

שאלות תיאורטיות

1. מחסנית היא אזור בזיכרון RAM שבו התוכנית משתמשת לשמירת כתובת לחזרה מרוטינה ופסיקה ובשביל עוד מקום לאגירת מידע בנוסף לרגיסטרים. בתחילה נאתחל את ערכו לערך של סוף הזיכרון ה RAM (בשביל גודל מקסימלי של המחסנית) בכל שימוש PUSH. ה DATA נכנס ל TOS. וכתובת ה SP קטנה ב – 2 (כתובת ה SP מורה על ה TOS).

2. כאשר יש צורך בקטע קוד שחוזרים על עצמם, נוכל להגדיר "רוטינה", שזו כמו פונקציה בשפת תכנות הילית, אליה נוכל לקרוא בלי לשכפל קטעי קוד. הרוטינה יושבת בזיכרון ה flash. כדי להגדירה, הרוטינה תתחיל בשם המשתמש destination לפקודת call, הקוראת לרוטינה מהקוד הראשי. היתרונות המרכזיים הם שאנו נמנעים מניפוח זיכרון הבקר, וניתן לדבג בנוחות (בניגוד למקרו) כשקוראים לרוטינה, המיקום הבא ב PC (2+PC) נכנס ל TOS של המחסנית, ה SP קטן ב -2, וכתובת הרוטינה מועברת ל PC. בסוף הרוטינה, חוזרים ל Main באמצעות פקודת RET, שבו ערך ה SP גדל חזרה ב – 2.

3. פונקציית ה MACRO מהווה תחליף לרוטינה באופן הבא: בכל שורת קוד בו תופיע שימוש בפונק' ה MACRO, תוחלף השורה בקטע הקוד אותו הגדרנו מראש בתחילת התכנית. להבדיל מרוטינה, השימוש בפונק' ה MACRO, מועתקות כל שורות גוף הפונק' למיקום בו קראנו לפונ' בתכנית המרכזית.

יתרונות	חסרונות
עבור קטע קוד שהוא נורא קצר אנחנו חוסכים את השהייה ביצאה לרוטינה ובחזרה ממנה	-לא ניתן לדבג -מנפח את גודל התוכנית (בגלל העתק הדבק)

גודל התוכנית – hex(5E)-94

$$202 \times 0.954 \times 10^{-6} = 1.927 \times 10^{-4}$$

זמין ריצה –