



VB 语音模块 定制词条流程

版本 V1.0

版权 ©2021

免责声明和版权公告

本文中的信息，包括供参考的 URL 地址，如有变更，恕不另行通知。

文档“按现状”提供，不负任何担保责任，包括对适销性、适用于特定用途或非侵权性的任何担保，和任何提案、规格或样品在他处提到的任何担保。本文档不负任何责任，包括使用本文档内信息产生的侵犯任何专利权行为的责任。本文档在此未以禁止反言或其他方式授予任何知识产权使用许可，不管是明示许可还是暗示许可。

文中所得测试数据均为安信可实验室测试所得，实际结果可能略有差异。

文中提到的所有商标名称、商标和注册商标均属其各自所有者的财产，特此声明。

最终解释权归深圳市安信可科技有限公司所有。

注意

由于产品版本升级或其他原因，本手册内容有可能变更。深圳市安信可科技有限公司保留在没有任何通知或者提示的情况下对本手册的内容进行修改的权利。本手册仅作为使用指导，深圳市安信可科技有限公司尽全力在本手册中提供准确的信息，但是深圳市安信可科技有限公司并不确保手册内容完全没有错误，本手册中的所有陈述、信息和建议也不构成任何明示或暗示的担保。

目录

一、软件准备.....	4
二、词条静态库定制.....	4
三、CDK 修改.....	6
四、提示音制作.....	7
五、固件烧录.....	10
六、串口控制播放提示音.....	11
七、播放音量调节.....	12
联系我们.....	12

Ai-Thinker

一、软件准备

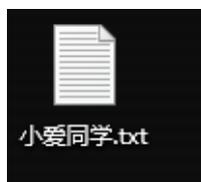
1. 编程软件：CDK
2. 固件烧录工具：CSKYFlashProgrammer.exe
3. 提示音装换工具：WnvcEncoderTool_200922_V1.4.exe

注意：这三个工具的所在路径不能有中文，

4. 文字转语音工具：推荐 迅捷文字转语音工具

二、词条静态库定制

1. 创建词条文件，格式 txt，如：



2. 在该文件中输入需要定制的唤醒词和命令词，如：

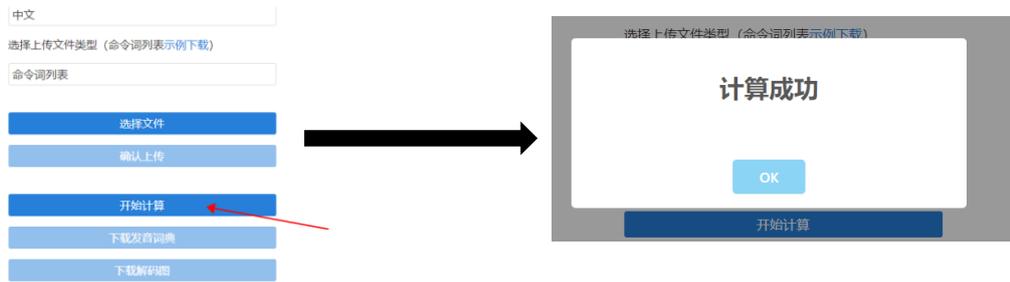


3. 编辑好之后，在浏览中打开[华镇电子 AI 语音识别在线训练平台](http://47.111.14.84/)：

<http://47.111.14.84/>注册好账号之后登录，然后上传刚刚编辑好的词条文件，如下：



4. 上传成功之后点击“开始计算”，计算成功之后点击“下载解码图”，如果需要做优化的话，就下载发音词典。

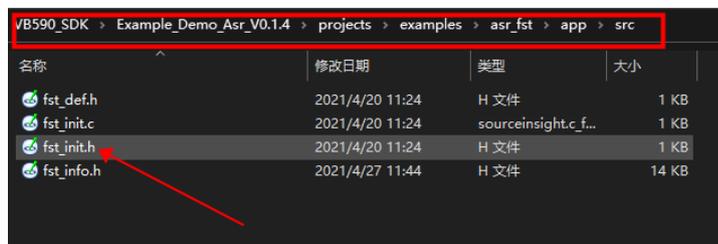


说明：
 下载发音词典是一个名为：**lex** 且无后缀的文件；
 下载解码图则是一个.h 文件：**fst_info.h**



5. 然后把 **fst.info.h** 文件复制到：

VB590_SDK→**Example_Demo_Asr_V0.1.4**→**projects**→**examples**→**asr_fst**→**app**→**src**



6. 回到“**asr_fst**”目录下，进入“**CDK**”文件夹中点击“**libasrfst.cdkws**”文件（需要安装好 CDK 软件），打开工程，选择工程名字，鼠标右键选择“**Build All**”，如下：



然后查看编译结果，没有错误后，会在“**libasrfst**”文件夹的“**Obj**”下生成一个静态库：**libasrfst.a**

```

fsts->final_states = final_states;
^
H:/VB590_SDK/Example_Demo_Asr_V0.1.4/projects/examples/asr_fst/app/src/fst_init.c:11:15: warning: assignment from incompatible pointer type
fsts->words = words;
^
sky-elfabiv2-ar rcu Obj/libasrfst.a @"libasrfst.txt"
make[1]: Leaving directory 'h:/VB590_SDK/Example_Demo_Asr_V0.1.4/projects/examples/asr_fst/CDK/libasrfst'
===0 errors, 4 warnings, total time : 35773ms===

```

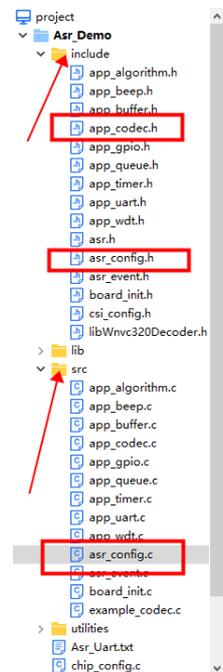
最后把“**libasrfst.a**”静态库文件复制到，Asr_Deme 文件夹的“**lib**”文件夹中，路径如下：
VB590_SDK → Example_Demo_Asr_V0.1.4 → projects → examples → **Asr_Demo** → lib

三、CDK 修改

完成语音词条静态库的定制之后，还需要修改 Demo 的一点代码才算是完成词条定制，我们说出的指令才能被 VB 语音模块识别。

1. 需要修改的几个文件：

- (1) asr_config.c : 词条定义文件
- (2) asr_config.h : 词条配置数量等配置文件
- (3) app_codec.h : 主要修改音量配置



2. 修改唤醒词：**asr_config.c**

在 CKD 中打开 asr_config.c 文件，把“小蜂管家”和“小蜂小蜂”改成和我们的词条文件一样的指令，（注意：两者不能有一个词的区别），如：

```

/**
unsigned char wakeup_cmd_data[ASR_WAKE_INDEX][ASR_WAKE_LEN] =
{
    {"小蜂管家"},
    {"小蜂小蜂"},
};
*/

```

修改前

```

1. 将所需要的唤醒词填入改数组中；
2. 所填入的唤醒词必须在数组 asr_cmd_data 中存在，且识别指令库中存在该指令（即：能够出识
3. 所填入的指令，如果工作区间是 utf-8 编码格式，则可以直接填入汉字，不然需要填入汉字对应的
4. 根据现有宏命令 ASR_WAKE_INDEX/ASR_WAKE_LEN 填入指令，且每个指令长度不能超过 10 个且每个唤醒
*/
unsigned char wakeup_cmd_data[ASR_WAKE_INDEX][ASR_WAKE_LEN] =
{
    {"小爱同学"},
    {"你好小艾"},
};
/**
因为“艾”不一样，识别失败
1. 将所有需要识别表的指令填入改数组中；
2. 所填入的识别指令词必须在识别指令库中存在；
3. 所填入的指令，如果工作区间是 utf-8 编码格式，则可以直接填入汉字，不然需要填入汉字对应的
4. 根据现有宏命令 ASR_WAKE_INDEX/ASR_WAKE_LEN 填入指令，且每个指令长度不能超过 10 个且每个唤醒
*/

```


2. 重命名 wav 文件:

把已经生成的语音文件重名为: 00.wav、01.wav、02.wav....等排序名。

说明: 00.wav 为开机提示音或复位提示音。

02.wav 为退下提示音,

01.wav 一般作为唤醒词的提示音

其他: 就是命令词提示音了



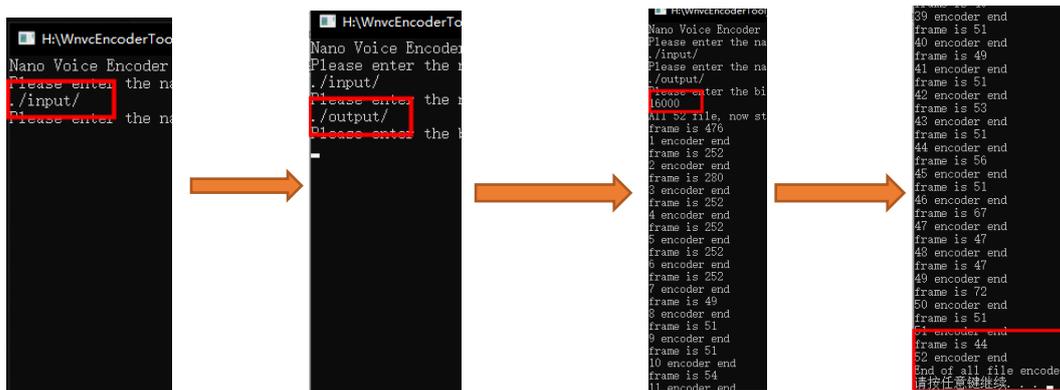
然后把这些已经重命名的语音文件复制并替换, WnvcEncoderTool_200922_V1.4 文件夹下的“input”内的文件, 其路径:

VB590_Tools → WnvcEncoderTool_200922_V1.4 → input

3. 把语音文件转换成语音库

到: VB590_Tools → WnvcEncoderTool_200922_V1.4 目录下, 双击“WnvcEncoderTool_200922_V1.4.exe”,

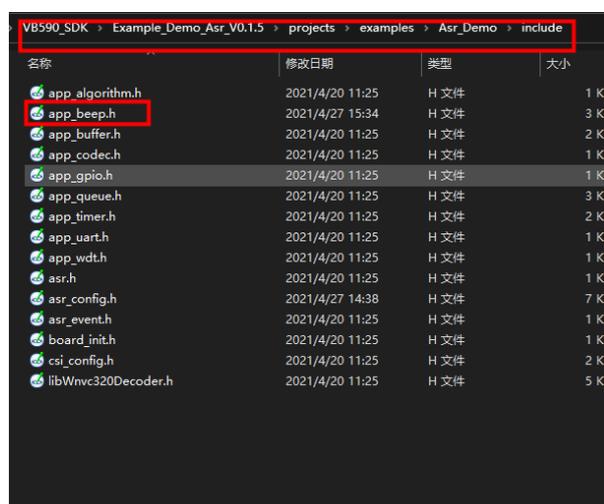
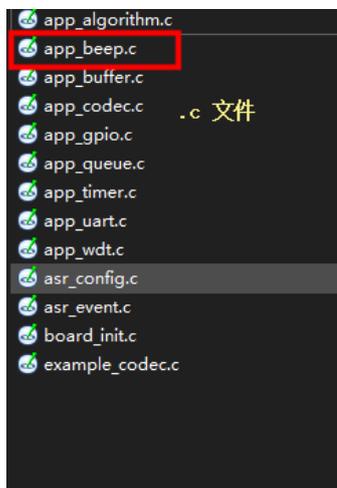
然后在面板中输入: ./input/ → 回车 → ./output/ → 回车 → 16000 → 回车。输入不要漏掉“/”



经过以上流程后, 正常情况下会在“output”文件夹中生成两个文件: app_beep.c 和 app_beep.h。

把 app_beep.c 复制到: VB590_SDK\Example_Demo_Asr_V0.1.4\projects\examples\Asr_Demo\src

把 app_beep.h 复制到: VB590_SDK\Example_Demo_Asr_V0.1.4\projects\examples\Asr_Demo\include

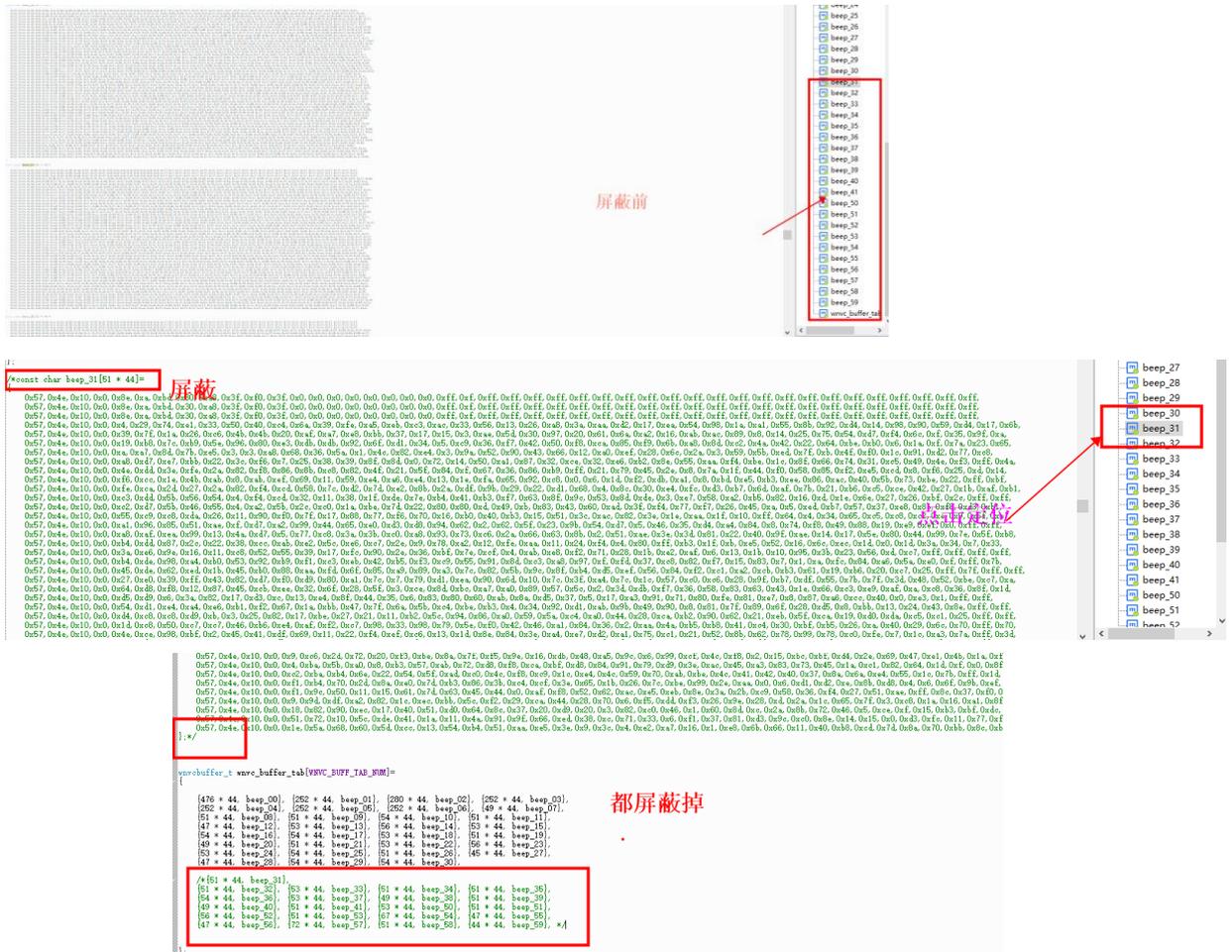


完成之后重新编译 CDK，等待编译完成之后，会在 **Asr_Demo** 工程目录下的“Obj”文件夹下生成：**Asr_Demo.elf** 固件。路径：**VB590_SDK\Example_Demo_Asr_V0.1.5\projects\examples\Asr_Demo\Obj**

复制语音文件后可能出现的问题。如下：

```
linking...
d:/c-sky/cdk/csky/mingw/csky-abi2-elf-toolchain/bin/./lib/gcc/csky-elfabi2/6.3.0/../../../../
d:/c-sky/cdk/csky/mingw/csky-abi2-elf-toolchain/bin/./lib/gcc/csky-elfabi2/6.3.0/../../../../
d:/c-sky/cdk/csky/mingw/csky-abi2-elf-toolchain/bin/./lib/gcc/csky-elfabi2/6.3.0/../../../../
d:/c-sky/cdk/csky/mingw/csky-abi2-elf-toolchain/bin/./lib/gcc/csky-elfabi2/6.3.0/../../../../
collect2.exe: error: ld returned 1 exit status
make[1]: *** [Obj/Asr_Demo] Error 1
Asr_Demo.mk:86: recipe for target 'Obj/Asr_Demo' failed
make[1]: Leaving directory 'H:/VB590_SDK/Example_Demo_Asr_V0.1.5/projects/examples/Asr_Demo'
make: *** [All] Error 2
Makefile:4: recipe for target 'All' failed
====1 errors, 0 warnings, total time : 21s909ms====
```

那是因为语音数组占内存空间太大了，VB059 的 flash 装不下，**解决办法**：在 CDK 中打开 **app_deep.c** 文件，把一些不要的提示音数组给屏蔽掉就行了。如下：



屏蔽之后，这些数据不占内存，就可以编译通过了。

五、固件烧录

1. 打开 VB590_Tools 文件夹，把“danica_qflash.elf”文件复制到：FlashProgrammer\VB590-V1.0.8 目录下：



回到“FlashProgrammer”文件夹下双击“CSKYFlashProgrammer.exe”，这样就能打开固件烧录软件，如下：



Please wait...
Erase data...

烧录中

Cancel

Information

Success

烧录成功

OK

Error

DebugServer Start error, can't connect to target.

遇到这个请先检查接线，然后拔插下载器

OK

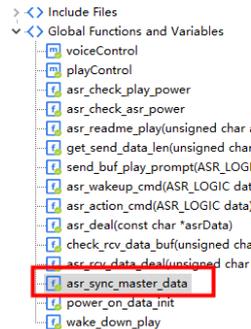
六、串口控制播放提示音

我所拿到手的 Demo 中，只能用 UART1 去控制提示音（出厂固件不会有这个问题），但是在使用 USB 供电的时候就无法使用 UART1，这时，就要使用 UART0 去控制，而且 VB 模块的串口输出也是从 UART0 进行输出，这时就需要去添加串口数据的来源。

修改方法：

1. 在 CDK 中打开“asr_event.c”，定位到“asr_sync_master_data”

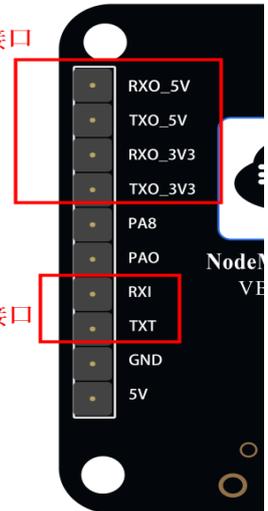
函数：



点一下就能定位

UART0 接口

UART1 接口



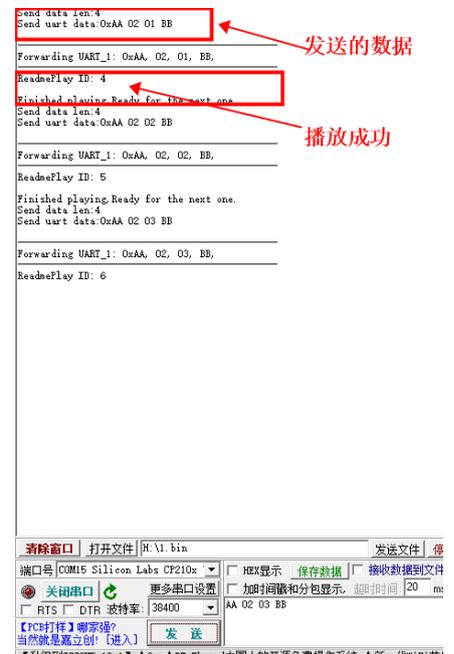
2. 复制“asr_rcv_data_deal(pbyte, bufLen);”粘贴到这个函数的结尾，如图：

```
g.c  app_algorithm.c  example_codec.c  asr_event.c  app_queue.c  app_uart.c
\brief Process received data.
\return NONE.
*/
void asr_sync_master_data()
{
    unsigned char *pbyte = NULL;
    unsigned char buflen = 0;

    pbyte = app_uart_get_rx_data(&buflen); //获取uart1接收数据
    if (pbyte)
    {
        //接收到数据
        asr_rcv_data_deal(pbyte, buflen); //处理接收到的数据，比如说要播放提示音等
    }
    pbyte = asr_rcv_log_uart_data_get(&buflen); //获取uart0接收数据
    if (pbyte)
    {
        //接收到数据
        app_uart_tx(pbyte, buflen); //将回去的uart0 rx数据通过uart1 tx发送出去
    }
    #if APP_LOG_ENABLE
        unsigned int i = 0;
        printf("\n-----\n");
        printf("Forwarding UART_1: 0x");
        for (i = 0; i < buflen; i++)
        {
            printf("%02X, ", pbyte[i]);
        }
        printf("\n-----\n");
    #endif
    asr_rcv_data_deal(pbyte, buflen);
}
```

1. 复制这一句

2. 粘贴到这里，位置别错了是在 #endif 的下方



七、播放音量调节

在 CDK 中打开“[app_codec.h](#)”，找到“`#define CODEC_OUT_MAX_VOLUME_DB`”宏定义，修改后面的 16 进制数即可修改音量：

最小音量：0x27

最大音量：0x3F

```
app_algorithm.c  example_codec.c  asr_event.c  app_queue.c  app_uart.c  app_codec.h X
#ifndef __APP_CODEC_H__
#define __APP_CODEC_H__

#include <stdint.h>
#include <stdbool.h>

#define CODEC_TX_FRAME_BYTE_SIZE      1280//20ms ,@16k 16bits, stereo
#define CODEC_RX_FRAME_BYTE_SIZE      1920//30ms ,@16k 16bits, stereo

/*
 * 0x2f: 幅值2.62V
 * 0x2B: 幅值1.86V
 * 0x29: 幅值1.54V (客户测得1.1V)
 * 0x27: 幅值1.28V
 */
#define CODEC_OUT_MAX_VOLUME_DB      0x3A// 0dB
// N*0.75 -35.25 (dB)  修改这个值
```

联系我们

官方官网：<https://www.ai-thinker.com>

开发 DOCS：<https://docs.ai-thinker.com>

官方论坛：<http://bbs.ai-thinker.com>

样品购买：<https://anxinke.taobao.com>

商务合作：sales@aithinker.com

技术支持：support@aithinker.com

公司地址：深圳市宝安区西乡固戍华丰智慧创新港 C 栋 410

联系电话：0755-29162996

