

文档版本	V1.0
发布日期	20191108

APT32F172 LED 驱动应用开发指南



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 硬件 LED 驱动配置	1
4. 程序下载和运行	3
5. 改版历史	4

1 概述

本文介绍了在APT32F172中使用LED驱动的应用范例。

2. 适用的硬件

该例程使用于 APT32F172 开发板 APT-DB172

3. 应用方案代码说明

基于 APT32F172 完整的库文件系统，可以很方便的对 LED 驱动进行配置。

3.1 硬件 LED 驱动配置

软件配置：

开启内部主频 20MHz, 并作为系统时钟。

使用硬件 8COM*6SEG 驱动共阴 LED 数码管。

使能 8COM 对应 IO 强驱动模式。

```
/*  
//LED Functions  
//EntryParameter:NONE  
//ReturnValue:NONE  
*/  
void GPIO_CONFIG(void)  
{  
    GPIO_DriveStrength_EN(GPIOA1,3);    //PA1.3 强驱动使能    COM0  
    GPIO_DriveStrength_EN(GPIOA1,4);    //PA1.4 强驱动使能    COM1  
    GPIO_DriveStrength_EN(GPIOA1,5);    //PA1.5 强驱动使能    COM2  
    GPIO_DriveStrength_EN(GPIOC0,0);    //PC0.0 强驱动使能    COM3  
    GPIO_DriveStrength_EN(GPIOC0,1);    //PC0.1 强驱动使能    COM4  
    GPIO_DriveStrength_EN(GPIOB0,0);    //PB0.0 强驱动使能    COM5  
    GPIO_DriveStrength_EN(GPIOB0,1);    //PB0.1 强驱动使能    COM6  
    GPIO_DriveStrength_EN(GPIOA0,0);    //PA0.0 强驱动使能    COM7  
}  
  
/*  
//LED Init
```

```

//EntryParameter:NONE
//ReturnValue:NONE
/*****/
void LED_CONFIG(void)
{
    LED_RESET_VALUE();                //LED 所有寄存器复位赋值
    LED_seg_io_initial(LED_SEG_0,0);   //led seg0 io 初始化
    LED_seg_io_initial(LED_SEG_1,0);   //led seg1 io 初始化
    LED_seg_io_initial(LED_SEG_4,0);   //led seg4 io 初始化
    LED_seg_io_initial(LED_SEG_5,0);   //led seg5 io 初始化
    LED_seg_io_initial(LED_SEG_6,0);   //led seg6 io 初始化
    LED_seg_io_initial(LED_SEG_7,0);   //led seg7 io 初始化
    LED_Control_Config(LEDCLK_DIV32,0xff,LED_Bright_100,80,1);//F_LedClk=32/20M;COM0~COM7 使能;亮度
    100%;COM 显示周期长度=(8*32/20M)*(80+7)=1.1MS;相邻 COM 互不交叠区时间时间=2 个时钟周期;
    LED_SEGDATX_data_write(SEGDAT_NUM0 , 0xaa);    //SegDat0=0xaa
    LED_SEGDATX_data_write(SEGDAT_NUM1 , 0xbb);    //SegDat1=0xbb
    LED_SEGDATX_data_write(SEGDAT_NUM2 , 0xcc);    //SegDat2=0xcc
    LED_SEGDATX_data_write(SEGDAT_NUM3 , 0xdd);    //SegDat3=0xdd
    LED_SEGDATX_data_write(SEGDAT_NUM4 , 0xee);    //SegDat4=0xee
    LED_SEGDATX_data_write(SEGDAT_NUM5 , 0xff);    //SegDat5=0xff
    LED_SEGDATX_data_write(SEGDAT_NUM6 , 0x99);    //SegDat6=0x99
    LED_SEGDATX_data_write(SEGDAT_NUM7 , 0x88);    //SegDat7=0x88
    LED_SCAN_ENABLE(ENABLE);                      //LED Scan enable
}
    
```

代码说明:

LED_Control_Config(LEDCLK_DIV32, 0xff, LED_Bright_100, 80, 1);-----

LEDCLK_DIV32 显示周期分频设定, 0xFF 分别表示各 COM 口使能位, LED_Bright_100 最高亮度, 8*DCOMCNT:COM 显示周期长度, 1*NOVCNT:相邻 COM 时间

LED_SEGDATX_data_write(SEGDAT_NUMX, 0xaa);-----表示对应 COM 口 SEG 置位, 该函数放在主循环中。

```

/*****/
//APT32F172_init
//EntryParameter:NONE
//ReturnValue:NONE
/*****/
void APT32F172_init(void)
{
    SYSCON_WDT_CMD(DISABLE);                //关闭 WDT

    SYSCON->PCER0=0xFFFFFFFF;                //使能 IP
    SYSCON->PCER1=0xFFFFFFFF;                //使能 IP
    while(!(SYSCON->PCSR0&0x1));              //判断 IP 是否使能
}
    
```

```
SYSCON_Int_Enable(); //使能 SYSCON 中断向量
SYSCON->IECR=ISOSC_ST|IMOSC_ST|EMOSC_ST|SYSCLK_ST;//使能 ISOSC 时钟稳定中断,使能 IMOSC
时钟稳定中断,使能 EMOSC 时钟稳定中断
CK_CPU_EnAllNormalIrq(); //打开全局中断

SYSCON_CONFIG(); //syscon 参数 初始化

GPIO_CONFIG(); //GPIO 初始化
LED_CONFIG (); //LED 初始化
}
```

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. 将 IO 连接到对应 LED 数码管上
3. 程序编译后仿真运行
4. 修改 LED_SEGDATX_data_write 函数，查看数码管显示情况。

5. 改版历史

版本	修改日期	修改概要
V1.0	2019-11-08	初版