



מערכות ספרתיות ומבנה המחשב (044252)

סמסטר חורף תשפ"ב

בחינה סופית – מועד א

7 בפברואר 2022

טור 1

--	--	--	--	--	--	--	--	--	--

מספר סטודנט

משך המבחן: 3 שעות (180 דקות). **תכננו את זמנכם היטב.**

חומר עזר: אין להשתמש בכל חומר עזר בכתב, מודפס או אלקטרוני, פרט לדפי העזר ולמחשבון.

הנחיות והוראות:

- הבחינה כתובה על גבי 24 עמודים כולל עמוד זה (לא רלוונטי עבור קובץ הפתרון) **(בדקו בתחילת הבחינה שלא חסרים לכם עמודים)**. בסה"כ ישנן 17 שאלות: 14 שאלות אמריקאיות, ו-3 שאלות פתוחות מרובות סעיפים.
- בתחילת הבחינה תקבלו חוברת בחינה, מחברת טיוטה, דפי עזר וטופס תשובות ממוחשב. בסיום הבחינה, החזירו את חוברת הבחינה וטופס התשובות הממוחשב בלבד.
- יש לענות על כל השאלות הפתוחות בגוף המבחן, **במלבנים המסומנים לכך בלבד**.
- אין **לתלוש או להפריד** דפים מחוברת הבחינה, ממחברת הטיטה ומדפי העזר.
- יש לכתוב את התשובות באמצעות עט שחור או כחול בלבד. **אין לכתוב או לצייר בעט אדום**.
- רשמו את מספר הסטודנט שלכם על חוברת הבחינה (בראש עמוד זה). **ודאו כי על מחברת הבחינה ועל טופס התשובות האמריקאי מודבקת מדבקת הנבחן שלכם**.
- לא מורדות נקודות (אין "קנס") בגין תשובה שגויה. לכן, בשאלות האמריקאיות כדאי לסמן תשובה כלשהי לכל שאלה.
- ציון השאלות האמריקאיות ייקבע על סמך סריקה ממוחשבת של טופס התשובות בלבד. **לא לשכוח לסמן בטופס התשובות הממוחשב את מספר הטור שלכם (מופיע בראש עמוד זה)**.
- אסור שימוש בכל חומר חיצוני מלבד מחשבון. אסורה העברת חומר כלשהו בין הנבחנים, ואסורה כל תקשורת עם אנשים אחרים או כל מקור מידע. האיסור חל על כל צורות התקשורת – מילולית, חזותית, כתובה, אלקטרונית, אלחוטית, טלפנית, או אחרת. בפרט, אין להחזיק בטלפון סלולארי.

בהצלחה!



שאלה 1 (5 נקודות):

נתון הרכיב my_module הבא:

```
module my_module (
    input logic my_in,
    output logic [3:0] my_out
);

    logic [3:0] my_vec;
    assign my_vec[0] = my_in;

    genvar i;
    generate
        for (i=0; i<3 ; i++)
            begin
                unit #(
                    .SHIFT_NUM(i)
                ) unit_inst (
                    .unit_in(my_vec[i]),
                    .unit_out(my_vec[i+1])
                );
            end
        endgenerate

    assign my_out = my_vec;
endmodule
```

כאשר הרכיב unit הינו:

```
module unit (
    input logic unit_in,
    output logic unit_out
);

    parameter SHIFT_NUM;
    logic [3:0] unit_vec;

    assign unit_vec = 4'b1101 >> SHIFT_NUM;
    xor(unit_out, unit_vec[0], unit_in);
endmodule
```

(תזכורת: במוצא הפעולה: $2 \gg 4'b1000$. נקבל $4'b0010$)



מה המוצא my_out של הרכיב my_module בהרצת סימולציה שבה
my_in=1'b0?

א. 4'b0110

ב. 4'b1101

ג. 4'b0010

ד. 4'b1110

ה. ישנה שגיאה בקוד כתוצאה מהתנגשות. כלומר - ביט אחד מקבל שני ערכים שונים במקביל.

שאלה 2 (5 נקודות):

מהנדס מעוניין לייצר קוד המקבל מספר המורכב מסיביות בינאריות באורך כלשהו, ומסוגל לייצג את שני סוגי המספרים הבאים:

סוג א': מספרים בקידוד המשלים ל-2, בטווח $[-2^8, 2^8 - 1]$

סוג ב': מספרים בקידוד unsigned, בטווח $[322, 833]$

מהו המספר המינימלי של ביטים הנדרשים עבור קוד זה?

רמז – התייחסו גם ליכולת להבדיל בין הסוגים השונים.

א. 9

ב. 10

ג. 11

ד. 12

ה. 13



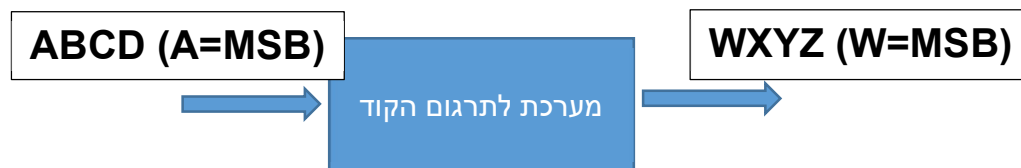
שאלה 3 (5 נקודות):

תכנונו מערכת שמתרגמת בין שני ייצוגים שונים.
המערכת מקבלת כקלט ספרה עשרונית המיוצגת בעזרת 4 סיביות שכתובות בייצוג משוקלל (6, 4, 2, -3) כאשר 6 מייצג את משקל סיבית ה-MSB.

תזכורת: ספרה המיוצגת בייצוג משוקלל היא: $N = \sum_i w_i d_i$
כאשר N מייצג ספרה בין 0 ל-9. שימו לב, ייתכן שלחלק מהקלטים קיים ייצוג כפול.

המערכת מוציאה כפלט את אותה ספרה עשרונית אך מיוצגת בעזרת קוד BCD.

משתני הכניסה הם A,B,C,D (A=MSB)
משתני היציאה הם W,X,Y,Z (W=MSB)



דוגמה:

ABCD=0110 מייצג את הספרה העשרונית "6"
ולכן נקבל במוצא: WXYZ=0110. שימו לב שגם הקלט ABCD=1000 הינו חוקי, ובמוצא המערכת נקבל גם WXYZ=0110.
מבין האפשרויות הבאות, מהו הביטוי המצומצם ביותר כסכום של מכפלות עבור המוצאים W ו-X של המערכת?

א. $W = AD + BCD$
 $X = B'CD + BD' + ABC'$

ב. $W = AD' + BCD'$
 $X = B'CD + BD' + ABC'$

ג. $W = ABC + ACD'$
 $X = B'CD + ABC' + AC'D' + BD'$

ד. $W = AD' + BCD$
 $X = B'CD + BD' + ABC'$

ה. תשובות א-ד אינן מבטאות נכונה את התרגום הנדרש לקביעת W ו-X.



שאלה 4 (5 נקודות):

סטודנטית בקורס מעוניינת לממש את הפונקציה:

$$f(A, B, C, D, E) = \Sigma(19, 21, 22, 27, 29, 30)$$

הסטודנטית מעוניינת לממש את הפונקציה הנתונה, בעזרת מפענחים ושערים לוגים. כלל המפענחים בשאלה בעלי כניסת Enable.

לרשות הסטודנטית עומד שער OR בעל 32 כניסות **שאינו** יספר במניין השערים הלוגים בתשובות. הסטודנטית יכולה להשתמש בקבועים '0' ו-'1' ללא הגבלה.

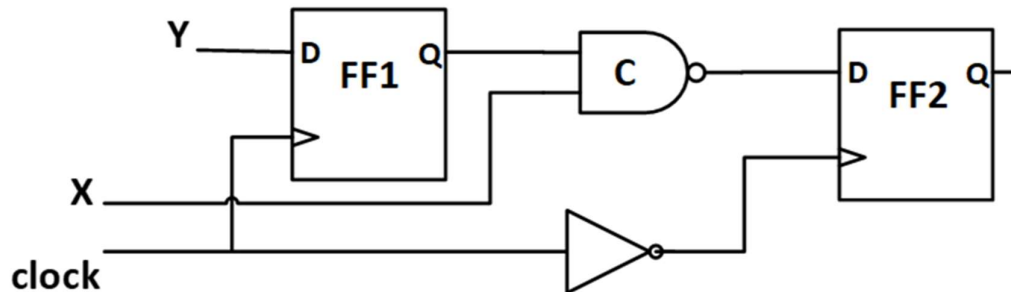
סמנו את התשובה **המאפשרת מימוש של הפונקציה** ומכילה רכיבים לפי סדר עדיפות הסטודנטית, כאשר התשובות מסודרות לפי עדיפות. כלומר, תשובה א' היא הכי פחות עדיפה ותשובה ה' הכי עדיפה.

- א. ניתן לממש את הפונקציה בעזרת מפענח יחיד מסוג $32 \rightarrow 5$, וללא תוספת שערים לוגים.
- ב. ניתן לממש את הפונקציה בעזרת מפענח יחיד מסוג $16 \rightarrow 4$, וללא תוספת שערים לוגים.
- ג. ניתן לממש את הפונקציה בעזרת **שני** מפענחים מסוג $8 \rightarrow 3$, ובתוספת שערים לוגים.
- ד. ניתן לממש את הפונקציה בעזרת מפענח יחיד מסוג $8 \rightarrow 3$, ובתוספת שערים לוגים.
- ה. ניתן לממש את הפונקציה בעזרת מפענח יחיד מסוג $8 \rightarrow 3$, וללא תוספת שערים לוגים.

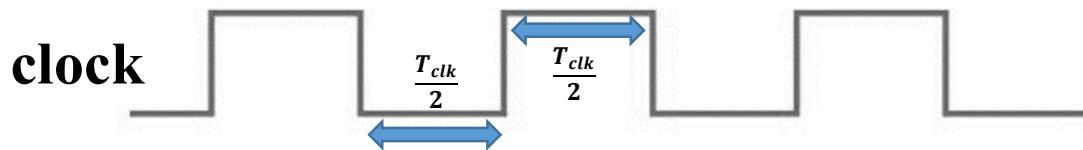


שאלה 5 (5 נקודות):

נתון המעגל הבא:



-	t_{cd}	t_{pd}/t_{pCQ}	t_{setup}	t_{hold}
FF1/FF2	5	40	20	5
NAND	2	22		
NOT	2	3		



השעון שמחובר ל-FF1 הוא בעל $Duty\ cycle=0.5$, כלומר חצי מהמחזור הוא שווה ל-'1', וחצי מהזמן שווה ל-'0'.
שימו לב שהשעון נכנס למהפך לפני שהוא עובר ל-FF2.

בנוסף, יש להתעלם מבדיקת תנאי hold במעגל, ונתון שהכניסות X ו-Y עומדות בתנאי התזמון של המעגל.

מהו זמן המחזור המינימלי המאפשר עבודה תקינה של המערכת?

א. 62

ב. 80

ג. 102

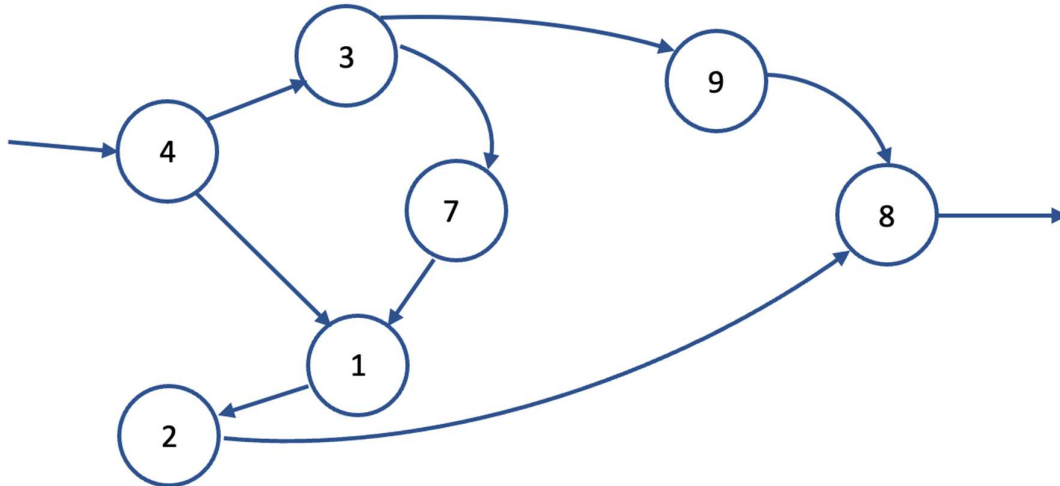
ד. 124

ה. 160



שאלה 6 (5 נקודות):

נתונה המערכת הצירופית הבאה:



זמן ההשהיה של כל רכיב צירופי מצוין על גביו.

מהי כמות הרגיסטרים המינימלית אשר דרושה לצורך צינור המערכת כדי שתתקבל תפוקה (throughput) מקסימלית בעדיפות ראשונה, והשהייה מינימלית בעדיפות שניה?

לצורך הצינור, כל הרגיסטרים הנתונים אידיאליים (בעלי זמן השהייה 0, ותנאי hold מתקיים בכולם).

א. 4

ב. 6

ג. 7

ד. 8

ה. 9



שאלה 7 (5 נקודות):

הוחלט לשנות את פרוטוקול התקשורת UART שנלמד בכיתה
(start bit = 0, 8 bits of data, stop bit = 1) באופן הבא:

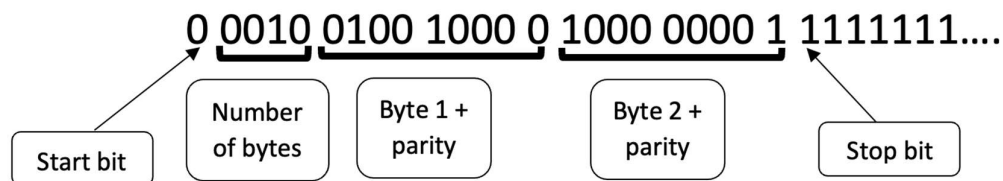
משדרים start bit = 0, ולאחר מכן 4 סיביות המתארות את מספר ה**ביתים** (כל בית מכיל 8 סיביות) שנשדר ברצף. לאחר מכן, על מנת לשפר את יכולת גילוי השגיאות של הפרוטוקול, משדרים כל בית עם סיבית הזוגיות שלו. לבסוף משדרים stop bit = 1.

שימו לב: משמעות שידור בתים ברצף היא שאין ביניהם start bit/stop bit.

בנוסף מוגדר בפרוטוקול שמספר הבתים המשודר גדול מ-0 (כלומר הסיביות 1:4 לא יכולים להיות כולם 0).

השידור מתחיל מה-LSB.

לדוגמה, אם נרצה לשדר את הבתים 0x12 ו-0x01 (בסדר הזה) נשדר את סדרת הסיביות הבאה:



מספר הבתים המשודר הוא $0 < N \leq 15$.

סטודנט מעוניין לשלוח הודעה בעלת N בתים בזמן הקצר ביותר (זמן שליחת סיבית אחת בפרוטוקול החדש ובפרוטוקול הסטנדרטי זהה). מה ה- N המינימלי עבורו הסטודנט יעדיף להשתמש בפרוטוקול החדש?

א. לכל $0 < N \leq 15$ עדיף להשתמש בפרוטוקול החדש.

ב. $N = 3$

ג. $N = 7$

ד. $N = 8$

ה. לכל $0 < N \leq 15$ עדיף להשתמש בפרוטוקול הסטנדרטי.



שאלה 8 (5 נקודות):

עליכן לתכנן מכונת מצבים מסוג מילי, אשר תזהה רצפים באורך זוגי גדול מ-0 של 1-ים. עם סיומו של כל רצף זוגי של '1' (כלומר, עם קבלת '0' לאחר רצף של '1') המכונה תוציא '1'.

להלן דוגמה של ההתנהגות הרצויה של המכונה:

Input	0	1	1	1	0	1	1	0	1	0
Output	0	0	0	0	0	0	0	1	0	0

כמה מצבים נדרשים לצורך מימוש מכונה זו?

א. 2

ב. 3

ג. 4

ד. 5

ה. 6



שאלה 9 (5 נקודות):

נתון קטע הקוד הבא:

```
addi    t0, x0, 0x8
slli    t0, t0, 3
add     t0, t0, t0
srli    t0, t0, 2
```

מה ערכו של רגיסטר t0 עם סיום התוכנית?
שימו לב שערכי המספרים בתשובות הינם בבסיס עשרוני.

- א. 4
- ב. 8
- ג. 16
- ד. 32
- ה. 64



שאלה 10 (5 נקודות):

נתון מערך A השמור בזיכרון. כל איבר במערך הוא בן 4 בתיים.
כתובת הבסיס שלו נמצאת ב-t1, ואיברי המערך הם: $A=\{0,1,2,3,4,5\}$.

נתון קוד האסמבלי הבא:

```
Addi t2, x0, 6
Add s1, x0, x0
Loop: beq t2, x0, end
      Lw t3, 0(t1)
      Add s1, s1, t3
      Addi t1, t1, 8
      Addi t2, t2, -2
      J loop
End:
```

מה יהיה הערך של s1 בסוף הריצה?

- א. 0
- ב. 1
- ג. 6
- ד. 5
- ה. 10



שאלה 11 (5 נקודות):

עבור הפקודות הבאות, איזו פקודה ניתן לממש כפקודה אמיתית (לא פסאודו פקודה) במעבד ה-**single cycle RISC-V**?

ניתן לבצע שינויים בבקר, הוספת בוררים וחיוטים במסלול הנתונים של המעבד ולהתאים את זמן המחזור, אך **אסור** לבצע שינויים ביחידות: Register file, ImmGen, Memory, ALU.

א. פקודת `mv2 rd1, rd2, rs` אשר מעתיקה את הערך ברגיסטר `rs` לרגיסטר `rd1` וגם לרגיסטר `rd2`.

ב. פקודת `subi20 rd, rs, imm` אשר מבצעת חיסור בין 32 הביטים של רגיסטר `rs` לבין `immediate` בגודל 20 ביט (שלאחר מכן עובר דרך ה- ImmGen ומתורגם למספר בן 32 סיביות) ושומרת את התוצאה ב-`rd`.

ג. פקודת `cp rs1, rs2` אשר מעתיקה מילה מהזכרון מהכתובת שנתונה ברגיסטר `rs1` לכתובת בזיכרון אשר נתונה ברגיסטר `rs2`.

ד. פקודת `addsub rd, rs1, rs2` אשר מבצעת את הפעולה $reg[rd] = reg[rd] + reg[rs1] - reg[rs2]$ (שומרת את התוצאה לרגיסטר `rd`).

ה. לא ניתן לממש אף פקודה מהפקודות הנ"ל.



שאלה 12 (5 נקודות):

נתונה תכנית אשר מורכבת מ-N פקודות.

בנוסף נתון פילוח פקודות התוכנית אשר בפועל רצות על המעבד לפי סוג:

סוג הפקודה	אחוז הפקודות מתוך סך הפקודות שרצות
R-type	35%
Beq	30%
LW	20%
SW	15%

נתונים זמני המחזור עבור שלוש הארכיטקטורות שנלמדו בקורס:

$$T_{single\ cycle} = 6ns$$

$$T_{multicycle} = 4ns$$

$$T_{pipelined} = 4ns$$

מעבד ה- Pipeline RISC-V בעל הנתונים הבאים:

- למעבד יחידות Forwarding **מלא** בין השלבים WB->Decode, Mem->Exe ו-WB->Exe, כנלמד בקורס.
- למעבד יחידת Hazard detection unit.
- החלטות על ביצוע הקפיצה מתקבלות בשלב ה- Exe.

נתון כי במחצית מפקודות ה-lw מתרחש Load hazard וכי ב-50% מפקודות ה-beq **לא מתבצעת קפיצה** (כלומר ב-50% כן מתבצעת קפיצה).



סמנו את התשובה הנכונה ביותר ביחס לזמן ריצת הקוד על גבי המעבדים השונים. בכל תשובה שם המעבד מייצג את זמן ריצת הקוד עליו.

הניחו שמספר הפקודות N גדול מאוד.

א. $pipelined < multi\ cycle < single\ cycle$

ב. $pipelined < single\ cycle < multi\ cycle$

ג. $pipelined < single\ cycle = multi\ cycle$

ד. $single\ cycle < multi\ cycle < pipelined$

ה. $pipelined = single\ cycle < multi\ cycle$



שאלה 13 (5 נקודות):

המהנדס ריק בחברת MortyCycle הוסיף פקודה חדשה למעבד RISC-V
:Multi Cycle

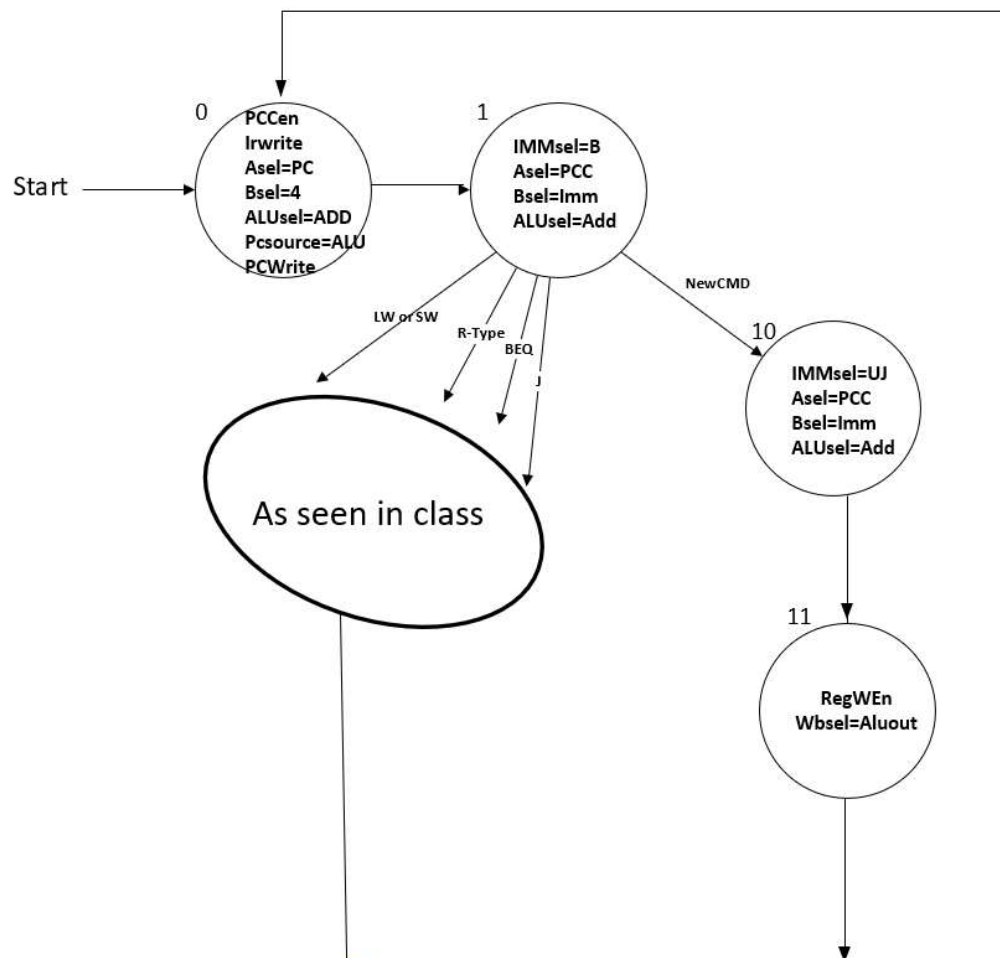
newCMD rd

הפקודה היא בפורמט JJ, ומקודדת כך:

0x000000	rd	opcode
----------	----	--------

על מנת לתמוך בפקודה, המהנדס שינה את מכונת המצבים של המעבד,
והשאיר את ה-Datpath ללא שינוי.

מכונת המצבים המעודכנת הינה:





המהנדס מריץ את קטע הקוד הבא:

0x24: addi t8, x0, 1

0x28: newCMD t8

המהנדס מעוניין גם לדעת האם ניתן לממש את הפקודה החדשה כפסאודו-פקודה של פקודות המכונה עליהן למד בקורס:
Jal, R-type, LW, SW, beq, addi
(תיאור הפקודות מופיע בדפי העזר)

סמנו את התשובה הנכונה:

- א. הערך של t8 לאחר הרצת הקוד הינו 0x1.
ניתן לממש את הפקודה החדשה כפסאודו-פקודה שתתורגם לפקודת מכונה אחת.
- ב. הערך של t8 לאחר הרצת הקוד הינו 0x24.
ניתן לממש את הפקודה החדשה כפסאודו-פקודה שתתורגם לפקודת מכונה אחת.
- ג. הערך של t8 לאחר הרצת הקוד הינו 0x24.
ניתן לממש את הפקודה החדשה כפסאודו-פקודה שתתורגם לשתי פקודות מכונה, ולא ניתן לממש את הפקודה ע"י פחות פקודות מכונה.
- ד. הערך של t8 לאחר הרצת הקוד יהיה 0x28.
ניתן לממש את הפקודה החדשה כפסאודו-פקודה שתתורגם לפקודת מכונה אחת.
- ה. הערך של t8 לאחר הרצת הקוד יהיה 0x28.
ניתן לממש את הפקודה החדשה כפסאודו-פקודה שתתורגם לשתי פקודות מכונה, ולא ניתן לממש את הפקודה ע"י פחות פקודות מכונה.



שאלה 14 (5 נקודות):

מהנדס בונה מערכת משדר מקלט מבוסס UART כנלמד בהרצאות. המשדר מבוסס על מעבד Pipeline-RISCV בעל הנתונים הבאים:

- למעבד יחידות Forwarding **מלא** בין השלבים WB->Decode, Mem->Exe ו-WB->Exe, כנלמד בקורס.
- למעבד יחידת Hazard Detection כנלמד בקורס.
- למעבד קיימת יחידת branch-prediction החוזה את יעד קפיצת פעולות branch כבר בשלב ה-fetch, **והחיזוי שלה תמיד נכון**.

נתון כי המשדר משדר את התוכן של כתובת הזיכרון $0x100$ כל מחזור שעון, **החל מהמחזור לאחר שהערך עודכן**.

בנוסף, נתון כי זמן מחזור השעון הינו $10ns$, וזמן T_{bit} הינו $100ns$.

נתון הקטע קוד הבא:

```
1 Main: addi x5, x0, 0x100 // address of bit-to-send
2       addi x6, x0, 0x300 // address of Array-to-send
3       addi x4, x0, 1 // x4 holds 1 for stopbit
4       sw x0, 0(x5) // Prepare start bit
5       addi x28, x0, 8 // x28: number of data bits left
6       // Insert nops here
7 Start: lw x7, 0(x6) // Load new bit
8       sw x7, 0(x5) // Prepare new bit
9       // Insert nops here
10      addi x6, x6, 4 // Advance to next bit
11      addi x28, x28, -1
12      bne x28, x0, start
13      // Insert nops here
14      sw x4, 0(x5) // Prepare stop bit
15      10 X nop
16      // rest of code
```



בתוכנית ישנם סימונים שבהם יש מקום להוספת Q כדי לעמוד בתנאים (אין חובה להוסיף Q בכל מקום).

רמז – מטרת הכנסת ה- Q היא יצירת T_{bit} באורך תקין.

כמה Q יש להוסיף (לא צריך להתחשב בתשובות ב- 10 ה- Q שמופיעים בשורה 15)?

א. 9

ב. 10

ג. 11

ד. 12

ה. 13



החל מהעמוד הבא מתחיל החלק של
השאלות פתוחות
(שאלות 15 – 17)



שאלה 15 (10 נקודות):

בשאלה הזאת נתון קטע הקוד הבא הכתוב ב assembly עם הכתובת של כל פקודה:

Address	
0x1000	function: li t0, 0
0x1004	addi t1, a0, 0
0x1008	loop: bge t0, a1, end
0x100C	mul a0, a0, t1
0x1010	addi t0, t0, 1
0x1014	jal x0, loop
0x1018	end jr ra

בנוסף נתון כי מאתחלים את הרגיסטרים באופן הבא:

$pc = 1000$

$a0 = 2$

$a1 = 5$

$t0 = 0$

$t1 = 0$

$ra = \text{address of the end of the code}$

א. כמה פעמים הקוד מבצע את הפקודה בכתובת 1010 ?



ב. כעת נתון $a_0=x$, $a_1=y$ בתחילת הריצה.
מה הערך של הרגיסטר a_0 בסוף ריצת הקוד כפונקציה של x ו- y ?

ג. מבצעים את השינוי הבא בקוד: הפקודה בכתובת $0x1014$ מוחלפת
בפקודה `j loop`.

איך השינוי הזה ישפיע על הקוד (התחשבו רק בקוד הנתון)? נמקו.



שאלה 16 (2 נקודות לכל סעיף, ניתן לקבל 10 נקודות לכל היותר):

נתונה מערכת עקיבה (FSM) מסוג מילי בעלת N מצבים (N לא ידוע) עם כניסה יחידה X , ומוצא יחיד Z .

דני ויוסי מצאו סדרות הפרדה שונות המתחילות משני מצבים כלשהם A ו- B של המערכת, עבורן המערכת מוציאה פלט שונה בסוף הסדרה. פלט מכונת המצבים במהלך הסדרות לא ידוע.

הסדרה שדני מצא באורך 12 קלטים והסדרה שיוסי מצא באורך 8 קלטים.

לגבי כל אחד מהמשפטים בטבלה להלן, סמנו X רק בעמודה המתאימה. העמודות מסמלות אם המשפט נכון, לא בהכרח נכון (כלומר, ניתן למצוא דוגמה בה מתקיים המשפט ודוגמה בה הוא לא מתקיים) או בהכרח לא נכון.

	נכון	לא בהכרח נכון	בהכרח לא נכון
1. ייתכן ש- $N=2$			
2. קיימת סדרת הפרדה נוספת, באורך 8 סיביות או קצרה יותר, השונה מסדרתו של יוסי			
3. ייתכן ש-8 הכניסות הראשונות של הסדרה של דני זהות לכניסות של סדרתו של יוסי			
4. ניתן למצוא סדרות הפרדה בכל כפולה של 8 כניסות (סדרה באורך 16, ..., 24)			
5. אם $N=8$ והמכונה מצומצמת, קיימת סדרת הפרדה קצרה יותר מזו של יוסי			



שאלה 17 (10 נקודות):

במעבד Multicycle RISC-V הוחלט לטפל בשני סוגי חריגות בלבד: חלוקה באפס ופקודה לא חוקית, עם הקידוד הבא:

סוג החריגה	קידוד
פקודה לא חוקית	1
חלוקה באפס	2

במקרה של חלוקה באפס צריך לכתוב לרגיסטר המכנה את הערך 1.
במקרה של פקודה לא חוקית צריך לדלג על הפקודה הלא חוקית ולהמשיך לפקודה הבאה.

הניחו שכל הרגיסטרים מסוג s_i , t_i ו- a_i , מאותחלים ל-0 ושהגישה לרגיסטרים SCAUSE ו- SEPC זהה לגישה לרגיסטר מה- RegFile.
התוכנית רצה החל מ- main.

השלימו את הקוד בעמוד הבא כך שירוצ באופן תקין ויתמוך בחריגות הנתונות בשאלה.



0x10000000 main: addi t0, x0, 2

0x10000004 sub t2, t1, t1

0x10000008 unknown instruction

0x1000000C div t3, t0, t2

0x1C090000 interrupt handler: addi sp, sp, _____

0x1C090004 sw s0, 0(sp)

0x1C090008 sw _____, 4(sp)

0x1C09000C addi s0, x0, 1

0x1C090010 addi s1, x0, _____

0x1C090014 beq SCAUSE, s0, _____

0x1C090018 beq SCAUSE, _____, label2

0x1C09001C done: lw s0, 0(sp)

0x1C090020 lw _____, 4(sp)

0x1C090024 addi sp, sp, 8

0x1C090028 jr _____

0x1C091000 label1: addi _____, _____, 4

0x1C091004 j done

0x1C091008 label2: addi t2, x0, _____

0x1C09100C j done

0x1C091008 label: