

Arduino 开源物联网网关

——自开发网关,用户定义产品! **USR-EG118**



指导手册

联网找有人

可信赖的智慧工业物联网伙伴



1	_		_	_		ы.
	\mathbf{n}	n	T.	$\boldsymbol{\Box}$	n	т
0	U		- L	\sim		- L

1.1. 概述 4 1.2. Arduino 简介 4 1.3. 硬件规格 4 1.4. 引脚咬射 5 1.5. 指示灯 6 1.6. 硬件结构 6 2. Arduino 开发环境简介 7 2.1. Arduino 开发环境简介 7 2.2. 健立开发环境的主要步骤 7 2.3.1. 下载并安装 Arduino IDE 7 2.3.2. 添加 Boards Manager URLs 7 2.3.3. 安装 ESP32 板 8 2.3.4. 安装所需要的库文件 8 2.3.5. 添加 USR-EG118 库文件 10 3.1. 硬件连接 11 3.2. 通过 Arduino IDE 烧束 11 3.2.1. 下载参数设置 11 3.2.2. 烧菜履程序 13 3.3. 通过 ESP32 下载工具进行烧束、 14 3.3. 通过 ESP32 工具进行提序烧束 14 3.3. 通过 ESP32 工具进行提序烧束 14 3.3. 通过 ESP32 工具进行程序烧束 14 3.3. 通过 ESP32 工具进行程序烧束 14 3.3. 通过 ESP32 工具进行程序烧束 14 3.3.1. 导出已编译的二进制文件 14 3.3.2. 通过 ESP32 工具进行程序烧束 14 3.3.1. 导出已编译的二进行发展 14 4. 功能介绍和测试 16 4.1. 开始 16 4.2.	1.	产品介绍4
1.2. Arduino 简介 4 1.3. 硬件规格 4 1.4. 引脚映射 5 1.5. 指示灯 6 1.6. 硬件结构 6 2. Arduino 开发环境介绍 7 2.1. Arduino 开发环境介绍 7 2.1. Arduino 开发环境向介 7 2.2. 建立开发环境的主要步骤 7 2.3. 具体步骤说明 7 2.3. 具体步骤说明 7 2.3. 二、下载并安装 Arduino IDE 7 2.3.1. 下载并安装 Arduino IDE 7 2.3.2. 添加 Boards Manager URLs 7 2.3.3. 安装 ESP32 板 8 2.3.4. 安装所需要的库文件 8 2.3.5. 添加 USR-EG118 库文件 10 3. 硬序绕录 11 3.1. 硬件连接 11 3.2.1. 下载参数设置 11 3.2.2. 烷灵程序 13 3.3. 通过 ESP32 下载工具进行提示操家 14 3.3.1. 导出已编译的二进制文件 14 3.3.2. 通过 ESP32 工具进行程序端录 14 3.3.2. 通过 ESP32 工具进行程序操录 14 3.3.3. 通过 ESP32 工具进行程序操录 14 3.3.1. 导出已编译的二进制文件 16 4. 功能介绍和测试 16 4.1. 开始 16 4.2. IO 接口 17		1.1. 概述
1.3. 硬件规格 4 1.4. 引脚映射 5 1.5. 指示灯 6 1.6. 硬件结构 6 2. Arduino 开发环境价组 7 2.1. Arduino 开发环境的主要步骤 7 2.2. 建立开发环境的主要步骤 7 2.3. 具体步骤说明 7 2.3. 具体步骤说明 7 2.3.1. 下载井安装 Arduino IDE 7 2.3.2. 添加 Boards Manager URLs 7 2.3.3. 安装 ESP32 板 8 2.3.4. 安裝所需要的庫文件 8 2.3.5. 添加 USR-EG118 库文件 10 3. 超序焼录 11 3.1. 硬件连接 11 3.2. 進过 Arduino IDE 烧灵 11 3.2. 進过 Arduino IDE 烧灵 11 3.2. 進过 Arduino IDE 烧灵 11 3.2. 通过 Arduino IDE 烧灵 11 3.2. 進过 Arduino IDE 烧灵 11 3.2. 進过 Braya 14 3.3. 通过 ESP32 T、載工行檢察 14 4.0 功能介绍和測試 16 4.1. 开始 16 4.1. 开始 16 4.2. 10 接口		1.2. Arduino 简介
1.4. 引脚映射 5 1.5. 指示灯 6 1.6. 硬件结构 6 2. Arduino 开发环境介绍 7 2.1. Arduino 开发环境的主要步骤 7 2.2. 建立开发环境的主要步骤 7 2.3. 具体步骤说明 7 2.3. 具体步骤说明 7 2.3.1. 下載井安装 Arduino IDE 7 2.3.2. 添加 Boards Manager URLs 7 2.3.3. 安装 ESP32 板 8 2.3.4. 安装所需要的库文件 8 2.3.5. 添加 USR-EG118 库文件 10 3. 程序烷录 11 3.2. 通过 Arduino IDE 際景 11 3.2. 通过 Arduino IDE 際景 11 3.2.1. 下载参数设置 11 3.2.2. 源泉程序 13 3.3. 通过 ESP32 下载工具进行烧泉 14 3.3.1. 导出已编译的二进制文件 14 3.3.1. 导出已编译的二进制文件 14 3.3.2. 通过 ESP32 工具进行院娘录 14 4. 功能介绍和测试 16 4.1. 开始 16 4.1. 开始 17 4.3. STA 设置 17 4.4. 添加拓展机 18 4.5. CF 模式电口网口透传 19 4.6. MQTT 模式 MQTT 串口透传 20 70 7 7 </th <td></td> <td>1.3. 硬件规格</td>		1.3. 硬件规格
1.5. 指示灯 6 1.6. 硬件结构 6 2. Arduino 开发环境介绍 7 2.1. Arduino 开发环境简介 7 2.2. 建立开发环境的主要步骤 7 2.3. 具体步骤说明 7 2.3.1. 下载并安装 Arduino IDE 7 2.3.2. 添加 Boards Manager URLs 7 2.3.3. 安装 ESP32 板 8 2.3.4. 安装所需要的库文件 8 2.3.5. 添加 USR-EG118 库文件 10 3. 健作连接 11 3.1. 硬件连接 11 3.2. 通过 Arduino IDE 烧录 11 3.2. 通过 Arduino IDE 烧录 11 3.2. 通过 ESP32 下载工具进行烧录 14 3.3. 通过 ESP32 下载工具进行烧录 14 3.3. 通过 ESP32 工具进行健常缺量 14 3.3. 通过 ESP32 工具进行健康 14 3.3. 通过 ESP32 工具进行健康 17 4. 功能介绍和测试 16 4.1. 开始 16 4.1. 开始 16 4.2. IO 接口 17 4.3. STA 设置 17		1.4. 引脚映射
1.6. 硬件结构 6 2. Arduino 开发环境简介 7 2.1. Arduino 开发环境简介 7 2.2. 建立开发环境的主要步骤 7 2.3. 具体步骤说明 7 2.3. 具体步骤说明 7 2.3.1. 下载并安装 Arduino IDE. 7 2.3.2. 添加 Boards Manager URLs 7 2.3.3. 安装 ESP32 板 8 2.3.4. 安装所需要的库文件 8 2.3.5. 添加 USR-EG118 库文件 10 3. 程序炼录 11 3.1. 硬件连接 11 3.2. 通过 Arduino IDE 烧录 11 3.2. 通过 Arduino IDE 烧录 11 3.2. 通过 ESP32 下载工具进行烧录 14 3.3. 通过 ESP32 工具进行程序烧录 14 3.3.1. 导出已编译的二进制文件 14 3.3.1. 导出已编译的二进制文件 14 3.3.1. 导出已编译的二进制文件 14 3.3.1. 导出已编译的二进制文件 14 3.3. 通过 ESP32 工具进行程序烧录 14 4. 功能介绍和测试 16 4. 小能介绍和测试 16 4.1. 开始 16 4.2. IO 接口 17 4.3. STA 设置 17 4.4. 添加拓展机 18 4.5. TCP 模式串口网口透传 19 4.6. MQTT 每口透传		1.5. 指示灯
2. Arduino 开发环境简介		1.6. 硬件结构
2.1. Arduino 开发环境简介	2.	Arduino 开发环境介绍
2.2. 建立开发环境的主要步骤 7 2.3. 具体步骤说明 7 2.3.1. 下载并安装 Arduino IDE 7 2.3.2. 添加 Boards Manager URLs 7 2.3.3. 安装 ESP32 板 8 2.3.4. 安装所需要的库文件 8 2.3.5. 添加 USR-EG118 库文件 10 3. 程序烧录 11 3.1. 硬件连接 11 3.2. 通过 Arduino IDE 烧录 11 3.2. 通过 Arduino IDE 烧录 11 3.2.1. 下载参救设置 11 3.2.2. 烧录程序 13 3.3. 通过 ESP32 下载工具进行烧录 14 3.3.1. 导出已编译的二进制文件 14 3.3.2. 通过 ESP32 工具进行烧束烧录 14 4. 功能介绍和测试 16 4. 1. 开始 16 4.1. 开始 16 4.2.10 接口 17 4.3.5TA 设置 17 4.4. 添加拓展机 18 4.5. TCP 模式串口网口透传 19 4.6. MQTT 模式 MQTT 串口透传 20		2.1. Arduino 开发环境简介
2.3. 具体步骤说明. 7 2.3.1. 下载并安装 Arduino IDE. 7 2.3.2. 添加 Boards Manager URLs 7 2.3.3. 安装 ESP32 板 8 2.3.4. 安装所需要的库文件 8 2.3.5. 添加 USR-EG118 库文件 10 3. 程序烧录 11 3.1. 硬件连接 11 3.2. 通过 Arduino IDE 烧录 11 3.2. 通过 Arduino IDE 烧录 11 3.2. 通过 Arduino IDE 烧录 11 3.2. 通过 ESP32 下载工具进行烧录 11 3.3. 通过 ESP32 下载工具进行烧录 14 3.3.1. 导出已编译的二进制文件 14 3.3.2. 通过 ESP32 工具进行提序烧录 14 4. 功能介绍和测试 16 4.1. 开始 16 4.2. IO 接口 17 4.3. STA 设置 17 4.4. 添加拓展机 18 4.5. TCP 模式串口网口透传 19 4.6. MQTT 樓式 MQTT 串口透传 20		2.2. 建立开发环境的主要步骤
2.3.1.下载并安装 Arduino IDE. 7 2.3.2.添加 Boards Manager URLs. 7 2.3.3. 安装 ESP32 板 8 2.3.4. 安装所需要的库文件. 8 2.3.5.添加 USR-EG118 库文件. 10 3. 程序烧录 11 3.1. 硬件连接 11 3.2. 通过 Arduino IDE 烧录 11 3.2. 通过 Arduino IDE 烧录 11 3.2. 通过 Arduino IDE 烧录 11 3.2. 通过 ESP32 下载工具进行烧录 11 3.3. 通过 ESP32 下载工具进行烧录 14 3.3.1. 导出已编译的二进制文件 14 3.3.2. 通过 ESP32 工具进行程序烧录 14 4. 功能介绍和测试 16 4.1. 开始 16 4.2.10 接口 17 4.3. STA 设置 17 4.4. 添加拓展机 18 4.5. TCP 模式串口网口透传 19 4.6. MQTT 模式 MQTT 串口透传 20 5. WAT # 20		2.3. 具体步骤说明
2.3.2. 添加 Boards Manager URLs 7 2.3.3. 安装 ESP32 板 8 2.3.4. 安装所需要的库文件 8 2.3.5. 添加 USR-EG118 库文件 10 3. 程序烧录 11 3.1. 硬件连接 11 3.2. 通过 Arduino IDE 烧录 11 3.2.1. 下载参数设置 11 3.2.2. 烧灵程序 13 3.3. 通过 ESP32 下载工具进行烧灵 14 3.3.1. 导出已编译的二进制文件 14 3.3.2. 通过 ESP32 工具进行程序烧灵 14 3.3.2. 通过 ESP32 工具进行程序烧录 14 4. 功能介绍和测试 16 4.1. 开始 16 4.2. IO 接口 17 4.3. STA 设置 17 4.4. 添加拓展机 18 4.5. TCP 模式串口网口透传 19 4.6. MQTT 模式 MQTT 串口透传 20 5. WAT 20		2.3.1. 下载并安装 Arduino IDE
2.3.3. 安装 ESP32 板. 8 2.3.4. 安装所需要的库文件. 8 2.3.5. 添加 USR-EG118 库文件. 10 3. 程序烧录. 11 3.1. 硬件连接. 11 3.2. 通过 Arduino IDE 烧录. 11 3.2. 通过 Arduino IDE 烧录. 11 3.2. 通过 Arduino IDE 烧录. 11 3.2. 通过 ESP32 下载工具进行烧录. 13 3.3. 通过 ESP32 下载工具进行烧录. 14 3.3.1. 导出已编译的二进制文件. 14 3.3.2. 通过 ESP32 工具进行程序烧录. 14 4. 功能介绍和测试. 16 4.1. 开始. 16 4.2.10 接口. 17 4.3.5TA 设置 17 4.4. 添加拓展机. 18 4.5. TCP 模式串口网口透传. 19 4.6. MQTT 模式 MQTT 串口透传. 20		2.3.2. 添加 Boards Manager URLs
2.3.4. 安装所需要的库文件. 8 2.3.5. 添加 USR-EG118 库文件 10 3. 程序烧录 11 3.1. 硬件连接 11 3.2. 通过 Arduino IDE 烧录 11 3.2.1. 下载参数设置 11 3.2.2. 烧录程序 13 3.3. 通过 ESP32 下载工具进行烧录 14 3.3.1. 导出已编译的二进制文件 14 3.3.2. 通过 ESP32 工具进行烧录 14 3.3.1. 导出已编译的二进制文件 14 3.3.2. 通过 ESP32 工具进行烧录 14 3.3.1. 导出已编译的二进制文件 16 4.1. 开始 16 4.2. IO 接口 17 4.3. STA 设置 17 4.4. 添加拓展机 18 4.5. TCP 模式串口网口透传 19 4.6. MQTT 模式 MQTT 串口透传 20 5. TCP 模式 串口 20 5. TCP 模式 串口 20		2.3.3. 安装 ESP32 板
2.3.5. 添加 USR-EG118 库文件 10 3. 程序烧录 11 3.1. 硬件连接 11 3.2. 通过 Arduino IDE 烧录 11 3.2.1. 下载参数设置 11 3.2.2. 烧录程序 13 3.3. 通过 ESP32 下载工具进行烧录 14 3.3.1. 导出已编译的二进制文件 14 3.3.2. 通过 ESP32 工具进行程序烧录 14 3.3.2. 通过 ESP32 工具进行程序烧录 14 4. 功能介绍和测试 16 4.1. 开始 16 4.2. IO 接口 17 4.3. STA 设置 17 4.4. 添加拓展机 18 4.5. TCP 模式串口网口透传 19 4.6. MQTT 模式 MQTT 串口透传 20		2.3.4. 安装所需要的库文件
3. 程序焼泉 11 3.1. 硬件连接 11 3.2. 通过 Arduino IDE 烧泉 11 3.2.1. 下載参数设置 11 3.2.2. 烧录程序 13 3.3. 通过 ESP32 下載工具进行烧泉 14 3.3.1. 导出已编译的二进制文件 14 3.3.2. 通过 ESP32 工具进行程序烧录 14 3.3.2. 通过 ESP32 工具进行程序烧录 14 3.3.1. 导出已编译的二进制文件 14 3.3.2. 通过 ESP32 工具进行程序烧录 14 4. 功能介绍和测试 16 4.1. 开始 16 4.2. IO 接口 17 4.3. STA 设置 17 4.4. 添加拓展机 18 4.5. TCP 模式串口网口透传 19 4.6. MQTT 模式 MQTT 串口透传 20		2.3.5. 添加 USR-EG118 库文件
3.1. 硬件连接 11 3.2. 通过 Arduino IDE 烧录 11 3.2.1. 下载参数设置 11 3.2.2. 烧录程序 13 3.3. 通过 ESP32 下载工具进行烧录 14 3.3.1. 导出已编译的二进制文件 14 3.3.2. 通过 ESP32 工具进行烧录 14 3.3.2. 通过 ESP32 工具进行程序烧录 14 4. 功能介绍和测试 16 4.1. 开始 16 4.2. IO 接口 17 4.3. STA 设置 17 4.4. 添加拓展机 18 4.5. TCP 模式串口网口透传 19 4.6. MQTT 槔式 MQTT 串口透传 20	3.	程序烧录11
3.2. 通过 Arduino IDE 烧录 11 3.2.1. 下载参数设置 11 3.2.2. 烧录程序 13 3.3. 通过 ESP32 下载工具进行烧录 14 3.3.1. 导出已编译的二进制文件 14 3.3.2. 通过 ESP32 工具进行烧录 14 3.3.2. 通过 ESP32 工具进行程序烧录 14 4. 功能介绍和测试 16 4.1. 开始 16 4.2. IO 接口 17 4.3. STA 设置 17 4.4. 添加拓展机 18 4.5. TCP 模式串口网口透传 19 4.6. MQTT 模式 MQTT 串口透传 20		3.1. 硬件连接
3.2.1. 下载参数设置 11 3.2.2. 烧录程序 13 3.3. 通过 ESP32 下载工具进行烧录 14 3.3.1. 导出已编译的二进制文件 14 3.3.2. 通过 ESP32 工具进行程序烧录 14 4. 功能介绍和测试 16 4.1. 开始 16 4.2. IO 接口 17 4.3. STA 设置 17 4.4. 添加拓展机 18 4.5. TCP 模式串口网口透传 19 4.6. MQTT 模式 MQTT 串口透传 20		3.2. 通过 Arduino IDE 烧录 11
3.2.2. 烧录程序. 13 3.3. 通过 ESP32 下载工具进行烧录. 14 3.3.1. 导出已编译的二进制文件. 14 3.3.2. 通过 ESP32 工具进行程序烧录. 14 4. 功能介绍和测试. 16 4.1. 开始. 16 4.2. IO 接口. 17 4.3. STA 设置. 17 4.4. 添加拓展机. 18 4.5. TCP 模式串口网口透传. 19 4.6. MQTT 模式 MQTT 串口透传. 20		3.2.1. 下载参数设置
3.3. 通过 ESP32 下载工具进行烧录 14 3.3.1. 导出已编译的二进制文件 14 3.3.2. 通过 ESP32 工具进行程序烧录 14 4. 功能介绍和测试 16 4.1. 开始 16 4.2. IO 接口 17 4.3. STA 设置 17 4.4. 添加拓展机 18 4.5. TCP 模式串口网口透传 19 4.6. MQTT 模式 MQTT 串口透传 20		3.2.2. 烧录程序
3.3.1. 导出已编译的二进制文件 14 3.3.2. 通过 ESP32 工具进行程序烧录 14 4. 功能介绍和测试 16 4.1. 开始 16 4.2. IO 接口 17 4.3. STA 设置 17 4.4. 添加拓展机 18 4.5. TCP 模式串口网口透传 19 4.6. MQTT 模式 MQTT 串口透传 20		3.3. 通过 ESP32 下载工具进行烧录
3.3.2. 通过 ESP32 工具进行程序烧录 14 4. 功能介绍和测试 16 4.1. 开始 16 4.2. IO 接口 17 4.3. STA 设置 17 4.4. 添加拓展机 18 4.5. TCP 模式串口网口透传 19 4.6. MQTT 模式 MQTT 串口透传 20		3.3.1. 导出已编译的二进制文件
4. 功能介绍和测试 16 4. 1. 开始 16 4. 2. 10 接口 17 4.3. STA 设置 17 4.4. 添加拓展机 18 4.5. TCP 模式串口网口透传 19 4.6. MQTT 模式 MQTT 串口透传 20		3.3.2. 通过 ESP32 工具进行程序烧录
4. 1.开始	4.	功能介绍和测试
 4.2. IO 接口		4.1. 开始 16
 4.3. STA 设置		4.2. IO 接口
 4.4. 添加拓展机		4.3. STA 设置
 4.5. TCP 模式串口网口透传		4.4. 添加拓展机
4.6. MQTT 模式 MQTT 串口透传		4.5. TCP 模式串口网口透传
		4.6. MQTT 模式 MQTT 串口透传
2.	5.	联系方式

6.	免责声明	2
----	------	---

1.产品介绍

1.1.概述

USR-EG118 是一个基于 Arduino 平台编程开发的积木式开源物联网网关,以乐鑫公司的 ESP32 模块为核心配备 Xtensa®双核 32 位 LX6 CPU, 240 MHz 主频, 8MB PSRAM 和 4MB SPI Flash。集成了丰富的硬件接口:RS485/232、DO*2、DI*1、AI*1、程序烧录口、Wi-Fi、BLE 和以太网口,并可以在网关右侧插入最多 6 块 IO 扩展机实现更强大的 IO 控制功能。

本设备基于 C 语言和 Arduino 编程,需要用户具备一定的开发能力。其中 Arduino 编程的很大一部分工作是对函数库的使用,网络上可以找到数以千计的函数库,而最好的函数库可以直接通过编辑器找到并安装。

有人物联网为用户开发应用程序提供基本的 DEMO,用户可参考借鉴此 DEMO 进行协助开发。设备具备丰富的硬件接口和网络,该 设备可应用于多种场景,特别是用户需要同时采集和控制串行设备及 IO 设备的场景,如:农业灌溉、废水处理、数字化工厂等。

1.2.Arduino 简介

Arduino 平台自 2005 年成立以来,已发展成为电子和嵌入式设计领域最知名的品牌之一。Arduino 是一个基于易于使用的硬件和 软件的开源电子平台。您可以通过向电路板上的 MCU 发送一组指令来告诉电路板要做什么。

Arduino 诞生于 Ivrea Interaction Design Institute,是一种简单的快速原型设计工具,面向没有电子和编程背景的学生。当进入更 广泛的社区,Arduino 板就开始改变以适应新的需求和挑战,产品从简单的 8 位板到应用于物联网、可穿戴设备、3D 打印和嵌入式环境 的产品。

多年来,Arduino 一直是数千个项目的大脑,从日常物品到复杂的科学仪器。一个由学生、业余爱好者、艺术家、程序员和专业人 士组成的全球创客社区聚集在这个开源平台周围,他们的贡献加起来提供了数量惊人的可访问知识,对新手和专家都有很大帮助。由于 其简单易用的用户体验,Arduino 已被用于数千个不同的项目和应用程序。Arduino 软件对于初学者来说易于使用,但对于高级用户来 说却足够灵活。它可以在 Mac、Windows 和 Linux 上运行。

1.3.硬件规格

设备参数		描述
	芯片	ESP32-WROVER-E-N4R8
设备规格	处理器	双核 Xtensa®32 位 LX6 CPU,240MHz
	PSRAM	8M
	Flash	4M
	电源	9~36V
	RS232/485	1*RS232/485; 默认参数: 115200/N/8/1
硬件接口	DI	1*DI: 检测 0~2 V 判断为低电平,9~36 V 检测为高电平
	DO	2*DO; NO: 10A/28V DC, 10A/277V AC; NC: 5A/250V AC
	AI	1*AI (4~20 mA)
Wi-Fi 规格	无线标准	IEEE802.11b/g/n

表 1. 硬件规格

	理论速率	高达 150Mbps, A-MPDU 和 A-MSDU 聚合, 支持 0.4 µs 保护间隔
	频段	2412 ~ 2484 MHz
蓝牙	BLE	兼容 BEL4.2

1.4.引脚映射

表 2 描述了 USR-EG118 的硬件接口引脚与主控 ESP32 引脚的对应关系。 用户还可以下载 USR-EG118 对应的硬件原理图。下载链接: https://www.usr.cn/wiki/puba/QMJcMJNVg#top

USR-EG118 主控引脚		3 主控引脚	ESP32 引朋	卽	功能
序号	功能接口	引脚	引脚名称	引脚号	描述
1		RMII_MDIO	GPIO18	30	
2		RMII_MDC	GPIO23	37	
3		RMII_REF_CLK	GPIO0	25	
4		RMII_TXD0	GPIO19	31	
5	RMII 接口	RMII_TXD1	GPIO22	36	田王以大网端口
6		RMII_RX0	GPIO25	10	用了以太网项口
7		RMII_RX1	GPIO26	11	
8		RMII_TXEN	GPIO21	33	
9		RMII_CRS_DV	GPIO27	12	
10	PHY 复位	phy_reset	GPIO5	29	
11		UART1_TX	GPIO32	8	
12	RS485/232	UART1_RX	GPIO33	9	串行端口
13		RS485_EN1	GPI012	14	
14		UART2_TX	GPIO13	16	
15	板件 RS485	UART2_RX	GPIO34	6	用于 IO 扩展机
16		RS485_EN2	GPIO14	13	
17		OE	GPIO4	26	
10	锁存器控制	LE	GPIO15	23	寄存器输入/控制引脚
10		LL	GPIO2	24	
19	DO	DO1	DO1	Q1	数字输出(未直接连接到
20	00	D02	DO2	Q2	ESP32)
21	DI	DI1	GPIO39	5	数字输入
22	AI	ADC1	GPIO35	7	模拟输入
23	喂狗 IO	MCU_WDI	RXD0	34	给看门狗喂食的引脚

表 2. 引脚映射表



24	恢复出厂	Reload	sensor_vp	4	1.重置为出厂设置 2.启用程序烧录模式
25	LED	引线工作	U0_RX	34	工作指标
26	复位	RESET	en	3	看门狗的复位引脚
27	核录口	SWCLK	U0_TX	35	程序核型
28		SWCIO	U0_RX	34	

1.5.指示灯

USR-EG118 共具备 2 个设备状态指示灯,具体说明如下。

表 3. 指示灯

参数	描述
	长亮: 电源供应正常。
FOW	熄灭:无电源或电源异常。
WORK	1S 闪烁 1 次:系统工作正常。

1.6.硬件结构

▶对导轨安装和壁面安装的支撑。

▶79.6*58*110.0mm (左侧,不含配件)



2.Arduino 开发环境介绍

2.1.Arduino 开发环境简介

用户可以使用 Arduino 提供的 Arduino IDE 软件进行编程开发。本文是基于 Arduino IDE 开发环境进行介绍的。

Arduino 编程的很大一部分是库的使用,通过在 Arduino IDE 编程软件中的库管理器可以搜索并安装函数库。Arduino 软件作为开 源工具发布,可供有经验的程序员扩展。该语言可以通过 C++库进行扩展,想要了解技术细节的人可以从 Arduino 跳转到它所基于的 AVR-C 编程语言。同样,如果需要,您可以将 AVR-C 代码直接添加到 Arduino 程序中。

大多数 Arduino 板被设计为在微控制器上运行单个程序。该程序可以设计为执行单个操作,例如闪烁 LED。它还可以设计为在一个 周期内执行数百个操作。范围因程序而异。加载到微控制器的程序将在通电后立即开始执行。每个程序都有一个称为"循环"的函数。 在循环函数中,例如:读取传感器、打开一盏灯、检查是否满足条件。程序的速度非常快,除非我们让它放慢速度。这取决于程序的大 小以及 MCU 执行程序所需的时间,但通常以微秒(百万分之一秒)为单位。

2.2.建立开发环境的主要步骤

有关构建开发环境的步骤,请参见下图。具体的操作步骤将在下面的章节中详细介绍。而有关 Arduino IDE 编程软件更多的操作说明,可查阅 Arduino 官网说明文档:

https://docs.arduino.cc/software/ide-v2/tutorials/ide-v2-installing-a-library/



2.3.具体步骤说明

2.3.1.下载并安装 Arduino IDE

下载并安装 Arduino IDE: <u>https://www.arduino.cc/en/Main/Software</u>

2.3.2.添加 Boards Manager URLs

安装 IDE 后,添加额外的 Boards Manager URLs:**https://dl.espressif.com/dl/package_esp32_index.json** 依次点击:File->Preference-> Add Additional Boards Manager URLs

M100	/20240222v2 Arduino IDE 2.3.2		- ð ×
	Select Board		-,Q: ↓
	LIBRARY MANAGER Filter your search. Type: All Type: Communication Anduino Cloud Provider Examples by Arduino Examples of how to connect various Arduino bards to cloud providers More info	M100v20240222222 ino 0 L PETTENDECENNENN(); 222 writetoEEPROMRAM(deviceMode_ADDR, DEVICE_Mode_AT_COMMAND); 233 wpdate_RAM_toEEPROM(); 244) 255 static void EEPROM_setdeviceMode_MOTT_DTU_ONLY(void) (256 referebEEPROMRAM(deviceMode_ADDR, DEVICE_MODE_MOTT_DTU_ONLY); 287 writetoEEPROMRAM(deviceMode_ADDR, DEVICE_MODE_MOTT_DTU_ONLY); 288 writetoEEPROMRAM(deviceMode_ADDR, DEVICE_MODE_MOTT_DTU_ONLY); 289) 381 static void Proferences ×	
	Arduino Uno WiFi Dev Ed Library by Arduino This library allows users to use network features like reat and mqt. Includes some tools for the SPB226. Use this library only More info INSTALL ArduinoBLE by Arduino	000 Windexes Sketchbook location: BROWSE 833 c:UsersHP-DocumentsArduino BROWSE 835 static void Ghow in the inside Sketches 837 refreshEP Caldro font size: 14 838 wintetoEPF Interface scale: 2 Automatic 100 % 840 Theme: Light. V 841 static void Ghow verbose output during compile upload 844 update_FAM Kone verbose output during Compile upload 844 update_FAM Compile variangs None v 845 // refreshEP Compile variangs None v	:
	Enables Bluetooth® Low Energy connectivity on the Arduino MKR WiFi 1010, Arduino UNO	Bail void sinvance Editor Quick Suggestions Output Additional boards manager URLs: https://dilespressif.com/dilpackage_esp32_index.json	
	WH Rez Arduno Nano 33 lof, Arduino Na More info 1.3.6 INSTALL ArduinoBearSSL by Arduino Port of BearSSL to Arduino. This library depends on ArduinoECXX8. More info 1.7.3 INSTALL	CANCEL	
8	ArduinoCloudThing by Arduino		n 825 Col 33 × No board selected D

2.3.3.安装 ESP32 板

依次点击: Tools->Board Manager->Search "ESP" ->esp by Espressif Systems->INSTALL。

注:如果无法下载,则需要电脑连接国外网站进行下载。下载 2.0.11 版本的 ESP32。

Sketch <u></u> File Edi	feb27a Arduino IDE 2.3.2 t Sketch Tools Help	-	0	×
	Select Board		\checkmark	۰ © ۰
e F Ik Ø	BOARDS MANAGER ESP 输入ESP后回车进行 Type: All Arduino AVR Boards by Arduino 1.8.6 installed Boards included in this package: Arduino Micro, Arduino NG or older, LilyPad Arduino USB, Arduino Mega or Mega SE60. Arduine Ereler, Address Linguages	<pre>sketh_feb27a.ino void setup() { // put your setup code here, to run once: // a } void loop() { // put your main code here, to run repeatedly:</pre>		•••
Q	More Info 1.8.6 REMOVE Arduino ESP32 Boards by Arduino Boards Included In this package: Arduino Nano ESP32. More Info 2.0.13 INSTALL	10		
8	esp32 by Espressif Systems Boards Included in this package: ESP32 Dev Board, ESP32-S3 Dev Board, ESP32-S3 Dev Board, ESP32-C3 Dev Board More info 2.0.11 V INSTALL	Output Keyboard@1.0.5 Installing Keyboard@1.0.5 Downloading Mouse@1.0.1 Mouse@1.0.1 Installing Mouse@1.0.1 Installed Mouse@1.0.1	=	IX A
		Ln 10, Col 1 × No board self	cted 💭	

2.3.4.安装所需要的库文件

如果用户测试我司提供的演示代码,则需要首先安装以下库文件:

ESPAsyncWebServer、ArduinoRS485、ArduinoModbus、PubSubclient、OneButton、Adc、AsyncTCP、

Atcommand、AWS_IOT、Drv74hc595、Drvdout、eg118setting、Ethernet、huaweicloud、OneButton、

pctool_modbus_demo、system_parameter、USR_IO

依次点击: Sketch->Library Manager->Search "ESPAsyncWebSrv" ->Install

用户还可以下载 Arduino IDE 上需要的其他函数库。官网文件也有提供的库文件



如果您需要使用函数库,首先需要将库包含在程序顶部。



几乎所有的库都带有您可以使用的示例。这些示例您可以通过 File->Examples->{Library}->{Example}进行访问。在下图示例,我

们选择的是 Communication->SerialPassthrough。

File Edit Stetch. Tools. Help Open. Chri-N Open. Chri-N Open. Chri-N Stetch. Edit. Stetch. Stetch. Stetch. Open. Chri-N Stetch. Stetch. Open. Chri-N Stetch. Stetch. Open. Chri-N Open. Chri-N Open. Chri-N Open. Stetch. Open. Chri-Shirts Open. Chri-Shirts Open. Chri-Shirts Open. Stetch. Open. Chri-Shirts Open. Chri-Shirts Open. Chri-Shirts Open. Chri-Shirts Open. Chri-Shirts Open. <th>х</th>	х
New Stadiels Aki-Stadiel Open Lew Clasiels Aki-Stadiel Open Lew Clasiels Aki-Stadiel Open Lew Clasiels CH-O Open Reett S22 Stadielso Buil-In nameles Clasie Ch-H Oben Reett Buil-In nameles Clasie Ch-H Oben Reett Oben Reett Saw AL. Ch-H Oben Reett Saw AL. Oben Reetter Saw AL. Oben Reetter Saw AL. Out- Sa	
New Classifier & Ak+Cell*N MOX/0224022222 Ino Open Orti-O Open Orti-O Open Recert • Stattibook • Optimization • Advanced • Obtic Chile Optimization • Optimization • Optimization • Optimization •	<u>o</u>
Open Cri-0 M00/2202/22/22 ///2 Set-block Adv Press/Cri-Nover(); Set-block Set-block Set-block Set-block Set-block Set-block Set-block Dill-in-reamples thill-in-reamples Col-e Cri-N Set-block set-block See Cri-N Set-block set-block See Cri-N Set-block set-block See Cri-N Set-block Set-block See Cri-N Set-block Set-block_PMORENCIP See Cri-N Set-block_PMORENCIP set-block_PMORENCIP Advanced Set-block Dimmer det/set/Get/Set/Block_PMORENCIP Set-block Dimmer det/set/Get/Set/Block_PMORENCIP Midi Set-block Dimmer det/set/Get/Set/Block_PMORENCIP Midi Set-block Dimmer det/set/Get/Set/Block_PMORENCIP SetialBleponse Moler Disserblock Presceter SetialBleponseSCI SetialBleponse SetialBle	
Gene Reart 0 022 022 0	
Steerblook	
Lambine Los maydes Bubine manydes Eatic void EEPROF actions (NOT NOT NOT NOT NOT NOT NOT NOT NOT NOT	
Close Critike 018/asis + attic void EPROF MacRed Mode_MORY_void) {	
Sive Cut: Cut:Strikes Strikes Strikes <td></td>	
Save As. Od1-Skifrik's OdAcomunication Addamced Addamced Od Addamced Od.Scientral Odimoner Obit Ctrl Q Odimoner Graph Addamced Odiso Official Odimoner Addamced Odiso Official Odiso Arduino Uno WiFi Det Odiso Physical/Seid This Ibrary allows uters to it Odiso Physical/Seid Odit Odiso Earoples from Custom Libraries SerialCallesponer Earoples from Custom Libraries SerialCallesponer Icel/ode_ADDR, DEVICE_MODE_MODEUS_ONLY (void) { Oli or its Fraduro Mice Nife Toto Fraduro Mice Nife Toto SerialSerialSeria More info EsRbaynet/Geria Icel/ode_ADDR, DEVICE_MODE_MODEUS_ONLY (void) { Serial Callesponer Icel/ode_ADDR, DEVICE_MODE_MODEUS_ONLY (void) { Nature Colonkier Icel/ode_ADDR, DEVICE_MODE_MODEUS_ONLY (void) { Odit Vitual/Colonkier Icel/ode_ADDR, DEVICE_MODEUS_ONLY (void) { More info SerialSer	
Peferences Cut+325 Odc.:miniation A COITable Advanced Odc.:miniation Cethod Odc.:miniation Cethod Out Cethod Odc.:miniation Cethod Odc.:miniation Out Cethod Odc.:miniation Cethod Odc.:miniation Performance Cethod Odc.:miniation Cethod Odc.:miniation Performance Cethod Odc.:miniation Physical/Dect Modice Out Cethod Odc.:miniation Cethod ADDR::miniation Cethod Performance Cethod Odc.:miniation Cethod ADDR::miniation Cethod Out Cethod Odc.:miniation Cethod ADDR::miniation Cethod ADDR::miniation Physical/Dector Cethod ADDR::miniation Cethod::miniation Cethod::miniation Cethod::miniation Tothod::miniation Cethod::miniation Cethod::miniation Cethod::miniation Cethod::miniation Cethod::miniation Tothod::miniation Cethod::miniation Cethod::miniation Cethod::miniation Cethod::miniation Tothod::miniation Cethod::miniation Cethod::miniation Cethod::miniation Cethod::miniation Tothod::mini:miniation Cethod:	
Advanced OSC-ordrol OSC-ordrol Dimmer pdeviceMode_ADDR, DEVICE_MODE_SCONTROLID_ONLY(void) { Out OSC-ordrol OSC-ordrol Osc-ordrol Osc-ordrol Advanced Osc-ordrol Serial/Single-orde Advance/Osc-ordrol Serial/Single-orde Inder Single/Single Serial/Single-orde IndeviceMode_ADDR, DEVICE_MODE_MODBUS_ONLY); Serial/Single-orde IndeviceMode/Single Serial/Single-orde Serial/Single-orde Serial/Single-orde IndeviceMode/Single Serial/Single-orde	
Opic Opic Opic Opic Opic Opic Opic Opic Arduino Uno WiFi Ion Opic Opic Opic Arduino BLE by Arduino Opic Serialization Opic Arduino BLE by Arduino NISTALL Spanorket/Sinder IceNode_ADDR, DEVICE_MODE_MODBUSCONTROLID); Arduino BLE by Arduino Opic Serialization Opic Serialization Arduino BLE by Arduino NISTALL Spanorket/Sinder IceNode_ADDR, DEVICE_MODE_MODBUSCONTROLID); Arduino BLE by Arduino Serialization IceNode_ADDR, DEVICE_MODE_MODBUSCONTROLID); IceNode_ADDR, DEVICE_MODE_MODE_MODBUSCONTROLID); Arduino BLE by Arduino Serialization IceNode_ADDR, DEVICE_MODE_MODBUS_ONTY); Serialization More info INSTALL Spanorket/Sin IceNode_ADDR, DEVICE_MODE_MODBUS_ONTY); Serialization	
Dat Chrid 07.0 ipplay Mdi Provide	- 1
Arduino Uno WiFi DY Dy Arduino 08.Strings MuldiSerial Arduino Sure NG Dy Arduino 69.US PhycicalPistel OFUSE 09.US PhycicalPistel More info SerialCallReponse r1ceMode_ADMS_DULY(void) { O.0.3 v INSTALL SerialCallReponse r1ceMode_ADMS_DULY(void) { SerialCallReponse r1ceMode_ADMS_DULY(void) {); SerialCallReponse r1ceMode_ADMS_DULY(void) {); SerialCallReponse r1ceMode_ADMS_DULY(void) {); SerialCallReponse r1ceMode_ADMS_DULY(void) {); SerialCallReponse r1ceMode_ADMS_DULY(void) { ; SerialCallReponse r1ceMode_ADMS_DULY(void) { ; SerialCallReponse r1ceMode_ADMS_DULY(void) { ; SerialCallReponse r1ceMode_ADMS_DULY(void) { ; SerialCallReponse r1ceMode_ADMS_DULY; ; VitualColonMiser r1ceMode_ADMS_DULY; ; Marge Main r1ceMode_ADMS_DULY; ; VitualColonMiser r1ceMode_ADMS_DULY; ; Worl Brazer, Admin More SWIF 1010, Arduino Nao ; ; Wri Brazer, Admin Man	
Arduino Uno WiFi Puo Toss Iterary allows surgers to tools for the EstRadue, Mark and the More informed for the Stradue Mark Mark and the Stradue Mark and the More informed for the Stradue Mark Mark and the Stradue Mark and the More informed for the Stradue Mark Mark and the Stradue Mark and the More informed for the Stradue Mark Mark and the Stradue Mark and the More informed for the Stradue Mark and the Stradue Mark and the Stradue Mark and the More informed for the Stradue Mark and the Stra	
Arduino Uno Wirf IPW by Arduino Vino Wirf 1012, Arduino Nuo 10.3arter/dit Basic/dit ReadASCIString idevice/idode_AGS_ORLYY(void) { Based Sci IString idevice/idode_AGOR, DEVICE_MODE_AGOR, DEVICE_MODE_AGOR, DEVICE_MODE_AGOR, DEVICE_MODE_AGOR, DEVICE_MODE_AGOR, DEVICE_MODE_AGOR, DEVICE_MODE_AGOR, DEVICE_MODE_MODEUS_ORLYY; SerialCallResponse Based Sci IString SerialCallResponse /ceMode_AGOR, DEVICE_MODE_AGOR, DEVICE_MODE_AGOR, DEVICE_MODE_AGOR, DEVICE_MODE_MODEUS_ORLYY; SerialCallResponse CeViceMode_AGOR, DEVICE_MODE_MODEUS_ORLYY; SerialCallResponse Based Sci INSTALL ESPAnynCV/EbSiv SerialCallResponse Color INSTALL ESPAnynCV/EbSiv SerialCallResponse ArduinoBLE by Arduino 844 VituaColorMixer /ceVice_MODE_MODEUS_ORLY); World exturne Mark Wiri 1010, Arduino Nuxo 847 void sr74hc595srclk_low(void) { Output Output Output	
b) y Alculuo 11.AduinolSP 5evialCallReporce // LeeVode_ADDR, DEVICE_MODE_AWS_ONLY); This library allows users to tools for the SP3260, UB; 11.AduinolSP 5evialCallReporce // LeeVode_ADDR, DEVICE_MODE_AWS_ONLY); SevialCallReporce SevialCallReporce // LeeVode_ADDR, DEVICE_MODE_AWS_ONLY); SevialCallReporce More info INSTALL SevialCallReporce // LeeVode_ADDR, DEVICE_MODE_MODEUS_ONLY); SevialCallReporce SevialCallReporce // LeeVode_ADDR, DEVICE_MODE_MODEUS_ONLY); SevialCallReporce 844 VirualColoNView // LeeVode_ADDR, DEVICE_MODE_MODEUS_ONLY); VirualColoNView // LeeVode_ADDR, DEVICE_MODE_MODEUS_ONLY); // refreshEEPROVRUM(); // refreshEEPROVRUM(); 847 void sor7Ahc:959srclk_low(void) { Output Output Output Image: Sevial Coll Coll Coll Coll Coll Coll Coll Co	
Fool to find use like rest and rendt tools for the SP3206, Use More info Evengles from Custom Ubraries SPAAmC/CP SerialCallReporceASCL); 0.0.3 INSTALL ESPAAmC/CP SerialCallReporceASCL); SerialCallReporceASCL); SerialCallReporceASCL); ValueCallReporceASCL); SerialCallReporceASCL); ValueCallReporceASCL); SerialCallReporceASCL); SerialCallReporceASCL); SerialCallReporceASCL); Updat SerialCallReporceASCL); SerialCallReporceASCL); SerialCallReporceASCL SerialCallReporceASCL);	
More ESPECIAL Use: College time ESPECIAL Use: Serial Restriction College time More information College time Serial Restriction College time Serial Restriction College time 0.0.3 INSTALL ESPAnymCVE Serial Restriction College time Serial Restriction College time Serial Restriction College time Serial Restriction College time ArduinoBLE by Arduino 844 Virual College time IncelVade_ADDR, DEVICE_MODE_MODBUS_ONLY); Virual College time 844 Virual College time Virual College time ArduinoBLE by Arduino 845 Virual College time Virual College time No the Anduino MLK Wini 1010, Arduino Nuo Colleget Output Colleget	
More into ESMaynC(D) Joint (CD) jdevice/dode_JODBUS_ONLY(void) { 0.3 v INSTALL ESMaynC/Weisov 843 Virtua/ColoNMore //cet/dode_JODBUS_ONLY); 0.4 ArduinoBLE by Arduino 844 Virtua/ColoNMore //cet/dode_JODBUS_ONLY); 0.6 ArduinoBLE by Arduino 844 Virtua/ColoNMore //cet/dode_JODBUS_ONLY); 0.6 ArduinoBLE by Arduino 845 // refreshEEPROMRUM(); 845 847 void sa78hc/95srclk_low(void) { 0u/put 0u/put	
0.3 INSTALL ESPAgn/Webgiv Fall 843 Virula/Colom/Mare // ceMode_ADDR, DEVICE_MODE_MODBUS_ONLY); 9 Virula/Colom/Mare // refreshEPRORRAM(); ArduinoBLE by Arduno 846 Void sn74hc595srclk_low(void) { 0.1 Void sn74hc595srclk_low(void) { Output Output	
ArduinoBLE by Arduno 844 update_Maintower(w), bt FLC_node_node(s)_cut(), bt FLC_node(s)_cut(), bt FLC_node_node(s)_cut(), bt FLC_node(s)_cut	
ArduinoBLE by Arduino 845 // refreshEERR/RAM(); Bables Bluetooth® Low Energy connective; 845 // In the Arduino MKB Writ 1010, Arduino NNAS 847 void sn74hc595srclk_low(void) { Output Output Image: Control of the Arduino NNAS Image: Control of the Arduino NNAS	
ArduinoBLE by Arduino 846 847) Enables Bluetooth® Low Energy connectivity on the Arduino MKR 3010, Arduino Nuno WR Rev2, Arduino Nuno 3010, Arduino Nuno 0xtput	
Enables Bluetooth® Low Energy connectivity on the Arduino Nuck Win 1010, Arduino Nuck Win Rev: Arduino Nuck Si Arduino Nuck	
on the Arduine NKR Wir 1010, Arduine NNO Output WiFi Rev.2, Arduine Namo 33 107, Arduine Namo	
WiFi Rev.2, Arduino Nano 33 IoT, Arduino Na	6
Marciale	
ArduinoBearSSL by Arduino	
Port of Bear/SL to Arduino. This library depends on ArduinoECCX08. Merve info	
1.7.3 V INSTALL	

2.3.5.添加 USR-EG118 库文件

我司提供的 USR-EG118 使用的库文件,访问下方链接下载 DEMO 文件:

https://www.usr.cn/wiki/puba/12xmaDwKB#top

注意:下载的库应该被复制到 Arduino IDE 的安装路径上。

3.程序烧录

有两种主要的固件烧录方式:通过 Arduino IDE 和 ESP32 提供的工具。下面介绍了这两种烧录工具的具体使用方法。无论采用哪种方法进行烧录,下载调试线都需要提前连接。

- 3.1.硬件连接
 - 在你开始之前,需要做一些准备。
 - •USR-EG118*1
 - ●USB 串口 TTL 串口线*1
 - ●Wi-Fi 天线*1
 - ●USB 转 RS485/232 串口线*1
 - ●12V/1A 电源适配器*1

硬件连接如下图所示:



3.2.通过 Arduino IDE 烧录

3.2.1.下载参数设置

Tools->Board->esp32-WROOM-DA Module

File Edit Sketh Tools Help ESP32C3 Dev Module SP32C3 Dev Module Note Sketh Ctrl+Shift+I SP32C3 Dev Module SP32C3 Dev Module BOARDS Manage Libraries Ctrl+Shift+I SP32C3 Dev Module SP32C3 Dev Module ESP Srial Monitor Ctrl+Shift+I SP32C3 Dev Module SP32C3 Dev Module SP32C3 Dev Module File Srial Monitor Ctrl+Shift+I Post SP32C3 Dev Module SP32C3 Dev Module SP32C3 Dev Module File Srial Monitor Ctrl+Shift+I Post SP32C3 Dev Module SP32C3 Dev Module Dev Module Dev Module SP32C4 Dev Module SP32C3 Dev Module Dev Modul	🔤 sketch	n_feb27a Ard	duino IDE 2.3.2			ESP32S3 Dev Module		٥	×
Auto Format Ctrl+T BOARDS Marchive Sketch BOARDS Serial Monitor ESP Serial Monitor Type: Serial Plotter Firmware Updater code here, to run once: Uplad SSL Root Certificates Boards Manager Rott Ctrl+Shift+I Kortino Ctrl+Shift+I Uplad SSL Root Certificates Boards Manager Rott Ctrl+Shift+I Boards Rott Rott Ctrl+Shift+I Uplad SSL Root Certificates Boards Manager Ctrl+Shift+I ESP32-WROOM-DA Module* Boards Rott Rott Ctrl+Shift+I Kortino Ctrl+Shift+I Ctrl+Shift+I Boards Manager Rott Ctrl+Shift+I Boards Rott Rott Ctrl+Shift+I Kortino Ctrl+Shift+I Ctrl+Shift+I Boards Manager Rott Ctrl+Shift+I Rott Ctrl+Shift+I Rott Ctrl+Shift+I Rott Ctrl+Shift+I Rott Ctrl+Shift+I Rott Boards Manager Rott Ctrl+Shift+I Rott Ctrl+Shift+I <tr< td=""><td>File Ed</td><td>lit Sketch</td><td>Tools Help</td><td></td><td><u></u></td><td>ESP32C3 Dev Module</td><td></td><td></td><td></td></tr<>	File Ed	lit Sketch	Tools Help		<u></u>	ESP32C3 Dev Module			
BARChive Sketch Ctrl + Shift + H ESP32 Dev Module ESP32 WROW-DA Module ESP3 Serial Monitor Ctrl + Shift + H code here, to run once: ESP32 WROW-DA Module Type: Serial Potter Firmware Updater ESP32 WROW-DA Module ESP32 Wrowr Module Modifie Upload SSL Root Certificates Board: Manager Ctrl + Shift + B ESP32 Shift + B ESP32 Shift + B Modifie Upload SSL Root Certificates Board: Manager Ctrl + Shift + B ESP32 Shift + B ESP32 Shift + B Arduino Modifie Get Board Info Arduino AVR Boards ESP32 Wrowr Kit (all versions) ESP32 Shift + B 18.6 Core Debug Levei: "None" Ctrl + Shift + B ESP32 Shift + B ESP32 Wrowr Kit (all versions) 18.6 Core Debug Levei: "None" UM TinyPiCO UM TinyPiCO UM TinyS2 Isah Mode: "QIO" Flash Mode: "QIO" UM RAP UM TinyS3 UM RAP Flash Mode: "QIO" Flash Mode: "QIO" UM RAP UM RAP UM RAP Hash Size: "4MB (32Mb)" Hash Size: "4MB (32Mb)" UM RAP UM RAP UM RAP Upload Speet: "921600" UM RAP UM RAP UM RAP <td< td=""><td></td><td>→</td><td>Auto Format</td><td>Ctrl+T</td><td></td><td>ESP32S2 Dev Module</td><td></td><td>\checkmark</td><td>·Q··</td></td<>		→	Auto Format	Ctrl+T		ESP32S2 Dev Module		\checkmark	·Q··
b0kets Manage Libraries Ctrl +Shift+1 ESP Serial Monitor Ctrl +Shift+1 Type: Serial Monitor Ctrl +Shift+1 Firmware Updater Firmware Updater Firmware Updater Firmware Updater Board's II Board's ESP32-WROOM-DA Module Board's II Board's ESP32-WROOM-DA Module* Board's II Port Arduino Ctrl +Shift+B Board's II Fort First Hold's Core Ctrl +Shift+B Board's II Fort First Hold's Core Ctrl +Shift+B <td>-</td> <td>BOARDS</td> <td>Archive Sketch</td> <td></td> <td></td> <td>ESP32 Dev Module</td> <td></td> <td></td> <td></td>	-	BOARDS	Archive Sketch			ESP32 Dev Module			
ESP Serial Monitor Ctrl + Shift+M Ctrl + Shift+M Code here, to run once: ESP32 Wrover Module Firmware Updater Firmware Updater ESP32 Wrover Module ESP32 Co-D4 ESP32 Si Co-D4 Monitor Upload SSL Root Certificates Board's Manager Ctrl + Shift+B ESP32 Si SoN 18.6 in Board's "ESP32-WROOM-DA Module" Board's Manager Ctrl + Shift+B ESP32 Si SoN GO Arduino Arduino Get Board Info Esp32 Si CAM LCD esp32 ESP32 Wrover Kit (all versions) 18.6 in Board's "Forguency: '240MHz (WiFi/BT)" Esp32 Si CAM LCD ESp32 Wrover Kit (all versions) 18.6 in Core Debug Levei: 'None" UM TinyPICO UM TinyPICO 18.6 in Risk Before Sketch Upload: 'Disabled" UM TinyS2 UM RAP UM RAP Flash Mode: 'QIO' UM Risk Size: '4MB (32Mb)' UM Risk Size: '4MB (32Mb)' UM RAP Flash Size: '4MB (32Mb)' UM Risk Siffs (1.2MB APP/1.5MB SPIFFS)* UM FeatherS3 UM FeatherS3 Upload Speed: '92f600* Upload Speed: '92f600* UM FeatherS3 UM FeatherS3		FOR	Manage Libraries	Ctrl+Shift+I		✓ FSP32-WROOM-DA Module			200
Type: Serial Plotter ESP32 More Info: ESP32 PICO-D4 Ibs: Firmware Updater ESP32 PICO-D4 ESP32 S3-80x Ibs: Boards TESP32-WROOM-DA Module* Boards Manager Ctrl+Shift+B ESP32-S3-80x Boards Port ESP32 S3-MILCD ESP32 S2-S3-USB-OTG Arduino Get Board Info Esp32 S2-S3-USB-OTG ESP32 S2-S3-USB-OTG CPU Frequency: '240MHz (WiFi/BT)* Esp32 S2-S3-USB-OTG ESP32 S2 Xative USB Core Debug Level: 'None* UM TinyPICO ESP32 Wrover Kit (all versions) I.8.6 Frequency: '240MHz (WiFi/BT)* UM TinyPICO Fase All Flash Before Sketch Upload: 'Disabled* UM TinyS2 UM TinyS2 Flash Mode: "QIO* Flash Mode: "QIO* UM TinyS3 UM TinyS3 Flash Mode: "CIO* Parttion Scheme: 'Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS)* UM PROS3 UM PROS3	-	ESP	Serial Monitor	Ctrl+Shift+M	p code here, to run once:	ESP32 Wrover Module			
Firmware Updater Euror Lifectors Nore Inf Boards Info CPU Frequency: '240MHz (WiFi/8T)" Arduino AVR Boards 1.8.6 Core Debug Leve: 'None" 1.8.6 Core Debug Leve: 'None" 1.8.6 Core Debug Leve: 'None" I.8.6 Core Debug Leve: 'None" I.8.7 Flash Redore: 'Core 1' I.8.8 Frash II Gas Manager (Ctrl+Shift+B I.8.9 Core Debug Leve: 'None" I.8.9 Core Debug Leve: 'None" I.8.9	1	Type:	Serial Plotter			ESP32 PICO-D4			
Ardun Upload SSL Root Certificates Expresension 18.6 in Boards tresp32-WROOM-DA Module* Boards Manager Ctrl+Shift+B ESP32-S3-USB-OTG Boards Info Port Arduino AVR Boards ESP32S CAM LCD Arduino Get Board Info ESP32-S3-USB-OTG ESP32S CAM LCD More Inf CPU Frequency: '240MHz (WiF/BT)* ESP32S CAM LCD ESP32S Varver Kit (all versions) 18.6 Core Debug Level: 'None* UM TinyPICO ESP32S Varver Kit (all versions) 18.6 Core Debug Level: 'Disabled* UM FeatherS2 UM FeatherS2 Flash Frequency: '80MHz' Hz' UM TinyS2 UM RMP Flash Size: '4MB (32Mb)* Hadian Runs On: 'Core 1* UM TinyS3 UM PROS3 Partition Scheme: 'Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS)* UM FeatherS3 UM FeatherS3		Andrite	Firmware Updater			5032-52-Boy			
1.8.6 in Board: "ESP32-WROOM-DA Module" Boards Manager Ctrl+Shift+B ESP32-34-USB-OTG Boards Manager Ctrl+Shift+B ESP32-34-USB-OTG ESP32-34-USB-OTG Boards Manager Ctrl+Shift+B ESP32-34-USB-OTG ESP32-34-USB-OTG Boards Manager Ctrl+Shift+B ESP32-34-USB-OTG ESP32-34-USB-OTG More Inf Arduino Arduino Arduino Arduino Arduino Arduino More Inf Fort Get Board Info ESP32-34-USB-OTG ESP32-34-USB-OTG 1.8.6 CPU Frequency: '240MHz (WiF//BT)* Fort Get Board Info ESP32-20-USB ESP32-20-USB 1.8.6 Core Debug Level: 'None* Fort Subort UM TinyPICO ESP32-20-USB 1.8.6 Core Debug Level: 'None* UM TinyPICO UM FeatherS2 Neo UM TinyS2 Flash Frequency: '80MHz* Fort SubOrt UM RMP UM TinyS3 UM TinyS3 Flash Size: '4MB (32Mb)* Fort Subort UM PROS3 UM PROS3 UM PROS3 Upload Speed: '921600* UM FeatherS3 UM FeatherS3 UM FeatherS3		Arduin	Upload SSL Root Certificates			ESP32-S3-BOX			
Boards Port Arduino AVR Boards ESP32S CAM LCD Arduino Arduino Arduino AVR Boards ESP32S CAM LCD More Inf CPU Frequency: '240MHz (WiF//BT)* ESP32 1.8.6 Core Debug Level: 'None* UM TinyPICO I.8.6 Core Debug Level: 'Disabled* UM TinyPICO Frase AII Flash Before Sketch Upload: 'Disabled* UM TinyPICO Flash Frequency: '80MHz' UM TinyS2 Flash Size: '4MB (32Mb)* UM RAPP Flash Size: '4MB (32Mb)* UM TinyS3 Arduino Runs On: 'Core 1* UM RPOS3 Parition Scheme: 'Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS)* UM FeatherS3 Upload Speed: '921600* UM FeatherS3		1.8.6 in	Board: "ESP32-WROOM-DA Module"	•	Boards Manager Ctrl+Shift+B	ESP32-S3-USB-UIG			
Arduino Get Board Info ESP32 ESP32 ESP32 ESP32 ESP32 More Inf CPU Frequency: '240MHz (WiF//BT)* ESP32 ESP32 ESP32 ESP32 1.8.6 Core Debug Level: 'None* UM TinyPICO ESP32 ESP32 ESP32 1.8.6 Core Debug Level: 'None* Image: March Info UM TinyPICO Image: March Info 1.8.6 Core Debug Level: 'Disabled* Image: March Info UM FeatherS2 Image: March Info 1.8.6 Frequency: '80MHz* Image: March Info UM TinyS1 Image: March Info Flash Frequency: '80MHz* Image: March Info Image: March Info Image: March Info Flash Node: 'QIO* Image: March Info Image: March Info Image: March Info Flash Node: 'QIO* Image: March Info Image: March Info Image: March Info Arduino Runs On: 'Core 1* Image: March Info Image: March Info Image: March Info Arduino Runs On: 'Core 1* Image: March Info Image: March Info Image: March Info Arduino Scheme: 'Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS)* Image: March Info Image: March Info Image: March Info <t< td=""><td>₽ B</td><td>Arduino</td><td>Port</td><td>,</td><td>Arduino AVR Boards</td><td>SESPECT AND LED</td><td></td><td></td><td></td></t<>	₽ B	Arduino	Port	,	Arduino AVR Boards	SESPECT AND LED			
18.6 CPU Frequency: "240MHz (WiFi/BT)" ESP32 Wrover Kit (all versions) 18.6 Core Debug Level: "None" UM TinyPICO Erase All Flash Before Sketch Upload: "Disabled" V UM FeatherS2 Events Run On: "Core 1" V UM TinySi Flash Frequency: "80MHz" V UM TinySi Flash Size: "4MB (32Mb)" V UM RMP Arduino Runs On: "Core 1" UM TinySi Arduino Runs On: "Core 1" UM TinySi Upload Speed: "921600" UM ReatherS3		Arduino More inf	Get Board Into		• esp32	ESP3252 Native USB			
1.8.6 Core Debug Level: "None" UM TinyPICO Erase All Flash Before Sketch Upload: "Disabled" UM FeatherS2 Events Run On: "Core 1" V UM TinyS2 Flash Frequency: "80MHz" V UM TinyS2 Flash Node: "QIO" V UM RMP Flash Size: "4MB (32Mb)" V UM TinyS3 Arduino Runs On: "Core 1" V UM ROS3 Upload Speed: "921600" VM FeatherS3	Q	4.0.0	CPU Frequency: "240MHz (WiFi/BT)"	•		ESP32 Wrover Kit (all versions)			
Frase All Flash Before Sketch Upload: "Disabled" UM FeatherS2 Events Run On: "Core 1" UM FeatherS2 Neo Flash Frequency: '80MHz' UM TinyS2 Flash Mode: "QIO" UM RMP Flash Size: "4MB (32Mb)" UM TinyS3 Arduino Runs On: "Core 1" UM PROS3 Parition Scheme: "Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS)" UM FeatherS3 Upload Speed: "921600" UM FeatherS3		1.8.6	Core Debug Level: "None"			UM TinyPICO			
Events Kun On: "Core 1" VM FeatherS2 Neo Flash Frequency: "80MHz" VM TinyS2 Flash Mode: "QIO" VM RMP Flash Size: "4MB (32Mb)" VM TinyS3 Arduino Runs On: "Core 1" VM ROS3 Parition Scheme: "Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS)" VM FeatherS3 Upload Speed: "921600" VM FeatherS3			Erase All Flash Before Sketch Upload: "Disabled	-		UM FeatherS2			
Hash Frequency: 'BUMH2' UM TinyS2 Flash Mode: 'QIO' UM RMP Flash Size: 'AMB (32Mb)' UM TinyS3 Arduino Runs On: 'Core 1' UM ROS3 Partition Scheme: 'Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS)' UM PROS3 Upload Speed: '921600' UM FeatherS3			Events Run On: "Core 1"			UM FeatherS2 Neo			
Hash Mode: UM RMP Flash Size: VM RMS Flash Size: VM Size: Arduino Runs On: Core 1' Partition Scheme: UM ROS3 Upload Speed: 921600'			Flash Frequency: BUMHZ			UM TinyS2			
Partition Scheme: "Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS)" UM TinyS3 Upload Speed: "921600" UM PROS3			Flash Circu #4MB (22MAb)*			UM RMP			
Partition Scheme: "Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS)" UM PROS3 Upload Speed: "921600" UM FeatherS3			Arduine Pune On: "Core 1"			UM TinyS3			
Upload Speed: "921600" MIX Spins (1.2MD XCP/1.5MD SPINS)			Partition Scheme: "Default AMB with spiffs (1 2MB A)	DD/1 SMR SDIEES\"		UM PROS3			
opioud speed. Selood			Lipload Speed: "921600"	FF/1.5M0 SFIFTS)		UM FeatherS3			
Programmer S.ODI Ultra v1			Programmer			S.ODI Ultra v1			
Programmer LilyGo T-Display-S3			Burn Bootloader	5		LilyGo T-Display-S3			
micro52			Juin Boortoader			microS2			
And				Output		MagicBit		=	
SUBSYDETCD/1 2 A Turta IoT Node				ESPAsyncTCP@1_2_4		Turta IoT Node	_	=×	
Installing ESPAsyncTCP@1.2.4 TTGO LoRa32-OLED				Installing ESPAsyncTCP@	1.2.4	TTGO LoRa32-OLED			
Installed ESPAsyncTCP@1.2.4 TTGO T1				Installed ESPAsyncTCP@1	.2.4	TTGO T1			
Downloading ESPAsynckebSrv@1.2.7				Downloading ESPAsyncWeb	Srv@1.2.7	TTGO T7 V1.3 Mini32			
Installing ESPAsyncWebSrv@1.2.7 TTGO T7 V1.4 Mini32				Installing ESPAsyncWebS	rv@1.2.7	TTGO T7 V1.4 Mini32			
Installed ESPAsyncWebSrv@1.2.7 TIGO T-OI PLUS RISC-V ESP32-C3	Q			Installed ESPAsyncWebSr	v@1.2.7	TTGO T-OI PLUS RISC-V ESP32-C3			
							and the second second		-

USR-EG118 和电脑连接 USB 转 TTL 串口线(此时 USR-EG118 为断电状态),依次点击 Tools->Board->Port->选择对应的端口号。 用户可以在设备管理器中检查 COM 端口,如下图所示。

如果设备管理器中没有此 COM 端口, 用户可以安装串行口驱动程序。

下载链接: https://www.pusr.com/support/download/Serial-Driver-USB-To-Serial-Driver.html

文件(F)		
DESKTOP-7PQEDUO		
> 🖵 SIMATIC NET		
> 🏺 USB 连接器管理器		
> 🔐 安全设备		
> 🔲 处理器		
> 🔚 传感器		
> 🔜 磁盘驱动器		
> 🍲 存储控制器		
> 🚍 打印队列		
> 🗐 打印机		
> 🤪 电池		
✓ ☐ 端口 (COM 和 LPT)		
💭 USB-SERIAL CH340 (COM3)		
> 🧾 固件		
> 🔜 计算机		
> 🛄 监视器		
> 🔤 键盘		
> 😣 蓝牙		
> 🔜 内存技术设备		
✓ ↓ 人体学输入设备		
🙀 GPIO 笔记本电脑或平板电脑指示器驱动程序		
HID PCI Minidriver for ISS		
🕅 HID PCI Minidriver for ISS		
HID PCI Minidriver for ISS		
HID PCI Minidriver for ISS		
🛲 HID-compliant device		

😁 sketc	h_feb27a Ard	uino IDE 2.3.2			- 0	Х
File Ec	lit Sketch	Tools Help				
		Auto Format	Ctrl+T		\checkmark	·Q··
		Archive Sketch				
	BOARDS	Manage Libraries	Ctrl+Shift+I			
	ESP	Serial Monitor	Ctrl+Shift+M	a cada hana ita nun ancar		
行)	Type:	Serial Plotter		b code nere, co run once.		
		Firmware Updater				
Ith	Arduin	Upload SSL Root Certificates				
1110	1.8.6 in:	Board: "ESP32-WROOM-DA Module"	•	code here to run repeatedly:		
	Boards in	Port	•	Serial ports		
0	Arduino	Get Board Info		СОМЗ		
0	More inf	CPU Frequency: "240MHz (WiFi/BT)"	•			
\sim	1.8.6	Core Debug Level: "None"	•			
		Erase All Flash Before Sketch Upload: "Disabled"	•			
		Events Run On: "Core 1"	•			
		Flash Frequency: "80MHz"	•			
		Flash Mode: "QIO"	•			
		Flash Size: "4MB (32Mb)"	•			
		Arduino Runs On: "Core 1"	•			
		Partition Scheme: "Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS)"	•			
		Upload Speed: "921600"	•			
		Programmer	•			
		Burn Bootloader				
		Output			ī	≣ 6
		ESPAsyncTC Installing Installed Downloadin ESPAsyncWe Installing	P@1.2.4 ESPAsyncTCP@ ESPAsyncTCP@1 g ESPAsyncWeb bSrv@1.2.7 ; ESPAsyncWebS	11.2.4 .2.4 Srv@1.2.7 rv@1.2.7		

3.2.2.烧录程序

用户可以在 Arduino IDE 上编辑、编译和下载代码。

注: 点击"Upload"按钮,先编译代码,当编译即将完成,下方报文出现 CONNECTING 时,按住 Reload 按钮,然后给 USR-EG118 通电。在程序下载过程中不要释放 Reload 按钮,否则程序将无法烧录。

当出现以下报文时,代表程序烧录完毕,可以松开 Reload 按键。

🔤 sketch	_feb27a Arduino IDE 2.3.2	-	Ō	×
File Ed	it Sketch Tools Help			
	→ 🚱 🖞 ESP32-WROOM-DA M 🔻		~	·Q··
Ph	BOARDS MANAGER	sketch_feb27a.ino		
	ESP	<pre>void setup() {</pre>		
힘	Type: Arduino V	<pre>2 // put your setup code here, to run once: 3</pre>		
	Arduino AVR Boards by Arduino 1.8.6 installed Boards included in this package: Arduino Uno Mini, Arduino Leonardo ETH, Arduino Ethernet, Arduino Fio,	<pre>5 6 void loop() { 7 // put your main code here, to run repeatedly: 8 </pre>		-
0	Arduino NG or older, Arduino Uno, Arduino Mega or More info	9 } 10		
		Output	Ξ	ž 6
8		Writing at 0x00043ca0 (100 %) Wrote 230864 bytes (127839 compressed) at 0x00010000 in 2.5 seconds (effective 751.0 kbit/s) Hash of data verified. Leaving Hard resetting via RTS pin		
		Ln 7, Col 49 ESP32-WROOM-DA Module on COM	3 🗘 4	

3.3.通过 ESP32 下载工具进行烧录

用户可以通过 Arduino 或其他工具导出 BIN 文件,然后通过 flash 下载工具下载 BIN 文件。

3.3.1.导出已编译的二进制文件

用户可以在 Arduino IDE 上编辑代码,并导出已编译的二进制文件。



打开 flash_download_tool, Chipset—ESP32

					and the second of		1 IS SE 10 77		10400-000		50 - 15 - 15 - 15 - 1		
✿ D:\0产品部\3	、产品上架\USR-N	M100-3.8\M10	0开源版本\flas	sh_download	d_tool_3.9.5_0	0\flash_d	ownload_tool	_3.9.5\fla	ash_dow	/nload_to	ol_3.9.5		×
													^
						18				_			
							DOW			×			
							ChipType:	ESP32		~			
							West Meder	Develo	n	~			
							workiviode:			_			
							LoadMode:	UART					
								OK					
													~

3.3.2.通过 ESP32 工具进行程序烧录

xxx.ino.bin 对应寄存器起始地址为 0x10000

xxx.ino.bootloader.bin 对应寄存器起始地址为 0x1000

xxx.ino.partitions.bin 对应寄存器起始地址为 0x8000

烧录程序和分区可以从我们的官方网站下载:

选择 COM 端口, BAUD 为 921600。

SPIFlashConfig: 40MHz, DIO, DoNotChgBin

1	ESP32 FLAS	H DOWNLOAD	TOOL V3.9	9.5		-		×
ſ	SPIDownloa	d						
	 ✓ J人DEMC ✓ E:/产品管 ✓ E:/产品管 ✓ □ □ <l< th=""><th>D固件\M100vi 理\M100-Ard 理\M100-Ard</th><th>20240222v uino\M100 uino\M100 分别选</th><th>r2.ino.bir D-ARD测 D-ARD测 过圣天寸应</th><th></th><th></th><th>0x10000 0x1000 0x8000 天件</th><th></th></l<>	D固件\M100vi 理\M100-Ard 理\M100-Ard	20240222v uino\M100 uino\M100 分别选	r2.ino.bir D-ARD测 D-ARD测 过圣天寸应			0x10000 0x1000 0x8000 天件	
	SPIFlashConfig	9				600		
	SPI SPEED 40MHz 26.7MHz 20MHz 80MHz	SPI MODE QIO QOUT DIO DOUT FASTRD	Co	lotChgBi Settings mbineBin Default	n	D	etectedInfo	4
								Ŧ
- 1	DownloadPan	el 1						
	IDLE 等待							
	START	STOP	ERASE	COM: BAUD:	CON 9216	43 500		~ ~

1>关闭 USR-EG118 的电源, 按住 Reload 按钮后, 再给 USR-EG118 设备通电, 点击 "START"按钮, 开始编程。

ESP32 FLAS	H DOWNLOA	D TOOL V3.	9.5		-		×
SPIDownloa	d						
SPIFlashConfil SPIFlashConfil SPI SPEED 40MHz 26.7MHz 20MHz	D副律\M100-Ar 理\M100-Ar 理\M100-Ar 理\OVERSIGNED SPI MODE QIO QUO QUU QUU QUU QUU QUU QUU QUU QUU	v20240222 duino\M10 duino\M10 Lock	v2.ino.bir i0-ARD i0-AR		0 0 0 0 0 0 0 0 D	0x10000 0x1000 0x8000 etectedInfo	
DownloadPan	el 1						
IDLE 等待							*
START	STOP	ERASE	COM: BAUD:	CON 921	√13 600		~

2>完成编程。

ESP32 FLAS	h downloae	D TOOL V3.	9.5		_		×
SPIDownloa	d						
	9 SPI MODE ②QIO ③QOUT ③DIO ③DOUT ③FASTRD	20240222 Juino\M10 Juino\M10 Loc Cc	v2.ino.bir)0-ARD測)0-ARD測 NotChgB kSettings ombineBi Default	1 1	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Ox10000 Ox1000 Ox8000 Cx8000 Sabo vendor: Sabo vendor: Sa	
DownloadPar	iel 1						
FINISH 完成	P: 441793E1 T: 441793E1	14235 ST/ 14236 ETI	A: 44179 HERNET:	3E142 4417	234 793E	14237	•
START	STOP	ERASE	COM: BAUD:	CON 921	/13 500		> >

4.功能介绍和测试

我司提供了一个简单的 DEMO 程序,以方便客户测试各种硬件接口功能。演示程序的主要功能包括: IO 端口测试、设置 AP/STA 模式、TCP/MQTT 协议上的数据透明传输。

注意:本节中的以下所有测试都是基于有人物联网提供的 DEMO, 仅供参考。

4.1.开始

\leftarrow	WLAN	
18	JIZUUL-ILJI	
	U200-TEST-5G	
P	USR-EG118_A971 安全	
	输入网络安全密钥	
	<u> </u>	
	下一步	取消

USR-EG118 的默认参数如下表所示。

参数	描述
SSID	EG118_MAC
Password	012345678
IP 地址	192.168.1.1

用户可以通过 Wi-Fi 连接到 USR-EG118, 然后在浏览器输入 192.168.1.1 访问内置网页。

2 ☆)



OSK-LOTTO. LOTTO TOOLWED

 1 EG118_config
 2 EG118_demo
 3 EG118_rs485_WORKFUNC
 4 EG118_device_info

4.2.IO 接口

点击网页上的第一个按钮 "EG118_config",它就会导航到下一页,然后在点击 "eg118_io_config"进入下一页。用户可以通过 该界面上的开关来控制 DO 通断和读取 DI 和 AI。



USR-EG118: 1set DOUT0 ON OFF DOUT1 ON OFF get ai current

1 OUT1_ON 2 OUT1_OFF 3 OUT2_ON 4 OUT_2OFF 5 eg118_ai 6 eg118_di return

4.3.STA 设置

点击网页上的第一个按钮 "EG118_config", 它就会导航到下一页, 然后在点击"wifi_sta"进入下一页, 然后点击 "sta_wifisetting"进入配置页面。并输入 USR-EG118 将连接到的路由器的 SSID 和密码。在此 DEMO 规则下, 名称和密码的长度应该 小于 10 个字符, 不要出现中文。



← C ▲ 不安全 192.168.1.1/PAGE_EG118_CONFIG
: -[EG118_CONFIG]
1 wifi_sta 2 eg118_io_config 3 wifi_softap connect_mode 5 usrioconfig return
← C ▲ 不安全 192.168.1.1/PAGE_wifi_sta_modeBUTTON
sta_wifisetting return
ssid (maxlen 8) : G806s ssid
password (maxlen 8) : 12345678
submit reset 密码

最后点击 "submit"。该设置将在重新启动后生效。为了提高效率,用户可以一次性设置所有参数,然后重新启动。

4.4.添加拓展机

点击网页上的第一个按钮 "EG118_config",它就会导航到下一页,然后在点击 "usrioconfig"进入下一页,然后配置拓展机类型。 配置完点击 "submit"。该设置将在重新启动后生效。

← C ▲ 不安全 192.168.1.1	
USR-EG118: EG118 rootweb	
1 EG118_config 2 EG118_demo 3 EG118_rs485_WORKFUNC 4 EG118_device_info	
: -[EG118_CONFIG]	
1 wifi_sta 2 eg118_io_config 3 wifi_softap connect_mode 5 usrioconfig return	

type of device in slot1 : USR_IO4040 v type of device in slot2 : USR_IO4040 v type of device in slot3 : USR_IO4040 v submit reset

4.5.TCP 模式串口网口透传

单击 "EG118_rs485_WORKFUNC" 按钮,进入下一页, 然后点击 "RS485_TCPDTU",进入下一页,点击 "tcpserversetting" 进入配置页面,设置服务器的 IP 和端口。在 TCP 模式下,设备目前只能在 TCP 客户端模式下工作。

USR-EG118: EG118 rootweb

1 EG118_config 2 EG118_demo 3 EG118_rs485_WORKFUNC 4 EG118_device_info

: -[USR-EG118 RS485_WORKFUNC]

1 RS485AT 2 RS485_MQTT_DTU 3 RS485_TCPDTU 4 th485pe demo return

USR-EG118: SET USR-EG118 work serial date ontransparent transmission MODE over tcp connect when USR-EG118 POWER ON

tcpserversetting return

填写目标服务器的 IP 和端口号,下图示例中填写的是服务器 IP: 172.16.11.221 和服务器端口: 8081。

tcpserverip: 172.16.11.221 tcpserverport: 8081 submit reset

在 TCP 服务器端和串口端监听窗口可以看出,数据可以在串口和 TCP 服务器之间透明传输。



4.6.MQTT 模式 MQTT 串口透传

单击 "EG118_rs485_WORKFUNC " 按钮, 进入下一页, 然后点击 "RS485_MQTT_DTU", 进入下一页, 点击 "mqttserversetting"进入配置页面,设置服务器的 IP 和端口。在 TCP 模式下,设备目前只能在 TCP 客户端模式下工作。

USR-EG118: EG118 rootweb

1 EG118_config 2 EG118_demo 3 EG118_rs485_WORKFUNC 4 EG118_device_info

: -[USR-EG118 RS485_WORKFUNC]

1 RS485AT 2 RS485_MQTT_DTU 3 RS485_TCPDTU 4 th485pe demo return

USR-EG118: SET USR-EG118 work rs485 date transparent transmission MODE over mqtt connect when USR-EG118 POWER ON

mqttserversetting return

在 MQTT 模式下,用户可以设置服务器 IP、用户名和密码来连接 MQTT 服务器进行通信。该设置将在重新启动后生效。用户可以一次性设置所有参数,然后重新启动。

在本测试案例中,我们设置的参数如下:

●服务器 IP: 192.168.2.44

- ●用户名: usr.cn
- ●密码: usr.cn
- ●端口号: 1883

●发布: rs485_to_mqtt

●订阅: mqtt_to_rs485

端口号和发布订阅主题可在 DTU 库文件的代码中修改

>USR-EG118 设置如图:

mqttse	rverip:
192.168.	2.44
mqttse	rveruser:
usr.cn	
mqttsei	rverpass:
usr.cn	
submit	reset

▶数据调试如图:



5.联系方式

- 公 司: 济南有人物联网技术有限公司
- 地 址: 山东省济南市历下区茂岭山三号路中欧校友产业大厦 12、13 层有人物联网
- 网址: http://www.usr.cn
- 用户支持中心: http://im.usr.cn
- 邮 箱: sales@usr.cn
- 电话: 4000-255-652 或者 0531-66592361

有人定位:可靠的智慧工业物联网伙伴 有人愿景:成为工业物联网领域的生态型企业 有人使命:连接价值 价值连接 价值观:天道酬勤 厚德载物 共同成长 积极感恩 产品理念:可靠 易用 价格合理 企业文化:联网的事情找有人

6.免责声明

本文档提供有关 USR-EG118 产品的信息,本文档未授予任何知识产权的许可,并未以明示或暗示,或以禁止发言或其它方式授予任 何知识产权许可。除在其产品的销售条款和条件声明的责任之外,我公司概不承担任何其它责任。并且,我公司对本产品的销售或使用 不作任何明示或暗示的担保,包括对产品的特定用途适用性,适销性或对任何专利权,版权或其它知识产权的侵权责任等均不作担保。 本公司可能随时对产品规格及产品描述做出修改,恕不另行通知。





天猫旗舰店: https://youren.tmall.com 京东旗舰店: https://youren.jd.com 官方网站: www.usr.cn 技术支持工单: im.usr.cn 战略合作联络: ceo@usr.cn 软件合作联络: console@usr.cn 电话: 4000 255 652



关注有人微信公众号

登录商城快速下单

地址:山东省济南市历下区茂岭山三号路中欧校友产业大厦 12、13 层有人物联网