NUC505 MP3 Decoder AMR Encoder

Example Code Introduction for 32-bit NuMicro® Family

Information

Application	This code uses NUC505 internal audio codec to play MP3 file and record AMR file in the SD card.
BSP Version	NUC505 Series BSP CMSIS V3.03.000
Hardware	NuTiny-SDK-NUC505

1 Function Description

1.1 Introduction

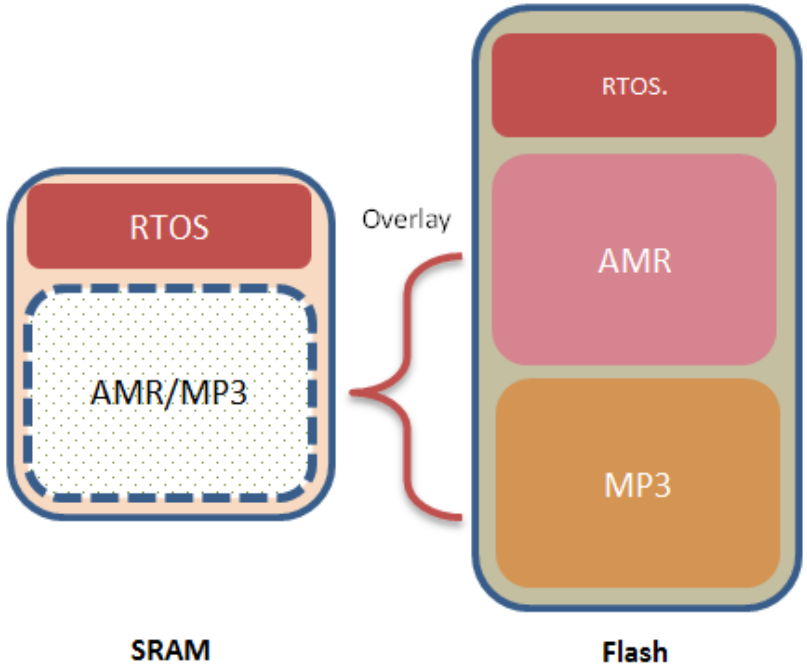
This example code uses the NUC505 internal audio decoder to play the MP3 audio file (36.MP3) in the SD card. Then, it records for 20 seconds voice. The NUC505 will encode the voice data into AMR format and store it in the SD card.

1.2 Principle

FreeRTOS manages the example code. In the FreeRTOS application, there is a scheduler for deciding which program to run when, and provides the illusion of simultaneous execution by rapidly switching between the MP3 decoder and AMR encoder.

In this example, the function of overlay SRAM is implemented. Since the application of recording and playback requires a high performance program execution to decode and encode immediately. The program data will be read from the internal SPI Flash of NUC505 to SRAM. Then, NUC505 can execute program in the SRAM. However, the MP3 decoding and AMR encoding library are quite large, there is only one function that can be put into the SRAM at the same time. The program flow will decode the MP3 file when the MP3 decoding library is put into the SRAM. After the MP3 playback is completed, the AMR Library can be overwritten in the SRAM to start the recording function. The two applications share the same SRAM space.

1.3 Demo Result




```
UART #1
Initial SDH Driver (20160602)
rc=0
Relocate vector table in SRAM (0x20000000) for fast interrupt handling. count:188
Loading overlay: EXEC=0x20010510, LOAD=0x00008b0c, LENGTH=63416 ...
Loading overlay: EXEC=0x20010510, LOAD=0x00008b0c, LENGTH=63416 DONE
FreeRTOS is starting ...
Check Task is running by message
APP1: AMR is running on SRAM.
Initial SDH Driver (20160602)
```

2 Code Description

Create MP3 decoder task, AMR encoder task and switch task：

```
void vStartAPPTasks(unsigned portBASE_TYPE uxPriority)
{
    /* Spawn the task. */
    xTaskCreate(vSWITCHTask, (signed char *) "SWITCH", (unsigned short) 120, NULL,
     uxPriority, &SWITCH);
    xTaskCreate(vAMRTask, (signed char *) "AMR", (unsigned short) 2000, NULL, uxPriority,
     &AMR);
    xTaskCreate(vMP3Task, (signed char *) "MP3", (unsigned short) 1000, NULL, uxPriority,
     &MP3);


    vTaskSuspendAll();
    load_overlay(ovly_A); /*load AMR library to SRAM*/


    xTaskResumeAll();
    vTaskSuspend(MP3);    /*Suspend MP3 task*/
}
```

Set AMR encoder task：

```
static portTASK_FUNCTION(vAMRTask, pvParameters)
{
    /* The parameters are not used. */
    (void) pvParameters;
    for(;;) {
        printf("APP1: AMR is running on SRAM.\n");
        I2S_Open(I2S, I2S_MODE_MASTER, 8000, I2S_DATABIT_16, I2S_MONO, I2S_FORMAT_I2S,
        I2S_ENABLE_INTERNAL_CODEC);


        /* Open MCLK*/
        I2S_EnableMCLK(I2S, 8000 * 256);


        /* Init AMR codec */
        amrInitEncode();
        /* Create AMR file */
        amr_recorder_th(AMR_FILE);


        /* Init one of three input source */
```

```
        demo_LineIn();

        /* record 20 seconds then close file */
        taskENTER_CRITICAL();
        amr_recorder_bh(20);
        taskEXIT_CRITICAL();

        /* Clean AMR codec */
        amrFinishEncode();

        vTaskDelay(pollqTASK_DELAY);
    }
}
```

Set MP3 decoder task :

```
static portTASK_FUNCTION(vMP3Task, pvParameters)
{
    /* The parameters are not used. */
    (void) pvParameters;

    for(;;) {
        if(Mp3App_StartDecode(u8MP3FileName) == FALSE) {/* Decode MP3 header*/
            goto _app_end_;
        }

        while(app_end) {
            i32TxDmaLastAddr = i32TxDmaCurAddr;
            i32TxDmaCurAddr = I2S_GET_TXDMA_CADDR(I2S);
            i32TxDmaDiff = i32TxDmaCurAddr - i32TxDmaLastAddr;

            if((TX_SAMPLE_SIZE - i32TxDiff) >= (MP3APP_PCM_BUF_SIZE * 2 * 2)) {
                 /* Transfer MP3 data to PCM data*/
                if(Mp3_DecodeProcess(g_au8Mp3WorkBuf,g_au8Mp3TempBuf,g_aiWorkingBuf,Mp3App
                _ResLdrGetMp3DataFromSPI,NULL) != MP3_DECODE_SAMPLE_PER_FRAME*2)
                {
                    break;
                }

                /* Copy working buffer to I2S TX DMA buffer*/
                for(u32i = 0; u32i < (MP3APP_PCM_BUF_SIZE * 2); u32i++) {
```

```
                    if(u32TxIdx >= TX_SAMPLE_COUNT) {
                        u32TxIdx = 0;
                        u32TxBufFlag = 1;
                    }
                    g_aiTxBuf[u32TxIdx++] = g_aiWorkingBuf[u32i];
                }
            }
        }
        /*MP3 playback stop*/
        I2S_DISABLE_TX(I2S);
        I2S_DISABLE_TXDMA(I2S);
        f_close(&mp3FileObject);
        printf("Stop ...\n");

        vTaskDelay(pollqTASK_DELAY);
    }
}
```

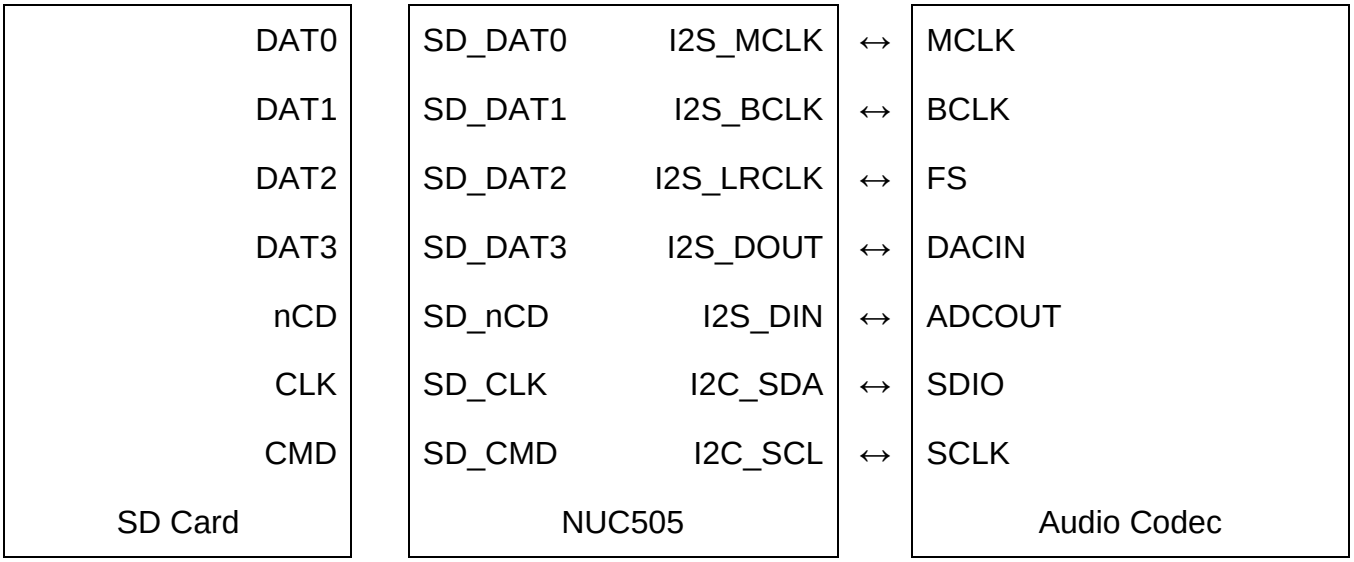
3 Software and Hardware Environment

Software Environment

- BSP version
  - NUC505 Series BSP CMSIS V3.03.000
- IDE version
  - Keil uVersion 5.26

Hardware Environment

- Circuit components
  - NuTiny-SDK-NUC505
- Diagram



4 Directory Information

- EC_NUC505_MP3_Decoder_AMR_Encoder_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
  - ThirdParty
    - FatFs — A generic FAT file system module for small embedded systems. Its official website is: http://elm-chan.org/fsw/ff/00index_e.html.
    - FreeRTOS — A real time operating system available for free download. Its official website is: http://www.freertos.org/
    - LibMAD — A MPEG audio decoder library which currently supports MPEG-1 and the MPEG-2 extension to lower sampling frequencies, as well as the de facto MPEG 2.5 format. All three audio layers — Layer I, Layer II, and Layer III (i.e. MP3) are fully implemented. This library is distributed under GPL license. Please contact Underbit Technologies (http://www.underbit.com/) for the commercial license
    - LibAMR — An AMR audio encoder library

5 How to Execute Example Code

1. Browsing into sample code folder by Directory Information (section 4) and double click NUC505_MP3_Decoder_AMR_Encoder.uvproj.
2. Enter Keil compile mode
   a. Build
   b. Download
   c. Start/Stop debug session
3. Enter debug mode
   a. Run

6 Revision History

Date	Revision	Description
Jul. 11, 2019	1.00	1. Initially issued.

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.