
SparkRoad Brief

Designed by Verimake

2019 年 5 月

目 录

第一章 SparkRoad 开发板简介.....	3
1.1 开发板资源.....	3
1.2 主控 FPGA 资源.....	4
1.3 开发资料获取方式.....	4
第二章 开发环境准备.....	5
2.1 Windows 环境.....	5
2.2 Ubuntu 环境.....	6
第三章 Hello LED 开发板功能演示.....	8
3.1 Windows 环境.....	8
3.2 Ubuntu 环境.....	9
第四章 Hello RISC-V 软核开发.....	10
4.1 Windows 环境.....	10
4.2 Ubuntu 环境.....	12
第五章 开发板资源介绍.....	14
5.1 电源拓扑.....	14
5.2 按键和 RGB LED.....	15
5.3 micro SD 接口.....	15
5.4 ESP32-S module 接口.....	16
5.5 FPGA 管脚资源分配表.....	17
5.6 micro:bit 拓展接口.....	19
版本信息.....	19

第一章 SparkRoad 开发板简介

1.1 开发板资源

- 1) 平台通过 USB-C 提供稳定电源。
- 2) 板载 USB JTAG 可支持 FPGA 下载、调试、flash 烧录等功能
- 3) 两个用户按键
- 4) 三色 RGB LED 灯
- 5) FPGA 配置 flash，六线连接 FPGA
- 6) 8-Pad WSON Package 六线 flash 接口
- 7) 六线 microSD 卡接口
- 8) DIP-38 Package ESP32-S 接口
- 9) FPGA 芯片内嵌 64Mbit SDRAM
- 10) 0.5mmFPC-24P 标准 DVP Camera 接口
- 11) micro: bit 金手指兼容大量 BBC 外设

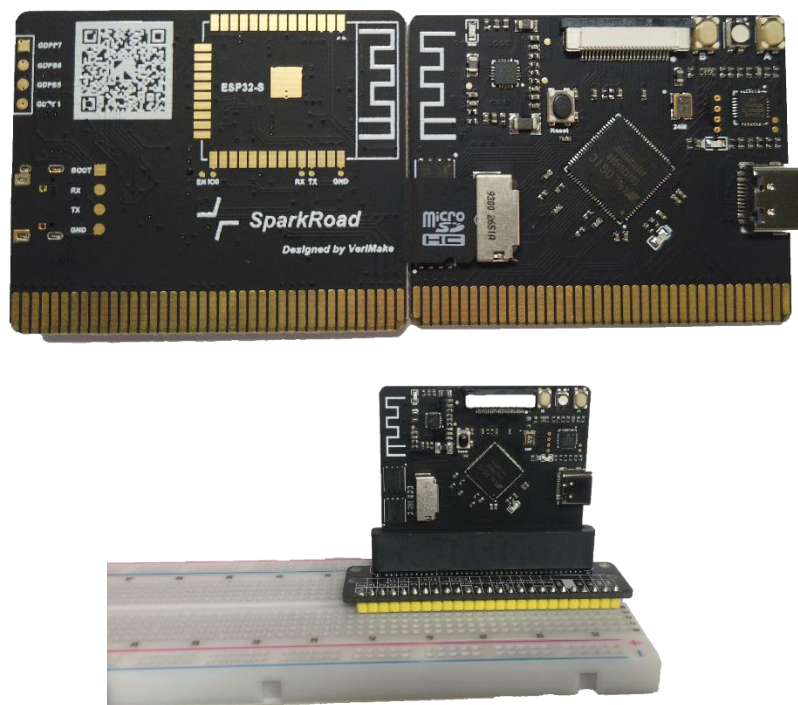


图 1.1-1 SparkRoad 实物图

1.2 主控 FPGA 资源

SparkRoad 开发板主控采用上海安路科技的高性能 FPGA 产品 EG4S20NG88。

- 1) 等效 23520 个四输入查找表
- 2) 最大 156.8K 分布式 RAM
- 3) 64 块 9Kb 嵌入式 ram 和 16 块 32Kb 嵌入式 ram
- 4) 64Mbit SDRAM, 最高 200MHz 工作频率
- 5) 16 个全局时钟, 3 个 PLL
- 6) 1MSPS 12-bit SAR 型 ADC

1.3 开发资料获取方式

Github 网址: <https://github.com/verimake-team/SparkRoad-FPGA>

微信公众号: VeriMake

Bilibili ID: VeriMake



第二章 开发环境准备

2.1 Windows 环境

● TD 软件安装

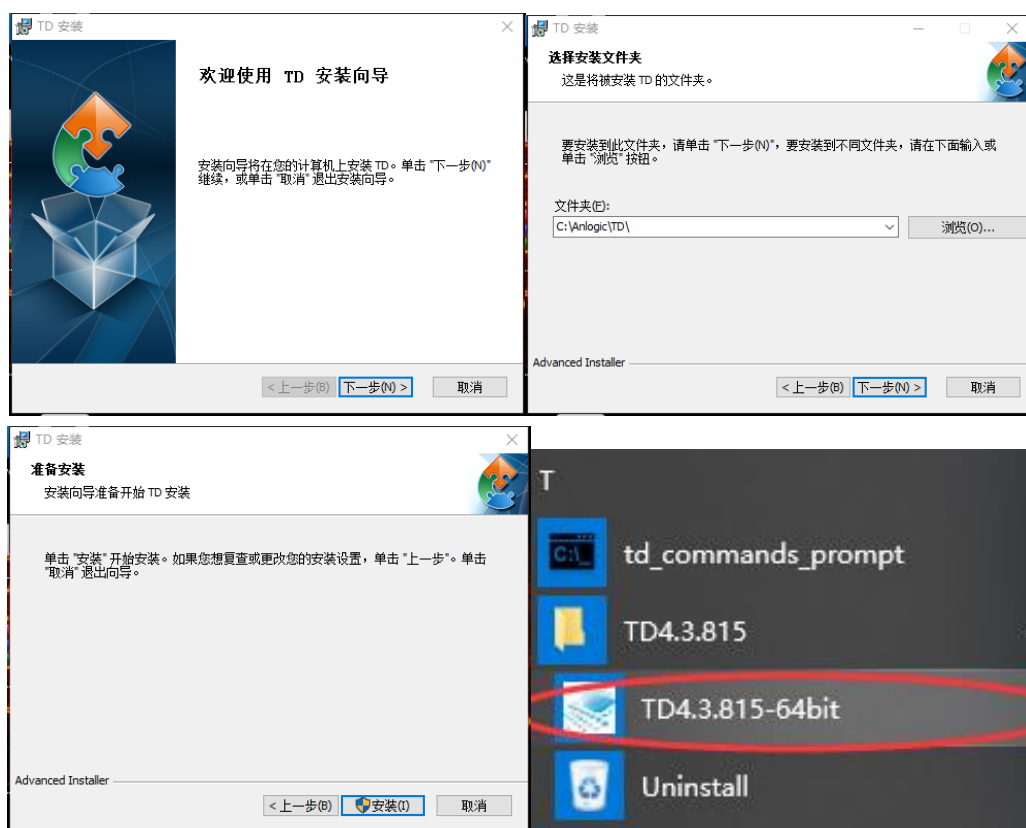


图 2.1-1 windows 环境 TD 安装流程

● 开发板驱动安装

- 1) 开发板通过 USB-C 线连接到电脑。右键“计算机->设备管理器”，在弹出的新设备中点右键选择“更新驱动程序软件”，在弹出的选项中点击“浏览计算机以查找驱动程序软件”。



→ 自动搜索更新的驱动程序软件(S)

Windows 将搜索你的计算机和 Internet 以获取适合你设备的最新驱动程序软件，除非你已在设备安装设置中禁用此功能。

→ 浏览我的计算机以查找驱动程序软件(R)

手动查找并安装驱动程序软件。

- 2) 选择 TD 软件安装路径下的“driver”目录，并点击下一步完成更新。

浏览计算机上的驱动程序

在以下位置搜索驱动程序:

C:\Anlogic\TD4.3.815\driver\win8_10_64

浏览(R)...

☒ 包括子文件夹(I)

- 3) 进入启动选项，禁用 windows 强制数字签名（也可永久禁用避免重复操作，安全后果自负）。“win 键->shift 键+重启->疑难解答->高级选项->启动设置->禁用驱动程序强制签名”

2.2 Ubuntu 环境

● TD 软件安装

- 1) 确保和 TD 程序的压缩包在同一目录下
- 2) 在/opt 中创建 TD IDE 的安装位置

```
sudo mkdir /opt/TD_DECEMBER2018
```

- 3) 即将 TD 提取到创建的目录

```
sudo tar -xvf TD_DECEMBER2018_GOLDEN_RHEL.tar.gz -d  
/opt/TD_DECEMBER2018/
```

-
- 4) 在/usr/bin/td 目录下创建一个链接文件来执行/opt/TD_DECEMBER2018/bin/td

```
sudo ln -s /opt/TD_DECEMBER2018/bin/td /usr/bin/td
```

- 5) 以 GUI 模式打开 TD IDE

```
td -gui
```

● 开发板驱动安装

- 1) 开发板通过 USB-C 线连接到电脑。执行

```
lsusb
```

```
vertmake@ubuntu:~$ lsusb
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 002 Device 005: ID 0547:1002 Anchor Chips, Inc. Python2 WDM Encoder
Bus 002 Device 003: ID 0e0f:0002 VMware, Inc. Virtual USB Hub
Bus 002 Device 002: ID 0e0f:0003 VMware, Inc. Virtual Mouse
Bus 002 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
```

确保 USB 设备的 VID: PID 为 0547: 1002

- 2) 创建 udev rule file

```
sudo vi /etc/udev/rules.d/91-anlogic-jtag.rules
```

- 3) 输入以下信息

```
SUBSYSTEMS=="usb", ATTRS{idVendor}=="0547",
ATTRS{idProduct}=="1002", \
    GROUP="plugdev", \
    MODE="0660"
```

- 4) 重启 udev service

```
sudo service udev restart
```

- 5) 查看 TD IDE 能否识别驱动设备

```
./td -gui
```

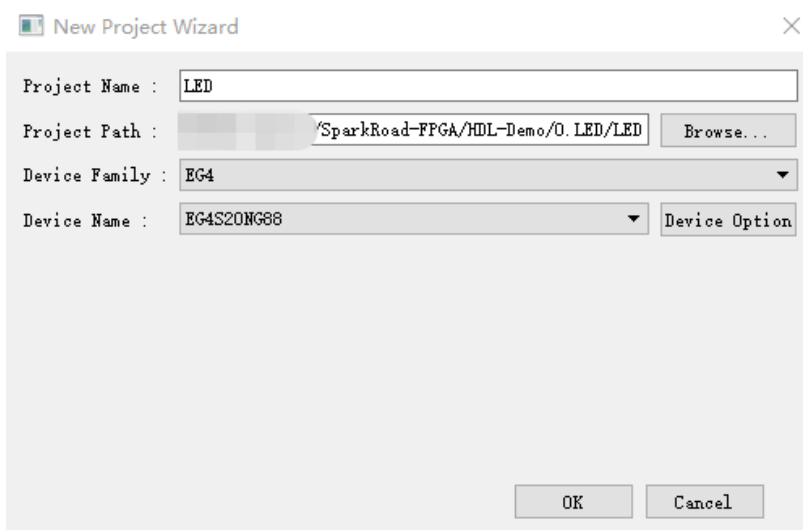
点击”tools->download”查看是否识别出 usb 设备

第三章 Hello LED 开发板功能演示

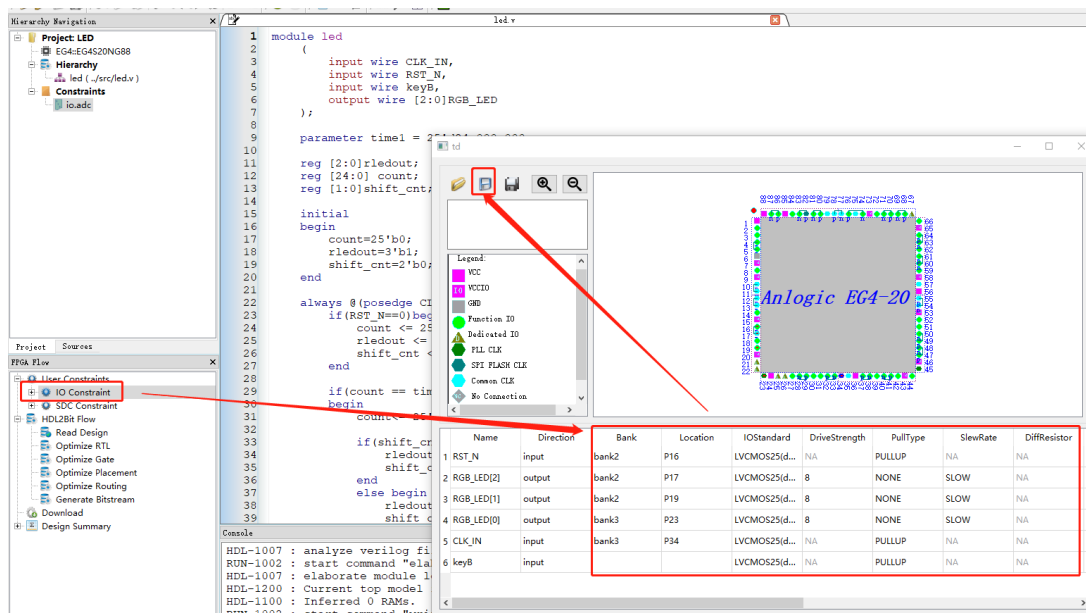
此处添加 led 演示流程，包括建立工程、编写代码、编译、下载、演示

3.1 Windows 环境

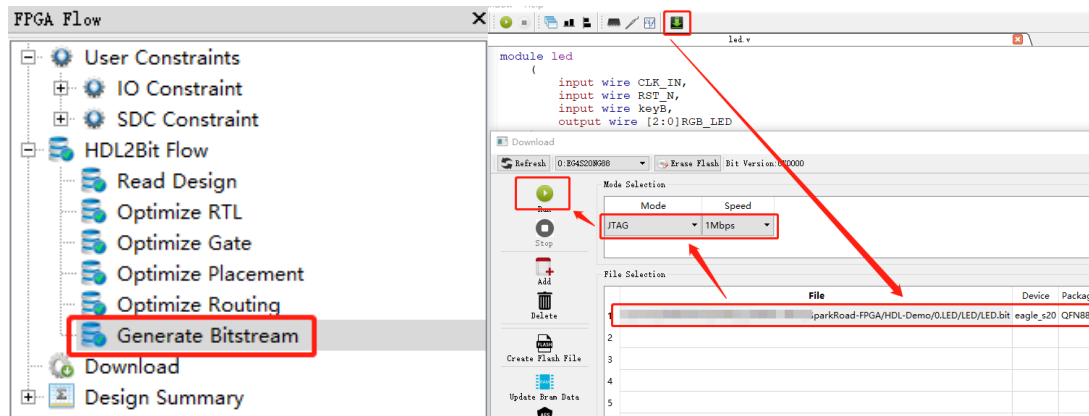
- 1) 打开 TD 软件，顶部菜单选择“Project->New Project”按照下图，方式新建工程。目录自选



- 2) 顶部工具栏“Source->Add Source”选择/SparkRoad/HDL-Demo/0.LED/src 目录下的 led.v 添加进工程之后，再按照下图方式添加并保存引脚约束文件



- 3) 综合工程生成 bitstream 文件，将 SparkRoad 用 USB 线缆连接到电脑上，并按照下图方式将 bit 文件下载到 SparkRoad 上。下载选项可选 Jtag 仿真或者 Flash 下载。



3.2 Ubuntu 环境

```
./td -gui
```

启动 TD-gui 界面，其余流程与 windows 一致。

第四章 Hello RISC-V 软核开发

4.1 Windows 环境

- 1) 下载安路科技提供的 riscv-toolchain.exe 并安装

链接: https://pan.baidu.com/s/160Iu03p4Nv1cNUw18_msDQ
提取码: fv59

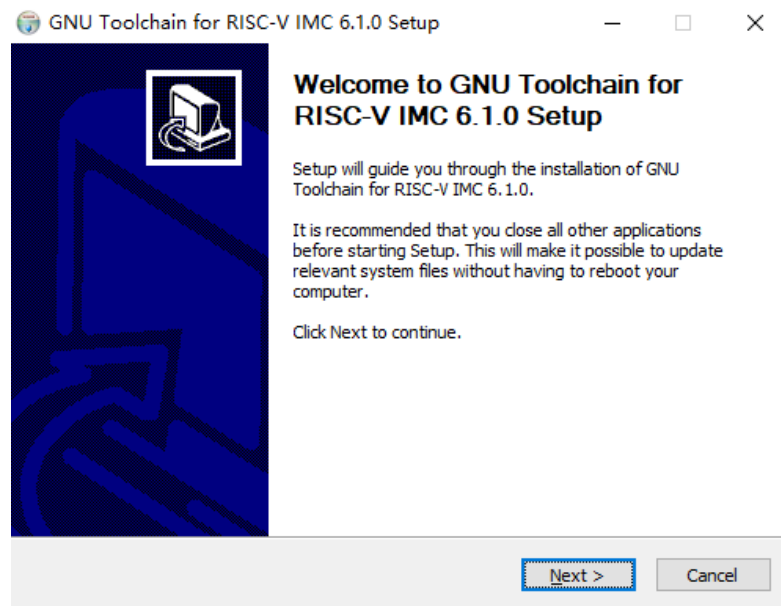


图 3.1-1 GNU toolchain for RISC-V

- 2) 配置环境变量

右键“计算机->属性->高级系统设置”把上面 toolchain 的安装目录下的 bin 目录添加到 PATH

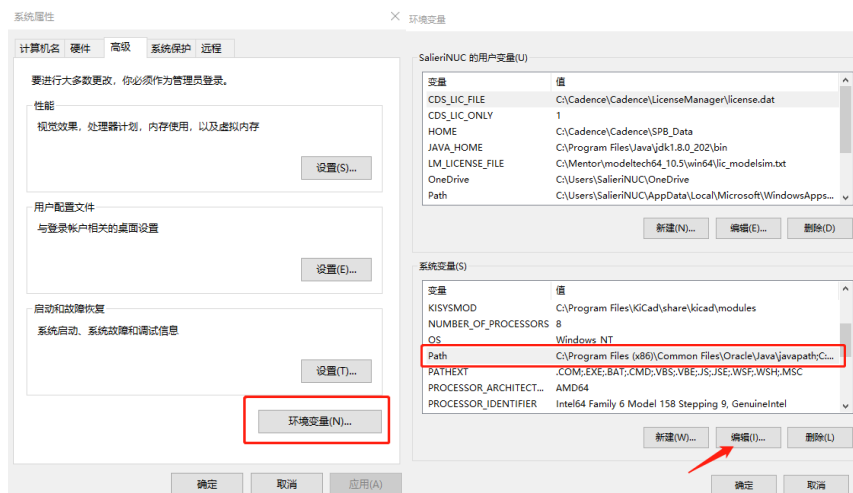
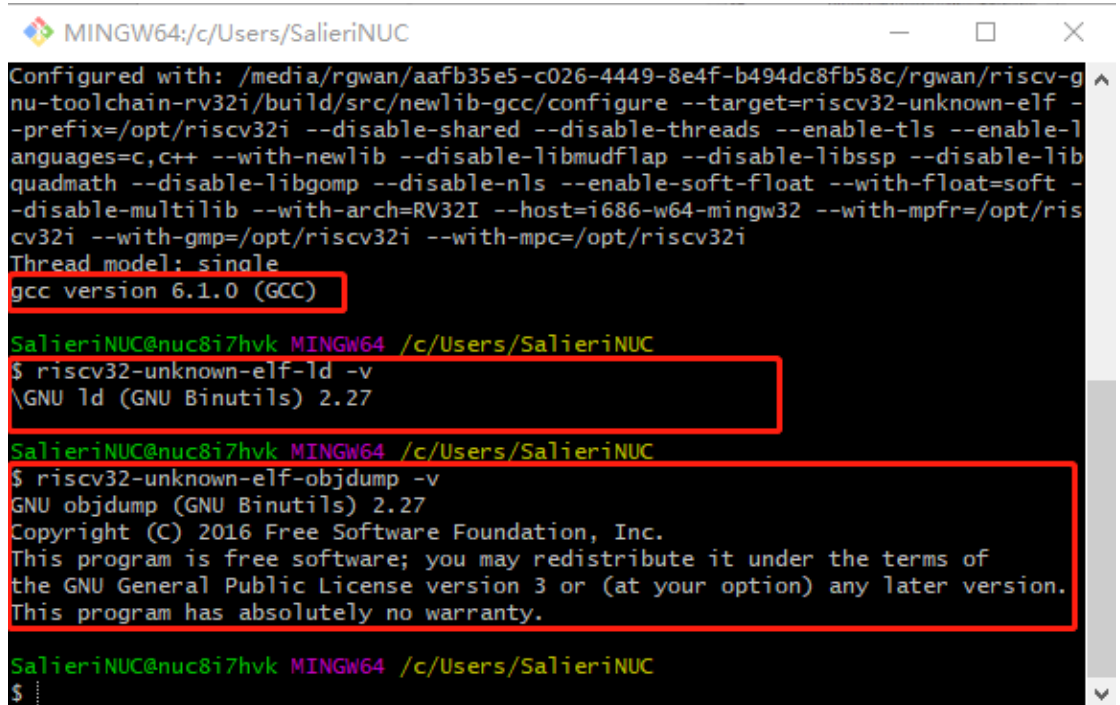


图 3.1-2 windows 环境添加环境变量

3) 在任意 terminal 中测试 toolchain



```
MINGW64:/c/Users/SalieriNUC
Configured with: /media/rgwan/aafb35e5-c026-4449-8e4f-b494dc8fb58c/rgwan/riscv-g
nu-toolchain-rv32i/build/src/newlib-gcc/configure --target=riscv32-unknown-elf -
-prefix=/opt/riscv32i --disable-shared --disable-threads --enable-tls --enable-l
anguages=c,c++ --with-newlib --disable-libmudflap --disable-libssp --disable-lib
quadmath --disable-libgomp --disable-nls --enable-soft-float --with-float=soft -
-disable-multilib --with-arch=RV32I --host=i686-w64-mingw32 --with-mpfr=/opt/ris
cv32i --with-gmp=/opt/riscv32i --with-mpc=/opt/riscv32i
Thread model: single
gcc version 6.1.0 (GCC)

SalieriNUC@nuc8i7hvk MINGW64 /c/Users/SalieriNUC
$ riscv32-unknown-elf-ld -v
GNU ld (GNU Binutils) 2.27

SalieriNUC@nuc8i7hvk MINGW64 /c/Users/SalieriNUC
$ riscv32-unknown-elf-objdump -v
GNU objdump (GNU Binutils) 2.27
Copyright (C) 2016 Free Software Foundation, Inc.
This program is free software; you may redistribute it under the terms of
the GNU General Public License version 3 or (at your option) any later version.
This program has absolutely no warranty.

SalieriNUC@nuc8i7hvk MINGW64 /c/Users/SalieriNUC
$
```

图 3.1-3 Git bash 演示

4) 从 github 上 clone 资料包

```
git clone https://github.com/verimake-team/SparkRoad-FPGA.git
```

进入软件代码目录

```
\SparkRoad-FPGA\RISC-V\firmware
```

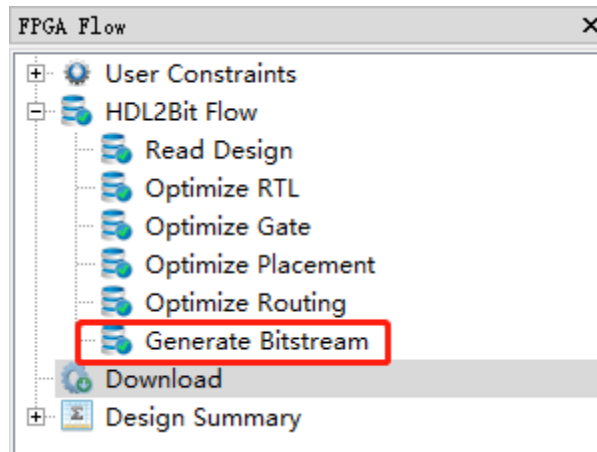
在此目录下，依次用 terminal 执行以下指令编译软件代码（以 git bash 为例）

```
riscv32-unknown-elf-gcc -ffreestanding -nostdlib -o
firmware.elf firmware.S firmware.c -march=rv32i --std=gnu99 -
Wl,-Bstatic,-T,firmware.lds,-Map,firmware.map,--strip-debug -
lgcc
riscv32-unknown-elf-objcopy -O binary firmware.elf
firmware.bin
rom2mif firmware.bin hi.mif mh.mif ml.mif lo.mif
mv *.mif ../hardware/Picorv32/al_ip/
```

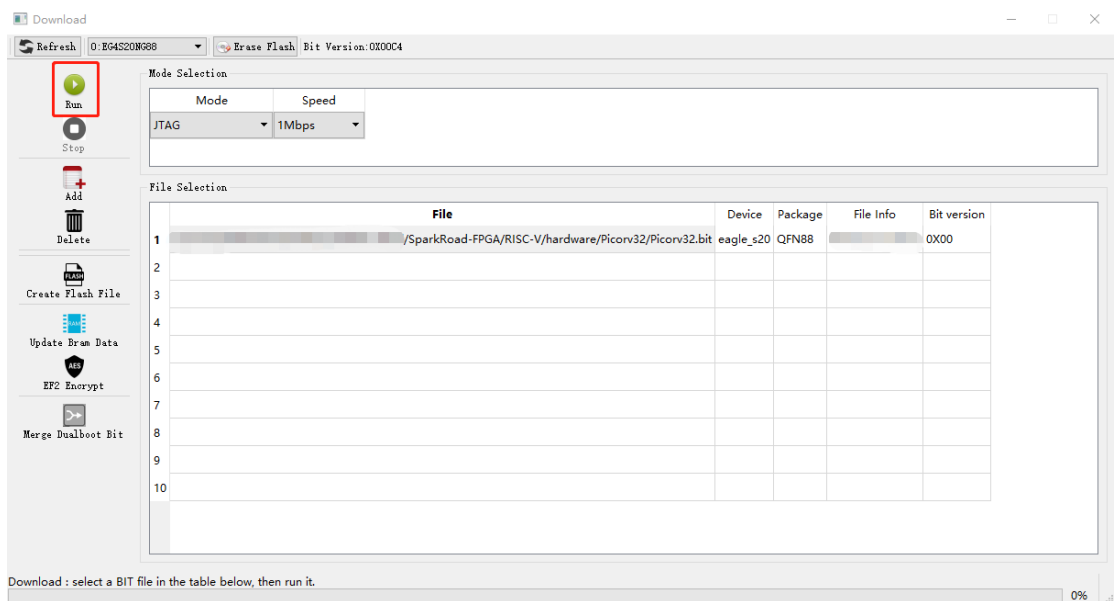
5) 进入 RTL 代码目录

```
\SparkRoad-FPGA\RISC-V\hardware\Picorv32
```

用 TD 工具打开工程文件 Picorv32.al 对 FPGA 工程进行综合



随后从顶部菜单进入“Tools->Download”仿真/下载上一步生成的 Bit 文件



将金手指的 P13 (TX) 和 P14 (RX) 连接上 USB-TTL 串口。打开任意串口助手类工具，按下 SparkRoad 最右端按键，复位系统即可看见以下 log

```
PicoRV32 (RV32IMC ISA) running on SparkRoad(EG4S20NG88)
Apr.2019 Serial port
www.socchina.net
```

4.2 Ubuntu 环境

- 1) 获取软件包并进入目录

```
git clone https://github.com/verimake-team/SparkRoad-FPGA.git
cd SparkRoad-FPGA
```

- 2) 安装编译环境。由于网络环境问题，我们也额外提供一份百度云云盘。

链接: https://pan.baidu.com/s/160Iu03p4Nv1cNUw18_msDQ
提取码: fv59

-
- 3) 将百度云盘下载好的 riscv-gnu-toolchain-rv32i.tar.gz 放在/SparkRoad-FPGA 目录下再依次执行以下命令

```
sudo mkdir /opt/riscv32i
sudo chown $USER /opt/riscv32i
make get-tools
sudo tar -zxvf riscv-gnu-toolchain-rv32i.tar.gz
sudo chown $USER riscv-gnu-toolchain-rv32i
cd riscv-gnu-toolchain-rv32i; mkdir build; cd build
../configure --with-arch=rv32i --prefix=/opt/riscv32i
make -j$(nproc)
cd ../../RISC-V/firmware
gcc -Wall rom2mif -o rom2mif
sudo mv rom2mif /opt/riscv32i/bin
```

- 4) 执行完以上步骤之后，编译环境就已经被安装在/opt/rv32i 目录下。只需要再配置环境变量即可(配置完环境变量后，系统用户重新登陆才能生效)。

```
vi /etc/profile
```

并在末尾加入

```
export PATH="$PATH:/opt/riscv32i/bin"
```

- 5) 系统重新登陆后进入/SparkRoad-FPGA/RISC-V/firmware 依次执行以下指令编译软件代码

```
make clean
make firmware.elf
make firmware.bin
make lo.mif
```

- 6) 进入 RTL 工程目录/SparkRoad-FPGA/RISC-V/hardware/Picorv32。之后的流程及现象与 windows 环境下相同，参考上一节。

第五章 开发板资源介绍

5.1 电源拓扑

开发板采用 USB-C 供电，一般使用普通电脑 USB 口供电足够，若焊接了 ESP32-S 模块，可能会超过电脑 USB 供电能力，供电不足会导致开发板不稳定，可通过 micro: bit 上的金手指提供外部 5V 供电。

开发板的 5V 电源输入通过三路 Buck DC-DC 电源管理芯片 EA3036 转换，其中两路转换为 3.3V 和 1.2V 供 FPGA 主控使用。额外一路 Buck 输出连接到 ESP32-S 模块的供电接口上，确保 WIFI 能够稳定工作。

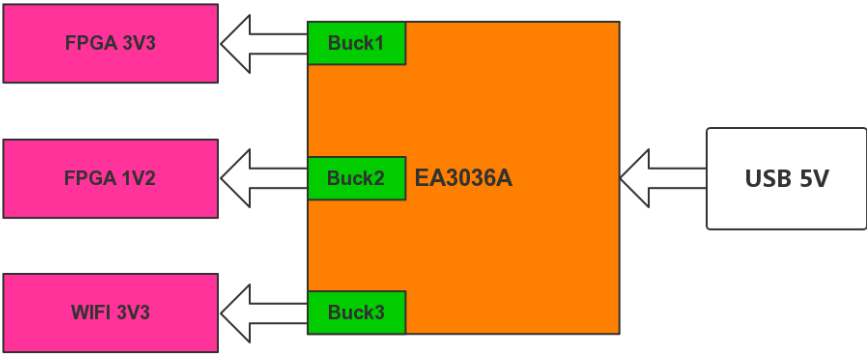


图 5.1-1 EA3036 电源拓扑

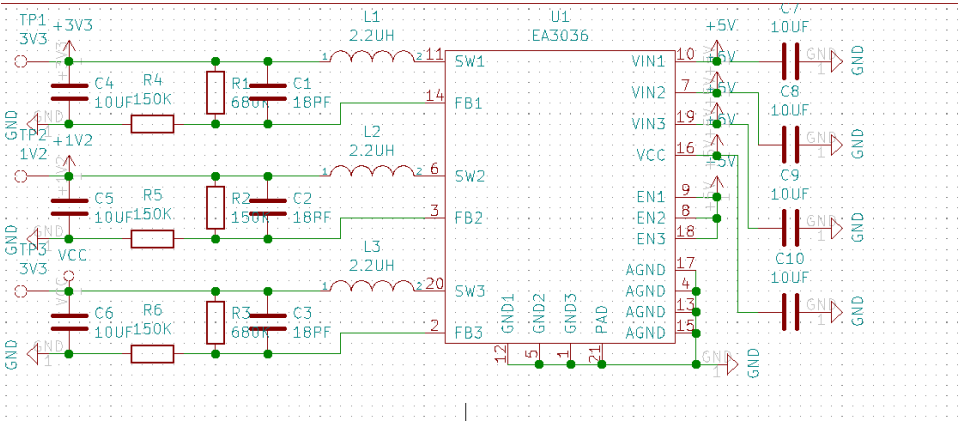


图 5.1-2 EA3036 硬件原理图

5.2 按键和 RGB LED

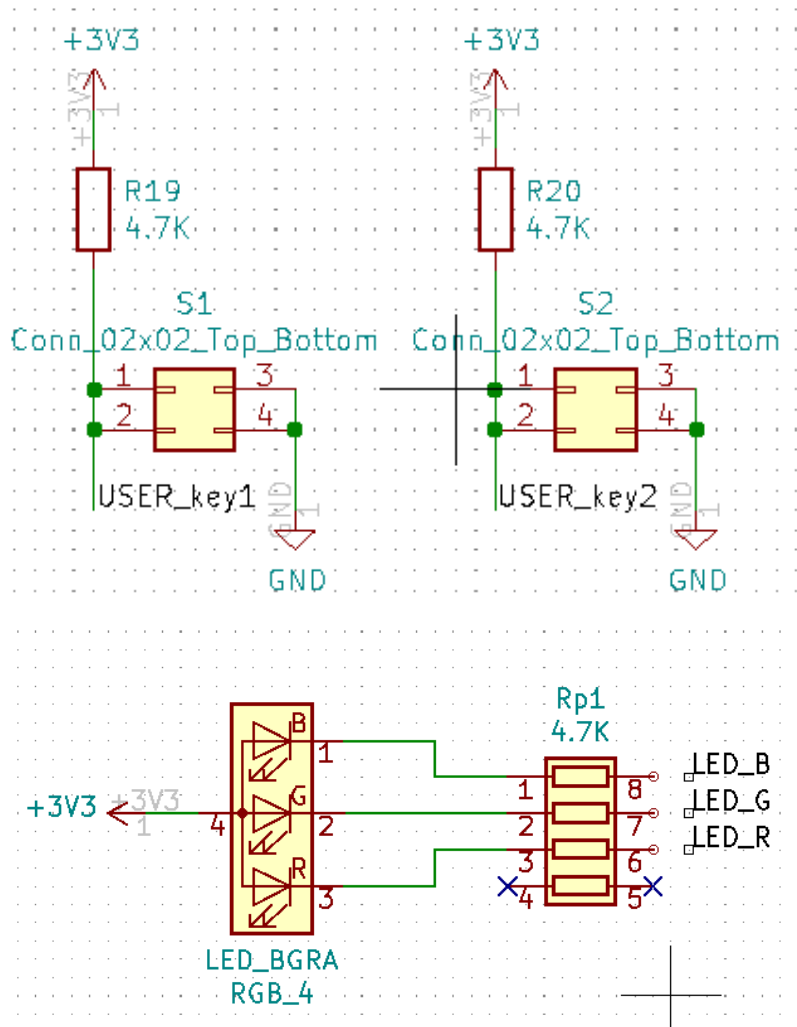


图 5.2-1 按键和 RGB 硬件原理图

5.3 micro SD 接口

microSD 卡接了六个 IO 到 FPGA 上，并且预留了上拉焊盘。可根据具体需求设计成不同读写模式。

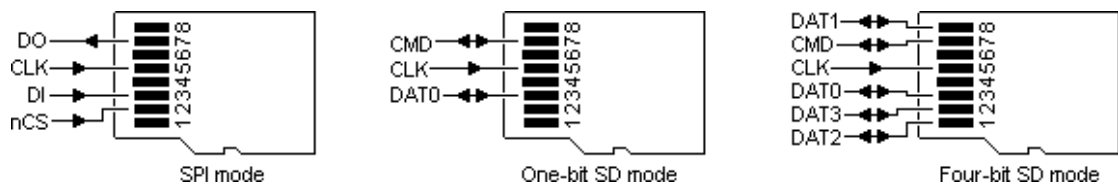


图 5.3-1 SD 卡不同控制模式

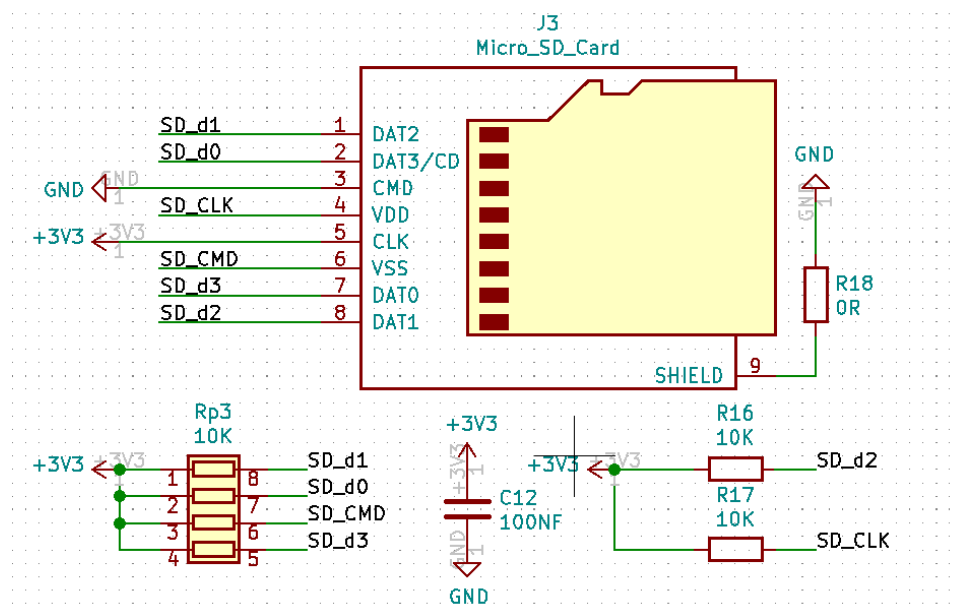


图 5.3-2 SD 硬件原理图

5.4 ESP32-S module 接口

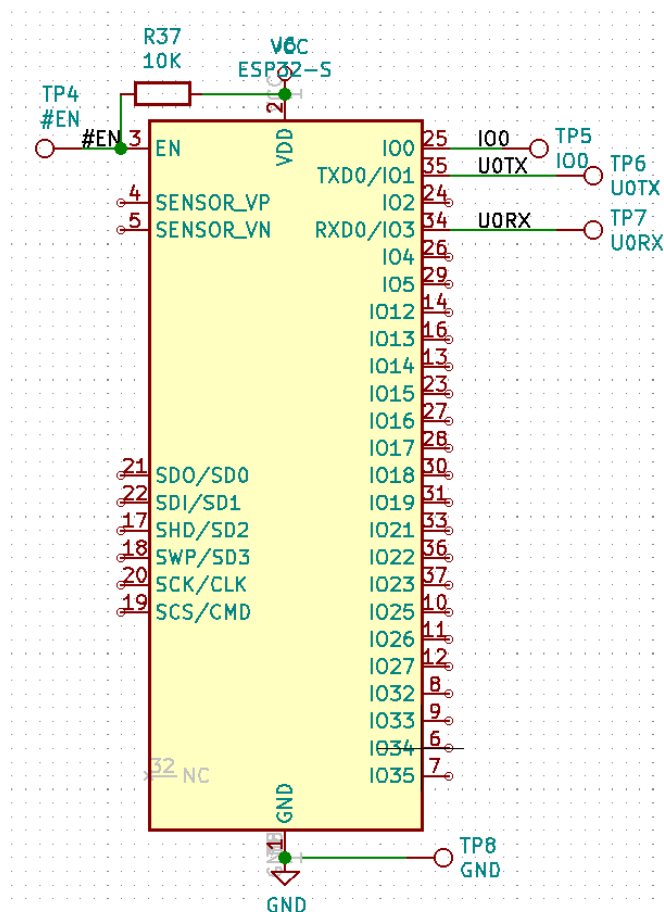


图 5.4-1 ESP32-S 接口原理图

5.5 FPGA 管脚资源分配表

	网络标号	管脚名称
24M OSC	24M	P34
PROGRAM RST BOTTON	PROGRAM	P67
USER BOTTON A	USER_key1	P16
USER BOTTON B	USER_key2	P18
RGB LED	LED_B	P17
RGB LED	LED_G	P19
RGB LED	LED_R	P23
FPGA JTAG	JTAG_TDO	P21
FPGA JTAG	JTAG_TMS	P22
FPGA JTAG	JTAG_TDI	P25
FPGA JTAG	JTAG_TCK	P26
DVP	d0	P27
DVP	d1	P28
DVP	d2	P32
DVP	d3	P30
DVP	d4	P29
DVP	d5	P31
DVP	d6	P33
DVP	d7	P37
DVP	d8	P38
DVP	d9	P40
DVP	PCLK	P35
DVP	HREF	P41
DVP	VSYNC	P45
DVP	XCLK	P39
DVP	PWDN	P42
DVP	CAMSIOC	P48
DVP	CAMSIOD	P49
DVP	CAMRST	P47
USER_FLASH	USER_cs	P54
USER_FLASH	USER_miso	P55

USER_FLASH	USER_wp	P57
USER_FLASH	USER_hold	P61
USER_FLASH	USER_cclk	P60
USER_FLASH	USER_mosi	P59
microSD	SD_d1	P62
microSD	SD_d0	P63
microSD	SD_CLK	P64
microSD	SD_CMD	P66
microSD	SD_d3	P74
microSD	SD_d2	P75
ESP32-S	U0TX	P51
ESP32-S	U0RX	P50
ESP32-S	#EN	P72
ESP32-S	IO0	P52

5.6 micro:bit 拓展接口

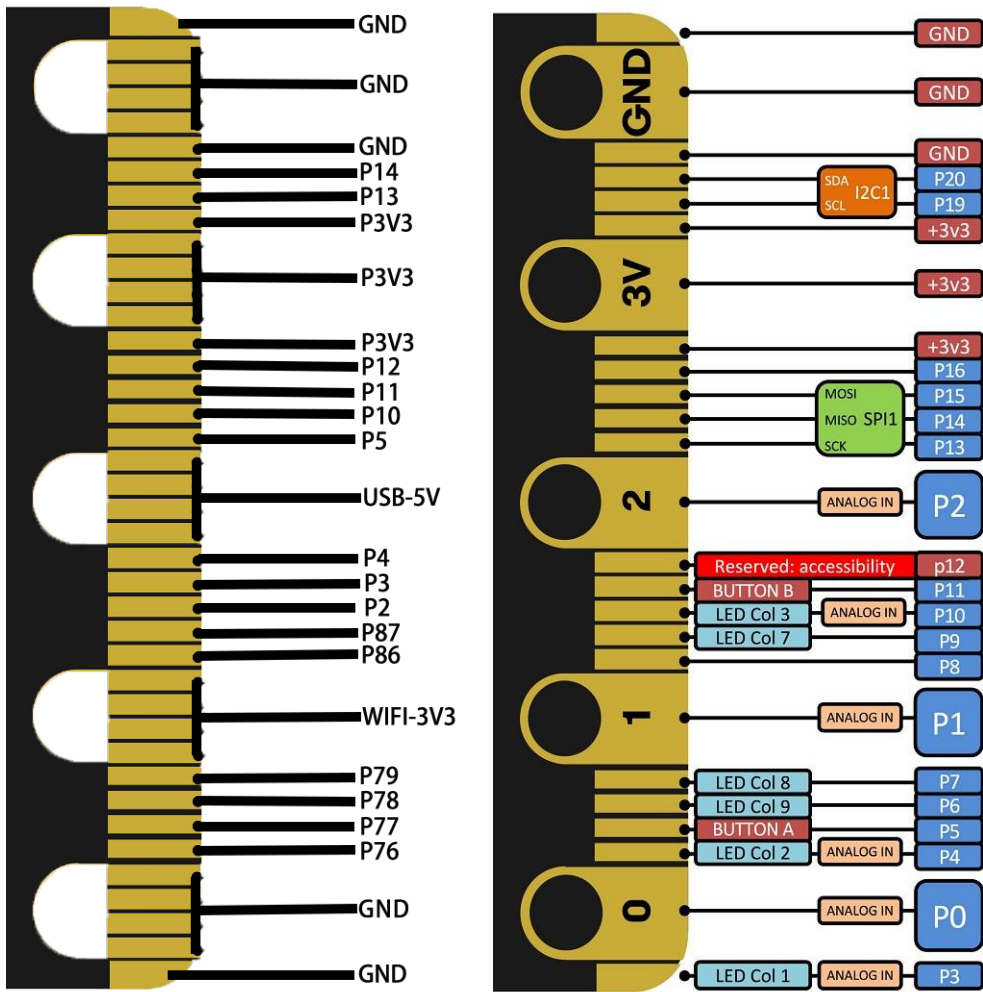


图 5-5-1 SparkRoad(左)对比 micro:bit(右)接口

版本信息

日期	版本	修订记录
5/5/2019	1.0	初版建立