

שלום לכולם

בשיעור זה נדבר על סוגי הפקודות השונים, ולאחר מכן ניתן דוגמה למרבית הפקודות המוכרות.

נציג תחילה מספר סוגי פקודות שונים :

- פקודות מסוג R : מיועדות לפעולות אריתמטיות בין 2 אוגרים (למשל חיבור, חיסור, and, or וכו..). כמו add \$10, \$8, \$9
 - פקודות Shift מסוג R : מבצעות הזזה ימינה או שמאלה של 32 הסיביות.
 - לדוגמה אם נניח שבאוגר \$9 שמור הערך 1110000101000001010000010100001 והמשמעות של הזזה שמאלה 2 יחידות תהיה למחוק את שתי הספרות השמאליות, ולהכניס מימין 2 אפסים כך שנקבל : 11000010100000101000001010000100.
 - פקודות לוגיות מסוג I : פעולה בין אוגר לבין מספר קבוע. למשל addi \$10, \$8, 256.
 - פקודות לקבלת החלטות מסוג I : בפקודות אלו לרוב מדובר על קפיצה, שתבצע אם יתקיים תנאי מסוים, כמו שקורה בלולאות. למשל beq \$8, \$9 L1 שתבדוק האם ערכי האוגרים 8 ו-9 שווים, ואם כן תקפוץ לתווית L1 שמייצגת כתובת כלשהי בזיכרון.
 - פקודות זיכרון מסוג I : לוקחות ערך מסוים מהזיכרון ושמות אותו באוגר מסוים, או להפך. לדוגמה lw \$t0, 12(\$s2).
- גודל הערך יכול להיות למשל בית אחד, חצי מילה, מילה..

הערות	היכן תישמר התוצאה	אופרנד שני	אופרנד ראשון	סוג	פקודה	משמעות
תוצאת פעולת הכפל תאוחסן ב-2 אוגרים Lo ו Hi	\$8	\$10	\$9	R	add \$8, \$9, \$10	חיבור אוגרים
					sub \$8, \$9, \$10	חיסור אוגרים
					and \$8, \$9, \$10	פעולת "וגם" על כל סיבית
					or \$8, \$9, \$10	פעולת "או" על כל סיבית
					xor \$8, \$9, \$10	פעולת "xor" על כל סיבית
					mult \$8, \$9, \$10	כפל אוגרים
	\$8	250	\$9	I	addi \$8, \$9, 250	חיבור אוגר עם מספר
	\$8	\$10	\$9	R	slt \$8, \$9, \$10	אם הערך באוגר 9 קטן מהערך באוגר 10 במשלים ל-2, אוגר 8 יקבל את הערך 1, אחרת 0.

היכן תישמר התוצאה	אופרנד שני	אופרנד ראשון	סוג	פקודה	משמעות
\$8	\$10	\$9	R	addu \$8, \$9, \$10	חיבור אוגרים תוך התייחסות לערכיהם שמיוצגים בשיטת ללא סימן
				subu \$8, \$9, \$10	חיסור אוגרים ללא סימן
\$8	150	\$9	I	sltiu \$8, \$9, 150	האם הערך באוגר 9 כמספר ללא סימן קטן מ 150, אוגר 8 יקבל ערך '1'.

היכן תישמר התוצאה	סוג	פקודה	משמעות
\$t0	R	sll \$t0, \$t1, 3	הזזה שמאלה 3 סיביות של המידע המאוחסן ב t1, והכנסת אפסים מימין.
		srl \$t0, \$t1, 3	הזזה ימינה 3 סיביות של המידע המאוחסן ב t1, והכנסת אפסים משמאל.
		sra \$t0, \$t1, 3	הזזה ימינה 3 סיביות של המידע המאוחסן ב t1, ושכפול הספרה השמאלית ביותר 3 פעמים משמאל

היכן תישמר התוצאה	אופרנד שני	אופרנד ראשון	סוג	פקודה	שם
t0	12	\$s2	I	lw \$t0, 12(\$s2)	הערך שנמצא בזיכרון בכתובת שהיא סכום האופרנדים ייכנס לאוגר \$t0
t0	12	\$s2	I	sw \$t0, 12(\$s2)	הערך שנמצא באוגר \$t0 ייכתב בזיכרון בכתובת שהיא סכום האופרנדים

אופרנד שני	אופרנד ראשון	סוג	פקודה	משמעות
\$s4	\$s3	I	beq \$s3,\$s4, L1	לקפוץ לכתובת שמיוצגת ע"י התווית L1 אם ערכי האוגרים שווים
			bne \$s3,\$s4, L1	לקפוץ לכתובת שמיוצגת ע"י התווית L1 אם ערכי האוגרים שונים

הערות	סוג	פקודה	משמעות
	I	j Target	לקפוץ לכתובת שמיוצגת ע"י התווית Target
		jal Target	לקפוץ לכתובת שמיוצגת ע"י התווית Target ולשמור באוגר 31 את כתובת הפקודה שבאה אחריה (לפני הקפיצה)
		jr \$t0	לקפוץ לכתובת שמאוחסנת ברגיסטר t0

אופרנד שני	אופרנד ראשון	סוג	פקודה	משמעות
\$s4	\$s3	I	lbu \$t5, 2(\$s3)	ללכת לבית בזיכרון שכתובתו היא חיבור הערך של s3, עם היסט של 2 (בתים) ולשים את תוכנו ב t5 ומשמאלו אפסים
			lb \$t5, 2(\$s3)	ללכת לבית בכתובת שהיא חיבור של s3, עם היסט של 2 ולשים את תוכנו ב t5 מורחב ל-32 סיביות (שכפול ספרה שמאלית)

סוג	פקודה	משמעות
I	lui \$s0, 72	<p>פקודה זו טוענת את הערך המיידני 72 כמספר בינארי של 16 סיביות -</p> <p>ב-16 הסיביות העליונות של הרגיסטר s0, ומאפסת כל 16 הסיביות הנמוכות.</p> <p>אם ברצוננו לטעון ערך של 32 סיביות לאוגר מסוים, אז קודם כל נטען את 16 הסיביות העליונות בעזרת lui, ואז נבצע פעולת ori עם 16 הסיביות הנמוכות לא נוכל להשתמש ב addi מכיוון שהיא מבצעת הרחבת סימן.</p>