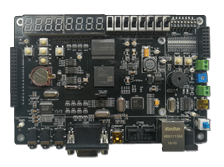
ARTIX-7 FPGA开发板硬件手册V1.0

---安徽福讯 F1 开发板教程



安徽福讯信息技术有限公司

公司地址：安徽省合肥市高新区石楠路高新集团科技实业园C14栋二层

淘宝店铺 ：https://shop493359083.taobao.com/

官方网站： www.ahfx.com

技术支持QQ群：827974317

商务联系：13965036605 17755102323

QQ:36568169 407136580

咨询电话： 0551-65306196

目 录

[第一篇 硬件篇 5](#_Toc11747)

[第一章 实验平台简介 6](#_Toc16960)

[1.1 ARTIX-7 F1 FPGA开发板资源初探 6](#_Toc19886)

[1.1.1 资源概述： 6](#_Toc22403)

[1.1.2 板载资源介绍： 7](#_Toc25752)

[1.1.3 开发板特点： 8](#_Toc28879)

[1.2 F1 FPGA开发板资源说明 9](#_Toc22582)

[1.2.1 硬件资源说明 9](#_Toc19087)

[1. 数码管 9](#_Toc21451)

[2. TFT-LCD 10](#_Toc18802)

[3. 按键 11](#_Toc24025)

[4. 温度传感器 11](#_Toc1085)

[5. EEPROM 11](#_Toc23351)

[6. 实时时钟 11](#_Toc22791)

[7. 音频接口 12](#_Toc29649)

[8. VGA 12](#_Toc20849)

[9. USB下载 12](#_Toc11909)

[10. JTAG下载 12](#_Toc30114)

[11. 千兆网口 13](#_Toc1200)

[12. 电源接口 5V2A 13](#_Toc27654)

[13. 电源开关 13](#_Toc29552)

[14. USB串口 13](#_Toc29732)

[15. DAC接口 13](#_Toc32674)

[16. ADC接口 13](#_Toc32007)

[17. 蜂鸣器 14](#_Toc2855)

[18. 摄像头接口 14](#_Toc23567)

[19. 16路LED 14](#_Toc3315)

[20. 8路贴片拨码 14](#_Toc3874)

[21. 8路拨动开关 14](#_Toc2802)

[22. SRAM 14](#_Toc3480)

[23. 复位按钮 14](#_Toc27601)

[24. 下载按键 15](#_Toc1369)

[25. FPGA 15](#_Toc10265)

[26. FLASH 15](#_Toc3582)

[27. TF 15](#_Toc23890)

[1.2.2 软件资源说明 15](#_Toc27131)

[1.2.3 F1 开发板IO引脚分配 16](#_Toc6914)

[第二章 硬件资源详解 25](#_Toc28187)

[2.1 开发板原理图详解 25](#_Toc16569)

[2.1.1 FPGA 25](#_Toc10707)

[2.1.2 时钟输入 29](#_Toc15769)

[2.1.3 JTAG 接口 29](#_Toc32569)

[2.1.4 时钟/复位/按键电路 29](#_Toc17653)

[2.1.5 LED 31](#_Toc3999)

[2.1.6 有源蜂鸣器 32](#_Toc26868)

[2.1.7 拨码开关 32](#_Toc18982)

[2.1.8 拨动开关 33](#_Toc21354)

[2.1.9 数码管 34](#_Toc5060)

[2.1.10 VGA 接口 34](#_Toc22072)

[2.1.11 LCD 模块接口 36](#_Toc31883)

[2.1.12 温度传感器接口 37](#_Toc6132)

[2.1.13 实时时钟 38](#_Toc21567)

[2.1.14 AD/DA接口 38](#_Toc1718)

[2.1.15 I2S音频编解码 40](#_Toc16505)

[2.1.16 SRAM 41](#_Toc18985)

[2.1.17 以太网接口（RJ45 ） 41](#_Toc18655)

[2.1.18 USB串口 42](#_Toc30554)

[2.1.19 摄像头模块接口 43](#_Toc12869)

[2.1.21 TF卡接口 44](#_Toc7135)

[2.1.22 SPI FLASH 44](#_Toc1969)

[2.1.23 FPGA 电源设计 45](#_Toc9030)

[2.1.24 电源适配器输入接口 46](#_Toc26774)

[2.2 开发板使用注意事项 46](#_Toc28911)

[2.3 FPGA的学习方法 47](#_Toc11708)

第一篇 硬件篇

实践出真知，要想学好FPGA，实验平台必不可少！本篇我们将详细介绍用来学习FPGA的硬件平台：福讯技术 ARTIX-7 F1 FPGA开发板，通过该篇的介绍，你将了解到我们的学习平台福讯技术 ARTIX-7 F1 FPGA开发板的功能及特点。

为了让读者更好的使用福讯技术 ARTIX-7 F1 FPGA开发板，本篇还介绍了开发板的一些使用注意事项，请读者在使用开发板的时候一定要注意。

本篇将分为如下三章：

1. 实验平台简介；

2. 实验平台硬件资源详解；

**第一章 实验平台简介**

本章内容主要向大家简要介绍我们的实验平台：福讯科技ARTIX-7 F1 FPGA开发板。通过本章的学习，你将对我们后面使用的实验平台有个大致的了解，为后面的学习做铺垫。

本章包括以下几个部分：

1.1 F1开发板资源初探

1.2 F1开发板资源说明

**1.1 ARTIX-7 F1 FPGA开发板资源初探**

**1.1.1 资源概述：**

首先，我们来看 F1 FPGA开发板的资源图，如图 1.1.1所示：

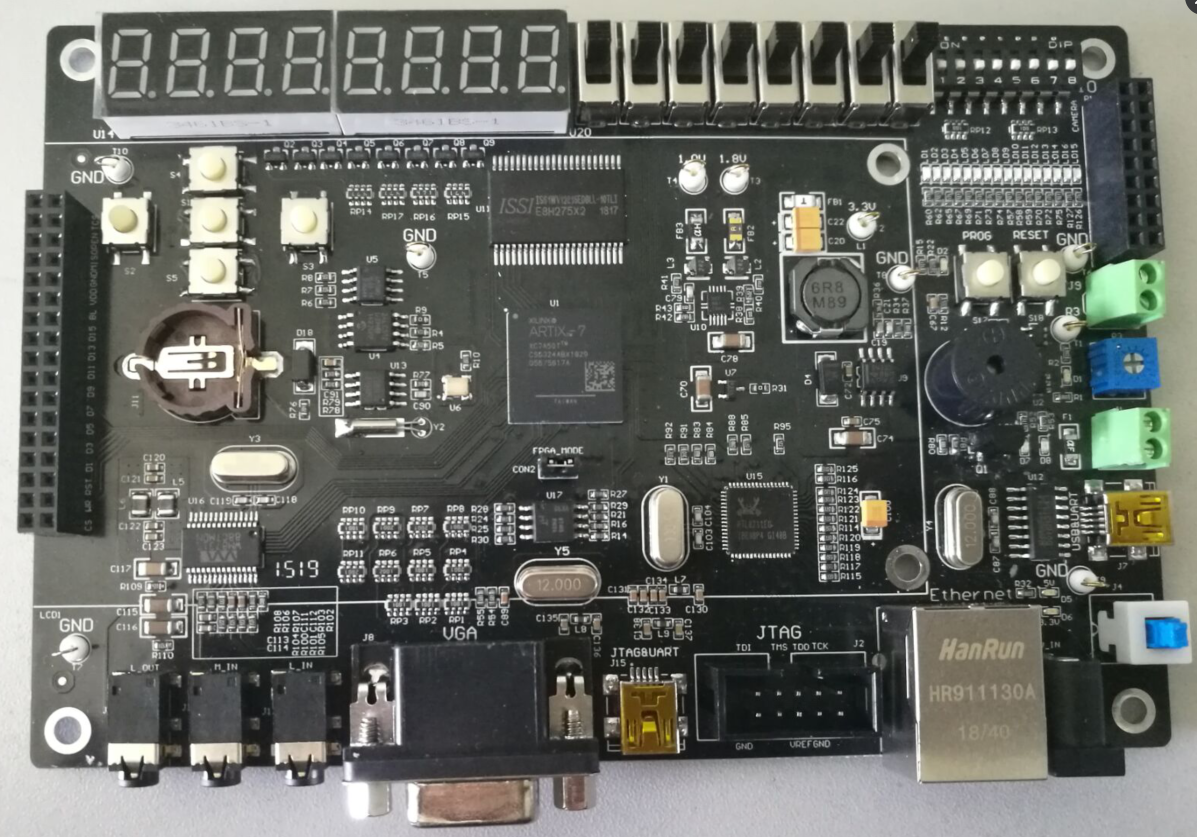




图 1.1.1 F1 FPGA开发板资源图

从图 1.1.1可以看出，F1 FPGA开发板的资源十分丰富，把FPGA XC7A50T-1CSG324I的内部资源发挥到了极致，同时扩充了丰富的接口和功能模块，整个开发板显得十分大气。

开发板的外形尺寸为100mm\*150mm，板子的设计充分考虑了人性化设计，并结合福讯信息多年的开发板设计经验，经过多次改进，最终确定了这样的设计。

**1.1.2 板载资源介绍：**

F1 FPGA开发板板载资源如下：

* 主控芯片：XC7A50T-1CSG324I，封装：BGA324
*  晶振：100Mhz
*  FLASH：MT25QL128，容量：128Mbit（16M字节）
*  SRAM：IS61WV12816BLL-12，容量：2Mbit（256K字节）
*  EEPROM：24LC512，容量：512Kbit（32K字节）
* 2个电源指示灯 5V指示和3.3V指示
*  16个状态指示灯（D1~D16）
* 8个0.36共阳数码管
* 1个标准的3.5/4.3MCU TFT-LCD 接口，支持电阻/电容触摸屏
* 5个独立按键
* 2个专用按键复位和下载
* 8个大拨码开关
* 1个8路2.54拨码开关
* 1个USB串口
*  1个有源蜂鸣器
* 1个温度传感器，采用LM75芯片
*  1路实时时钟，采用PCF8563芯片
*  1个高性能音频编解码芯片WM8731，集成24bits ADC和DAC
* 1个千兆以太网接口（RJ45）
* 1个摄像头模块接口 配合摄像头OV7725模块
*  1个TF卡接口（在板子背面）
*  1个VGA接口，数据格式为RGB565
*  1个标准的JTAG调试下载口
* 1路DAC输出 采用**DAC081S101** 0-3.3V
* 1路ADC输入 采用**ADC081S101**  0-3.3V
*  1个录音头（MIC/麦克风）,LINEIN 和HP输出
* 1路电源输入接口 DC5V@2A
* 1个电源按键开关；

**1.1.3 开发板特点：**

F1 FPGA开发板的特点包括：

1) 接口丰富。板子提供了丰富的标准外设接口，可以方便的进行各种外设的实验和开发。

2) 设计灵活。板上很多资源都可以灵活配置，以满足不同条件下的使用。其中芯片两侧引出17x2和2x9扩展口，共52个扩展IO口。

3) 资源充足。主控芯片采用自带600Kbit嵌入式RAM块的XC7A50T-

1CSG324I ，并外扩512Kbit 的EEPROM，满足大内存需求和大数据存储。板载高性能音频编解码芯片、千兆网卡、温度传感器，TFT液晶屏以及其它各种接口芯片，满足各种不同应用的需求。

4) 人性化设计。接口位置设计合理，方便顺手。资源搭配合理，物尽其用。

**1.2 F1 FPGA开发板资源说明**

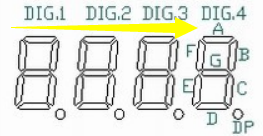
F1 FPGA开发板资源说明分为两个部分：硬件资源说明和软件资源说明。

**1.2.1 硬件资源说明**

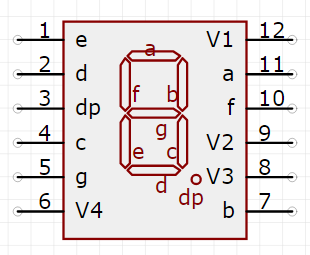
这里我们首先详细介绍F1 FPGA开发板的各个部分（图 1.1.1中的标注部分）的硬件资源，我们将按逆时针的顺序依次介绍。

1. **数码管**

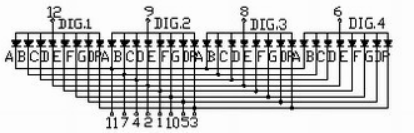
开发板上有2组4位0.36寸共阳数码管，可以同时显示8位数字输出；每组数码管从左到右计数，分别是第一位，第二位，第三位和第四位；

每个数码管的段是按照顺时针顺序的，按照顺时针12点方向开始，端的顺序是A,B,C,D,E,F,G+DP;DP对应的是点，用于指示带小数的应用；4个数码管对应的共阳端分别是12,9,8,6管脚，

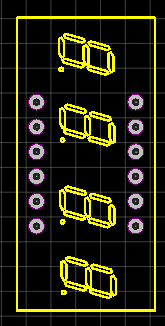


数码管管脚图



数码管内部原理图

共阳数码管的驱动需要从管脚12（9，8，6）提供驱动电源，如驱动第一数码管的A位亮，则需要在11脚（对应A）串一个限流电阻通过一个低电平来实现，驱动电流通过限流电阻来调整；



数码管PCB 封装图

1. **TFT-LCD**

开发板板载的MCU TFT-LCD模块接口（16位并口数据），支持4.3寸TFT-LCD 模块，并且支持电阻/电容触摸功能，接口在板子左侧；该模块屏幕分辨率为800\*480，16 位真彩显示，采用NT35510 驱动，该芯片直接自带GRAM，无需外加驱动器，因而任何单片机，都可以轻易驱动。模块采用电容触摸屏，支持5 点同时触摸，具有非常好的操控效果。

1. **按键**

开发板板载的5个机械式按键（S1-S5）是直接连接在FPGA的IO口上的，可以作为人机交互的输入信号。这5个按键信号默认都是高电平的，当按键被按下之后，按键信号变为低电平。

1. **温度传感器**

开发板板载的LM75BD数字温度传感器，采用I2C接口，可以提供高精度温度测量。

1. **EEPROM**

开发板板载EEPROM采用芯片24LC512,具有64KB存储空间，接口使用I2C接口；用于存储一些掉电不能丢失的重要数据，比如系统设置的一些参数等。可以方便的实现掉电数据保存

1. **实时时钟**

开发板板载PCF8563实时时钟芯片，PCF8563是PHILIPS公司推出的一款工业级多功能时钟/日历芯片，具有报警功能、定时器功能、时钟输出功能以及中断输出功能，能完成各种复杂的定时服务。

1. **音频接口**
2. 耳机输出接口 J14

这是开发板板载的音频输出接口（PHONE），该接口可以插入3.5mm的耳机。当WM8731播放音乐的时候，就可以通过在该接口插入耳机来欣赏音乐。

1. 音频输入接口 J12

这是开发板板载的音频输入接口（LINE\_IN），该接口可以用来连接电脑或者手机的耳机输出接口。

1. MIC(麦克风) J13

这是开发板的板载录音输入口（MIC，即麦克风），该麦克风直接连接到WM8731的录音输入通道上，可以实现录音的功能。

1. **VGA**

开发板板载通过电阻网络实现了16bits的RGB565的VGA接口，可以实现65536种颜色的显示；该接口可以连接在带有VGA接口的显示器上，FPGA通过VGA接口来驱动VGA显示器，使其显示出彩条、图片以及视频图像等。

1. **USB下载**

开发板板载MINI USB下载接口，可以脱离专用XILINX下载工具，方便下载；

1. **JTAG下载**

开发板板载IDC10下载接口，用于专用XILINX下载工具，方便下载；和USB下载接口不能同时使用；

1. **千兆网口**

开发板板载的网口（ETHNET），可以用来连接网线，实现网络通信功能。该接口连接到开发板上的PHY芯片（RTL8211EG），支持10Mbps/100Mbps/1000Mbps的通信速率。

1. **电源接口 5V2A**

开发板板载了芯片（TPS54331+TPS62400）组合，用于给开发板提供高效、稳定的3.3V,1.8V和1.0V电源，电源输入5V2A，接口采用DC005 标准接口。在耗电比较大的情况下，比如用到4.3寸屏/网口/音频的时候，使用外部电源供电，可以提供足够的电流给开发板使用。

1. **电源开关**

开发板板载的电源开关（J4）。该开关用于控制整个开发板的供电，如果通过开关切断电源，则整个开发板都将断电，电源指示灯（PWR）会随着此开关的状态而亮灭。

1. **USB串口**

板载了USB串口芯片CH340G ,硬件接口采用MINI USB，可以方便大家进行USB串口通信的试验。

1. **DAC接口**

板载了数字转模拟芯片DAC081S101，通过SPI接口与FPGA进行通讯，将数据转成模拟量输出。注意输出的电压范围在0-3.3V

1. **ADC接口**

板载了模拟转数字芯片ADC081S101，通过SPI接口与FPGA进行通讯，将模拟量转成数据输出。注意输入的电压范围在0-3.3V

1. **蜂鸣器**

开发板的板载蜂鸣器（BEEP），可以实现简单的报警/闹铃

1. **摄像头接口**

开发板板载的一个摄像头模块接口（P1），可以实现与摄像头OV7725模块实现图像输入，通过FPGA实现VGA显示或者网络传输； 注意使用时 拨码开关JP1-JP8都拨码到下部，就接口都拉高！

1. **16路LED**

开发板板载16路LED，用于基础试验输出信号的指示；可以组合流水灯或者状态指示。

1. **8路贴片拨码**

开发板板载贴片8位拨码用于基础试验的静态输入；可以组合状态的输入使用。

1. **8路拨动开关**

开发板板载8路JP1-JP8拨码开关,用于基础试验的静态/动态输入；可以组合状态的输入使用。

1. **SRAM**

开发板板载SRAM芯片IS61WV12861BLL，速度12ns ，容量2Mbit，用于大数据量应用；

1. **复位按钮**

这是开发板板载的复位按键（RESET），可以作为FPGA程序执行的复位信号，注意按键复位信号默认是高电平的，当复位按钮按下之后为低电平。

1. **下载按键**

用于擦除FPGA的配置FLASH

1. **FPGA**

开发板FPGA芯片采用ARTIX-7 XC7A50T-1CSG324I，该款芯片拥有52160个逻辑单元、2700Kbits的嵌入式存储资源、5个通用锁相环、10个全局时钟网络、120个DSP slices，6个用户I/O BANK和最大250个用户I/O，是一款非常具有性价比的芯片。

1. **FLASH**

开发板板载MT25QL128用于程序下载，芯片容量128Mb (16M x 8) ，接口采用SPI ，速度高达133MHz。

1. **TF**

开发板板载TF-07卡座，可用于文件系统等试验。

**1.2.2 软件资源说明**

开发板提供的标准例程多达44个，我们提供的这些例程，基本都是原创，拥有非常详细的注释、代码风格统一、循序渐进，非常适合初学者入门。而其他开发板的例程，注释比较少且工程文件管理不统一，对初学者来说不那么容易入门。

F1 FPGA开发板的例程列表如表1.2.2.1所示：

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 工程 | 名称 | 功能描述 |
| 1 | at7\_prj1 | 门电路测试试验 | 试验加法、乘法逻辑门电路，并用在线调试工具查看 |
| 2 | at7\_prj2 | PLL+RAM的IP核配置测试试验 | 例化PLL和RAM的IP核，并用在线调试工具查看RAM的写入读出 |
| 3 | at7\_prj3 | 复位键和蜂鸣器测试 | 对时钟做分频，产生1Hz的分频信号控制蜂鸣器发声；按复位键完成复位，蜂鸣器停止 |
| 4 | at7\_prj4 | 流水灯测试试验 | 显示16个LED流水灯 |
| 5 | at7\_prj5 | 拨码开关测试试验 | LED显示按键测试，按一下则对应的led灯亮 |
| 6 | at7\_prj6 | 按键和数码管测试 | 按下按键，则8位数码管显示对应的按键值 |
| 7 | at7\_prj7 | SRAM测试试验 | 测试SRAM接口，并用在线调试工具查看有规律变化的SRAM数据读写 |
| 8 | at7\_prj8 | EEPROM测试试验 | EEPROM掉电缓存数据，并观察数据读写 |
| 9 | at7\_prj9 | ADC测试试验 | 定时采集输入模拟电压，显示到数码管上 |
| 10 | at7\_prj10 | DAC测试试验 | 产生一个0-255循环递增的数据，通过SPI接口不断的写入到DAC中，输出的模拟电压可以控制LED的亮暗变化 |
| 11 | at7\_prj11 | 串口测试试验 | 实现PC端通过UART发送数据到FPGA，FPGA将所接收到的数据同样通过UART发回PC端 |
| 12 | at7\_prj12 | RTC测试试验 | 读取RTC芯片中的时、分、秒数据，并通过led显示秒数据 |
| 13 | at7\_prj13 | TF卡测试试验 | TF卡写入，读出，并通过led显示读出结果 |
| 14 | at7\_prj14 | 温度模块采集试验 | 实现对温度传感器的采集，显示到数码管上 |
| 15 | at7\_prj15 | 以太网测试试验 | 实现PC端通过千兆网口发送数据到FPGA，FPGA数据同样是通过千兆网口发回PC端 |
| 16 | at7\_prj16 | VGA测试试验 | VGA使屏幕产生彩色条纹 |
| 17 | at7\_prj17 | 液晶屏显示测试试验 | 驱动彩色液晶屏，显示8色ColorBar |
| 18 | at7\_prj18 | 音频输出测试试验 | TF卡存储一首歌曲，并通过耳机收听 |

表1.2.2.1 F1 FPGA开发板例程

FPGA开发板的例程是非常丰富的，并且扩展了很多有价值的例程。各个例程的安排是循序渐进的，首先从最基础的流水灯开始，然后一步步深入，从简单到复杂，有利于大家的学习和掌握，所以，F1 FPGA开发板是非常适合初学者的。当然，对于想深入学习FPGA开发的朋友，F1 FPGA开发板也绝对是一个不错的选择。

**1.2.3 F1 开发板IO引脚分配**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 信号名 | 方向 | 管脚 | | BANK | 端口说明 |
| 系统时钟（100Mhz） | | | | | |
| **CLK\_100M** | input | **E3** | | **BANK35** | 系统时钟，频率：100Mhz |
| 系统复位（RESET） | | | | | |
| **RST** | input | **T16** | | **BANK35** | 系统复位，低电平有效 |
| 5个按键 | | | | | |
| S1 | input | **E15** | | **BANK15** | 按下低电平有效 |
| S2 | input | **D14** | | **BANK15** | 按下低电平有效 |
| S3 | input | **C14** | | **BANK15** | 按下低电平有效 |
| S4 | input | **B16** | | **BANK15** | 按下低电平有效 |
| S5 | input | **B13** | | **BANK15** | 按下低电平有效 |
| 16个LED灯 | | | | | |
| D1 | output | **B11** | | **BANK15** | 高电平点灯 LED1 |
| D2 | output | **B12** | | **BANK15** | 高电平点灯 LED2 |
| D3 | output | **C15** | | **BANK15** | 高电平点灯 LED3 |
| D4 | output | **C16** | | **BANK15** | 高电平点灯 LED4 |
| D5 | output | **C17** | | **BANK15** | 高电平点灯 LED5 |
| D6 | output | **D15** | | **BANK15** | 高电平点灯 LED6 |
| D7 | output | **D17** | | **BANK15** | 高电平点灯 LED7 |
| D8 | output | **F13** | | **BANK15** | 高电平点灯 LED8 |
| D9 | output | **G14** | | **BANK15** | 高电平点灯 LED9 |
| D10 | output | **J13** | | **BANK15** | 高电平点灯 LED10 |
| D11 | output | **H16** | | **BANK15** | 高电平点灯 LED11 |
| D12 | output | **G16** | | **BANK15** | 高电平点灯 LED12 |
| D13 | output | **F16** | | **BANK15** | 高电平点灯 LED13 |
| D14 | output | **G17** | | **BANK15** | 高电平点灯 LED14 |
| D15 | output | **H15** | | **BANK15** | 高电平点灯 LED15 |
| D16 | output | **H17** | | **BANK15** | 高电平点灯 LED16 |
| 蜂鸣器 | | | | | |
| BEEP | output | **M13** | | **BANK14** | 高电平有效 |
| 8位数码管 | | | | | |
| 信号名 | 方向 | 管脚 | BANK | | 端口说明 |
| **DG\_A0** | output | **H1** | **BANK35** | | 第1个数码管位选信号 |
| **DG\_A1** | output | **K3** | **BANK34** | | 第2个数码管位选信号 |
| **DG\_A2** | output | **K2** | **BANK35** | | 第3个数码管位选信号 |
| **DG\_A3** | output | **L3** | **BANK34** | | 第4个数码管位选信号 |
| **DG\_A4** | output | **H5** | **BANK35** | | 第5个数码管位选信号 |
| **DG\_A5** | output | **J5** | **BANK35** | | 第6个数码管位选信号 |
| **DG\_A6** | output | **F4** | **BANK35** | | 第7个数码管位选信号 |
| **DG\_A7** | output | **A9** | **BANK16** | | 第8个数码管位选信号 |
| **SDG\_A** | output | **J2** | **BANK35** | | 数码管段选A |
| **SDG\_B** | output | **G3** | **BANK35** | | 数码管段选B |
| **SDG\_C** | output | **H4** | **BANK35** | | 数码管段选C |
| **SDG\_D** | output | **G4** | **BANK35** | | 数码管段选D |
| **SDG\_E** | output | **M4** | **BANK34** | | 数码管段选E |
| **SDG\_F** | output | **L4** | **BANK34** | | 数码管段选F |
| **SDG\_G** | output | **L5** | **BANK34** | | 数码管段选G |
| **SDG\_DP** | output | **K5** | **BANK34** | | 数码管段选DP |
| USB串口 | | | | | |
| **uart\_rxd** | input | **E16** | **BANK15** | | USB串口接收 |
| **uart\_txd** | output | **K13** | **BANK15** | | USB串口发送 |
| VGA接口 | | | | | |
| **vga\_hs** | output | **V9** | **BANK34** | | VGA行同步 |
| **vga\_vs** | output | **V7** | **BANK34** | | VGA场同步 |
| **vga\_rgb[0]** | output | **U6** | **BANK34** | | **VGA蓝色** |
| **vga\_rgb[1]** | output | **U4** | **BANK34** | | **VGA蓝色** |
| **vga\_rgb[2]** | output | **U3** | **BANK34** | | **VGA蓝色** |
| **vga\_rgb[3]** | output | **U7** | **BANK34** | | **VGA蓝色** |
| 信号名 | 方向 | 管脚 | BANK | | 端口说明 |
| **vga\_rgb[4]** | output | **T5** | **BANK34** | | **VGA蓝色** |
| **vga\_rgb[5]** | output | **U2** | **BANK34** | | **VGA绿色** |
| **vga\_rgb[6]** | output | **V10** | **BNAK14** | | **VGA绿色** |
| **vga\_rgb[7]** | output | **T9** | **BNAK14** | | **VGA绿色** |
| **vga\_rgb[8]** | output | **U8** | **BANK34** | | **VGA绿色** |
| **vga\_rgb[9]** | output | **T10** | **BNAK14** | | **VGA绿色** |
| **vga\_rgb[10]** | output | **U9** | **BANK34** | | **VGA绿色** |
| **vga\_rgb[11]** | output | **T8** | **BANK34** | | **VGA红色** |
| **vga\_rgb[12]** | output | **T6** | **BANK34** | | **VGA红色** |
| **vga\_rgb[13]** | output | **R10** | **BANK14** | | **VGA红色** |
| **vga\_rgb[14]** | output | **T4** | **BANK34** | | **VGA红色** |
| **vga\_rgb[15]** | output | **T3** | **BANK34** | | **VGA红色** |
| TFT-LCD接口 | | | | | |
| **LCD\_DB0** | output | **R6** | **BANK34** | | LCD数据D0 |
| **LCD\_DB1** | output | **R7** | **BANK34** | | LCD数据D1 |
| **LCD\_DB2** | output | **P5** | **BANK34** | | LCD数据D2 |
| **LCD\_DB3** | output | **P3** | **BANK34** | | LCD数据D3 |
| **LCD\_DB4** | output | **N5** | **BANK34** | | LCD数据D4 |
| **LCD\_DB5** | output | **M6** | **BANK34** | | LCD数据D5 |
| **LCD\_DB6** | output | **K6** | **BANK34** | | LCD数据D6 |
| **LCD\_DB7** | output | **J4** | **BANK35** | | LCD数据D7 |
| **LCD\_DB8** | output | **R8** | **BANK34** | | LCD数据D8 |
| **LCD\_DB9** | output | **G1** | **BANK35** | | LCD数据D9 |
| **LCD\_DB10** | output | **R2** | **BANK34** | | LCD数据D10 |
| **LCD\_DB11** | output | **R5** | **BANK34** | | LCD数据D11 |
| **LCD\_DB12** | output | **N6** | **BANK34** | | LCD数据D12 |
| **LCD\_DB13** | output | **M2** | **BANK34** | | LCD数据D13 |
| **LCD\_DB14** | output | **L6** | **BANK34** | | LCD数据D14 |
| **LCD\_DB15** | output | **J3** | **BANK35** | | LCD数据D15 |
| **LCD\_RD** | output | **L1** | **BANK34** | | LCD读使能 |
| **LCD\_WR** | output | **M1** | **BANK34** | | LCD写使能 |
| **LCD\_RS** | output | **N1** | **BANK34** | | 命令/数据控制（0：命令 1：  数据） |
| **LCD\_RST** | output | **K1** | **BANK35** | | LCD复位信号 |
| **LCD\_CS** | output | **R1** | **BANK34** | | 片选信号 |
| **T\_MO** | input | **R3** | **BANK34** | | 忙信号 |
| **T\_CS** | output | **P4** | **BANK34** | | 复位信号 |
| **T\_CLK** | output | **N4** | **BANK34** | | 触摸屏 IIC\_SCL 信号 |
| **T\_PEN** | output | **F6** | **BANK35** | | 触摸屏中断信号 |
| **T\_MOSI** | inout | **G2** | **BANK35** | | 触摸屏 IIC\_SDA 信号 |
| **RT\_MOSI** | output | **G6** | **BANK35** | | NC |
| **BL\_CTR** | output | **H6** | **BANK35** | | 背光控制引脚（高电平点亮背光，低电平关闭） |
| RTC实时时钟 | | | | | |
| RTC\_SCL | output | **T11** | **BANK14** | | IIC时钟信号线 |
| RTC\_SDA | inout | **U11** | **BANK14** | | IIC双向数据线 |
| AD | | | | | |
| AD\_CS | output | **K16** | **BANK15** | | AD SPI片选 |
| AD\_CLK | output | **J15** | **BANK15** | | AD SPI时钟 |
| AD\_SDATA | inout | **K15** | **BANK15** | | AD SPI串行输出数据 |
| DA | | | | | |
| DA\_SCK | output | **J14** | **BANK15** | | DA SPI时钟 |
| DA\_SYN | output | **J18** | **BANK15** | | DA SPI同步 |
| DA\_DIN | inout | **J17** | **BANK15** | | DA SPI串行输出数据 |
| EEPROM | | | | | |
| EEF\_SCL | output | **C9** | **BANK16** | | IIC时钟信号线 |
| EEF\_SDA | inout | **D9** | **BANK16** | | IIC双向数据线 |
| 温度传感器 | | | | | |
| TEM\_SCL | output | **N2** | **BANK34** | | IIC时钟信号线 |
| TEM\_SDA | inout | **M3** | **BANK34** | | IIC双向数据线 |
| TEM\_OS | input | **P2** | **BANK34** | | 超温输出 |
| 音频 | | | | | |
| VM\_SCLK | output | **V6** | **BANK34** | | WM8731的IIC配置时钟信号线 |
| VM\_SDIN | inout | **V5** | **BANK34** | | WM8731的IIC配置双向数据线 |
| VM\_BCLK | output | **T1** | **BANK34** | | WM8731的位时钟 |
| VM\_DACDAT | inout | **U1** | **BANK34** | | WM8978的DAC数据线 |
| VM\_DACLRC | output | **V1** | **BANK34** | | WM8978的DAC时钟 |
| VM\_ADCDAT | inout | **V2** | **BANK34** | | WM8978的ADC数据线 |
| VM\_ADCLRC | output | **V4** | **BANK34** | | WM8978的ADC时钟 |
| 网络 | | | | | |
| **E1\_MDC** | output | **R11** | **BANK14** | | MD时钟 |
| **E1\_MDIO** | inout | **L18** | **BANK14** | | MD数据线 |
| **E1\_GTXC** | output | **M18** | **BANK14** | | GMII时钟 |
| **E1\_TXEN** | output | **R12** | **BANK14** | | 发送使能 |
| **E1\_TXER** | output | **R13** | **BANK14** | | 发送错误 |
| **E1\_RXER** | inout | **P17** | **BANK14** | | 接收错误 |
| **E1\_CRS** | input | **U16** | **BANK14** | | 载波检测 |
| **E1\_RXCOL** | input | **V17** | **BANK14** | | 碰撞输出 |
| **E1\_PME** | input | **U12** | **BANK14** | | 电源管理事件输出 |
| **E1\_TXC** | output | **M17** | **BANK14** | | 发送时钟 |
| **E1\_RXC** | input | **N15** | **BANK14** | | 接收时钟 |
| **E1\_RDV** | input | **N16** | **BANK14** | | 接收数据有效标志输出 |
| **E1\_RESET** | output | **V12** | **BANK14** | | 硬件复位 低有效 |
| **E1\_ENSWEREG** | output | **U14** | **BANK14** | | 电压使能输出 |
| **E1\_TXDO** | output | **T13** | **BANK14** | | 发送数据总线 |
| **E1\_TXD1** | output | **R18** | **BANK14** | | 发送数据总线 |
| **E1\_TXD2** | output | **T18** | **BANK14** | | 发送数据总线 |
| **E1\_TXD3** | output | **N14** | **BANK14** | | 发送数据总线 |
| **E1\_TXD4** | output | **P14** | **BANK14** | | 发送数据总线 |
| **E1\_TXD5** | output | **N17** | **BANK14** | | 发送数据总线 |
| **E1\_TXD6** | output | **P18** | **BANK14** | | 发送数据总线 |
| **E1\_TXD7** | output | **M16** | **BANK14** | | 发送数据总线 |
| **E1\_RXD0** | input | **R17** | **BANK14** | | 接收数据总线 |
| **E1\_RXD1** | input | **P15** | **BANK14** | | 接收数据总线 |
| **E1\_RXD2** | input | **R15** | **BANK14** | | 接收数据总线 |
| **E1\_RXD3** | input | **T14** | **BANK14** | | 接收数据总线 |
| **E1\_RXD4** | input | **T15** | **BANK14** | | 接收数据总线 |
| **E1\_RXD5** | input | **V16** | **BANK14** | | 接收数据总线 |
| **E1\_RXD6** | input | **U17** | **BANK14** | | 接收数据总线 |
| **E1\_RXD7** | input | **U18** | **BANK14** | | 接收数据总线 |
| 摄像头 | | | | | |
| **FIFO\_D0** | inout | **D18** | **BANK15** | | FIFO 输出数据口 |
| **FIFO\_D1** | inout | **H14** | **BANK15** | | FIFO 输出数据口 |
| **FIFO\_D2** | inout | **A16** | **BANK15** | | FIFO 输出数据口 |
| **FIFO\_D3** | inout | **A15** | **BANK15** | | FIFO 输出数据口 |
| **FIFO\_D4** | inout | **A14** | **BANK15** | | FIFO 输出数据口 |
| **FIFO\_D5** | inout | **A13** | **BANK15** | | FIFO 输出数据口 |
| **FIFO\_D6** | inout | **A11** | **BANK15** | | FIFO 输出数据口 |
| **FIFO\_D7** | inout | **G13** | **BANK15** | | FIFO 输出数据口 |
| **FIFO\_PRST** | output | **B18** | **BANK15** | | FIFO 读指针复位 |
| **FIFO\_WRST** | output | **F18** | **BANK15** | | FIFO 写指针复位 |
| **FIFO\_OE** | output | **A18** | **BANK15** | | FIFO输出使能 |
| **FIFO\_WEN** | output | **F15** | **BANK15** | | FIFO写使能 |
| **FIFO\_RCLK** | output | **F14** | **BANK15** | | FIFO读时钟 |
| **OV\_VSYNC** | output | **E17** | **BANK15** | | 帧同步信号 |
| **OV\_SCL** | output | **G18** | **BANK15** | | SCCB通讯时钟信号 |
| **OV\_SDA** | inout | **E18** | **BANK15** | | SCCB通讯数据信号 |
| 拨码开关 | | | | | |
| **SWO** | input | **G18** | **BANK15** | | 拨码开关0 |
| **SW1** | input | **F18** | **BANK15** | | 拨码开关1 |
| **SW2** | input | **E18** | **BANK15** | | 拨码开关2 |
| **SW3** | input | **B18** | **BANK15** | | 拨码开关3 |
| **SW4** | input | **D18** | **BANK15** | | 拨码开关4 |
| **SW5** | input | **A18** | **BANK15** | | 拨码开关5 |
| **SW6** | input | **A16** | **BANK15** | | 拨码开关6 |
| **SW7** | input | **H14** | **BANK15** | | 拨码开关7 |
| 拨动开关 | | | | | |
| **SW8** | input | **A14** | **BANK15** | | 拨码开关8 |
| **SW9** | input | **A15** | **BANK15** | | 拨码开关9 |
| **SW10** | input | **A11** | **BANK15** | | 拨码开关10 |
| **SW11** | input | **A13** | **BANK15** | | 拨码开关11 |
| **SW12** | input | **F14** | **BANK15** | | 拨码开关12 |
| **SW13** | input | **G13** | **BANK15** | | 拨码开关13 |
| **SW14** | input | **F15** | **BANK15** | | 拨码开关14 |
| **SW15** | input | **E17** | **BANK15** | | 拨码开关15 |
| SRAM | | | | | |
| **M\_A0** | output | **A6** | **BANK35** | | SRAM地址信号 |
| **M\_A1** | output | **A5** | **BANK35** | | SRAM地址信号 |
| **M\_A2** | output | **A4** | **BANK35** | | SRAM地址信号 |
| **M\_A3** | output | **A3** | **BANK35** | | SRAM地址信号 |
| **M\_A4** | output | **A1** | **BANK35** | | SRAM地址信号 |
| **M\_A5** | output | **F1** | **BANK35** | | SRAM地址信号 |
| **M\_A6** | output | **E1** | **BANK35** | | SRAM地址信号 |
| **M\_A7** | output | **F3** | **BANK35** | | SRAM地址信号 |
| **M\_A8** | output | **H2** | **BANK35** | | SRAM地址信号 |
| **M\_A9** | output | **D10** | **BANK16** | | SRAM地址信号 |
| **M\_A10** | output | **B8** | **BANK16** | | SRAM地址信号 |
| **M\_A11** | output | **A8** | **BANK16** | | SRAM地址信号 |
| **M\_A12** | output | **D8** | **BANK35** | | SRAM地址信号 |
| **M\_A13** | output | **D7** | **BANK35** | | SRAM地址信号 |
| **M\_A14** | output | **C6** | **BANK35** | | SRAM地址信号 |
| **M\_A15** | output | **C5** | **BANK35** | | SRAM地址信号 |
| **M\_A16** | output | **C4** | **BANK35** | | SRAM地址信号 |
| **M\_D0** | inout | **B1** | **BANK35** | | SRAM数据信号 |
| **M\_D1** | inout | **B2** | **BANK35** | | SRAM数据信号 |
| **M\_D2** | inout | **B3** | **BANK35** | | SRAM数据信号 |
| **M\_D3** | inout | **B4** | **BANK35** | | SRAM数据信号 |
| **M\_D4** | inout | **B6** | **BANK35** | | SRAM数据信号 |
| **M\_D5** | inout | **B7** | **BANK35** | | SRAM数据信号 |
| **M\_D6** | inout | **C7** | **BANK35** | | SRAM数据信号 |
| **M\_D7** | inout | **C1** | **BANK35** | | SRAM数据信号 |
| **M\_D8** | inout | **D4** | **BANK35** | | SRAM数据信号 |
| **M\_D9** | inout | **D3** | **BANK35** | | SRAM数据信号 |
| **M\_D10** | inout | **D2** | **BANK35** | | SRAM数据信号 |
| **M\_D11** | inout | **E5** | **BANK35** | | SRAM数据信号 |
| **M\_D12** | inout | **E2** | **BANK35** | | SRAM数据信号 |
| **M\_D13** | inout | **D5** | **BANK35** | | SRAM数据信号 |
| **M\_D14** | inout | **E6** | **BANK35** | | SRAM数据信号 |
| **M\_D15** | inout | **E7** | **BANK35** | | SRAM数据信号 |
| **M\_WR** | output | **C2** | **BANK35** | | SRAM写使能信号 |
| **M\_OE** | output | **F5** | **BANK35** | | SRAM读使能信号 |
| TF卡 | | | | | |
| **SD\_DATA0** | output | **B9** | **BANK16** | | TF卡数据信号 |
| **SD\_DATA1** | output | **D12** | **BANK15** | | TF卡数据信号 |
| **SD\_DATA2** | output | **C12** | **BANK15** | | TF卡数据信号 |
| **SD\_DATA3** | output | **C11** | **BANK16** | | TF卡数据信号，可做片选信号 |
| **SD\_CMD** | input | **C10** | **BANK16** | | TF卡数据输入 |
| **SD\_CLK** | output | **A10** | **BANK16** | | TF卡时钟 |

**第二章 硬件资源详解**

本章，我们将向大家详细介绍F1 FPGA开发板各部分的硬件原理图，让大家对该开发板的各部分硬件原理有个深入理解，并向大家介绍开发板的使用注意事项，为后面的学习做好准备。

本章包括以下几个部分：

2.1 开发板原理图详解

2.2 开发板使用注意事项

2.3 FPGA的学习方法

**2.1 开发板原理图详解**

**2.1.1 FPGA**

F1 FPGA开发板选择的是ARITX-7系列的XC7A50T-1CSG324I，该芯片是极具功耗和性价比优势，它拥有52160个逻辑单元、8150个slices，65200个CLB单元，600Kb的嵌入式存储资源、120个嵌入式DSP SLICE、1个通用锁相环、5个全局时钟网络、5个用户IO BANK和最大250个用户I/O，1个2代PCIE，1组XADC,1个可配置AES加密块，8个GTP传输器；了解器件的整体硬件资源，有助于我们在设计时根据器件提供的资源，对设计进行合理的优化，以达到最佳的性价比。具体这些资源在设计中怎么使用，我们将在后续章节为大家介绍。

另外需要了解的一点是XC7A50T-1CSG324I芯片的IO口分成了5组，每一组称为一个IO Bank，同一个Bank中的所有IO供电相同，而各个Bank的IO供电可以不同。FPGA的5个I/O BANK的引脚连接如下：

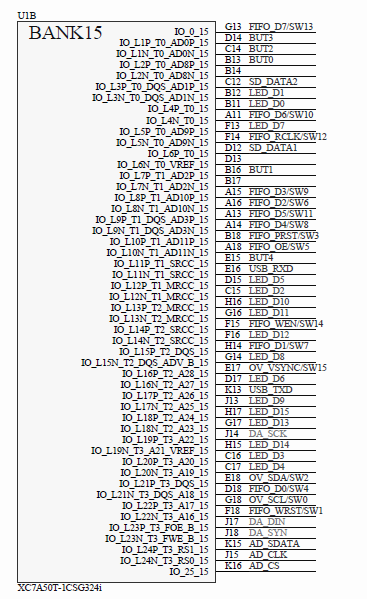
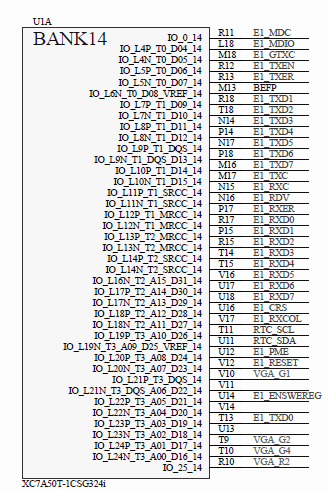


图 2.1.1.1 BANK14 15 引脚连接图

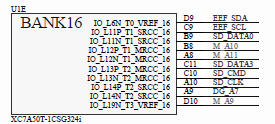


图 2.1.1.2 BANK16 引脚连接图

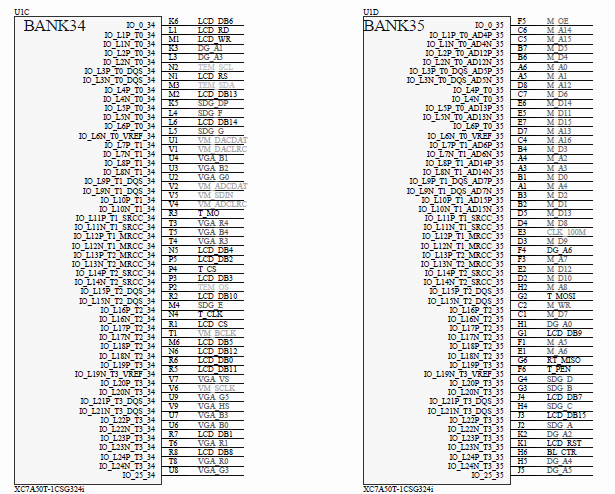


图 2.1.1.3 BANK34 ，35 引脚连接图

通过查看各BANK引脚与对应模块的端口连接即可确定管脚分配。

**2.1.2 时钟输入**

F1 FPGA开发板提供给FPGA的时钟晶振电路如下图所示：

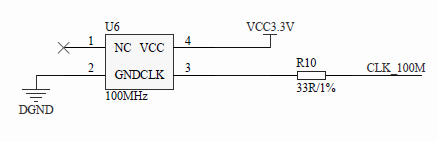


图 2.1.2.1 时钟晶振电路

由上图可知，晶振提供给FPGA的时钟为100MHz，当需要其它频率的时钟时可通过FPGA内部的锁相环 PLL产生。

**2.1.3 JTAG 接口**

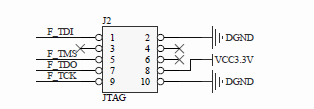
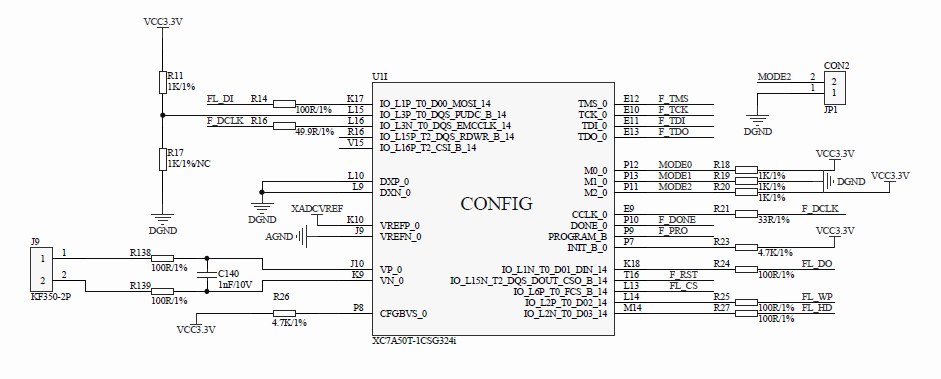


图 2.1.3.1 JTAG 接口电路

我们采用的是标准的JTAG接法（10脚），使用FPGA TCK、FPGA TDO、FPGA TDI、FPGA TMS四个引脚来对FPGA进行下载测试。

**2.1.4 时钟/复位/按键电路**

F1 FPGA开发板的时钟/复位/按键电路与FPGA的引脚连接如下图所示：



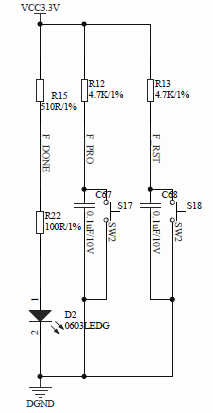


图 2.1.4.1 时钟/复位/按键接口

S18为复位按键，按下时低电平复位;其中5个输入按键的电路原理图如下图所示：

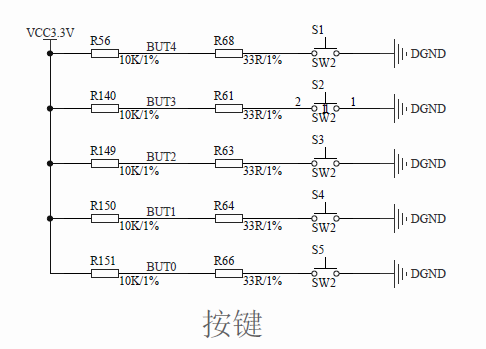


图 2.1.4.2 复位、按键电路

，S1-S5为5个独立按键外接上拉电阻，未按下时按键端口输入高电平，按下时输入低电平。按键作为最简单的输入设备，适合在需要给系统输入控制信号的场合使用。

**2.1.5 LED**

F1 FPGA开发板板载了16个LED，电路原理图如下图所示：

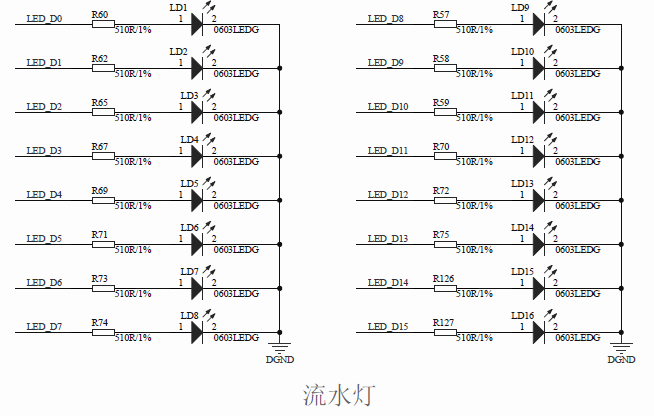


图 2.1.5.1 LED

16个LED均由FPGA驱动，当FPGA输出高电平时电亮对应的LED灯，点亮LED灯。

**2.1.6 有源蜂鸣器**

F1 FPGA开发板板载了一个有源蜂鸣器，电路原理图如下图所示：

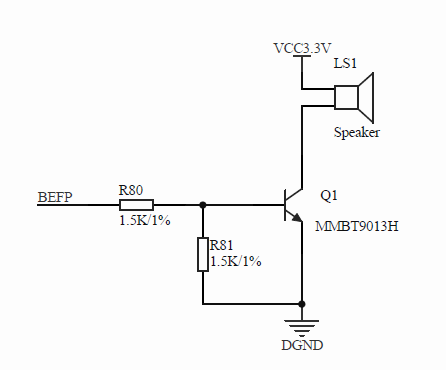


图 2.1.6.1 蜂鸣器

蜂鸣器采用三极管进行驱动，三极管型号MMBT9013,采用共发射级电路，驱动信号从FPGA的BEEP进来，通过R80驱动三极管，蜂鸣器在集电极；BEEP信号高电平控制三极管导通，从而电流经蜂鸣器和三极管，导通后蜂鸣器会发声。

**2.1.7 拨码开关**

F1 FPGA开发板板载了1组拨码开关，包含8个开关，电路原理图如下图所示：

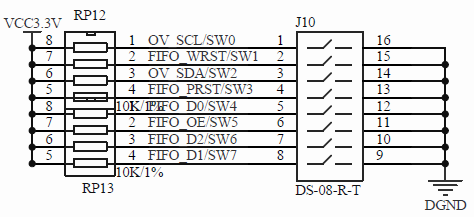
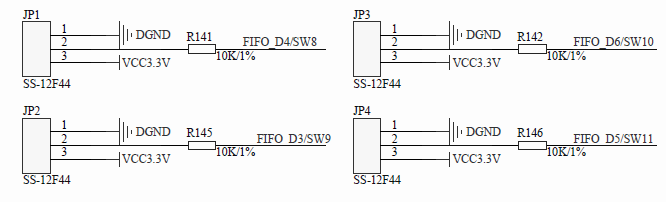


图2.1.7.1 拨码开关

8位拨码开关导通时，输出低电平；板子设计时拨码开关和摄像头模块共用了8个管脚，注意使用摄像头模块时，拨码必须全部拨到OFF状态；

**2.1.8 拨动开关**

F1 FPGA开发板板载了1组拨动开关，包含8个开关，电路原理图如下图所示：



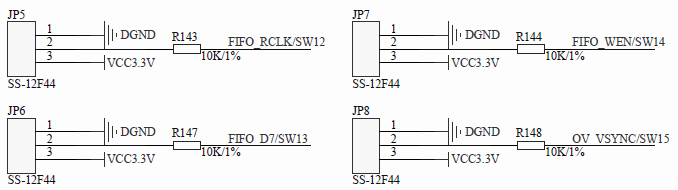


图2.1.8.1 拨动开关

拨动开关JP1-JP8，8位拨动开关导通时，输出低电平；板子设计时拨动开关和摄像头模块共用了8个管脚，注意使用摄像头模块时，拨动开关JP1-JP8必须全部拨到下部，即开关的输出都是上拉状态；

**2.1.9 数码管**

F1 FPGA开发板板载了8个共阳数码管，电路原理图如下图所示：

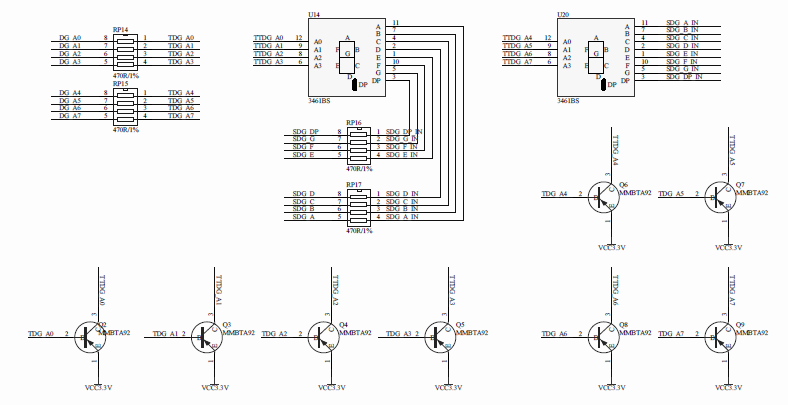


图2.1.9.1 数码管

需要注意的是，这里是用PNP型三极管作为开关控制位选信号，当基极为低电平时，三极管导通，输出高电平；

**2.1.10 VGA 接口**

F1 FPGA开发板板载了一个RGB565数据格式的VGA接口，电路原理图如下图所示：

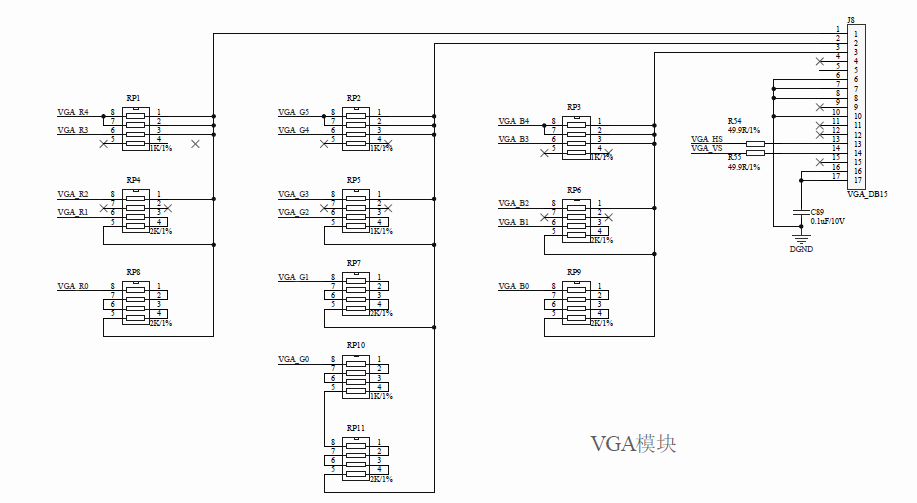


图 2.1.11.1 权电阻网络全1时等效电路

VGA的全称是Video Graphics Array，即视频图形阵列，是一个使用模拟信号进行视频传输的标准。VGA接口协议规定，红绿蓝三基色模拟电压的范围为0~0.714V，其中0V代表无色，0.714V代表满色。RGB565格式，即红色占5位数据，绿色占6位数据，蓝色占5位数据，一共可表示65536种颜色。

F1 FPGA开发板用权电阻网络来实现RGB565信号到VGA接口三基色信号转换的电路，即用5位权电阻网络来实现红色信号R，6位权电阻网络实现绿色信号G，5位权电阻网络实现蓝色信号B。以红色信号R为例来说明权电阻网络参数的选取。由于R对应的数字信号为5位，故当该5位全为1的时候对应的模拟电压信号值应为0.714V，当5位全为 0 的时候对应的模拟电压信号值应为0V。等效电路如下图所示（终端电阻75Ω在显示器连接头内部）

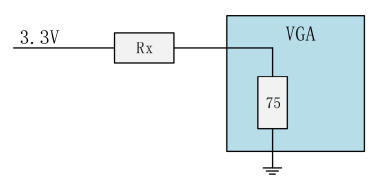


图 2.1.11.2 权电阻网络全1时等效电路

其中Rx为5位权电阻网络的等效电阻，5位全为1时有：

Rx =R||2R||4R||8R||16R



可得等效电阻Rx = 271.6Ω，R = 526.3Ω。考虑到所有电阻（使用的是贴片电阻）都必须取标称值以及减小与理论计算的误差，故将R取500Ω，后面的以标称值取1KΩ、2KΩ、4KΩ、8KΩ。

**2.1.11 LCD 模块接口**

F1 FPGA开发板板载的 LCD模块接口电路如下图所示：

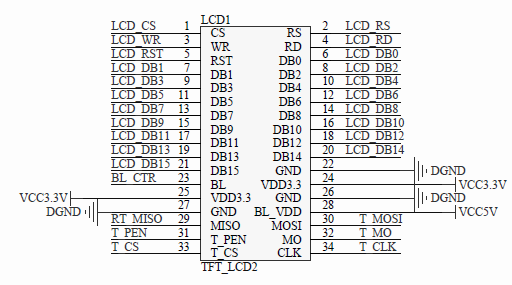


图2.1.11.1 LCD 模块接口

图中 LCD是一个通用的液晶模块接口，支持4.3寸的TFT LCD模块。

图中的T\_MISO/T\_MOSI/T\_PEN/T\_SCK/T\_CS用来实现对液晶触摸屏的控制（支持电阻屏和电容屏）。LCD\_BL用于控制LCD的背光。

**2.1.12 温度传感器接口**

F1 FPGA开发板板载了一个温度传感器接口，电路原理图如下图所示：

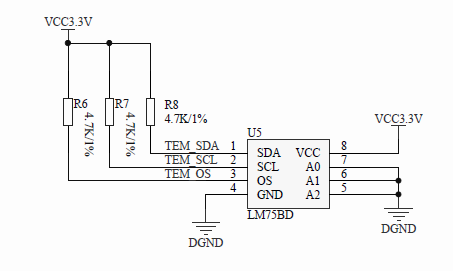


图2.1.12.1 温度传感器接口

温度传感器芯片使用的是LM75BD，该芯片的测温范围-55℃ ～ +125℃，精度±3℃。 这里我们把A0~A2均接地，对LM75BD来说也就是把地址位设置成了0了，写代码的时候要注意这点。TEM\_SCL接在FPGA的N2引脚，TEM\_SDA接在FPGA的M3引脚，另外LM75BD采用IIC协议进行数据的读写，而IIC的串行时钟线SCL和数据线SDA均是开漏的，所以需要接上拉电阻。TEM\_OS是超温报警输出,接在FPGA的P3引脚；

**2.1.13 实时时钟**

F1 FPGA开发板板载了一个实时时钟RTC，电路原理图如下图所示：

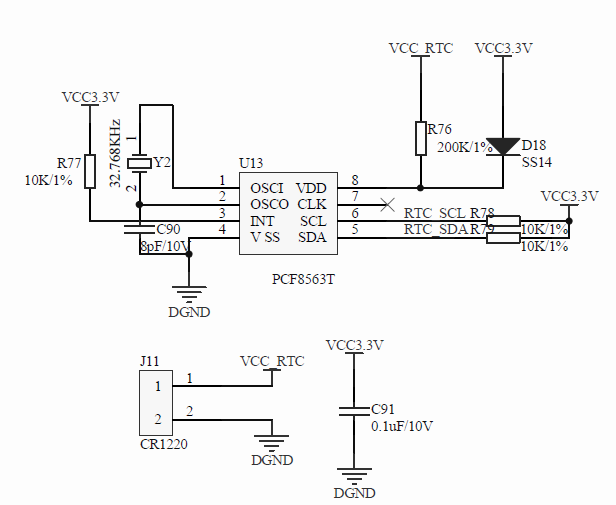
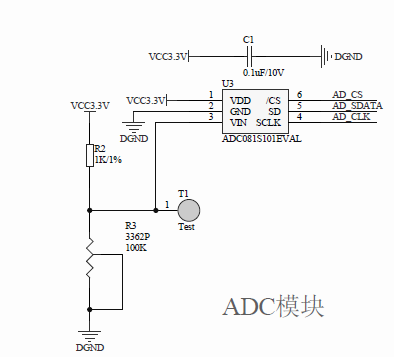


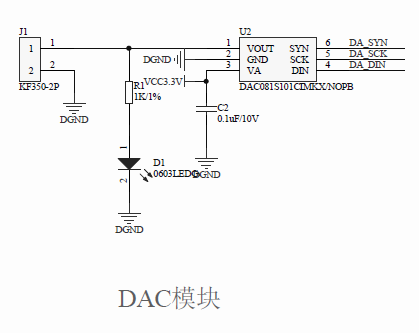
图2.1.13.1 实时时钟 RTC

实时时钟芯片为PCF8563，采用直接供电和备用电源供电两种供电方式，采用IIC协议进行通信。备用电源供电时使用纽扣电池CR1220供电，安放在开发板的电池座。

**2.1.14 AD/DA接口**

F1 FPGA开发板板载了一个AD/DA接口，电路原理图如下图所示：





F1开发板上板载的AD/DA器件为采用SPI协议进行通信的ADC081S101和DAC081S101器件。ADC测量电压变化从0-3.3V，DAC输出范围0-3.3V

**2.1.15 I2S音频编解码**

F1 FPGA开发板板载WM8731高性能音频编解码芯片，电路原理图如下图所示：

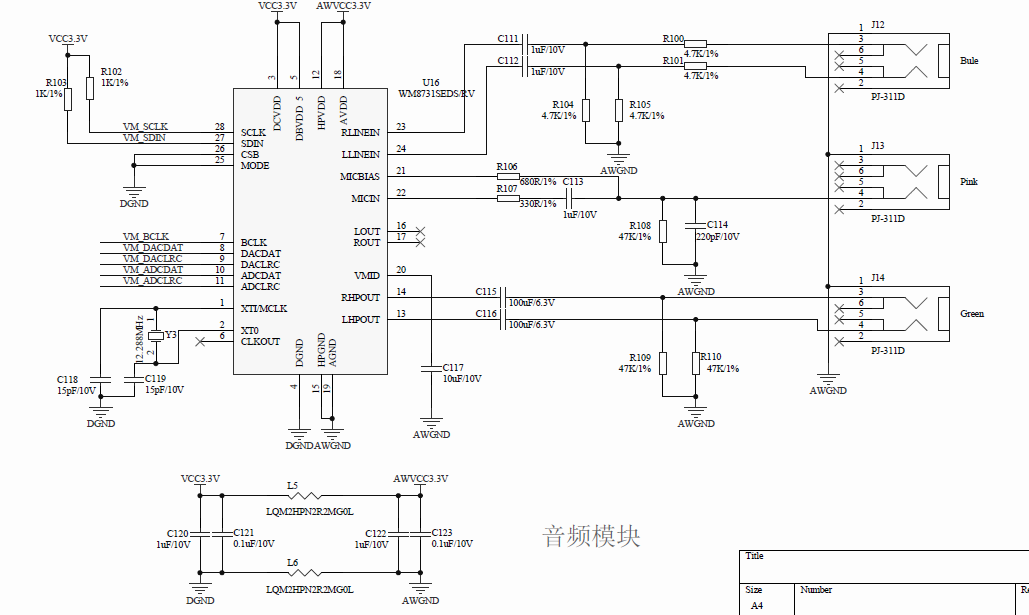
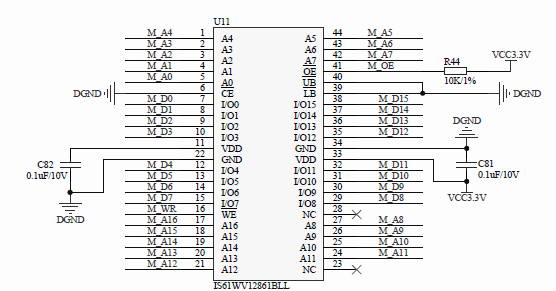


图2.1.15.1 WM8731

WM8731是一颗低功耗、高性能的立体声多媒体数字信号编解码器。该芯片内部集成了24位高性能DAC&ADC，可以播放最高96K@24bit的音频信号。该芯片还结合了立体声差分麦克风的前置放大与扬声器、耳机和差分、立体声线输出的驱动，减少了应用时必需的外部组件，直接可以驱动耳机（16Ω@50mW）和（32Ω/30mW），无需外加功放电路。 MIC是麦克输入，可用于录音机实验，实现录音。PHONE是3.5mm耳机输出接口，可以用来插耳机。LINE\_IN 则是线路输入接口，可以用来外接线路输入，实现立体声录音。

**2.1.16 SRAM**

F1 FPGA开发板板载了SRAM，此部分电路图如下图所示：



图中，U23就是SRAM芯片，型号为IS61WV12816BLL，容量为2Mb字节。该芯片可以满足FPGA的各种大内存需求场合，实现数据的缓存。

**2.1.17 以太网接口（RJ45 ）**

开发板板载了一个以太网接口（RJ45），电路原理图如下图所示：

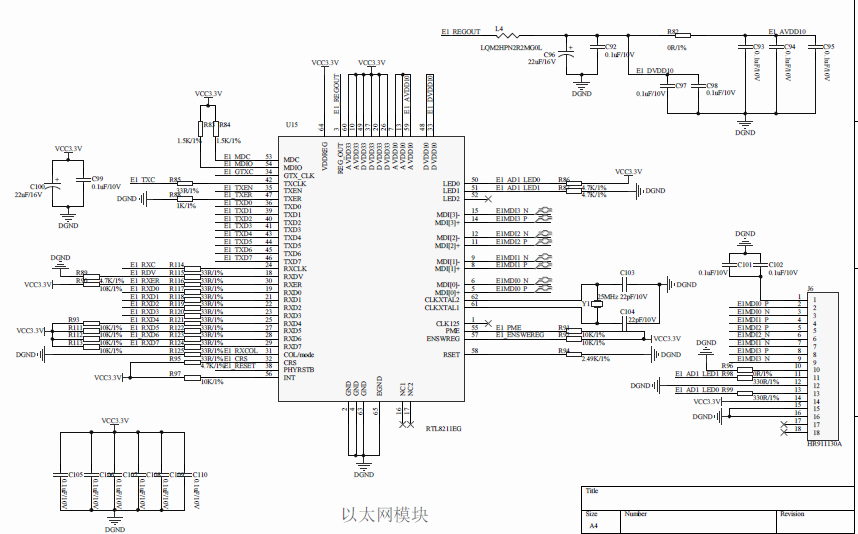
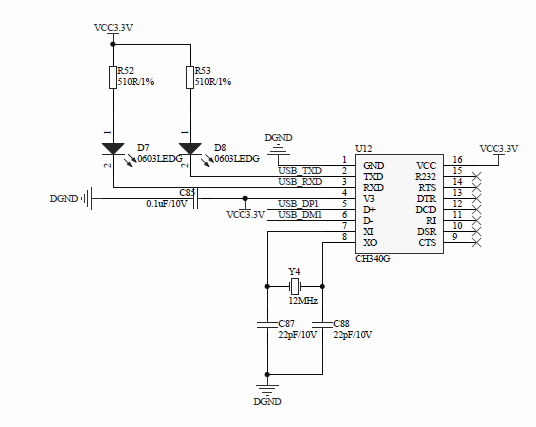


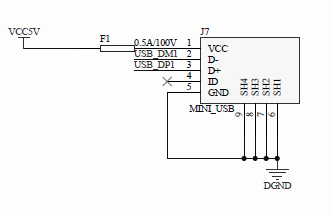
图2.1.17.1 以太网接口

这里我们选择的是RTL8211EG这颗芯片作为FPGA的PHY芯片用于以太网通信。RTL8201是一块 10M/100M/1000M自适应以太网收发器，提供MII/SNI接口的MAC连接。板载一个自带网络变压器的RJ45头（HR911130A）。

**2.1.18 USB串口**

开发板板载了一个USB串口，其原理图如下图所示：





USB转串口，我们选择的是CH340G，是国内芯片公司南京沁恒的产品，稳定性经测试还不错，所以还是多支持下国产。

USB\_232是一个MiniUSB座，提供CH340G和电脑通信的接口，同时可以给开发板供电，VUSB就是来自电脑USB的电源，USB\_232是本开发板的可选供电口 .

**2.1.19 摄像头模块接口**

开发板板载了一个摄像头模块接口，电路原理图如下图所示：



图2.1.19.1 摄像头模块接口

图中P1接口可以用来连接摄像头模块。摄像头模块需要用到全部引脚。

**2.1.21 TF卡接口**

开发板板载了一个TF卡接口，电路原理图如下图所示：

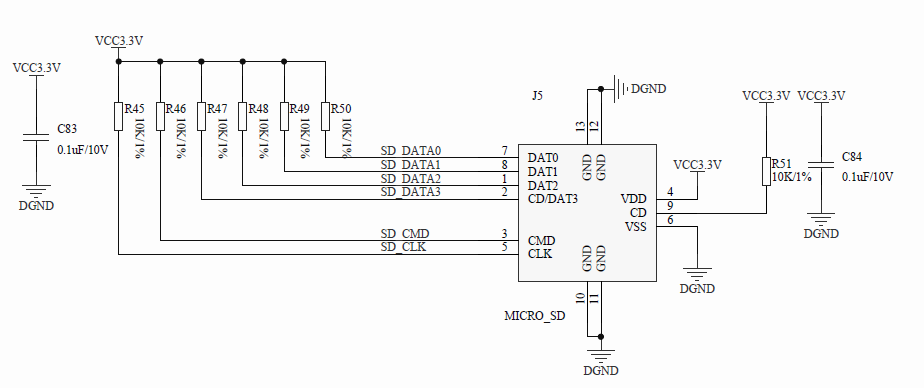


图 2.1.21.1 TF 卡接口

图中MICRO\_SD为TF卡接口，该接口在开发板的底面。TF卡采用4位SDIO方式驱动，理论上最大速度可以达到24MB/S，非常适合需要高速存储的情况。

**2.1.22 SPI FLASH**

板载的SPI FLASH电路原理图如下图所示：

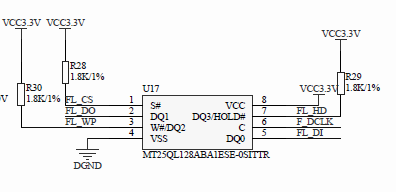
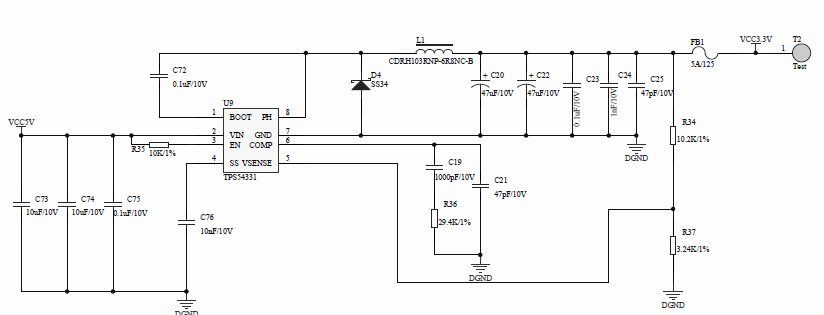


图2.1.22.1 SPI FLASH芯片

SPI FLASH芯片型号为MT25Q128，存储容量为128Mb（16M字节），采用SPI协议和FPGA进行通信，可做为FPGA的配置芯片（完全兼容EPCS16芯片），以保证FPGA在重新上电后仍能继续工作。

**2.1.23 FPGA 电源设计**

ARTIX-7 XC7A50T-1CSG324I FPGA需要3组供电，分别为3.3V,1.8V和1.0V。开发板FPGA供电电源电路如下图所示



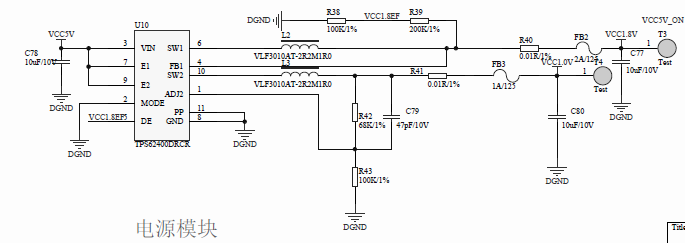


图2.1.23.1 FPGA供电电源电路

**2.1.24 电源适配器输入接口**

开发板由电源适配器供电的电源接口原理图如下图所示：

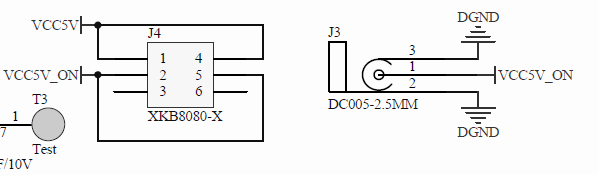


图2.1.24.1 电源适配器供电电源接口

VCC5V\_ON用于外部直流电源输入，输入电压（5V）经过都经过电源开关J4控制是否对开发板供电。J4按下5V和3.3V会点亮电源指示灯。

**2.2 开发板使用注意事项**

为了让大家更好的使用F1 FPGA开发板，我们在这里总结该开发板使用的时候尤其要注意的一些问题，希望大家在使用的时候多多注意，以减少不必要的问题。

1. USB供电电流最大500mA，且由于导线电阻存在，供到开发板的电压，一般都不会有5V，如果使用了很多大负载外设，比如4.3寸屏、7寸屏、摄像头模块等，那么可能引起USB供电不足，所以开发板如果连接大负载模块的朋友，或者同时用到多个模块的时候，建议大家使用一个电源适配器供电5V2A。

2. 当你想使用某个IO口用作其他用处的时候，请先看看开发板的原理图，该IO口是否有连接在开发板的某个外设上，如果有，该外设的这个信号是否会对你的使用造成干扰，先确无干扰，再使用这个IO,尤其使用摄像头模块时。

3. 当液晶显示白屏的时候，请先检查液晶模块是否插好（拔下来重新插试试），如果还不行，可以通过串口看看LCD ID是否正常，再做进一步的分析。

**2.3 FPGA的学习方法**

FPGA作为目前最热门的非CPU类处理器，正在被越来越多的公司选择使用。学习FPGA的朋友也越来越多，初学者可能会认为FPGA很难学，以前只学过51，或者甚至连51都没学过的，一看到FPGA就懵了。其实，万事开头难，只要掌握了方法，学好FPGA，还是非常简单的，这里我们总结学习FPGA的几个要点：

1. 一款实用的开发板。

这个是实验的基础，有时候软件仿真通过了，在板上并不一定能跑起来，而且有个开发板在手，什么东西都可以直观的看到，效果不是仿真能比的。但开发板不宜多，多了的话连自己都不知道该学哪个了，觉得这个也还可以，那个也不错，那就这个学半天，那个学半天，结果学个四不像。倒不如从一而终，学完一个再学另外一个。

2. 掌握方法，勤学慎思。

FPGA不是妖魔鬼怪，不要畏难，FPGA的学习和普通单片机一样，基本方法就是：

a) 了解FPGA的基本结构。

学习FPGA之前需要先对FPGA的基本结构和其功能有个大概的了解，如锁相环PLL、FIFO等。需要知道PLL是用来产生不同频率的时钟，如使用SDRAM时需要产生100MHz的驱动时钟，使用WM8731（音频编解码芯片）时需要生成12MHz的时钟；FIFO用于数据的缓存和异步时钟域数据的传递等。

b) 了解Verilog HDL基本语法

没有软件的硬件就如同行尸走肉一般。软件是硬件的灵魂，硬件是软件的舞台。好的软件设计才能发挥硬件的性能，而软件的精髓在于代码。学习FPGA也是这样，Verilog HDL做为一种硬件描述语言，是对数字电路的一种描述，而数字电路是并行工作的，因而在编写Verilog HDL时要有并行的思想，不同于软件设计语言，软件设计语言是由CPU统一进行处理，一条指令一条指令的串行运行，所以软件设计语言是基于串行的设计思想，因而在写Verilog HDL代码的时候要注意这种差别。另外对于Verilog HDL的基本语法是务必要掌握的，如一般常用的module/endmodule、input/output/inout、wire/reg、begin/end、posedge/negedge、always/assign、if/else、case/default/endcase/parameter/localparam等关键字要清楚它们的作用和区别。掌握了Verilog HDL的基本语法和Verilog HDL的并行设计思想后，会觉得Verilog HDL和C语言一样简单。

3. 多思考，多动手。

所谓熟能生巧，先要熟，才能巧。如何熟悉？这就要靠大家自己动手，多多练习了，光看或说是没什么太多用的。只有在使用FPGA的过程中，才会一点点的熟悉，也只有动手实练，才能对FPGA有切实的感受熟悉了之后，就应该进一步思考，也就是所谓的巧了。我们提供了几十个例程，供大家学习，跟着例程走，无非就是熟悉FPGA的过程，只有进一步思考，才能更好的掌握FPGA，也即所谓的举一反三。例程是死的，人是活的，所以，可以在例程的基础上，自由发挥，实现更多的其他功能，并总结规律，为以后的学习和使用打下坚实的基础，如此，方能信手拈来。

所以，学习一定要自己动手，光看视频，光看文档，是不行的。举个简单的例子，你看视频，教你如何煮饭，几分钟估计你就觉得学会了。实际上你可以自己测试下，是否真能煮好？机会总是留给有准备的人，只有平时多做准备，才可能抓住机会。

只要以上三点做好了，学习FPGA基本上就不会有什么太大问题了。