

PET-Q13R3568产品规格书

13.3寸安卓工控一体机



一、产品特点

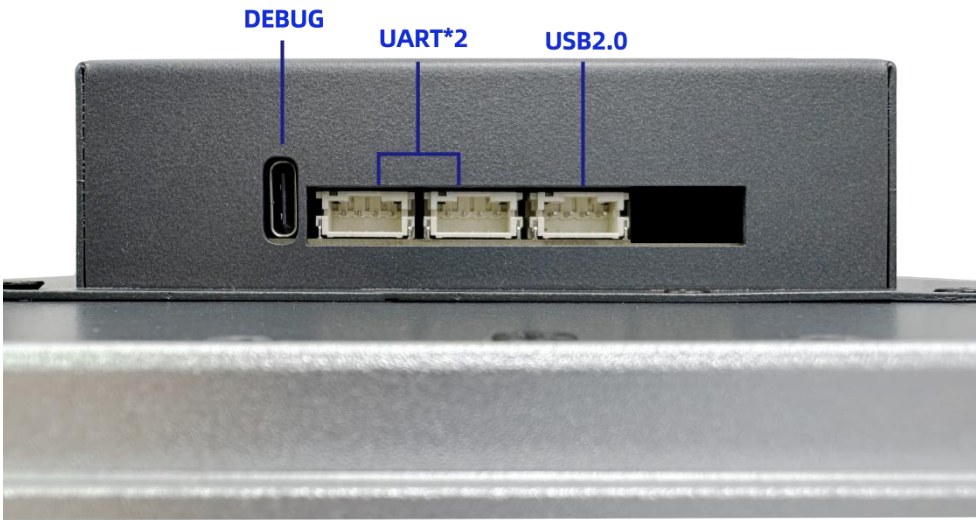
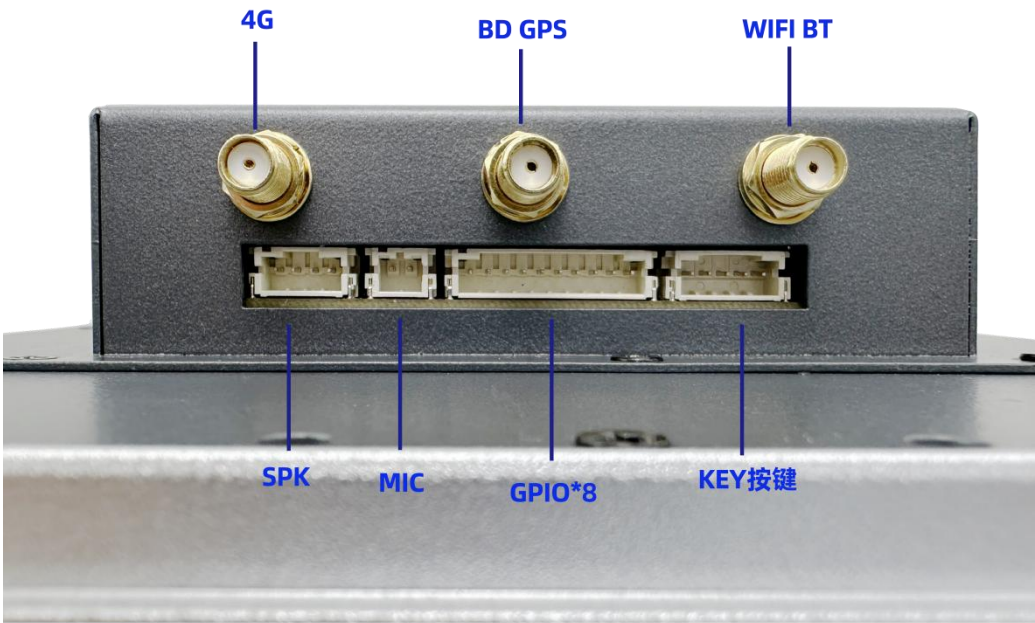
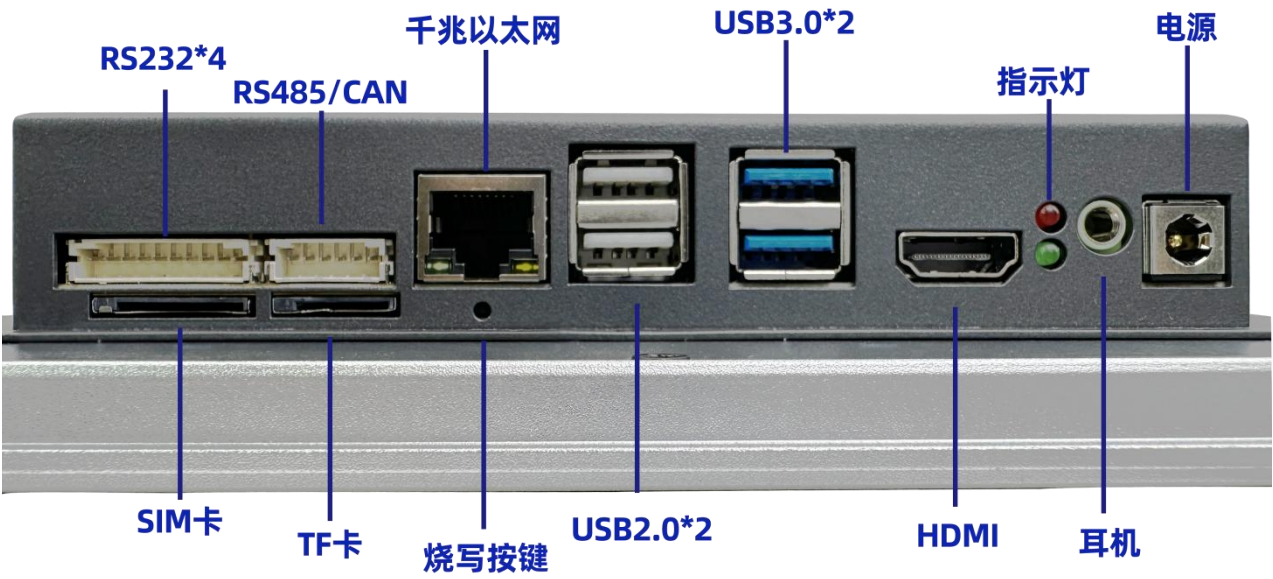
- RK3568处理器
- 电容式触摸， 触摸精准灵敏
- 内外嵌式安装、壁挂式安装
- 超窄边框， 嵌入后与机柜融为一体
- 7X24小时稳定运行



内外嵌式安装

多种效果安装方式
嵌入后与机柜融为一体

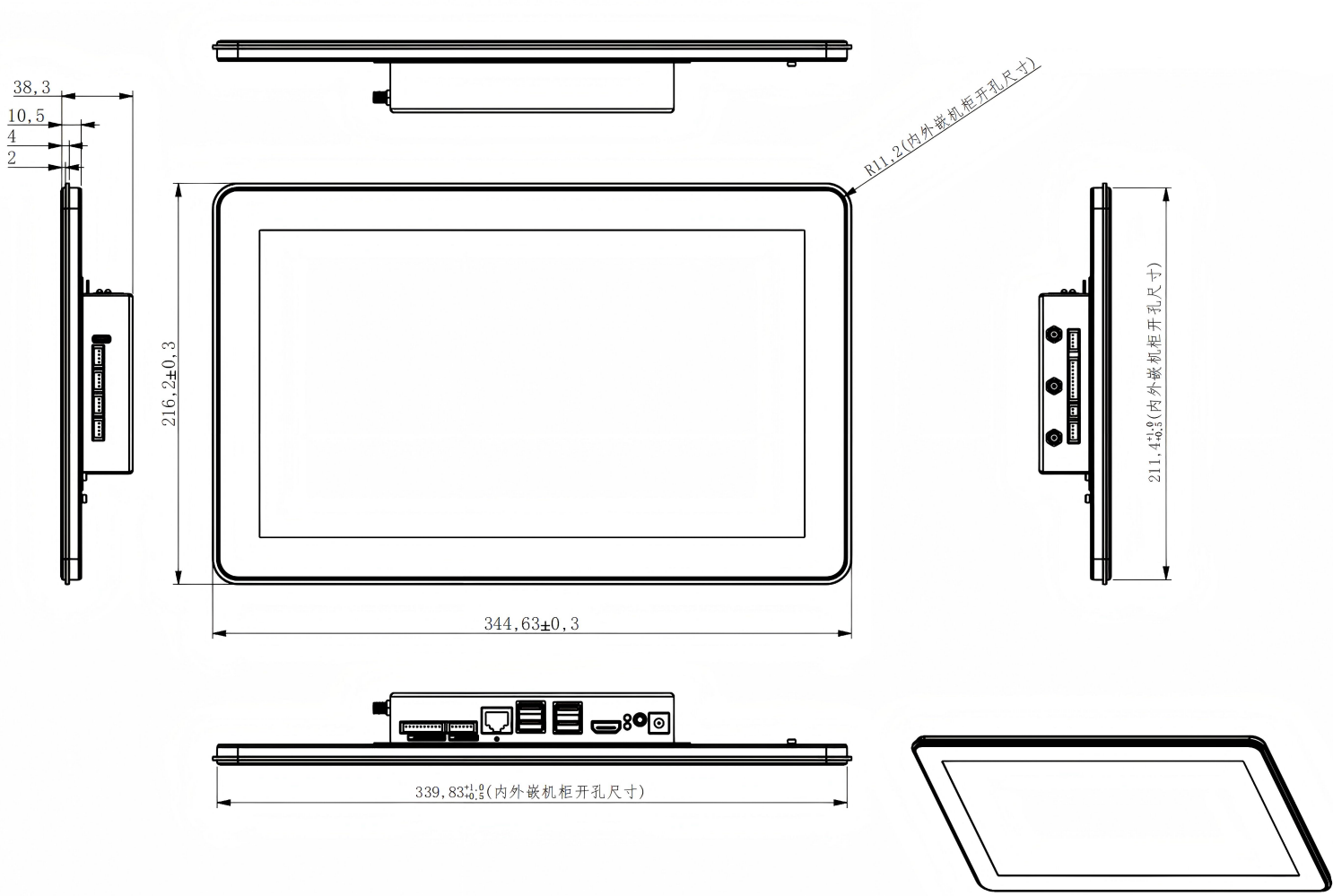
二、产品参数



三、产品介绍

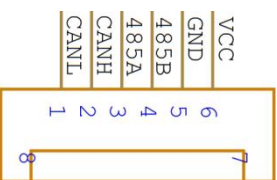
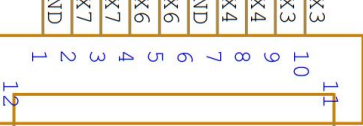
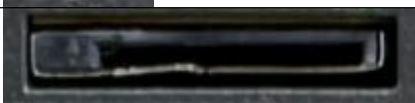
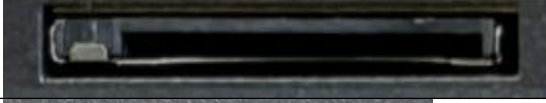
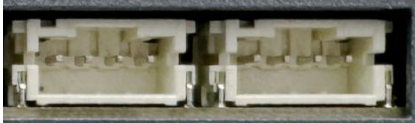
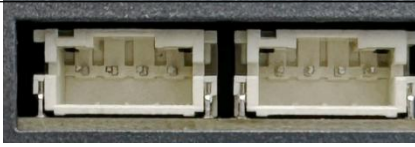
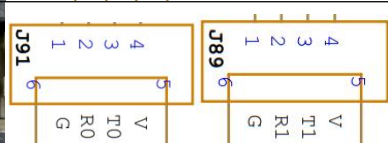
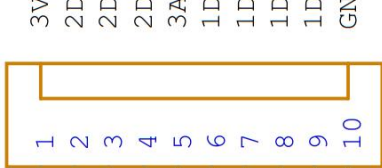
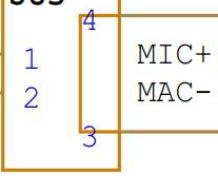
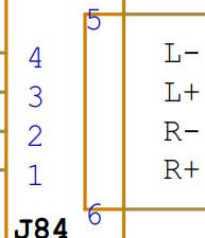
主要参数：		
CPU 处理器	RK3568 Cortex A55 四核 64 位 ARM64，最高主频 2.0G	
操作系统	安卓 Android 12	
运行内存	标配2G	
EMMC 存储	标配16G	
内置 NPU	RKNN NPU，1TOPS 算力，用于 AI 应用	
GPU	MALI-G52-2EE，支持 OpenCL 2.0，OpenGL ES 1.1、2.0、3.2，Vulkan1.1	
显示性能参数：		
尺寸	13.3寸	
分辨率	1920*1080, 16:9	
触摸形式	10 点电容触摸，钢化玻璃材质	
亮度	260 cd/m²	
视角	全视角 IPS	
外设 、 接口 、 配件参数		
电源接口	1路	DC5.5*2.1MM 12V/3A
音频接口	1个	3.5mm耳机接口
MIC接口	1个	
SPK接口	1个	喇叭2路（15 瓦）
指示灯	2个	1 个电源灯，1 个运行灯
HDMI接口	1路	HDMI-A 接口，支持 4K分辨率
USB HOST	4路	USB3.0-HOST 2 路、USB2.0-HOST 2路
千兆以太网接口	1路	10/100/1000Mbps 自适应网络接口
RS485接口	1路	
CAN接口	1路	5V
RS232接口	4路	
TF卡接口	1路	TF卡接口
KEY按键	1个	ADC 方式可实现多个按键，可引出电源键
GPIO接口	8路	可配置为输入或输出模式
WIFI/蓝牙天线接口	1个	WIFI/BT 二合一模组， WIFI2.4G，蓝牙 4.2
4G天线接口	1个	支持安装移远的PCIE 4G全网通模组
SIM卡	1路	标准SIM接口
USB OTG	1路	Type-C座子，可连接PC机进行应用开发和系统烧写
TTL串口	2路	4P接口座
USB2.0接口	2路	4P接口座
RTC功能	板载	支持定时开关机，板载独立 RTC 芯片以及纽扣电池
其他参数		
开机方式	来电自启	
系统管理	原生态 Android 系统，开放 root 权限 实时远程监控，系统崩溃自恢复，7*24 小时无人值守 支持 USB升级，支持 wifi display	
看门狗	支持CPU 内置看门狗	
安装方式	内嵌式、外嵌式、壁挂式	
输入电压	12V	
电源性能	DC 12V外置电源适配器	
工作温度	- 20～ 70° C	
储存温度	- 20～ 70° C	
工作湿度	10%～80%相对湿度, 无冷凝	

四、产品尺寸



五、接口详细说明

接口说明		
		标准12V
		3.5mm耳机音频输出接口
		1 个电源灯，1 个运行灯
		HDMI 输出接口支持 4K 分辨率
		USB3.0-HOST 2 路
		USB2.0-HOST 2 路
		10/100/1000Mbps 自适应网络接口

 	RS485/CAN (设备名 /dev/ttyS9)
 	RS232*4 设备名: /dev/ttyS7, /dev/ttyS6, /dev/ttyS4, /dev/ttyS3
	烧写按键
	TF卡接口
	标准SIM接口
 	USB2.0*2, 4P接口座
 	TTL串口*2, 4P接口座 设备名: J91:/dev/ttyS0, J89:/dev/ttyS1
	DEBUG
	4G天线、GPS/北斗天线、WIFI/蓝牙天线
 	KEY按键, ADC 方式可实现多个按键, 可引出电源键
 	GPIO*8, 可配置为输入或输出模式
 	MIC接口
 	SPK接口, 喇叭2路 (15 瓦)

六、镜像文件烧写

1、 安装驱动并连接硬件

解压开发工具目录下的DriverAssitant_USB驱动程序.7z，右击以管理员权限运行 DriverInstall.exe，安装驱动程序。

注意win10或win11系统需要关闭操作系统的驱动签名验证功能才能正常安装驱动，win11不能永久关闭这个功能，建议安装一个win7的虚拟机，在虚拟机内进行驱动安装和固件烧写。

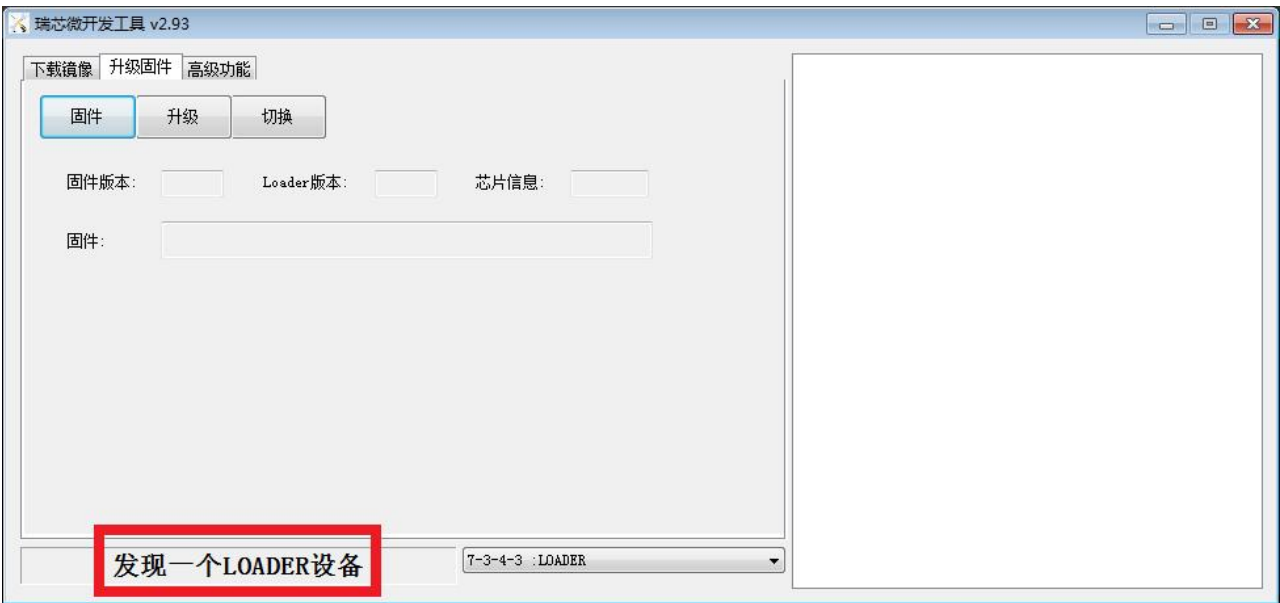
将主板与PC机用TYPE-C线（注意有些数据线仅支持充电，无法传输数据）连接好，如果出现无法识别的情况可以通过重新连接、更换PC机USB接口、更换USB线、更换PC机等方式重试。

解压开发工具目录下的RKDevTool.7z，右击以管理员权限运行 RKDevTool.exe

2、 进入烧写模式

主板处于Loader或Maskrom模式时可以对系统进行格式化和烧写系统镜像文件操作。

进入Loader模式

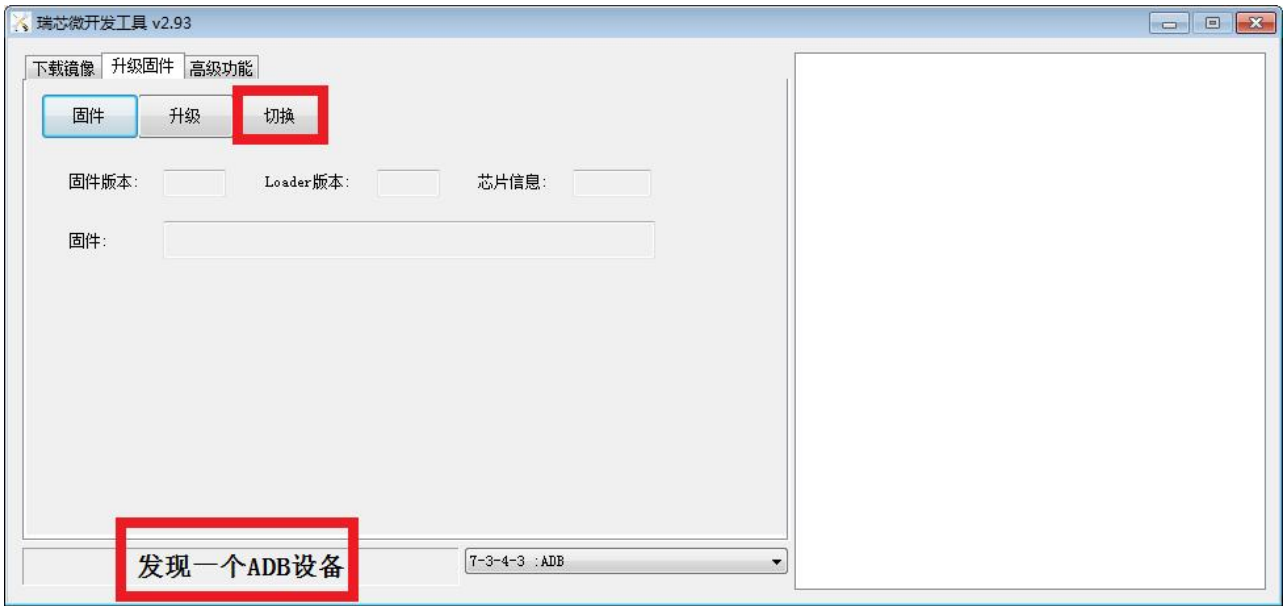


方式一、将主板断电，首先用TYPE-C线将主板与PC机连接好，按下主板上的烧写键，并保持按下状态，然后再上电开机，主板会进入Loader操作模式，注意这种方式适用于主板上的BootLoader可正常工作的情况。



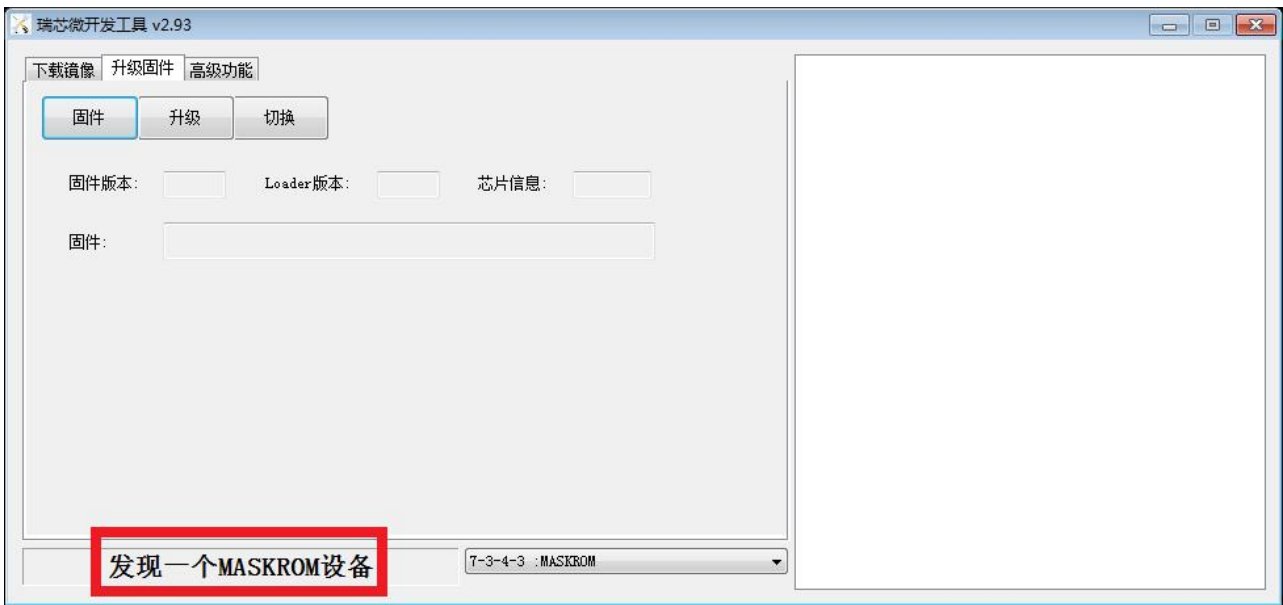
方式二、将主板接通电源，进入系统，用TYPE-C线将主板与PC机连接好如下图，RKDevTool软件会识别到一个ADB设备，单击切换按钮，主板会重启进入Loader

模式

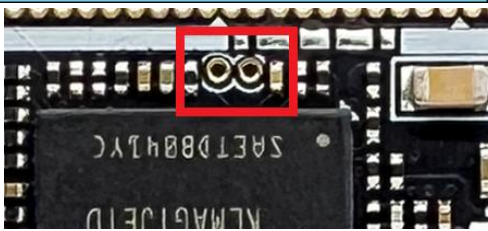


方式三、在调试串口控制台或其他控制终端输入 reboot loader，主板会重启进入Loader模式

3、 进入Maskrom模式（无法进入Loader模式时使用）



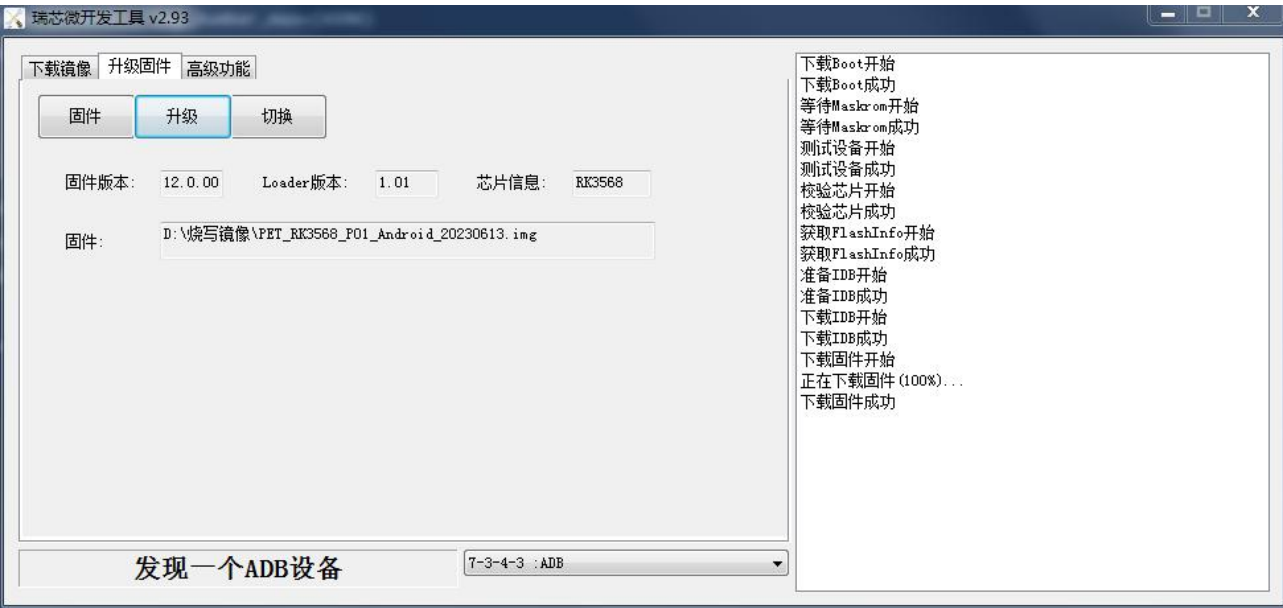
一般仅在uboot损坏，无法进入Loader模式时使用。
首先准备好一个金属镊子，将主板断电
用镊子短接右图中的核心板上两个沉金通孔
然后再上电开机，**进入Masktom模式后取消短接，松开镊子**



继续进行擦除flash、升级固件等操作。

4、 系统烧写流程

首先将主板进入Loader或Maskrom模式，打开RKDevTool软件，点击固件按钮选择需要烧写的镜像文件，然后点击升级按钮，右侧窗口会显示烧写进度，烧写完成后，主板会自动重启开机。



七、安卓应用开发

1、 GPIO编程参考

通过 sysfs 方式控制 GPIO，GPIO 的操作接口包括 direction 和 value 等，direction 控制 GPIO 输入和输入模式，而 value 可控制 GPIO 输出或获得 GPIO 输入。

例如控制调试灯GPIO操作如下（串口终端命令行方式）：

调试灯GPIO设置为输出 echo out > /sys/class/gpio/gpio23/direction

调试灯GPIO输出高电平 echo 1 > /sys/class/gpio/gpio23/value

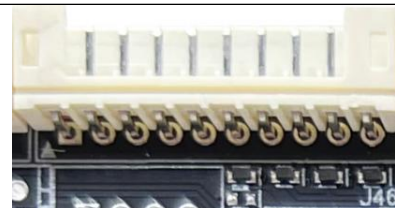
调试灯GPIO输出高低平 echo 0 > /sys/class/gpio/gpio23/value

调试灯GPIO设置为输入 echo in > /sys/class/gpio/gpio23/direction

读取调试灯GPIO输出输入电平 cat /sys/class/gpio/gpio23/value

当GPIO处于输出和输入模式时都可以读取，当设置为输入模式时读取的是GPIO实际电平，当设置为输出模式时读取的是设置的值（如果设置为高电平输出，外部将引脚电平拉低后，读取的值依然是1）。

应用程序控制请参考源码下的demo程序源码

	3V3 2D6 2D4 2D5 3A7 1D1 1D2 1D3 1D4 GND	1 2 3 4 5 6 7 8 9 10
2脚	GPIO	/sys/class/gpio/gpio94
3脚	GPIO	/sys/class/gpio/gpio92
4脚	GPIO	/sys/class/gpio/gpio93
5脚	GPIO	/sys/class/gpio/gpio103
6脚	GPIO	/sys/class/gpio/gpio57
7脚	GPIO	/sys/class/gpio/gpio58
8脚	GPIO	/sys/class/gpio/gpio59
9脚	GPIO	/sys/class/gpio/gpio60

	NFC-IO2 NFC-IO1 I2C2_SDA_M1 SPI1-MOSI SPI1-CS0	1 3 5 7 9 11	2 4 6 8 10 12	NFC_INT NFC-IO3 I2C2_SCL_M1 SPI1-CLK SPI1-MISO
1脚	GPIO	/sys/class/gpio/gpio33		
7脚	GPIO	/sys/class/gpio/gpio113		
8脚	GPIO	/sys/class/gpio/gpio115		
9脚	GPIO	/sys/class/gpio/gpio97		

10脚	GPIO	/sys/class/gpio/gpio114
-----	------	-------------------------

2、 串口UART编程参考

	TTL串口，设备名 /dev/ttyS1
	TTL串口，设备名 /dev/ttyS0
	RS485接口，设备名 /dev/ttyS9
	RS232接口，设备名： /dev/ttyS7, /dev/ttyS6, /dev/ttyS4, /dev/ttyS3

安卓系统串口编程请参考demo程序源码或以下链接：

<https://github.com/yutils/YSerialPort>
<https://github.com/Acccord/AndroidSerialPort>
<https://github.com/Geek8ug/Android-SerialPort>

3、 WatchDog看门狗编程参考

进入内核后默认会启动看门狗，内核崩溃等情况出现，会在 15 秒内自动复位主板。
上层应用程序打开看门狗后，内核将看门狗控制权交由上层应用程序控制，上层应用程序的喂狗间隔建议不大于3秒。

看门狗的使用流程为 打开看门狗→循环喂狗→停止喂狗→关闭看门狗
喂狗之前必须先打开看门狗，关闭看门狗之前需停止喂狗操作。
打开看门狗后如果15秒内没有喂狗或关闭看门狗，系统会自动复位。

命令行测试：

打开看门狗：echo 1 >/sys/class/gzpeite/user/watch_dog
喂狗： echo 2 >/sys/class/gzpeite/user/watch_dog
关闭看门狗：echo 0 >/sys/class/gzpeite/user/watch_dog

应用程序控制请参考源码下的demo程序源码

4、 获取root权限

系统默认已开启root权限，上层应用app可直接获取root权限并进行相关操作，可以参考源码目录下的demo程序

5、 系统签名

系统签名文件位于源代码目录下，使用对应的文件对APK进行签名即可。

6、 动态隐藏/显示系统状态栏和导航栏

隐藏状态栏和导航栏在应用app里面向系统发送广播
gzpeite.intent.systemui.hidenavigation和gzpeite.intent.systemui.hidestatusbar
显示状态栏和导航栏在应用app里面向系统发送广播
gzpeite.intent.systemui.shownavigation和gzpeite.intent.systemui.showstatusbar

测试命令如下：

```
am broadcast -a "gzpeite.intent.systemui.hidenavigation"
am broadcast -a "gzpeite.intent.systemui.hidestatusbar"

am broadcast -a "gzpeite.intent.systemui.shownavigation"
am broadcast -a "gzpeite.intent.systemui.showstatusbar"
```

请参考源码下的demo程序源码

7、 静默安装/卸载应用

安装APK时， 向系统发送gzpeite.intent.action.install_apk 广播

卸载APK时， 向系统发送gzpeite.intent.action.uninstall_apk 广播

测试命令如下：

```
am broadcast -a "gzpeite.intent.action.install_apk" --es apk_path "/mnt/media_rw/0000-4823/GPSTest.apk"
am broadcast -a "gzpeite.intent.action.uninstall_apk" --es pkg_name "com.android.gptest"
```

8、 重启、关机操作

重启： 向系统发送gzpeite.intent.action.reboot 广播

关机向系统发送gzpeite.intent.action.shutdown 广播

测试命令如下：

```
重启(有确认提示): am broadcast -a "gzpeite.intent.action.reboot" --ez confirm true
重启(无确认提示): am broadcast -a "gzpeite.intent.action.reboot" --ez confirm false
关机(有确认提示): am broadcast -a "gzpeite.intent.action.shutdown" --ez confirm true
关机(无确认提示): am broadcast -a "gzpeite.intent.action.shutdown" --ez confirm false
```

应用程序控制请参考源码下的demo程序源码

9、 获取MAC地址

原生Android12系统默认禁止应用获取MAC地址，为了兼容更早期的应用程序，我司已对系统代码进行优化允许应用app获取WIFI及以太网的MAC地址，详见源代码目录下的demo程序源码。

九、 动态修改开机logo和动画

将logo.bmp、 logo_kernel.bmp、 bootanimation.zip，复制到系统 /mnt/logo目录下即可

```
adb push logo.bmp /mnt/logo/
adb push logo_kernel.bmp /mnt/logo/
adb push bootanimation.zip /mnt/logo/
```

两个logo文件必须为bmp文件格式，一个是在uboot阶段加载显示，一个是在内核阶段加载显示，文件名不可修改，文件内容可以完全一样。

开机动画bootanimation.zip 仅支持安卓系统，制作方式可以通过搜索引擎查询相关教程。

安全指导

版权声明

本使用手册所提及的商标与名称，均属其合法注册的公司所有。

责任声明

本使用手册受著作权保护，所撰写的内容均为本公司所拥有。

本使用手册所提及的产品规格或相关信息，本公司保留修改的权利。

本使用手册所提及的产品规格或相关信息有任何修改或变更时，恕不另行通知。

未事先经由本公司书面允许，不得以任何形式复制、修改、转载、传播或出版本使用手册内容。

产品质量保证

从购买之日起，本公司为原购买商提供一年的免费保修，但对那些未经授权的维修人员维修过的产品并不予提供质量保证。对于不正确的使用、灾难、错误安装产生的问题有免责权利。如果本公司产品出现故障，在质保期内我们提供免费维修或更换服务。对于出保产品，我们将会酌情收取材料费、人工服务费用。请联系相关销售人员了解详细情况。

安全指示

1. 请仔细阅读此安全操作说明。
2. 请妥善保管好此用户手册，以便日后参考。
3. 对于使用电源线的设备，设备周围必须有容易接触到的电源插座。
4. 当您连接设备到电源插座上时，请确认电源插座的电压是否符合要求。
5. 请将电源线布置在人们不易绊到的位置，并不要在电源线上覆盖任何杂物。
6. 清洁设备前，请从插座拔下电源线。请不要使用液体或去污喷雾剂直接清洁设备，应借助抹布使用。
7. 请在安装前确保设备放置在可靠的平面上，防止意外跌落。
8. 如果长时间不使用设备，请将其同电源插座断开，避免设备被超标的电压波动损坏。
9. 请不要把设备存储在超出我们建议的温度范围的环境。即不要低于-20°C或高于70°C否则可能会损坏设备。
10. 请不要让任何液体流入设备，以免引起短路或者火灾。
11. 请不要自行打开设备。为了确保您的安全，请由专业的技术人员或经过认证的工程师来打开设备。