

文档版本	V1.0
发布日期	20210621

# APT32F102 CORET 应用指南

**APT**CHIP



## 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 CORET 配置 .....	1
3.2 定时输出.....	3
4. 程序下载和运行 .....	5

# 1 概述

本文介绍了在APT32F102中CORET应用

## 2. 适用的硬件

该例程使用于 APT32F102x 系列学习板

## 3. 应用方案代码说明

### 3.1 CORET 配置

- **硬件配置:**

系统定时器 CORET 提供了一个 24 位循环递减的计数器，当计数器递减到 0 时，会向中断控制器发起中断请求。可以作为简单的计时，也可作为操作系统的 SYSTICK 定时器。

- **注意:**

- 1.CORET 的计数器不具有复位清零功能。在每次复位后，需要通过软件进行初始化。
- 2.产生的中断具有最高的优先级

- **模块框图:**

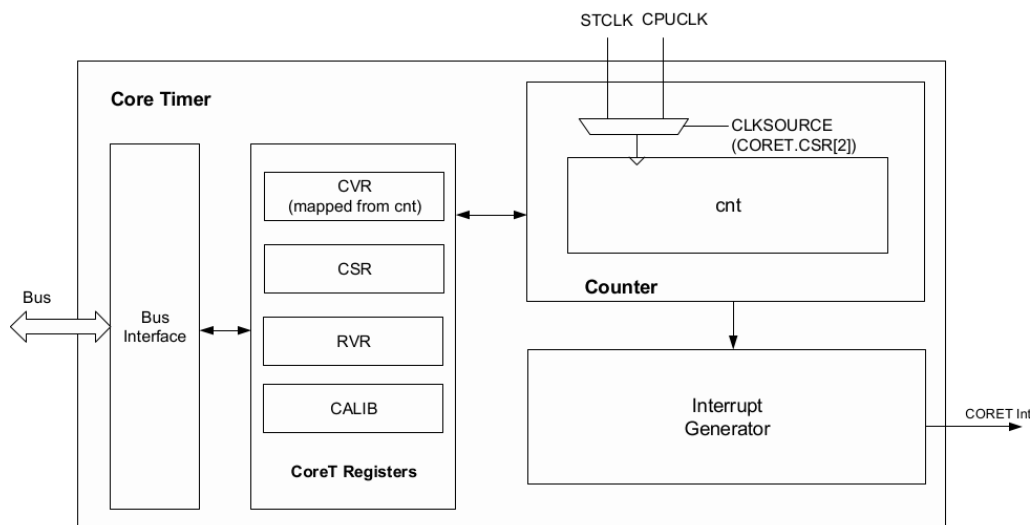


图 3.1.1 模块框图

● **软件配置:**

可在 apt32f102\_initial.c 文件中 CORET\_CONFIG()进行初始化的配置.

```

/*****
//CORET Functions
//EntryParameter:NONE
//ReturnValue:NONE
*****/
void CORET_CONFIG(void)
{
    CORET_DeInit();
    SYSCON_General_CMD(ENABLE,ENDIS_SYSTICK);
    CK801->CORET_RVR=59999;
    //CORETCLK=sysclock/8=48M/8=6Mhz e.g:10ms=(CORET_RVR+1)*(8/48M);CORET_RVR=60000-1=59999
    CORET_reload();
    CORET_CLKSOURCE_EX(); //use ex clock
    CORET_TICKINT_Enable(); //clear CORET counter
    CORET_stop();
    //CORET_start();
    CORET_Int_Enable();
    //CORET_WakeUp_Enable();
}

```

● **代码说明:**

- CORET\_DeInit();           ---- 用于恢复默认值
- SYSCON\_General\_CMD();   ---- 用于配置 CORET 时钟
- CORET\_reload();           ---- 用于清除计数值
- CORET\_CLKSOURCE\_EX();   ---- 用于配置时钟源
- CORET\_TICKINT\_Enable();  ---- 用于配置中断
- CORET\_start();           ---- 用于使能 CORET
- CORET\_stop();            ---- 用于关闭 CORET
- CORET\_Int\_Enable();      ---- 用于开启 CORET 中断
- CORET\_WakeUp\_Enable();   ---- 用于开启 CORET 唤醒中断

### 3.2 定时输出

系统时钟选择内部 48Mhz，利用 Coret 定时产生中断，PA0.14 输出周期为 20ms，占空比为 10ms

```

void GPIO_CONFIG(void)
{
    GPIO_Init(GPIOA0,14,0);
    GPIO_Write_High(GPIOA0,14);
}

/*****/
//CORET Functions
//EntryParameter:NONE
//ReturnValue:NONE
/*****/
void CORET_CONFIG(void)
{
    CORET_DeInit();
    SYSCON_General_CMD(ENABLE,ENDIS_SYSTICK);
    CK801->CORET_RVR=59999;
    //CORETCLK=sysclock/8=48M/8=6Mhz e.g:10ms=(CORET_RVR+1)*(8/48M);CORET_RVR=60000-1=59999
    CORET_reload();
    CORET_CLKSOURCE_EX(); //use ex clock
    CORET_TICKINT_Enable(); //clear CORET counter
    CORET_start();

    CORET_Int_Enable();
    //CORET_WakeUp_Enable();
}
void APT32F102_init(void)
{
    //-----/
    //Peripheral clock enable and disable
    //EntryParameter:NONE
    //ReturnValue:NONE
    //-----/
    SYSCON->PCER0=0xFFFFFFFF; //PCLK Enable 0x00410071
    SYSCON->PCER1=0xFFFFFFFF; //PCLK Enable
    while(!((SYSCON->PCSR0&0x1)); //Wait PCLK enabled
    //-----/
    //ISOSC/IMOSC/EMOSC/SYSCLK/IWDT/LVD/EM_CMFAIL/EM_CMRCV/CMD_ERR OSC stable interrupt
    //EntryParameter:NONE
    
```

```

//ReturnValue:NONE
//-----/
    SYSCON_CONFIG();                //syscon initial
    CK_CPU_EnAllNormalIrq();        //enable all IRQ
//-----/
//Other IP config
//-----/
    GPIO_CONFIG();                  //GPIO initial
    CORET_CONFIG();                 //CORET initial
}
void CORETHandler(void)
{
    static U8_T temp;
    // ISR content ...
    CK801->CORET_CVR = 0x0;         // Clear counter and flag
    nop;
    if(temp==0)
    {
        temp = 1;
        GPIO_Write_Low(GPIOA0,14);
    }
    else
    {
        temp = 0;
        GPIO_Write_High(GPIOA0,14);
    }
}
/*****/
//main
/*****/
int main(void)
{
    //delay_nms(5000);
    APT32F102_init();
    while(1)
    {
        SYSCON_IWDCNT_Reload();
    }
}

```

● 波形输出：

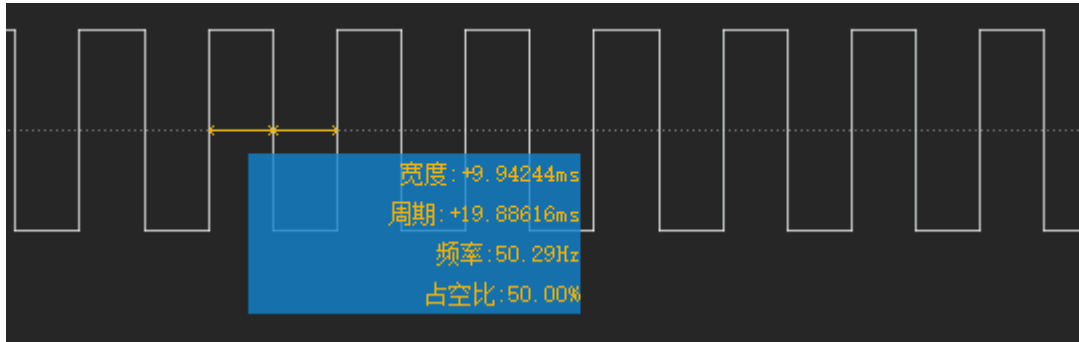


图 3.2.1 IO 翻转波形

## 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.通过示波器或逻辑分析仪查看输出波形。如图 3.2.1 所示