MS51 SPI 实现 ARGB LED 彩虹灯灯效

NuMicro® 8 位系列微控制器范例代码介绍

文件信息

应用简述	本范例代码使用MS51 SPI 来模拟ARGB LED 的时序并实现彩虹灯灯效
BSP 版本	MS51 Series BSP V1.00.003
开发平台	NuMaker-MS51PC Ver1.0

The information described in this document is the exclusive intellectual property of Nuvoton Technology Corporation and shall not be reproduced without permission from Nuvoton.

Nuvoton is providing this document only for reference purposes of NuMicro microcontroller and microprocessor based system design. Nuvoton assumes no responsibility for errors or omissions.

All data and specifications are subject to change without notice.

For additional information or questions, please contact: Nuvoton Technology Corporation.

www.nuvoton.com

1 概述

此范例代码使用MS51 系列的SPI 来仿真ARGB LED 的时序，演示ARGB LED 彩虹灯效果。此范例代码使用一组timer 来定时更新ARGB LED 彩虹灯的颜色及一组SPI 来模拟ARGB LED 的时序。

1.1 原理

1.1.1 ARGB LED 数据传输时间(TH+TL=1.25 us ± 600 ns)

ARGB LED 时序图如图 1-1与图 1-2所示。逻辑’0’定义为高准位0.4us ± 150ns 与低准位0.85us ± 150ns; 逻辑’1’定义为高准位0.8us ± 150ns 与低准位0.45us ± 150ns。

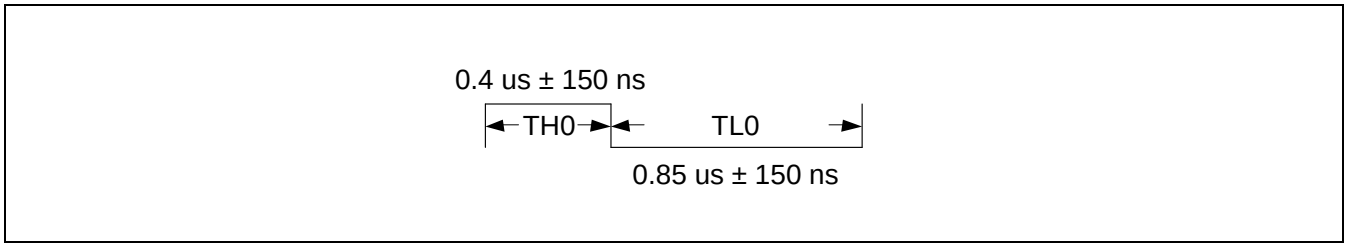

图 1-1 逻辑’0’

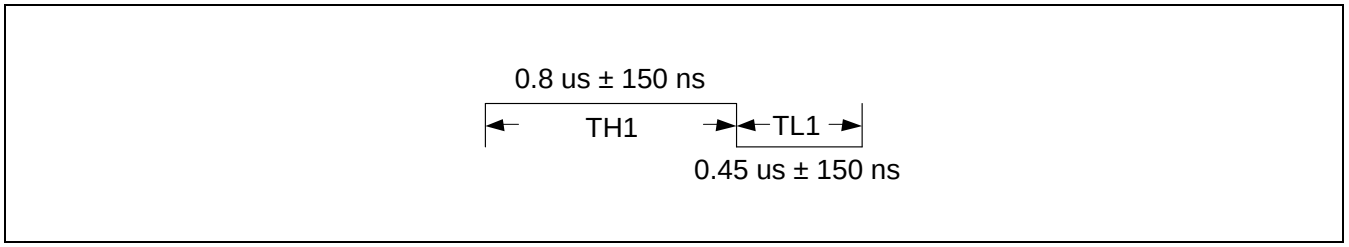

图 1-2 逻辑’1’

图 1-3 重置

1.1.2 SPI 时钟频率

程序中设定系统时钟HIRC 为24 MHz 以及SPI 时钟除频值为8，由此可计算出SPI 时钟速率为3 MHz，这代表一个SPI 时钟为 0.333 us。换句话说，逻辑’0’ 映像为SPI 位样式为‘1000’; 逻辑’1’ 映像为SPI 位样式为‘1110’，亦即3 个字节的RGB 数据格式可转化成12 个字节的SPI 数据格式。

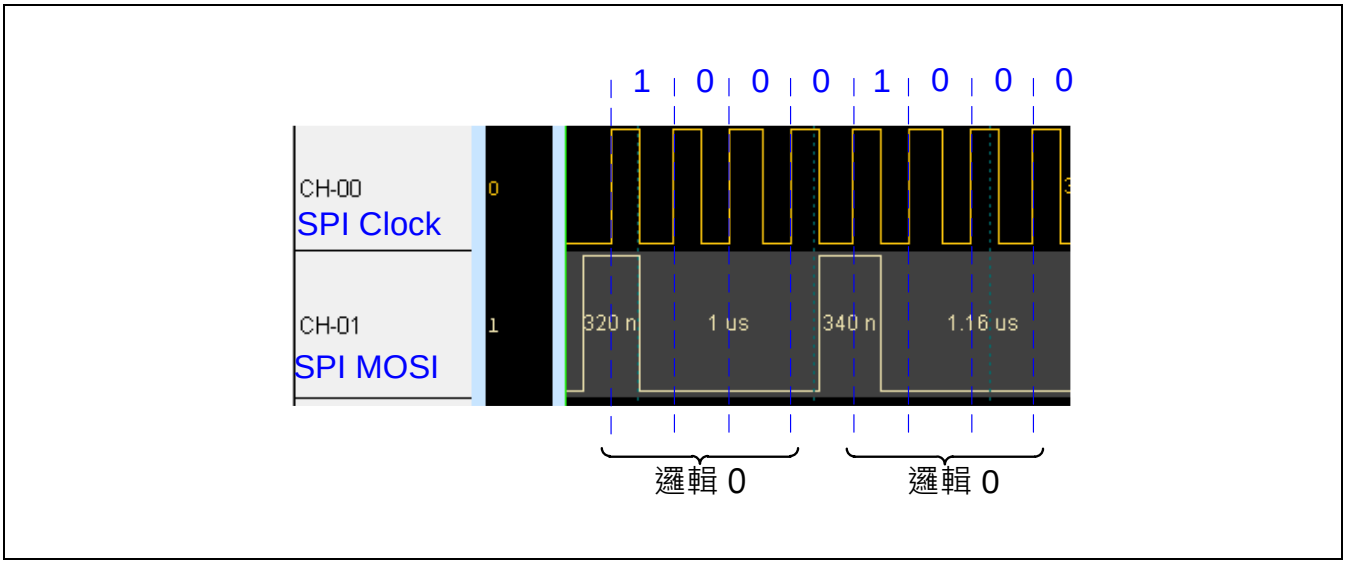

图 1-4 逻辑'0' 映像为 SPI 位样式

1.2 执行结果

执行结果。

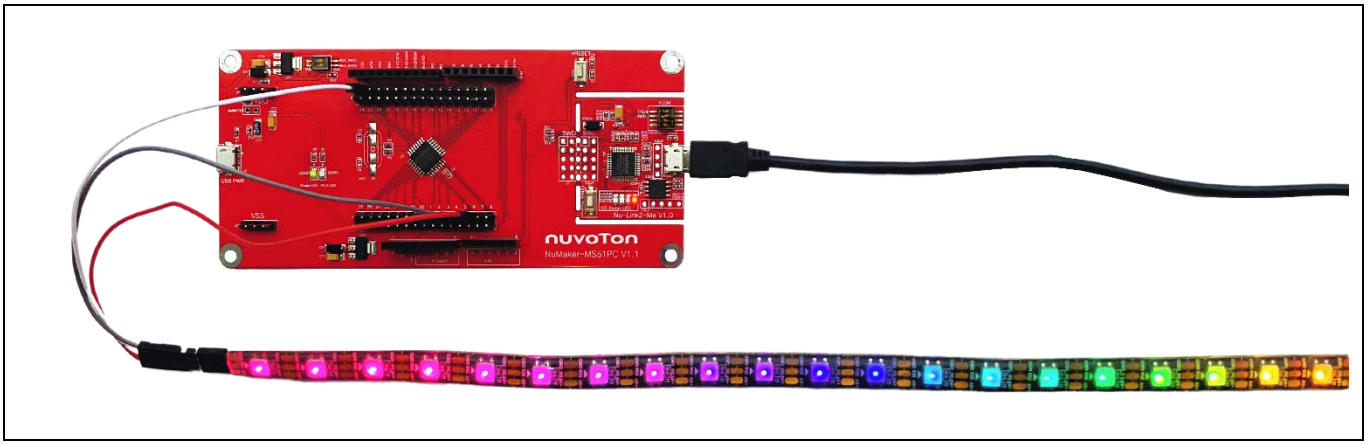

图 1-5 执行结果

2 代码介绍

代码工作流程如下：

1. main.c 中初始化 MS51 相关外设，包含 HIRC、SPI、与 Timer 3
2. 在 while 循环中呼叫 RainbowWave_TOPDOWN()来计算出相对应的 ARGB LED 颜色
3. 使用 SPI_Send() 来模拟 ARGB LED 时序

```
void main (void)
{
   MODIFY_HIRC(HIRC_24);

    /* Set LED output */
    P00 = 1;
    P00_INPUT_MODE;
    ENABLE_P00_PULLDOWN;

    SPI_Init();

    /* Enable Timer3 to add time counter */
    Timer3_1ms_initial();


    /* Init LED to all disable */
    Disable_TOPDOWN();
    SPI_Send();

    while(1)
    {

    /* Control LED */
        {
            RainbowWave_TOPDOWN();
            SPI_Send();
        }
    }
}
```

当 RainbowWave_TOPDOWN()被呼叫时，在函数内部使用它来计算周期内的 ARGB LED 颜色。

```
void RainbowWave_TOPDOWN(void)
{
    unsigned char i, j;
    /* Velocity = LEDs / Second */
    unsigned char Velocity[3] = {5, 9, 13};
    unsigned long deltaT, TotalDuration, CurrentTime, NextTime;
    unsigned char ColorCursor1, ColorCursor2;

    /* Duration for one time effect */
    TotalDuration = HDIV_Div(30 * 1000, Velocity[TOPDOWNSpeed]);
    /* Next LED time */
    NextTime = HDIV_Div(TotalDuration, 30);
    /* Current Time */
    CurrentTime = HDIV_Mod(TOPDOWNTimeCounter, TotalDuration);
    /* Unit time for each color */
    deltaT = HDIV_Div(TotalDuration, 7);
```

```
    /* Reset CountingTime */
    if(TOPDOWNTimeCounter >= TotalDuration)
    {
        TOPDOWNTimeCounter -= TotalDuration;
    }

    /* Mix Color */
    for(j = 0; j < TOP_DOWN_LED_Num; j++, CurrentTime += NextTime)
    {
        if(CurrentTime >= TotalDuration)
            CurrentTime -= TotalDuration;

        if(CurrentTime >= (deltaT * 7))
        {
            /* Color Cursor */
            ColorCursor1 = 0;
            ColorCursor2 = 1;

            for(i = 0; i < 3; i++)
            {
                TOPDOWNRGBArray[(TOP_DOWN_LED_Num-j-1)][i]=RainbowColor[ColorCursor1][i];
            }
        }
        else
        {
            /* Color Cursor */
            ColorCursor1 = HDIV_Mod(HDIV_Div(CurrentTime, deltaT), 7);
            ColorCursor2 = HDIV_Mod((ColorCursor1 + 1), 7);

            for(i = 0; i < 3; i++)
            {
                TOPDOWNRGBArray[(TOP_DOWN_LED_Num-j-1)][i] =
HDIV_Div(((RainbowColor[ColorCursor1][i] * ((ColorCursor1 + 1) * deltaT - CurrentTime)) +\

(RainbowColor[ColorCursor2][i] * (CurrentTime - ColorCursor1 * deltaT))), deltaT);
            }
        }
    }
}
```

3 颜色计算公式

以所在区域的比例，计算前后颜色的RGB混合比例

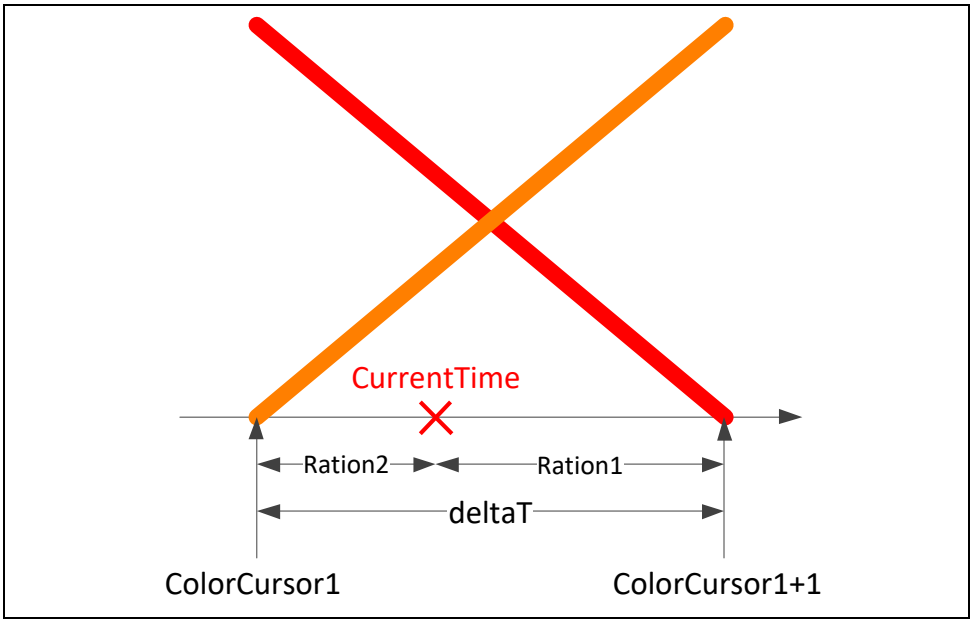

图 3-1 计算 RGB 混合比例示意图

Ration1 = (ColorCursor1 + 1) * deltaT – CurrentTime
Ration2 = CurrentTime - ColorCursor1 * deltaT

颜色计算公式:

$$Color = \frac{Color1 \times Ration1}{deltaT} + \frac{Color2 \times Ration2}{deltaT} =$$

$$\frac{Color1 \times ((ColorCursor1+1) \times deltaT - CurrentTime) + Color2 \times (CurrentTime - Cursor1 \times deltaT)}{deltaT}$$

4 软件与硬件需求

- 软件需求
  - BSP 版本
    - MS51 Series BSP V1.00.003
  - IDE 版本
    - Keil uVersion 5.26
- 硬件需求
  - 电路组件
    - NuMaker-MS51PC Ver1.0
    - ARGB LED
  - 示意图

使用 SPI MOSI 脚位来模拟 ARGB LED 时序图。

	VCC	↔	VCC	
	SPI0_MOSI (P0.0)	↔	Din	
	GND	↔	GND	
	MS51		ARGB LED	

图 4-1 接线示意图

注：ARGB LED 工作电压为 5V，故 NuMaker-MS51PC 开发板为 ARGB LED 供电电压必须更换为 5V。可以改变 Nu-Link2-Me 上的 ICEJPR1(MCUVCC)电阻使其输出 5V 电压。

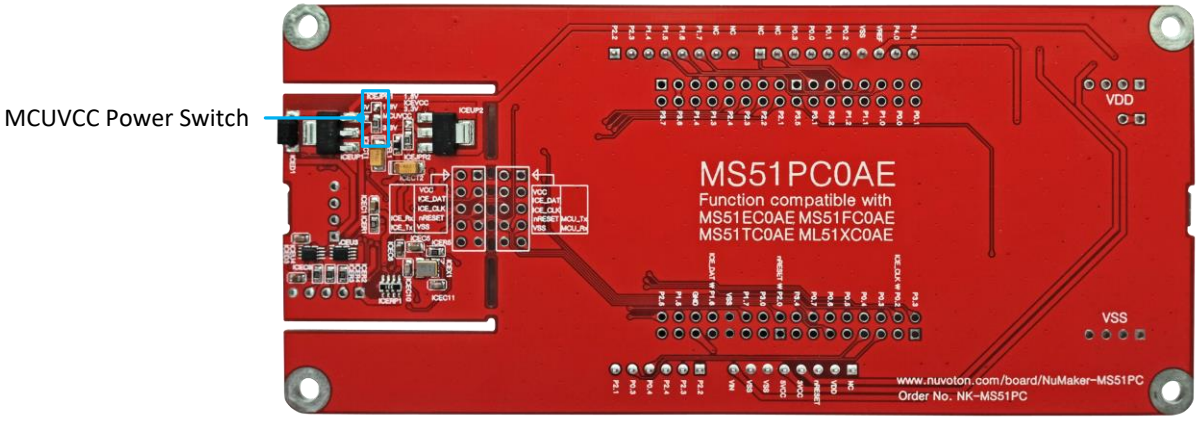

图 4-2 MCUVCC 电源切换至 5V

5 目录信息

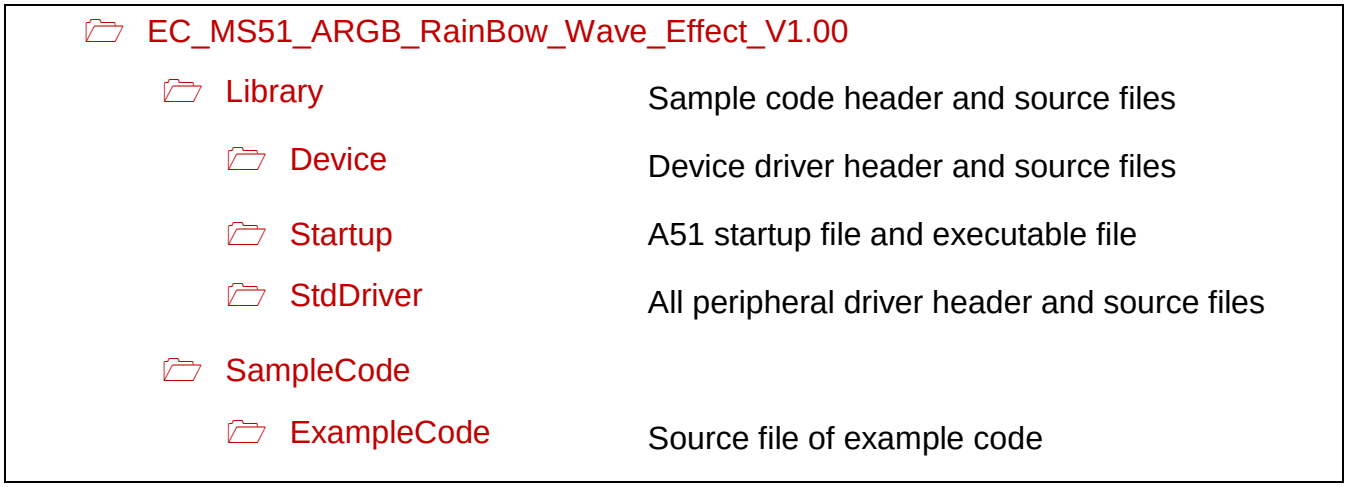

图 5-1 目录结构

6 范例程序执行

1. 根据目录信息章节进入 ExampleCode 路径中的 KEIL 文件夹，双击 *MS51_ARGB_RainBow_Wave_Effect.uvproj*。
2. 进入编译模式接口
   a. 编译
   b. 下载代码至内存
   c. 进入 / 离开除错模式
3. 进入除错模式接口
   a. 执行代码

7 修订纪录

Date	Revision	Description
2021.08.20	1.00	1. 初始发布.

Important Notice

Nuvoton Products are neither intended nor warranted for usage in systems or equipment, any malfunction or failure of which may cause loss of human life, bodily injury or severe property damage. Such applications are deemed, "Insecure Usage".

Insecure usage includes, but is not limited to: equipment for surgical implementation, atomic energy control instruments, airplane or spaceship instruments, the control or operation of dynamic, brake or safety systems designed for vehicular use, traffic signal instruments, all types of safety devices, and other applications intended to support or sustain life.

All Insecure Usage shall be made at customer's risk, and in the event that third parties lay claims to Nuvoton as a result of customer's Insecure Usage, customer shall indemnify the damages and liabilities thus incurred by Nuvoton.

Please note that all data and specifications are subject to change without notice.
All the trademarks of products and companies mentioned in this datasheet belong to their respective owners.