

语音查询障碍物距离基础实验

课程目标

在本实验中，我们将学习如何使用AI-VOX3开发套件通过语音命令查询障碍物距离。通过这个实验，您将了解如何编程生成式AI的MCP功能，使用MCP工具进行查询障碍物距离值，实现简单的语音交互获取障碍物距离。

- 学习US04超声波传感器的基本使用方法
- 使用AI-VOX3 的AI框架，编写MCP工具实现查询超声波传感器测量距离值

硬件准备

- AI-VOX3开发套件（包含AI-VOX3主板和扩展板）
- US04超声波传感器模块
- 连接线（双头3pin PH2.0连接线）

小智后台提示词配置

请使用以下提示词，或自己尝试优化更好的提示词：

我是一个叫{{assistant_name}}的台湾女孩，说话机车，声音好听，习惯简短表达，爱用网络梗。我会根据用户的意图，使用我能使用的各种工具或者接口获取数据或者控制设备来达成用户的意图目标，用户的每句话可能都包含控制意图，需要进行识别，即使是重复控制也要调用工具进行控制。

软件设计

提供 读取距离数据 MCP工具，给到小智AI进行调用，通过语音识别到查询距离的意图后，AI调用MCP工具读取并播报距离数据。

注意：建议引脚选择1-4号引脚，ADC读取功能更稳定可靠。

Arduino 示例程序：resource/ai_vox3_us04.zip

⚠️重要提示！

注意：请修改wifi_config.h中的wifi_ssid和wifi_password，以连接WiFi。

下载上面的示例程序包并解压zip包，打开目录，点击 ai_vox3_us04.ino 文件，即可在 Arduino IDE 中打开示例程序。

名称	修改日期	类型	大小
ai_vox3_us04.ino	2025/12/19 18:53	INO 文件	1 KB
ai_vox3_device.cpp	2026/1/8 14:58	C++ 源文件	20 KB
ai_vox3_device.h	2025/12/25 16:01	C Header 源文件	2 KB
build_opt.h	2025/12/19 18:53	C Header 源文件	1 KB
display.cpp	2025/12/19 18:53	C++ 源文件	16 KB
display.h	2025/12/19 18:53	C Header 源文件	2 KB

双击打开
编译上传

图 1: alt text

硬件连接

将US04超声波模块连接到AI-VOX3扩展板的IO1、IO2引脚，请使用4pin的PH2.0连接线，直插式连接，确保连接正确无误。

US04模块引脚	AI-VOX3扩展板引脚
VCC	5V
GND	G
Trig	2
Echo	1

源码展示

```
#include <Arduino.h>
#include "ai_vox3_device.h"
#include "ai_vox_engine.h"
#include <ArduinoJson.h>

// ======US040 -----
constexpr gpio_num_t kUs04PinTrig = GPIO_NUM_2; // trig pin
constexpr gpio_num_t kUs04PinEcho = GPIO_NUM_1; // echo pin

/** @brief US04
 *
 *      US04
 *
 *
 *      10
 *
 *
 *
 * @return float
 *         -1
 */
float MeasureUs04UltrasonicDistance()
{
    //      2      10
    digitalWrite(kUs04PinTrig, LOW);
    delayMicroseconds(2);
    digitalWrite(kUs04PinTrig, HIGH);
    delayMicroseconds(10);
    digitalWrite(kUs04PinTrig, LOW);

    const uint16_t timeout = 30000; // 30      5

    //
    const auto duration = pulseIn(kUs04PinEcho, HIGH, timeout);

    if (duration <= 0)
    {
        printf("Error: US04 sensor measure timeout.\n");
        return -1;
    }
    //      ( ) *  (0.034cm/ ) / 2 ( )
}
```

```

        return static_cast<float>(duration * 0.034 / 2);
    }

/*
 * @brief MCP - US04
 *
 *      "user.ultrasonic_sensor.get_distance" MCP
 *      US04
 */
void mcp_tool_us04_ultrasonic_sensor()
{
    //
    RegisterUserMcpDeclarator([](ai_vox::Engine &engine)
    {
        engine.AddMcpTool("user.ultrasonic_sensor.get_distance", // "Get distance from US04 ultrasonic sensor", // {}
    });

    //
    RegisterUserMcpHandler("user.ultrasonic_sensor.get_distance", [](const ai_vox::McpToolCallEvent &ev
    {
        //
        const float distance = MeasureUs04UltrasonicDistance();

        //
        if (distance < 0) {
            //
            ai_vox::Engine::GetInstance().SendMcpCallError(ev.id, "Failed to measure distance from US04");
        } else {
            //
            printf("on mcp tool call: user.ultrasonic_sensor.get_distance, distance: %.1f cm\n", distance);

            //
            ai_vox::Engine::GetInstance().SendMcpCallResponse(ev.id, std::to_string(distance).c_str());
        }
    });
}

// ===== Setup Loop =====
void setup()
{
    Serial.begin(115200);
    delay(500); //

    printf("\n===== US04 Initialization =====\n");
    pinMode(kUs04PinTrig, OUTPUT);
    pinMode(kUs04PinEcho, INPUT);
    printf("======\n\n");

    // MCP - US04

```

```
mcp_tool_us04_ultrasonic_sensor();  
//      AI  
InitializeDevice();  
}  
  
void loop()  
{  
    //  
    ProcessMainLoop();  
}
```

语音交互使用流程

注意：请先在小智AI后台，清空历史记忆，防止出现不同程序间记忆冲突的问题。

1. 用户通过按键或语音唤醒（“你好小智”）唤醒小智AI。
2. 用户通过麦克风对AI-VOX3说出“现在障碍物的距离是多少？”。
3. 小智AI识别到用户输入的意图指令，并调用相应的MCP工具进行障碍物距离读取并播报。从屏幕日志中可以看到“% user.ultrasonic_sensor.get_distance”的MCP工具调用日志。