RJM8L003 用户开发指南

一、 RJM8L003 最新资料链接，不定期更新此链接中的文档等技术资料，请自行不定期 进入下载。数据手册可以直接自行到公司官网 www.runjetic.com 下载。

链接：https://pan.baidu.com/s/1FUHQAs_ZY24kOU69b0WF9Q?pwd=c42p

提取码：c42p

二、 RJM8L003 通用驱动文件链接，供学习和使用，仅供参考！

链接：https://pan.baidu.com/s/16v_v3QRwAM1_urZ6Z0xDuw?pwd=1una

提取码：1una

三、 更新日志中出现增加，补充说明，变更字样的可以直接看对应注意点，如出现删除字样的请对照该文档的上一次更新，详细见文档最后一页修改日志，文中标注 红色字体的，开发前需要重点阅读!!!

FQ&A：

1、 外部晶振能为 24MHz 吗？

答：可以，外部无源晶振选用范围（4-24MHz），SOP16 和 SOP14 封装才有，外部无源晶体大于 16MHz 时，需要二分频，另外支持有源晶振（1-16M）或 方波信号源输入，**目前的方波信号源测试到 48MHz，但是需要分频到有效频率范围内。超过有效频率范围会发生意想不到的现象！！！**

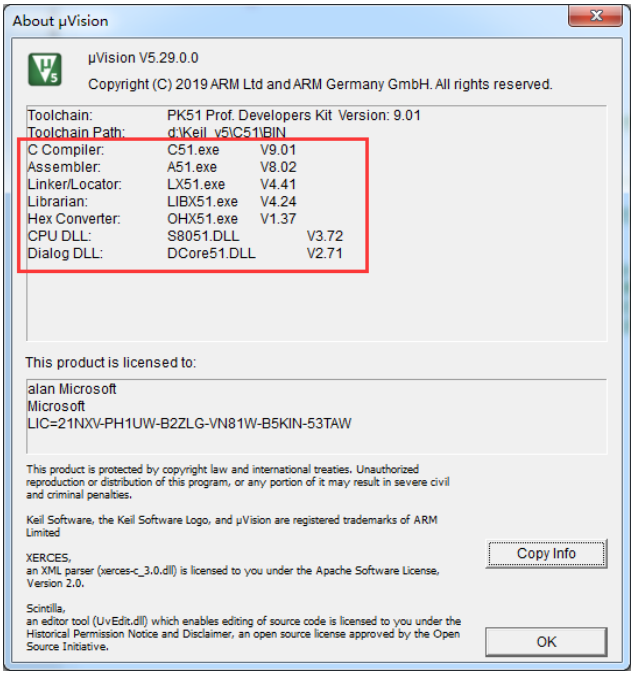
2、 支持在线仿真吗？

答：不支持。

3、 是否支持 IAP？

答：支持。

4、 **RJM8L003 开发环境搭建：** Keil_C51编译链接补丁.rar **，请按此压缩包中的文档进行替换安装，此文件在 RJM8L003 资料 链接中可以下载安装成功后可以在 keil C51 软件下工具栏 help 查看 About uVision 的版本信息，对照编译链接工具版本，如下图示。**



5、 **RJM8L003、RJM8L151S 和 RJM8L151 的简易烧录器硬件相同，且固件也做了兼容请及时更新烧录器固件。**如果连接开发板后无法复位目标芯片，请确认烧录器固件是否是最新的烧录器固件版本，新版上位机烧录软件需要配合新版的烧录器固件，将烧录器与电脑连接好后可以按下烧录器上的复位按钮，用

串口助手查看烧录器固件版本的打印信息，串口波特率 115200。烧录器固件更新方法可以查看简易版烧录器固件更新说明书。

6、 串口使用注意事项：

1、 printf 函数在 51 单片机中使用需要特别注意格式化输出的数据类型。否则很容易出现数据大小溢出。**一定不要在中断函数中直接使用 printf 函数，或直接使用串口发送函数。**

2、 开启串口中断后，发送中断和接收中断同时开启了，因此 MCU 发送数据的时候也会进中断服务函数，需要看中断服务函数有没有判断发送完成标志。**MCU 串口发送数据时，最好先关闭串口接收及中断，发送完成后再打开串口接收及中断，否则在多个中断的系统应用中时，会有可能出现发送一组数据（8 个为例）时，数据分段的情况。**

3、 当系统有多个中断源应用时，注意串口优先级，处于低优先级时可能会导致接收时丢失数据。

7、 在使用低功耗各模式时，注意事项：

1> 没有用到（悬空）的 GPIO 将 GPIO 的输入、输出使能关闭，用到的 GPIO 根据外围电路状态输出高电平或低电平。

8.4.12. 管脚输入使能寄存器 P0_IEN（偏移：F8H）

比特	名称	属性	复位值	描述
7:0	P0_IEN	RW	0xFF	0：不使能管脚输入功能； 1：使能管脚输入功能； 例如：P0_IEN[5]=1,，GPIO0.5 对应的 IO 输入功能使能；

8.4.13. 管脚输入使能寄存器 P1_IEN（偏移：F9H）

比特	名称	属性	复位值	描述
7:0	P1_IEN	RW	0xFF	0：不使能管脚输入功能； 1：使能管脚输入功能； 例如：P1_IEN[5]=1,，GPIO1.5 对应的 IO 输入功能使能；

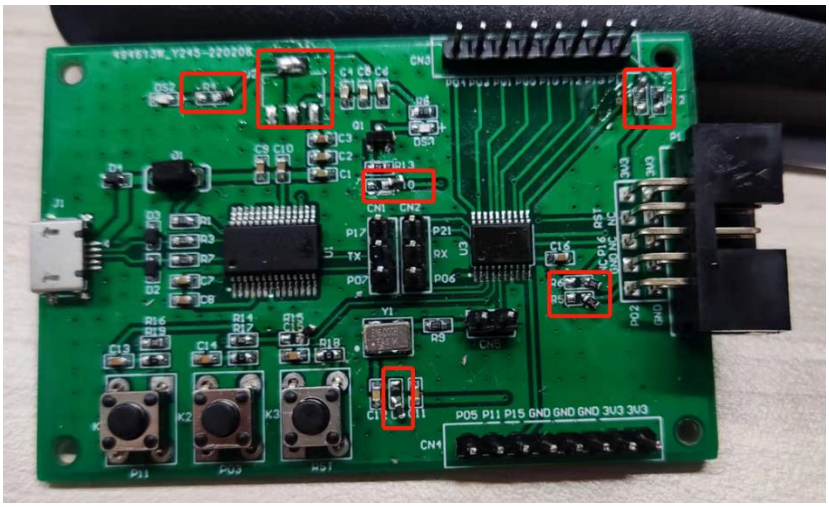
8.4.14. 管脚输入使能寄存器 P2_IEN（偏移：FAH）

比特	名称	属性	复位值	描述
7:2	RSV	-	-	保留
1:0	P2_IEN	RW	0x3	0：不使能管脚输入功能； 1：使能管脚输入功能； 例如：P2_IEN[1]=1,，GPIO2.1 对应的 IO 输入功能使能；

8.5.2. 输出使能寄存器 Px_OEN（偏移：A1H、A2H、A3H）

比特	名称	属性	复位值	描述
7:0	Px_OEN	RW	0x00	配置 GPIOx 口的输出使能 例如：将 GPIO1.3 配置成输出，则 P1_OEN=0x08；

2> 如果想使用开发板测试芯片功耗，可以参考下图拆掉一些元件：



3> 低功耗应用时：GPIO 口做为 IO 唤醒源后是不能读取这个 IO 的状态的

除非将 IO 唤醒源使能位关闭。

8、 ADC 使用注意事项：

1> 使用时 ADC 各通道口需要增加滤波对地电容

9、 开发板使用时，注意事项：

1> 当使用烧录器和开发板同时供电时，上电后开发板不运行是由于开发板复位脚把 MCU 复位脚钳住了。

2> 当使用开发板供电时，未使用烧录器供电，且未断开烧录器与烧录板的连接时也会出现上电后开发板不运行的现象，也是由于开发板复位脚把 MCU 复位脚钳住了，此时断开开发板与烧录器的连接线方可运行。

3> 在使用高级定时器的 PWM 输出通道 2 的反向输出口时，需要将板上的电容去掉后再测试。

10、 烧录口 IO 复用硬件注意事项

答：当使用烧录口的 IO 时，配置 GPIO 功能需放置到最后初始化，另外烧录口 IO **不能有下拉电阻**和**对地电容**，否则会影响固件烧录；烧录程序的时候需要接复位脚。

11、 SOP14 和 SOP16 封装 OSCIN，OSCOUT 可否用为 GPIO？

答：外部晶振脚不能用作 GPIO 功能，20Pin 的没有外部晶振脚，4 种封装均没有外部 32768Hz 时钟接口。

12、 Flash 擦除、读写，在供电范围内（2.0 – 5.0V），是否都可以？

答： 可以

13、 高阻输入怎么设置？

答：GPIO 没有高阻输入模式

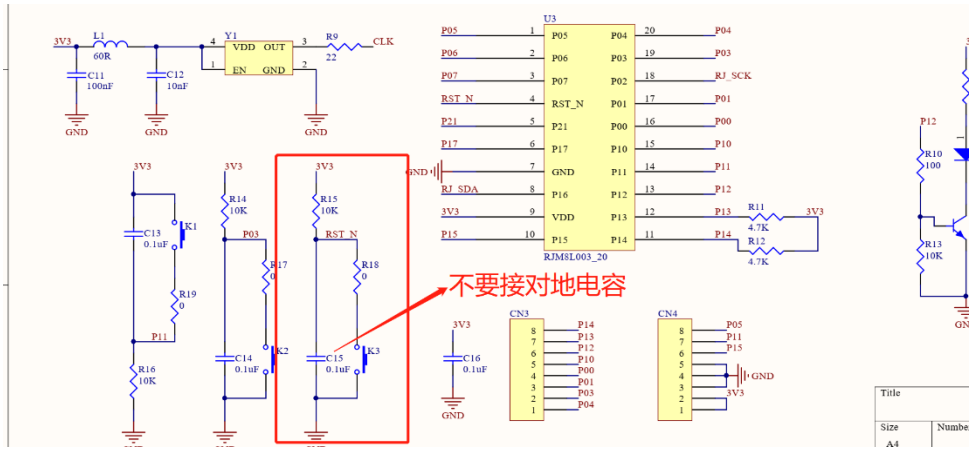
14、 **RJM8L 简易版烧录器烧录时使用注意事项：**

a、使用时烧录器烧录时请注意不要外挂大电流器件。

15、 **如果烧录脚的 IO 在实际应用中被复用为功能口，就要预留 RST?**

答：是的，因此建议 PCB 设计时烧录口统一预留出 RST 脚。

16、 **开发板的参考电路中，003/151s 的复位电路的电容不要焊接，但电路可以保留，增加了反而会出现上电不工作的现象!!!**



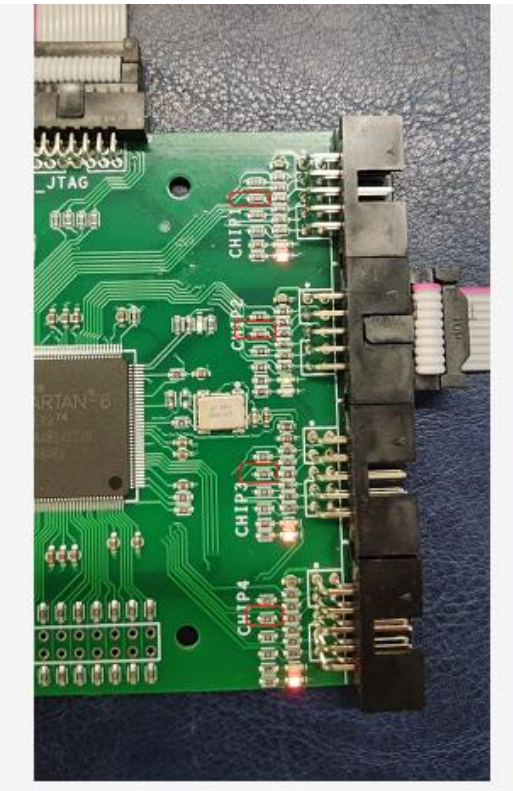
特别需要注意：烧录口不能有上拉电阻!!!

17、 003/151S 正常工作时的功耗峰值有多大？

答：小于 4mA。

18、 注意：003 的早期样品没有写入 UID，出厂的芯片会写入。

19、 003/151s 的 1 拖 4 烧录器 硬件上需要去掉 4 颗电阻，不去掉会影响在线烧录，不影响脱机烧录，如下图所示，在线烧录速度会比较慢，不建议使用。



20、 如果是 2022 年 6 月 14 日之前送样发送的资料，gpio 的库文件里的 GPIO 输入输出函数配置存在书写错误，请及时在 003 资料包中下载新的库文件替换掉原 gpio 文件，最新库版本更新至 V1.3。如图：

正确的：

```
  * @Name     IO_SetDirecion
  * @brief    设置IO的输入/输出
  * @param    GPIOx: [输入]  端口组
**           GPIO_Pin: [输入]  端口组的引脚号
**           Direcion: [输入] IO方向
  * @retval   None
  * @author   Runjet Application Team
  * @Data     2022-03-03
 **/
void IO_SetDirecion(GPIOx_Typedef GPIOx, uint8_t GPIO_Pin, GPIOx_PinDirTypedef Direcion)
{
    if(Direcion == GPIO_OutPut)
    {
        if(GPIOx == GPIO_P0)
        {
            P0_OEN |= GPIO_Pin;
            P0_IEN &= (~GPIO_Pin);
        }

        else if(GPIOx == GPIO_P1)
        {
            P1_OEN |= GPIO_Pin;
            P1_IEN &= (~GPIO_Pin);
        }

        else if(GPIOx == GPIO_P2)
        {
            P2_OEN |= GPIO_Pin;
            P2_IEN &= (~GPIO_Pin);
        }
```

错误的：

```
        }

        else if(GPIOx == GPIO_P1)
        {
            P1_OEN |= GPIO_Pin;
            P1_IEN &= (~GPIO_Pin);
        }

        else if(GPIOx == GPIO_P2)
        {
            P2_OEN |= GPIO_Pin;
            P2_IEN &= (~GPIO_Pin);
        }
    }

    else
    {
        if(GPIOx == GPIO_P0)
        {
            P0_IEN |= GPIO_Pin;
            P0_OEN &= (~GPIO_Pin);
        }

        else if(GPIOx == GPIO_P1)
        {
            P0_IEN |= GPIO_Pin;
            P1_OEN &= (~GPIO_Pin);
        }

        else if(GPIOx == GPIO_P2)
        {
            P0_IEN |= GPIO_Pin;
            P2_OEN &= (~GPIO_Pin);
        }
    }
```

21、 低功耗休眠状态下全局变量会被保存下来吗？

答：是的。

22、 全局变量被修改的问题

1、优化的太狠了（8 级有时候也发现了）会导致函数临时变量地址重叠，这时候如果一个函数正在运行，突然来了中断，刚好函数内的临时变量和中断服务函数内的变量地址编译器编译的一样，就会发生中断退出后，回到原来函数执行的时候，变量被修改的问题

2、中断内最好不要有执行函数，做一些标记判定，执行标记变量定义的时候用全局变量，这样在编译的时候不会有地址折叠问题，就不会因为进出中断导致外部变量数据被改变。顺序执行的时候变量是不会有问题，就是中断的时候如果地址折叠重复利用就会出现变量被修改

3、注意：数据量大小不要超越定义的数组大小范围，否则可能导致全局变量被修改的现象

23、 **注意：低功耗唤醒后，用到的外设最好重新初始化，不能省掉相关配置**

例如：串口功能的配置

```
152     LVD_PvdCmd(ENABLE);
153     LVD_BorCmd(ENABLE);
154
155     IRQ_Enable(IRQ_IntRtc);//打开
156     Uartx_init( UART0 , BAUDRATE_115200);          //初始化串口，相关配置不能省
157
158     Lowpower_AwakeSource_off();                    //关闭唤醒源
159
```

24、 **注意：低功耗定时器的时钟源和 RTC 的时钟源是同时选择的，有且只有内部低速时钟源。**

25、 **旧版上位机连接 MCU 无法正常烧录，现象如下图所示：**

排查方法如下：

1、检查烧录器是否连接目标板。

2、检查烧录器与目标板是否接线错误

3、检查目标板的烧录脚是否有其电路，不能有下拉电阻和对地电容，同时 FPGA 烧录器脱机烧录，且是带目标板烧录时烧录脚也不能有上拉电阻及对地电容。

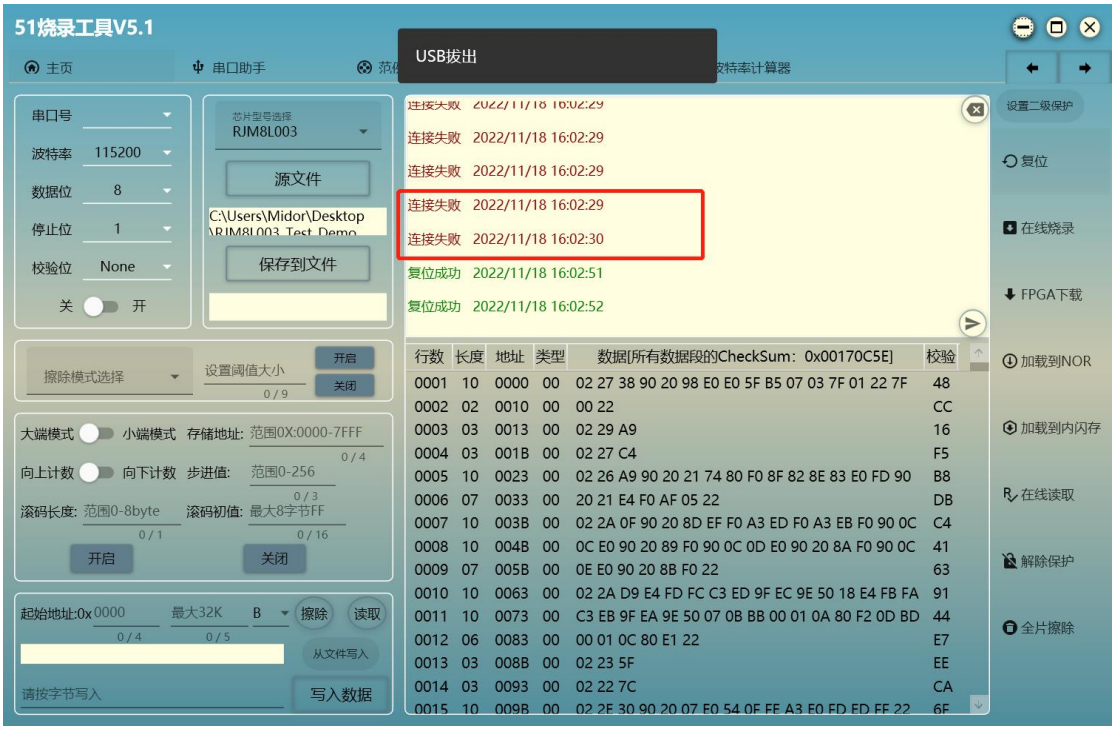
4、如有使用烧录口复用为其它功能，建议把这些 IO 复用功能的配置放到最后初始化。

5、确认烧录器固件版本是否是 RJM8LPR-00B_V2.1 版本，如果是，请确认烧录器的 R12、R20 电阻是否去掉

6、请使用新版上位机软件进行项目开发调试

7、现象：RJM8HPR_00C 脱机烧录器无法离线烧录，可以在线烧录时，请检查一下烧录模式是否是纯烧录模式，复位烧录器用串口助手即可以查看 log 信息

8、



FPGA 一拖一 或一拖四 脱机烧录时出现以上现象的可能原因：

1) 检查烧录器的 STM32 固件是否是高性能的量产烧录器固件，特别注意固件版本的区别，高性能的脱机烧录软件的固件命是：RJM8HPR_00B_Vxxx，简易版的烧录器固件命名是：RJM8LPR_00B_Vxxx。

2) 检查烧录器的时钟和数据线线序是否连接反。

9、

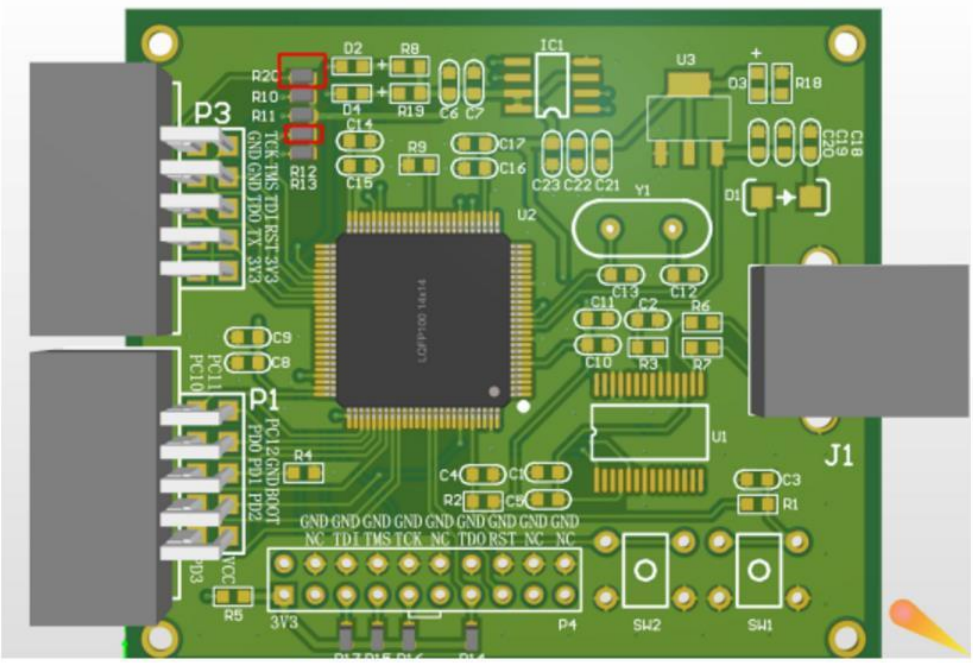
26、 使用串口唤醒时，需要注意唤醒响应时间过短导致接收唤醒数据不能及时显示的现象。可以在串口唤醒后增加最少 8-10ms 的延时等待。

```c
main.c | uart.c | rtc.c | functions.c | gpio.c
212     printf("result = %f\r\n",result);
213     return 0;
214 }
215 //休眠唤醒测试
216 void Sleep_Awake_Test(void)
217 {
218     static uint8_t i=0;
219
220     if(P11_Awake_flag){
221         ss++;
222         IRQ_Disable(IRQ_IntUart0);
223         printf("ss = %02bx\r\n", ss);
224         IRQ_Enable(IRQ_IntUart0);
225         P11_Awake_flag=0;
226     }
227     if(uart0_rx_flag == 1){
228         uart0_rx_flag=0;
229         P15 = 1;
230         for(i=0;i<in_cnt;i++)
231             uart_sendbyte(UART0,rx3data[i]);
232     }
233     if(LPtimer_awake_Flag ==1){
234         LPtimer_awake_Flag =0;
235         printf("LPtimer_Awake\r\n");
236     }
237     if(Uart0_awake_Flag ==1){
238         Uart0_awake_Flag =0;
239         printf("s\r\n");
240     }
```

```c
lowp.c | LOWP.H
151     /* 在进Halt前需要将打开了的中断全部关闭，然后进halt。*/
152     /* 例如使用了uart0中断，其操作如下:IRQ_Disable(IRQ_IntUart0);  LP_HaltEnter
153     LP_HaltEnter();
154
155     LVD_PvdCmd(ENABLE);
156     LVD_BorCmd(ENABLE);
157
158     LPTIMER_IntCmd(DISABLE);                                    //唤醒后先关闭
159     LPTIMER_Cmd(DISABLE);
160
161     IRQ_Enable(IRQ_IntRtc);//打开
162     IRQ_Enable(IRQ_IntUart0);
163
164 //    Uartx_init( UART0 , BAUDRATE_115200);                     //初始化串口，
165
166     Delay_ms(8);                                                //由于唤醒的时
167     Lowpower_AwakeSource_off();                                 //关闭唤醒源
168
169     if(RTC_Awake_flag == 1){
170         RTC_Awake_flag =0;
171         P15=1;
172     }else if(P11){
173         P04=1;
174         P11_Awake_flag=1;//标记唤醒口
175     }
176 //    else{
177 //        P11_Awake_flag=0;
178 //    }
179
```

27、 简易版烧录器的固件最新版本为 RJM8LPR-00B_V2.4，及以后版本，需要去掉两颗电阻，支持全片擦除功能。RJM8LPR-00B_V2.3 及以后版本需要配合新版 51 烧录工具才能使用，建议及时同步更新为最新烧录器固件，如下如所示：

1、 简易烧录器需要去除两颗上拉电阻（R12和R20），电阻位置如下图所示：



新版烧录软件 V5.1：

新版上位机软件常见问题及注意事项：

1、该软件是 64 位的软件，目前只支持 win7/win10 64 位系统，如使用的是 win7 32 位系统，会存在兼容性问题，可能无法使用。

2、加载源文件后提示源文件不存在请检查源文件是否有被人为编辑过加入了奇怪的符号!!! 造成的结果是无法烧录!

28、 注意：定义成 code 类型数组后不能成功写入 flash，例如：

```
uint8 code data2[]=
{
0x55, 0x55, 0x55, 0x55, 0x55, 0x55, 0x55, 0x55, 0x55, 0x55, 0x55, 0x55, 0x55, 0x55, 0x55, 0x55,
0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA,
0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA,
0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA,
0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA,
0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA,
0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA,
0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA
};
```

29、 RTC 使用外部时钟源时可以不用配置寄存器 **RTC_SECOND_CNT_x(x=3~0)，复位值是 32768**

15.4.14. 秒计数校准寄存器 RTC_SECOND_CNT_x(x=3~0)(偏移：10H~13H)

比特	名称	属性	复位值	描述
31:0	SEC_CNT	RW	0x8000	秒计数校准寄存器 当采用内部 LSI ~~或外部 32.768k~~ 时钟作为 RTC 计数时钟时，均需要配置此寄存器。配置值为使用时钟计数 1 秒需要的 cycle 个数 W：配置秒计数初始值； R：读取秒计数初始值；

注：该寄存器由 Sec_cnt_3、Sec_cnt_2、Sec_cnt_1、Sec_cnt_0 依次拼接而成，其中 Sec_cnt_3 为最高字节，Sec_cnt_0 为最低字节。

30、 在上位烧录机软件上操作 flash 功能时注意：上位机读写或擦除 flash 时，需要注意的是烧录器都会复位 MCU，当进行上位机擦除 flash 中指定地址数据时，如果程序初始化进行写 flash 操作且每次写的数据相同，那么会造成一个上位机擦写 flash 失败的现象。把程序中初始化写 flash 那段程序屏蔽即可。

31、 在配置 SPI 做主机，快速模式下，连续发送数据时，每个数据间会有 1us 的延时间隔，建议 SPI 最大时钟速率是控制在 4MHz 及以下，否则在大量数据传输的情况下会有几率出现数据传输错误。

32、

如有其它疑问及时联系我司！

修改日志：

1. 2022-03-25：更新 RJM8L003 底层开发库至 RJM8L003_DEMO&LIB_v1.2.rar。添加 RJM8L003 资料链接，不定期更新此链接中的文档等技术资料，请自行不定期进入下载。数据手册可以直接上公司官网 www.runjetic.com 下载，谢谢！
   资料链接：https://pan.baidu.com/s/1Q0kVDhT05jsP4Bt7sLr_jg 请点击
   提取码：v0nd （此资料链接不在维户！）
2. 2022-03-29：增加 开发使用注意项 第 6，7 点
3. 2022-04-01：补充说明 开发使用注意项 第 7 点
   增加 开发使用注意事项 第 8 点

4. 2022-04-08：增加 开发使用注意项 第 9 点
5. 2022-04-11：补充说明 开发注意示项 第 9 点的第 3 项
   增加 开发注意事项 第 10 、11、12 点
6. 2022-04-22：修改并补充说明 开发注意事项 第 5 点
7. 2022-04-29：补充说明 开发注意事项 第 7 项的第 3 项
8. 2022-05-09：增加 开发注意事项 第 13 、14、15 点
9. 2022-06-02：增加 开发注意事项 第 16 点
10. 2022-06-09：增加 开发注意事项 第 17 点
11. 2022-06-09：增加 开发注意事项 第 18,19,20 点
12. 2022-06-14：删除 开发注意事项 第 7 点的第 3 项
    003 资料新链接：https://pan.baidu.com/s/15MaOLMlas8kkA919wPXCgQ

    提取码：q9ex （此链接不再使用）

    用户手册更新至 V1.2 版本，请及时下载最新手册！
13. 2022-06-15：修改并补充说明 开发注意事项 第 17 点
14. 2022-06-17：补充说明 开发注意事项 第 7 点的第 1 项
15. 2022-06-24：增加 RJM8L003 通用驱动链接，供学习和使用，仅供参考！
    链 接 ： https://pan.baidu.com/s/16v_v3QRwAM1_urZ6Z0xDuw?pwd=1una
    提取码：1una
16. 2022-06-30：增加 开发注意事项 第 7 点的第 3 项
17. 2022-07-08：修改并补充 开发注意事项 第 6 点
    增加 开发注意事项 第 21 ，22，23 点
18. 2022-07-11：修改 开发注意事项 第 5 点
19. 2022-07-12：补充说明 开发注意事项 第 7 点的第 3 项
20. 2022-07-14：增加 开发注意事项 第 24 点
21. 2022-07-19：增加 开发注意事项 第 25，26 点
22. 2022-07-22：增加 开发注意事项 第 27，26 点
    补充说明 开发注意事项 第 25 点的第 5 项
    删除 原开发注意事项 第 16 点
23. 2022-08-08：增加 开发注意事项 第 28 点
24. 2022-08-12：补充说明 开发注意事项 第 27 点
25. 2022-08-24：修改并补充 开发注意事项 第 5，20，25，27 点
26. 2022-08-29：补充说明 开发注意事项 第 25 点
27. 2022-09-02： 003 资料链接变更：
    https://pan.baidu.com/s/1FUHQAs_ZY24kOU69b0WF9Q?pwd=c42p
    提取码：c42p
28. 2022-09-19：补充说明 开发注意事项 第 6 点第 3 项
29. 2022-09-23：补充说明 开发注意事项 第 22 点第 2 项
30. 2022-09-29：补充说明 开发注意事项 第 16，25，27 点
31. 2022-11-18：增加 开发注意事项 第 25 点的第 8 项
32. 2022-11-21：更新 RJM8L003 底层开发库至 RJM8L003_DEMO&LIB_v1.4.rar。

33. 2022-11-29：增加 开发注意事项 第 29、30 点
34. 2023-02-08：增加 开发注意事项 第 31 点
35.