



杭州电子科技大学
HANGZHOU DIANZI UNIVERSITY

实验项目

A composite image showing a computer keyboard and mouse in a blue and green color scheme, overlaid with binary code (0s and 1s) and a faint grid pattern.

主讲教师：章复嘉

实验四 寄存器堆设计

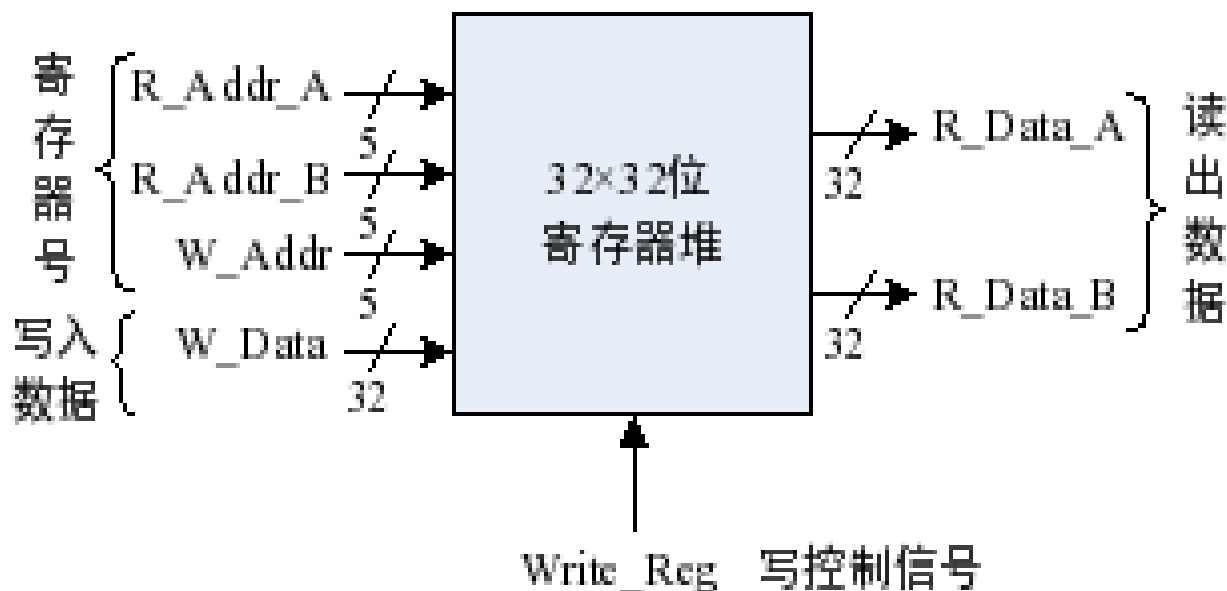
❖ 1、实验目的

- 学习使用 Verilog HDL 语言进行**时序电路**的设计方法；
- 掌握灵活运用 Verilog HDL 语言进行**各种描述与建模的技巧和方法**；
- 学习**寄存器堆的数据传送与读写工作原理**，掌握寄存器堆的设计方法。

实验四 寄存器堆设计

❖ 2、实验内容与原理

- 设计一个 **32×32 位** 的寄存器堆（即含有 32 个寄存器，每个寄存器 32 位）



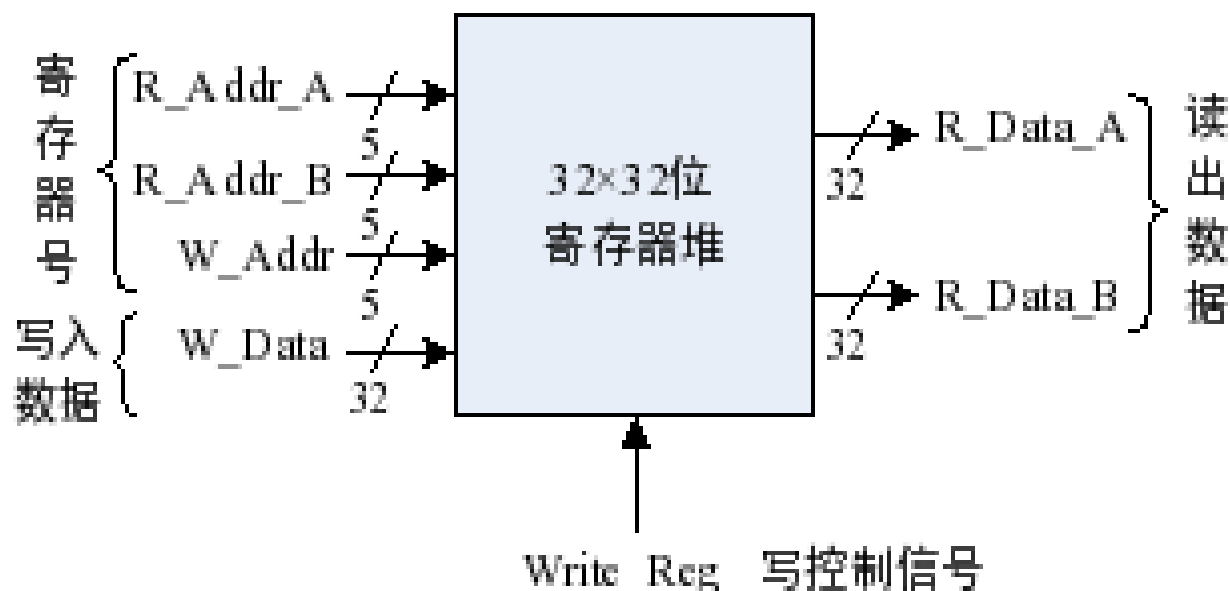
输入输出端口示意图

❖ 双端口读： 2 个读端口

❖ 单端口写： 1 个写端口

实验四 寄存器堆设计

- ❖ **读访问操作：无需时钟同步**，只要给出寄存器地址，即可读出寄存器中的数据。
- ❖ **写访问操作：需要时钟同步**，所有写入操作的输入信号必须在时钟边沿来临时，已经有效（**Write_Reg=1、地址和数据**）。



实验四 寄存器堆设计

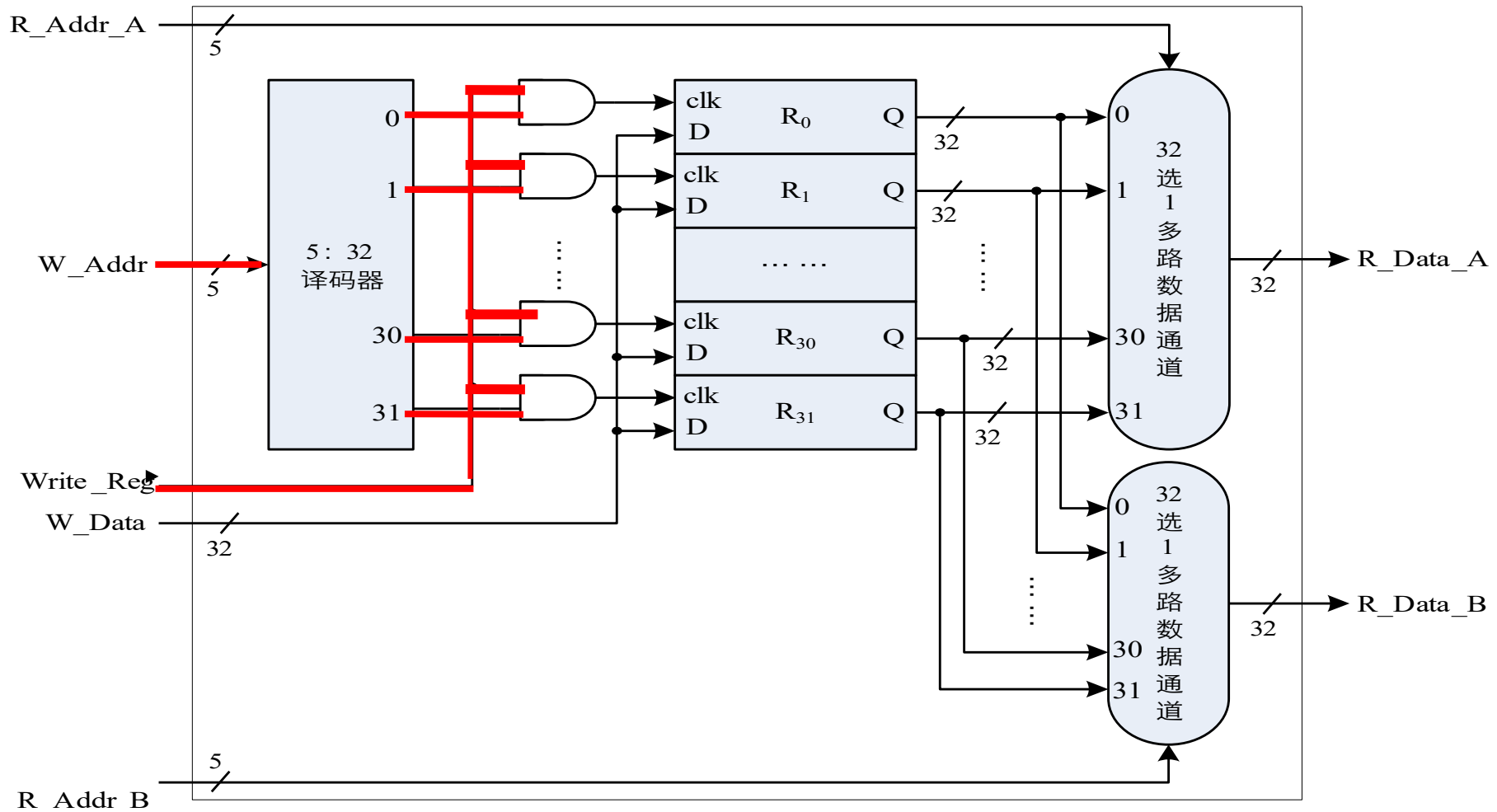
❖2、实验内容与原理

寄存器堆功能表

输入信号					输出信号		操作
R_Ad dr_A	R_Ad dr_B	Write _Reg	W_Ad dr	W_Da ta	R_Dat a_A	R_Dat a_B	
寄存器 号	—	—	—	—	A 口数 据	—	读 A 口
—	寄存器 号	—	—	—	—	B 口数 据	读 B 口
—	—	1	寄存器 号	写入数 据	—	—	写操 作

实验四 寄存器堆设计

❖ 2、实验内容与原理 寄存器堆逻辑结构图



❖ 实验内容与原理

- 实验实现：

- 寄存器堆： **reg** 类型信号的**数组**

```
reg [31:0] REG_Files[0:31];
```

- 读操作：**组合逻辑电路**

```
assign R_Data_A = REG_Files[R_Addr_A];
```

```
assign R_Data_B = REG_Files[R_Addr_B];
```

- 写操作：**时序逻辑电路**

- 需要 **Reset** 信号：用于初始化寄存器（全部清零）

- 需要 **clk** 信号：用于写入寄存器

实验四 寄存器堆设计

```
always @(posedge Clk or posedge Reset)
begin
    if(Reset) // 高电平有效， =1 则初始化
        .....// 初始化 32 个寄存器
    else
        begin
            if (Write_Reg)
                .....// 写入寄存器；
        end
    end
end
```


❖ 3、实验要求

- 编程实现基本的寄存器堆模块，并通过仿真验证；
- 编写一个实验验证的顶层模块，调用该寄存器堆模块
- 参考方法如下：
 - 使用 5 位开关提供读写的寄存器地址；
 - 1 位开关提供 Write_Reg 信号；指定 Write_Reg=0 时执行读操作；=1 时执行写操作；
 - 2 位开关作为复用控制：若为读操作时，用于选择读出的 32 位数据的某个字节到 8 位 LED 灯显示；若为写操作，则选择 4 个指定数据之一作为写入数据。
 - 1 个按钮提供 Clk；1 个按钮提供 Reset；一个按钮作为读 A 端口 /B 端口的选择；

❖ 3、实验要求

- **实验室任务：**
 - **配置管脚：**见下表
 - **生成 *.bit 文件，下载到 Nexys3 实验板中。**
 - **完成板级验证。**
- **撰写实验报告。**

实验四 寄存器堆设计

❖ 信号配置表

	信号	配置设备管脚	功能说明
输入信号	地址信号	5 个逻辑开关	提供读 A、读 B、写地址
	写信号	1 个逻辑开关	=1 为写操作； =0 为读操作
	选择信号	2 个逻辑开关	读操作时，选择显示的字节； 写操作时，选择要写入的数据
	Clk、Reset	2 个按钮	必须连接到时钟引脚 BTND (C9) 或 BTNR (D9)
	A/B 读端口选择	1 个按钮	选择读出 A 端口还是 B 端口
输出			

❖ 4、实验步骤

- 在 Xilinx ISE 中创建工程，编源码，然后编译、综合
- 编写激励代码，观察仿真波形，直至验证正确
- **实验准备：**
 - 设置 N3 板卡电源开关跳线 J1，选择从 USB 取电；
 - 用 USB 电缆连接 PC 机和 N3 板卡；
 - 开 N3 实验板的电源开关；
- 在 PC 机上打开工程文件，进行**管脚配置**。
- **生成编程文件 *.bit，下载到板卡中。**
- **实验。**

实验四 寄存器堆设计

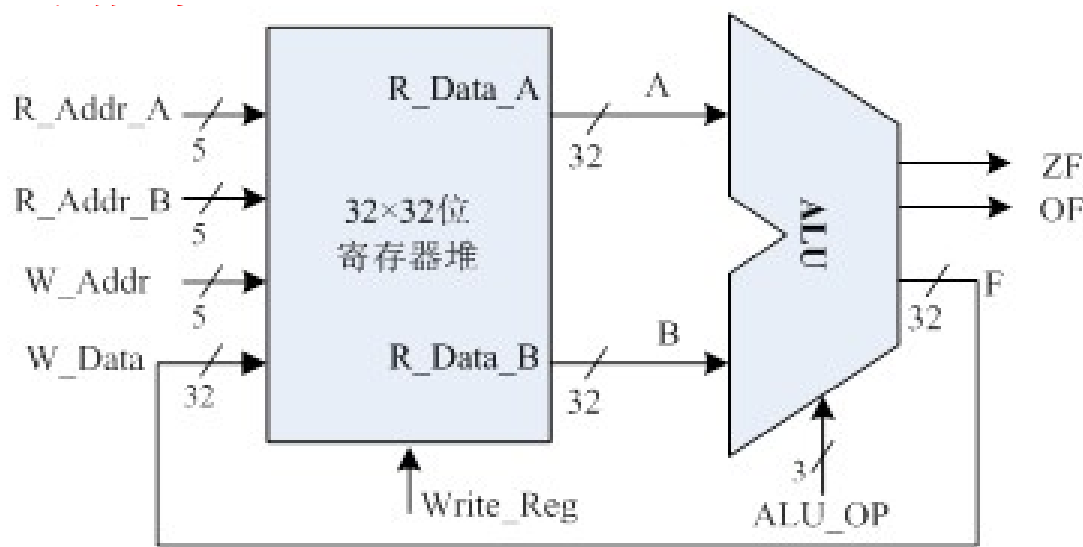
❖ 5、思考与探索：必做（1）和（2）


- （1）选择 8 个寄存器执行读写操作，将实验结果记录到表中，结果是否符合预期；如果不符，分析原因。
- （2）修改基本寄存器堆模块，实现 MIPS 计算机的寄存器堆，以供后续 MIPS CPU 的设计使用。
- （3）谈谈实验中读操作和写操作在时序上有何区别？反映到电路实现上，又有何不同？

实验四 寄存器堆设计

❖ 思考与探索

- （4）利用实验三 ALU 模块和本实验的寄存器堆模块，编写一个顶层模块，完成 $R_i \theta R_j \rightarrow R_k$ 的操作（即 2 个寄存器数据做某种运算，结果送回第 3 个寄存器中）运算功能 θ 由 ALU 模块中的 ALU_OP 作





The End!