

实验二：组合逻辑电路设计

一、实验目的

1. 学习 Verilog HDL 基本语法；
2. 巩固 Vivado 环境下的 Verilog HDL 编程设计的基础；
3. 掌握组合逻辑器件的 Verilog HDL 语言程序设计的方法。

二、实验内容

1. 实现以下组合逻辑功能：优先编码器，比较器，算术逻辑单元。

三、实验要求

1. 在 PC 机上完成相应的逻辑功能仿真，对结果进行分析；
2. 完成下载，在实验板上对程序进行验证；
3. 信号的输入通过 Basys3 上的拨码开关实现，信号的输出通过 Basys3 上的 LED 灯显示。

四、实验步骤

1. 编写一段 Verilog 程序，利用 Basys3 实现 8 线-3 线优先编码器。

编码器通常分为两大类：普通编码器和优先编码器。其中普通编码器就是对某一个给定时刻只能对一个输入信号进行编码的编码器，它的输入端口不允许同一时刻出现两个以上的有效输入信号，否则输出将发生混乱；优先编码器就是对某一个给定时刻只对优先级最高的输入信号进行编码的编码器，它的输入端口允许多个输入信号同时有效。

8 线-3 线优先编码器的逻辑框图如图 2.1 所示，有 8 个信号输入端，3 个二进制编码输出端，输入输出均以高电平作为有效电平。EN 为输入使能端，EN=1 时编码器工作。GS、GC 为工作状态指示信号，当 EN=1 且至少有一个有效输入时 GS 输出值为 1、GC 输出值为 0，当 EN=1 且没有有效输入时 GS 输出值为 0、GC 输出值为 1，其真值表如表 2.1 所示。

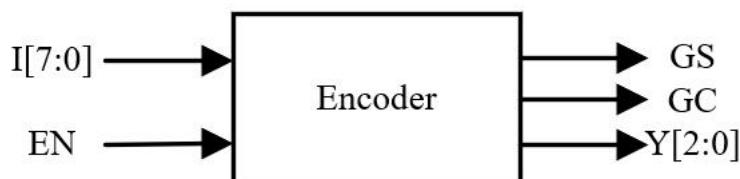


图 2.1 8 线-3 线优先编码器逻辑框图

表 2.1 8 线-3 线优先编码器真值表

EN	输入								输出				
	I[7]	I[6]	I[5]	I[4]	I[3]	I[2]	I[1]	I[0]	Y[2]	Y[1]	Y[0]	GS	GC
0	X	X	X	X	X	X	X	X	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	1
1	1	X	X	X	X	X	X	X	1	1	1	1	0
1	0	1	X	X	X	X	X	X	1	1	0	1	0
1	0	0	1	X	X	X	X	X	1	0	1	1	0
1	0	0	0	1	X	X	X	X	1	0	0	1	0
1	0	0	0	0	1	X	X	X	0	1	1	1	0
1	0	0	0	0	0	1	X	X	0	1	0	1	0

1	0	0	0	0	0	0	1	X	0	0	1	1	0
1	0	0	0	0	0	0	0	1	0	0	0	1	0

其中，I[7]-I[0]对应拨码开关 SW7-SW0，Y[2]-Y[0]对应 LED 灯 LD2-LD0，EN 对应拨码开关 SW10；GS 对应 LED 灯 LD10，GC 对应 LED 灯 LD11。（使用 case 语句或者 if 语句）

2. 编写一段 Verilog 程序，利用 Basys3 实现比较器。

比较器的逻辑框图如图 2.2 所示，A、B 为比较器输入端口，RST 为比较器使能端口，AGTB、AEQB、ALTB 为比较器输出端口。数字比较器的设计过程中，通常依据两组二进制数码的数值大小来进行比较，即 A>B、A=B 或 A<B，这三种情况有一种值为真。



图 2.2 比较器逻辑框图

各引脚说明：A、B：皆为二位输入信号，RST 为使能信号。

AGTB：当 A>B 时，其值为 1，否则为 0；

AEQB：当 A=B 时，其值为 1，否则为 0；

ALTB：当 A<B 时，其值为 1，否则为 0；

当 RST 为低电平时，比较器停止工作，AGTB、AEQB、ALTB 值都为 0。

其中，A 对应拨码开关 SW1、SW0，B 对应拨码开关 SW3、SW2；RST 对应拨码开关 SW5，AGTB 对应 LD0，AEQB 对应 LD1，ALTB 对应 LD2。

3. 编写一段 Verilog 程序，利用 Basys3 实现算术逻辑单元。

算术逻辑单元(Arithmetic Logic Unit, ALU)是 CPU 的核心组成部分，是专门执行算术运算和逻辑运算的电路。ALU 执行运算操作时，必须首先分析这条指令的操作码是什么，然后决定具体的操作类型。其中 opcode 为指令，a、b 为操作数，out 为 ALU 的输出端口，简单的 ALU 逻辑图如图 2.3 所示。

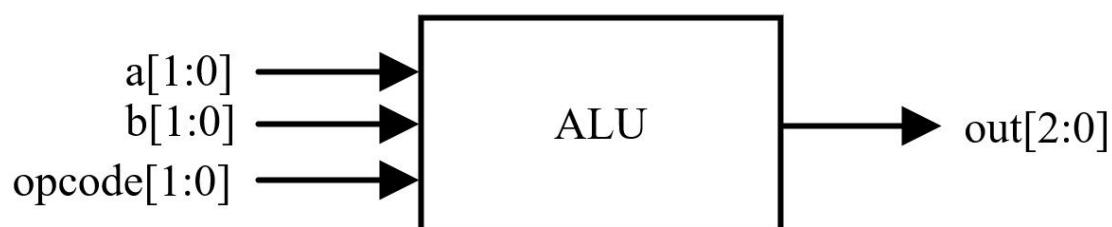


图 2.3 ALU 逻辑图

表 2.2 ALU 指令列表

操作指令	操作类型
opcode=0	求和

opcode=1	相减
opcode=2	与
opcode=3	或

其中，opcode 对应拨码开关 SW11、SW10，a 对应拨码开关 SW8、SW7，b 对应拨码开关 SW5、SW4，out 对应 LED 灯 LD9、LD8、LD7。(建议使用 Verilog 中的“define”编译指令定义 ALU 操作指令)