

## ספרתיות סימולציה 1

326286705	מוחמד זנגריה
212269294	איימן עואד

## 2.1

נרשום את כל הערכים של  $d1d0s$  בטבלת האמת ועמודת הפלט ( $cout$ ) ואז נרשום אותם אותם למפת קרנו.

D1	D0	S	COUT
0	0	0	0
0	0	1	0
1	0	0	0
1	0	1	1
0	1	0	1
0	1	1	0
1	1	0	1
1	1	1	1

מפת קרנו:

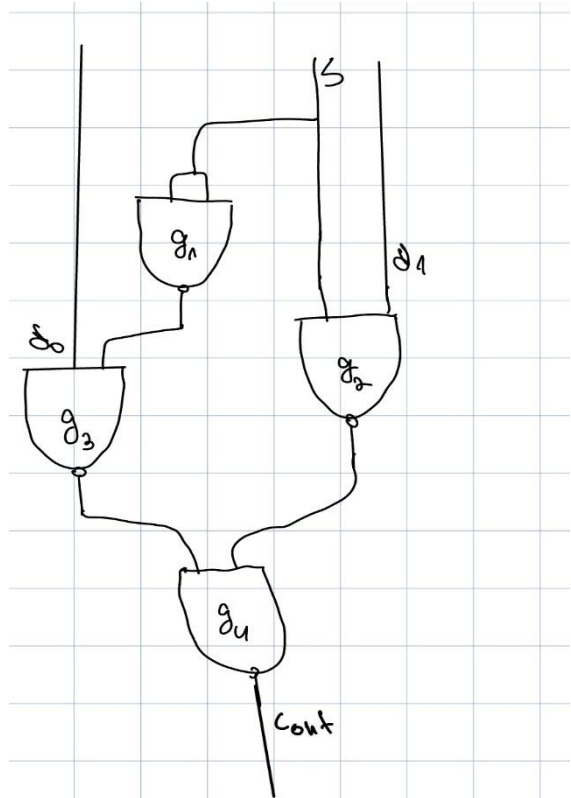
$S/d1d0$	00	01	11	10
0		1	1	
1			1	1

$$f(d1, d0, s) = d0s' + d1s$$

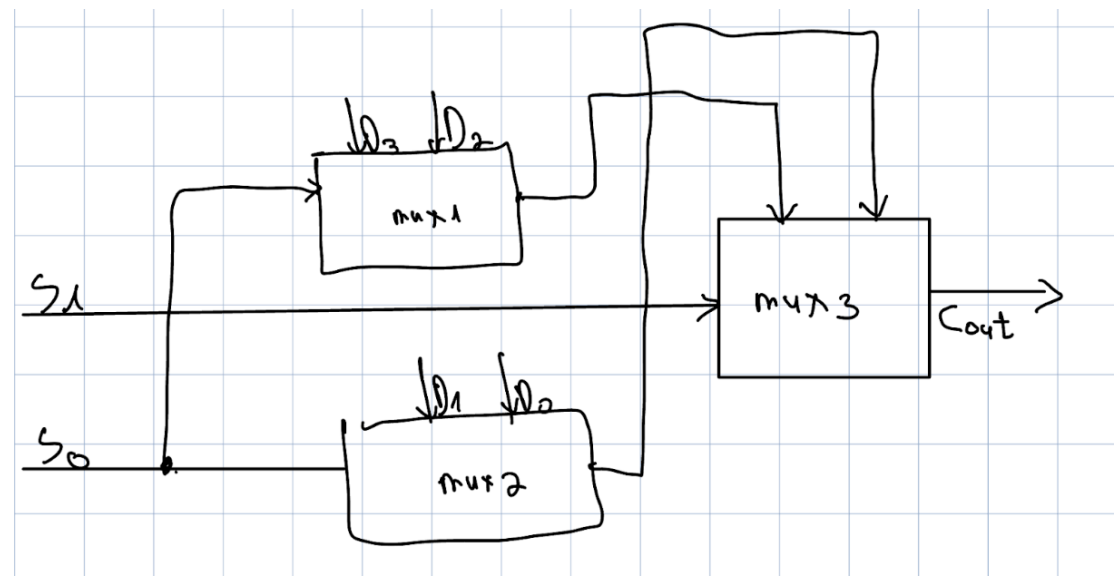
לאחר הפעלת דה מורגן נקבל :

$$f(d1, d0, s) = ((d0s')'(d1s'))'$$

PATH	D0	D1	S	Tpd
D0g3g4	0a1	0	0	8+2=10
D0g3g4	1a0	0	0	8+2=10
D1g2g4	0	0a1	0	8+2=10
D1g2g4	0	1a0	0	8+2=10
Sg2g4	0	1	0a1	8+2=10
Sg2g4	0	1	1a0	8+2=10
Sg1g3g4	1	0	0a1	2*8+2=18
Sg1g3g4	0	0	1a0	2*2+8=12



2.2

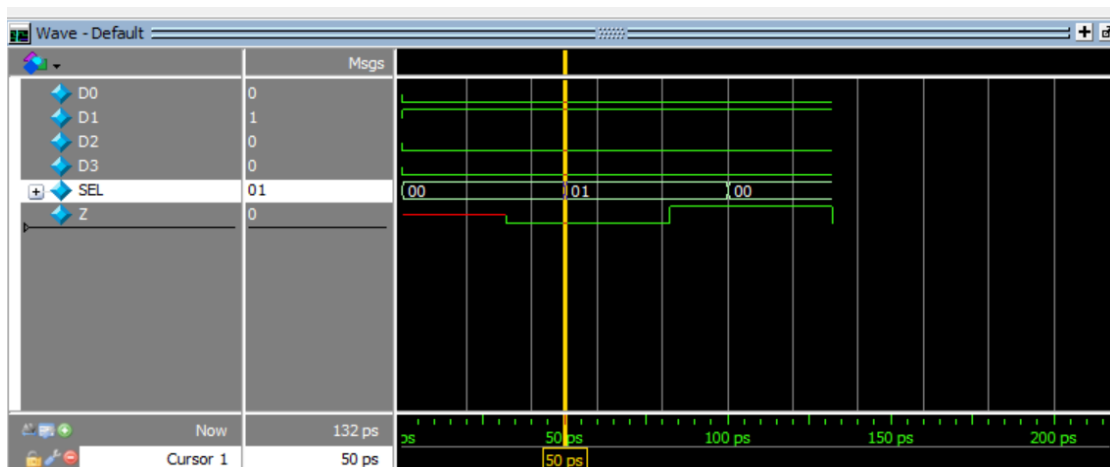
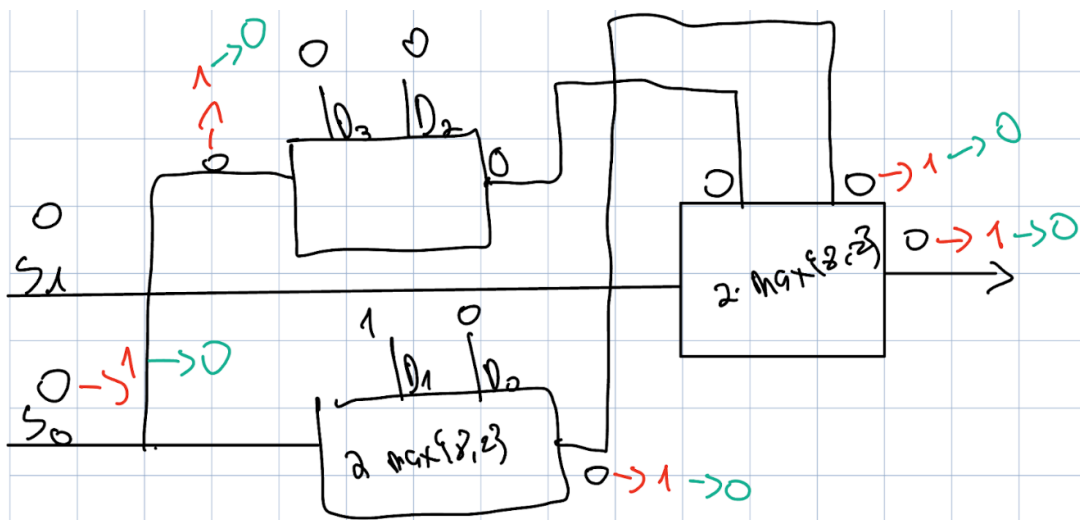


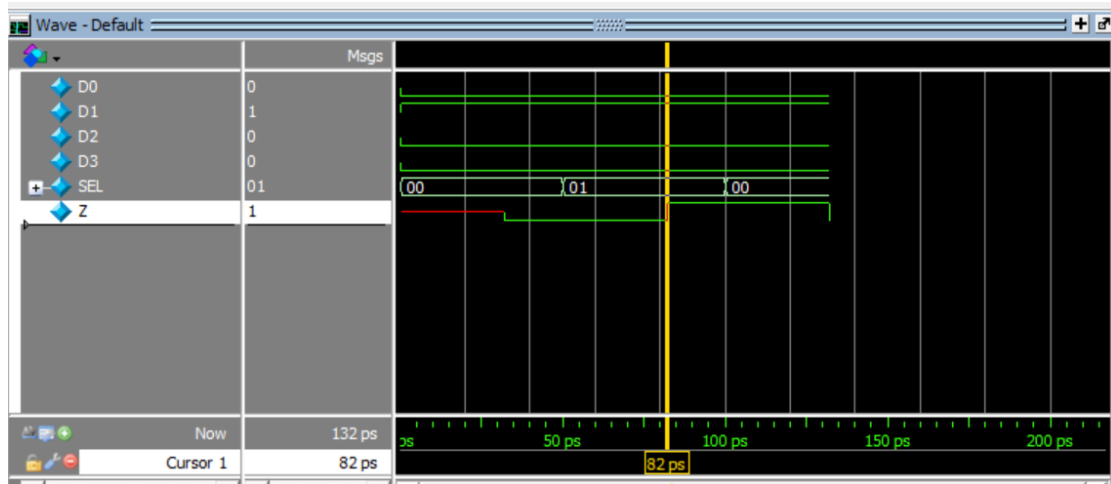
נבחר את הכניסות הבאות כך שנשנה רק את  $s_0 = 0a1a0$ ,

S0	S1	D3	D2	D1	D0
0a1	0	0	0	1	0
1a0	0	0	0	1	0

כאשר מעלים את  $s_0$  מ 0 ל 1 ומורידים אותו מ 1 ל 0 רק בורר 2 ובורר 3 משנים את היציאה שלהם ולפי סעיף קודם לפי הטבלה זמן השהייה של כל אחד מהם במקרה של כניסות אלו (בטבלה למעלה) לפי הסעיף הקודם הינו  $2 * \max\{8,2\}$  ולכן סה"כ נקבל זמן השהייה בשני המקרים הנ"ל בטבלה:  $Tpd = 2 * 2 * \max\{8,2\} = 32ns$ ,

תמונה להבנה:





וקיבלנו את הזמן שהייה הרצוי כפי שנראה בתמונות הנ"ל , לאחר  $32ns$  משינוי הכניסה  $sel[0]$  מ 0 ל 1 מתעדכן המוצא z מ 0 ל 1 כפי שחישבנו במודל התיאוריטי.

## 2.3

נשתמש בשתי מפות קרנו בשביל למצוא ביטוי לוגי עבור כל אחת מהיציאות  $SUM$  ,  $Cout$ . נקח את מספר השכנים המקסימלי בכל פעם כפי שנלמד בהרצאה והתרגול בשביל לקבל ביטוי עם מספר ליטרלים מינימלי.

Cout:

$Cin, a_{ns/a,b}$	00	01	11	10
00		1		
01				
11		1	1	1
10	1	1	1	

קיצור  $a_{ns} = n$

$$Cout = bCin + a'Cin*n' + a'bn' + aCin*n + abn$$

$$= (b + Cin)(a'n' + an) + bc$$

$$= (aXNORn)(b + Cin) + bc$$

$$= (((aXNORn)(b + Cin))'(bc)')'$$

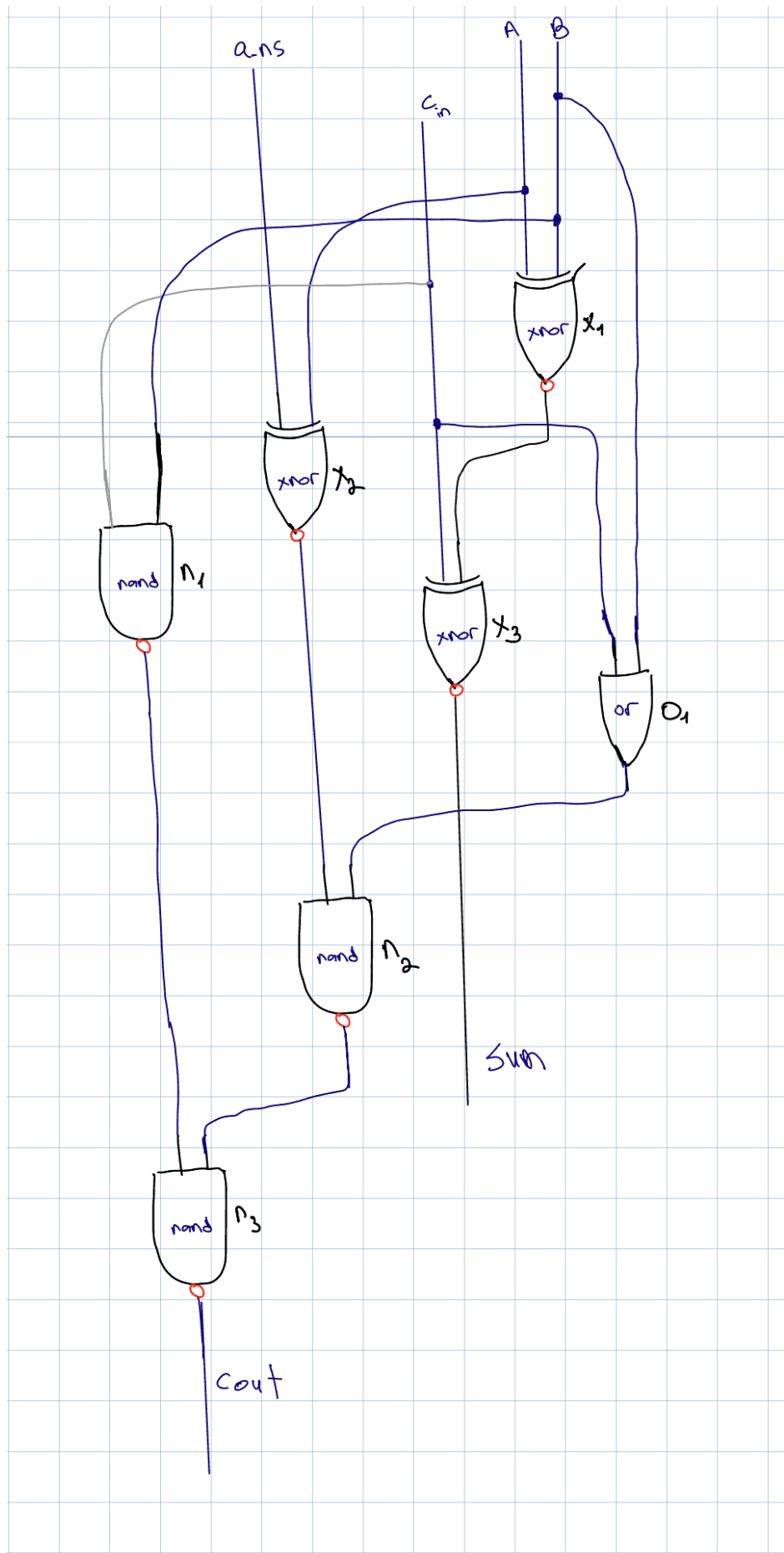
Sum:

<b><i>Cin,a_ns/a,b</i></b>	<b>00</b>	<b>01</b>	<b>11</b>	<b>10</b>
<b>00</b>		1		1
<b>01</b>		1		1
<b>11</b>	1		1	
<b>10</b>	1		1	

$$\text{Sum} = a'bCin + ab'Cin' + a'b'Cin + abCin = a'(bCin' + b'Cin) + a(b'Cin' + bCin) =$$

$$a'(bXORCin) + a(bXNORCin) = (aXNORb)XNORCin$$

מעבר אחרון :  $a'x' + ax = aXNORx$ ,  $x' = bXORcin$ ,  $x = bXNORcin$

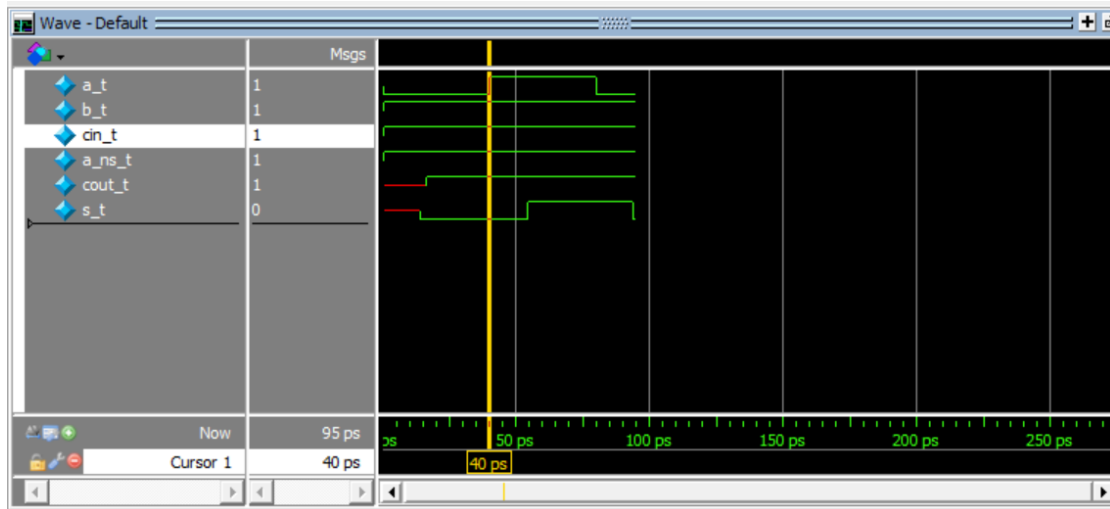


<i>path</i>	<i>A_ns</i>	<i>A</i>	<i>B</i>	<i>Cin</i>	<i>delay</i>
b-o1-n2-n3-cout	0	0	0a1	0	$Tpd(or)+Tpd(NAND)+Tpd(NAND)=22ns$
b-x1-x3-s	1	0	1a0	1	$Tpd(XNOR)+Tpd(XNOR)=12ns$
a-x2-n2-n3-Cout	1	0a1	1	0	$Tpd(xnor)+Tpd(nand)+Tpd(nand)=23ns$
A-x1-x3-s	1	0a1	1	1	$Tpd(xnor)+Tpd(xnor)=14ns$
Cin-o1-n2-n3-Cout	0	0	0	0a1	$Tpd(or)+Tpd(NAND)+Tpd(NAND)=22ns$
Cin-x3-s	1	0	0	0a1	$Tpd(xnor)=7ns$
A_ns-x2-n2-n3-Cout	0a1	1	0	0	$Tpd(xnor)+Tpd(nand)+Tpd(nand)=23ns$

נבחר במקרה הבא מהטבלה הנ"ל:

<i>A_ns</i>	<i>A</i>	<i>B</i>	<i>Cin</i>	<i>delay</i>
1	0a1	1	1	$Tpd(xnor)+Tpd(xnor)=14ns$
1	1a0	1	1	$Tpd(xnor)+Tpd(xnor)=14ns$

נראה כי התקבל הזמן שהייה הרצוי:

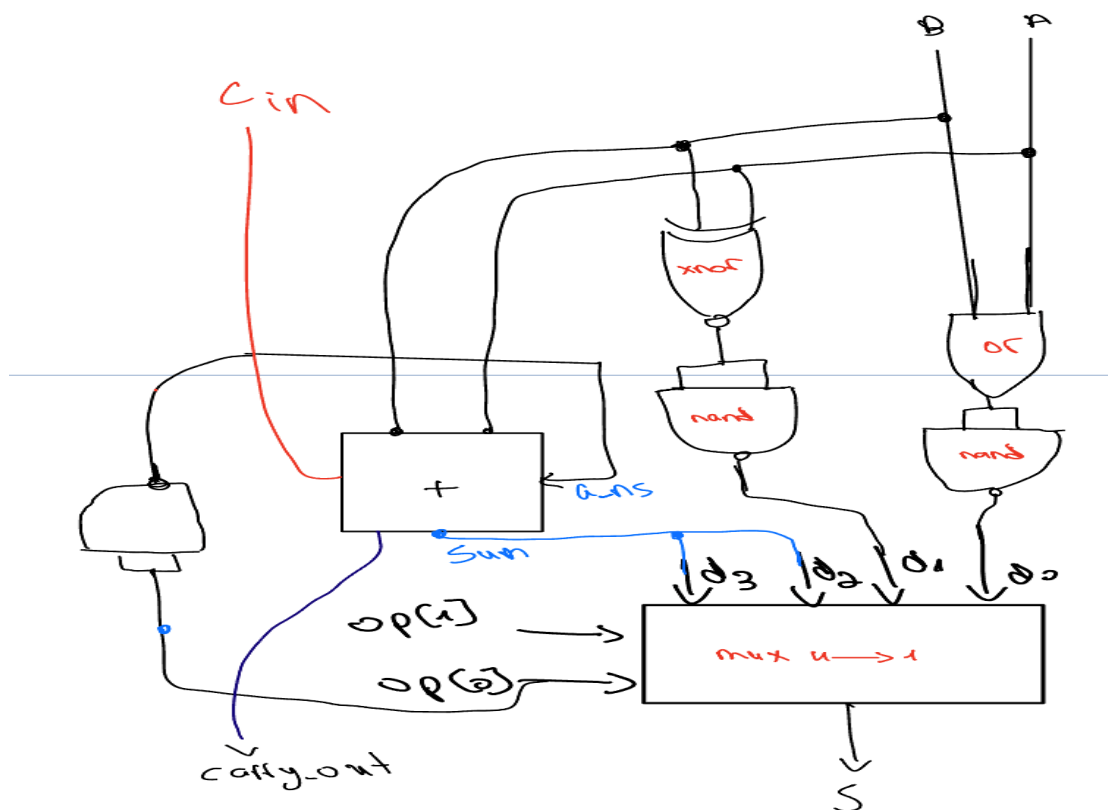


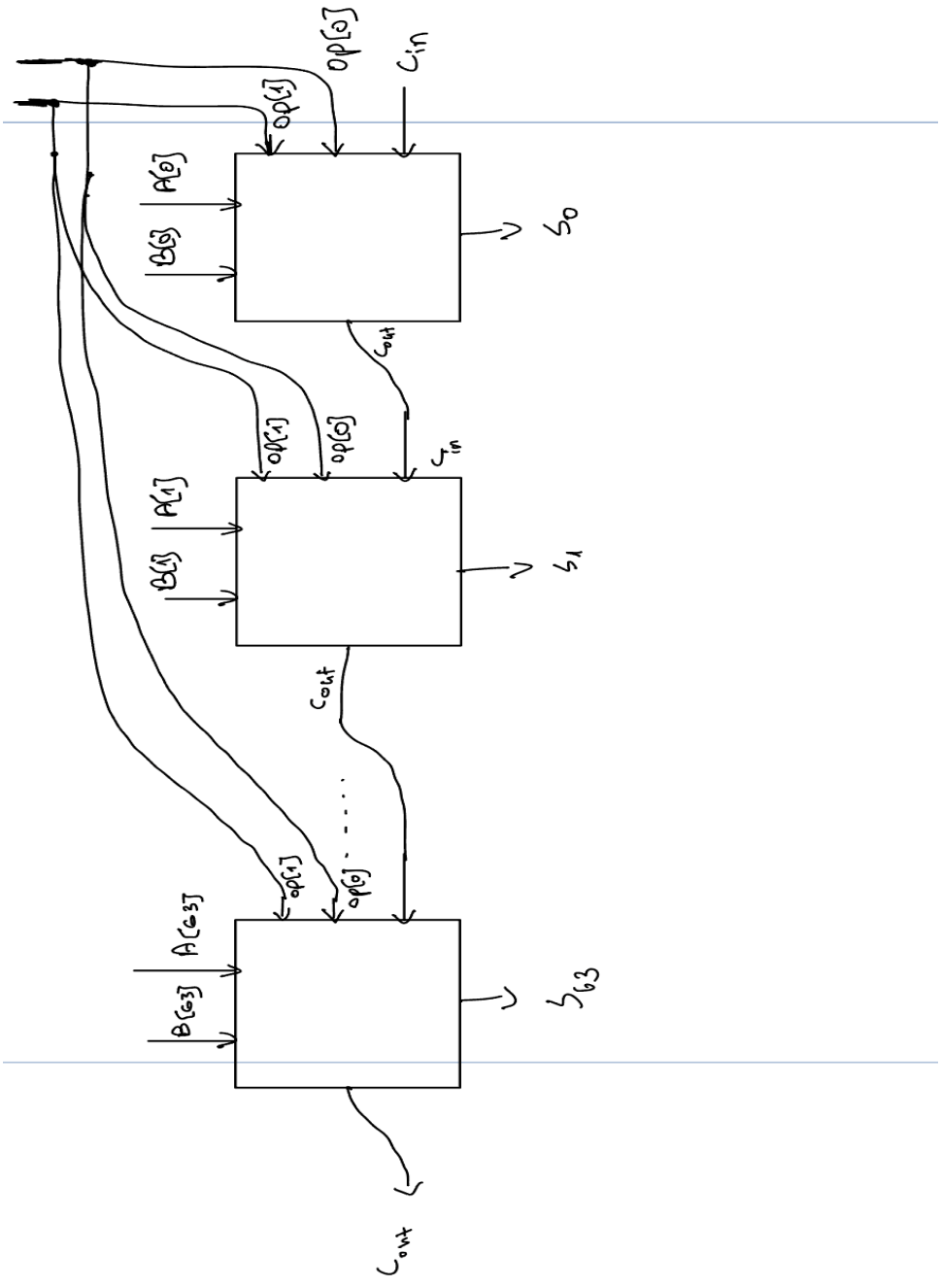




רק לאחר  $14ns$  שכיסה ש מתעדכנת מ 0 ל 1 ה  $sum$  משתנה מ 0 ל 1 כמו שחישבנו בתיאוריטי.

(2.4



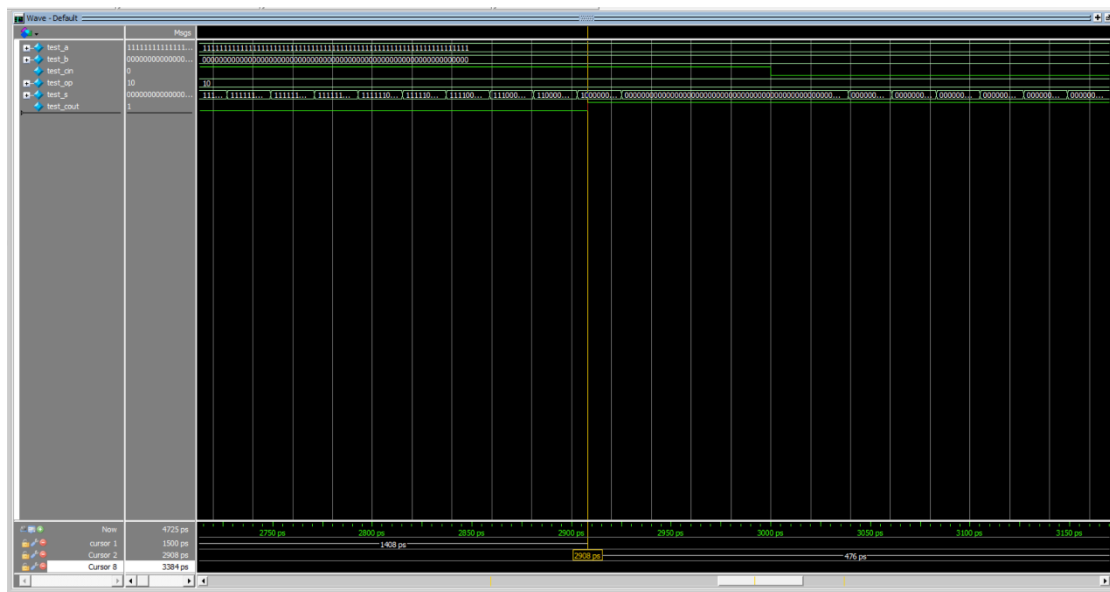


המסלול שגורם להשהיה הכי מקסימלית מכניסה ליציאה הוא מבין כל האפשרויות הוא:

א	ב	op[1:0]	cin	ב
(1,1,...,1)	(0,0,...,0)	10	0->1	$64 \cdot (Tpd\_OR + 2 \cdot Tpd\_NAND)$ $= 64 \cdot (6 + 2 \cdot 8) = 1408$

מכיוון שהשינוי של cin מ 0 ל 1 גורם לכל ה cout\_i (לכל  $1 \leq i \leq 64$ ) להשתנות בכל alu וגורם לשינויים במספר שערים הכי גדול ולכן הוא המסלול שגורם להשהיה הכי מקסימלית.

## תוצאת הסימולציה על הבדיקות שנדרשנו לעשות:



(המספרים בתוכנה טיפה קטנים) אבל אפשר לראות בתמונה הראשונה שבזמן 1500ps שינינו את cin מ 0 ל 1 אחרי שהתיצבו הערכים והיציאות, ובמתונה השנייה אחרי 1408ps ברגע (2908 = 1500 + 1408) השתנתה היציאה cout וזה תואם לחישובים שלנו בטבלה למעלה והגדרתנו לזמני ההשהיות בסעיף 2.2 .