

PWM2DUTY9~8: PWM2 duty data MSB 2 bits.

PWM1DUTY9~8: PWM1 duty data MSB 2 bits.

3.1.25 ANAEN (Analog Circuit Enable Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ANAEN	R	0x1A	CMPEN	-	-	-	-	-	-	-
R/W Property			R/W	-	-	-	-	-	-	-
Initial Value			0	X	X	X	X	X	X	X

CMPEN: Enable/disable voltage comparator.

CMPEN=1, enable voltage comparator.

CMPEN=0, disable voltage comparator.

3.1.26 RFC (RFC Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RFC	R	0x1B	RFCEN	-	-	-	PSEL[3:0]			
R/W Property			R/W	-	-	-	R/W			
Initial Value			0	X	X	X	0			

RFCEN: Enable/disable RFC function.

RFCEN=1, enable RFC function.

RFCEN=0, disable RFC function.

PSEL[3:0]: Select RFC pad.

PSEL[3:0]	RFC PAD
0000	PA0
0001	PA1
0010	PA2
0011	PA3
0100	PA4

PSEL[3:0]	RFC PAD
0101	PA5
0110	PA6
0111	PA7
1000	PB0
1001	PB1
1010	PB2
1011	PB3
1100	PB4
1101	PB5
1110	-
1111	-

Table 3 RFC pad select

3.1.27 TM4RH (Timer4 High Byte Register)

Name	SFR Type	Addr	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
TM4RH	R	0x1C	TMR49	TMR48	-	-	PWM4 DUTY9	PWM4 DUTY8	PWM3 DUTY9	PWM3 DUTY8
R/W Property			R/W	R/W	-	-	R/W	R/W	R/W	R/W
Initial Value			X	X	X	X	X	X	X	X

TMR49, TMR48: Timer4 MSB 2 bits. Write these 2 bits will overwrite the 10-bit Timer4 load value of bit 9 and 8. Read these 2 bits will get the Timer4 bit9-8 current value.

PWM4DUTY9~8: PWM4 duty data MSB 2 bits.

PWM3DUTY9~8: PWM3 duty data MSB 2 bits.

3.1.28 OSCCAL (Internal Oscillator Calibration)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
OSCCALH	R	0x1D	-	-	-	-	-	OSC CAL10	OSC CAL9	OSC CAL8
R/W Property			-	-	-	-	-	R/W	R/W	R/W
Initial Value			X	X	X	X	X	HIRC 8 bits option trim date bit[7:5]		

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
OSCCALL	R	0x1E	OSC CAL7	OSC CAL6	OSC CAL5	OSC CAL4	OSC CAL3	OSC CAL2	OSC CAL1	OSC CAL0
R/W Property			R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value			I_HRC 8 bits option trim data bit[4:0]					1	0	0

OSCCAL10~0: I_HRC 8 bits calibration data load to OSCCAL10~OSCCAL3 and user can adjust OSCCAL10~0 at slow mode.

3.1.29 INTE2 (Interrupt Enable and Flag 2nd. Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
INTE2	R	0x1F	INT2IF	T4IF	-	-	INT2IE	T4IE	-	-
R/W Property			R/W	R/W	-	-	R/W	R/W	-	-
Initial Value			0	0	X	X	0	0	X	X

INT2IF: External interrupt 2 flag bit.
INT2IF=1, external interrupt 2 is occurred.
INT2IF must be clear by firmware.

T4IF: Timer4 underflow interrupt flag bit.
T4IF=1, Timer4 underflow interrupt is occurred.
T4IF must be clear by firmware.

INT2IE: External interrupt 2 enable bit.
INT2IE=1, enable external interrupt 2.
INT2IE=0, disable external interrupt 2.

T4IE: Timer4 underflow interrupt enable bit.
T4IE=1, enable Timer4 underflow interrupt.
T4IE=0, disable Timer4 underflow interrupt.

3.2 T0MD Register

T0MD is a readable/writeable register which is only accessed by instruction T0MD / T0MDR.

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
T0MD	-	-	LCKTM0	GP6	T0CS	T0CE	PS0WDT	PS0SEL[2:0]		
R/W Property			R/W							
Initial Value(note*)			0	0	1	1	1	111		

PS0SEL[2:0]: Prescaler0 dividing rate selection. The rate depends on Prescaler0 is assigned to Timer0 or WDT. When Prescaler0 is assigned to WDT, the dividing rate is dependent on which timeout mechanism is selected.

PS0SEL[2:0]	Dividing Rate		
	PS0WDT=0 (Timer0)	PS0WDT=1 (WDT Reset)	PS0WDT=1 (WDT Interrupt)
000	1:2	1:1	1:2
001	1:4	1:2	1:4

PS0SEL[2:0]	Dividing Rate		
	PS0WDT=0 (Timer0)	PS0WDT=1 (WDT Reset)	PS0WDT=1 (WDT Interrupt)
010	1:8	1:4	1:8
011	1:16	1:8	1:16
100	1:32	1:16	1:32
101	1:64	1:32	1:64
110	1:128	1:64	1:128
111	1:256	1:128	1:256

Table 4 Prescaler0 Dividing Rate

PS0WDT: Prescaler0 assignment.

PS0WDT=1, Prescaler0 is assigned to WDT.

PS0WDT=0, Prescaler0 is assigned to Timer0.

Note: Always set PS0WDT and PS0SEL[2:0] before enabling watchdog or timer interrupt, or reset or interrupt may be falsely triggered.

T0CE: Timer0 external clock edge selection.

T0CE=1, Timer0 will increase one while high-to-low transition occurs on pin EX_CKI0.

T0CE=0, Timer0 will increase one while low-to-high transition occurs on pin EX_CKI0.

Note: T0CE is also applied to Low Oscillator Frequency as Timer0 clock source condition.

T0CS: Timer0 clock source selection.

T0CS=1, External clock on pin EX_CKI0 or Low Oscillator Frequency (I_LRC or E_LXT) is selected.

T0CS=0, Instruction clock F_{INST} is selected.

GP6: General register.

LCKTM0: When T0CS=1, timer 0 clock source can be optionally selected to be low-frequency oscillator.

T0CS=0, Instruction clock F_{INST} is selected as Timer0 clock source.

T0CS=1, LCKTM0=0, external clock on pin EX_CKI0 is selected as Timer0 clock source.

T0CS=1, LCKTM0=1, Low Oscillator Frequency (I_LRC or E_LXT, depends on configuration word Low Oscillator Frequency) output replaces pin EX_CKI0 as Timer0 clock source.

Note: For more detail descriptions of Timer0 clock source select, please see Timer0 section.

3.3 F-page Special Function Register

3.3.1 IOSTA (PortA I/O Control Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IOSTA	F	0x5	IOPA7	IOPA6	IOPA5	IOPA4	IOPA3	IOPA2	IOPA1	IOPA0
R/W Property			R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value			1	1	1	1	1	1	1	1

IOPAx: PAx I/O mode selection, 0 ≤ x ≤ 7.

IOPAx=1, PAx is input mode.

IOPAx=0, PAx is output mode.

3.3.2 IOSTB (PortB I/O Control Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IOSTB	F	0x6	-	-	IOPB5	IOPB4	IOPB3	IOPB2	IOPB1	IOPB0
R/W Property			-	-	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value			X	X	1	1	1	1	1	1

IOPBx: PBx I/O mode selection, 0 ≤ x ≤ 5.

IOPBx=1, PBx is input mode.

IOPBx=0, PBx is output mode.

3.3.3 APHCON (PortA Pull-High Resistor Control Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
APHCON	F	0x9	/PHPA7	/PHPA6	/PLPA5	/PHPA4	/PHPA3	/PHPA2	/PHPA1	/PHPA0
R/W Property			R/W							
Initial Value			1	1	1	1	1	1	1	1

/PHPAx: Enable/disable Pull-High resistor of PAx, x=0~4, 6~7.

/PHPAx=1, disable Pull-High resistor of PAx.

/PHPAx=0, enable Pull-High resistor of PAx.

/PLPA5: Enable/disable Pull-Low resistor of PA5.

/PLPA5=1, disable Pull-Low resistor of PA5.

/PLPA5=0, enable Pull-Low resistor of PA5.

Note: When PA6 and PA7 are used as crystal oscillator pads, the Pull-High resistor should not enable. Or the oscillation may fail.

3.3.4 PS0CV (Prescaler0 Counter Value Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PS0CV	F	0xA	PS0CV[7:0]							
R/W Property			R							
Initial Value			1	1	1	1	1	1	1	1

While reading PS0CV, it will get current value of Prescaler0 counter.

3.3.5 BODCON (PortB Open-Drain Control Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
BODCON	F	0xC	-	-	ODPB5	ODPB4	ODPB3	ODPB2	ODPB1	ODPB0
R/W Property			-	-	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value			X	X	0	0	0	0	0	0

ODPBx: Enable/disable open-drain of PBx, 0 ≤ x ≤ 5.

ODPBx=1, enable open-drain of PBx.

ODPBx=0, disable open-drain of PBx.

3.3.6 CMPCR (Comparator voltage select Control Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CMPCR	F	0xE	GP7	RBIAS_H	RBIAS_L	CMPF_INV	PS1	PS0	NS1	NS0
R/W Property			R/W							
Initial Value			0	0	0	0	1	1	0	0

NS[1:0]: Comparator inverting input select.

NS[1:0]	Inverting input
00	PA1
01	PA3
10	Bandgap (0.6V)
11	Vref

PS[1:0]: Comparator non-inverting input select

PS[1:0]	Non-inverting input
00	PA0
01	PA2
10	Vref
11	---

CMPF_INV: Comparator output inverse control bit.

CMPF_INV = 1, Inverse comparator output.

CMPF_INV = 0, Non-inverse comparator output.

RBIAS_L, RBIAS_H: Set corresponding voltage reference levels

(please refer to chapter 3.16.1)

Note: RBIAS_H and RBIAS_L must be set as "0" to avoid power consumption in Halt mode or Standby mode.

3.3.7 PCON1 (Power Control Register1)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PCON1	F	0xF	GIE	LVDOUT	LVDS3	LVDS2	LVDS1	LVDS0	GP1	T0EN
R/W Property			R/W(1*)	R	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value			0	X	1	1	1	1	0	1

T0EN: Enable/disable Timer0.

T0EN=1, enable Timer0.

T0EN=0, disable Timer0.

LVDS3~0: Select one of the 14 LVD voltage.

LVDS[3:0]	Voltage
0000	1.9V
0001	2.0V
0010	2.2V
0011	2.4V
0100	2.6V
0101	2.8V
0110	2.9V
0111	3.0V
1000	3.15V
1001	3.30V
1010	3.45V
1011	3.60V
1100	3.75V
1101	3.90V
1110	4.05V
1111	4.15V

Table 7 LVD voltage select

LVDOUT: Low voltage detector output, read-only.

GIE: Global interrupt enable bit.

GIE=1, enable all unmasked interrupts.

GIE=0, disable all interrupts.

GP5, GP1: General purpose read/write register.

(1*) : set by instruction ENI, clear by instruction DISI, read by instruction IOSTR.

3.4 S-page Special Function Register

3.4.1 TMR1 (Timer1 Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
TMR1	S	0x0	TMR1[7:0]							
R/W Property			R/W							
Initial Value			XXXXXXXX							

When reading register TMR1, it will obtain current value of 10-bit down-count Timer1 at TMR1[9:0]. When writing register TMR1, it will write data from TMRH[5:4] and Timer1 reload register to TMR1[9:0] current content.

3.4.2 T1CR1 (Timer1 Control Register1)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
T1CR1	S	0x1	PWM1OEN	PWM1OAL	TM1OE	-	TM1_HRC	T1OS	T1RL	T1EN
R/W Property			R/W	R/W	R/W	-	R/W	R/W	R/W	R/W
Initial Value			0	0	0	X	0	0	0	0

This register is used to configure Timer1 functionality.

T1EN: Enable/disable Timer1.

T1EN=1, enable Timer1.

T1EN=0, disable Timer1.

T1RL: Configure Timer1 down-count mechanism while Non-Stop mode is selected (T1OS=0).

T1RL=1, initial value is reloaded from reload register TMR1[9:0].

T1RL=0, continuous down-count from 0x3FF when underflow is occurred.

T1OS: Configure Timer1 operating mode while underflow is reached.

T1OS=1, One-Shot mode. Timer1 will count once from the initial value to 0x00.

T1OS=0, Non-Stop mode. Timer1 will keep down-count after underflow.

T1OS	T1RL	Timer1 Down-Count Functionality
0	0	Timer1 will count from reload value down to 0x00. When underflow is reached, 0x3FF is reloaded and continues down-count.
0	1	Timer1 will count from reload value down to 0x00. When underflow is reached, reload value is reloaded and continues to down-count.
1	x	Timer1 will count from initial value down to 0x00. When underflow is reached, Timer1 will stop down-count.

Table 8 Timer1 Functionality

TM1_HRC: Timer1 clock source selection.

TM1_HRC =1, PWM1,2,3 & Timer 1 clock source is High Oscillator Clock.

TM1_HRC =0, PWM1,2,3 & Timer 1 clock source selection depends on T1CS register bit.

PWM1OAL: Define PWM1 output active state.

PWM1OAL=1, PWM1 output is active low.

PWM1OAL=0, PWM1 output is active high.

PWM1OEN: Enable/disable PWM1 output.

PWM1OEN=1, PWM1 output will be present on PB1/PB4.

PWM1OEN=0, PB1/PB4 is GPIO.

TM1OE: Enable/disable Timer1 match output, T1OUT toggle output when Timer1 underflow occurs.

TM1OE=1, enable T1OUT output to pad PB4.

TM1OE=0, PB4 is GPIO.

3.4.3 T1CR2 (Timer1 Control Register2)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
T1CR2	S	0x2	-	-	T1CS	T1CE	/PS1EN	PS1SEL[2:0]		
R/W Property			-	-	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value			X	X	1	1	1	1	1	1

This register is used to configure Timer1 functionality.

PS1SEL[2:0]: Prescaler1 dividing rate selection.

PS1SEL[2:0]	Dividing Rate
000	1:2
001	1:4
010	1:8
011	1:16
100	1:32
101	1:64
110	1:128
111	1:256

Table 9 Prescaler1 Dividing Rate

Note: Always set PS1SEL[2:0] at /PS1EN=1, or interrupt may be falsely triggered.

/PS1EN: Disable/enable Prescaler1.

/PS1EN=1, disable Prescaler1.

/PS1EN=0, enable Prescaler1.

Note: When High Oscillator are selected as clock source must set /PS1EN=1(disable Prescaler1)

T1CE: Timer1 external clock edge selection.

T1CE=1, Timer1 will decrease one while high-to-low transition occurs on pin EX_CKI0.

T1CE=0, Timer1 will decrease one while low-to-high transition occurs on pin EX_CKI0.

T1CS: Timer1 clock source selection.

T1CS=1, External clock on pin EX_CKI0 is selected.

T1CS=0, Instruction clock or High Oscillator are selected.

3.4.4 PWM1DUTY (PWM1 Duty Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PWM1DUTY	S	0x3	PWM1DUTY[7:0]							
R/W Property			W							
Initial Value			XXXXXXXX							

The reload value of 10-bit Timer1 stored on registers TMRH[5:4] and TMR1[7:0] is used to define the PWM1 frame rate, and registers TMRH[1:0] and PWM1DUTY[7:0] is used to define the duty cycle of PWM1.

3.4.5 PS1CV (Prescaler1 Counter Value Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PS1CV	S	0x4	PS1CV[7:0]							
R/W Property			R							
Initial Value			1	1	1	1	1	1	1	1

While reading PS1CV, it will get current value of Prescaler1 counter.

3.4.6 BZ1CR (Buzzer1 Control Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
BZ1CR	S	0x5	BZ1EN	-	-	-	BZ1FSEL[3:0]			
R/W Property			W	-	-	-	W			
Initial Value			0	X	X	X	1	1	1	1

BZ1FSEL[3:0]: Frequency selection of BZ1 output.

BZ1FSEL[3:0]	BZ1 Frequency Selection	
	Clock Source	Dividing Rate
0000	Prescaler1 output	1:2
0001		1:4
0010		1:8
0011		1:16
0100		1:32
0101		1:64
0110		1:128
0111		1:256
1000	Timer1 output	Timer1 bit 0

BZ1FSEL[3:0]	BZ1 Frequency Selection	
	Clock Source	Dividing Rate
1001		Timer1 bit 1
1010		Timer1 bit 2
1011		Timer1 bit 3
1100		Timer1 bit 4
1101		Timer1 bit 5
1110		Timer1 bit 6
1111		Timer1 bit 7

Table 10 Buzzer1 Output Frequency Selection

BZ1EN: Enable/Disable BZ1 output.

BZ1EN=1, enable Buzzer1.

BZ1EN=0, disable Buzzer1.

3.4.7 IRCR (IR Control Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IRCR	S	0x6	IROSC358M	-	-	-	-	IRCSEL	IRF57K	IREN
R/W Property			W	-	-	-	-	W	W	W
Initial Value			0	X	X	X	X	0	0	0

IREN: Enable/Disable IR carrier output.

IREN=1, enable IR carrier output.

IREN=0, disable IR carrier output.

IRF57K: Selection of IR carrier frequency.

IRF57K=1, IR carrier frequency is 57KHz.

IRF57K=0, IR carrier frequency is 38KHz.

IRCSEL: Polarity selection of IR carrier.

IRCSEL=0, IR carrier will be generated when I/O pin data is 1.

IRCSEL=1, IR carrier will be generated when I/O pin data is 0.

IROSC358M: When external crystal is used, this bit is determined according to what kind of crystal is used.

This bit is ignored if internal high frequency oscillation is used.

IROSC358M=1, crystal frequency is 3.58MHz.

IROSC358M=0, crystal frequency is 455KHz.

Note:

1. Only high oscillation (F_{HOSC}) (See section 3.17) can be used as IR clock source.

2. Division ratio for different oscillation type.

OSC. Type	57KHz	38KHz	Conditions
I_HRC(13.6MHz)	64	96	The input clock to IR module is 13.6M/4=3.4MHz 3.4M/64 =53KHz , 3.4M/96 =35.4KHz
I_HRC(14.4MHz)	64	96	The input clock to IR module is 14.4M/4=3.6MHz 3.6M/64 =56.3KHz , 3.6M/96 =37.5KHz
I_HRC(16M/8M/4M/2M/1MHz)	64	96	The input clock to IR module is 16M/4=4MHz 4M/64 =62.5KHz , 4M/96 =41.67KHz
I_HRC (19.2MHz)	64	96	The input clock to IR module is 19.2M/5=3.84MHz 3.84M/64 =60KHz , 3.84M/96 =40KHz
I_HRC(20.8MHz)	64	96	The input clock to IR module is 20.8M/5=4.16MHz 4.16M/64 =65KHz , 4.16M/96 =43.3KHz
Xtal 3.58MHz	64	96	Xtal mode & IROSC358M=1
Xtal 455KHz	8	12	Xtal mode & IROSC358M=0

Table 11 Division ratio for different oscillation type

3.4.8 TBHP (Table Access High Byte Address Pointer Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
TBHP	S	0x7	-	-		-	-	TBHP2	TBHP1	TBHP0
R/W Property			-	-		-	-	R/W	R/W	R/W
Initial Value			X	X	X	X	X	X	X	X

When instruction CALLA, GOTOA or TABLEA is executed, the target address is constituted by TBHP[2:0] and ACC. ACC is the Low Byte of PC[10:0] and TBHP[2:0] is the high byte of PC[10:0].

3.4.9 TBHD (Table Access High Byte Data Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
TBHD	S	0x8	-	-	TBHD5	TBHD4	TBHD3	TBHD2	TBHD1	TBHD0
R/W Property			-	-	R	R	R	R	R	R
Initial Value			X	X	X	X	X	X	X	X

When instruction TABLEA is executed, high byte of content of addressed ROM is loaded into TBHD[5:0] register. The Low Byte of content of addressed ROM is loaded to ACC.

3.4.10 P2CR1 (PWM2 Control Register1)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P2CR1	S	0xA	PWM2OEN	PWM2OAL	-	-	-	-	-	-
R/W Property			R/W	R/W	-	-	-	-	-	-
Initial Value			0	0	X	X	X	X	X	X

PWM2OAL: Define PWM2 output active state.

PWM2OAL=1, PWM2 output is active low.

PWM2OAL=0, PWM2 output is active high.

PWM2OEN: Enable/disable PWM2 output.

PWM2OEN=1, PWM2 output will be present on PA4/PB5.

PWM2OEN=0, PA4/PB5 is GPIO.

3.4.11 PWM2DUTY (PWM2 Duty Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PWM2DUTY	S	0xC	PWM2DUTY[7:0]							
R/W Property			W							
Initial Value			XXXXXXXX							

The reload value of 10-bit Timer1 stored on registers TMRH[5:4] and TMR1[7:0] is used to define the PWM2 frame rate, and registers TMRH[3:2] and PWM2DUTY[7:0] is used to define the duty cycle of PWM2.

3.4.12 OSCCR (Oscillation Control Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
OSCCR	S	0xF	-	CMPOE	-	-	OPMD[1:0]		STPHOSC	SELHOSC
R/W Property			-	R/W	-	-	R/W		R/W	R/W
Initial Value			X	0	X	X	00		0	1

SELHOSC: Selection of system oscillation (F_{OSC}).

SELHOSC=1, F_{OSC} is high-frequency oscillation (F_{HOSC}).

SELHOSC=0, F_{OSC} is low-frequency oscillation (F_{LOSC}).

STPHOSC: Disable/enable high-frequency oscillation (F_{HOSC}).

STPHOSC=1, F_{HOSC} will stop oscillation and be disabled.

STPHOSC=0, F_{HOSC} keep oscillation.

OPMD[1:0]: Selection of operating mode.

OPMD[1:0]	Operating Mode
00	Normal mode
01	Halt mode
10	Standby mode
11	reserved

Table 15 Selection of Operating Mode by OPMD[1:0]

CMPOE: Disable/enable comparator output to pad PB3.

CMPOE=1, enable comparator output to pad PB3.

CMPOE=0, disable comparator output to pad PB3.

Note: Comparator output to pad PB3 has higher priority than pwm1/buzzer1 output to pad PB3.

Note: STPHOSC cannot be changed with SELHOSC or OPMD at the same time. STPHOSC cannot be changed with OPMD at the same time during SELHOSC1.

3.4.13 P3CR1 (PWM3 Control Register1)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P3CR1	S	0x11	PWM3OEN	PWM3OAL	-	-	-	-	-	-
R/W Property			R/W	R/W	-	-	-	-	-	
Initial Value			0	0	X	X	X	X	X	X

PWM3OAL: Define PWM3 output active state.

PWM3OAL=1, PWM3 output is active low.

PWM3OAL=0, PWM3 output is active high.

PWM3OEN: Enable/disable PWM3 output.

PWM3OEN=1, PWM3 output will be present on PA2/PA7.

PWM3OEN=0, PA2/PA7 is GPIO.

3.4.14 PWM3DUTY (PWM3 Duty Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PWM3DUTY	S	0x13	PWM3DUTY[7:0]							
R/W Property			W							
Initial Value			XXXXXXXX							

The reload value of 10-bit Timer1 stored on registers TMRH[5:4] and TMR1[7:0] is used to define the PWM3 frame rate, and registers TM4RH[1:0] and PWM3DUTY[7:0] is used to define the duty cycle of PWM3.

3.4.15 TMR4 (Timer4 Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
TMR4	S	0x15	TMR4[7:0]							
R/W Property			R/W							
Initial Value			XXXXXXXX							

When reading register TMR4, it will obtain current value of 10-bit down-count Timer4 at TMR4[7:0]. When writing register TMR4, it will write data from TM4RH[7:6] and Timer4 reload register to Timer4[9:0] current content.

3.4.18 T4CR1 (Timer4 Control Register1)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
T4CR1	S	0x16	PWM4OEN	PWM4OAL	-	-	TM4_HRC	T4OS	T4RL	T4EN
R/W Property			R/W	R/W	-	-	R/W	R/W	R/W	R/W
Initial Value			0	0	X	X	0	0	0	0

This register is used to configure Timer3 functionality.

T4EN: Enable/disable Timer4.

T4EN=1, enable Timer4.

T4EN=0, disable Timer4.

T4RL: Configure Timer4 down-count mechanism while Non-Stop mode is selected (T4OS=0).

T4RL=1, initial value is reloaded from reload register TMR4[9:0].

T4RL=0, continuous down-count from 0x3FF when underflow is occurred.

T4OS: Configure Timer4 operating mode while underflow is reached.

T4OS=1, One-Shot mode. Timer4 will count once from the initial value to 0x00.

T4OS=0, Non-Stop mode. Timer4 will keep down-count after underflow.

T4OS	T4RL	Timer4 Down-Count Functionality
0	0	Timer4 will count from reload value down to 0x00. When underflow is reached, 0x3FF is reloaded and continues down-count.
0	1	Timer4 will count from reload value down to 0x00. When underflow is reached, reload value is reloaded and continues to down-count.
1	x	Timer4 will count from initial value down to 0x00. When underflow is reached, Timer4 will stop down-count.

Table 16 Timer4 Functionality

TM4_HRC: Timer4 clock source selection.

TM4_HRC =1, PWM4 & Timer 4 clock source is High Oscillator Clock.

TM4_HRC =0, PWM4 & Timer 4 clock source selection depends on T4CS register bit.

PWM4OAL: Define PWM4 output active state.

PWM3OAL=1, PWM4 output is active low.

PWM3OAL=0, PWM4 output is active high.

PWM4OEN: Enable/disable PWM4 output.

PWM4OEN=1, PWM4 output will be present on PB3.

PWM4OEN=0, PB3 is GPIO.

3.4.19 T4CR2 (Timer4 Control Register2)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
T4CR2	S	0x17	-	-	T4CS	T4CE	/PS4EN	PS4SEL[2:0]		

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
R/W Property			-	-	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value			X	X	1	1	1	1	1	1

This register is used to configure Timer4 functionality.

PS4SEL[2:0]: Prescaler4 dividing rate selection.

PS4SEL[2:0]	Dividing Rate
000	1:2
001	1:4
010	1:8
011	1:16
100	1:32
101	1:64
110	1:128
111	1:256

Table 17 Prescaler4 Dividing Rate

Note: Always set PS4SEL[2:0] at /PS4EN=1, or interrupt may be falsely triggered.

/PS4EN: Disable/enable Prescaler4.

/PS4EN=1, disable Prescaler4.

/PS4EN=0, enable Prescaler4.

Note: When High Oscillator are selected as clock source must set /PS4EN=1(disable Prescaler4)

T4CE: Timer4 external clock edge selection.

T4CE=1, Timer4 will decrease one while high-to-low transition occurs on pin EX_CKI1.

T4CE=0, Timer4 will decrease one while low-to-high transition occurs on pin EX_CKI1.

T4CS: Timer4 clock source selection.

T4CS=1, External clock on pin EX_CKI1 is selected.

T4CS=0, Instruction clock or High Oscillator are selected.

3.4.20 PWM4DUTY (PWM4 Duty Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PWM4DUTY	S	0x18	PWM4DUTY[7:0]							
R/W Property			W							
Initial Value			XXXXXXXX							

The reload value of 10-bit Timer4 stored on registers TM4RH[7:6] and TMR4[7:0] is used to define the PWM4 frame rate, and registers TM4RH[3:2] and PWM4DUTY[7:0] is used to define the duty cycle of PWM4.

3.4.21 PS4CV (Prescaler4 Counter Value Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PS4CV	S	0x19	PS4CV[7:0]							
R/W Property			R							
Initial Value			1	1	1	1	1	1	1	1

While reading PS4CV, it will get current value of Prescaler4 counter.

3.4.22 TMR5 (Timer5 Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
TMR5	S	0x1A	TMR5[7:0]							
R/W Property			R/W							
Initial Value			XXXXXXXX							

When reading register TMR5, it will obtain current value of 10-bit down-count Timer5 at TMR5[7:0]. When writing register TMR5, it will write data from TM5RH[5:4] and Timer5 reload register to Timer5[9:0] current content.

3.4.23 T5CR1 (Timer5 Control Register1)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
T5CR1	S	0x1B	PWM5OEN	PWM5OAL	-	-	TM5_HRC	T5OS	T5RL	T5EN
R/W Property			R/W	R/W	-	-	R/W	R/W	R/W	R/W
Initial Value			0	0	X	X	0	0	0	0

This register is used to configure Timer3 functionality.

T5EN: Enable/disable Timer5.

T5EN=1, enable Timer5.

T5EN=0, disable Timer5.

T5RL: Configure Timer5 down-count mechanism while Non-Stop mode is selected (T5OS=0).

T5RL=1, initial value is reloaded from reload register TMR5[9:0].

T5RL=0, continuous down-count from 0x3FF when underflow is occurred.

T5OS: Configure Timer5 operating mode while underflow is reached.

T5OS=1, One-Shot mode. Timer5 will count once from the initial value to 0x00.

T5OS=0, Non-Stop mode. Timer5 will keep down-count after underflow.

T5OS	T5RL	Timer5 Down-Count Functionality
0	0	Timer5 will count from reload value down to 0x00. When underflow is reached, 0x3FF is reloaded and continues down-count.
0	1	Timer5 will count from reload value down to 0x00. When underflow is reached, reload value is reloaded and continues to down-count.

T5OS	T5RL	Timer5 Down-Count Functionality
1	x	Timer5 will count from initial value down to 0x00. When underflow is reached, Timer5 will stop down-count.

Table 16 Timer4 Functionality

TM5_HRC: Timer5 clock source selection.

TM5_HRC =1, PWM5 & Timer 5 clock source is High Oscillator Clock.

TM5_HRC =0, PWM5 & Timer 5 clock source selection depends on T5CS register bit.

PWM5OAL: Define PWM5 output active state.

PWM5OAL=1, PWM5 output is active low.

PWM5OAL=0, PWM5 output is active high.

PWM5OEN: Enable/disable PWM5 output.

PWM5OEN=1, PWM5 output will be present on PB2.

PWM5OEN=0, PB2 is GPIO.

3.4.24 T5CR2 (Timer5 Control Register2)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
T5CR2	S	0x1C	-	-	T5CS	T5CE	/PS5EN	PS5SEL[2:0]		
R/W Property			-	-	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value			X	X	1	1	1	1	1	1

This register is used to configure Timer5 functionality.

PS5SEL[2:0]: Prescaler5 dividing rate selection.

PS5SEL[2:0]	Dividing Rate
000	1:2
001	1:4
010	1:8
011	1:16
100	1:32
101	1:64
110	1:128
111	1:256

Table 17 Prescaler5 Dividing Rate

Note: Always set PS5SEL[2:0] at /PS5EN=1, or interrupt may be falsely triggered.

/PS5EN: Disable/enable Prescaler5.

/PS5EN=1, disable Prescaler5.

/PS5EN=0, enable Prescaler5.

Note: When High Oscillator are selected as clock source must set /PS5EN=1(disable Prescaler5)

T5CE: Timer5 external clock edge selection.

T5CE=1, Timer5 will decrease one while high-to-low transition occurs on pin EX_CKI1.

T5CE=0, Timer5 will decrease one while low-to-high transition occurs on pin EX_CKI1.

T5CS: Timer5 clock source selection.

T5CS=1, External clock on pin EX_CKI1 is selected.

T5CS=0, Instruction clock or High Oscillator are selected.

3.4.25 PWM5DUTY (PWM5 Duty Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PWM5DUTY	S	0x1D	PWM5DUTY[7:0]							
R/W Property			W							
Initial Value			XXXXXXXX							

The reload value of 10-bit Timer5 stored on registers TM5RH[5:4] and TMR5[7:0] is used to define the PWM5 frame rate, and registers TM5RH[1:0] and PWM5DUTY[7:0] is used to define the duty cycle of PWM5.

3.4.26 PS5CV (Prescaler5 Counter Value Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PS5CV	S	0x1E	PS5CV[7:0]							
R/W Property			R							
Initial Value			1	1	1	1	1	1	1	1

While reading PS5CV, it will get current value of Prescaler5 counter.

3.4.27 TM5RH (Timer High Byte Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
TM5RH	S	0x1F	-	-	TMR59	TMR58	-	-	PWM5DUTY9	PWM5DUTY8
R/W Property			-	-	R/W	R/W	-	-	R/W	R/W
Initial Value			X	X	X	X	X	X	X	X

TMR59~8: Timer5 MSB 2 bits. Write these 2 bits will overwrite the 10-bit Timer5 load value of bit 9 and 8. Read these 2 bits will get the Timer5 bit9-8 current value.

PWM5DUTY9~8: PWM5 duty data MSB 2 bits.

3.5 T-page Special Function Register

3.5.1 INTE3 (Interrupt Enable 3 Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
INTE3	T	0x7	-	EEIE	T5IE/CCPIE	-	-	-	-	-
R/W Property			-	R/W	R/W	-	-	-	-	-
Initial Value			X	0	0	X	X	X	X	X

T5IE/CCPIE: When CCP capture or compare mode is enabled, this interrupt is used as CCP interrupt enable bits, otherwise it is T5 underflow interrupt enable bit.
T5IE/CCPIE=1, enable interrupt.
T5IE/CCPIE=0, disable interrupt.

EEIE: End of EEPROM Writing interrupt enable bit.
EEIE=1, enable End of EEPROM Writing interrupt.
EEIE=0, disable End of EEPROM Writing interrupt.

3.5.2 INTF3 (Interrupt Flag 3 Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
INTF3	T	0x8	-	EEIF	T5IF/CCPIF	-	-	-	-	-
R/W Property			-	R/W	R/W	-	-	-	-	-
Initial Value(note*)			X	0	0	X	X	X	X	X

T5IF/CCPIF: When CCP capture or compare mode is enabled, this interrupt is used as CCP interrupt flag bits, otherwise it is T5 underflow interrupt flag bit.
T5IF/CCPIF=1, interrupt is occurred.
T5IF/CCPIF must be clear by firmware.

EEIF: Enf of EEPROM writing interrupt flag bit.
EEIF=1, Enf of EEPROM writing interrupt is occurred.
EEIF must be clear by firmware.

3.5.3 CCPCON (CCP Control Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CCPCON	T	0x1E	PWM5 M1	PWM5 M0	FBCH1	FBCH0	CCPM3	CCPM2	CCPM1	CCPM0
R/W Property			R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value(note*)			0	0	0	0	0	0	0	0

CCPM[3:2]=00/01/10: Capture or compare mode. PB2 is capture input or compare output.

CCPM[3:2]=11:

PWM5M1~0 = 00 → PWM single output.

PWM5M1~0 = 01 → PWM Full-Bridge output forward.

PWM5M1~0 = 10 → PWM Half-bridge output.

PWM5M1~0 = 11 → PWM Full-Bridge output reverse.

T5IF/CCPIF=1, interrupt is occurred.

Note: T5IF/CCPIF must be clear by firmware.

CCPM[3:0]:

0000 = CCP off.

0010 = Compare mode, toggle output on match.

0100 = Capture mode, capture at every falling edge.

0101 = Capture mode, capture at every rising edge.

0110 = Capture mode, capture at every 4th rising edge.

0111 = Capture mode, capture at every 16th rising edge.

1000 = Compare mode, set output and interrupt on match.

1001 = Compare mode, clear output and interrupt on match.

1010 = Compare mode, only interrupt on match.

1011 = Compare mode, trigger ADC and interrupt on match.

1100 = PWM mode, P1A/P1C active high, P1D/P1B active high.

1101 = PWM mode, P1A/P1C active high, P1D/P1B active low.

1110 = PWM mode, P1A/P1C active low, P1D/P1B active high.

1111 = PWM mode, P1A/P1C active low, P1D/P1B active low.

Note: P1A is PB2, P1B is PA5, P1C is PA2, P1D is PA4.

FBCH1~0: Full band change direction gap

00 = 1 CPU cycle.

01 = 4 CPU cycle.

1x = 16 CPU cycle.

Note: Compare/Capture mode step:

a. Set timer5 operating mode : set T1OS/T1RL, set T5EN = PWM5OEN = 0.

b. Set timer4 clock source, timer4 prescaler.

c. Set timer4 / timer5 reload/initial value.

d. Set pwm4duty / pwm5duty.

e. Enable compare / capture mode.

f. Disable compare / capture mode before update pwm / timer data.

3.5.4 PWMDB (Dead band control Register)

Name	SFR Type	Addr.	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PWMDB	T	0x1F	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
R/W Property			R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value(note*)			0	0	0	0	0	0	0	0

DB[7:0]: Define the dead band width of the CCP PWM mode. The dead band unit is in CPU cycle.

td= F_{INST} *(DB[7:0])

3.6 I/O Port

AT8BE62D provides 14 I/O pins which are PA[7:0] and PB[5:0]. User can read/write these I/O pins through registers PORTA and PORTB respectively. Each I/O pin has a corresponding register bit to define it is input pin or output pin. Register IOSTA[7:0] define the input/output direction of PA[7:0]. Register IOSTB[5:0] define the input/output direction of PB[5:0].

When an I/O pin is configured as input pin, it may have Pull-High resistor or Pull-Low resistor which is enabled or disabled through registers. Register APHCON[7:6], PCON[4] and APHCON[4:0] are used to enable or disable Pull-High resistor of PA[7:0]. Register APHCON[5], PCON[6] and ABPLCON[3:0] are used to enable or disable Pull-Low resistor of PA[5:0]. Register BPHCON[5:0] are used to enable or disable Pull-High resistor of PB[5:0]. Register ABPLCON[7:4] are used to enable or disable Pull-Low resistor of PB[3:0].

When an PortB I/O pin is configured as output pin, there is a corresponding and individual register to be selected as Open-Drain output pin. Register BODCON[5:0] determine PB[5:0] is Open-Drain or not. When an

The summary of Pad I/O feature is listed in the table below.

Feature		PA[5:0]	PA[7:6]	PB[3:0]	PB[5:4]
Input	Pull-High Resistor	V	V	V	V
	Pull-Low Resistor	V	X	V	X
Output	Open-Drain	X	X	V	V

Table 19 Summary of Pad I/O Feature

The level change on each I/O pin of PA and PB may generate interrupt request. Register AWUCON[7:0] and BWUCON[5:0] will select which I/O pin of PA and PB may generate this interrupt. As long as any pin of PA and PB is selected by corresponding bit of AWUCON and BWUCON, the register bit PABIF (INTF[1]) will set to 1 if there is a level change occurred on any selected pin. An interrupt request will occur and interrupt service routine will be executed if register bit PABIE (INTE[1]) and GIE (PCON1[7]) are both set to 1.

There are three external interrupts provided by AT8BE62D. When register bit EIS0 (INTEDG[4]) is set to 1, PB0 or PB5 (according to INT0 input configuration word) is used as input pin for external interrupt 0. When register bit EIS1 (INTEDG[5]) is set to 1, PB1 or PA3 (according to INT1 input configuration word) is used as input pin for external interrupt 1. When register bit EIS2 (INTEDG[6]) is set to 1, PA5 is used as input pin for external interrupt 2 when INT2 input configuration word is properly selected.

Note: When PB0 / PB1 / PA3 / PB5 / PA5 are both set as level change operation and external interrupt, the external interrupt will have higher priority, and the PB0 / PB1/ PA3 / PB5 / PA5 level change operation will be disabled.

AT8BE62D provides IR carrier generation output. As the table 20 shows, the IR operation is depended both on register IREN(IRCR[0]) and configuration word IR_PAD : when IREN=1 and IR_PAD=1, IR carrier output will be presented on PB1 pad. When IREN=1 and IR_PAD=0, IR carrier output will be presented on PA3.

PA5 can be used as external reset input which is enabled or disabled by configuration word PA.5. When an active-low signal is applied to PA5, it will cause AT8BE62D to enter reset process.

When external crystal (E_HXT, E_XT or E_LXT) is adopted for high oscillation or low oscillation according to setting of configuration words, PA6 will be used as crystal input pin (Xin) and PA7 will be used as crystal output pin (Xout).

When I_HRC or I_LRC mode is selected as system oscillation and E_HXT, E_XT or E_LXT is not adopted, instruction clock is observable on PA7 if configuration word Instruction Clock is enabled.

Moreover, PA4 can be timer 0 external clock source EX_CKI0 if T0MD T0CS=1 and LCK_TM0=0. PA4 can be timer 1 external clock source EX_CKI0 if T1CS=1. PA1 or PA2 (according to External Clock 1 configuration word) can be Timer4/Timer5 external clock source EX_CKI1 if T4CS/T5CS=1.

PB1 or PB4 (according to PWM1 output configuration word) can be PWM1 output If P1CR1[7] PWM1OEN=1. PB4 can be T1OUT output if T1CR1[5] TM1OE=1.The output priority of PB4 is T1OUT > PWM1.

PA4, PB5 (according to PWM2 output configuration word) can be PWM2 output If P2CR1[7] PWM2OEN=1. PB2 can be Buzzer2 output if BZ2CR[7] BZ2EN=1.

PA2 or PA7 (according to PWM3 output configuration word) can be PWM3 output If T3CR1[7] PWM3OEN=1. PA2 can be Buzzer3 output if BZ3CR[7] BZ3EN=1.

PB3 can be PWM4 output If T4CR1[7] PWM4OEN=1. PB3 can be Buzzer1 output if BZ1CR[7] BZ1EN=1.

PB2 can be PWM5 output If T5CR1[7] PWM5OEN=1.

When configured as output, the sink current of each pin can be small(4mA for VDD =3V), normal (16mA for VDD =3V) or large (23mA for VDD =3V) according to configuration words. When configured as output, the drive current of each pin can be small (1.0mA for VDD =3V) or large (10mA for VDD =3V) according to configuration words. Check the following table for drive/sink current mode setting:

Configuration Word	Small Sink	Normal Sink	Large Sink
PXcurrent	X	0	1
PXcsc	X	0	0
Small Sink	0	1	1

Configuration Word	Small drive	Large Drive
Small Drive & Sink	0	1

Table 21 Drive / Sink current mode setting (X=A, B)

3.6.1 Block Diagram of IO Pins

IO_SEL: set pad attribute as input or output

WRITE_EN: write data to pad.

READ_EN: read pad.

PULLUP_ENB: enable Pull-High.

PULLDOWN_EN: enable Pull-Low.

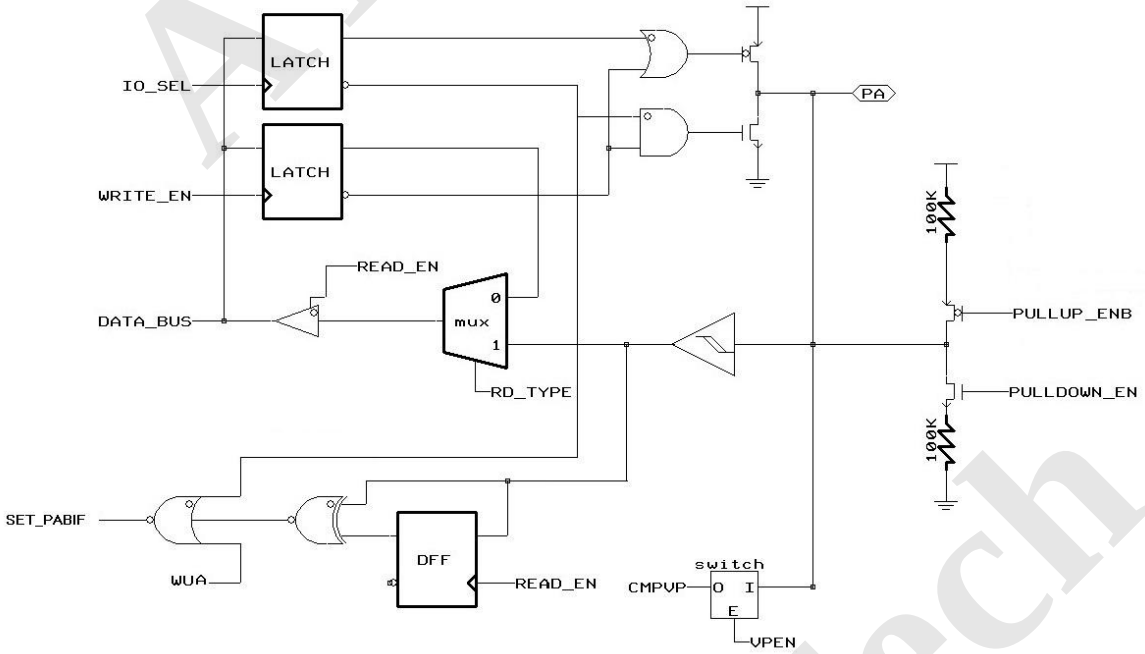
VPEN: enable pad to comparator input.

CMPVP: to comparator input.

RD_TYPE: select read pin or read latch.

WUA: port A wake-up enable.

SET_PBIF: port B wake-up flag.



.Figure 5 Block Diagram of PA0, PA1, PA2

IO_SEL: set pad attribute as input or output.

WRITE_EN: write data to pad.

READ_EN: read pad.

PULLUP_ENB: enable 100KΩ Pull-High.

PULLDOWN_EN: enable Pull-Low.

VPEN: enable pad to comparator input.

CMPVP: to comparator input.

EIS: external interrupt function enable.

EX_INT: external interrupt signal.

RD_TYPE: select read pin or read latch.

WUA: port A wake-up enable.

SET_PBIF: port B wake-up flag.

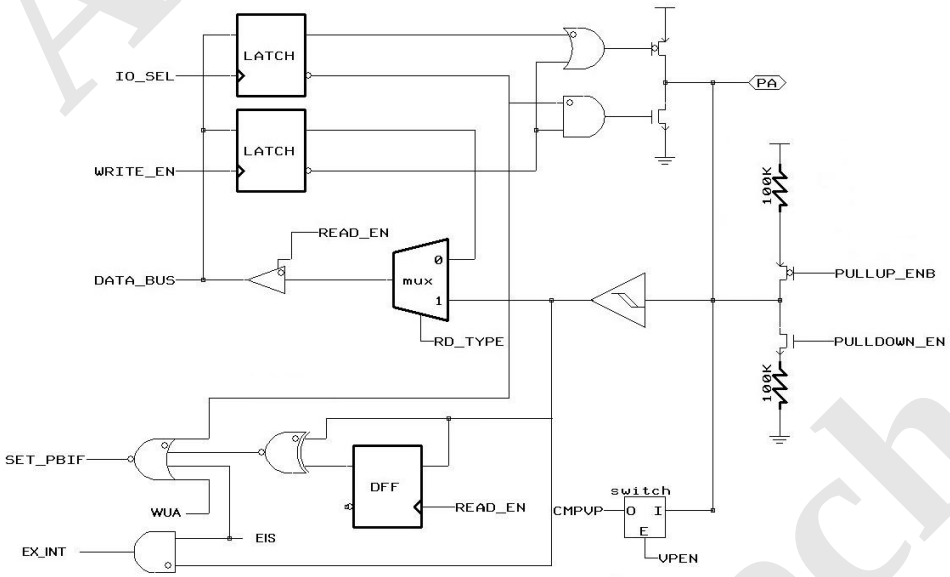


Figure 8 Block Diagram of PA3

IO_SEL: set pad attribute as input or output.

WRITE_EN: write data to pad.

READ_EN: read pad.

PULLUP_ENB: enable Pull-High.

RD_TYPE: select read pin or read latch.

WUA: port A wake-up enable.

SET_PBIF: port B wake-up flag.

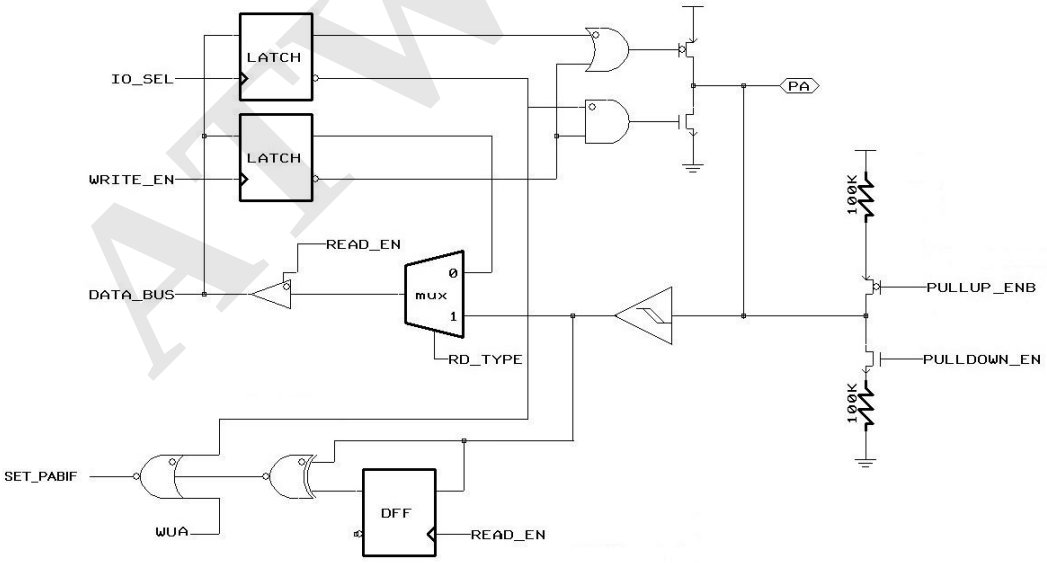


Figure 9 Block Diagram of PA4

RSTPAD_EN: enable PA5 as reset pin.

RSTB_IN: reset signal input.

IO_SEL: set pad attribute as input or output.

WRITE_EN: write data to pad.

READ_EN: read pad.

PULLUP_ENB: enable Pull-High.

PULLDOWN_EN: enable Pull-Low.

EIS: external interrupt function enable.

INTEDG[7]: external interrupt edge select.

EX_INT: external interrupt signal.

RD_TYPE: select read pin or read latch.

WUA: port A wake-up enable.

SET_PBIF: port B wake-up flag.

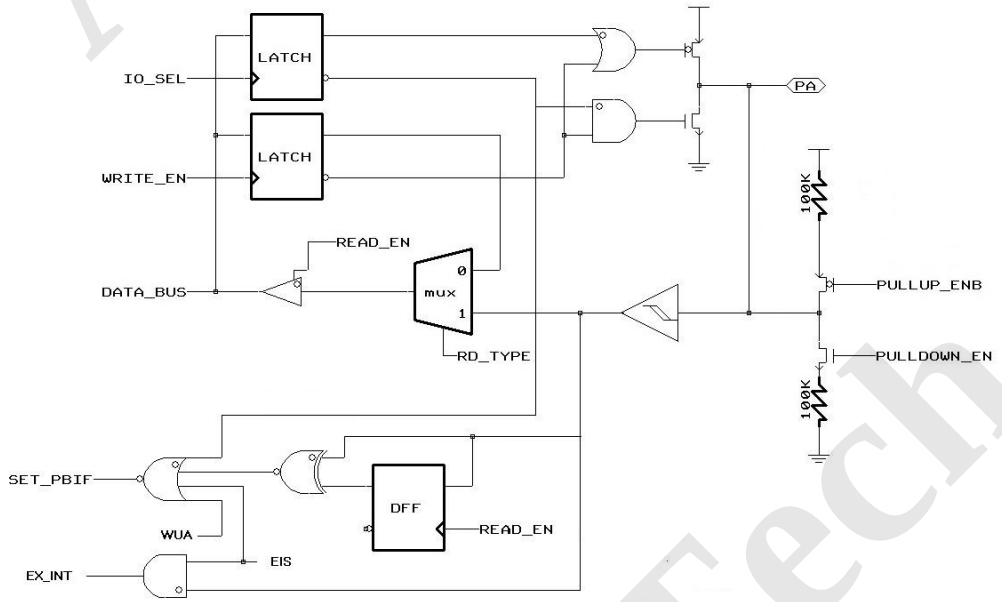


Figure 10 Block Diagram of PA5

XTL_EN: enable crystal oscillation mode.

IO_SEL: set pad attribute as input or output.

WRITE_EN: write data to pad.

READ_EN: read pad.

PULLUP_ENB: enable Pull-High.

RD_TYPE: select read pin or read latch.

WUA: port A wake-up enable.

SET_PBIF: port B wake-up flag.

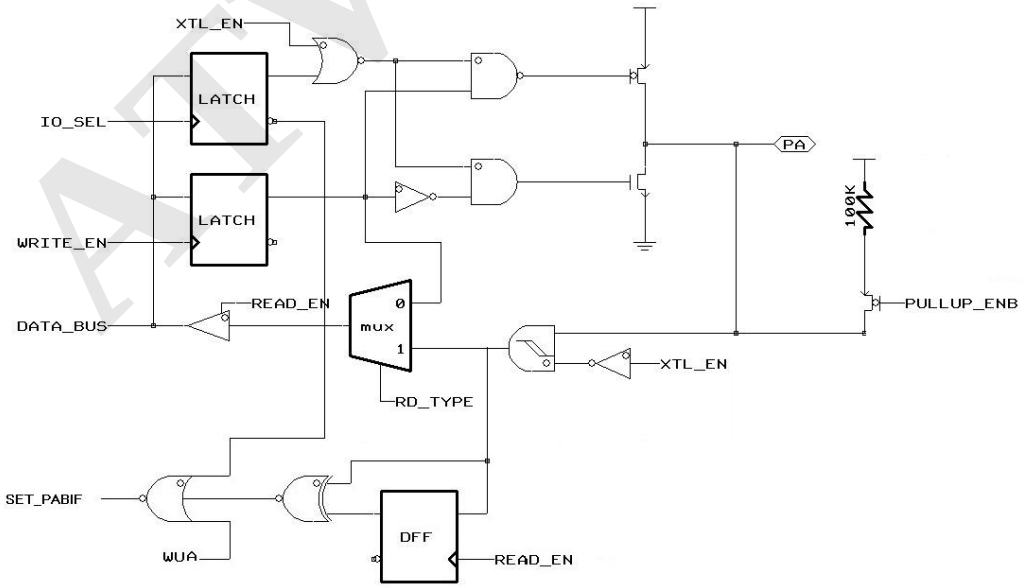


Figure 11 Block Diagram of PA6, PA7

IO_SEL: set pad attribute as input or output.

WRITE_EN: write data to pad.

READ_EN: read pad.

OD_EN: enable Open-Drain.

PULLUP_ENB: enable Pull-High.

PULLDOWN_EN: enable Pull-Low.

RD_TYPE: select read pin or read latch.

WUB: port B wake-up enable.

SET_PBIF: port B wake-up flag.

EIS: external interrupt function enable.

EX_INT: external interrupt signal.

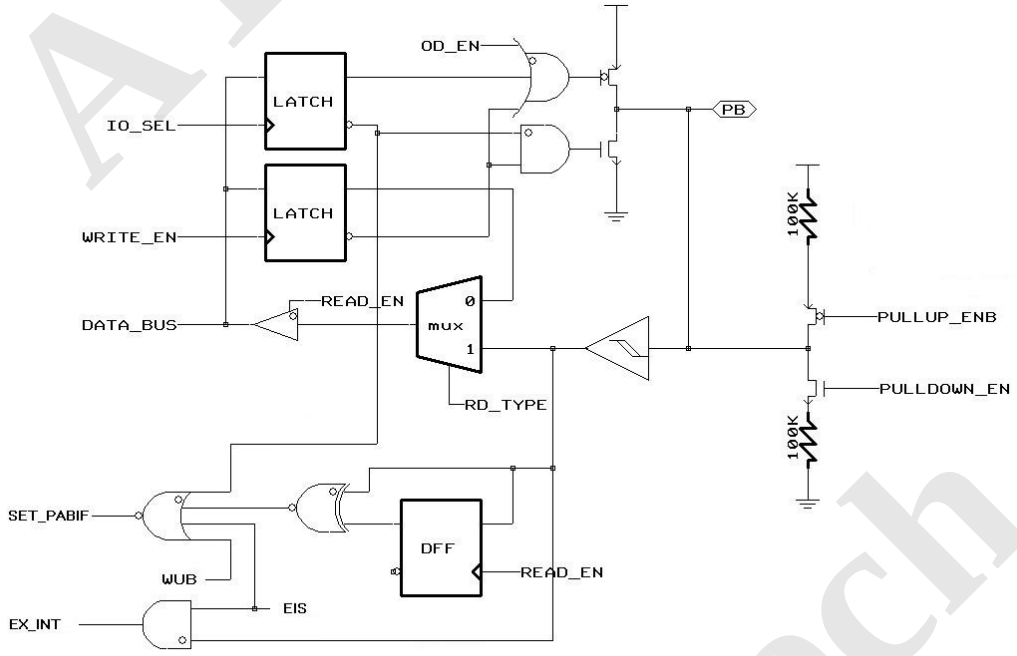


Figure 12 Block Diagram of PB0, PB1

IO_SEL: set pad attribute as input or output.

WRITE_EN: write data to pad.

READ_EN: read pad.

OD_EN: enable open-Drain.

PULLUP_ENB: enable Pull-High.

PULLDOWN_EN: enable Pull-Low.

RD_TYPE: select read pin or read latch.

WUB: port B wake-up enable.

SET_PBIF: port B wake-up flag.

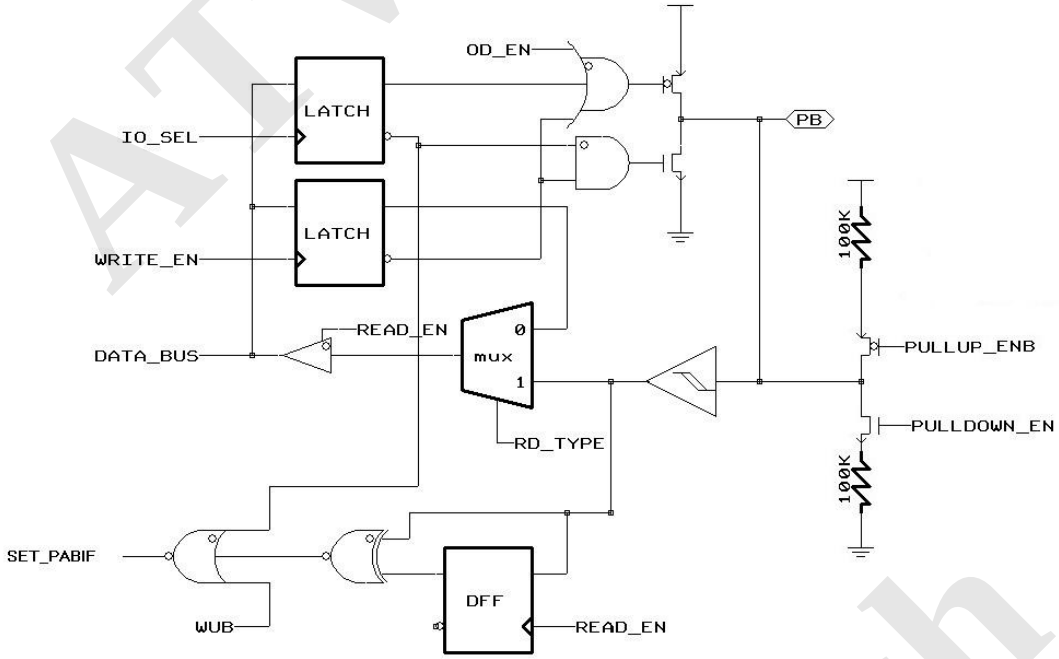


Figure 14 Block Diagram of PB2 & PB3

IO_SEL: set pad attribute as input or output.

WRITE_EN: write data to pad.

READ_EN: read pad.

OD_EN: enable open-Drain.

PULLUP100K_ENB: enable Pull-High.

RD_TYPE: select read pin or read latch.

WUB: port B wake-up enable.

SET_PBIF: port B wake-up flag.

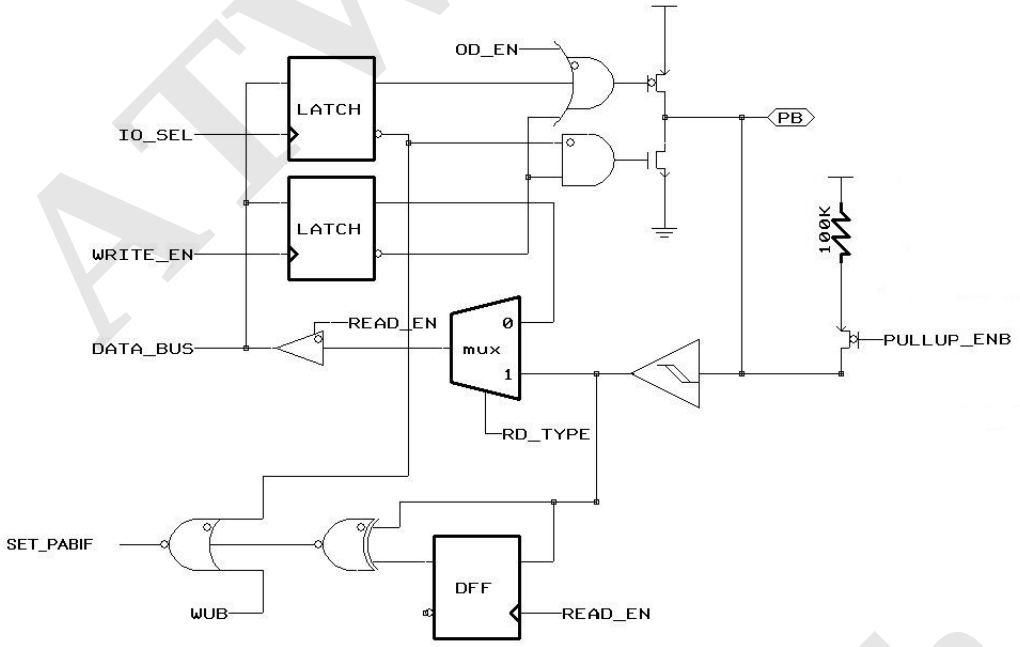


Figure 15 Block Diagram of PB4

IO_SEL: set pad attribute as input or output.

WRITE_EN: write data to pad.

READ_EN: read pad.

OD_EN: enable open-Drain.

PULLUP_ENB: enable Pull-High.

RD_TYPE: select read pin or read latch.

WUB: port B wake-up enable.

SET_PBIF: port B wake-up flag.

EIS: external interrupt function enable.

EX_INT: external interrupt signal.

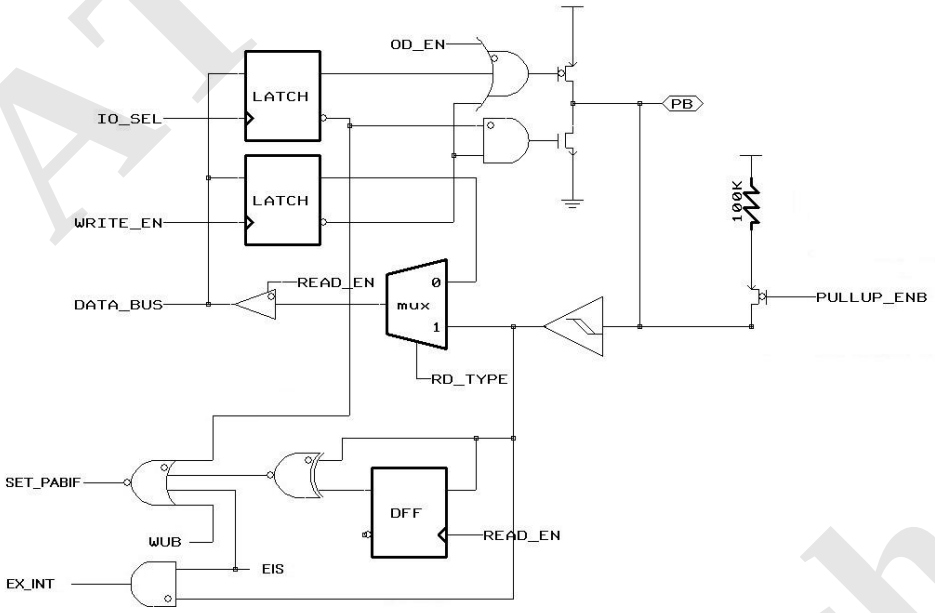


Figure 16 Block Diagram of PB5

3.7 Timer0

Timer0 is an 8-bit up-count timer and its operation is enabled by register bit T0EN (PCON1[0]). Writing to Timer0 will set its initial value. Reading from Timer0 will show its current count value.

The clock source to Timer0 can be from instruction clock, external pin EX_CKI0 or low speed clock Low Oscillator Frequency according to register bit T0CS and LCK_TM0 (T0MD[5] and T0MD[7]). When T0CS is 0, instruction clock is selected as Timer0 clock source. When T0CS is 1 and LCK_TM0 is 0, EX_CKI0 is selected as Timer0 clock source. When T0CS is 1 and LCK_TM0 is 1 (and Timer0 source must set to 1), Low Oscillator Frequency (I_LRC or E_LXT, depends on configuration word) output is selected. Summarized table is shown below. (Also check Figure 18)

Timer0 clock source	T0CS	LCKTM0	Timer0 source	Low Oscillator Frequency
Instruction clock	0	X	X	X
EX_CKI0	1	0	X	X
		X	0	
E_LXT	1	1	1	1
I_LRC	1	1	1	0

Table 22 Summary of Timer0 clock source control

Moreover the active edge of EX_CKI0 or Low Oscillator Frequency to increase Timer0 can be selected by register bit T0CE (T0MD[4]). When T0CE is 1, high-to-low transition on EX_CKI0 or Low Oscillator Frequency will increase Timer0. When T0CE is 0, low-to-high transition on EX_CKI0 or Low Oscillator Frequency will increase Timer0. When using Low Oscillator Frequency as Timer0 clock source, it is suggested to use prescaler0 (see below descriptions) and the ratio set to more than 4, or missing count may happen.

Before Timer0 clock source is supplied to Timer0, it can be divided by Prescaler0 if register bit PS0WDT (T0MD[3]) is clear to 0. When writing 0 to PS0WDT by instruction, Prescaler0 is assigned to Timer0 and Prescaler0 will be clear after this instruction is executed. The dividing rate of Prescaler0 is determined by register bits PS0SEL[2:0] which is from 1:2 to 1:256.

When Timer0 is overflow, the register bit T0IF (INTF[0]) will be set to 1 to indicate Timer0 overflow event is occurred. If register bit T0IE (INTE[0]) and GIE are both set to 1, interrupt request will occur and interrupt service routine will be executed. T0IF will not be clear until firmware writes 0 to T0IF.

The block diagram of Timer0 and WDT is shown in the figure below.

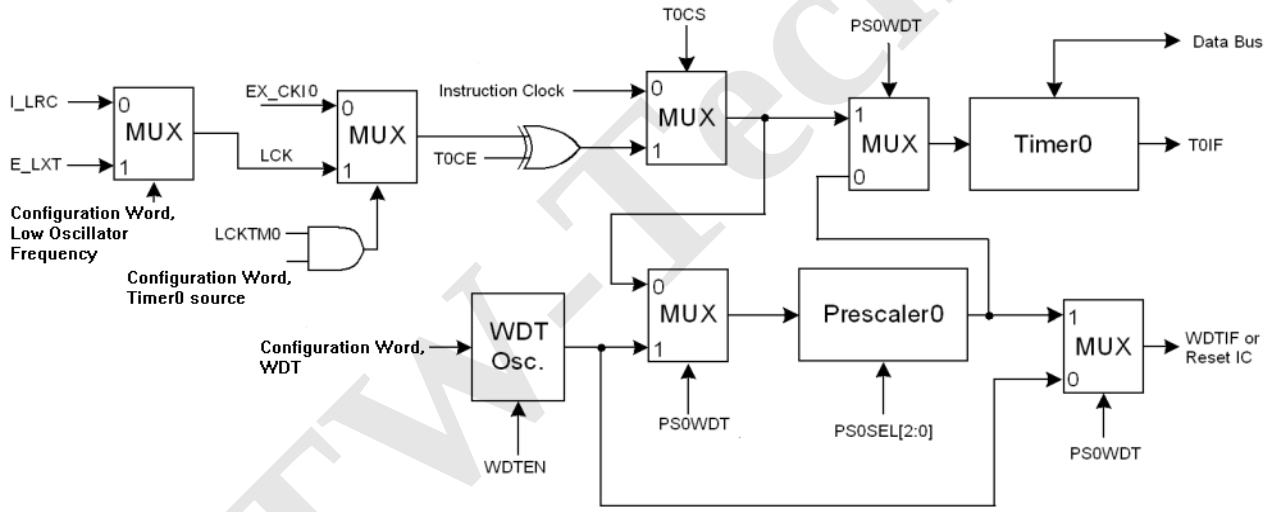
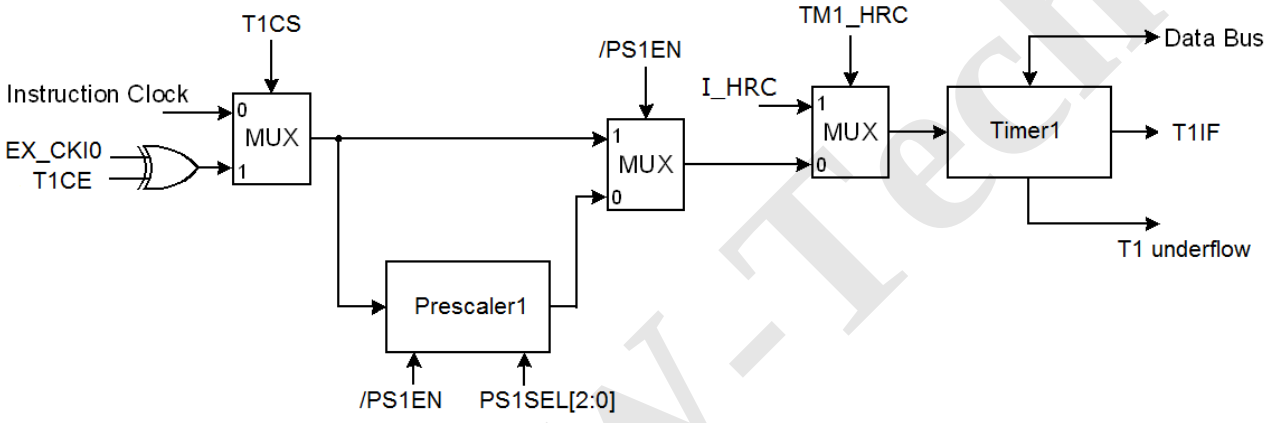


Figure 18 Block Diagram of Timer0 and WDT

3.8 Timer1 / PWM1 / Buzzer1

Timer1 is an 10-bit down-count timer with Prescaler1 whose dividing rate is programmable. The output of Timer1 can be used to generate PWM1 output and Buzzer1 output. Timer1 builds in auto-reload function and Timer1 reload register stores reload data with double buffers. When user write Timer1 reload register, write Timer1 MSB 2 bits(TMRH[5:4]) first and write TMR1 second, Timer1 reload register will be updated to Timer1 counter after Timer1 overflow occurs when T1EN=1. If T1EN=0, Timer1 reload register will be updated to Timer1 counter after write TMR1 immediately. A read to the Timer1 will show the content of the Timer1 current



count value. The block diagram of Timer1 is shown in the figure below.

Figure 19 Block Diagram of Timer1

The operation of Timer1 can be enabled or disabled by register bit T1EN (T1CR1[0]). After Timer1 is enabled, its clock source can be instruction clock, pin EX_CKI0 or I_HRC output, which is determined by register bit T1CS

(T1CR2[5]) and TM1_HRC(T1CR1[3]). When T1CS is 1 and TM1_HRC is 0, EX_CKI0 is selected as clock source. When T1CS is 0 and TM1_HRC is 0, instruction clock is selected as clock source. When TM1_HRC is 1, I_HRC output is selected as clock source. When EX_CKI0 is selected, the active edge to decrease Timer1 is determined by register bit T1CE (T1CR2[4]). When T1CE is 1, high-to-low transition on EX_CKI0 will decrease Timer1. When T1CE is 0, low-to-high transition on EX_CKI0 will decrease Timer1. The selected clock source can be divided further by Prescaler1 before it is applied to Timer1. Prescaler1 is enabled by writing 0 to register bit /PS1EN (T1CR2[3]) and the dividing rate is from 1:2 to 1:256 determined by register bits PS1SEL[2:0] (T1CR2[2:0]). Current value of Prescaler1 can be obtained by reading register PS1CV.

Timer1 provides two kinds of operating mode: one is One-Shot mode and the other is Non-Stop mode. When register bit T1OS (T1CR1[2]) is 1, One-Shot mode is selected. Timer1 will count down once from initial value stored on register TMR1[9:0] to 0x00, i.e. underflow is occurred. When register bit T1OS (T1CR1[2]) is 0, Non-Stop mode is selected. When underflow is occurred, there are two selections to start next down-count which is determined by register bit T1RL (T1CR1[1]). When T1RL is 1, the initial value stored on register TMR1[9:0] will be restored and start next down-count from this initial value. When T1RL is 0, Timer1 will start next down-count from 0x3FF.

When Timer1 is underflow, the register bit T1IF (INTF[3]) will be set to 1 to indicate Timer1 underflow event is occurred. If register bit T1IE (INTE[3]) and GIE are both set to 1, interrupt request will occur and interrupt service routine will be executed. T1IF will not be clear until firmware writes 0 to T1IF.

The timing chart of Timer1 is shown in the following figure.

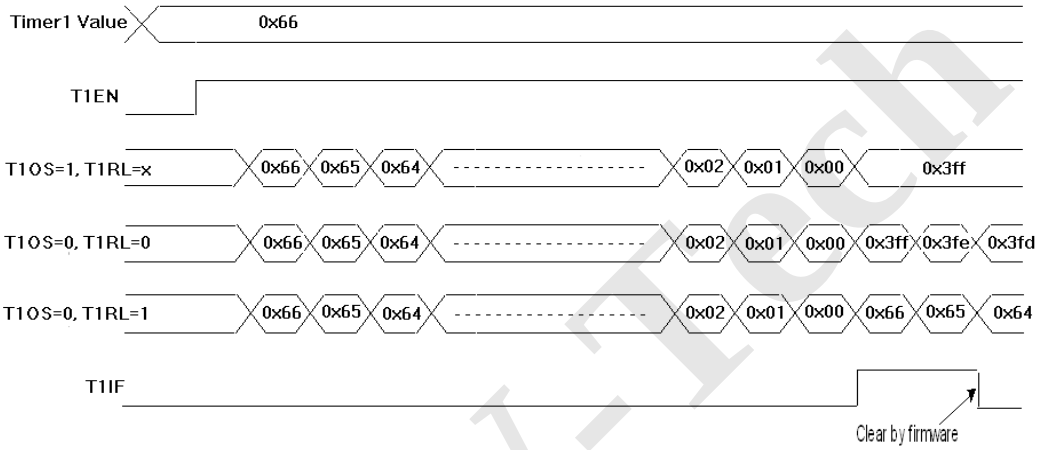


Figure 20 Timer1 Timing Chart

The PWM1 output can be available on I/O pin PB1 or PB4 according to PWM1 output configuration word. When register bit PWM1OEN (T1CR1[7]) is set to 1, the corresponded PWM1 pin will became output pin automatically. The active state of PWM1 output is determined by register bit PWM1OAL (T1CR1[6]). When PWM1OAL is 1, PWM1 output is active low. When PWM1OAL is 0, PWM1 output is active high. Moreover, the duty cycle and frame rate of PWM1 are both programmable. The duty cycle is determined by registers TMRH[1:0] and

PWM1DUTY[7:0]. When PWM1DUTY is 0, PWM1 output will be never active. When PWM1DUTY is 0x3FF, PWM1 output will be active for 1023 Timer1 input clocks. The frame rate is determined by TMRH[5:4] + TMR1[7:0] initial value. Therefore, PWM1DUTY value must be less than or equal to TMRH[5:4] + TMR1[7:0]. When user write PWM1DUTY, write PWM1DUTY[9:8] MSB 2 bits(TMRH[1:0]) first and write PWM1DUTY[7:0] second, PWM1 duty register will be updated after Timer1 overflow occurs. The block diagram of PWM1 is illustrated in the following figure.

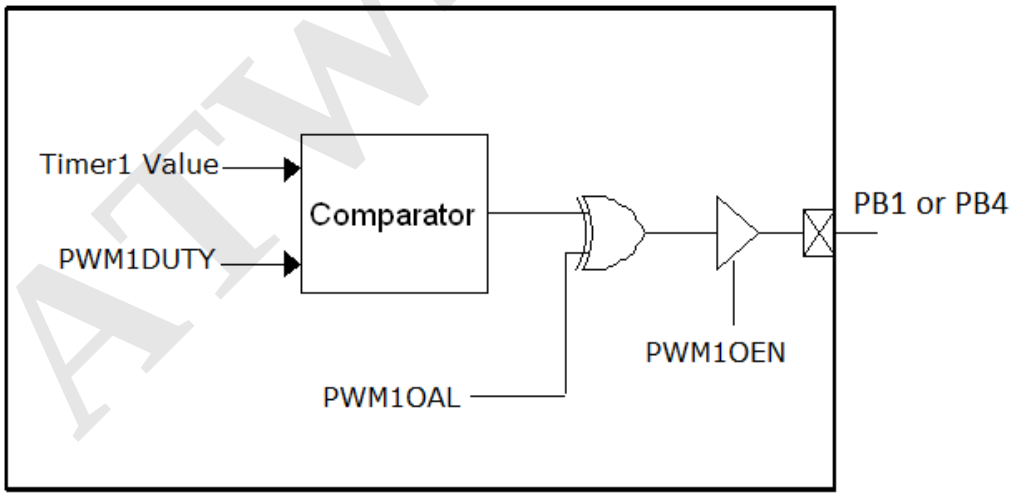


Figure 21 PWM1 Block Diagram

The Buzzer1 output (BZ1) can be available on I/O pin PB3 when register bit BZ1EN (BZ1CR1[7]) is set to 1. Moreover, PB3 will become output pin automatically. The frequency of BZ1 can be derived from Timer1 output or Prescaler1 output and dividing rate is determined by register bits BZ1FSEL[3:0] (BZ1CR[3:0]). When BZ1FSEL[3] is 0, Prescaler1 output is selected to generate BZ1 output. When BZ1FSEL[3] is 1, Timer1 output is selected to generate BZ1 output. The dividing rate can be from 1:2 to 1:256 in order to generate all kinds of frequency. The block diagram of Buzzer1 is illustrated in the following figure.

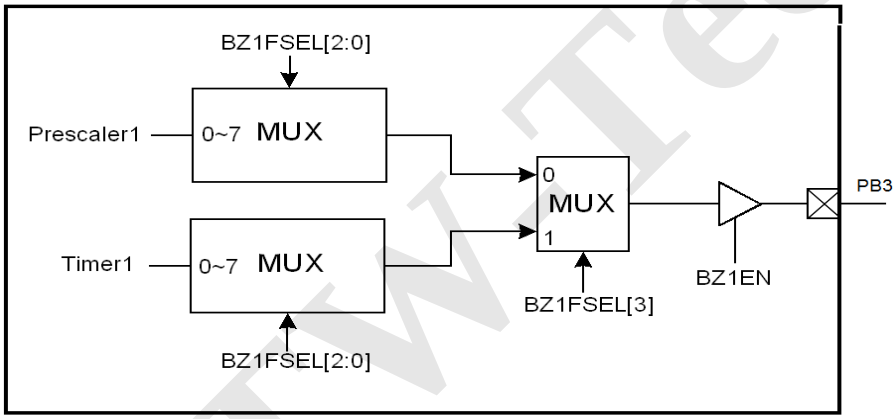


Figure 22 Buzzer1 Block Diagram

Note: When PWM1 and Buzzer1 are both enabled, PWM1 will have the higher priority for PB3 output.

3.9 PWM2

The PWM2 output can be available on I/O pin PA4 or PB5 according to PWM2 output configuration word. When register bit PWM2OEN (P2CR1[7]) is set to 1, the corresponded PWM2 pin will became output pin automatically. The active state of PWM2 output is determined by register bit PWM2OAL (P2CR1[6]). When PWM2OAL is 1, PWM2 output is active low. When PWM2OAL is 0, PWM2 output is active high. Moreover, the duty cycle and frame rate of PWM2 are both programmable. The duty cycle is determined by register TMRH[3:2], PWM2DUTY[7:0]. When PWM2DUTY is 0, PWM2 output will be never active. When PWM2DUTY is 0x3FF, PWM2 output will be active for 1023 Timer2 input clocks. The frame rate is determined by TMRH[5:4], TMR1[7:0] initial value. Therefore, PWM2DUTY value must be less than or equal to TMR1[9:0]. When user write PWM2DUTY, write PWM2DUTY[9:8] MSB 2 bits(TMRH[3:2]) first and write PWM2DUTY[7:0] second, PWM2 duty register will be updated after Timer1 overflow occurs. The block diagram of PWM2 is illustrated in the following figure.

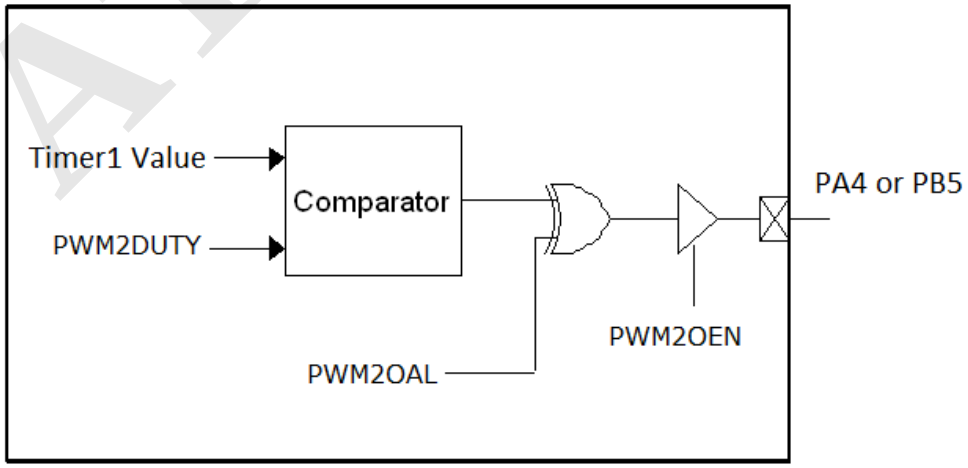


Figure 25 PWM2 Block Diagram

3.10 PWM3

The PWM3 output can be available on I/O pin PA2 or PA7 according to PWM3 output configuration word. When register bit PWM3OEN (P3CR1[7]) is set to 1, the corresponded PWM3 pin will became output pin automatically. The active state of PWM3 output is determined by register bit PWM3OAL (P3CR1[6]). When PWM3OAL is 1, PWM3 output is active low. When PWM3OAL is 0, PWM3 output is active high. Moreover, the duty cycle and frame rate of PWM3 are both programmable. The duty cycle is determined by register TM4RH[1:0] , PWM3DUTY[7:0]. When PWM3DUTY is 0, PWM3 output will be never active. When PWM3DUTY is 0x3FF, PWM3 output will be active for 1023 Timer3 input clocks. The frame rate is determined by TMRH[5:4], TMR1[7:0] initial value. Therefore, PWM3DUTY value must be less than or equal to TMR1[9:0]. When user write PWM3DUTY, write PWM3DUTY[9:8] MSB 2 bits(TM4RH[1:0]) first and write PWM3DUTY[7:0] second, PWM3 duty register will be updated after Timer1 overflow occurs. The block diagram of PWM3 is illustrated in the following figure.

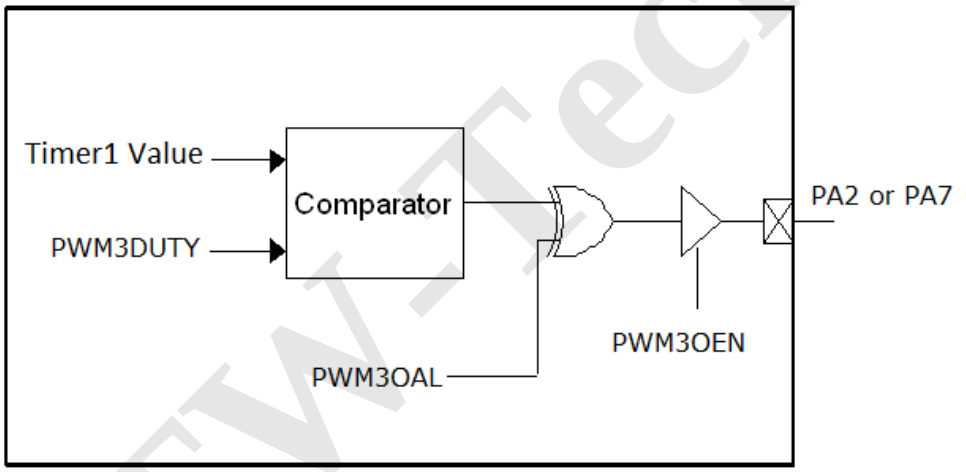
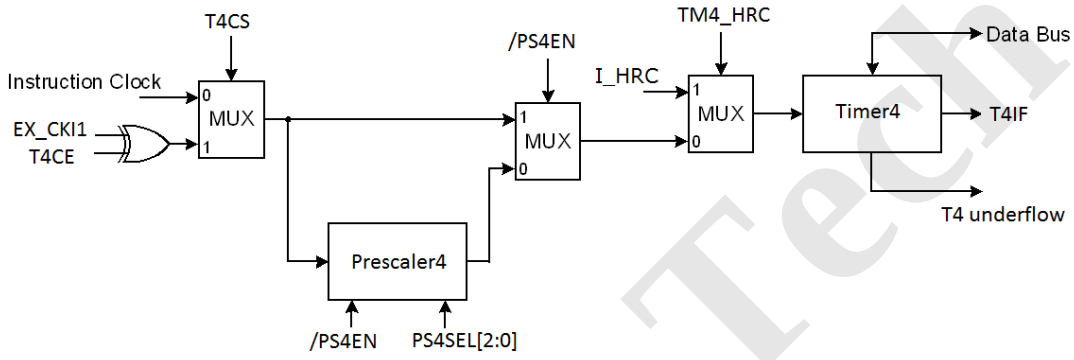


Figure 29 PWM3 Block Diagram

3.11 Timer4 / PWM4

Timer4 is an 10-bit down-count timer with Prescaler4 whose dividing rate is programmable. The output of Timer4 can be used to generate PWM4 output. Timer4 builds in auto-reload function and Timer4 reload register stores reload data with double buffers. When user write Timer4 reload register, write Timer4 MSB 2 bits(TM4RH[7:6]) first and write TMR4 second, Timer4 reload register will be updated to Timer4 counter after Timer4 overflow occurs when T4EN=1. If T4EN=0, Timer4 reload register will be updated to Timer4 counter after write TMR4 immediately. A read to the Timer4 will show the content of the Timer4 current count value. The



block diagram of Timer4 is shown in the figure below.

Figure 19 Block Diagram of Timer4

The operation of Timer4 can be enabled or disabled by register bit T4EN (T4CR1[0]). After Timer4 is enabled, its clock source can be instruction clock, pin EX_CKI1 or I_HRC output, which is determined by register bit T4CS (T4CR2[5]) and TM4_HRC(T4CR1[3]). When T4CS is 1 and TM4_HRC is 0, EX_CKI1 is selected as clock source. When T4CS is 0 and TM4_HRC is 0, instruction clock is selected as clock source. When TM4_HRC is 1, I_HRC output is selected as clock source. When EX_CKI1 is selected, the active edge to decrease Timer4 is

determined by register bit T4CE (T4CR2[4]). When T4CE is 1, high-to-low transition on EX_CKI1 will decrease Timer1. When T4CE is 0, low-to-high transition on EX_CKI1 will decrease Timer4. The selected clock source can be divided further by Prescaler4 before it is applied to Timer4. Prescaler4 is enabled by writing 0 to register bit /PS4EN (T4CR2[3]) and the dividing rate is from 1:2 to 1:256 determined by register bits PS4SEL[2:0] (T4CR2[2:0]). Current value of Prescaler4 can be obtained by reading register PS4CV.

Timer4 provides two kinds of operating mode: one is One-Shot mode and the other is Non-Stop mode. When register bit T4OS (T4CR1[2]) is 1, One-Shot mode is selected. Timer4 will count down once from initial value stored on register TMR4[9:0] to 0x00, i.e. underflow is occurred. When register bit T4OS (T4CR1[2]) is 0, Non-Stop mode is selected. When underflow is occurred, there are two selections to start next down-count which is determined by register bit T4RL (T4CR1[1]). When T4RL is 1, the initial value stored on register TMR4[9:0] will be restored and start next down-count from this initial value. When T4RL is 0, Timer1 will start next down-count from 0x3FF.

When Timer4 is underflow, the register bit T4IF (INTE2[6]) will be set to 1 to indicate Timer4 underflow event is occurred. If register bit T4IE (INTE2[2]) and GIE are both set to 1, interrupt request will occur and interrupt service routine will be executed. T4IF will not be clear until firmware writes 0 to T4IF.

The timing chart of Timer4 is shown in the following figure.

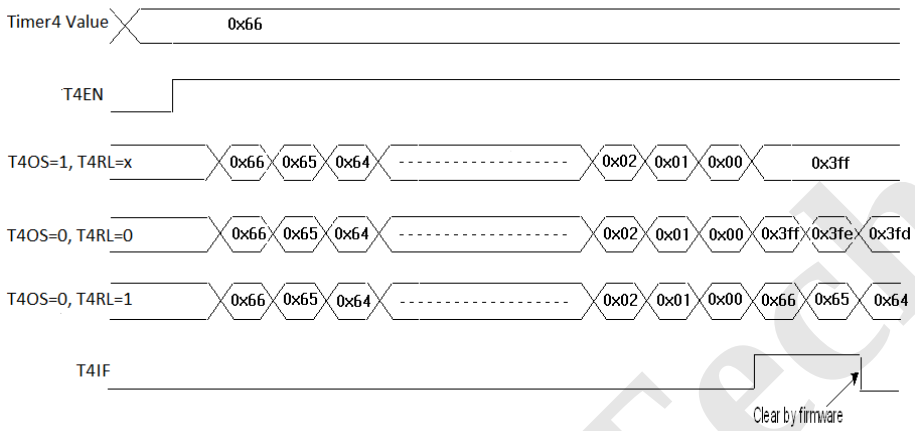


Figure 20 Timer4 Timing Chart

The PWM4 output is available on I/O pin PB3. When register bit PWM4OEN (T4CR1[7]) is set to 1, the corresponded PWM4 pin will became output pin automatically. The active state of PWM4 output is determined by register bit PWM4OAL (T4CR1[6]). When PWM4OAL is 1, PWM4 output is active low. When PWM4OAL is 0, PWM4 output is active high. Moreover, the duty cycle and frame rate of PWM4 are both programmable. The duty cycle is determined by registers TM4RH[3:2] and PWM4DUTY[7:0]. When PWM4DUTY is 0, PWM4 output will be never active. When PWM4DUTY is 0x3FF, PWM4 output will be active for 1023 Timer4 input clocks. The frame rate is determined by TM4RH[7:6] + TMR4[7:0] initial value. Therefore, PWM4DUTY value must be less than or equal to TM4RH[7:6] + TMR4[7:0]. When user write PWM4DUTY, write PWM4DUTY[9:8] MSB 2

bits(TM4RH[3:2]) first and write PWM4DUTY[7:0] second, PWM4 duty register will be updated after Timer4 overflow occurs. The block diagram of PWM4 is illustrated in the following figure.

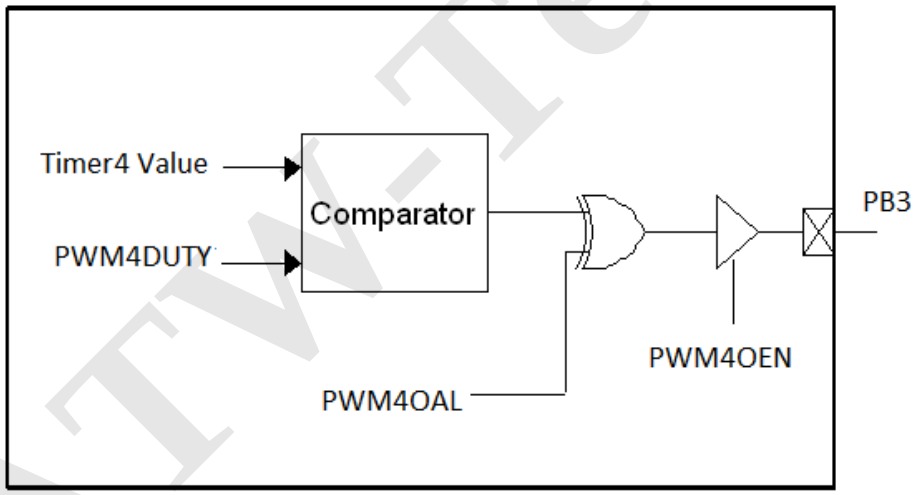


Figure 21 PWM4 Block Diagram

3.12 Timer5 / PWM5

Timer5 is an 10-bit down-count timer with Prescaler5 whose dividing rate is programmable. The output of Timer5 can be used to generate PWM5 output. Timer5 builds in auto-reload function and Timer5 reload register stores reload data with double buffers. When user write Timer5 reload register, write Timer5 MSB 2 bits(TM5RH[5:4]) first and write TMR5 second, Timer5 reload register will be updated to Timer5 counter after Timer5 overflow occurs when T5EN=1. If T5EN=0, Timer1 reload register will be updated to Timer5 counter after write TMR5 immediately. A read to the Timer5 will show the content of the Timer5 current count value. The block diagram of Timer5 is shown in the figure below.

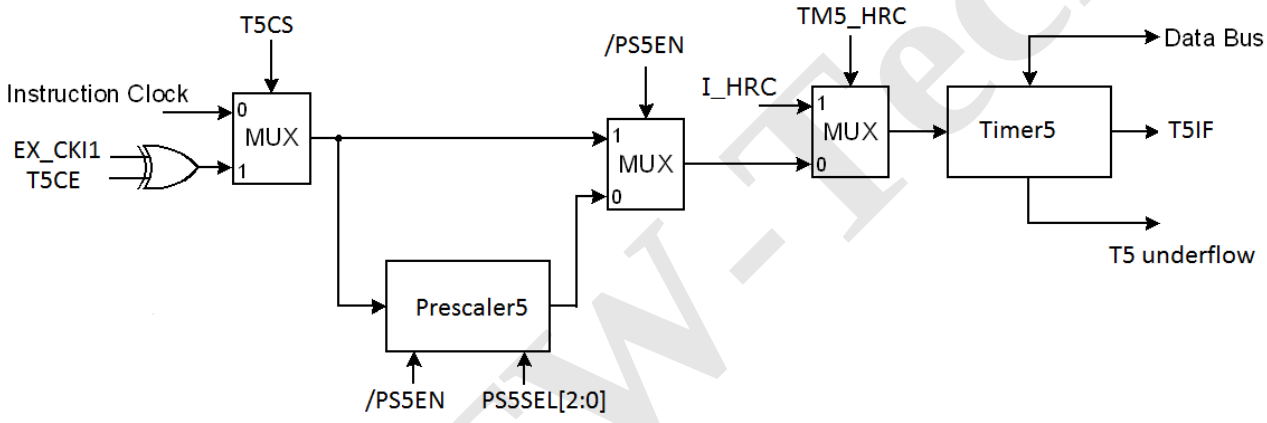


Figure 19 Block Diagram of Timer5

The operation of Timer5 can be enabled or disabled by register bit T5EN (T5CR1[0]). After Timer5 is enabled, its clock source can be instruction clock, pin EX_CKI1 or I_HRC output, which is determined by register bit T5CS (T5CR2[5]) and TM5_HRC(T5CR1[3]). When T5CS is 1 and TM5_HRC is 0, EX_CKI1 is selected as clock source. When T5CS is 0 and TM5_HRC is 0, instruction clock is selected as clock source. When TM5_HRC is 1, I_HRC output is selected as clock source. When EX_CKI1 is selected, the active edge to decrease Timer5 is determined by register bit T5CE (T5CR2[4]). When T5CE is 1, high-to-low transition on EX_CKI1 will decrease Timer1. When T5CE is 0, low-to-high transition on EX_CKI1 will decrease Timer5. The selected clock source can be divided further by Prescaler5 before it is applied to Timer5. Prescaler5 is enabled by writing 0 to register bit /PS5EN (T5CR2[3]) and the dividing rate is from 1:2 to 1:256 determined by register bits PS5SEL[2:0] (T5CR2[2:0]). Current value of Prescaler1 can be obtained by reading register PS5CV.

Timer5 provides two kinds of operating mode: one is One-Shot mode and the other is Non-Stop mode. When register bit T5OS (T5CR1[2]) is 1, One-Shot mode is selected. Timer5 will count down once from initial value stored on register TMR5[9:0] to 0x00, i.e. underflow is occurred. When register bit T5OS (T5CR1[2]) is 0, Non-Stop mode is selected. When underflow is occurred, there are two selections to start next down-count which is determined by register bit T5RL (T5CR1[1]). When T5RL is 1, the initial value stored on register TMR5[9:0] will be restored and start next down-count from this initial value. When T5RL is 0, Timer1 will start next down-count from 0x3FF.

When Timer5 is underflow, the register bit T5IF (INTF3[5]) will be set to 1 to indicate Timer5 underflow event is occurred. If register bit T5IE (INTE3[5]) and GIE are both set to 1, interrupt request will occur and interrupt service routine will be executed. T5IF will not be clear until firmware writes 0 to T5IF.

The timing chart of Timer5 is shown in the following figure.

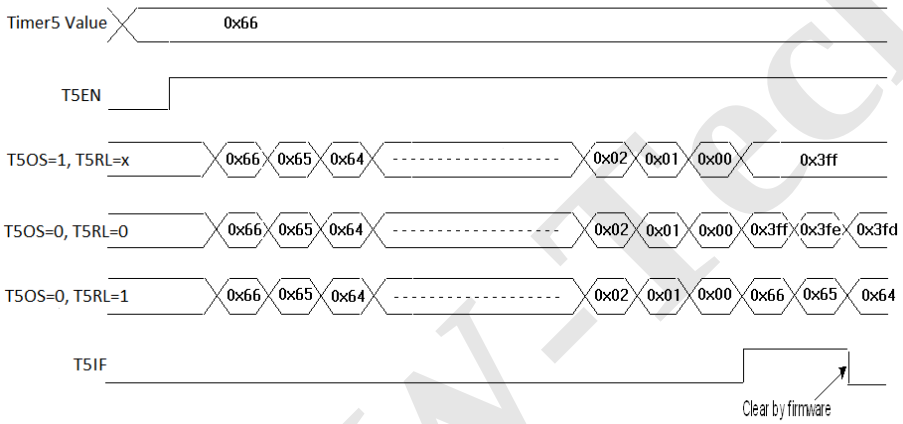


Figure 20 Timer5 Timing Chart

The PWM5 output is available on I/O pin PB2. When register bit PWM5OEN (T5CR1[7]) is set to 1, the corresponded PWM5 pin will became output pin automatically. The active state of PWM5 output is determined by register bit PWM5OAL (T5CR1[6]). When PWM5OAL is 1, PWM5 output is active low. When PWM5OAL is 0,

PWM5 output is active high. Moreover, the duty cycle and frame rate of PWM5 are both programmable. The duty cycle is determined by registers TM5RH[1:0] and PWM5DUTY[7:0]. When PWM5DUTY is 0, PWM5 output will be never active. When PWM5DUTY is 0x3FF, PWM5 output will be active for 1023 Timer5 input clocks. The frame rate is determined by TM5RH[5:4] + TMR5[7:0] initial value. Therefore, PWM5DUTY value must be less than or equal to TM5RH[5:4] + TMR5[7:0]. When user write PWM5DUTY, write PWM5DUTY[9:8] MSB 2 bits(TM5RH[1:0]) first and write PWM5DUTY[7:0] second, PWM5 duty register will be updated after Timer5 overflow occurs. The block diagram of PWM5 is illustrated in the following figure.

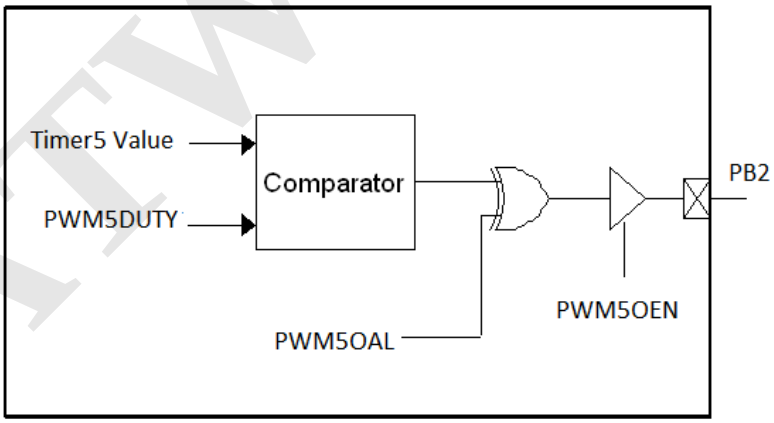


Figure 21 PWM5 Block Diagram

3.13 CCP Mode

The CCP (Capture/Compare/PWM) register (CCPR) is comprised of two 8-bit registers: CCPRL (low byte) and CCPRH (high byte). The CCPCON and PWMDB registers control the operation of CCP. The capture, compare mode use16-bit timer and PWM mode use10-bit timer respectively. The CCP timers and registers utilize the existing timers and registers. The following table shows the registers and timers resources shared with CCP modules. Note that when AT8BE62D is in CCP mode, related timer/PWM function are disabled.

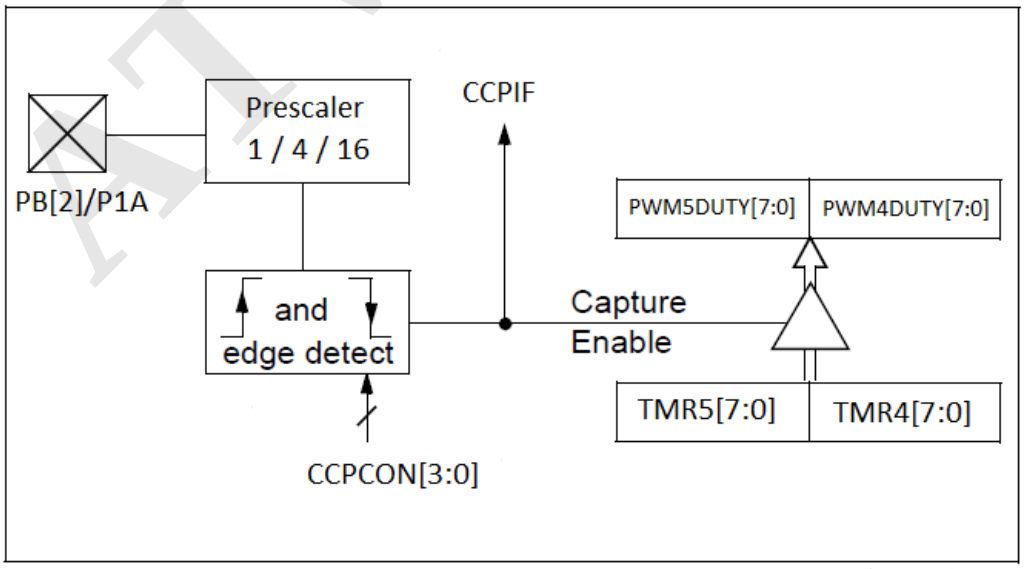
CCP mode	CCP register/timer	Shared Timers	Shared Registers
Capture/Compare	CCPL	-	PWM4DUTY[7:0]
Capture/Compare	CCPH	-	PWM5DUTY[7:0]
PWM	PWMDUTY	-	PWM5DUTY[9:0]
PWM	PWMDB	-	-
Capture/Compare	Capture/Compare timer	Timer5[7:0] (MSB) + Timer4[7:0] (LSB)	-
PWM	PWM timer	Timer5[9:0]	-

3.13.1 Capture Mode

In capture mode, CCPRH:CCPRL (That is, PWM5DUTY[7:0]:PWM4DUTY[7:0]) captures the 16-bit value of capture timer register when an event occurs on pin P1A (PB2). An event is defined as:

- every falling edge
- every rising edge
- every 4th rising edge
- every 16th rising edge

The block diagram of capture mode is as the following:



An event is selected by control bits CCPM[3:0]. When a capture is made, the interrupt request flag bit CCPIF is set.

In capture mode, the P1A(PB2) should be configured ad an input. If it is an output pin, a write to the port can cause a capture condition.

In capture mode, capture timer must be running in CPU clock synchronous mode, or the capture operation may not work.

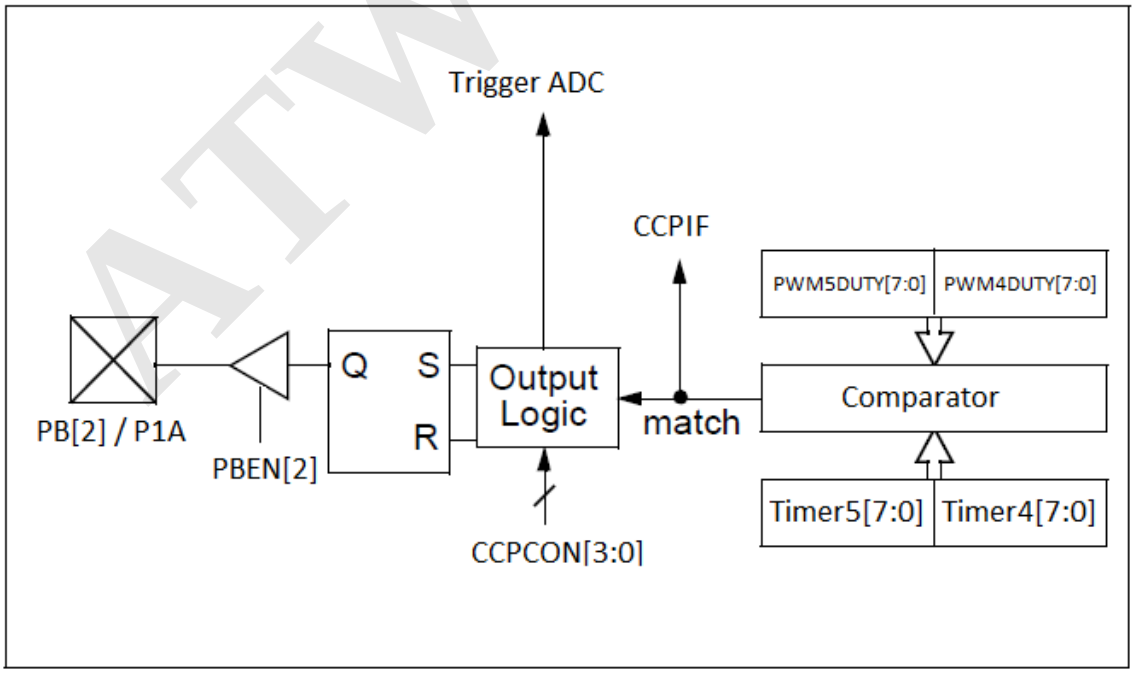
The capture event counter (or prescaler) is cleared when not in capture mode. Switching from one capture prescaler to another may generate an interrupt. Also, the prescaler counter will not be cleared.

3.13.2 Compare Mode

In compare mode, the 16-bit CCPR (That is, PWM5DUTY[7:0]:PWM4DUTY[7:0]) register value is constantly compared against the compare timer value. When a match occurs, the CCP1 (PB2) pin is :

- driven high
- driven low
- toggle output
- remain unchanged (only interrupt)
- Trigger ADC if ADC is enabled

The compare mode block diagram is as the below shown:



The action on the pin is based on the value of control bit CCPM[3:0]. When compare match occurs, interrupt flag bit CCPIF is set.

In compare mode, the user must configure the CCP (PB2) pin as an output.

In compare mode, compare timer must be running in CPU clock synchronous mode, or the compare operation may not work.

3.13.3 CCP PWM Mode

In CCP PWM mode, the CCP module produces up to a 10-bit resolution PWM output. The PWM pads are P1A(PB2), P1B(PA5), P1C(PA2) and P1D(PA4). The PWM period and duty are specified by Timer5[9:0] and PWM5DUTY[9:0] registers.

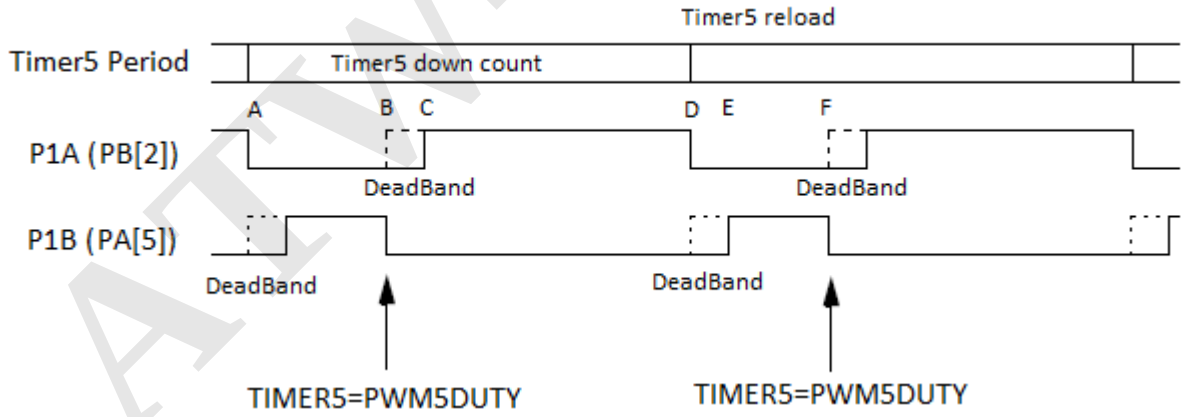
Register PWM5M[1:0] in the CCPCON register allows one of the following configurations:

- Single output: P1A output. P1B, P1C, P1D assigned as normal I/O.
- Half-Bridge output: P1A, P1B modulated with deadband control. P1C, P1D assign as normal I/O.

- Full-Bridge output, Forward mode: P1D modulated, P1A active. P1B, P1C inactive.
- Full-Bridge output, Reverse mode: P1B modulated, P1C active. P1A, P1D inactive.

In single output mode, P1A (PB2) pin is used as PWM output. PB2 must be set as output.

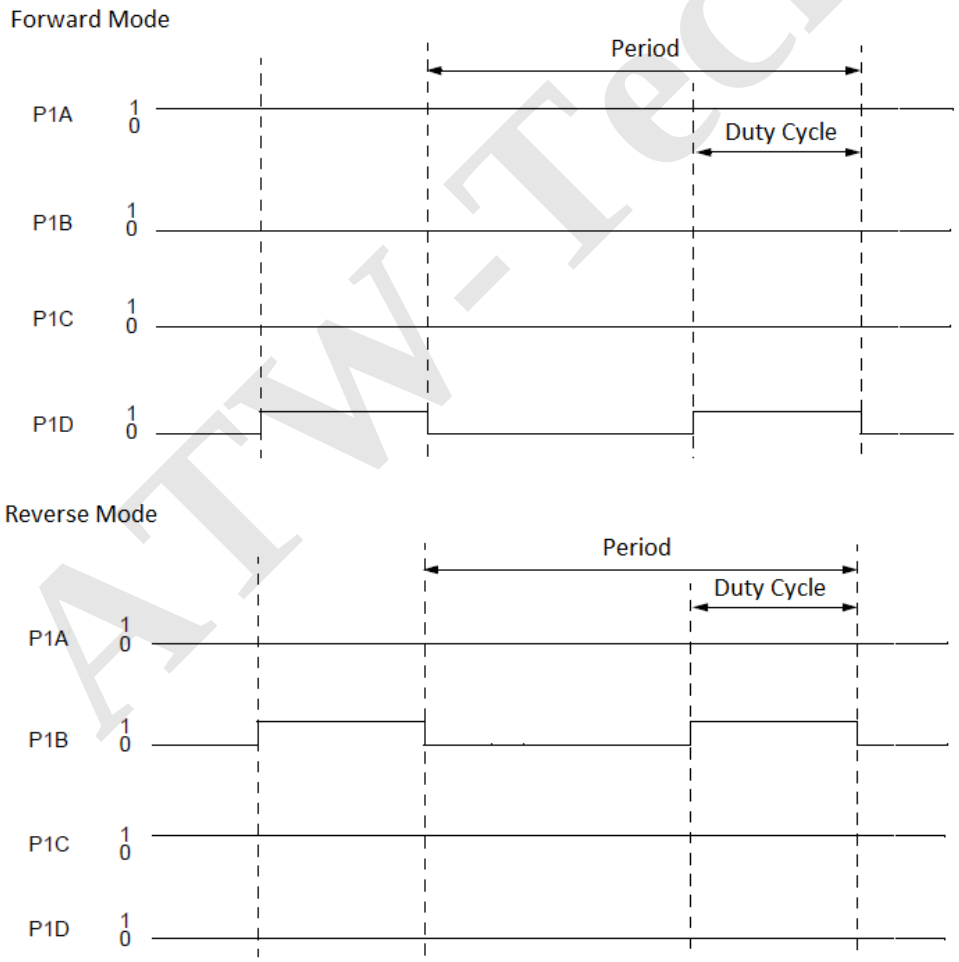
In Half-Bridge mode, P1A (PB2) pin has the PWM output signal, P1B (PA5) pin has the complementary PWM output signal. In this mode, PB2 and PA5 must be set as outputs.



In the Half-Bridge output mode, a digitally programmable deadband delay is available to avoid shoot-through current from destroying the bridge power switch. PWMDB[7:0] is deadband delay count for Half-Bridge mode, the delay unit is in CPU cycle.

As the above timing shown, timer5 is a 10-bit down counter, P1A is PWM5 output, P1B is PWM5 complementary output. AT point B, Timer5 value is equal to PWM5DUTY. At point D, Timer5 count to zero and reload value. If there no deadband control, P1A output should be active from point B to Point D. P1B output should be active from point D to point F. A non-zero deadband will delay P1A rising point from B to C and delay P1B rising point from D to E. The deadband zone are the time from B to C and from D to E.

In Full-Bridge mode, four pins are used as outputs; however, only two outputs are active at a time. In the Forward mode, P1A(PB2) is continuously activated and P1D(PA4) is modulated. In the Reverse mode, P1C(PA2) is continuously activated and P1B(PA5) is modulated. In this mode, PB2, PA5, PA2 and PA4 must be set as outputs. The following timing diagram shows Full-Bridge Forward and Reverse condition.



3.14 RFC Mode

AT8BE62D has built-in RFC mode. Once RFC mode is enabled, the selected input pad state will take control of the Timer1 counting. When the selected input pad is recognized as 0 state (The input pad voltage is smaller than V_{IL}), Timer1 keeps counting. When this selected pad is recognized as 1 (The input pad voltage is larger than V_{IH}), Timer1 stops counting. The following figure shows how RFC mode operates: PSEL3~0 is used to select one RFC input pad out of 16 AT8BE62D pads. RFCEN is used to switch the Timer1 enable signal between the normal enable signal T1EN and RFC selected input state.

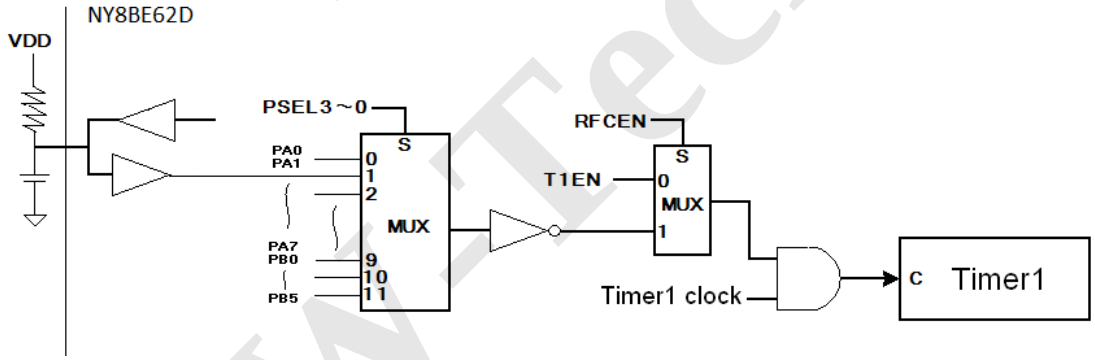


Figure 31 RFC Block Diagram

One application of RFC mode is to measure the capacitor-resistor charging time, As the figure shows, when PSEL3~0=0x01, PA1 is selected as RFC input pad. At first the PA1 is set as output low (the voltage of PA1 is discharged to 0). Next step, clear Timer1 content, set PA1 as input and enable RFC mode. Then Timer1 will start counting, and the RC circuit will start charging PA1. As PA1 is charged to the V_{IH} voltage, the Timer1 counting is stopped because PA1 input is high. The Timer1 content will show the RC circuit charging time. (Note: Timer1 is down-count.)

3.15 IR Carrier

According to the IR PAD configuration, the IR output pad of AT8BE62D can be PB1 or PA3. The IR carrier will be generated after register bit IREN (IRCR[0]) is set to 1. Moreover, the IP pad will become output pin automatically. When IREN is clear to 0, the IP pad will become general I/O pin as it was configured.

The IR carrier frequency is selectable by register bit IRF57K (IRCR[1]). When IRF57K is 1, IR carrier frequency is 57KHz. When IRF57K is 0, IR carrier frequency is 38KHz. Because IR carrier frequency is derived from high frequency system oscillation F_{HOSC}, it is necessary to specify what frequency is used as system oscillation when external crystal is used. Register bit IROSC358M (IRCR[7]) is used to provide AT8BE62D this information. When IROSC358M is 1, frequency of external crystal is 3.58MHz and when IROSC358M is 0, frequency of external crystal is 455KHz. When internal high frequency oscillation is adopted, this register will be ignored

The active state (polarity) of IR carrier is selectable. When register bit IRCSEL (IRCR[2]) is 1, IR carrier will be present on the IR pad when its output data is 0. When register bit IRCSEL (IRCR[2]) is 0, IR carrier will be present on the IR pad when its output data is 1. The polarity of IR carrier is shown in the following figure.

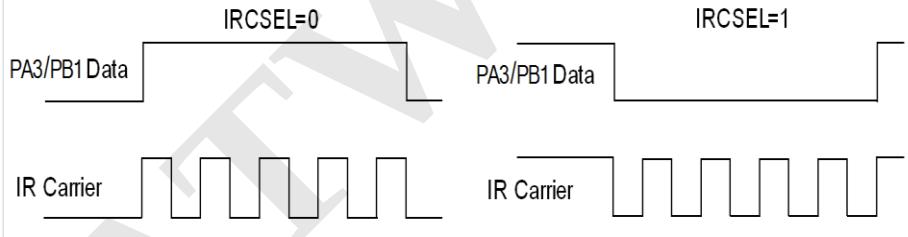


Figure 32 Polarity of IR Carrier vs. Output Data

3.16 Low Voltage Detector (LVD)

The low voltage detector (LVD) and comparator function are integrated in the AT8BE62D. PS[1:0] (Register CMPCR[3:2]) = 10 and NS[1:0] (Register CMPCR[1:0])=10 set AT8BE62D in LVD mode. LVD is built-in a precise band-gap reference for accurately detecting V_{DD} level. If LVDEN(register PCON[5])=1 and V_{DD} voltage value falls below LVD voltage which is selected by LVDS[3:0] as table shown below, the LVD output will be low. If the LVD interrupt is enabled, the LVD interrupt flag will be high and if GIE=1 the program will start interrupt service routine. Moreover, LVD real-time output can be polled by register PCON1[6]. The following is LVD block diagram:

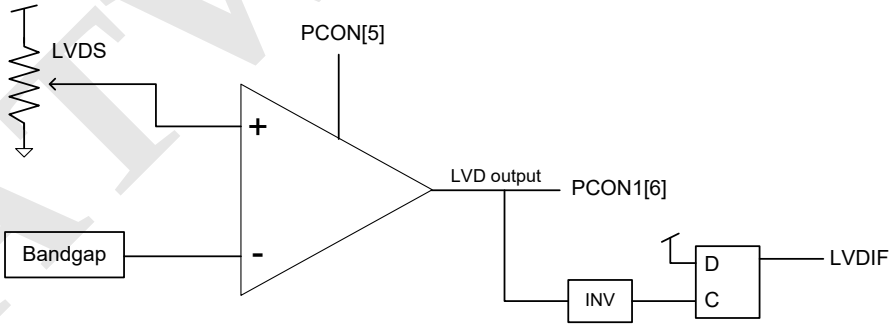


Figure 33 LVD block diagram

LVDS[3:0]	Voltage
0	1.9V
1	2.0V
10	2.2V
11	2.4V
100	2.6V
101	2.8V
110	2.9V
111	3.0V
1000	3.15V
1001	3.30V
1010	3.45V
1011	3.60V
1100	3.75V
1101	3.90V
1110	4.05V
1111	4.15V

The following table is LVD voltage select table.

LVDS[3:0]	Voltage
0	--
1	--
10	(2.2+0.1) V
11	(2.4+0.1) V
100	(2.6+0.1) V
101	(2.8+0.1) V
110	(2.9+0.1) V
111	(3.0+0.1) V
1000	(3.15+0.1) V
1001	(3.30+0.1) V
1010	(3.45+0.1) V
1011	(3.60+0.1) V
1100	(3.75+0.1) V
1101	(3.90+0.1) V
1110	(4.05+0.1) V
1111	(4.15+0.1) V

Table 23 LVD voltage select

Note: The hysteresis voltage (from low to high) of LVD is about 0.1V. In battery charging applications (detected voltage is from low to high), the LVD voltage select table should be as followed:

The LVD control flow is as the following:

Step1: Select LVD voltage by LVDS[3:0]

Step2: Set CMPCR = 0x0A

Step3: Set PCON[5]=1 (enable LVD)

Step4: Check LVD status by PCON1[6]

Note: If LVD voltage LVDS[3:0] is changed, user must wait at least 50us(@F_{HOSC}=1MHz) to get correct LVD status by PCON1[6]

3.17 Voltage Comparator

AT8BE62D provides voltage comparator and internal reference voltage with various analog comparing mode. The comparator non-inverting and inverting input can share with GPIO.

CMPEN (register PCON[2]) is used to enable and disable comparator. When CMPEN=0(default), comparator is disabled. When CMPEN=1, the comparator is enabled. In halt mode the comparator is disabled automatically.

The structure of comparator is shown in the following figure:

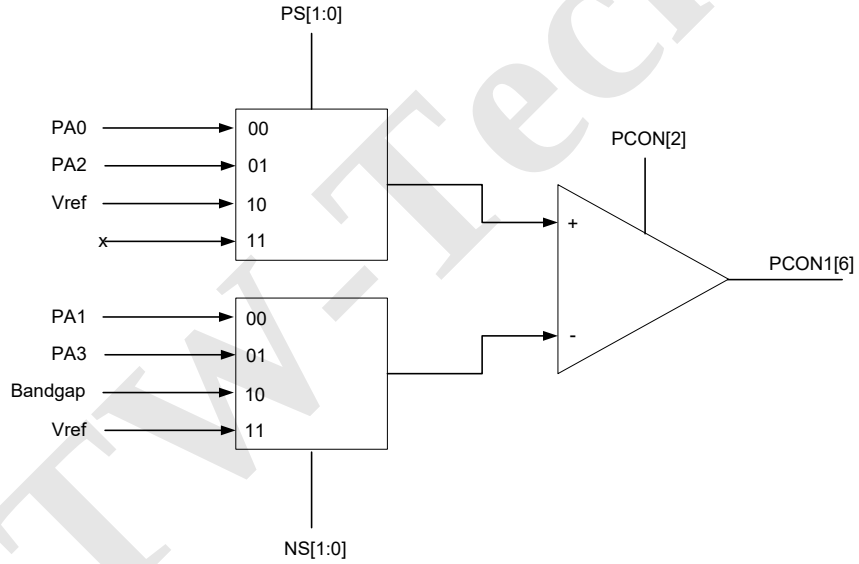
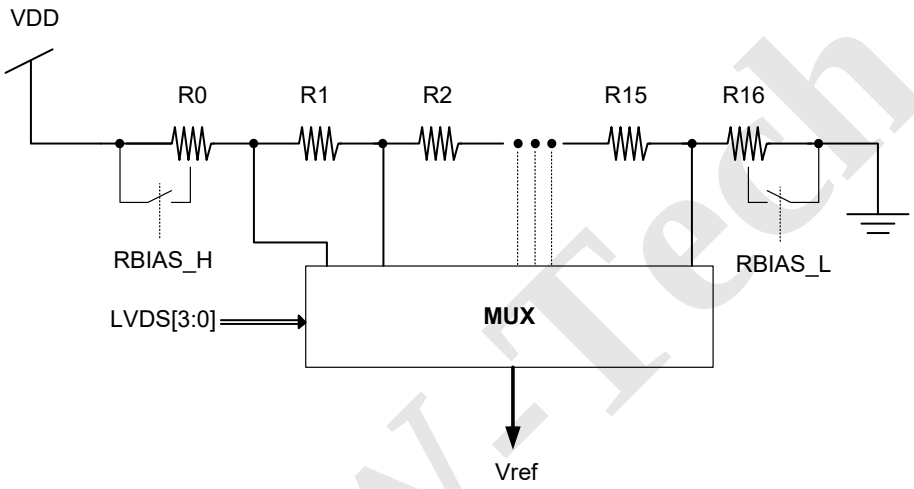


Figure 20 Comparator block diagram

3.17.1 Comparator Reference Voltage (Vref)

The internal reference voltage Vref is built by series resistance to provide different level of reference voltage. RBIAS_H and RBIAS_L are used to select the maximum and minimum values of Vref, and LVDS[3:0] are used to select one of 16 voltage levels.



The **Vref** is determined by RBIAS_H, RBIAS_L and LVDS[3:0]. The LVDS[3:0] is used to select one out of 16 reference voltages, the table shown below.

Symbol	Condition				
	LVD		trim		
	RBIAS[H:L]	LVDS[3:0]	VDD=5V V_{IN-} = $VDD_{\_REF}$ (V)		
Vref (V)	10	0000b	67.891/128*VDD =	0.530*VDD =	2.652
	10	0001b	64.563/128*VDD =	0.504*VDD =	2.522
	10	0010b	58.957/128*VDD =	0.461*VDD =	2.303
	10	0011b	54.579/128*VDD =	0.426*VDD =	2.132
	10	0100b	50.458/128*VDD =	0.394*VDD =	1.971
	10	0101b	46.763/128*VDD =	0.365*VDD =	1.827
	10	0110b	45.483/128*VDD =	0.355*VDD =	1.777
	10	0111b	44.117/128*VDD =	0.345*VDD =	1.723
	10	1000b	42.069/128*VDD =	0.329*VDD =	1.643
	10	1001b	39.851/128*VDD =	0.311*VDD =	1.557
	10	1010b	38.477/128*VDD =	0.301*VDD =	1.500
	10	1011b	36.693/128*VDD =	0.287*VDD =	1.433
	10	1100b	35.243/128*VDD =	0.275*VDD =	1.377
	10	1101b	34.099/128*VDD =	0.266*VDD =	1.332
	10	1110b	33.109/128*VDD =	0.259*VDD =	1.293
	10	1111b	32.171/128*VDD =	0.251*VDD =	1.257
	01	0000b	34.901/128*VDD =	0.273*VDD =	1.363
	01	0001b	32.085/128*VDD =	0.251*VDD =	1.253
	01	0010b	28.468/128*VDD =	0.222*VDD =	1.112
	01	0011b	25.173/128*VDD =	0.197*VDD =	0.983
	01	0100b	22.272/128*VDD =	0.174*VDD =	0.870
	01	0101b	19.968/128*VDD =	0.156*VDD =	0.780
	01	0110b	18.773/128*VDD =	0.147*VDD =	0.733
	01	0111b	17.664/128*VDD =	0.138*VDD =	0.690
	01	1000b	16.299/128*VDD =	0.127*VDD =	0.637
	01	1001b	15.019/128*VDD =	0.117*VDD =	0.587
	01	1010b	13.653/128*VDD =	0.107*VDD =	0.533
	01	1011b	12.544/128*VDD =	0.098*VDD =	0.490
	01	1100b	11.947/128*VDD =	0.093*VDD =	0.467
	01	1101b	10.923/128*VDD =	0.085*VDD =	0.427
	01	1110b	10.069/128*VDD =	0.079*VDD =	0.393
	01	1111b	9.557/128*VDD =	0.075*VDD =	0.373

Table 13 The reference voltage Vref selection table

Note:

1. The deviation of Vref is ±0.1V.

2. RBIAS_H and RBIAS_L must be set as "0" to avoid power consumption in Halt mode or Standby mode.

The non-inverting input of the comparator is determined by PS[1:0] (register CMPCR[3:2]). The table is shown below

PS[1:0]	Non-inverting input
00	PA0
01	PA2
10	Vref
11	---

Table 14 Non-inverting input select

The inverting input of the comparator is determined by NS[1:0] (register CMPCR[1:0]). The table is shown below

NS[1:0]	Inverting input
00	PA1
01	PA3
10	Bandgap (0.6V)
11	Vref

Table 15 Inverting input select

There are two ways to get the comparator output result: one is through register polling, the other is through probing output pad.

Comparator output can be polled by LVDOUT (register PCON1[6]).

To probe comparator output at output pad, set CMPOE (register OSCCR[6]) to 1, then PB1 will be the real-time state of the comparator output. It is noted that when CMPOE=1, the PWM3 function will be disabled if it is enabled.

3.18 Analog-to-Digital Convertor (ADC)

AT8BE62D provide 11+2 channel 12-bit SAR ADC to transfer analog signal into 12-bits digital data. The ADC high reference voltage is selectable. They can be external voltage from PA0, or internal generated voltage VDD, 4V, 3V or 2V. The Analog input is selected from analog signal input pin PA0~PA4, PB0~PB5, internal generated 1 / 4 *VDD or Ground. The ADC clock ADCLK can be selected to be F_{INST}/1, F_{INST}/2, F_{INST}/8 or F_{INST}/16. The Sampling pulse width can be selected to be ADCLK*1, ADCLK*2, ADCLK*4 or ADCLK*8. Set ADEN=1 before ADC take into operation. Then set START=1, the ADC will start to convert analog signal to digital. EOC=0 means ADC is in processing. EOC=1 indicate ADC is at end of conversion. If ADIE=1 and global interrupt is enabled, the ADC interrupt will issue after EOC low go high. The block diagram is as following figure.

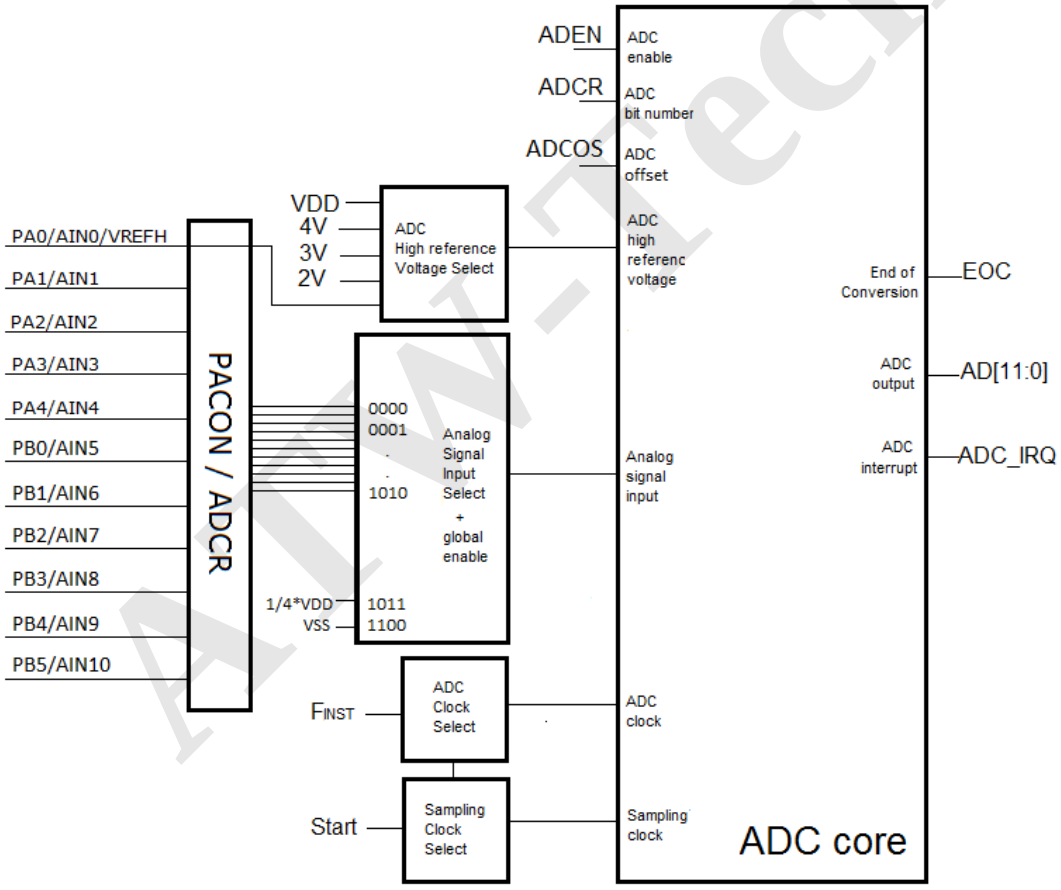


Figure 36 ADC block diagram

3.18.1 ADC reference voltage

ADC is built-in five high reference voltage source controlled by ADVREFH register. These high reference voltage source are one external voltage source (PA0) and four internal voltage source (VDD, 4V, 3V, 2V). When EVHENB bit is “1”, ADC reference voltage is external voltage source from PA0. In this mode PA0 must input a voltage between VDD and 2V. If EVHENB bit is 0, ADC reference voltage is from internal voltage source selected by VHS[1:0]. If VHS[1:0] is “11”, ADC reference voltage is VDD. If VHS[1:0] is “10”, ADC reference voltage is 4V. If VHS[1:0] is “01”, ADC reference voltage is 3V. If VHS[1:0] is “00”, ADC reference voltage is 2V. The limitation of internal reference voltage application is VDD can't below each of internal voltage level, or the level is equal to VDD. ADC sampling voltage range is limited by high/low reference voltage. The ADC low reference voltage is VSS and not changeable. The ADC high reference voltage includes internal VDD/4V/3V/2V and external reference voltage source from PA0 pin. The ADC reference voltage range limitation is (ADC high reference voltage – low reference voltage) ≥ 2V. ADC low reference voltage is VSS=0V. So ADC high reference voltage range is 2V ~ VDD.

ADC analog input signal voltage must be from ADC low reference voltage to ADC high reference voltage. If the ADC analog input signal voltage is over this range, The ADC converting result is unexpected (full scale or zero).

EVHENB	VHS[1:0]	Reference voltage
1	x x	PA0
0	1 1	VDD
0	1 0	4V
0	0 1	3V
0	0 0	2V

Table 27 ADC reference voltage select

3.18.2 ADC analog input channel

ADC use CHS[3:0] and GCHS to select analog input source. GCHS is global channel select. Namely, GCHS must be 1 before any analog input source can be selected and converted.

GCHS	CHS[3:0]	ADC analog input source
0	xxxx	x
1	0000	PA0
1	0001	PA1
1	0010	PA2
1	0011	PA3
1	0100	PA4
1	0101	PB0
1	0110	PB1
1	0111	PB2
1	1000	PB3
1	1001	PB4
1	1010	PB5
1	1011	1 / 4 * VDD
1	1100	VSS
1	1101	N.C.
1	111x	N.C.

Table 28 ADC analog input source select

ADC input pins are shared with digital I/O pins. Connect an analog signal to these pin may cause extra current leakage in I/O pins. In the power down mode, the above leakage current will be a big problem. Register bit PACONx / PBCONx is PAx / PBx configuration register bit to solve above problem. Write “1” to PACONx / PBCONx register bit will configure related PAx / PBx pin as pure analog input pin to avoid current leakage, and it can't be use as normal I/O.

Except setting the PACONx / PBCONx register bit, the selected analog input pin must be set as input mode and the internal pull-high / pull-down must be disabled, otherwise the analog input level may be affected.

3.18.3 ADC clock (ADCLK), sampling clock (SHCLK) and bit number

Conversion speed and conversion accuracy are affected by the selection of the ADC clock (ADCLK), sampling pulse width (SHCLK) and conversion bit number. ADCLK is the base clock of ADC. During the operation of SAR ADC, bit operation is synchronized with ADCLK. SHCLK is the duration of analog signal sampling time, larger SHCLK will restore original analog signal level more closely but it will slow down the ADC conversion speed. Vise versa. The ADC can select different conversion bit number which is depended on ADCR[1:0] register bits. There are 2 bit number to select, which is 12-bit, 10-bit and 8-bit. Less conversion bit number will speed up the ADC conversion rate but the effective ADC bit is less. More conversion bit number will slow down the conversion rate but the accuracy is more.

The ADC clock is derived from F_{INST} and is selectable from ADCK[1:0].

ADCK[1:0]	ADC clock
0 0	F_{INST}/16
0 1	F_{INST}/8
1 0	F_{INST}/1
1 1	F_{INST}/2

Table 29 ADC clock select

The Sampling clock width is derived from ADCLK and is selectable from SHCK[1:0].

SHCK[1:0]	Sampling clock
0 0	1 ADCLK
0 1	2 ADCLK
1 0	4 ADCLK
1 1	8 ADCLK

Table 30 ADC sampling clock select

ADC bit number select is from ADCR[1:0].

ADCR[1:0]	Conversion bit number
0 0	8-bit
0 1	10-bit
1 x	12-bit

Table 31 conversion bit number select

The ADC converting time is from START(Start to ADC convert) to EOC=1 (End of ADC convert). The duration is depending on ADC resolution and ADC clock rate and sampling clock width.

ADC conversion time ≈ sampling clock width + (ADC bit number + 2) * ADCLK width.

The following table is some example conversion time and conversion rate of ADC.

Bit No.	ADC clock	SHCLK	Conversion Time (ADCLK No.)	F_{INST}=2MHz		F_{INST} =250K	
				Time	Rate	Time	Rate
12	F_{INST}/16	8 ADCLK	22	176us	5.68kHz	1408us	710Hz
12	F_{INST}/1	1 ADCLK	15	7.5us	133.3kHz	60us	16.7kHz
10	F_{INST}/1	1 ADCLK	13	6.5us	153.8kHz	52us	19.2kHz
8	F_{INST}/1	1 ADCLK	11	5.5us	181.8kHz	44us	22.7kHz

Table 32 ADC Conversion time

3.18.4 ADC offset calibration

ADC offset error is defined as the deviation between the first ideal code transition and the first actual code transition. The first ideal code transition take place at 0.5LSB. ADC offset error varies with temperature, process and voltage. ADC offset error can be real-time adjusted in AT8BE62D through ADJMD register. ATW provides the NYIDE example code "ADC_Interrupt_AutoK" for ADC offset calibration process.

3.18.5 ADC operation

Set ADC clock (ADCLK), sampling clock width (SHCLK), conversion bit number (ADCR), ADC high reference voltage (ADVREFH), select input channel and PACON related bit. Then set ADEN=1.

After setting ADEN=1, it must wait at least 256us (ADC internal bias stable time) before ADC can operate. Write START to 1 to start an ADC conversion session. During ADC is in processing EOC=0. Polling EOC=1 or wait for ADC interrupt at the end of ADC conversion.

3.19 Watch-Dog Timer (WDT)

There is an on-chip free-running oscillator in AT8BE62D which is used by WDT. As this oscillator is independent of other oscillation circuits, WDT may still keep working during Standby mode and Halt mode.

WDT can be enabled or disabled by a configuration word. When WDT is enabled by configuration word, its operation still can be controlled by register bit WDTEN (PCON[7]) during program execution. Moreover, the mechanism after WDT time-out can reset AT8BE62D or issue an interrupt request which is determined by another configuration word. At the same time, register bit /TO (STATUS[4]) will be clear to 0 after WDT time-out.

The baseline of WDT time-out period can be 3.5 ms, 15 ms, 60 ms or 250 ms which is determined by two configuration words. The time-out period can be lengthened if Prescaler0 is assigned to WDT. Prescaler0 will be assigned to WDT by writing 1 to register bit PS0WDT. The dividing rate of Prescaler0 for WDT is determined by register bits PS0SEL[2:0] and depends on WDT time-out mechanism. The dividing rate is from 1:1 to 1:128 if WDT time-out will reset AT8BE62D and dividing rate is from 1:2 to 1:256 if WDT time-out will interrupt AT8BE62D.

When Prescaler0 is assigned to WDT, the execution of instruction CLRWDT will clear WDT, Prescaler0 and set /TO flag to 1.

If user selects interrupt for WDT time-out mechanism, register bit WDTIF (INTF[6]) will set to 1 after WDT is expired. It may generate an interrupt request if register bit WDTIE (INTE[6]) and GIE both set to 1. WDTIF will not be clear until firmware writes 0 to WDTIF.

3.20 Interrupt

AT8BE62D provides two kinds of interrupt: one is software interrupt and the other is hardware interrupt. Software interrupt is caused by execution of instruction INT. There are 12 hardware interrupts:

- Timer0 overflow interrupt.
- Timer1 underflow interrupt.
- Timer4 underflow interrupt.
- Timer5/CCP underflow interrupt.
- WDT timeout interrupt.
- PA/PB input change interrupt.
- External 0 interrupt.
- External 1 interrupt
- External 2 interrupt
- LVD/comparator interrupt.
- ADC end-of-convert interrupt.
- EEPROM write complete interrupt.

GIE is global interrupt enable flag. It has to be 1 to enable hardware interrupt functions. GIE can be set by ENI instruction and clear to 0 by DISI instruction.

After instruction INT is executed, no matter GIE is set or clear, the next instruction will be fetched from address 0x001. At the same time, GIE will be clear to 0 by AT8BE62D automatically. This will prevent nested interrupt from happening. The last instruction of interrupt service routine of software interrupt has to be RETIE. Execution of this instruction will set GIE to 1 and return to original execution sequence.

While any of hardware interrupts is occurred, the corresponding bit of interrupt flag will be set to 1. This bit will not be clear until firmware writes 0 to this bit. Therefore user can obtain information of which event causes hardware interrupt by polling the corresponding bit of interrupt flag. Note that only when the corresponding interrupt enable bit is set to 1, will the corresponding interrupt flag be read. And if the corresponding interrupt enable bit is set to 1 and GIE is also 1, hardware interrupt will occur and next instruction will be fetched from 0x008. At the same time, the register bit GIE will be clear by AT8BE62D automatically. If user wants to implement nested interrupt, instruction ENI can be used as the first instruction of interrupt service routine which will set GIE to 1 again and allow other interrupt events to interrupt AT8BE62D again. Instruction RETIE has to be the last instruction of interrupt service routine which will set GIE to 1 and return to original execution sequence.

It should be noted that ENI instruction cannot be placed right before RETIE instruction because ENI instruction in interrupt service routine will trigger nested interrupt, but RETIE will clear internal interrupt processing after jump out of ISR, so it is possible for interrupt flag to be falsely cleared.

3.20.1 Timer0 Overflow Interrupt

Timer0 overflow (from 0x00 to 0xFF) will set register bit T0IF. This interrupt request will be serviced if T0IE and GIE are set to 1.

3.20.2 Timer1 Underflow Interrupt

Timer1 underflow (from 0x3FF to 0x00) will set register bit T1IF. This interrupt request will be serviced if T1IE and GIE are set to 1.

3.20.3 Timer4 Underflow Interrupt

Timer4 underflow (from 0x3FF to 0x00) will set register bit T4IF. This interrupt request will be serviced if T4IE and GIE are set to 1.

3.20.4 Timer5 Underflow/CCP Interrupt

Timer5 underflow (from 0x3FF to 0x00) will set register bit T5IF. This interrupt request will be serviced if T5IE and GIE are set to 1. When CCP capture / compare mode is enabled, timer5 interrupt is replaced by CCP interrupt.

3.20.5 WDT Timeout Interrupt

When WDT is timeout and the configuration word selects WDT timeout will generate interrupt request, it will set register bit WDTIF. This interrupt request will be serviced if WDTIE and GIE are set to 1.

3.20.6 PA/PB Input Change Interrupt

When PAx, $0 \le x \le 7$, PBy, $0 \le y \le 7$ is configured as input pin and corresponding register bit WUPAx, WUPBx is set to 1, a level change on these selected I/O pin(s) will set register bit PABIF. This interrupt request will be serviced if PABIE and GIE are set to 1. Note when PB0, PB1, PA3, PA4 or PA5 is both set as level change interrupt and external interrupt, the external interrupt enable EIS0, EIS1 or EIS2=1 will disable PB0, PB1, PA3, PA4 or PA5 level change operation.

3.20.7 External 0 Interrupt

According to the configuration of EIS0=1 and INTEDG, the selected active edge on I/O pin PB0 or PB5 will set register bit INT0IF and this interrupt request will be served if INT0IE and GIE are set to 1.

3.20.8 External 1 Interrupt

According to the configuration of EIS1=1 and INTEDG, the selected active edge on I/O pin PB1 or PA3 will set register bit INT1IF and this interrupt request will be served if INT1IE and GIE are set to 1.

3.20.9 External 2 Interrupt

According to the configuration of EIS2=1 and INTEDG, the selected active edge on I/O pin PA5 will set register bit INT2IF and this interrupt request will be served if INT2IE and GIE are set to 1.

3.20.9 LVD/comparator Interrupt

When V_{DD} level falls below LVD voltage or comparator output change state, LVD/comparator IF will be set to 1. This interrupt request will be serviced if LVD/comparator IE and GIE are set to 1.

3.20.10 ADC end of conversion Interrupt

The ADC interrupt is triggered whenever an ADC end-of-convert signal is issued. This interrupt request will be serviced if ADIE and GIE are set to 1.

3.20.11 EEPROM write complete Interrupt

When EEPROM write is completed, the EEPROM write complete flag is issued. This interrupt request will be serviced if EEIE and GIE are set to 1.

3.21 Oscillation Configuration

Because AT8BE62D is a dual-clock IC, there are high oscillation (F_{HOSC}) and low oscillation (F_{LOSC}) that can be selected as system oscillation (F_{OSC}). The oscillators which could be used as F_{HOSC} are internal high RC oscillator (I_HRC), external high crystal oscillator (E_HXT) and external crystal oscillator (E_XT). The oscillators which could be used as F_{LOSC} are internal low RC oscillator (I_LRC) and external low crystal oscillator (E_LXT).

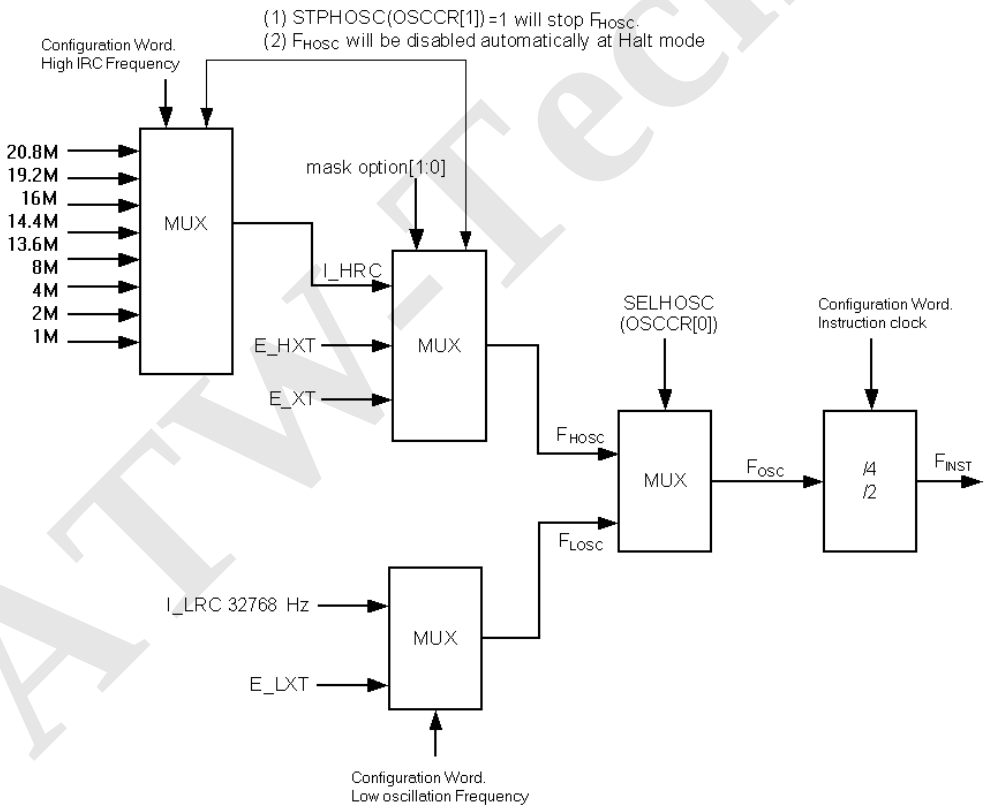


Figure 37 Oscillation Configuration of AT8BE62D

There are two configuration words to determine which oscillator will be used as F_{HOSC}. When I_HRC is selected as F_{HOSC}, I_HRC output frequency is determined by three configuration words and it can be 1M, 2M, 4M, 8M, 16M, 13.6MHz, 14.4MHz, 19.2MHz, 20.8MHz. Moreover, external crystal oscillator pads PA6 and PA7 can be used as I/O pins. On the other hand, PA7 can be the output pin of instruction clock according to a configuration word's setting. If F_{HOSC} required external crystal whose frequency ranges from 8MHz to 20MHz, E_HXT is recommended. If F_{HOSC} required external crystal whose frequency ranges from 455KHz to 6MHz, E_XT is recommended. When E_HXT or E_XT is adopted, PA6/PA7 cannot be used as I/O pins. They must be used as crystal output pin and input pin. PA7 is crystal output pin (Xout) and PA6 is crystal input pin (Xin).

There is one configuration word to determine which oscillator will be used as F_{LOSC}. When I_LRC is selected, its frequency is centered on 32768Hz. If F_{LOSC} required external crystal, E_LXT is selected and only 32768Hz crystal is allowed. When E_LXT is adopted, PA6/PA7 cannot be used as I/O pins. They must be used as crystal output pin and input pin. PA7 is crystal output pin (Xout) and PA6 is crystal input pin (Xin). The dual-clock combinations of F_{HOSC} and F_{LOSC} are listed below.

No.	F_{HOSC}	F_{LOSC}
1	I_HRC	I_LRC
2	E_HXT or E_XT	I_LRC
3	I_HRC	E_LXT

Table 33 Dual-clock combinations

When E_HXT, E_XT or E_LXT is used as one of oscillations, the crystal or resonator is connected to Xin and Xout to provide oscillation. Moreover, a resistor and two capacitors are recommended to connect as following figure in order to provide reliable oscillation, refer to the specification of crystal or resonator to adopt appropriate C1 or C2 value. The recommended value of C1 and C2 are listed in the table below.

Oscillation Mode	Crystal Frequency (Hz)	C1, C2 (pF)
E_HXT	16M	5 ~ 10
	10M	5 ~ 30
	8M	5 ~ 20
E_XT	4M	5 ~ 30
	1M	5 ~ 30
	455K	10 ~ 100
E_LXT	32768	10 ~ 30

Table 34 Recommended C1 and C2 Value for Different Kinds of Crystal Oscillation

For 20MHZ resonator in 2 clock CPU cycle mode, an 18pF C2 capacitor is a must.

To get precise and stable 32.768k frequency, choosing the right C1 and C2 value is important. You need to match the C1 / C2 capacitance to the specific crystal you chose. Every crystal datasheet lists something called the Load Capacitance (CL), C1 and C2 value is chosen with the following formula:

C1=C2=2*CL

Where Cbt is the AT8BE62D crystal pad built-in capacitance, which is about 5pF. For example, for crystal CL=12.5P, C1=C2=20pF is recommended.

Please reference “Frequency vs. VDD of I_HRC” and “Frequency vs. Temperature of I_HRC” chapters for detail. For application requires ±1%, please also check writer and socket conditions.

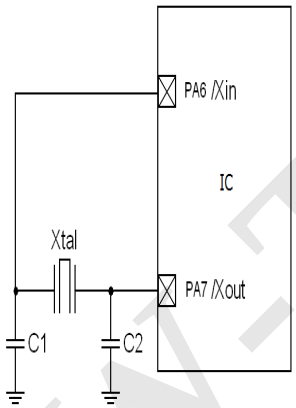


Figure 38 Connection for External Crystal Oscillation

Either F_{HOSC} or F_{LOSC} can be selected as system oscillation F_{OSC} according to the value of register bit SELHOSC (OSCCR[0]). When SELHOSC is 1, F_{HOSC} is selected as F_{OSC}. When SELHOSC is 0, F_{LOSC} is selected as F_{OSC}. Once F_{OSC} is determined, the instruction clock F_{INST} can be $F_{OSC}/2$ or $F_{OSC}/4$ according to value of a configuration word.

3.22 Operating Mode

AT8BE62D provides four kinds of operating mode to tailor all kinds of application and save power consumptions. These operating modes are Normal mode, Slow mode, Standby mode and Halt mode. Normal mode is designated for high-speed operating mode. Slow mode is designated for low-speed mode in order to save power consumption. At Standby mode, AT8BE62D will stop almost all operations except Timer0/Timer1/Timer4/ Timer5/WDT in order to wake-up periodically. At Halt mode, AT8BE62D will sleep until external event or WDT trigger IC to wake-up. The block diagram of four operating modes is described in the following figure.

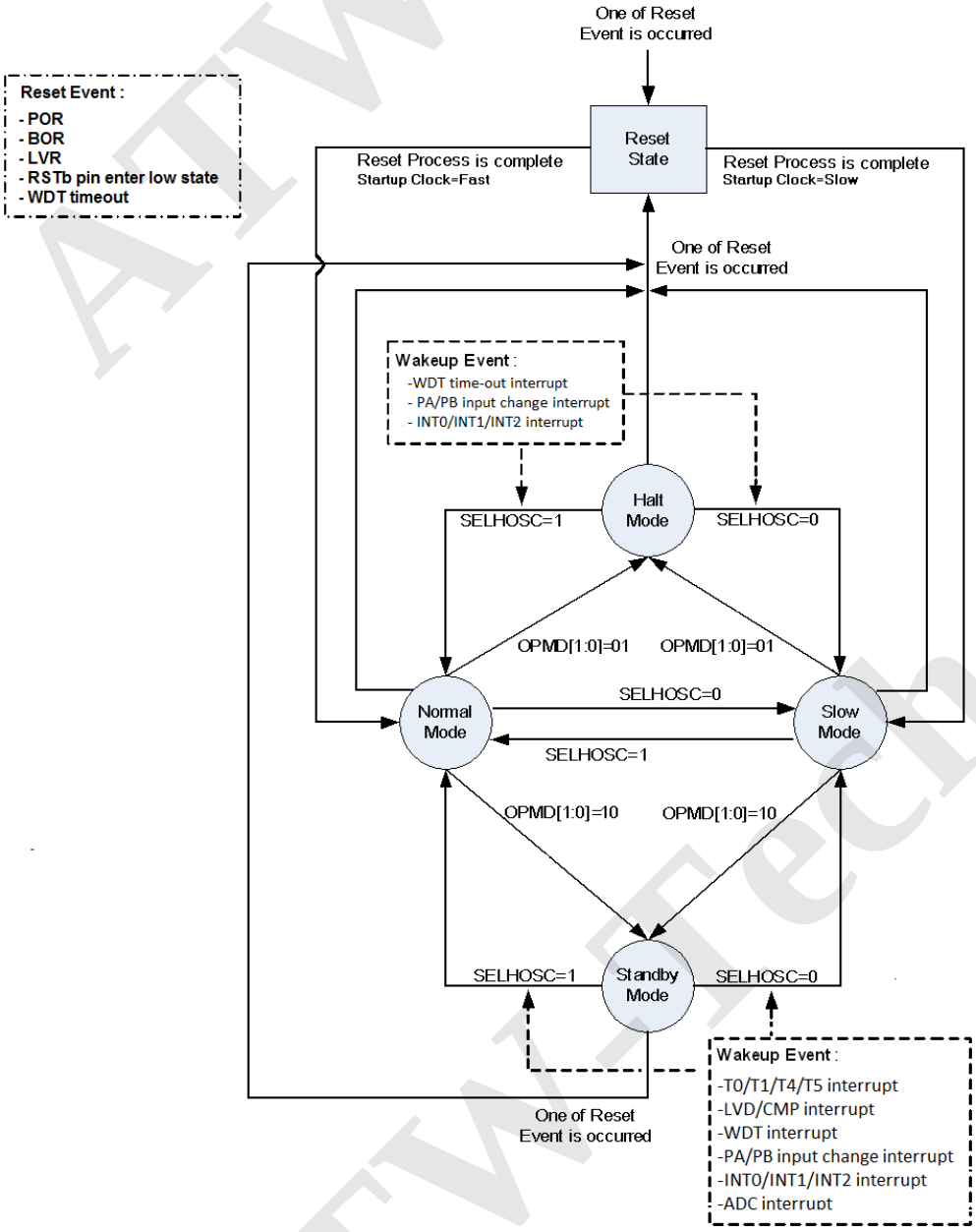


Figure 39 Four Operating Modes

3.22.1 Normal Mode

After any Reset Event is occurred and Reset Process is completed, AT8BE62D will begin to execute program under Normal mode or Slow mode. Which mode is selected after Reset Process is determined by the Startup Clock configuration word. If Startup Clock=fast, AT8BE62D will enter Normal mode, if Startup Clock=Slow, AT8BE62D will enter Slow mode. At Normal mode, F_{HOSC} is selected as system oscillation in order to provide highest performance and its power consumption will be the largest among four operating modes. After power on or any reset trigger is released, AT8BE62D will enter Normal mode after reset process is completed.

- Instruction execution is based on F_{HOSC} and all peripheral modules may be active according to corresponding module enable bit.
- The F_{LOSC} is still active and running.
- IC can switch to Slow mode by writing 0 to register bit SELHOSC (OSCCR[0]).
- IC can switch to Standby mode or Halt mode by programming register bits OPMD[1:0] (OSCCR[3:2]).
- For real time clock applications, the AT8BE62D can run in normal mode, at the same time the low-frequency clock Low Oscillator Frequency connects to Timer0 clock. This is made possible by setting LCKTM0 to 1 and corresponding configuration word Timer0 source setting to 1.

3.22.2 Slow Mode

AT8BE62D will enter Slow mode by writing 0 to register bit SELHOSC. At Slow mode, F_{LOSC} is selected as system oscillation in order to save power consumption but still keep IC running. However, F_{HOSC} will not be disabled automatically by AT8BE62D. Therefore user can write 1 to register bit STPHOSC (OSCCR[1]) in slow mode to reduce power consumption further. But it is noted that it is forbidden to enter slow mode and stop F_{HOSC} at the same time, one must enter slow mode first, then disable F_{HOSC}, or the program may hang on.

- Instruction execution is based on F_{LOSC} and all peripheral modules may be active according to corresponding module enable bit.
- F_{HOSC} can be disabled by writing 1 to register bit STPHOSC.
- IC can switch to Standby mode or Halt mode by programming register bits OPMD[1:0].
- IC can switch to Normal mode by writing 1 to SELHOSC.

3.22.3 Standby Mode

AT8BE62D will enter Standby mode by writing 10b to register bits OPMD[1:0]. At Standby mode, however, F_{HOSC} will not be disabled automatically by AT8BE62D and user has to enter slow mode and write 1 to register bit STPHOSC in order to stop F_{HOSC} oscillation. Most of AT8BE62D peripheral modules are disabled but Timer can be still active if register bit T0EN/T1EN/T4EN/T5EN is set to 1. Therefore AT8BE62D can wake-up after Timer0/Timer1/Timer4/Timer5 is expired. The expiration period is determined by the register TMR0/TMR1[9:0]/TMR4[9:0]/TMR5[9:0], F_{INST} and other configurations for Timer0/Timer1/Timer4/Timer5.

- Instruction execution is stop and some peripheral modules may be active according to corresponding module enable bit.
- F_{HOSC} can be disabled by writing 1 to register bit STPHOSC.
- The F_{LOSC} is still active and running.
- IC can wake-up from Standby mode if any of (a) Timer0/Timer1/Timer4/Timer5 (overflow/underflow) interrupt, (b) WDT timeout interrupt, (c) PA/PB input change interrupt, (d) INT0/1/2 external interrupt is happened, (e) LVD/Comparator Interrupt or (f) ADC end of conversion Interrupt.
- After wake-up from Standby mode, IC will return to Normal mode if SELHOSC=1, IC will return to Slow mode if SELHOSC=0.
- It is not recommended to change oscillator mode (normal to slow / slow to normal) and enter standby mode at the same time.

3.22.4 Halt Mode

AT8BE62D will enter Halt mode by executing instruction SLEEP or writing 01b to register bits OPMD[1:0]. After entering Halt mode, register bit /PD (STATUS[3]) will be clear to 0, register bit /TO (STATUS[4]) will be set to 1 and WDT will be clear but keep running.

At Halt mode, all of peripheral modules are disabled, instruction execution is stop and AT8BE62D can only wake-up by some specific events. Therefore, Halt mode is the most power saving mode provided by AT8BE62D.

- Instruction execution is stop and all peripheral modules are disabled.
- F_{HOSC} and F_{LOSC} are both disabled automatically.
  - IC can wake-up from Halt mode if any of (a) WDT timeout interrupt, (b) PA/PB input change interrupt or (c) INT0/1/2 or external interrupt is happened.
- After wake-up from Halt mode, IC will return to Normal mode if SELHOSC=1, IC will return to Slow mode if SELHOSC=0.

  Note: Users can change STPHOSC and enter Halt mode in the same instruction.
- It is not recommended to change oscillator mode (normal to slow or slow to normal) and enter halt mode at the same time.

3.22.5 Wake-up Stable Time

The wake-up stable time of Halt mode is determined by Configuration word: High Oscillator Frequency or Low Oscillator Frequency. If one of E_HXT, E_XT and E_LXT is selected, the wake-up period would be 512*F_{OSC}. And if no XT mode are selected, 16*Fosc would be set as wake up period. On the other hand, there is no need of wake-up stable time for Standby mode because either F_{HOSC} or F_{LOSC} is still running at Standby mode.

Before AT8BE62D enter Standby mode or Halt mode, user may execute instruction ENI. At this condition, AT8BE62D will branch to address 0x008 in order to execute interrupt service routine after wake-up. If instruction DISI is executed before entering Standby mode or Halt mode, the next instruction will be executed after wake-up.

3.22.6 Summary of Operating Mode

The summary of four operating modes is described in the following table.

Mode	Normal	Slow	Standby	Halt
F_{HOSC}	Enabled	STPHOSC	STPHOSC	Disabled
F_{LOSC}	Enabled	Enabled	Enabled	Disabled
Instruction Execution	Executing	Executing	Stop	Stop
Timer0/1/4/5	TxEN	TxEN	TxEN	Disabled
WDT	Option and WDTEN	Option and WDTEN	Option and WDTEN	Option and WDTEN
Other Modules	Module enable bit	Module enable bit	Module enable bit	All disabled
Wake-up Source	-	-	- Timer0 overflow - Timer1/4/5 underflow - WDT timeout - PA/PB input change - INT0/1/2 - LVD/Comparator interrupt - ADC end-of-convert	- WDT timeout - PA/PB input change - INT0/1/2

Table 35 Summary of Operating Modes

3.23 Reset Process

AT8BE62D will enter Reset State and start Reset Process when one of following Reset Event is occurred:

- Power-On Reset (POR) is occurred when V_{DD} rising is detected.
- Low-Voltage Reset (LVR) is occurred when operating V_{DD} is below pre-defined voltage.
- Pin RSTb is low state.
- WDT timeout reset.

Moreover, value of all registers will be initialized to their initial value or unchanged if its initial value is unknown. The status bits /TO and /PD could be initialized according to which event causes reset. The /TO and /PD value and its associated event is summarized in the table below.

Event	/TO	/PD
POR, LVR	1	1
RSTb reset from non-Halt mode	unchanged	unchanged
RSTb reset from Halt mode	1	1
WDT reset from non-Halt mode	0	1
WDT reset from Halt mode	0	0
SLEEP executed	1	0
CLRWDT executed	1	1

Table 36 Summary of /TO & /PD Value and its Associated Event

After Reset Event is released, AT8BE62D will start Reset Process. It will wait certain amount of period for oscillation stable no matter what kind of oscillator is adopted. This period is called power-up reset time and is determined by three-bit configuration words which can be 140us, 4.5ms, 18ms, 72ms or 288ms. After power-up reset time, AT8BE62D will wait for further oscillator start-up time (OST) before it starts to execute program. OST=1 clock cycle of F_{OSC} if the previous power-up time is 140us, OST=16 clock cycles of F_{OSC} if the previous power-up time is 4.5ms, 18ms, 72ms or 288ms.

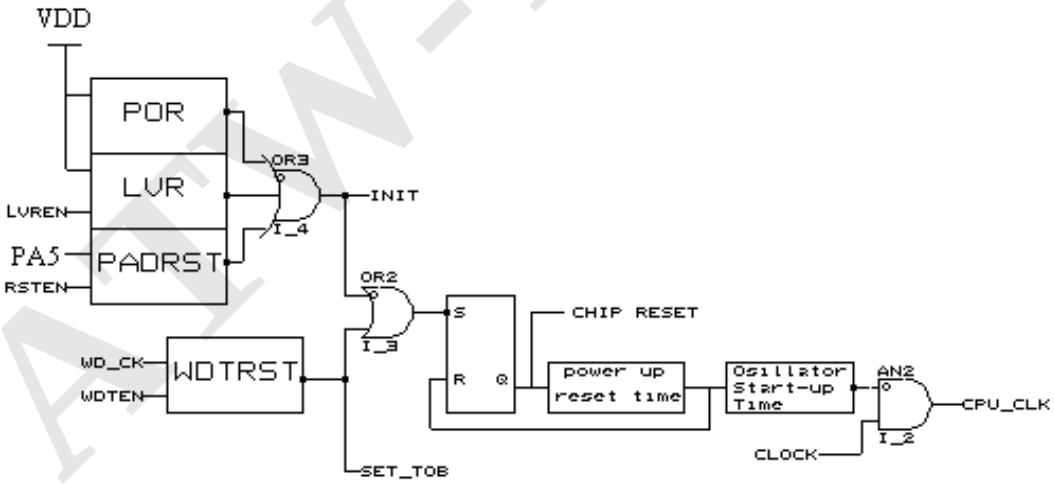


Figure 40 Block diagram of on-chip reset circuit

For slow V_{DD} power-up, it is recommended to use RSTb reset, as the following figure.

- It is recommended the R value should be not greater than 40KΩ.
- The R1 value=100Ω to 1KΩ will prevent high current, ESD or Electrical overstress flowing into reset pin.
  - The diode helps discharge quickly when power down.

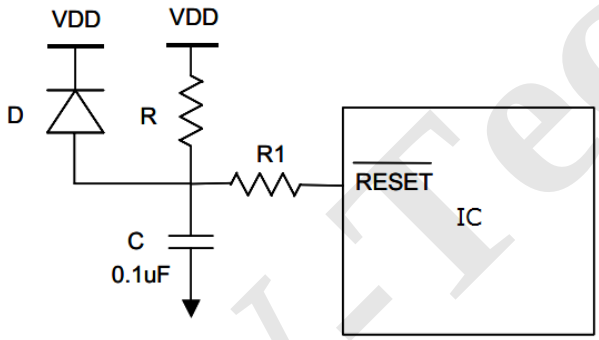


Figure 41 Block Diagram of Reset Application

3.24 On-Chip Debug (OCD)

3.24.1 Function Description

AT8BE62D is embedded in an on-chip debugger(OCD) providing developers with a low cost method for debugging user code. The OCD gives debug capability of complete program flow control with 3 hardware address breakpoints, 1 conditional register break, single step, free running, and non-intrusive commands for memory access. The OCD system does not occupy any locations in the memory map and does not share any on-chip peripherals.

The OCD system uses a two-wire serial interface, SCL and SDA, to establish communication between the target device and the controlling debugger host. SDA is an input/output pin for debug data transfer and SCL is an input pin for synchronization with SDA. The AT8BE62D also use SCL and SDA as control pins to write and read MTP (note that writer needs further pin (VPP) for high-voltage writing of MTP cells).

3.24.2 Limitation of OCD

AT8BE62D is a fully-featured microcontroller that multiplexes several functions on its limited I/O pins. Some device functionality must be sacrificed to provide resources for OCD system. The OCD has the following limitations:

1. The SCL/SDA pins are physically located on the same pin PB5/PB4 or PA4/PA2. Therefore, neither its I/O function nor shared multi-functions can be emulated.
2. System clock cannot be turned off because OCD uses this clock to monitor its internal status: When the system is in halt mode, it is invalid to perform ram/registers accesses because parts of the device may not be clocked. A read access could return garbage or a write access might not succeed. But the following accesses are not effected by the system halt : Read current program address , current PCL, current break condition and current halt status.

4. Instruction Set

AT8BE62D provides 53 powerful instructions for all kinds of applications.

Inst.	OP 1	OP 2	Operation	Cyc.	Flag
Arithmetic Instructions					
ANDAR	R	d	dest = ACC & R	1	Z
IORAR	R	d	dest = ACC | R	1	Z
XORAR	R	d	dest = ACC ⊕ R	1	Z
ANDIA	i		ACC = ACC & i	1	Z
IORIA	i		ACC = ACC | i	1	Z
XORIA	i		ACC = ACC ⊕ i	1	Z
RRR	R	d	Rotate right R	1	C
RLR	R	d	Rotate left R	1	C
BSR	R	bit	Set bit in R	1	-
BCR	R	bit	Clear bit in R	1	-
INCR	R	d	Increase R	1	Z
DECR	R	d	Decrease R	1	Z
COMR	R	d	dest = ~R	1	Z
Conditional Instructions					
BTRSC	R	bit	Test bit in R, skip if clear	1 or 2	-
BTRSS	R	bit	Test bit in R, skip if set	1 or 2	-
INCRSZ	R	d	Increase R, skip if 0	1 or 2	-
DECRSZ	R	d	Decrease R, skip if 0	1 or 2	-
Data Transfer Instructions					
MOVAR	R		Move ACC to R	1	-
MOVR	R	d	Move R	1	Z
MOVIA	i		Move immediate to ACC	1	-
SWAPR	R	d	Swap halves R	1	-
IOST	F		Load ACC to F-page SFR	1	-
IOSTR	F		Move F-page SFR to ACC	1	-
SFUN	S		Load ACC to S-page SFR	1	-
SFUNR	S		Move S-page SFR to ACC	1	-
T0MD			Load ACC to T0MD	1	-
T0MDR			Move T0MD to ACC	1	-
TABLEA			Read ROM	2	-

Inst.	OP 1	OP 2	Operation	Cyc.	Flag
Arithmetic Instructions					
ADDAR	R	d	dest = R + ACC	1	Z, DC, C
SUBAR	R	d	dest = R + (~ACC)	1	Z, DC, C
ADCAR	R	d	dest = R + ACC + C	1	Z, DC, C
SBCAR	R	d	dest = R + (~ACC) + C	1	Z, DC, C
ADDIA	i		ACC = i + ACC	1	Z, DC, C
SUBIA	i		ACC = i + (~ACC)	1	Z, DC, C
ADCIA	i		ACC = i + ACC + C	1	Z, DC, C
SBCIA	i		ACC = i + (~ACC) + C	1	Z, DC, C
DAA			Decimal adjust for ACC	1	C
CMPAR	R		Compare R with ACC	1	Z, C
CLRA			Clear ACC	1	Z
CLRR			Clear R	1	Z
Other Instructions					
NOP			No operation	1	-
SLEEP			Go into Halt mode	1	/TO, /PD
CLRWDT			Clear Watch-Dog Timer	1	/TO, /PD
ENI			Enable interrupt	1	-
DISI			Disable interrupt	1	-
INT			Software Interrupt	3	-
RET			Return from subroutine	2	-
RETIE			Return from interrupt and enable interrupt	2	-
RETIA	i		Return, place immediate in ACC	2	-
CALLA			Call subroutine by ACC	2	-
GOTOA			unconditional branch by ACC	2	-
LCALL	adr		Call subroutine	2	-
LGOTO	adr		unconditional branch	2	-

Table 37 Instruction Set

ACC: Accumulator.

adr: immediate address.

bit: bit address within an 8-bit register R.

C: Carry/Borrow bit

C=1, carry is occurred for addition instruction or borrow is **NOT** occurred for subtraction instruction.

C=0, carry is not occurred for addition instruction or borrow **IS** occurred for subtraction instruction.

d: Destination

If d is “0”, the result is stored in the ACC.

If d is “1”, the result is stored back in register R.

DC: Digital carry flag.

dest: Destination.

F: F-page SFR, F is 0x5 ~ 0xF.

i: 8-bit immediate data.

PC: Program Counter.

PCHBUF: High Byte Buffer of Program Counter.

/PD: Power down flag bit

/PD=1, after power-up or after instruction CLRWDT is executed.

/PD=0, after instruction SLEEP is executed.

Prescaler: Prescaler0 dividing rate.

R: R-page SFR, R is 0x0 ~0x7F.

S: S-page SFR, S is 0x0 ~ 0x1F.

T0MD: T0MD register.

TBHP: The high-Byte at target address in ROM.

TBHD: Store the high-Byte data at target address in ROM.

/TO: Time overflow flag bit

/TO=1, after power-up or after instruction CLRWDT or SLEEP is executed.

/TO=0, WDT timeout is occurred.

WDT: Watchdog Timer Counter.

Z: Zero flag

.

ADCAR	Add ACC and R with Carry
Syntax:	ADCAR R, d
Operand:	0 ≤ R ≤127 d = 0, 1.
Operation:	R + ACC + C dest
Status affected:	Z, DC, C
Description:	Add the contents of ACC and register R with Carry. If d is 0, the result is stored in ACC. If d is 1, the result is stored back to R.
Cycle	1
Example:	ADCAR R, d before executing instruction: ACC=0x12, R=0x34, C=1, d=1, after executing instruction: R=0x47, ACC=0x12, C=0.

ADDAR	Add ACC and R
Syntax:	ADDAR R, d
Operand:	0 ≤ R ≤127 d = 0, 1.
Operation:	ACC + R → dest
Status affected:	Z, DC, C
Description:	Add the contents of ACC and R. If d is 0, the result is stored in ACC. If d is 1, the result is stored back to R.
Cycle:	1
Example:	ADDAR R, d before executing instruction: ACC=0x12, R=0x34,C=1, d=1, after executing instruction: R=0x46, ACC=0x12, C=0.

ADCIA	Add ACC and Immediate with Carry
Syntax:	ADCIA i
Operand:	0 ≤ i ≤ 255
Operation:	ACC + i + C ACC
Status affected:	Z, DC, C
Description:	Add the contents of ACC and the 8-bit immediate data i with Carry. The result is placed in ACC.
Cycle:	1
Example:	ADCIA i before executing instruction: ACC=0x12, i=0x34, C=1, after executing instruction: ACC=0x47, C=0.

ADDIA	Add ACC and Immediate
Syntax:	ADDIA i
Operand:	0 ≤ i ≤ 255
Operation:	ACC + i ACC
Status affected:	Z, DC, C
Description:	Add the contents of ACC with the 8-bit immediate data i. The result is placed in ACC.
Cycle:	1
Example:	ADDIA i before executing instruction: ACC=0x12, i=0x34, C=1, after executing instruction: ACC=0x46, C=0.

ANDAR	AND ACC and R
Syntax:	ANDAR R, d
Operand:	0 ≤ R ≤ 127 d = 0, 1.
Operation:	ACC & R → dest
Status affected:	Z
Description:	The content of ACC is AND'ed with R. If d is 0, the result is stored in ACC. If d is 1, the result is stored back to R.
Cycle:	1
Example:	ANDAR R, d before executing instruction: ACC=0x5A, R=0xAF, d=1. after executing instruction: R=0x0A, ACC=0x5A, Z=0.

ANDIA	AND Immediate with ACC
Syntax:	ANDIA i
Operand:	0 ≤ i ≤ 255
Operation:	ACC & i → ACC
Status affected:	Z
Description:	The content of ACC register is AND'ed with the 8-bit immediate data i. The result is placed in ACC.
Cycle:	1
Example:	ANDIA i before executing instruction: ACC=0x5A, i=0xAF, after executing instruction: ACC=0x0A, Z=0.

BCR	Clear Bit in R
Syntax:	BCR R, bit
Operand:	0 ≤ R ≤ 127 0 ≤ bit ≤ 7
Operation:	0 → R[bit]
Status affected:	--
Description:	Clear the bitth position in R.
Cycle:	1
Example:	BCR R, B2 before executing instruction: R=0x5A, B2=0x3, after executing instruction: R=0x52.

BSR	Set Bit in R
Syntax:	BSR R, bit
Operand:	0 ≤ R ≤ 127 0 ≤ bit ≤ 7
Operation:	1 → R[bit]
Status affected:	--
Description:	Set the bitth position in R.
Cycle:	1
Example:	BSR R, B2 before executing instruction: R=0x5A, B2=0x2, after executing instruction: R=0x5E.

BTRSC	Test Bit in R and Skip if Clear
Syntax:	BTRSC R, bit
Operand:	0 ≤ R ≤ 127 0 ≤ bit ≤ 7
Operation:	Skip next instruction, if R[bit] = 0.
Status affected:	--
Description:	If R[bit] = 0, the next instruction which is already fetched is discarded and a NOP is executed instead. Therefore it makes this instruction a two-cycle instruction.
Cycle:	1 or 2(skip)
Example:	BTRSC R, B2 Instruction1 Instruction2 before executing instruction: R=0x5A, B2=0x2, after executing instruction: because R[B2]=0, instruction1 will not be executed, the program will start execute instruction from instruction2.

BTRSS	Test Bit in R and Skip if Set
Syntax:	BTRSS R, bit
Operand:	0 ≤ R ≤ 127 0 ≤ bit ≤ 7
Operation:	Skip next instruction, if R[bit] = 1.
Status affected:	--
Description:	If R[bit] = 1, the next instruction which is already fetched is discarded and a NOP is executed instead. Therefore it makes this instruction a two-cycle instruction.
Cycle:	1 or 2(skip)
Example:	BTRSS R, B2 Instruction2 Instruction3 before executing instruction: R=0x5A, B2=0x3, after executing instruction: because R[B2]=1, instruction2 will not be executed, the program will start execute instruction from instruction3.

CALLA	Call Subroutine
Syntax:	CALLA
Operand:	--
Operation:	PC + 1 → Top of Stack {TBHP, ACC} → PC
Status affected:	--
Description:	The return address (PC + 1) is pushed onto top of Stack. The contents of TBHP[2:0] is loaded into PC[10:8] and ACC is loaded into PC[7:0].
Cycle:	2
Example:	CALLA before executing instruction: TBHP=0x02, ACC=0x34. PC=A0. Stack pointer=1. after executing instruction: PC=0x234, Stack[1]=A0+1, Stack pointer=2

CLRA	Clear ACC
Syntax:	CLRA
Operand:	--
Operation:	00h → ACC 1 → Z
Status affected:	Z
Description:	ACC is clear and Z is set to 1.
Cycle:	1
Example:	CLRA before executing instruction: ACC=0x55, Z=0. after executing instruction: ACC=0x00, Z=1.

CLRR	Clear R
Syntax:	CLRR R
Operand:	0 ≤ R ≤ 127
Operation:	00h → R 1 → Z
Status affected:	Z
Description:	The content of R is clear and Z is set to 1.
Cycle:	1
Example:	CLRR R before executing instruction: R=0x55, Z=0. after executing instruction: R=0x00, Z=1.

COMR	Complement R
Syntax:	COMR R, d
Operand:	0 ≤ R ≤ 127 d = 0, 1.
Operation:	~R → dest
Status affected:	Z
Description:	The content of R is complemented. If d is 0, the result is stored in ACC. If d is 1, the result is stored back to R.
Cycle:	1
Example:	COMR, d before executing instruction: R=0xA6, d=1, Z=0. after executing instruction: R=0x59, Z=0.

CLRWDT	Clear Watch-Dog Timer
Syntax:	CLRWDT
Operand:	--
Operation:	00h → WDT, 00h → WDT prescaler 1 → /TO 1 → /PD
Status affected:	/TO, /PD
Description:	Executing CLRWDT will reset WDT, Prescaler0 if it is assigned to WDT. Moreover, status bits /TO and /PD will be set to 1.
Cycle:	1
Example:	CLRWDT before executing instruction: /TO=0 after executing instruction: /TO=1

CMPAR	Compare ACC and R
Syntax:	CMPAR R
Operand:	0 ≤ R ≤ 127
Operation:	R - ACC → (No restore)
Status affected:	Z, C
Description:	Compare ACC and R by subtracting ACC from R with 2's complement representation. The content of ACC and R is not changed.
Cycle:	1
Example:	CMPAR R before executing instruction: R=0x34, ACC=12, Z=0, C=0. after executing instruction: R=0x34, ACC=12, Z=0, C=1.

DAA	Convert ACC Data Format from Hexadecimal to Decimal
Syntax:	DAA
Operand:	--
Operation:	ACC(hex) → ACC(dec)
Status affected:	C
Description:	Convert ACC data format from hexadecimal to decimal after addition operation and restore result to ACC. DAA instruction must be placed immediately after addition operation if decimal format is required. Please note that interrupt should be disabled before addition instruction and enabled after DAA instruction to avoid unexpected result.
Cycle:	1
Example:	DISI ADDAR R,d DAA ENI before executing instruction: ACC=0x28, R=0x25, d=0. after executing instruction: ACC=0x53, C=0.

DECRSZ	Decrease R, Skip if 0
Syntax:	DECRSZ R, d
Operand:	0 ≤ R ≤ 127 d = 0, 1.
Operation:	R - 1 → dest, Skip if result = 0
Status affected:	--
Description:	Decrease R first. If d is 0, the result is stored in ACC. If d is 1, the result is stored back to R. If result is 0, the next instruction which is already fetched is discarded and a NOP is executed instead. Therefore it makes this instruction a two-cycle instruction.
Cycle:	1 or 2(skip)
Example:	DECRSZ R, d instruction2 instruction3 before executing instruction: R=0x1, d=1, Z=0. after executing instruction: R=0x0, Z=1, and instruction will skip instruction2 execution because the operation result is zero.

DECR	Decrease R
Syntax:	DECR R, d
Operand:	0 ≤ R ≤ 127 d = 0, 1.
Operation:	R - 1 → dest
Status affected:	Z
Description:	Decrease R. If d is 0, the result is stored in ACC. If d is 1, the result is stored back to R.
Cycle:	1
Example:	DECR R, d before executing instruction: R=0x01, d=1, Z=0. after executing instruction: R=0x00, Z=1.

DISI	Disable Interrupt Globally
Syntax:	DISI
Operand:	--
Operation:	Disable Interrupt, 0 → GIE
Status affected:	--
Description:	GIE is clear to 0 in order to disable all interrupt requests.
Cycle:	1
Example:	DISI before executing instruction: GIE=1, After executing instruction: GIE=0.

ENI	Enable Interrupt Globally
Syntax:	ENI
Operand:	--
Operation:	Enable Interrupt, 1 → GIE
Status affected:	--
Description:	GIE is set to 1 in order to enable all interrupt requests.
Cycle:	1
Example:	ENI before executing instruction: GIE=0, After executing instruction: GIE=1.

INCR	Increase R
Syntax:	INCR R, d
Operand:	0 ≤ R ≤ 127 d = 0, 1.
Operation:	R + 1 → dest.
Status affected:	Z
Description:	Increase R. If d is 0, the result is stored in ACC. If d is 1, the result is stored back to R.
Cycle:	1
Example:	INCR R, d before executing instruction: R=0xFF, d=1, Z=0. after executing instruction: R=0x00, Z=1.

GOTOA	Unconditional Branch
Syntax:	GOTOA
Operand:	--
Operation:	{TBHP, ACC} → PC
Status affected:	--
Description:	GOTOA is an unconditional branch instruction. The content of TBHP[2:0] is loaded into PC[10:8] and ACC is loaded into PC[7:0].
Cycle:	2
Example:	GOTOA before executing instruction: PC=A0. TBHP=0x02, ACC=0x34. after executing instruction: PC=0x234

INCRSZ	Increase R, Skip if 0
Syntax:	INCRSZ R, d
Operand:	0 ≤ R ≤ 127 d = 0, 1.
Operation:	R + 1 → dest, Skip if result = 0
Status affected:	--
Description:	Increase R first. If d is 0, the result is stored in ACC. If d is 1, the result is stored back to R. If result is 0, the next instruction which is already fetched is discarded and a NOP is executed instead. Therefore it makes this instruction a two-cycle instruction.
Cycle:	1 or 2(skip)
Example:	INCRSZ R, d instruction2, instruction3. before executing instruction: R=0xFF, d=1, Z=0. after executing instruction: R=0x00, Z=1. And the program will skip instruction2 execution because the operation result is zero.

INT	Software Interrupt
Syntax:	INT
Operand:	--
Operation:	PC + 1 → Top of Stack, 001h → PC
Status affected:	--
Description:	Software interrupt. First, return address (PC + 1) is pushed onto the Stack. The address 0x001 is loaded into PC[10:0].
Cycle:	3
Example:	INT before executing instruction: PC=address of INT code after executing instruction: PC=0x01

IORIA	OR Immediate with ACC
Syntax:	IORIA i
Operand:	0 ≤ i ≤ 255
Operation:	ACC | i → ACC
Status affected:	Z
Description:	OR ACC with 8-bit immediate data i. The result is stored in ACC.
Cycle:	1
Example:	IORIA i before executing instruction: i=0x50, ACC=0xAA, Z=0. after executing instruction: ACC=0xFA, Z=0.

IORAR	OR ACC with R
Syntax:	IORAR R, d
Operand:	0 ≤ R ≤ 127 d = 0, 1.
Operation:	ACC | R → dest
Status affected:	Z
Description:	OR ACC with R. If d is 0, the result is stored in ACC. If d is 1, the result is stored back to R.
Cycle:	1
Example:	IORAR R, d before executing instruction: R=0x50, ACC=0xAA, d=1, Z=0. after executing instruction: R=0xFA, ACC=0xAA, Z=0.

IOST	Load F-page SFR from ACC
Syntax:	IOST F
Operand:	5 ≤ F ≤ 15
Operation:	ACC → F-page SFR
Status affected:	--
Description:	F-page SFR F is loaded by content of ACC.
Cycle:	1
Example:	IOST F before executing instruction: F=0x55, ACC=0xAA. after executing instruction: F=0xAA, ACC=0xAA.

IOSTR	Move F-page SFR to ACC
Syntax:	IOSTR F
Operand:	5 ≤ F ≤ 15
Operation:	F-page SFR → ACC
Status affected:	--
Description:	Move F-page SFR F to ACC.
Cycle:	1
Example:	IOSTR F before executing instruction: F=0x55, ACC=0xAA. after executing instruction: F=0x55, ACC=0x55.

LGOTO	Unconditional Branch
Syntax:	LGOTO adr
Operand:	0 ≤ adr ≤ 2047
Operation:	adr → PC[10:0].
Status affected:	--
Description:	LGOTO is an unconditional branch instruction. The 11-bit immediate address adr is loaded into PC[10:0].
Cycle:	2
Example:	LGOTO Level before executing instruction: PC=A0. after executing instruction: PC=address of Level.

LCALL	Call Subroutine
Syntax:	LCALL adr
Operand:	0 ≤ adr ≤ 2047
Operation:	PC + 1 → Top of Stack, adr → PC[10:0]
Status affected:	--
Description:	The return address (PC + 1) is pushed onto top of Stack. The 11-bit immediate address adr is loaded into PC[10:0].
Cycle:	2
Example:	LCALL SUB before executing instruction: PC=A0. Stack level=1 after executing instruction: PC=address of SUB, Stack[1]= A0+1, Stack pointer =2.

MOVAR	Move ACC to R
Syntax:	MOVAR R
Operand:	0 ≤ R ≤ 127
Operation:	ACC → R
Status affected:	--
Description:	Move content of ACC to R.
Cycle:	1
Example:	MOVAR R before executing instruction: R=0x55, ACC=0xAA. after executing instruction: R=0xAA, ACC=0xAA.

MOVIA	Move Immediate to ACC
Syntax:	MOVIA i
Operand:	$0 \leq i \leq 255$
Operation:	$i \rightarrow ACC$
Status affected:	--
Description:	The content of ACC is loaded with 8-bit immediate data i.
Cycle:	1
Example:	MOVIA i before executing instruction: i=0x55, ACC=0xAA. after executing instruction: ACC=0x55.

NOP	No Operation
Syntax:	NOP
Operand:	--
Operation:	No operation.
Status affected:	--
Description:	No operation.
Cycle:	1
Example:	NOP before executing instruction: PC=A0 after executing instruction: PC=A0+1

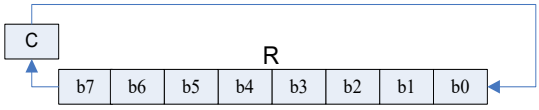
MOVR	Move R to ACC or R
Syntax:	MOVR R, d
Operand:	$0 \leq R \leq 127$ d = 0, 1.
Operation:	$R \rightarrow dest$
Status affected:	Z
Description:	The content of R is move to destination. If d is 0, destination is ACC. If d is 1, destination is R and it can be used to check whether R is zero according to status flag Z after execution.
Cycle:	1
Example:	MOVR R, d before executing instruction: R=0x0, ACC=0xAA, Z=0, d=0. after executing instruction: R=0x0, ACC=0x00, Z=1.

RETIE	Return from Interrupt and Enable Interrupt Globally
Syntax:	RETIE
Operand:	--
Operation:	Top of Stack $\rightarrow$ PC $1 \rightarrow GIE$
Status affected:	--
Description:	The PC is loaded from top of Stack as return address and GIE is set to 1.
Cycle:	2
Example:	RETIE before executing instruction: GIE=0, Stack level=2. after executing instruction: GIE=1, PC=Stack[2], Stack pointer=1.

RETIA	Return with Data in ACC
Syntax:	RETIA i
Operand:	0 ≤ i ≤ 255
Operation:	i → ACC, Top of Stack → PC
Status affected:	--
Description:	ACC is loaded with 8-bit immediate data i and PC is loaded from top of Stack as return address.
Cycle:	2
Example:	RETIA i before executing instruction: Stack pointer =2. i=0x55, ACC=0xAA. after executing instruction: PC=Stack[2], Stack pointer =1. ACC=0x55.

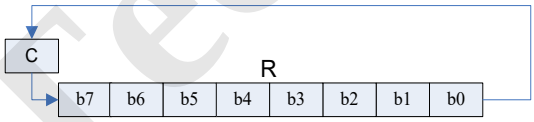
RET	Return from Subroutine
Syntax:	RET
Operand:	--
Operation:	Top of Stack → PC
Status affected:	--
Description:	PC is loaded from top of Stack as return address.
Cycle:	2
Example:	RET before executing instruction: Stack level=2. after executing instruction: PC=Stack[2], Stack level=1.

RLR	Rotate Left R Through Carry
Syntax:	RLR R, d
Operand:	0 ≤ R ≤ 127 d = 0, 1.
Operation:	C → dest[0], R[7] → C, R[6:0] → dest[7:1]



Status affected:	C
Description:	The content of R is rotated one bit to the left through flag Carry. If d is 0, the result is placed in ACC. If d is 1, the result is stored back to R.
Cycle:	1
Example:	RLR R, d before executing instruction: R=0xA5, d=1, C=0. after executing instruction: R=0x4A, C=1.

RRR	Rotate Right R Through Carry
Syntax:	RRR R, d
Operand:	0 ≤ R ≤ 127 d = 0, 1.
Operation:	C → dest[7], R[7:1] → dest[6:0], R[0] → C



Status affected:	C
Description:	The content of R is rotated one bit to the right through flag Carry. If d is 0, the result is placed in ACC. If d is 1, the result is stored back to R.
Cycle:	1
Example:	RRR R, d before executing instruction: R=0xA5, d=1, C=0. after executing instruction: R=0x52, C=1.

SBCAR	Subtract ACC and Carry from R
Syntax:	SBCAR R, d
Operand:	0 ≤ R ≤ 127 d = 0, 1.
Operation:	R + (~ACC) + C → dest
Status affected:	Z, DC, C
Description:	Subtract ACC and Carry from R with 2's complement representation. If d is 0, the result is placed in ACC. If d is 1, the result is stored back to R.
Cycle:	1
Example:	SBCAR R, d (a) before executing instruction: R=0x05, ACC=0x06, d=1, C=0, after executing instruction: R=0xFE, C=0. (-2) (b) before executing instruction: R=0x05, ACC=0x06, d=1, C=1, after executing instruction: R=0xFF, C=0. (-1) (c) before executing instruction: R=0x06, ACC=0x05, d=1, C=0, after executing instruction: R=0x00, C=1. (-0), Z=1. (d) before executing instruction: R=0x06, ACC=0x05, d=1, C=1, after executing instruction: R=0x1, C=1. (+1)

SBCIA	Subtract ACC and Carry from Immediate
Syntax:	SBCIA i
Operand:	0 ≤ i ≤ 255
Operation:	i + (~ACC) + C → dest
Status affected:	Z, DC, C
Description:	Subtract ACC and Carry from 8-bit immediate data i with 2's complement representation. The result is placed in ACC.
Cycle:	1
Example:	SBCIA i (a) before executing instruction: i=0x05, ACC=0x06, C=0, after executing instruction: ACC=0xFE, C=0. (-2) (b) before executing instruction: i=0x05, ACC=0x06, C=1, after executing instruction: ACC=0xFF, C=0. (-1) (c) before executing instruction: i=0x06, ACC=0x05, C=0, after executing instruction: ACC=0x00, C=1. (-0), Z=1. (d) before executing instruction: i=0x06, ACC=0x05, C=1, after executing instruction: ACC=0x1, C=1. (+1)

SFUN	Load S-page SFR from ACC
Syntax:	SFUN S
Operand:	0 ≤ S ≤ 31
Operation:	ACC → S-page SFR
Status affected:	--
Description:	S-page SFR S is loaded by content of ACC.
Cycle:	1
Example:	SFUN S before executing instruction: S=0x55, ACC=0xAA. after executing instruction: S=0xAA, ACC=0xAA.

SFUNR	Move S-page SFR from ACC
Syntax:	SFUNR S
Operand:	0 ≤ S ≤ 31
Operation:	S-page SFR → ACC
Status affected:	--
Description:	Move S-page SFR S to ACC.
Cycle:	1
Example:	SFUNR S before executing instruction: S=0x55, ACC=0xAA. after executing instruction: S=0x55, ACC=0x55.

SLEEP	Enter Halt Mode
Syntax:	SLEEP
Operand:	--
Operation:	00h → WDT, 00h → WDT prescaler 1 → /TO 0 → /PD
Status affected:	/TO, /PD
Description:	WDT and Prescaler0 are clear to 0. /TO is set to 1 and /PD is clear to 0. IC enter Halt mode.
Cycle:	1
Example:	SLEEP before executing instruction: /PD=1, /TO=0. after executing instruction: /PD=0, /TO=1.

SUBAR	Subtract ACC from R
Syntax:	SUBAR R, d
Operand:	0 ≤ R ≤ 127 d = 0, 1.
Operation:	R – ACC → dest
Status affected:	Z, DC, C
Description:	Subtract ACC from R with 2's complement representation. If d is 0, the result is placed in ACC. If d is 1, the result is stored back to R.
Cycle:	1
Example:	SBCAR R, d (a) before executing instruction: R=0x05, ACC=0x06, d=1, after executing instruction: R=0xFF, C=0. (-1) (b) before executing instruction: R=0x06, ACC=0x05, d=1, after executing instruction: R=0x01, C=1. (+1)

SUBIA	Subtract ACC from Immediate
Syntax:	SUBIA i
Operand:	0 ≤ i ≤ 255
Operation:	i – ACC → ACC
Status affected:	Z, DC, C
Description:	Subtract ACC from 8-bit immediate data i with 2's complement representation. The result is placed in ACC.
Cycle:	1
Example:	SUBIA i (a) before executing instruction: i=0x05, ACC=0x06. after executing instruction: ACC=0xFF, C=0. (-1) (b) before executing instruction: i=0x06, ACC=0x05, d=1, after executing instruction: ACC=0x01, C=1. (+1)

SWAPR	Swap High/Low Nibble in R
Syntax:	SWAPR R, d
Operand:	0 ≤ R ≤ 127 d = 0, 1.
Operation:	R[3:0] → dest[7:4]. R[7:4] → dest[3:0]
Status affected:	--
Description:	The high nibble and low nibble of R is exchanged. If d is 0, the result is placed in ACC. If d is 1, the result is stored back to R.
Cycle:	1
Example:	SWAPR R, d before executing instruction: R=0xA5, d=1. after executing instruction: R=0x5A.

T0MD	Load ACC to T0MD
Syntax:	T0MD
Operand:	--
Operation:	ACC → T0MD
Status affected:	--
Description:	The content of T0MD is loaded by ACC.
Cycle:	1
Example:	T0MD before executing instruction: T0MD=0x55, ACC=0xAA. after executing instruction: T0MD=0xAA.

TABLEA	Read ROM data
Syntax:	TABLEA
Operand:	--
Operation:	ROM data{ TBHP, ACC } [7:0] → ACC ROM data{TBHP, ACC} [13:8] → TBHD.
Status affected:	--
Description:	The 8 least significant bits of ROM pointed by {TBHP[2:0], ACC} is placed to ACC. The 6 most significant bits of ROM pointed by {TBHP[2:0], ACC} is placed to TBHD[5:0].
Cycle:	2
Example:	TABLEA before executing instruction: TBHP=0x02, CC=0x34. TBHD=0x01. ROM data[0x234]= 0x35AA after executing instruction: TBHD=0x35, ACC=0xAA.

T0MDR	Move T0MD to ACC
Syntax:	T0MDR
Operand:	--
Operation:	T0MD → ACC
Status affected:	--
Description:	Move the content of T0MD to ACC.
Cycle:	1
Example:	T0MDR before executing instruction T0MD=0x55, ACC=0xAA. after executing instruction ACC=0x55.

XORAR	Exclusive-OR ACC with R
Syntax:	XORAR R, d
Operand:	0 ≤ R ≤ 127 d = 0, 1.
Operation:	ACC ⊕ R → dest
Status affected:	Z
Description:	Exclusive-OR ACC with R. If d is 0, the result is placed in ACC. If d is 1, the result is stored back to R.
Cycle:	1
Example:	XORAR R, d before executing instruction: R=0xA5, ACC=0xF0, d=1. after executing instruction: R=0x55.

XORIA	Exclusive-OR Immediate with ACC
Syntax:	XORIA i
Operand:	0 ≤ i ≤ 255
Operation:	ACC ⊕ i → ACC
Status affected:	Z
Description:	Exclusive-OR ACC with 8-bit immediate data i. The result is stored in ACC.
Cycle:	1
Example:	XORIA i before executing instruction: i=0xA5, ACC=0xF0. after executing instruction: ACC=0x55.

5. Configuration Words

Item	Name	Options
1	High Oscillator Frequency	1. I_HRC 2. E_HXT 3. E_XT
2	Low Oscillator Frequency	1. I_LRC 2. E_LXT
3	High IRC Frequency	1. 13.6MHz 2. 14.4MHz 3. 16MHz 4. 19.2MHz 5. 20.8MHz
4	Divider	1. div 1 2. div 2 3. div 4 4. div 8 5. div 16
5	High Crystal Oscillator	1. 8MHz > F_{HOSC} > 6MHz 2. 10MHz > F_{HOSC} ≧ 8MHz 3. 12MHz > F_{HOSC} ≧ 10MHz 4. 16MHz > F_{HOSC} ≧ 12MHz 5. 20MHz > F_{HOSC} ≧ 16MHz 6. 20MHz
6	Instruction Clock	1. 2 oscillator period 2. 4 oscillator period
7	WDT	1. Watchdog Enable (Software control) 2. Watchdog Disable (Always disable)
8	WDT Event	1. Watchdog Reset 2. Watchdog Interrupt
9	Timer0 Source	1. EX_CKI0 2. Low Oscillator (I_LRC/E_LXT)
10	PA.5	1. PA.5 is I/O 2. PA.5 is reset
11	PA.7	1. PA.7 is I/O 2. PA.7 is instruction clock output
12	IR Pad	1. PB1 2. PA3
13	Startup Time	1. 140us 2. 4.5ms 3. 18ms 4. 72ms 5. 288ms
14	WDT Time Base	1. 3.5ms 2. 15ms 3. 60ms 4. 250ms
15	Noise Filter (High_EFT)	1. Enable 2. Disable
16	LVR Setting	1. Register Control 2. Register Control + Halt mode Off 3. Always On 4. Operation mode On + Halt mode Off
17	LVR Voltage	1. 1.6V 2. 1.8V 3. 2.0V 4. 2.2V 5. 2.4V 6. 2.7V 7. 3.0V 8. 3.3V 9. 3.6V 10. 4.2V
18	VDD Voltage	1. 3.0V 2. 4.5V 3. 5.0V
19	EEWR time-out	1. Enable 2. Disable
20	EEPL MODE	1. Once 2. Toggle
21	Sink current	1. Small 2. Normal 3. Large
22	Drive current	1. Small 2. Normal
23	Comparator Input pin select	1. Enable 2. Disable
24	Read Output Data	1. I/O Port 2. Register
25	E_LXT Backup Control	1. Auto Off 2. Register Off
26	EX_CKI0 to Inst. Clock	1. Sync 2. Async.
27	Startup Clock	1. Fast (I_HRC/E_HXT/E_XT) 2. Slow (I_LRC/E_LXT)
28	Input High Voltage (VIH)	1. 0.7VDD 2. 0.5VDD
33	Input Low Voltage (VIL)	1. 0.3VDD 2. 0.2VDD
36	Input Voltage (VIH/VIL)	1. Schmitt 2. No Schmitt
37	PWM1 output	1. PB1 2. PB4
38	PWM2 output	1. PA4 2. PB5
39	PWM3 output	1. PA2 2. PA7

Item	Name	Options	
40	INT0 input	1. PB0	2. PB5
41	INT1 input	1. PB1	2. PA3
42	INT2 input	1. NO	2. PA5
43	External clock 1	1. PA1	2. PA2

Table 38 Configuration Words

6. Electrical Characteristics

6.1 Absolute Maximum Rating

Symbol	Parameter	Rated Value	Unit
V_{DD} - V_{SS}	Supply voltage	-0.5 ~ +6.0	V
V_{IN}	Input voltage	V_{SS}-0.3V ~ V_{DD}+0.3	V
T_{OP}	Operating Temperature	-20 ~ +85	°C
T_{ST}	Storage Temperature	-40 ~ +125	°C

6.2 DC Characteristics

(All refer $F_{INST}=F_{HOSC}/4$, F_{HOSC}=16MHz@I_HRC, WDT enabled, ambient temperature T_A=25˚C unless otherwise specified.)

Symbol	Parameter	V_{DD}	Min.	Typ.	Max.	Unit	Condition
V_{DD}	Operating voltage	--	3.3	--	5.5	V	F_{INST}=20.8MHz @ I_HRC/2
			3.0				F_{INST}=19.2MHz @ I_HRC/2
			2.7				F_{INST}=16MHz @ I_HRC/2 & E_HXT/2
			2.7				F_{INST}=14.4MHz @ I_HRC/2
			2.7				F_{INST}=13.6MHz @ I_HRC/2
			2.2				F_{INST}=20.8MHz @ I_HRC/4
			2.2				F_{INST}=19.2MHz @ I_HRC/4
			2.0				F_{INST}=14.4MHz @ I_HRC/4
			2.0				F_{INST}=13.6MHz @ I_HRC/4
			2.0				F_{INST}=16MHz @ I_HRC/4 & E_HXT/4
			2.0				F_{INST}=8MHz @ I_HRC/2 & E_HXT/2
			1.8				F_{INST}=8MHz @ I_HRC/4 & E_HXT/4
							F_{INST}=4MHz @ I_HRC/2 & E_XT /2
			1.6				F_{INST}=4MHz @ I_HRC/4
							F_{INST}=32KHz @ I_LRC/4 & I_LRC/2
							F_{INST}=32KHz @ E_LXT/4 & E_LXT/2
V_{IH}	Input high voltage	5V	4.0	--	--	V	RSTb (0.8 V_{DD})
		3V	2.4	--	--		
		5V	3.5	--	--	V	All other I/O pins, EX_CKI0/1, INT0/1 CMOS option (0.7 V_{DD})
		3V	2.1	--	--		
		5V	2.5	--	--	V	All other I/O pins, EX_CKI0/1 TTL option (0.5 V_{DD})
		3V	1.5	--	--		
		5V	2.0	--	--	V	All other I/O pins, EX_CKI, INT No Schmitt Trigger (0.4V_{DD})
		3V	1.2	--	--		
V_{IL}	Input low voltage	5V	--	--	1.0	V	RSTb (0.2 V_{DD})
		3V	--	--	0.6		
		5V	--	--	1.5	V	All other I/O pins, EX_CKI0/1, INT0/1 CMOS option (0.3 V_{DD})
		3V	--	--	0.9		

Symbol	Parameter	V_{DD}	Min.	Typ.	Max.	Unit	Condition
		5V	--	--	1.0	V	All other I/O pins, EX_CKI0/1 TTL option (0.2 V_{DD})
		3V	--	--	0.6		
		5V	--	--	2.0	V	All other I/O pins, EX_CKI, INT No Schmitt Trigger (0.4V_{DD})
		3V	--	--	1.2		
I_{OH}	Output high current (Small current)	5V	--	1.7	--	mA	V_{OH}=4.0V
		3V	--	0.9	--		V_{OH}=2.0V
	Output high current (Normal current)	5V	--	18	--	mA	V_{OH}=4.0V
		3V	--	10	--		V_{OH}=2.0V
I_{OL}	Output low current (Small current)	5V	--	7.7	--	mA	V_{OL}=1.0V
		3V	--	4.2	--		
	Output low current (Normal current)	5V	--	28	--	mA	V_{OL}=1.0V
		3V	--	16	--		
I_{OL}	Output low current (Large current)	5V	--	40	--	mA	V_{OL}=1.0V
		3V	--	24	--		
	Output low current (Constant current)	5V	--	26	--	mA	V_{OL}=1.0V
		3V	--	24	--		
I_{LIR}	Large IR sink current	5V	--	420	--	mA	V_{OL}=1.0V, Large IR
		3V	--	340	--		
I_{OP}	Operating current	**Normal Mode**					
		5V	--	4.1	--	mA	F_{HOSC}=20.8MHz @ I_HRC/2
		3V	--	2.6	--		
		5V	--	3.1	--	mA	F_{HOSC}=20.8MHz @ I_HRC/4
		3V	--	2.0	--		
		5V	--	3.8	--	mA	F_{HOSC}=19.2MHz @ I_HRC/2
		3V	--	2.5	--		
		5V	--	2.9	--	mA	F_{HOSC}=19.2MHz @ I_HRC/4
		3V	--	1.9	--		
		5V	--	3.3	--	mA	F_{HOSC}=14.4MHz @ I_HRC/2
		3V	--	2.2	--		
		5V	--	2.6	--	mA	F_{HOSC}=14.4MHz @ I_HRC/4
		3V	--	1.8	--		
		5V	--	3.1	--	mA	F_{HOSC}=13.6MHz @ I_HRC/2
		3V	--	2.0	--		
		5V	--	2.5	--	mA	F_{HOSC}=13.6MHz @ I_HRC/4
		3V	--	1.7	--		
		5V	--	3.3	--	mA	F_{HOSC}=16MHz @ I_HRC/2 & E_HXT/2
		3V	--	2.1	--		
		5V	--	2.6	--	mA	F_{HOSC}=16MHz @ I_HRC/4 & E_HXT/4
		3V	--	1.8	--		
		5V	--	2.5	--	mA	F_{HOSC}=8MHz @ I_HRC/2 & E_HXT/2

Symbol	Parameter	V_{DD}	Min.	Typ.	Max.	Unit	Condition
		3V	--	1.7	--		
		5V	--	1.6	--	mA	F_{HOSC}=8MHz @ I_HRC/4 & E_HXT/4
		3V	--	1.1	--		
		5V	--	1.6	--	mA	F_{HOSC}=4MHz @ I_HRC/2 & E_XT/2
		3V	--	1.1	--		
		5V	--	1.2	--	mA	F_{HOSC}=4MHz @ I_HRC/4 & E_XT/4
		3V	--	0.8	--		
		5V	--	0.9	--	mA	F_{HOSC}=1MHz @ I_HRC/2 & E_XT/2
		3V	--	0.5	--		
		5V	--	0.7	--	mA	F_{HOSC}=1MHz @ I_HRC/4 & E_XT/4
		3V	--	0.4	--		
		Slow Mode					
		5V	--	14	--	uA	F_{HOSC} disabled, F_{LOSC}=32KHz @ I_LRC/2
		3V	--	10	--		
		5V	--	19	--	uA	F_{HOSC} disabled, F_{LOSC}=32KHz @ E_LXT/2
		3V	--	12	--		
		5V	--	9	--	uA	F_{HOSC} disabled, F_{LOSC}=32KHz @ I_LRC/4
		3V	--	6	--		
		5V	--	13	--	uA	F_{HOSC} disabled, F_{LOSC}=32KHz @ E_LXT/4
		3V	--	7	--		
I_{STB}	Standby current	5V	--	3.8	--	uA	Standby mode, F_{HOSC} disabled, F_{LOSC}=32KHz @ I_LRC/4
		3V	--	1.6	--		
I_{HALT}	Halt current	5V	--	--	0.5	uA	Halt mode, WDT disabled.
		3V	--	--	0.2		
		5V	--	--	5.0	uA	Halt mode, WDT enabled.
		3V	--	--	2.0		
R_{PH}	Pull-High resistor	5V	--	60	--	KΩ	Pull-High resistor (not include PA5)
		3V	--	120	--		
		5V	--	90	--	KΩ	Pull-High resistor (PA5)
		3V	--	90	--		
R_{PL}	Pull-Low resistor	5V	--	50	--	KΩ	Pull-Low resistor
		3V	--	100	--		

6.3 OSC Characteristics

(Measurement conditions V_{DD} Voltage, T_A Temperature are equal to programming conditions.)

Parameter	Min.	Typ.	Max.	Unit	Condition
I_HRC deviation by socket			±1	%	Socket installed directly on writer.
I_HRC deviation by handler			±3	%	Handler condition with correct setup.
I_LRC deviation by handler			±5	%	

6.4 Comparator / LVD Characteristics

(V_{DD}=5V, V_{SS}=0V, T_A=25°C unless otherwise specified.)

Symbol	Parameter	Min.	Typ.	Max.	Unit	Condition
V_{IVR}	Comparator input voltage range	0	--	5	V	F_{HOSC}=1MHz
T_{ENO}	Comparator enable to output valid	--	20	--	us	F_{HOSC}=1MHz
I_{CO}	Operating current of comparator	--	135	--	uA	F_{HOSC}=1MHz, P2V mode
I_{LVD}	Operating current of LVD	--	150	--	uA	F_{HOSC}=1MHz, LVD=4.3V
E_{LVD}	LVD voltage error	--	3	--	%	F_{HOSC}=1MHz, LVD=4.3V

6.5 ADC Characteristics

(V_{DD}=5V, V_{SS}=0V, T_A=25°C unless otherwise specified.)

Symbol	Parameter	Min.	Typ.	Max.	Unit	Condition
V_{REFH}	VREFH input voltage	2V	--	V_{DD}	V	Ext. reference voltage
V_{REF4}	Int. 4V reference voltage, V_{DD}=5V	3.96	4	4.04	V	
V_{REF3}	Int. 3V reference voltage, V_{DD}=5V	2.97	3	3.03	V	
V_{REF2}	Int. 2V reference voltage, V_{DD}=5V	1.98	2	2.02	V	
V_{REF}	Int. V_{DD} reference voltage, V_{DD}=5V	--	V_{DD}	--	V	
	Internal reference supply voltage	V_{REF}+0.5	--	--	V	Minimum supply voltage
	ADC analog input voltage	0	--	V_{REFH}	V	
$I_{OP(ADC)}$	ADC current consumption	--	0.5	--	mA	
ADCLK	ADC Clock Frequency	32K	--	1M	Hz	
ADCYCLE	ADC Conversion Cycle Time	16	--		1/ADCLK	SHCLK=2 ADC clock
ADC_{sample}	ADC Sampling Rate	--	--	125	K/sec	V_{DD}=5V
DNL	Differential Nonlinearity	±1	--	--	LSB	V_{DD}=5.0V, AVREFH=5V, FADSMP=62.5K
INL	Integral Nonlinearity	±2	--	--	LSB	
NMC	No Missing Code	10	11	12	Bits	

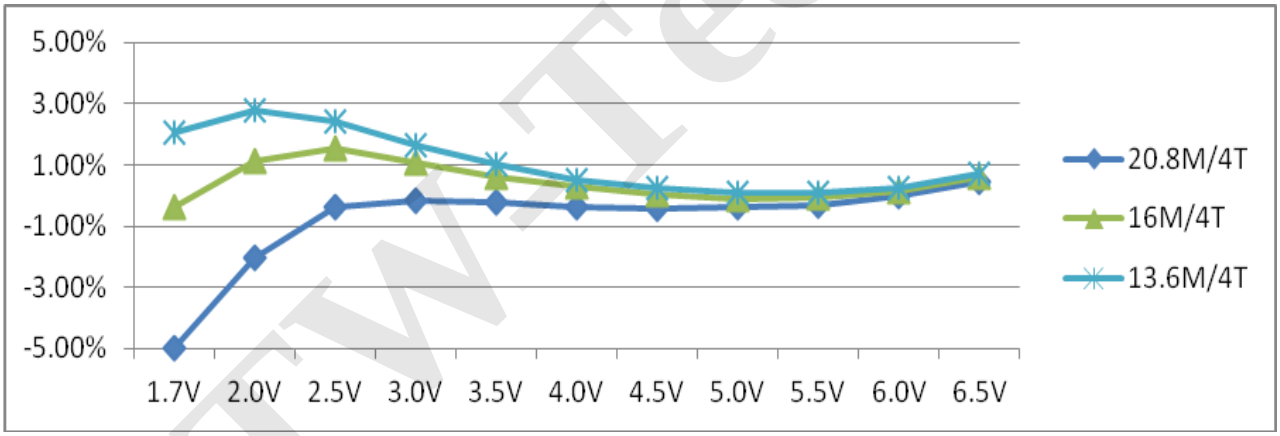
6.6 EEPROM Characteristics

(T_A=25°C unless otherwise specified.)

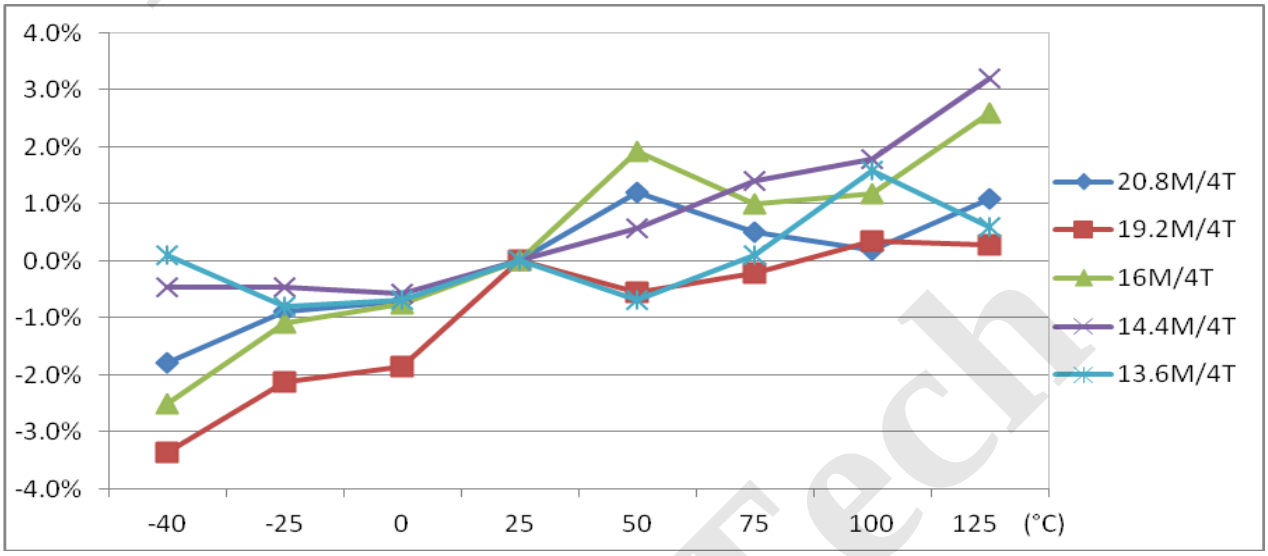
Symbol	Parameter	Min.	Typ.	Max.	Unit	Condition
V_{EEWR}	EEPROM write voltage	2.0	--	5.5	V	T = -20℃ ~ 85℃
I_{EEWR}	EEPROM write current	--	7	--	mA	VDD = 5 V
		--	7	--		VDD = 3 V
T_{EEWR}	EEPROM write time (per byte)	--	--	10	ms	VDD = 2.0V ~ 5.5V
N_{EE}	Endurance of EEPROM	--	40,000	--	cycles	T = 85℃
		--	30,000	--		T = 25℃
		--	10,000	--		T = -20℃

6.7 Characteristic Graph

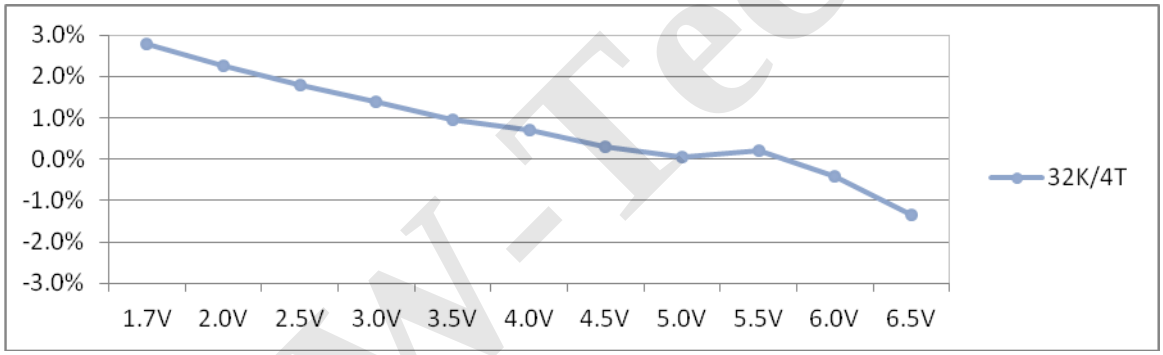
6.7.1 Frequency vs. VDD of I_HRC



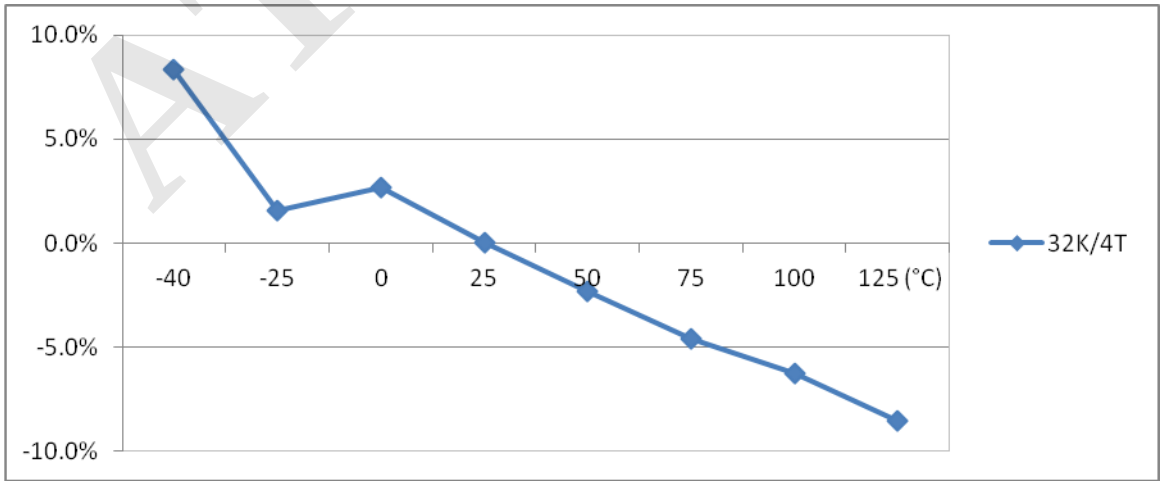
6.7.2 Frequency vs. Temperature of I_HRC



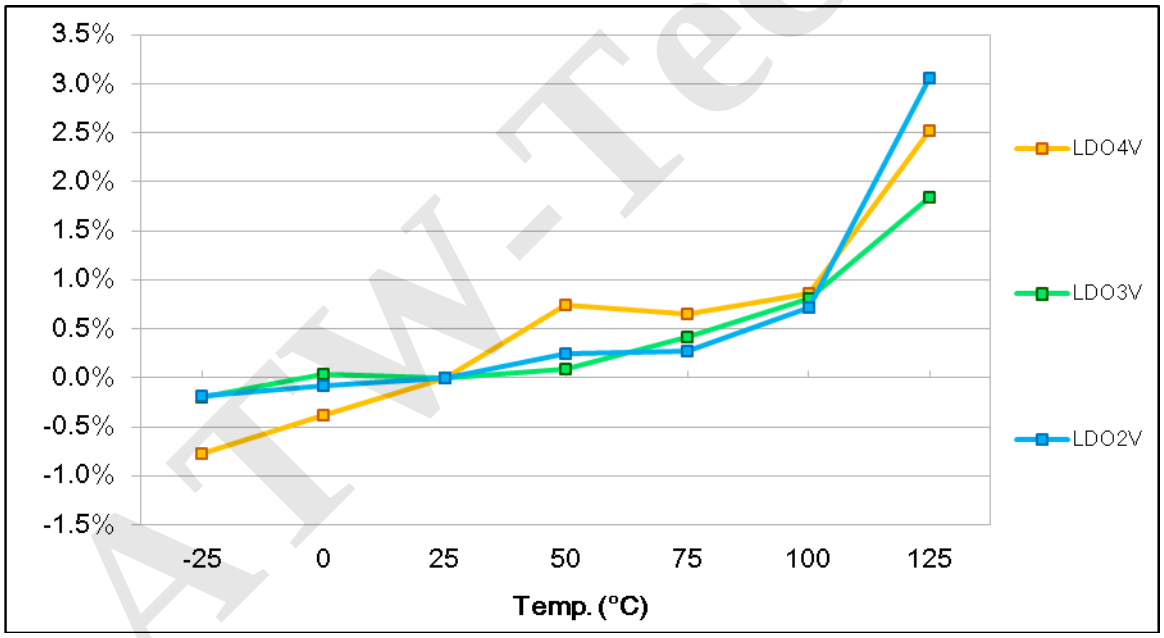
6.7.3 Frequency vs. V_{DD} of I_LRC



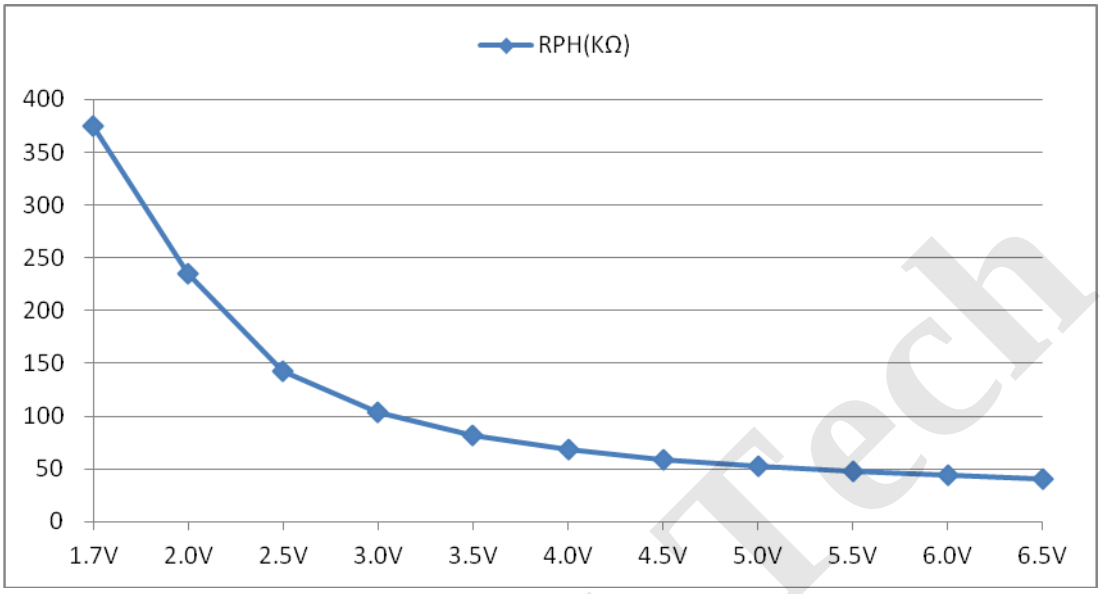
6.7.4 Frequency vs. Temperature of I_LRC



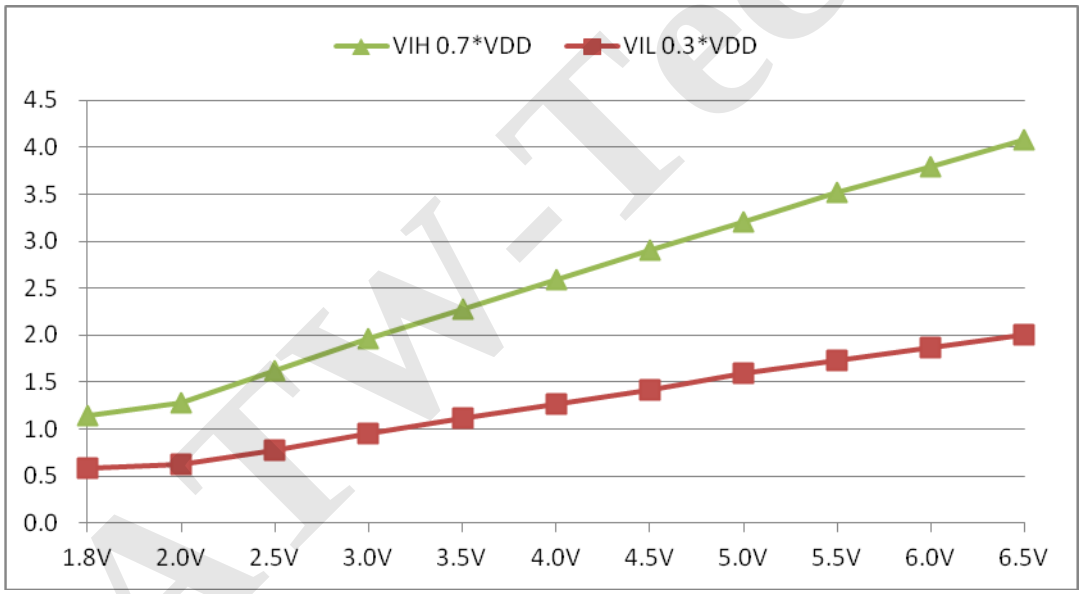
6.7.5 Low Dropout Regulator vs. Temperature

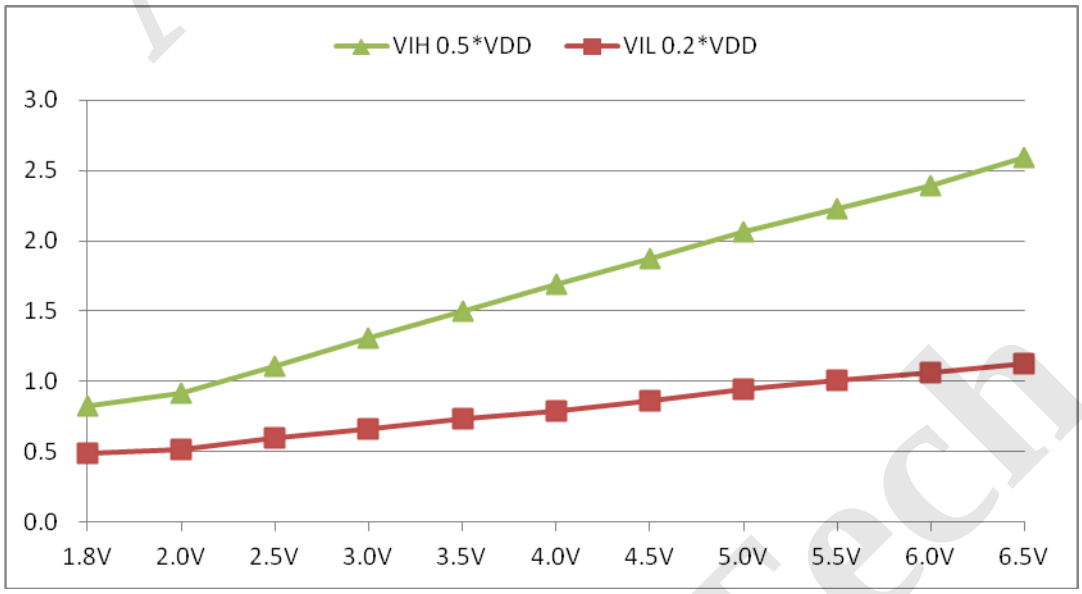


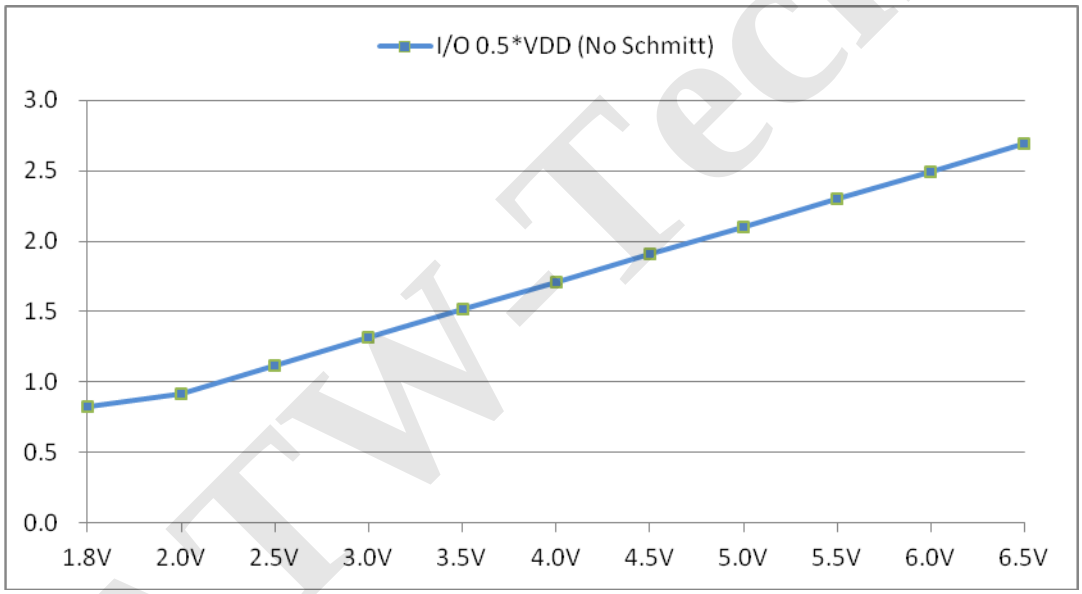
6.7.6 Pull High Resistor vs. VDD

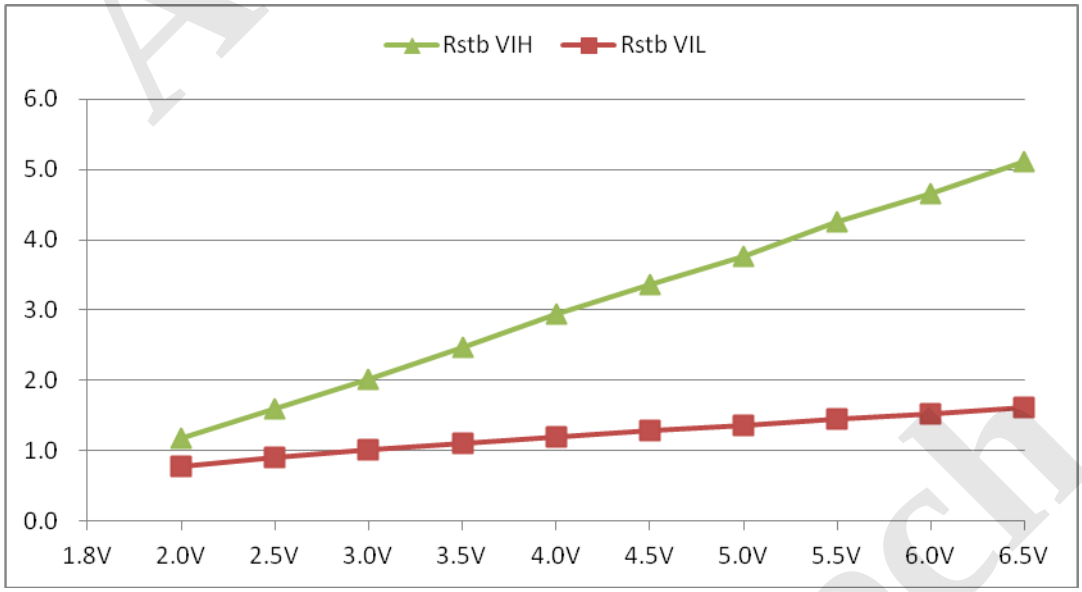


6.7.7 VIH/VIL vs. VDD







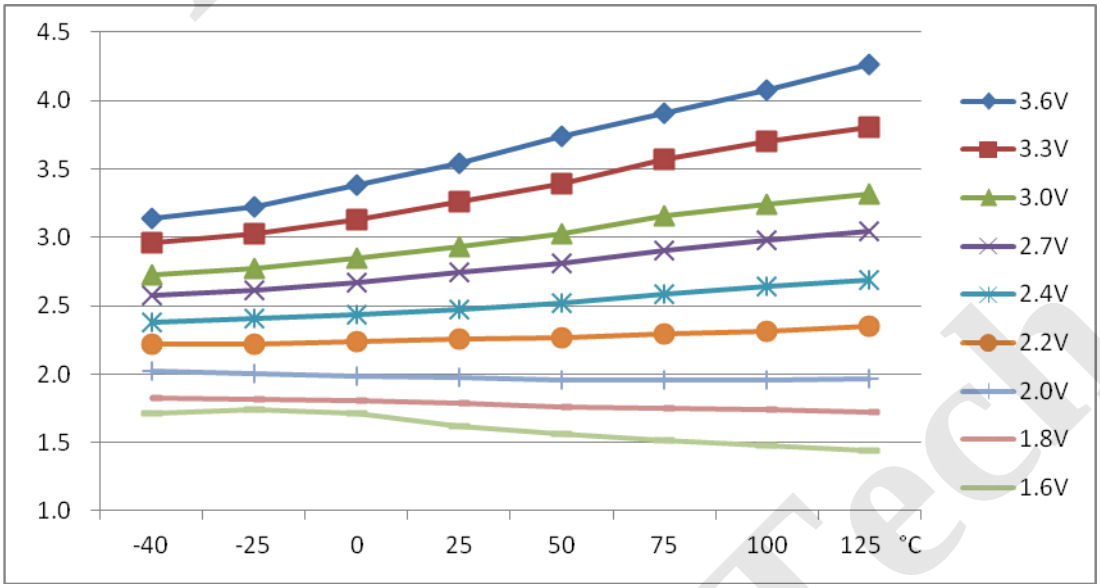


6.8 Recommended Operating Voltage

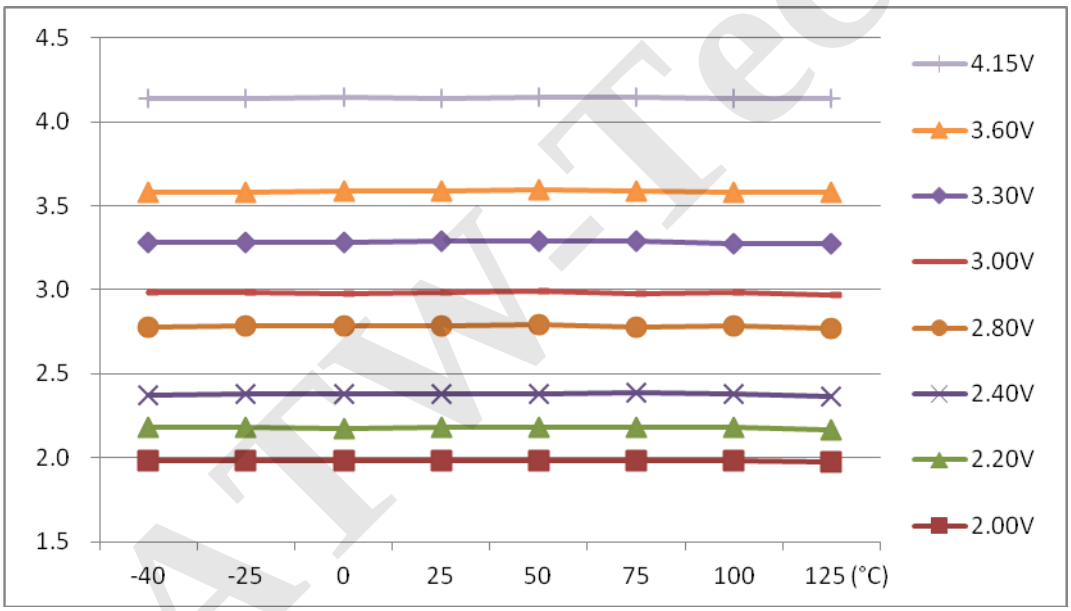
Recommended Operating Voltage (Temperature range: -20°C ~ +85°C)

Frequency	LVR default Setting	LVR Setting (Min. @ 25℃)
20.8M/2T	3.6V	3.6V
16M/2T	3.3V	3.0V
20.8M/4T	2.7V	2.4V
16M/4T	2.4V	2.2V
8M/4T	2.2V	2.0V
4M/4T	2.2V	2.0V

6.9 LVR vs. Temperature

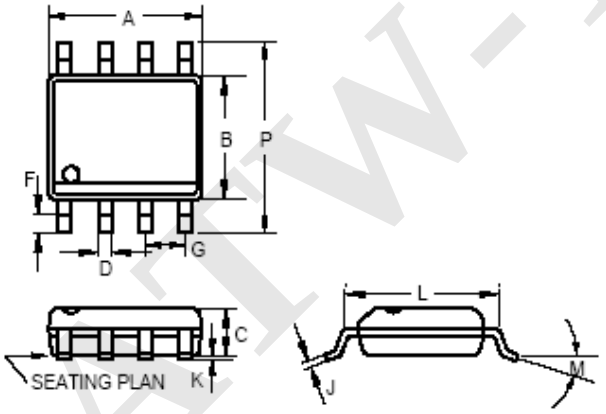


6.10 LVD vs. Temperature



7. Package Dimension

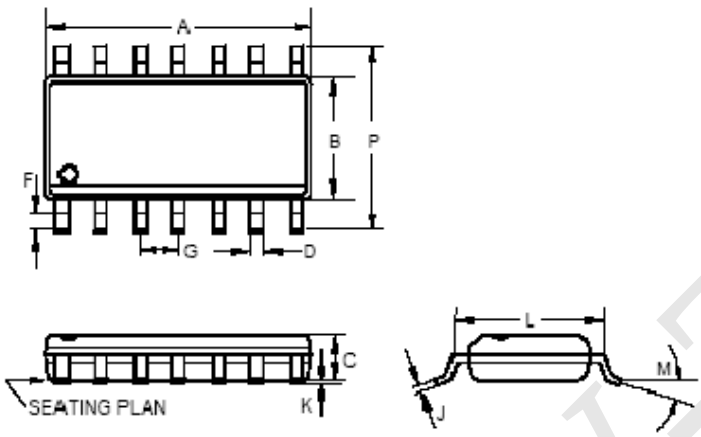
7.1 8-Pin Plastic SOP (150 mil)



	INCHES			MILLIMETERS		
	MIN	TYP	MAX	MIN	TYP	MAX
A	0.183	-	0.202	4.65	-	5.13
B	0.144	-	0.163	3.66	-	4.14
C	0.068	-	0.074	1.35	-	1.88
D	0.010	-	0.020	0.25	-	0.51
F	0.015	-	0.035	0.38	-	0.89
G	0.050 BSC			1.27 BSC		
J	0.007	-	0.010	0.19	-	0.25
K	0.005	-	0.010	0.13	-	0.25
L	0.189	-	0.205	4.80	-	5.21
M	-	-	8°	-	-	8°
P	0.228	-	0.244	5.79	-	6.20

Note: For 8-Pin SOP, 100 units per tube.

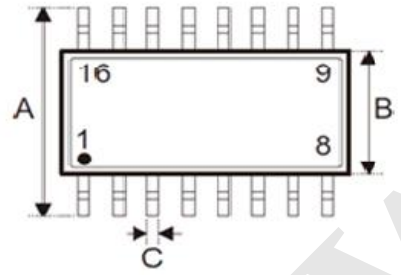
7.2 14-Pin Plastic SOP (150 mil)



	INCHES			MILLIMETERS		
	MIN	TYP	MAX	MIN	TYP	MAX
A	0.337	-	0.344	8.55	-	8.75
B	0.144	-	0.163	3.66	-	4.14
C	0.068	-	0.074	1.73	-	1.88
D	0.017	-	0.020	0.35	-	0.51
F	0.016	-	0.044	0.40	-	1.12
G	0.050 BSC			1.27 BSC		
J	-	0.004	-	-	0.10	-
K	0.005	-	0.010	0.13	-	0.25
L	0.189	-	0.205	4.80	-	5.21
M	-	-	8°	-	-	8°
R	0.228	-	0.244	5.80	-	6.20

Note: For 14-Pin SOP, 50 units per tube.

7.3 16-Pin Plastic SOP (150 mil)



	INCHES			MILLIMETERS		
	MIN	TYP	MAX	MIN	TYP	MAX
A	0.236 BSC			6.00 BSC		
B	0.154 BSC			3.90 BSC		
C	0.012	-	0.020	0.31	-	0.51
C'	0.390 BSC			9.90 BSC		
D	0.065	-	0.069	1.64	-	1.75
E	0.050 BSC			1.27 BSC		
F	0.004	-	0.010	0.10	-	0.25
G	0.016	-	0.050	0.40	-	1.27
H	0.004	-	0.010	0.10	-	0.25
α	-	-	8°	-	-	8°

Note: For 16-pin SOP, 50 units per tube.

8. Ordering Information

P/N	*Package Type*	*Pin Count*	*Package Width*	*Shipping*
AT8BE62DS8	SOP	8	150 mil	Tape & Reel: 2.5K pcs per Reel Tube: 100 pcs per Tube
AT8BE62DS14	SOP	14	150 mil	Tape & Reel: 2.5K pcs per Reel Tube: 50 pcs per Tube
AT8BE62DS16	SOP	16	150 mil	Tape & Reel: 2.5K pcs per Reel Tube: 50 pcs per Tube