



UTM
UNIVERSITI TEKNOLOGI MALAYSIA

SEEL5123

ADVANCED MICROPROCESSOR SYSTEM

SECTION 01

MILESTONE 4

GROUP 2

GOH JUN YI	A19EE0333
SOLEHAH NAJIIHAH BINTI ABD JAMAL	A19EE0494
TEE MEI XIN	A19EE0411

LECTURER: DR. MOHD AFZAN BIN OTHMAN

1. Software/Tools setup

In this project, STM32CubeIDE is used as the development environment. Some external libraries are imported for signal processing steps, such as the DSP library. For the hardware, the Nucleo-F446RE board and INMP441 Microphone is used.

2. Configuration Steps

DMA Configuration

DMA Mode and Configuration			
Configuration			
DMA1	DMA2	MemToMem	
DMA Request	Stream	Direction	Priority
SPI2_RX	DMA1 Stream 3	Peripheral To Memory	Low
USART2_TX	DMA1 Stream 6	Memory To Peripheral	Low

USART2 Configuration

USART2 Mode and Configuration			
Mode			
Mode	Asynchronous		
Hardware Flow Control (RS232)	Disable		
Configuration			
Reset Configuration			
NVIC Settings	DMA Settings	GPIO Settings	
Parameter Settings		User Constants	
NVIC Interrupt Table	Enabled	Preemption Priority	Sub Priority
DMA1 stream6 global interrupt		0	0
USART2 global interrupt		0	0

Configuration			
Reset Configuration			
<input checked="" type="checkbox"/> NVIC Settings	<input checked="" type="checkbox"/> DMA Settings	<input checked="" type="checkbox"/> GPIO Settings	
<input checked="" type="checkbox"/> Parameter Settings		<input checked="" type="checkbox"/> User Constants	
DMA Request	Stream	Direction	Priority
USART2_TX	DMA1 Stream 6	Memory To Periphe...	Low

Configuration	
Reset Configuration	
<input checked="" type="checkbox"/> NVIC Settings	<input checked="" type="checkbox"/> DMA Settings
<input checked="" type="checkbox"/> Parameter Settings	<input checked="" type="checkbox"/> User Constants
Configure the below parameters :	
<input type="text" value="Search (Ctrl+F)"/> <input type="button" value="↺"/> <input type="button" value="↻"/> <input type="button" value="i"/>	
<div> <div>Basic Parameters</div> <div> <div>Baud Rate</div> <div>115200 Bits/s</div> </div> <div> <div>Word Length</div> <div>8 Bits (including Parity)</div> </div> <div> <div>Parity</div> <div>None</div> </div> <div> <div>Stop Bits</div> <div>1</div> </div> </div>	
<div> <div>Advanced Parameters</div> <div> <div>Data Direction</div> <div>Receive and Transmit</div> </div> <div> <div>Over Sampling</div> <div>16 Samples</div> </div> </div>	

I2S2 Configuration

I2S2 Mode and Configuration	
Mode	
Mode	Half-Duplex Master
<input type="checkbox"/> Master Clock Output	

Configuration

Reset Configuration

Parameter Settings

User Constants

NVIC Settings

DMA Settings

GPIO Settings

DMA Request	Stream	Direction	Priority
SPI2_RX	DMA1 Stream 3	Peripheral To Memory	Low

Add

Delete

DMA Request Settings

Mode	Circular	Increment Address	<input type="checkbox"/>	Peripheral	<input type="checkbox"/>	Memory	<input checked="" type="checkbox"/>
Use Fifo	<input type="checkbox"/>	Threshold		Data Width	Half Word	Half Word	
				Burst Size			

Configuration

Reset Configuration

NVIC Settings

DMA Settings

GPIO Settings

Parameter Settings

User Constants

Configure the below parameters :

Search (Ctrl+F)

Generic Parameters

Transmission Mode	Mode Master Receive
Communication Standard	I2S Philips
Data and Frame Format	16 Bits Data on 32 Bits Frame
Selected Audio Frequency	32 KHz
Real Audio Frequency	31.914 KHz
Error between Selected and Real	-0.26 %

Clock Parameters

Clock Source	I2S PLL Clock
Clock Polarity	Low

3. Steps for firmware development

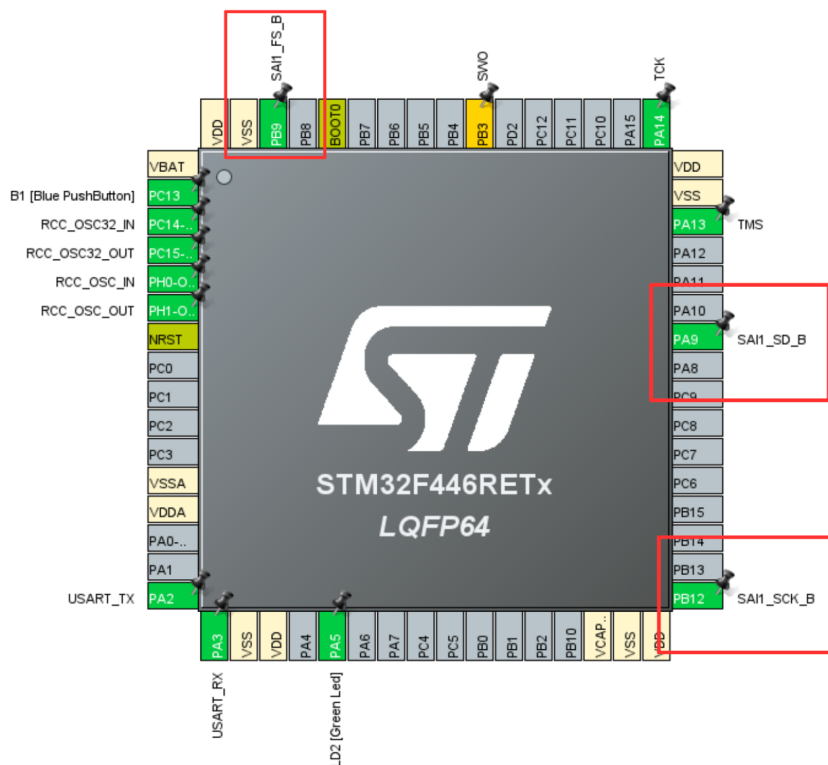
The first step of firmware development is importing required libraries. This can be done in the software packs section in pinout and configuration. The X-CUBE-ALGOBUILD package has a DSP library, which can be included in the project later.

The second step is to construct i2s data array and recording function. The input button is set to alter the recording flag. When recording is started and the i2s array is fully received, indicated by interrupt HAL_I2S_RxHalfCpltCallback and HAL_I2S_RxCpltCallback, it will be processed and save the data to the output variables.

4. Steps for hardware development

We are using Omnidirectional Microphone Module I2S Interface INMP441 MEMS to capture the audio input for our project. In order to configure the microphone with the board, STM32CubeMX is used to set up the microphone. The figure below shows the input pin of the board (in red circle) that is connected to the microphone pin.

The Serial Audio Interface (SAI) is a synchronous serial bus interface that is used for connecting digital audio devices. It is the most common means for transferring two channels of audio data across system devices. In this project, the protocol used for transferring digital audio is by using I2S based on the microphone datasheet. It used I2S Philips standard.



SAI1 Mode and Configuration

Mode

> SAI A
 < SAI B

Mode Master

☒ I2S/PCM Protocol

The SAI B is configured as Master mode which means the SAI provides the timing signals such as bit clock (SCK) and frame synchronization (FS)

Configuration

Reset Configuration

✔ Parameter Settings
✔ User Constants
✔ NVIC Settings
✔ DMA Settings
✔ GPIO Settings

Configure the below parameters :

< SAI B

Synchronization Inputs	Asynchronous
Basic Parameters	
Audio Mode	Master Receive
Output Mode	Stereo
Companding Mode	No companding mode
Protocol Parameters	
Protocol	I2S Standard
Data Size	24 Bits
Number of Slots (only Even Values)	2
Clock Parameters	
Master Clock Divider	Enabled
Audio Frequency	32 KHz
Real Audio Frequency	31.25 KHz
Error between Selected	-2.34 %
Advanced Parameters	
Fifo Threshold	Empty
Output Drive	Disabled

Audio mode is set as Master Receive as the board receives the audio input from the microphone.

DMA Mode and Configuration

Configuration

✔ DMA1
✔ DMA2
✔ MemToMem

DMA Request	Stream	Direction	Priority
SAI1_B	DMA2 Stream 4	Peripheral To Memory	Low

Direct Access Memory is added to allow the storing of audio input from the peripheral directly into the board's memory.