EGR 234 – Digital Logic Design Lab 7: Modular Approach for Adder Implementation

Lab Report by: Christopher Parisi Lab Partner: No Partner October 15, 2010

Exercise 1

```
VHDL code for 1-bit full adder
-- 1-bit full adder
-----
library ieee;
use ieee.std logic 1164.all;
entity fa is
port (x, y, cin : in std logic;
            s, cout : out std logic);
end fa;
architecture fa arch of fa is
      signal input : std logic vector (2 downto 0);
      signal output : std logic vector (1 downto 0);
begin
      input <= cin & x & y;
with input select
      output <= "00" when "000",
                    "10" when "001",
                    "10" when "010",
                    "01" when "011",
                    "10" when "100",
                    "01" when "101",
                    "01" when "110",
                    "11" when "111",
                    "00" when others;
      s \leq output(1);
      cout <= output(0);</pre>
end fa arch;
```

Functional Simulation Waveform

D3	Master	Time Bar:		0 ps	Pointer:	513 ps	Interval:	513 ps	Start:	End	:[
A ⊛		Name	Val <mark>ue at</mark> 0 ps	0 ps 0 ps J	10.0 ns	20.0 ns	30.0 ns	40.0 ns	50.0 ns	60.0 ns	70.0 ns	80.0 n
_		cin x	B 0 B 0									
44	2 3	y s	BOBO	3		_						
₩.,	@4	cout	BO	1								

Timing Simulation Waveform

5	Master 1	Time Bar:		0 ps	Pointer:	10.87 ns	Interval:	10.87 ns	Start:	End		
1			Value at	0 ps	10.0 ns	20.0 ns	30.0 ns	40.0 ns	50.0 ns	60.0 ns	70.0 ns	0.08
•		Name	0 ps	0 ps								
<u> </u>	@>0	cin	BO									
	⊡ •1	x	BO									
-[₽ 2	У	BO									
۱		s	BO									
+ [@4	cout	BO									

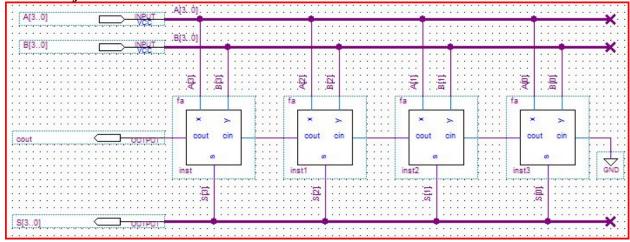
Truth Table

cin	Х	у	S	cout
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

This truth table coincides with the waveform and shows the functionality of the 1-bit adder.

Exercise 2

Schematic for 4-bit adder



Maste	r Time Bar:	0 ps	Pointer:	1.74 ns	Interval:	1.74 ns	Start:	Er	nd:
		0 ps	10.0 ns	20.0 ns	30.0 ns	40.0 ns	50.0 ns	60.0 ns	70.0 ns 80.0
	Name	0 ps							
P 0	E A	1	χ 3	χ 5	χ 7	X	9 Х	в Х	D X F
1	— A[3]	1		-					
₽2									
■ 3	— A[1]			1					
_	A[0]								
₽5	B	0	χ2	X4	X6	X	8 X	<u>AX</u>	с <u>х</u> е
6									
7									
8									
9		<u> </u>	Y 5	X 9	Ý		1	5 Y	9 X D
@10 @11			~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~		L	^		° /	3 1 0
@ 12									
13	-S[1]								
@ 14									
@ 15									

Function Simulation Waveform

Exercise 3

Schematic for completed adder

	dder4			⊃ cout	T
A[30]	A[30] cout	hex7seg			PIN_AE23
A[3.0]	B[30] S[30]	hex[30] seg[60]		⊃ seg[80]	<u></u>
PIN_C13 PIN_AE14					PIN_V13 PIN_V14
PIN_AC13 PIN_P25 iiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiii	nst8	inst			PIN_AE11
PIN_AF14 PIN_N25			J		PIN_AD11 PIN_AC12
					PIN_AB12
					PIN_AF10
					enter interior

This schematic was then programmed into the DE2 board and tested. The results match up with the waveform from Exercise 2. This shows that the Modular Approach is capable of solving many problems. The modular approach is easy and organizes the problem making it easier to read and implement.