**IC8001 DEMO板**

**用户手册**

# 第一章 概述

## **1.1 IC8001 DEMO板特性**

IC8001 DEMO板以自主研发的IC8001虹膜编码识别芯片为核心，集成国际领先水平的虹膜编码识别算法,识别准确快速，二次开发应用简单。

主要具有以下功能：

1. 虹膜采集（注册）；
2. 虹膜比对（识别）；
3. 模板储存；
4. 模板删除

主要应用领域：虹膜门锁、虹膜保险柜、身份识别授权、门禁系统等领域。

开发者可以根据本手册提供的技术资料，开发出从简单到复杂的虹膜识别应用系统。

公司还提供基于单片机的C语言参考SDK开发包，具体可与我公司联系。

## 1.2 虹膜编码识别模组组成

IC8001 DEMO板主要由两部分组成，M4核心板和IC8001核心板（外加摄像机模块和LCD显示屏）。

IC8001核心板主要实现对虹膜图像进行处理生成虹膜模板，模板存储，模板删除，模板比对。

M4核心板实现与IC8001核心板通信(通过UART,SPI,IIC外部通信接口)发送控制命令（如注册，识别，删除以及导出图片和模板数据），以及图像回显。

IC8001 DEMO板各部分组成如下图：

图1.1 IC8001 DEMO板各部分框图

摄像机+红外灯

IC8001核心板

按键，LED,蜂鸣器等外设

LCD显示

M4核心板

# 主要技术指标

表2.1 技术指标

|  |  |
| --- | --- |
| 虹膜模板存储容量 | ≤1024个 |
| 编码识别处理时间 | ＜0.08S |
| 采集/识别方式 | 单目、双目、定焦非接触 |
| 通信接口 | IIC,UART,SPI |
| 额定电压 | 5V |
| 额定电流 | (待测试) |
| 错误接受率（FAR） | 10-7 |
| 错误拒绝率（FRR） | 10-3 |
| 使用距离 | 20～50cm（可定制） |
| 额定功率（MAX） | (待测试) |
| 使用环境 | 非强光直射 |
| 显示 | 2.4寸LCD显示屏，320\*240pixel |
| 红外辅助照明 | 近红外LED×2(黑暗环境下可用) |
| 工作温度 | -20℃ ～ 55℃ |
| 储存温度 | -30℃ ～ 60℃ |
| 储存湿度 | < 80%RH |

# 硬件接口

## 3.1 IC8001核心板硬件接口

表3.1 IC8001核心板接口

|  |  |  |
| --- | --- | --- |
| **接口** | **接口型号** | **功能描述** |
| P3 |  | Boot模式选择，1：boot1模式 0：boot0模式 |
| P5 | 20Pin |  |
| P6 | 20Pin | 通讯与控制接口 |
| J3 | WP7\_S030VA1 | 摄像机模块数据接口（与摄像机模块J3相连） |

Boot方式选择

5vDC



JTAG

IC8001

烧录方式选择

J3

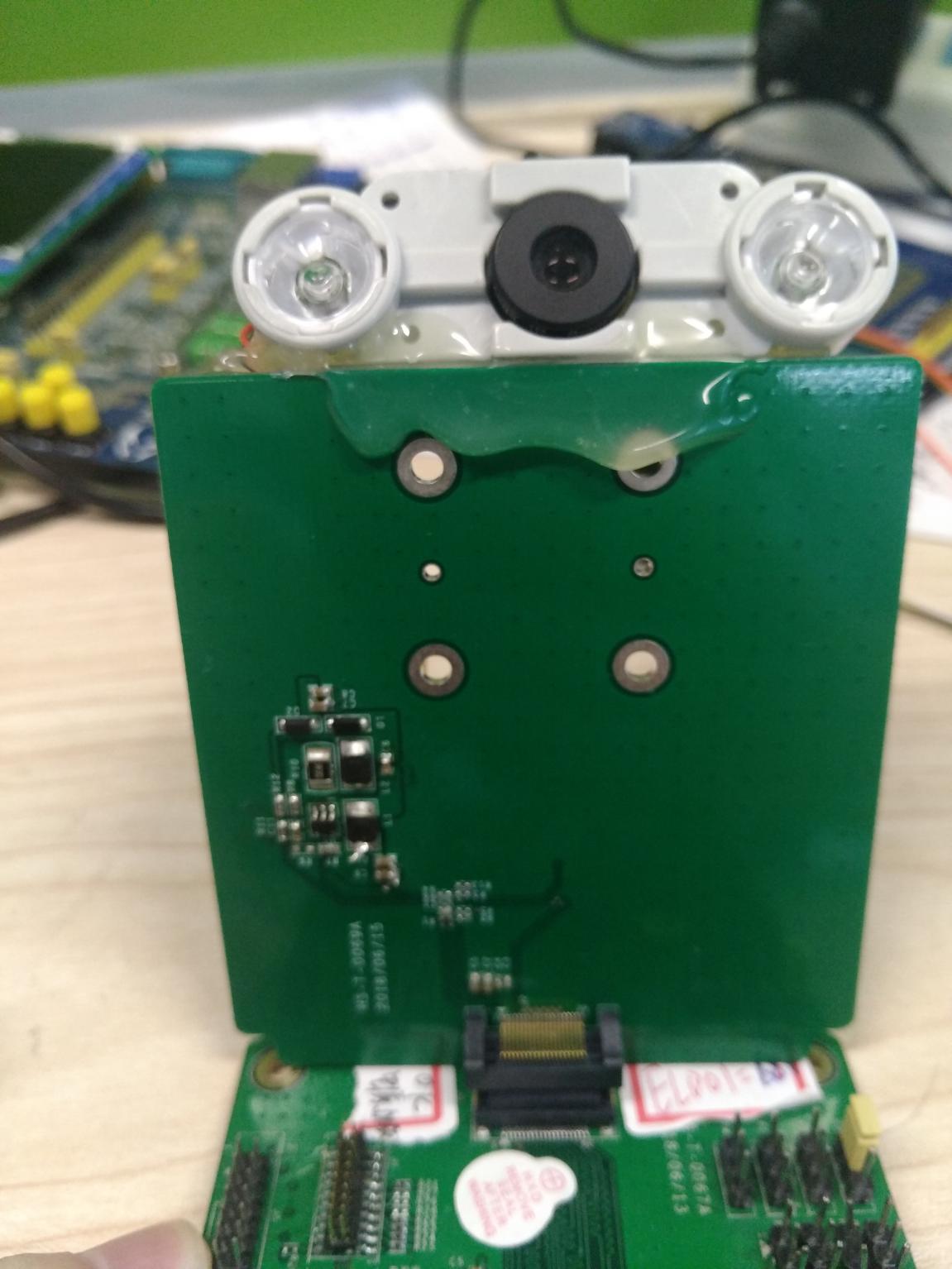
Sw1

MiniUSB

P6

P5

## 3.2 虹膜摄像机硬件接口



摄像机

## 3.3 M4核心板硬件接口



Sw6

Sw5

Sw4

Sw3

2.4寸LCD

红外灯

JTAG

Sw7

蜂鸣器

表3.2 M4核心板接口

|  |  |  |
| --- | --- | --- |
| **接口** | **接口型号** | **功能描述** |
| Sw3 | 轻触开关 | 识别按钮 |
| Sw4 | 轻触开关 | 删除全部模板 |
| Sw5 | 轻触开关 | 删除单个模板 |
| Sw6 | 轻触开关 | 注册按钮 |
| Sw7 | 拨码开关 | 选择通讯接口（默认都打到OFF位置） |

# 操作流程

## 4.1 UART通信方式流程

4.1.1 把M4核心板和IC8001核心板及相机按下图组装在一起，将拨码开关的1,3号位往上 拨到“ON”位置，其余位保持在“OFF”位置，M4核心板的UART与IC8001核心板的 UART连接在一起

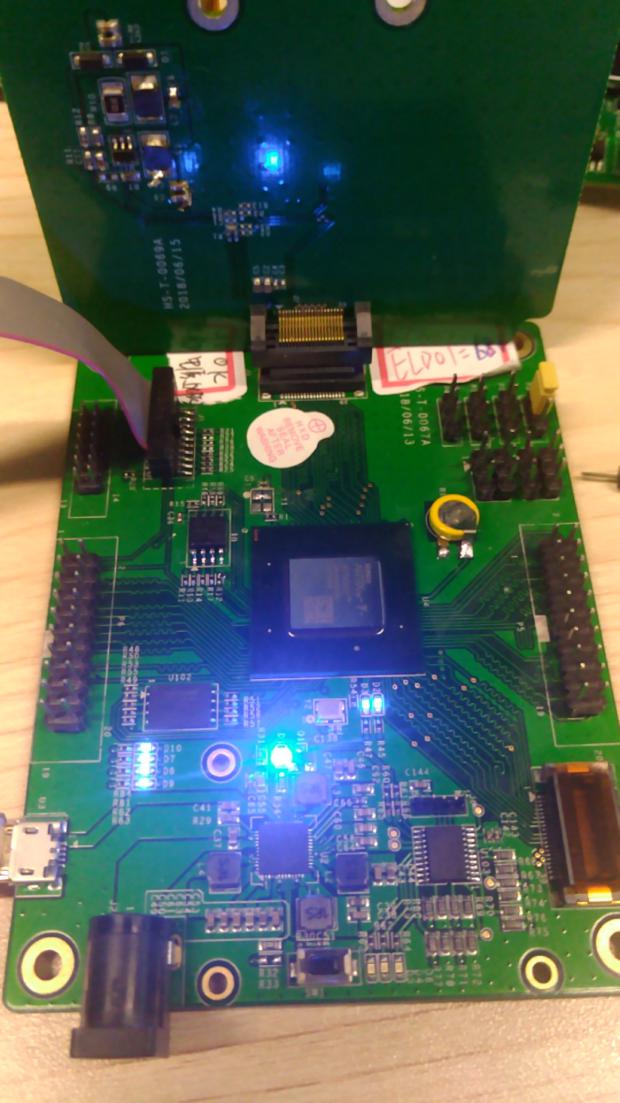


相机和红外灯模块

M4核心板

IC8001核心板

4.1.2 DEMO连接5v适配器供电，或者通过MiniUSB口供电，长按SW1 保持3秒,D1亮起表明IC8001核心板上电成功，等待大约10s以后，D2,D3,D7,D8,D9,D10也亮起表明芯片复位完成，连接JLINK调试器，此时可以开始给IC8001开发板烧写程序，如下图所示

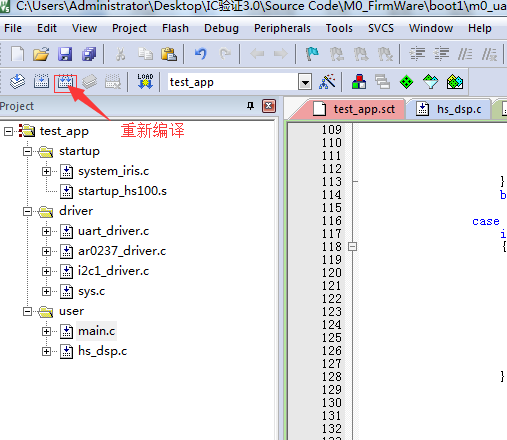


USB电源

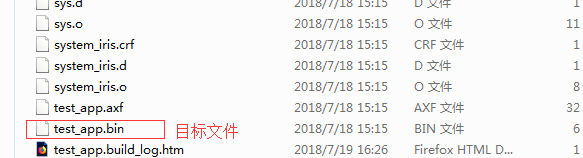
连接JLINK

4.1.3打开目录下的m0\_uart\_firmware工程（工程是在Keil5下创建的，关于keil5环境的安装在此不赘述，请自行百度），代码分为Release版本和Debug版本，分别对应boot0模式和boot1模式，通过P3跳帽可以切换boot模式

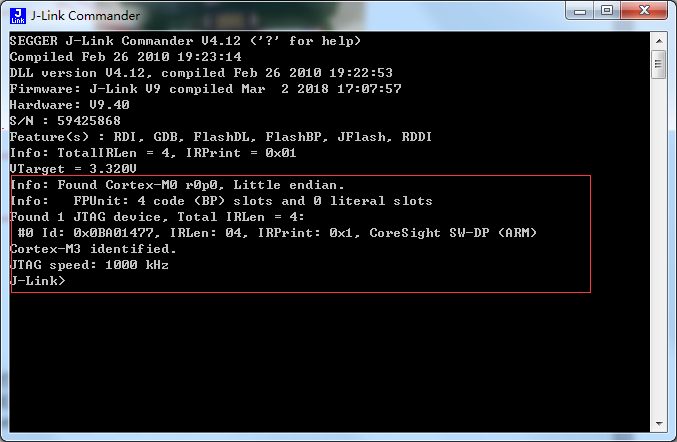
Boot0模式：即工作模式，打开Release版本工程，点击“重新编译按钮”如下图所示，



在工程下的output文件夹下找到“test\_app.bin”文件，

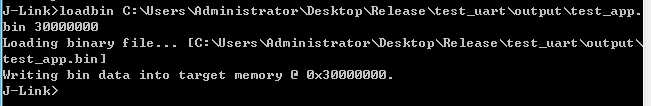


打开JLINK（关于JLINK驱动安装在此不赘述，请自行百度）下的J-Link Commander工具如下图



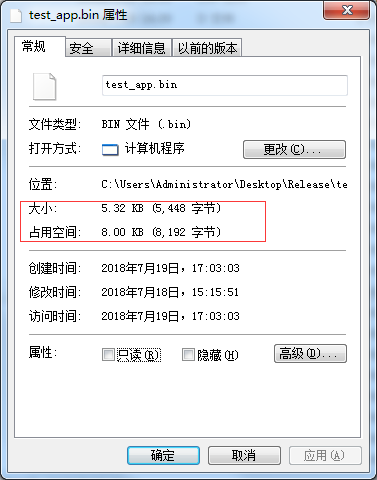
出现“Found Cortex-M0 r0p0”字样，表明JTAG连接成功，可以开始调试

在J-Link Commander中输入loadbin命令，然后输入需要导入的bin文件的完整路径，最后输入需要加载到的RAM地址，对于IC8001而言，此处加载地址为0x30000000，本例中，我们的输入如下图所示；



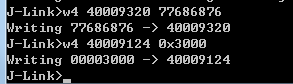
出现“Writing bin data into target memmory”字样表明bin数据已经加载在指定内存地址上

往codesize寄存器写bin文件的大小，查看bin文件属性可以知道bin文件大小，如下图所示



板上的flash每页大小为256byte，这点特别注意，写入codesize大小必须大于或等于代码所占flash空间大小（比如我们代码实际大小为5448byte，所占页数为5448/256=21.28，即所占页数为22个page，那么我们写入的codesize要大于或者等于256\*22=5632byte）,此处我们写入codesize大小为8k,即0x2000,另外根据bootloader代码的要求，需要在此大小上面增加4K大小，所以此处我们的codesize大小为0x3000,

写入codesize的值之前需要打开此寄存器的写使能，如下图所示



输入mem命令读codesize寄存器的值，如图表明codesize写入成功



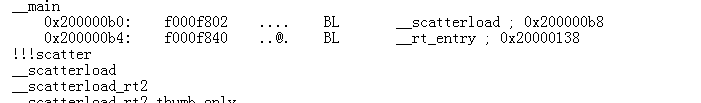
因为boot代码需要从0x30000000+codesize-0x1000这个地址取得MSP的值，由SCT文件和启动文件可以计算得到这个值为20018ff0（此处不赘述，只要不改启动文件和SCT文件，这个值可以认为就是0x20018ff0）



输入mem命令读取写入的值，如图表明MSP的值写入成功



因为boot代码需要从0x30000000+codesize-0x1000+0x04这个地址取得main入口地址的值，查看bin文件的反汇编查到此地址值为0x200000b4,如下图所示





输入mem命令读取写入的值，如图表明MSP的值写入成功



输入config device info命令将DSP相关寄存器配置写入到FLASH



输入mem命令查询DSP status。如下表明写入成功，否则写入失败



输入config usercode命令将bin文件写入到flash



输入mem命令查询DSP status。如下表明写入成功，否则写入失败



至此，usercode已经写入到FLASH中

Boot1模式：即调试模式，打开debug版本工程，点击“重新编译”按钮，然后点击“debug”按钮

