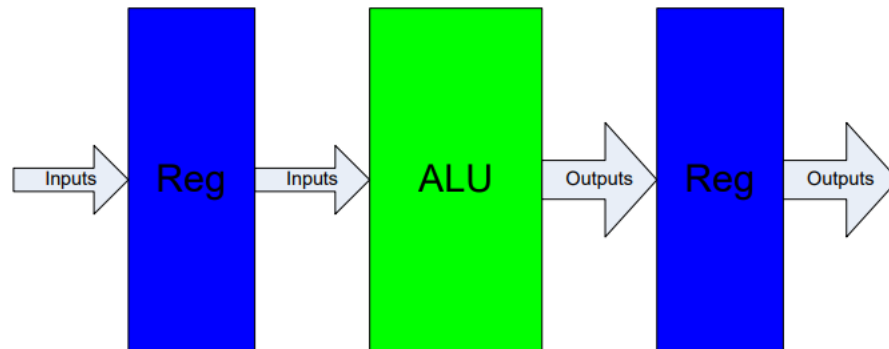


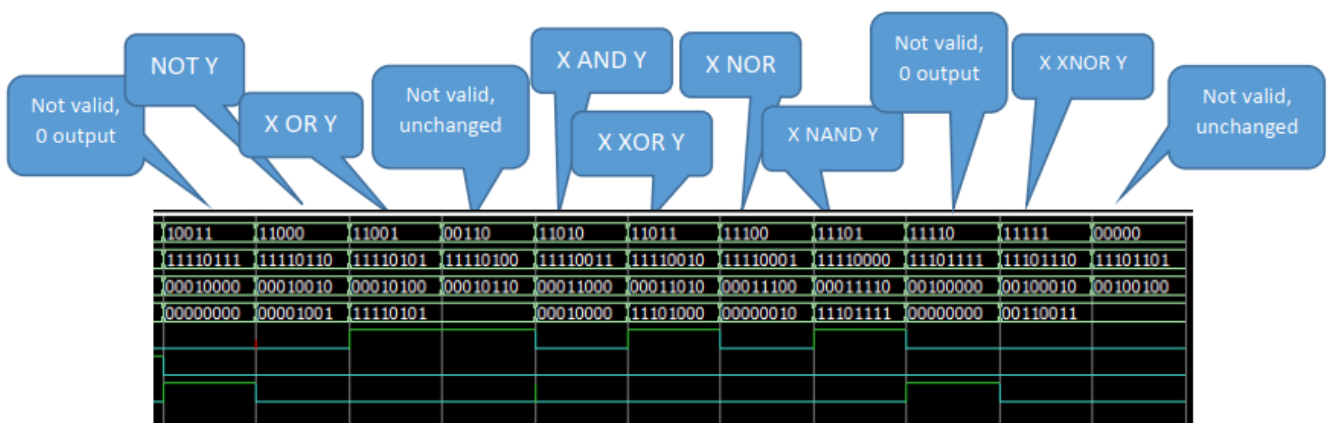
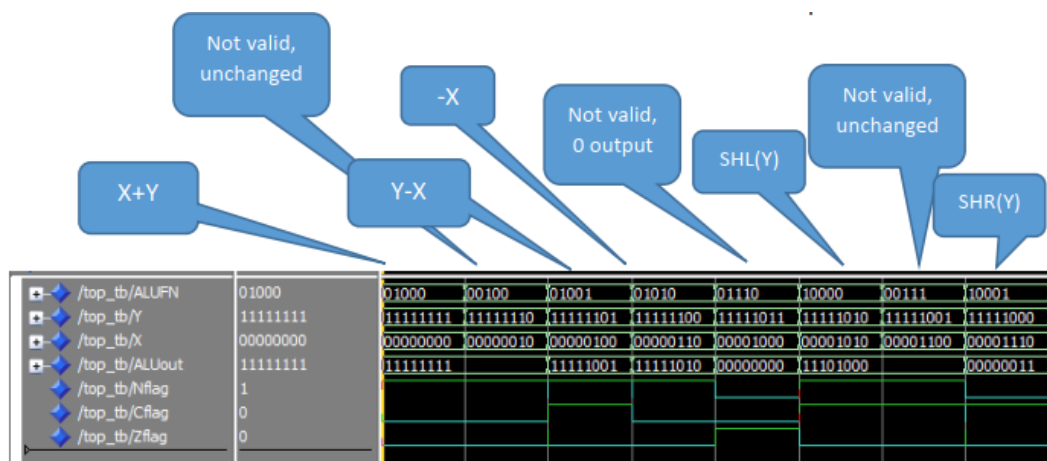
דו"ח מכין- ניסוי 4:

בניסוי זה נדרשנו לצרוב את הקוד של ניסוי מס. 1 ל-FPGA.

בשלב הראשון, לצורך חישוב Fmax ומציאת הcritical path, חיברנו את יחידת הALU לרגיסטרים באופן הבא:



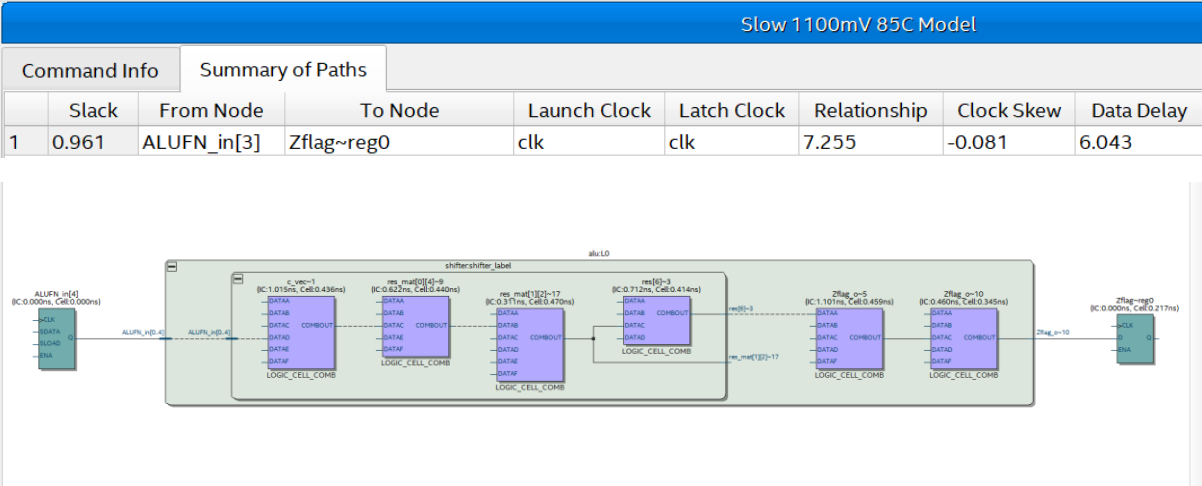
לפנינו wave במודל modedsim של testbench של הליבה ALU:



כעת נסתכל על Fmax שהתקבל:

Slow 1100mV 85C Model Fmax Summary			
<<Filter>>			
	Fmax	Restricted Fmax	Clock Name
1	137.85 MHz	137.85 MHz	clk

לפנינו המסלול הקריטי:



ניתן להסיק מהמסלול הקריטי שהאופרציה שמגבילה את הזמן בצורה הכי משמעותית החישוב של ה-Zflag לאחר היציאה מshifter, וזאת מכיוון שבshifter ישנן מספר רמות שקורות בצורה כרונולוגית, ולא מקבילית, ולאחר מכן, גם חישוב דגל ה-zero הוא ארוך משאר הדגלים האחרים. למרות זאת, היינו מצפים שהמסלול הקריטי יעבור דווקא דרך Adder\sub, מכיוון ששם מתבצעות 8 פעולות בצורה טורית. לכן התוצאה מעידה על תכנון לא אידיאלי של ה-shifter במעבדה 1.

כעת נציג RTL viewer לכל בלוק.

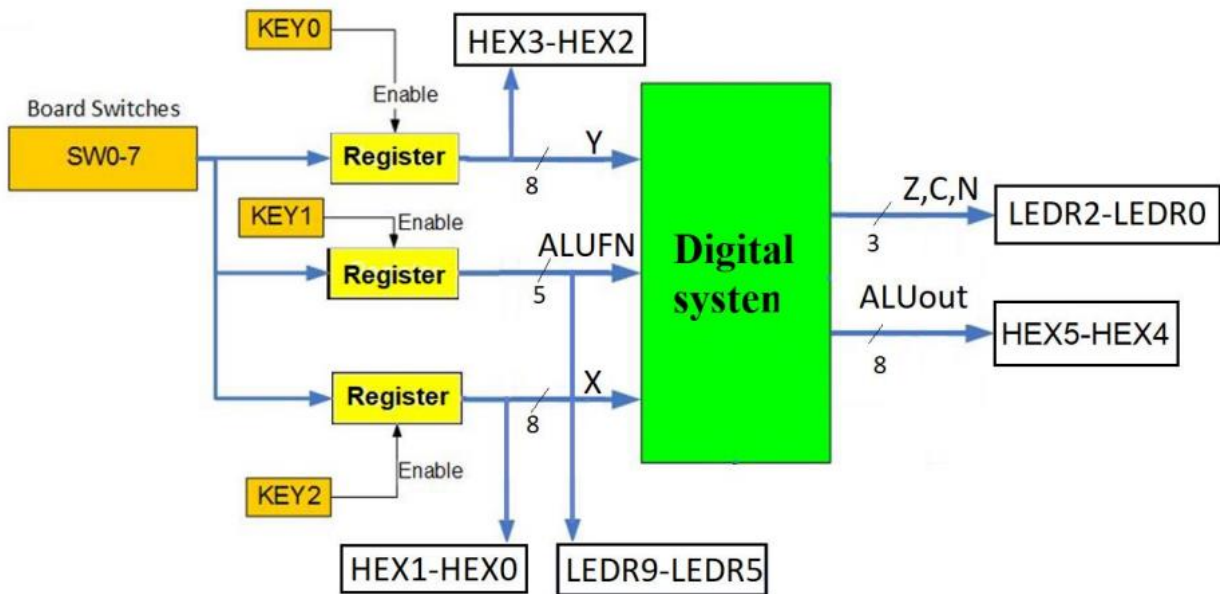
לפנינו הlogic usage report:

Flow Summary	
<<Filter>>	
Flow Status	Successful - Sun May 21 14:02:20 2023
Quartus Prime Version	21.1.0 Build 842 10/21/2021 SJ Lite Edition
Revision Name	quartusproject
Top-level Entity Name	fpga_conc
Family	Cyclone V
Device	5CSXFC6D6F31C6
Timing Models	Final
Logic utilization (in ALMs)	126 / 41,910 (< 1 %)
Total registers	21
Total pins	62 / 499 (12 %)
Total virtual pins	0
Total block memory bits	0 / 5,662,720 (0 %)
Total DSP Blocks	0 / 112 (0 %)
Total HSSI RX PCSs	0 / 9 (0 %)
Total HSSI PMA RX Deserializers	0 / 9 (0 %)
Total HSSI TX PCSs	0 / 9 (0 %)
Total HSSI PMA TX Serializers	0 / 9 (0 %)
Total PLLs	0 / 15 (0 %)
Total DLLs	0 / 4 (0 %)

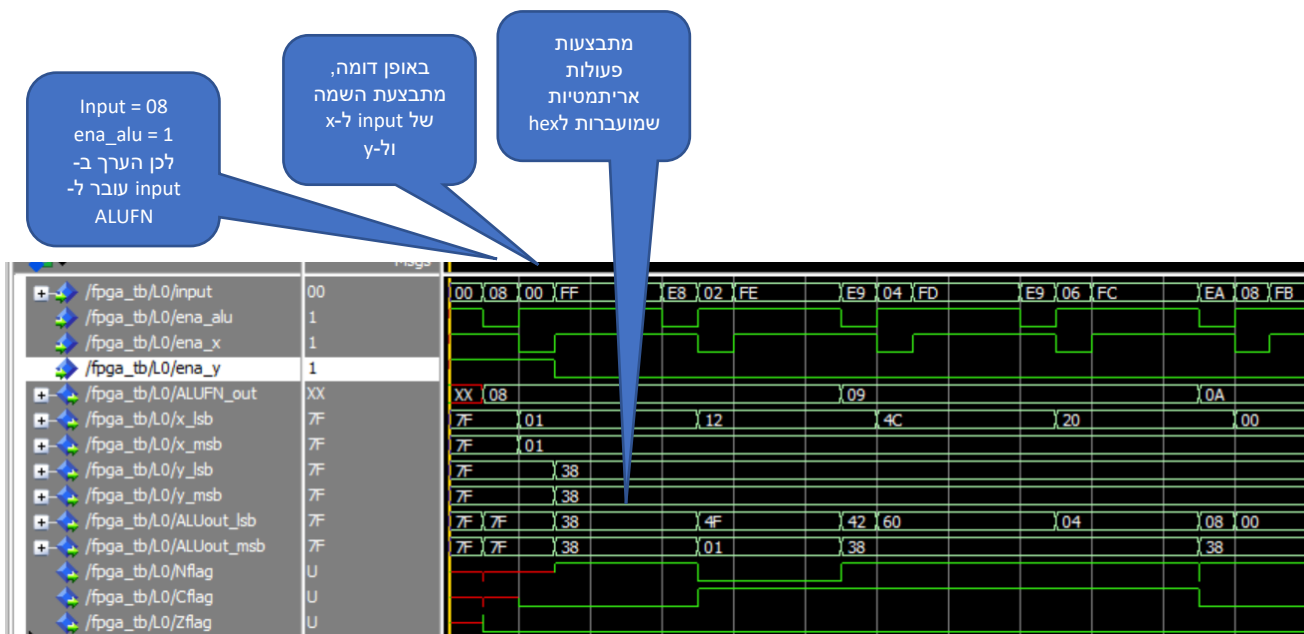
Analysis & Synthesis Resource Utilization by Entity				
<<Filter>>				
	Compilation Hierarchy Node	Combinational ALUTs	Dedicated Logic Registers	Pins
1	▼ fpga_conc	215 (42)	21 (21)	62
1	▼ alu:L0	173 (80)	0 (0)	0
1	▼ AdderSub:addersub_label	29 (11)	0 (0)	0
1	FA:\rest:1:chain	1 (1)	0 (0)	0
2	FA:\rest:2:chain	2 (2)	0 (0)	0
3	FA:\rest:3:chain	4 (4)	0 (0)	0
4	FA:\rest:4:chain	2 (2)	0 (0)	0
5	FA:\rest:5:chain	2 (2)	0 (0)	0
6	FA:\rest:6:chain	3 (3)	0 (0)	0
7	FA:\rest:7:chain	2 (2)	0 (0)	0
8	FA:first	2 (2)	0 (0)	0
2	LOGIC:logic_label	15 (15)	0 (0)	0
3	shifter:shifter_label	49 (49)	0 (0)	0

ניתן לראות שכל הרגיסטרים יושבים בשכבה המחוברת בין ה-alu לבקר. זאת מכיוון שהalu הוא א-סינכרוני ולכן לא מכיל רגיסטרים.

לאחר השלב המקדים, חיברנו את הALU לFPGA באופן הבא:










לפני שצרבנו את הקוד לבקר, ביצענו בדיקה באמצעות testbench שיצרנו לחיבורים. לפנינו צילום של waven מאותה הבדיקה:



לאחר שלב זה, צרבנו את הקוד לבקר באמצעות Quartus. לפנינו signal tap עבור שתי פעולות שונות, אחת פעולת add והשנייה shift:

:Add .1

log: Trig @ 2023/05/19 13:17:14 (0:0)			click to insert time bar			
Type	Alias	Name	0	1024	2048	3072
		⊕ alufn[4..0]			01000b	
		⊕ y[7..0]			01h	
		⊕ x[7..0]			03h	
		⊕ alu:L0 ALUout o[7..0]			04h	
		Cflag				
		Zflag				
		Nflag				

:shift .2

log: Trig @ 2023/05/19 13:19:32 (0:0)			click to insert time bar			
Type	Alias	Name	0	1024	2048	3072
R		alufn[4..0]			10000b	
R		y[7..0]			00000100b	
R		x[7..0]			00000001b	
C		alu:L0 ALUout o[7..0]			00001000b	
*		Cflag				
*		Zflag				
*		Nflag				