

שאלות חלק תיאורטי – הרצת קוד לדוגמא בסביבת IAR:

1. הסבר את ההבדל בין משתנים לוקאלים למשתנים גלובליים.

ההבדלים בין משתנה לוקאלי למשתנה גלובלי מתבטאים בכמה אופנים:

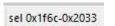
מיקום הגדרת המשתנה - משתנה גלובלי מוגדר מחוץ לכל הפונקציות, בדרך כלל בתחילת התוכנית וערכו הדיפולטיבי הינו 0. בעוד שמשתנה לוקאלי מוגדר בתוך פונקציה וערכו הדיפולטיבי הוא ערך זבל.

"זמן החיים" - משתנה גלובלי יהיה קיים בזיכרון לכל אורך התוכנית וניתן יהיה לשנות את ערכו ולבצע עליו חישובים בכל חלקי התוכנית. בעוד שמשתנה לוקאלי מקבל הקצאת זיכרון (במחסנית) בשורת הגדרתו בתוך הפונקציה ונמחק בסיומה.

דוגמא למשתנה גלובלי: maxTrace והסקופ שלו הוא מהגדרתו ועד סוף התכנית. דוגמא למשתנה לוקאלי: Max שהוא משתנה לוקאלי של הפונקציה DiagonalsMax והסקופ שלו הוא מהגדרתו ועד סיום הפונקציה.

בזיכרון ומה טווח הכתובות אותו הוא מכסה. מהו סוג זיכרון 2. מה כתובת המערך 2 Mat 2 בזיכרון ומה טווח הכתובות אותו הוא מכסה. מהו סוג זיכרון זה

עבור הIAR:

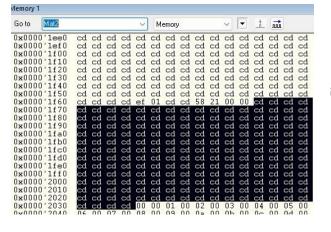


כתובת המערך הינה 0x1f6c והטווח הינו עד 0x2033.

סך הכל 200 כתובות, כמספר הערכים של מטריצה obyte 2. בזיכרון שהוא

cd cd הערך של המערך בכל איבריו הוא

המערך נמצא בזיכרון הstack מכיוון שהוא משתנה לוקאלי של הפונקציה הראשית main.



עבור ה CCS:

Name	Туре	Value	Location
> 🥭 Mat2	int[10][10]	[[0,0,0,0,0,],[0,0,0,0,0],[0,0,0,	0x04C6

המערך הוא מערך בגודל 10*10 של משתנים מסוג Int, לכן אמור לחסות כ100*גודל משתנה בזיכרון, ניתן לראות כי:

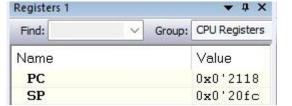
0x04C6 start, 0x058C end לכן מכסה בסה"כ 200 בתים (הכתובת סיום היא מקום המשתנה האחרון המכיל 2 בתים גם)

3. רשום את כתובת תחילת מיקום המחסנית בזיכרון הנקבע ע"י המהדר.

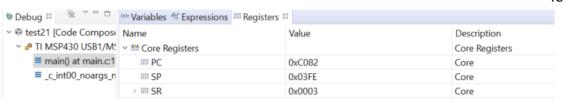
עבור IAR:

בתחילת הריצה רגיסטר SP הוא המצביע על של המחסנית מאותחל לכתובת הראשונה של המחסנית מכיוון שעוד לא נכנס אליה שום ערך. להלן הכתובת :

0x020fc



עבור CCS:



ניכנס לחלונית הרגיסטרים ונראה את כתובת ה SP בתחילת ההרצה: לכן כאן SP הוא Ox3FE

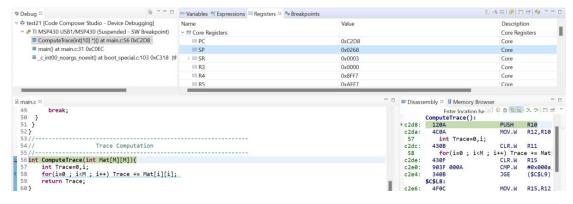
4. רשום את תוכן SP כאשר רגיסטר PC כאשר רגיסטר SP 4. ComputeTrace

:IAR עבור

ניתן לראות כי הכתובת שעליה מצביע הSP השתנתה כי בקריאה לפונקציה הכתובת האחרונה שבה היה הPC. נכנסת למחסנית על מנת שבסיום התכנית נחזור לנקודה האחרונה בה התכנית הקודמת הייתה.

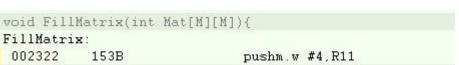


עבור CCS:



נראה כי כתובת הPC מבציעה על השורה הראושנה בפונקציה לפי ההדגשה בחלקון מראה כי כתובת PC מבציעה על השורה הראושנה בפונקציה לפי ההדגשה בחלקון הדיסאסמבלי. ולכן כתובת SP כעת היא

5. רשום את כתובת הפונקציה FillMatrix בזיכרון .מה גודל קוד הפונקציה 5. בבתים ? מהו סוג זיכרון זה?



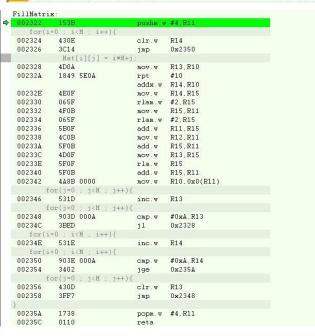
בAAI:

בחלון ה dissamsemblly ניתן לראות את הפונקציה במלואה עם כל שורות הקוד שלה וכיצד הפקודות מאוחסנות בזיכרון.

כתובת הפונקציה היא הכתובת הראשונה והיא, מסכיסיס

גודל הקוד הוא מהכתובת האחרונה פחות הראשונה: bvte 0x0235C-0x2322 = 59

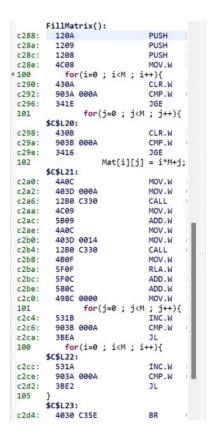
סוג הזיכרון לאכסון הפונקציות הוא זיכרון הflash.



:CCSa

נראה את קוד האסמבלי החלון הדיסאסמבלי:

נוכל לחסר בין הכתובת האחרונה לראשונה בפונקציה ונראה כי גודל הפוקנציה הוא 76 בתים.



6. מהו זמן ריצת הקוד של הפונקציה FillMatrix ביחידות מחזור של

:IAR₂

לפני ריצת הפונקציה:

CYCLECOUNTER 58

לאחר ריצת הפונקציה:

CYCLECOUNTER 3'407

סך הכל 3349 מחזורי שעון.

7. מהו ה scope -של משתנה Trace2mat בתוכנית, מהו מיקומו בזמן ה- scope

המשתנה Trace2mat הוא משתנה לוקאלי של פונקציית הmain ולכן ה scope שלו הוא כל escope משתנה משתנה משתנה לוקאלי של פונקציית הmain. בזמן הscope המשתנה נמצא במחסנית.

maxTrace = את קוד האסמבלי המתורגם ע"י המהדר עבור שורת הקוד הבאה .8 mat1Trace > mat2Trace ? mat2Trace : mat2Trace;

:IARב

:disassembly

002158	93.00	emp.w	R10, R12
	A CONTRACTOR OF THE CONTRACTOR		1
00215A	3403	jge	0x2162
00215C	4A82 1100	mov.w	R10,&maxTrace
002160	3C02	jmp	0x2166
002162	4C82 1100	mov.w	R12,&maxTrace

:CCSa

```
33
            maxTrace = mat1Trace > mat2Trace ? mat1Trace : mat2Trace;
c0f8:
        990C
                          CMP.W
                                 R9,R12
c0fa:
        3401
                          JGE
                                  ($C$L7)
c0fc:
        490C
                          MOV.W
                                 R9,R12
      $C$L7:
c0fe: 4C82 0202
                         MOV.W R12,&maxTrace
35
           break;
```

נוכל לראות שהפוקודת זהות כמעט לגמרי וההבדל הוא ברגיסטרים ובלייבלים בלבד.