



הטכניון - מכון טכנולוגי לישראל

מבנה מחשבים (236267)

מבחן מסכם מועד א'

11 פברואר 2022

מרצים: ליהוא רפופורט, עדי יועז.
מתרגלים: פרנק סלה, איתי רביד, אנטון חנה.

מס. ת.ז. :

- משך הבחינה: שלוש שעות.
- מותר חומר עזר מודפס ומחשבון בלבד, אסור חומר עזר במחשב ולא בכל אמצעי מקוון אחר.
- יש לכתוב בקיצור ככל האפשר, אך יש לנמק כל תשובה.
- יש לכתוב בכתב ברור וקריא.
- המבחן כולל 5 שאלות, יש לענות על כולן. במבחן 11 עמודים.

שאלה 1	OOO	/ 30
שאלה 2	BP	/ 12
שאלה 3	power	/ 13
שאלה 4	VM	/ 21
שאלה 5	cache	/ 24
ציון סופי		/100

בהצלחה !

שאלה 1 – Out-Of-Order Execution (30 נק')

יש למלא את הטבלה שבהמשך. לכל פקודה יש לרשום:

- | | | |
|---------------------------------------|---|--|
| מולאו בחלקם, יש להשלים את החסר | { | • R3, R2, R1 – ערכי הרגיסטרים הארכיטקטוניים לאחר retire של הפקודה. עבור פקודה ממסלול שגוי: הערך שחושב עבור הרגיסטר הפיזי בזמן execute. |
| | | • addr – כתובת הגישה לזיכרון – עבור פקודות load ו-store בלבד. |
| | | • data – ערך זיכרון שנקרא או נכתב – עבור פקודות load ו-store בלבד. |
| | | |
- T alloc: הזמן בו מבוצעת אלוקציה: עד 3 פקודות בכל מחזור, החל מ- $t=1$. ניתן לבצע אלוקציה רק כאשר לכל הפקודות שמספק ה-frontend במחזור יש מקום ב-ROB. ב-ROB יש 9 כניסות. כל פקודה (כולל פקודות Store) תופסת מקום אחד ב-ROB.
 - src1, src2: מספרי הרגיסטרים המשמשים כ-sources לפקודה: P_i עבור רגיסטר פיזי, ו- R_i במידה וקוראים ישירות את הרגיסטר הארכיטקטוני. עבור store: src1 – הרגיסטר המשמש לחישוב הכתובת. src2 – הרגיסטר המכיל את הנתון.
 - T src1 ready, T src2 ready: הזמן בו מוכן כל אחד ערכי ה-sources לפקודה. אם ה-src כבר מוכן בזמן האלוקציה, אז זמן זה יהיה שווה לזמן האלוקציה. אחרת, זמן זה שווה ל-T data ready של הפקודה שמחשבת את הערך של ה-src.
 - T exe: הזמן בו הפקודה נשלחת לביצוע. הניחו כי ישנן אינסוף יחידות ביצוע.
 - פקודה יכולה להיכנס לביצוע לכל המוקדם במחזור שלאחר האלוקציה.
 - פקודה נכנסת לביצוע במחזור השעון שלאחר המחזור בו כל ה-src-ים מוכנים. פקודת store נכנסת לביצוע במחזור השעון שלאחר המחזור בו src1 (המשמש לחישוב הכתובת) מוכן.
 - Load block code: עבור load שנשלח לביצוע בזמן $t=T_{exe}$, או שהוסר עבורו תנאי חסימה קודם בזמן t , תנאי החסימה נבדקים בזמן $t+1$ לפי הסדר (רישמו את כל תנאי החסימה לפי הסדר):
 1. unknown store address (התנאי מוסר בזמן T addr ready של ה-store החוסם)
 2. waiting for store data (התנאי מוסר בזמן T data ready של ה-store החוסם)
 - T addr ready: ממולא עבור פקודות load ו-store בלבד: $T_{exe}+1$
 - עבור load ו-store המבוצעים באותו זמן t : תנאי החסימה של ה-load נבדקים בזמן $t+1$, בזמן זה הכתובת של ה-store ידועה, ולכן ה-load לא נחסם על unknown store address ע"י ה-store.
 - T data ready
 - עבור פקודות add, sub, conditional jump: $T_{exe}+1$. עבור פקודת mul: $T_{exe}+3$.
 - עבור load שהוסרו עבורו כל תנאי החסימה בזמן t או שבוצע בזמן t ולא נחסם:
 - במידה וה-load פוגע ב-cache, או שיש store to load forwarding: בזמן $t+2$.
 - אחרת, במידה ובוצע load/store לאותה שורה ב-cache בזמן $t' < t$: בזמן $\max(t'+10, t+2)$.
 - אחרת, בזמן $t+10$.
 - עבור store: $T_{src2\ ready} + 1$

- עבור פקודת Jump עם חיזוי שגוי מבוצע flush בזמן $T_{exe}+1$, וכבר במחזור זה אין אלוקציה של פקודות. הפקודות מהמסלול הנכון מבצעות אלוקציה החל מזמן $T_{exe}+4$ (במידה ואין סיבה אחרת שמעכבת את האלוקציה).
- T retire: הזמן בו הפקודה מבצעת retire. ניתן לבצע retire לעד 3 פקודות בכל מחזור. פקודה (למעט store) יכולה לבצע retire החל מזמן $T_{data\ ready}+1$. פקודת store יכולה לבצע retire החל מזמן $\max(T_{addr\ ready}, T_{data\ ready}) + 1$.
- #ROB entries: מספר הכניסות התפוסות ב-ROB מיד לאחר שהפקודה ביצעה אלוקציה. יש להתחשב גם בפקודות שהוצאו מה-ROB. פקודה שמבצעת retire בזמן t מוצאת מה-ROB במחזור $t+1$ וכבר במחזור זה פקודה חדשה יכולה לבצע alloc במקום שהתפנה.
- הנחות:
 - בטבלה רשומות הפקודות שמבצעות אלוקציה, כולל פקודות מהמסלול השגוי.
 - הכתובות הן פיזיות (אין צורך בתרגום). כל הערכים המספריים הם בבסיס 16.
 - L1 data cache: גודל שורה $20_{16}B = 32_{10}B$. write allocate. ריק בתחילת הביצוע.
 - בתחילת הביצוע כתובת N בזיכרון מכילה את הערך N.

Pdst	instruction	R1	R2	R3	addr	data	src1 addr	src2 data	T alloc	#OB entries	T src1 ready	T src2 ready	T exe	T addr ready	T data ready	block code	T retire
1	load R1 \leftarrow m[R2+30]	<u>40</u>	10	50	40	40											
2	Store m[R3+10] \leftarrow R1	40	10	50	60	40											
3	mul R3 \leftarrow R2 \times 2	40	10	<u>20</u>													
4	load R1 \leftarrow m[R3+30]	<u>50</u>	10	20	50	50											
5	load R3 \leftarrow m[R2+50]	50	10	<u>40</u>	60	40											
6	mul R1 \leftarrow R2 \times 2	<u>20</u>	10	40													
7	if (R1>0) PC \leftarrow 3000 חזיון שגוי																
8	Store m[R1+40] \leftarrow R2																
9	load R3 \leftarrow m[R2+45]																
10	mul R2 \leftarrow R3 \times 2																
11	load R1 \leftarrow m[R2-20]																

שאלה 2 – חיזוי קפיצות (12 נק')

נתון חזאי קפיצות מסוג gshare בעל היסטוריה באורך 3, המצביע אל מערך חיזוי בן 8 מונים. כל מונה הוא בן סיבית אחת, ומאותחל ל-0. המונה אליו פונים מחושב ע"י ביצוע XOR בין ההיסטוריה לבין 3 הסיביות התחתונות של הכתובת בה נמצא ה-jump. נתונה סידרת הקפיצות המחזורית הבאה:

ABB ABB ABB, ABB ABB ABB
101 101 **000**, 101 101 **000**, ...

$$\begin{aligned} A \text{ XOR } 0 &= A \\ A \text{ XOR } 1 &= \overline{A} \end{aligned}$$

האיברים בסידרה המודגשים בקו תחתון שייכים ל-jump A, הנמצא בכתובת שהסיביות התחתונות שלה הן 000. שאר האיברים שייכים ל-jump B, הנמצא בכתובת שהסיביות התחתונות שלה הן 111.

א. (6 נק') יש למלא את הטבלה הבאה. היסטוריה לחיזוי – ההיסטוריה כפי שעודכנה ע"י הקפיצה הקודמת ומשמשת לחיזוי הקפיצה הנוכחית, מאותחלת ל-000. מספר המונה – המשמש לחיזוי.

[illegible]

ב. (6 נק') מהו אחוז החיזוי הנכון במצב היציב עבור jump B ? יש להסביר

[illegible]

שאלה 3 – Power/Performance (13 נק')

נתון מעבד שבו 8 ליבות, 4 ליבות גדולות, ו-4 ליבות קטנות. מאפייני שני סוגי הליבות נתונים בטבלאות. ההספק הכולל המוקצה למעבד הוא 15W. ניתן לקבוע את התדר והמתח של כל ליבה באופן עצמאי. ליבות שאינן בשימוש ניתן לכבות לחלוטין.

מתח Volt	תדר GHz Small	תדר GHz Big
0.8	1.5	1.5
0.9	2.0	2.0
1.0		3.0

Small	Big	
0.2W	0.5W	Leakage Power
840pF	1520pF	Cdyn עבור אפליקציה A
3	4	IPC עבור אפליקציה A

א. (6 נק') השלימו את הטבלה הבאה עבור אפליקציה A, השתמשו בשתי ספרות דיוק אחרי הנקודה:

	Small Core				Big Core			
מתח Volt	תדר GHz	power W	IPS Ginst/sec	IPS/W	תדר GHz	power W	IPS Ginst/sec	IPS/W
0.8	1.5				1.5			
0.9	2.0				2.0			
1.0					3.0			

$IPS = IPC \times Freq (GHz)$. $IPS - IPS/W$ מחולק ב-power.

ב. (7 נק') המערכת מריצה מספר תהליכים במקביל, כולם מאפליקציה A הנתונה בטבלה. עבור מספר תהליכים כלשהו, מהו ה-IPS הכולל המקסימלי של המעבד במסגרת ההספק הנתונה? יש להסביר את השיקולים ואת החישוב לבחירת הקונפיגורציה שמשיגה IPS כולל מקסימלי.

שאלה 4 – זיכרון וירטואלי (21 נק')

נתון מעבד דמוי X86 בעל מבנה הכתובת הבא:

63	44	43	36	35	28	27	20	19	12	11	0
sign ext.	PML4	PDP	DIR	PTE	Offset						

במעבד קיים PMH הכולל translation caches עבור כל-אחת מרמות התרגום, כל אחד בגודל כניסה בודדת, ושהגישה אליהם מתבצעת במקביל. כמו-כן במעבד קיימים TLB גדול, ו-data cache גדול בעל גודל שורה של 64 בתים. בכל רמת תרגום, טבלה היא בגודל חצי דף.

א. (3 נקודות) מה מכיל המצביע של כניסה בטבלת ב-DIR: _____

באיזה סיביות של הכתובת הווירטואלית משתמשים בפניה ל-DIR cache: _____

- ב. (14 נקודות) בטבלה שלהלן נתונה סידרת פניות לכתובות וירטואליות בבסיס 16. עבור כל אחת מהפניות ציינו:
- עבור ה-TLB וכל אחד מה-translation caches, האם הניב hit או miss, או שלא ניגשו אליו (NA).
 - מספר הגישות ל-Data Cache שהסתיימו ב-hit ומספר הגישות שהסתיימו ב-miss (עבור התרגום בלבד, לא כולל הפניות להבאת הנתון עצמו).
 - בשורות בהן לא רשומה כתובת, רישמו כתובת שעבורה יתקיימו הנתונים הכתובים בטבלה והשלימו את שאר הנתונים. יש לשנות סיפורה הקסדצימאלית אחת בכתובת ביחס לכתובת שבשורה הקודמת.
 - הניחו שבתחילת הסידרה כל ה-caches ריקים, ושמה שנכתב ל-Data cache נשאר בו לאורך השאלה.

# Data\$ hits	# Data\$ misses	PML\$	PDP\$	DIR\$	TLB	Address (HEX)
						0x0000 0050 4020 1FDC
1				H		
					H	
	2		H	M		
1		M				

ג. (4 נקודות) מהו מספר הדפים הנדרש עבור טבלאות הדפים (בכל הרמות) הנדרשות למיפוי מערך (רציף) בגודל $0x100000$ איברים, כאשר כל איבר במערך הוא בגודל 8 בתים והמערך מתחיל מכתובת $0x00$? הסבירו.

שאלה 5 – זיכרון מטמון (24 נק')

נתון מעבד עם שתי רמות זיכרון מטמון. עיקרון ההכלה לא מתקיים.

L1: גודל 16KB, 4-way set associative, גודל שורה 16 byte, מדיניות כתיבה write back, מדיניות write-allocate, מדיניות פינוי LRU.

L2: גודל 64KB, 16-way set associative, גודל שורה 16 byte, מדיניות כתיבה write back, מדיניות write allocate, מדיניות פינוי LRU.

גודל הכתובת 40 סיביות.

א. (3 נקודות) מהו מבנה הכתובת (tag-set, offset) של L1 ושל L2 ?

מריצים את קטע הקוד הבא על המעבד:

```
int arr[20][256];
For (k=0; k<3; k++)
  For (i=0; i < 20 ; i++)
    For (j=0; j < 256 ; j++)
      sum += arr[i][j];
```

הנחות: המשתנים i, j, k ו- sum שמורים ברגיסטרים.

המספרים בתוכנית הם בבסיס דצימלי.

המטמון ריק בתחילת התוכנית.

המערך arr מתחיל בכתובת 2000H (בסיס הקסדצימלי). גודל משתנה מסוג INTEGER הוא 4 בתים.

ב. (4 נקודות) מהו ה-hit rate ב-L1 עבור כל אחת מהאיטרציות בלולאה החיצונית $K=0/1/2$?

ג. (4 נקודות) מהו מספר הגישות ל-L2 עבור כל אחת מהאיטרציות בלולאה החיצונית $K=0/1/2$?

ד. (4 נקודות) מהו ה-hit rate ב-L2 עבור כל אחת מהאיטרציות בלולאה החיצונית $K=0/1/2$?

בהמשך להרצת קטע הקוד הקודם (כעת המטמון אינו ריק), מריצים את קטע הקוד הבא על המעבד.

```
int arr[20][256]
For (k=0; k<3; k++)
  For (i=19; i >= 0; i--)
    For (j=255; j >= 0; j--)
      sum += arr[i][j];
```

ה. (3 נקודות) מהו ה-hit rate ב-L1 עבור $K=0$?

ו. (3 נקודות) מהו מספר הגישות ל-L2 עבור $K=0$?

ז. (3 נקודות) מהו ה-hit rate ב-L2 עבור $K=0$?
