

数字逻辑基础课程 project_3 设计报告

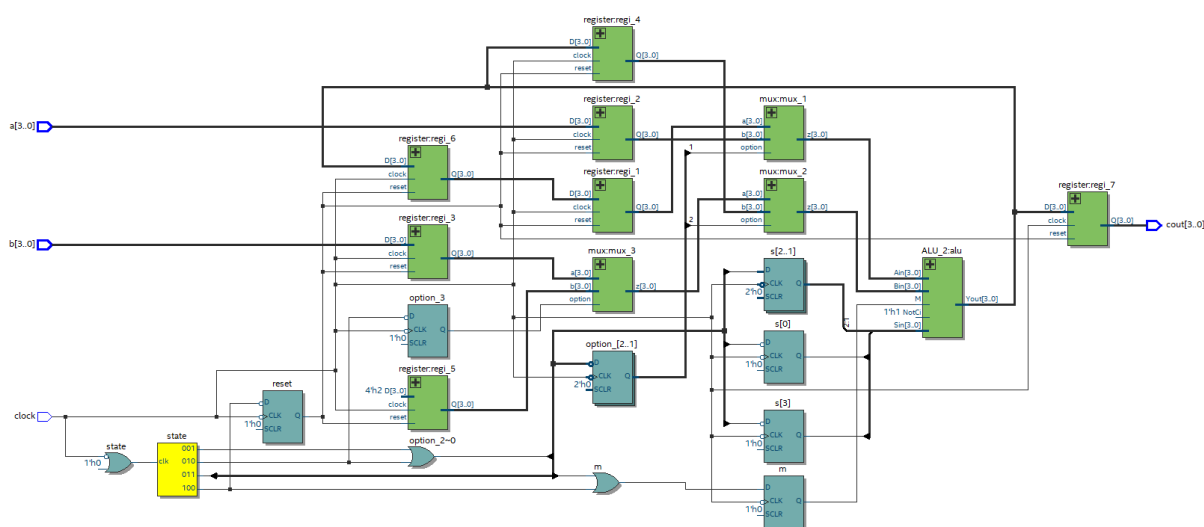
——使用 ALU 完成 $Y = (A+2) \text{ XOR } (A+B)$

学号：13013013013 姓名：朕是太上皇

一 • 设计思路

由于本次pj要求只是用一个ALU模块,而要分别完成 $A+2$, $A+B$, XOR 三个操作,因此一个时钟周期是无法完成的。故而应采用寄存器分别存储操作对象和结果,以类似“排队”的方式先后进过ALU。同时采用状态机控制每次进入ALU的对象,在若干时钟周期后完成该操作。

下面是代码综合得到的电路图:



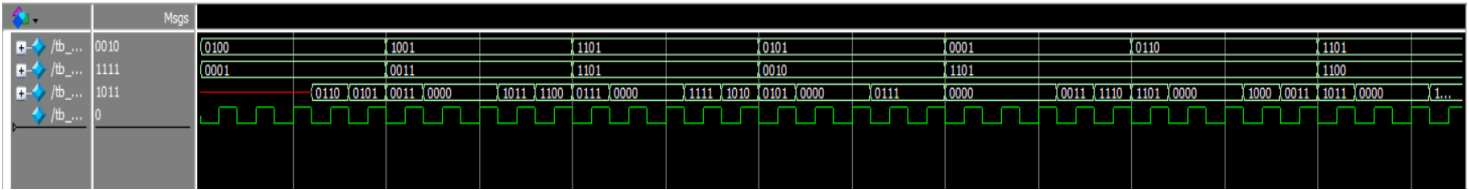
较为重要元件的功能:

mux_1: 用来选择 a 或是操作完成的 $(a+2)$ 进入 ALU;

mux_2 及 mux_3: 选择 b , $4'h2$, 操作得到的 $a+b$ 中一个进入 ALU;

regi_7 : 存储输出的 y ; state: 状态机, 控制各参数的取值;

二 • 仿真波形图



如图所示，上次操作完成后，顺序输出 $a+2$, $a+b$, $(a+2) \text{ xor } (a+b)$ 的结果，然后清 0；

三 • 综合结果（硬件开销与时钟频率）

通过使用 quartus prime 应用，可得到该电路运行在 50ns 的周期当中。以下为硬件开销：

	Resource	Usage
1	Estimate of Logic utilization (ALMs needed)	20
2		
3	▼ Combinational ALUT usage for logic	19
1	-- 7 input functions	3
2	-- 6 input functions	4
3	-- 5 input functions	4
4	-- 4 input functions	1
5	-- <=3 input functions	7
4		
5	Dedicated logic registers	27
6		
7	I/O pins	13
8		
9	Total DSP Blocks	0
10		
11	Maximum fan-out node	clock~input
12	Maximum fan-out	27
13	Total fan-out	167
14	Average fan-out	2.32

四 • 时序约束条件（sdc）

另附 sdc 文件。

//本次 project 感谢社会人士“古今第一龌龊人”的贴心帮助。