

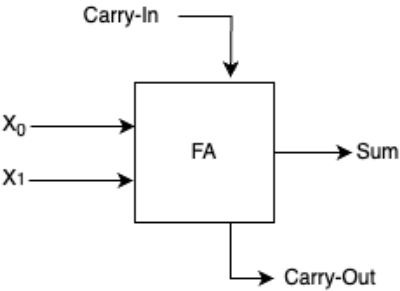
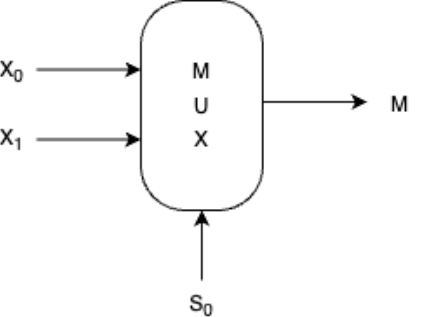
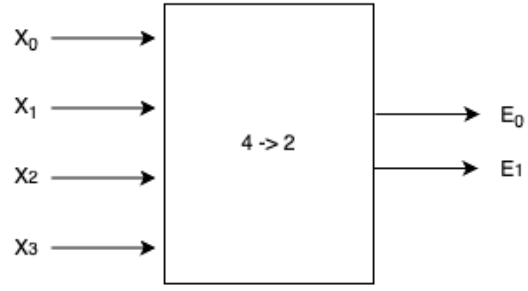
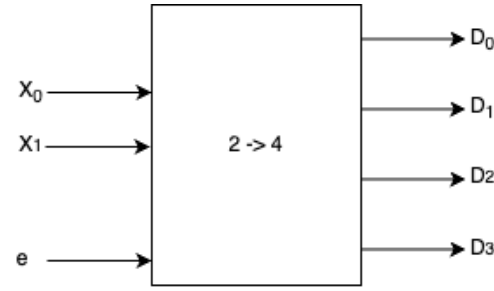


# מעבד חד מחזורי

- הקדמה ורכיבים לוגיים בסיסיים
- בניית ה-ALU (לוגיקה צירופית)
- בניית מקבץ האוגרים (לוגיקה סדרתית)
- שלבי הריצה של פקודה במעבד
- זרימת הנתונים במעבד חד מחזורי
- זרימת הבקרה במעבד חד מחזורי

# מעבד חד מחזורי - סיכום

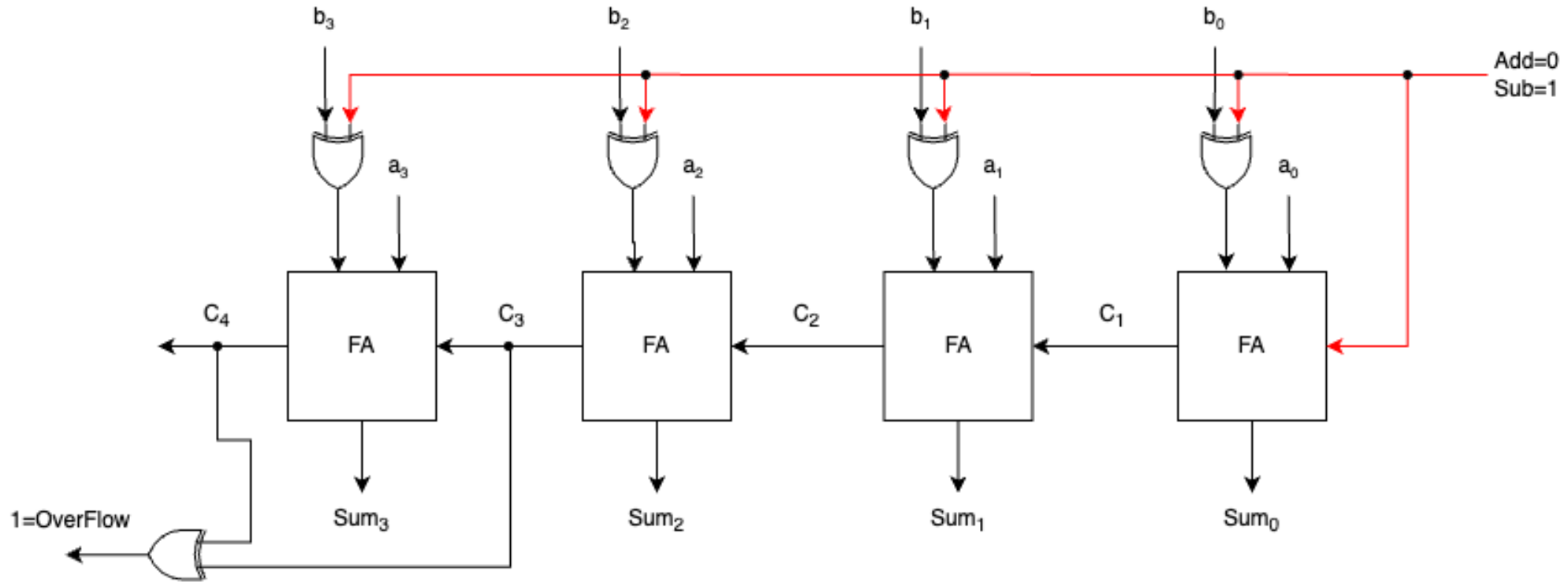


מחבר (Adder)	מרבב (Multiplexer)	מקודד (Encoder)	מפענח (Decoder)																																																																																																										
																																																																																																													
<table border="1" data-bbox="89 615 552 922"> <thead> <tr> <th>C-I</th> <th><math>X_1</math></th> <th><math>X_0</math></th> <th>C-O</th> <th>SUM</th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>1</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr> </tbody> </table>	C-I	$X_1$	$X_0$	C-O	SUM	0	0	0	0	0	0	0	1	0	1	0	1	0	0	1	0	1	1	1	0	1	0	1	1	0	1	1	0	1	0	1	1	1	1	1	<table border="1" data-bbox="771 697 952 842"> <thead> <tr> <th><math>S_0</math></th> <th>M</th> </tr> </thead> <tbody> <tr><td>0</td><td><math>X_0</math></td></tr> <tr><td>1</td><td><math>X_1</math></td></tr> </tbody> </table>	$S_0$	M	0	$X_0$	1	$X_1$	<table border="1" data-bbox="1140 668 1669 872"> <thead> <tr> <th><math>X_3</math></th> <th><math>X_2</math></th> <th><math>X_1</math></th> <th><math>X_0</math></th> <th><math>E_1</math></th> <th><math>E_0</math></th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td></tr> </tbody> </table>	$X_3$	$X_2$	$X_1$	$X_0$	$E_1$	$E_0$	0	0	0	1	0	0	0	0	1	0	0	1	0	1	0	0	1	0	1	0	0	0	1	1	<table border="1" data-bbox="1705 668 2234 872"> <thead> <tr> <th><math>X_1</math></th> <th><math>X_0</math></th> <th><math>D_3</math></th> <th><math>D_2</math></th> <th><math>D_1</math></th> <th><math>D_0</math></th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td></tr> </tbody> </table>	$X_1$	$X_0$	$D_3$	$D_2$	$D_1$	$D_0$	0	0	0	0	0	1	0	1	0	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0
C-I	$X_1$	$X_0$	C-O	SUM																																																																																																									
0	0	0	0	0																																																																																																									
0	0	1	0	1																																																																																																									
0	1	0	0	1																																																																																																									
0	1	1	1	0																																																																																																									
1	0	1	1	0																																																																																																									
1	1	0	1	0																																																																																																									
1	1	1	1	1																																																																																																									
$S_0$	M																																																																																																												
0	$X_0$																																																																																																												
1	$X_1$																																																																																																												
$X_3$	$X_2$	$X_1$	$X_0$	$E_1$	$E_0$																																																																																																								
0	0	0	1	0	0																																																																																																								
0	0	1	0	0	1																																																																																																								
0	1	0	0	1	0																																																																																																								
1	0	0	0	1	1																																																																																																								
$X_1$	$X_0$	$D_3$	$D_2$	$D_1$	$D_0$																																																																																																								
0	0	0	0	0	1																																																																																																								
0	1	0	0	1	0																																																																																																								
1	0	0	1	0	0																																																																																																								
1	1	1	0	0	0																																																																																																								
<p>מקבל <math>2^n</math> כניסות ומוציא את תוצאת החיבור שלהן וכן את הנשא המתקבל כתוצאה מהחיבור. מבדילים בין חצי מחבר (Half Adder) שאיננו מקבל נשא נכנס אלא רק שתי כניסות, לבין מחבר מלא (Full Adder) המקבל גם <math>2^n</math> כניסות וגם נשא נכנס.</p>	<p>מקבל <math>2^n</math> כניסות ו-n כניסות בקרה ומעביר את אחת הכניסות ליציאה על פי הערך בכניסת הבקרה. נקרא גם "בורר" או "mux".</p>	<p>מקבל <math>2^n</math> כניסות, ומוציא n יציאות. המקודד מדליק את היציאה שהמספר שלה מיוצג בינארית בכניסות למפענח.</p>	<p>מקבל n כניסות, ומוציא <math>2^n</math> יציאות. המפענח מדליק את היציאה שהמספר שלה מיוצג בינארית בכניסות למפענח. אפשר להוסיף כניסת enable למפענח וכאשר היא שווה 0, כל היציאות יהיו שוות 0.</p>																																																																																																										

# מעבד חד מחזורי - סיכום



מחבר ומחסר מלא לכמה סיביות (בשיטת משלים ל-2)



# מעבד חד מחזורי - סיכום



## SOP (sum of product) סכום המכפלות ו-POS (product of sum) מכפלת הסכומים

בהינתן טבלת אמת של מספר משתנים, נוכל לבנות פונקציה לוגית המשתמשת בשערים AND, OR ו-NOT בכדי להציג את טבלת האמת. ישנן שתי גישות בכדי להגיע לביטוי אלגברי המשיג את דרישות טבלת האמת:

POS מכפלת הסכומים		SOP סכום המכפלות	
נתייחס רק לשורות שבהן ערך הפונקציה שווה 0. כל משתנה המופיע בשורה זו כ-0, יופיע בסכום כפי שהוא. כל משתנה המופיע כ-1, יופיע בסכום כהופכי של המשתנה. נכפול את כל הסכומים.		נתייחס רק לשורות שבהן ערך הפונקציה שווה 1. כל משתנה המופיע בשורה זו כ-1, יופיע במכפלה כפי שהוא. כל משתנה המופיע כ-0, יופיע במכפלה כהופכי של המשתנה. נסכום את כל המכפלות.	
0	שורות רלוונטיות	1	שורות רלוונטיות
$\bar{x}$	משתנה שווה 1	$x$	משתנה שווה 1
$x$	משתנה שווה 0	$\bar{x}$	משתנה שווה 0
maxterm	סכום סטנדרטי	minterm	מכפלה סטנדרטית
$F(x, y, z) = \prod M_i$	סימון הפונקציה	$F(x, y, z) = \sum m_i$	סימון הפונקציה

לדוגמה:

SOP סכום המכפלות

$$F(x, y, z) = \sum m_i = \sum (2,3,4) = \bar{x}y\bar{z} + \bar{x}yz + x\bar{y}\bar{z} + xyz = x\bar{y}\bar{z} + yz + \bar{x}y$$

POS מכפלת הסכומים

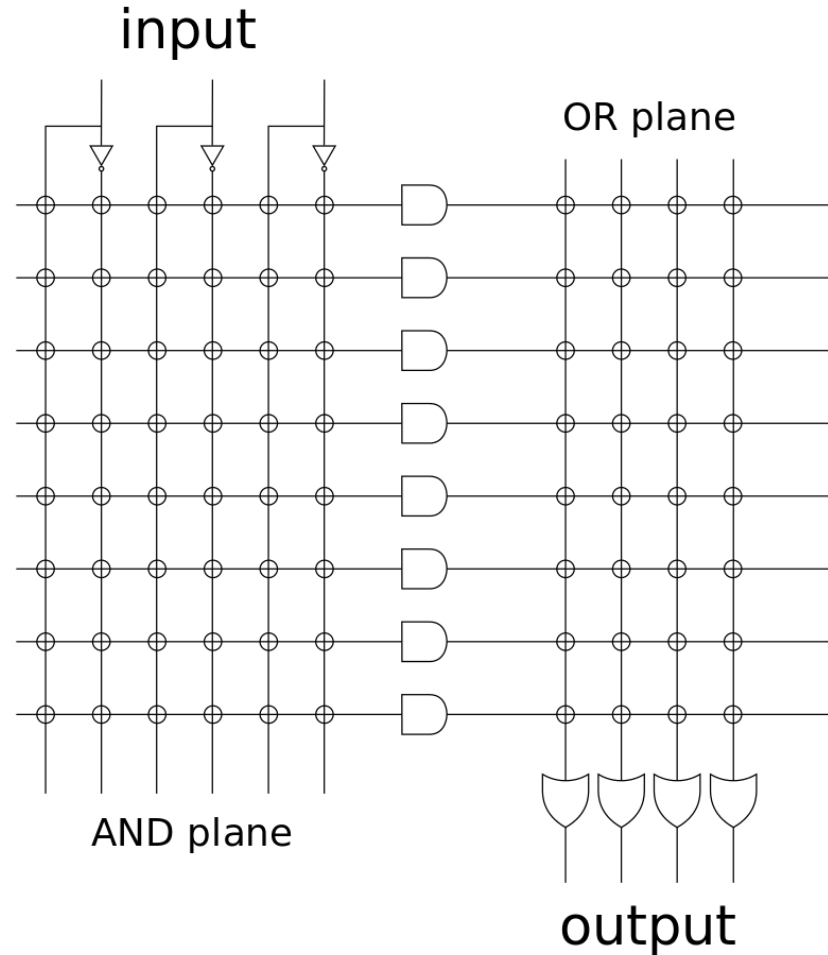
$$F(x, y, z) = \prod M_i = \prod (0,1,5,6) = (x + y + z)(x + y + \bar{z})(\bar{x} + y + \bar{z})(\bar{x} + \bar{y} + z) = x\bar{y}\bar{z} + yz + \bar{x}y$$

	x	y	z	f
$m_0$	0	0	0	0
$m_1$	0	0	1	0
$m_2$	0	1	0	1
$m_3$	0	1	1	1
$m_4$	1	0	0	1
$m_5$	1	0	1	0
$m_6$	1	1	0	0
$m_7$	1	1	1	1

# מעבד חד מחזורי - סיכום



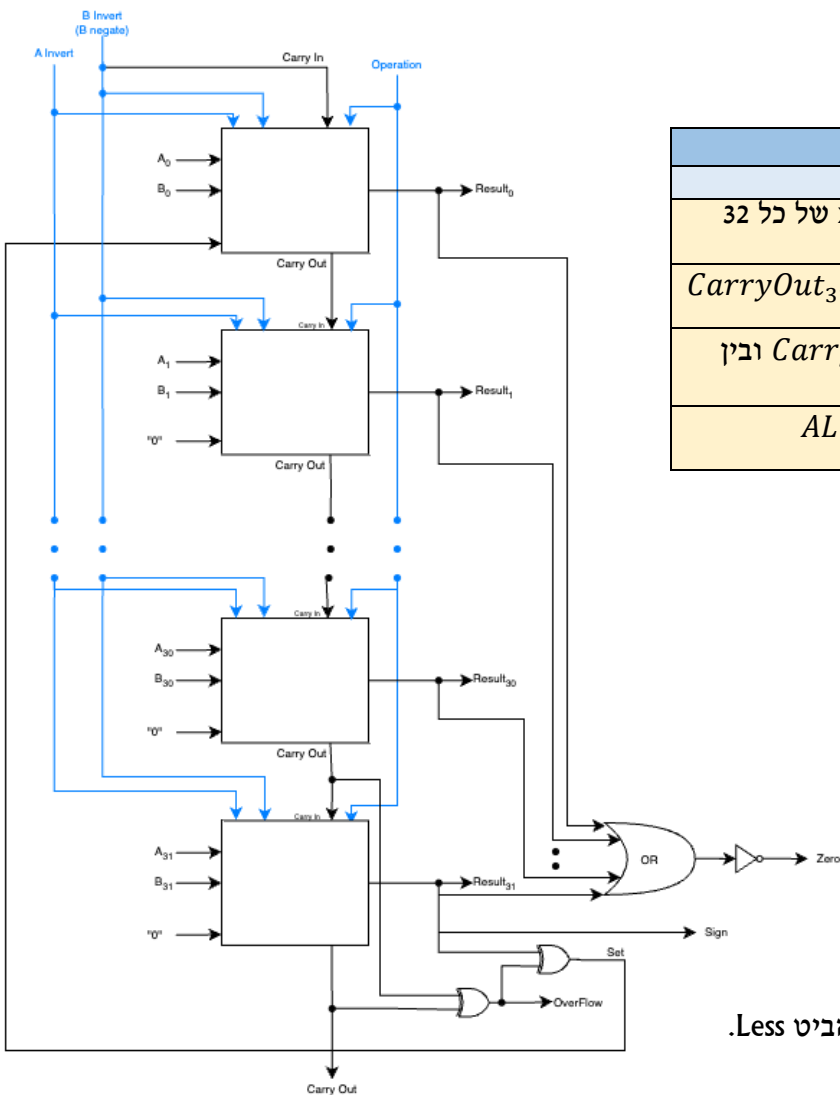
## PLA - Programmable logic array



# מעבד חד מחזורי - סיכום



## שרשור 32 יחידות ALU



### חיווי על תוצאות ה-ALU

סימון	שם	משמעות	בשביל מה זה טוב?	איך זה קורה?
Z	Zero	ביט השווה 1 אם כל הביטים של התוצאה שווים 0	חיווי בפקודות branch האם יש לבצע את הקפיצה או לא	היפוך סימן לפעולת xor של כל 32 התוצאות
C	Carry Out	ביט השווה 1 אם חיבור שני הביטים ב-MSB הוציא נשא	חיווי לגלישה בשיטת ייצוג ללא סימן	תוצאת הנשא האחרון $CarryOut_{31}$
O	Over Flow	ביט השווה 1 אם יש גלישה לפי משלים ל-2	חיווי לגלישה בשיטת משלים ל-2	פעולת xor בין $CarryOut_{30}$ ובין $CarryOut_{31}$
S	Sign	ביט השווה 1 אם תוצאת ה-MSB שווה 1	חיווי למספר שלילי בשיטת משלים ל-2	תוצאת החיבור של $ALU_{31}$

### פעולות ה-ALU כתוצאה מכניסות הבקרה

למה זה קורה?	Operation	B Invert	A Invert	מה יקרה?
	00	0	0	"וגם"
	01	0	0	"או"
	10	0	0	"חיבור"
$A + (-B) = A - B$	10	1	0	"חיסור"
(לוגי) $\overline{(A + B)} = \overline{A} \cdot \overline{B}$	00	1	1	"לא או"
(לוגי) $\overline{(A \cdot B)} = \overline{A} + \overline{B}$	01	1	1	"לא וגם"
	11	0	1	"הדלק אם קטן מ"

הזמן שלוקח לבצע פעולה לוגית קצר מכיוון שכל הביטים מחושבים במקביל.  
 הזמן שלוקח לבצע פעולה אריתמטית יותר ארוך כיוון שהחישוב מבוצע בטור.

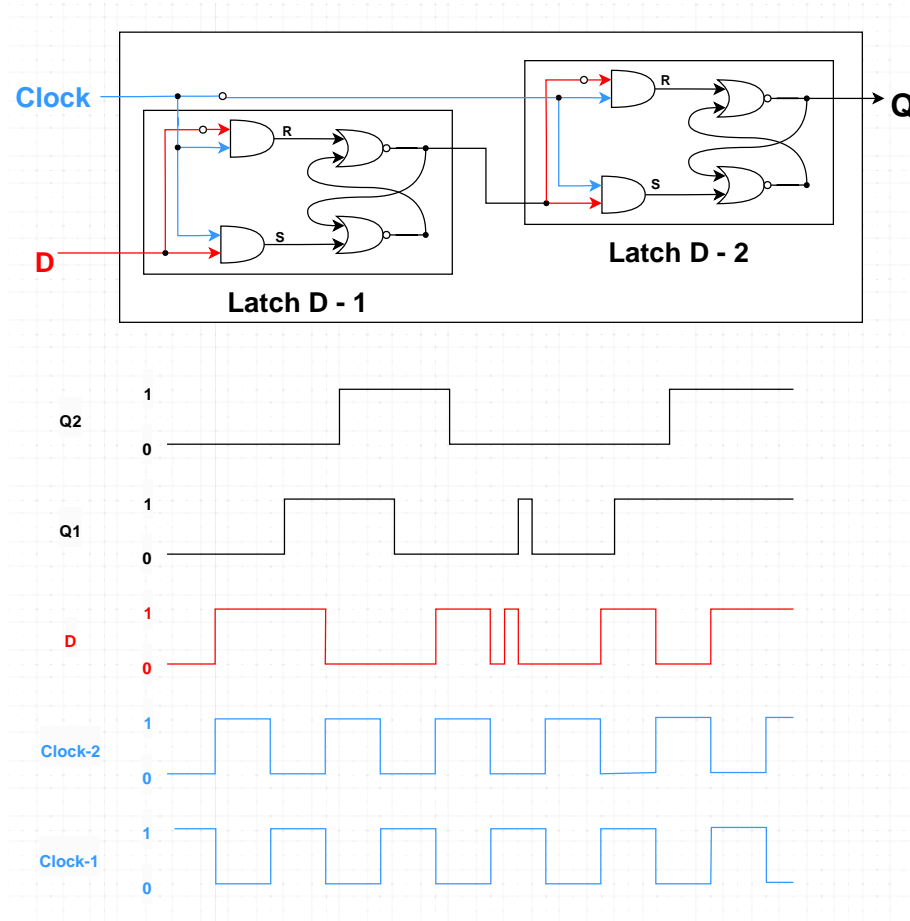
זמן הפעולה של ה-ALU הוא קצת יותר ארוך מפעולה אריתמטית כיוון שבפקודת SLT יש להוסיף את הזמן של הדלקת הביט Less.

# מעבד חד מחזורי - סיכום



## מנעול בקצה שעות - Latch D Edge Triggered

מנעול זה משורשר 32 פעמים עבור כל סיבית באוגר ואז מוכפל 32 פעמים עבור 32 אוגרים

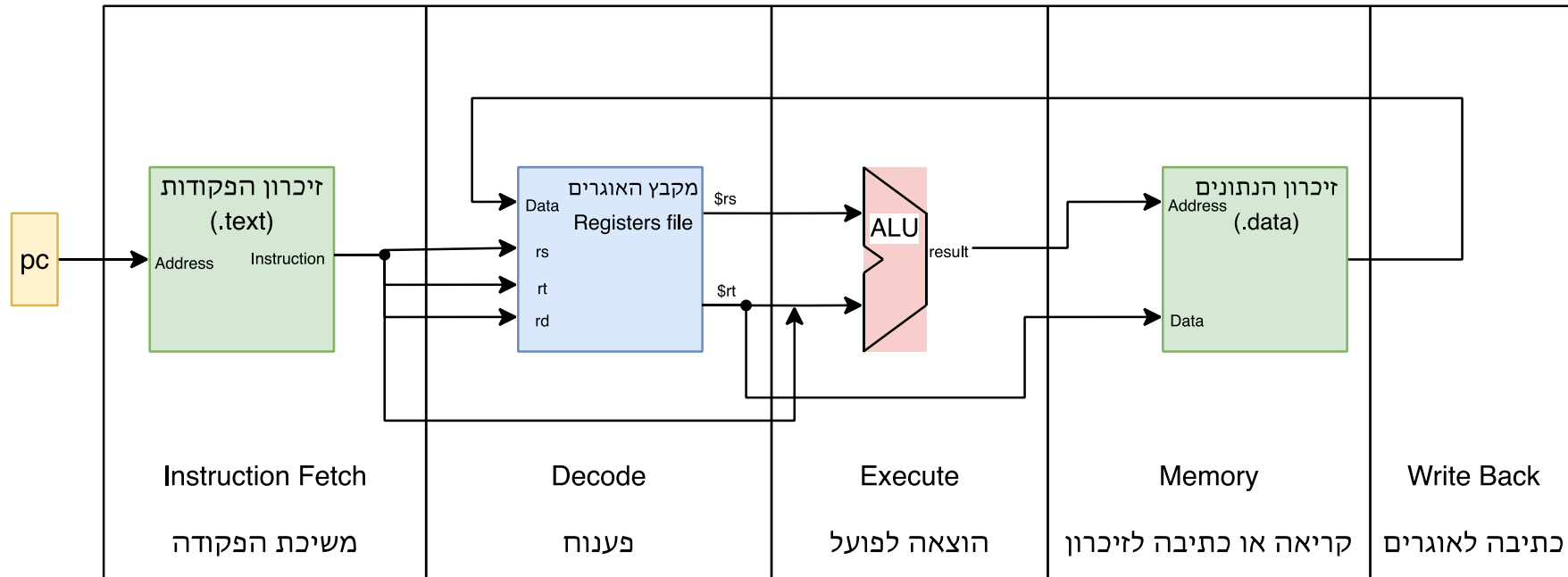


# מעבד חד מחזורי - סיכום



## חמשת שלבי הביצוע של פקודות במעבד

מתי רלוונטי?	שלב		
תמיד	משיכת הפקודה	Instruction Fetch	1
	פענוח (וקריאת האוגרים)	Decode	2
תמיד חוץ מפקודת jump	הוצאה לפועל	Execute	3
רק בפקודות lw, sw	קריאה או כתיבה לזיכרון	Memory	4
בכל הפעולות האריתמטיות / לוגיות ובפקודה lw	כתיבה לאוגרים	Write Back	5

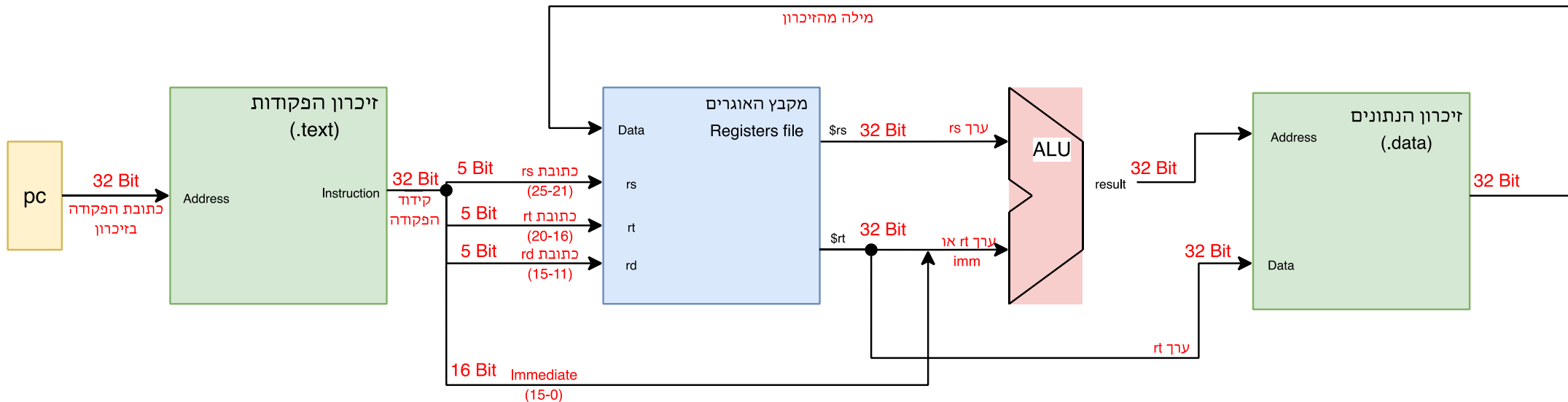




# מעבד חד מחזורי - סיכום



## רוחב הפסים ומשמעות המידע הזורם בהם



# מעבד חד מחזורי - סיכום



## מעבד המסוגל לבצע את הפקודות - ADD, SUB, AND, OR, LW, SW, BEQ, J

פקודה	RegDst	ALUSrc	MemToReg	RegWrite	MemRead	MemWrite	Branch	Jump	ALUOp1	ALUOp0
טיפוס R	1	0	0	1	0	0	0	0	1	0
sw	x	1	x	0	0	1	0	0	0	0
lw	0	1	1	1	1	0	0	0	0	0
beq	x	0	x	0	0	0	1	0	0	1
j	x	x	x	0	0	0	x	1	x	x

