HYCON TECHNOLOGY

HY12P Family

User's Guide

Digital Multimeter

Table of Contents

1. Reading Guidance

1.1. About This User Guide

Attention:

1、 HYCON Technology Corp. reserves the right to change the content of this datasheet without further notice. For most up-to-date information, please constantly visit our website: http://www.hycontek.com .

2、 HYCON Technology Corp. is not responsible for problems caused by figures or application circuits narrated herein whose related industrial properties belong to third parties.

3、 Specifications of any HYCON Technology Corp. products detailed or contained herein stipulate the performance, characteristics, and functions of the specified products in the independent state. We does not guarantee of the performance, characteristics, and functions of the specified products as placed in the customer's products or equipment. Constant and sufficient verification and evaluation is highly advised.

4、 Please note the operating conditions of input voltage, output voltage and load current and ensure the IC internal power consumption does not exceed that of package tolerance. HYCON Technology Corp. assumes no responsibility for equipment failures that resulted from using products at values that exceed, even momentarily, rated values listed in products specifications of HYCON products specified herein.

5、 Notwithstanding this product has built-in ESD protection circuit, please do not exert excessive static electricity to protection circuit.

6、 Products specified or contained herein cannot be employed in applications which require extremely high levels of reliability, such as device or equipment affecting the human body, health/medical equipments, security systems, or any apparatus installed in aircrafts and other vehicles.

7、 Despite the fact that HYCON Technology Corp. endeavors to enhance product quality as well as reliability in every possible way, failure or malfunction of semiconductor products may happen. Hence, users are strongly recommended to comply with safety design including redundancy and fire-precaution equipments to prevent any accidents and fires that may follow.

8、 Use of the information described herein for other purposes and/or reproduction or copying without the permission of HYCON Technology Corp. is strictly prohibited.

1.2. Terms and Definition

1.2.1. Glossary

1KB	1KiloByte
ADC	Analog to Digital Converter
Bit	Bit
BOR	Brown-Out Reset
BSR	Bank Select Register
Byte	Byte
CPU	Central Processing Unit
DM	Data Memory
FSR	File Select Register
GPR	General Purpose Register
HAO	High Accuracy Oscillator
LNOP	Low Noise OP AMP
LPO	Low Power Oscillator
LSB	Least Significant Bit
MEM	Memory
MPM	Main Program Memory
MSB	Most Significant Bit
OTP	One Time Program-EPROM
PC	Program Counter
PPF	PWM and PFD
SD18	Sigma-Delta ADC
SR	Special Register
SRAM	Static Random Access Memory
STK	Stack
WDT	Watch Dog Timer
WREG	Work Register

1.2.2. Register Related Glossary

[ ]	Register length
< >	Register value
ABC[7:0]	ABC register had 0 to 7bit
ABC<111>	ABC register had 3bit and value had 111 of binary
ABC<11x>	x：can be neglected, it can be set as 1 or 0

rw	Read/Write
r	Read only
r0	Read as 0
r1	Read as 1
w	Write only
w0	Write as 0
w1	Write as 1
h0	cleared by Hardware
h1	set by Hardware
u0	cleared by User
u1	set by User
-	Not use
!	users are forbidden to change
u	unchanged
x	unknown
d	depends on condition

2. CPU

2.1. CPU Core

CPU Core (H08A) adopts Harvard architecture concept in order to enhance execution efficiency. Separate program memory and data memory incorporated in program memory address increases user convenience of program writing.

CPU features include：

- Isolated design frame of program memory and data memory upgrades instruction execution speed and CPU efficiency.
- Maximum address ability: 8KW for program memory and 256KB for data memory.
- At most 67 instructions including 16-bit look-up-table, 8x8 hardware multiplier and program memory block switch and stack control.
- One instruction accomplished data movement from register A to register B without changing work register data.
- One instruction accomplished utmost 16-bit FSR register data movement and address 8KW program memory look-up-table instruction.
- Data memory operation includes Program Counter (PC), Status Register (Status) and Stack Register (Stack) data movement.
- Processor core is H08A core.

2.2. Memory

Memory is composed by program memory (OTP) and data memory (SRAM). Memory size differs from diverse part number; hence product data sheets should be read with extra caution.

Program Memory：
Main Program Memory (MPM)
Program Counter (PC)
Stack (STK)
Data Memory：
Special Register (SR)
General Purpose Register (GPR)
Memory Related Registers：(x：Means it constitutes several registers).

PC[12:0]	PCHSR[4:0],PCLATH[4:0],PCLATL[7:0]
TOS[12:0]	TOSH[4:0],TOSL[7:0]
FSRx[8:0]	FSRxH[8],FSRxL[7:0]
INDFx	INDF0[7:0],INDF1[7:0]
POINCx	POINC0[7:0], POINC1[7:0]
PODECx	PODEC0[7:0], PODEC1[7:0]
PRINCx	PRINC0[7:0], PRINC1[7:0]
PLUSWx	PLUSW0[7:0], PLUSW1[7:0]
STKCN	STKFL[0],STKOV[0],STKUN[0],STKPRT[2:0]
PSTATUS	SKERR[0]
BSRCN	BSR[0]

2.2.1. Program Memory

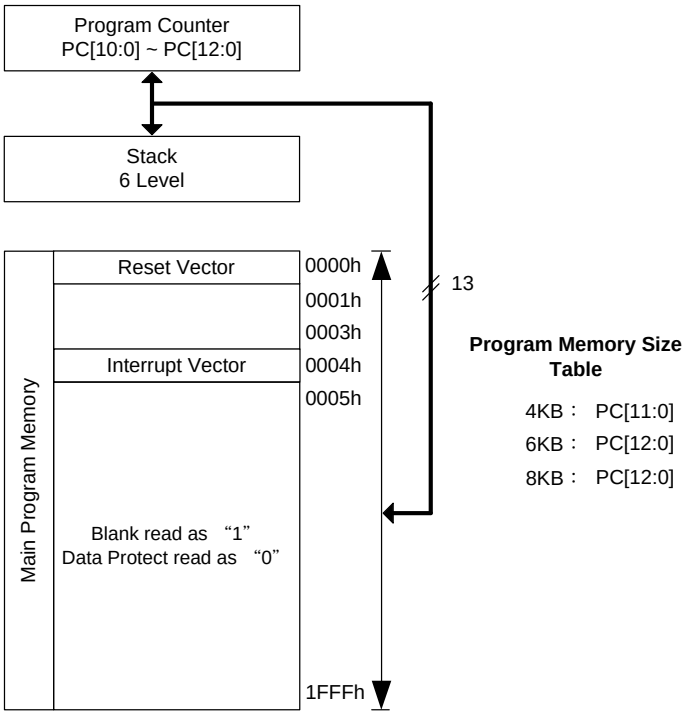


Figure 2-1 Program Memory Flame

2.2.1.1. Main Program Memory, MPM

The frame of main program memory is as follows：

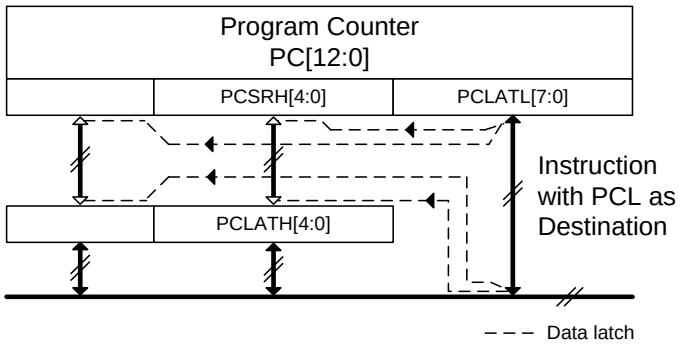
- Interrupt Vector
- Reset Vector

Maximum Address ability starting from 0x00000h to 0x1FFFh, the entire capacity is 8191 characters and it will vary with different part numbers.

Before the IC being written, data type of all bits is 1. After programming, the bit will show 1 or 0 according to the written data type. Please be noticed that if the emulation software (HYIDE) compiling option has been configured the programming protection function, all data type will only be read as 0.

2.2.1.2. Program Counter, PC

Program Counter (PC) includes shift register PCSR and buffer register PCLAT, as Figure 2-2 implicated.



4KB： PC[11:0]
PCLATL[7:0]
PCLATH[3:0]
6KB： PC[12:0]
PCLATL[7:0]
PCLATH[4:0]
8KB： PC[12:0]
PCLATL[7:0]
PCLATH[4:0]

Figure 2-2 Program Counter Frame

PC [12:0][1] of the ICE equips with 13 bit data length and is composed by two registers: PCSRH [4:0] and PCLATL [7:0]. PCLATL [7:0] and PCLATH [4:0] can be directly read/written but PCSRH [4:0] cannot. Buffer register, PCLATH [4:0] must be applied to carry out indirect read and write.

- To read PC[12:0], PCLATL[7:0] must be read first then to read PCLATH[4:0] in order to obtain correct data. Any reverse order may result in incorrect data.
- To write PC[12:0], PCLATH[4:0] must be written first then to write PCLATL[7:0]. Any reverse order may result in incorrect data.

[1] Program memory address ability varies from every product scheme. Common capacities are 4KB (0xFFFh), 6KB (0x17FFh, HY12P65 emulation IC capacity).

```
ORG   0000
      JMP   START
ORG   0004H
      RETI
      ...
START:                    ;jump to 0109h
      MVFF PCLATL,B1
      INF   PCLATH,F,ACCE
      MVL   2
      ADDF B1,W,ACCE
      MVF   PCLATL,F,ACCE
      ...
ORG   0109H
      NOP   ...
```

Example 2-1 Read/Write PCLAT Example Program

2.2.1.3. Stack, STK

Stack, STK is mainly composed by Stack Index Control Register (STKCN), Top-of-Stack Register (TOSx), Stack Layer Register (STKn[2]), Stack Error Flag Bit (SKERR) and Stack Error Reset Controller (SKRST[0]). As presented in Figure 2-3.

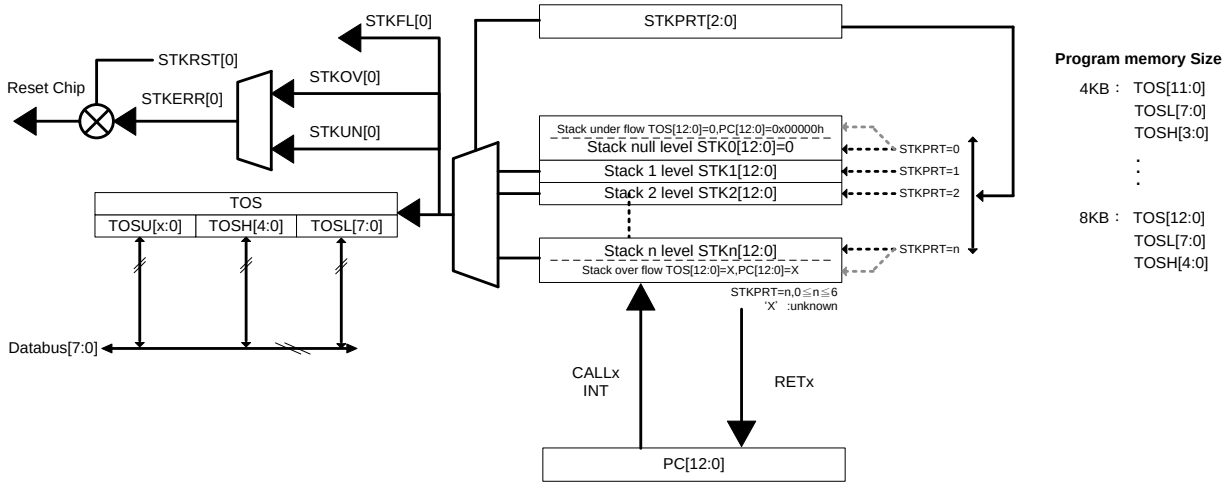


Figure 2-3 Stack architecture diagram

Top-Of-Stack register, TOS[12:0] is 13 bit wide and is constituted by two registers, TOSH[4:0] and TOSL [7:0]. When STKPRT[2:0]=<0>,TOS[12:0]=<0> null. When program executes CALL instruction or an interrupt (INT) is acknowledged, stack pointer STKPRT[2:0] will add 1 and write the PC address into current register, TOS[12:0]. When program executes instruction RETx, STKPRT[2:0] will subtract 1. Before subtracting 1, TOS[12:0] data will be written to PC[12:0]. After completion, STKPRT[2:0] will subtract 1 and change current TOS[12:0] value.

- There is no special regulation of reading register, TOS[12:0]. It can be read directly.
- CALL instruction or interrupt (INT) can be utilized to write PC[12:0] data to register, TOS[12:0]. POP instruction can discard current TOS[12:0] data and may result in 1 decrement of STKPRT[2:0] and may load in new TOS[12:0] data.

STKFL[0] (Stack full), STKOV[0](Stack overflow) or STKUN[0](Stack underflow) may happen during stack operation processes. Stack full is the early warning flag of stack overflow, executes POP instruction this time can discard current TOS[12:0] data and STKPRT[2:0] may subtract 1 and rewrite the newly appointed stack layer data into TOS[12:0]. Users must be aware that when STKPRT[2:0]=<0>, executes POP instruction may not lead to stack underflow, STKPRT[4:0] data can still be <0>. Therefore, users must determine if it is blank stack.

Stack overflow and stack underflow may result in unexpected result of program execution. If there is a necessary, it is suggested to restart the IC. In the processes of

[2] Stack layer register, STKn：Every stack layer has the same length data register as that of top-of-stack register, TOS. When stack index STKPRT being designated, the content of data register will be sent to TOS.

program development, stack reset control bit, SKRST[0][3] can be configured as <1> through software. When stack overflow or stack underflow take place, reset signal will be generated and SKERR[0] will be set as <1> to restart the IC.

- Stack Full: Configure STKFL[0] as <1>, PC[12:0] is not influenced.
- Stack Underflow: Configure STKUN[0] as <1>, PC[12:0] moves to 0x00000h, STKPRT points to 0 Level. If SKRST[0] is set as <1>, reset signal will be aroused after stack underflow and SKERR[0] may be configured as <1>, STKUN[0] will be <0> after reset.
- Stack Overflow: Configure STKOV[0] as <1>, PC[12:0] is not influenced but STKPRT remains at the last layer and new values may be written in. That is to say, the latest written-in data may be saved after stack full. If SKRST[0] is configured as <1>, reset signal may be generated after stack overflow and SKERR[0] may be set as <1>. STKOV[0] will be set as <0> after reset.
- Error: Configure SKERR[0] as <1>, stack error occurred. If SKRST[0] is configured <1>, reset signal will be generated after stack overflow and SKERR[0] will be placed <1>. STKUN[0] and STKOV[0] will be configured as <0> after reset.
- If stack overflow that resulted from ignorance of stack full situation and stack underflow that caused by continuously execute POP instruction happened, STKFL[0], STKOV[0] and STKOV[0] must be configured as <1> in the same time. It is recommended to implement flag clearance action in order to prevent program misjudgment.

To ignore the known stack overflow status when writing program, it is suggested to use POP instruction to erase stack overflow flag, then to continue implementing program. Otherwise, the Interrupt/Call instruction that generated from stack overflow may cape current TOS[12:0] data.

[3] SKRST[0] is the generated reset signal control bit of stack error. Instead of direct read or write, it only can be set by developing software at the program development stage. That is to say, whether to generate stack error reset signal must be determined at program developing stage. If reset is chosen, after IC on powered, SKRST [0] is set as 1, the opposite situation is set as 0.

2.2.1.4. Register Description-Program Memory Controller

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1												
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition												
Address	**File Name**	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**	**A-RESET**	**i-RESET**	**R/W**
16H	TOSH				TOS[12]	TOS[11]	TOS[10]	TOS[9]	TOS[8]	...0 0000	...0 0000	-,-,*,*,*,*,*
17H	TOSL	Top-of-Stack Low Byte (TOS<7:0>)								0000 0000	0000 0000	*,*,*,*,*,*,*
18H	STKPTR	STKFL	STKUN	STKOV			STKPRT[2]	STKPRT[1]	STKPRT[0]	000. .000	000. .000	r,rw0,rw0,- -,r,r,r
1AH	PCLATH				PC[12]	PC[11]	PC[10]	PC[9]	PC[8]	...0 0000	...0 0000	-,-,*,*,*,*,*
1BH	PCLATL	PC Low Byte for PC<7:0>								0000 0000	0000 0000	*,*,*,*,*,*,*
2CH	PSTATUS	PD	TO	IDLEB	BOR		SKERR			000d .0..	uduu .d..	rw0,rw0,rw0,rw0 -,rw0,-,-

Table 2-1 Program Memory Control Register

TOSU/TOSH/TOSL：Top-Of-Stack Register

TOSH：TOS[12:8]

TOSL：TOS[7:0]

STKPTR：Stack Controller

STKFL：Stack full flag

1：Happened

0：Not happened

STKUN：Stack underflow flag

1：Happened

0：Not happened

STKOV：Stack overflow flag

1：Happened

0：Not happened

STKPRT[2:0]：Stack pointer register

110：the 6th layer

101：the 5th layer

000：the 0 layer, TOS[12:0]=0x0000h

PCLATU/PCLATH/PCLATL：Program Counter, PC[12:0]

PCLATH：PC[12:8]

PCLATL：PC[7:0]

PSTATUS：Status Register

SKERR：Stack error generated reset flag

1：Happened

0：Not happened

2.2.2. Data Memory, DM

Data Memory comprises Special Register (SR) and General Purpose Register (GPR), every 256byte is a segment. Segment 0 is particular; include 128byte SR and 128byte GPR respectively. Other segments only contain GPR, as illustrated in Figure 2-4.

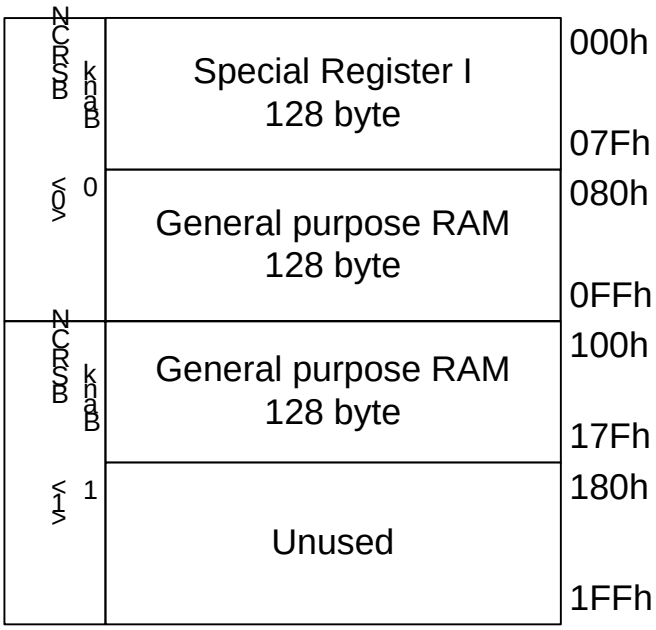


Figure 2-4 Data Memory architecture diagram

2.2.2.1. Memory and Instruction

H08A instruction set is quite flexible in their memory application, such as address ability, hardware multiplier, look-up-table instruction, assistant function and arguments definition. Only definition of instruction memory arguments is illustrated in this chapter. Detail description of instruction arguments is depicted in Instruction chapter.

Instructions that contain address operation function of the instruction set have three arguments, namely "f", "d" and "a".

"f" is Data or Data Memory Address

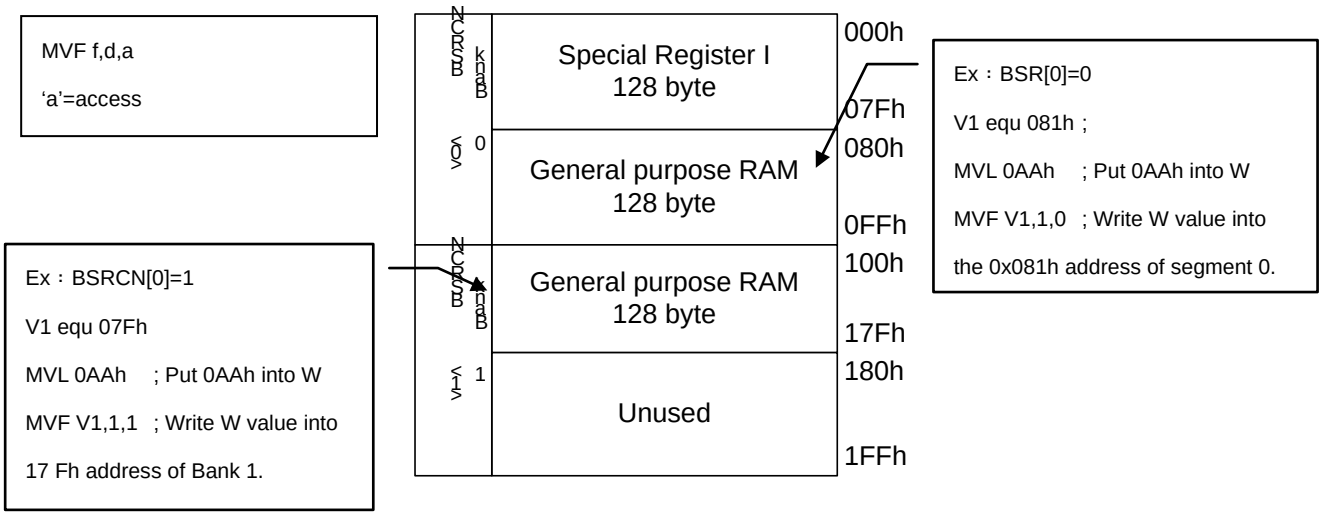
"d" is data storage place after operation. d=0 is saved in WREG register, d=1 is saved in Data Memory Register.

"a" is the designated memory operation segment；a=0 is operated in segment 0, a=1 is operated in designated segment of BSRCN[0].

2.2.2.2. Segment Select Control Register

Every 256 byte of data memory is set to be one segment (000h～0FFh). To read or write address 0FFh register data, it is necessary to set correct segment control register BSRCN [0] and instruction argument "a". Description is as follows:

- When a = 0, no matter register BSRCN [0] appoints to which segment, data memory read/write instruction will only show in segment 0.
- When a = 1, read/write instruction of H08A CPU Core to data memory will be in compliance with the assigned segment of BSRCN[0].



Example 2-2 Relation between Segment Selector Example Program And Data Memory

2.2.2.3. Special Register

Special register comprises CPU Core and peripheral function related registers, mainly are control function registers and data returned registers. Undefined address or address bit of data register will show 0 while reading and writing.

There are several instruction collocation registers contained in special register, only two common types, working register (WREG) and indirect address register (FSR) are introduced herein. Other special registers will be illustrated in depth in other chapter.

2.2.2.3.1. Working Register, WREG

Working register is abbreviated as W, which acts as the most frequently used register for data movement, operation and diagnosis.

2.2.2.3.2. Indirect Address Register, FSR and INDF

File select register, FSR includes instruction register FSR0 [8:0], FSR1 [8:0] and index register, INDF0 [7:0] and INDF1 [7:0]. Because of function similarity, only FSR0 is explained in this chapter.

FSR0[8:0] can be separated into two registers, FSR0H[0] and FSR0L[7:0]. There is no need to set up BSR [0] to address different segments. Through special instruction, only applying one instruction can write 16-bit data.

NDF0[7:0] is index register that can read FSR0[8:0] appointed address data of data memory.

H08A instruction set supports enhanced index register, the functions are characterized as follows：

- POINC0[7:0]：Events that ensued by read/write POINC0 [7:0] register by instruction.
  - The address value that FSR0 [8:0] pointed to will be sent back first.
  - Then pointer register, FSR0[8:0] value will add 1 and points to the next instruction.
- PODEC0[7:0]：Events that followed by read/write PODEC0 [7:0] register by instruction.
  - The address value that FSR0 [8:0] pointed to will be sent back first.
  - Then pointer register, FSR0[8:0] value will subtract 1 and points to the last address.
- PRINC0[7:0]：Events that followed by read/write PRINC0[7:0] register by instruction.
  - Pointer register FSR0[8:0] value will add 1 and points to the next address.
  - Current value of FSR0[8:0] appointed to address will be sent back.
- PLUSW0 [7:0]：Events that ensued by read/write PLUSW0 [7:0] register by instruction.
  - Add pointer register, FSR0[8:0] value together with working register, W value.
  - Send back current FSR0[8:0] appointed address value. The register W value will be±128d.

2.2.2.3.3. General Purpose Register, GPR

General purpose register, GPR is the free area for users to conduct data storage, operation, flag bit setup…etc.

2.2.2.4. Register Description- Data Memory Controller

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1												
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition												
Address	**File Name**	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**	**A-RESET**	**i-RESET**	**R/W**
00H	INDF0	Contents of FSR0 to address data memory˙ value of FSR0 not changed								N/A	N/A	*,*,*,*,*,*,*,*
01H	POINC0	Contents of FSR0 to address data memory˙ value of FSR0 post-incremented								N/A	N/A	*,*,*,*,*,*,*,*
02H	PODEC0	Contents of FSR0 to address data memory˙ value of FSR0 post-decremented								N/A	N/A	*,*,*,*,*,*,*,*
03H	PRINC0	Contents of FSR0 to address data memory˙ value of FSR0 pre-incremented								N/A	N/A	*,*,*,*,*,*,*,*
04H	PLUSW0	Contents of FSR0 to address data memory˙ value of FSR0 offset by W								N/A	N/A	*,*,*,*,*,*,*,*
05H	INDF1	Contents of FSR1 to address data memory˙ value of FSR0 not changed								N/A	N/A	*,*,*,*,*,*,*,*
06H	POINC1	Contents of FSR1 to address data memory˙ value of FSR0 post-incremented								N/A	N/A	*,*,*,*,*,*,*,*
07H	PODEC1	Contents of FSR1 to address data memory˙ value of FSR0 post-decremented								N/A	N/A	*,*,*,*,*,*,*,*
08H	PRINC1	Contents of FSR1 to address data memory˙ value of FSR0 pre-incremented								N/A	N/A	*,*,*,*,*,*,*,*
09H	PLUSW1	Contents of FSR1 to address data memory˙ value of FSR0 offset by W								N/A	N/A	*,*,*,*,*,*,*,*
0FH	FSR0H								FSR0[8]	x	....u	-,-,-,- -,-,-,*
10H	FSR0L	Indirect Data Memory Address Pointer 0 Low Byte,FSR0[7:0]								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
11H	FSR1H								FSR1[8]	x	....u	-,-,-,- -,-,-,*
12H	FSR1L	Indirect Data Memory Address Pointer 1 Low Byte,FSR1[7:0]								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
29H	WREG	Working Register								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
2AH	BSRCN								BSR[0]	 0000	 0000	-,-,-,- *,*,*,*

Table 2-2 Data Memory Control Register

INDF0/POINC0/PODEC0/PRINC0/PLUSW0：Different Functional Index Register

INDF0[7:0]：Please refer to **File Select Register, FSR and INDF** description.

POINC0[7:0]：Please refer to **File Select Register, FSR and INDF** description.

PODEC0[7:0]：Please refer to **File Select Register, FSR and INDF** description.

PRINC0[7:0]：Please refer to **File Select Register, FSR and INDF** description.

PLUSW0[7:0]：Please refer to **File Select Register, FSR and INDF** description.

FSR0：Force Select Register

FSR0H[0]：Please refer to **File Select Register, FSR and INDF** description.

FSR0L[7:0]：Please refer to **File Select Register, FSR and INDF** description.

FSR1：Force Select Register

FSR1H[0]：Please refer to **File Select Register, FSR and INDF** description.

FSR1L[7:0]：Please refer to **File Select Register, FSR and INDF** description.

WREG：Force Select Register

WREG[7:0]：Please refer to **Working Register, WREG** description.

BSRCN：Memory Segment Read/Write Control Register

BSR[0]：Memory read/write segment pointer register

1: Segment 1, address 0x100h~0x1FFh.

0: Segment 0, address 0x000h~0x0FFh.

2.2.3. Register List-Data Memory

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition

Address	File Name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	A-RESET	i-RESET	R/W
00H	INDF0	Contents of FSR0 to address data memory˙ value of FSR0 not changed								N/A	N/A	*,*,*,*,*,*,*,*
01H	POINC0	Contents of FSR0 to address data memory˙ value of FSR0 post-incremented								N/A	N/A	*,*,*,*,*,*,*,*
02H	PODEC0	Contents of FSR0 to address data memory˙ value of FSR0 post-decremented								N/A	N/A	*,*,*,*,*,*,*,*
03H	PRINC0	Contents of FSR0 to address data memory˙ value of FSR0 pre-incremented								N/A	N/A	*,*,*,*,*,*,*,*
04H	PLUSW0	Contents of FSR0 to address data memory˙ value of FSR0 offset by W								N/A	N/A	*,*,*,*,*,*,*,*
05H	INDF1	Contents of FSR1 to address data memory˙ value of FSR0 not changed								N/A	N/A	*,*,*,*,*,*,*,*
06H	POINC1	Contents of FSR1 to address data memory˙ value of FSR0 post-incremented								N/A	N/A	*,*,*,*,*,*,*,*
07H	PODEC1	Contents of FSR1 to address data memory˙ value of FSR0 post-decremented								N/A	N/A	*,*,*,*,*,*,*,*
08H	PRINC1	Contents of FSR1 to address data memory˙ value of FSR0 pre-incremented								N/A	N/A	*,*,*,*,*,*,*,*
09H	PLUSW1	Contents of FSR1 to address data memory˙ value of FSR0 offset by W								N/A	N/A	*,*,*,*,*,*,*,*
0FH	FSR0H								FSR0[8]	x	....u	-,-,-,- -,-,-,*
10H	FSR0L	Indirect Data Memory Address Pointer 0 Low Byte,FSR0[7:0]								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
11H	FSR1H								FSR1[8]	x	....u	-,-,-,- -,-,-,*
12H	FSR1L	Indirect Data Memory Address Pointer 1 Low Byte,FSR1[7:0]								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
16H	TOSH				TOS[12]	TOS[11]	TOS[10]	TOS[9]	TOS[8]	...0 0000	...0 0000	-,-,*,*,*,*,*,*
17H	TOSL	Top-of-Stack Low Byte (TOS<7:0>)								0000 0000	0000 0000	*,*,*,*,*,*,*,*
18H	STKPTR	STKFL	STKUN	STKOV			STKPRT[2]	STKPRT[1]	STKPRT[0]	000. .000	000. .000	r,rw0,rw0,- -,r,r,r
1AH	PCLATH				PC[12]	PC[11]	PC[10]	PC[9]	PC[8]	...0 0000	...0 0000	-,-,*,*,*,*,*,*
1BH	PCLATL	PC Low Byte for PC<7:0>								0000 0000	0000 0000	*,*,*,*,*,*,*,*
1DH	TBLPTRH				TBLPTR[12]	TBLPTR[11]	TBLPTR[10]	TBLPTR[9]	TBLPTR[8]	...0 0000	...0 0000	-,-,*,*,*,*,*,*
1EH	TBLPTRL	Program Memory Table Pointer Low Byte (TBLPTR<7:0>)								0000 0000	0000 0000	*,*,*,*,*,*,*,*
1FH	TBLDH	Program Memory Table Latch High Byte								0000 0000	0000 0000	*,*,*,*,*,*,*,*
20H	TBLDL	Program Memory Table Latch Low Byte								0000 0000	0000 0000	*,*,*,*,*,*,*,*
21H	PRODH	Product Register of Multiply High Byte								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
22H	PRODL	Product Register of Multiply Low Byte								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
23H	INTE1	GIE		TMCIE		TMAIE	WDTIE	E1IE	E0IE	0.0. 0000	0.0. 0000	*,-,*,- *,*,*
24H	INTE2	TXIE	RCIE	RMSIE	LPFIE	AD1IE		CTIE		0000 000.	0000 000.	*,*,*,*,*,-
25H	INTE3	E24IE	E25IE	E26IE	E27IE					0000	0000	*,*,* -,-,-,-
26H	INTF1			TMCIF		TMAIF	WDTIF	E1IF	E0IF	..0. 0000	..0. 0000	-,-,*,- *,*,*
27H	INTF2	TXIF	RCIF	RMSIF	LPFIF	AD1IF		CTIF		0000 000.	0000 000.	*,*,*,*,*,-
28H	INTF3	E24IF	E25IF	E26IF	E27IF					0000	0000	*,*,*,* -,-,-,-
29H	WREG	Working Register								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
2AH	BSRCN								BSR[0]	 0000	 0000	-,-,-,- *,*,*,*
2BH	STATUS				C	DC	N	OV	Z	...x xxxx	...u uuuu	-,-,-,*,*,*,*,*
2CH	PSTATUS	PD	TO	IDLEB	BOR		SKERR			000d .0..	uduu .d..	rw0,rw0,rw0,rw0 -,rw0,-,-
2DH	LVDCN1	ENLVD	LVD	VJ1	VJ2	VLDX[3:0]						
2EH	LVDCN2	VSL	SVIN[2:0]			SVIP[3:0]						
2FH	SBMSET1	SKRST	HAOTR[5:0]							x.xx xxxx	u.uu uuuu	*,-,*,*,*,*,*,*
30H	MCKCN1	HSSEL	CPUCK[1:0]		HSS[1:0]		HSCK	ENXT	ENHAO	0000 0001	0000 0001	*,*,*,*,*,*,*,*
31H	MCKCN2	LCDS[2:0]			ADCCK	PERCK	BZS[2:0]			0000 0000	0000 0000	*,*,*,*,*,*,*,*
32H	TMACN	ENTMA	TMACK	TMAS[1:0]		ENWDT	WDTS[2:0]			0000 0000	0000 0000	*,*,*,* w1,*,*,*
33H	TMAR	TimerA data register								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
34H	TMCCN	ENTMC	TMCCK[1:0]		TMCS1[2:0]			TMCS0[1:0]		0000 0000	0000 0000	*,*,*,*,*,*,*,*
35H	PRC	TimerC programmable register								1111 1111	1111 1111	*,*,*,*,*,*,*,*
36H	TMCR	TimerC register								0000 0000	0000 0000	r,r,r,r r,r,r,r
37H	PWMCN	ENPWM	ENPFD	PWMRL[1:0]						0000 0000	0000 0000	*,*,*,*,*,*,*,*
38H	PWMR	PWM MSB Byte register								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
39H	LCDCN1	ENLCD	LCDPR	VLCDX[1:0]		LCDBF	LCDBI[1:0]			0000 000.	0000 000.	*,*,*,*,*,*,-
3AH	LCDCN2	LCDBL	LCDMX[1:0]							000.	000.	*,*,*,- -,-,-,-
3BH	LCD0	Segment SEG1@[7:4] and SEG0@[3:0] data register of LCD								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
3CH	LCD1	Segment SEG3@[7:4] and SEG2@[3:0] data register of LCD								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
3DH	LCD2	Segment SEG5@[7:4] and SEG4@[3:0] data register of LCD								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
3EH	LCD3	Segment SEG7@[7:4]and SEG6@[3:0] data register of LCD								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
3FH	LCD4	Segment SEG9@[7:4] and SEG8@[3:0] data register of LCD								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
40H	LCD5	Segment SEG11@[7:4] and SEG10@[3:0] data register of LCD								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
41H	LCD6	Segment SEG13@[7:4] and SEG12@[3:0] data register of LCD								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
42H	LCD7					Segment SEG14@[3:0] data register of LCD				 xxxx	.... uuuu	-,-,-,- *,*,*,"
46H	URCON	ENSP	ENTX	TX9	TX9D	PARITY			WUE	0000 0..0	0000 0..0	*,*,*,*,-,-,*
47H	URSTA		RC9D	PERR	FERR	OERR	RCIDL	TRMT	ABDOVF	.000 0110	.000 0110	-,r,r,r r,r,r,rw0
48H	BAUDCON					ENCR	RC9	ENADD	ENABD	 0000	 0000	-,-,-,- *,*,*,*
49H	BRGRH	Baud Rate Generator Register High Byte								...x xxxx	...u uuuu	-,-,-,*,*,*,*,*
4AH	BRGRL	Baud Rate Generator Register Low Byte								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
4BH	TXREG	UART Transmit Register								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
4CH	RCREG	UART Receive Register								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r

Table 2-3 Data Memory List

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1												
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition												
Address	**File Name**	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**	**A-RESET**	**i-RESET**	**R/W**
4DH	PT1	PT1.7	PT1.6	PT1.5	PT1.4	PT1.3	PT1.2	PT1.1	PT1.0	xxxx xxxx	uuuu uuuu	*,*,*,* r,r,r,r
4EH	TRISC1	TC1.7	TC1.6	TC1.5	TC1.4	TC1.3	TC1.2	TC1.1	TC1.0	0000 0000	0000 0000	*,*,*,*,*,*,*,*
4FH	PT1PU	PU1.7	PU1.6	PU1.5	PU1.4	PU1.3	PU1.2	PU1.1	PU1.0	0000 0000	0000 0000	*,*,*,*,*,*,*,*
50H	PT1M1	PM1.7	PM1.6	PM1.5	PM1.4	INTEG1[1:0]		INTEG0[1:0]		0000 0000	0000 0000	*,*,*,*,*,*,*,*
51H	PT2	PT2.7	PT2.6	PT2.5	PT2.4	PT2.3	PT2.2	PT2.1	PT2.0	xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
52H	TRISC2	TC2.7	TC2.6	TC2.5	TC2.4	TC2.3	TC2.2	TC2.1	TC2.0	0000 0000	0000 0000	*,*,*,*,*,*,*,*
53H	PT2DA	DA2.7	DA2.6	DA2.5	DA2.4	DA2.3		PM2.2[1:0]		0000 0.00	0000 0.00	*,*,*,*,-,*,*
54H	PT2PU	PU2.7	PU2.6	PU2.5	PU2.4	PU2.3	PU2.2	PU2.1	PU2.0	0000 0000	0000 0000	*,*,*,*,*,*,*,*
55H	PT3	PT3.7	PT3.6	PT3.5		TC3.7	TC3.6	TC3.5		xxx. 000.	uuu. 000.	*,*,*,- *,*,*,-
56H	PT3PU	PU3.7	PU3.6	PU3.5		PM3.7		DA3.5		000. 0.0.	000. 0.0.	*,*,*,- *,-,*,-
57H	PAX6					PS6	DS6	FS6	SS6			
58H	PA54	PS5	DS5	FS5	SS5	PS4	DS4	FS4	SS4			
59H	PA32	PS3	DS3	FS3	SS3	PS2	DS2	FS2	SS2			
5AH	PA10	PS1	DS1	FS1	SS1	PS0	DS0	FS0	SS0			
5BH	PWRCN	DMMBIAS	SAGND[1:0]		ENVS	ENREFO	ENLDO	LDOC[1:0]				
5CH	PWRCN2	MCUBIAS	ENCPVGG	ENCMP	ENCNTI	ENCTR	RSTCOMB	RSLPF	RSRMS			
5DH	ADCN1	SDIO	SREFO	SFT1<1:0>		SFUVR<3:0>						
5EH	ADCN2	SMODE<7:0>										
5FH	ADCN3	SCMPRH<3:0>				SCMPRL<3:0>						
60H	ADCN4	SCMPI<2:0>			AD1CHOP<1:0>		AD1OSR<2:0>					
61H	ADCN5	SAD1FP<3:0>				HSAD	SAD1FN<2:0>					
62H	ADCN6	SAD1RH<2:0>			SAD1RL<2:0>			SAD1I<1:0>				
63H	ADCN7	ENAD1	AD1IG<1:0>		AD1RG	AD1RHBUF	AD1RLBUF	AD1IPBUF	AD1INBUF			
64H	RMSCN	ENRMS	ENLPF	ENRMSAVG	LPFBW<1:0>		ENPKH					
65H	CTAU	CTA<23:16>								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
66H	CTAH	CTA<15:8>								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
67H	CTAL	CTA<7:0>								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
68H	CTBU	CTB<23:16>								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
69H	CTBH	CTB<15:8>								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
6AH	CTBL	CTB<7:0>								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
6BH	CTCU	CTC<23:16>								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
6CH	CTCH	CTC<15:8>								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
6DH	CTCL	CTC<7:0>								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
6EH	CTSTA	CNTI	ACPO	CMPHO	CMPLO				CTBOV			
6FH	PKHMAXU	PKHMAX<18:11>								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
70H	PKHMAXH	PKHMAX<10:3>								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
71H	PKHMAXL	PKHMAX<2:0>								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
72H	PKHMINU	PKHMIN<18:11>								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
73H	PKHMINH	PKHMIN<10:3>								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
74H	PKHMINL	PKHMIN<2:0>								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
75H	RMSDATA4	RMS<37:30>								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
76H	RMSDATA3	RMS<29:22>								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
77H	RMSDATA2	RMS<21:14>								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
78H	RMSDATA1	RMS<13:6>								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
79H	RMSDATA0	RMS<5:0>								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
7AH	LPFDATAU	LPF<18:11>								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
7BH	LPFDATAH	LPF<10:3>								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
7CH	LPFDATAL	LPF<2:0>								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
7DH	AD1DATAU	AD1<18:11>								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
7EH	AD1DATAH	AD1<10:3>								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
7FH	AD1DATAL	AD1<2:0>								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
80H ~ FFH	GPR0	General Purpose Register as 128Byte								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
100H~17FH	GPR1	General Purpose Register as 128Byte								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*

Table 2-4 Data Memory List (continued)

3. Oscillator, Clock Sources and Power Managed Modes

HY12P Series has three clock sources, HAO, LPO and XT as shown in Table 3-1. Through clock sources controller register set-up, it helps to flexibly manage CPU and peripheral operating frequency. Moreover, it also appropriately adjusts IC's consumed power as to reach the purpose of energy economy.

Clock Source Control Register:

MCKCN1 HSSEL[0],CPUCK[1:0],HSS[1:0],HSCK[0],ENXT[0],ENHAO[0]

MCKCN2 LCDS[2:0],ADCCK[0],PERCK[0],BZS[2:0]

Oscillator			Oscillation Way
Symbol	Attributes	Frequency	
HAO	Internal	4MHz	RC
LPO		32KHz	RC
XTL/S/H	External	32768Hz ~ 8MHz	crystal/ RC

Table 3-5 IC Clock Sources

3.1. Oscillator

3.1.1. XT External Crystal/ Resonator Oscillator

Clock sources control register, MCKCN1[7:0] must be set up in accordance with external operating frequency of oscillator, as Table 3-2. Figure 3-1 depicts the block diagram for HY12P Series external oscillator.

Using external oscillator, I/O must be set in input mode and cannot use internal pull high resistance. MCKCN1 register control bit have to be configured in compliance with its oscillation frequency in order to determine external crystal/resonator operating frequency.

As Figure 3-1 illustrates, C1, C2 and R1 value changes according to different external crystal oscillator or resonator frequency. What is more, even with the same oscillator, capacitance value will have small discrepancy resulted from different PCB layout. Under not particularly consideration, capacitance C1 and C2 can be left out; Table 3-2 lists the value for design guidance, not optimized value.

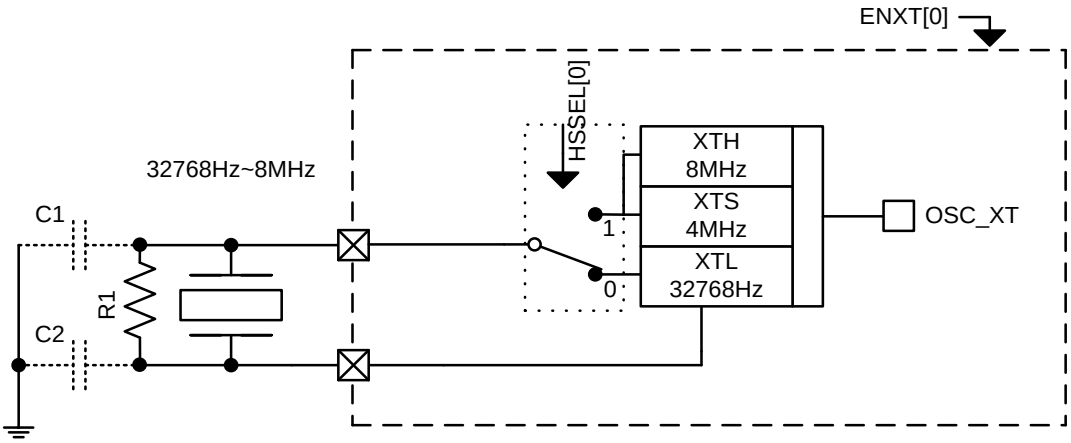


Figure 3-5 Block Diagram of External Oscillator

External oscillator is sorted into three oscillation frequency levels, each level has diverse driving current and thus, different configuration must be made on the apparatus, as Table 3-2 exhibits. When external oscillator is applied, ENXT [0] must be set as 1. If the external oscillator frequency is 4MHz or higher, HSSEL[0] must be set up as 1 as well. On the contrary, if no changes have been made to HSSEL[0] based on different external oscillation frequencies; external oscillator may not be started-up and may cause huge power consumption.

"x": ignore, resistor unit:Ω,capacitor unit: F

Symbol	Frequency	Ceramic Resonator			Crystal Oscillator			MCKCN1 Configuration	
		C1	C2	R1	C1	C2	R1	ENXT	HSSEL
XTL	455Hz							1	0
	32768Hz			10M	20p	20p	10M	1	0
XTS	4.0MHz			1M	20p	20p	1M	1	1
XTH	8.0MHz			1M	20p	20p	1M	1	1

Table 3-6 Oscillator Optimized Capacitor Value and MCKCN1 Register Set Up

3.1.2. HAO Oscillator

HAO is internal high speed RC oscillator; the typical output frequency is 4.0MHz. Compared to external XT oscillator, internal HAO equips with fast Start-up and better anti-interference character, thus, HAO is called restated CPU operation clock source.

When CPU of HY12P Series utilizes other oscillator as operating clock source, HAO oscillator can be turned off by setting ENHAO[0] as <0>.

3.1.3. LPO Oscillator

LPO is internal low speed RC oscillator, the typical output frequency is 32KHz. LPO consumes 0.7uA current; therefore, it is mostly implemented in low speed and power efficient CPU operation and Watch Dog Timer clock source.

After executing Sleep instruction, LPO oscillator of HY12P Series products will be shut off and will oscillate automatically as the IC being awakened.

3.2. CPU and Peripheral Circuit Clock Sources

3.2.1. Clock Sources Configuration

Three sets of oscillator output（OSC_XT、OSC_HAO、OSC_LPO）will firstly pass through pre-operating clock divider to initiate start/stop, switch and prescaling, then to CPU and other peripheral configurations of the IC. Pre-operating clock divider can produce four different kinds of clock source frequency. As Figure 3-2 indicates, based on their interdependence and frequency speed, the permutation is shown as:

HS_CK≧HSS_CK≧HS_DCK or LS_CK.

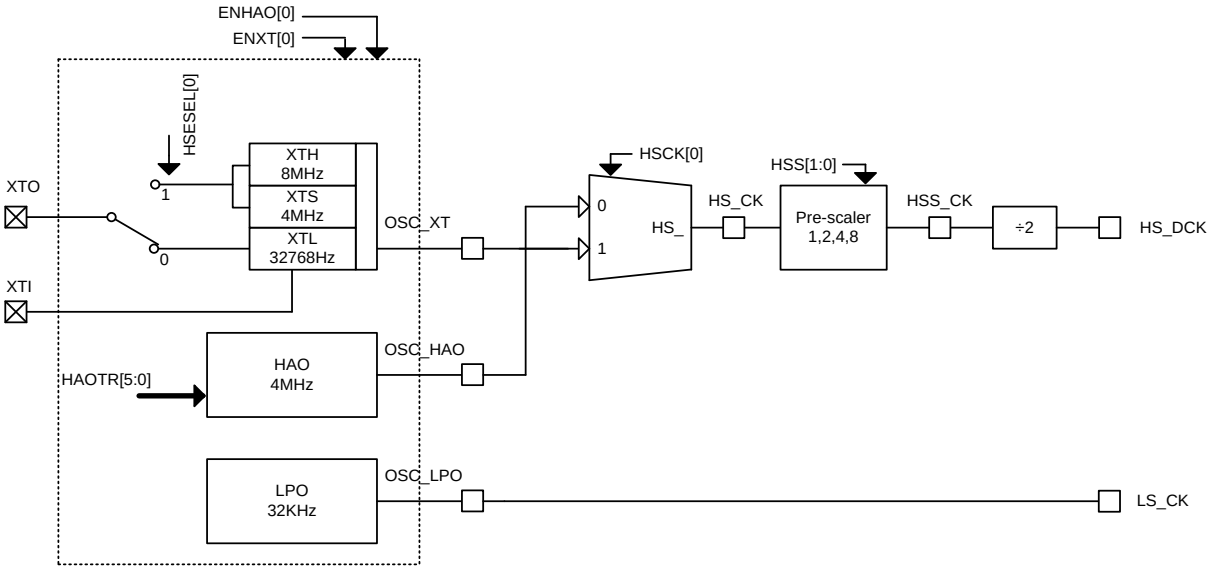


Figure 3-6 Pre-Operating Clock Divider

```
Example program：
Setting up the internal secondary oscillator HAO as primary clock source,
HS_DCK outputs 1MHz frequency. Setting LS_CK as internal LPO outputs 32KHz
frequency.
MVL     00101001b     ；Starting HAO
MVF     MCKCN1,1,0    ；Set up HS_CK clock source as OSC_HAO
                        Set up LS_CK clock source as OSC_LPO
                       First, HSS[1:0] prescale 2
                        Then, internal prescale 2, can get HS_DCK= 1MHz
```

Example 3-2 HS_DCK Outputs 1MHz Example Program

```
Example program：
Configure external  8MHz oscillator as the primary clock source. HS_DCK outputs  4MHz
frequency and LS_CK is configured as internal  LPO output frequency  32KHz.
CLRF    TRISC2,0      ; Set up  PT2.0, PT2.1 for external oscillator input signal
CLRF    PT2PU,0
MVL     10000011b     ; Set up external 8MHz oscillation
MVF     MCKCN1,1,0    ; Set up  CPU_CK as  HS_DCK
CALL    DELAY         ; DELAY LOOP is vice program of delay time
                      ; Reserve about 30msec delay time for oscillation
MVL     10100110b     ; Set up  CPU_CK as  HS_DCK and switches to external oscillator
                      ; HS_DCK clock source is OSC_XT_XT
                      ; LS_CK clock source is OSC_LPO
                      ; Instruction cycle is  INTR_CK=8M/2/4=1MHz
                      ;Turn off internal OSC_HAO frequency source and switches to
                      external oscillator for power-saving
MVF     MCKCN1,1,0    ;
NOP
```

Example 3-2 HS_DCK Outputs 4MHz (External Oscillator) Example Program

3.2.2. CPU Clock Source

There are four clock sources of IC CPU core operating frequency namely, HS_CK, HSS _CK, HS_DCK and LS_CK.

Instruction execution cycle is CPU_CK/4 and PERA_CK is the main clock source of peripherals, as Figure 3-3. Table 3-3 briefly indicates the relation between CPU Operating Frequency and Instruction Execution Cycle.

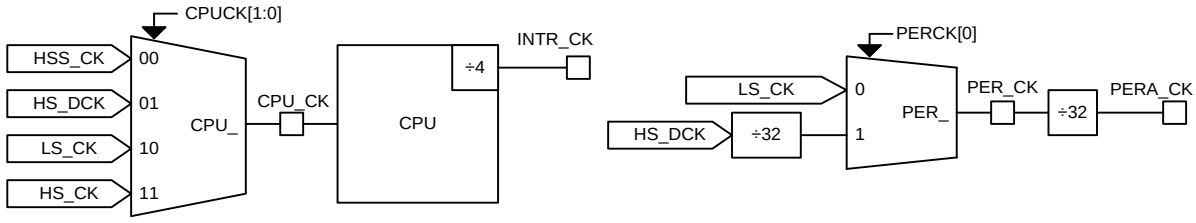


Figure 3-7 CPU and Peripheral Operating Clock Sources

Operation Clock Source HAO, LPO		CPU Operation Frequency CPU_CK	Instruction Operation Cycle INST_CK
HSS_CK	4MHz	4MHz	1us
HS_DCK	4MHz	2MHz	2us
LS_CK	32KHz	32KHz	125us
HS_CK	4MHz	4MHz	1us

Table 3-7 HSS[1:0]=01b, CPU Operating Frequency and Instruction Execution Cycle

3.2.3. CPU Clock Source

Clock source of HY12P Series peripheral circuit are configured by different distribution controller and prescaler. The configuration will be fully illustrated in each peripheral unit; hence Figure 3-4 only presents the peripheral clock sources configuration.

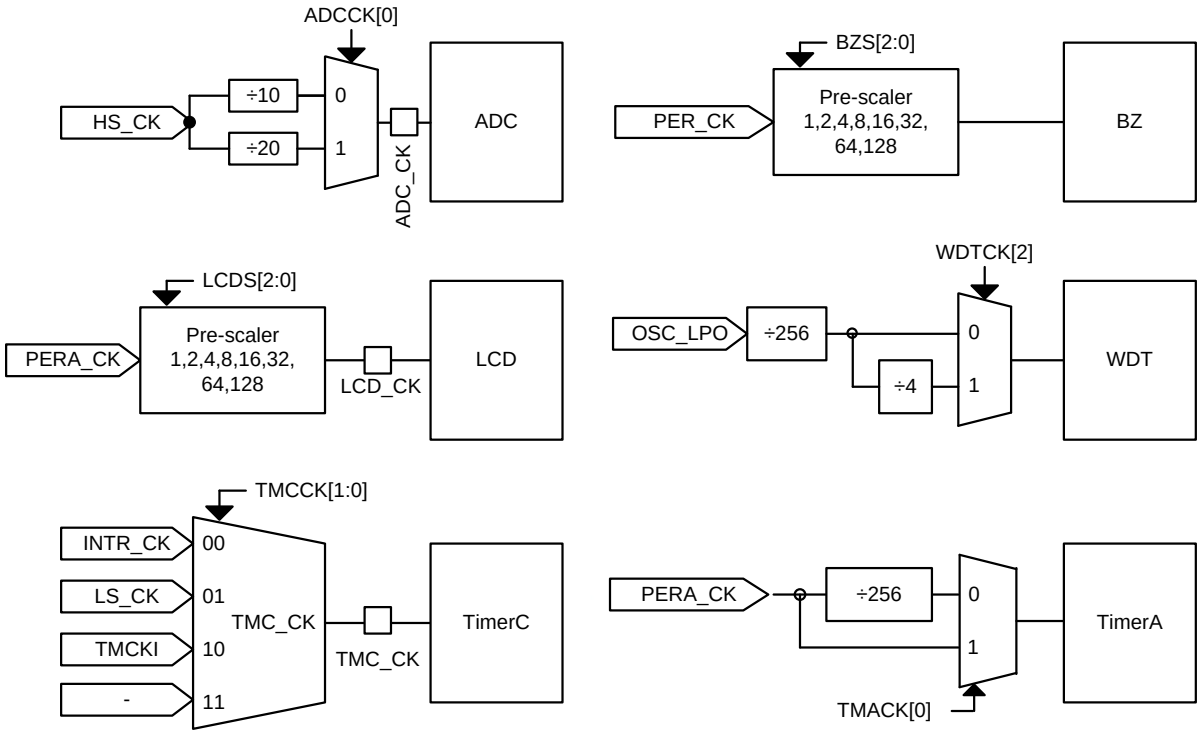


Figure 3-8 Configuration of Peripheral Operating Clock Source

3.3. Register Description-Operating Clock Source Controller

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1												
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition												
Address	**File Name**	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**	**A-RESET**	**i-RESET**	**R/W**
30H	MCKCN1	HSSEL	CPUCK[1:0]		HSS[1:0]		HSCK	ENXT	ENHAO	0000 0001	0000 0001	*,*,*,*,*,*,*,*
31H	MCKCN2	LCDS[2:0]			ADCCK	PERCK	BZS[2:0]			0000 0000	0000 0000	*,*,*,*,*,*,*,*
51H	PT2							PT2.1	PT2.0	xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
52H	TRISC2							TC2.1	TC2.0	0000 0000	0000 0000	*,*,*,*,*,*,*,*
54H	PT2PU							PU2.1	PU2.0	0000 0000	0000 0000	*,*,*,*,*,*,*,*

Table 3-8 Operating Clock Source Control Register

MCKCN1：Operating Clock Source Controller 1

HSSEL：Control bit of external oscillator frequency selection

1：XTS/XTH mode

0：XTL mode

CPUCK[1:0]：Control bit of CPU operating clock selection

11：HS_CK

10：LS_CK

01：HS_DCK

00：HSS_CK

HSS[1:0]：Prescaler of high speed clock source

11：HS_CK/8

10：HS_CK/4

01：HS_CK/2

00：HS_CK/1

HSCK：Control bit of high speed clock source selector

When ENXT = 1

1：OSC_XT

0：OSC_HAO

When ENXT = 0

1：Cannot be configured

0：OSC_HAO

ENXT：Control bit of starting external crystal/resonator

1：Active

0：Inactive

ENHAO：Control bit of starting internal HAO (4MHz)

1：Active

0：Inactive

MCKCN2：Operating Clock Source Controller 2

LCDS[2:0]：Prescaler of LCD peripheral operating frequency

111：PERA_CK/128

110：PERA_CK/64

101：PERA_CK/32

100：PERA_CK/16

011：PERA_CK/8

010：PERA_CK/4

001：PERA_CK/2

000：PERA_CK/1

ADCCK：Selector of SD18 peripheral operating clock source

1：HSCK/20

0：HSCK/10

PERCK：Control bit for peripheral clock source selection

1：HS_DCK/32。

0：LS_CK。

BZS[2:0]：Prescaler of BZ peripheral operating frequency

111：PER_CK/128。

110：PER_CK/64。

101：PER_CK/32。

100：PER_CK/16。

011：PER_CK/8。

010：PER_CK/4。

001：PER_CK/2。

000：PER_CK/1。

PT2：PORT2 Status Control Register

PT2.1：Control bit of external pins

1：high potential

0：low potential

PT2.0：Control bit of external pins

1：high potential

0：low potential

TRISC2：Input/Output Control Register

TC2.1：Input/output control bit of external pins

1：output

0：input

TC2.0：Input/output control bit of external pins

1：output

0：input

PT2PU：Pull-Up Resistor Control Register

PU2.1：Pull-up resistor control bit of external pins

1：Start

0：Shutoff

PU2.0：Pull-up resistor control bit of external pins

1：Start

0：Shutoff

3.4. Power Managed and Operation Mode

HY12P Series CPU provides three operating modes for users to acquire the best managed circumstance between operating efficiency and power economy. These modes are run mode, idle mode and sleep mode.

3.4.1. Run Mode

Run mode means that CPU dealt with every occurred event in accordance with the clock source. At this time, all peripherals operate normally and consume maximum power within the same clock source.

3.4.2. Idle Mode

IDLE Mode means that CPU stops running to wait for awakening as long as it accesses into energy efficient status. The IDLEB [0] flag of PSTATUS reset register is set as 1. At this time, peripherals still operate normally. When interrupt event occurs, CPU[4] will then be awakened. Moreover, under this mode, signals produced when WDT counter ceased, are part of interrupt signals instead of reset signals. Example program can be referred to Example 3-4。

3.4.3. Sleep Mode

The chip stops operation in Sleep Mode. CPU, internal oscillator (HAO & LPO) and external oscillator (XT) also ceases operation. Peripherals that use HAO, LPO, XT related clock sources will stop operation and PD flag bit of PSTATUS reset register will be configured as <1>. Under this mode, the chip will be awakened by interrupt event for partial peripherals still retaining in open status despite stop operation. In order to maintain the lowest power consumption under Sleep Mode, peripherals that do not related to awaking the chip must be further turned off. Example program can be founded in Example 3-5。

[4] When CPU was awoken by interrupt signals, PC will switch to interrupt vector address (0x004h). For detail PSTATUS reset register and interrupt service vector description, please refer to *Reset, Interrupt* chapter.

```
MVL     01000000B
MVF     MCKCN1, F, ACCE     ; Setup internal LPO as CPU clock
                            ; Turn off all external clock sources and internal HAO
CLRF    TMACN               ; Turn off clock counter modules,
CLRF    TMCCN               ;
CLRF    ADCCN7              ; Turn off ADC function module
CLRF    RMSCN               ; Turn off RMS function module.
CLRF    PWRCN               ; Turn off Power Source
CLRF    PWRCN2              ; Turn off Power Source
CLRF    SSPCON1             ; Turn off SPI function module.
CLRF    URCON               ; Turn off EUART function module.
                            ; Other modules if be turned off depend on applications.
BCF     TRISC1, 0, ACCE     ; Set PT1.0 as external interrupt awaken input
BSF     PT1PU, 0, ACCE
                            ; Configuration can base on both PT1 and PT2 at the same time.
BCF     INTF1, 0, ACCE
MVL     10000001B
MVF     INTE1, F, ACCE      ; Set external interrupt PT1.0 to enable
IDLE
NOP
```

Example 3-4 Idle Mode Example Program

```
CLRF    TMACN       ; Turn off Timer A clock counter module
CLRF    TMCCN       ; Turn off Timer C clock counter module
CLRF    PWMCN       ; Turn off PWM module
CLRF    ADCCN7      ; Turn off ADC function module
CLRF    RMSCN       ; Turn off RMS function module
CLRF    PWRCN       ; Turn off Power Source
CLRF    PWRCN2      ; Turn off Power Source
CLRF    SSPCON1     ; Turn off SPI module
CLRF    URCON       ; Turn off EUART module
CLRF    LVDCN1      ; Turn off LVD module
CLRF    LCDCN1      ; Turn off LCD module
CLRF    TRISC1      ; Set PT1.0 as external interrupt awaken input, PT1 PULL UP
SETF    PT1PU
CLRF    TRISC2      ; Set PT2 PULL UP
SETF    PT2PU
CLRF    TRISC3      ; Set PT3 PULL UP
MVL     11000010b
MVF     PT3PU,F,ACCE
BCF     INTF1, 0, ACCE
MVL     10000001B
MVF     INTE1, F, ACCE  ; Set external interrupt PT1.0 to enable
SLP
NOP
```

Example 3-5 Sleep Mode Example Program

4. RESET

HY12P Series reset circuit includes the following four events to trigger reset signal. Reset block diagram is as Figure 4-1.

- **BOR** Power interference reset.
- **RST** External reset input pin.
- **WDT** watch dog timer reset.
- **SKERR** Stack error reset (determined by users).

Operating Status Register:

PSTATUS PD[0],TO[0],IDLEB[0],BOR[0],SKERR[0]

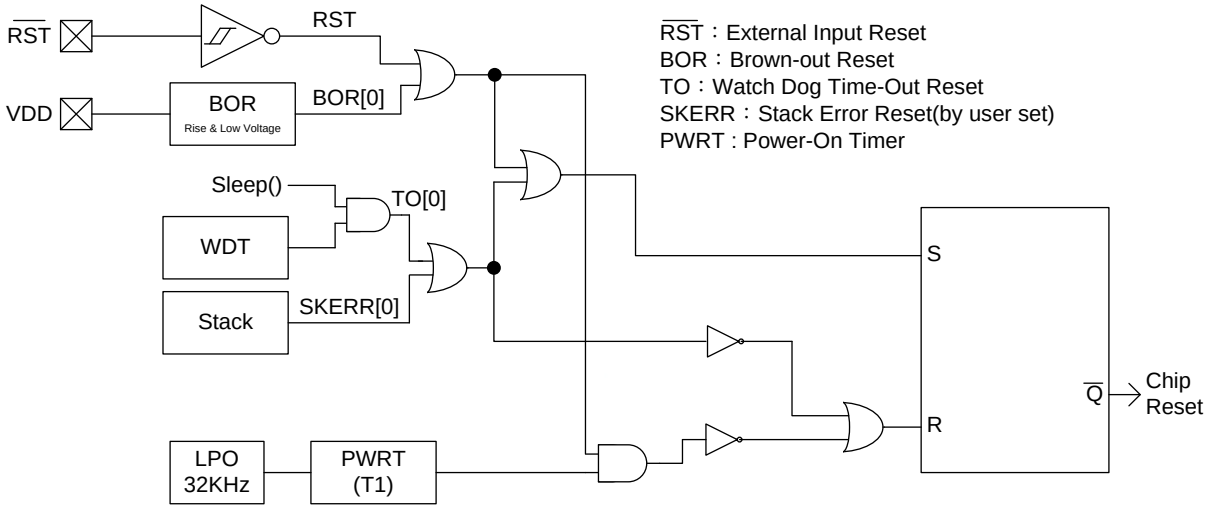


Figure 4-9 Reset Block Diagram

These reset events can be divided into hardware reset and software reset as depicted in Table 4-1. After CPU reset, program will start from 0x0000h.

Reset Type	Event	Symbol	Description
Hardware Reset	BOR RST	I-RESET	CPU restarts. It is until the end of internal oscillator counting, will CPU return to normal operating status.
Software Reset	WDT SKERR	A-RESET	Only partial registers will be cleared, CPU gets back to normal operating status soon.

Table 4-9Reset Level

4.1. Reset Events Description

4.1.1. BOR Power Interference Reset

When CPU is interfered during power on process or by external power, it will enter into normal operating voltage from abnormal over-low operating voltage. If CPU cannot enter into reset status in over-low operating voltage, it may result in CPU crash and abnormal peripheral circuit operations. Thus, when BOR circuit detects that operating voltage is interfered and the voltage lowers than default value, reset signal will be aroused and the IC will enter into restart status. Until operating voltage returned to default value, will the reset signal be released and the IC enters into normal run mode.

When BOR reset occurred, BOR[0] flag of register PSTATUS[7:0] will be configured as <1> to record the occurred event.

BOR circuit of HY12P Series will generate approximately lower than 1uA current consumption; there is no other program or configuration methods can shut it off.

4.1.2. RST External Reset Input

Reset signal will be produced when external RST pin voltage is lower than default value[5], at this time, the IC will enter into restart status. The IC will return to normal operating mode till RST voltage regain to default arguments.

4.1.3. Watch Dog timer Reset, WDT

The IC will enter into quick restart status when reset signal is produced after WDT operating mode ceased. As soon as WDT reset happens, TO [0] flag of register, PSTATUS [7:0] is set as <1> in order to record the occurred event.

Please be noticed that there are two kinds of signals which happened after WDT ceased. Reset signal is produced when the IC is under RUN Mode. If it is in IDLE Mode, CPU will be awakened by the signals generated by interrupt events. Detailed instruction description can be referred to in ***Watch Dog Timer, WDT*** chapter.

4.1.4. SKERR Stack Error Reset

When reset signal is generated from program stack overflow/underflow, the IC will access into quick restart status. For recording the occurred events, SKERR [0] flag of register, PSTATUS[7:0] is set as <1> while SKERR stack error reset happened. Detailed instruction description will be elaborated on ***Memory*** chapter.

[5] The pin includes two other different functions. One is that when RST input voltage conforms with V_{IU} specification, the IC will get into OTP programming mode, the other one is that when RST input voltage matches to V_{IL} specification, the IC will enter into current leakage detect mode.

4.2. Status Registers

IC operating status is displayed in reset register, PSTATUS [7:0]; the interrelation is indicated in Table 4-2.

"0"：not happened, "1"：happened, "u"：unchanged, "-"：unused

Name/Status	Address	7	6	5	4	3	2	1	0
PSTATUS	02CH	PD	TO	IDLEB	BOR	-	SKERR	-	-
Hardware Reset (A-RESET)	BOR	0	0	0	1	-	0	-	-
	RST	0	0	0	u	-	0	-	-
Software Reset (I-RESET)	WDT	u	1	u	u	-	u	-	-
	SKERR	u	u	u	u	-	1	-	-

Table 4-10 Interrelation of Reset Status Flags

4.2.1. Reset Status Sequence

Figure 4-2 presents the time frame from hardware reset signal happened to IC accesses into operating status. As for the time frame from different reset signals occurred to IC enters into operating status, please refer to Table 4-3.

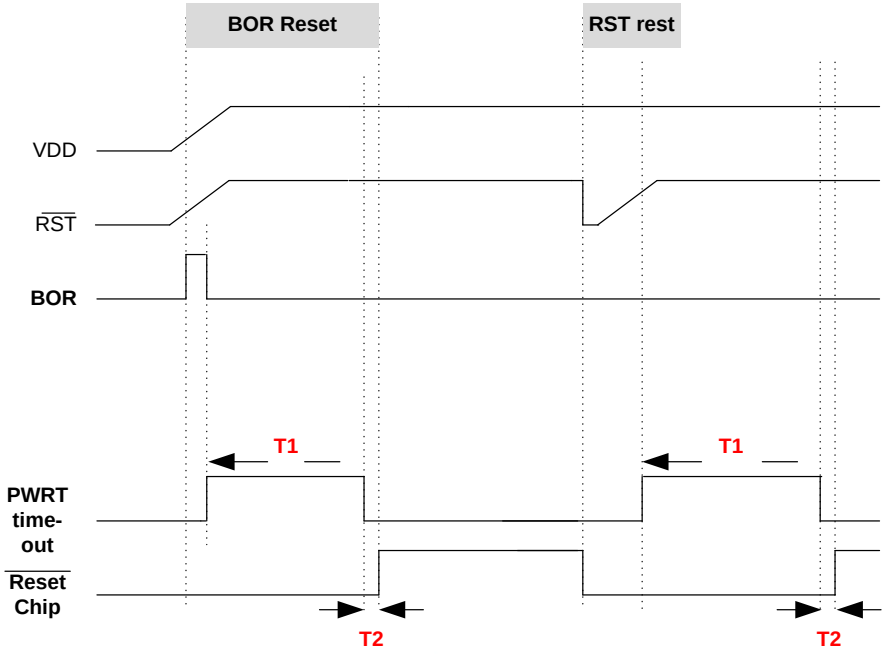


Figure 4-10 Reset Status Sequence

"-": no definition

Reset Signal	Delay Time			Operating Status		
	Symbol	T1[6]	T2[7]	Run	Idle	Sleep
BOR	t_{RST}	T1+T2		Valid	Valid	Valid
RST		T1+T2		Valid	Valid	Valid
WDT[8]	-	-		Valid	Valid	Invalid
SKERR	-	-		Valid	Invalid	Invalid

Table 4-11 Interrelation of Reset Status Delay Time and Operating Status

6 T1:2048 oscillation delay cycle, using LPO(32KHz) clock source.

7 T2:1024 oscillation delay cycle, using HAO(4MHz) clock source.

8 Two signals will be generated after WDT ceased, please refer to ***Watch Dog, WDT*** chapter for detailed instruction description.

4.2.2. Register Description-Reset Status

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1												
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition												
Address	File Name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	A-RESET	i-RESET	R/W
2CH	PSTATUS	PD	TO	IDLEB	BOR		SKERR			000d .0..	uduu .d..	rw0,rw0,rw0,rw0 -,rw0,-,-

Table 4-12 Reset Register

PSTATUS：Status Register

PD：Sleep Mode flag

1：Set up <1> to execute SLEEP instruction

0：Cleared through BOR, RST or instruction

TO：WDT flag

1：Set up <1>, when WDT ceased

0：Cleared through BOR, RST or instruction

IDLEB：Idle Mode flag

1：Set up <1> to execute IDLE instruction

0：Cleared through BOR, RST or instruction

BOR：Power interference reset flag

1：Set up <1> to operate BOR

0：Cleared through instruction

SKERR：Stack error reset flag

1：Stack error set up <1>

0：Cleared through BOR, RST or instruction

5. Interrupt

Interrupt involves interrupt starting controller, INTE and interrupt event flag, INTF. If interrupt event occurs while interrupt service established, PC will move to interrupt vector address, 0x0004h of program memory to execute interrupt service program.

Interrupt Control Register：

INTE1 GIE[0],TMCIE[0],TMAIE[0],WDTIE[0],E1IE[0],E0IE[0]

INTE2 TXIE[0],RCIE[0],RMSIE[0],LPFIE[0],AD1IE[0], CTIE[0]

INTE3 E24IE[0],E25E[0], E26IE[0],E27IE[0]

NTF1 TMCIF[0],TMAIF[0],WDTIF[0],E1IF[0],E0IF[0]

INTF2 TXIF[0],RCIF[0],RMSIF[0],LPFIF[0],AD1IF[0], CTIF[0]

INTF3 E24IF[0],E25F[0], E26IF[0],E27IF[0]

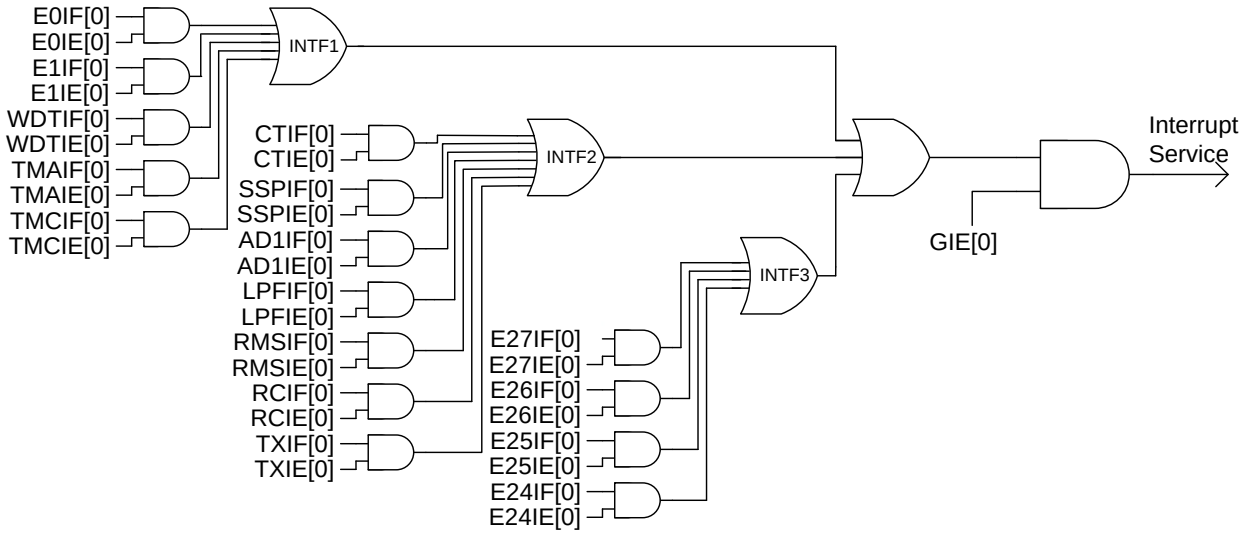


Figure 5-11 Interrupt Vector Block Diagram

There are two layers of interrupt service event controller. The top layer is global interrupt enable, GIE [0] and the bottom layer is the starting control bit for interrupt event.

- To initiate interrupt event, set up the control bit that relatives to interrupt control register, INTEx [7:0] to be <1>. Conversely, set up< 0> to close the interrupt event.
- To initiate interrupt service, set up the control bit, GIE [0] that relatives to interrupt control register INTE1 [7:0] to be <1>. Conversely, set up< 0> to close the interrupt service.

GIE [0] will automatically be set up as <0> when it enters into interrupt service vector. After interrupt service program has been completely executed and would like to return to interrupt occurred address, interrupt return instruction, RETI can be applied directly. At this time, GIE [0] will automatically be set up as <1>. Moreover, if return instruction RET is conducted, GIE[0] status will remain as 0.

5.1. Register Description-Interrupt

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1												
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition												
Address	**File Name**	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**	**A-RESET**	**i-RESET**	**R/W**
23H	INTE1	GIE		TMCIE		TMAIE	WDTIE	E1IE	E0IE	0.0. 0000	0.0. 0000	*,-,*,- *,*,*,*
24H	INTE2	TXIE	RCIE	RMSIE	LPFIE	AD1IE		CTIE		0000 000.	0000 000.	*,*,*,*,*,*,*,-
25H	INTE3	E24IE	E25IE	E26IE	E27IE					0000	0000	*,*,*,*,-,-,-,-
26H	INTF1			TMCIF		TMAIF	WDTIF	E1IF	E0IF	..0. 0000	..0. 0000	-,-,*,- *,*,*,*
27H	INTF2	TXIF	RCIF	RMSIF	LPFIF	AD1IF		CTIF		0000 000.	0000 000.	*,*,*,*,*,*,*,-
28H	INTF3	E24IF	E25IF	E26IF	E27IF					0000	0000	*,*,*,*,-,-,-,-

Table 5-13 Interrupt Registers

INTE1：Interrupt Starting Control Register 1

GIE[0]：Global interrupt enable

1：Start

0：Shutoff

TMCIE[0]：Timer-C interrupt event starting controller

1：Start (Timer C, TMC)

0：Shutoff

TMAIE[0]：Timer-A interrupt event starting controller

1：Start (Timer A, TMA)

0：Shutoff

WDTIE[0]：Watch Dog interrupt event starting controller

1：Start (WDT)

0：Shutoff

E1IE[0]：Interrupt event starting controller of input pin 1

1：Start (external input pin, PT1.1)

0：Shutoff

E0IE[0]：Interrupt event starting controller of input pin 0

1：Start (external input pin, PT1.0)

0：Shutoff

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1												
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition												
Address	File Name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	A-RESET	i-RESET	R/W
23H	INTE1	GIE		TMCIE		TMAIE	WDTIE	E1IE	E0IE	0.0. 0000	0.0. 0000	*,-,*,- *,*,*,*
24H	INTE2	TXIE	RCIE	RMSIE	LPFIE	AD1IE		CTIE		0000 000.	0000 000.	*,*,*,*,*,*,*,-
25H	INTE3	E24IE	E25IE	E26IE	E27IE					0000	0000	*,*,*,*,-,-,-,-
26H	INTF1			TMCIF		TMAIF	WDTIF	E1IF	E0IF	..0. 0000	..0. 0000	-,-,*,- *,*,*,*
27H	INTF2	TXIF	RCIF	RMSIF	LPFIF	AD1IF		CTIF		0000 000.	0000 000.	*,*,*,*,*,*,*,-
28H	INTF3	E24IF	E25IF	E26IF	E27IF					0000	0000	*,*,*,*,-,-,-,-

INTE2：Interrupt Starting Control Register 2

TXIE[0]：TX interrupt event starting controller

1：Start (Communication interface, EUART)

0：Shutoff

RCIE[0]：RC interrupt event starting controller

1：Start (Communication interface, EUART)

0：Shutoff

RMSIE[0]：True RMS interrupt event starting controller

1：Start

0：Shutoff

LPFIE[0]：Low Pass Filter interrupt event starting controller

1：Start

0：Shutoff

AD1IE[0]：ADC interrupt event starting controller

1：Start

0：Shutoff

CTIE[0]：CONTER interrupt event starting controller

1：Start

0：Shutoff

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1												
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition												
Address	File Name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	A-RESET	i-RESET	R/W
23H	INTE1	GIE		TMCIE		TMAIE	WDTIE	E1IE	E0IE	0.0. 0000	0.0. 0000	*,-,*,- *,*,*,*
24H	INTE2	TXIE	RCIE	RMSIE	LPFIE	AD1IE		CTIE		0000 000.	0000 000.	*,*,*,*,*,*,*,-
25H	INTE3	E24IE	E25IE	E26IE	E27IE					0000	0000	*,*,*,*,-,-,-,-
26H	INTF1			TMCIF		TMAIF	WDTIF	E1IF	E0IF	..0. 0000	..0. 0000	-,-,*,- *,*,*,*
27H	INTF2	TXIF	RCIF	RMSIF	LPFIF	AD1IF		CTIF		0000 000.	0000 000.	*,*,*,*,*,*,*,-
28H	INTF3	E24IF	E25IF	E26IF	E27IF					0000	0000	*,*,*,*,-,-,-,-

INTE3：Interrupt Starting Control Register 3

E24IE[0]：Interrupt event starting controller of input pin 4

1：Start (External input pin, PT2.4)

0：Shutoff

E25IE[0]：Interrupt event starting controller of input pin 5

1：Start (External input pin, PT2.5)

0：Shutoff

E26IE[0]：Interrupt event starting controller of input pin 6

1：Start (External input pin, PT2.6)

0：Shutoff

E27IE[0]：Interrupt event starting controller of input pin 7

1：Start (External input pin, PT2.7)

0：Shutoff

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1												
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition												
Address	File Name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	A-RESET	i-RESET	R/W
23H	INTE1	GIE		TMCIE		TMAIE	WDTIE	E1IE	E0IE	0.0. 0000	0.0. 0000	*,-,*,- *,*,*,*
24H	INTE2	TXIE	RCIE	RMSIE	LPFIE	AD1IE		CTIE		0000 000.	0000 000.	*,*,*,*,*,*,*,-
25H	INTE3	E24IE	E25IE	E26IE	E27IE					0000	0000	*,*,*,*,-,-,-,-
26H	INTF1			TMCIF		TMAIF	WDTIF	E1IF	E0IF	..0. 0000	..0. 0000	-,-,*,- *,*,*,*
27H	INTF2	TXIF	RCIF	RMSIF	LPFIF	AD1IF		CTIF		0000 000.	0000 000.	*,*,*,*,*,*,*,-
28H	INTF3	E24IF	E25IF	E26IF	E27IF					0000	0000	*,*,*,*,-,-,-,-

INTF1：Interrupt Event Flag Register 1

TMCIF[0]：Timer-C interrupt event flag

1：Happened (Timer C, TMC)

0：Not happened

TMAIF[0]：Timer-A interrupt event flag

1：Happened (Timer A, TMA)

0：Not happened

WDTIF[0]：Watch Dog interrupt event flag

1：Happened (WDT)

0：Not happened

E1IF[0]：Interrupt event flag of input pin 1

1：Happened (External input pin, PT1.1)

0：Not happened

E0IF[0]：Interrupt event flag of input pin 0

1：Happened (External input pin, PT1.0)

0：Not happened

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1												
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition												
Address	File Name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	A-RESET	i-RESET	R/W
23H	INTE1	GIE		TMCIE		TMAIE	WDTIE	E1IE	E0IE	0.0. 0000	0.0. 0000	*,-,*,- *,*,*,*
24H	INTE2	TXIE	RCIE	RMSIE	LPFIE	AD1IE		CTIE		0000 000.	0000 000.	*,*,*,*,*,*,*,-
25H	INTE3	E24IE	E25IE	E26IE	E27IE					0000	0000	*,*,*,*,-,-,-,-
26H	INTF1			TMCIF		TMAIF	WDTIF	E1IF	E0IF	..0. 0000	..0. 0000	-,-,*,- *,*,*,*
27H	INTF2	TXIF	RCIF	RMSIF	LPFIF	AD1IF		CTIF		0000 000.	0000 000.	*,*,*,*,*,*,*,-
28H	INTF3	E24IF	E25IF	E26IF	E27IF					0000	0000	*,*,*,*,-,-,-,-

INTF2：Interrupt Event Flag Register 2

TXIF[0]：TX interrupt event flag

1：Happened (Communication interface, EUART)

0：Not happened

RCIF[0]：RC interrupt event flag

1：Happened (Communication interface, EUART)

0：Not happened

RMSIF[0]：True RMS interrupt event flag

1：Happened

0：Not happened

LPFIF[0]：Low Pass Filter interrupt event flag

1：Happened

0：Not happened

AD1IF[0]：ADC interrupt event flag

1：Happened

0：Not happened

CTIF[0]：COUNTER interrupt event flag

1：Happened

0：Not happened

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1												
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition												
Address	File Name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	A-RESET	i-RESET	R/W
23H	INTE1	GIE		TMCIE		TMAIE	WDTIE	E1IE	E0IE	0.0. 0000	0.0. 0000	*,-,*,- *,*,*,*
24H	INTE2	TXIE	RCIE	RMSIE	LPFIE	AD1IE		CTIE		0000 000.	0000 000.	*,*,*,*,*,*,*,-
25H	INTE3	E24IE	E25IE	E26IE	E27IE					0000	0000	*,*,*,*,-,-,-,-
26H	INTF1			TMCIF		TMAIF	WDTIF	E1IF	E0IF	..0. 0000	..0. 0000	-,-,*,- *,*,*,*
27H	INTF2	TXIF	RCIF	RMSIF	LPFIF	AD1IF		CTIF		0000 000.	0000 000.	*,*,*,*,*,*,*,-
28H	INTF3	E24IF	E25IF	E26IF	E27IF					0000	0000	*,*,*,*,-,-,-,-

INTF3：Interrupt Event Flag Register 3

E24IF[0]：Interrupt event flag of input pin 4

1：Happened (External input pin, PT2.4)

0：Not happened

E25IF[0]：Interrupt event flag of input pin 5

1：Happened (External input pin, PT2.5)

0：Not happened

E26IF[0]：Interrupt event flag of input pin 6

1：Happened (External input pin, PT2.6)

0：Not happened

E27IF[0]：Interrupt event flag of input pin 7

1：Happened (External input pin, PT2.7)

0：Not happened

6. Hardware Multiplier

H08A instruction set has 8x8 hardware multiplier processing instruction "MULF and MULL". The operation outcome of 8x8 hardware multiplier will store at multiplier register PRODH[7:0] and PRODL[7:0], and will not change any sign of STATUS[7:0] status register. Users must be cautioned that PRODH[7:0] and PRODL[7:0] are read only registers.

Hardware multiplier can conduct signed value and unsigned value operation, as Example 6-1 and Example 6-2 illustrates.

```
Ex 1：V1 x V2 = V
MVL     V1
MVF     BUF0,1,0      ; Put V1 value in register BUF0 of memory segment 0
MVL     V2            ; Put V2 value to register W
MULF    BUF0,0        ; Execute V1 x V2 and place the result to PRODH/PRODL
```

Example 6-3 Unsigned Value Operation

```
Ex 2：N1 x N2 = N ,s=7b
MVL     N1            ; Put N1 value to register W
MVF     BUF0,1,0      ; Put N1 value in register BUF0 of memory segment 0
MVL     N2            ; Put N2 value to register W
MVF     BUF1,1,0      ; Put N2 value to register BUF1
MULF    BUF0,0        ; Execute V1 x V2 and place the result to PRODH/L
MVFF    PRODH,SWP     ; Put register PRODH value to register SWP
BTSZ    BUF0,s        ; Judge N1, if it is negative then
SUBF    SWP,1,0       ; Place SWP – N2 to register SWP
MVF     BUF0,0,0      ; Put N1 value to register W
BTSZ    BUF1,s        ; Judge N2, if it is negative then
SUBF    SWP,1,0       ;Place SWP – N1 to SWP. After operation, N = SWP/PRODL
; ----------------------------------------------------------------------------------
; N1=07Fh,N2=0FFh, after multiplier operation, obtained PRODH/L = 7E81h
; Determine whether N1 is minus, if so, then set PRODH – N2
; Determine whether N2 is minus, if so, then set PRODH – N1
; After operation, signed N value will be acquired.
; 7Fh x FFh     = 7Fh x ( 0FFh – 100h )
;               = 7Fh x 0FFh – 7Fh x 100h
;               = 7E81h – 7F00h
;               = FF81h
```

Example 6-4 Signed Value Operation

7. Input/Output Port, I/O

Every 8 I/O is a port that can be utilized as digital input/output and analog signal measurement channel. Every port is controlled by a set of registers. Small discrepancies of I/O register composition exist in different products.

I/O Related Registers:

PT	PT1[7:0], PT2[7:0], PT3[7:5]
TRISC	TC1[7:0], TC2[7:0], TC3[7:5],
PTDA	DA2[7:3], DA3[5]
PTPU	PU1[7:0], PU2[7:0], PU3[7:5]
PTM	PM1[7:4], INTEG1[1:0], INTEG0[1:0], PM2[1:0]

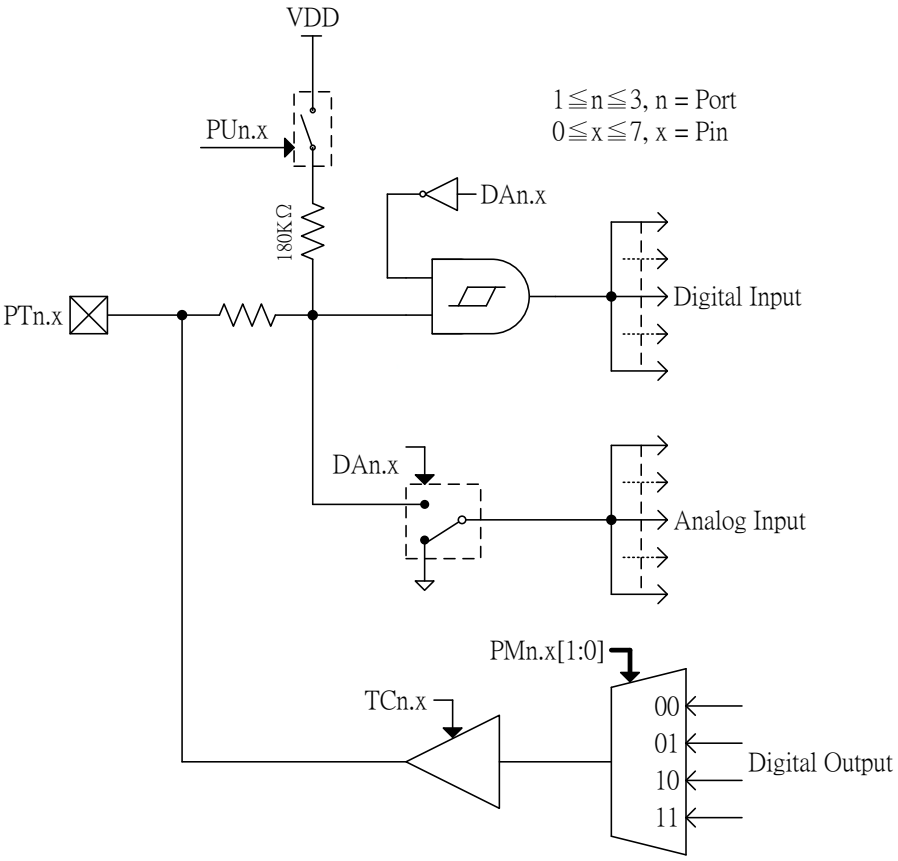


Figure 7-12 Block Diagram of I/O Frame

7.1. PORT Related Register Introduction

PORT mainly offers digital or analog signal I/O pin.

7.1.1. PT Status Control Register

When I/O is set as input, in corresponding register address, current I/O status can be read. If the value is 1, the I/O inputs high potential, if the value is 0 and the I/O inputs low potential.

When I/O is set as output, the correlative register site can control output status. If <1> is set, the I/O outputs high potential. If the value is set as <0>, the I/O outputs low potential.

7.1.2. TRISC Input/Output Control Register

Selecting I/O to be input or output, set <1> and the I/O is in output status, set <0> and the I/O is in input status. If I/O is configured as input status, an explicit input potential must be made once the IC enters into sleep mode so as to avoid IC leakage status.

7.1.3. PTDA Digital or Analog Input Control Register

Selecting I/O to be analog/digital input status, set <1> will be analog input; set <0> will be digital input. Other I/O related registers configuration status must also be taken into accounts as to avoid interference of digital/analog signal.

7.1.4. PTPU Pull-Up Resistor Control Register

Configure the register as <1>; the I/O is enabled with the pull-up resistor function. Configure the register set <0>, the function is disabled. Before the IC gets into sleep mode, if I/O is configured as digital input status, the external circuit connected way will cause I/O floating phenomenon. At this time, the pull-up resistor may be enabled as to avoid current leakage.

7.1.5. PTM Digital Output Mode Select Register

I/O output mode selector, PMn.x[1:0] ($1 \leqq n \leqq 2, 0 \leqq x \leqq 7$) can configure I/O output signal. Partial I/O equips at least 1 kind of digital peripheral circuit output signals. Thus, correctly configures PMn.x[1:0] can get expected output signal.

7.1.6. PTINT Interrupt Signal Generated Conditions

Diverse I/O external input potential produces different interrupt signals. Potential change conditions can be separated into rising edge (0→1) change, falling edge (1→0) change and potential conversion (0→1 or 1→0) change. Which PT2[7:4] was fixed to potential conversion interrupt.

When detection conditions are potential conversion changes, the status of PT1[1:0] or PT2[7:4] have to be read first then active potential conversion (0→1 or 1→0) change. When the result of PT1[1:0] status is differ to before, the interrupt signals will be appeared.

Regarding to the configuration of potential conversion (0→1), the pins, PT1.0/PT1.1 have to be configured as low voltage and implement read the status of PT1[1:0] first then start INTEGx[1:0]=10b or 11b mode ($0 \leqq x \leqq 1$). When the pins, PT1.0/PT1.1 are detected to rising edge, the interrupt signals will be appeared.

Regarding to the configuration of potential conversion (1→0), the pins, PT1.0/PT1.1 have to be configured as high voltage and implement read the status of PT[1:0] first then start INTEGx[1:0]=10b or 11b mode(0≦x≦1). When the pins, PT1.0/PT1.1 are detected to falling edge, the interrupt signals will be appeared.

7.2. Buzzer

BZ can produce several different frequencies to drive external buzzer. BZ operating frequency prescaler, BZS[2:0] can set diverse output frequency.

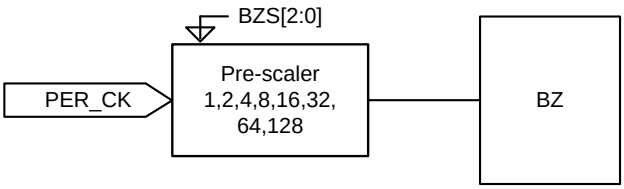


Figure 7-13 BZ Block Diagram

7.3. Input/Output Port 1, I/O Port1

"i":input, "o":output, "a":analog, "c":cmos i/o, "s":schmitt trigger, "x":undefined, "p":power

Pin Name	Design		Register Configuration			Description
	Type	Buffer	TC[0]	DA[0]	PM[0]	
PT1.0	I/O	S	0/1	X	X	Digital input/output pin
INT0	I	S	0	X	X	External interrupt source
PSCK	I	S	0	X	X	OTP programming interface PSCK pin
PT1.1	I/O	S	0/1	X	X	Digital input/output pin
INT1	I	S	0	X	X	External interrupt source
PSDI	I	S	0	X	X	OTP programming PSDI pin
PT1.2	I/O	S	0/1	X	X	Digital input/output pin
PT1.3	I	S	0/1	X	X	Digital input/output pin
TST	I	S	0	X	X	Reserved by manufacturer
RC	I	S	0	X	X	EUART communication interface RC pin
PT1.4	I/O	S	0/1	X	0	Digital input/output pin
TX	O	C	1	X	1	EUART communication interface TX pin
PT1.5	I/O	S	0/1	X	0	Digital input/output pin
PT1.6	I/O	S	0/1	X	0	Digital input/output pin
PT1.7	I/O	S	0/1	X	0	Digital input/output pin
BZ	O	C	1	X	1	Buzzer output pin
PSDO	O	C	1	X	0	OTP programming pin, PSDO
Set at PT1PU[7:0]=00h						

Table 7-14 PORT1 Function

7.3.1. Register Description –PORT1

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1												
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition												
Address	File Name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	A-RESET	i-RESET	R/W
23H	INTE1	GIE						E1IE	E0IE	0.0. 0000	0.0. 0000	*,-,*,- *,*,*
26H	INTF1							E1IF	E0IF	..0. 0000	..0. 0000	-,-,*,- *,*,*
31H	MCKCN2						BZS[2:0]			0000 0000	0000 0000	*,*,*,*,*,*,*
4DH	PT1	PT1.7	PT1.6	PT1.5	PT1.4	PT1.3	PT1.2	PT1.1	PT1.0	xxxx xxxx	uuuu uuuu	*,*,* r,r,r,r
4EH	TRISC1	TC1.7	TC1.6	TC1.5	TC1.4	TC1.3	TC1.2	TC1.1	TC1.0	0000 0000	0000 0000	*,*,*,*,*,*,*,*
4FH	PT1PU	PU1.7	PU1.6	PU1.5	PU1.4	PU1.3	PU1.2	PU1.1	PU1.0	0000 0000	0000 0000	*,*,*,*,*,*,*,*
50H	PT1M1	PM1.7	PM1.6	PM1.5	PM1.4	INTEG1[1:0]		INTEG0[1:0]		0000 0000	0000 0000	*,*,*,*,*,*,*,*

Table 7-15 PORT1 Control Register

INTE1/INTF1：Please refer to ***Interrupt*** chapter.

MCKCN2：Please refer to ***Oscillator, Clock Sources and Power Managed Modes*** chapter

PT1：PORT1Status Control Register

PT1.x：External pin flag/control bit（0≦x≦7）

1：High potential flag/High potential output

0：Low potential flag/Low potential output

TRISC1：Input/Output Control Register

TC1.x：External input/output pin control bit（0≦x≦7）

1：Output

0：Input

PT1PU：Pull-Up Resistor Control Register

PU1.x：External pull-up resistor pin control bit（0≦x≦7）

1：Start

0：Shutoff

PT1M1：Digital Output mode Select Register

PM1.7：PT1.7 Buzzer output control bit

1：Start

0：Shutoff

PM1.4：PT1.4 EUART TX output control bit

1：Start

0：Shutoff

INTEG1[1:0]：PT1.1 Interrupt signal generated conditions

11：Potential conversion (0→1 or 1→0)

10：Potential conversion (0→1 or 1→0)

01：Rising edge (0→1)

00：Falling edge (1→0)

INTEG0[1:0]：PT1.0 Interrupt signal generated conditions

11：Potential conversion (0→1 or 1→0)

10：Potential conversion (0→1 or 1→0)

01：Rising edge (0→1)

00：Falling edge (1→0)

7.4. Input/Output Port 2, I/O Port2

"i": input, "o": output, "a": analog, "c": cmos i/o, "s": schmitt trigger, "x": undefined, "p": power

Pin Name	Design		Register Configuration			Description
	Type	Buffer	TC[0]	DA[0]	PM[1:0]	
PT2.0	I/O	S	0/1	X	X	Digital input/output pin
XTO	O	A	0	X	X	External oscillator pin
PT2.1	I/O	S	0/1	X	X	Digital input/output pin
XTI	I	A	0	X	X	External oscillator pin
PT2.2	I/O	C	0/1	X	00	Digital input/output pin
PFD	O	C	1	X	01	PFD output pin
PWM	O	C	1	X	10	PWM output pin
PT2.3	I/O	S	0/1	0	X	Digital input/output pin
TMCKI	I	S	0	0	X	TIMER-C clock source input pin
LVDIN	I	A	0	1	X	LVD External signal input pin
PT2.4	I/O	S	0/1	0	X	Digital input/output pin
INT24	I	S	0	0	X	Interrupt source input pin
CMP0	I	A	0	1	X	Comparator input interface pin
PT2.5	I/O	S	0/1	0	X	Digital input/output pin
INT25	I	S	0	0	X	Interrupt source input pin
CMP1	I	A	0	1	X	Comparator input interface pin
PT2.6	I/O	S	0/1	0	X	Digital input/output pin
INT26	I	S	0	0	X	Interrupt source input pin
CMP2	I	A	0	1	X	Comparator input interface pin
PT2.7	I/O	S	0/1	0	X	Digital input/output pin
INT27	I	S	0	0	X	Interrupt source input pin
CMP3	I	A	0	1	X	Comparator input interface pin
Set at PT2PU[7:0]=00h						

Table 7-16 PORT2 Function

7.4.1. Register Description-PORT2

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1												
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition												
Address	File Name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	A-RESET	i-RESET	R/W
51H	PT2	PT2.7	PT2.6	PT2.5	PT2.4	PT2.3	PT2.2	PT2.1	PT2.0	xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
52H	TRISC2	TC2.7	TC2.6	TC2.5	TC2.4	TC2.3	TC2.2	TC2.1	TC2.0	0000 0000	0000 0000	*,*,*,*,*,*,*,*
53H	PT2DA	DA2.7	DA2.6	DA2.5	DA2.4	DA2.3		PM2.2[1:0]		0000 0.00	0000 0.00	*,*,*,*,-,*,*
54H	PT2PU	PU2.7	PU2.6	PU2.5	PU2.4	PU2.3	PU2.2	PU2.1	PU2.0	0000 0000	0000 0000	*,*,*,*,*,*,*,*

Table 7-17 PORT2 Control Register

PT2：PORT2 Status Control Register

PT2.x：External pin flag/control bit（0≦x≦7）

1：High potential flag/High potential output

0：Low potential flag/Low potential output

TRISC2：Input/Output Control Register

TC2.x：External pin input/output control bit（0≦x≦7）

1：Output

0：Input

PT2DA：Digital/Analog Input Control Register

DA2.x：External analog/digital input signal pin control bit（3≦x≦7）

1：Analog

0：Digital

PM2.2[1:0]：PT2.2 output mode control bit

11：Reserved

10：Timer C PWM output starts.

01：Timer C PFD output starts.

00：Shutoff

PT2PU：Pull-Up Resistor Control Register

PU2.x：External pull-up resistor pin control bi（0≦x≦7）

1：Start

0：Shutoff

7.5. Input/Output Port 3 , I/O Port3

"i":input, "o":output, "a":analog, "c":cmos i/o, "s": schmitt trigger, "x":undefined, "p":power,

Pin Name	Design		Register Configuration			Description
	Type	Buffer	TC[0]	DA[0]	PM[0]	
PT3.5	I/O	C	0/1	1	X	Digital input/output pin
PB5	I	A	0	0	X	Analog channels input pin
PT3.6	I/O	C	0/1	X	X	Digital input/output pin
CNT	I	A	0	X	X	Frequency counting input pin
PT3.7	I/O	C	0/1	X	0	Digital input/output pin
CMPO	O	C	1	X	1	Comparator output pin
Set at PT3PU[7:5]=000b						

Table 7-18 PORT3 Function

7.5.1. Register Description -PORT3

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1												
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition												
Address	**File Name**	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**	**A-RESET**	**i-RESET**	**R/W**
55H	PT3	PT3.7	PT3.6	PT3.5		TC3.7	TC3.6	TC3.5		xxx. 000.	uuu. 000.	*,*,*,- *,*,*,-
56H	PT3PU	PU3.7	PU3.6	PU3.5		PM3.7		DA3.5		000. 0.0.	000. 0.0.	*,*,*,- *,-,*,-

Table 7-19 PORT3 Control Register

PT3：PORT3 Status Control Register

PT3.x：External pin flag/control bit（5≦x≦7）

1：High potential flag/High potential output

0：Low potential flag/Low potential output

TC3.x：External pin input/output control bit（5≦x≦7）

1：Output

0：Input

PT3PU：Pull-Up Resistor Control Register

PU3.x：External pull-up resistor pin control bit（5≦x≦7）

1：Start

0：Shutoff

PM3.7：PT3.7 CMPO output control bit

1：Start

0：Shutoff

DA3.5：PT3.5 External analog/digital input signal pin control bit

~~0：Analog~~1：Digital

0：Analog

8. Watch Dog Timer, WDT

Watch dog timer (WDT) is IC guardian that mainly being applied in awakening event.

- Run Mode
  - WDT overflow generates reset signal, IC will be restarted.
  - Timer can be cleared zero through software
- Sleep Mode
  - Shutoff WDT, it is inactive
- Idle Mode
  - WDT overflow generates interrupt event to awake IC

WDT Related Registers:

TMACN	ENWDT[0],WDTS[2:0]
PSTATUS	TO[0]
INTF1	WDTIF[0]
INTE1	WDTIE[0]

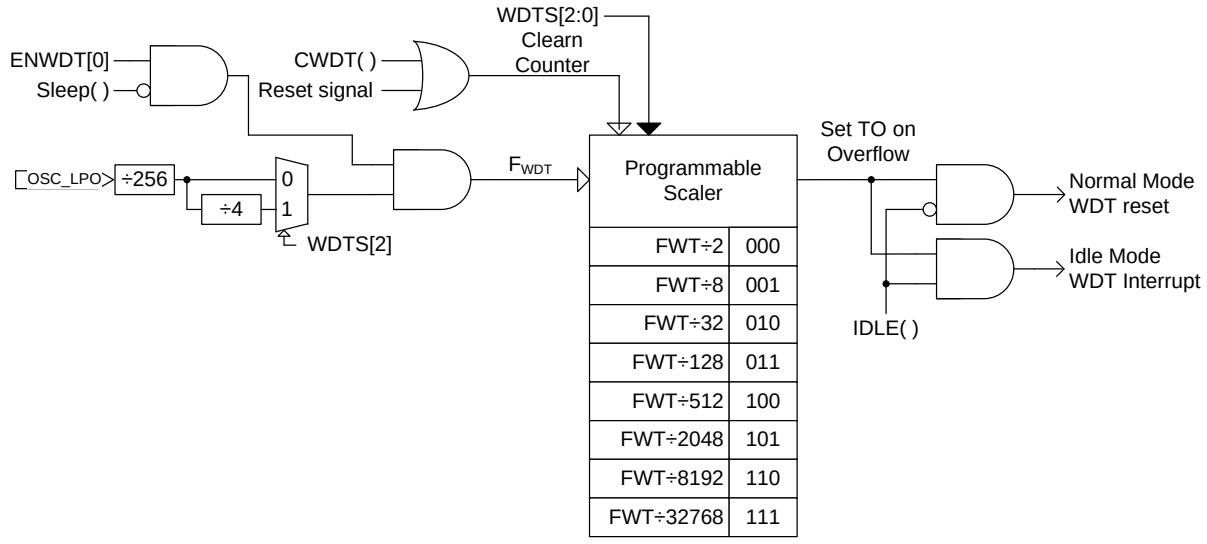


Figure 8-14 WDT Block Diagram

8.1. WDT Manual

8.1.1. WDT Initial Configuration

WDT counting controller, WDTS[2:0] can decide WDT counter operating frequency F_{WDT} and overflow. WDT reset signal TO or interrupt event, WDTIF[9] will be produced after the counter being overflowed.

[9] WDT uses internal clock source, LPO. It can be operated under Run Mode and Idle Mode. Under Run Mode, WDT can be zeroed by software as to avoid IC reset. However, WDT cannot be zeroed by any means under Idle Mode.

8.1.2. WDT Interrupt Event Service

WDT interrupt event can only be operated while the IC is in Idle Mode. When WDTIE[0] and GIE[0] is configured as <1>, after WDT overflowed, interrupt event, WDTIF[0] is set as <1> and PC will jump to interrupt vector address, <0>x0004h. By contrast, WDTIE[0] and GIE[0] is configured as <0>, no interrupt will take place.

8.1.3. WDT Initiation

WDT must be started when the IC is under Run Mode. That is to say, WDT starting controller, ENWDT[0] must be set as <1> to initiate WDT. Once the WDT is started, ENWDT[0] cannot be configured as <0> by software. Moreover, in Idle Mode, hardware will automatically set ENWDT[0] as <0> if WDT is awakened by interrupt even

```
MVL 00Ah
MVF TMACN,1,0       ; Start WDT and configure WDTS[2:0] = 010b
….                  ; WDT overflow time is around 4Hz
CWDT                ; Zero WDT
….
```

Example 8-5 WDT Reset Event Example Program

```
MVL   00Ah
MVF   TMACN,1,0         ; Start WDT and configure WDTS[2:0] = 010b
….                      ; WDT overflow time is around 4Hz
CWDT                    ; Zero WDT
IDLE                    ; Enter into Idle Mode.
….
Idle Interrupt :        ; Interrupt Service Program
BCF   INTF1,WDTIF,0     ; Clear WDT Interrupt event flag
….
RETI                    ; Interrupt service return
```

Example 8-6 WDT Interrupt Event Example Program

8.2. Register Description-WDT

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1												
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition												
Address	**File Name**	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**	**A-RESET**	**i-RESET**	**R/W**
23H	INTE1	GIE					WDTIE			0.0. 0000	0.0. 0000	*,-,*,- *,*,*,*
26H	INTF1						WDTIF			..0. 0000	..0. 0000	-,-,*,- *,*,*,*
2CH	PSTATUS		TO							000d .0..	uduu .d..	rw0,rw0,rw0,rw0 -,rw0,-,-
32H	TMACN					ENWDT	WDTS[2:0]			0000 0000	0000 0000	*,*,*,* w1,*,*,*

Table 8-20 WDT Control Register

INTE1/INTF1：Please refer to ***interrupt*** Chapter

PSTATUS：Please refer to ***RESET*** Chapter

TMACN：Timer-A Control Register

ENWDT：WDT Starting Controller

1：Start

0：Shutoff；（cannot be configured <0> through software）

WDTS[2:0]：Configure WDT overflow time

111：F_{WDT}/32768

110：F_{WDT}/8192

101：F_{WDT}/2048

100：F_{WDT}/512

011：F_{WDT}/128

010：F_{WDT}/32

001：F_{WDT}/8

000：F_{WDT}/2

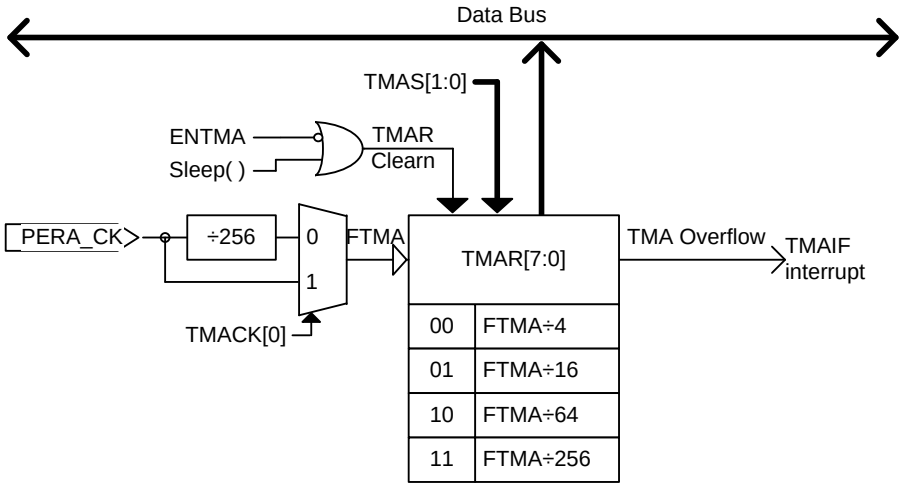
9. Timer-A

Timer-A is designed in 8-bit frame. TMA can function in Run Mode and Idle Mode.

- Ascending counter
- 4-step overflow value select
- Overflow generated interrupt event
- Counter value is readable

TMA Registers:

TMACN	ENTMA[0],TMACK[0],TMAS[1:0]
TMAR	TMAR[7:0]
INTE1	TMAIE[0]
INTF1	TMAIF[0]



Configure TMAS[1:0]=00b, when TMAR[7:0]=00000100b, first interrupt occurs,
The next time interrupt occurs TMAR[7:0]=00001000b。
So, every time interrupt occurs TMAR[7:0]=TMAR[7:0]+4.

Configure TMAS[1:0]=10b, when TMAR[7:0]=01000000b, first interrupt occurs,
The next time interrupt occurs TMAR[7:0]=10000000b。
So, every time interrupt occurs TMAR[7:0]=TMAR[7:0]+64.

Figure 9-15 Timer-A Block Diagram

9.1. TMA Manual

9.1.1. TMA Initial Configuration

TMA operating frequency is provided by PERA_CK, by configuring TMACK [0] can implement PERA_CK pre-frequency divided, F_{TMA} operating frequency is PERA_CK/256 or PERA_CK.

TMAR[7:0] is the counter of TMA, overflow or interrupt event TMAIF[0] time can be configured separately through TMAS[1:0]. TMAS[1:0] can configure TMAR[7:0] to generate overflow in +4, +16, +64 or +256[10] 。

9.1.2. TMA Interrupt Event Service

After TMA counter TMAR[7:0] overflowed, interrupt event will be produced and TMAIF[0] will be configured as <1>. If interrupt service is needed at this stage, TMAIE[0] and GIE[0]must be configured as <1>.

- For example, if TMAS[1:0] were configured as <00>, TMAR[7:0] value will change from 00000011b to 00000100b, interrupt event will be generated at this point. When next interrupt event occurred to TMAR[7:0], it will change from 00000111b to 00001000b. If TMAS[1:0] were configured as <10>, TMAR[7:0] value will change from 00111111b to 01000000b, interrupt event will be generated at this point. Next time when interrupt event occurred, TMAR[7:0] value will change from 01111111b to 10000000b.

9.1.3. TMA Initiation

ENTMA[0] is set as <1> to start TMA, TMAR[7:0] begin counting. If it is configured as <0>, TMA will be closed and TMAR[7:0] value will be set as zero.

```
BCF   INTF1,TMAIF,0     ; Clear TMAIF Flag
MVL   088h
MVF   INTE1,1,0         ; Configure Timer A Interrupt Service
MVL   0D0h              ; Start Timer A and configure operating frequency as PERA_CK
MVF   TMACN,1,0         ; Configure TMAS[1:0] =01b, the frequency of TMAR overflow is:
                        ; PERA_CK/16 Hz. This is to say, every generated interrupt event is: 1/ (PERA_CK/16) second
…
TMA Interrupt :         ; Timer A interrupt event service program
BCF   INTF1,TMAIF,0     ; Clear TMA interrupt event flag and TMAR=TMAR+16. Notice: every time TMAR overflowed
                        ; Whether interrupt event service is started TMAR=TMAR+16
RETI                    ; Interrupt service return
```

Example 9-7 TM A Interrupt Event Example Program

[10] During counting, TMAS[1:0] overflow generated time may lead to TMA mis-action, users must use with extra attention.

9.2. Register Description -TMA

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1												
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition												
Address	**File Name**	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**	**A-RESET**	**i-RESET**	**R/W**
23H	INTE1	GIE				TMAIE				0.0. 0000	0.0. 0000	*,-,*,- *,*,*,*
26H	INTF1					TMAIF				..0. 0000	..0. 0000	-,-,*,- *,*,*,*
32H	TMACN	ENTMA	TMACK	TMAS[1:0]						0000 0000	0000 0000	*,*,*,* w1,*,*,*
33H	TMAR	TimerA data register								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r

Table 9-21 TMA Control Register

INTE1/INTF1：Please refer to ***interrupt*** Chapter

TMACN：Timer-A Control Register

ENTMA：Timer-A starting controller

1：Start

0：Shutoff；Zero counters

TMACK：Timer-A operating frequency selector

1：F_{TMA}=PERA_CK

0：F_{TMA}=PERA_CK / 256

TMAS[1:0]：Timer-A overflow controller

11：F_{TMA} / 256；Every overflow occurs interrupt event, TMAR[7:0]=TMAR[7:0]+256

10：F_{TMA} / 64；Every overflow occurs interrupt event, TMAR[7:0]=TMAR[7:0]+64

01：F_{TMA} / 16；Every overflow occurs interrupt event, TMAR[7:0]=TMAR[7:0]+16

00：F_{TMA} / 4；Every overflow occurs interrupt event, TMAR[7:0]=TMAR[7:0]+4

TMAR：TMA ascending counter, readable but not writable

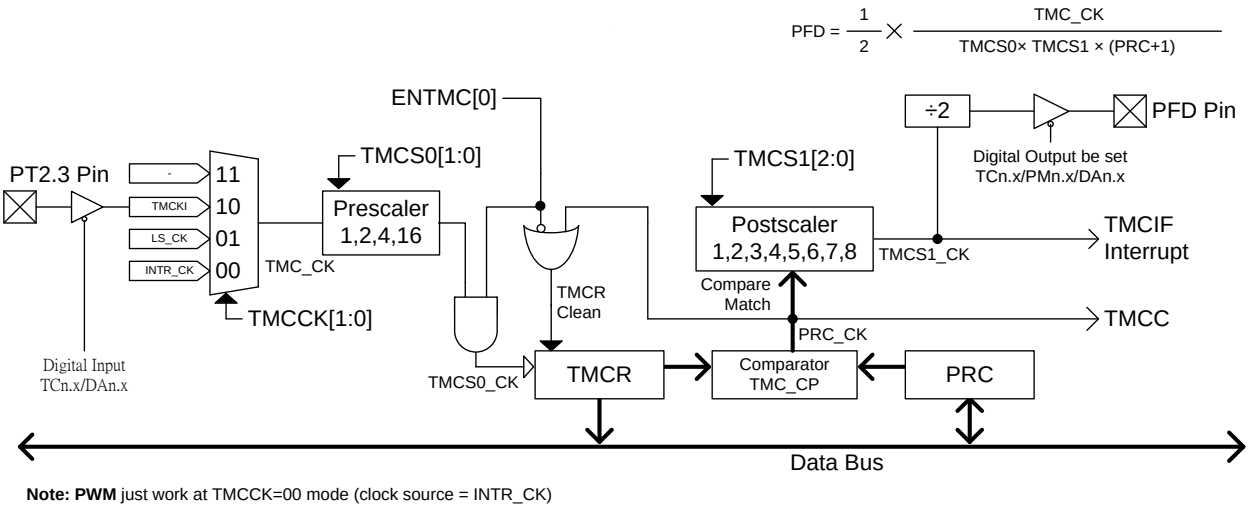
10. Timer-C

Timer C is designed in 8-bit frame. TMC counter is composed by two value registers and one comparator. The overflow event is generated by postscaler. It can be operated in Run Mode, Idle Mode and Sleep Mode.

- Equip with 8-bit frequency controller, value comparator and counter
- Ascending counter
- Value comparator
- Supporting PWM function
- Supporting PFD function
- Plan overflow value
- Overflow generated interrupt event

Timer-C Registers:

TMCCN	ENTMC[0], TMCCK[1:0],TMCS1[2:0],TMCS0[1:0]
TMCR	TMCR[7:0]
PRC	PRC[7:0]
INTE1	TMCIE[0]
INTF1	TMCIF[0]



Note: PWM just work at TMCCK=00 mode (clock source = INTR_CK)

Figure 10-16 Timer-C Block Diagram

10.1. Timer-C Manual

10.1.1. TMC Initial Configuration

TMC operating frequency is configured through operating frequency selector, TMCCK [1:0]. It can set TMC_CK as INTR_CK[11]、LS_CK or TMCKI[12] operating frequency. TMC can be operated under Run, Idle or Sleep Mode by configuring diverse frequency.

Configuration of prescale controller, TMCS0[1:0] will prescale TMC_CK to generate TMCS0_CK. The configuration of overflow controller, TMCS1[2:0] will prescale PRC_CK and TMCS1_CK will be produced.

Counter, TMCR[7:0][13], frequency controller, PRC[7:0] and value comparator, TMC_CP compose PRC_CK signal generated scheme. That is, when TMCR[7:0] and PRC[7:0] register content is the same, PRC_CK signal will be aroused to postscaler, TMCS1[2:0].

10.1.2. TMC Interrupt Event Service

PRC_CK, after postscaler, will generate overflow interrupt event as TMCS1[2:0] configured condition is satisfied. TMCIF[0] is configured as <1>. If interrupt event service is required, TMCIE[0] and GIE[0] must be set up as <1>.

10.1.3. Timer-C Initiation

Configuring ENTMC[0] as <1> can start TMC and TMCR[7:0] will begin counting. If it is configured as <0>, TMC will be shut off and TMCR[7:0] counter value will be zeroed automatically. Thus, in order to obtain correct value, users must first write the value in PRC[7:0] then starting TMC.

```
BCF   INTF1,TMCIF,0     ; Clear TMC interrupt event flag
MVL   0A0h
MVF   INTE1,1,0         ; Configure Timer C interrupt service
MVL   01Fh              ; Write 01Fh into PRC
MVF   PRC,1,0           ; Interrupt frequency is around: INTR_CK/(1Fh x 2h)
MVL   084h              ; Start Timer C configured operating frequency
MVF   TMCCN,1,0         ; Operating frequency is INTR_CK, not prescale but postscale, configure TMCS1[2:0] = 001b
….
TMC Interrupt :         ; TMC interrupt event service program
BCF   INTF1,TMCIF,0     ; Clear TMC interrupt event flag
….
```

Example 10-8 Timer-C Interrupt Event Example Program

[11] When using PWM peripherals, TMC operating frequency must be configured as this argument.

[12] When input frequency chooses external TMCKI, I/O pin must be configured correctively; otherwise the signal may not be entered and will bring about abnormal execution. For detail register description, please refer to ***Input/Output Port, I/O*** Chapter.

[13] TMCR[7:0] only can be read, any writing action to TMCR[7:0] or TMCCN[7:0] will be deemed as zeroing counter, TMCR[7:0] and the value of prescaler and postscaler will be zeroed as well.

10.2. Register Description -TMC

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1												
".unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition												
Address	File Name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	A-RESET	i-RESET	R/W
23H	INTE1	GIE		TMCIE						0.0. 0000	0.0. 0000	*,-,*,- *,*,*,*
26H	INTF1			TMCIF						..0. 0000	..0. 0000	-,-,*,- *,*,*,*
34H	TMCCN	ENTMC	TMCCK[1:0]		TMCS1[2:0]			TMCS0[1:0]		0000 0000	0000 0000	*,*,*,*,*,*,*,*
35H	PRC	TimerC programmable register								1111 1111	1111 1111	*,*,*,*,*,*,*,*
36H	TMCR	TimerC register								0000 0000	0000 0000	r,r,r,r r,r,r,r

Table 10-22 TMC Control Register

INTE1/INTF1：Please refer to ***interrupt*** Chapter

TMCCN：Timer-C Control Register

ENTMC：Timer-C starting control bit

1：Start

0：Shutoff: clear zero counter value

TMCCK[1:0]：TMC operating frequency select controller

11：Reserved

10：TMCKI；This configuration does not support PWM peripheral circuits

01：LS_CK；This configuration does not support PWM peripheral circuits

00：INTR_CK

TMCS1[2:0]：Timer-C counter overflow controller

111：PRC_CK/8

110：PRC_CK/7

101：PRC_CK/6

100：PRC_CK/5

011：PRC_CK/4

010：PRC_CK/3

001：PRC_CK/2

000：PRC_CK/1

TMCS0[1:0]：Timer-C operating frequency prescaler

11：TMC_CK/16

10：TMC_CK/4；This configuration support partial PWM peripheral circuits, please refer to CH.11 Frequency generator, PWM / PDF description

01：TMC_CK/2；This configuration support partial PWM peripheral circuits, please refer to CH.11 Frequency generator, PWM / PDF description

00：TMC_CK/1；This configuration support partial PWM peripheral circuits, please refer to CH.11 Frequency generator, PWM / PDF description

TMCR：Timer C Counter

Ascending counter of Timer-C that is can only read. Any write in action to TMCR[7:0] or TMCCN[7:0] will be deemed as zeroing TMCR[7:0].

PRC：Frequency Control Register

Frequency controller of Timer-C, TMC_CP will contrast the content of TMCR[7:0] and PRC[7:], when the value is in equality, reversed PRC_CK status.

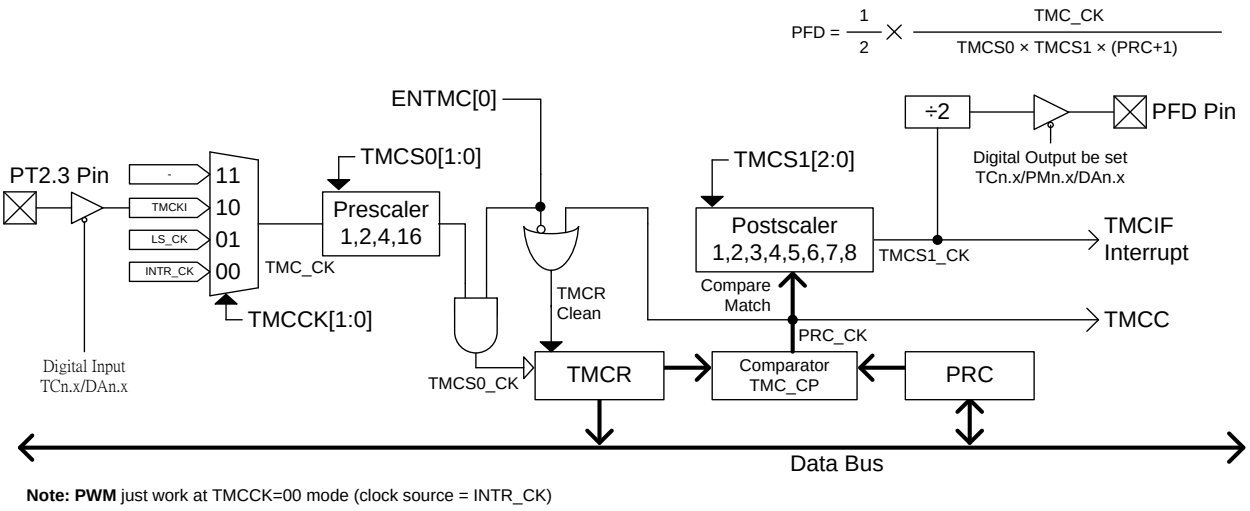
11. Frequency Generator, PWM/PFD

Frequency generator contains two modes, one is Pulse Width Modulation, PWM, and the other is Pulse Divider, PFD.

- Must be used in collocation with Timer-C
- Multiplexed PWM and PFD Mode
- 10-bit frequency controller

PFD and PWM Registers:

TMCCN	ENTMC[0], TMCCK[1:0],TMCS1[2:0],TMCS0[1:0]
TMCR	TMCR[7:0]
PRC	PRC[7:0]
PWMCN	ENPWM[0], ENPFD[0],PWMRL[1:0]
PWMR[9:0]	PWMRH[7:0] ,PWMRL[1:0]



Note: PWM just work at TMCCK=00 mode (clock source = INTR_CK)

Timer-C Block Diagram

11.1. PFD Mode Manual

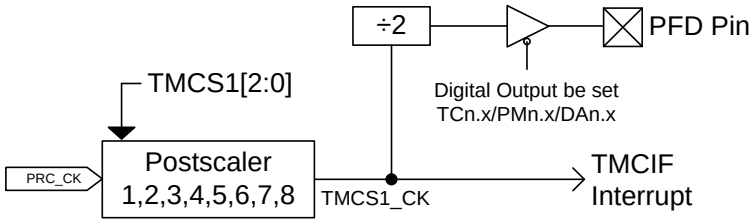


Figure 11-17 PFD Block Diagram

Frequency modulation, PFD Mode must be utilized with Timer-C supportability; configuration of output frequency modulation must through Timer-C. PFD frequency output must be set as digital output and the output signal must select PFD.

11.1.1. PFD Mode Initial Configuration

Timer-C has to be started first, relative configuration please refer to Timer-C description.

PFD operating frequency is PRC_CK, through TMC overflow controller configuration, TMCS1[2:0], can change PFD pin14 output frequency, PFD frequency modulation formula is as Equation 11-1.

Equation 11-1

$$PFD = \frac{1}{2} \times \frac{TMC\_CK}{TMCS0 \times TMCS1 \times (PRC+1)}$$

PFD Modulation Frequency Equation

11.1.2. PFD Mode Initiation

Configure ENPFD[0] as <1> to start PFD Mode. When ENPFD[0] is set as <0> will shut off PFD Mode.

```
BCF  PT2PU, 2, 0          ; PORT relative configuration
BSF  TRISC2, 2, 0
BCF  PT2DA, 1, 0          ;Configure PTn.x as digital output interface and output PFD signal
BSF  PT2DA, 0, 0
MVL  01Fh                 ; Write 01Fh to PRC
MVF  PRC,1,0              ; Start Timer C
MVL  084h                 ; Configure operating frequency as INTR_CK
MVF  TMCCN,1,0            ; Do not prescale but postscale configurated TMCS1[2:0] = 001b

                          ; So PFD frequency is INTR_CK/(1h x 2h x 20h)
BSF  PWMCN.ENPFD.0        ; Start PFD
```

[14] PFD Mode must correctly configure I/O PORT setup while using or the signal may not be output and PFD Mode will have abnormal function. Detailed description please refers to *Input/Output Port, I/O* Chapter.

Example 11-9 PFD Output Example Program

11.2. PWM Mode Manual

- Pulse width modulation, PWM equips with the following functions:
  - PWM Output
  - Must have Timer-C support

11.2.1. PWM Single Output Manual

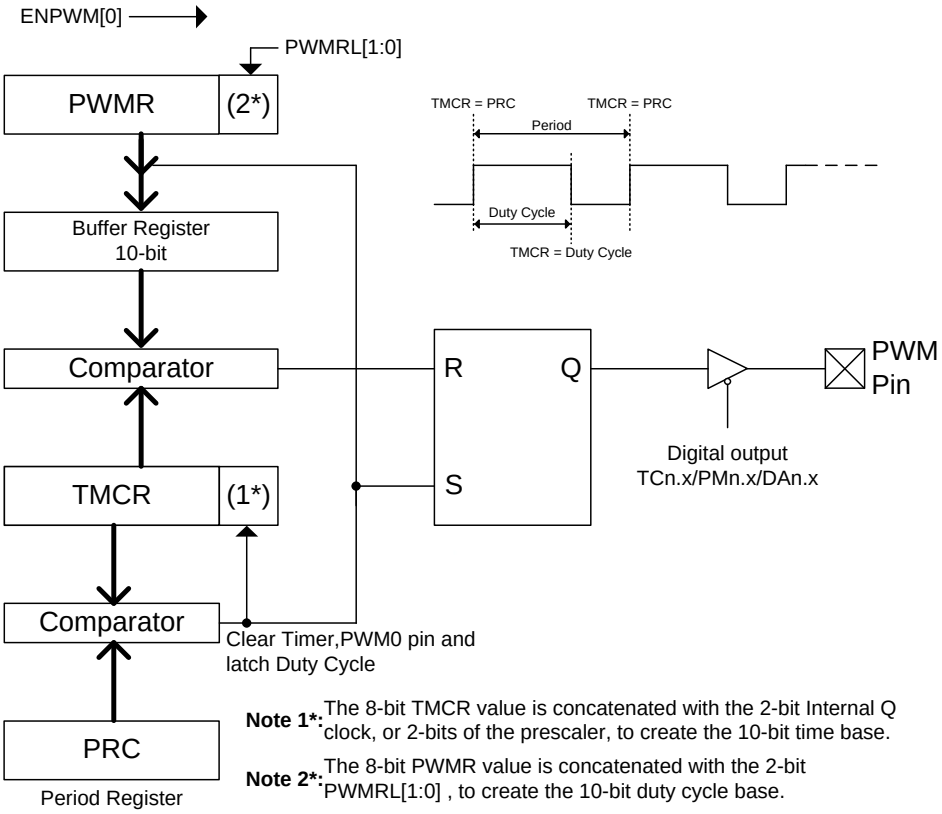


Figure 11-2 PWM Single Output Block Diagram

PWM single output pulse width modulation signal has single output PWM0 pin[15] .Before using, Timer-C must be configured first to scheme PWM frequency and duty cycle.

[15] PWM Mode must correctly configure PORT related configuration while using, or the signal may not be output and PFD Mode will have abnormal function. Detailed register description please refer to ***Input/Output Port, I/O*** Chapter.

PWM Single Output Initial Configuration

Frequency (Period) Configuration

Frequency controller, PRC[7:0][16] has 8-bit length, change its configuration parameter can determine PWM period (frequency). The equation is shown in Equation 0-1：

Equation 0-1 (a)

$$\text{PWM Period} = \frac{\text{TMCS0} \times (\text{PRC}+1)}{\text{TMC\_CK}}$$

PWM Period Equation

Equation 0-1 (b)

$$\text{PWM Frequency} = \frac{1}{\text{PWM Period}}$$

PWM Frequency Equation

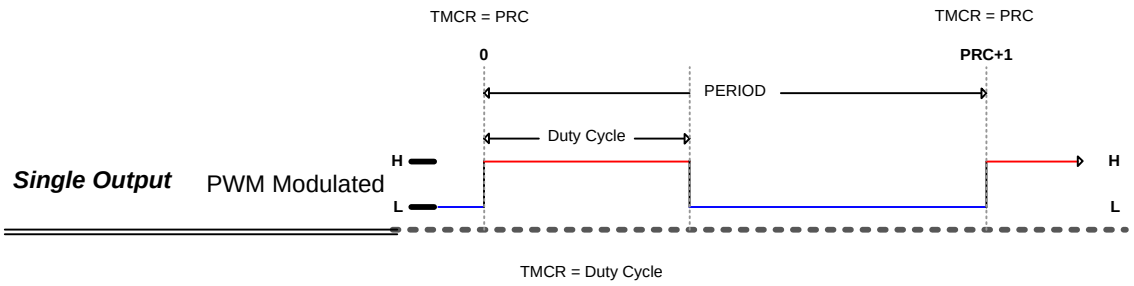
Duty Cycle Configuration

Pulse width modulation controller, PWMR[9:0][17] has 10-bit length and is composed by PWMRH[7:0] and PWMRL[1:0]. PWM's pulse width is determined by changing its configuration parameters, as Equation 0-2.

Equation 0-2

$$\text{PWM Duty Cycle} = \frac{\text{TMCS0} \times (\text{PWMR[9:0]} + 4)}{\text{TMC\_CK} \times 4}$$

PWM Duty Cycle Equation



PWM OUTPUT RELATIONSHIPS

Figure 11-3 PWM Waveform Chart

[16] When the value of TMCR[7:0] and PRC[7:0] is equal, it will bring about two results：a. PWMx output pin will be configured as 1. (If PWMR[9:0]=0，PWMx output pin will not be set as 1). b. TMCR will be cleared zero.

[17] When PWM is in operating status, PWMR will show the written value until the end of this period. If PWMR value is bigger than PRC value, PWMx output pin will not be configured as 0.

Resolution Configuration

TMC_CK and PWM output frequency must be taken into account while computing PWM greatest resolution, as shown in Equation 0-3.

Equation 0-3

$$\text{PWM Resolution (max)} = \frac{\log\left(\frac{\text{TMC\_CK} \times 4}{\text{TMCS0} \times \text{PWM Frequency}}\right)}{\log(2)}$$

PWM Resolution Equation

Timer-C Configuration

PWM must adopt Timer-C to generate frequency and duty cycle. Therefore, when Timer-C supports PWM Mode, its operating frequency selector, TMCCK[1:0] and prescaler, TMCS0[1:0] configuration will be restrained.

If CPU operating frequency equals to or heightens than 2MHz, PWM's operating frequency selector, TMCCK[1:0] and prescaler, TMCS0[1:0] configuration is as follows:

In order to provide operating frequency, TMC for PWM utilization, TMCCK[1:0] can only be configured as <00> through INTR_CK or to be configured as <01> through LS_CK.

4 prescaling arguments of TMCS0[1:0] configuration can be used.

If CPU operating frequency equals to or lowers than 32KHz, PWM's operating frequency selector, TMCCK[1:0] and prescaler, TMCS0[1:0] configuration is as follows:

Configure TMCCK[1:0] as <00>, INTR_CK provides operating frequency to TMC. Under this configuration, 4 prescaling arguments of TMCS0[1:0] configuration can be used.

Configure TMCCK[1:0] as <01>, LS_CK provides operating frequency to TMC. Under this configuration, only TMCS0[1:0] <11> is applicable.

Detailed description please refers to ***Register Description-TMC*** Chapter.

11.2.1.1. PWM Single Output Initiation

Start controller, ENPWM[0] is configured as <1> to initiate PWM Mode. When ENPWM[0] is set as <0>, PWM Mode will be shut off.

```
BCF  PT2PU, 2, 0          ; PORT relative configuration
BSF  TRISC2, 2, 0
BSF  PT2DA, 1, 0          ; Configure PTn.x as digital output interface and output PWM signal
BCF  PT2DA, 0, 0
MVL      0FFh             ; Write 0FFh to PRC, Set PWM Period
MVF      PRC,1,0          ; PWM Period=1*(255)/500khz =512us
MVL      07FH             ; PWM Duty cycle (07FH)
MVF      PWMR,1,0         ; PWM duty cycle=1*127/2mhz =254us
BSF      PWMCN,5,0        ; high duty percentage=49.61%
BSF      PWMCN,4,0
MVL      084h
MVF      TMCCN,1,0        ; Start TMC, configure operating frequency as INTR_CK,
                          ; do not prescale but postscale configurated TMCS1[2:0] = 001b
MVL      0B0H             ; Start PWM
MVF      PWMCN,1,0
```

Example 11-2 PWM Output Example Program

11.3. Register Description-PFD/PWM

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1												
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition												
Address	**File Name**	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**	**A-RESET**	**i-RESET**	**R/W**
23H	INTE1	GIE		TMCIE						0.0. 0000	0.0. 0000	*,-,*,- *,*,*,*
26H	INTF1			TMCIF						..0. 0000	..0. 0000	-,-,*,- *,*,*,*
34H	TMCCN	ENTMC	TMCCK[1:0]		TMCS1[2:0]			TMCS0[1:0]		0000 0000	0000 0000	*,*,*,*,*,*,*,*
35H	PRC	TimerC programmable register								1111 1111	1111 1111	*,*,*,*,*,*,*,*
36H	TMCR	TimerC register								0000 0000	0000 0000	r,r,r,r r r,r,r,r
37H	PWMCN	ENPWM	ENPFD	PWMRL[1:0]						0000 0000	0000 0000	*,*,*,* *,*,*,*
38H	PWMR	PWM MSB Byte register								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
51H	PT2						PT2.2			xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
52H	TRISC2						TC2.2			0000 0000	0000 0000	*,*,*,*,*,*,*,*
53H	PT2DA							PM2.2[1:0]		0000 0.00	0000 0.00	*,*,*,*,-,*,*
54H	PT2PU						PU2.2			0000 0000	0000 0000	*,*,*,* *,*,*,*

Table 11-23 PFD/PWM Register

INTE1/INTF1：Please refer to **Interrupt** Chapter.

TMCCN/TMCR：**Please refer to Timer-C Chapter.**

PT2/TRISC2/PT2DA/PT2PU：Please refer to ***Input/Output Port, I/O*** Chapter.

PRC：**PWM Period Controller, TMC Frequency controller**

PWM relative equation：

PWM Period = (PRC+1) × (1/TMC_CK) × TMCS0

PWM Duty Cycle = TMCS0 x PWMR ÷ (TMC_CK x 4)

PWM Resolution = log(TMC_CK / PWM Frequency) / log(2)

PWMCN：PWM Control Register

ENPWM：PWM Start Controller

1：Start

0：Shutoff

ENPFD：PFD Start Controller

1：Start

0：Shutoff

PWMRL[1:0]: PWMR[9:0] low bit

PWMR[9:0] = PWMRH[7:0] + PWMRL[1:0]

ENPRS：Auto start condition controller

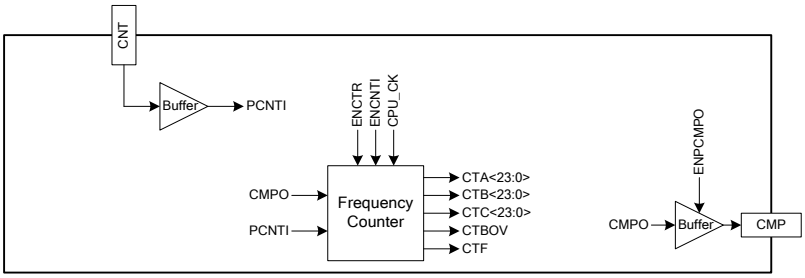
1：Hardware automatically configures PASF as 0. PWMx modulated output will restart in next period.

0：Users employ software to configure PASF as 0. PWMx modulated output will restart in next period

PWMR：PWM Duty Cycle High Bit Register

PWM duty cycle high byte [9:2]

12. Frequency Counter, CNT And CMP Pin



Frequency Counter standby signals can be chose by ENCNTI as DMM Comparator Network output CMPO or PT3.6 input CNTI. Frequency Counter includes 3 group 24bits Counters. They are CTA, CTB and CTC. CTA and input clock source of CTC is SYSCLK, input clock source of CTB is standby signals. The counter only can be active when the standby signal of CTC is high. Frequency Counter operation counting procedure as follows:

(1) Configure ENCTR=0 and CTA<7:0>, CTB<23:0>, CTC<23:0> will be configured as 0.

(2) Write in counter initial value as CTA<23:8>, then set Gate Time=[1000000h-CTA<23:0>]/F_{SYSCLK}.

(3) After configure ENCTR=1, start counting when the first positive edge of standby signals come up until CTA<23:0> overflow generated and the first positive edge of standby signals stop counting. The start and stop of counter is the completed cycle of standby signal numbers. The interrupt signal will be appeared when the counter stop counting.

(4) Read CTA<23:0>, CTB<23:0>, CTC<23:0> and CTBOV.

(5) If CTBOV=1, it means Gate Time was set to too long and standby signal frequency is too high, so CTB<23:0> overflow generated. If so, the information produced this time is useless. The procedure has to be restarted from step (1) and reset Gate Time, count again.

(6) If CTBOV=0, it means the information produced this time is useful. The frequency of standby signal, Duty Cycle can be calculated by produced information.

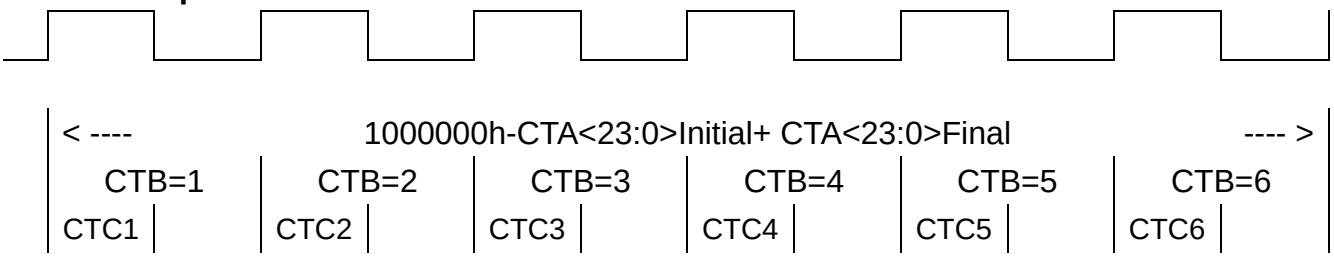
T (Count time)=[1000000h-CTA<23:0>$_{Initial}$+ CTA<23:0>$_{Final}$]/F_{SYSCLK}

Standby signals frequency=CTB<23:0>/T

Standby signal, Duty Cycle=CTC<23:0>/[1000000h-CTA<23:0>$_{Initial}$+ CTA<23:0>$_{Final}$]

Within it, F_{SYSCLK} is the frequency of SYSCLK, CTA<23:0>$_{Initial}$ is the value before count operated. CTA<23:0>$_{Final}$ is the value occurred after count operated.

12.1. Example of Calculation



Calculation element description (Taking 1kHz / 50% as example)

FSYSCLK：System oscillator frequency, supposed it is 4MHz

CTA<23:0>Initial：Default value before CTA counter, CTA<23:8> program defaults C000h, and CTA<7:0>is cleared as 00h

CTA<23:0>Final：Value after CTA counter, CTA<23:0> Initial value is C00000h; it will be 000760h under 1kHz.

CTB<23:0>：Cycles within the time period, CTA<23:0> Initial value is C00000h, it will be 000419h under 1kHz

CTC<23:0>：High time sum, CTA<23:0> Initial value is C00000h, it will be 20043Ah under Duty 50%.

Count time:

T = [1000000h-CTA<23:0>Initial+ CTA<23:0>Final]/FSYSCLK

=(1000000h-C00000h +000760h)/3D0900h --- >hexadecimal

=(16777216-12582912+1888)/4000000=1.0490 --- >decimal

Standby signals frequency:

Freq = CTB<23:0>/T

= 1049/1.0490=1000 Hz

Standby signal, Duty Cycle:

Duty Cycle = CTC<23:0>/[1000000h-CTA<23:0>Initial + CTA<23:0>Final]

= 20043Ah/400760h --- >hexadecimal

= 2098234/4196192=0.5=50% --- >decimal

12.2. Register Description- Frequency Counter

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1											
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition											
Address	**File Name**	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**	**A-RESET**	**i-RESET**
27H	INTF2	TXIF	RCIF	RMSF	LPFF	AD1F	SSPIF	CTF		0000 000.	0000 000.
5CH	PWRCN2	MCUBIAS	ENCPVGG	ENCMP	ENCNTI	ENCTR	RSTCOMB	RSLPF	RSRMS		
65H	CTAU	CTA<23:16>									
66H	CTAH	CTA<15:8>									
67H	CTAL	CTA<7:0>									
68H	CTBU	CTB<23:16>									
69H	CTBH	CTB<15:8>									
6AH	CTBL	CTB<7:0>									
6BH	CTCU	CTC<23:16>									
6CH	CTCH	CTC<15:8>									
6DH	CTCL	CTC<7:0>									
6EH	CTSTA	CNTI	CMPO	CMPHO	CMPLO				CTBOV		

Related register description as follows:

(1) ENCTR: Register bit, can enable Frequency Counter. 1=Enable; 0=Disable and clear CTA<23:0>, CTB<23:0>, CTC<23:0> and CTBOV to 0.

(2) ENCNTI: Register bit, can choose Frequency Counter input source. CNTI can be read by CTSTA<7> at the same time.

ENCNTI	0	1
Frequency Counter Input	CMPO	PCNTI

(3) CTA<7:0>: Data Register, when ENCTR=0, CTA<7:0> will be cleared to 0.

(4) CTA<23:8>: Data Register, when ENCTR=0, CTA<23:8> will not be cleared to 0. When ENCTR=0, CTA<23:8> can be written in MCU directly. When ENCTR=1, CTA<23:8> only can be increased progressively by Frequency Counter.

(5) CTB<23:0>: Data Register, when ENCTR=0, CTB<23:0> will be cleared to 0. When ENCTR=1 and interrupt occurred by the time count finished, CTB<23:0> will record the numbers of full cycle of standby signals. The numbers can be used to calculate the frequency of standby signals.

(6) CTC<23:0>: Data Register, when ENCTR=0, CTC<23:0> will be cleared to 0. When ENCTR=1 and interrupt occurred by the time count finished, CTC<23:0> will record SYSCLK numbers under the situation of standby signals is high. The numbers can be used to calculate the Duty Cycle of standby signals.

(7) CTBOV: Data Register, when CTB<23:0> is overflow, CTBOV will be set as 1. When read the register, CTSTA or ENCTR=0, CTBOV will be set as 0.

(8) CTF: CTF is the flag of Frequency Counter events occurred. The signals will be sent in INTF register.

(9) ENPCMPO: PT3PU[PM3.7] register bit, can enable CMPO Pin if output by PT3.7. 1=Enable; 0=Disable. Configure CMPO is output by PT3.7 and need to configure PT3.7 is output as well.

13. LCD

LCD driving circuit is suitable for LCD production process like TN-LCD and STN-LCD, and it has the following features：

- Regulated charge pump
- 4 steps adjustable driving voltage state
- Supporting 4 LCD waveform types operation
  - Static operation
  - 2-mux, 1/3 bias (2-mux, 1/3 bias)
  - 3-mux, 1/3 bias (3-mux, 1/3 bias)
  - 4-mux, 1/3 bias (4-mux, 1/3 bias)
- Selectable input clock source and output frequency
- Equips with Blinking capability

LCD Registers：

LCDCN1 ENLCD[0],LCDPR[0] ,VLCDX[1:0],LCDBF[0],LCDBI[1:0]

LCDCN2 LCDBL[0],LCDMX[1:0]

LCD[159:0] LCD0[7:0]~ LCD6[7:0], LCD7[3:0]

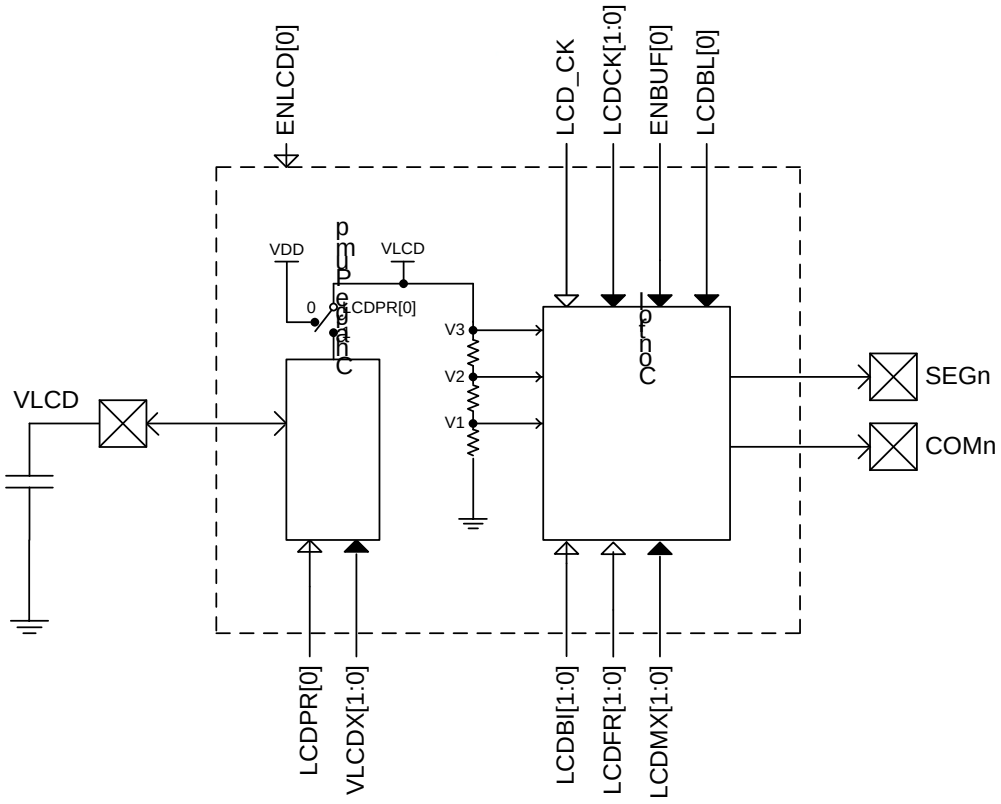


Figure 13-18 LCD Block Diagram

13.1. LCD Manual

13.1.1. LCD Initial Configuration

Operating Frequency and Output Frame Frequency Configuration.

Operating frequency is offered by PERA_CK, after pre-scaler, LCDS[2:0] offers appropriate operating frequency to LCD to output frame frequency. Output waveform controller, LCDMX[1:0] can configure LCD operating waveform, frame frequency and operating waveform must be configured correctly in accordance with external LCD monitor specification, otherwise LCD monitor will display shadow or abnormal digit display. LCD operating frequency and frame frequency is presented in Table 13-1。

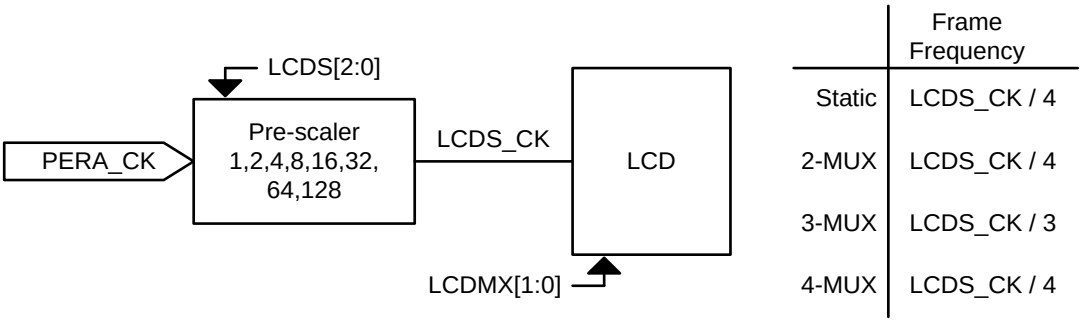


Figure 13-19 LCD Operating Frequency and Frame Frequency

PERA_CK=1953Hz (PERCK[0]=1b, PERA_CK~4MHz/2/32/32)		LCDS[2:0]							
		000	001	010	011	100	101	110	111
LCDS_CK		1953	977	488	244	122	61	31	15
Frame Frequency	Static, LCDMX[1:0]=00	488	244	122	61	31	15	8	4
	2-MUX, LCDMX[1:0]=01	488	244	122	61	31	15	8	4
	3-MUX, LCDMX[1:0]=10	651	326	163	81	41	20	10	5
	4-MUX, LCDMX[1:0]=11	488	244	122	61	31	15	8	4

Unit：Hz

Table 13-24 LCD Operating Frequency Configuration

Charge Pump Circuit and LCD Operating Voltage Configuration

There are two ways to generate LCD operating voltage, VLCD:

- VLCD is input externally, LCDPR[0] must be configured as <0> and charge pump must be disabled. Voltage comes through external VLCD pin in order to determine LCD operating voltage. When using external input, the configuration of VLCDX[1:0] will not influence LCD operating voltage.
  - When driving over-sized or over-loaded LCD monitor, LCD output buffer, LCDBF[0] can be set up as <1> and can initiate buffer to increase LCD driving ability. On the contrary, if LCDBF[0] is configured as <0>, LCD consumed current will decrease as the buffer being shut off.
- VLCD is generated by internal charge pump. By setting up charge pump circuit controller, LCDPR[0] as <1> and configure charge pump voltage state controller, VLCDX[1:0] , VLCD will be produced to supply LCD power that prevents it against IC operating voltage changes.
  - VLCDX[1:0] can be configured as 4 different kind of functional operating voltage if the charge pump is being initiated first. Charge pump circuit may influence analog-to-digital convertor, SD18 performance in high resolution conversion.
  - Using internal charge pump circuit to produce VLCD, LCD buffer will be started automatically by internal hardware circuit.

Bias and Blinking Configuration

Waveform bias controller, LCDBI[1:0] can configure every LCD frame is consisted by how many bias. There are two options, configuring <00> LCD waveform in static operation and configuring <10> LCD waveform operation in 1/3 bias.

Blinking effect enables LCD to switch from display status to off status or switching it back to display status. This cycling process only needs digit blink controller, LCDBL[0] to configured as <1> for off or by configuring <0> for full display. Hence, configuring LCDBL[0] as <1>, LCD monitor will not lighten up any digit. That is to say, if LCDBL[0] is set as <0>, LCD monitor will light up according to digit register's configuration of LCD1[7:0]~ LCD7[3:0].

LCD Digit Register

Every digit register, LCDn[7:0] controls two digit pins SEGn, and each digit has 4-bit control bit, SEGn[3:0]. Whether the control bit is valid or invalid, it depends on waveform output controller, LCDMX[1:0] configuration. For instance, when output waveform is 4-mux, SEGn[3:0] 4-bit is valid. Besides, if output waveform is 2-mux, only the lowest SEGn[3:0] 2-bit is valid. Because of the valid and invalid characteristic, LCD0 and LCD1 digit register has SEG0[3:0] and SEG1[3:0] multiplexed design.

13.1.2. LCD Initiation

Configuring ENLCD[0] as <1> can start LCD driver. On the other hand, when ENLCD[0] is configured as <0>, LCD driver will be closed.

```
MVL     01100000B
MVF     LCDCN2,1,0      ;1/4 duty, LCD digit display
MVL     11011100B
MVF     LCDCN1,1,0      ;1/3 bias, LCD start. Initiating LCD charge pump power VLCD=3V
CALL    DELAY           ;LCD charge pump power regulated time (at VLCD CAP-4.7uF)
                        ;VDD=2.2V, VLCD=3V, Stable time ~ 85msec
                        ;VDD=3.6V, VLCD=3V, Stable time ~ 15msec
```

Example 13-10 LCD Example Program

13.2. LCD Output Waveform

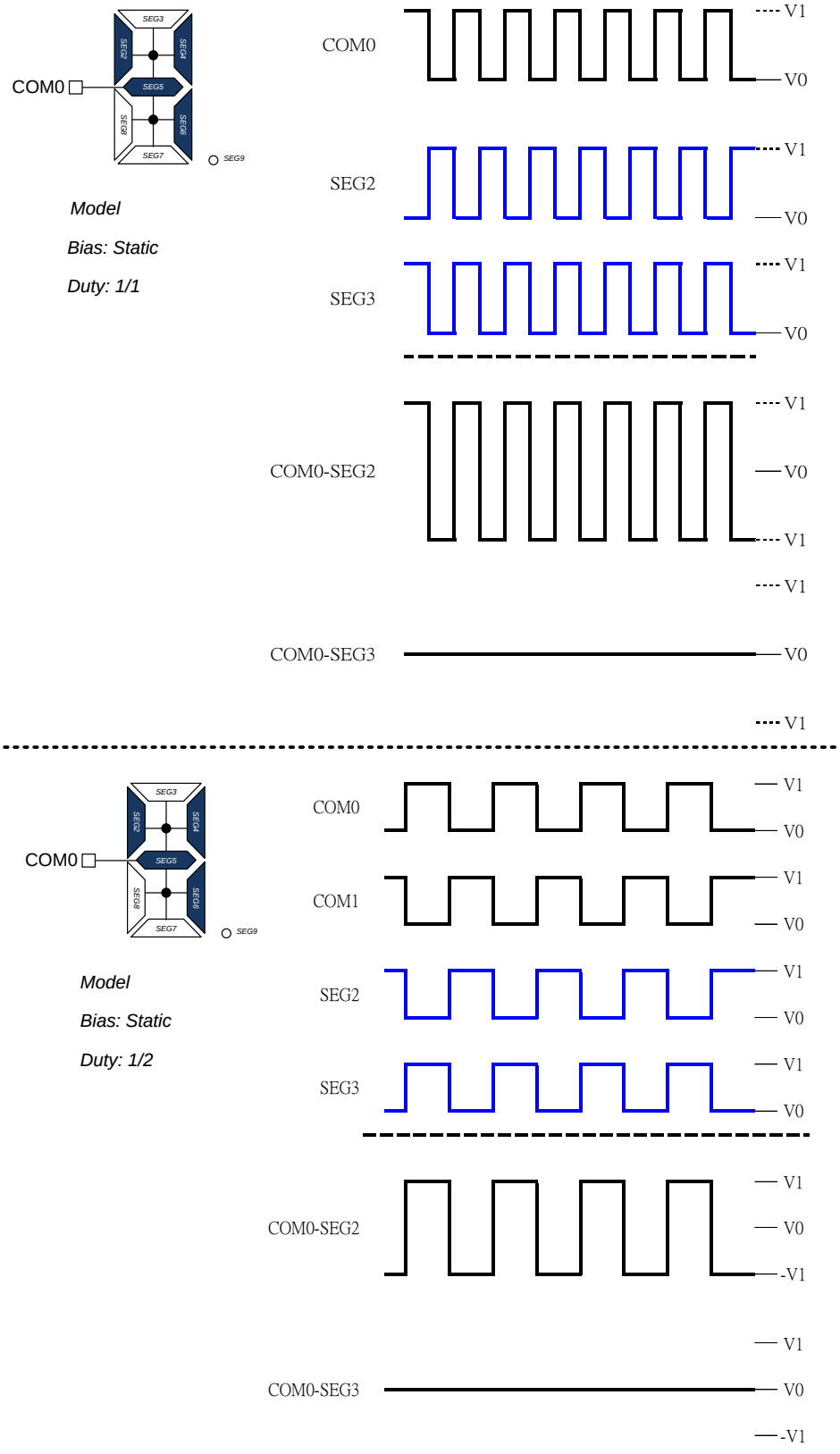


Figure 13-20(a) Output Waveform-Static Operation

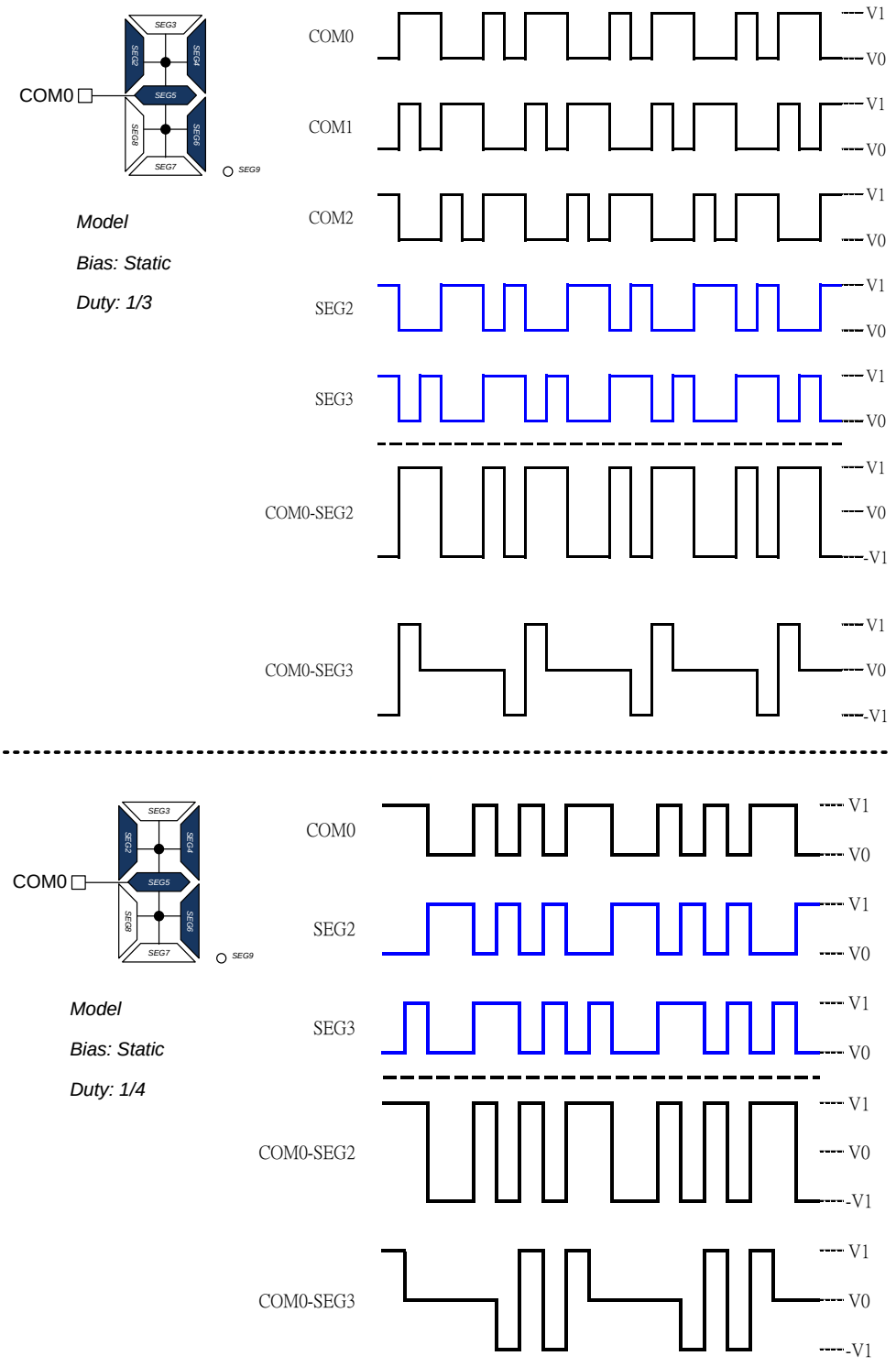


Figure 13-3(b) Output Waveform-Static Operation (continued)

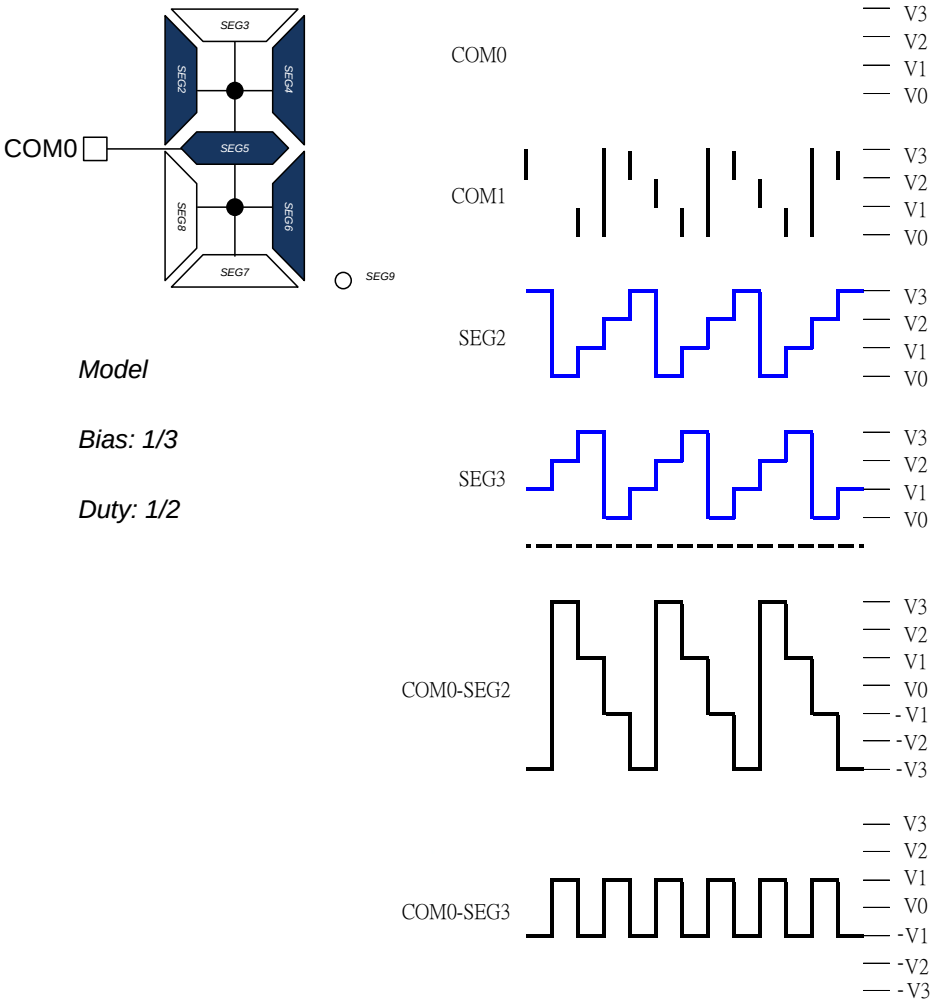


Figure 13-21 Output Waveform -2-Mux

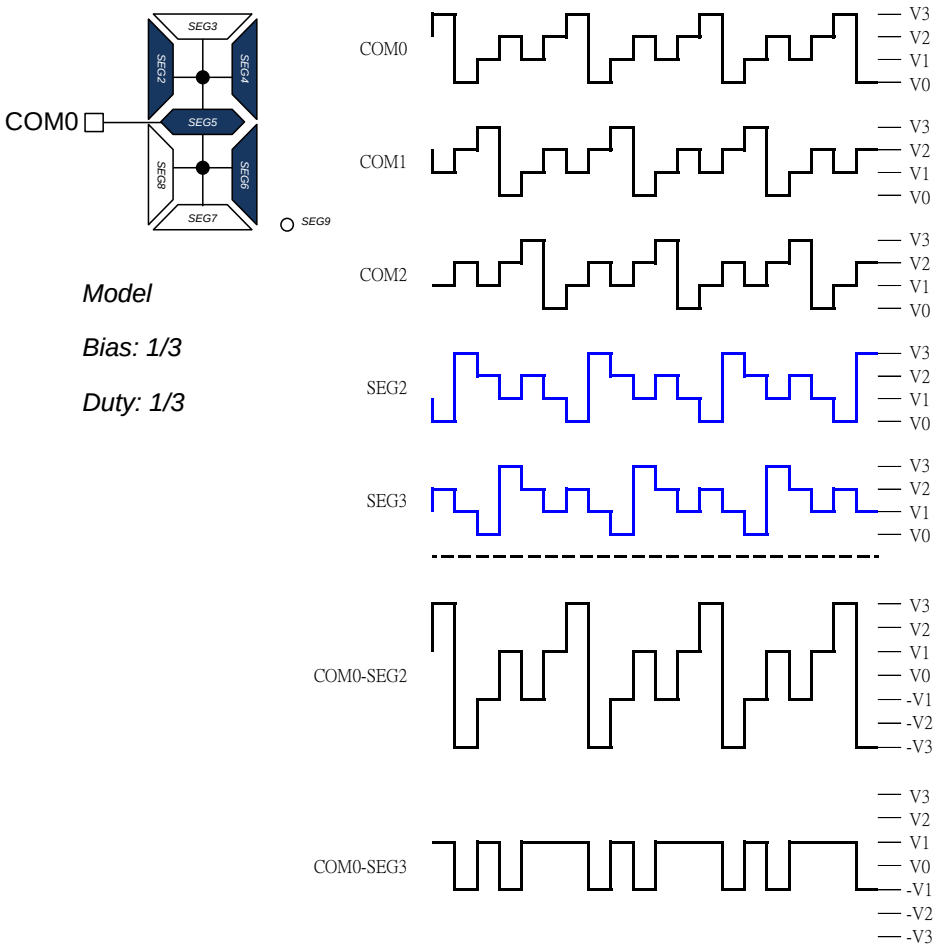


Figure 13-22 Output Waveform-3-Mux

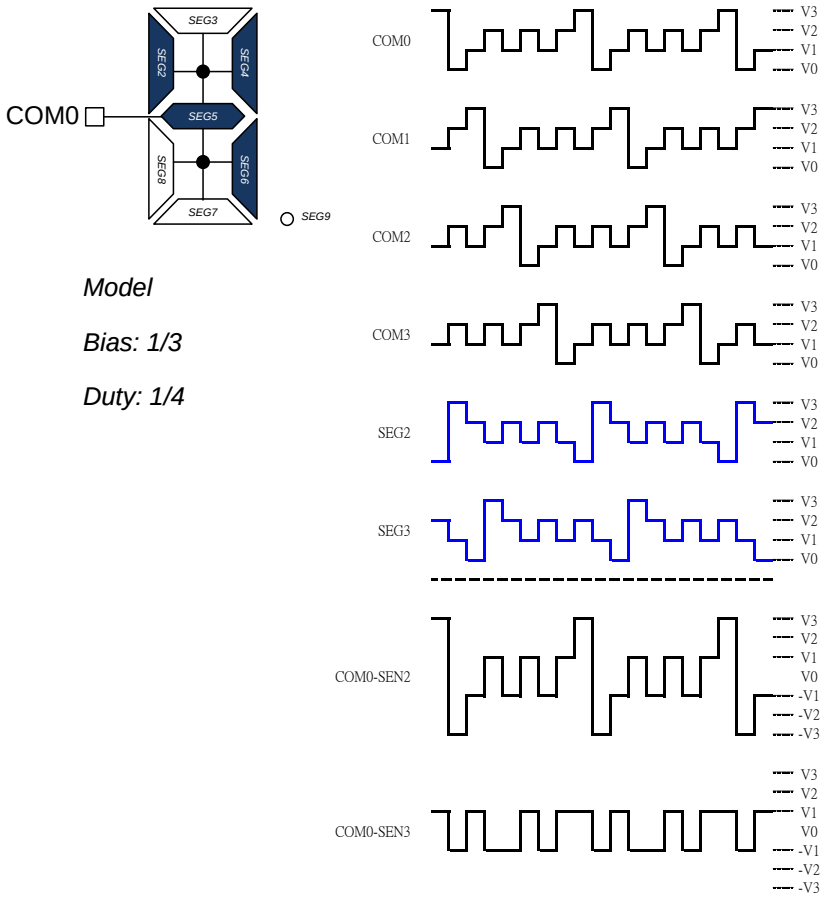


Figure 13-23 Output Waveform-4-Mux

13.3. Register Description -LCD

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1												
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition												
Address	**File Name**	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**	**A-RESET**	**i-RESET**	**R/W**
31H	MCKCN2	LCDS[2:0]								0000 0000	0000 0000	*,*,*,*,*,*,*,*
39H	LCDCN1	ENLCD	LCDPR	VLCDX[1:0]		LCDBF	LCDBI[1:0]			0000 000.	0000 000.	*,*,*,*,*,*,*,-
3AH	LCDCN2	LCDBL	LCDMX[1:0]							000.	000.	*,*,*,- -,-,-,-
3BH	LCD0	Segment SEG1@[7:4] and SEG0@[3:0] data register of LCD								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
3CH	LCD1	Segment SEG3@[7:4] and SEG2@[3:0] data register of LCD								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
3DH	LCD2	Segment SEG5@[7:4] and SEG4@[3:0] data register of LCD								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
3EH	LCD3	Segment SEG7@[7:4]and SEG6@[3:0] data register of LCD								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
3FH	LCD4	Segment SEG9@[7:4] and SEG8@[3:0] data register of LCD								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
40H	LCD5	Segment SEG11@[7:4] and SEG10@[3:0] data register of LCD								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
41H	LCD6	Segment SEG13@[7:4] and SEG12@[3:0] data register of LCD								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
42H	LCD7					Segment SEG14@[3:0] data register of LCD				 xxxx	.... uuuu	-,-,-,- *,*,*,*

Table 13-25 LCD Register

MCKCN2：Please refer to **Oscillator, Clock Sources and Power Managed Modes** Chapter.

LCDCN1：**LCD Control Register 1**

ENLCD：LCD start controller

1：Start

0：Shutoff

LCDPR：LCD charge pump circuit controller

1：Start；Voltage source, VLCD is generated from internal IC

0：Shutoff；Voltage source, VLCD is inputted from external pin

VLCDX[1:0]：Charge pump voltage state select controller

11：VLCD = 2.55V。

10：VLCD = 2.8V。

01：VLCD = 3.05V。

00：VLCD = 3.3V。

LCDBF：LCD output buffer

1：Start

0：Shutoff

LCDBI[1:0]：LCD waveform bias controller

11：Unused

10：1/3 bias

01：Reserved

00：Static operation

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1												
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition												
Address	**File Name**	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**	**A-RESET**	**i-RESET**	**R/W**
31H	MCKCN2	LCDS[2:0]								0000 0000	0000 0000	*,*,*,*,*,*,*,*
39H	LCDCN1	ENLCD	LCDPR	VLCDX[1:0]		LCDBF	LCDBI[1:0]			0000 000.	0000 000.	*,*,*,*,*,*,*,-
3AH	LCDCN2	LCDBL	LCDMX[1:0]							000.	000.	*,*,*,- -,-,-,-
3BH	LCD0	Segment SEG1@[7:4] and SEG0@[3:0] data register of LCD								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
3CH	LCD1	Segment SEG3@[7:4] and SEG2@[3:0] data register of LCD								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
3DH	LCD2	Segment SEG5@[7:4] and SEG4@[3:0] data register of LCD								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
3EH	LCD3	Segment SEG7@[7:4]and SEG6@[3:0] data register of LCD								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
3FH	LCD4	Segment SEG9@[7:4] and SEG8@[3:0] data register of LCD								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
40H	LCD5	Segment SEG11@[7:4] and SEG10@[3:0] data register of LCD								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
41H	LCD6	Segment SEG13@[7:4] and SEG12@[3:0] data register of LCD								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
42H	LCD7					Segment SEG14@[3:0] data register of LCD				 xxxx	.... uuuu	-,-,-,- *,*,*,*"

LCDCN2：LCD Control Register 2

LCDBL：LCD digit blink controller

1：LCD digit off

0：LCD digit display

LCDMX[1:0]：LCD Waveform output controller

00：Static status (COM0)。

01：1/2 duty (COM0,COM1)。

10：1/3 duty (COM0,COM1,COM2)。

11：1/4 duty (COM0,COM1,COM2,COM3)

LCD0~LCD7：LCD Digit Data Register

14. Enhanced Universal Asynchronous Receiver Transmitter

Enhanced Universal Asynchronous Receiver Transmitter, EUART peripheral is usually called serial communications interface or SCI. The EUART can be configured as a full-duplex asynchronous system that can communicate with peripheral devices, such as CRT terminals and personal computers. It can also be configured as a half-duplex synchronous system that can communicate with peripheral devices, such as A/D or D/A integrated circuits, serial EEPROM, etc.

The Enhanced EUART implements additional features, including Frame error detection and auto address identification. Frame error detection can determine whether a frame is effective or not through frame stop bit. Auto address identification function can compare address frame content with single chip address; serial interrupt can only be generated when matching. These 2 functions are implemented through hardware circuit and software respectively.

EUART can be configured in the following modes:

- Asynchronous (full-duplex) with:
  - Auto-baud rate generator
- Frame Error Mode
  - Frame error detection [18]
  - Overrun error detection [19]
  - Hardware same bit check code(Parity check code)
- Data Transmission and reception
  - Asynchronous transmission(8 bit or 9 bit)
  - Asynchronous receive(8 bit or 9 bit)
- Auto-wake-up on character reception

EUART Registers:

URCON	ENSP[0],ENTX[0],TX9[0],TX9D[0],PARITY[1:0]
URSTA	RC9D[0],PERR[0],FERR[0],OERR[0],RCIDL[0],TRMT[0],ABDOVF[0]
BAUDCON	ENCR[0],RC9[0],ENADD[0],ENABD[0]
BRGR[15:0]	BRGRH[4:0], BRGRL[7:0]
TXREG	TXREG[7:0]
RCREG	RCREG[7:0]

[18] Frame error detection (FERR)：UART does not receive the initial bit that usually aroused from the noise on signal line. UART cannot ontain correct data from shift register.

[19] Overflow error detection (OERR)：The latest data has covered the previous data.

14.1. EUART Manual

14.1.1. To Set up an Asynchronous Data Transmission

- Configure TX of register, TRISC1 as output pin. Configure TX pinout mode of PT1M1 register.
- Configure TXIE bit of INTE2 register and GIE bit of INTE1 register to determine whether to allow transmit enabled interrupt (TXIF bit of INTF2 register is default as High, related enabled interrupt must be configured after confirmation).
- Configure BRGRH, BRGRL register to define appropriate baud rate.
- Configure ENSP bit of URCON register to start EUART serial I/O module.
- Configure TX9 bit of URCON register to decide whether the 9th bit data transmit function is enabled (If the function is enabled, the data must be written into TX9D bit. The 9th bit can be address or data).
- Configure ENTX bit of URCON register to start data transmission function.
- Write the data to TXREG register; determine to transmit data (Start transmitting after writing the data).

14.1.2. To Set up an Asynchronous Data Reception

- Configure TRISC1 register, set RC as input pin.
- Configure RCIE bit of INTE2 register and GIE bit of INTE1 register to determine whether to allow enabled interrupt reception.
- Configure BRGRH, BRGRL register, determine the appropriate baud rate.
- Configure ENSP bit of URCON register to start EUART serial I/O module.
- Configure RC9 bit of BAUDCON register to determine whether to enable the 9^{th} bit data reception function.
- Configure ENCR bit of BAUDCON register to start data reception function.
- Read RC9D bit of URSTA register to obtain the 9^{th} bit value and determine if error occurred during reception processes.
- Read RCREG register to obtain the received 8 bit data.
- Read whether the FERR bit of URSTA register is configured. Make sure if the read data is wrong. FERR can be cleared through ENCR bit.

14.1.3. To Set up an Asynchronous Data Reception (9 Bit, RS-485 Mode)

- Configure TRISC1 register. Set RC as input pin.
- Configure BRGRH, BRGRL register, to determine appropriate baud rate.
- Configure ENSP bit of URCON register to start EUART serial I/O module.
- Configure RC9 bit of BAUDCON register and determine whether to start the 9th bit data reception function.
- Configure ENADD bit of BAUDCON register enabled address detect function.
- Configure ENCR bit of BAUDCON register to start data reception function.
- Configure RCIE bit of INTE2 register and GIE bit of INTE1 register to determine whether to allow enabled interrupt reception. When data reception accomplished, RCIF bit will be configured.
- Read RC9D bit of URSTA register to obtain the 9th value of the received data. Determine whether error occurred during reception processes.
- Read RCREG register to obtain 8 bit reception data.
- Read whether FERR bit of URSTA register is configured. Make sure the creativeness of the read data. FERR bit can be cleared through ENCR bit.
- Configure ENADD bit of BAUDCON register to shut off address detect, making the next data as reception.

14.2. Baud Rate Generator, BRG

BRG is a dedicated 13 bit generator that supports asynchronous mode of the EUART. BRGR[12:0] register that controls the period of a free-running timer. Table 14-1 is the equation of baud rate computation that only apply in Master mode.

Given the desired baud rate and the operating clock source is CPU_CK, the nearest integer value for BRGR[12:0] register can be calculated using the equation in Table 14-1. From this, the error of baud rate is determined.

Example 14-1 illustrated the calculation of Baud Rate Error.

BRG/EUART MODE	Baud Rate Equation
13 bit/ asynchrony	CPU_CK÷[4 (n + 1)]
CPU_CK= operating frequency； n = BRGRH:BRGRL register correct value	

Table 14-26 Baud Rate Equation

Operating under asynchronous mode, the operating frequency is CPU_CK (4MHz) . And the desired baud rate is 9600bps. What is BRGR[12:0]=< ? > , BRGRH[7:0]:BRGRL[7:0]=< ? >

The known equation：desired baud rate＝OSC_HAO÷(4 (<BRGR[12:0]>+1))：

Therefore BRGR[12:0]＝((OSC_HAO÷desired baud rate)÷4)–1

＝((4000000÷9600)÷4)–1

＝103.2

≒103 means BRGRH[4:0]=<00>，BRGRL[7:0]=<67>；Note: 67 is hexadecimal.

Actually, BRG calculated result is: actual baud rate＝4000000÷(4×(103+1))＝9615.38

Certain error exists, the computed way is：

Error rate＝(actual baud rate– desired baud rate)/ desired baud rate

＝(9615-9600)/9600

＝0.16%

Example 14-11 Calculation of Baud Rate Error

14.2.1. Operation in Power Managed Modes

The IC clock is used to generate desired baud rate. When one the power managed modes is entered, the new clock source may be operating at different frequency. This may require adjustment to the value of BRGR[12:0] register.

14.2.2. RC Sampling

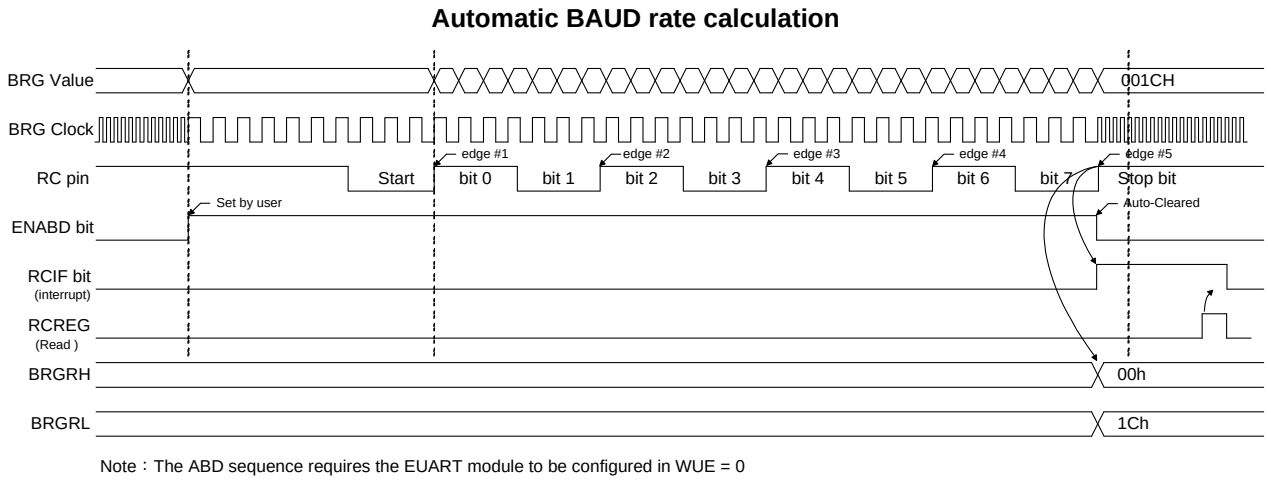
Sampling circuit will conduct sampling in the center point of baud rate period to determine if a high or a low level is presented at the RC pin.

14.2.3. Auto Baud Rate

EUART module supports auto detection and calibration of baud rate. Auto baud rate is only active in awakening start controller; WUE[0] is cleared. To start this function, ENABD[0] must be set as 1.

Auto baud rate detection begins whenever a start bit is received (the reception value must be 055H). After auto detection and calibration finished, the calculated result will be written to register, BRGRH[7:0] and BRGRL[7:0]. Relative sequences are detailed in Figure 14-1.

When BRGR[12:0] overflowed, the content generates overflow from 01FFFH to 00000H, auto baud rate flag, ABDOVF[0] will be placed 1. Users can clear ABDOVF[0] by instruction or through configure ENABD[0] as 0 to make ABDOVF[0] to be 0. After ABDOVF[0] is set up as 1, ENABD[0] status will remain as 1. Please refer to Figure 14-2.



Note：The ABD sequence requires the EUART module to be configured in WUE = 0

Figure 14-24 Auto Baud Rate Calculation Waveform

BRG Overflow Sequence

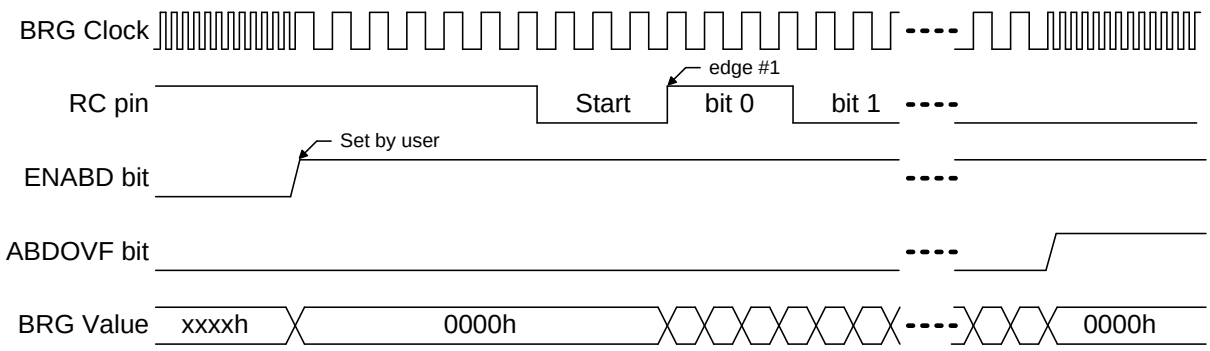


Figure 14-25 Auto Baud Rate Overflow (ABDOVF) Waveform

14.3. Hardware Parity Check

EUART supports hardware odd/even calibration function; the calibrated bit is saved in the 9th data bit. Parity check (ENADD[0]=0) is carried out in accordance with registers configuration, relative configurations are shown in Table 14-2

Transmit/Receive 8/9 bit data		PARITY	Status
TX9	RC9		
0	0	0	Transmit/Receive data no parity check message
0	0	1	Transmit/Receive data no parity check message
0	1	0	Data received has parity l check function, even parity
0	1	1	Data received has parity check function, odd parity
1	0	0	Data transmitted has parity check function, even parity
1	0	1	Data transmitted has parity check function, odd parity
1	1	0	
1	1	1	
Note: When RC9[0] is configured as 1, parity check function is started. When even/odd parity error, PERR[0] is configured as 1. If RC9[0] and ENADD[0] is configured as a in the same time, do not care error value of PERR[0] bit.			

Table 14-27 Parity Check Status

14.4. EUART Asynchronous Mode

In this mode, the EUART uses standard "Non-Return-to-Zero, NRZ" format (one Start bit, eight or nine data bits and one Stop bit). The most common data format is 8 bit. An on-chip dedicated 13-bit Baud Rate Generator can be applied to derive standard baud rate frequencies from the oscillator.

Moreover, the EUART transmits and receives the last LSB. The transmitter and receiver are functionally independent but use the same data format and baud rate. Parity is supported by hardware and can be stored as the 9th data bit.

14.4.1. EUART Asynchronous Transmitter

Figure 14-3 is the sequence of EUART transmitter. The core of the transmitter is Transmit Shift Register, TSR, users cannot read/write TSR.

TSR obtains data from the Read/Write Transmit Buffer register, TXREG[7:0]. The TXREG[7:0] register is loaded with data in software. The TSR register is not loaded until the Stop bit has been transmitted from the previous load. Once the Stop bit is transmitted, the TSR is loaded with new data from the TXREG register (if available).

As soon as the TXREG register transfer the data to the TSR register, the TXREG register is empty and TXIF flag is set from 1 to 0 (when ENTX bit of URCON register is configured, TXIF bit will be configured as 1). TXIF will not be cleared immediately when new data is loaded. Instead, it is cleared in the second instruction period following the load instruction. TXIF will be configured as 1 again at the end of one instruction cycle. The interrupt can be enabled or disabled by setting or clearing the interrupt enable bit, TXIE. TXIE will be set regardless of the state of TXIE; when interrupt occurs, TXIF will be configured from 1 to 0 and cannot be cleared in software. TXIF will be configured as 1 again at the end of one instruction cycle. If TSR register data has not been transmitted from the previous load and data has been written into TXREG register. TXIF is cleared in the second instruction period following the load instruction. TXIF will be configured as 1 again when Stop bit occurred.

Hence, after new data is loaded into TXREG register, TXIF will indicate the status of TXREG. Another bit, TRMT, presents TSR register status. TRMT is only readable, it will be set up as 1 when TSR register is empty (no loaded action). TRMT bit has no connection with any interrupt logic. Hence, users can only check to the status of TSR to determine whether it is empty. Asynchronous data transmission sequence can be referred to Figure 14-4 and Figure 14-5.

- UART actions are irrelevant to CPU instruction cycle except read and write action.
- TXIF and RCIF is for interrupt purpose, they are irrelevant to other events.
- When using CPU to monitor peripheral components, be cautioned about the corresponding operating speed.

EUART TRANSMIT BLOCK DIAGRAM

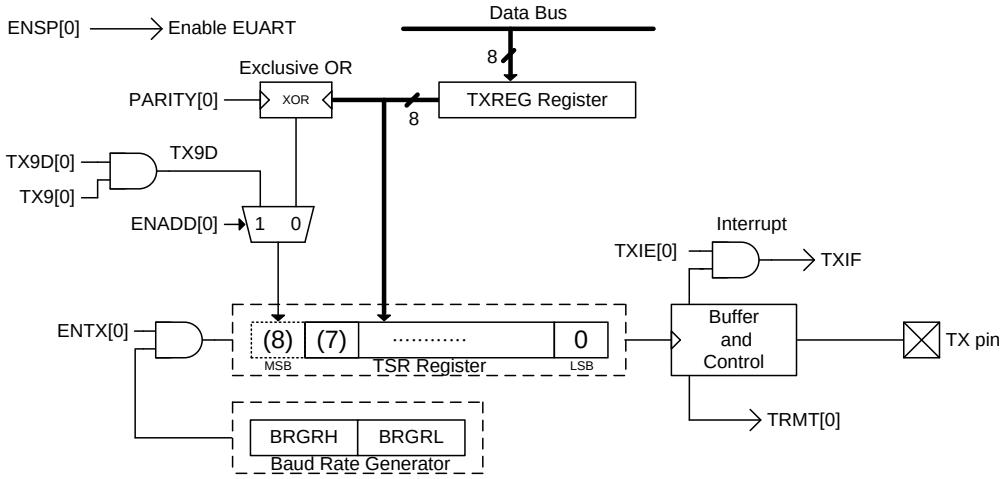


Figure 14-26 EUART Transmission Block Diagram

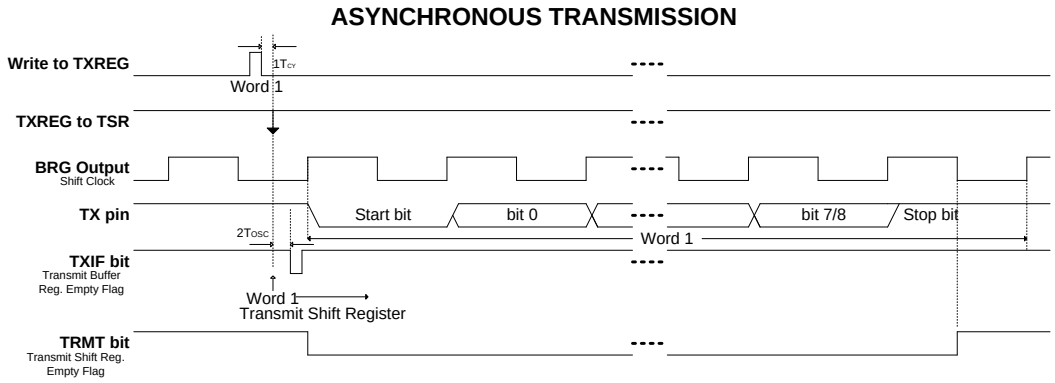


Figure 14-27 Asynchronous Transmission Sequence

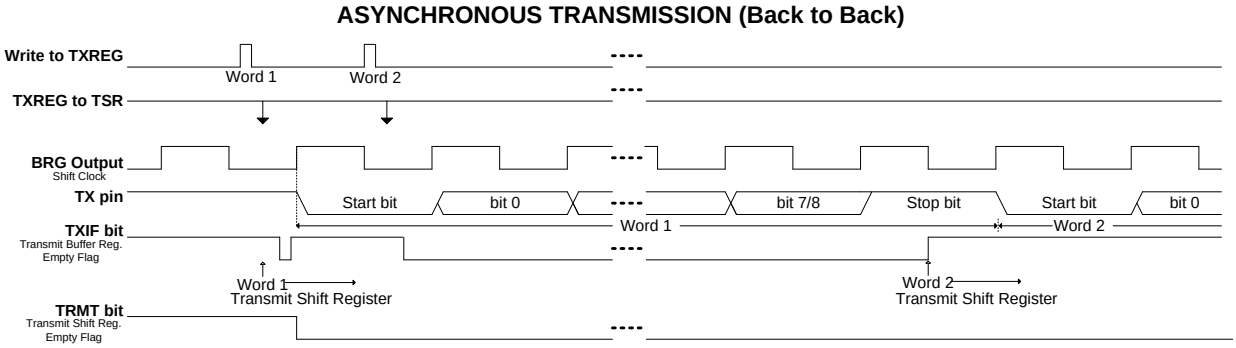


Figure 14-28 Asynchronous Transmission Sequence (Back to Back)

14.4.2. EUART Asynchronous Receiver

Figure 14-6 and Figure 14-7 receiver block diagram while Figure 14-8 illustrates asynchronous receive sequence. The data is received on the RC pin drives the data recovery circuit. Data recovery circuit is actually a high-speed shifter that operating at 13-bit auto baud rate, whereas the main receive serial shifter operates at baud rate or at OSC_HAO. This mode is typically be used in RS-232 systems.

If RC pin does not receive complete byte (Start bit, 8(9) bit data, Close bit), FERR bit will be set as 1 and it can be cleared by ENCR bit.

When RC pin has received two complete byte data, OERR bit will be configured as 1 when receiving the third complete byte data (have not read the data of RCREG register). OERR bit can be cleared by ENCR bit.

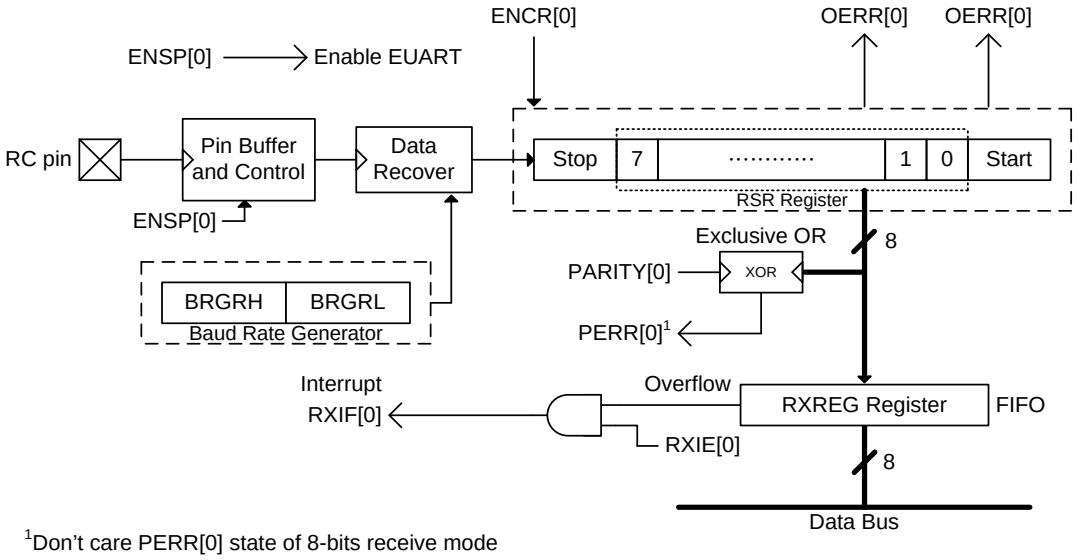
When complete data reception accomplished, RCIF bit of INTF2 register will be set up. Instructions cannot clear RCIF bit once it has been configured. RCIF status can be cleared by reading RCREG register.

RCIDL bit of URSTA register will reflect if it is in reception status. By determining the status, users can know whether the data reception has been completed.

When receiving data, hardware will conducts the received 8 bit data exclusive or. If RC9 is set as 1, the received RC9D data (9 bit) will be calculated by exclusive or. After

operation, the result will be calculated again by exclusive or with PARITY bit and it will be displayed in PERR bit. If the received data is correct, PERR is configured as 0. Conversely, if the data received is incorrect, PERR will be set as 1. PERR bit cannot be cleared in software. PERR will be set as 0 whenever the next data is being correctly received.

EUART 8-BITs RECEIVE BLOCK DIAGRAM



[1]Don't care PERR[0] state of 8-bits receive mode

Figure 14-29 EUART 8-bits Receive Block Diagram

EUART 9-BITs RECEIVE BLOCK DIAGRAM

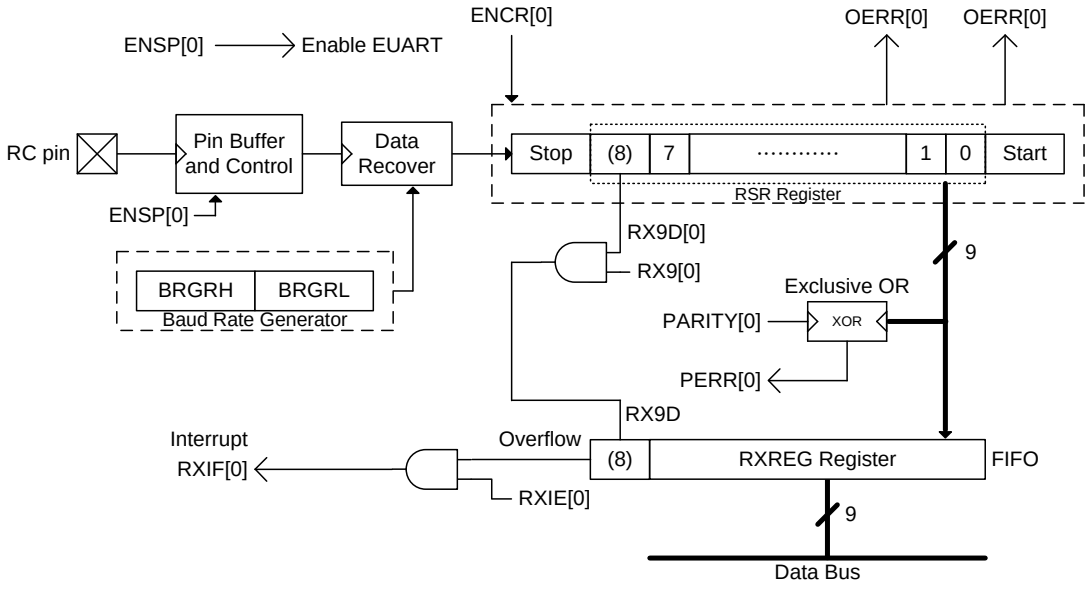


Figure 14-30 EUART 9-bits Receive Block Diagram

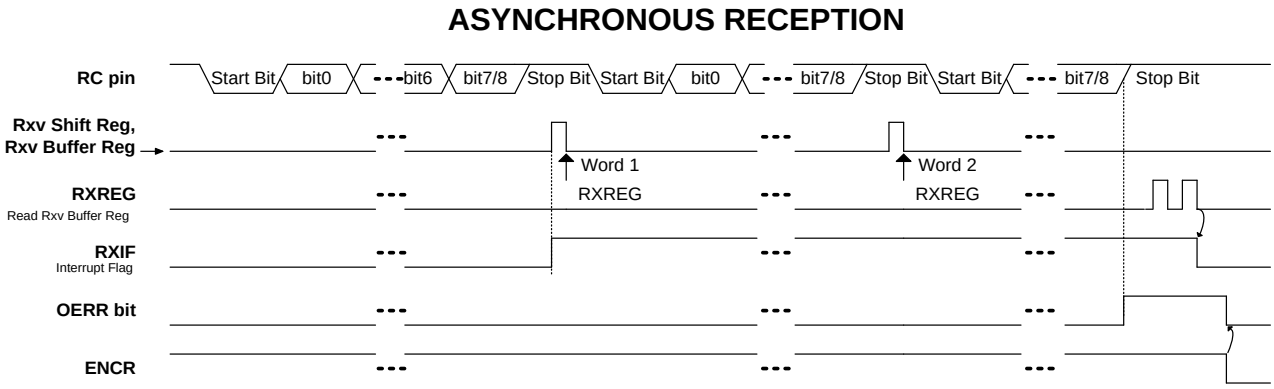


Figure 14-31 Asynchronous Receive Sequence

14.4.3. 9 Bit Address Detect Mode

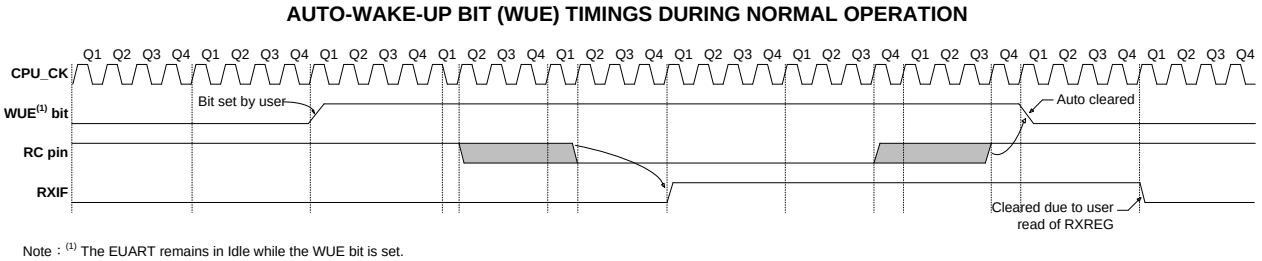
This mode is typically be used in RS-485 systems. Users can operate asynchronous reception with reference to ***EUART Manual*** section. ENADD bit of BAUDCON register can be used to configure address detect or data detect.

14.4.4. Auto-Wake-Up on Character Reception

Under Sleep mode or Idle mode, all clocks to the EUART are suspended. As a result, the Baud Rate Generator is inactive and a correct byte reception cannot be conducted. The auto-wake-up function allows the controller to be awakened up when the activity on the RC line while the EUART is operating in Asynchronous mode. The auto-wake-up function is enabled by configuring the WUE bit of URCON register. After initiation, the typical receive sequence on RC is disabled and the EUART remains in an Idle state, monitoring for a wakeup event（it is not related with CPU Run mode）.

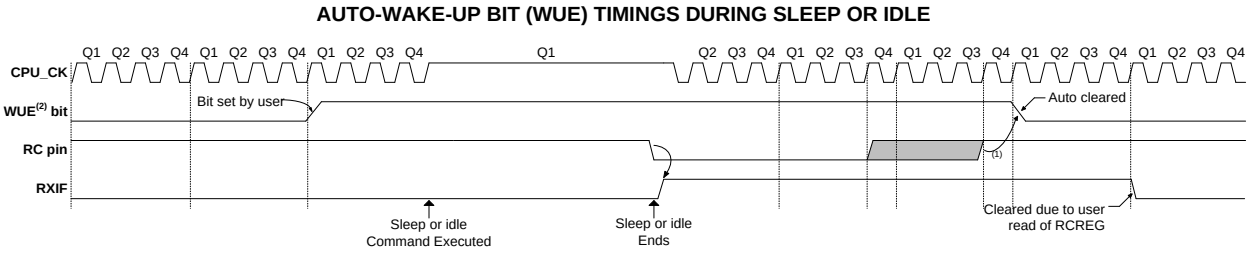
A wake up event consists of a high state to low state transition on the RC line. Followed by a wake up event, the module generates an RCIF interrupt. The interrupt is generated synchronously to the Q clock in normal operating mode. (Please refer to Figure 14-9); If the IC is in Sleep or Idle mode, it is asynchronously. (Please refer to Figure 14-10). The interrupt is cleared by reading RCREG register.

After wake up event, when low state to high state transition occurs on the RC line, WUE bit is automatically be cleared. At this time, EUART module returns to normal Run mode from Idle mode. Users can know from the signals that the event is over.



Note：(1) The EUART remains in Idle while the WUE bit is set.

Figure 14-32 Auto-Wake-Up Sequence in Normal Mode



Note：(1) If the wake-up event requires long oscillator warm-up time, the auto-clear of the WUE bit can occur before the oscillator is ready. This sequence should not depend on the presence of Q clocks.
(2) The EUART remains in Idle while the WUE bit is set.

Figure 14-33 Auto-Wake-Up Sequence in Sleep or Idle Mode

14.4.5. Notice of Using Auto-Wake-Up Function

Due to the fact that auto-wake-up functions by sensing rising edge transitions on RC, information with any state changes before the Stop bit may output a false character and result in data or frame errors. Thus, the initial character in the transmission must be all "0" bit. This can be 00h (8 bit) for standard RS-232 ICs.

Oscillator start-up time must be considered as well, particularly in applying oscillators with longer start-up delay. The auto-wake-up character must be of sufficient length and of sufficient length of time interval to allow enough time for the selected oscillator to start and offer appropriate initialization of the EUART.

14.4.6. Notice of Using WUE Bit

Using WUE and RCIF event timing to determine the validity of received data may bring about some confusion. As noted, setting the WUE as 1 may place the EUART to an standby mode. The wake up event generates a receive interrupt and RCIF is placed 1. The WUE bit is cleared after a rising edge is seen on RC. The interrupt condition is cleared by reading the RCREG register.

Under normal condition, the data of RCREG after wake up is ineffective and should be discarded. The fact that WUE bit has been cleared (or is still set as 1) and RCIF flag is set should not be used as an indicator of the integrity of the data in RCREG. Users should consider deploying a firmware method to verify received data integrity. In order to assure no effective data is lost, check the RCIDL bit to verify that a receive operation is not in progress. If a receive operation is not executed, the WUE can be placed 1, forcing the IC entering the Sleep mode.

```
BSF     INTE2,6,0       ; Configure RCIE receive interrupt service
MVL     010H            ; Configure PT14 (TX) as digital output pin
MVF     TRISC1,1,0      ; Configure PT13 (RX) as digital input pin
BSF     PT1M1,4,0
MVL     000H            ; Set Baud rate as 9600hz
MVF     BRGRH,1,0
MVL     067H
MVF     BRGRL,1,0
MVL     0F0H            ; Start EUART function, initiate data transmission and
                          the 9th bit output
MVF     URCON,1,0       ; Configure TX9D=1. Set parity check code as 0
MVL     00CH            ; Enable data reception, start the 9th bit reception, the 9th
                          bit is data
MVF     BAUDCON,1,0
MVL     055H            ; Write EUART transmission data
MVF     TXREG,1,0
....
RC Interrupt :          ; Receive interrupt event service program
BCF     INTF2,RCIF,0    ; Clear receive interrupt event flag
BTSZ    URSTA,5,0       ; Determine if PERR bit is 0, to assure data integrity
JMP     FAIL_LOOP       ; Data error determine loop
MVFF    RCREG,BUF0      ; Move the received data to BUF0 register
MVFF    URSTA,BUF1      ; Move the 9th received data to BUF1 register
....
RETI                    ; Return from interrupt
FAIL_LOOP:              ; Data receive error  loop
```

Figure 14-34 EUART Example Program

14.5. Register Description-EUART

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1												
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition												
Address	**File Name**	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**	**A-RESET**	**i-RESET**	**R/W**
23H	INTE1	GIE								0.0. 0000	0.0. 0000	*,-,*,- *,*,*
24H	INTE2	TXIE	RCIE							0000 000.	0000 000.	*,*,*,*,*,*,*,-
27H	INTF2	TXIF	RCIF							0000 000.	0000 000.	*,*,*,*,*,*,*,-
46H	URCON	ENSP	ENTX	TX9	TX9D	PARITY			WUE	0000 0..0	0000 0..0	*,*,*,*,-,-,*
47H	URSTA		RC9D	PERR	FERR	OERR	RCIDL	TRMT	ABDOVF	.000 0110	.000 0110	-,r,r,r r,r,r,rw0
48H	BAUDCON					ENCR	RC9	ENADD	ENABD	 0000	 0000	-,-,-,- *,*,*,*
49H	BRGRH	Baud Rate Generator Register High Byte								...x xxxx	...u uuuu	-,-,-,- *,*,*,*
4AH	BRGRL	Baud Rate Generator Register Low Byte								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
4BH	TXREG	UART Transmit Register								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
4CH	RCREG	UART Receive Register								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
4DH	PT1				PT1.4	PT1.3				xxxx xxxx	uuuu uuuu	*,*,*,* r,r,r,r
4EH	TRISC1				TC1.4	TC1.3				0000 0000	0000 0000	*,*,*,*,*,*,*,*
4FH	PT1PU				PU1.4	PU1.3				0000 0000	0000 0000	*,*,*,*,*,*,*,*
50H	PT1M1				PM1.4					0000 0000	0000 0000	*,*,*,*,*,*,*,*

Table 14-28 EUART Register

INTE1/INTE2/INTF2：Please refer to Interrupt Chapter.

PT1/TRISC1/PT1PU/PT1M1：Please refer to Input/Output Port, I/O Chapter.

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1												
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition												
Address	File Name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	A-RESET	i-RESET	R/W
23H	INTE1	GIE								0.0. 0000	0.0. 0000	*,-,*,- *,*,*,*
24H	INTE2	TXIE	RCIE							0000 000.	0000 000.	*,*,*,*,*,*,*,-
27H	INTF2	TXIF	RCIF							0000 000.	0000 000.	*,*,*,*,*,*,*,-
46H	URCON	ENSP	ENTX	TX9	TX9D	PARITY			WUE	0000 0..0	0000 0..0	*,*,*,*,-,-,*
47H	URSTA		RC9D	PERR	FERR	OERR	RCIDL	TRMT	ABDOVF	.000 0110	.000 0110	-,r,r,r r,r,r,rw0
48H	BAUDCON					ENCR	RC9	ENADD	ENABD	 0000	 0000	-,-,-,- *,*,*,*
49H	BRGRH	Baud Rate Generator Register High Byte								...x xxxx	...u uuuu	-,-,-,- *,*,*,*
4AH	BRGRL	Baud Rate Generator Register Low Byte								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
4BH	TXREG	UART Transmit Register								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
4CH	RCREG	UART Receive Register								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
4DH	PT1				PT1.4	PT1.3				xxxx xxxx	uuuu uuuu	*,*,*,* r,r,r,r
4EH	TRISC1				TC1.4	TC1.3				0000 0000	0000 0000	*,*,*,*,*,*,*,*
4FH	PT1PU				PU1.4	PU1.3				0000 0000	0000 0000	*,*,*,*,*,*,*,*
50H	PT1M1				PM1.4					0000 0000	0000 0000	*,*,*,*,*,*,*,*

URCON: UART Control Register

ENSP：UART port function bit

1：Start UART port and dispose TX and RC pins for UART port

0：Close UART port and dispose TX and RC pins for I/O.

Note：When UART series port is initiated, input or output pin must be configured appropriately.

ENTX：UART transmission function enable bit

1：Enable

0：Disable

TX9：Enable the transmission of the ninth bit

1：Enable

0：Disable

TX9D：Send the ninth bit data

1：Data is "1"

0：Data is"0"

PARITY：Odd/even bit check

1：Odd bit check

0：Even bit check

WUE：Character receiving auto wake up enable bit

1：Enable

0：Disable

URSTA: UART Status Register

RC9D：Receiving the ninth bit data

1：Data is"1"

0：Data is"0"

PERR：Data parity check result flag

1：Receiving parity check error

0：Receiving parity check correctness

FERR：Incomplete UART data receiving (start, 8(9) bit data, end) flag

1：Represents incomplete data

0：Represents complete data

OERR：Received the second unprocessed data flag

1：Happened

0：Not happened

RCIDL：Response whether it is in receiving status flag

1：In receiving status

0：Not in receiving status

TRMT：Represents transmission register (TSR) status flag

1：Represents TSR register is empty

0：Represents TSR register has data

ABDOVF：Auto Baud Rate overstate flag

1：Happened

0：Not happened

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1												
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition												
Address	File Name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	A-RESET	i-RESET	R/W
23H	INTE1	GIE								0.0. 0000	0.0. 0000	*,-,*,- *,*,*
24H	INTE2	TXIE	RCIE							0000 000.	0000 000.	*,*,*,*,*,*,*,-
27H	INTF2	TXIF	RCIF							0000 000.	0000 000.	*,*,*,*,*,*,*,-
46H	URCON	ENSP	ENTX	TX9	TX9D	PARITY			WUE	0000 0..0	0000 0..0	*,*,*,*,-,-,*
47H	URSTA		RC9D	PERR	FERR	OERR	RCIDL	TRMT	ABDOVF	.000 0110	.000 0110	-,r,r,r r,r,r,rw0
48H	BAUDCON					ENCR	RC9	ENADD	ENABD	 0000	 0000	-,-,-,- *,*,*,*
49H	BRGRH	Baud Rate Generator Register High Byte								...x xxxx	...u uuuu	-,-,-,- *,*,*,*
4AH	BRGRL	Baud Rate Generator Register Low Byte								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
4BH	TXREG	UART Transmit Register								xxxx xxxx	uuuu uuuu	*,*,*,*,*,*,*,*
4CH	RCREG	UART Receive Register								xxxx xxxx	uuuu uuuu	r,r,r,r r,r,r,r
4DH	PT1				PT1.4	PT1.3				xxxx xxxx	uuuu uuuu	*,*,*,* r,r,r,r
4EH	TRISC1				TC1.4	TC1.3				0000 0000	0000 0000	*,*,*,*,*,*,*,*
4FH	PT1PU				PU1.4	PU1.3				0000 0000	0000 0000	*,*,*,*,*,*,*,*
50H	PT1M1				PM1.4					0000 0000	0000 0000	*,*,*,*,*,*,*,*

BAUDCON：UART receiving data control register

ENCR：Data receiving function enable bit

1：Enable

0：Disable

RC9：The ninth bit receiving function enable bit

1：Enable

0：Disable

ENADD：Address detection bit

1：Enable

0：Disable

ENABD：Auto Baud rate controller enable bit

1：Enable

0：Disable

BRGRH/BRGRL: Baud rate Control Register

TXREG: UART transmit Control Register

RCREG: UART receive Control Register

PT1: PORT1 status control register

PT1.4：External pin control bit (PT1.4)

1：High potential flag or high potential output.

0：Low potential flag or low potential output.

PT1.3：External pin control bit (PT1.3)

1：High potential flag or high potential output.

0：Low potential flag or low potential output.

TRISC1: Input/output control register

TC1.4：External pin input/output control bit 4

1：Output

0：Input

TC1.3：External pin input/output control bit 3

1：Output

0：Input

PT1PU: Pull up resistor control register

PU1.4：External pin pull up resistor control bit 4

1：Enable

0：Disable

PU1.3：External pin pull up resistor control bit 3

1：Enable

0：Disable

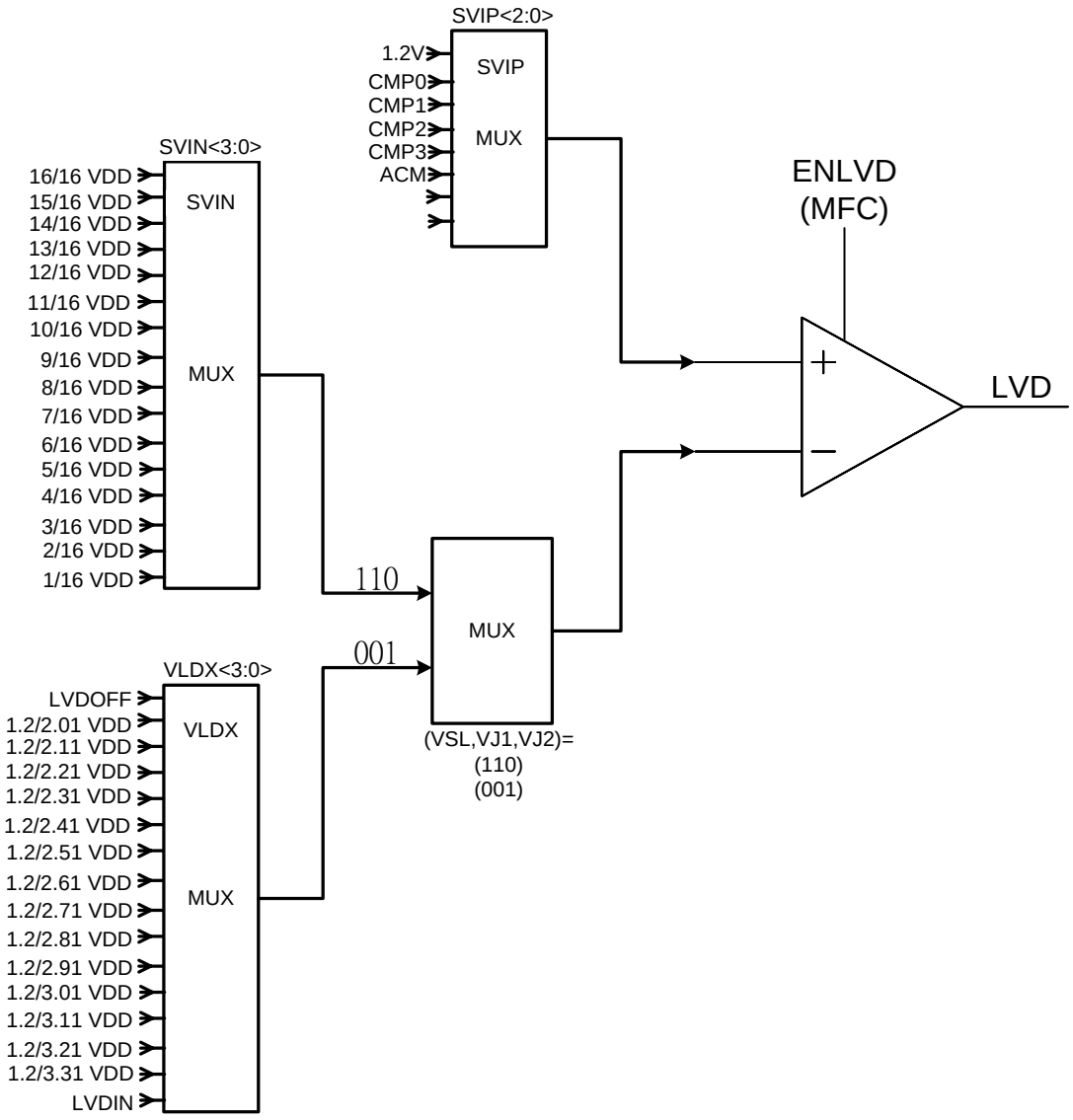
PT1M1: Digital output mode select register

PM1.4：PT1.4 EUART TX output control bit

1：Enable

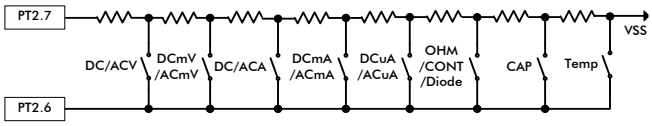
0：Disable

15. Multi-function Comparator



Multi-function Comparator is composed of VDD divide voltage network, input multiplexer and comparator. The main function is to select VDD divide voltage and compare with internal 1.2V to constitute 2~3.3V low battery voltage determination. Comparing external input to internal 1.2V can form an external comparator of low battery voltage detection. Moreover, comparing external input pins (PT2.4~PT2.7) to VDD divide voltage to form a simple 4 bits ADC.

15.1. Scan Key Description



Take the above graph as an example; PT2.7 is Digital output that connected with 8 external resistors (20K ohm) as to generate different divide voltage. When PT2.6 is set as analog input and MFC is switched as Scan Key mode, it can compare voltage with different internal divide voltage nodes to determine which range it should belong to (it is suggested to keep every scan key in max. 8 ranges as to avoid VDD power noise and comparator noise). When in sleep mode, PT2.7 can be set as output low to deduce power consumption.

15.2. Example Program

To be provided.

15.3. Register Description- Multi-function Comparator

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1												
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition												
Address	File Name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	A-RESET	i-RESET	R/W
2DH	LVDCN1	ENLVD	LVD	VJ1	VJ2	VLDX[3:0]						
2EH	LVDCN2	VSL	SVIN[2:0]			SVIP[3:0]						
2FH	SBMSET1	SKRST		HAOTR[5:0]						x.xx xxxx	u.uu uuuu	*,-,*,*,*,*,*

LVDCN1：Multi-function Comparator Control Register 1

ENLVD：Multi-Function comparator enable controller

1：Active

0：Inactive

LVD：Comparator react flag

1：Happened

0：Not happened

VSL,VJ1,VJ2：Source divider controller

111：Inactive

110：Select VDD as 4 Bits divide voltage mode (SVIN)

101：Inactive

100：Inactive

011：Inactive

010：Inactive

001：Select low battery voltage detection mode (LVDX).

000：Inactive

LVDX[3:0]：Low battery voltage detection mode (must configure: VSL, VJ1, VJ2=001)

VLDX<3:0>	0000	0001	0010	0011	0100	0101	0110	0111
VLDX	LVDOFF	1.2/2.01VDD	1.2/2.11VDD	1.2/2.21VDD	1.2/2.31VDD	1.2/2.41VDD	1.2/2.51VDD	1.2/2.61VDD
VLDX<3:0>	1000	1001	1010	1011	1100	1101	1110	1111
VLDX	1.2/2.71VDD	1.2/2.81VDD	1.2/2.91VDD	1.2/3.01VDD	1.2/3.11VDD	1.2/3.21VDD	1.2/3.31VDD	LVDIN

LVDCN2:

SVIP: Multiplexer inputted via comparator positive end

SVIP	000	001	010	011	100	101
SVIP	1.2V	CMP0	CMP1	CMP2	CMP3	ACM

SVIN: VDD is 4 Bits divide power mode (must configure: VSL, VJ1, VJ2=110)

SVIN<3:0>	0	1	10	11	100	101	110	111
SVIN	16/16VDD	15/16VDD	14/16VDD	13/16VDD	12/16VDD	11/16VDD	10/16VDD	9/16VDD
SVIN<3:0>	1000	1001	1010	1011	1100	1101	1110	1111
SVIN	8/16VDD	7/16VDD	6/16VDD	5/16VDD	4/16VDD	3/16VDD	2/16VDD	1/16VDD

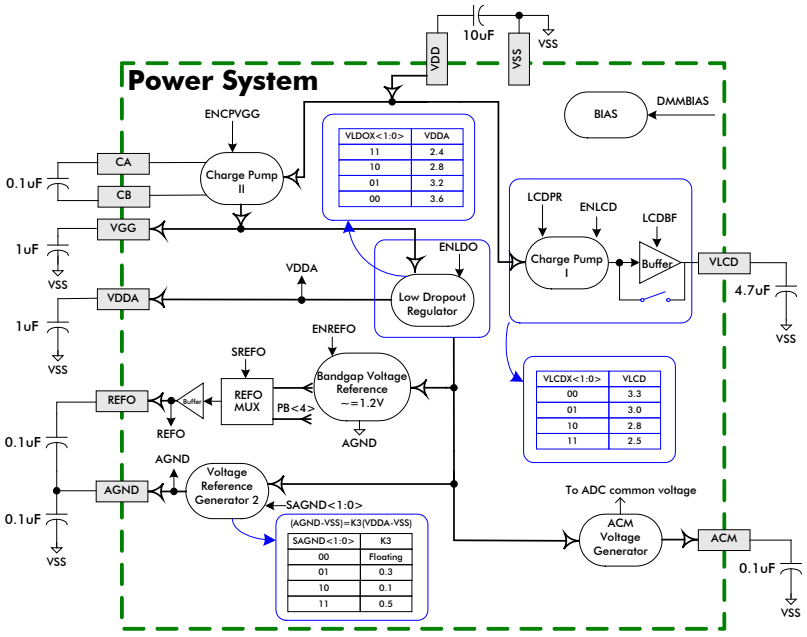
15.4. Low Power Voltage Detection Description

15.4.1. LVD Initial Setup

Configure voltage monitor selector, VLDX[3:0], can determine whether LVD detection voltage is generated by divide voltage of operation voltage or by external input pin, LVDIN.

When VLDX[3:0] is set up as detection operation voltage, 14-step of voltage monitor points can be achieved by matching the divide voltage resistors. If it is set as external input, users may need to design divide voltage circuit to produce proper voltage signal and input via LVDIN pin to LVD comparator.

16. Charge Pump Regulator and VDDA LDO



In order to ensure that ADC can measure AC500mV signal, the operation voltage of DMM Analog Front End must achieve 3.6V and the input power source of HY12P65 must ranges from 2.4V~3.6V. HY12P65 has built-in Charge Pump Regulator and LDO to generate operation voltage (VDDA) of DMM Analog Front End.

Charge Pump Regulator connects capacitors through CA and CB and boosts VGG to 4.2V. CA and CB connects to 0.1~1μF while CA connects to positive end of capacitor and VGG connects to1~10μF capacitor for power regulator.

VDDA LDO input power source is VGG; VDDA output can be regulated at 2.4V to 3.6V by program. VDDA must connect to 1μF capacitor for power regulator.

16.1. Register Description- Charge Pump Regulator

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1											
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition											
Address	**File Name**	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**	**A-RESET**	**i-RESET**
5BH	PWRCN					-	ENLDO	LDOC[1:0]			
5CH	PWRCN2		ENCPVGG								

PWRCN:

ENLDO：Register bit, can enable VDDA LDO.

1：Active

0：Inactive

Note：When ENLDO is activated, REFO will be enabled as well.

LDOC[1:0]：Register bit, can select LDO output voltage

00=3.6V

01=3.2V

10=2.8V

11=2.4V

PWRCN2:

ENCPVGG：Register bit, can enable charge pump regulator.

1：Active

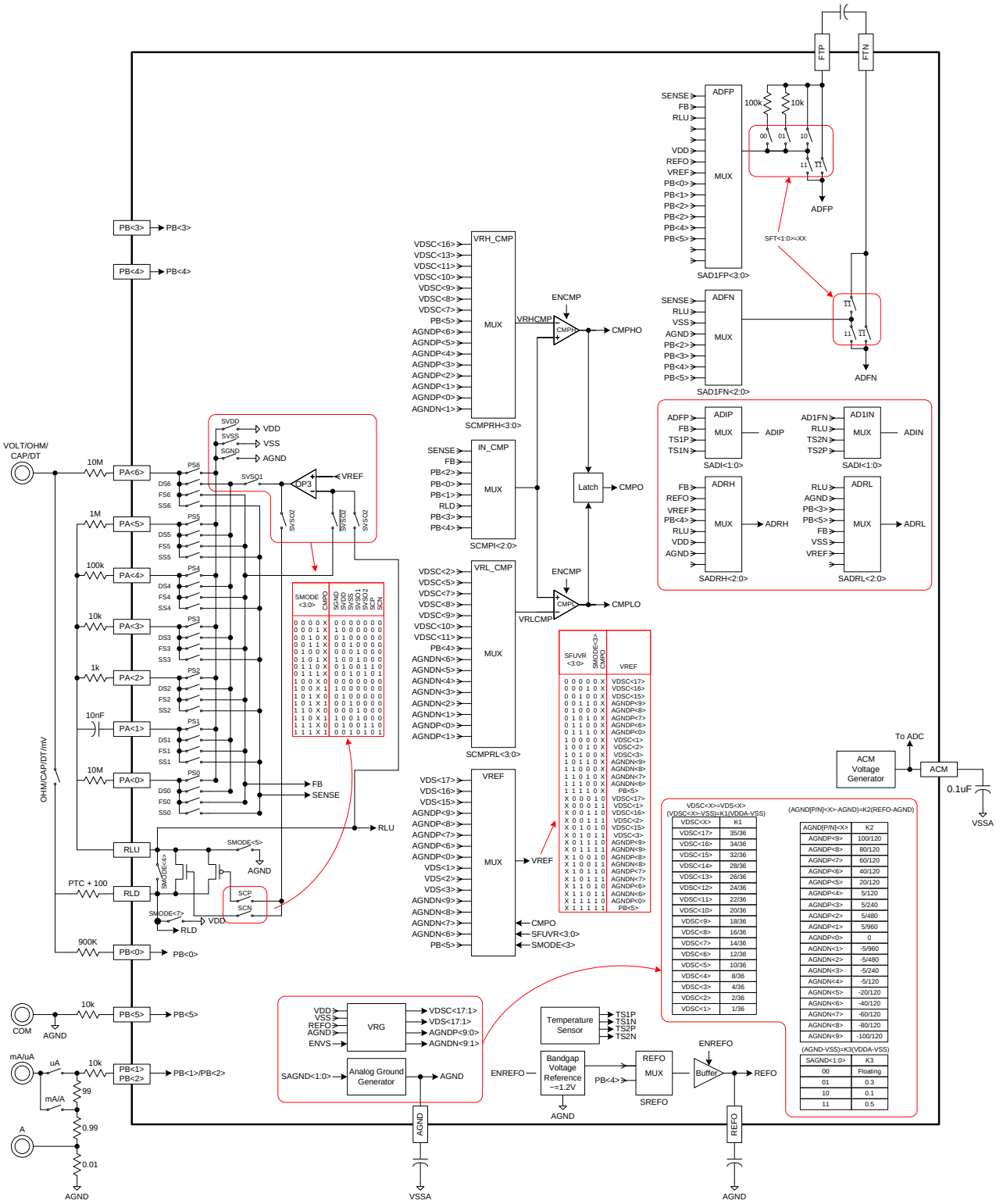
0：Inactive

16.2. Power on Example Program

Example code：

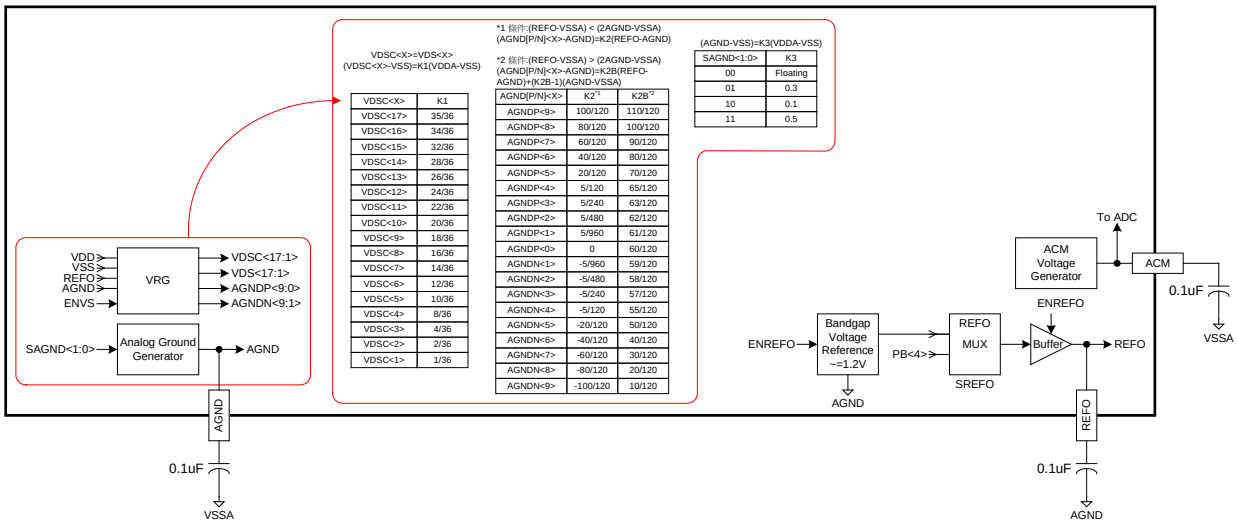
```
MVL     11000000b   ;Configure MCUBIAS，enable Charge pump regulator
MVF     PWRCN2,F,A
MVL     11111100b   ；Configure DMMBIAS，AGND voltage selection 0.5xVDDA，enable Voltage
                    ; Reference Generator，Enable VDDA LDO, LDO output voltage selection 3.6V
MVF     PWRCN,F,A
```

17. Auto Range DMM Multi-Function Network



Auto Range DMM Multi-Function Network includes Voltage Reference Generator, Analog Switch Network, DMM Comparator Network and Pre-Filter, ADC Input MUX and Temperature Sensor. Detailed description will be separately given in below.

17.1. Voltage Reference Generator (VRG)



VDSC<X>	K1
VDSC<17>	35/36
VDSC<16>	34/36
VDSC<15>	32/36
VDSC<14>	28/36
VDSC<13>	26/36
VDSC<12>	24/36
VDSC<11>	22/36
VDSC<10>	20/36
VDSC<9>	18/36
VDSC<8>	16/36
VDSC<7>	14/36
VDSC<6>	12/36
VDSC<5>	10/36
VDSC<4>	8/36
VDSC<3>	4/36
VDSC<2>	2/36
VDSC<1>	1/36

AGND[P/N]<X>	K2*1	K2B*2
AGNDP<9>	100/120	110/120
AGNDP<8>	80/120	100/120
AGNDP<7>	60/120	90/120
AGNDP<6>	40/120	80/120
AGNDP<5>	20/120	70/120
AGNDP<4>	5/120	65/120
AGNDP<3>	5/240	63/120
AGNDP<2>	5/480	62/120
AGNDP<1>	5/960	61/120
AGNDP<0>	0	60/120
AGNDN<1>	-5/960	59/120
AGNDN<2>	-5/480	58/120
AGNDN<3>	-5/240	57/120
AGNDN<4>	-5/120	55/120
AGNDN<5>	-20/120	50/120
AGNDN<6>	-40/120	40/120
AGNDN<7>	-60/120	30/120
AGNDN<8>	-80/120	20/120
AGNDN<9>	-100/120	10/120

SAGND<1:0>	K3
00	Floating
01	0.3
10	0.1
11	0.5

AGND is Analog ground and its pin must connect to 0.1μF capacitor for power regulator. REFO offers voltage reference that measurement required and its relative voltage to AGND is about 1.2V, the pin must connect to an external 0.1μF capacitor for power regulator. REFO Buffer input is selected by SREFO, users can choose from internal Band-gap Voltage Reference or PB<4> voltage. ACM Voltage Generator produces common mode reference point of internal ADC. ACM pin must connect to 0.1μF capacitor for power regulator. Voltage reference generator (VRG) produces different voltage reference for ADC and Comparator usage. VDS<17:1> is the divided voltage that resulted from VDDA-VSS while AGNDP<9:0> and AGNDN<9:1> are the divided voltage that resulted from +/-REFO-VSS.

17.1.1. Register Description- Voltage Reference

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1											
".unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition											
Address	**File Name**	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**	**A-RESET**	**i-RESET**
5BH	PWRCN	DMMBIAS	SAGND[1:0]		ENVS	-					
5DH	ADCN1	SDIO	SREFO	SFT1<1:0>		SFUVR<3:0>					

PWRCN:

DMMBIAS：Bias current, providing LDO power to all analog IP

SAGND[1:0]：Register bit can select AGND power.

00：Disable AGND Generator and AGND pin is in floating status

01：0.3xVDDA

10：0.1xVDDA

11：0.5xVDDA

ENVS：Register bit, can enable Voltage Reference Generator.

1：Enable

0：Disable

VDSC<N> is the node that generated from dividing (VDDA,VSS) voltage, AGNDP<N> is the node that generated from dividing (REFO,AGND) voltage and AGNDN<N> is the node that generated from dividing (-REFO,AGND) voltage. All relative voltage is listed in below.

*1 condition:(REFO-VSSA) < (2AGND-VSSA)
(AGND[P/N]<X>-AGND)=K2(REFO-AGND)

*2 condition:(REFO-VSSA) > (2AGND-VSSA)
(AGND[P/N]<X>-AGND)=K2B(REFO-AGND)+(K2B-1)(AGND-VSSA)

VDSC<X>=VDS<X>
(VDSC<X>-VSS)=K1(VDDA-VSS)

VDSC<X>	K1
VDSC<17>	35/36
VDSC<16>	34/36
VDSC<15>	32/36
VDSC<14>	28/36
VDSC<13>	26/36
VDSC<12>	24/36
VDSC<11>	22/36
VDSC<10>	20/36
VDSC<9>	18/36
VDSC<8>	16/36
VDSC<7>	14/36
VDSC<6>	12/36
VDSC<5>	10/36
VDSC<4>	8/36
VDSC<3>	4/36
VDSC<2>	2/36
VDSC<1>	1/36

AGND[P/N]<X>	K2*1	K2B*2
AGNDP<9>	100/120	110/120
AGNDP<8>	80/120	100/120
AGNDP<7>	60/120	90/120
AGNDP<6>	40/120	80/120
AGNDP<5>	20/120	70/120
AGNDP<4>	5/120	65/120
AGNDP<3>	5/240	63/120
AGNDP<2>	5/480	62/120
AGNDP<1>	5/960	61/120
AGNDP<0>	0	60/120
AGNDN<1>	-5/960	59/120
AGNDN<2>	-5/480	58/120
AGNDN<3>	-5/240	57/120
AGNDN<4>	-5/120	55/120
AGNDN<5>	-20/120	50/120
AGNDN<6>	-40/120	40/120
AGNDN<7>	-60/120	30/120
AGNDN<8>	-80/120	20/120
AGNDN<9>	-100/120	10/120

ENREFO：Register bit, can enable internal bandgap Voltage Reference and start REFO Buffer

1：Enable

0：Disable, REFO pin is in floating state

ADCN1:

SDIO：Short control bit of PB<0> and PB<2>

0：Open

1：Short

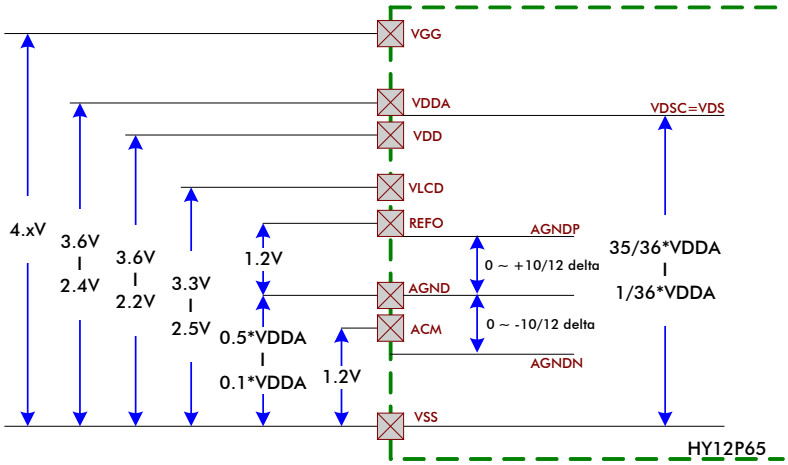
SREFO：Register bit, can select REFO Buffer input source

0：Select internal Band-gap Voltage Reference

1：Select PB<4> pin

17.2. Power System

The power relation graph is as follows:



17.3. AGND for Different Applications

In DMM application, AGND voltage is set as 1/2 VDDA mostly. In resistor range, due to the reason that resistors are all positive, by configuring AGND as 0.3 VDDA can enlarge the input voltage when measuring resistors and enhance resolution. In Diode range measurement, for the linearity requirements of measuring barrier potential is looser and the ability to measure the largest open circuit is important, AGND is then configured as 0.1 VDDA.

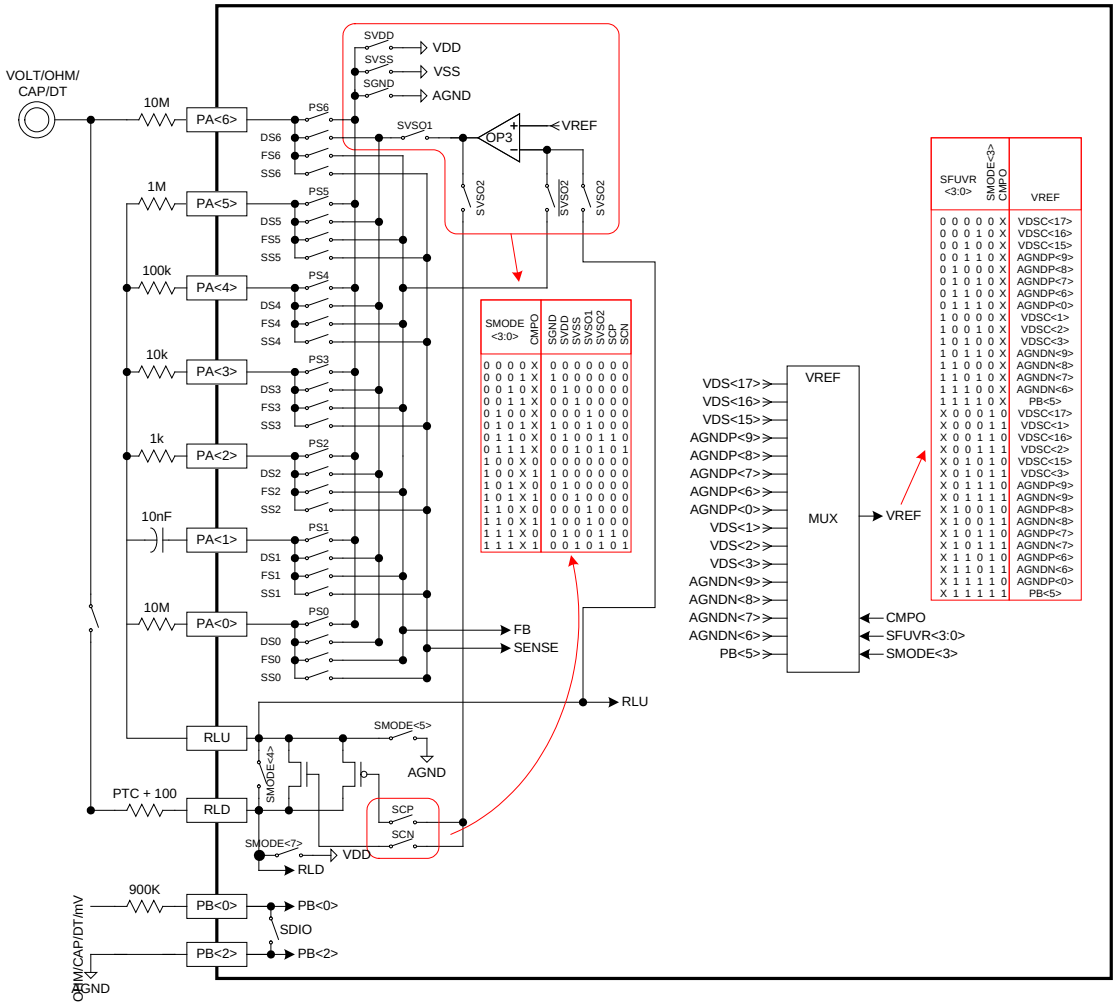
(AGND-VSS)=K3(VDDA-VSS)

SAGND<1:0>	K3	Function
01	0.3	Resistance
10	0.1	Diode
11	0.5	Capacitance/V/A

17.4. Example Program

To be provided.

17.5. Analog Switch Network



Analog Switch Network provides auto range switch for voltage/resistor/capacitor/diode measurement switching circuit. Nodes signal, including FB, SENSE, RLU, RLD are generated after switching and output to ADC or Comparator Network. Measurement mode is primarily selected by SMODE[7:0].

17.5.1. Register Description- Analog Switch Network

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1											
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition											
Address	**File Name**	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**	**A-RESET**	**i-RESET**
57H	PAX6					PS6	DS6	FS6	SS6		
58H	PA54	PS5	DS5	FS5	SS5	PS4	DS4	FS4	SS4		
59H	PA32	PS3	DS3	FS3	SS3	PS2	DS2	FS2	SS2		
5AH	PA10	PS1	DS1	FS1	SS1	PS0	DS0	FS0	SS0		
5DH	ADCN1	SDIO	SREFO	SFT1<1:0>		SFUVR<3:0>					
5EH	ADCN2	SMODE<7:0>									

PAX6:

PS6：PA<6> Power select control bit

1：Connect

0：Disconnect

DS6：PA<6> OP3 output select control bit

1：Connect

0：Disconnect

FS6：PA<6> Feedback select control bit

1：Connect

0：Disconnect

SS6：PA<6> Sense end select control bit

1：Connect

0：Disconnect

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1											
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition											
Address	**File Name**	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**	**A-RESET**	**i-RESET**
57H	PAX6					PS6	DS6	FS6	SS6		
58H	PA54	PS5	DS5	FS5	SS5	PS4	DS4	FS4	SS4		
59H	PA32	PS3	DS3	FS3	SS3	PS2	DS2	FS2	SS2		
5AH	PA10	PS1	DS1	FS1	SS1	PS0	DS0	FS0	SS0		
5DH	ADCN1	SDIO	SREFO	SFT1<1:0>		SFUVR<3:0>					
5EH	ADCN2	SMODE<7:0>									

PA54:

PS5：PA<5> Power select control bit

1：Connect

0：Disconnect

DS5：PA<5> OP3 output select control bit

1：Connect

0：Disconnect

FS5：PA<5> Feedback select control bit

1：Connect

0：Disconnect

SS5：PA<5> Sense select control bit

1：Connect

0：Disconnect

PS4：PA<4> Power select control bit

1：Connect

0：Disconnect

DS4：PA<4> OP3 select control bit

1：Connect

0：Disconnect

FS4：PA<4> Feedback select control bit

1：Connect

0：Disconnect

SS4：PA<4> Sense end select control bit

1：Connect

0：Disconnect

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1											
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition											
Address	**File Name**	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**	**A-RESET**	**i-RESET**
57H	PAX6					PS6	DS6	FS6	SS6		
58H	PA54	PS5	DS5	FS5	SS5	PS4	DS4	FS4	SS4		
59H	PA32	PS3	DS3	FS3	SS3	PS2	DS2	FS2	SS2		
5AH	PA10	PS1	DS1	FS1	SS1	PS0	DS0	FS0	SS0		
5DH	ADCN1	SDIO	SREFO	SFT1<1:0>		SFUVR<3:0>					
5EH	ADCN2	SMODE<7:0>									

PA32:

PS3：PA<3> power select control bit

1：Connect

0：Disconnect

DS3：PA<3> OP3 output select control bit

1：Connect

0：Disconnect

FS3：PA<3> Feedback select control bit

1：Connect

0：Disconnect

SS3：PA<3>Sense end select control bit

1：Connect

0：Disconnect

PS2：PA<2> Power select control bit

1：Connect

0：Disconnect

DS2：PA<2> OP3 output select control bit

1：Connect

0：Disconnect

FS2：PA<2> Feedback select control bit

1：Connect

0：Disconnect

SS2：PA<2>Sense select control bit

1：Connect

0：Disconnect

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1											
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition											
Address	**File Name**	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**	**A-RESET**	**i-RESET**
57H	PAX6					PS6	DS6	FS6	SS6		
58H	PA54	PS5	DS5	FS5	SS5	PS4	DS4	FS4	SS4		
59H	PA32	PS3	DS3	FS3	SS3	PS2	DS2	FS2	SS2		
5AH	PA10	PS1	DS1	FS1	SS1	PS0	DS0	FS0	SS0		
5DH	ADCN1	SDIO	SREFO	SFT1<1:0>		SFUVR<3:0>					
5EH	ADCN2	SMODE<7:0>									

PA10:

PS1：PA<1> Power select control bit

1：Connect

0：Disconnect

DS1：PA<1> OP3 output select control bit

1：Connect

0：Disconnect

FS1：PA<1> Feedback select control bit

1：Connect

0：Disconnect

SS1：PA<1> Sense end select control bit

1：Connect

0：Disconnect

PS0：PA<0> Power select control bit

1：Connect

0：Disconnect

DS0：PA<0> OP3 output select control bit

1：Connect

0：Disconnect

FS0：PA<0> Feedback select control bit

1：Connect

0：Disconnect

SS0：PA<0> Sense end select control bit

1：Connect

0：Disconnect

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1											
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition											
Address	**File Name**	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**	**A-RESET**	**i-RESET**
57H	PAX6					PS6	DS6	FS6	SS6		
58H	PA54	PS5	DS5	FS5	SS5	PS4	DS4	FS4	SS4		
59H	PA32	PS3	DS3	FS3	SS3	PS2	DS2	FS2	SS2		
5AH	PA10	PS1	DS1	FS1	SS1	PS0	DS0	FS0	SS0		
5DH	ADCN1	SDIO	SREFO	SFT1<1:0>		SFUVR<3:0>					
5EH	ADCN2	SMODE<7:0>									

ADCN1:

SDIO：Short control bit of PB<0> and PB<2>

0：Open

1：Short

SREFO：Register bit, can select input source of REFO Buffer

0：Select internal Band-gap Voltage Reference

1：Select PB<4> pin

SFT1<1:0>：Pre-filter path

00：100KΩ。

01：10KΩ。

10：0Ω；Short。

11：Close；Without Pre-filter path

SFUVR<3:0>：Select voltage reference source (as below table)

SFUVR <3:0>	SMODE<3>	CPO	VREF
0	0	0	VDS<17>
0	0	1	VDS<16>
0	1	0	VDS<15>
0	1	1	AGNDP<9>
1	0	0	AGNDP<8>
1	0	1	AGNDP<7>
1	1	0	AGNDP<6>
1	1	1	AGNDP<0>
2	0	0	VDS<1>
2	0	1	VDS<2>
2	1	0	VDS<3>
2	1	1	AGNDN<9>
3	0	0	AGNDN<8>
3	0	1	AGNDN<7>
3	1	0	AGNDN<6>
3	1	1	PB<5>
4	0	0	VDS<17>
4	0	1	VDS<1>
5	0	0	VDS<16>
5	0	1	VDS<2>
6	0	0	VDS<15>
6	0	1	VDS<3>
7	0	0	AGNDP<9>
7	0	1	AGNDN<9>
8	0	0	AGNDP<8>
8	0	1	AGNDN<8>
9	0	0	AGNDP<7>
9	0	1	AGNDN<7>
10	0	0	AGNDP<6>
10	0	1	AGNDN<6>
11	0	0	AGNDP<0>
11	0	1	PB<5>

ADCN2:

SMODE<7:0>	Measurement Mode Description
00000000	Voltage range divider, using external switch to ground
00100000	Voltage range divider, using internal switch to ground
00010100	Constant voltage resistor measurement or diode measurement
00010101	Negative constant voltage resistor measurement or diode measurement
00000110	Constant current resistor measurement or diode measurement
00000111	Negative constant current resistor measurement or diode measurement
00010101	Constant voltage resistor reference measurement
10100000	Diode measurement, VDDA driving voltage. 10M/1.111M resistor divider
0001101x	Capacitor measurement, VDDA/VSS charge/discharge is determined by CMPO
0001110x	Capacitor measurement, constant voltage charge/discharge is determined by CMPO
0000111x	Capacitor measurement, constant current charge/discharge is determined by CMPO

Select PS6~PS0, DS6~DS0, FS6~FS0 and SS6~SS0 can determine voltage range divider resistor and resistor reference of constant voltage/current. Voltage reference of constant voltage/current is decided by SFUVR, SMODE[3] and CMPO(Comparator output). Relative switch impedance is listed in below:

@VDDA=3.6V

Switch Name	PS6 ~ PS2 DS6 ~ DS2	PS1 ~ PS0	DS1 ~ DS0	FS6 ~ FS0	SS6 ~ SS0
Switch Impedance(Ω)	80	16	40	400	400

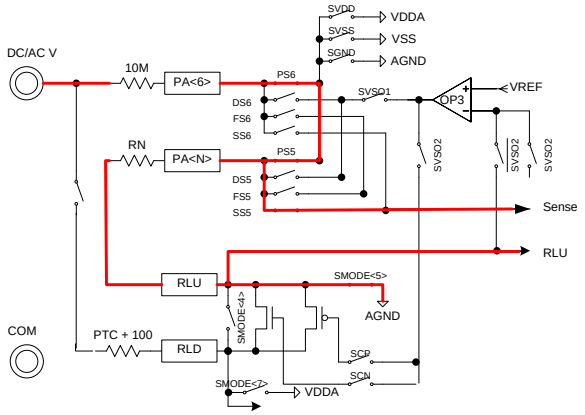
Register bit, SDIO, can control whether to connect PB<2> and PB<0> pin.

17.5.2. Multi-function Network Description

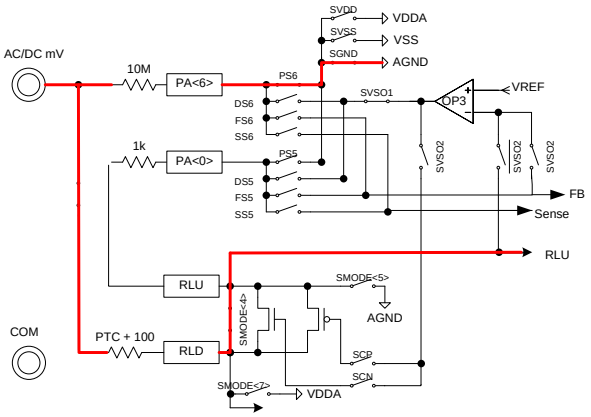
Please refer to 17.5.1 SMODE<7:0> configuration, the corresponding value of SMODE is as follows:

SMODE <3:0>	CMPO	SGND	SVDD	SVSS	SVSO1	SVSO2	SCP	SCN
0 0 0 0	X	0	0	0	0	0	0	0
0 0 0 1	X	1	0	0	0	0	0	0
0 0 1 0	X	0	1	0	0	0	0	0
0 0 1 1	X	0	0	1	0	0	0	0
0 1 0 0	X	0	0	0	1	0	0	0
0 1 0 1	X	1	0	0	1	0	0	0
0 1 1 0	X	0	1	0	0	1	1	0
0 1 1 1	X	0	0	1	0	1	0	1
1 0 0 X	0	0	0	0	0	0	0	0
1 0 0 X	1	1	0	0	0	0	0	0
1 0 1 X	0	0	1	0	0	0	0	0
1 0 1 X	1	0	0	1	0	0	0	0
1 1 0 X	0	0	0	0	1	0	0	0
1 1 0 X	1	1	0	0	1	0	0	0
1 1 1 X	0	0	1	0	0	1	1	0
1 1 1 X	1	0	0	1	0	1	0	1

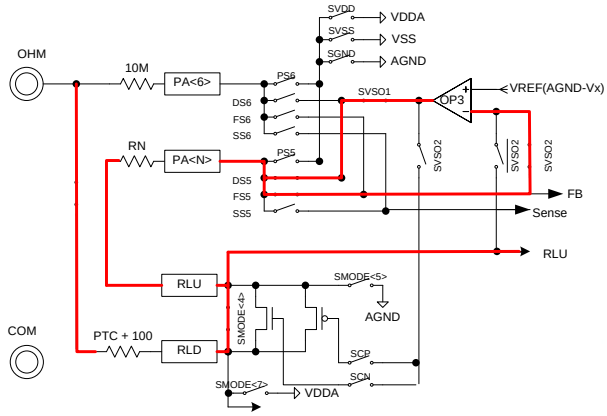
Multi-function Network can develop related network:



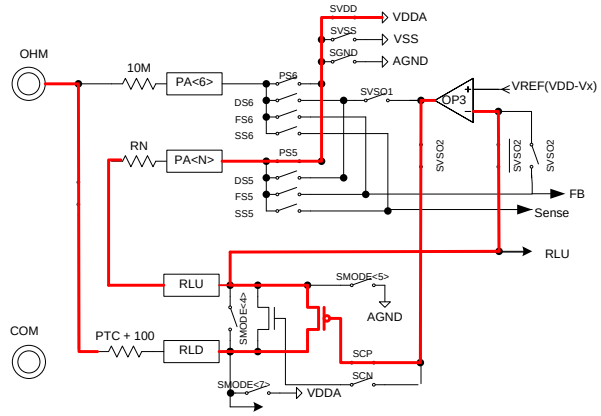
DC/AC V Voltage Divider



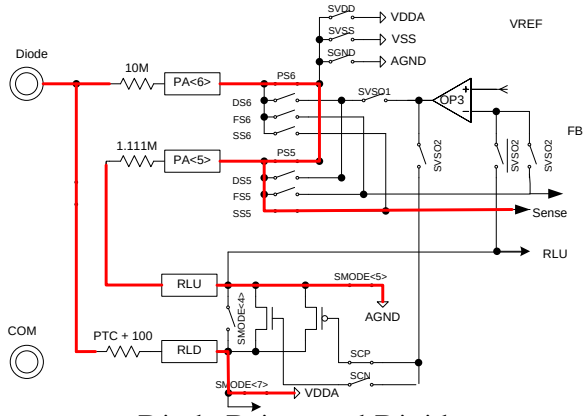
mV with 10 M Input Impendence



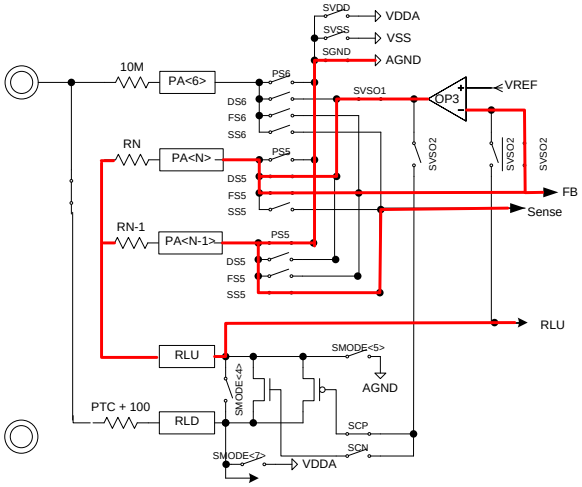
Constant Voltage Source



Constant Current Source

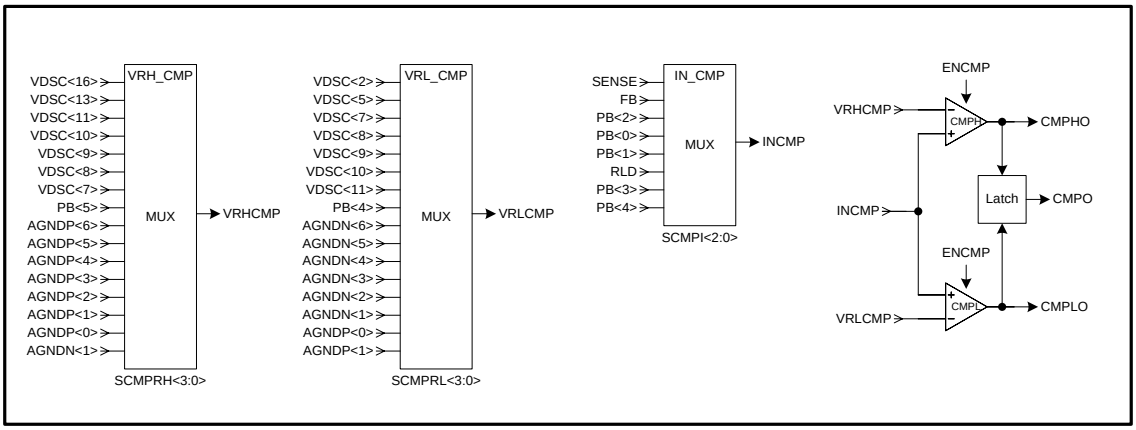


Diode Driver and Divider



Reference Resistor Measure

17.6. DMM Comparator Network



DMM Comparator Network constitutes two analog comparators and input multiplexers. Hysteresis window comparator is composed of CMPH and CMPL comparator. VRHCMP, the positive input end of CMPH and VRLCMP, the negative input end of CMPL respectively equip with high/low potential comparison of hysteresis window comparator that can select by input multiplexer. CMPO is the output of window comparator while CMPHO and CMPLO are output of CMPH and CMPL comparator respectively.

Comparator Network is mainly used to measure frequency, test short circuit and measure capacitor range.

17.6.1. Register Description- DMM Comparator Network

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1											
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition											
Address	**File Name**	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**	**A-RESET**	**i-RESET**
5FH	ADCN3	SCMPRH<3:0>				SCMPRL<3:0>					
60H	ADCN4	SCMPI<2:0>			AD1CHOP<1:0>		AD1OSR<2:0>				
6EH	CTSTA	CNTI	CMPO	CMPHO	CMPLO				CTBOV		

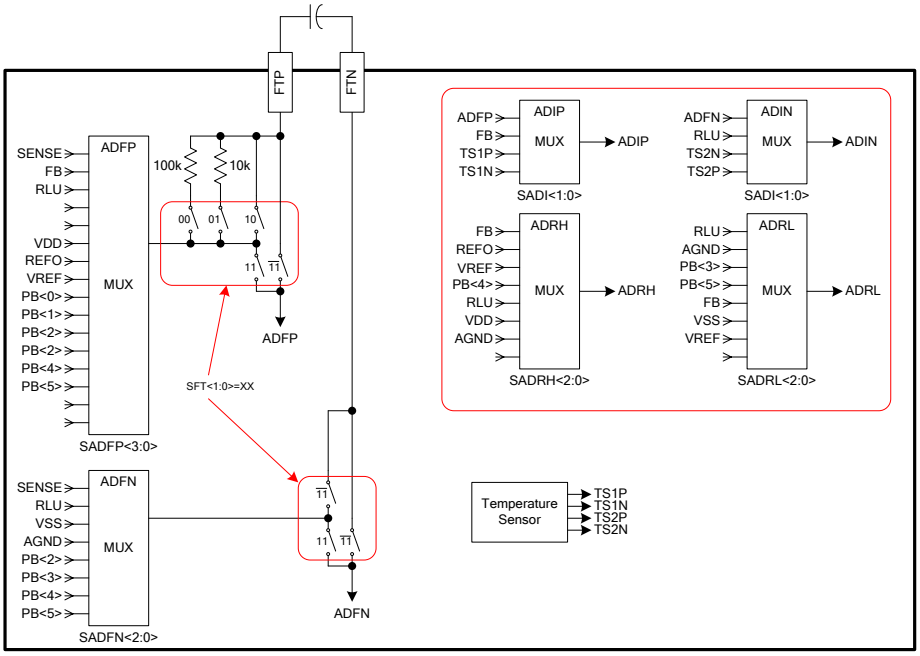
(1) Comparator output, CMPHO, CMPLO and CMPO can be read by CTSTA register, it will output to frequency counter simultaneously.

(2) Comparator input is connected by MUX and is controlled by register bit, SCMPRH<3:0>, SCMPRL<3:0> and SCMPI<2:0>.

SCMPRH<3:0>	0000	0001	0010	0011	0100	0101	0110	0111
VRHCMP	VDSC<16>	VDSC<13>	VDSC<11>	VDSC<10>	VDSC<9>	VDSC<8>	VDSC<7>	PB<5>
SCMPRH<3:0>	1000	1001	1010	1011	1100	1101	1110	1111
VRHCMP	AGNDP<6>	AGNDP<5>	AGNDP<4>	AGNDP<3>	AGNDP<2>	AGNDP<1>	AGNDP<0>	AGNDN<1>
SCMPRL<3:0>	0000	0001	0010	0011	0100	0101	0110	0111
VRLCMP	VDSC<2>	VDSC<5>	VDSC<7>	VDSC<8>	VDSC<9>	VDSC<10>	VDSC<11>	PB<4>
SCMPRL<3:0>	1000	1001	1010	1011	1100	1101	1110	1111
VRLCMP	AGNDN<6>	AGNDN<5>	AGNDN<4>	AGNDN<3>	AGNDN<2>	AGNDN<1>	AGNDP<0>	AGNDP<1>
SCMPI<2:0>	000	001	010	011	100	101	110	111
INCMP	SENSE	FB	PB<2>	PB<0>	PB<1>	RLD	PB<3>	PB<4>

(3) ENCMP：Register bit, can enable CMPH and CMPL comparator. 1=Enable, 0=Disable.

17.7. Pre-Filter, ADC Input MUX and Temperature Sensor



Input signal and reference signal of ADC are connected by MUX. ADC preamplifier can bypass Pre-filter through selection. Additionally, the chip has a built-in Temperature Sensor that helps to measure chip temperature through ADC measurement.

17.7.1.1. Register Description-Pre-Filter, ADC Input MUX and Temperature Sensor

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1											
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition											
Address	**File Name**	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**	**A-RESET**	**i-RESET**
5DH	ADCN1			SFT<1:0>		SFUVR<3:0>					
61H	ADCN5	SADFP<3:0>					SADFN<2:0>				
62H	ADCN6	SADRH<2:0>			SADRL<2:0>			SADI<1:0>			

Pre-Filter:

(1) There is a resistor network in AD1 input preamplifier. A Filter will be formed by connecting an external filter capacitor in between FTP and FTN. This filter helps to filter noise and stabilize signal. The positive/negative input signal is connected by MUX while SADFP and SADFN are the positive/negative output signal of the filter respectively. They were controlled by register bit, SAD1FP<3:0>, SAD1FN<2:0> and SFT1<1:0>.

SAD1FP<3:0>	0000	0001	0010	0011	0100	0101	0110	0111
Filter Positive Input	SENSE	FB	RLU	X	X	VDD	REFO	VREF
SAD1FP<3:0>	1000	1001	1010	1011	1100	1101	1110	1111
Filter Positive Input	PB<0>	PB<1>	PB<2>	PB<2>	PB<4>	PB<5>	X	X
SAD1FN<2:0>	000	001	010	011	100	101	110	111
Filter Negative Input	SENSE	RLU	VSS	AGND	PB<2>	PB<3>	PB<4>	PB<5>

(2) SFT1<1:0>：Register bit, can select filter capacitor as 100K, 10K, 0 or None.

ADC Input MUX:

ADC input signal and reference signal are connected by MUX that controlled by register bit respectively.

(1) ADIP and ADIN：ADC positive/negative input signal that commonly controlled by register bit, SAD1I<1:0>.

(2) SADRH and SADRL：ADC positive/negative signal reference that commonly controlled by register bit SAD1RH<2:0> and SAD1RL<2:0>.

SAD1I<1:0>	00	01	10	11				
AD1IP	ADFP	FB	TS1P	TS1N				
AD1IN	ADFN	RLU	TS2N	TS2P				
SAD1RH<2:0>	000	001	010	011	100	101	110	111
SADFRP	FB	REFO	VREF	PB<4>	RLU	VDD	AGND	X
SAD1RL<2:0>	000	001	010	011	100	101	110	111
SADFRN	RLU	AGND	PB<3>	PB<5>	FB	VSS	VREF	X

Temperature Sensor:

There is a built-in temperature sensor of the IC that has two sets of voltage (TS1P,TS2N) and (TS1N,TS2P). These two voltage sets can be measured by AD1. Temperature calculation is as follows: (Set ADC Input buffer off, ADCN7[AD1IPNUF]= ADCN7[AD1IPNUF]=0b)

(1) Set SAD1I<1:0>=10, ADC measured a digital code, TCode1.

(2) Set SAD1I<1:0>=11, ADC measured a digital code, TCode2.

(3) Calculate TCode=(TCode2 - TCode1)/2, this action can erase Temperature Sensor Offset.

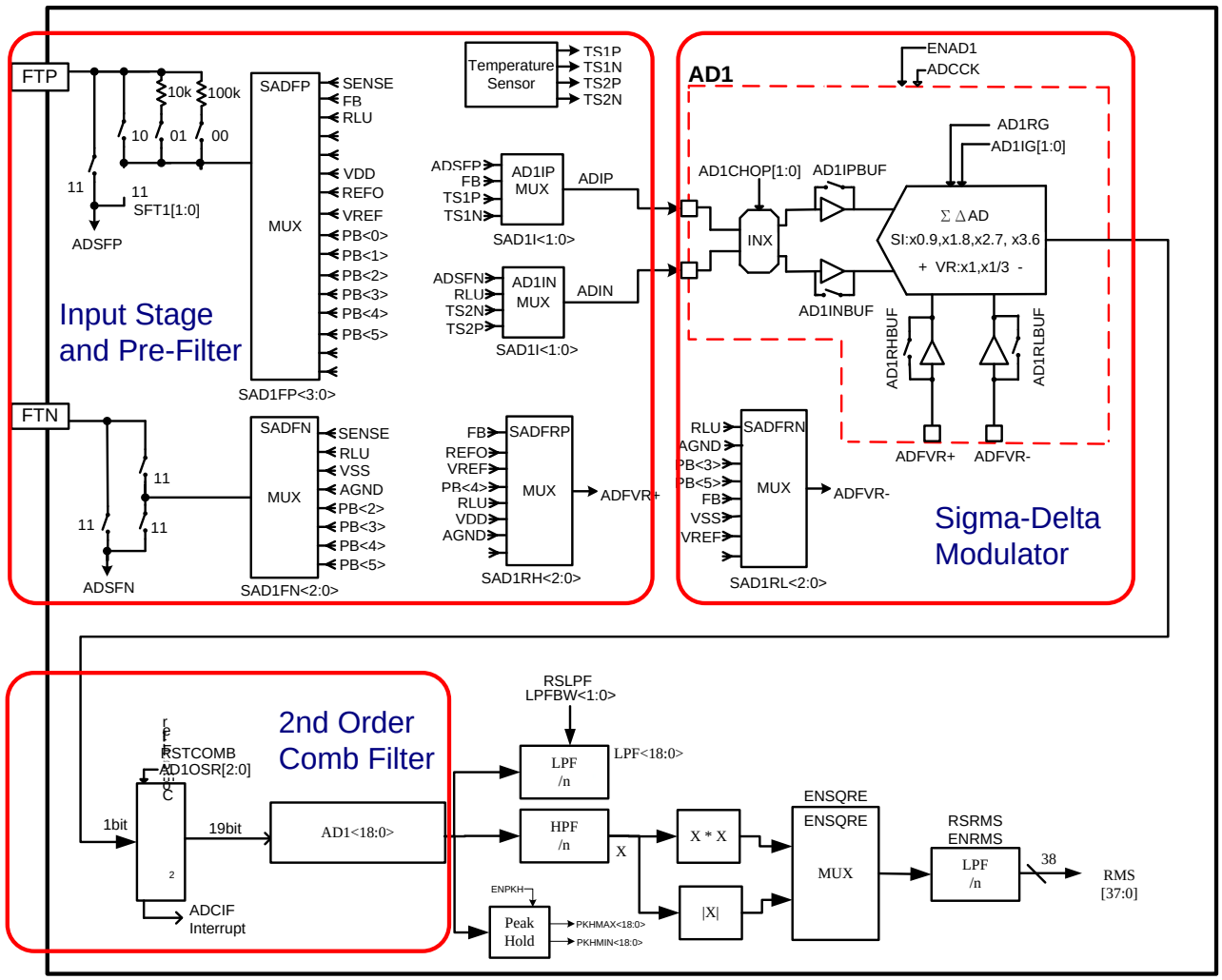
(4) Supposed that one point was calculated at 25℃, and then TCode@25℃ can be obtained. Due to the fact that there is a level shift, an offset will be added. Temperature curve slope, G can be gained as follows:

$$G = \frac{TCode@25^{o}C}{25 + 273.15 + T_{OS}}$$, T_{OS} is offset, about $8^{O}K$.

(5) Supposed that the temperature-to-be-measured is T_x℃, then we can gained:

$$T_X = \frac{TCode@T_X{}^{o}C}{G} - [273.15 + T_{OS}] \quad {}^{o}C$$

18. ΣΔADC, Low Pass Filter, RMS Converter and Peak Hold



18.1. Register Data Synchronization

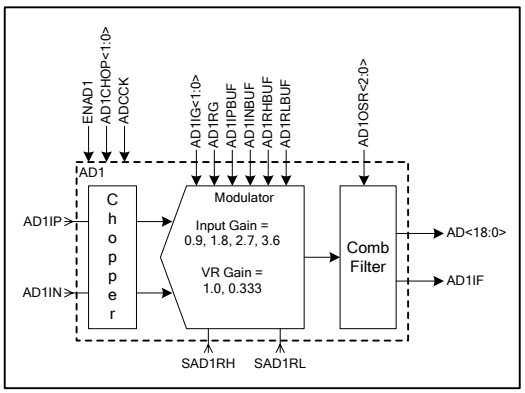
"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1											
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition											
Address	**File Name**	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**	**A-RESET**	**i-RESET**
6FH	PKHMAXU	PKHMAX<18:11>									
70H	PKHMAXH	PKHMAX<10:3>									
71H	PKHMAXL	PKHMAX<2:0>									
72H	PKHMINU	PKHMIN<18:11>									
73H	PKHMINH	PKHMIN<10:3>									
74H	PKHMINL	PKHMIN<2:0>									
75H	RMSDATA4	RMS<37:30>									
76H	RMSDATA3	RMS<29:22>									
77H	RMSDATA2	RMS<21:14>									
78H	RMSDATA1	RMS<13:6>									
79H	RMSDATA0	RMS<5:0>									
7AH	LPFDATAU	LPF<18:11>									
7BH	LPFDATAH	LPF<10:3>									
7CH	LPFDATAL	LPF<2:0>									
7DH	AD1DATAU	AD1<18:11>									
7EH	AD1DATAH	AD1<10:3>									
7FH	AD1DATAL	AD1<2:0>									

When reading PKHMAX, PKHMIN, RMSDATA, LPFDATA, AD1DATA register, 2byte and above data was read, a Latch is needed to ensure that no other record was updated to the register during continuous reading process. This Latch mechanism is when low byte was read, high byte will be updated as the same record. Thus, reading should start from the register that has bigger address.

Ex: When reading AD1, 7FH address must be read first. 7EH and 7DH data will be fixed when reading 7FH to ensure that user can read the same AD value.

After AD1DATAL was read, the data read from AD1DATAH will be wrong if LPFDATAL was read first, then to read AD1DATAH.

18.2. ΣΔADC



ADC of HY12P65 comprises four parts, Input/VR Buffer and Chopper Control, Gain Stage, Modulator and three-stage Comb Filter. ADIP and ADIN are positive/negative input signal while SAD1RH and SAD1RL are positive/negative reference signal.

18.2.1. Register Description-ΣΔADC

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1											
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition											
Address	**File Name**	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**	**A-RESET**	**i-RESET**
5CH	PWRCN2	MCUBIAS	ENCPVGG	ENCMP	ENCNTI	ENCRT	RSTCOMB	RSLPF	RSRMS		
60H	ADCN4				AD1CHOP<1:0>		AD1OSR<2:0>				
61H	ADCN5	SAD1FP<3:0>				HSAD	SAD1FN<2:0>				
62H	ADCN6	SAD1RH<2:0>			SAD1RL<2:0>			SAD1I<1:0>			
63H	ADCN7	ENAD1	AD1IG<1:0>		AD1RG	AD1RHBUF	AD1RLBUF	AD1IPBUF	AD1INBUF		

(1) ENAD1：Register bit which enables ADC. 1=Enable; 0=Disable and clear AD<18:0> as 0.

(2) AD1CHOP<1:0>：Register bit that configure the form of Chop ADC input signal. The result will be reflected at ADC output, AD<18:0>.

Supposed VOS is the offset output code when AD1CHOP=00, VX is the output code that deducted offset. When using different AD1CHOP configurations, ADC output code is listed in below table. When AD1CHOP=1x, ADC switches input signal in accordance with OSR time configuration to achieve Offset deduction function. This way, ADC Offset Voltage can be lowered than 10uV, and ADC output rate will be one time slower.

ADCHOP<1:0>	00	01	10	11
AD<18:0>	VX+VOS	VX-VOS	VX	VX

(3) AD1IG<1:0>：Register bit, can configure gain of ADC input signal.

(4) AD1RG：Register bit, can configure gain of ADC reference signal.

ADIG<1:0>	00	01	10	11	ADRG	0	1
AD Input Gain	0.9	1.8	2.7	3.6	AD Reference Gain	1.0	0.333

(6) AD1IPBUF：Register bit, can configure whether to let ADC positive input signal passes through Buffer. 1=Enable; 0=Disable.

(7) AD1INBUF：Register bit, can configure whether to let ADC negative input signal passes through the Buffer. 1=Enable; 0=Disable.

(8) AD1RHBUF：Register bit, can configure whether to let ADC positive reference signal passes through Buffer. 1=Enable; 0=Disable.

(9) AD1RLBUF：Register bit, can configure whether to let ADC negative reference signal passes through Buffer. 1=Enable; 0=Disable.

(10) The frequency of Modulator1 sampling signal of ADC is 200KHz(ADCCK=1) or 400KHz(ADCCK=0) after frequency divider.

AD1Cdata output rate=F_{ADCLK}/OSR1, F_{ADCLK} is the frequency of ADCCK. When OSR1=0xx, ADC is under fast output mode. HSAD=1 and AD1CHOP=0x must be configured. In different OSR configurations, the maximum output value of ADC output code (Comb Filter Gain Factor) will be different as follows:

AD1OSR<2:0>	000	001	010	011	100	101	110	111
OSR1	32	64	128	256	2500	5000	10000	20000
+Max	3FD7CH	3FFFFH	3FFFFH	3FFFFH	17D79H	17D79H	17D79H	17D79H
-Min	40288H	40000H	40000H	40000H	68288H	68288H	68288H	68288H

(12) AD1DATA<18:0>：ADC data output register, total 19 bits.

(13) AD1IF：Flag for ADC event occurrence; this signal will be sent to INTF register.

(14) RSTCOMB：Reset Comb Filter.

<0>Clear; Write "0" must be written back to "1".

<1>Normal.

18.3. Suggested Configurations for DMM Application

18.3.1. INPUT STAGE

- Set ADC input and Reference base on different range measurement.
- DC passes Pre-filter, AC and Peak hold bypass Pre-filter.

18.3.2. ADC Modulator

ADC Gain configurations are as follows:

Measurement Range	ADIG	ADRG	ADC Gain	VREF
DC/AC 500mV, DC/ACV,DC/AC 10A, 500mA, 5000uA, Diode	0.9	1	0.9	(REFO,AGND)
DC/AC 50mV , DC/AC 5A, 50mA, 500uA	3.6	0.33	10.8	(REFO(AGND)
500/5k/50k OHM	0.9	1	0.9	(RLU,FB)
500k/5M/50M OHM	0.9	0.33	2.7	(VDDA,VSS)

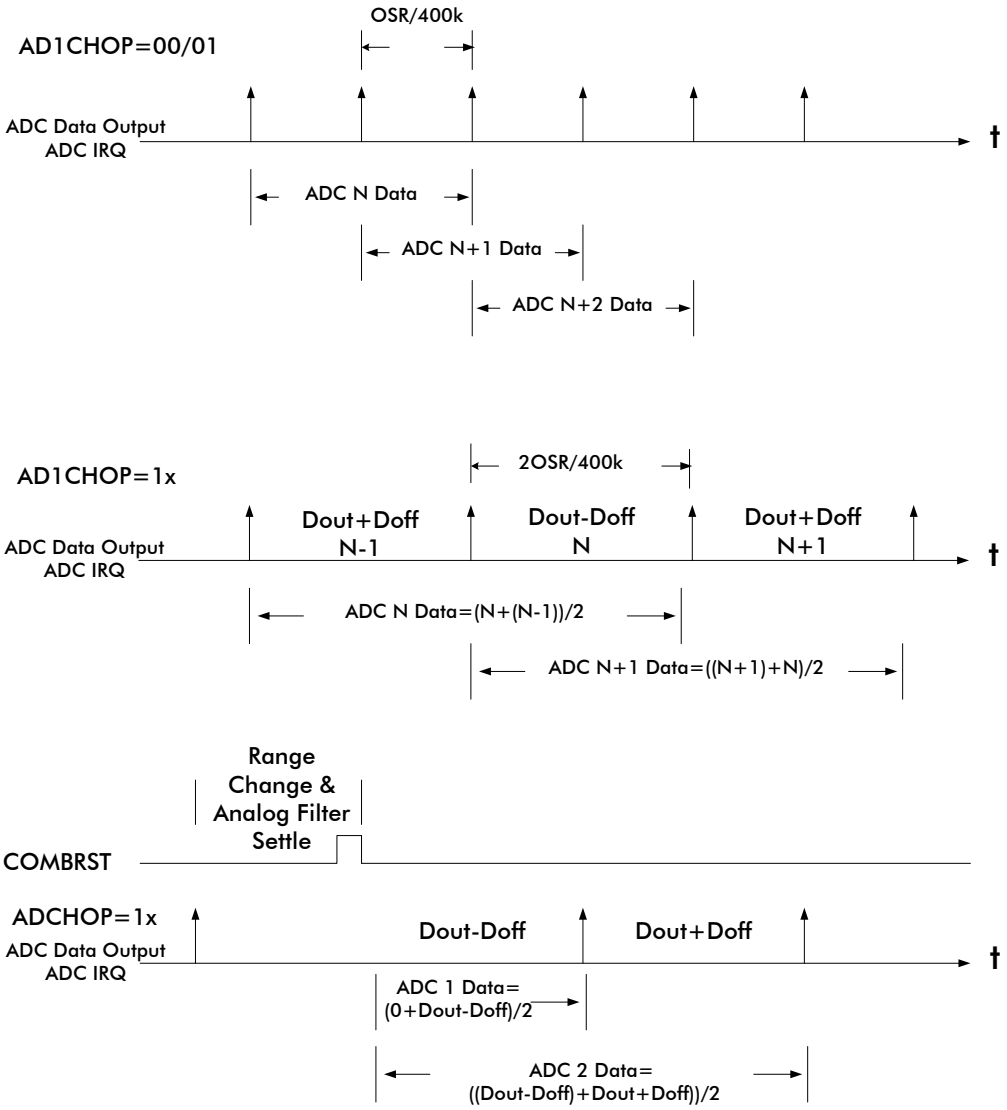
18.3.3. COMB Filter and Chopper

The influences of Comb filter and Chopper configurations on output rate.

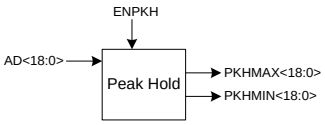
ADC Clock=400KHz			ADC Output Rate	
AD1OSR	OSR	Comb Filter	AD1Chop=0x	AD1hop=1x
111	20000	2nd	20Hz	10Hz
110	10000	2nd	40Hz	20Hz
101	5000	2nd	80Hz	40Hz
100	2500	2nd	160Hz	80Hz
AD1OSR=111, AD1CHop=1x is the suggested configuration of DC measurement				

The output rate of ADC is 400kHz/OSR. However, ADC sampled 2*OSR of every data and when AD1Chop started, it will switch to ADC input end and the output rate will decrease a half time. Input Offset is smaller than 10uV. When switching ranges, users need to wait until the analog filter was stabilized. Additionally, RSTCOMB bit can be used to reset comb filter as to gain accurate output by waiting for the second ADC data.

The sequence is as follows:


AD1CHOP=00/01
OSR/400k
ADC Data Output
ADC IRQ
t
ADC N Data
ADC N+1 Data
ADC N+2 Data
AD1CHOP=1x
2OSR/400k
Dout+Doff
N-1
Dout-Doff
N
Dout+Doff
N+1
ADC Data Output
ADC IRQ
t
ADC N Data=(N+(N-1))/2
ADC N+1 Data=((N+1)+N)/2
Range
Change &
Analog Filter
Settle
COMBRST
ADCHOP=1x
ADC Data Output
ADC IRQ
Dout-Doff
Dout+Doff
t
ADC 1 Data=
(0+Dout-Doff)/2
ADC 2 Data=
((Dout-Doff)+Dout+Doff))/2

18.4. Peak Hold



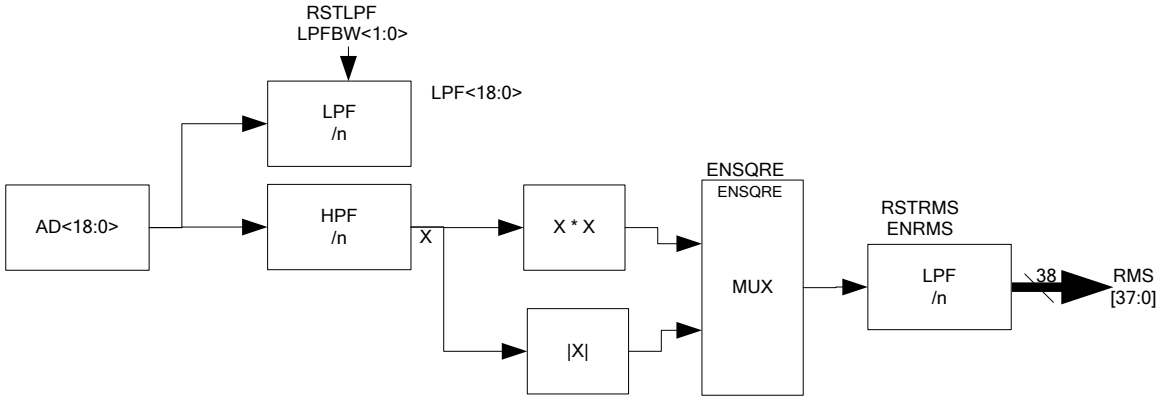
Peak Hold can store the Maximum /minimum ADC output value to PKHMAX and PHHMIN registers.

18.4.1. Register Description-Peak Hold

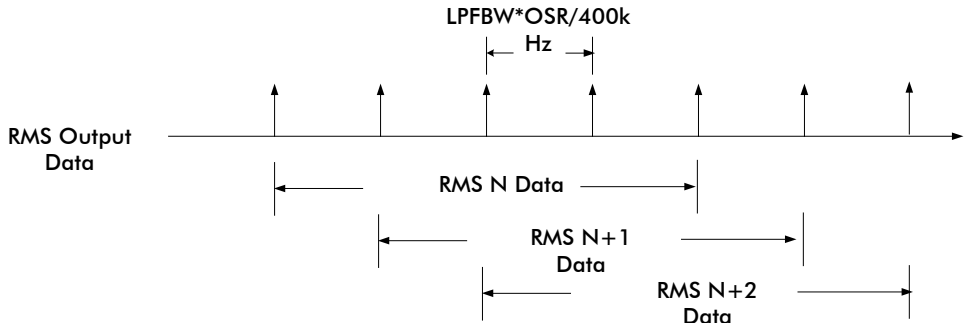
"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1											
"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition											
Address	**File Name**	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**	**A-RESET**	**i-RESET**
64H	RMSCN	ENRMS	ENLPF	ENSQRE	LPFBW<1:0>		ENPKH				

When ENPKH=0, PKHMAX<18:0>=40000h, PKHMIN<18:0>=3FFFFh. When ENPKH=1, compared ADC output to PKHMAX and PHHMIN respectively. If the result is bigger than PKHMAX, then PKHMAX=AD. If the result is smaller than PKHMIN, then PKHMIN=AD, or please remain the original value unchanged.

18.5. Low Pass Filter & RMS Converter



AC measurement of HY12P65 used internal digital signal process unit to calculate its AC value when ADC is under fast output mode. Configure ENSQRE separately to calculate its true RMS value or the mean of absolute value. Before calculating its AC value, it will pass through HPF (High Pass Filter) to remove DC. If its DC signal is required, users can read LPF<18:0>. AC signal after square or absolute value, will pass through $Sinc^4$ Low Pass Filter to gain RMS<37:0> output. For true RMS value measurement, MCU is required to implement radical. The output time sequence is as follows:



18.5.1. Register Description- Low Pass Filter & RMS Converter

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1											
."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition											
Address	**File Name**	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**	**A-RESET**	**i-RESET**
5CH	PWRCN2	MCUBIAS	ENCPVGG	ENCMP	ENCNTI	ENCTR	RSTCOMB	RSLPF	RSRMS		
64H	RMSCN	ENRMS	ENLPF	ENSQRE	LPFBW<1:0>		ENPKH				

(1) LPF is when AD1 OSR1 is 32 ~ 128 and its AC calculation Low Pass Filter, which is $Sinc^4$ Filter.

(2) ENLPF：Register bit, can enable Low Pass Filter. 1=Enable；0=Disable, and clear LPF<18:0> to 0.

(3) LPFBW<2:0>：Register bit, can configure Over Sampling Ratio (OSR4) of Low Pass Filter.

Low Pass Filter data output rate=data input rate/OSR4.

LPFBW<2:0>	00	01	10	11
OSR4	256	512	1024	2048

(4) LPF<18:0>：Output data register of AD1 Low Pass Filter.

(5) LPFIF：Flag bit of Low Pass Filter event; this signal will be sent to INTF register.

(6) RSLPF：Reset Low Pass Filter.

<0>Clear; Write "0" must be written back to "1".

<1>Normal.

RMSCN Register:

(1) ENRMS：Register bit, can enable RMS Converter. 1=Enable；0=Disable and clear RMS<37:0> to 0.

(2) RMS<37:0>：Output data register of RMS Converter.

RMS data output rate=Low Pass Filter data output rate.

(3) Supposed that X=AD1<18:0> passes through High Pass Filter, N is the OSR of Low Pass Filter, which is configured by LPFBW<2:0>.

Then, $RMS<37:0>=\sum\frac{X^2}{N}$, when users would like to gain RMS value, radical must be implemented via external MCU software.

(4) RMSIF：Flag bit of RMS Converter event; this signal will be sent to INTF register.

(5) ENSQRE: "1" will display $RMS<37:0>=\sum\frac{X^2}{N}$.

"0" will display $RMS<37:0>=\sum\frac{|X|}{N}$

(6) RSRMS：Reset RMS Low Pass Filter。

<0>Clear; Write "0" must be written back to "1".

<1>Normal.

19. DMM Range Application Example

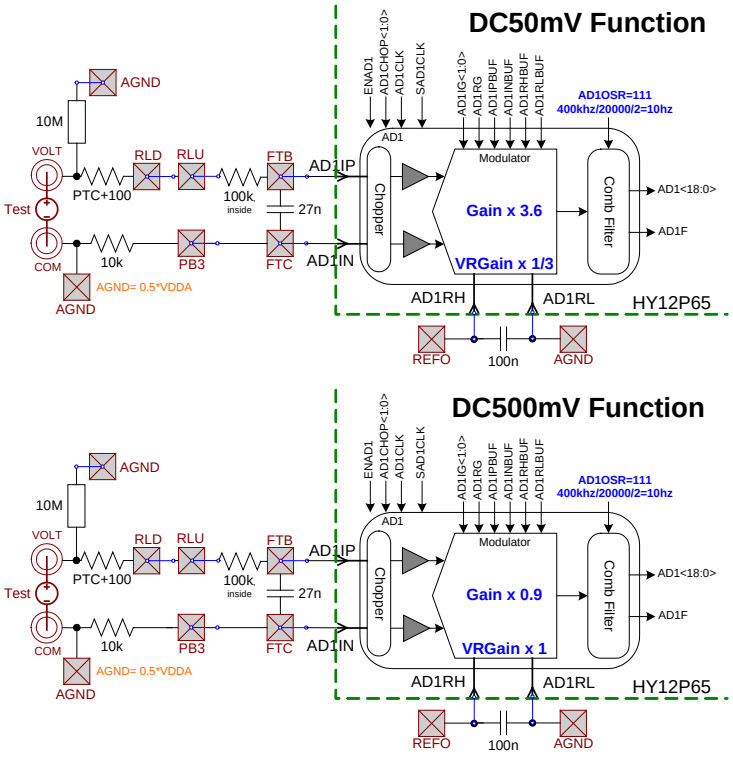
19.1. DC mV

19.1.1. Register Configuration

DC 500mV:

DC 50mV:

19.1.2. Example



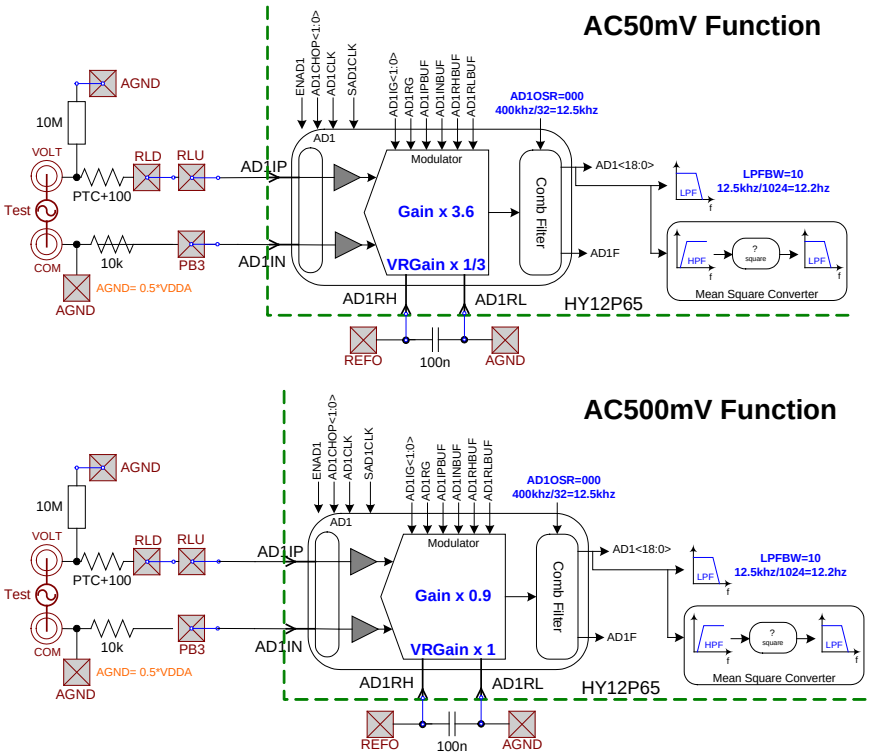
19.2. AC mV

19.2.1. Register Configuration

AC 500mV:

AC 50mV:

19.2.2. Example



19.3. DCV

19.3.1. Register Configuration

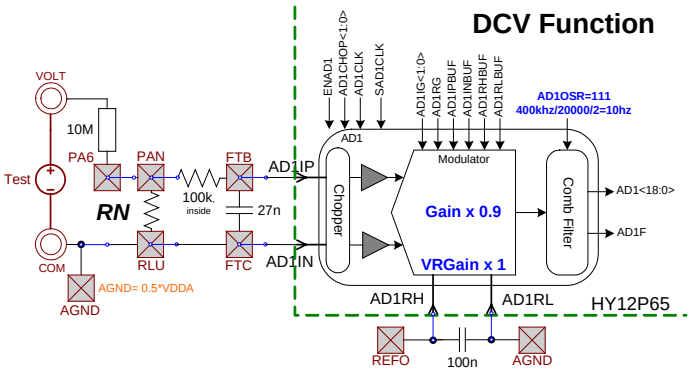
5V:

50V:

500V:

1KV:

19.3.2. Example



19.4. ACV

19.4.1. Register Configuration

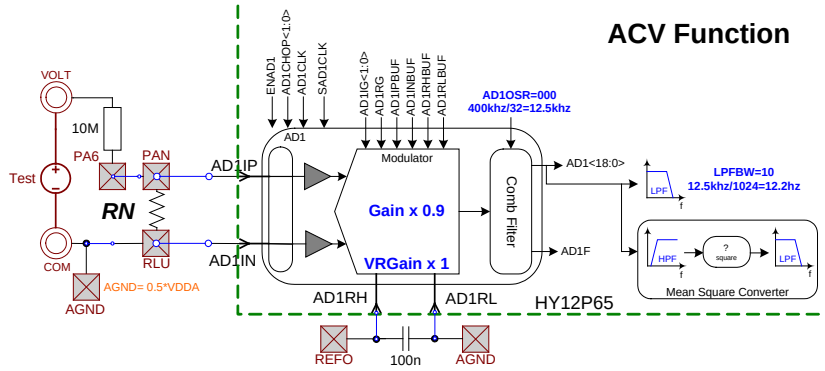
5V:

50V:

500V:

1KV:

19.4.2. Example



19.5. 500~50Kohm

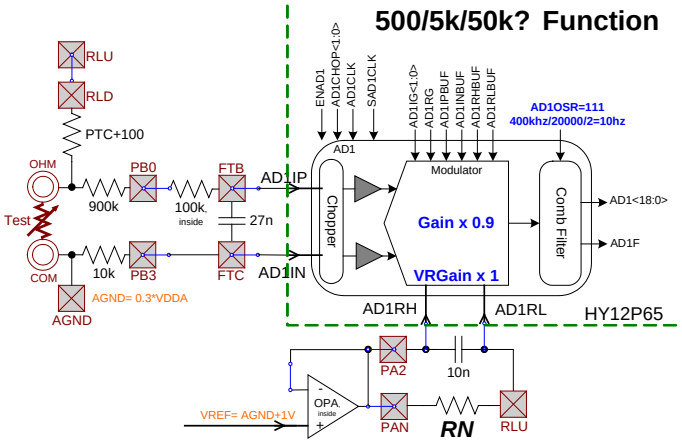
19.5.1. Register Configuration

500ohm:

5Kohm:

50Kohm:

19.5.2. Example



19.6. 500K~50Mohm

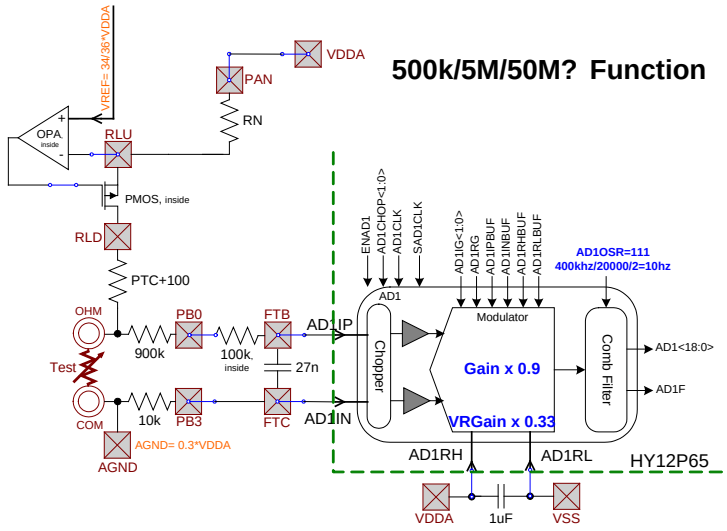
19.6.1. Register Configuration

500K ohm:

5M ohm:

50M ohm:

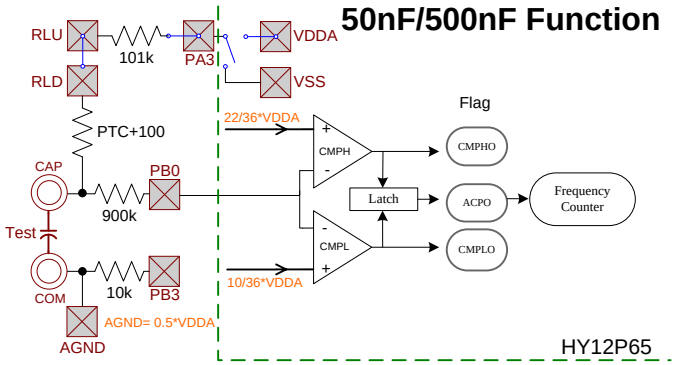
19.6.2. Example



19.7. 5nF~500nF

19.7.1. Register Configuration

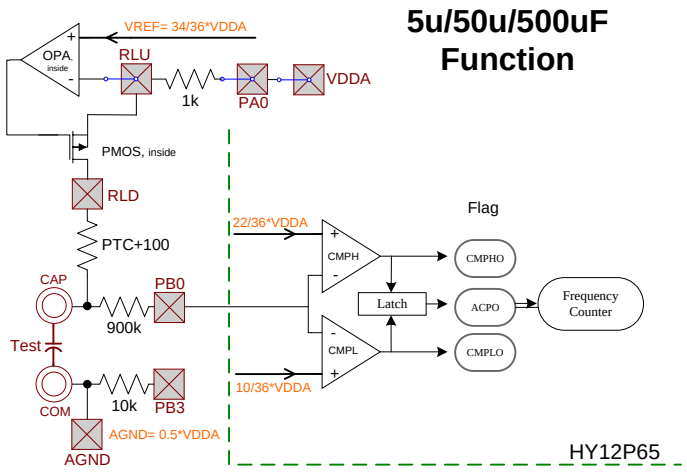
19.7.2. Example



19.8. 5uF~500uF

19.8.1. Register Configuration

19.8.2. Example



20. Build-In EPROM

Build-In EPROM(BIE) applies HY12P62 Series, use BIE function to store product No., password, and the data that after program calculation…etc. External hardware only need to connect VBIE with 6V voltage in VPP/RST pin then use this function. The stored address range is 00H~3FH total 64 words equal to 128 bytes.

When use external VBIE power supply (6V) to program BIE area, able to program one byte (word) one time data in BIE area through order.。

※**Attention**：HY12S65 development kit unable to simulate BIE programming function.

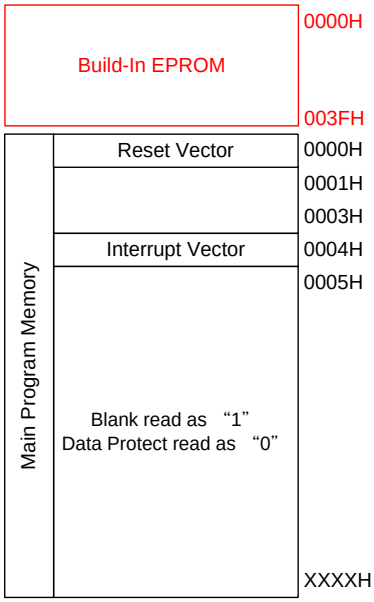


Figure 20-35 Build-In EPROM Architecture

BIE Register summary：

BIECN	VPPHV[0], BIEWR[0], BIERD[0]
BIEARH	ENBIE[0]
BIEARL	BIE_ADDR[5:0]
BIEDRH	BIE_DATA[15:8]
BIEDRL	BIE_DATA[7:0]

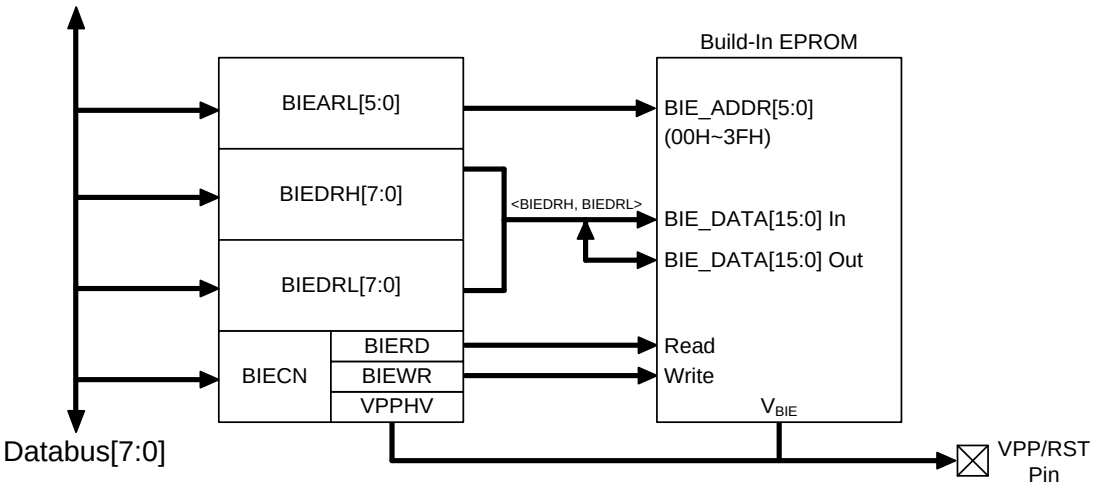


Figure 20-2 BIE diagram

20.1. BIE instruction

20.1.1. BIE read function

- When user read EPROM：
  - ENBIE[0] set <1> to able BIE function and VPP cannot be 0V。
  - When ENBIE[0] set <1>，have to set BIEARH[6:0] as <0000000>(default)，

if use command(Recommendation)：

BSF BIEARH,ENBIE,ACCE or

BCF BIEARH,ENBIE,ACCE

Then, would not influence value of BIEARH[6:0].

- Write in the specific address to BIEARL[7:0] address register，the valid write in length is BIEARL[5:0](up most is 3FH).
- BIERD[0] set <1> to return the specified address data 16Bit to BIEDRH [7: 0] and BIEDRL[7:0],high bits data return to BIEDRH[7:0]，low bits data return to BIEDRL[7:0]。
- Data after complete read, BIERD [0] is automatically set by the hardware as <0>, and BIEARL [7: 0] address register content is automatically incremented until 3Fh that is no longer incremented.

20.1.2. BIE write in (programing) function

- Pins connect with external high voltage, which is VPPBIE external 6V voltage programming.
  - Detect if connect with external 6V voltage and if the flag status is stable.
    - When the VPP pin connector 6V voltage VPPHV [0] is automatically set to <1>. VPPHV [0] for the immediate reaction VPP pin is connected to 6V voltage state flag bit, it can be used to detect 6V voltage is stable.
  - ENBIE[0] set <1> to enable BIE function.
  - When ENBIE[0] set <1>, need set BIEARH[6:0] as <0000000>(default)，if use orders：

    BSF BIEARH,ENBIE,ACCE or
    BCF BIEARH,ENBIE,ACCE

    Will not influence the value of BIEARH[6:0].
  - Write data to be stored here to BIEARL [7: 0], and the data are written to the highest burning bytes BIEDRH [7: 0] and the lower bytes BIEDRL [7: 0] register.
  - BIEWR [0] set <1> to write in data to the specified address.
  - Write in data after completion BIERD [0] is automatically set by the hardware as <0> and BIEARL [7: 0] address register content is automatically incremented until 3Fh that is no longer incremented.

- *BIE function is enabled and specify the location of more than 3FH, will BIEWR [0] <1> or BIERD [0] <1> will not have to program data movement*
- *BIE EPROM write operation will increase the IC instantaneous current consumption can affect ADC measurement accuracy.*
- *BIERD [0] and BIEWR [0] Do not set both <1> to avoid operating anomalies occur.*
- *BIE read operation has nothing to do with the VPP voltage, but not for the low voltage; when use the BIE to program, the power order is VDD first, then VPP*

(1)Read EPROM

```
        LBSR    001H                ;claim Bank address，BIE related register control is in Bank 1
        BCF     ADCN7,ENAD1,ACCE    ;turn off ADC function before read EPROM
        BSF     BIEDRH, ENBIE, BANK ;enable BIE
        MVL     00000000B
        MVF     BIEARL, F, BANK     ; to definite EPROM address as 00H
                                BSF     BIECN, BIERD, BANK      ;Oder to read EPROM，and
save the data in register BIEDRH, BIEDRL
                                ; After finishing reading EPROM，BIECN[BIERD]automatically
                                ; become 0, BIEARL(BIE_ADDR)automatically add 1, to 3FH upmost.
WAITRDBIE:
        BTSZ    BIECN, BIERD, BANK  ; Wait for judge EPROM is finished reading,
                                    ; BIECTRL[BIERD]automatically become 0
        JMP     WAITRDBIE
        MVF     BIEDRL, W, BANK
        MVF     BUF0, F, ACCE       ; BIEDRL data move to BUF0
        MVF     BIEDRH, W, BANK
        MVF     BUF1, F, ACCE       ; BIEDRH data move to BUF1
        ;BSF    ADCN7,ENAD1,ACCE    ; open the ADC function according to user requirement.
```

(2) Write in EPROM

```
        LBSR    001H                ; Claim Bank address，BIE related register control is in Bank 1
        BCF     ADCN7,ENAD1,ACCE    ; Turn off ADC function before write in EPROM
        BSF     BIEDRH, ENBIE, BANK ; Enable BIE
VPPCHK:
        BTSS    BIECN, BIEHV, BANK  ; Check external VPP=6V voltage exist or not, if exit, then
                                    ; continue next step.
        JMP     VPPCHK
        MVL     00000000B
        MVF     BIEARL, F, BANK     ; Definite EPROM address as 00H
        MVL     12H                 ; Definite write in data [BIEDRH, BIEDRL]=[12H,34H]
        MVF     BIEDRH, F, BANK
        MVL     34H
        MVF     BIEDRL, F, BANK
        BSF     BIECN, BIEWR, BANK  ; After finish writing EPROM，BIECN[BIEWR] automatically be 0.
                                    ; BIEARL(BIE_ADDR)automatically add 1, to 3FH upmost.
WAITWRBIE:
        BTSZ    BIECN, BIEWR, BANK  ;
```

Figure 20-3 H08A BIE Example program (applyHY12P62)

20.2. Register Description-BIE

"-"no use,"*"read/write,"w"write,"r"read,"r0"only read 0,"r1"only read 1,"w0"only write 0,"w1"omly write 1											
"$"for event status ,"."unimplemented bit,"x"unknown,"u"unchanged,"d"depends on condition											
Address	**File Name**	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**	**A-RESET**	**i-RESET**
181H	BIECN	-	-	-	-	BIEHV	-	BIEWR	BIERD	1... $000	1... $uuu
182H	BIEARH	ENBIE	-	-	-	-	-	-	-	0...	u... ....
183H	BIEARL	-	-	BIE Address Register as BIEARL[5:0]						..xx xxxx	..uu uuuu
184H	BIEDRH	BIE High Byte Register								xxxx xxxx	uuuu uuuu
185H	BIEDRL	BIE Low Byte Register								xxxx xxxx	uuuu uuuu

BIECN：BIE Control Register

VPP_HIGH：Check VPP

0：VPP without external program power supply 6V

1：VPP with external program power supply 6V

BIEWR：Write in EPROM control bit

0：Unavailable to write in

1：Available to write in (BIE write in)

BIERD：Read EPROM control bit

0：Unavailable to read

1：Available to write in (BIE read)

BIEARH：EPROM Control Register

ENBIE：

0：not enable BIE function

1：enable BIE function

BIEARL：EPROM address definition

BIE_ADDR[5:0]：EPROM address, only 00H~3FH，total 64 words

BIEDRH：EPROM High Byte data definition

BIEDRL：EPROM Low Byte data definition

21. Revision Record

Major differences are stated thereinafter:

Version	Page	Revision Summary
V02	All	New Release
V03	All	Register name revised.
	77	Add Example of Calculation.
	114~116	Revise AGND[P/N] voltage table.
	126	Revise Multifunction network application circuit.
	146~147	Revise capacitor range application circuit
V04	21, 106	Add LVDCN description
	22	Add BIE related register
	26	External oscillator example program
	30	ADCCK description
	74	Counter icon SYSCLK modify CPU CK.
	131	AD1 output data range
	145~148	Add BIE instruction
V05	27	Revise HS_DCK Outputs 4MHz (External Oscillator) Example Program
	56, 57	Revise DA3.5 description
	91, 92	Revise UART operating frequency as CPU_CK
	133, 134	Revise ADCOSR<2:0> and ADC Min value Add RSTCOMB description
	139	Add RSLPF description Add RSRMS description