

205854599	צחי אדרי
931202808	רפאל שרביט

חלק יבש

2.1

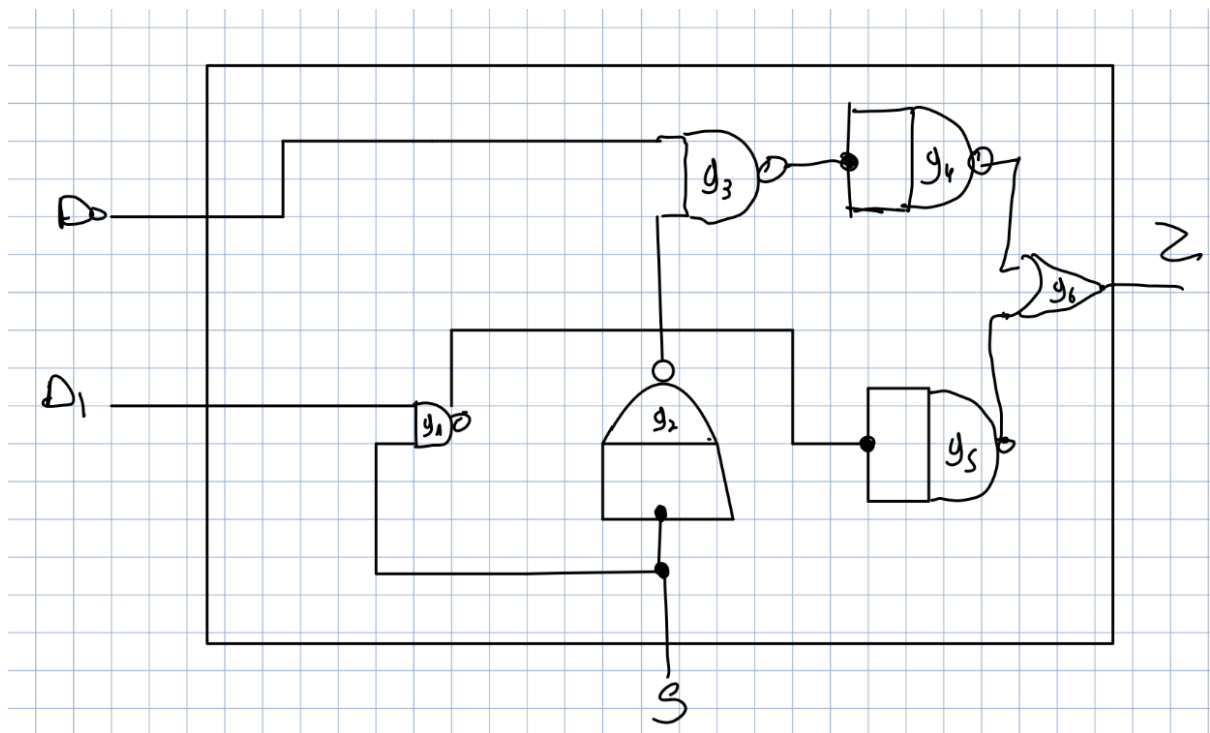
נממש בורר 2-1 בעזרת שערי NAND עם 2 כניסות, שערי OR עם 2 כניסות ושערי XNOR עם 2 כניסות.

טבלת אמת לבורר:

D0	D1	Selector	Z
0	0	0	0
0	1	0	0
1	0	0	1
1	1	0	1
0	0	1	0
0	1	1	1
1	0	1	0
1	1	1	1

נשים לב ש $Z = \text{Sel}' * D0 + \text{Sel} * D1$

לאחר פישוטים אלגבריים על מנת לממש בעזרת הרכיבים שקיימים לנו במערכת, נקבל את המעגל הבא:



נחשב את זמני השהייה:

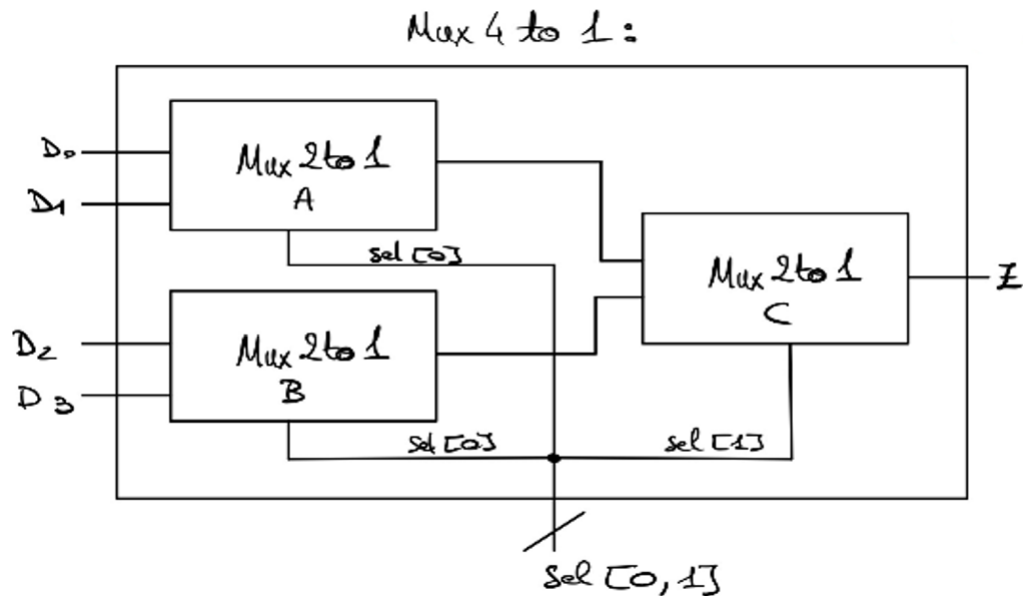
Path	D0	D1	Selector	TPD
Sel->g1->g5->g6->Z	0	1	0->1	23
Sel->g2->g3->g4->g6->Z	1	0	1->0	33
Sel->g1->g5->g6->Z	0	1	1->0	20
Sel->g2->g3->g4->g6->Z	1	0	0->1	25
D1->g1->g5->g6->Z	0	0->1	1	23
D1->g1->g5->g6->Z	0	1->0	1	20
D0->g3->g4->g6->Z	0->1	1	0	23
D0->g3->g4->g6->Z	1->0	1	0	20
D1->g1->g5->g6->Z	1	0->1	1	23
D1->g1->g5->g6->Z	1	1->0	1	20
D0->g3->g4->g6->Z	0->1	0	0	23
D0->g3->g4->g6->Z	1->0	0	0	20

הTPD חושב לפי הת"ר 205854599:

	tpdlh	tpdhl
NAND2	B=10	C=5
OR2	D=8	E=5
XNOR2	F=4	G=5

2.2

על מנת לממש בורר $1 < 4$ בעזרת בוררי $1 < 2$, נצטרך שלושה בוררי $1 < 2$, כאשר ה Selectorn כוקטור של שתי ביטים יכריע האם ניגש לשני הבוררים השמאליים או לימני:



חישוב זמני ההשהייה יתבצע לפי הטבלה הבאה:

*	tpdlh	tpdhl
NAND2	10	10
OR2	8	8
XNOR2	5	5

נבחר את כניסה D0 להיות הכניסה שמשתנה, ונקבע את שאר הכניסות

$D1 = D2 = D3 = \text{Sel}[0] = \text{Sel}[1]$ ל0.

ההשהיה המקסימלית כאשר D0 עובר מ0 ל1:

- בבוררים הראשונים שדרכם עוברים הכניסות D_i , זמן ההשהייה הוא כזה:
 $28 = \text{Tpdl}(\text{Nand}) + \text{Tpdl}(\text{Nand}) + \text{Tpdl}(\text{or})$
הוא:

Path: D0->g3->g4->g6->Z

כעת בבורר הבא, אשר אליו נכנסים היציאות של הבוררים הקודמים, זמן ההשהייה והמסלול שבו עוברים זהה. לכן סך הכל $56 = 28 * 2$.

ההשהייה המקסימלית כאשר D0 עובר מ 1 ל 0:

- באופן דומה, בבורים הראשונים זמן ההשהייה היה $28 = Tpd(Nand) + Tpd(Nand) + Tpd(or)$, כאשר המסלול בתוך הבורר 2-1 עצמו הוא:

Path: D0->g3->g4->g6->Z

גם כאן, זמן ההשהייה והמסלול בבורר הבא זהה. לכן בסך הכל זמן ההשהייה 56.

2.3

תחילה נשים לב כי הsum output זהה עבור אותם קלטים בחיבור או בחיסור. לכן ממה שראינו בהרצאה מתקיים

$$Sum = XOR(XOR(a,b), cin)$$

נממש אותה בעזרת

$$Sum = XNOR(XNOR(b,cin), a)$$

עבור הoutput cout נשתמש במפת קרנו:

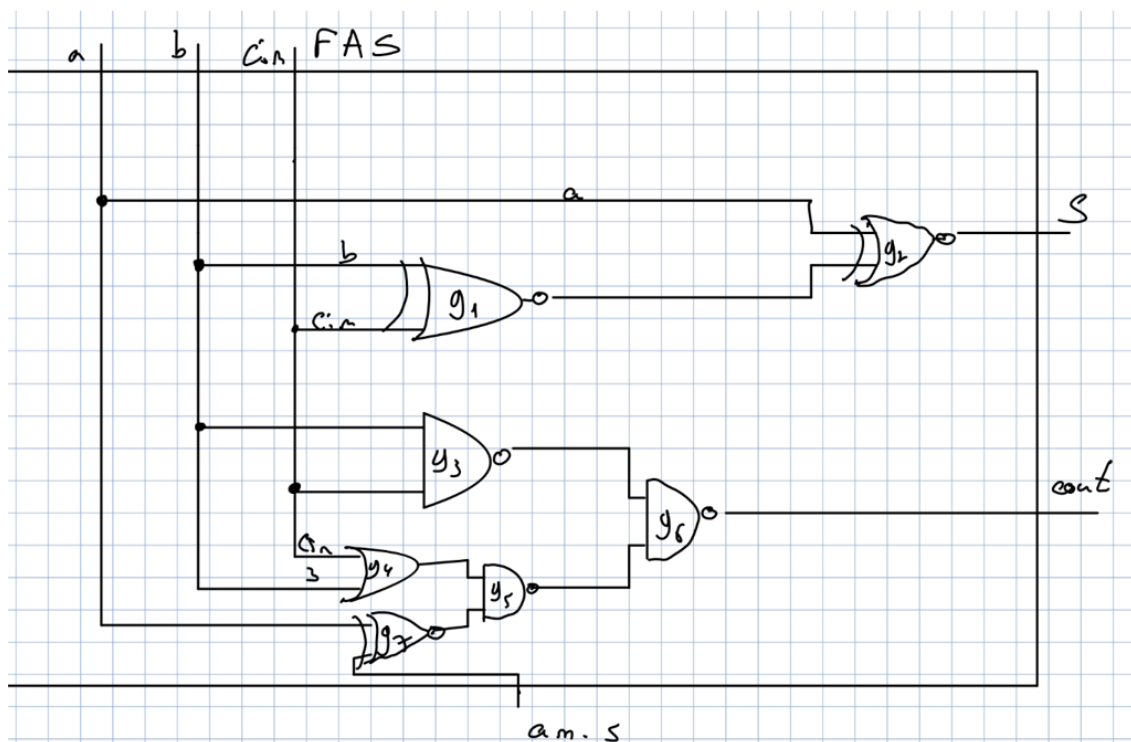
ans,cin b,a	00	01	11	10
00		1		
01			1	
11		1	1	1
10	1	1	1	

$$cout = cin \cdot b + ans \cdot b \cdot a' + ans' \cdot cin \cdot a' + ans \cdot cin \cdot a + ans \cdot b$$

ברשותנו שערי NAND2 OR2 XNOR בלבד. לכן נממש באופן הבא:

$$cout = NAND(NAND(b, cin), NAND(OR(b, cin), XNOR(a, an_s)))$$

הפתרון שלנו מכיל סהכ 7 שערים לכן עומד בדרישה.



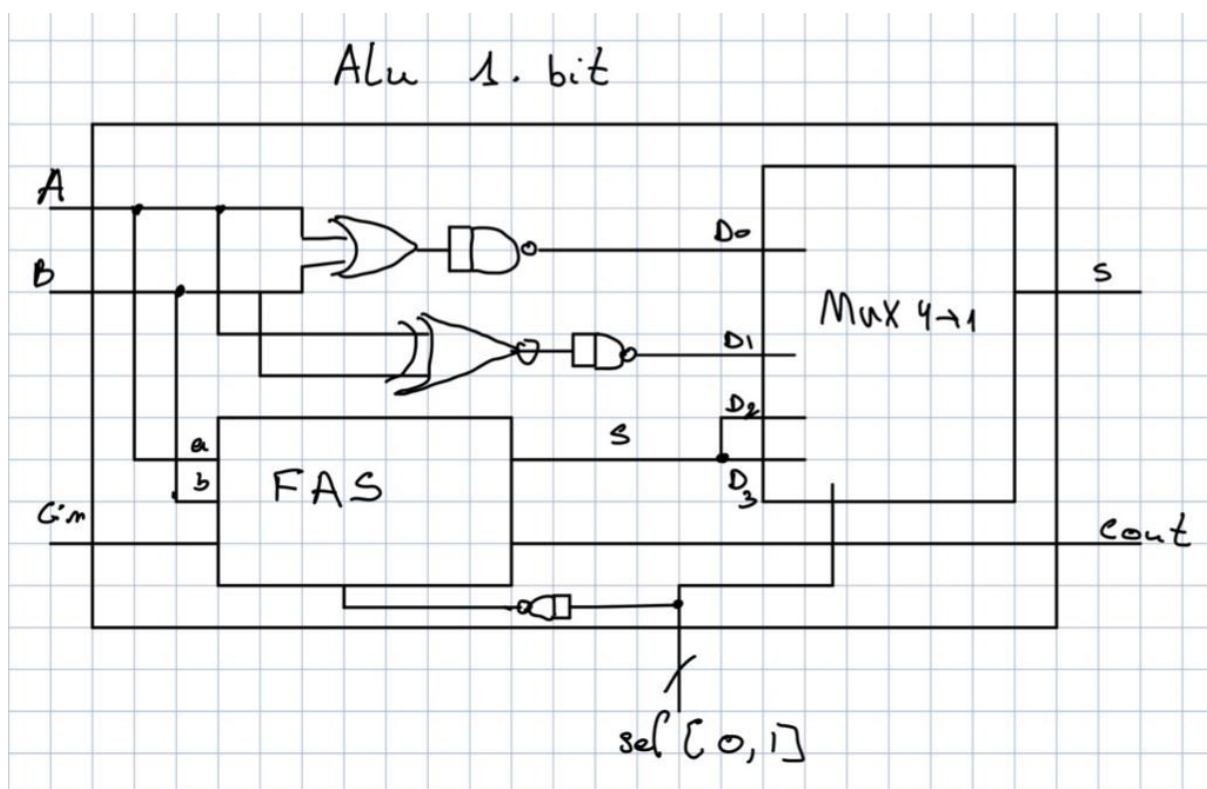
לפי השרטוט הנ"ל והטבלה מסעיף 2.2 נחשב את ההשהיות המקסימליות מכל הכניסה לכל יציאה:

	Path	a	b	cin	an_s	s	cout	tpd
1	a->g1->g2->s	1->0	0	0	0	1->0	1	5
2	a->g7->g5->g6->cout	1->0	1	0	0	0	0->1	25
3	b->g1->g2->s	0	0->1	0	1	0->1	0	10
4	b->g4->g5->g6->cout	0	0->1	0	0	0->1	0->1	28
5	cin->g1->g2->s	0	0	1->0	1	1->0	0	10
6	cin->g4->g5->g6->cout	0	0	1->0	0	1->0	1->0	28
7	an_s->s	∅	∅	∅	∅	∅	∅	0
8	an_s->g7->g5->g6->cout	0	1	0	0->1	1	1->0	25

2.4

נממש ALU אשר הוא מקבל בכניסה מילת בקרה קס בעל שתי סיביות לפי הטבלה הבא :

Operation	פעולת ALU
00	NOR (בין A ל-B)
01	XOR (בין A ל-B)
10	חיבור (כמו ב-FAS)
11	חיסור (כמו ב-FAS)



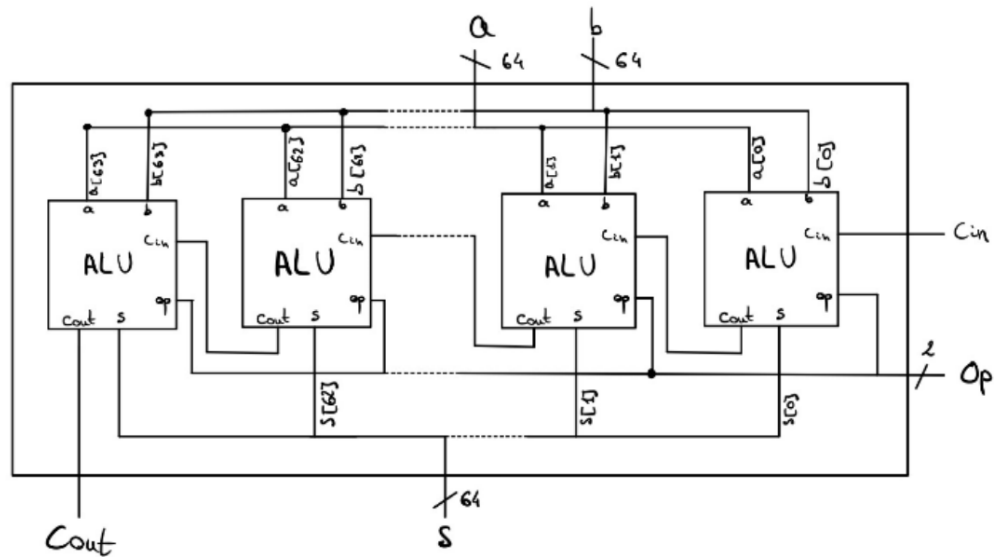
חישוב ההשהיות המקסימליות מכל כניסה לכל יציאה:

Path	A	B	Cin	Op	Sum	Cout	Tpd
A->fas->mux->S	0->1	0	0	00	1->0	0	56+18 = 74

A- >fas- >cout	0->1	1	0	10	1->0	0->1	25
B->fas- >mux- >S	0	0->1	0	00	1->0	0	74
B->fas- >cout	1	0->1	0	10	1->0	0->1	28
Cin- >fas- >mux- >s	0	0	0->1	10	0->1	0	66
Cin- >fas- >cout	1	0	0->1	10	1->0	0->1	28
Op- >mux- >s	1	1	0	00->01	0->1	∅	56
Op- >nand- >fas- >cout	0	1	0	10->11	1	0->1	35

2.5)

alu64bit :



נממש ALU בעל כניסה Data ברוחב 64 סיביות, בעזרת היחידה מסעיף 2.4. והרכיבים שבמערכת שלנו כמו שנתן לראות הנל.

זמן ה tpd המקסימלי הוא עבור מסלול שיחולק לשלושה חלקים באופן הבא:

- החלק הראשון הוא עבור מסלול המתחיל בop, ומסתיים בcout של הרכיב הרשאון.
- החלק השני הינו 62 רכיבי ALU, אשר עוברים בCin ומתסיימים בCout.
- החלק השלישי הינו המסלול שמתחיל ברכיב Cin של הרכיב האחרון, ומסתיים בSum של אותו הרכיב.

לפיכך הTPD המקסימלי הינו:

$$tpd(op \rightarrow cout) + 62 * tpd(cin \rightarrow cout) + tpd(cin \rightarrow sum) = 35 + 62 * 28 + 66 = 1837.$$

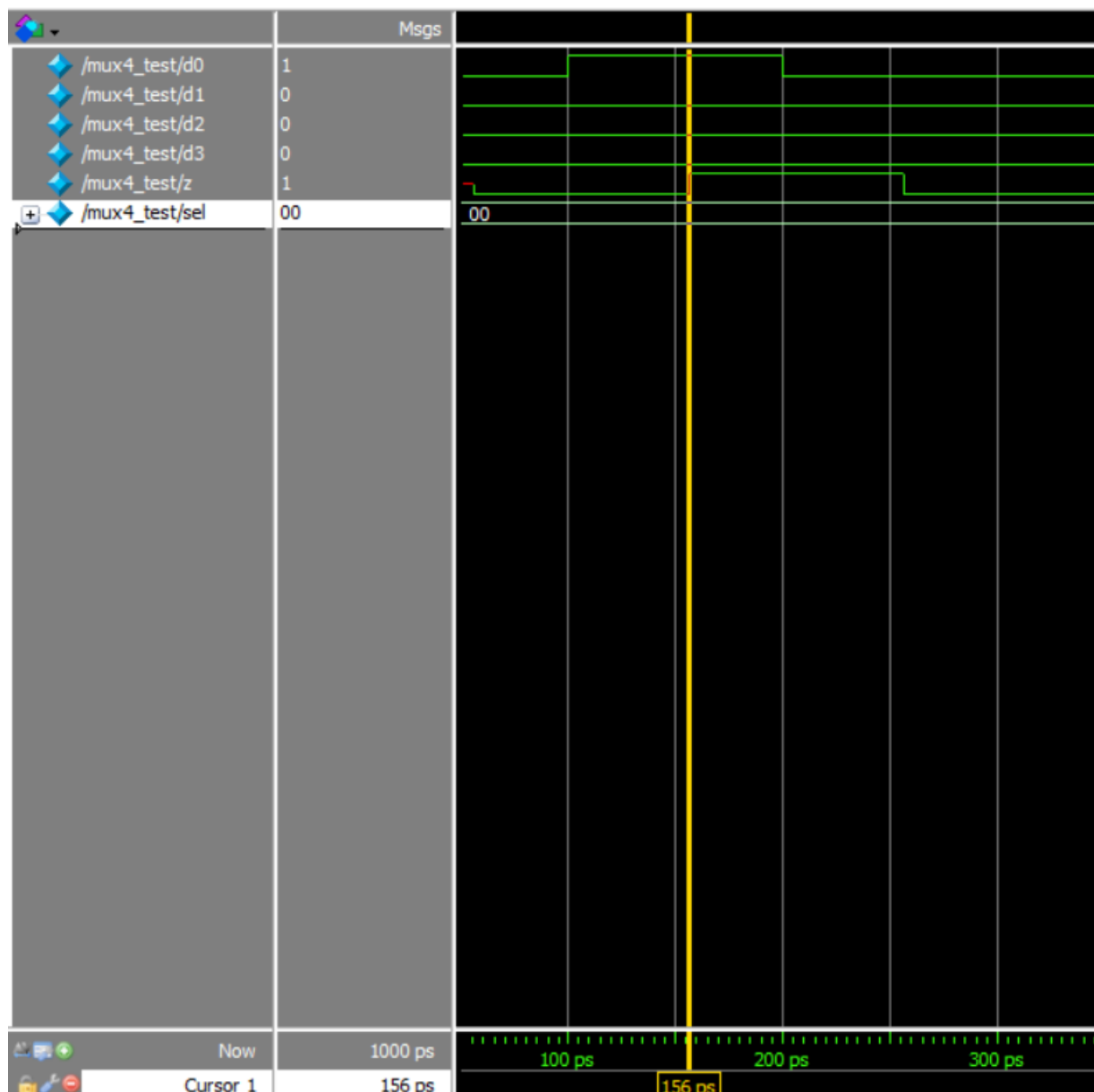
חלק רטוב

3.3

בחרנו את הקלט הבא

$$D0=D1=D2=D3=sel[0]=sel[1]=0$$

ואנחנו משנים את D0 ל 1 . בשאלה 2.2 חישבנו את הזמן המקסימלי ויצא לנו 56. שמנו את אותם קלטים ב Modelsim. הצילום מסך נמצא מתחת. בזמן $t=100ns$ שינינו את D0 מ 0 ל 1 ואז קיבלנו שהשינוי ב z קורה בזמן $t=156$. $t=156-100=56$ כדרוש.



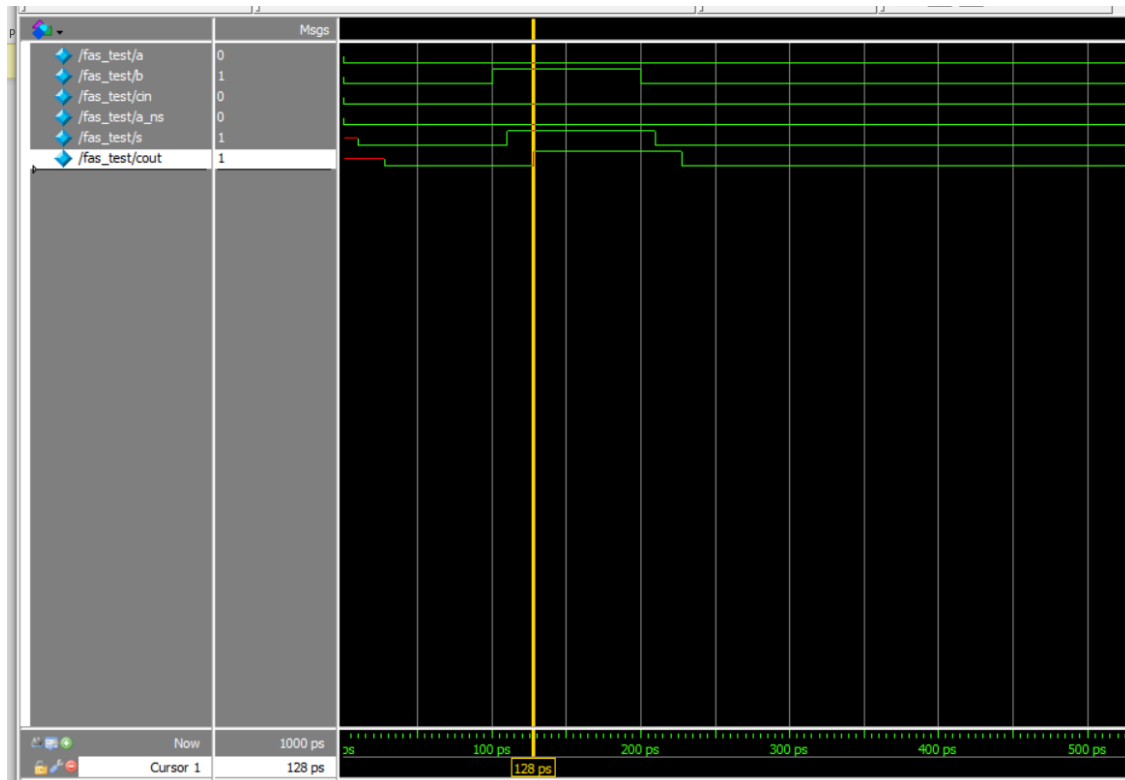
ואז חוזרים לערך המקורי. השינוי קלט קורה בזמן $t=200$ והשינוי פלט בזמן $t=256$ לכן $256-200=56$ כדרוש.

3.5

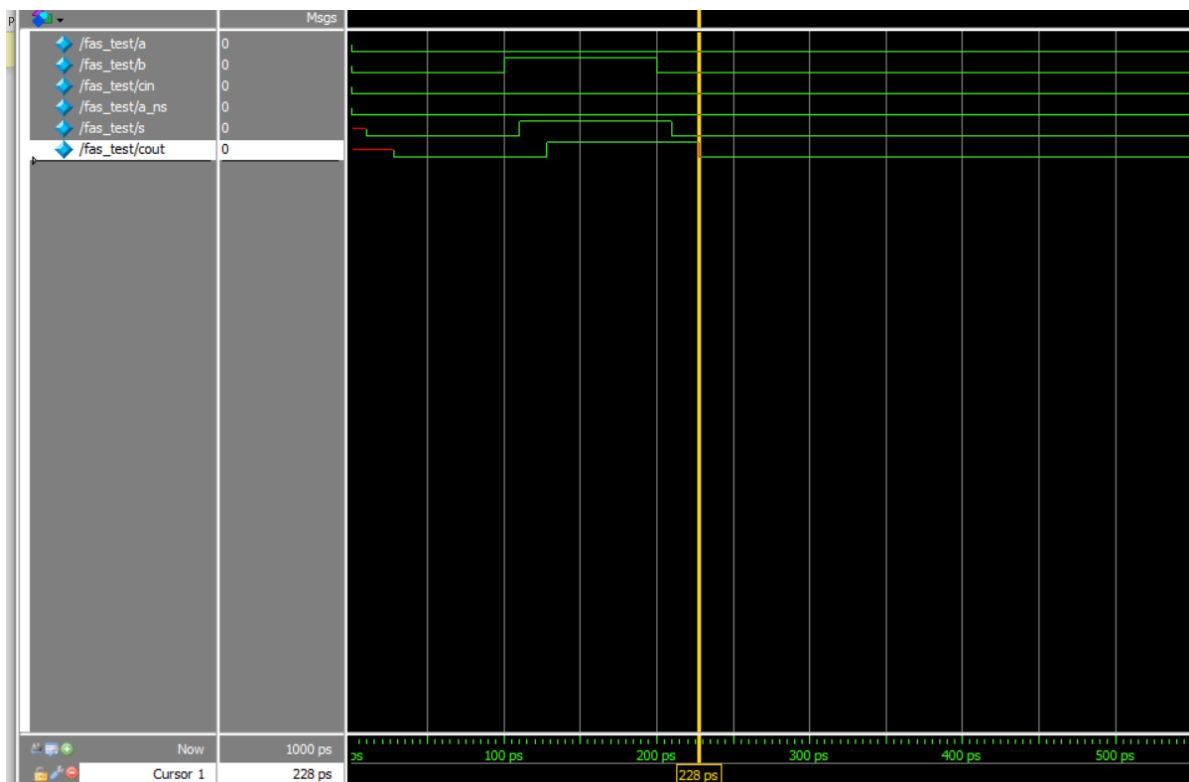
בסעיף זה ההשהיה המקסימלית הייתה כאשר $a=cin=a_{ns}=0$ וב משתנה מהערך 0 לערך 1. החישוב התחיל בזמן $t=100$, שם שינינו את הקלט של b ל 1.

היציאה $cout$ השתנתה בזמן $t=128$, כלומר לאחר $t=28$ כמו שציפינו בחישוב שבסעיף

2.3

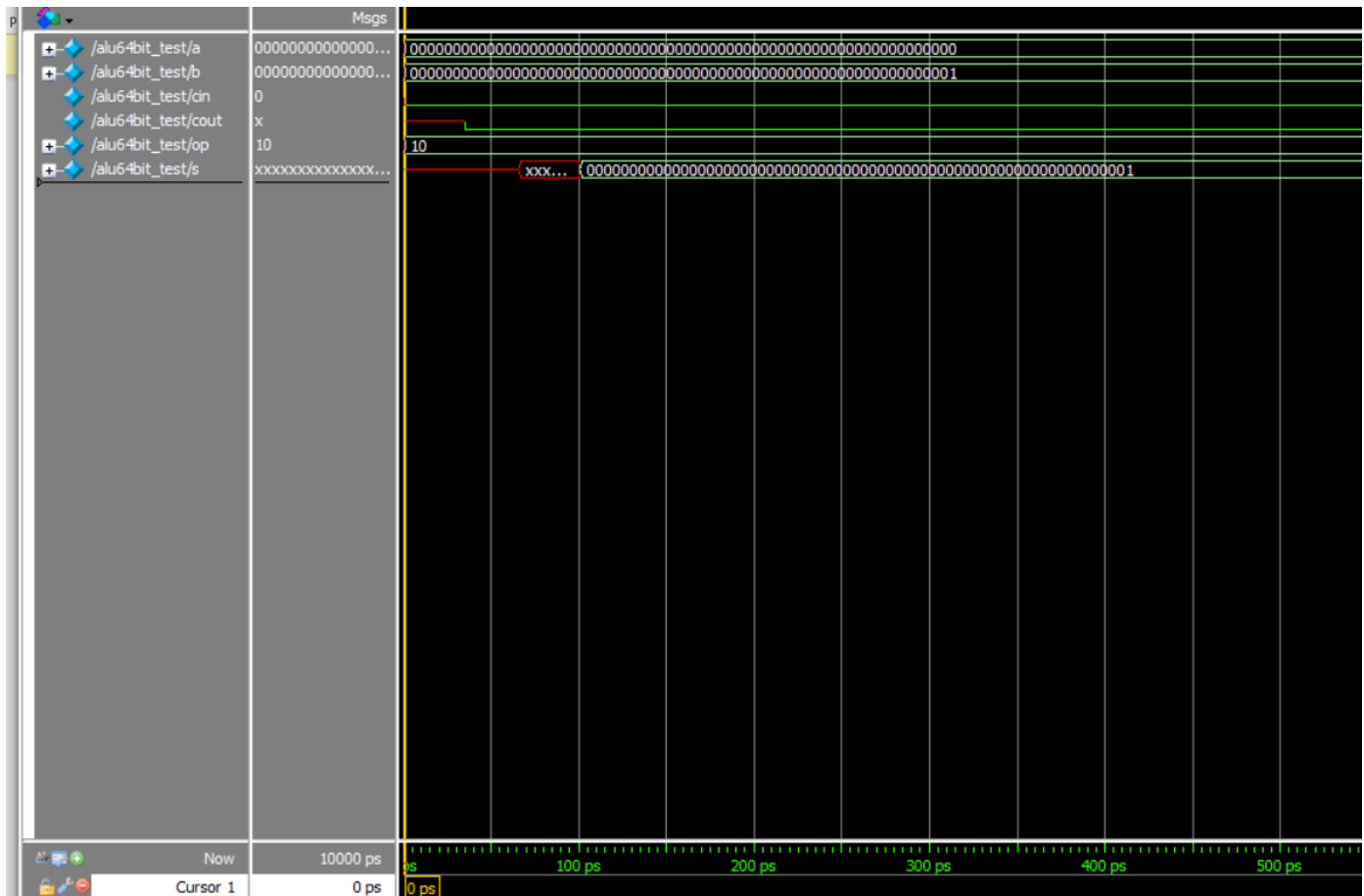


לאחר השהייה על מנת שהפלט יתייצב, שינינו בחזרה את b מ 1 ל 0 בזמן $t=200$, וראינו כי כפי שציפינו, גם כאן לקח לשינוי $t=28$ להתבצע.

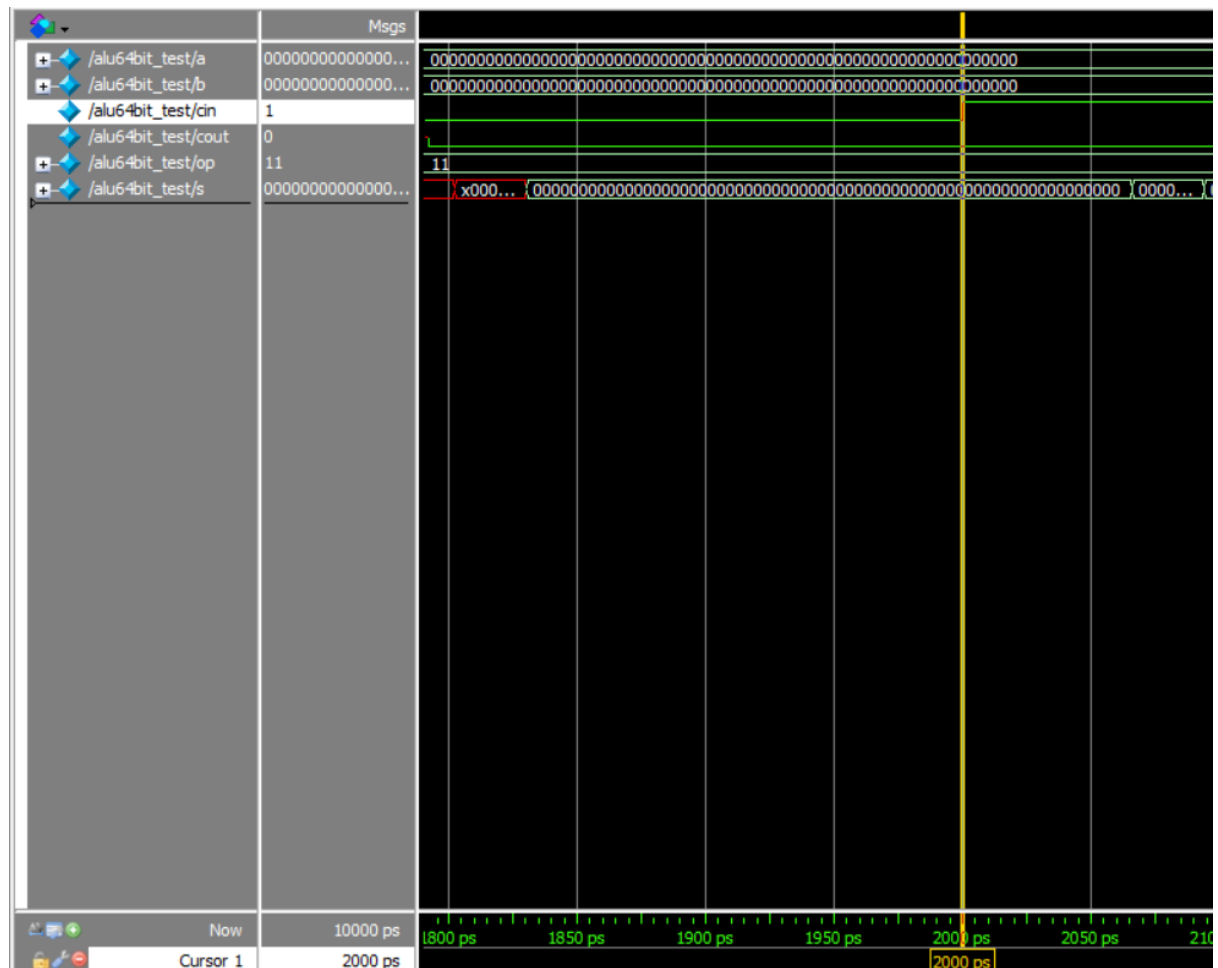


3.8

בסעיף זה נבחר את הקלט הבא :
כל הביטים של a הם 0 וכל הביטים של b הם גם 0 חוץ מ LSB שלו שהוא 1. cin גם שווה ל 0 וקס משתנה מ 10 ל 11.



בזמן $t=2000\text{ns}$ נשנה את קס מ 10 ל 11. כפי שניתן לראות בתמונה מתחת.



שינוי יציאה קורה בזמן $t = 3837$ כאשר שינינו את הכניסה בזמן $t = 2000$ לכן t_{pd} הוא $1837 = 3837 - 2000$, בהתאם לתוצאה שמצאנו בסיף 2.5.