

BAR-ILAN UNIVERSITY (RA)

Faculty of Engineering

Ramat-Gan 52900, Israel



אוניברסיטת בר-אילן (ע"ר)

הפקולטה להנדסה

רמת-גן 52900

Tel: 03-5317722

engbi@mail.biu.ac.il

מבנה מחשבים ספרתיים

תשע"ט סמסטר ב' מועד ב'

83-301

מרצה: פרופ' שמואל וימר

מתרגל: מר בנימין פרנקל

- יש לקרוא היטב את ההוראות.
- חובה לענות על כל השאלות.
- ציון מקסימלי בבחינה: 100 נקודות.
- יש לנמק את כל תשובותיכם.
- יש להקפיד על כתב יד קריא!
- יש לרשום תשובות בתוך הטבלאות המצורפות במקום שנדרש.
- חומר עזר מותר בשימוש: מחשבון, ספר הקורס ושקפי ההרצאות בלבד! כל חומר עזר אחר אסור בהחלט!
- משך הבחינה: שלוש שעות.
- יש לצרף את שאלוני הבחינה למחברת!

בהצלחה!

שאלה מספר 1 (55 נקודות):

Loop:	LD	R2, 0(R1)
	LD	R4, 0(R3)
	DADD	R2, R2, R4
	DSUB	R4, R4, R2
	SD	R2, 0(R1)
	SD	R4, 0(R3)
	DADDIU	R1, R1, #8
	DADDIU	R3, R3, #8
	BNE	R2, R5, Loop

נתונה התכנית הבאה:

התכנית עוברת על איבריהם של שני מערכים המכילים מספרי integer ומבצעת פעולות אריתמטיות על כל צמד איברים.

נתון מעבד המסוגל להוציא לדרך 2 פקודות במקביל (two-issue processor) וכן תומך ב-dynamic scheduling. הנח כי המעבד מסוגל גם לסיים שתי פקודות מכל סוג שהוא במחזור שרון אחד. במעבד קיימות יחידות ביצוע נפרדות עבור חישוב הכתובת (יחידה אחת), ביצוע פעולות ALU (יחידה אחת) ובדיקת תנאי הקפיצה המותנית (יחידה אחת). נתון כי ביצוע של פעולת ALU אורך מחזור שרון אחד. כמו-כן, נתון כי הזיכרון הינו dual-ported.

- מריצים את התכנית על המעבד הנ"ל כאשר המעבד לא תומך ב-speculation (זאת-אמרת שעובדים על-פי אלגוריתם Tomasulo, אך ללא ROB).

א. מלא את הטבלה הבאה כך שתתאר את התקדמות הפקודות לפי מספר מחזור השרון במהלך ריצת התכנית:

Loop iteration	Instruction	Issue	Execute	Mem access	Write CDB	Comment
1	LD R2, 0(R1)	1	2	3	4	First issue
1	LD R4, 0(R3)	1	3	4	5	Wait for HW unit
1	DADD R2, R2, R4	2	6		7	Wait for LW
1	DSUB R4, R4, R2	2	8		9	Wait for DADD
1	SD R2, 0(R1)	3	4	8		Wait for DADD
1	SD R4, 0(R3)	3	5	10		Wait for DSUB
1	DADDIU R1, R1, #8	4	5		6	Execute directly
1	DADDIU R3, R3, #8	4	7		8	Wait for ALU
1	BNE R2, R5, Loop	5	8			Wait for DADD
2	LD R2, 0(R1)	6	9	10	11	Wait for BNE
2	LD R4, 0(R3)	6	10	11	12	Wait for HW unit
2	DADD R2, R2, R4	7	13		14	Wait for LW
2	DSUB R4, R4, R2	7	15		16	Wait for DADD
2	SD R2, 0(R1)	8	11	15		Wait for DADD
2	SD R4, 0(R3)	8	12	17		Wait for DSUB
2	DADDIU R1, R1, #8	9	10		11	Execute directly
2	DADDIU R3, R3, #8	9	11		12	Wait for ALU
2	BNE R2, R5, Loop	10	15			Wait for DADD
3	LD R2, 0(R1)	11	16	17	18	Wait for BNE
3	LD R4, 0(R3)	11	17	18	19	Wait for HW unit
3	DADD R2, R2, R4	12	20		21	Wait for LW
3	DSUB R4, R4, R2	12	22		23	Wait for DADD
3	SD R2, 0(R1)	13	18	22		Wait for DADD
3	SD R4, 0(R3)	13	19	24		Wait for DSUB
3	DADDIU R1, R1, #8	14	16		17	Wait for BNE
3	DADDIU R3, R3, #8	14	17		18	Wait for ALU
3	BNE R2, R5, Loop	15	22			Wait for DADD

ב. לאיזה ערך שואף ה-CPI עבור מספר גדול של איטרציות?
 בין כל איטרציה לאיטרציה הבאה חולפים 7 מחזורי שעות בהם מתבצעות 9 פקודות, ולכן,

$$\text{עבור מספר גדול של איטרציות ה-CPI שואף ל- } \frac{7}{9} = 0.7778$$

- מהנדס א', מהנדס צעיר בצוות ארכיטקטורה, נדרש לנסות ולשפר את ביצועי המעבד עבור התכנית הנתונה. כמובן, למהנדס ניתנה האפשרות להוסיף חומרה למערכת, ובלבד שהדבר יביא לשיפור בביצועים.
 המהנדס החליט לכפיל את מספר יחידות הביצוע, כך שכעת יש למעבד 2 יחידות עבור חישוב הכתובת, 2 יחידות ALU ו-2 יחידות לבדיקת תנאי הקפיצה המותנית. יחידת הזיכרון נותרה ללא שינוי (dual-ported).

ג. מלא את הטבלה הבאה עבור התכנית שרצה על המעבד על-פי הצעתו של מהנדס א':

Loop iteration	Instruction	Issue	Execute	Mem access	Write CDB	Comment
1	LD R2, 0(R1)	1	2	3	4	First issue
1	LD R4, 0(R3)	1	2	3	4	First issue
1	DADD R2, R2, R4	2	5		6	Wait for LW
1	DSUB R4, R4, R2	2	7		8	Wait for DADD
1	SD R2, 0(R1)	3	4	7		Wait for DADD
1	SD R4, 0(R3)	3	4	9		Wait for DSUB
1	DADDIU R1, R1, #8	4	5		6	Execute directly
1	DADDIU R3, R3, #8	4	6		7	Wait for ALU
1	BNE R2, R5, Loop	5	7			Wait for DADD
2	LD R2, 0(R1)	6	8	9	10	Wait for BNE
2	LD R4, 0(R3)	6	8	10	11	Wait for Mem port
2	DADD R2, R2, R4	7	12		13	Wait for LW
2	DSUB R4, R4, R2	7	14		15	Wait for DADD
2	SD R2, 0(R1)	8	9	14		Wait for DADD
2	SD R4, 0(R3)	8	9	16		Wait for DSUB
2	DADDIU R1, R1, #8	9	10		11	Execute directly
2	DADDIU R3, R3, #8	9	10		12	Wait for CDB
2	BNE R2, R5, Loop	10	14			Wait for DADD
3	LD R2, 0(R1)	11	15	16	17	Wait for BNE
3	LD R4, 0(R3)	11	15	17	18	Wait for Mem port
3	DADD R2, R2, R4	12	19		20	Wait for LW
3	DSUB R4, R4, R2	12	21		22	Wait for DADD
3	SD R2, 0(R1)	13	16	21		Wait for DADD
3	SD R4, 0(R3)	13	16	23		Wait for DSUB
3	DADDIU R1, R1, #8	14	15		16	Execute directly
3	DADDIU R3, R3, #8	14	15		16	Execute directly
3	BNE R2, R5, Loop	15	21			Wait for DADD

ד. לאיזה ערך שואף ה-CPI עבור מספר גדול של איטרציות? האם תוספת החומרה שהוסיף מהנדס א' שיפרה את הביצועים? אם כן, בכמה? אם לא, מדוע?
 בין כל איטרציה לאיטרציה הבאה חולפים 7 מחזורי שעות בהם מתבצעות 9 פקודות, ולכן,
 עבור מספר גדול של איטרציות ה-CPI שואף ל- $\frac{7}{9} = 0.7778$. אין שיפור בביצועים למרות

שנוספו יחידות ביצוע, וזאת משום שכל איטרציה עדיין לוקחת 7 מחזורי שעון (יש עיכוב בגלל mem access, בגלל תלויות מידע ובגלל פקודת ה-BNE).

- מהנדס ב', מאותו צוות ארכיטקטורה של מהנדס א', ניסה לשפר את ביצועי המעבד בדרך אחרת, וזאת על ידי הרחבת החומרה כך שתוכל להוציא לדרך 3 פקודות במקביל (three-issue processor). הנח כי כעת המעבד מסוגל לסיים שלוש פקודות מכל סוג שהוא במחזור שעון אחד, וכן הנח כי הזיכרון כעת הינו triple-ported. שים-לב, כי על-פי הצעתו של מהנדס ב' ישנה רק יחידת ביצוע אחת מכל סוג, כמו במעבד המקורי המתואר בתחילת השאלה.

ה. מלא את הטבלה הבאה עבור התכנית שרצה על המעבד בהתאם להצעתו של מהנדס ב':

Loop iteration	Instruction		Issue	Execute	Mem access	Write CDB	Comment
1	LD	R2, 0(R1)	1	2	3	4	First issue
1	LD	R4, 0(R3)	1	3	4	5	Wait for HW unit
1	DADD	R2, R2, R4	1	6		7	Wait for LW
1	DSUB	R4, R4, R2	2	8		9	Wait for DADD
1	SD	R2, 0(R1)	2	4	8		Wait for DADD
1	SD	R4, 0(R3)	2	5	10		Wait for DSUB
1	DADDIU	R1, R1, #8	3	4		5	Execute directly
1	DADDIU	R3, R3, #8	3	5		6	Wait for ALU
1	BNE	R2, R5, Loop	3	8			Wait for DADD
2	LD	R2, 0(R1)	4	9	10	11	Wait for BNE
2	LD	R4, 0(R3)	4	10	11	12	Wait for HW unit
2	DADD	R2, R2, R4	4	13		14	Wait for LW
2	DSUB	R4, R4, R2	5	15		16	Wait for DADD
2	SD	R2, 0(R1)	5	11	15		Wait for DADD
2	SD	R4, 0(R3)	5	12	17		Wait for DSUB
2	DADDIU	R1, R1, #8	6	9		10	Wait for BNE
2	DADDIU	R3, R3, #8	6	10		11	Wait for ALU
2	BNE	R2, R5, Loop	6	15			Wait for DADD
3	LD	R2, 0(R1)	7	16	17	18	Wait for BNE
3	LD	R4, 0(R3)	7	17	18	19	Wait for HW unit
3	DADD	R2, R2, R4	7	20		21	Wait for LW
3	DSUB	R4, R4, R2	8	22		23	Wait for DADD
3	SD	R2, 0(R1)	8	18	22		Wait for DADD
3	SD	R4, 0(R3)	8	19	24		Wait for DSUB
3	DADDIU	R1, R1, #8	9	16		17	Wait for BNE
3	DADDIU	R3, R3, #8	9	17		18	Wait for ALU
3	BNE	R2, R5, Loop	9	22			Wait for DADD

- לאיזה ערך שואף ה-CPI עבור מספר גדול של איטרציות? האם תוספת החומרה שהוסיף מהנדס ב' שיפרה את הביצועים? אם כן, בכמה? אם לא, מדוע?
בין כל איטרציה לאיטרציה הבאה חולפים 7 מחזורי שעון בהם מתבצעות 9 פקודות, ולכן עבור מספר גדול של איטרציות ה-CPI שואף ל- $\frac{7}{9} = 0.7778$. אין שיפור ביצועים למרות שנוספה חומרה, וזאת משום שכל איטרציה עדיין לוקחת 7 מחזורי שעון (עדיין יש עיכוב בגלל תלויות מידע ובגלל פקודת ה-BNE).

- מהנדס ג', בוגר מצטיין של הפקולטה להנדסה באוניברסיטת בר-אילן, הציע למהנדס ב' להוסיף למעבד תמיכה ב- speculation (עם ROB), וזאת בנוסף לשיפורי החומרה שמהנדס ב' כבר הוסיף (three-issue processor, triple-ported memory). שיים-לב, כי על-פי הצעה זו ישנה רק יחידת ביצוע אחת מכל סוג, כמו במעבד המקורי המתואר בתחילת השאלה.

ז. מלא את הטבלה הבאה עבור התכנית שרצה על המעבד בהתאם להצעתו של מהנדס ג'.

Loop iteration	Instruction	Issue	Execute	Read access	Write CDB	Commit	Comment
1	LD R2, 0(R1)	1	2	3	4	5	First issue
1	LD R4, 0(R3)	1	3	4	5	6	Wait for HW unit
1	DADD R2, R2, R4	1	6		7	8	Wait for LW
1	DSUB R4, R4, R2	2	8		9	10	Wait for DADD
1	SD R2, 0(R1)	2	4			10	Commit in order
1	SD R4, 0(R3)	2	5			10	Commit in order
1	DADDIU R1, R1, #8	3	4		5	11	Commit in order
1	DADDIU R3, R3, #8	3	5		6	11	Wait for ALU
1	BNE R2, R5, Loop	3	8			11	Wait for DADD
2	LD R2, 0(R1)	4	6	7	8	12	Wait for HW unit
2	LD R4, 0(R3)	4	7	8	9	12	Wait for HW unit
2	DADD R2, R2, R4	4	10		11	12	Wait for LW
2	DSUB R4, R4, R2	5	12		13	14	Wait for DADD
2	SD R2, 0(R1)	5	8			14	Wait for HW unit
2	SD R4, 0(R3)	5	9			14	Wait for HW unit
2	DADDIU R1, R1, #8	6	7		8	15	Commit in order
2	DADDIU R3, R3, #8	6	9		10	15	Wait for ALU
2	BNE R2, R5, Loop	6	12			15	Wait for DADD
3	LD R2, 0(R1)	7	10	11	12	16	Wait for HW unit
3	LD R4, 0(R3)	7	11	12	13	16	Wait for HW unit
3	DADD R2, R2, R4	7	14		15	16	Wait for LW
3	DSUB R4, R4, R2	8	16		17	18	Wait for DADD
3	SD R2, 0(R1)	8	12			18	Wait for HW unit
3	SD R4, 0(R3)	8	13			18	Wait for HW unit
3	DADDIU R1, R1, #8	9	11		12	19	Wait for ALU
3	DADDIU R3, R3, #8	9	13		14	19	Wait for ALU
3	BNE R2, R5, Loop	9	16			19	Wait for DADD

ח. לאיזה ערך שואף ה- CPI עבור מספר גדול של איטרציות? האם השינוי שהציע מהנדס ג' תרם לשיפור הביצועים של המעבד? אם כן, בכמה? אם לא, מדוע?

בין כל איטרציה לאיטרציה הבאה חולפים 4 מחזורי שעון בהם מתבצעות 9 פקודות, ולכן, עבור מספר גדול של איטרציות ה- CPI שואף ל- $\frac{4}{9} = 0.4444$. יש שיפור ביצועים, וזאת משום שכבר אין עיכוב בגלל פקודת ה- BNE).

שאלה מספר 2 (55 נקודות):

בשאלה זו נעסוק בתכנון וניתוח ביצועים של מערכות זיכרון.

- נתון מעבד עם מערכת זיכרון בעלת רמה אחת של זיכרון מטמון (L1). נתון כי זמן הגישה הממוצע לזיכרון הוא 2.4 מחזורי שעון, כאשר:
 - אם המידע נמצא בזיכרון המטמון אזי זמן הגישה לזיכרון הוא מחזורי שעון יחיד,
 - אם המידע לא נמצא בזיכרון המטמון אזי נדרשים עוד 80 מחזורי שעון על מנת להביא את המידע מהזיכרון הראשי (main memory).
- צוות מהנדסים מנסה לשפר את זמן הגישה הממוצע לזיכרון, במטרה להשיג שיפור של 65% בזמן הגישה הממוצע ($speedup=1.65$). צוות המהנדסים החליט להוסיף רמה נוספת של זיכרון מטמון (L2), כאשר נתון כי:
 - זמן הגישה ל-L2 הינו 6 מחזורי שעון.
 - הוספת ההיררכיה של L2 למערכת הזיכרון איננה משפיעה כלל על רצף הגישות ל-L1 או על זמני הגישות ל-L1.
 - זמן הגישה לזיכרון הראשי נותר כשהיה (80 מחזורי שעון).
- א. מה צריך להיות ה- hit rate של L2 על מנת להשיג את השיפור הרצוי ב- AMAT?

ראשית, עלינו למצוא מהו ה- miss rate של L1:

$$AMAT = HitTime + MissRate_{L1} \times MissPenalty$$

$$2.4 = 1 + MissRate_{L1} \times 80$$

$$MissRate_{L1} = 1.75\%$$

בשלב הבא, נרצה לחשב מהו ה- AMAT שעלינו לחתור אליו על מנת להשיג את השיפור הנדרש בשאלה:

$$Speedup = \frac{Time_{old}}{Time_{new}}$$

$$1.65 = \frac{2.4}{Time_{new}}$$

$$Time_{new} = 1.4545 \text{ [clock cycles]}$$

כעת, נוכל להשתמש שוב בנוסחת ה- AMAT על מנת למצוא את ה- miss rate של L2 שניב את השיפור הרצוי:

$$1.4545 = 1 + 0.0175 \times (6 + MissRate_{L2} \times 80)$$

מפתרון של המשוואה הנ"ל עולה כי ה- miss rate הגבוהה ביותר האפשרי עבור L2 הוא:
 ≈ 0.2497 , ולכן, ה- hit rate של L2 צריך להיות לפחות: $\sim 75\%$

- נתונה מערכת בעלת מעבד יחיד (single core system). המעבד הינו מצונר (pipelined), כך שה- CPI האידיאלי שלו הינו מחזורי שעון יחיד. שים-לב, כי ה- CPI האידיאלי איננו כולל את התקורה (overhead) הנובעת מהחטאות בזיכרון המטמון (cache misses).
על המערכת הנ"ל הריצו benchmark מסוים, כאשר מערכת הזיכרון של המעבד הכילה L1 cache. מתוך ניתוח ריצת ה- benchmark עולה כי עבור כל 10,000,000 גישות לזיכרון המטמון היו 308,752 החטאות (L1 cache misses), כאשר:

- אם המידע נמצא בזיכרון המטמון אזי זמן הגישה לזיכרון הוא מחזורי שעות יחיד, ולא נוצר שום עיכוב ב-pipeline.
 - אם המידע לא נמצא בזיכרון המטמון אזי נדרשים עוד 10 מחזורי שעות על מנת להביא את המידע.
- ראש צוות הארכיטקטורה בחברה שבה משתמשים במערכת המתוארת לעיל (single core system) קיבל החלטה להחליף את המערכת הנ"ל במערכת מרובת ליבות (multi-core system). במערכת החדשה לכל core יש זיכרון מטמון (L1) משלו, כאשר נתון כי:
- כל הליבות (cores) מתייחסות לזיכרון מרכזי משותף אחד,
 - התנגשויות פוטנציאליות עבור נתונים משותפים מטופלות בשיטת snooping ובפרוטוקול MSI עבור cache coherence.
- על המערכת החדשה הריצו את אותו benchmark שבו השתמשו עבור ניתוח ביצועי המערכת הישנה, ומתוך ניתוח ריצת ה-benchmark עולה כי כעת ישנם 452,977 החטאות (L1 cache misses) עבור כל 10,000,000 גישות לזיכרון המטמון, כאשר:
- אם המידע נמצא בזיכרון המטמון אזי זמן הגישה לזיכרון נותר מחזורי שעות יחיד.
 - אם המידע לא נמצא בזיכרון המטמון אזי נדרשים עוד 14 מחזורי שעות בממוצע על מנת לטפל ב-L1 cache miss שאירע.
- ב. מה חייב להיות ערכו של ה-CPI האידאלי של ה-multi-core system על מנת שהחלטתו של ראש צוות הארכיטקטורה (לעבור מה-single core system ל-multi-core system) תהיה כדאית מבחינת ביצועים? האם הדבר אפשרי? הסבר.

ראשית, נרצה לחשב את ה-CPI של המערכת בעלת מעבד יחיד. לשם כך, נחשב את התקורה הנובעת מהחטאות בגישה לזיכרון L1, ונקבל כי:

$$CPI = 1 + \left(\frac{308,752}{10,000,000} \times 10 \right) = 1.308752$$

על מנת שההחלטה לעבור ל-multi-core system תהיה כדאית מבחינת ביצועים, עלינו לדאוג שיתקיים:

$$1.308752 > X + \left(\frac{452,977}{10,000,000} \times 14 \right)$$

$$1.308752 > X + 0.6341678$$

$$0.6745842 > X$$

ולכן, ה-CPI האידאלי צריך להיות קטן מ-0.6745842. הדרישה הזאת אפשרית בהחלט, שהרי מדובר במערכת מרובת ליבות העובדות במקביל, ולכן אפשרי להשיג CPI שכזה.

- נדרש לתכנן זיכרון מטמון היררכי. על זיכרון המטמון להיות בעל **Inclusive policy**. נתון כי גודלו של כל בלוק בזיכרון הינו 16B. הוחלט כי ה-L1 cache יהיה בגודל של 32KB, משום שזהו הגודל המקסימלי של היררכיה בזיכרון המטמון שאליה ניתן לגשת במחזורי שעות יחיד. לדוגמא, אם היו מוסיפים ל-L1 עוד בלוק אחד נוסף הרי שכבר היו נדרשים שני מחזורי שעות על מנת לגשת ל-L1.
- כעת, רוצים להוסיף היררכיית זיכרון מטמון נוספת (L2), כאשר נתון כי:
 - זמן הגישה להיררכיה בזיכרון המטמון הינו לינארי בגודל ההיררכיה, כך שזמן הגישה להיררכיה בגודל $32KB \times n$ הינו n מחזורי שעות. כמובן, שאם n איננו מספר שלם, הרי שמספר מחזורי השעות הנדרש לצורך גישה להיררכיה הוא $\lceil n \rceil$.
 - ה-Miss Rate של ההיררכיה ה- i בזיכרון המטמון נתון ע"י הביטוי:

$$MissRate_{L_i} = \frac{i \times 0.2 \times 32K}{|L_i| - |L_{i-1}|}$$

כאשר $|L_i|$ הינו מספר הבתים (Bytes) בהיררכיה ה- i , וכן מוגדר כי: $|L_0| = 0$
 ○ זמן הגישה ל- main memory הינו 100 מחזורי שעון.

ג. עליך לקבוע את $|L_2|$ כך שיתקבל AMAT מינימלי. לשם כך, אין הגבלה על כמות החומרה.
 פרט ונמק את צעריך. רשום את ערכו המדויק של ה- AMAT המינימלי המתקבל עבור $|L_2|$ שמצאת.

ראשית, נחשב את ה- Miss rate של כל אחת מההיררכיות, וכן את זמן הגישה לכל היררכיה:

$$MissRate_{L_1} = \frac{1 \times 0.2 \times 32K}{|L_1| - |L_0|} = \frac{1 \times 0.2 \times 32K}{32K - 0} = 0.2$$

$$MissRate_{L_2} = \frac{2 \times 0.2 \times 32K}{|L_2| - |L_1|} = \frac{0.4 \times 32K}{|L_2| - 32K}$$

$$T_{L_1} = \left\lceil \frac{32K}{32K} \right\rceil = 1$$

$$T_{L_2} = \left\lceil \frac{|L_2|}{32K} \right\rceil$$

הביטוי הכללי של זמן הגישה הממוצע לזיכרון הוא:

$$AMAT = T_{L_1} + MR_{L_1} \times T_{L_2} + MR_{L_1} \times MR_{L_2} \times T_{MM}$$

$$AMAT = 1 + 0.2 \times \left\lceil \frac{|L_2|}{32K} \right\rceil + 0.2 \times \frac{0.4 \times 32K \times 100}{|L_2| - 32K} \quad \text{נציב ונקבל משוואה עם נעלם אחד:}$$

זוהי פונקציה שאינה רציפה, משום שהביטוי של T_{L_2} תלוי בגודל של L_2 באופן לא רציף. בנוסף, גם $|L_2|$ בעצמו איננו משתנה רציף, שהרי הוא חייב להיות כפולה שלמה של 16 (גודל בלוק).
 בשלב הראשון נפתור את הבעיה הרציפה ונניח כאילו T_{L_2} תלוי בגודל של L_2 באופן רציף, וכן כי $|L_2|$ הינו משתנה רציף. לאחר מכן, נתחשב בעובדה כי המציאות מנציבה פונקציה לא רציפה.
 נגזור את הפונקציה הרציפה ונשווה את הנגזרת לאפס על מנת למצוא מינימום:

$$\frac{\partial AMAT}{\partial |L_2|} = \frac{0.2}{32K} - 0.2 \times \frac{0.4 \times 32K \times 100}{(|L_2| - 32K)^2}$$

$$\frac{\partial AMAT}{\partial |L_2|} = 0$$

$$\frac{0.2}{32K} = 0.2 \times \frac{0.4 \times 32K \times 100}{(|L_2| - 32K)^2}$$

$$(|L_2| - 32K)^2 = (40 \times 32K) 32K$$

$$|L_2| = \sqrt{40} \times 32K + 32K = 240011.0287$$

זוהי נקודת האופטימום של הפונקציה הרציפה. כמובן, שאין כל אפשרות להגדיר היררכיה של זיכרון עם שברי בתים, וגם לא עם שברי בלוקים, ולכן, נמצא את מספר הבלוקים השלם הקרוב לערך הזה. אם נעגל כלפי מטה נקבל:

$$\left\lfloor \frac{240011.0287}{16} \right\rfloor = 15000 [\text{Blocks}]$$

$$|L_2| = 240000 [\text{Bytes}]$$

$$AMAT = 1 + 0.2 \times \frac{240000}{32K} + 0.2 \times \frac{0.4 \times 32K \times 100}{240000 - 32K} = 3.729822132$$

אם נעגל את מספר הבלוקים כלפי מעלה נקבל תוצאה אפילו קצת יותר טובה:

$$\left\lceil \frac{240011.0287}{16} \right\rceil = 15001 [\text{Blocks}]$$

$$|L_2| = 240016 [\text{Bytes}]$$

$$AMAT = 1 + 0.2 \times \frac{240016}{32K} + 0.2 \times \frac{0.4 \times 32K \times 100}{240016 - 32K} = 3.729822129$$

זה יכול היה להיות ה-AMAT אם T_{L2} היה יכול להיות מספר לא שלם של מחזורי שעון, אך מכיוון

שבפועל אנחנו משלמים רק מספר שלם של מחזורי שעון, הרי שעבור $|L_2| = 240016 [\text{Bytes}]$

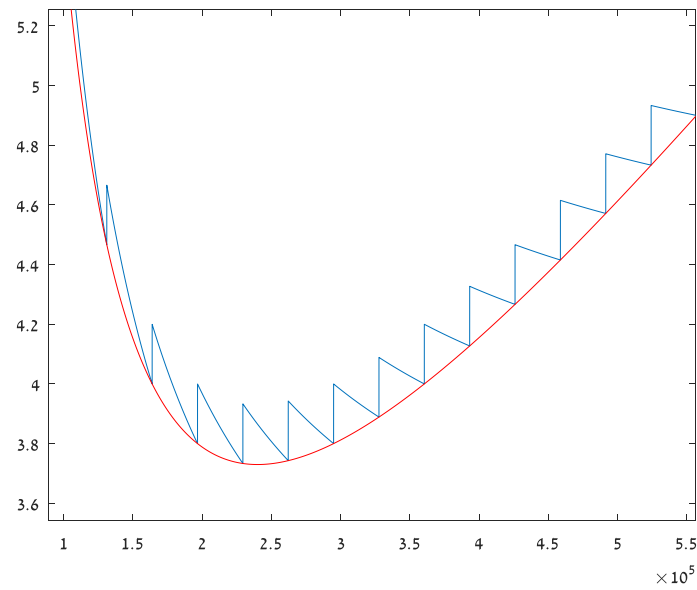
אנחנו נשלם 8 מחזורי שעון.

$$AMAT = 1 + 0.2 \times \underbrace{\left\lceil \frac{240016}{32K} \right\rceil}_{=8} + 0.2 \times \frac{0.4 \times 32K \times 100}{240016 - 32K} = 3.86488$$

מובן כי אין כל היגיון לקבוע את $|L_2| = 240016 [\text{Bytes}]$, שהרי באותו מחיר (של 8 מחזורי שעון)

אנחנו יכולים להוסיף עוד 1383 בלוקים, שהם 22128 בתים, ובכך להקטין את ה-miss rate של L2 וזאת מבלי לשלם שום מחיר מבחינת זמן הגישה ל-L2.

נדרש להבין כי תנאי הבעיה מגדירים פונקציה דיסקרטית (מתוארת בגרף בכחול), אשר ערכה מתלכד עם ערך הפונקציה הרציפה (מתוארת בגרף באדום) במקומות שבהם $|L_2|$ הינו כפולה שלמה של 32K:



לכן, כל שנותר לעשות הוא רק לבדוק איזו אפשרות מבין שתי האפשרויות שהן כפולה שלמה של 32K הסמוכות לנקודת המינימום של הפונקציה הרציפה נותן AMAT טוב יותר:

$$AMAT = 1 + 0.2 \times \left\lceil \frac{8 \times 32K}{32K} \right\rceil + 0.2 \times \frac{0.4 \times 32K \times 100}{8 \times 32K - 32K} = 3.742857$$

$$AMAT = 1 + 0.2 \times \left\lceil \frac{7 \times 32K}{32K} \right\rceil + 0.2 \times \frac{0.4 \times 32K \times 100}{7 \times 32K - 32K} = 3.7333333$$

ולכן, נקבע את $|L_2|$ להיות $7 \times 32K$, ובכך נשיג: $AMAT = 3.7333333[\text{cycles}]$