10-1、I2S CODEC MAX98357

1. 参考《0-1》文档，插入ESP32模块。
2. 用导线连接IO口与I2S接口，并插上喇叭，如下图所示：

红色线<------->DIN

蓝色线<------->BCLK

绿色线<------->LRCLK

MX1.25插座<------->8Ω/2W喇叭



喇叭

1. 代码示例

#main.py

from machine import I2S

from machine import Pin

"""

GPIO13 -- DIN

GPIO12 --- BCLK

GPIO14 -- LRC

GND -- GND

5V或3.3V -- VCC

"""

# 初始化引脚定义

sck\_pin = Pin(12) # 串行时钟输出

ws\_pin = Pin(14) # 字时钟

sd\_pin = Pin(13) # 串行数据输出

"""

sck 是串行时钟线的引脚对象

ws 是单词选择行的引脚对象

sd 是串行数据线的引脚对象

mode 指定接收或发送

bits 指定样本大小（位），16 或 32

format 指定通道格式，STEREO（左右声道） 或 MONO(单声道)

rate 指定音频采样率（样本/秒）

ibuf 指定内部缓冲区长度（字节）

"""

# 初始化i2s

audio\_out = I2S(1, sck=sck\_pin, ws=ws\_pin, sd=sd\_pin, mode=I2S.TX, bits=16, format=I2S.STEREO, rate=44100, ibuf=20000)

wavtempfile = "test.wav"

with open(wavtempfile,'rb') as f:

# 跳过文件的开头的44个字节，直到数据段的第1个字节

pos = f.seek(44)

# 用于减少while循环中堆分配的内存视图

wav\_samples = bytearray(1024)

wav\_samples\_mv = memoryview(wav\_samples)

print("start playing...")

#并将其写入I2S DAC

while True:

try:

num\_read = f.readinto(wav\_samples\_mv)

# WAV文件结束

if num\_read == 0:

break

# 直到所有样本都写入I2S外围设备

num\_written = 0

while num\_written < num\_read:

num\_written += audio\_out.write(wav\_samples\_mv[num\_written:num\_read])

except Exception as ret:

print("exception...", ret)

break

1. 上传main.py和test.wav文件到模块中，按F5或点击运行按钮，可以听到喇叭播放出来的声音，如想使用其他音频文件请注意文件格式及文件大小。



