

---

# **AD14N\_ANS\_PATCH**

**发行版本 2025 年 3 月 4 日 - v1.1**

**JIELI AD Toys Project Group - liujie**

**2025 年 03 月 05 日**

---

## 补丁说明:

---

<b>1</b>	<b>补丁说明</b>	<b>2</b>
1.1	补丁等级说明	2
1.2	补丁等级说明	2
1.3	AD14N ANS 语音降噪补丁	2
1.4	修正记录	3
<b>2</b>	<b>ANS 语音降噪说明</b>	<b>4</b>
2.1	ANS 语音降噪算法依赖介绍	4
2.2	Step 1. 添加新增的 ANS 文件到工程中	5
2.2.1	1、拷贝所有源文件	5
2.2.2	2、添加库文件到工程配置	5
2.2.3	3、添加新的头文件目录到工程配置	6
2.2.4	4、添加新的 C 文件到工程配置	6
2.2.5	5、保存工程配置	6
2.3	Step 2. 修改源文件	7
2.3.1	app/src/mcu/sh54/app_modules.h 的修改	8
2.3.2	include_lib/encoder/encoder_mge.h 的修改	9
2.3.3	app/bsp/common/encoder/encoder_api.c 的修改	10
2.3.4	app/post_build/sh54/voice_toy/app_ld.c 的修改	11

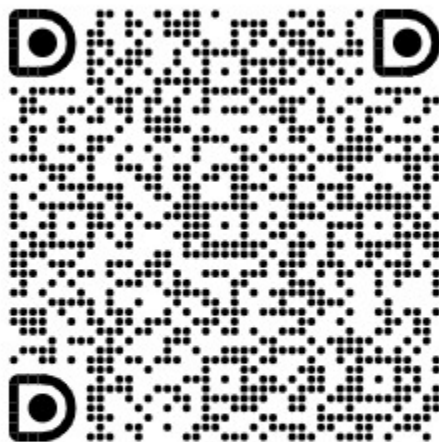
开源社区仓库链接:

<https://gitee.com/Jieli-Tech/fw-AD15N>

<https://github.com/Jieli-Tech/fw-AD15N>

开源社区交流群:

AD系列32位芯片 杰理开源社区 



此二维码365天内有效 (2025-10-08前)

 钉钉扫一扫群二维码, 立即加入群聊

开源社区钉钉群二维码

作者: 刘杰

### 1.1 补丁等级说明

A 类：问题修复，客户必须修复

B 类：性能优化，客户可选择修

C 类：功能增加，客户可选择

### 1.2 补丁等级说明

- A 类：问题修复，客户必须修复
- B 类：性能优化，客户可选择修
- C 类：功能增加，客户可选择

### 1.3 AD14N ANS 语音降噪补丁

日期：2025 年 3 月 4 日

作用 SDK 版本：AD1x-4578\_AC104-release\_v1.8.0

补丁等级：C

## 1.4 修正记录

2025 年 3 月 5 日

- 1、添加内容“app/bsp/common/sound\_effect\_list/ans\_api.c”文件配置进工程
  - 2、完善遗漏的 ld.c 部分修改说明
-

---

## ANS 语音降噪说明

---

ANS 语音降噪，该功能用于将 16 位语音信号进行降噪处理。本章节会介绍如何将 ANS 降噪算法添加到自己的 AD14N 工程中。

**注：**后续一些小节对原有文件的修改，仅仅针对添加 ANS 而言，客户的源代码文件可能会存在差异，在修改时有变化的源代码文件时，应该仅仅对图上所示的变化进行修改，而不是对整个文件进行替换。

### 2.1 ANS 语音降噪算法依赖介绍

ANS 语音降噪，该功能用于将 16 位语音信号进行降噪处理。

#### 1、新增库文件

```
include_lib/liba/ARCH/pi32_lto/libNoiseSuppress_pi32_OnChip.a
include_lib/liba/ARCH/pi32_lto/lib_SW_FFT_pi32_OnChip.a
```

#### 2、新增头文件

```
include_lib/ans/NoiseSuppressLib.h
include_lib/ans/ans_api.h
```

#### 3、变化的头文件

```
app/src/voice_toy/sh54/app_modules.h
include_lib/encoder/encoder_mge.h
```

#### 4、新增的 C 文件。

```
app/bsp/common/sound_effect_list/ans_api.c
```

#### 5、变化的 C 文件。

```
app/bsp/common/encoder/encoder_api.c
app/post_build/sh54/voice_toy/app_ld.c
```

## 2.2 Step 1. 添加新增的 ANS 文件到工程中

### 2.2.1 1、拷贝所有源文件

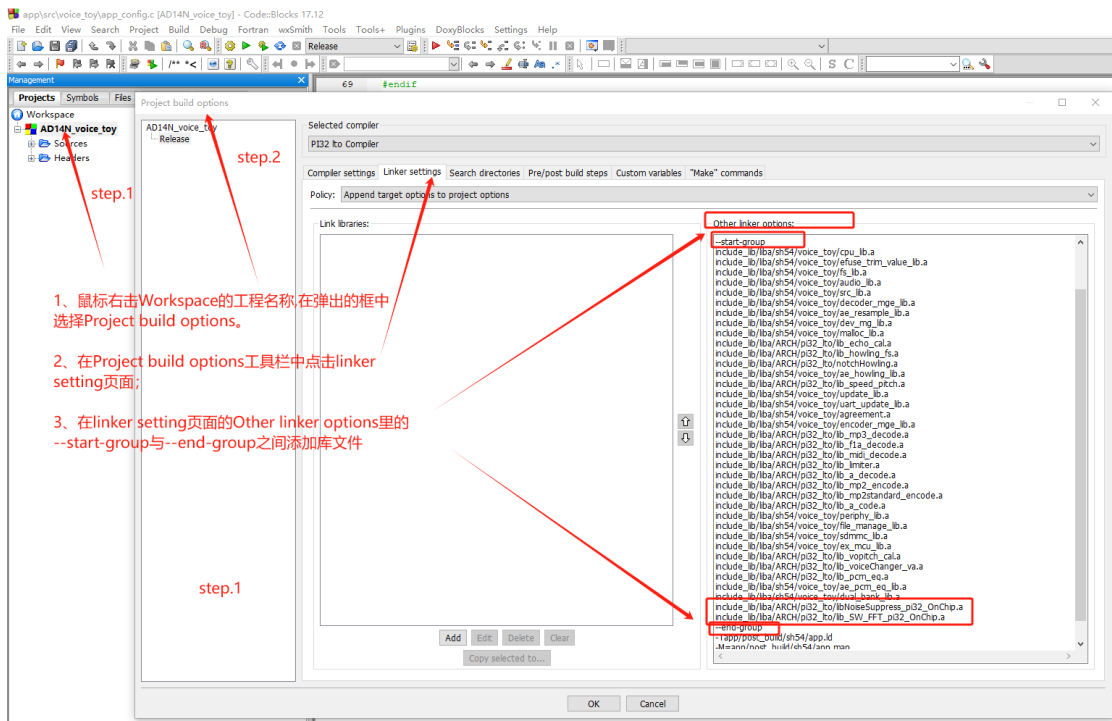
将以下文件从补丁包中拷贝到目标工程相应的目录中

```
include_lib/liba/ARCH/pi32_lto/libNoiseSuppress_pi32_OnChip.a
include_lib/liba/ARCH/pi32_lto/lib_SW_FFT_pi32_OnChip.a
include_lib/ans/NoiseSuppressLib.h
include_lib/ans/ans_api.h
app/bsp/common/sound_effect_list/ans_api.c
```

### 2.2.2 2、添加库文件到工程配置

按照以下步骤或图示将库文件添加到工程配置中：

- 1、鼠标右击 Workspace 的工程名称, 在弹出的框中选择 Project build options;
- 2、在 Project build options 工具栏中点击 linkersetting 页面;
- 3、在 linker Setting 页面的 Other linker options 里的--start-group 与--end-group 之间添加库文件。

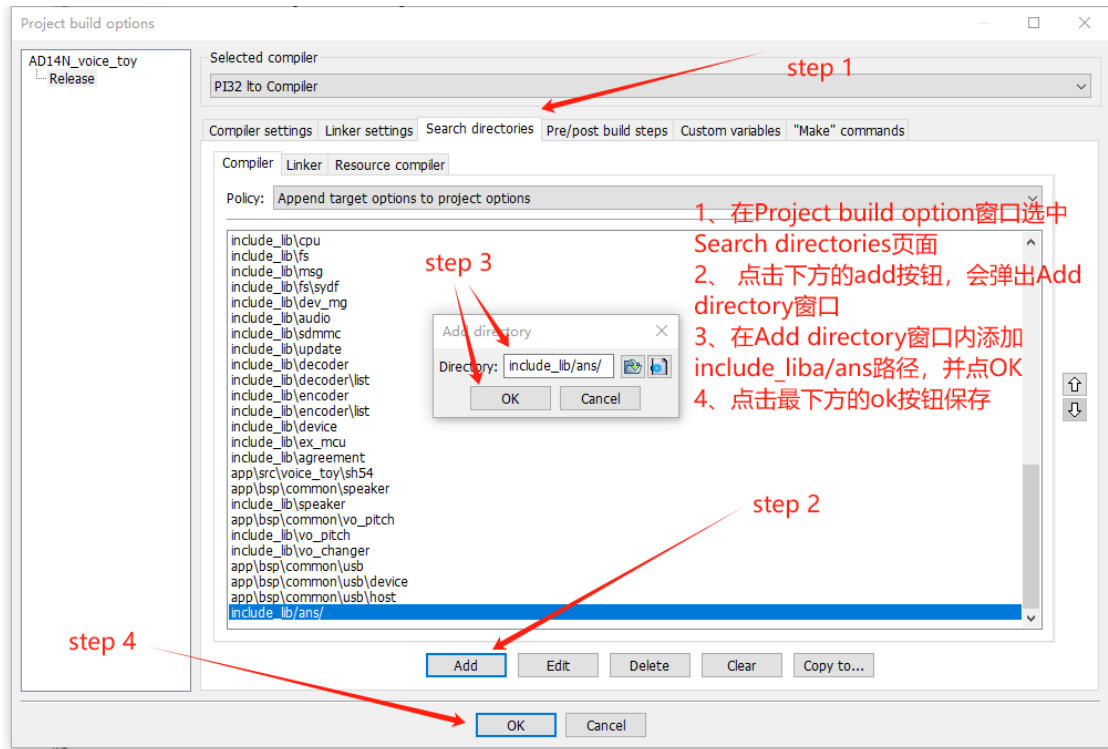


添加库文件到工程配置

### 2.2.3 3、添加新的头文件目录到工程配置

在上一步打开的 Project build options 中，按照以下步骤或图示添加新的头文件目录到工程配置：

- 1、在 Project build option 窗口选中 Search directories 页面；
- 2、点击下方的 add 按钮，会弹出 Add directory 窗口；
- 3、在 Add directory 窗口内添加 include liba/ans 路径，并点 OK；
- 4、点击最下方的 ok 按钮保存；



添加新的头文件目录到工程配置

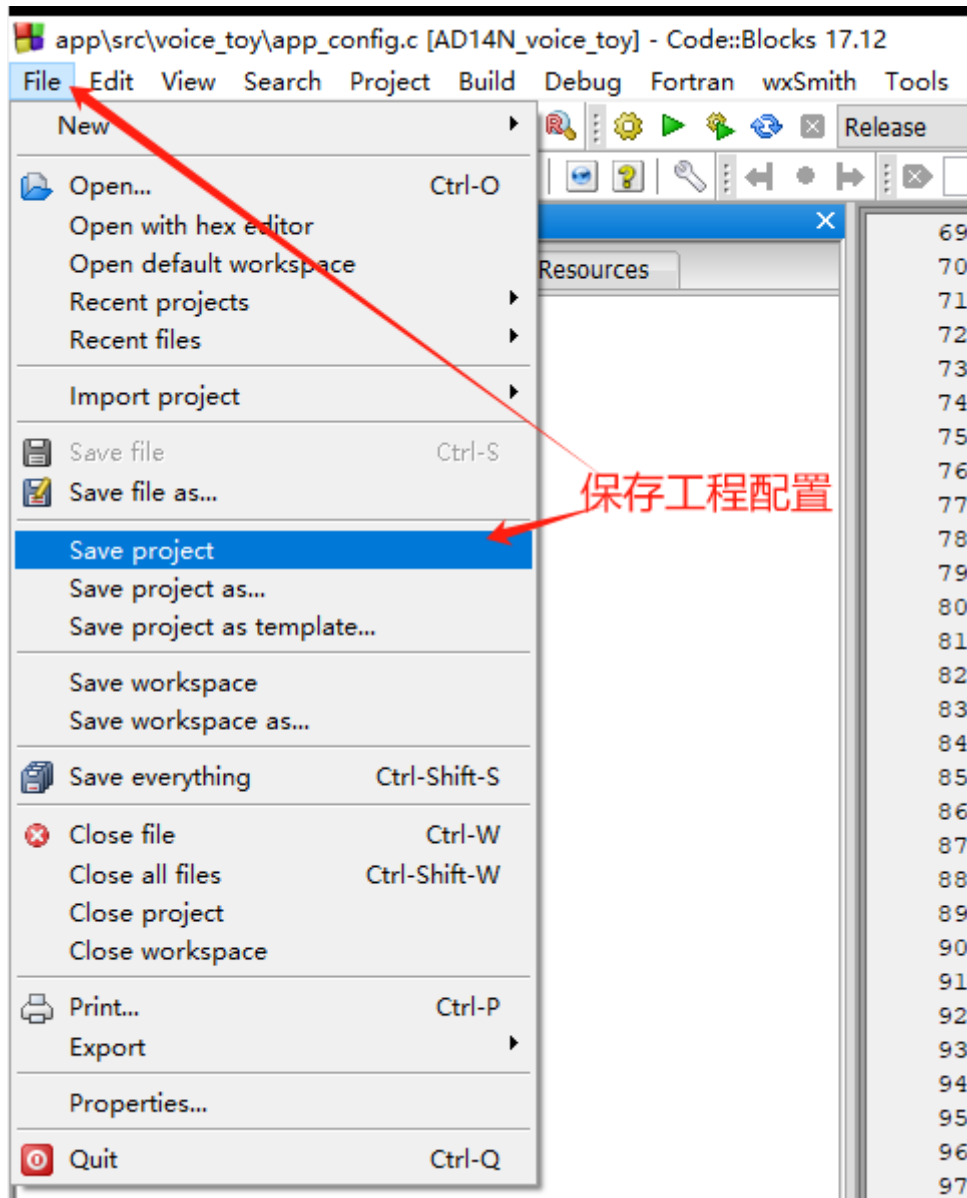
### 2.2.4 4、添加新的 C 文件到工程配置

- 1、鼠标右击 Workspace 的工程名称，在弹出的框中选择 “Add files…”；
- 2、选中文件 “app/bsp/common/sound\_effect\_list/ans\_api.c”添加进工程参与编译；

### 2.2.5 5、保存工程配置

经过上面两步骤在 Project build options 的修改后，需要如下图所示点击” File->Save project” 结果：



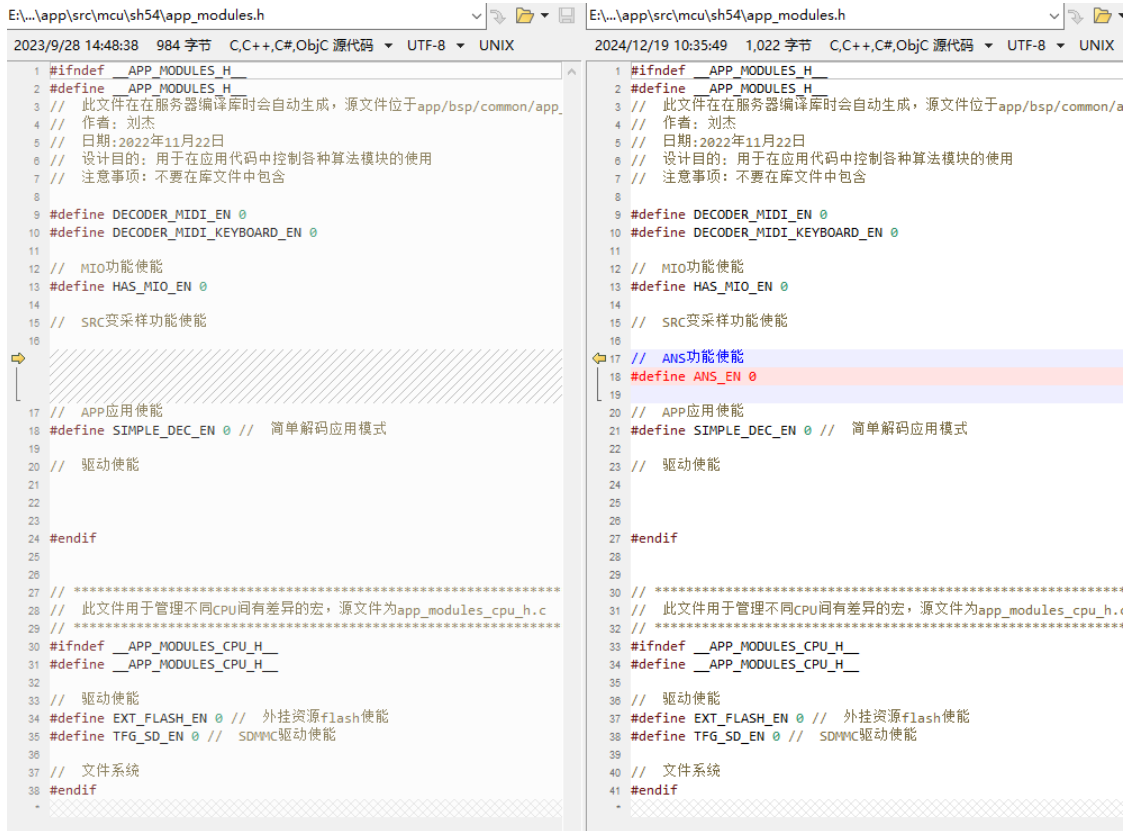


保存工程配置

## 2.3 Step 2. 修改源文件

本小节所有修改都基于对 AD1x-4578\_AC104-release\_v1.8.0 的比较。

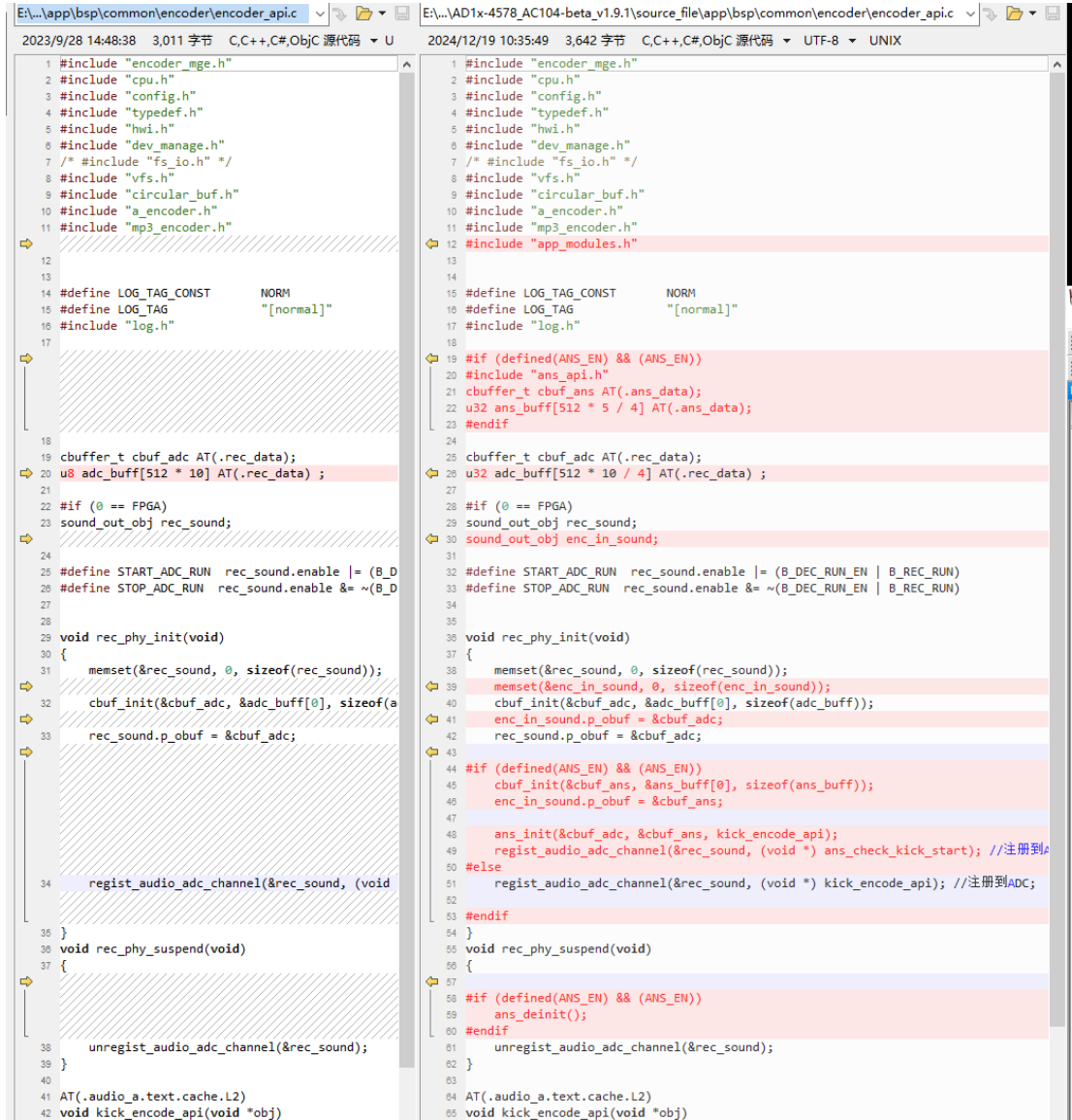
## 2.3.1 app/src/mcu/sh54/app\_modules.h 的修改



app\_modules.h 的修改



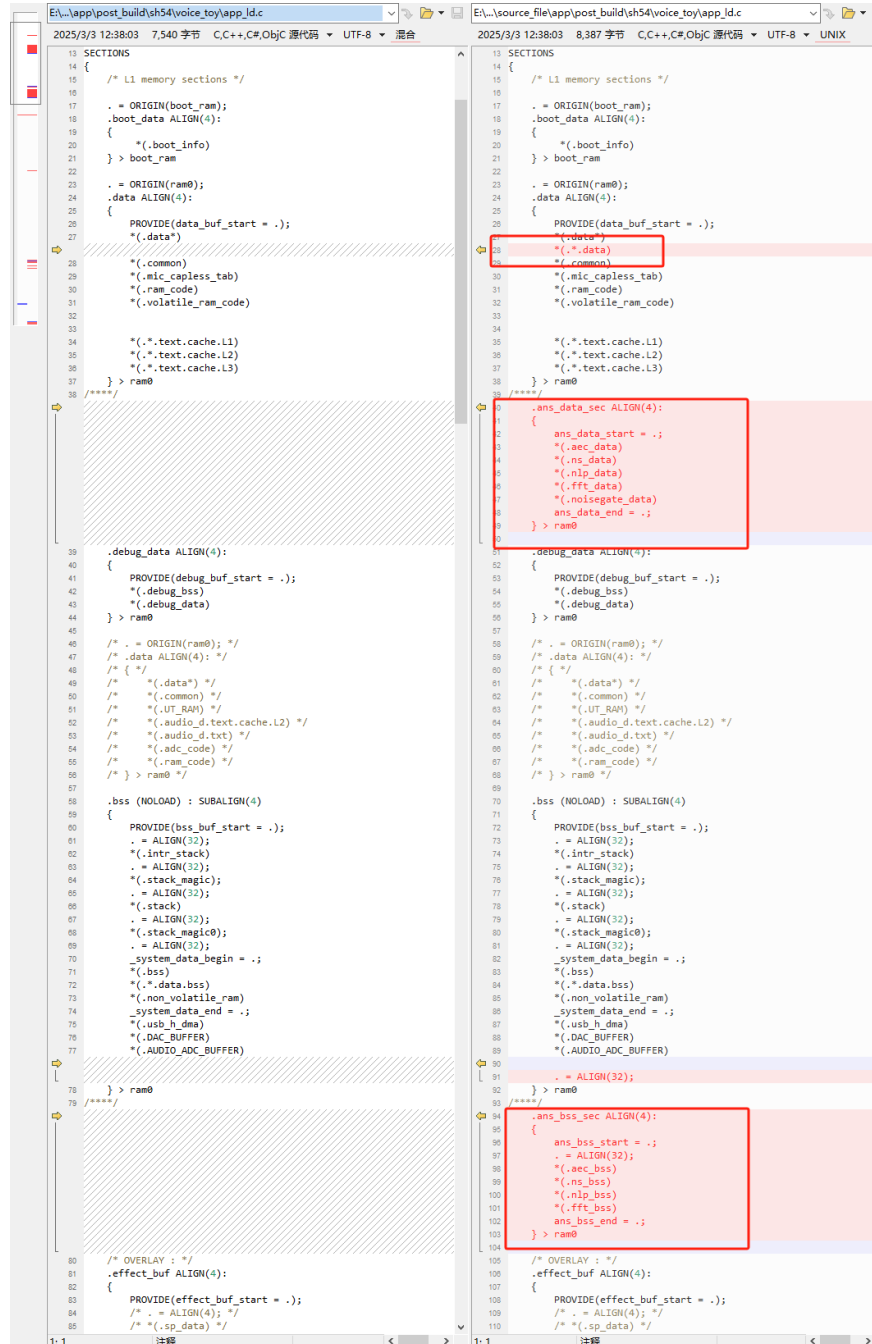
### 2.3.3 app/bsp/common/encoder/encoder\_api.c 的修改



encoder\_api.c 的修改

## 2.3.4 app/post\_build/sh54/voice\_toy/app\_ld.c 的修改

注：本小节介绍仅仅针对添加 ANS 而言，客户的 ld 源文件可能会存在差异，此处修改应该仅对图上所示的增量进行修改，而不是整个替换 app\_ld.c。



```

99  /* . = ORIGIN(ram1); */
100 OVERLAY : AT(0x200000)
101 {
102     .d_toy_music
103     {
104         PROVIDE(toy_music_buf_start = .);
105         *(.toy_music_data);
106         PROVIDE(toy_music_buf_end = .);
107     }
108     d_speed
109     {
110         . = toy_music_buf_end;
111         PROVIDE(speed_buf_start = .);
112         . = ALIGN(4);
113         *(.sp_data)
114         PROVIDE(speed_buf_end = .);
115     }
116     .d_a
117     {
118         . = speed_buf_end;
119         PROVIDE(a_buf_start = .);
120         *(.a_data);
121         PROVIDE(a_buf_end = .);
122     }
123
124     .d_midi
125     {
126         . = a_buf_end;
127         PROVIDE(midi_buf_start = .);
128         *(.midi_buf);
129         PROVIDE(midi_buf_end = .);
130         PROVIDE(midi_ctrl_buf_start = .);
131         *(.midi_ctrl_buf);
132         PROVIDE(midi_ctrl_buf_end = .);
133     }
134     .d_ump3
135     {
136         . = a_buf_end;
137         PROVIDE(ump3_buf_start = .);
138         *(.ump3_data);
139         PROVIDE(ump3_buf_end = .);
140     }
141     .d_mp3_st
142     {
143         . = a_buf_end;
144         PROVIDE(mp3_st_buf_start = .);
145         *(.mp3_st_data);
146         PROVIDE(mp3_st_buf_end = .);
147     }
148     .d_wav
149     {
150         . = a_buf_end;
151         PROVIDE(wav_buf_start = .);
152         *(.wav_data);
153         PROVIDE(wav_buf_end = .);
154     }
155     .d_fla
156     {
157         . = a_buf_end;
158         PROVIDE(fla_1_buf_start = .);
159         *(.fla_1_buf);
160         PROVIDE(fla_1_buf_end = .);
161         PROVIDE(fla_2_buf_start = .);
162         *(.fla_2_buf);
163         PROVIDE(fla_2_buf_end = .);
164     }
165     .d_rec
166     {
167         *(.rec_data)
168         rec_data_end = .;
169     }
170     .d_enc_ima
171     {
172         . = rec_data_end;
173         *(.enc_a_data)
174     }
175     .d_enc_mp3
176     {
177
124  /* . = ORIGIN(ram1); */
125  OVERLAY : AT(0x4000000)
126  {
127      .d_toy_music
128      {
129          PROVIDE(toy_music_buf_start = .);
130          *(.toy_music_data);
131          PROVIDE(toy_music_buf_end = .);
132      }
133      d_speed
134      {
135          . = toy_music_buf_end;
136          PROVIDE(speed_buf_start = .);
137          . = ALIGN(4);
138          *(.sp_data)
139          PROVIDE(speed_buf_end = .);
140      }
141      .d_a
142      {
143          . = speed_buf_end;
144          PROVIDE(a_buf_start = .);
145          *(.a_data);
146          PROVIDE(a_buf_end = .);
147      }
148
149      .d_midi
150      {
151          . = a_buf_end;
152          PROVIDE(midi_buf_start = .);
153          *(.midi_buf);
154          PROVIDE(midi_buf_end = .);
155          PROVIDE(midi_ctrl_buf_start = .);
156          *(.midi_ctrl_buf);
157          PROVIDE(midi_ctrl_buf_end = .);
158      }
159      .d_ump3
160      {
161          . = a_buf_end;
162          PROVIDE(ump3_buf_start = .);
163          *(.ump3_data);
164          PROVIDE(ump3_buf_end = .);
165      }
166      .d_mp3_st
167      {
168          . = a_buf_end;
169          PROVIDE(mp3_st_buf_start = .);
170          *(.mp3_st_data);
171          PROVIDE(mp3_st_buf_end = .);
172      }
173      .d_wav
174      {
175          . = a_buf_end;
176          PROVIDE(wav_buf_start = .);
177          *(.wav_data);
178          PROVIDE(wav_buf_end = .);
179      }
180      .d_fla
181      {
182          . = a_buf_end;
183          PROVIDE(fla_1_buf_start = .);
184          *(.fla_1_buf);
185          PROVIDE(fla_1_buf_end = .);
186          PROVIDE(fla_2_buf_start = .);
187          *(.fla_2_buf);
188          PROVIDE(fla_2_buf_end = .);
189      }
190      .d_rec
191      {
192          *(.rec_data)
193          *(.ans_data)
194          rec_data_end = .;
195      }
196      .d_enc_ima
197      {
198          . = rec_data_end;
199          *(.enc_a_data)
200      }
201      .d_enc_mp3
202      {

```

app\_ld.c 的修改之二

<pre> 261 262 263     . = ORIGIN(app_code); 264     .app_code ALIGN(32): 265     { 266         *startup.o(.text) 267         . = ALIGN(32); 268         *(.fla_code) 269         _VERSION_BEGIN = .; 270         KEEP(*(.version)) 271         _VERSION_END = .; 272         *(.debug) 273         *(.debug_const) 274         *(.debug_code) 275         *(.debug_string) 276 277         /* *memset.o(.text .rodata*) */ 278         /* *memcpy.o(.text .rodata*) */ 279         *(*.text.const) 280 281         *(*.text) 282         *(*.text) 283 284         *(.app_root) 285         *(.vm) 286         . = ALIGN(32); 287         _SPI_CODE_START = .; 288         *(.sni_code) </pre>	<pre> 287 288 289     . = ORIGIN(app_code); 290     .app_code ALIGN(32): 291     { 292         *startup.o(.text) 293         . = ALIGN(32); 294         *(.fla_code) 295         _VERSION_BEGIN = .; 296         KEEP(*(.version)) 297         _VERSION_END = .; 298         *(.debug) 299         *(.debug_const) 300         *(.debug_code) 301         *(.debug_string) 302         . = ALIGN(4); 303 304         *(.fft_const) 305         *(.fft_code) 306         /* *memset.o(.text .rodata*) */ 307         /* *memcpy.o(.text .rodata*) */ 308         *(*.text.const) 309         *(*.text.const) 310         *(*.text) 311         *(*.text) 312         *(*.text.*) 313         *(.app_root) 314         *(.vm) 315         . = ALIGN(32); 316         _SPI_CODE_START = .; 317         *(.sni_code) </pre>
--	---

app\_ld.c 的修改之三