



ניהול היררכיית זיכרון

- הקדמה
- מיפוי ישיר
- מיפוי אסוציאטיבי מלא
- מיפוי אסוציאטיבי חלקי

ניהול היררכיית זיכרון - סיכום



ניהול היררכיית זיכרון נועד כדי להרוויח את המהירות של זיכרונות נדיפים מסוג SRAM מבלי לשלם מחיר גבוה על נפח גדול של זיכרון מסוג זה אלא על ידי שימוש בעיקר בזיכרון מסוג DRAM שהוא אמנם איטי בצורה משמעותית אך גם זול מאוד.

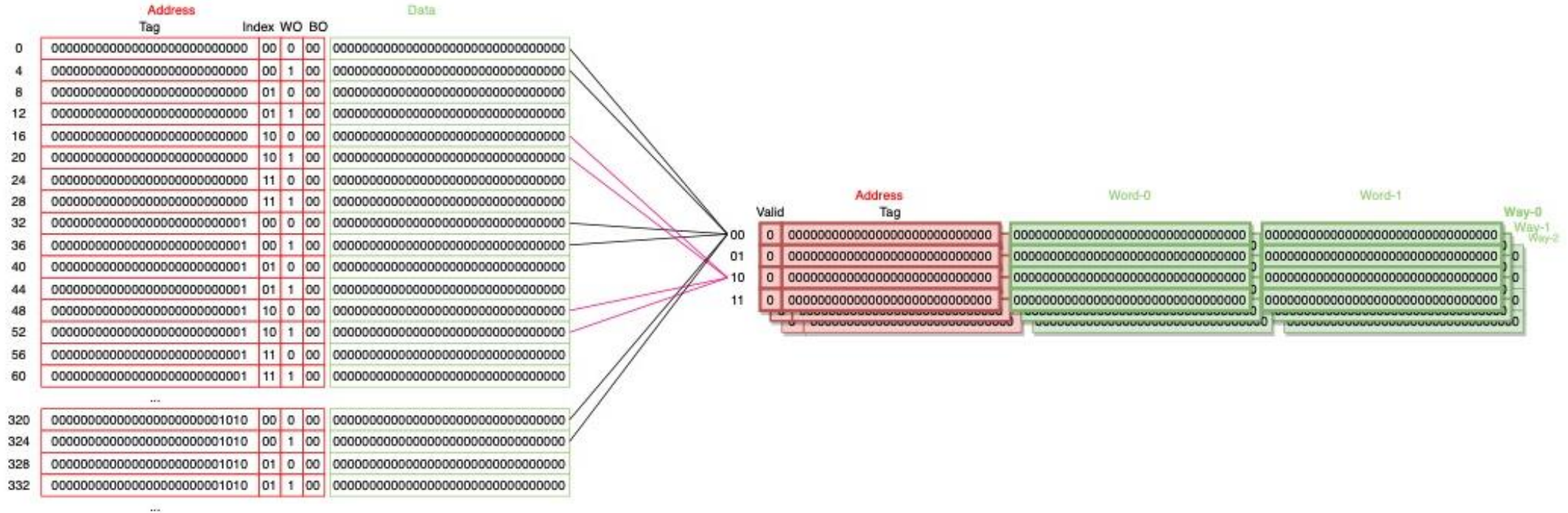
מושגים שונים בניהול היררכיית זיכרון			
גישה של המעבד לזיכרון המטמון ומציאת הנתון המבוקש	פגיעה	Hit	מושגי יסוד
גישה של המעבד לזיכרון המטמון וגילוי שהנתון המבוקש לא נמצא בו	החטאה	Miss	
הזמן שלוקח לגשת לזיכרון המטמון ולקבל ממנו נתון שנמצא בו	זמן פגיעה	Hit Time	
הזמן שלוקח לגשת לזיכרון המטמון, לגלות שאין בו את הנתון המבוקש, לחכות שהזיכרון יפנה לרמה שמתחתיו ויביא את הנתון המבוקש אליו וקבלת הנתון המבוקש אל המעבד	זמן החטאה	Miss Penalty	
אחוז הפעמים שהמעבד פונה לזיכרון המטמון ומוצא בו את הנתון המבוקש	שיעור הפגיעה	Hit Rate	
אחוז הפעמים שהמעבד פונה לזיכרון המטמון ולא מוצא בו את הנתון המבוקש ($Hit Rate + Miss Rate = 1$)	שיעור החטאה	Miss Rate	
לרוב, בלוק יהיה בגודל של מילה אחת / 2 מילים / 4 מילים / 8 מילים וכן הלאה	חבילה של נתונים בגודל כלשהו	Block	
במיפוי אסוציאטיבי חלקי, שדה האינדקס של הכתובת מהזיכרון הראשי יקבע לאיזה סט הכתובת תיכנס	כניסה לשורה ספציפית בזיכרון	Set	
במיפוי אסוציאטיבי חלקי, לאחר שמיפוינו כתובת כלשהי לסט מסוים, הנתון יוכל להישמר בכל אחד מה-Wayים שיש באותו סט	מקום בתוך הסט שאפשר לשמור בו בלוק	Way	
הזמן הממוצע שלוקח להביא נתון מהזיכרון למעבד: $AMAT = Hit Time + Miss Rate \cdot Miss Penalty$	זמן ממוצע לגישה לזיכרון	AMAT	
החטאה שנגרמה כתוצאה מכך שהתא עדיין ריק וסיבית ה-Valid שלו שווה 0	החטאה כי התא ריק	Miss Valid	סיבות להחטאה
החטאה שנגרמה כתוצאה מכך שהתא מלא (סיבית ה-Valid שלו שווה 1), אבל שדה ה-Tag שנמצא בו לא זהה לשדה ה-Tag של הכתובת המבוקשת	החטאה כי התא מלא במשהו אחר	Miss Tag (Miss Conflict)	
שיטת כתיבה לזיכרון הראשי כך שבכל פעם שהמעבד כותב משהו לזיכרון המטמון המידע מועבר מיידית גם לזיכרון הראשי (על פי רוב משתמשים ב-Buffer כדי לא להשהות את המעבד סתם)	כתיבה מיידית	Write Through	שיטות כתיבה
שיטת כתיבה לזיכרון הראשי כך שבכל פעם שהמעבד כותב משהו לזיכרון המטמון הבלוק שנכתב אליו מסומן ב-Dirty Bit, ורק כאשר ינסו להחליף את הבלוק בבלוק אחר, המידע שיש בו ייכתב לזיכרון הראשי	כתיבה רק בהחלפת בלוק	Write Back	
שיטת מיפוי לפיה כל בלוק מהזיכרון הראשי יכול להיות במיקום אחד בלבד בזיכרון המטמון. שיטה שקלה מאוד לניהול ולשליפת נתונים מזיכרון המטמון אך עלולה לגרום להרבה Miss Tag	מיפוי ישיר	DMC - Direct Mapped Cache	שיטות מיפוי
שיטת מיפוי לפיה כל בלוק מהזיכרון הראשי יכול להיות בכל מקום בזיכרון המטמון. שיטה זו אמנם מאפשרת להשתמש בכל מרחב הכתובות בזיכרון המטמון לפני שניתקל ב-Miss Tag הראשון, אך בפועל קשה מאוד לנהל אותה ויש לשלם מחיר כבד מידי של זמן שליפה ארוך וחומרה מסובכת ויקרה	מיפוי אסוציאטיבי מלא	Fully Associative Cache	
שיטת מיפוי לפיה כל בלוק מהזיכרון הראשי יכול להיות במיקום אחד בלבד בזיכרון המטמון אך בתוך כל מיקום כזה יש מספר מקומות פנויים ואפשר שיהיו בתוך הסט הזה כמה בלוקים שונים בעת ובעונה אחת	מיפוי אסוציאטיבי חלקי	K Way Set Associative Cache	
	פינוי אקראי	Random	שיטות פינוי זיכרון אסוציאטיבי
	פינוי הבלוק האחרון שהיה בשימוש	LRU	
	פינוי נכנס ראשון יוצא ראשון	FIFO	
	פינוי נכנס ראשון יוצא ראשון בתנאי שלא היה בשימוש לאחרונה	NMRU	

ניהול היררכיית זיכרון - סיכום



נוסחאות חישוב					נדרש למצוא
נוסחות					
000000000000000000000000000000	0000	00	00		
Tag N-n-m-2	Index n=?	Block Offset m=?	Byte Offset		
נסמן את כמות הביטים שיש בכתובת ב-N, כמות המילים שיש בזיכרון הראשי הינה: 2^{N-2}					כמות מילים בזיכרון הראשי
נסמן את כמות הביטים שיש בכתובת ב-N, כמות הבתים שיש בזיכרון הראשי הינה: 2^N					כמות בתים בזיכרון הראשי
נסמן את כמות הביטים שיש בכתובת ב-N, כמות הביטים שיש בזיכרון הראשי הינה: $2^N \cdot 8$ או $2^{N-2} \cdot 32$					כמות ביטים בזיכרון הראשי
כמות הביטים באינדקס היא n, וכמות הביטים ב-Word Offset היא m, אם יש מילה אחת בבלוק 2^n מילים, אם יש מספא מילים בבלוק 2^{n+m}					כמות המילים בזיכרון המטמון במיפוי ישיר
אם יש מילה אחת בבלוק 2^{n+2} מילים, אם יש מספר מילים בבלוק 2^{n+m+2}					כמות הבתים בזיכרון המטמון במיפוי ישיר
אם יש מילה אחת בבלוק $k \cdot 2^n$ מילים, אם יש מספא מילים בבלוק $k \cdot 2^{n+m}$					כמות המילים בזיכרון המטמון אסוציאטיבי עם k way בכל סט
אם יש מילה אחת בבלוק $k \cdot 2^{n+2}$ מילים, אם יש מספר מילים בבלוק $k \cdot 2^{n+m+2}$					כמות הבתים בזיכרון המטמון אסוציאטיבי עם k way בכל סט
$k \cdot 2^n \cdot (32 \cdot 2^m + valid + dirty + Tag) = k \cdot 2^n \cdot (32 \cdot 2^m + 1 + 1 + Tag)$ יש לבדוק אם יש בזיכרון סיבית dirty, אם לא, להוריד 1					כמות הסיביות שיש במטמון באופן כללי (כולל הסיביות של המידע ושל המיפוי)

ניהול היררכיית זיכרון - סיכום



ניהול היררכיית זיכרון - סיכום

