

אוניברסיטת תל-אביב

מדעי המחשב

סמסטר א , תשס"ה

מבחן במבנה המחשב

מועד א'

המבחן כולל 10 עמודים (כולל עמוד זה) עם 5 שאלות,

הנחיות מיוחדות:

עליך לענות על * כל * השאלות !

יש לענות על השאלות על גבי טופס המבחן בלבד.

המחברות הן לטייטה בלבד ולא תיבדקנה.

משך המבחן: 3 שעות

מלא/י את תעודת הזהות בכל העמודים!

פרופ' יהודה אפק, פרופ' נתן אינטרטור, מר אורי שלו

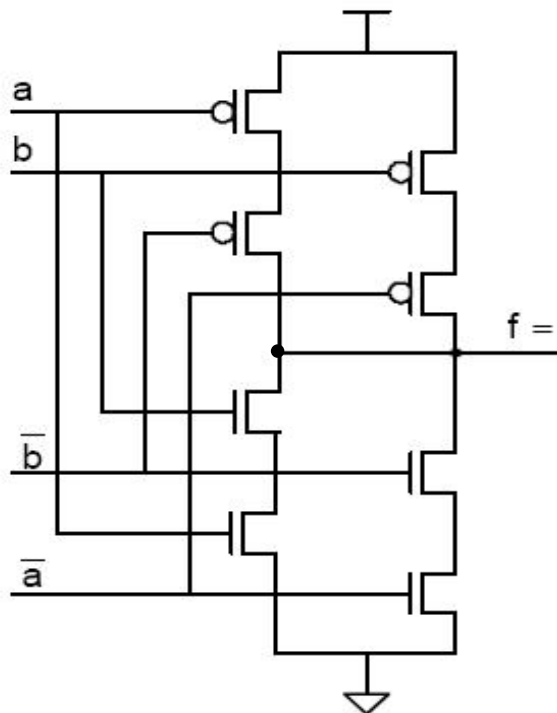
מועד המבחן: 11.2.2005

חומר עזר מותר בשימוש: כל חומר עזר כתוב

בהצלחה!

שאלה 1 (15%):

נתון המעגל הבא :

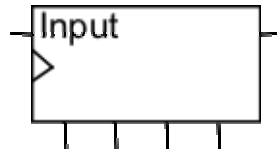


		a	
		0	1
b	0		
	1		

- א. (8%) מלא את טבלת האמת של המעגל. לפחות לכניסה אחת בטבלה הסבר ונמק מדוע הערך שמלאת.
- ב. (5%) צייר מעגל לוגי משערי AND OR ו- NOT שמממש את אותה פונקציה.
- ג. (2%) מה שם הפונקציה שמעגל זה מממש?

שאלה 2 (20%)

נתון אוגר הזזה 4-bit.



בשמונה פעימות השעון הבאות יכנסו אל כניסת הקלט של האוגר הביטים של מספר בן שמונה ביטים, אחד אחרי השני. שרטט מעגל הכולל את האוגר הנ"ל, ובו קו יציאה "equal", שערכו יהיה "1" אם רביעיית הביטים הראשונים זהה לרביעייה השניה, ו-"0" אחרת.

א. (8%) כאשר לרשותך מינימום שערים לוגים, 4 יחידות FF (הסוג – לבחירתך) וקבועים. עליך להשתמש בכל ארבעת ה-FF ולתת פתרון פשוט.

ב. (12%) כאשר לרשותך מינימום שערים לוגים, ורק 3 יחידות FF לכל היותר וקבועים.

שאלה 3 (20%) :

Four-bit-adder הוא רכיב המחבר שני מספרים בני ארבעה ביטים.

א. (8%) שרטט מעגל המקבל שני מספרים בני ארבעה ביטים ומשתמש ב-four-bit-adder בשערים לוגים ובקבועים; פלט המעגל הוא קו "equal" שערכו 1 אם המספרים שווים, 0 אחרת.

ב. (12%)

הקוד הבא מתאר רכיב להשוואת שני מספרים בני ארבעה ביטים בשפת VHDL. השלם את הקוד כך שפלט המעגל יבצע את המבוקש בסעיף א.

```
entity fourbitcompare is
    port (a, b: in std_logic_vector(3 downto 0);
          equal: out std_logic);
end fourbitcompare;

architecture compare_structure of fourbitcompare is
    component fourbitadd
        port (a, b: in std_logic_vector(3 downto 0);
              Cin : in std_logic;
              sum : out std_logic_vector (3 downto 0);
              Cout, overflow: out std_logic);
    end component;
```

שאלה 4 (20%) :

נתון ה- **data-path** המצולם בדף אחרי הבא. המעבד הזה מניח בברירת מחדל שפקודת **conditional-branch** לא נלקחת ואם הקפיצה כן נלקחת אז משלמים קנס של מחזור שעון אחד (**stall** אחד).

עבור התוכנית הבאה קבע את ערכו של כל קו בקרה בטבלה הבאה במחזורי שעון 3 עד 9 כולל. כמו כן הראה איזה פקודת מכונה תופסת כל תחנה ב**pipeline** בכל מחזור שעון. (הנח שקו הבקרה **write-enable IF/ID** מקבל את הערך ההפוך של קו הבקרה **Stall**).

המצב ההתחלתי שבו מתחיל המעבד הינו :

- **PC = 0**
- **All pipeline registers contain 0**
- **Registers: All registers in register file contain 0**
- **Memory: All data-memory locations contain 0**

עליך להשתמש ב **data-forwarding** ככל שניתן.
כל כניסות ה **mux**-ים ממוספרות אנכית מלמטה למעלה 0, 1, 2 וכי' בהתאם ל**mux**.
הקוד ל **ALU** הוא :

Value	Desired ALU Action
00	Add
01	Subtract
10	Determine by decoding funct field

Instruction memory (זכרון הפקודות) contains:

00 : addi \$3, \$zero, 4

04 : lw \$4, 100(\$3)

08: add \$2, \$4, \$3

0c: beq \$4, \$zero, 0x14

10: addi \$3, \$3, 1

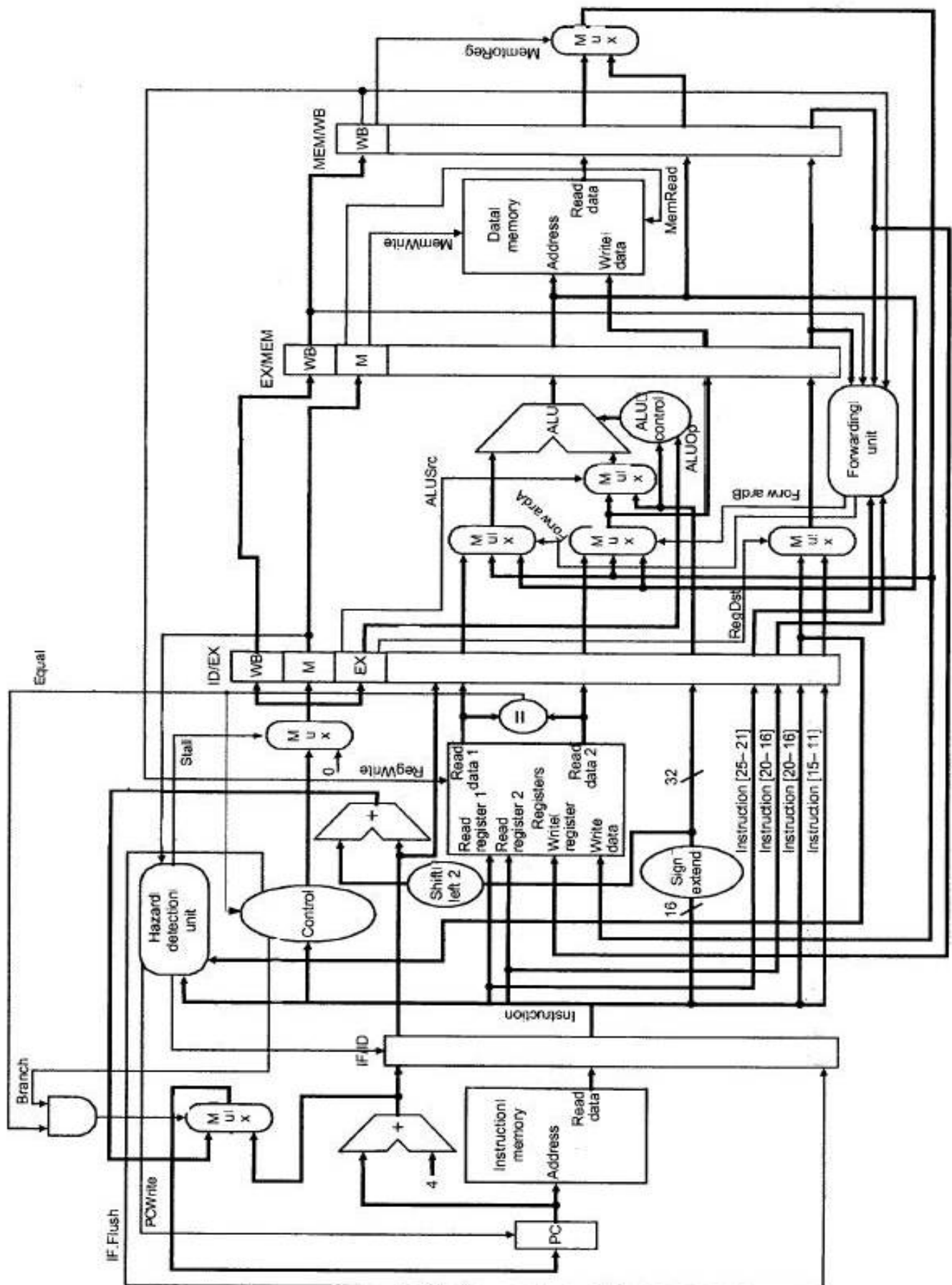
14: add \$2, \$2, \$3

כל שאר זכרון הפקודות מכיל 0.

	Time	PCWrite	IF.Flush	Branch	Stall	ALUSrc	ALUOp	RegDst	ForwardA	ForwardB	MemRead	MemWrite	MemToReg	הערות
	IF		ID		EX					M		WB		הערות
0	00: addi		--		--					--		--		
		1	0	0	0	0	00	0	00	00	0	0	0	
1	04: lw		00: addi		--					--		--		
		1	0	0	0	0	00	0	00	00	0	0	0	
2	08: add		04: lw		00: addi					--		--		
		1	0	0	0	1	00	0	00	00	0	0	0	
3														
4														
5														
6														
7														
8														
9														

מס' ת.ז. : _____ מחברת מספר: _____ 7

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



שאלה 5 (25%) :

- א. (5%) במערכת זכרון מטמון (cache memory) בשיטת המיפוי הישיר (direct mapped) כאשר :
- המחשב עובד עם זכרון מטמון יחיד,
 - בהנחה שתוכן ה-Instruction Register (IF/ID) אינו נשמר בזמן Cache-miss, כלומר רגיסטר IF/ID נמחק בזמן Cache miss.
- מה מספר ה-block-ים המינימלי שדרוש כך שמעבד יוכל להתקדם. שלא יהיה מצב בו הוא לא מסוגל להתקדם. (הנח שכל בלוק מכיל מילה אחת). נמק ותן דוגמא.

- ב. (5%) במערכת זכרון מדומה (virtual memory) מה מספר הדפים הפיסיים המינימלי שדרוש לתוכנית על מנת שתוכל להתקדם. נמק והסבר.

ג. (7%) עבור יחידת ה-TLB כפי שנלמדה בכיתה כמה ביטים של זכרון דרושים לה (כולל ה valid bit) אם :

- כתובת בזכרון המדומה (virtual memory) הינה בת 32 ביטים.
- כתובת בזכרון הפיסי תופסת 31 ביטים.
- גודל הדף הוא 2K בתים.
- ה TLB מכיל 16 כניסות מטבלת הדפים.

הסבר את דרך הפתרון.

ד. (8%) מהנדסי Intel מבקשים שתחשבו באיזה אחוז תשתפר יעילות העבודה של CPU אם יחידת ה **Branch Prediction** תשפר את יכולת הניבוי באחוז אחד ובשני אחוזים. לשם כך עליך לענות על השאלות הבאות. כל החישובים הם עפ"י ההנחות הבאות:

- אחת לחמש (1: 5) פקודות מכונה היא פקודת קפיצה מותנת (**conditional branch**)
- מחיר ניבוי לא נכון הוא 12 stalls. (הנח שאין stalls אחרים מלבד אלה שנובעים מניחוש לא נכון).
- לפני השיפור ה **Branch Prediction Unit** מנחש נכון ב 95% מהקפיצות (**Branches**).

(a) (4%) מה אחוז ה stalls כאשר היחידה מנחשת נכון ב- 95% מהקפיצות המותנות (בריצה ארוכה איזה אחוז ממחזורי השעון יהיו stalls)?

(b) (2%) מה אחוז ה stalls כאשר היחידה מנחשת נכון ב- 96% מהקפיצות המותנות?

(c) (2%) מה אחוז ה stalls כאשר היחידה מנחשת נכון ב- 97% מהקפיצות המותנות?

הסבר את חישוביך.

בהצלחה!