

# CPU Architecture

## Task 1 report

### ALU Design with VHDL

עומר לוקסמבורג 205500390

עילי נוריאל 312538580

## תוכן עניינים

3	הגדרת תכנון המערכת
5	תיאור הרכיבים
6	תיאור הבדיקות
11	File based simulation of DUT

## הגדרת תכנון המערכת

התבקשנו לכתוב תיאור חומרה למערכת הבאה :

תכנון ALU בסיסי, עם רגיסטרים בכניסה וביציאה שלו. ה-ALU מורכב מ:

1. יחידה אריתמטית המבצעת פעולות מתמטיות.
2. יחידת הזזה – shift – המבצעת פעולות הזזה שונות.
3. בורר יציאה – Selector – בין היחידה האריתמטית ל-shift unit.

המערכת מקבלת שני Inputים A,B באורך ברירת מחדל של 8 ביט, OPCODE באורך 5 ביט ו-Cin ומוציאה את תוצאות הפעולה התואמת ל-OPCODE לרגיסטר RES המורכב משני רגיסטרים באורך 8 ביט (LOW ו-HIGH).

### דרישות נוספות:

- השכבה העליונה של ה-ALU צריכה להיות Structural.
- יש להשתמש בקבצים top ו-aux\_package שניתנו עם המשימה.
- יש להשתמש במודול אחד של כפל ומודול אחד של חיבור-חיסור בלבד.
- נדרש לכתוב Test Bench למערכת כולה.

### תרשים המערכת:

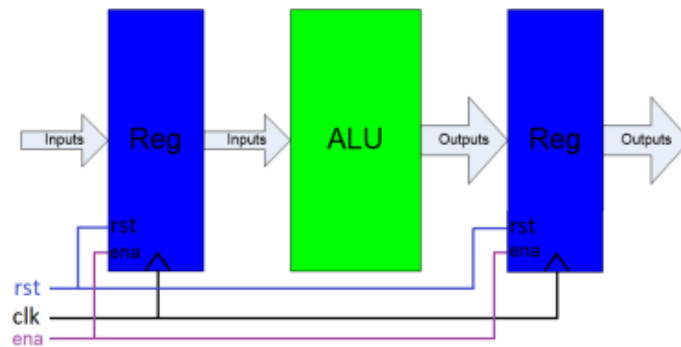


Figure 1: Overall DUT structure

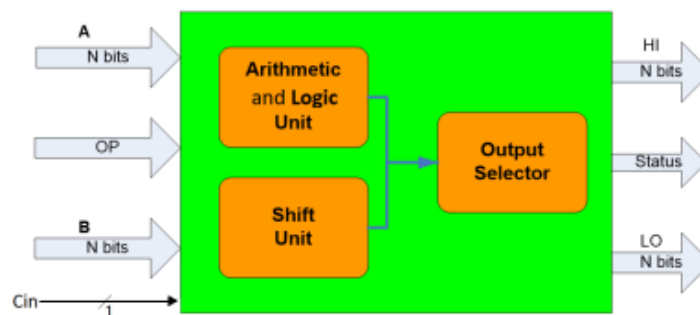


Figure 2: ALU structure

## חיבורים:

$A, B$  – ערוצי הכניסה בעלי  $n$  ביטים כל אחד.

$C_{in}$  – כניסה של ביט נוסף (carry).

OPCODE – בחירת הפעולה שתבצע על הכניסות הנקבעת באופן הבא:

Instruction	Opcode	Operation	Note
ADD	00001	$RES(HI,LO) = A + B$	As in lab1 requirement (see FAQ file)
SUB	00010	$RES(HI,LO) = A - B$	As in lab1 requirement (see FAQ file)
ADDC	00011	$RES(HI,LO) = A + B + C_{in}$	As in lab1 requirement (see FAQ file)
MULT	00100	$RES(HI,LO) = A * B$	Multiply two integer <b>unsigned</b> numbers (Result is $N*2$ bits)
MAC	00101	$ACC = ACC + A * B$ $RES(HI,LO) = ACC$	Multiply Accumulate (MAC is internal $N*2$ bits register) integer <b>unsigned</b> numbers
MAC_RST	00110	$MAC = 0$	Reset MAC
MAX	00111	$RES(0,LO) = \text{Max}(A, B)$	maximum between A and B
MIN	01000	$RES(0,LO) = \text{Min}(A, B)$	minimum between A and B
AND	01001	$RES(0,LO) = \text{AND}(A, B)$	Bitwise operation
OR	01010	$RES(0,LO) = \text{OR}(A, B)$	Bitwise operation
XOR	01011	$RES(0,LO) = \text{XOR}(A, B)$	Bitwise operation
RLA	01100	$RES(0,LO) = \text{rotation result}$	Rotate left A arithmetically B(2..0) times
RLC	01101	$RES(0,LO) = \text{rotation result}$	Rotate left A through carry B(2..0) times
RRA	01110	$RES(0,LO) = \text{rotation result}$	Rotate right A arithmetically B(2..0) times
RRC	01111	$RES(0,LO) = \text{rotation result}$	Rotate right A through carry B(2..0) times

Table 1 : ALU Op Codes

Hi, Low – ערוצי היציאה באורך  $n$  ביטים כל אחד.

Status – יציאה באורך 2 ביטים המייצגת דגלי Zero ו-Carry של הפעולה האחרונה שהתבצעה:

Status Flag Name	Condition	Indicator $(status_1, status_0)_2$
C	Last OP produces carry	01
Z	Last OP result is zero	10
	Last OP produces both (C&Z)	11
	else	00

Table 2 : Status Bus

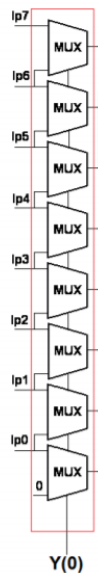
## תיאור הרכיבים

לפי הגדרת המשימה הגדרנו את המודולים ותתי המודולים הבאים (מסודרים לפי היררכיית השכבות):

aux_package	top			
	ALU			
	Selector	Shift_Unit		Arithmetic_Logic_Unit
		RRA	RLA	Adder_Substructor
		Yblock	Yblock	FA
		MUX2	MUX2	

### הערות – לגבי המודולים ממעבדה 1:

- מודול ה-Yblock כתוב בצורה גנרית ומתאר שכבה אחת ברכיב ה-shifter (כמתואר בדרישות המעבדה הראשונה) כלומר עמודה כזו:



- מודול ה-FA מתאר רכיב Full Adder לשני ביטים ומודול ה-MUX2 מתאר מרבב 1 → 2.

## תיאור הלוגיקה של המודולים בעמוד הבא

## תיאור הבדיקות

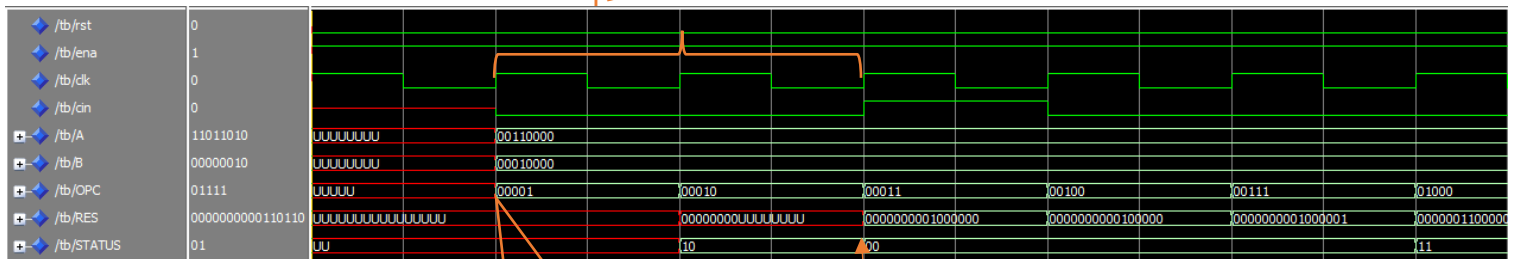
### File based simulation of DUT

כתבנו מודול tb היוצר מקובץ ה-input.txt סימולציה למערכת.  
 בקובץ ה-input הכנסנו תרחישים לכל הפקודות השונות אותן המערכת צריכה לבצע (כל ה-opcodes),  
 וכן מקרי קצה.  
 כלומר הוא מכיל ערכים לכל הקלטות של המערכת. כמו כן המודול tb כותב את הפלט לקובץ  
 .output.txt

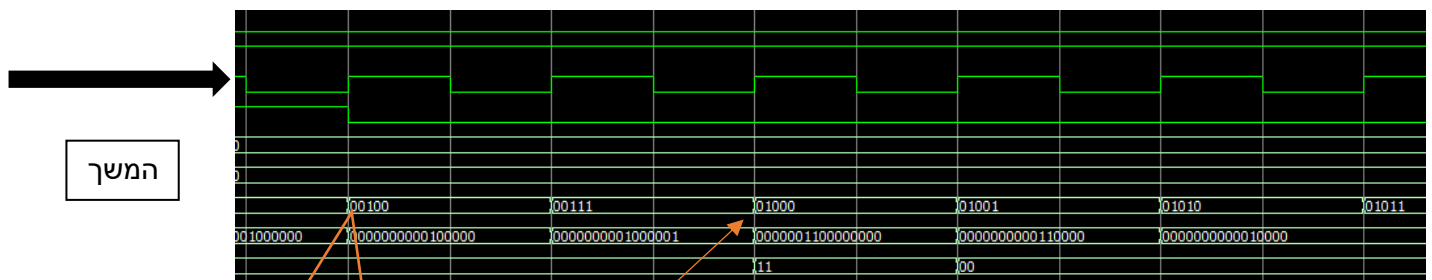
### ניתוח תוצאות הסימולציה בתצוגת wave form :

בדיקת הפעולות ADD, SUB, ADDC, MULT, MAX, MIN, AND, OR, XOR :

#### שני מחזורי שעות

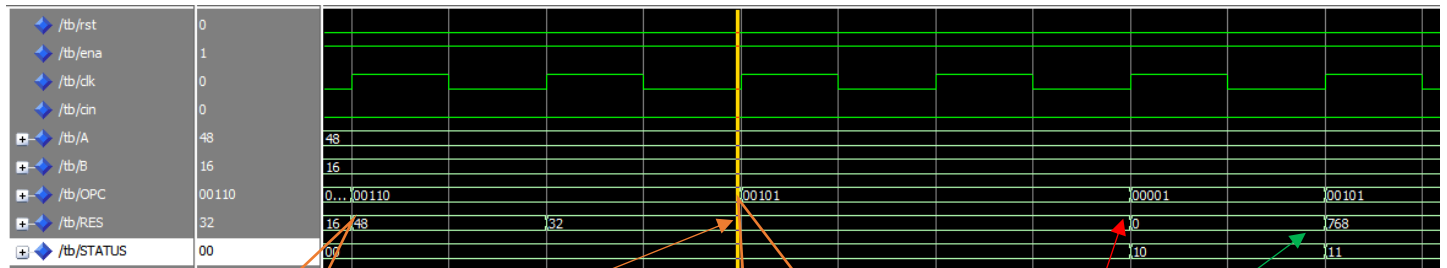


ה-opcode משתנה ל-1 כלומר פעולת ADD.  
 נשים לב כי הפלט התואם לפקודה הזו ייצא רק  
 לאחר שני מחזורים שעות בעקבות ההשהיה של  
 רגיסטרי הכניסה והיציאה.  
 לאחר שני מחזורי שעות ערך היציאה הוא אכן  
 $A + B = 64 = (1000000)_2$



ה-opcode משתנה ל-100 כלומר פעולת MULT.  
 נשים לב כי הפלט התואם לפקודה הזו ייצא רק לאחר שני מחזורים שעות בעקבות  
 ההשהיה של רגיסטרי הכניסה והיציאה.  
 לאחר שני מחזורי שעות ערך היציאה הוא אכן  
 $A \cdot B = 1536 = (11000000000)_{10}$ , כאשר  $A=32$ ,  $B=48$

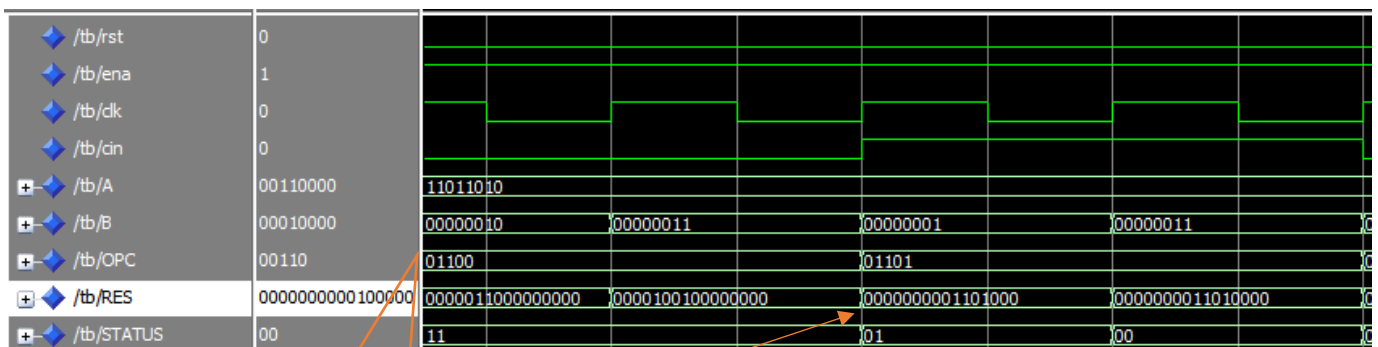
## בדיקת הפעולות MAC, MAC\_RST :



1. ה opcoden משתנה ל110 כלומר פעולת MAC\_RST. ערך RES לא משתנה

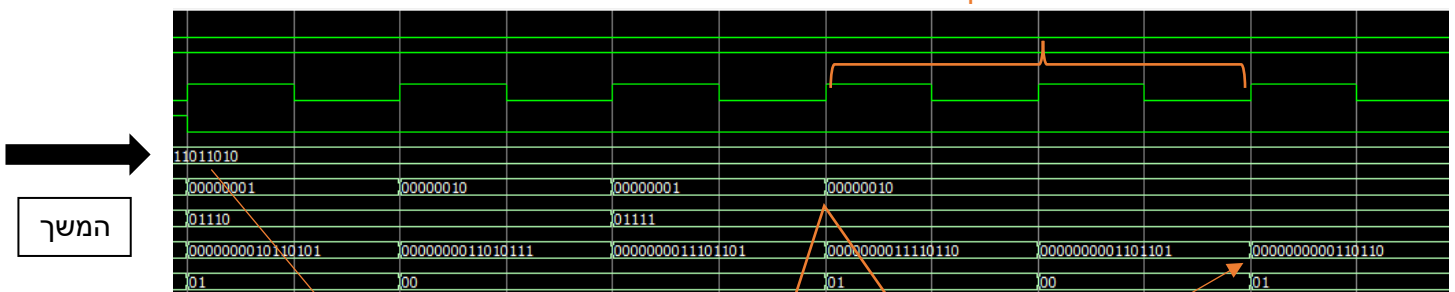
2. ה opcoden משתנה ל101 כלומר פעולת MAC ונשאר במשך עוד שני מחזורים. החץ האדום מציג את המצב בו ACC=0 - הערך 0 הוא המוצא עדיין. החץ הירוק מציג את הערך שבו ACC כבר שונה מ-0.

## בדיקת הפעולות RLA, RRA, RRC, RLC :



ה opcoden משתנה ל1100 כלומר פעולת RLA. B=2 לכן יש הזזה של 2 עם הוספת "00" מימין.

### שני מחזורי שעון



המשך

ה opcoden הוא 01111 כלומר פעולת RRC. B=2 A=11011010 - כלומר יתבצעו 2 הזזות על A כאשר cin=0. נשים לב שזה אומר שביט Carry צריך להיות 1 (הביט האחרון שיצא מ-RES(LO)). בנוסף התוצאה תהיה RES(LO)=01101101 כאשר ה-0 האדום הוא ביט הקארי ומשמאל אליו ביט ה-LSB.