

文档版本	V1.0
发布日期	20210613

APT32S003 SYSCON 应用指南

APTCHIP



目录

1 概述	1
2. 适用的硬件.....	1
3. 应用方案代码说明	1
3.1 SYSCON 配置.....	1
3.2 HFOSC 做系统时钟	5
3.3 IMOSC 做系统时钟	7
3.4 EMOS 做系统时钟	8
3.5 系统时钟切换.....	10
3.6 IWDT 配置.....	14
3.7 LVD/LVR 配置	14
4. 程序下载和运行	15

1 概述

本文介绍了在APT32S003中SYSCON系统控制器。

2. 适用的硬件

该例程使用于 APT32S003 系列学习板

3. 应用方案代码说明

基于 APT32S003 完整的库文件系统，进行配置 SYSCON

3.1 SYSCON 配置

● 硬件配置

SYSCON 模块可以管理和配置系统的时钟以及和系统工作相关的功能模块，比如看门狗设置、低电压报警和复位、RESET 历史记录。

在芯片上电的初始化时，系统将自动选择 IMOSC 最高频率作为缺省工作时钟，完成上电复位和硬件初始化后，可以通过软件设置到希望的时钟源。

需要注意高速时钟 (>16MHZ) 下工作，Flash 读取速度匹配问题，必须要设置合适的 Flash wait 节拍以匹配 CPU 速度。

	WAIT	SPEED
$24\text{MHz} < \text{SYSCLK} \leq 48\text{MHz}$	2	1
$16\text{MHz} < \text{SYSCLK} \leq 24\text{MHz}$	1	1
$\text{SYSCLK} \leq 16\text{MHz}$	0	0

图 3.1.1 时钟等待

系统支持将内部时钟通过外部管脚（CLO）输出。

● 时钟结构示意图:

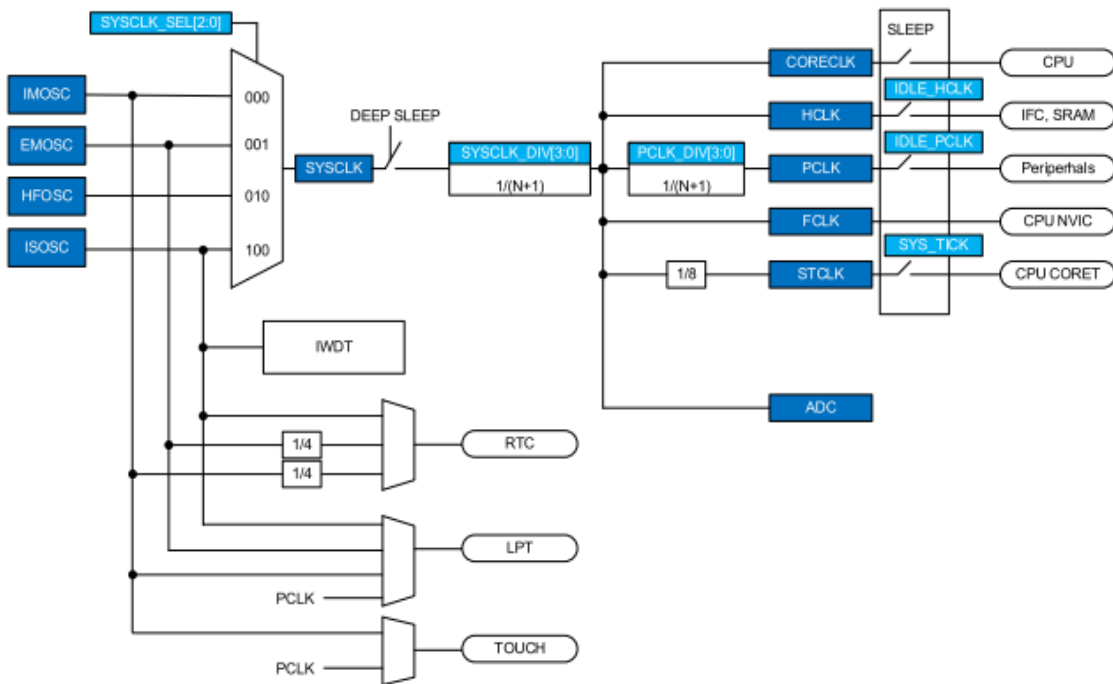


图 3.1.2 功能框图

内部主振时钟 **IMCLK**: Internal Main Clock: 131.072KHz / 2.097MHz / 4.194MHz / 5.556MHz (default) (1%偏差@典型值)

内部高速振荡时钟 **HFCLK**: High Frequency Clock: 24MHz/48MHz (1%偏差@典型值)

外部晶振时钟 **EMCLK**: External Main Clock: 400KHz 到 24MHz , 支持独立的 32.768K 配置

内部辅振 **ISCLK**: Internal Sub Clock: 27KHz (5%偏差@典型值)

● 软件配置:

可在 apt32S003_initialr.c 文件中 SYSCON_CONFIG 函数进行初始化的配置。

/*.....*/

```

//syscon Functions

//EntryParameter:NONE

//ReturnValue:NONE

/*****/

void SYSCON_CONFIG(void)
{
//-----SYSTEM CLK AND PCLK FUNTION-----/

    SYSCON_RST_VALUE(); //SYSCON all register clr

    SYSCON_General_CMD(ENABLE,ENDIS_ISOSC);

    //EMOSC_OSTR_Config(0xAD,0X1f,EM_LFSEL_EN,EM_FLEN_EN,EM_FLSEL_10ns);

    //EM_CNT=0X3FF,0xAD(36K),EM_GM=0,Low F modedisable,EM filter disable,if enable,cont set 5ns

    //SYSCON_General_CMD(ENABLE,ENDIS_EMOSC);

    SYSCON_HFOSC_SELECTE(HFOSC_SELECTE_48M); //HFOSC selected 48MHz

    //system clock set, Hclk div ,Pclk div set system clock=SystemCLK/Hclk div/Pclk div

    SystemCLK_HCLKDIV_PCLKDIV_Config(SYSCLK_HFOSC,HCLK_DIV_1,PCLK_DIV_1,HFOSC_48M);

//----- WDT FUNTION -----/

    SYSCON_IWDCNT_Config(IWDT_TIME_2S,IWDT_INTW_DIV_7); //WDT TIME 1s,WDT alarm interrupt time=1s-1s*1/8=0.875S

    SYSCON_WDT_CMD(DISABLE); //enable WDT

    SYSCON_IWDCNT_Reload(); //reload WDT

    //IWDT_Int_Enable();

//----- WWDT FUNTION -----/

    WWDT_CNT_Load(0xFF);

    WWDT_CONFIG(PCLK_4096_DIV0,0xFF,WWDT_DBGDIS);

    WWDT_Int_Config(DISABLE);

    //WWDT_CMD(ENABLE);

//----- CLO -----/

    //SYSCON_CLO_CONFIG(CLO_PA02);

    //SYSCON->OPT1=(SYSCON->OPT1&0xFFFF8000)|(0X01<<12)|(0X04<<8)|(0X00<<4);

//----- LVD FUNTION -----/

//LVDLVR Enable/Disable

    SYSCON_LVD_Config(DISABLE_LVDEN,INTDET_LVL_3_3V,RSTDET_LVL_1_9V,DISABLE_LVD_INT,INTDET_POL_fall);

    //LVD_Int_Enable();

//----- EVTRG function -----/

/* SYSCON->EVTRG=0X00|0x01<<20; //SYSCON_trgsrc0

    SYSCON->EVPS=0X00;

    SYSCON->IMER =EM_EVTRG0_ST;*/

//----- SYSCON Vector -----/

    //SYSCON_Int_Enable(); //SYSCON VECTOR

    //SYSCON_WakeUp_Enable(); //Enable WDT wakeup INT

}

```

● 代码说明:

- SYSCON_RST_VALUE();-----用于复位寄存器初值
- SYSCON_General_CMD();-----用于使能指定系统时钟使能/禁止
- SYSCON_HFOSC_SELECTE();-----用于选择 HFOSC 的频率
- EMOSC_OSTR_Config(); ----用于配置外部配置
- SystemCLK_HCLKDIV_PCLKDIV_Config();-----用于选择系统分频
- SYSCON_IWDCNT_Config(); ----用于配置 IWDT
- SYSCON_WDT_CMD(); ----用于使能 IWDT
- SYSCON_IWDCNT_Reload(); ----用于重新载入 IWDT 值
- WWDT_CNT_Load(); ----用于窗口看门狗计数刷新
- WWDT_CONFIG(); -----用于配置窗口看门狗
- WWDT_Int_Config(); ----用于使能 WWDT
- SYSCON_CLO_CONFIG(); ----用于选择 CLO 输出引脚
- SYSCON_LVD_Config(); ----用于配置 LVD 及 LVR
- SYSCON_Int_Enable(); -----用于使能 SYSYCON 中断
- SYSCON_WakeUp_Enable(); ---用于使能 SYSCON 唤醒中断

● 函数参数说明:



3.2 HFOSC 做系统时钟

HFOSC 有 48M/24M/12M/6M 可选，这里设为 48MHZ, HCLK 分频设置为 1, PCLK 分频设置为 1。

```

/*****/
//syscon Functions
//EntryParameter:NONE
//ReturnValue:NONE
/*****/
void SYSCON_CONFIG(void)
{
//-----SYSTEM CLK AND PCLK FUNTION-----/
    SYSCON_RST_VALUE();
    SYSCON_HFOSC_SELECTE(HFOSC_SELECTE_48M);          //HFOSC selected 48MHz
    //system clock set, Hclk div ,Pclk div  set system clock=SystemCLK/Hclk div/Pclk div
    SystemCLK_HCLKDIV_PCLKDIV_Config(SYSCLK_HFOSC,HCLK_DIV_1,PCLK_DIV_1,HFOSC_48M);
//----- WDT FUNTION -----/
    SYSCON_IWDCNT_Config(IWDT_TIME_2S,IWDT_INTW_DIV_7); //WDT TIME 1s,WDT alarm interrupt time=1s-1s*1/8=0.875S
    SYSCON_WDT_CMD(ENABLE); //enable WDT
    SYSCON_IWDCNT_Reload(); //reload WDT
    //IWDT_Int_Enable();
//----- WWDT FUNTION -----/
    WWDT_CNT_Load(0xFF);
    WWDT_CONFIG(PCLK_4096_DIV0,0xFF,WWDT_DBGDIS);
    WWDT_Int_Config(ENABLE);
    //WWDT_CMD(ENABLE);
//----- CLO -----/
    SYSCON_CLO_CONFIG(CLO_PA02);
    SYSCON->OPT1=(SYSCON->OPT1&0XFFF8000)|((0X04<<12)|((0X04<<8)|(0x00<<4)));
//----- LVD FUNTION -----/
    //LVD LVR Enable/Disable
    SYSCON_LVD_Config(DISABLE_LVDEN,INTDET_LVL_3_3V,RSTDET_LVL_1_9V,DISABLE_LVD_INT,INTDET_POL_fall);
    //LVD_Int_Enable();
//----- EVTRG function -----/
    //SYSCON->EVTRG=0X00|0x01<<20; //SYSCON_trgsrc0
    //SYSCON->EVPS=0X00;

    //SYSCON->IMER =EM_EVTRG0_ST;
//----- SYSCON Vector -----/
    //SYSCON_Int_Enable(); //SYSCON VECTOR
    //SYSCON_WakeUp_Enable(); //Enable WDT wakeup INT
}

```

- 代码说明:

SYSCON_HFOSC_SELECTE(HFOSC_SELECTE_48M); ----用于选择系统时钟

SystemCLK_HCLKDIV_PCLKDIV_Config(SYSCLK_HFOSC,HCLK_DIV_1,PCLK_DIV_1,HFOSC_48M);----用于配置 HFOSC 时钟

- 验证方法：可以通过 CLO (PA0.2) 输出系统时钟，但输出高频 (>10MHZ) 时需要设置分频。

SYSCON_CLO_CONFIG(CLO_PA02);---用于配置 GPIO 口为 CLO 复用功能

SYSCON->OPT1=(SYSCON->OPT1&0XFFFF8000)|(0X04<<12)|(0X04<<8)|(0x00<<4);

---用于配置输出时钟以及 CLO 分频配置。

- 参数说明：



SYSCON->OPT1=(SYSCON->OPT1&0XFFFF8000)|(0X04<<12)|(0X04<<8)|(0x00<<4)

- 波形图：

CLO 选择 8 分频输出波形 48MHZ/8 = 6MHZ.

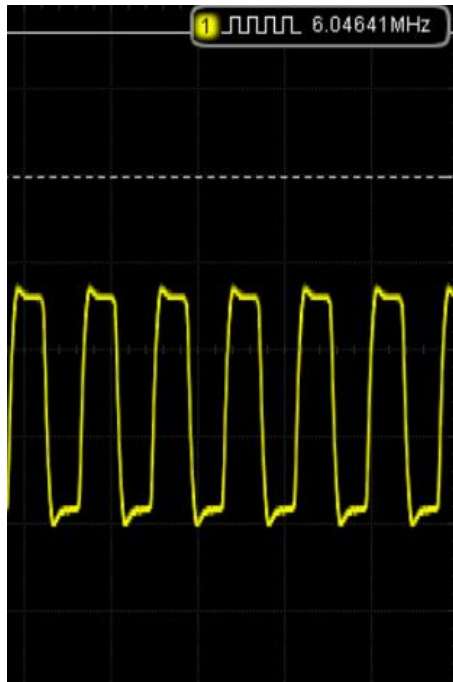


图 3.2.1 HFOSC 时钟

3.3 IMOSC 做系统时钟

IMOSC 有 5.556M/4.194M/2.097M/131K 可选，这里选择 5.556MHZ。

```

/*****/
//syscon Functions
//EntryParameter:NONE
//ReturnValue:NONE
/*****/
void SYSCON_CONFIG(void)
{
//-----SYSTEM CLK AND PCLK FUNTION-----/
    SYSCON_RST_VALUE();
    SYSCON_HFOSC_SELECTE(IMOSC_SELECTE_5556K);
    SystemCLK_HCLKDIV_PCLKDIV_Config(SYSCON_IMOSC,HCLK_DIV_1,PCLK_DIV_1,IMOSC);
//----- WDT FUNTION -----/
//WDT TIME 1s,WDT alarm interrupt time=1s-1s*1/8=0.875S
    SYSCON_IWDCNT_Config(IWDT_TIME_2S,IWDT_INTW_DIV_7);
    SYSCON_WDT_CMD(ENABLE); //enable WDT
    SYSCON_IWDCNT_Reload(); //reload WDT
    //IWDT_Int_Enable();
//----- WWDT FUNTION -----/
    WWDT_CNT_Load(0xFF);
    WWDT_CONFIG(PCLK_4096_DIV0,0xFF,WWDT_DBGDIS);
    WWDT_Int_Config(ENABLE);
    //WWDT_CMD(ENABLE);
//----- CLO -----/
    SYSCON_CLO_CONFIG(CLO_PA02);
    SYSCON->OPT1=(SYSCON->OPT1&0xFFFF8000)|((0X01<<12)|((0X01<<8)|(0x00<<4)));
//----- LVD FUNTION -----/
//LVD LVR Enable/Disable
    SYSCON_LVD_Config(DISABLE_LVDEN,INTDET_LVL_3_3V,RSTDET_LVL_1_9V,DISABLE_LVD_INT,INTDET_POL_fall);
    //LVD_Int_Enable();
//----- EVTRG function -----/
//SYSCON_trgsrc0
//SYSCON->EVTRG=0X00|0x01<<20;
//SYSCON->EVPS=0X00;
//SYSCON->IMER =EM_EVTRG0_ST;
//----- SYSCON Vector -----/
//SYSCON VECTOR
//SYSCON_Int_Enable();
//Enable WDT wakeup INT
//SYSCON_WakeUp_Enable();
}

```

● 代码说明:

SYSCON_HFOSC_SELECTE(IMOSC_SELECTE_5556K);----用于使能时钟

SystemCLK_HCLKDIV_PCLKDIV_Config(SYSCLK_IMOSC,HCLK_DIV_1,PCLK_DIV_1,IMOSC);----用于配置时钟分频

● CLO 输出波形验证:

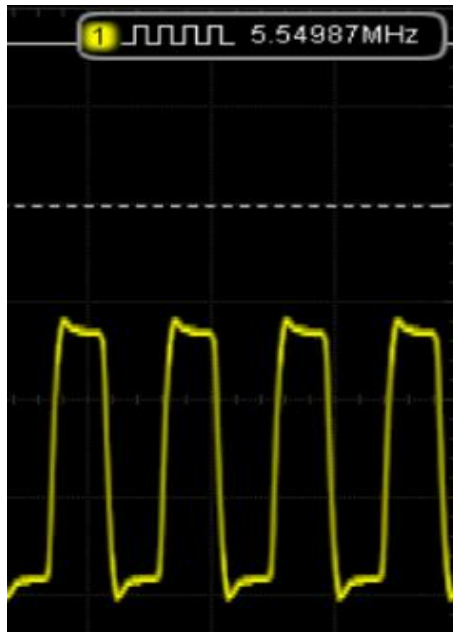


图 3.3.1 IMOSC 时钟

3.4 EMOS 做系统时钟

选择外部晶振 EMOSC，作为系统时钟。有 24M/16M/12M/8M/4M/32.768K 可选，这里选择外部 24MHZ.

● 硬件配置:

外接晶振（普通模式）	-		0.4	-	24	MHz
------------	---	--	-----	---	----	-----

图 3.4.1 引脚配置

C1/C2 电容范围 20-30pF.

```

/*****/

//syscon Functions

//EntryParameter:NONE

//ReturnValue:NONE

/*****/

void SYSCON_CONFIG(void)

{

//-----SYSTEM CLK AND PCLK FUNTION-----/

SYSCON_RST_VALUE();

EMOSC_OSTR_Config(0XAD,0X1f,EM_LFSEL_DIS,EM_FLEN_EN,EM_FLSEL_10ns);

//EM_CNT=0X3FF,0xAD(36K),EM_GM=0,Low F modedisable,EM filter disable,if enable,cont set 5ns

SYSCON_General_CMD(ENABLE,ENDIS_EMOSC);

SystemCLK_HCLKDIV_PCLKDIV_Config(SYSCLK_EMOSC,HCLK_DIV_1,PCLK_DIV_1,EMOSC_24M);

//----- WDT FUNTION -----/

//WDT TIME 1s,WDT alarm interrupt time=1s-1s*1/8=0.875S

SYSCON_IWDCNT_Config(IWDT_TIME_2S,IWDT_INTW_DIV_7);

SYSCON_WDT_CMD(ENABLE); //enable WDT

SYSCON_IWDCNT_Reload(); //reload WDT

//IWDT_Int_Enable();

//----- WWDT FUNTION -----/

WWDT_CNT_Load(0xFF);

WWDT_CONFIG(PCLK_4096_DIV0,0xFF,WWDT_DBGDIS);

WWDT_Int_Config(ENABLE);

//WWDT_CMD(ENABLE);

//----- CLO -----/

SYSCON_CLO_CONFIG(CLO_PA02);

SYSCON->OPT1=(SYSCON->OPT1&0XFFF8000)|(0X04<<12)|(0X03<<8)|(0x00<<4);

//----- LVD FUNTION -----/

//LVD LVR Enable/Disable

SYSCON_LVD_Config(DISABLE_LVDEN,INTDET_LVL_3_3V,RSTDET_LVL_1_9V,DISABLE_LVD_INT,INTDET_POL_fall);

//LVD_Int_Enable();

//----- EVTRG function -----/

//SYSCON_trgsrc0

//SYSCON->EVTRG=0X00|0x01<<20;

//SYSCON->EVPS=0X00;

//SYSCON->IMER =EM_EVTRG0_ST;

//----- SYSCON Vector -----/

//SYSCON VECTOR

//SYSCON_Int_Enable();

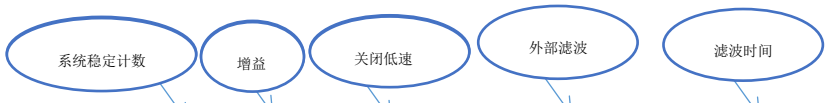
//Enable WDT wakeup INT

//SYSCON_WakeUp_Enable();

}

```

● 代码说明:



```
EMOSC_OSTR_Config(0XAD,0X1f,EM_LFSEL_DIS,EM_FLEN_EN,EM_FLSEL_10ns);
---用于配置时钟参数

SYSCON_General_CMD(ENABLE,ENDIS_EMOSC);---用于开启 EMOSC 时钟

SystemCLK_HCLKDIV_PCLKDIV_Config(SYSCLOCK_EMOSC,HCLK_DIV_1,PCLK_DIV_1,EMOSC_24M);
--- 用于配置系统时钟分频
```

● 波形图: CLO 选择 8 分频输出波形 24MHZ/8 = 3MHZ.

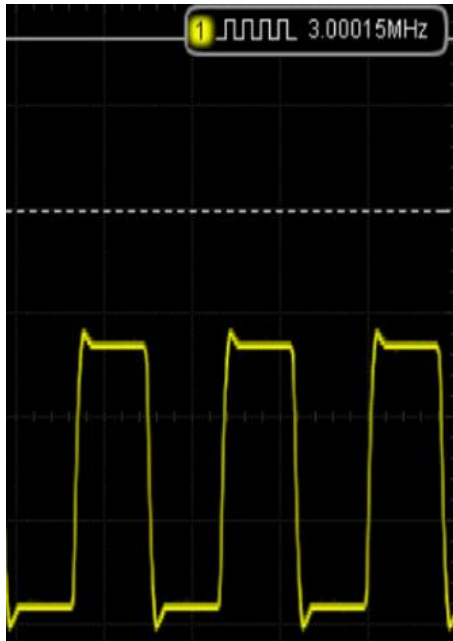


图 3.4.3 EMOS 时钟

3.5 系统时钟切换

系统时钟可以根据不同应用要求，支持在多个时钟源间进行切换。系统上电时，缺省选择 IMOSC 的 5.556MHz 作为工作时钟。在切换时钟源时，建议先执行 SystemCLK_Clear();操作将时钟源寄存器恢复为默认以保证时钟源切换稳定。

● 时钟源:

IMOSC (Internal Main OSC)	5.556MHz/4.194MHz/2.097MHz/131.072KHz
HFOSC (High Frequency OSC)	48MHz/24MHZ/12MHZ/6MHZ
EMOSC (External Main OSC)	32.768KHz / 36KHz ~ 24MHz
ISOSC (Internal Sub OSC)	27KHz

例 1: 系统完成上电复位和硬件初始化后,使能 EMOSC 时钟,再切换到内部 HFOSC,并把 HFOSC 作为系统时钟。

```

/*****/
//syscon Functions
//EntryParameter:NONE
//ReturnValue:NONE
/*****/
void SYSCON_CONFIG(void)
{
//-----SYSTEM CLK AND PCLK FUNTION-----/
SYSCON_RST_VALUE(); //SYSCON all register clr
SYSCON_General_CMD(ENABLE,ENDIS_ISOSC); //SYSCON enable/disable clock source

SYSCON_General_CMD(ENABLE,ENDIS_EMOSC);
EMOSC_OSTR_Config(0xAD,0x1f,EM_LFSEL_DIS,EM_FLEN_EN,EM_FLSEL_10ns);
//system clock set, Hclk div ,Pclk div set system clock=SystemCLK/Hclk div/Pclk div
SystemCLK_HCLKDIV_PCLKDIV_Config(SYSCLK_EMOSC,HCLK_DIV_1,PCLK_DIV_1,EMOSC_24M);
//
SystemCLK_Clear();
SYSCON_HFOSC_SELECTE(HFOSC_SELECTE_48M);
SystemCLK_HCLKDIV_PCLKDIV_Config(SYSCLK_HFOSC,HCLK_DIV_1,PCLK_DIV_1,HFOSC_48M);

//----- WDT FUNTION -----/
//WDT TIME 1s,WDT alarm interrupt time=1s-1s*1/8=0.875S
SYSCON_IWDCNT_Config(IWDT_TIME_2S,IWDT_INTW_DIV_7);
SYSCON_WDT_CMD(DISABLE); //enable WDT
SYSCON_IWDCNT_Reload(); //reload WDT
//IWDT_Int_Enable();
//----- WWDT FUNTION -----/
WWDT_CNT_Load(0xFF);
WWDT_CONFIG(PCLK_4096_DIV0,0xFF,WWDT_DBGDIS);
WWDT_Int_Config(DISABLE);

```

```

//WWDT_CMD(ENABLE); //enable wwdt

//----- CLO -----/
SYSCON->CLO_CONFIG(CLO_PA02); //CLO output setting
SYSCON->OPT1=(SYSCON->OPT1&0XFFF00FF)|(0X04<<12)|(0X03<<8)|(0x01<<4);

//----- LVD FUNTION -----/
//LVD LVR Enable/Disable
SYSCON->LVD_Config(DISABLE_LVDEN,INTDET_LVL_3_3V,RSTDET_LVL_1_9V,DISABLE_LVD_INT,INTDET_POL_fall);
//LVD_Int_Enable();

//----- EVTRG function -----/
SYSCON->EVTRG=0X00|0x01<<20; //SYSCON_trgsrc0
SYSCON->EVPS=0X00;
SYSCON->IMER =EM_EVTRG0_ST;

//----- SYSCON Vector -----/
//SYSCON_Int_Enable(); //SYSCON VECTOR
//SYSCON_WakeUp_Enable(); //Enable WDT wakeup INT
}
    
```

● **CLO 验证:**

8 分频输出主频: $8 \times 6M = 48MHz$



例 2: 系统完成上电复位和硬件初始化后,由 IMOSC 5.556MHz 时钟切换到 IMOSC 2.097MHz 时钟, 并把它作为系统时钟。

```

/*****/
//syscon Functions
//EntryParameter:NONE
//ReturnValue:NONE
    
```

```

/*****/
void SYSCON_CONFIG(void)
{
//-----SYSTEM CLK AND PCLK FUNTION-----/
SYSCON_RST_VALUE(); //SYSCON all register clr
SYSCON_General_CMD(ENABLE,ENDIS_ISOSC); //SYSCON enable/disable clock source
//
SystemCLK_Clear();
SYSCON_IMOSC_SELECTE(IMOSC_SELECTE_2097K);
SystemCLK_HCLKDIV_PCLKDIV_Config(SYSCON_IMOSC,HCLK_DIV_1,PCLK_DIV_1,IMOSC);
//----- WDT FUNTION -----/
//WDT TIME 1s,WDT alarm interrupt time=1s-1s*1/8=0.875S
SYSCON_IWDCNT_Config(IWDT_TIME_2S,IWDT_INTW_DIV_7);
SYSCON_WDT_CMD(DISABLE); //enable WDT
SYSCON_IWDCNT_Reload(); //reload WDT
//IWDT_Int_Enable();
//----- WWDT FUNTION -----/
WWDT_CNT_Load(0xFF);
WWDT_CONFIG(PCLK_4096_DIV0,0xFF,WWDT_DBGDIS);
WWDT_Int_Config(DISABLE);
//WWDT_CMD(ENABLE); //enable wwdt
//----- CLO -----/
SYSCON_CLO_CONFIG(CLO_PA02); //CLO output setting
SYSCON->OPT1=(SYSCON->OPT1&0XFFF00FF)|((0X04<<12)|(0X01<<8)|(0X01<<4));
//----- LVD FUNTION -----/
//LVD LVR Enable/Disable
SYSCON_LVD_Config(DISABLE_LVDEN,INTDET_LVL_3_3V,RSTDET_LVL_1_9V,DISABLE_LVD_INT,INTDET_POL_fall);
//LVD_Int_Enable();
//----- EVTRG function -----/
SYSCON->EVTRG=0X00|0x01<<20; //SYSCON_trgsrc0
SYSCON->EVPS=0X00;
SYSCON->IMER =EM_EVTRG0_ST;
//----- SYSCON Vector -----/
//SYSCON_Int_Enable(); //SYSCON VECTOR
//SYSCON_WakeUp_Enable(); //Enable WDT wakeup INT
}

```

● **CLO 验证:**

8 分频输出主频: 8*260.779KHZ = 2086KHZ



3.6 IWDT 配置

```
//----- WDT FUNTION -----/
//WDT TIME 1s,WDT alarm interrupt time=1s-1s*1/8=0.875S
SYSCON_IWDCNT_Config(IWDT_TIME_2S,IWDT_INTW_DIV_7);
SYSCON_WDT_CMD(ENABLE); //enable WDT
SYSCON_IWDCNT_Reload(); //reload WDT
//IWDT_Int_Enable();
```

IWDT 设置为 2s，分频为 7，则报警中断时间为 $2s \times 7/8 = 1.75S$ ，用户需要在

IWDT 设置时间内进行清狗操作，若达到报警时间未清狗，在 IWDT 中斷开启时会产生 IWDT 报警中断，报警中断产生后程序将进入中断程序 SYSCONIntHandler 的 IWDT_INT_ST 位置，报警中断后需要进行喂狗操作，否则会产生 IWDT 复位。

● 函数参数说明：



SYSCON_IWDCNT_Config(IWDT_TIME_2S,IWDT_INTW_DIV_7);

SYSCON_WDT_CMD(ENABLE); ---IWDT 使能

SYSCON_IWDCNT_Reload(); ---喂狗函数

IWDT_Int_Enable(); ----使能看门狗中断

3.7 LVD/LVR 配置

```
//----- LVD FUNTION -----/
```



```
//LVD LVR Enable/Disable
SYSCON_LVD_Config(DISABLE_LVDEN,INTDET_LVL_3_3V,RSTDET_LVL_1_9V,DISABLE_LVD_INT,INTDET_POL_fall);
//LVD_Int_Enable();
```

● 函数说明:



```
SYSCON_LVD_Config(DISABLE_LVDEN,INTDET_LVL_3_3V,RSTDET_LVL_1_9V,DISABLE_LVD_INT,INTDET_POL_fall);
```

LVD_Int_Enable(); ----用于开启 LVD 中断

LVD 电压等级为 2.1/2.4/2.7/3.0/3.3/3.6/3.9V

LVR 电压等级为 1.9/2.2/2.5/2.8/3.1/3.4/3.7/4.0V

4. 程序下载和运行

1. 将目标板与仿真器连接，分别为 VDD SCLK SWIO GND
2. 程序编译后仿真运行
3. 通过示波器查看，图 3.2.1、图 3.3.1、图 3.4.3 所示波形