DOI: 10.35598/mcfpga.2025.015

Application of the Voice Channel for Controlling a Microcontroller-Based Digital System

Oleksandr Vorgul
Scientific adviser
ORCID 0000-0002-7659-8796
dept.Microprocessor Technologies and System
Kharkiv National University of Radio Electronics
Kharkiv, Ukraine
oleksandr.vorgul@nure.ua

Hlib Serbskyi
dept.Microprocessor Technologies and System
Kharkiv National University of Radio Electronics
Kharkiv, Ukraine
hlib.serbskyi@nure.ua

***Abstract*— This article provides a structured overview of the capabilities of the STM32F4Discovery development board (based on the STM32F407VGT6 microcontroller) in the context of audio input and output processing. The main focus is on physical interfaces, hardware resources, software control models, and basic audio stream encoding/decoding operations, excluding speech command analysis and recognition tasks.**

***Keywords*— STM32F4Discovery, STM32F407, audio I/O, I2S, CS43L22 DAC, MEMS microphone, DMA, audio codecs, FATFS, SD card.**

I. Introduction

Modern embedded systems increasingly require the integration of audio processing capabilities, whether it is simple notification playback or complex speech command processing. The STM32F4Discovery development board, based on the STM32F407VGT6 microcontroller, is a versatile platform that allows developers to explore and implement various aspects of audio in the context of embedded applications.

The board's audio capabilities are based on a carefully designed architecture that combines a powerful Cortex-M4F core with dedicated audio components. This combination opens up broad prospects for applications that can not only play and record audio, but also process it in real time.

II. Architectural foundations of the sound subsystem

The STM32F4Discovery audio subsystem is built around a central element - the Cirrus stereo DAC Logic CS43L22 [2] This 24-bit converter, with support for sampling rates up to 192 kHz, is much more than a simple digital-to-analog converter. It includes an integrated power amplifier sufficient to drive headphones directly, eliminating the need for external amplifier stages for most applications.

The interaction between the STM32F407 microcontroller and the CS43L22 audio chip is carried out via two key interfaces. The main channel for transmitting digital audio is the I2S interface, using the PB10 (CK), PC7 (SD), PA4 (WS) and optionally PA6 (MCK) pins [1, 2, 4]. In parallel with this, the CS43L22 parameters are controlled - the choice of signal source, volume level, operating modes - via the I2C interface, using the PB6 (SCL) and PB9 (SDA) pins [1, 2].

A special feature of this architecture is the absence of a dedicated hardware audio codec, which is typical for many other debug boards. Instead, the main load on processing the audio stream falls on the STM32F407 central processor [1, 4]. This approach, on the one hand, requires more careful planning of computing resources, but on the other hand, it gives the developer full control over the sound processing algorithms.

III. Audio Capture Capabilities and Their Limitations

To capture audio, the board uses an integrated MEMS microphone, typically the MP45DT02 from STMicroelectronics or similar [3]. This omnidirectional microphone has a typical bandwidth of 100 Hz to 10 kHz and a sensitivity of about -26 dBFS . The analog signal from the microphone is fed directly to a dedicated CS43L22 DAC input, where it is digitized using the ADC path [2, 3].

This input path organization scheme, although it ensures the compactness of the solution, imposes certain restrictions on the flexibility of the system. In particular, the absence of a linear input on the CS43L22 [2] limits the possibilities of connecting external sound sources, which may be critical for some applications.

At the same time, the STM32F407 microcontroller has built-in 12-bit ADCs (3 modules, 16 channels) with a sampling frequency of up to 3 MSPS [1, 5], which theoretically can be used to digitize external analog audio signals. However, the implementation of this approach requires careful design of the analog path and does not provide direct support for the I2S protocol, which complicates integration with the rest of the audio system.

IV. The Role of the I2S Interface in an Audio System

Interface I2S (Inter-IC Sound) plays a central role in the STM32F4Discovery audio architecture, serving as the primary transport mechanism for transferring digital audio between the processor and the audio chip [2, 4]. This dedicated serial synchronous interface was designed exclusively for working with digital audio and provides high-quality transfer of PCM data without the jitter inherent in more general-purpose interfaces such as USB or UART.

The operating principle of I2S is based on the use of three main signal lines. The clock signal CK (Bit Clock) determines the data transfer rate, its frequency is calculated as the product of the sampling frequency by the number of bits per sample and the number of channels. Signal WS (Word Select) is used to separate the left and right channels in a stereo system, while its frequency is equal to the sampling frequency. The data itself is transmitted via the SD line (Serial Data) synchronously with the clock signal CK [4].

Additionally, the MCK (Master) signal can be used. Clock) is a high-frequency clock signal supplied from the master to the slave to synchronize the internal phase-locked loop (PLL) detectors of audio codecs. On STM32F4Discovery, the PA6 (MCO1) pin can be used to generate MCK [1].

In the context of this board, the STM32F407 typically acts as the I2S Master, generating all necessary clock signals and controlling the data flow [1, 4]. This configuration provides full control over the timing characteristics of the audio system and allows performance to be optimized for specific tasks.

V. *Software aspects of encoding and decoding*

The absence of a hardware codec on the STM32F4Discovery board means that all audio stream compression and decompression operations must be performed by software on the central processor. The STM32F407VGT6 microcontroller with a Cortex-M4F core operating at 168 MHz has sufficient computing resources to implement various sound processing algorithms [1, 5].

The key advantage of this processor is the presence of a hardware floating-point unit (FPU), which is critical for the efficient implementation of audio compression algorithms. Additionally, the processor supports a set of SIMD instructions (Single Instruction Multiple Data) and MAC (Multi-Accumulate), optimized for digital signal processing tasks [5].

The basic format for the system operation is PCM (pulse code modulation) - an uncompressed format that serves as the basis for interaction with the CS43L22 DAC/ADC via the I2S interface [2, 4]. Processing PCM data creates virtually no additional load on the processor, since the main work on data transfer is performed by the DMA controller.

For applications requiring data compression, the most practical solution is ADPCM (Adaptive Differential Pulse Code Modulation). This algorithm provides simple compression without or with low losses and is characterized by low computational complexity. The STM32CubeF4 library contains ready-made examples of ADPCM implementation, which significantly simplifies its integration into projects [6]. When working with a sampling frequency of 16 kHz and a bit depth of 16 bits, ADPCM usually takes up less than 10-20% of CPU resources and provides a delay of several milliseconds [5, 6].

More complex formats such as MP3 can also be implemented on this platform, although with certain limitations. MP3 decoding is possible using optimized libraries such as Helix or libmad , but requires significant CPU and memory resources. When working with a 44.1 kHz/128 kbps quality stream, MP3 decoding can take 30-70% of the processor time and introduce a delay of about 50-100 ms [1, 5]. Real-time high-quality MP3 encoding on the STM32F407 is usually impractical due to limited performance.

AAC, Ogg formats Vorbis and FLAC can theoretically also be implemented, but require even more computing resources and complex libraries, which makes their use on this platform very problematic for real-time tasks with high bit rates [5].

VI. Organizing audio stream management

Effective management of audio streams on the STM32F4Discovery requires coordination of several microcontroller subsystems. The process begins with clock initialization, which includes activation of clock signals for GPIO, I2C, SPI/I2S, and DMA controllers [1, 5]. Next comes configuration of the corresponding GPIO pins for operation with the I2S and I2C interfaces, and optionally for generating a master clock [1, 5].

The CS43L22 is configured via the I2C interface by writing parameters to the chip's control registers that determine the input signal source, output stage mode, volume level, I2S data format, and power consumption mode [2, 5, 6]. In parallel with this, the SPI/I2S module of the microcontroller is configured by setting the operating mode (Master Tx / Rx), data transmission standard (Philips /MSB/LSB), bit depth (16/24/32 bit) and sampling frequency [1, 4, 5].

A critical element of the system is the DMA controller, which provides automatic data transfer between the buffers in the RAM and the I2S data registers. The use of DMA is almost mandatory for the implementation of real-time audio applications, since it frees the central processor from the routine operations of transferring each audio sample [1, 5]. Setting up the DMA includes configuring separate channels for transmitting (TX) and receiving (RX) data, as well as setting interrupts for half-filled and full buffer events.

After initialization is complete, the audio stream is controlled via the DMA interrupt mechanism. The main application logic is executed in the interrupt handler for a half-buffer or a full buffer: for the output stream, it is filling the next half of the buffer with new audio data (the result of decoding or synthesis), for the input stream, it is processing the data received from the filled half of the buffer (saving, transferring for encoding or analysis) [5, 6].

VII. Integration with external storage devices

Although the STM32F4Discovery board itself does not contain integrated non-volatile storage, the STM32F407 microcontroller provides several interfaces for connecting external storage devices [1]. The most common solution is to use SD cards, which can be connected via the SPI interface (available via expansion connectors CN7, CN10, CN11) or via a dedicated hardware SDIO controller to achieve higher data exchange rates.

The STM32F407 SDIO interface provides significantly higher performance compared to the SPI mode, but requires connection to the corresponding microcontroller pins (PC8-PC12, PD2) via an expansion connector [1]. This approach is especially relevant for applications that work with high-quality audio streams or require simultaneous recording and playback.

A typical scenario for working with audio files on an SD card includes a sequence of operations: initialization of the interface (SPI or SDIO), initialization of the low-level disk driver, mounting the FATFS volume, opening the file, performing read or write operations on the audio data, and then closing the file with the volume unmounted [6, 7]. FATFS provides the ability to work with both raw PCM data (with a software-generated WAV header) and with files in compressed formats, provided that the data is pre-prepared in the buffer [5, 6, 7].

VIII. Advanced Features and Alternative Approaches

In addition to the basic audio components, the STM32F407 microcontroller provides a number of additional resources that can be used in audio applications [1, 5]. Built-in timers (TIM) can provide precise sampling frequency setting via DMA triggers or generation of various control signals. Built-in 12-bit DACs (2 channels) can be used to easily output low-quality monophonic audio bypassing the I2S interface, although their quality is significantly inferior to the CS43L22.

Of particular interest is the SAI module (Serial Audio Interface), which is a more flexible alternative to the I2S interface [1, 4]. SAI supports the TDM protocol for multi-channel systems, time-slot protocols such as AC'97 and a wider range of data formats. Although on the STM32F4 the SAI pins are not directly connected to the CS43L22, they can be used to integrate external audio devices.

USB OTG FS/HS interface opens up possibilities for implementing USB Audio devices Class (UAC), allowing the board to function as a USB microphone, USB headphones, or USB host for reading audio files from flash drives. However, implementing USB Audio requires significant software resources and careful consideration of protocol aspects [5, 8].

IX. Comparison with other STM32 family microcontrollers

In the context of the broader STM32 microcontroller family, the STM32F4Discovery board occupies an intermediate position in terms of audio processing capabilities [1]. The high-performance STM32F4xx series (F429/F7/H7) offer improved DSP capabilities, including an extended instruction set, cache memory, and more often integrate full hardware audio codecs or S/PDIF interfaces. The STM32H7 family stands out in particular, with significantly higher clock rates and performance sufficient for implementing complex codecs.

The entry-level STM32F0/F1/F3 microcontrollers also support the I2S interface, but typically lack a floating-point unit and are characterized by lower performance, which limits their use to PCM and simple ADPCM formats. The energy-efficient STM32WL/L4 series focus on low power consumption and are suitable for voice applications with low sampling rates (8-16 kHz) and simple encoding.

The most powerful solutions are represented by the STM32MP1 family — processors with the Cortex-A7 + Cortex-M4 architecture, designed to work under Linux or other high-level operating systems. Such systems have powerful software codecs such as FFmpeg or GStreamer , as well as support for complex interfaces, including HDMI Audio.

X. Conclusions

The STM32F4Discovery development board based on the STM32F407VGT6 microcontroller is a balanced platform for studying and developing embedded audio applications. The combination of a powerful Cortex-M4F computing core with a floating-point unit, a high-quality CS43L22 DAC and a flexible I2S interface creates the basis for implementing a wide range of tasks - from simple playback of notifications to basic compression and decoding of audio streams.

The STM32F4Discovery board with the STM32F407VGT6 microcontroller is a balanced platform for mastering and prototyping embedded applications related to audio and voice processing. Its strengths are the high-quality stereo CS43L22 DAC with a headphone amplifier, an integrated MEMS microphone, and a high-performance Cortex-M4F core with an FPU. This allows for efficient implementation of PCM capture and playback, basic compression/decompression of voice streams using ADPCM, and MP3 decoding for playback.

However, the limitations of the board in terms of computing power and functionality make it worth considering more powerful solutions from the STM32F7/H7/MP1 families for tasks that require complex real-time encoding, multi-channel processing, or ensuring the highest audio quality. Nevertheless, the STM32F4Discovery remains an excellent choice for mastering the principles of working with audio in embedded systems and creating functional prototypes of audio devices.

References

[1] RM0090 Reference manual : STM32F405/415, STM32F407/417, STM32F427/437 and STM32F429/439 advanced Arm®-based 32-bit MCUs [Text] / STMicroelectronics . — Rev. 18. - 2020. - 1981 p

[2] CS43L22: 24-Bit, 192 kHz Stereo D/A Converter with Integrated Headphone Amplifier [Text] : Data Sheet / Cirrus Logic Inc. — DS643F4. — 2013. — 55 p .

[3] *MP45DT02: Omnidirectional digital MEMS microphone* [Text] : Datasheet - production data / STMicroelectronics . — Rev. 3. - 2016. - 19 p .

[4] AN3126 Application note : Audio and waveform generation using the DAC in STM32 products [Text] / STMicroelectronics . — Rev. 4. — 2020. — 43 p .

[5] PM0214 Programming manual : STM32 Cortex®-M4 MCUs and MPUs programming manual [Text] / STMicroelectronics . — Rev. 10. - 2023. - 1107 p .

[6] *STM32CubeF4: Embedded software for STM32F4 series* [Software] / STMicroelectronics . — Version 1.27.1. — 2024. — URL: https://www.st.com/en/embedded-software/stm32cubef4.html (accessed: 26.07.2025).

[7] *FatFs - Generic FAT Filesystem Module* [Electronic resource] / ChaN . — URL: http://elm-chan.org/fsw/ff/00index_e.html (date of access: 07/26/2025).

[8] *USB Device Class Definition for Audio Devices* [Standard] / USB Implementers Forum . — Revision 1.0. — 1998. — 95 p