基于ADC 输入电压来调整PWM 占空比

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 依据 0~3V ADC 输入电压来输出占空比为 0~100%且频率为 10 kHz 的 PWM 波形。
BSP 版本	MS51_Series_BSP_Keil_V1.00.004
开发平台	NuTiny-MS51DA V1.1

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

本范例代码使用 MS51 依据 0~3V ADC 输入电压输出占空比为 0~100% 且频率为 10 kHz 的 PWM 波形。后续章节会介绍如何使用 ADC 采样输入电压并将其转换为 12 位二进制数据，也会介绍如何透过 ADC 转换结果来改变 PWM 占空比。

1.1 原理

MS51 内嵌 12 位 ADC，ADC 会将引脚上的输入电压转换为 12 位二进制数据，12 位分辨率意味 ADC 转换结果的范围为 0~4095。其转换结果会对应到 PWM 输出占空比为 0~100% 的波形。使用下列公式来表示 ADC 转换结果与 PWM 占空比的转换关系式。

$$\text{PWM 占空比} = \frac{\text{ADC 转换结果 x100}}{4095}$$

1.2 执行结果

本范例代码执行的结果如图 1-1 与图 1-2 示波器截图所示。其表示 ADC 输入电压与 PWM 占空比的关系。当 ADC 输入电压为 1.02 V 与 1.5 V 时，其 PWM 占空比分别为 34 % 与 50%。

[注] 通道 1 用来量测 ADC 输入电压；通道 2 用来量测 PWM 输出波形。

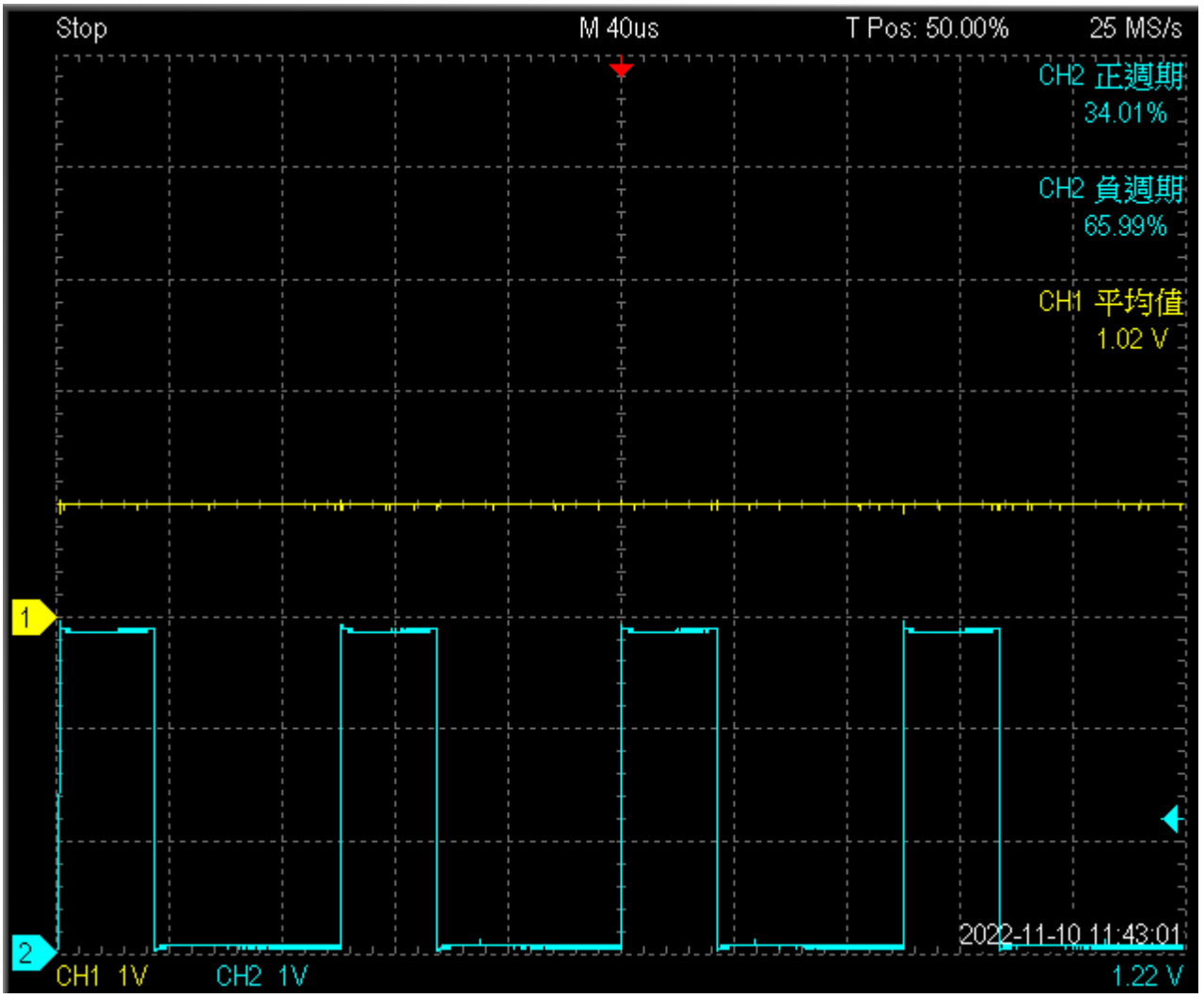

图 1-1 PWM 占空比 V.S. ADC 输入电压为 1.02 V

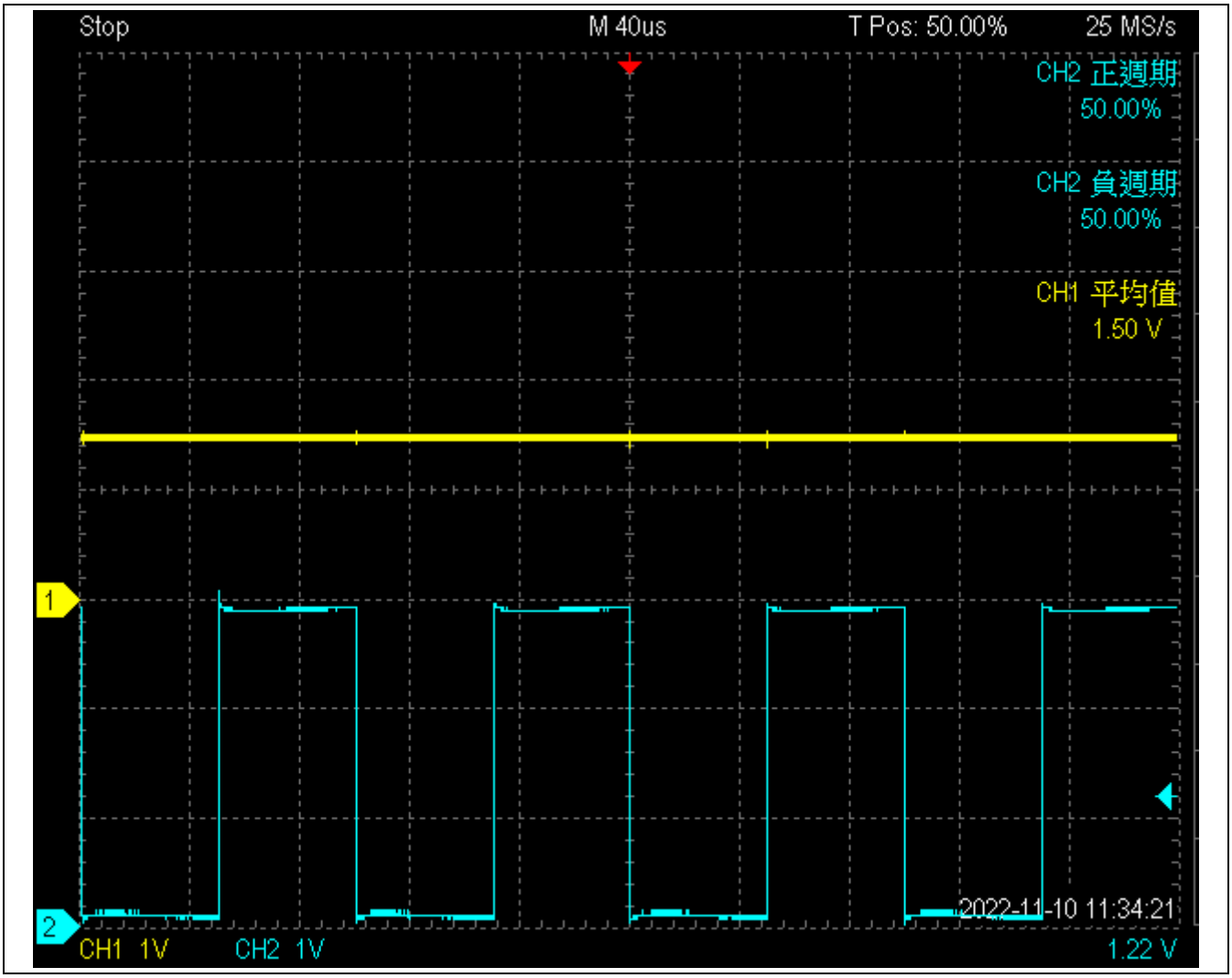

图 1-2 PWM 占空比 V.S. ADC 输入电压为 1.5V

2. 代码介绍

main() 函数可分为三个部分，第一部分为系统初始化，包含设定 HIRC 为 24 MHz 且初始化相关外设。PWM 与 ADC 相关设定请分别参考章节 2.1 与章节 2.2。 第二部分为在 IDLE_STATE 条件中启动 ADC 转换程序，且当 ADC 中断发生时获取 ADC 转换结果。最后，在 ADC_UPDATE_STATE 条件中依据 ADC 转换结果计算并更新 PWM 占空比。

```
void main(void)
{
    unsigned char state = IDLE_STATE;

    /* FSYS = 24MHz */
    MODIFY_HIRC(HIRC_24);
    PWM0_Init();
    ENABLE_GLOBAL_INTERRUPT;

    while (1)
    {
        switch (state)
        {
            case IDLE_STATE:
            default:

                ADC_Convert();
                state = ADC_UPDATE_STATE;
                break;


            case ADC_UPDATE_STATE:

                if (g_bADC_Flag == 1)
                {
                    g_bADC_Flag = 0;

                    i64_ADCdataConvert = u32_ADCdataAIN;
                    i64_ADCdataConvert = i64_ADCdataConvert * 100 / 4095;
                    /* Integer value */
                    i64_ADCdataConvert = (2 * i64_ADCdataConvert -
(int)(i64_ADCdataConvert));

                    PWM0_Reload((unsigned int)i64_ADCdataConvert);
                    DISABLE_ADC;
                    state = IDLE_STATE;
                }

                break;
        }
    }
}
```

2.1 PWM 相关函数

PWM0_Init() 函数用来初始化 PWM 模块与相关设定，包含时钟源、输出通道、PWM 输出频率与 PWM 占空比等。

PWM0_Reload() 函数用来計算並更新 PWM 占空比。

```
/**
* @brief Initialize the PWM module with specified configurations. It could be clock
source, PWM channel and frequency.
*        The default duty cycle is 50%.
*
* @param[in] None
*
* @return None
*
* @details None
*/
void PWM0_Init(void)
{
    /* Define PWM0 clock source and divider. */
    PWM0_ClockSource(PWM_FSYS, 1);

    /* Setting PWM0 channel 2 */
    ENABLE_PWM0_CH2_P10_OUTPUT;
    P10_PUSHPULL_MODE;
    PWM0_ConfigOutputChannel(2, Independent, EdgeAligned, PWM_FREQ, 50);

    PWM0_RUN();
}

/**
* @brief Update PWM duty cycle.
*
* @param[in] u16PWM0Duty Set PWM default duty cycle. The range is 0%~100%
*
* @return None
*
* @details The formula is:
*          PWM duty cycle = PWM duty register /  (PWM period register + 1)
*          -> PWM duty register = PWM duty cycle * (PWM period register + 1)
*/
void PWM0_Reload(unsigned int u16PWM0Duty)
{
    /* Reload PWM duty */
    unsigned int u16PWM0DutyCycleTemp = 0;
    u16PWM0DutyCycleTemp = ((PWM_FREQ + 1) / 100) * u16PWM0Duty;

    PWM2H = u16PWM0DutyCycleTemp >> 8;
    PWM2L = u16PWM0DutyCycleTemp;

    set_PWMCON0_LOAD;
}
```

2.2 ADC 相关函数

ADC_Convert() 函数用来初始化 ADC 模块与相关设定，包含时钟除频值、输入通道与 ADC 采样时间。 当转换完成后，可在 ADC_ISR() 函数中获取 ADC 转换结果。

```
/**
* @brief ADC interrupt service routine
*
* @param[in] None
*
* @return None
*
* @details Get ADC conversion result.
*/
void ADC_ISR(void) interrupt 11
{
    _push_(SFRS);

    clr_ADCCON0_ADCF;//Clear ADC interrupt flag

    u32_ADCdataAIN = ADCRH << 4;
    u32_ADCdataAIN |= ADCRL;
    g_bADC_Flag = 1;
    _pop_(SFRS);
}

/**
* @brief Initialize the ADC module with specified configurations and start ADC convert.
*
* @param[in] None
*
* @return None
*
* @details None
*/
void ADC_Convert(void)
{
    /*Enable channel 5 */
    ENABLE_ADC_CH5;
    ADCCON1 |= 0x30;          /* clock divider */
    ADCCON2 |= 0x0E;          /* AQT time */
    AUXR1 |= SET_BIT4;        /* ADC clock low speed */
    clr_ADCCON0_ADCF;
    set_ADCCON0_ADCS;

    /* Enable ADC interrupt */
    set_IE_EADC;
}
```

3. 软件与硬件需求

3.1 软件需求

- BSP 版本
  - MS51_Series_BSP_Keil_V1.00.004
- IDE 版本
  - Keil uVersion 5.26

3.2 硬件需求

- 电路组件
  - NuTiny-MS51DA V1.1
- 示意图
  - 将 ADC_CH5(P0.4) 脚位连接到外部电源供应器，用以得到不同的电压转换结果。
  - 将 PWM0_CH2(P1.0) 脚位连接到示波器，用以观察 PWM 输出信号。

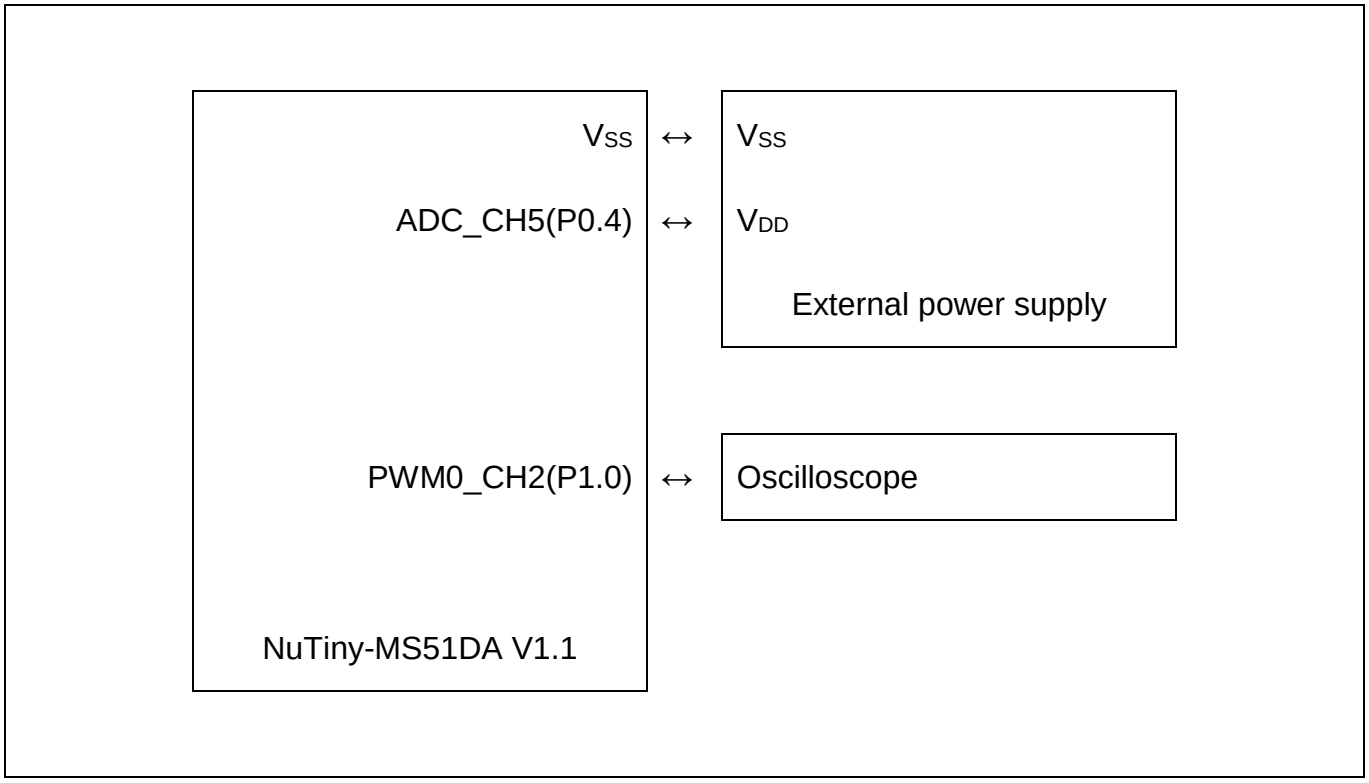

图 3-1 脚位连接

4. 目录信息

EC_MS51_ADC_PWM_V1.00	
Library	Sample code header and source files
Device	MS51 series flash device header file
Startup	A51 startup file and executable file
StdDriver	All peripheral driver header and source files
SampleCode	
ExampleCode	Source file of example code

图 4-1 目录信息

5. 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_ADC_PWM.uvproj*。
2. 进入编译模式界面
   - 编译
   - 下载代码至内存
   - 进入 / 离开仿真模式
3. 进入仿真模式界面
   - 执行代码

6. 修订纪录

Date	Revision	Description
2022.11.10	1.00	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.