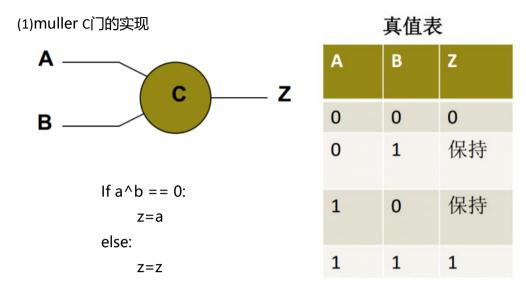
微流水线-实验报告(张里蒙)修改后

2019年5月17日 21:08





(2)用C单元实现两段握手

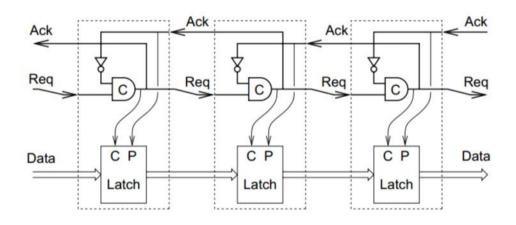
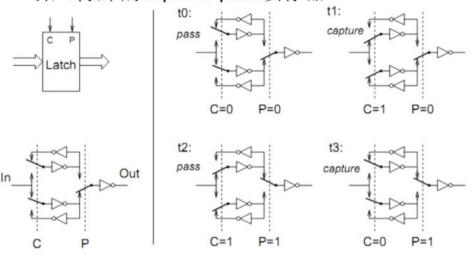


Figure 2.10. A simple 2-phase bundled-data pipeline.

(3)实现CP-latch

• 引入特殊的capture-pass锁存器



₩ Wave - Default ::::::::::::::::::::::::::::::::::::										
♦ 1 •	Msgs									
<pre>/cp_latch_tb/c</pre>	1h1									
<pre>/cp_latch_tb/p</pre>	1'h1									
/cp_latch_tb/rst_n	1h1	$\overline{}$								
- cp_latch_tb/data_in			3'h1		3'h0		3h1		3'h0	
/cp_latch_tb/data_out	3h1	3'h0	3'h1						3'h0	

(4)实现一个模块block, testbench中ack在req给出后2ns延迟响应;

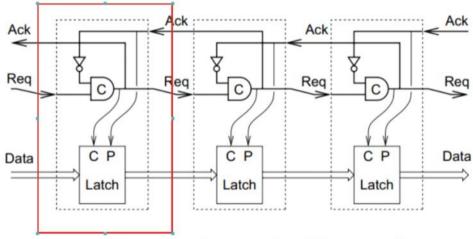
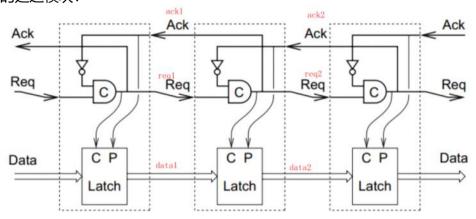


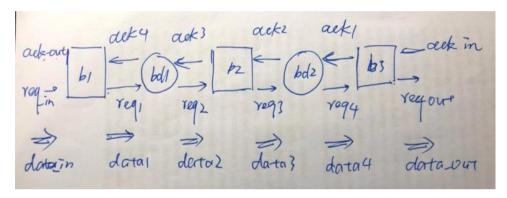
Figure 2.10. A simple 2-phase bundled-data pipeline.

<u>u</u> -	Msgs					
<pre>/block_tb/ack_in</pre>	1h1					
/block_tb/req_in	1'h0		_			
<pre>/block_tb/rst_n</pre>	1h1					
/block_tb/req_out	1'h0					
/block_tb/ack_out	1'h0					
+	3'h3		3h1)3 h 2	(3°h3	
+	3'h3	(31h0	3h1		3h3	

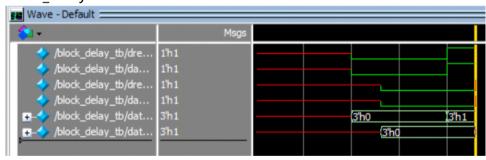
Ack_in	Req_in	С	Р	CP-latch	Data_In	Data_out
0	1	1	0	capture	1	1
0	1	1	0	capture	2	1
1	1	1	1	pass	3	3

(5)实现3个block的流水线(block_combine),每个block之间添加了3ns的延迟模块:





Block_delay:



实现了3ns延迟。



如上图所示,输入为12345时,输出端为12345,实现了FIFO功能

(6)流水线的右端没有不给应答信号,流水线充满后的状态



如图,流水线一直不给应答信号,流水线中只有数据1.

(7) 当testbench的req,ack采用两步握手的时候,如下图



(8) 分析:

吞吐率: 50%

流水线延迟: 3*3ns+2ns=11ns

利用率: 50%