

UT-GM9600

快速入门

日期	作者	版本	说明
2024/12/30	成剑辉、马敏捷	V1.0.1	初版

目录

第一章、产品介绍.....	1
一、数据手册.....	1
1.1 产品简介.....	1
1.1.1 产品概述.....	1
1.1.2 产品图片.....	2
1.1.3 主要参数.....	2
1.1.4 硬件资源.....	3
1.1.5 软件资源.....	4
1.1.6 应用范围.....	4
1.2 电气特性.....	5
1.2.1 工作条件.....	5
1.2.2 电源功率和功率特性.....	5
1.3 接口性能.....	5
1.3.1 系统频率参数.....	5
1.3.2 DI 接口性能参数.....	5
1.3.3 DO 接口性能参数.....	5
1.3.4 以太网接口性能参数.....	6
1.4 电磁兼容性.....	6
1.5 环境适应性.....	6
1.6 产品尺寸.....	7
1.7 已支持配件.....	7

1.8 装箱清单.....	7
二、硬件接口	8
2.1 接口分布.....	8
2.2 接口说明.....	9
2.2.1 电源接口.....	9
2.2.2 RS-232 接口.....	10
2.2.3 RS-485 接口.....	10
2.2.4 DI 接口	11
2.2.5 DO 接口	12
2.2.6 CAN 接口.....	13
2.2.7 AI 接口.....	14
2.2.8 LAN 口	14
2.2.9 USB 接口.....	15
2.2.10 Console 接口	15
2.2.11 HDMI 接口.....	15
2.3 按键.....	16
2.3.1 RST 按键	16
2.4 指示灯.....	16
第二章、快速入门.....	18
一、必备软硬件	18
1.1 必备硬件.....	18
1.2 必备软件.....	18

二、硬件连接	18
2.1 启动配置.....	18
2.2 电源连接.....	18
三、网络登录	19
3.1 IP 地址和密码	19
3.2 SSH 远程登录	19
四、系统接口基本操作.....	19
4.1 DI 的使用.....	19
4.2 DO 的使用.....	20
4.3 AI 的使用	21
4.4 RS232、RS485 串口.....	21
4.5 RTC 的使用.....	23
4.6 有线网络配置.....	23
4.7 WI-FI 网络配置.....	25
4.8 HDMI	26
4.9 LED	26
4.10 查看系统主频.....	27
4.11 查看系统核心温度	27
4.12 USB 设备.....	27
4.13 4G 设备使用方法	28
五、CAN 使用方法	28
5.1 CAN 端口在 down 的状态下可使用以下方法进行配置.....	28

5.2 启用或关闭 CAN	28
5.3 CAN 测试.....	28
六、 镜像更新与烧录.....	29
6.1 解压 update.img	29
6.2 更新镜像文件.....	30
6.3 打包 update.img	32
6.4 烧写固件.....	33

第一章、产品介绍

一、数据手册

1.1 产品简介

1.1.1 产品概述

UT-GM9600 系列，是一款面向工控市场的数字化集中控制器，采用 ARM 架构，搭载 RK3568J 处理器，专为要求独特结合实时处理和通信与应用处理的行业而设计，丰富的接口，优异的性能是产品的显著亮点，板载 4GB 内存，eMMC 存储器 32GB。

UT-GM9600 系列，接口方面拥有包括 4 路千兆以太网，2 路 USB2.0，3 路隔离 CAN2.0 接口，8 路隔离 RS-485，2 路 R-S232，18 路光耦隔离 DI，10 路继电器隔离 DO，4 路 ADC，1 个 M.2 座子 (SATA)，1 个 MiniPCIe 座子 (可选 4G 或 5G redcap)，Wi-Fi 可选，适用于更多行业。

UT-GM9600 系列，符合 EMC3 级设计，广泛应用于电力、石油、化工、交通、工业自动化控制等行业。特别是数字化变电站、分布式能源、发电厂、光伏、风电、储能、动力环境监控、楼宇数字化、数字化工厂等生产场景。

UT-GM9600 系列，产品充分分析了用户需求的多样性，使用场景的差异性，并结合行业的功能需求、技术规范，采用“求同存异”的设计策略，无缝的实现了：数据采集器、PLC 编程控制器、规约转换器、边缘网关等功能。

1.1.2 产品图片



UT-GM9600 前视图



UT-GM9600 后视图

1.1.3 主要参数

处理器	4 核 ARM Cortex-A55 处理器，最高主频 2.0GHz
RAM	4GB
eMMC	32GB
操作系统	Ubuntu20.04
看门狗	支持硬件看门狗
RTC	内置 RTC，RTC 计时功能
掉电保护	支持 5 秒
电源输入	DC 24V(12~36V 宽压供电)
复位按钮	支持设备复位重启

4G/5G	1 路, 支持全网通(4G/5G red cap 可选)
WIFI	1 路, 支持 2.4G/5.8G(可选)
EMC	GB/T17626-2、GB/T17626-3、GB/T17626-4、GB/T17626-5、GB/T17626-6、GB/T17626-8、GB/T17626-9、GB/T17626-10、GB/T17626-12(信号端口 2 级)
外壳	铝合金底板及钣金外壳设计、防护等级 IP40
外形尺寸(LxWxH)	LxWxH=230x114x50mm(不含安装配件)
工作温度	-40°C~+85°C
储存温度	-40°C~+85°C
相对湿度	5%~95%(无凝霜)

1.1.4 硬件资源

RS485	8 路, 光电隔离, 波特率 1200~115200bps
RS232	2 路, 光电隔离, 波特率 1200~115200bps
以太网接口	4 路 10/100/1000M 自适应网口(独立 IP 不同网段, 可支持桥接模式)
USB	2 路 USB2.0 HOST
DI	18 路 DI, 光耦隔离, 1-4 为湿接点, 5-18 为干接点, +9~24V
DO	10 路 DO, 继电器隔离, 常开, 触点阻性负载 3A/5A 250VAC/30VDC
CAN	3 路 CAN2.0, 500kbps 工业级隔离
AI	4xAI 0-20mA, 0-10V, 1%, 采样率 2ksps
TF 卡	1 路, TF 卡插槽, 最大适配 128G
M.2	1 路 M.2, 可支持 SATA SSD 固态硬盘
HDMI	1 路 HDMI 1.4/2.0

1.1.5 软件资源

- Ubuntu 系统;
- eMMC 驱动程序;
- ext4 文件系统;
- SSD 驱动程序;
- HDMI 显示驱动程序, 液晶背光驱动, 触摸屏驱动;
- USBHost 驱动程序, 支持 USB 键盘、USB 鼠标和 U 盘;
- 千兆以太网驱动程序;
- CAN 驱动程序;
- RS-485/RS-232 驱动程序;
- SD 卡驱动程序, 支持热插拔;
- I2C 驱动程序, 包含 RTC 驱动等;
- LED 指示灯驱动程序;
- 4G 模块驱动程序;
- 5G 模块驱动程序;
- 通用数字 I/O 驱动程序;
- 看门狗及复位驱动程序;

1.1.6 应用范围

工业控制

工业网关

1.2 电气特性

1.2.1 工作条件

工作温度	-40°C~+85°C
储存温度	-40°C~+85°C
相对湿度	5%~95%(无凝霜)

1.2.2 电源功率和功率特性

参数	最小	典型	最大	单位	备注
标准工作电压	12	24	36	V	
额定功率	21	TBD	19.8	W	

1.3 接口性能

1.3.1 系统频率参数

名称	最小	典型	最大	单位	说明
系统主频	1.4	1.4	1.8	GHz	根据负载变化

1.3.2 DI 接口性能参数

信号类型	输入电压范围
干节点	/
湿节点	9-24V

1.3.3 DO 接口性能参数

接口类型	触点负载(阻抗)	最大切换电压	最大切换电流	最大切换功率
信号继电器	100mΩ	250V@AC/30V@DC	5A	1250W@AC/ 150W@DC

1.3.4 以太网接口性能参数

参数	测试条件	最小	典型	最大	单位
发速率	单向发	896	TBD	940	Mb/s
收速率	单向收	834	TBD	927	Mb/s

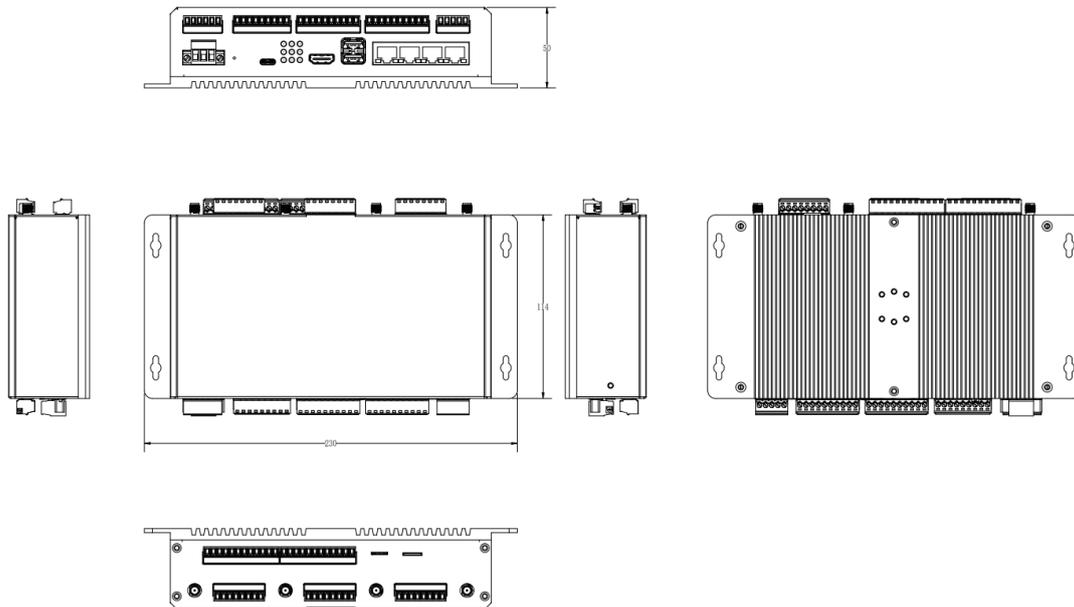
1.4 电磁兼容性

试验项目名称	试验接口	试验标准	试验等级	说明
静电放电抗扰度试验	电源和信号端口	GB/T 17626.2-2006	3	PASS
电快速瞬变脉冲群抗扰度试验	电源和信号端口	GB/T 17626.4-2006	3	PASS
雷击(浪涌)抗扰度试验	电源和信号端口	GB/T 17626.5-2006	3	PASS
射频场感应的传导骚扰抗扰度试验	电源和信号端口	GB/T 17626.6-2006	3	PASS

1.5 环境适应性

试验项目名称	试验标准	试验条件	试验结果
低温启动与运行试验	GB/T 2423.1-2008	-40°C	PASS
高温启动与运行试验	GB/T 2423.2-2008	+85°C	PASS
恒定湿热试验	GB/T 2423.3-2006	+85°C	PASS
正弦震动试验	GB/T 2423.10-2019	不带包装	PASS
自由跌落试验	GB/T 2423.1-2008	带盒	PASS

1.6 产品尺寸



产品尺寸：LxWxH=230x114x50mm(不含安装配件)

1.7 已支持配件

序号	名称	型号	厂商	说明
1	4G module	U9300C	龙尚	可选项
2	5G module	MT5710-CN	鼎桥	可选项

1.8 装箱清单

序号	名称	数量
1	4G/5G 吸盘天线	2
2	双频 WiFi 吸盘天线	2
3	合格证	1
4	保修卡	1

二、硬件接口

2.1 接口分布

如图 1-1、1-2 所示为 UT-GM9600 整机接口分布情况。

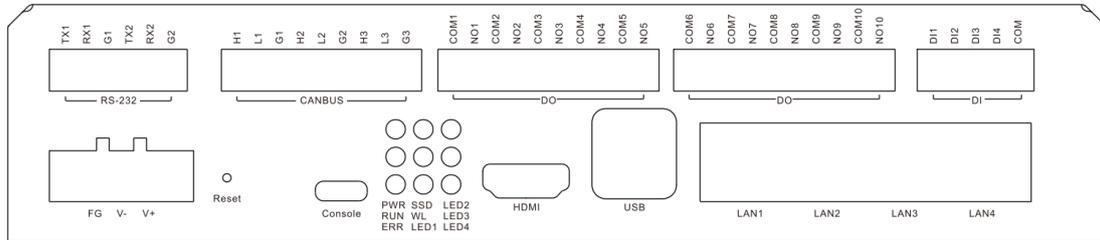


图 1-1 UT-GM9600 前面板接口分布图

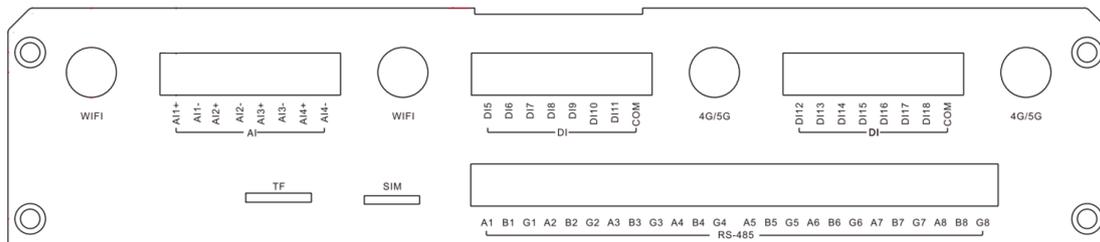


图 1-2 UT-GM9600 后面板接口分布图

UT-GM9600 背面接口如表 1-1 至表 1-2 所示。

表 1-1 UT-GM9600 前面板接口

序号	接口	端子/插座	说明
1	RS232	隔离 RS232 接口	3.81-6Pin
2	CANBUS	隔离 CAN 接口	3.81-9Pin
3	DO	隔离 DO 接口	3.81-10Pin
4	DO	隔离 DO 接口	3.81-10Pin
5	DI	隔离 DI 接口	3.81-4Pin
6	POWER	电源端子插座	5.08-3Pin
7	CONSOLE	调试接口	RJ45
8	HDMI	HDMI 接口	HDMI
9	USB	USB 2.0	Type-A
10	LAN1	10/100/1000Mbps 自适应网口	RJ45

11	LAN2	10/100/1000Mbps 自适应网口	RJ45
12	LAN3	10/100/1000Mbps 自适应网口	RJ45
13	LAN4	10/100/1000Mbps 自适应网口	RJ45

表 1-2 UT-GM9600 后面板接口

序号	接口	端子/插座	说明
1	AI	隔离 AI 接口	3.81-4Pin
2	DI	隔离 DI 接口	3.81-7Pin
3	DI	隔离 DI 接口	3.81-7Pin
4	RS485	隔离 RS485/RS232 接口	3.81-24Pin

2.2 接口说明

2.2.1 电源接口

UT-GM9600 整机使用+12~36VDC 宽压供电，请搭配合适的电源使用。接口物理形式为 3Pin、5.08mm 间距的绿色插座，插座自带螺丝孔，如图 1-3 所示。引脚定义如表 1-3 所示。

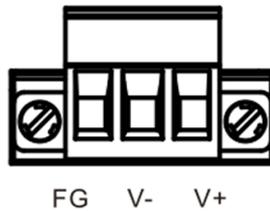


图 1-3 电源接口示意图

表 1-3 电源接口引脚定义

序号	接口	说明
1	FG	保护地
2	V-	系统直流电源地
3	V+	系统直流电源 12V~36V

2.2.2 RS-232 接口

UT-GM9600 提供 2 路隔离 RS-232 接口, 波特率 1200~115200bps, 使用一个 3.81-6Pin 的座子引出, 如图 1-4 所示。引脚定义从右到左如表 1-4 所示。

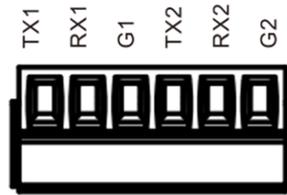


图 1-4 RS232 接口示意图

表 1-4 RS-232 座子引脚信号定义

接口	信号	说明
TX1	第一路 RS-232 发送信号	独立隔离
RX1	第一路 RS-232 接收信号	独立隔离
G1	第一路 RS-232 信号地	独立隔离
TX2	第二路 RS-232 发送信号	独立隔离
RX2	第二路 RS-232 接收信号	独立隔离
G2	第二路 RS-232 信号地	独立隔离

2.2.3 RS-485 接口

UT-GM9600 提供 8 路隔离 RS-485 接口, 波特率 1200~115200bps, 使用两个 3.81-12Pin 的座子引出, 如图 1-5 所示。表 1-5 列出了其中一个座子的信号, 剩余一个座子类似。



图 1-5 RS485 接口示意图

表 1-5 RS485 座子引脚信号定义

接口	信号	说明
A1	第一路 RS485 A 端信号	独立隔离
B1	第一路 RS485 B 端信号	独立隔离
G1	第一路 RS485 信号地	独立隔离
A2	第二路 RS485 A 端信号	独立隔离
B2	第二路 RS485 B 端信号	独立隔离
G2	第二路 RS485 信号地	独立隔离
A3	第三路 RS485 A 端信号	独立隔离
B3	第三路 RS485 B 端信号	独立隔离
G3	第三路 RS485 信号地	独立隔离
A4	第四路 RS485 A 端信号	独立隔离
B4	第四路 RS485 B 端信号	独立隔离
G4	第四路 RS485 信号地	独立隔离

2.2.4 DI 接口

UT-GM9600 提供 18 路光耦隔离 DI 接口，使用一个 3.81-5Pin 及两个 3.81-8Pin 的座子引出，如图 1-6 所示。表 1-6 列出了其中一个座子的信号，剩余一个座子类似。

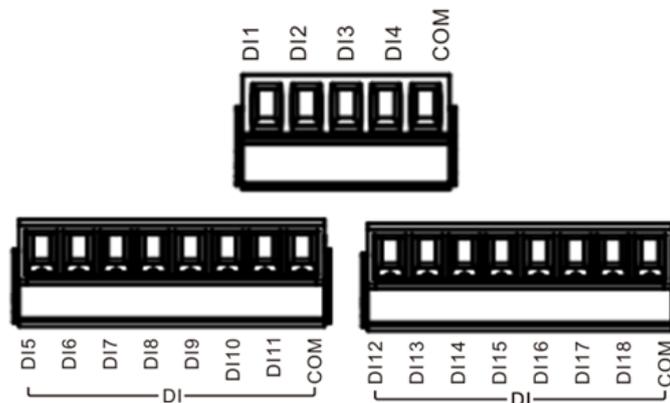


图 1-7 DI 接口示意图

表 1-7 DI 接口信号说明

接口	信号	说明
DI1	DI1 接口	光耦隔离
DI2	DI2 接口	光耦隔离
DI3	DI3 接口	光耦隔离
DI4	DI4 接口	光耦隔离
COM	公共端	光耦隔离

2.2.5 DO 接口

UT-GM9600 提供 10 路继电器常开输出接口，隔离触点负载（阻性）3A/5A/250VAC/30VDC，最大切换电压 250VAC/30VDC，最大切换电流 5A。使用两个 3.81-10Pin 的座子引出，如图 1-7 所示。表 1-7 列出了其中一个座子的信号，剩余一个座子类似。

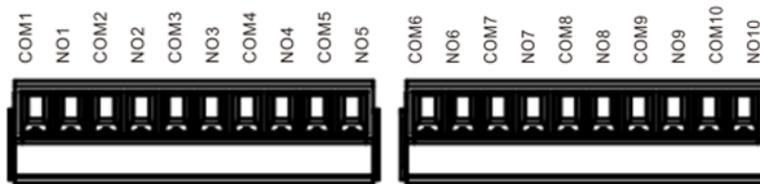


图 1-7 DO 接口示意图

表 1-7 DO 接口信号说明

接口	信号	说明
COM1	公共端 1	隔离触点负载(阻性)3A/5A/250VAC/30VDC
NO1	DO1 常开接口	隔离触点负载(阻性)3A/5A/250VAC/30VDC
COM2	公共端 2	隔离触点负载(阻性)3A/5A/250VAC/30VDC
NO2	DO2 常开接口	隔离触点负载(阻性)3A/5A/250VAC/30VDC
COM3	公共端 3	隔离触点负载(阻性)3A/5A/250VAC/30VDC
NO3	DO3 常开接口	隔离触点负载(阻性)3A/5A/250VAC/30VDC
COM4	公共端 4	隔离触点负载(阻性)3A/5A/250VAC/30VDC
NO4	DO4 常开接口	隔离触点负载(阻性)3A/5A/250VAC/30VDC

COM5	公共端 5	隔离触点负载(阻性)3A/5A/250VAC/30VDC
NO5	DO5 常开接口	隔离触点负载(阻性)3A/5A/250VAC/30VDC

2.2.6 CAN 接口

UT-GM9600 具有 3 路 CAN2.0 B 接口，500kbps 工业级隔离。使用一个 3.81-9Pin 的座子引出，如图 1-8 所示。引脚定义从左到右如表 1-8 所示。

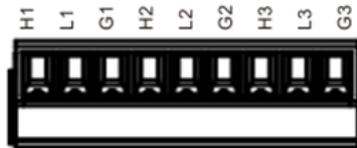


图 1-8 CAN 接口示意图

表 1-8 CAN 接口引脚定义

接口	信号	说明
H1	H 信号线 1	CAN2.0 B, 500kbps 工业级隔离
L1	L 信号线 1	CAN2.0 B, 500kbps 工业级隔离
G1	CAN1 隔离地	CAN2.0 B, 500kbps 工业级隔离
H2	H 信号线 2	CAN2.0 B, 500kbps 工业级隔离
L2	L 信号线 2	CAN2.0 B, 500kbps 工业级隔离
G2	CAN2 隔离地	CAN2.0 B, 500kbps 工业级隔离
H3	H 信号线 3	CAN2.0 B, 500kbps 工业级隔离
L3	L 信号线 3	CAN2.0 B, 500kbps 工业级隔离
G3	CAN3 隔离地	CAN2.0 B, 500kbps 工业级隔离

2.2.7 AI 接口

UT-GM9600 提供 4 路 AI 接口, 0-20mA, 0-10V, 1%, 采样率 2ksps。使用一个 3.81-8Pin 的座子引出, 如图 1-9 所示。引脚定义从左到右如表 1-9 所示。

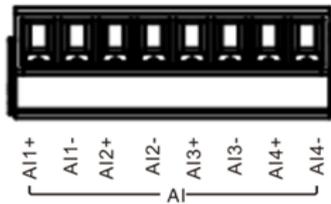


图 1-9 AI 接口示意图

表 1-9 AI 接口信号说明

接口	信号	说明
AI1+	AI1 接口+	0-20mA, 0-10V, 1%, 采样率 2ksps
AI1-	AI1 接口-	0-20mA, 0-10V, 1%, 采样率 2ksps
AI2+	AI2 接口+	0-20mA, 0-10V, 1%, 采样率 2ksps
AI2-	AI2 接口-	0-20mA, 0-10V, 1%, 采样率 2ksps
AI3+	AI3 接口+	0-20mA, 0-10V, 1%, 采样率 2ksps
AI3-	AI3 接口-	0-20mA, 0-10V, 1%, 采样率 2ksps
AI4+	AI4 接口+	0-20mA, 0-10V, 1%, 采样率 2ksps
AI4-	AI4 接口-	0-20mA, 0-10V, 1%, 采样率 2ksps

2.2.8 LAN 口

UT-GM9600 提供 4 路以太网接口, 物理接口使用 RJ45, 出厂固件设置默认 4 路网口均支持 10/100/1000Mbps 自适应, 推荐使用 1000Mbps 工作模式以获得最好的性能。网口号分布如图 1-10 所示。网口通道说明如表 1-10 所示。

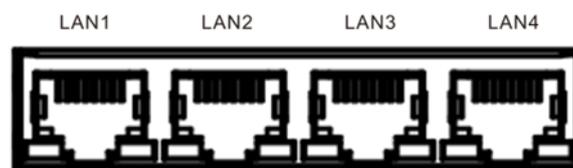


图 1-10 网口号分布图

表 1-10 网口通道说明

网口标号	网络总线	说明
LAN1	千兆以太网 1	独立 IP, 10/100/1000Mbps
LAN2	千兆以太网 2	独立 IP, 10/100/1000Mbps
LAN3	千兆以太网 3	独立 IP, 10/100/1000Mbps
LAN4	千兆以太网 4	独立 IP, 10/100/1000Mbps

2.2.9 USB 接口

UT-GM9600 提供 2 路标准 USB 2.0 TYPE-A 接口, 如图 1-11 所示。



图 1-11 USB 接口示意图

2.2.10 Console 接口

UT-GM9600 提供 1 路 USB TYPE-C 接口, 为调试接口, 如图 1-12 所示。



图 1-12 Console 接口示意图

2.2.11 HDMI 接口

UT-GM9600 提供 1 路 HDMI 显示接口, 如图 1-13 所示。



图 1-13 HDMI 接口示意图

2.3 按键

2.3.1 RST 按键

UT-GM9600 有 1 个 RST 复位按键，短按复位，隐藏式，如图 1-14 所示。



图 1-14 RST 按键示意图

2.4 指示灯

UT-GM9600 中有 9 个指示灯，分别为电源指示灯、SSD 运行状态灯、系统运行状态灯、4G 指示灯/WIFI 指示灯、ERR 错误指示灯、LED1 状态灯、LED2 状态灯、LED3 状态灯、LED4 状态灯，如图 1-15 所示。方便在任何的情况下，了解系统的运行状况，指示灯具体描述如表 1-15 所示。

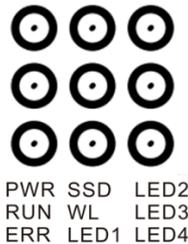


图 1-15 指示灯示意图

表 1-15 指示灯具体描述

标识	功能	说明
PWR	电源指示灯，绿色	亮：设备正常上电 灭：设备断电
SSD	SSD 状态灯，绿色	亮：SSD 运行正常 灭：SSD 未运行
RUN	系统运行状态灯，绿色	闪：系统运行正常 灭：系统故障或未运行
WL	4G 指示灯/WIFI 指示灯	亮：4G 拨号成功/WIFI 连接正常 灭：4G 未拨号成功/WIFI 未连接
ERR	错误指示灯	亮：运行异常 灭：运行正常
LED1	LED1 状态灯，绿色	闪/灭：用户自定义

LED2	LED2 状态灯, 绿色	闪/灭: 用户自定义
LED3	LED3 状态灯, 绿色	闪/灭: 用户自定义
LED4	LED4 状态灯, 绿色	闪/灭: 用户自定义

第二章、快速入门

一、必备软硬件

1.1 必备硬件

- 1) UT-GM9600 整机
- 2) 调试电脑主机

1.2 必备软件

- 串口终端软件：超级终端、Tera Term 或者 putty 等
- 串口终端波特率等设置：1500000, 8N1
- SSH 软件：putty、SSH Security Shell Client 等

二、硬件连接

2.1 启动配置

默认为 eMMC 启动。

2.2 电源连接

UT-GM9600 的电源供应：DC12-36V

三、网络登录

3.1 IP 地址和密码

默认 IP: lan1: 192.168.1.2、lan2: 192.168.2.2、

lan3: 192.168.3.2、lan4: 192.168.4.2

用户名: utek, 密码: admin

3.2 SSH 远程登录

网线接入 lan1、lan2、lan3、lan4, 调试终端选择 SSH 登录。

四、系统接口基本操作

4.1 DI 的使用

UT-GM9600 共有 18 路 DI, DI 设备名称路径和操作方法如下表。

表 2-1 DI 与 GPIO 的对应关系表

设备名称	路径	操作方法
DI1	/sys/class/dido/di/di1	cat /sys/class/dido/di/di1
DI2	/sys/class/dido/di/di2	cat /sys/class/dido/di/di2
DI3	/sys/class/dido/di/di3	cat /sys/class/dido/di/di3
DI4	/sys/class/dido/di/di4	cat /sys/class/dido/di/di4
DI5	/sys/class/dido/di/di5	cat /sys/class/dido/di/di5
DI6	/sys/class/dido/di/di6	cat /sys/class/dido/di/di6
DI7	/sys/class/dido/di/di7	cat /sys/class/dido/di/di7
DI8	/sys/class/dido/di/di8	cat /sys/class/dido/di/di8
DI9	/sys/class/dido/di/di9	cat /sys/class/dido/di/di9
DI10	/sys/class/dido/di/di10	cat /sys/class/dido/di/di10
DI11	/sys/class/dido/di/di11	cat /sys/class/dido/di/di11

DI12	/sys/class/dido/di/di12	cat /sys/class/dido/di/di12
DI13	/sys/class/dido/di/di13	cat /sys/class/dido/di/di13
DI14	/sys/class/dido/di/di14	cat /sys/class/dido/di/di14
DI15	/sys/class/dido/di/di15	cat /sys/class/dido/di/di15
DI16	/sys/class/dido/di/di16	cat /sys/class/dido/di/di16
DI17	/sys/class/dido/di/di17	cat /sys/class/dido/di/di17
DI18	/sys/class/dido/di/di18	cat /sys/class/dido/di/di18

4.2 DO 的使用

UT- GM9600 共有 10 路 DO，设备名称路径和操作方法如下表。

表 2-2 DO 与 GPIO 的对应关系表

设备名称	路径	操作方法
DO1	/sys/class/dido/do/do1	开: echo 1 > /sys/class/dido/do/do1 关: echo 0 > /sys/class/dido/do/do1
DO2	/sys/class/dido/do/do2	开: echo 1 > /sys/class/dido/do/do2 关: echo 0 > /sys/class/dido/do/do2
DO3	/sys/class/dido/do/do3	开: echo 1 > /sys/class/dido/do/do3 关: echo 0 > /sys/class/dido/do/do3
DO4	/sys/class/dido/do/do4	开: echo 1 > /sys/class/dido/do/do4 关: echo 0 > /sys/class/dido/do/do4
DO5	/sys/class/dido/do/do5	开: echo 1 > /sys/class/dido/do/do5 关: echo 0 > /sys/class/dido/do/do5
DO6	/sys/class/dido/do/do6	开: echo 1 > /sys/class/dido/do/do6 关: echo 0 > /sys/class/dido/do/do6
DO7	/sys/class/dido/do/do7	开: echo 1 > /sys/class/dido/do/do7 关: echo 0 > /sys/class/dido/do/do7
DO8	/sys/class/dido/do/do8	开: echo 1 > /sys/class/dido/do/do8 关: echo 0 > /sys/class/dido/do/do8
DO9	/sys/class/dido/do/do9	开: echo 1 > /sys/class/dido/do/do9 关: echo 0 > /sys/class/dido/do/do9
DO10	/sys/class/dido/do/do10	开: echo 1 > /sys/class/dido/do/do10 关: echo 0 > /sys/class/dido/do/do10

4.3 AI 的使用

UT-GM9600 共有 4 路 AI，采集电压量程 4~20mA/0~12V，采集精度 1%，采集速率 125SPS，AI 的对应路径和操作如表 2-3 所示。注意：开机默认位电压模式，如需要电流模式请自行修改。

表 2-3 AI 的对应路径表

设备名称	路径	操作方法
AI1	/sys/class/adc/shc6248/ch1_current 和 ch1_voltage	读电压: cat ch1_voltage 读电流: cat ch1_current
AI2	/sys/class/adc/shc6248/ch2_current 和 ch2_voltage	读电压: cat ch2_voltage 读电流: cat ch2_current
AI3	/sys/class/adc/shc6248/ch3_current 和 ch3_voltage	读电压: cat ch3_voltage 读电流: cat ch3_current
AI4	/sys/class/adc/shc6248/ch4_current 和 ch4_voltage	读电压: cat ch4_voltage 读电流: cat ch4_current
ch1_mode	/sys/class/adc/shc6248/ch1_mode	电流模式: echo 0 > ch1_mode 电压模式: echo 1 > ch1_mode
ch2_mode	/sys/class/adc/shc6248/ch2_mode	电流模式: echo 0 > ch2_mode 电压模式: echo 1 > ch2_mode
ch3_mode	/sys/class/adc/shc6248/ch3_mode	电流模式: echo 0 > ch3_mode 电压模式: echo 1 > ch3_mode
Ch4_mode	/sys/class/adc/shc6248/ch4_mode	电流模式: echo 0 > ch4_mode 电压模式: echo 1 > ch4_mode

4.4 RS232、RS485 串口

UT-GM9600 共有 2 路 RS-232，8 路 RS-485 接口,配置设备路径如下表。

RS232 是全双工，测试时 TX 短接 RX 即可收发测试，RS485 是半双工，同一时刻只能发送或者接收，测试时可以互相短接，例如 rs485-1 的 A1 接 rs485-2 的 A2，B1 接 B2 进行测试，测试完可以使用 killall uart-receive 关掉接收进程。

表 2-4 RS232、RS485 对应路径表

设备名称	路径	操作方法
rs232-1	/dev/rs232-1	接收: uart-receive 9600 /dev/rs232-1 & 发送: uart-send 9600 /dev/rs232-1 "hello world!"
rs232-2	/dev/rs232-2	接收: uart-receive 9600 /dev/rs232-2 & 发送: uart-send 9600 /dev/rs232-2 "hello world!"
rs485-1	/dev/rs485-1	接收: uart-receive 9600 /dev/rs485-1 & 发送: uart-send 9600 /dev/rs485-1 "hello world!"
rs485-2	/dev/rs485-2	接收: uart-receive 9600 /dev/rs485-2 & 发送: uart-send 9600 /dev/rs485-2 "hello world!"
rs485-3	/dev/rs485-3	接收: uart-receive 9600 /dev/rs485-3 & 发送: uart-send 9600 /dev/rs485-3 "hello world!"
rs485-4	/dev/rs485-4	接收: uart-receive 9600 /dev/rs485-4 & 发送: uart-send 9600 /dev/rs485-4 "hello world!"
rs485-5	/dev/rs485-5	接收: uart-receive 9600 /dev/rs485-5 & 发送: uart-send 9600 /dev/rs485-5 "hello world!"
rs485-6	/dev/rs485-6	接收: uart-receive 9600 /dev/rs485-6 & 发送: uart-send 9600 /dev/rs485-6 "hello world!"
rs485-7	/dev/rs485-7	接收: uart-receive 9600 /dev/rs485-7 & 发送: uart-send 9600 /dev/rs485-7 "hello world!"
rs485-8	/dev/rs485-8	接收: uart-receive 9600 /dev/rs485-8 & 发送: uart-send 9600 /dev/rs485-8 "hello world!"

4.5 RTC 的使用

hwclock -r: 显示 RTC 时间

hwclock -w: 将系统时间写到 RTC

hwclock -s: 将 RTC 时间设置为系统时间

4.6 有线网络配置

UT- GM9600 当前系统里已进行了简单的配置，开机即用。具体的 IP 地址如表 2-5 所示。

表 2-5 网口对应 IP 地址表

网络接口	IP 地址
lan1	192.168.1.2
lan2	192.168.2.2
lan3	192.168.3.2
lan4	192.168.4.2

(1) 临时修改 ip: ifconfig 配置，系统重启后配置不生效。

查看全部网卡信息：

```
$ ifconfig -a
```

查看指定网卡信息：

```
$ ifconfig lan1
```

启动/禁用指定网卡：

```
$ ifconfig lan1 up
$ ifconfig lan1 down
```

配置 IP、子网掩码、广播地址

```
$ ifconfig lan1 192.168.1.123 #配置 lan1 IP 地址
$ ifconfig lan1 192.168.1.123/24 #lan1 IP 地址和子网掩码 255.255.255.0
$ ifconfig lan1 192.168.1.123/24 broadcast 192.168.1.255 #配置 lan1 IP 地
```

址、子网掩码和广播地址

启用/禁用广播功能

```
$ ifconfig lan1 broadcast 192.168.1.255 #启用 lan1 广播功能
$ ifconfig lan1 -broadcast #禁用 lan1 广播功能
```

配置最大传输单元

```
$ ifconfig lan1 mtu 1500 #设置 lan1 能通过的最大数据包大小为 1500 bytes
```

开启/关闭混杂模式

```
$ ifconfig lan1 promisc #开启混杂模式
$ ifconfig lan1 -promisc #关闭混杂模式
```

(2) 永久修改 ip: netplan 配置

可以通过 netplan 指令配置网络配置。

打开 IP 配置文件:

```
$ sudo vi /etc/netplan/01-network-manager-all.yaml
```

按下 “i” 键进入编辑模式, 修改 ip 地址等参数, 以下为默认网络配置:

```
network:
  #配置版本, 一般都是 2
  version: 2
  #指定网络管理软件
  renderer: networkd
  #以太网配置
  ethernets:
    #指定网卡名称
    lan1:
      #DHCP 配置, 配置为 yes 则无需配置静态 IP 地址和子网掩码
      dhcp4: yes
    lan2:
      dhcp4: no
      #指定 IP 地址和子网掩码
      addresses:
        - 192.168.2.2/24
      #网关配置
      gateway4: 192.168.2.1
      #DNS 配置
      nameservers:
        addresses: [8.8.8.8]
    lan3:
```

```

dhcp4: no
addresses:
  - 192.168.3.2/24
gateway4: 192.168.3.1
nameservers:
  addresses: [8.8.8.8]
lan4:
dhcp4: no
addresses:
  - 192.168.4.2/24
gateway4: 192.168.4.1
nameservers:
  addresses: [8.8.8.8]

```

根据需要修改网络参数，修改完成后敲 Esc 键退出编辑模式，输入冒号 “:”，然后输入 “wq” 保存并退出，然后输入以下指令或重启系统使配置生效：

```
$ sudo netplan apply
```

注意：yaml 文件需要注意格式规范，缩进只能用 2 个空格，不能使用 Tab，冒号后边如果还要文本，需要一个空格，不然会导致配置不生效，建议修改前先备份。

4.7 WI-FI 网络配置

运行 ip addr 命令就会看到 wlan0 的网络接口

```

$ ip addr
...
10: wlan0: <NO-CARRIER,BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc
pfifo_fast state DORMANT group default qlen 1000
    link/ether 70:f7:54:ee:4f:0d brd ff:ff:ff:ff:ff:ff
...

```

此时可使用 NetworkManager 来管理无线网络连接，以下列出 nmcli 的常用 WIFI 配

置命令

```

# 列出 WIFI 列表
nmcli device wifi list

# 连接无密码网络，<ssid>是连接 WIFI 的 SSID
nmcli device wifi connect <ssid>

# 连接有密码网络，<ssid>是连接 WIFI 的 SSID，<psk>是连接 WIFI 的密码

```

```
nmcli device wifi connect <ssid> password <psk>
# 查看是否连接成功
nmcli device status
```

若 WIFI 模块支持 STA，可使用以下命令配置热点

```
# 创建 connection
# 以下命令中的 WIFI_NAME、WIFI_SSID、WIFI_PSK，可根据需求修改
nmcli device wifi hotspot ifname wlan0 con-name WIFI_NAME ssid
WIFI_SSID
password WIFI_PSK
nmcli connection modify WIFI_NAME connection.autoconnect yes
```

4.8 HDMI

即插即用。

4.9 LED

系统板载 9 颗 LED，分别是：

1	电源	2	run	3	User-led1~User-led4
4	4G/wifi	5	硬盘	6	5G

1、PWR

PWR 灯为系统“电源”指示灯，常亮为绿色。当 UT-GM9600 接上电源后就会亮起。

2、RUN

RUN 灯为系统的“运行”指示灯，点亮时为绿色。默认情况下，系统上电时为闪烁状态。如果 RUN 灯上电后是灭，说明系统加载异常，需要断电重启系统。

3、User-led

系统有 4 个 user-led1~user-led4，用户可自定义。

操作方法：echo 1 > /sys/class/leds/usr-led1/brightness 亮

操作方法：echo 0 > /sys/class/leds/usr-led1/brightness 灭

4、4G/wifi

4G 和 wifi 共用一个灯，4G 拨号成功或 wifi 有连接时绿灯常亮。

5、硬盘

SSD 正常（如有）工作绿灯闪烁。

6、错误

出现错误绿灯常亮。

4.10 查看系统主频

核心板主频可用如下命令查看：

```
root@rk3568~# cat
/sys/devices/system/cpu/cpufreq/policy0/scaling_cur_freq
408000
```

4.11 查看系统核心温度

核心温度可用如下命令查看：

```
root@rk3568:~# cat /sys/class/thermal/thermal_zone*/temp
44375
43750
```

以上查询到的温度结果除以 1000 则为对应的温度，单位为摄氏度，如 42500 的对应温度为 42.5 摄氏度。

4.12 USB 设备

即插即用。U 盘名称：/dev/sda*。

4.13 4G 设备使用方法

UT-GM9600 上的 4G 模块使用流程如下：拨号方法 longsung-CM 1 &。

五、CAN 使用方法

UT-GM9600 支持三路原生 CAN, 原生 CAN 口支持 CAN 2.0B, 支持波特率 5K、10K、100K、125K、250K、500K, 下面介绍原生 CAN 的配置及测试方法。

5.1 CAN 端口在 down 的状态下可使用以下方法进行配置

- 把 CAN 配置为 CAN2.0, 并设置波特率:

```
# /sbin/ip link set can0 type can bitrate 1000000
# /sbin/ip link set can1 type can bitrate 1000000
# /sbin/ip link set can2 type can bitrate 1000000
```

5.2 启用或关闭 CAN

- 启动 CAN

```
# ifconfig can0 up
# ifconfig can1 up
# ifconfig can2 up
```

- 关闭 CAN

```
# ifconfig can0 down
# ifconfig can1 down
# ifconfig can2 down
```

5.3 CAN 测试

在确保网络连通和 Ubuntu 源可用情况下, 可以安装 can-utils 来测试 CAN。

```
# sudo apt-get update
# sudo apt-get install can-utils
```

CAN 常用的测试工具是 candump、cansend、cangen, 下面简单介绍这三个工具的

用法:

candump 一般用于 CAN 帧接收, 如:

```
# candump can0
```

cansend 和 cangen 一般用于 CAN 帧发送, 如:

```
# cangen can0 -D 1122334455667788 -L 8
```

或

```
# cansend can0 123#1122334455667788
```

六、镜像更新与烧录

本章将举例详细介绍 update.img 镜像更新与烧录过程。

将 afptool、rkImageMaker、rk356x-mkupdate.sh、unpack.sh、update.img (镜像文件名字以实际为准) 复制到 linux 系统目录, 本例子路径如下:

```
root@ubuntu:/prj2/test# ls
afptool rk356x-mkupdate.sh rkImageMaker unpack.sh update.img
root@ubuntu:/prj2/test#
```

6.1 解压 update.img

执行以下指令解压 update.img 镜像文件:

```
#解压 update.img 文件

./unpack.sh update.img
```

```

root@ubuntu:/prj2/test# ./unpack.sh update.img
*****rkImageMaker ver 2.23*****
Unpacking image, please wait...
Exporting boot.bin
Exporting firmware.img
Unpacking image success.
Android Firmware Package Tool v2.27
Check file... OK
----- UNPACK -----
out/package-file      offset=0x800      size=0xF1
out/parameter.txt    offset=0x1000    size=0x209
out/MiniLoaderAll.bin offset=0x1800    size=0x6F9C0
out/uboot.img        offset=0x71800   size=0x400000
out/misc.img         offset=0x471800  size=0xC000
out/boot.img         offset=0x47D800  size=0x22CBE00
out/recovery.img     offset=0x2749800 size=0x2925C00
out/rootfs.img       offset=0x506F800 size=0x5B400000
out/oem.img          offset=0x6046F800 size=0x10A6000
out/grems.img        offset=0x61515800 size=0x403000
out/userdata.img     offset=0x61918800 size=0x446000
Unpack firmware OK!
----- OK -----
root@ubuntu:/prj2/test# █

```

解压完成生成的文件在 out 目录下，如图所示：

```

root@ubuntu:/prj2/test# ls out/
boot.bin  firmware.img  MiniLoaderAll.bin  oem.img  parameter.txt  rootfs.img  userdata.img
boot.img  grems.img     misc.img           package-file  recovery.img  uboot.img
root@ubuntu:/prj2/test# █

```

6.2 更新镜像文件

out 目录下有各种 img 文件，以下以修改根文件系统为例。

例如添加配置文件，改动配置文件：

```

#创建目录用来挂载 rootfs.img
mkdir rootfs

#挂载
sudo mount out/rootfs.img rootfs/

#查看是否挂载成功
df -h

```

```

root@ubuntu:/prj2/test# df -h
Filesystem      Size  Used Avail Use% Mounted on
udev            956M   0  956M   0% /dev
tmpfs           198M  24M  174M  12% /run
/dev/sda1       39G   29G   7.8G  79% /
tmpfs           986M  192K  985M   1% /dev/shm
tmpfs           5.0M   0   5.0M   0% /run/lock
tmpfs           986M   0   986M   0% /sys/fs/cgroup
/dev/sdb        40G   20G   18G  53% /prj2
tmpfs           198M  28K  197M   1% /run/user/108
tmpfs           198M   0   198M   0% /run/user/0
/dev/loop0      1.4G  1.2G  106M  92% /prj2/test/rootfs
    
```

直接去挂载目录下添加新的配置文件:

```

#跳转到挂载文件夹 etc 目录下
cd rootfs/etc

#在 etc 目录下创建一个测试文件
touch update-test
    
```

创建完成如下:

```

root@ubuntu:/prj2/test/rootfs/etc# ls
alsa                factory             inittab             localtime           pm                  socketcand.conf
bash.bashrc         fonts              input-event-daemon.conf mke2fs.conf        profile            ssh
bash_completion.d  fstab              input-event-daemon.conf.d mtab                profile.d          ssl
bindresvport.blacklist generate_logs.d     inputrc             my.cnf              protocols          timezone
collectd.conf       group              iproute2            mysql               pulse              udev
cups                hostapd            iptables.conf       netconfig           rc_keymaps        update-test
dbus-1              hostapd.accept    irqbalance.d        network             rc_local           usbmount
dhcpcd.conf         hostapd.conf       issue               nfsmount.conf       rc_maps.cfg       version
dnsmasq.conf       hostapd.deny      ld.so.cache         nsswitch.conf       resolv.conf       watchdog.conf
dropbear            hostname           libinput            ntp.conf            services          watchdogd.conf
ethertypes          hosts              libnl               os-release          shadow            wpa_supplicant.conf
exports             init.d             libnl               passwd              shells             xdg
    
```

创建完成取消挂载根文件系统:

```

#退出 rootfs/目录
cd ../../

#在/etc 目录下创建一个测试文件
sudo umount rootfs/
    
```

```

root@ubuntu:/prj2/test/rootfs/etc# cd ../../
root@ubuntu:/prj2/test# sudo umount rootfs/
root@ubuntu:/prj2/test# df -h
Filesystem      Size  Used Avail Use% Mounted on
udev            956M   0  956M   0% /dev
tmpfs           198M  24M  174M  12% /run
/dev/sda1       39G   29G   7.8G  79% /
tmpfs           986M  192K  985M   1% /dev/shm
tmpfs           5.0M   0   5.0M   0% /run/lock
tmpfs           986M   0   986M   0% /sys/fs/cgroup
/dev/sdb        40G   20G   18G  53% /prj2
tmpfs           198M  28K  197M   1% /run/user/108
tmpfs           198M   0   198M   0% /run/user/0
root@ubuntu:/prj2/test#

```

6.3 打包 update.img

执行以下指令打包 update.img 镜像文件:

```
#执行指令打包 update.img 文件
./rk356x-mkupdate.sh
```

打包成功如下所示:

```

root@ubuntu:/prj2/test# ./rk356x-mkupdate.sh
start to make update.img...
Android Firmware Package Tool v2.27
----- PACKAGE -----
Add file: out/package-file
package-file,Add file: out/package-file done,offset=0x800,size=0xf1,userspace=0x1
Add file: out/parameter.txt
parameter,Add file: out/parameter.txt done,offset=0x1000,size=0x215,userspace=0x1,flash_address=0x00000000
Add file: out/MiniLoaderAll.bin
bootloader,Add file: out/MiniLoaderAll.bin done,offset=0x1800,size=0x6f9c0,userspace=0xe0
Add file: out/uboot.img
uboot,Add file: out/uboot.img done,offset=0x71800,size=0x400000,userspace=0x800,flash_address=0x00004000
Add file: out/misc.img
misc,Add file: out/misc.img done,offset=0x471800,size=0xc000,userspace=0x18,flash_address=0x00006000
Add file: out/boot.img
boot,Add file: out/boot.img done,offset=0x47d800,size=0x22cbe00,userspace=0x4598,flash_address=0x00008000
Add file: out/recovery.img
recovery,Add file: out/recovery.img done,offset=0x2749800,size=0x2925c00,userspace=0x524c,flash_address=0x00028000
Add file: out/rootfs.img
rootfs,Add file: out/rootfs.img done,offset=0x506f800,size=0x5b400000,userspace=0xb6800,flash_address=0x00078000
Add file: out/oem.img
oem,Add file: out/oem.img done,offset=0x6046f800,size=0x10a6000,userspace=0x214c,flash_address=0x00878000
Add file: out/grems.img
grems,Add file: out/grems.img done,offset=0x61518000,size=0x403000,userspace=0x806,flash_address=0x008b8000
Add file: out/userdata.img
userdata,Add file: out/userdata.img done,offset=0x61918800,size=0x446000,userspace=0x88c,flash_address=0x010b8000
Add CRC...
Make firmware OK!
----- OK -----
*****rkImageMaker ver 2.23*****
Generating new image, please wait...
Writing head info...
Writing boot file...
Writing firmware...
Generating MD5 data...
MD5 data generated successfully!
New image generated successfully!
Making out/update.img OK.

```

在 test 目录下的 update.img 就是更新后的镜像固件

```

root@ubuntu:/prj2/test# ls
afptool  out  rk356x-mkupdate.sh  rkImageMaker  unpack.sh  update.img
root@ubuntu:/prj2/test#

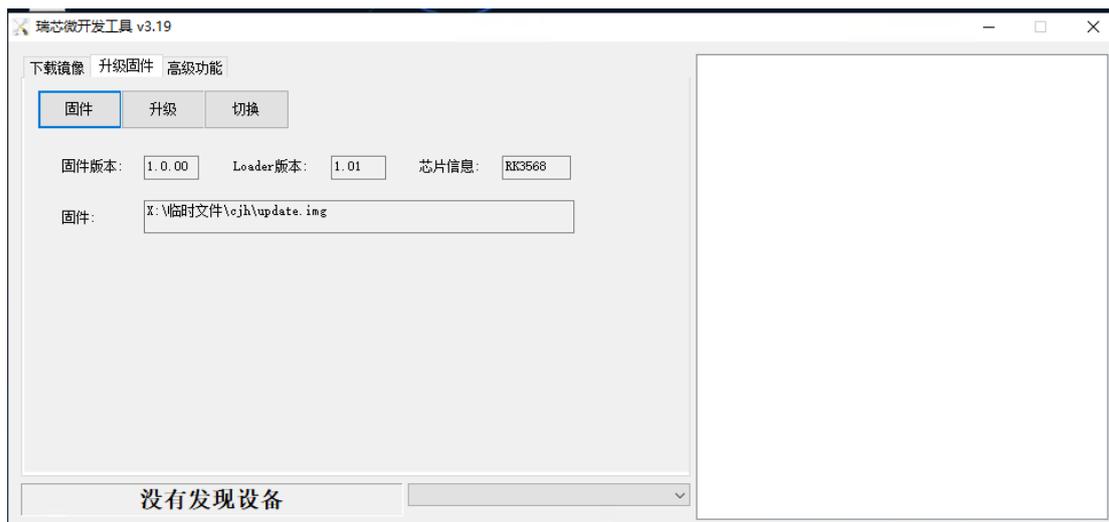
```

6.4 烧写固件

在 windows 系统下安装好瑞芯微的官方 RKTool 驱动，运行 RKDevTool.exe：

bin	2017/9/11 15:07	文件夹	
Language	2017/9/11 15:07	文件夹	
Log	2024/12/2 15:51	文件夹	
config.cfg	2024/12/2 15:51	CFG 文件	6 KB
config	2023/5/4 16:56	配置设置	2 KB
revision	2023/9/11 11:01	文本文档	4 KB
RKDevTool	2023/9/11 11:00	应用程序	1,222 KB
updateimg.cfg	2024/7/16 16:27	CFG 文件	6 KB
开发工具使用文档_v1.0	2021/8/27 10:28	Microsoft Edge ...	450 KB

选择“升级固件”选项卡，然后“固件”选项，在弹出的固件窗口选择更新后的 update.img 固件，如下图所示：



设备上需要拨码并重启设备使设备进入 LOADER 模式：



进入 LOADER 模式后点击“升级”，即可开始升级镜像。烧写过程中，烧写工具右边会有对应的打印信息，如下图所示：



烧写完成，烧写工具会提示下载固件成功，设备会自动重启。