FT61F02X

MSCK Application note

Table of contents

FT61F02x MSCKapplication

1.Oscillator and System Clock

system clock (SysClk)Can be selected as internal high-speed oscillator by commandHIRC, the internal low-speed oscillatorLIRC, or an external oscillator (EC, LP, XT,See " SCS",**surface1-2**). If an external oscillator is selected, then the initialization configuration register "FOSC" (**surface1-1**)Decided to choose3One of the external oscillators. The system clock can be further selected as the frequency division of the internal oscillator by instruction (refer toIRCF,**surface 1-2**). The system clock is used to generate the instruction clock (Instruction Clock):

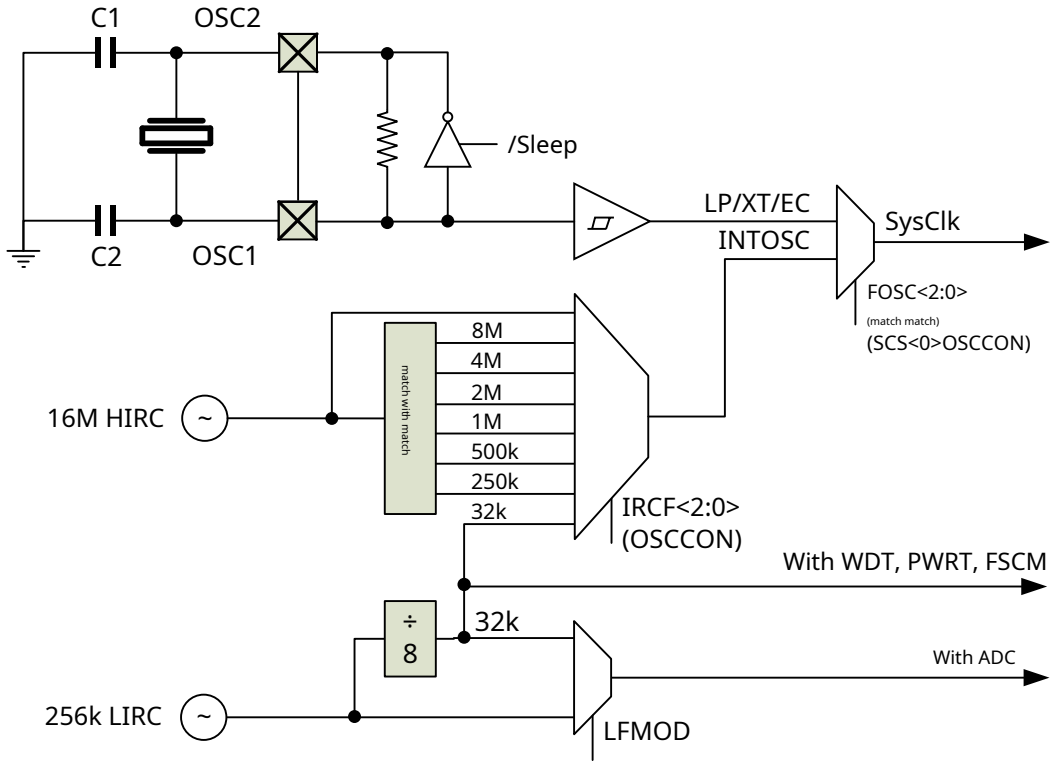
instruction clock =SysClk / N; N = 2 for 2T, 4 for 4T.

The external clock input and the internal instruction clock output are set by the initialization configuration register (seeFOSC).

TimersandADCThe modules have independent oscillators, so multiple oscillators can run simultaneously.

whenTimersWhen enabled, the selected oscillator will be automatically turned on, and theTimersIt remains valid during operation. When the corresponding oscillator is inSLEEPmode to keep running, theADC, TimersandPWMfunction is also available inSLEEPwork from time to time.

SLEEPIn the mode, the instruction stops running, and the instruction clock will also be stopped, so the peripheral module that selects the instruction clock as the clock source will also be in the SLEEPmode stops working.



picture1-1system clockSysClkClock Source Block Diagram

1.1. Oscillator Module Related Register Summary

name	Features	default
FOSC	- LP:PA7 (+)andPA6 (−)Connect to external low-speed crystal - oscillator XT:PA7 (+)andPA6 (−)Connect to external high-speed - crystal oscillator EC:PA7 (+)connected to an external clock - input,PA6forI/O INTOSC:PA6output "instruction clock",PA7for - I/O INTOSCIO :PA7andPA61forI/O	INTOSCIO
IESO	XT/LPTwo-Speed Clock Startup - Enable - closure	Enable
FCMEN	Fail-Safe Clock Monitor - Enable - closure	Enable
TSEL	Correspondence between instruction clock and system clock (2T or 4T) - 2 (instruction clock =SysClk/2) 4 (- instruction clock =SysClk/4)	2

surface1-1FOSCand Two-Speed Startup initialization configuration register

SysClksystem clock source			configuration			
			SCS	IRCF	LFMOD	OST
			OSCCON[0]	OSCCON[6:4]	OSCCON[7]	(Fixed value)
			0x8F			
			RW-0	RW-101	RW-0	
external	EC		0	-	-	-
	XT		0	-	-	1,024
	LP		0	-	-	32,768
internal	HIRC	16MHz	1	111	-	-
		8 MHz	1	110	-	-
		4 MHz	1	101	-	-
		2 MHz	1	100	-	-
		1 MHz	1	011	-	-
		500 kHz	1	010	-	-
		250 kHz	1	001	-	-
	LIRC	256 kHz[1]	1	000	1	-
		32 kHz[2]	1	000	0	-

surface1-2 SysClkSystem clock source setting related user registers

[1] 256 kHz LIRConly forADC (refer toADCSandLFMOD,Error! Reference source not found.)use. System

[2] Clock Source (IRCF=000),PWRT,FSCMandWDT (WCKSRC=00)Uniform useLIRCof8crossover, which is32 kHz,regardless ofLFMOD

Why is it worth it.

name	state	register	address	reset value
OSTS	Oscillator Start Timeout Status Bit (Latched) 1 =Running under the external oscillator (boot successfully) 0 =running from the internal oscillator	OSCCON[3]	0x8F	RO−x
HTS	HIRC ready (latch) 1 = Yes 0 =no	OSCCON[2]		RO−0
LTS	LIRC ready (latch) 1 = Yes 0 =no	OSCCON[1]		RO−0
CKMAVG	LIRCandHIRCWhen cross-calibrating4sub-averaged measurement mode 1 =Enable 0 =closure	MSCKCON[2]	0x1B	RW−0
CKCNTI	start upLIRCandHIRCThe cross-calibration function of 1 =start up 0 =Done (automatically cleared)	MSCKCON [1]		RW−0
SOSCPR	calibrationLIRCcycle requiredHIRCnumber of cycles	SOSCPR[11:0]	0x1D[3:0] 0x1C	RW−FFF

surface1-3 Oscillator Control Bits/Status Bits

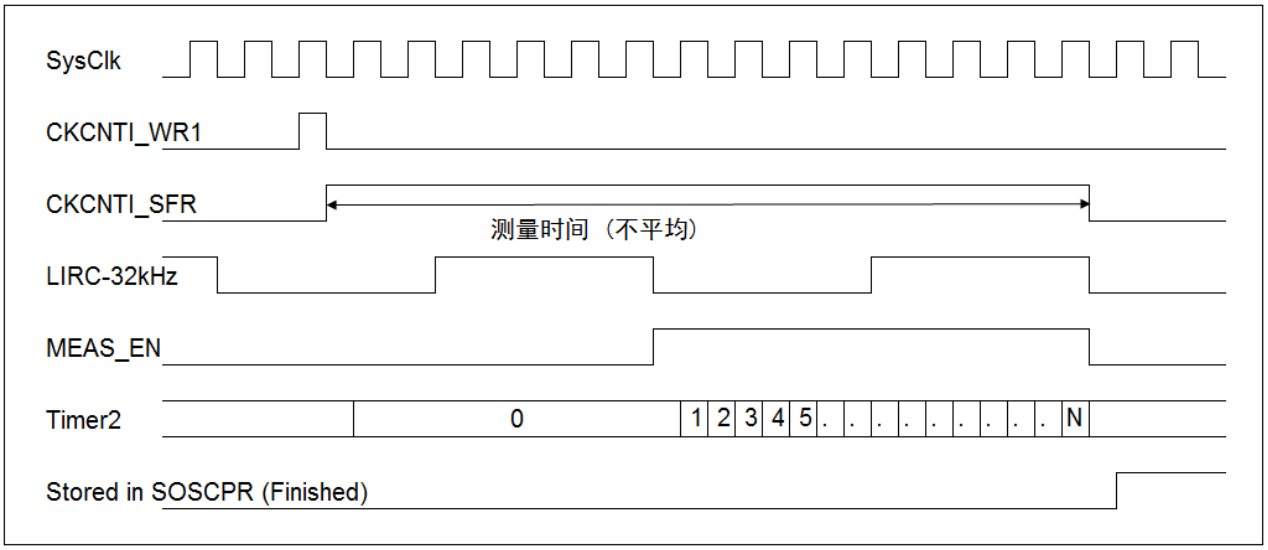
1.2. Internal Clock Mode (HIRCandLIRC)

Internal high frequency clock (Internal high frequency clock, HIRC) Factory calibrated to16 MHz @ 2.5V/25°C. Chip-to-chip frequency variation typical <±1.5% @2.5 − 5.5V/25°C, the typical temperature change is±4%@-40−+85°C.

HIRCAccuracy is calibrated at wafer test. The encapsulation process may result inHIRCfrequency drift. Writer software can choose whether to HIRCPerform a recalibration.

Internal low frequency clock (Internal low frequency clock, LIRC) Factory uncalibrated, operating frequency is32 kHz. Chip-to-chip frequency variation typical <±2% @2.5 − 5.5V/25°C, typical value of temperature change <±2%@-40−+85°C.

LIRCandHIRCCan be cross-calibrated with each other- in aLIRCPeriod (values are determined by "LFMOD"settings) usingTimer2to measure the number of instruction clocks (SysClkchoose16MHz HIRC), which is a built-in hardware function. becauseLIRCThe temperature coefficient is low, so when the temperature is unstable, it can be used by usingLIRCto calibrateHIRCfunction to achieve the same±2%temperature coefficient.



picture1-2 Single Measurement Timing Diagram

LIRCandHIRCCross-calibration steps:

1.set upIRCF = 111, SCS = 1 ; SysClkchoose16MHz HIRC (Other frequency settings will be less accurate)

2.set upCKMAVG = 1 ; 4Averaging of measurements, select0means no averaging

3.set upTMR2ON = 1 ;EnableTimer2

4.set upCKCNTI = 1 ;start calibration, defaultTimer2Prescaler =1,Postscaler =1, T2CKSRC = SysClk for 2T; SysClk/2 for 4T

5.When calibration is complete,CKCNTIautozero("CKCNTI=0"), CKMEAIFautoset("CKMEAIF = 1").

6.Measured values are stored inSOSCPRregister.

7.ifLIRCfor32kHz,andCPUoperating16MHz / 2T, the ideal matching value is500.

Note:

- LIRCandHIRCWhen cross-calibrating, do notSOSCPRH/LRegister write operation; LIRCandHIRCWhen cross-
- calibrating,Timer2Cannot be used by other peripherals; LIRCandHIRCCross-calibration function withIDEs
- The single-step debugging mode of is not compatible;

2.Application example

```
//*************************************************
********** /*file name:TEST_61F02x_MSCK.C
* Features:      FT61F02x-IOFast Clock Measurement Slow Clock Function
*IC:            Demonstration FT61F023_IO SOP16
* Crystal:       16M/2T
* illustrate:    Read fast clock measurement slow clock data in the program
*
*                    FT61F023 SOP16
*                    ---------------------------
* VDD----------|1(VDD)      (VSS)16|----------GND
* NC-----------|2(PA7)       (PA0)15|-----------NC
* NC-----------|3(PA6)       (PA1)14|-----------NC
* NC-----------|4(PA5)       (PA2)13|-----------NC
* NC-----------|5(PC3)       (PA3)12|-----------NC
* NC-----------|6(PC2)       (PC0)11|-----------NC
* NC-----------|7(PA4)       (PC1)10|-----------NC
* NC-----------|8(PC5)       (PC4)09|-----------NC
*                    ---------------------------
*/
//************************************************* ***********
# include "SYSCFG.h"
//**********************Macro definition ***************************
#define unint unsigned int

volatile unint       TestBuff;
/*---------------------------------------------- -
* Function name:POWER_INITIAL
* Features:    Power-on system initialization
* enter:    none
* output:    none
 ------------------------------------------------
--------- */ void POWER_INITIAL (void)
{
    OSCCON = 0B01110001;          //IRCF=111=16MHz/2=8MHz,0.125μs//Temporarily
    INTCON = 0;                   disable all interrupts
    PORTA = 0B00000000;
    TRISA = 0B00000000;           //PAinput Output     1-enter0-output
    PORTC = 0B00000000;
    TRISC = 0B00000000;           //PCinput Output     1-enter0-output
    WPUA = 0;                     //ban allPApull up
    WPUC = 0;                     //ban allPCpull up

    OPTION = 0B00001000;          //Bit3=1, WDT MODE, PS=000=WDT RATE 1:1
```

```
    MSCKCON = 0B00000000;
    //Bit6->0,prohibitPA4,PC5Regulated output
    //Bit5->0,TIMER2the clock isFosc //Bit4->0,
    prohibitLVR
    CMCON0 = 0B00000111;          //turn off the comparator,Cxfor numbersIOmouth
}
/*---------------------------------------------- ----
* Function name:Delay Us
* Features:    Short delay function --16M-2T--probably fast1%about.
* enter:       TimeDelay time length Delay time lengthTime μs none
* output:
 ------------------------------------------------
--------* / void DelayUs(unsigned char Time)
{
    unsigned char a;
    for(a=0;a<Time;a++)
    {
        NOP();
    }
}
/*---------------------------------------------- ----
* Function name:DelayMs
* Features:    short delay function
* enter:       TimeDelay time length Delay time lengthTime ms none
* output:
 ------------------------------------------------
--------* / void DelayMs(unsigned char Time)
{
    unsigned char a, b;
    for(a=0;a<Time;a++)
    {
        for(b=0;b<5;b++)
        {
            DelayUs(197);          //quick1%
        }
    }
}

/*---------------------------------------------- -
* Function name:SlowTimeTest
* Features:    Fast Clock Measures Slow Clock
* enter:       none
* output:      Slow clock clock measurementsTestTime
               Do not open the average mode slow clock frequency =16M/TestTime(2T)
```

```
                Open average mode slow clock frequency =16M/TestTime/4(2T)
  ------------------------------------------------
----- */ unint SlowTimeTest()
{
    unint TestTime;
    OSCCON = 0B01110001;        //IRCF=111=16MHz/2=8MHz,0.125μs//
    TMR2ON = 1;                 start timer2
    CKMEAIF = 0;                //clear flag
    CKMAVG = 0;                 //Disable average mode
                                //Note: Turn on averaging mode to output data as four clock cycles (single cycle*4)
    CKCNTI = 1;                 //Enable fast clock measurement bit, start measurement
    while(!CKMEAIF);
    CKMEAIF = 0;
    TestTime = SOSCPRH << 8;
    TestTime = TestTime + SOSCPRL;
    return TestTime;
}
/*---------------------------------------------- -
* Function name:main
* Features:   main function
* enter:    none
* output:   none
  ------------------------------------------------------
*/ void main()
{
    POWER_INITIAL();            //system initialization
    while(1)
    {
        TestBuff = SlowTimeTest();  //clock measurement
                                    //32768The value ≈488(Do not open the average mode - single cycle)
        NOP();
        NOP();
        NOP();
        DelayMs(200);               //time delay200ms
    }
}
```

Contact information

Fremont Micro Devices Corporation

5−8, 10/F, Changhong Building Ke-Ji
Nan 12 Road, Nanshan District,
Shenzhen, Guangdong, PRC 518057

Tel: (+86 755) 8611 7811
Fax: (+86 755) 8611 7810

Fremont Micro Devices (HK) Limited

16, 16/F, Block B, Veristong Industrial Centre, 34−36 Au Pui
Wan Street, Fotan, Shatin, Hong Kong SAR

Tel: (+852) 2781 1186
Fax: (+852) 2781 1144

http://www.fremontmicro.com