

## תרגיל בית יבש מספר 4

### שאלה 1

התוכנית NOLW היא תוכנית עם מספר פקודות גדול מאוד שאין בה אף פקודות LW. הוצע לייצר מעבד MULTI CYCLE RISC-V חדש אשר מיועד להרצת התוכנית NOLW בלבד, אשר אינו תומך כלל בפקודות LW.

נגדיר:

$$CPI = \frac{\text{cycles per instructions}}{\text{clock cycles to run the program}} = \frac{\text{number of instructions in the program}}{\text{clock cycles to run the program}}$$

סמנו את המשפט הנכון:

- א. במעבד החדש ניתן לוותר על DISPATCH ROM 2 וייתכן כי שימוש במעבד החדש יקטין את ה-CPI של התוכנית NOLW בהשוואה למעבד המקורי.
- ב. במעבד החדש לא ניתן לוותר על DISPATCH ROM 2 וייתכן כי שימוש במעבד החדש יקטין את ה-CPI של התוכנית NOLW בהשוואה למעבד המקורי.
- ג. במעבד החדש ניתן לוותר על DISPATCH ROM 2 ולא ייתכן כי שימוש במעבד החדש יקטין את ה-CPI של התוכנית NOLW בהשוואה למעבד המקורי.
- ד. במעבד החדש לא ניתן לוותר על DISPATCH ROM 2 ולא ייתכן כי שימוש במעבד החדש יקטין את ה-CPI של התוכנית NOLW בהשוואה למעבד המקורי.
- ה. לא ניתן לדעת האם ה-CPI של התוכנית NOLW יגדל או יקטן בהשוואה למעבד המקורי.

## שאלה 2

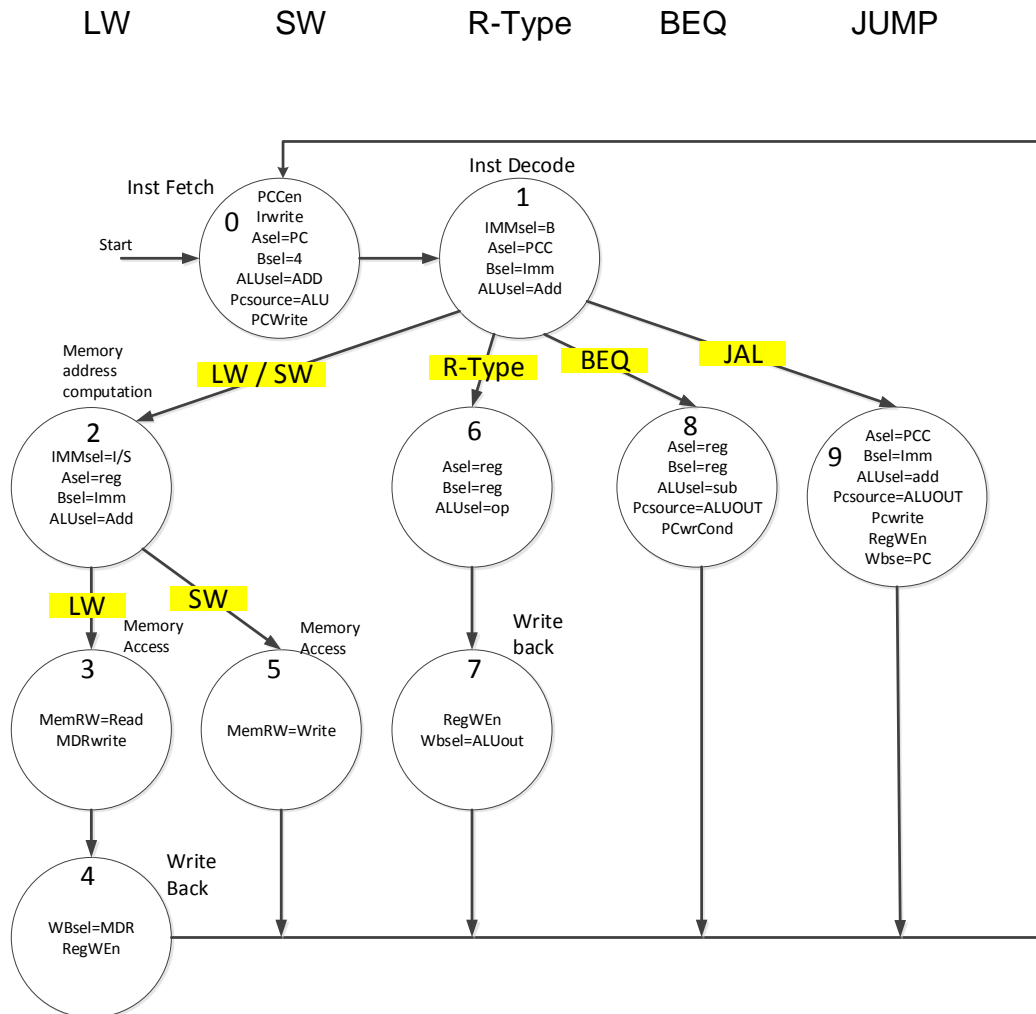
stack overflow הינה חריגה המתרחשת כאשר תוכנית קוד קריאה לשגרה מנסה לייצר frame חדש במחסנית, אבל חורג מגודל המקסימלי המותר למחסנית.

אילו מבין המשפטים הבאים נכון?

- א. שינוי חלוקת האחריות בין השגרה הקוראת והנקראת ימנע היווצרות stack overflow .
- ב. stack overflow מעידה על תקלה בקוד התוכנית.
- ג. stack overflow מעידה על תקלה במעבד.
- ד. stack overflow יכולה להיגרם רק ע"י קריאה רקורסיבית של שגרה לעצמה.
- ה. stack overflow מתגלה ע"י המעבד בעת ביצוע פקודת כתיבה לא חוקית לאזור שחורג מהמקום המוקצה למחסנית.

### שאלה 3

נתון מעבד RISC-V Cycle Multi כפי שנלמד בכיתה. בעת ייצור הבקר קרתה תקלה והקו IRWrite חובר קבוע ל '1' לוגי. הקף בעיגול את הפקודות אשר תמיד יעבדו כהלכה למרות התקלה:



## שאלה 4

נתונים 2 מעבדי Pipeline RISC-V בעלי load hazard detection unit ו-forwarding מלא. נתונה תוכנית בעלת מספר רב של פקודות ובעלת התפלגות הפקודות הדינאמית הבאה (התפלגות הפקודות אשר אכן בוצעו במהלך ריצת התוכנית):

Instruction	Percentage
R-Type	20%
BR	60%
Jump	0%
LW	10%
SW	10%

המעבד מניח כי ה-branch אינו מתרחש ועושה flush במקרה שכן. בתוכנית זו כאשר ישנה פקודת קפיצה מותנית, ישנו סיכוי של 75% שהקפיצה אכן תתבצע. כל פקודות ה-lw גורמות ל-load hazard, וכל פקודות ה-R-Type סובלות מ-Data Hazard.

- זמן מחזור של מעבד א': 2 ns. במעבד א' ה-branch מתבצע בשלב Memory.
- זמן מחזור של מעבד ב': 4 ns. במעבד ב' ה-branch מתבצע בשלב Decode.

בחרו את הטענה הנכונה:

- זמן הרצת התוכנית על מעבד ב' יהיה ארוך יותר מאשר על מעבד א', ויידרשו יותר מחזורי שעון להרצת התוכנית על מעבד ב' מאשר על מעבד א'.
- זמן הרצת התוכנית על מעבד ב' יהיה ארוך יותר מאשר על מעבד א', ויידרשו יותר מחזורי שעון להרצת התוכנית על מעבד א' מאשר על מעבד ב'.
- זמן הרצת התוכנית על מעבד א' יהיה ארוך יותר מאשר על מעבד ב', ויידרשו יותר מחזורי שעון להרצת התוכנית על מעבד ב' מאשר על מעבד א'.
- זמן הרצת התוכנית על מעבד א' יהיה ארוך יותר מאשר על מעבד ב', ויידרשו יותר מחזורי שעון להרצת התוכנית על מעבד א' מאשר על מעבד ב'.
- התוכנית תסיים לרוץ באותו הזמן על שני המעבדים.

## שאלה 5

נתון מחשב Pipeline RISC-V ללא יחידת forwarding וללא forwarding ב- Register File או יחידת hazard Detection. מהו המספר המינימלי של ops שיש להוסיף לתוכנית הבאה כדי שתתבצע נכון?

Sub \$t1, \$t1, \$t1

Add \$t2, \$t3, \$t4

Add \$t5, \$t2, \$t1

Lw \$t2, 0(\$t4)

Add \$t5, \$t2, \$t2

א- 5

ב- 6

ג- 7

ד- 8

ה- 9

# תשובות

## שאלה 1

תשובה ג' נכונה, במעבד החדש ניתן לוותר על DISPATCH ROM 2 ולא ייתכן כי שימוש במעבד החדש יקטין את ה CPI של התכנית NOLW .

ניתן לוותר על DISPATCH ROM 2 מכיוון שבמצב 2) עבור פקודות SW \ LW ( כבר לא קיים פיצול כתוצאה מהסרת התמיכה ב LW .

לא ייתכן כי המעבד החדש יקטין את ה CPI של התכנית NOLW כי הסרת התמיכה ב LW מסירה אך ורק מצבים שבכל מקרה התכנית לא הייתה מגיעה אליהם, ומספר המחזורים שיתבצעו בכל פקודה לא ישתנה.

## שאלה 2

תשובה ה'.

תשובה א' לא נכונה כל שינוי בחלוקת האחריות עדיין יצריך כי אחת מן השגרות תגבה חלק מתוכן – הרגיסטרים במחסנית. לכן, ה- frame של כל שגרה עדיין יהיה גדול מ- 0 ועדיין יכולה להתרחש חרגית stack overflow .

תשובה ב' לא נכונה ייתכן כי את אותו הקוד ניתן יהיה להריץ בצורה תקינה עבור מגבלה אחרת על גודל המחסנית המקסימלי.

תשובה ג' לא נכונה המעבד מבצע כל פקודה כהלכתה, רצף הפקודות הוא זה שגורם לבעיה.

תגובה ד' לא נכונה ניתן לבנות מעגל של קריאות רקורסיביות בין שגרות שונות למשל שגרה A קוראת לשגרה B , אשר קוראת לשגרה A וכו'. רצף כזה גם יכול לגרום לחריגה.

תשובה ה' נכונה מהות התקלה היא כתיבה של ערך מרגיסטר כלשהו לזיכרון במהלך יצירת frame חדש. הכתיבה לזיכרון מתבצעת לכתובת לא חוקית (מחוץ לתחום הכתובות המותר).

## שאלה 3

רק BEQ

משמעות התקלה הינה כי הרגיסטר IR ייכתב בכל מצב של מכונת המצבים (בניגוד לכתיבה אליו רק במצב 0 במעבד תקין). כלומר, הפער בין מעבד תקין למעבד הנתון נוצר בסוף מחזור השעון השני) כלומר, במעבר ממצב 1 למצב הבא.

כלומר, השאלה בעצם הצטמצמה לשאלה הבאה – "עבור אילו פקודות אין שימוש בתוכן ה- IR אחרי מצבים 0 ו-1 (ההחלטה על המעבר למצב הבא תתרחש באופן תקין במצב 1) התשובה לכך היא רק פקודת BEQ אשר במצב 8 (מחזור שעון שלישי) משווה את תוכן שני הרגיסטרים A ו-B ועל פיהם מחליטה אם לבצע את הקפיצה או לא. פקודת jump נעזרת בכתובת השמורה ב-IR. פקודות type-R משתמשות בשדה ה-func אשר שמור ב-IR. פקודות sw/lw נעזרת ב-opcode ע"מ להחליט על הפיצול במצב 2.

## שאלה 4

תשובה ב

כל פעם שישנה פקודת קפיצה, אנו נאלץ לבצע flush.

נבחן את זמן הריצה עבור תוכנית בעל ההתפלגות הנתונה ועבור  $N$  פקודות :  
עבור מעבד א' נבצע flush ל 3 שלבים ב pipe, ונסיים את ריצת התוכנית לאחר  
 $N(1+0.45*3+0.1)=2.45N$  מחזורי שעון. כלומר זמן ביצוע של  $4.9N[ns]$   
עבור מעבד ב' נבצע flush ל שלב יחיד ב pipe, ונסיים את ריצת התוכנית לאחר  
 $N(1+0.45*1+0.1)=1.55N$  מחזורי שעון. כלומר זמן ביצוע של  $6.2N[ns]$

## שאלה 5

תשובה ב'

```
Sub $t1, $t1, $t1
Add $t2, $t3, $t4
Nop
Nop
Nop
Add $t5, $t2, $t1
Lw $t2, ($t4)
Nop
Nop
Nop
Add $t5, $t2, $t2
```