使用M4 DSP做音讯频谱分析

NuMicro® 32 位系列微控制器范例代码介绍

文件信息

代码简述	本范例代码从LINE IN输入音讯并做频谱分析取得最大频率
BSP 版本	NUC505 Series BSP CMSIS V3.02.000
开发平台	NuTiny-EVB-NUC505 V1.6

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 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 功能介绍

1.1 简介

NUC505使用内部音频译码器，从LINE IN取得音讯，使用CMSIS DSP函式库做频谱分析，取得音频最大频率，用户接上耳机可以听到LINE IN的音讯。在范例程序中，仅分析右声道音频，其取样频率为8000赫兹，数据长度为16位。快速傅立叶运算的样本数为1024个，降低样本数可减少运算时间，但分辨率会变差。用户可以根据VolTH参数设定音量阈值，低于此值即认定为无声。

1.2 原理

快速傅立叶变换可以将时域中的讯号分解成频率域，如下图1-1为一类方波*f*，其为许多弦波组合而成，弦波组成其成分为图1-2，再经傅立叶变换后可以得到组成这一类方波所有弦波的频率以及强度如图1-3，因此一个讯号经过快速傅立叶变换后，可以很明显地了解其组成成分，并且根据结果来做讯号处理，如滤波等。

图 1-1 类方波 f

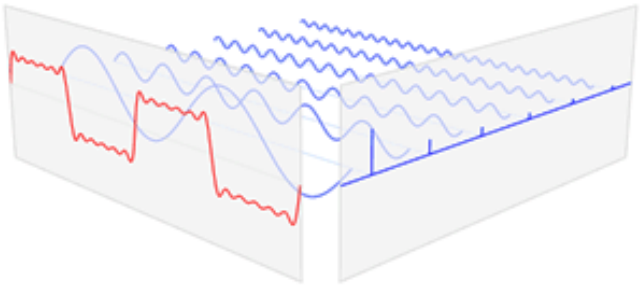


图 1-2 类方波为许多弦波组合而成

图 1-3 组合成类方波各弦波频率以及强度

CMSIS DSP函式库提供计算快速傅立叶(FFT)功能，用户输入数据即可得到最大频率强度以及位置。在此提供范例程序给用户参考，所使用到方程式定义方式如下表1-1~表1-3。

`arm_cfft_f32 (const arm_cfft_instance_f32 * S, float32_t * p1, uint8_t ifftFlag, uint8_t bitReverseFlag)`		
参数：	`*S`	[in] 指向浮点数CFFT结构
	`p1`	[in, out] 指向大小为2 * fftLen的复数矩阵(须为2的次方)，并处里完回存
	`ifftFlag`	[in] 执行正或反FFT运算(0为正，1为反)

	bitReverseFlag	[in] 是否位反转(1为正，0为反)
回传值：		无

表 1-1 复数快速傅立叶变换运算

arm_cmplx_mag_f32(float32_t *pSrc, float32_t *pDst, uint32_t numSamples)		
参数：	*pSrc	[in] 欲计算的复数数组
	*pDst	[out] 计算后实数数组
	numSamples	[in] 输入复数数组的样本数
回传值：		无

表 1-2 复数绝对值运算

arm_max_f32(float32_t *pSrc, uint32_t blockSize, float32_t *pResult, uint32_t *pIndex)		
参数：	*pSrc	[in] 欲计算的数组
	blockSize	[in] 需计算数值的样本数
	*pResult	[out] 所得最大值
	*pIndex	[out] 所得最大值的位置
回传值：		无

表 1-3 取得矩阵最大值运算

取得音频最大频率位置后，根据取样频率以及快速傅立叶样本数，即可算出音频最大频率，其算式如下：

$$\text{音频最大频率} = \frac{\text{取样频率}}{\text{FFT样本数}} \times \text{最大频率位置}$$

1.3 执行结果

执行后会打印出以下信息

```
+------------------------------------------------------------+
|      Line-In get spectrum Sample Code (internal CODEC)     |
+------------------------------------------------------------+
Sample rate is 8 kHz
NOTE: Need 3.5 mm Male to Male Cable.

The audio frequency is as follows:
406.0 Hz
406.0 Hz
414.0 Hz
421.0 Hz
421.0 Hz
429.0 Hz
437.0 Hz
445.0 Hz
445.0 Hz
453.0 Hz
Silence...
Silence...
Silence...
```

2 代码介绍

首先在main.c中对NUC505做初始化设定，使用DMA来收送数据。当一笔数据接收完成时，即会进行频谱分析，其中会判断声音是否大于阈值。

```
    /* This sample code only get audio data from right channel */
    while (1) {
        if (DataReady == 1) {
            /* Process the data through the CFFT/CIFFT module */
            arm_cfft_f32(&arm_cfft_sR_f32_len1024, RawData, ifftFlag, doBitReverse);
            /* Process the data through the Complex Magnitude Module for calculating the
magnitude at each bin */
            arm_cmplx_mag_f32(RawData, FFTOutput, BUFF_LEN / 2);
            /* Find maximum frequency */
            arm_max_f32(FFTOutput, BUFF_LEN / 2, &maxValue, &maxIndex);

            /* Check sound */
            if (maxValue < VolTH)
                printf("Silence... \n");
            else {
                /* Main frequency calculation, the resolution is sampleRate / fftSize */
                MainFreq = maxIndex * SampleRate / BUFF_LEN;
                printf("%0.1f Hz\n", MainFreq);
                DataReady = 0;
            }
        }
    }
```

当数据接收完毕时会进入I^2S中断，LINE IN输入数据存放在PcmRxBuff，HP OUT输出数据在PcmTxBuff，频谱分析数据存放在RawData。

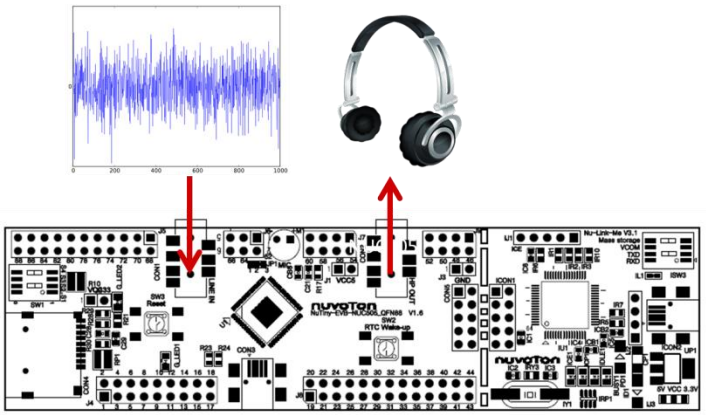
```
void I2S_IRQHandler(void)
{
    uint32_t u32I2SIntFlag;
    uint16_t i;

    /* Check interrupt flag */
    u32I2SIntFlag = I2S_GET_INT_FLAG(I2S, (I2S_STATUS_RDMATIF_Msk |
I2S_STATUS_RDMAEIF_Msk));

    /* Copy RX data to TX buffer */
```

```
    if (u32I2SIntFlag & I2S_STATUS_RDMATIF_Msk) {

        I2S_CLR_INT_FLAG(I2S, I2S_STATUS_RDMATIF_Msk);

        /* Copy RX data to TX buffer */
        for (i = 0; i < BUFF_LEN; i++) {
            PcmTxBuff[0][i * 2] = PcmRxBuff[0][i * 2];
            PcmTxBuff[0][i * 2 + 1] = PcmRxBuff[0][i * 2 + 1];
            /* Copy data to working buffer */
            RawData[i * 2] = PcmRxBuff[0][i * 2]; //real part
            RawData[i * 2 + 1] = 0; //imaginary part
        }
        /* Working buffer data is ready */
        DataReady = 1;

    } else if (u32I2SIntFlag & I2S_STATUS_RDMAEIF_Msk) {

        I2S_CLR_INT_FLAG(I2S, I2S_STATUS_RDMAEIF_Msk);

        if (s_flag1 == 0) {
            s_flag1 = 1;
            /* Enable TX -> start to play */
            I2S_ENABLE_TXDMA(I2S);
            I2S_ENABLE_TX(I2S);
        }
        /* Copy RX data to TX buffer */
        for (i = 0; i < BUFF_LEN; i++) {
            PcmTxBuff[1][i * 2] = PcmRxBuff[1][i * 2];
            PcmTxBuff[1][i * 2 + 1] = PcmRxBuff[1][i * 2 + 1];
        }
    }
}
```

3 软件与硬件环境

- 软件环境
  - BSP 版本
    - NUC505 Series BSP CMSIS V3.02.000
  - IDE 版本
    - Keil uVersion 5.26
- 硬件环境
  - 电路组件
    - NuTiny-EVB-NUC505 V1.6
    - 3.5 mm 音源线
    - 3.5 mm 接头耳机
  - 示意图

使用 3.5 mm 音源线连接板上的 LINE IN 底座，并用 3.5 mm 接头耳机连接板上的 HP OUT 底座，即可从耳机听到 LINE IN 声音，示意图如下。



4 目录信息

- EC_NUC505_LineInGetSpectrum_V1.00
  - Library — Sample code header and source files
    - CMSIS — Cortex® Microcontroller Software Interface Standard (CMSIS) by Arm® Corp.
    - Device — CMSIS compliant device header file
    - StdDriver — All peripheral driver header and source files
  - SampleCode
    - ExampleCode — Source file of example code

5 如何执行范例程序

1. 根据目录信息章节进入 ExampleCode 路径中的 KEIL 文件夹，双击 LineInGetSpectrum.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

6 修订纪录

Date	Revision	Description
Jul. 03, 2019	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.