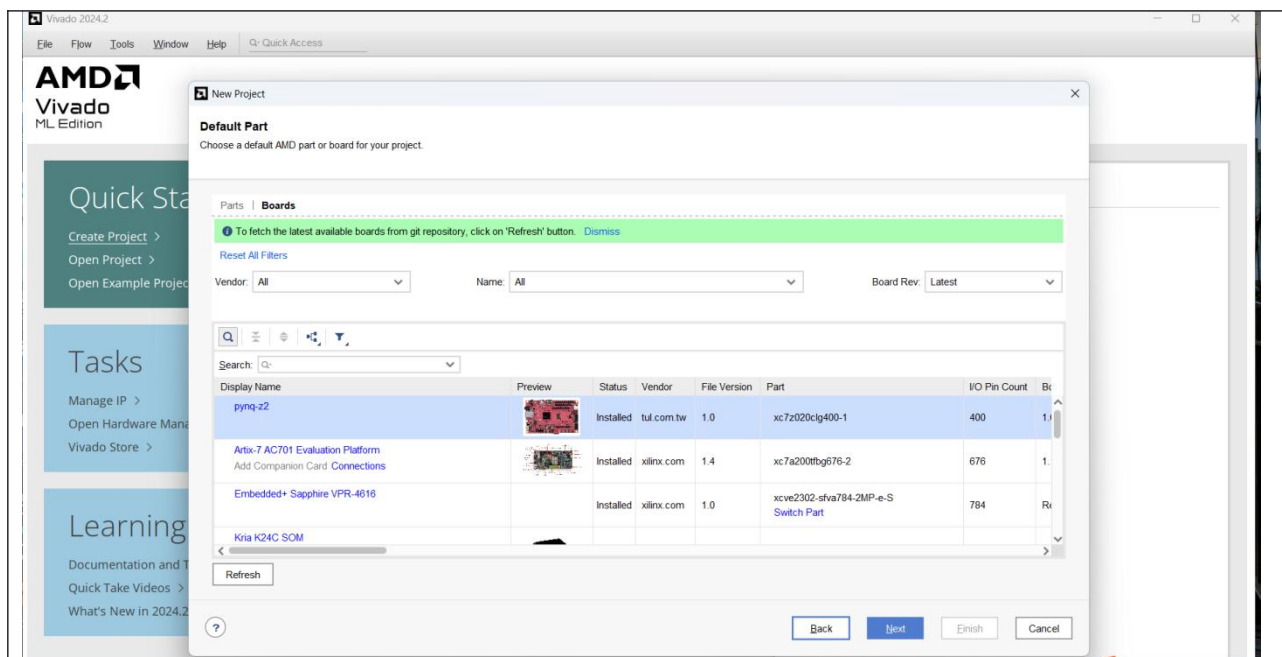
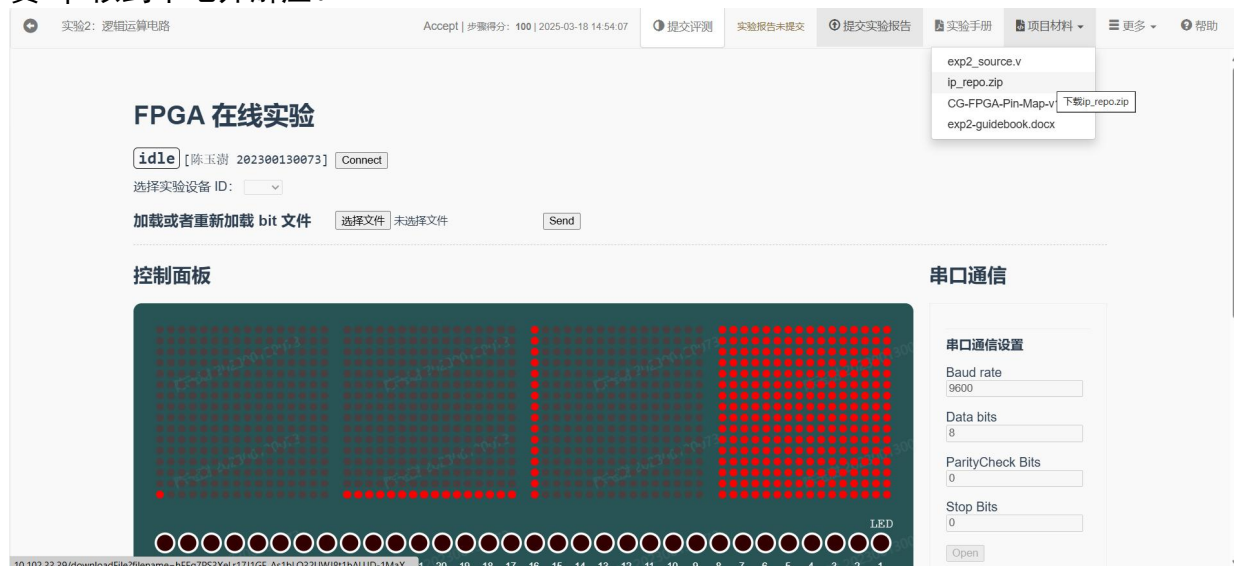


计算机组成与设计_____课程实验报告

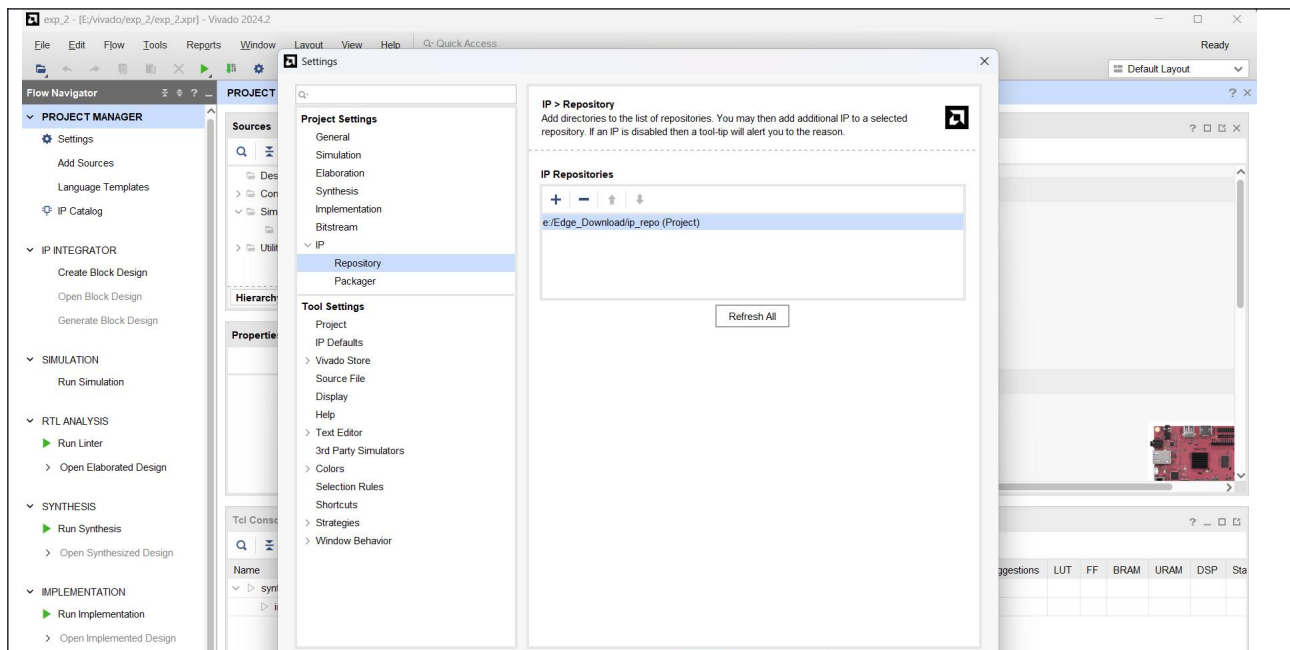
| | | |
|--|-----------------|----------------|
| 学号：202300130073 | 姓名：陈玉澍 | 班级：计科 23 级 1 班 |
| 实验题目： 实验 2 逻辑运算电路 | | |
| 实验学时：2 | 实验日期：2025. 3. 4 | |
| 实验目的： 设计一个能实现 1 位逻辑乘 ab 、逻辑或 $a+b$ 、半加 $(a\oplus b)$ 的逻辑运算电路。 | | |
| 实验软件和硬件环境： 软件环境： Vivado 软件、FPGA 实验平台 硬件环境： 1. 实验室台式机 2. FPGA 服务器，PYNQ-Z2 开发板 | | |
| 实验原理和方法： 一、逻辑运算基本原理 逻辑乘 (ab)：通过与门 (AND) 实现，仅当输入 a 和 b 均为 1 时输出 1。 逻辑或 ($a+b$)：通过或门 (OR) 实现，当输入 a 或 b 任意为 1 时输出 1。 半加 ($a\oplus b$)：通过异或门 (XOR) 实现，当输入 a 和 b 不同时输出 1，相同则输出 0。 二、硬件实现方法 使用 Vivado 软件进行 FPGA 设计，通过原理图输入方式连接基本逻辑门模块：利用 cg_fpga IP 核实现硬件接口（如 FIXED_IO、DDR 等）与逻辑电路的连接；通过封装 1 位逻辑运算电路为层次化模块，扩展实现 4 位逻辑运算功能。 | | |
| 实验步骤： (1) 创建工程：打开本地安装的 Vivado 2024. 2，新建项目，选择 pynq-z2 器件。 | | |



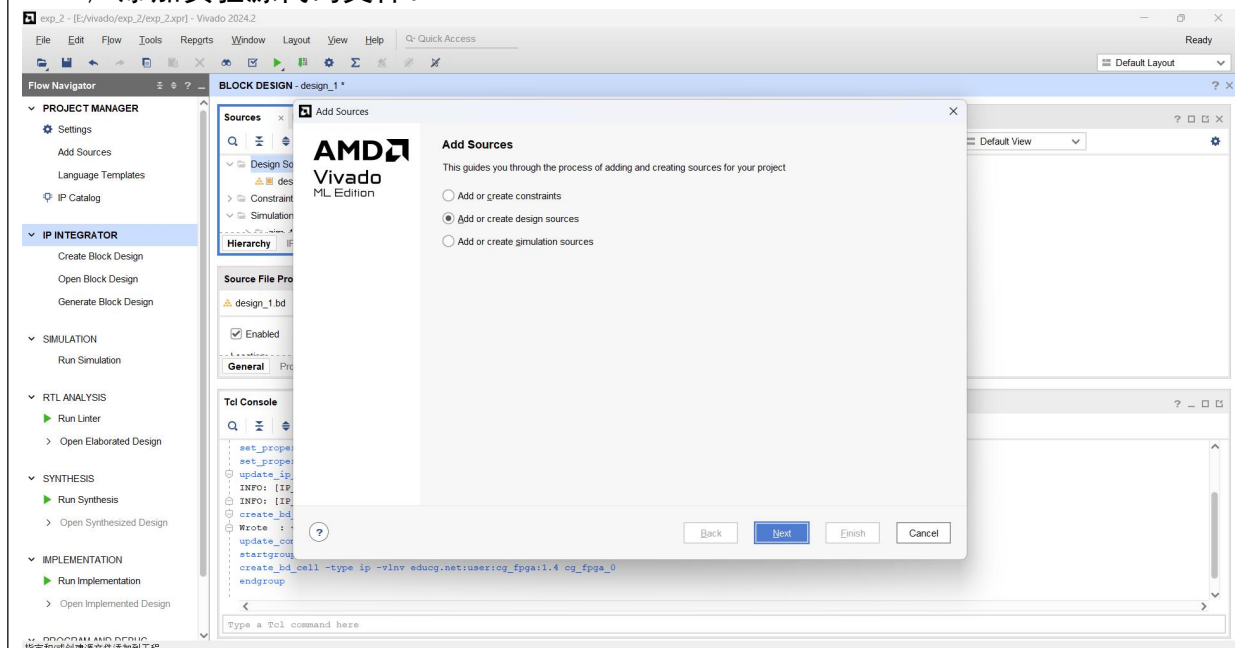
(2) 添加实验环境：进入 FPGA 在线实验环境，点击右上角项目材料下载实验源代码和希冀 ip 核到本地并解压。

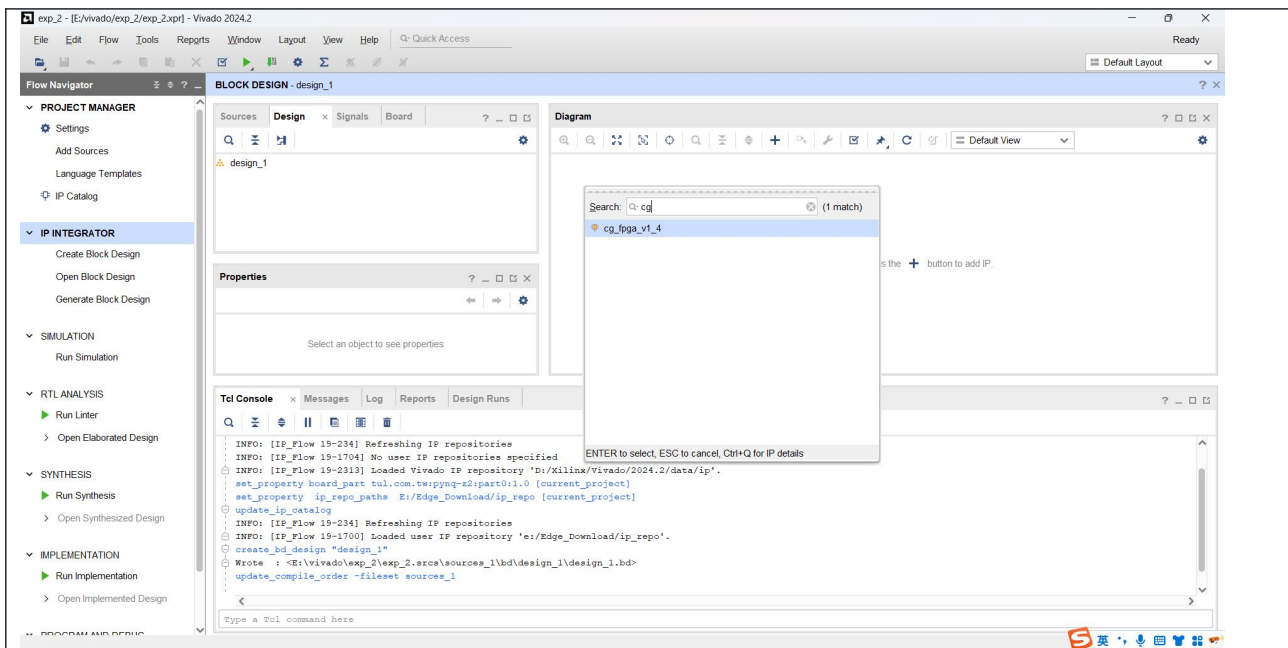


(3) 在 Vivado 项目中，点击 Settings→IP→Repository，将上一步解压后的 ip_repo 文件夹的位置添加进 IP 搜索目录。



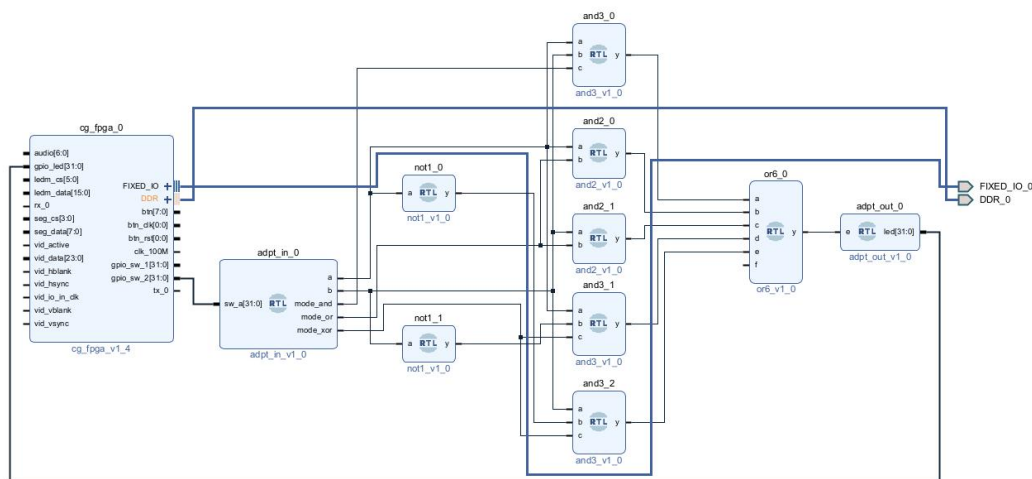
(4) 点击 Sources 窗口中的+, 选择 Add or create design sources → Next → Add File, 添加实验源代码文件。



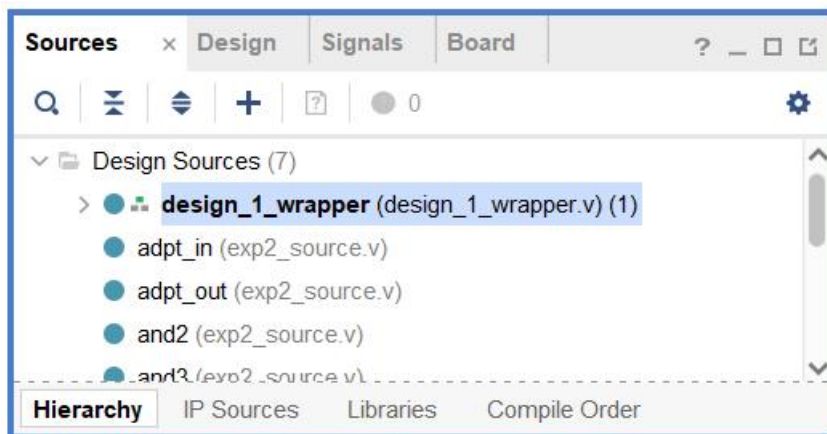
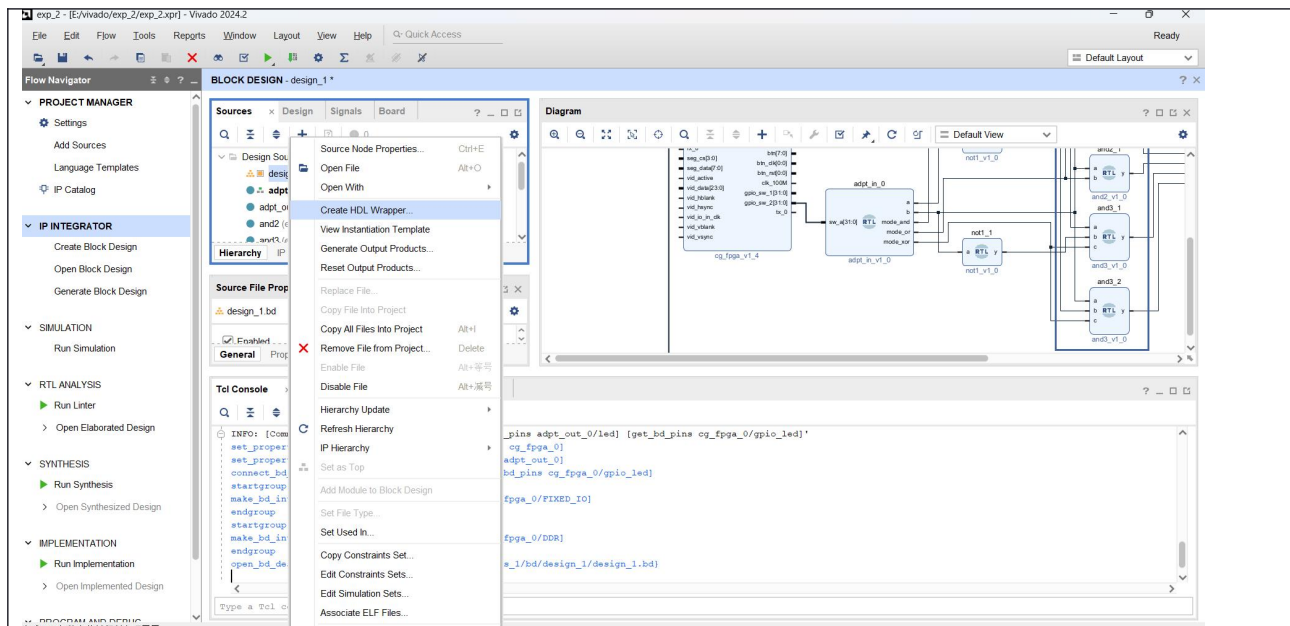


(6) 在 Sources 窗口下的 Design sources 中，根据电路图拖拽相应模块，完成原理图的输入。

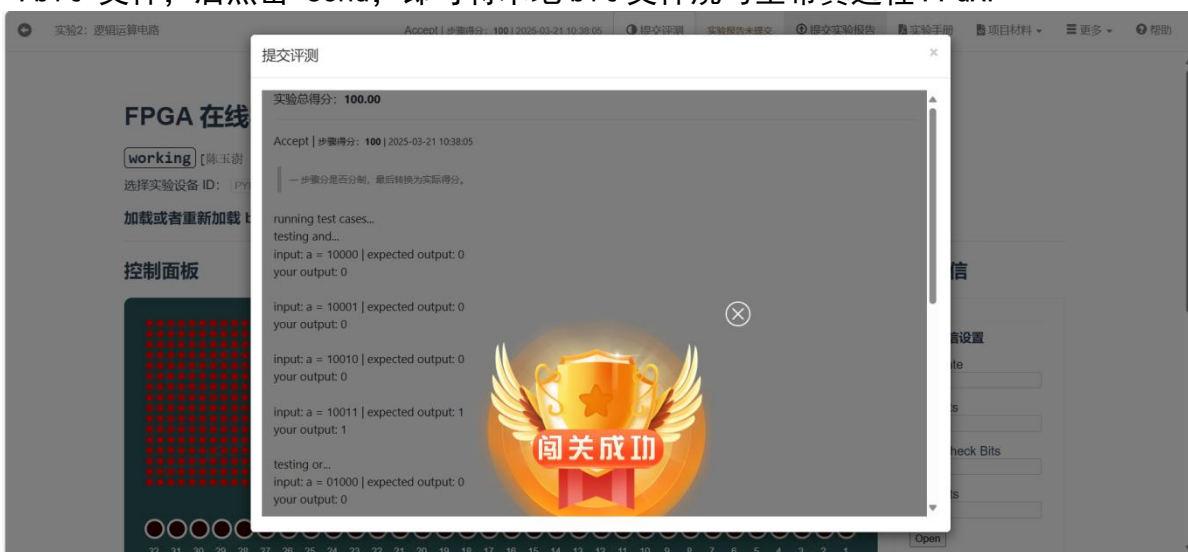
(7) 将输入的模块与 cg_fpga 如下图连接，并选择 cg_fpga 模块上的 FIXED_IO 和 DDR，点击右键→Make External。



(8) 右击 Sources 下顶层设计图标→Create HDL Wrapper，待 Wrapper 正确生成后，点击左下方 Generate Bitstream，开始综合并生成 bit 文件。注意：综合前 wrapper 模块应被设置为顶层（加粗表示），若自动设置错误，需右击 wrapper 图标点击 Set as Top 手动设置。



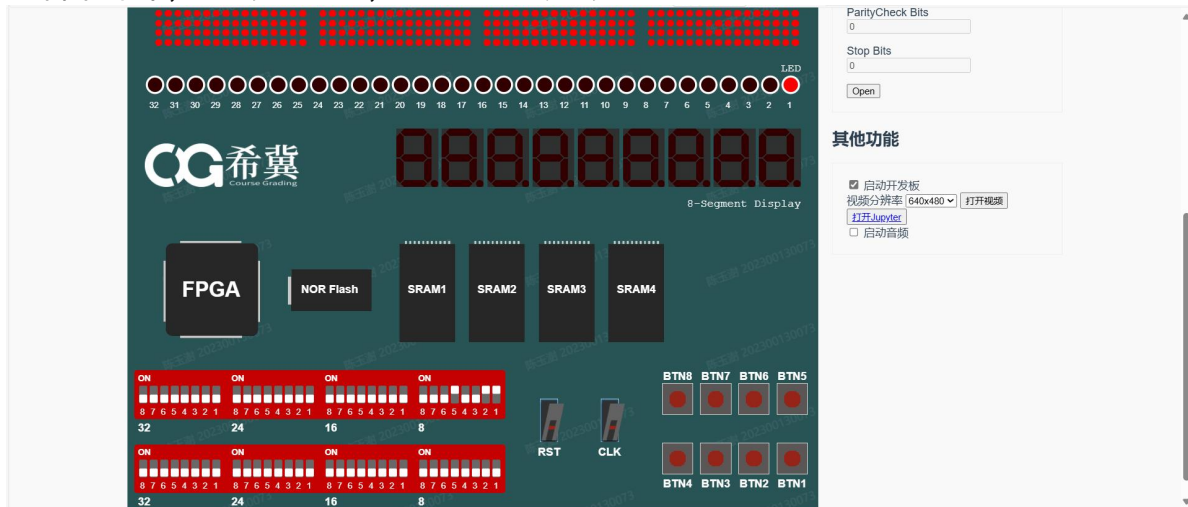
(9) 通过 FPGA 云实验平台，可在线分配远程 FPGA 硬件开发板。首先点击 connect 按钮，然后在下拉菜单中选择任意空闲的开发板，并点击 Choose File 中选择上一步生成的 *.bit 文件，后点击 send，即可将本地 bit 文件烧写至希冀远程 FPGA。



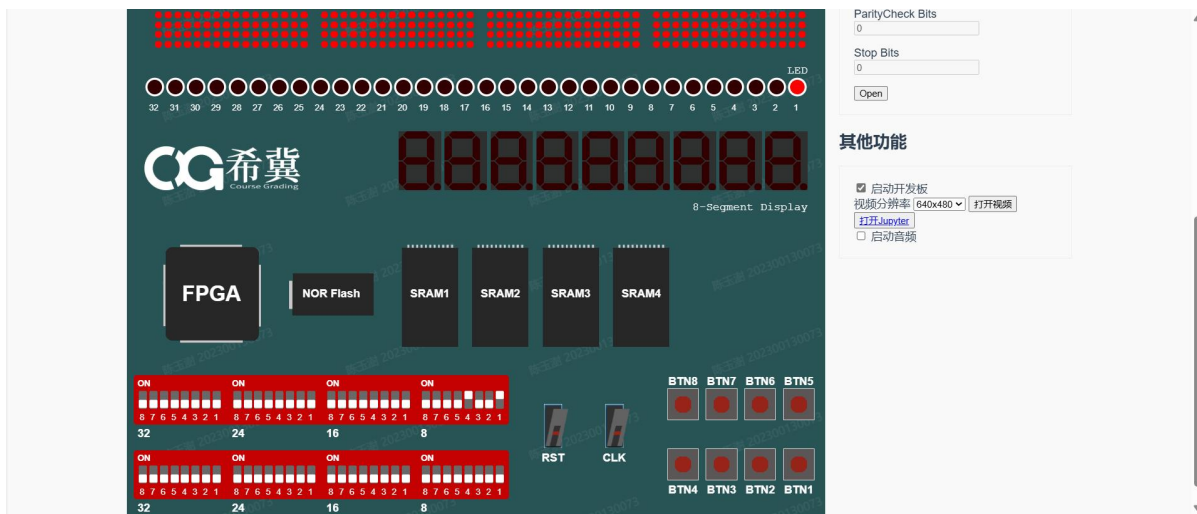
利用输入开关及发光二极管 LD 测试逻辑运算部件的功能并记录测试结果。

mode_and, mode_or, mode_xor 管脚分别绑定在上排拨码开关 5-3, a, b 管脚分别绑定在上排拨码开关 2-1, e 管脚绑定在 LED1 上。

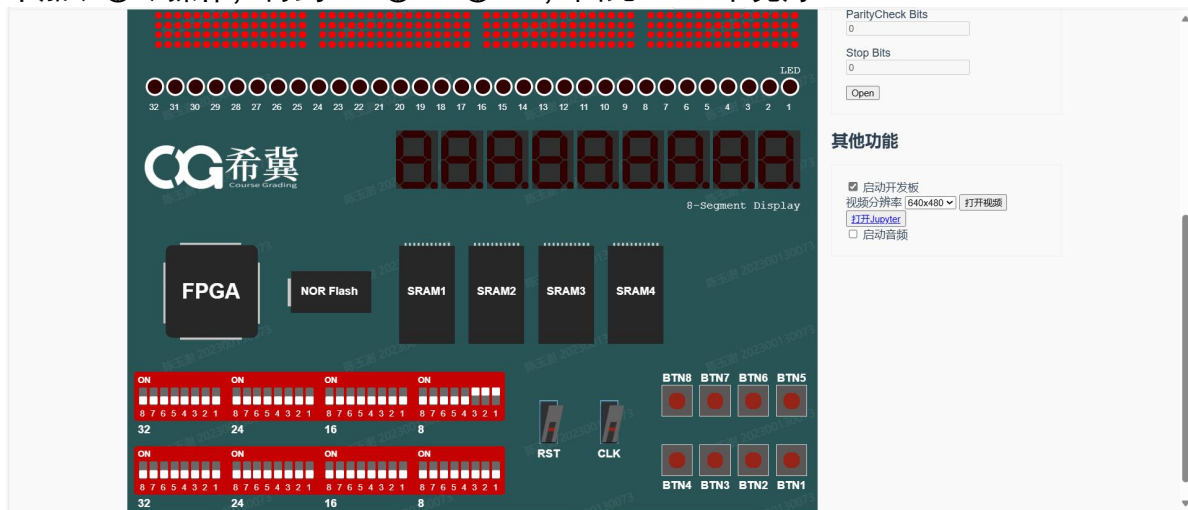
首先！输入为 10011 时，此时 $a=1, b=1, \text{more_and}=1, \text{more_or}=0, \text{more_xor}=0$ ，即 a 和 b 做逻辑乘操作，得到 $e=ab=1$ ，因此 LED1 亮灯。



下面是输入为 01001 时，此时 $a=0, b=1, \text{more_and}=0, \text{more_or}=1, \text{more_xor}=0$ ，即 a 和 b 做逻辑或操作，得到 $e=a+b=1$ ，因此 LED1 亮灯。



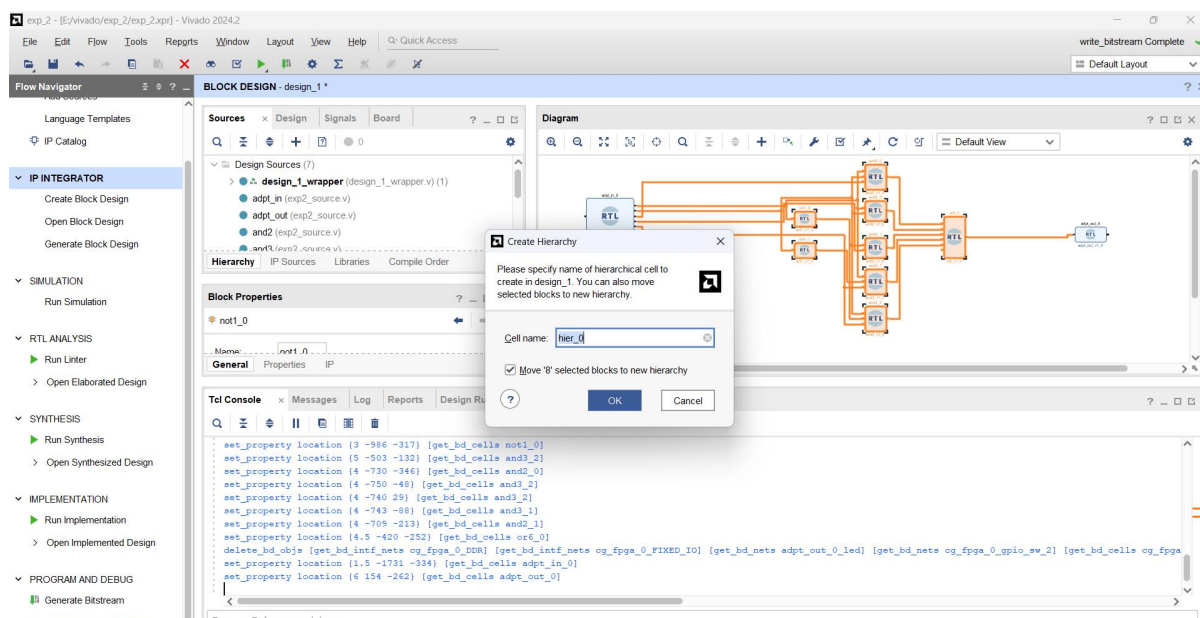
还有当输入为 00111 时，此时 $a=1, b=1, \text{more_and}=0, \text{more_or}=0, \text{more_xor}=1$ ，即 a 和 b 做半加 ($a \oplus b$) 操作，得到 $e=a \oplus b=1 \oplus 1=0$ ，因此 LED1 不亮灯。



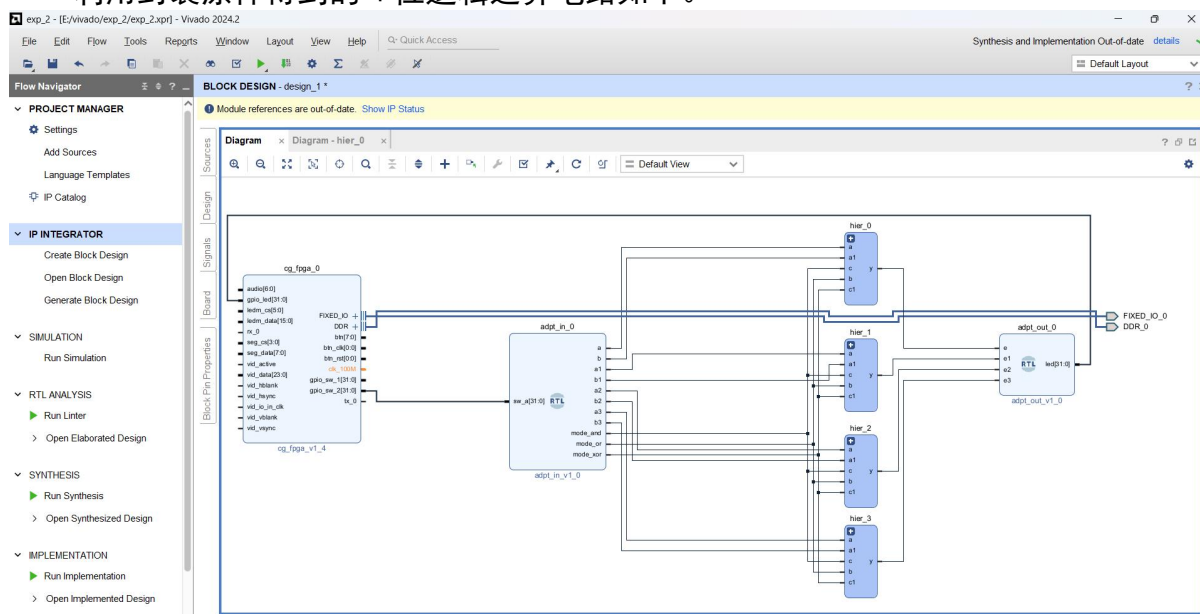
选做：

利用一位逻辑运算的结果实现两个 4 位二进制数 A (a3a2a1a0) 和 B(b3b2b1b0) 的逻辑运算并生成元件符号。使平台工作于模式 5，当按键开关不足时，可使用平台上红色的拨码开关。

由于此时是 4 位的逻辑运算电路，即 4 个上述电路。我们可以将上面的电路进行封装。选中要封装的电路部分，右键选择“Create Hierarchy”。



利用封装原件得到的 4 位逻辑运算电路如下。



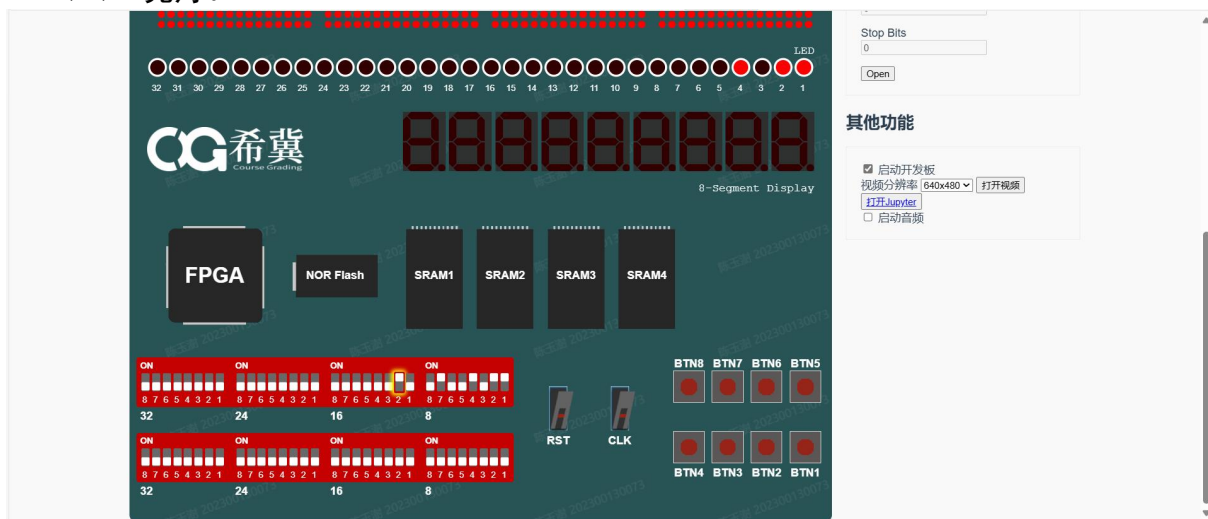
利用输入开关及发光二极管 LD 测试逻辑运算部件的功能并记录测试结果。

mode_and, mode_or, mode_xor 管脚分别绑定在上排第二组拨码开关 3-1，a3-a0 管脚分别绑定在上排拨码开关 7/5/3/1，b3-b0 管脚分别绑定在上排拨码开关 8/6/4/2，e3-e0 管脚绑定在 LED4-1 上。

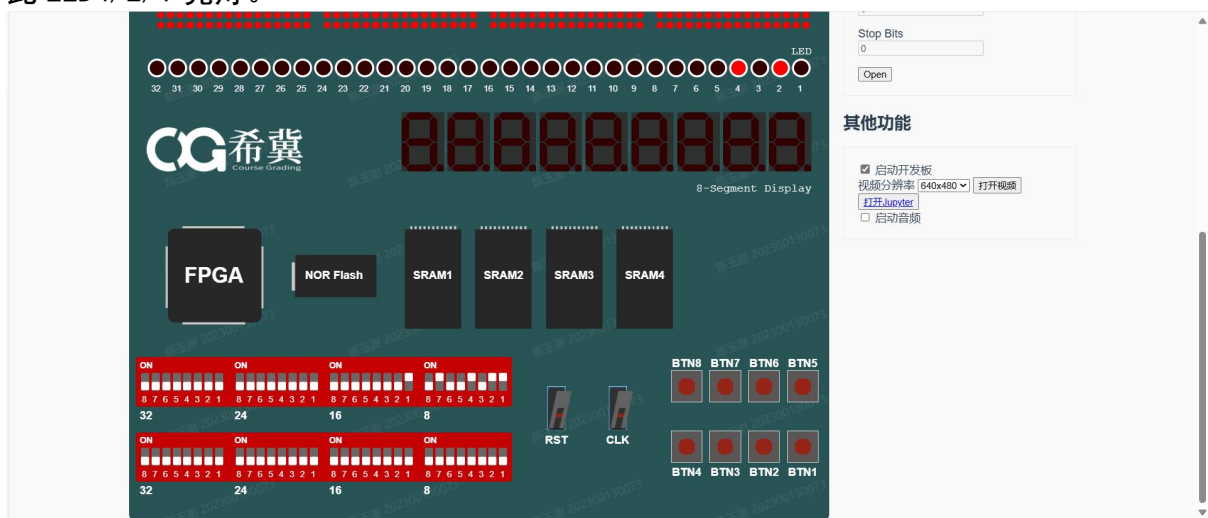
首先！输入为 10001001011 时，即 mode_and=1, mode_or=0, mode_xor=0, a3-a0=1001, b3-b0=0011。此时 a 和 b 做逻辑乘操作，得到 e=ab=0001，LED1 亮。



还有当输入为 01001001011 时，即 $\text{mode_and}=0$, $\text{mode_or}=1$, $\text{mode_xor}=0$, $a3-a0=1001$, $b3-b0=0011$ 。即 a 和 b 做逻辑或操作，得到 $e=a+b=1001+0011=1011$ ，因此 LED4/2/1 亮灯。



还有当输入为 00101001011 时，即 $\text{mode_and}=0$, $\text{mode_or}=0$, $\text{mode_xor}=1$, $a3-a0=1001$, $b3-b0=0011$ 。即 a 和 b 做半加 ($a \oplus b$) 操作，得到 $e=a \oplus b=1001 \oplus 0011=1010$ ，因此 LED4/2/1 亮灯。



结论分析与体会：

这次实验使我提高了对于 Vivado 软件的应用熟练度，对逻辑运算电路有了更加深入、深刻的认识和掌握。同时，选做实验锻炼了我举一反三的能力，通过封装和利用封装的元件，可以大大减小电路的复杂程度和工作量。

就封装时框选电路总会选上用不到的电路，无法只框选真正需要的部分电路问题处理的：
持续按住 shift，左键想要封装的元件和电路。