

# CPU Architecture

## LAB2 preparation report

### VHDL part2

Sequential code and Behavioral modeling

עומר לוקסמבורג 205500390

עילי נוריאל 312538580

## תוכן עניינים

3	הגדרת תכנון המערכת
4	תיאור הרכיבים
7	תיאור הבדיקות
7	בדיקת ה-sync delay
8	בדיקת ה-condition
9	בדיקת ה-counter
10	בדיקת ה-detector
11	בדיקת ה-top

## הגדרת תכנון המערכת

התבקשנו לכתוב תיאור חומרה למערכת הבאה :

מערכת סינכרונית שמזהה רצפים חוקיים של מספרים, לפי תנאי משתנה.

המערכת מורכבת מארבעה תתי מודולים :

1. Synchronous Delay – מודול סינכרוני היוצר השהייה בכניסה של סדרת המספרים.
2. cond - מודול א-סינכרוני הקובע אם התנאי מתקיים בין שני המספרים האחרונים שנכנסו בסדרה.
3. counter - מודול סינכרוני הסופר את באופן רציף את קיום התנאי מהמודול הקודם. אם התנאי לא מתקיים הוא מתאפס. סופר עד 7.
4. detector - מודול א-סינכרוני המקבל את ערך counter ומעלה דגל כאשר counter הגיע ל-7.

### דרישות נוספות:

- התכנון של השכבה העליונה יכול להיות behavioral.
- יש להשתמש בקבצי "top" ו-"aux\_package" הנתונים עם המשימה. מותר להוסיף קוד רק בקובץ ה-top.
- יש להשתמש בקובץ ה-"Adder" שנתון עם המשימה, פעם אחת בלבד.

### תרשים המערכת:

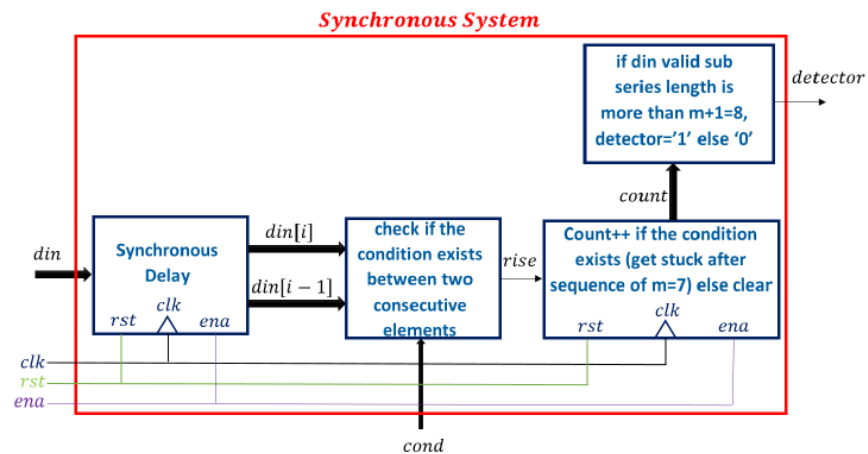


Figure 1 : System structure

### חיבורים:

- Clk – ביט אחד - אות שעון.
- rst – ביט אחד - מאפס את המערכת.
- ena – ביט אחד - מאפשר את פעולת המערכת.
- Din – וקטור של ביטים - כניסת סדרת המספרים.
- Detector – ביט אחד – מוצא המערכת, מציין האם סדרת המספרים מקיימת את התנאי.

Cond – שני ביטים – התנאי הנדרש על סדרת המספרים לפי הטבלה הבאה :

cond	Condition type	Explanation
0	$din[i] - din[i - 1] = 1$	ascending sub series by 1
1	$din[i] - din[i - 1] = 2$	ascending sub series by 2
2	$din[i] - din[i - 1] = 3$	ascending sub series by 3
3	$din[i] - din[i - 1] = 4$	ascending sub series by 4

## תיאור הרכיבים

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

aux_package	top			
	Sync_delay	condition	counter	detect
		Adder		

### הערות :

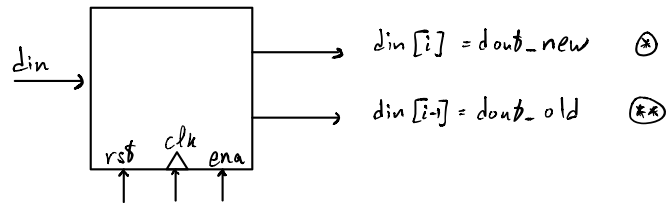
- המערכת (top) גנרית בערכים הבאים :
  - $N$  – מספר הביטים המייצגים את מספרי הסדרה.
  - $M$  – הערך המקסימלי של הcounter.
  - $K = \log_2 M$  – מספר הביטים הנדרש לייצוג  $M$ .
- מודול ה-Adder מתאר רכיב Full Adder ל- $n$  ביטים.

## תיאור הלוגיקה של המודולים בעמוד הבא

## פירט רכיבים

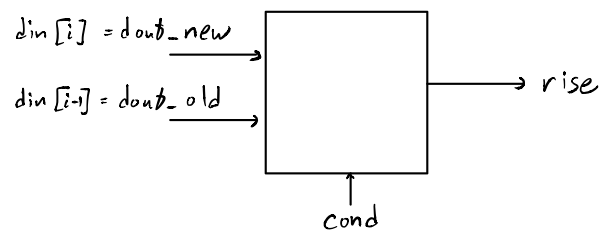
- בבל רכיבים הבאים:
- ena - קו אקטיוו בעל הנהגה.
- rst - קו ריסט האוסס מוצא (אסינכרוני).
- clk - כולל שער המערכת.

### Sync Delay (סינכרוני)



הפקיד מוקדל בה הוא ליגורט את  $din$  הנוכחי (⊗) ואת  $din$  מהחזור הקודם (⊗\*).

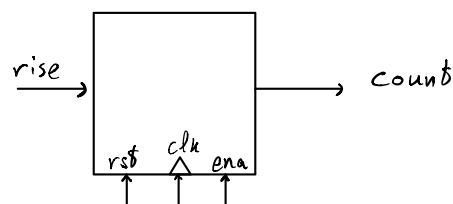
### Condition (אסינכרוני)



מוקדל בה כולל לסי האבלה הבאה -

cond	$d[i] - d[i-1]$	rise
0	1	1
1	2	1
2	3	1
3	4	1
else		0

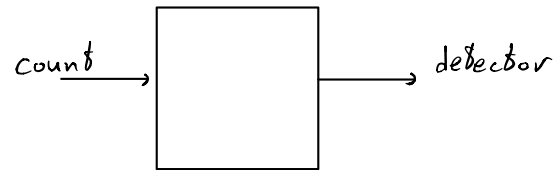
### Counter (סינכרוני)



מוקדל בה מונה את הבלל ה'י' לוי של  $rise$ , כאשר לאחר הפעם המונה 7 (מ) הוא נעצר עליו, ואחראס את  $rise$  הוא 'ס'.

Condition

(אסינכרוני)



כאשר count הוא 7 (אם האוסן שני מ), נקבל מולא הסרס '1'  
לוי -  $\text{detector}$ .

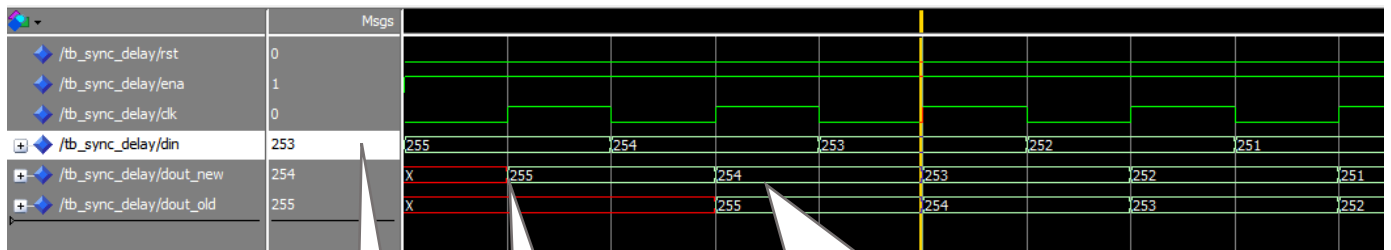
## תיאור הבדיקות

בתיקיית ההגשה מצורפים test branches למודולי ה-top, detect, Sync\_delay, condition, וה-counter.

## בדיקת ה-sync delay

- din – הערך המוכנס
- Doutnew – הערך המוכנס ללא השהייה
- Doutold – הערך המוכנס הקודם – כלומר עם השהייה

ps delta		/tb_sync_delay/rst		/tb_sync_delay/dout_old
		/tb_sync_delay/ena		
		/tb_sync_delay/clk		
		/tb_sync_delay/din		
		/tb_sync_delay/dout_new		
0	+0	0	0 0 00000000	UUUUUUUU UUUUUUUU
0	+1	0	1 0 11111111	UUUUUUUU UUUUUUUU
50000	+1	0	1 1 11111111	UUUUUUUU UUUUUUUU
50000	+2	0	1 1 11111111	UUUUUUUU UUUUUUUU
100000	+1	0	1 0 11111110	UUUUUUUU UUUUUUUU
150000	+1	0	1 1 11111110	UUUUUUUU UUUUUUUU
150000	+2	0	1 1 11111110	UUUUUUUU UUUUUUUU
200000	+1	0	1 0 11111101	UUUUUUUU UUUUUUUU
250000	+1	0	1 1 11111101	UUUUUUUU UUUUUUUU
250000	+2	0	1 1 11111101	UUUUUUUU UUUUUUUU
300000	+1	0	1 0 11111100	UUUUUUUU UUUUUUUU
350000	+1	0	1 1 11111100	UUUUUUUU UUUUUUUU
350000	+2	0	1 1 11111100	UUUUUUUU UUUUUUUU
400000	+1	0	1 0 11111011	UUUUUUUU UUUUUUUU
450000	+1	0	1 1 11111011	UUUUUUUU UUUUUUUU
450000	+2	0	1 1 11111011	UUUUUUUU UUUUUUUU
500000	+1	0	1 0 11111010	UUUUUUUU UUUUUUUU
550000	+1	0	1 1 11111010	UUUUUUUU UUUUUUUU
550000	+2	0	1 1 11111010	UUUUUUUU UUUUUUUU
600000	+1	0	1 0 11111001	UUUUUUUU UUUUUUUU
650000	+1	0	1 1 11111001	UUUUUUUU UUUUUUUU
650000	+2	0	1 1 11111001	UUUUUUUU UUUUUUUU
700000	+1	0	1 0 11111000	UUUUUUUU UUUUUUUU
750000	+1	0	1 1 11111000	UUUUUUUU UUUUUUUU
750000	+2	0	1 1 11111000	UUUUUUUU UUUUUUUU
800000	+1	0	1 0 11110111	UUUUUUUU UUUUUUUU
850000	+1	0	1 1 11110111	UUUUUUUU UUUUUUUU
850000	+2	0	1 1 11110111	UUUUUUUU UUUUUUUU
900000	+1	0	1 0 11110110	UUUUUUUU UUUUUUUU
950000	+1	0	1 1 11110110	UUUUUUUU UUUUUUUU
950000	+2	0	1 1 11110110	UUUUUUUU UUUUUUUU
1000000	+1	0	1 0 11111111	UUUUUUUU UUUUUUUU
1050000	+1	0	1 1 11111111	UUUUUUUU UUUUUUUU
1050000	+2	0	1 1 11111111	UUUUUUUU UUUUUUUU
1100000	+1	0	1 0 11111110	UUUUUUUU UUUUUUUU
1150000	+1	0	1 1 11111110	UUUUUUUU UUUUUUUU
1150000	+2	0	1 1 11111110	UUUUUUUU UUUUUUUU
1200000	+1	0	1 0 11111101	UUUUUUUU UUUUUUUU
1250000	+1	0	1 1 11111101	UUUUUUUU UUUUUUUU



אית הכניסה

ניתן לראות כי אית הכניסה נדגם בעליית שעות.

ניתן לראות כי נוצרה השהייה של מחזור שעות ובכל עליית שעות הפלט הוא din הנוכחי וזה שלפניו

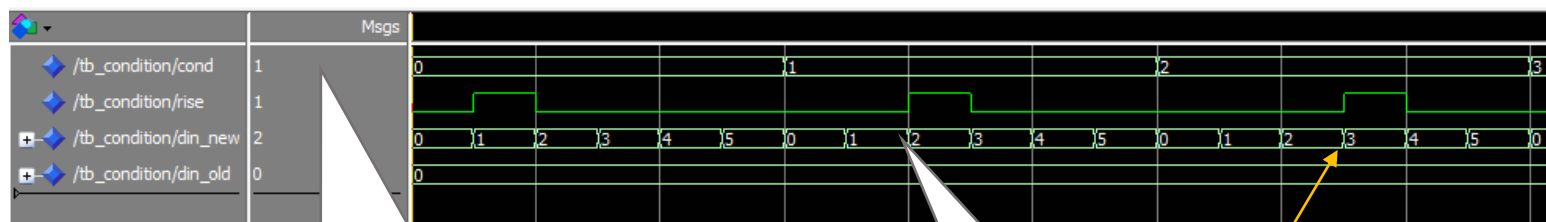
## בדיקת ה-condition

- Cond – התנאי
- Rise – 1 אם התנאי מתקיים
- Din\_new, Din\_old – המספרים ביניהם יש להשוות

בדקנו את כל התנאים, והכנסנו מספרים שמקיימים ולא מקיימים אותם בכדי לבדוק את תקינות Rise.

ps	delta	/tb_condition/cond	/tb_condition/rise	/tb_condition/din_new	/tb_condition/din_old
0	+0	0	0	000	000
0	+1	0	0	000	000
50000	+1	0	0	001	000
50000	+3	0	1	001	000
100000	+1	0	1	010	000
100000	+3	0	0	010	000
150000	+1	0	0	011	000
200000	+1	0	0	100	000
250000	+1	0	0	101	000
300000	+1	1	0	000	000
350000	+1	1	0	001	000
400000	+1	1	0	010	000
400000	+3	1	1	010	000
450000	+1	1	1	011	000
450000	+3	1	0	011	000
500000	+1	1	0	100	000
550000	+1	1	0	101	000
600000	+1	2	0	000	000
650000	+1	2	0	001	000
700000	+1	2	0	010	000
750000	+1	2	0	011	000
750000	+3	2	1	011	000
800000	+1	2	1	100	000
800000	+3	2	0	100	000
850000	+1	2	0	101	000
900000	+1	3	0	000	000
950000	+1	3	0	001	000
1000000	+1	3	0	010	000
1050000	+1	3	0	011	000
1100000	+1	3	0	100	000
1100000	+3	3	1	100	000
1150000	+1	3	1	101	000
1150000	+3	3	0	101	000





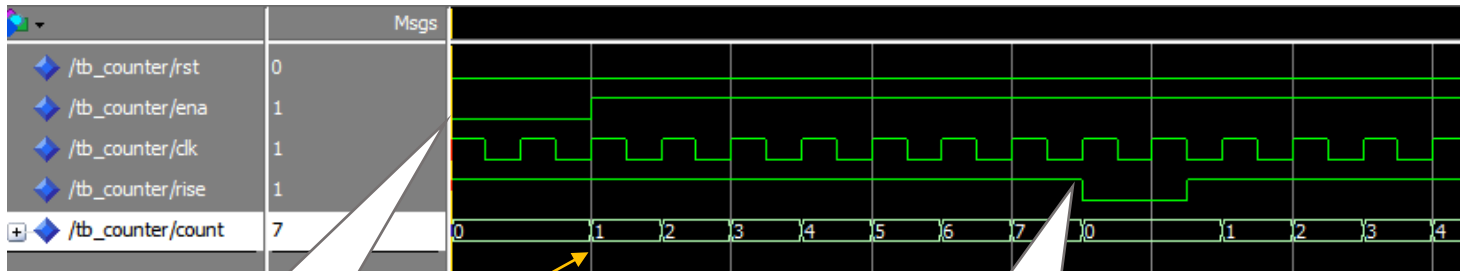
אות הכניסה של התנאי. עבור כל ערך של  $d[i-1]$  ו  $d[i]$  אותה הסדרה של ערכי  $d[i-1]$  ו  $d[i]$  קבוע 0. וכך בדקנו את כל התנאים

ניתן לראות כי השלב בו אות ה `rise` עולה משתנה בהתאם לתנאי

## בדיקת ה- counter

- Rise – 1 אם התנאי (מהמודול הקודם) מתקיים
  - count – מונה את מספר הפעמים הרצופות שהתנאי מתקיים. מפסיק ב-7.
- בדקנו כי ה- counter עולה התאם ל- `rise` ומפסיק לעלות כשהוא מגיע ל-7 גם אם `rise=1`.

ps	delta	/tb_counter/rst	/tb_counter/ena	/tb_counter/clk	/tb_counter/rise	/tb_counter/count
0	+0	0	0	0	0	000
0	+1	0	0	1	1	000
0	+2	0	0	1	1	000
50000	+1	0	0	0	1	000
100000	+1	0	0	1	1	000
150000	+1	0	0	0	1	000
200000	+1	0	1	1	1	000
200000	+3	0	1	1	1	001
250000	+1	0	1	0	1	001
300000	+1	0	1	1	1	001
300000	+3	0	1	1	1	010
350000	+1	0	1	0	1	010
400000	+1	0	1	1	1	010
400000	+3	0	1	1	1	011
450000	+1	0	1	0	1	011
500000	+1	0	1	1	1	011
500000	+3	0	1	1	1	100
550000	+1	0	1	0	1	100
600000	+1	0	1	1	1	100
600000	+3	0	1	1	1	101
650000	+1	0	1	0	1	101
700000	+1	0	1	1	1	101
700000	+3	0	1	1	1	110
750000	+1	0	1	0	1	110
800000	+1	0	1	1	1	110
800000	+3	0	1	1	1	111
850000	+1	0	1	0	1	111
900000	+1	0	1	1	0	111
900000	+3	0	1	1	0	000
950000	+1	0	1	0	0	000

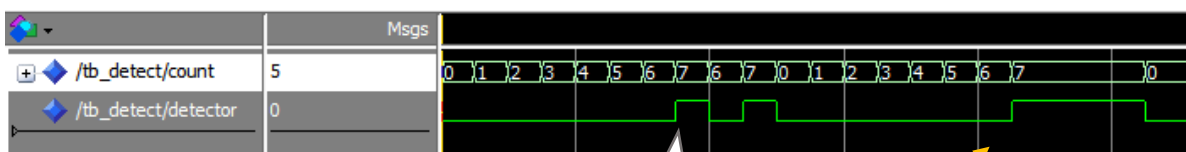


כשקו risen יורד counter מתאפס

### בדיקת ה-detector

- count – מונה את מספר הפעמים הרצופות שהתנאי מתקיים. מפסיק ב-7 (מהמודול הקודם).
- Detector – מוציא 1 אם  $\text{count} = 7$ , מתעדכן בעליית שעון ולכן נראה כאילו יש השהייה בתגובה שלו.

ps	delta	/tb_detect/count	/tb_detect/detector
0	+0	000	0
0	+1	000	1
0	+2	000	0
50000	+1	001	0
100000	+1	010	0
150000	+1	011	0
200000	+1	100	0
250000	+1	101	0
300000	+1	110	0
350000	+1	111	0
350000	+2	111	1
400000	+1	110	1
400000	+2	110	0
450000	+1	111	0
450000	+2	111	1
500000	+1	000	1
500000	+2	000	0
550000	+1	001	0



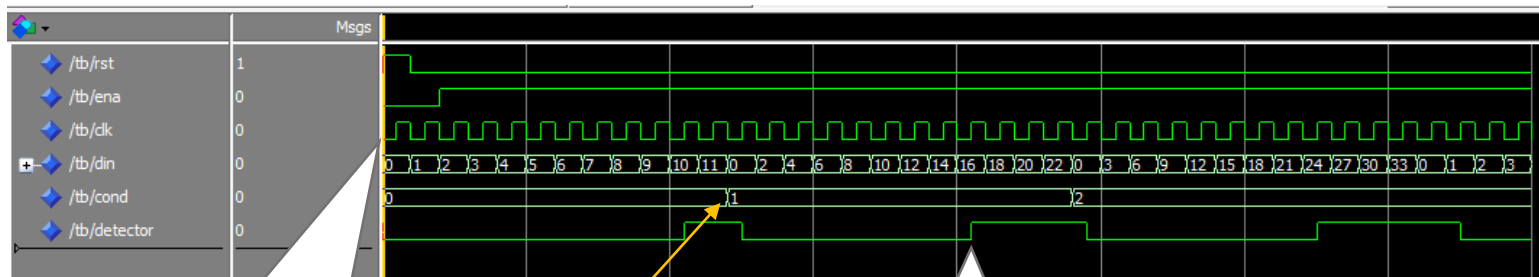
אנו מוודאים שה detector מוציא 1 רק כאשר ה counter הגיע ל-7

## בדיקת ה-top

בצענו בדיקות דומות לבדיקות המתוארות קודם, אך הפעם במודול הסופי שמאחד את כל תתי המודולים.

- Din – אות הכניסה של סדרת המספרים
- cond – התנאי (ההפרש הנדרש בין המספרים בסדרה)
- Detector – עולה ל-1 כאשר נכנסו 8 מספרים רצופים המקיימים את התנאי

ps↓ delta↓		/tb/rst↓ /tb/ena↓ /tb/clk↓	/tb/din↓ /tb/cond↓ /tb/detector↓
0	+0	0 0 0	00000000 0 0
0	+1	1 0 0	00000000 0 1
0	+2	1 0 0	00000000 0 0
50000	+1	1 0 1	00000000 0 0
100000	+1	0 0 0	00000001 0 0
150000	+1	0 0 1	00000001 0 0
200000	+1	0 1 0	00000010 0 0
250000	+1	0 1 1	00000010 0 0
300000	+1	0 1 0	00000011 0 0
350000	+1	0 1 1	00000011 0 0
400000	+1	0 1 0	00000100 0 0
450000	+1	0 1 1	00000100 0 0
500000	+1	0 1 0	00000101 0 0
550000	+1	0 1 1	00000101 0 0
600000	+1	0 1 0	00000110 0 0
650000	+1	0 1 1	00000110 0 0
700000	+1	0 1 0	00000111 0 0
750000	+1	0 1 1	00000111 0 0
800000	+1	0 1 0	00001000 0 0
850000	+1	0 1 1	00001000 0 0
900000	+1	0 1 0	00001001 0 0
950000	+1	0 1 1	00001001 0 0
1000000	+1	0 1 0	00001010 0 0
1050000	+1	0 1 1	00001010 0 0
1050000	+3	0 1 1	00001010 0 1
1100000	+1	0 1 0	00001011 0 1
1150000	+1	0 1 1	00001011 0 1
1200000	+1	0 1 0	00000000 1 1
1250000	+1	0 1 1	00000000 1 1
1250000	+3	0 1 1	00000000 1 0
1300000	+1	0 1 0	00000010 1 0
1350000	+1	0 1 1	00000010 1 0
1400000	+1	0 1 0	00000100 1 0
1450000	+1	0 1 1	00000100 1 0
1500000	+1	0 1 0	00000110 1 0
1550000	+1	0 1 1	00000110 1 0
1600000	+1	0 1 0	00001000 1 0
1650000	+1	0 1 1	00001000 1 0
1700000	+1	0 1 0	00001010 1 0
1750000	+1	0 1 1	00001010 1 0
1800000	+1	0 1 0	00001100 1 0
1850000	+1	0 1 1	00001100 1 0
1900000	+1	0 1 0	00001110 1 0
1950000	+1	0 1 1	00001110 1 0
2000000	+1	0 1 0	00010000 1 0
2050000	+1	0 1 1	00010000 1 0
2050000	+3	0 1 1	00010000 1 1
2100000	+1	0 1 0	00010010 1 1



הכנסנו אות שעון בתדר קבוע, וסדרת מספרים משתנה – תחילה מתקדמת ב-1 לאחר מכן ב-2 ולבסוף ב-3.

כמו כן התנאי משתנה בכדי לבדוק את כל התנאים

ניתן לראות כי ה-detector עולה רק לאחר שהתנאי התקיים 7 פעמים

הסבר :  $cond=1$  כלומר ההפרש צריך להיות של 2

1.  $2-0=2$
2.  $4-2=2$
3.  $6-4=2$
4.  $8-6=2$
5.  $10-8=2$
6.  $12-10=2$
7.  $14-12=2$

ואז בעליית שעון הבאה אכן ה-detector עולה ל-1