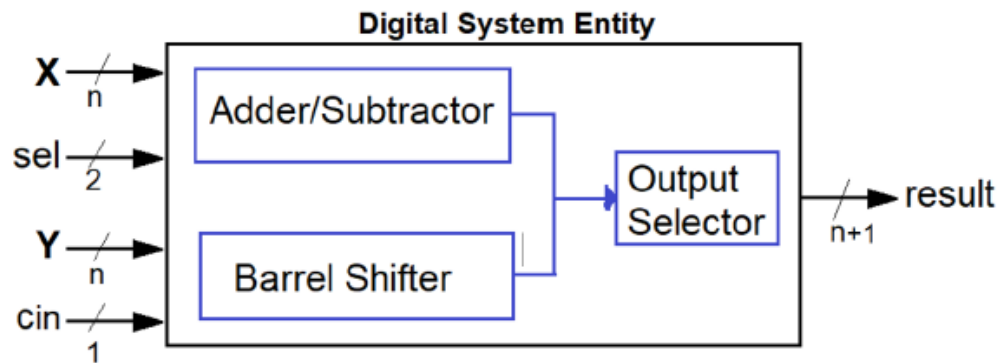


TOP:

בקובץ הזה נמצא ENTITY בשם top שמקבל 2 וקטורים בגודל N, וקטור מצב פעולה בגודל 2 ביט, ביט בשם CIN ומוציא תוצאה לפי מצב פעולה וקלט שהוא קיבל. המודול מכיל תת מודולים הנמצאים ב AdderSub.vhd Barrel.vhd,selector.vhd. בעצם מודול ראשי שמנהל את כל הלוגיקה.

איור המודל:



דיאגרמה נקיה:

[illegible]

דיאגרמה עם ההסבר:

[illegible]

דיאגרמת ה-LIST שנוכל לראות את טבלת אמת של WAVE:

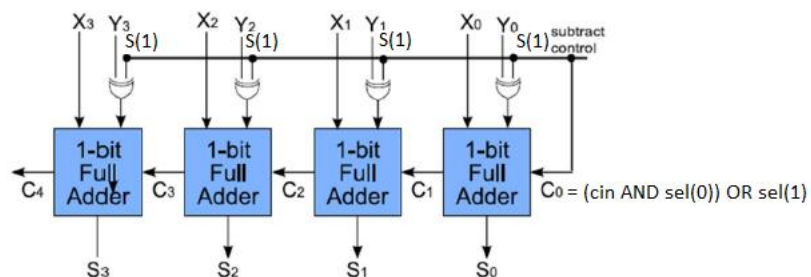
ps	/top_tb/cin	/top_tb/X	/top_tb/result
delta	/top_tb/sel	/top_tb/Y	
0 +0	0	00000000	00000000
0 +1	1	00000001	00000001
0 +4	1	00000001	00000001
0 +5	1	00000001	02222222
0 +11	1	00000001	00000001
5000000 +1	1	00000001	00000001
5000000 +11	1	00000001	00000010
10000000 +1	1	00000001	00000010
10000000 +9	1	00000001	00000100
10000000 +12	1	00000001	00000100
15000000 +1	1	00000001	00000100
15000000 +12	1	00000001	00000100
20000000 +1	0	11111111	00000100
20000000 +6	0	11111111	10000100
20000000 +7	0	11111111	11000000
20000000 +10	0	11111111	10010000
20000000 +12	0	11111111	11110000
25000000 +1	1	11111111	11110000
25000000 +12	1	11111111	11110000
30000000 +1	0	11111111	11110000
30000000 +10	0	11111111	11000000
30000000 +13	0	11111111	11100000
35000000 +1	1	11111111	11100000
35000000 +13	1	11111111	11000000
40000000 +1	1	00000001	00111111
40000000 +2	1	00000001	00111111
40000000 +3	1	00000001	01111111
40000000 +4	1	00000001	00111111
40000000 +5	1	00000001	00011100
40000000 +6	1	00000001	00011000
40000000 +7	1	00000001	00010000
40000000 +8	1	00000001	00000000
40000000 +9	1	00000001	00100000
50000000 +1	1	01000000	00100000
50000000 +4	1	01000000	00100001
60000000 +1	1	11111001	00000001
60000000 +3	1	11111001	01011101
60000000 +4	1	11111001	11111100
65000000 +1	1	01111111	00000001
65000000 +3	1	01111111	10111110
65000000 +4	1	01111111	00111110

Adder/Subtractor:

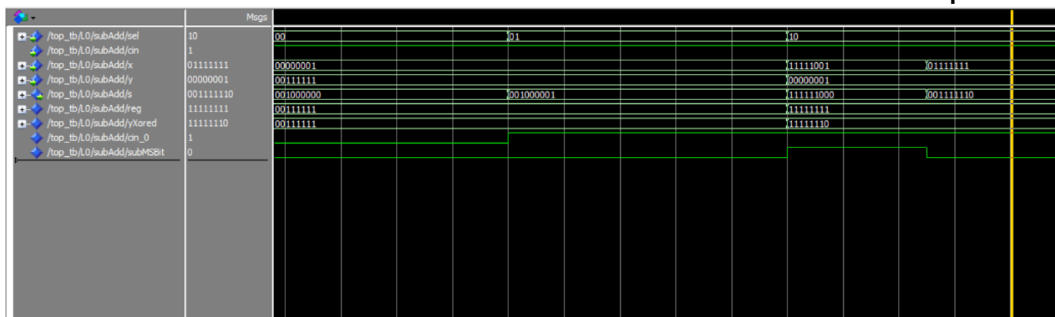
AdderTwo מקבל 2 וקטורים בגודל N, ביט בשם CIN ווקטור בגודל 2 ביט של מצב הפעולה של המודול הספיציפי הזה ומוציא וקטור של 1+N ביט של חישוב הנקבע על פי ווקטור הפעולה. מודול מבטיח פעולה תקינה רק עבור מצבי פעולה תקינים "01" "10" "10" כפי שהוגדר בקובץ הדרישות. זאת אומרת שאם מצב פעולה שווה ל"11" אנו לא מבטיחים פעולה תקינה מכיוון שזה לא מצב תקין עבור מודול הזה כפי שהוגדר בתרגיל אף מודול לא קורס. מודול הנוכחי משתמש בתת מודל בנמצא בקובץ FA.vhd.

איור המודול:

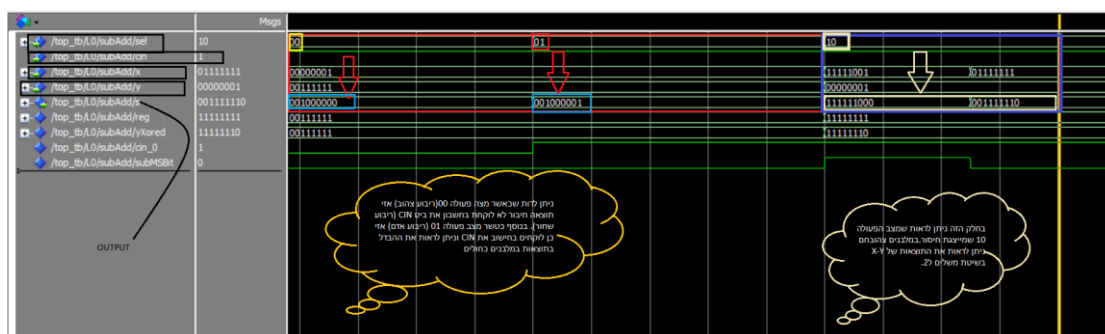
דוגמה עבור כניסה של 4 ביט (נבנה באופן גנרי)



דיאגרמה נקיה:



דיאגרמה עם ההסבר:



דיאגרמת ה-LIST שנוכל לראות את טבלת אמת של ה-WAVE:

ps	/top_tb/L0/subAdd/cin	/top_tb/L0/subAdd/s	/top_tb/L0/subAdd/reg
delta	/top_tb/L0/subAdd/x	/top_tb/L0/subAdd/y	/top_tb/L0/subAdd/yXored
	/top_tb/L0/subAdd/sel	/top_tb/L0/subAdd/cin_0	/top_tb/L0/subAdd/subMSBit
0 +0	0	00000000	00000000
0 +1	1	00000001	00000001
0 +2	1	00000001	00000001
0 +3	1	00000001	00000001
0 +4	1	00000001	00000001
0 +5	1	00000001	00000001
0 +6	1	00000001	00000001
5000000 +1	1	00000001	00000001
5000000 +2	1	00000001	00000001
5000000 +3	1	00000001	00000001
10000000 +1	1	00000001	00000001
10000000 +2	1	00000001	00000001
10000000 +3	1	00000001	00000001
10000000 +4	1	00000001	00000001
10000000 +5	1	00000001	00000001
15000000 +1	1	00000001	00000001
15000000 +2	1	00000001	00000001
15000000 +3	1	00000001	00000001
20000000 +1	0	11111111	00000000
20000000 +2	0	11111111	00000000
20000000 +3	0	11111111	00000000
20000000 +4	0	11111111	00000000
20000000 +5	0	11111111	00000000
20000000 +6	0	11111111	00000000
20000000 +7	0	11111111	00000000
20000000 +8	0	11111111	00000000
20000000 +9	0	11111111	00000000
25000000 +1	1	11111111	00000001
25000000 +2	1	11111111	00000001
25000000 +3	1	11111111	00000001
30000000 +1	0	11111111	00000000
30000000 +2	0	11111111	00000000
30000000 +3	0	11111111	00000000
35000000 +1	1	11111111	00000001
35000000 +2	1	11111111	00000001
35000000 +3	1	11111111	00000001
40000000 +1	1	00000001	00000000
40000000 +2	1	00000001	00000000
40000000 +3	1	00000001	00000000
40000000 +4	1	00000001	00000000
40000000 +5	1	00000001	00000000
40000000 +6	1	00000001	00000000
40000000 +7	1	00000001	00000000
40000000 +8	1	00000001	00000000
50000000 +1	1	00000001	00000000
50000000 +2	1	00000001	00000000
50000000 +3	1	00000001	00000000
60000000 +1	1	11111101	00000001
60000000 +2	1	11111101	00000001
60000000 +3	1	11111101	00000001
65000000 +1	1	01111111	00000001
65000000 +2	1	01111111	00000001
65000000 +3	1	01111111	00000001

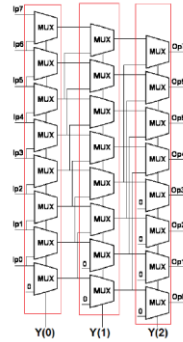
Barrel Shifter:

BARREL מקבל וקטור בגודל 8 ביט שאנו רוצים לבצע עליו פעולת הזזה ווקטור בגודל 3 ביט שבעצם אומר לנו כמה להזיז, מוציא וקטור בגודל 9 ביט כך שביט ה 9 זה CARRY.

BARREL משתמש ב2 תת מודולים הנמצאים בקבצי mux2on1.vhd וMuxCombined.vhd.

איור המודל:

כניסות בשם X וY פלט בשם Output. נבנה באופן גנרי.



דיאגרמה נקיה:

	Mags								
/top_tb/L0/BarrelEntity/x	00000001	00000001				11111111			
/top_tb/L0/BarrelEntity/y	111	000	001	010	011	100	101	110	111
/top_tb/L0/BarrelEntity/output	010000000	000000001	000000010	000000100	000001000	111110000	111100000	111000000	100000000
/top_tb/L0/BarrelEntity/mux1OutSig	00000010	000000001	000000010	000000010	000000010	111111111	111111110	111111111	111111110
/top_tb/L0/BarrelEntity/mux2OutSig	00001000	000000001	000000010	000000100	000001000	111111111	111111110	111111100	111110000
/top_tb/L0/BarrelEntity/firstReg	00000010	000000010				111111110			
/top_tb/L0/BarrelEntity/secondReg	00001000	00000100	00001000	00000100	00001000	111111100	11111000	11111100	11111000
/top_tb/L0/BarrelEntity/thirdReg	10000000	00010000	00100000	01000000	10000000	11110000	11100000	11000000	10000000
/top_tb/L0/BarrelEntity/outputSig	10000000	000000001	000000010	00000100	00001000	11110000	11100000	11000000	10000000
/top_tb/L0/BarrelEntity/coutSig	0								

דיאגרמה עם ההסבר:

	Mags								
/top_tb/L0/BarrelEntity/x	00000001	00000001				11111111			
/top_tb/L0/BarrelEntity/y	111	000	001	010	011	100	101	110	111
/top_tb/L0/BarrelEntity/output	010000000	000000001	000000010	000000100	000001000	111110000	111100000	111000000	100000000
/top_tb/L0/BarrelEntity/mux1OutSig	00000010	000000001	000000010	000000010	000000010	111111111	111111110	111111111	111111110
/top_tb/L0/BarrelEntity/mux2OutSig	00001000	000000001	000000010	000000100	000001000	111111111	111111110	111111100	111110000
/top_tb/L0/BarrelEntity/firstReg	00000010	000000010				111111110			
/top_tb/L0/BarrelEntity/secondReg	00001000	00000100	00001000	00000100	00001000	111111100	11111000	11111100	11111000
/top_tb/L0/BarrelEntity/thirdReg	10000000	00010000	00100000	01000000	10000000	11110000	11100000	11000000	10000000
/top_tb/L0/BarrelEntity/outputSig	10000000	000000001	000000010	00000100	00001000	11110000	11100000	11000000	10000000
/top_tb/L0/BarrelEntity/coutSig	0								

קצת לזאת ש Y מייצג וקטור בכמה לזוז את וקטור X ובנוסף קצת לזאת שאנו עושים טיפול בביט ה 9 כדי שמועד בתחילת (מסומן עם חץ)

דיאגרמת LIST שנוכל לראות את טבלת אמת של WAVE:

ps	/top_tb/L0/BarrelEntity/x	/top_tb/L0/BarrelEntity/firstReg	/top_tb/L0/BarrelEntity/secondReg	/top_tb/L0/BarrelEntity/thirdReg	/top_tb/L0/BarrelEntity/outputSig	/top_tb/L0/BarrelEntity/outputSig
delta	/top_tb/L0/BarrelEntity/y	/top_tb/L0/BarrelEntity/output	/top_tb/L0/BarrelEntity/output	/top_tb/L0/BarrelEntity/output	/top_tb/L0/BarrelEntity/output	/top_tb/L0/BarrelEntity/output
0	+0	00000000	00000000	00000000	00000000	00000000
0	+1	00000000	00000000	00000000	00000000	00000000
0	+2	00000000	00000000	00000000	00000000	00000000
0	+3	00000000	00000000	00000000	00000000	00000000
0	+4	00000000	00000000	00000000	00000000	00000000
0	+5	00000000	00000000	00000000	00000000	00000000
0	+6	00000000	00000000	00000000	00000000	00000000
0	+7	00000000	00000000	00000000	00000000	00000000
0	+8	00000000	00000000	00000000	00000000	00000000
0	+9	00000000	00000000	00000000	00000000	00000000
50000000	+2	00000000	00000000	00000000	00000000	00000000
50000000	+4	00000000	00000000	00000000	00000000	00000000
50000000	+5	00000000	00000000	00000000	00000000	00000000
50000000	+6	00000000	00000000	00000000	00000000	00000000
50000000	+7	00000000	00000000	00000000	00000000	00000000
50000000	+8	00000000	00000000	00000000	00000000	00000000
50000000	+9	00000000	00000000	00000000	00000000	00000000
100000000	+2	00000000	00000000	00000000	00000000	00000000
100000000	+4	00000000	00000000	00000000	00000000	00000000
100000000	+5	00000000	00000000	00000000	00000000	00000000
100000000	+6	00000000	00000000	00000000	00000000	00000000
100000000	+7	00000000	00000000	00000000	00000000	00000000
100000000	+8	00000000	00000000	00000000	00000000	00000000
100000000	+9	00000000	00000000	00000000	00000000	00000000
150000000	+2	00000000	00000000	00000000	00000000	00000000
150000000	+4	00000000	00000000	00000000	00000000	00000000
150000000	+5	00000000	00000000	00000000	00000000	00000000
150000000	+7	00000000	00000000	00000000	00000000	00000000
150000000	+8	00000000	00000000	00000000	00000000	00000000
150000000	+9	00000000	00000000	00000000	00000000	00000000
150000000	+10	00000000	00000000	00000000	00000000	00000000
200000000	+2	11111111	11111111	11111111	11111111	11111111
200000000	+3	11111111	11111111	11111111	11111111	11111111
200000000	+4	11111111	11111111	11111111	11111111	11111111
200000000	+5	11111111	11111111	11111111	11111111	11111111
200000000	+6	11111111	11111111	11111111	11111111	11111111
200000000	+7	11111111	11111111	11111111	11111111	11111111
200000000	+8	11111111	11111111	11111111	11111111	11111111
200000000	+9	11111111	11111111	11111111	11111111	11111111
200000000	+10	11111111	11111111	11111111	11111111	11111111
250000000	+2	11111111	11111111	11111111	11111111	11111111
250000000	+4	11111111	11111111	11111111	11111111	11111111
250000000	+5	11111111	11111111	11111111	11111111	11111111
250000000	+6	11111111	11111111	11111111	11111111	11111111
250000000	+7	11111111	11111111	11111111	11111111	11111111
250000000	+9	11111111	11111111	11111111	11111111	11111111
250000000	+10	11111111	11111111	11111111	11111111	11111111
300000000	+2	11111111	11111111	11111111	11111111	11111111
300000000	+4	11111111	11111111	11111111	11111111	11111111
300000000	+5	11111111	11111111	11111111	11111111	11111111
300000000	+7	11111111	11111111	11111111	11111111	11111111
300000000	+8	11111111	11111111	11111111	11111111	11111111
300000000	+10	11111111	11111111	11111111	11111111	11111111
350000000	+2	11111111	11111111	11111111	11111111	11111111
350000000	+4	11111111	11111111	11111111	11111111	11111111
350000000	+5	11111111	11111111	11111111	11111111	11111111
350000000	+7	11111111	11111111	11111111	11111111	11111111
350000000	+8	11111111	11111111	11111111	11111111	11111111

סיכום:

במהלך המעבדה הצלחנו לקבל כישורים בכתיבת קוד VHDL המכיל מבנה קוד, סוגי נתונים, מפעילים ותכונות, קוד במקביל, היררכיה עיצובית, חבילות ורכיבים. בנוסף התאמנו בהפעלת תוכנית לסימולציה בשם ModelSim. מעבר לזה הבנו במהלך התרגיל שמאוד חשוב לחלק מודול מסובך לתת מודולים בצורה לוגית טובה ושאכן חשוב לוודא שכל תת המודול לפני שמרכיבים אותם ביחד. הכירות עם התוכניות חדשות ושפה חדשה הייתה מאתגרת ואף הצלחנו להתמודד איתם אך אנו מצפים שבמעבדות הבאות נכיר את MODELSIM בצורה יותר מיטבית מפני שהרגשנו שהכרנו חלק מאוד קטן מיכולות שלו.