CHAPTER 4

RESULTS AND DISCUSSION

As results, after completed the analysis of the Wallace Tree multiplier, it has been proven that the pipelining method could increased the speed of the multiplier. However, the area of the multiplier designed is also increased because of the 4-stages D flip-flop that has been implemented in the design. Basically, the objective of this project is mainly focus on the speed of the Wallace Tree multiplier, so, the increased area of the multiplier does not take into count. Otherwise, the high speed 8-bits x 8-bits Wallace Tree multiplier design in this project was to be implementing on the field-programmable gate array (FPGA) Altera UP2 board, not to be fabricating on a chip. Figure 4.1 shows part of the source code for the high speed 8-bits x 8-bits Wallace Tree multiplier does not the speed 8-bits x 8-bits Wallace Tree multiplier does not be high speed 8-bits x 8-bits Wallace Tree for the source code for the high speed 8-bits x 8-bits Wallace Tree multiplier does not the speed 8-bits x 8-bits Wallace Tree for the source code for the high speed 8-bits x 8-bits Wallace Tree multiplier does not the speed 8-bits x 8-bits Wallace Tree for the source code for the high speed 8-bits x 8-bits Wallace Tree multiplier does not the speed 8-bits x 8-bits Wallace Tree multiplier does not the speed 8-bits x 8-bits Wallace Tree multiplier does not the high speed 8-bits x 8-bits Wallace Tree multiplier does not the high speed 8-bits x 8-bits Wallace Tree multiplier does not the high speed 8-bits x 8-bits Wallace Tree multiplier does not the high speed 8-bits x 8-bits Wallace Tree multiplier does not the high speed 8-bits x 8-bits Wallace Tree multiplier does not the high speed 8-bits x 8-bits Wallace Tree multiplier does not the high speed 8-bits x 8-bits Wallace Tree multiplier does not the high speed 8-bits x 8-bits Wallace Tree multiplier does not the high speed 8-bits x 8-bits Wallace Tree multiplier does not the high speed 8-bits x 8-bits Wallace Tree multiplier does not the high speed 8-bits x 8-bits Wallace Tree multiplier

The completed source code for both designs, conventional and pipelining highspeed 8-bits x 8-bits Wallace Tree multiplier included the sub-circuits are shown in the Appendices.

// Define an 8-bits x 8-bits Wallace Tree multiplier module wallace_t_pl(z, clk, set, rst, x, y);
<pre>// I/O port declarations output [16:0] z; input [7:0] x, y; input clk, set, rst;</pre>
<pre>// Internal nets reg [15:1] sh, ch; reg [49:1] sf, cf; reg [16:1] w;</pre>
// Instantiate an 8-bits x 8-bits Wallace Tree multiplier
<pre>d_ff d1 (w[1], clk, x[0], set, rst); d_ff d2 (w[2], clk, x[1], set, rst); d_ff d3 (w[3], clk, x[2], set, rst); d_ff d4 (w[4], clk, x[3], set, rst); d_ff d5 (w[5], clk, x[4], set, rst); d_ff d6 (w[6], clk, x[5], set, rst); d_ff d7 (w[7], clk, x[6], set, rst); d_ff d8 (w[8], clk, x[7], set, rst);</pre>
<pre>d_ff d9 (w[9], clk, y[0], set, rst); d_ff d10 (w[10], clk, y[1], set, rst); d_ff d11 (w[11], clk, y[2], set, rst); d_ff d12 (w[12], clk, y[3], set, rst); d_ff d13 (w[13], clk, y[4], set, rst); d_ff d14 (w[14], clk, y[5], set, rst); d_ff d15 (w[15], clk, y[6], set, rst); d_ff d16 (w[16], clk, y[7], set, rst);</pre>
halfadd ha1 (sh[1], ch[1], (w[1] & w[10]), (w[2] & w[9])); fulladd fa1 (sf[1], cf[1], (w[1] & w[11]), (w[2] & w[10]), (w[3] & w[9])); fulladd fa2 (sf[2], cf[2], (w[2] & w[11]), (w[3] & w[10]), (w[4] & w[9])); fulladd fa3 (sf[3], cf[3], (w[3] & w[11]), (w[4] & w[10]), (w[5] & w[9])); fulladd fa4 (sf[4], cf[4], (w[4] & w[11]), (w[5] & w[10]), (w[6] & w[9])); fulladd fa5 (sf[5], cf[5], (w[5] & w[11]), (w[6] & w[10]), (w[7] & w[9])); fulladd fa6 (sf[6], cf[6], (w[6] & w[11]), (w[7] & w[10]), (w[8] & w[9]));
· · · · · · · · · · ·
endmodule

Figure 4.1: Part of the source code for the high speed 8-bits x 8-bits Wallace Tree multiplier with pipelining method

The conventional circuit produces the maximum speed of 14.99 MHz or maximum delay of 66.7 nanoseconds to complete one process of 8-bits x 8-bits multiplication. However, after upgrading the conventional Wallace Tree into pipeline Wallace Tree, by adding D flip-flop stages at every input of the half adder and the full adder, the speed has improve much, increased to 54.05 MHz and the maximum delay has decreased to 18.3 nanoseconds. It was three times faster than the conventional design.

Table 4.1 show the multiplicands, multipliers and the results that were used to test and prove the functional of both design, the conventional and pipelining high speed 8-bits x 8-bits Wallace Tree multiplier.

	or the conventional and pipelin Wallace Tree multiplier	
Multiplicand	Multiplion	Multiplication Decult

Multiplicand	Multiplier	Multiplication Result				
(Hexadecimal)	(Hexadecimal)	(Hexadecimal)				
0F	FA	00EA6				
F0	FF	0EF10				
5A	F0	05460				
00	F0	00000				

Figure 4.2 show the result of the multiplication process done by the high speed 8bits x 8-bits Wallace Tree multiplier with pipelining that has been design. Figure 4.3 show the result of Timing Analyzer Tool for both of the Wallace Tree multiplier design.

	Name	Value at	0 ps	4.0 ns	8.0 ns	12.0 ns	16.0 ns	20.0 ns	24.0 ns	28.0 ns	32.0 ns	36.0 ns	40.0 ns	44.0 ns	48.0 ns	52.0 ns
	Name	Ops	0 ps													
	clk	BO														
	set	B 1														
	rst	B 1														
P	±×	H OF				0F			X				FÖ			
P	±у	H FA				FA			X				FF			
	+ z	H 00000		00000	00002	_0000E_X	00706	OOBE6 X	00CA6 X	00EA6 X	00EA4 X	04EA8 X	08430 X	0D030 X	08510 X	17810
		Value at	54.0 n	s 58.0 ns	62.0 ns	66.0 ns	70.0 ns	74.0 ns	78.0 ns	82.0 n:	s 86.0 n	s 90.0 n	s 94.0 r	ns 98.0	ns 102,0) ns 106,
	Name	54.0 ns	54.0 n	s							· · ·		· · ·			
	clk	B 1														
	set	B 1														
	rst	B 1														
	±Χ	H FO		FÜ	X			5A			X				00	
	±у	H FF		FF	X						F					
1	± z	H 0E710	(OE	710 <u>X</u> OF	F10 X	0EF10	<u> </u>	2F10 X 0	D310 X 0	6080 X ()3460 X		05460		07060 X	008E0)
	Val	ue at 5	9,0 ns		67.0 ns		75.0 ns		83. 0 ns		91.0 ns	9	39. 0 ns		107 _, 0 ns	115.0
Nam	Name 106.0 m													106	.0 ns	
(olk E	31														
\$		81														
		81														
± ;		00	FO	X			īΑ			<			00			
ر 🗉 ر		F0	FF OFF10		F10	V 00510	V 00 210	00000	00400	FO	05400	_	07000	00050	00000	00000
王 :	z Hl	10800	OFF10		F10	X 02F10	<u>0D310</u>	06080	03460	<u> </u>	05460		07060	(008E0)	00800	00000

Figure 4.2: Result of the multiplication process by the high speed 8-bits x 8-bits Wallace Tree multiplier with pipelining

🛎 Timing An	alyzer Tool				
Registered Pe	erformance tpd tsu tco th Custom Delays				
Clock: clk		•			
	Value				
From	d_ff:d3lq				
To	d_ff:d32lq				
Clock period					
Frequency	14.99 MHz				
	100 123 150 75 200 25 0 MHz 250				
	100 %				
00:00:05					
▶ Start	The Stop Report Number of paths to list: 10	List Paths			

The conventional high speed 8-bits x 8-bits Wallace Tree multiplier design

😕 Timing An	alyzer Tool 📃 🗖 🛛	×			
Registered Pe	rformance tpd tsu tco th Custom Delays				
Clock: clk	•				
	Value				
From	d_ff:d13lq				
To	fulladd_pl:fa23jd_ff:d1jq				
Clock period	18.500 ns				
Frequency	54.05 MHz				
	100 123 150 75 175 50 200 25 225 0 MHz 250				
	100 %				
00:00:15					
崎 Start	Stop Beport Number of paths to list: 10 List Paths				

The pipelining high speed 8-bits x 8-bits Wallace Tree multiplier design

Figure 4.3: The results of Timing Analyzer Tool

Finally, after completed the analysis of the Wallace Tree multiplier, it has been proven that the pipelining method could increased the speed of the multiplier.