



**CUBIEBOARD**  
<http://cubie.cc>

*Fucking cool*, 深利的嵌入式技术和应用讨论社区

## Einstein-A20 软件开发指导

Website:<http://cubieboard.org/>

Email: [support@cubietech.com](mailto:support@cubietech.com)



文档版本	作者	修改内容	审核
V-0.2-20150916	Payne		



## 内容目录

1.概述.....	4
2.文档撰写的目的.....	4
3.适用人群.....	4
4.Einstein 最小系统软件生态.....	4
4.1.概述.....	4
4.2.Android .....	4
5.系统升级.....	5
6.刷机步骤.....	6
6.1 Ubuntu 12.04.....	6
6.2 window.....	8
7.基于 SDK 的二次开发.....	8
7.1.开发环境搭建.....	8
7.1.1.JDK 安装.....	9
7.1.2.在线安装相关工具包.....	9
7.2.Einstein-A20 android SDK.....	10
7.3.SDK 编译.....	10
7.3.1. 编译 Lichee .....	10
7.3.2.编译 Android .....	11
8. 软件 SDK 开发指导.....	11
9.硬件开发指导.....	11



## 1. 概述

Einstein-A20 的主控采用全志 A20 芯片，采用双核 Cortex-A7 架构的 CPU，主频可达 1G (1008MHz)，功耗控制出色。图形方面，GPU 采用 Mali400MP2，兼容性更加出色。最高支持 2160P 的视频解码和 1080P@30fps 的编码，多媒体性能优异。A20 支持 1G 内存，另外 A20 还支持 1024x768 或 1024x600 等多种分辨率。

## 2. 文档撰写的目的

Einstein-A20 核心板为广大嵌入式技术开发者提供了便利，大大降低了产品开发难度，提高了项目开发成功率。通过该文档可以进一步提高软硬件开发效率。

## 3. 适用人群

Einstein-A20 为最小核心板，开发者可自行根据自己需求开发设计底板。Einstein-A20 板载 WIFI 模块，蓝牙模块，还支持丰富的 Camera 模块，3G 模块（电话系统），TF (SD) 卡扩展模块等以及多种传感器。可以广泛应用于工控，医疗，教育，通讯，家庭多媒体，互联网应用，车载，广告等多个领域。该文档适用于所有在此应用领域开发设计的人员。

## 4. Einstein 最小系统软件生态

### 4.1. 概述

Einstein-A20 最小系统 SDK 是基于 Cubieboard2 基础上，默认系统有 wifi、蓝牙、uart、adb 调试和串口调试等可用以及输出显示支持 av 和 hdmi 输出。

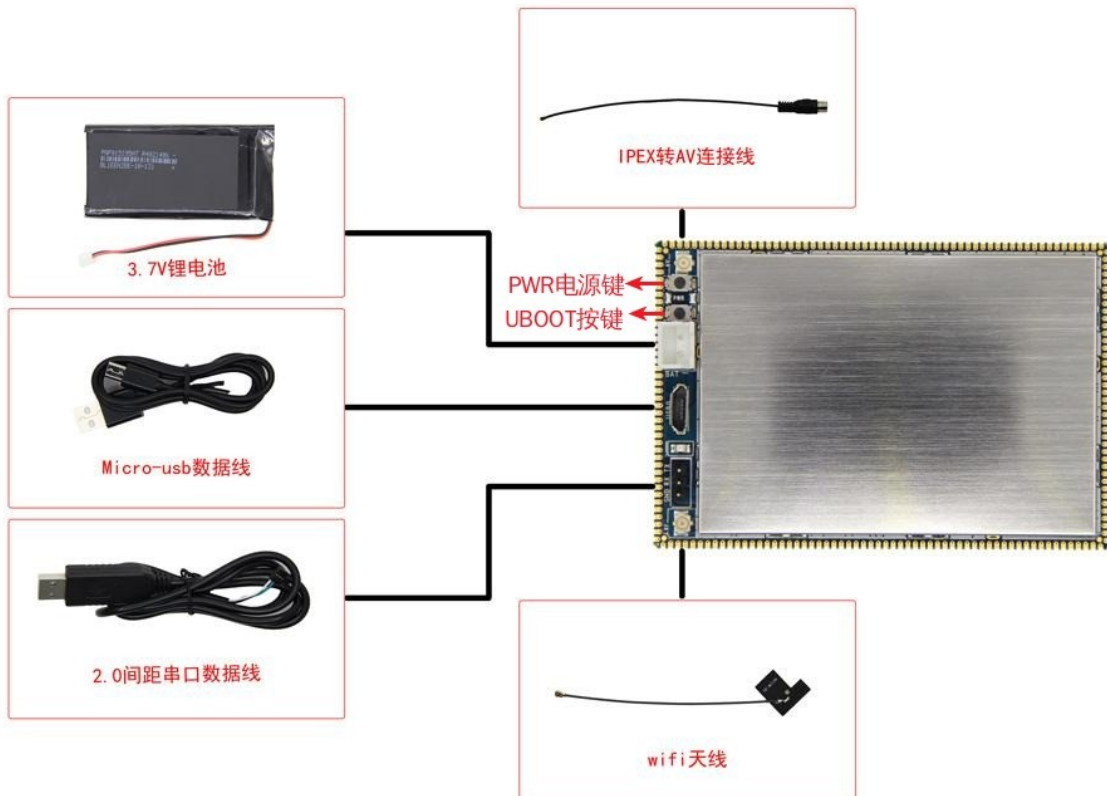
### 4.2. Android

A20 主控的核心板系统和软件平台是建立在 Android 4.2 平台基础上，linux 内核版本为 3.4。Android 生态系统支持影音，网络，娱乐，系统管理，个人助手等丰富的扩展。



## 5. 系统升级

Einstein-A20 板载 avout 接口，power 按键，UBOOT 按键，电池接口，Micro-usb 接口，uart0 串口以及 wifi 天线接口。烧写固件使用的是 Micro-usb 接口及 Micro-usb 数据线。



www.cubietech.com  
http://cubieboard.org

Einstein-A20 android 固件下载地址：[android4.2\\_einstein\\_A20](http://android4.2_einstein_A20)

固件解压：7z x android4.2\_A20\_Einstein\_v1.0.img.7z

当然爱好者也可以自行用解压工具解压



Windows 升级工具地址：<http://pan.baidu.com/s/1fbbGy>

Linux 刷机工具下载地址：

64 位操作系统：<http://pan.baidu.com/s/1XrsU>

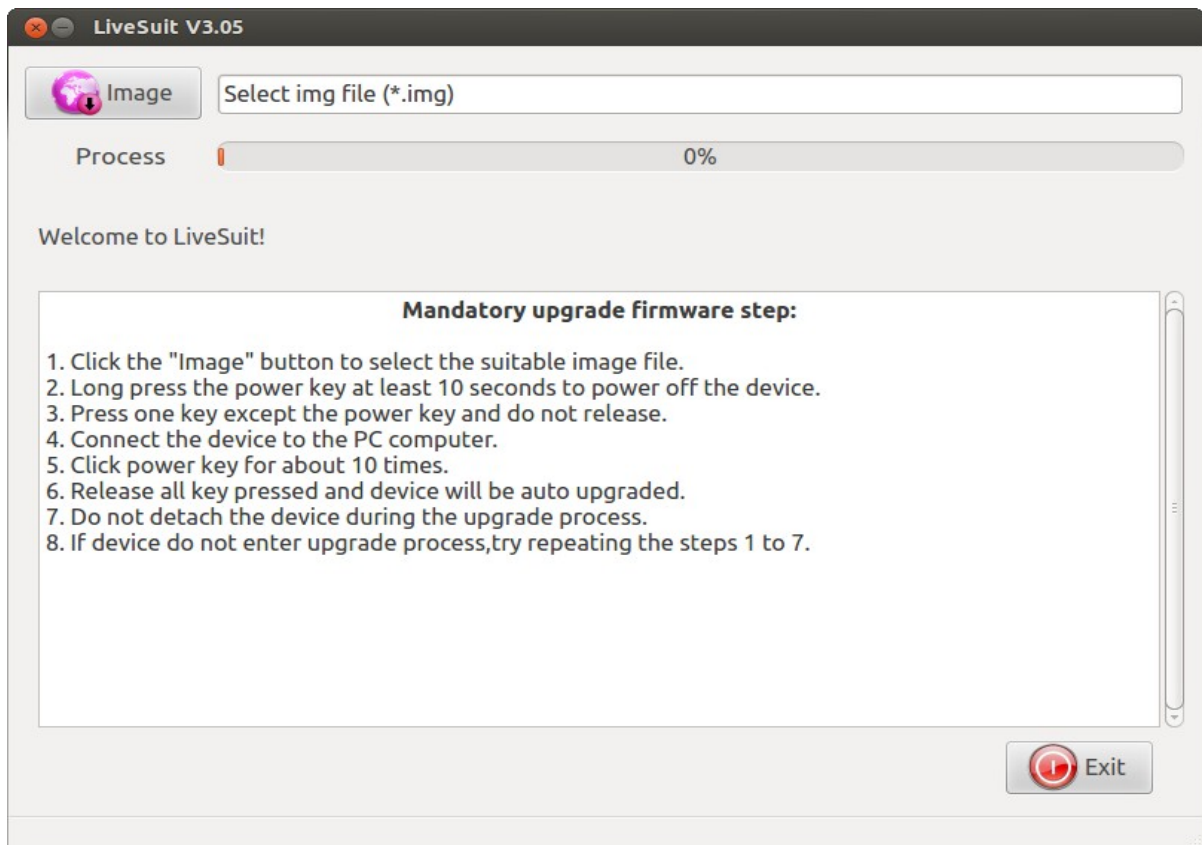
32 位操作系统 <http://pan.baidu.com/s/1eQnGzIC>

注：压缩包中自带刷机工具的安装和使用教程，这里就不再详细说明。

## 6. 刷机步骤

### 6.1 Ubuntu 12.04

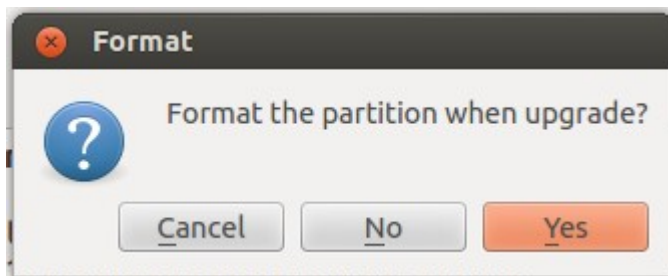
1)运行 Livesuit



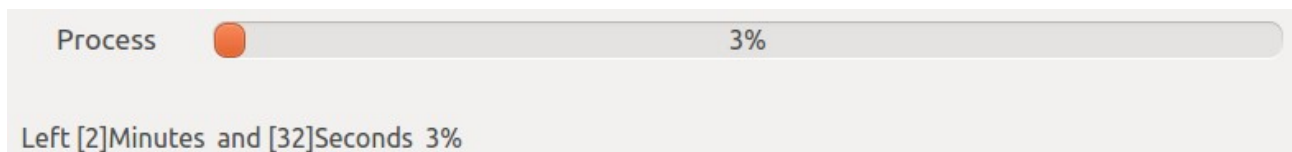


## 2)选择固件

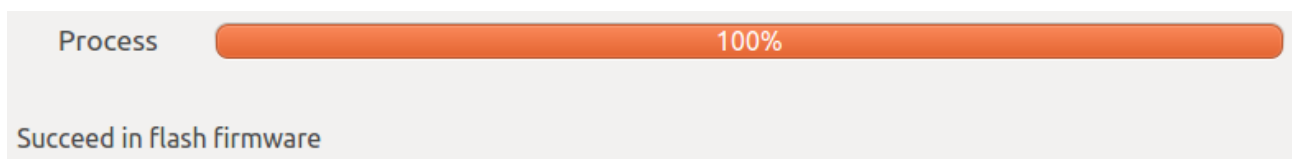
鼠标点击上图的 Image 按钮选择下载解压后的固件，按住板子的 UBOOT 按键进入刷机模式，Micro-usb 数据线的 USB 接口端连接 PC 机的 USB 接口，另一端连接 Einstein-A20 的 Micro-usb 刷机接口。弹出提示框即可放开 UBOOT 键，按“**Yes**”，Livesuit 将开始烧写固件到 EMMC。



正在刷机：



刷机成功：



刷完拔掉刷机线，插上电源即可上电运行



## 6.2 window

- 1) 运行 PhoenixSuit
- 2) 点击“一键刷机”
- 3) 选择固件,其余步骤与 Ubuntu 12.04 一样



## 7. 基于 SDK 的二次开发

这里提供出厂固件对应的 android SDK 和 SDK 开发指导，开发人员可根据此文档进行一系列的嵌入式软件开发。

### 7.1. 开发环境搭建

开发环境的搭建包括交叉编译环境的搭建和烧录工具的安装，推荐使用 Ubuntu12.04 64 位操作系统。下面以 Ubuntu12.04 64 位为例,搭建开发环境。





### 7.1.1.JDK 安装

```
$ wget dl.cubieboard.org/software/tools/android/jdk1.6.0_45.tar.gz
```

```
解压 : $sudo tar -zxvf jdk1.6.0_45.tar.gz
```

配置环境变量 :

```
$ sudo vim ~/.bashrc
```

增加 :

```
JAVA_HOME=/jdk_path
```

```
export JRE_HOME=/jdk_path/jre
```

```
export PATH=$JAVA_HOME/bin:$JRE_HOME/bin:$PATH
```

jdk\_path 为你下载解压后的 jdk 路径。

保存后使其生效且验证是否成功 :

```
$source ~/.bashrc&&java -version
```

成功的返回值如下:

```
java version "1.6.0_45"
```

```
Java(TM) SE Runtime Environment (build 1.6.0_45-b06)
```

```
Java HotSpot(TM) 64-Bit Server VM (build 20.45-b01, mixed mode)
```

### 7.1.2.在线安装相关工具包

```
$sudo apt-get update
```

```
$sudo apt-get upgrade
```

```
$sudo apt-get install build-essential u-boot-tools uboot-mkimage binutils-arm-linux-gnueabi
```

```
$sudo apt-get install gcc g++ gcc-arm-linux-gnueabi gcc-arm-linux-gnueabi-g++-multilib
```

```
$sudo apt-get install cpp-arm-linux-gnueabi libusb-1.0-0 libusb-1.0-0-dev wget fakeroot
```

```
$sudo apt-get install kernel-package zlib1g-dev libncurses5-dev build-essential
```

```
$sudo apt-get install texinfo texlive ccache zlib1g-dev gawk bison flex gettext uuid-dev
```

Website:<http://cubieboard.org/>

Email: [support@cubietech.com](mailto:support@cubietech.com)



```
$sudo apt-get install ia32-libs git gnupg flex bison gperf build-essential zip  
$sudo apt-get install curl libc6-dev x11proto-core-dev libx11-dev:i386 lib32ncurses5-dev  
$sudo apt-get install libreadline6-dev:i386 mingw32 tofrodos python-markdown  
$sudo apt-get install libxml2-utils xsltproc zlib1g-dev:i386 libgl1-mesa-dev
```

## 7.2. Einstein-A20 android SDK

该服务器节点存储了 Einstein-A20 所有的 Android 源码, 如存在有不同版本源码, 请下载最新版本的代码. 目前的 Android 源码只支持源码压缩包的方式进行更新。

下载链接: [Einstein-A20 android SDK](#)

## 7.3. SDK 编译

源码的编译过程大致如下: 先编译 lichee, 这部分包括 U-boot, 内核和驱动, 然后再编译 Android 层的代码, 编译步骤如下所示。

### 7.3.1. 编译 Lichee

拷贝 **kernel\_config**

```
$ cd lichee/linux-3.4/  
$cp arch/arm/configs/sun7i_einstein_core_defconfig .config
```

编译 **lichee**

```
$cd ..  
$./build.sh -p sun7i_android
```

出现下面所示表示编译成功

```
INFO: build u-boot OK.  
INFO: build rootfs ...  
INFO: skip make rootfs for android  
INFO: build rootfs OK.  
INFO: build lichee OK.
```



### 7.3.2.编译 Android

```
$cd android
$source build/envsetup.sh
$lunch sugar_einstein-eng
$extract-bsp
$make -j8 // -j* 建议等于 PC 的 CPU 核数以提高编译效率和速度
$pack
```

最终打包出来的固件在:

lichee/tools/pack/sun7i\_android\_sugar-einstein.img

## 8. 软件 SDK 开发指导

软件开发指导参考：[A20 软件开发指导](#)

## 9. 硬件开发指导

硬件开发指导参考：[Einstein-A20 模块规格书](#)