

MOR LIB.ASM ספריית - 27.2.2022

The only thing that you absolutely have to know is the location of the library. -Einstein

- ספרייה זו אוסף שירותים שמסופק בקובץ נוסף.
- ספריית MOR_LIB מספקת:
 - קריאת מקלדת לא חוסמת - MOR_GET_KEY
 - הדפסת מספר עשרוני - MOR_PRINT_NUM
 - השהיה קצובה - MOR_SLEEP
 - יצירת מספר רנדומלי - MOR_RANDOM
 - הצגת תמונת מסך ותמונה קטנה - MOR_SCREEN , MOR_LOAD_BMP
 - סטופר למדידת זמן - MOR_STOPPER_START , MOR_STOPPER_GET
 - ציור נקודה , קו ומלבן - drawPixel , drawLine , drawRect
 - הדפסת מחרוזת במצב גרפי - putMessage
- את קובץ הספרייה יש למקם באותה תיקייה כמו התוכנית שמשתמשת בה.
- בנוסף , בתוכנית יש להשתמש בפקודת INCLUDE שתגרום לספרייה להיכלל בתוכנית שלי. את הפקודה יש למקם בשורה הלפני אחרונה של התוכנית :

```
451  
452 INCLUDE "MOR_LIB.asm"  
453 END start  
454
```

- עתה עלי להכיר את הפרוצדורות השונות בספרייה ואת דרך העברת הפרמטרים לפרוצדורות.

הדפסת מספר עשרוני - MOR_PRINT_NUM

- הפרוצדורה מדפיסה למסך את הייצוג העשרוני של המספר המצוי ב AX

השהיה קצובה - MOR_SLEEP

- פרוצדורת ההשהיה דורשת ממני להכניס ב AX את זמן ההשהיה במילי-שניות ואז לקרוא לפרוצדורה שתשהה את התוכנית בהתאם לערך זה.
- הערה : MOR_SLEEP מבוססת על רכיב שעון המחשב שמתקדם כל 55 מילישניה ולכן הדיוק של ההשהיה יהיה בהתאם.

סטופר למדידת זמן

- משמש למדידת זמן בשניות
- MOR_STOPPER_START - איפוס
- MOR_STOPPER_GET - קבלת הזמן שעבר מהאיפוס (בשניות . מוחזר ב AX)
- הערה : הסטופר משתמש בשעון המובנה ומאפס אותו.
- מודגם בתוכנית MB15.ASM

קריאת מקלדת לא חוסמת - MOR_GET_KEY

- עד היום הכרנו את הפסיקה שאפשרה לנו לקרוא מידע מהמקלדת (INT21H כאשר AH=1)
- אך פעולה זו היתה **פעולה חוסמת (BLOCKING)**. כלומר עד שלא הוקש מקש התוכנית שלנו היתה חסומה.
- קריאה חוסמת של המקלדת מתאימה לחלק מהתוכניות כמו חישוב ממוצע ציונים.
- בתוכניות אחרות, כמו משחק חלליות, אנחנו שהמשחק יחיה וישתנה גם כאשר השחקן לא מקליד כלום.
- דוגמא נוספת היא התרגיל המתקדם עם **הלב הפועם** שנמשיך היום.
- **MOR_GET_KEY** היא פרוצדורה שבודקת את המקלדת אך חוזרת מיידית ולא חוסמת. לאחר הקריאה יש לבדוק אם נקלטה הקשת מקש:
 - במידה ולא נקלטה אזי הדגל ZERO יהיה 1 (TRUE)
 - במידה ונקלטה הקשת מקש אז הדגל ZERO יהיה 0 (FALSE) וקוד ה ASCII של המקש יהיה ברגיסטר AL
- בקשר ללחצנים מיוחדים (כמו חצים, מקש CONTROL ALT) וכן זיהוי לחיצה ושחרור של מקש - קראו בסוף המסמך

יצירת מספר רנדומלי - MOR_RANDOM

- הפרוצדורה MOR_RANDOM מייצרת מספר אקראי (בעצם פסאדו-רנדומלי)
- הפרוצדורה תייצר מספר אקראי בין 0 ל מספר שהוכנס ב - AX פחות 1.
- התוצאה תוחזר ב AX.

תוספות :

```
=====
;
; putMessage - print message on screen
; IN: DH= row number , DL = column number , cx = the message (offset)
; OUT: NONE
;     AFFECTED REGISTERS AND VARIABLES: NONE
```

```
=====
;
; drawPixel – draw a pixel at X,Y
; IN: CX=X , DX =Y , AL = COLOR
; OUT: NONE
;     AFFECTED REGISTERS AND VARIABLES : NONE
```

```
=====
;
; drawLine – draw a line starting at X,Y
; IN: CX=X , DX =Y , AL = COLOR , AH = WIDTH
; OUT: NONE
;     AFFECTED REGISTERS AND VARIABLES: NONE
```

```
=====
;
; drawRect – draw a rectangle starting at X,Y
; IN: CX=X , DX =Y , AL = COLOR , AH = WIDTH , BL = HIGHT
; OUT: NONE
;     AFFECTED REGISTERS AND VARIABLES: NONE
```

הצגת BMP

- יש שתי פרוצדורות לשם כך
 - **MOR_SCREEN** - מציגה תמונת מסך שלמה
 - **MOR_LOAD_BMP** - מציגה BMP קטן במיקום כלשהוא על המסך
- קבצי התמונה צריכים להיות בפורמט BMP של 256 צבעים
- כאשר מגדירים את שם הקובץ יש להוסיף את המספר 0 לציון סוף המחרוזת :

```
logo db 'BYE2.BMP',0
```

- מומלץ להשתמש בצייר של חלונות כדי ליצור, לערוך או להמיר תמונות מפורמטים אחרים
- תזכורת : ב DOS שמות הקבצים צריכים להיות בפורמט 8.3.
- עדיין אין תמיכה בפיקסל שקוף.
- ראו דוגמה בקובץ BMP_EXM.ASM

```
proc MOR_SCREEN
; DO : take a BMP and put it on screen ( starting from location 0,0)
; IN : ax filename
; OUT : None
; effected registers : None
```

```
proc MOR_LOAD_BMP near
; DO : load a BMP at location (x,y)
; IN : ax - filename cx - x dx - y
; OUT : None
; effected registers : None
```

הדגמת MOR_LIB ובאפר המקלדת

ראשית נסביר על באפר המקלדת - Type Ahead Buffer :

- באפר (בעברית חוצץ) הוא קטע זכרון הצובר מידע שיעובד בהמשך.
- זהו סוג של תור בגודל מסוים אליו נכנסים נתונים שהמעבד יטפל בהם בהמשך.
- למשל - באפר המקלדת - Type Ahead Buffer או TAB
- המחשב מכיל באפר שיכול לשמור עד 16 הקשות מקש ושומר אותם לשימוש התוכנה.
- התוכנה יכולה להיות באותו הזמן עסוקה במשהו אחר כמו חישוב מתמטי או ציור מסך.
- באותו זמן, הבאפר אוגר את הלחיצות.
- כאשר התוכנה רוצה לבדוק מה נלחץ היא פונה לרכיב ומקבלת את הלחיצות לפי הסדר שנלחצו.

תוכנית דוגמא MOR_EXM.ASM :

תוכנית הדוגמה תבצע 10 הגרלות של מספר אקראי (בעזרת **MOR_RANDOM**) כאשר בינהן תהיה השהיה של שניה אחת (בעזרת **MOR_SLEEP**). בזמן הזה יכול המשתמש להקליד על המקלדת הודעה קצרה . בשלב זה התוכנית תתעלם מההקלדה אך היא תישמר בבאפר (אבל מקסימום 16 הקלדות).

לאחר תום 10 השניות התוכנית תפנה ותקרא את ההקלדה. מאחר והתוכנית לא יודעת כמה אותיות הוקלדו היא תשתמש בשירות **MOR_GET_KEY** שאינו חוסם ומיידי מחזיר אם נותרו הקלדות .

נספחים

ללחצנים מיוחדים וכן זיהוי לחיצה ושחרור של מקש

המערכת יכולה לתת לנו גם חייווי על לחצנים מיוחדים (כמו חצים , מקש CONTROL ALT) וכן זיהוי לחיצה ושחרור של מקש . הנושא מפורט בספר בפרק 13 עמוד 250 אך ניתן לקבלם גם בעזרת MOR_GET_KEY.
לכל מקש מיוחד יש קידוד שנקרא SCAN CODE . את ערך הקוד אנחנו מקבלים ברגיסטר **AH** לאחר הקריאה ל MOR_GET_KEY (בתנאי שנלחץ לחצן כלשהו - שדגל ה ZERO הוא FALSE)

Key	Down	Up	Key	Down	Up	Key	Down	Up	Key	Down	Up
ESC	1	81	[{	1A	9A	, <	33	B3	center	4C	CC
1 !	2	82] }	1B	9B	. >	34	B4	right	4D	CD
2 @	3	83	Enter	1C	9C	/ ?	35	B5	+	4E	CE
3 #	4	84	Ctrl	1D	9D	R shift	36	B6	end	4F	CF
4 \$	5	85	A	1E	9E	PrtSc	37	B7	down	50	D0
5 %	6	86	S	1F	9F	alt	38	B8	pgdn	51	D1
6 ^	7	87	D	20	A0	space	39	B9	ins	52	D2
7 &	8	88	F	21	A1	CAPS	3A	BA	del	53	D3
8 *	9	89	G	22	A2	F1	3B	BB	/	E0 35	B5
9 (0A	8A	H	23	A3	F2	3C	BC	enter	E0 1C	9C
0)	0B	8B	J	24	A4	F3	3D	BD	F11	57	D7
- _	0C	8C	K	25	A5	F4	3E	BE	F12	58	D8
= +	0D	8D	L	26	A6	F5	3F	BF	ins	E0 52	D2
Bksp	0E	8E	; :	27	A7	F6	40	C0	del	E0 53	D3
Tab	0F	8F	"	28	A8	F7	41	C1	home	E0 47	C7
Q	10	90	` ~	29	A9	F8	42	C2	end	E0 4F	CF
W	11	91	L shift	2A	AA	F9	43	C3	pgup	E0 49	C9
E	12	92	\	2B	AB	F10	44	C4	pgdn	E0 51	D1
R	13	93	Z	2C	AC	Num	45	C5	left	E0 4B	CB
T	14	94	X	2D	AD	SCRL	46	C6	right	E0 4D	CD
Y	15	95	C	2E	AE	home	47	C7	up	E0 48	C8
U	16	96	V	2F	AF	up	48	C8	down	E0 50	D0
I	17	97	B	30	B0	pgup	49	C9	R alt	E0 38	B8
O	18	98	N	31	B1	-	4A	CA	R ctrl	E0 1D	9D
P	19	99	M	21	B2	left	4B	CB	pause	E1 1D	-

טבלת תווים ו-scan codes

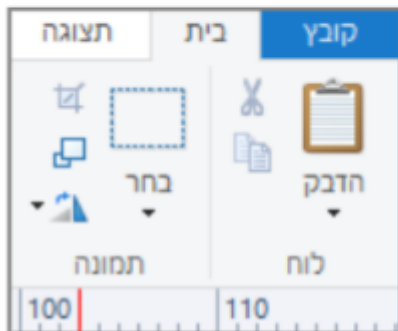
תמונות בתוכנית אסמבלי - עם MOR_LIB

את התמונות אפשר ליצור עם הרבה תוכנות, למשל הצייר של חלונות. יש להקפיד ש:

- הקובץ יישמר בפורמט 256 COLOR - BMP
- שם הקובץ יהיה עד 8 אותיות ומספרים (ואחר כך .BMP)
- מימדי התמונה יהיו כפולות של 8
- במידה ורוצים תמונת מסך מלאה המימדים הם 320 על 200 פיקסלים
- התמונה באה עם לוח-צבעים משלה PALLET שיכול להיות שונה מלוח הצבעים הבסיסי (אפשר לאפס את המסך בין תמונה מסוימת לתוכנית שלכם על ידי העברה נוספת למוד גרפי)

שימוש בצייר :

ציור בתוכנת צייר.



נצייר בצייר ריבוע שחור



נקבע את גודל התמונה לפיקסלים (16*16)

תפריט בית ⇨ שינוי גודל תמונה ⇨ נבחר בפיקסלים
נקבע את הגודל במישור האופקי (רחב התמונה) למספר
המתחלק ב - 8.

נשמור את התמונה בפורמט של - **bitmap 256** ⇨ br.bmp

כדי שנוכל להשתמש בתמונה באסמבלי היא צריכה להיות שמורה במחיצה של tasm / bin



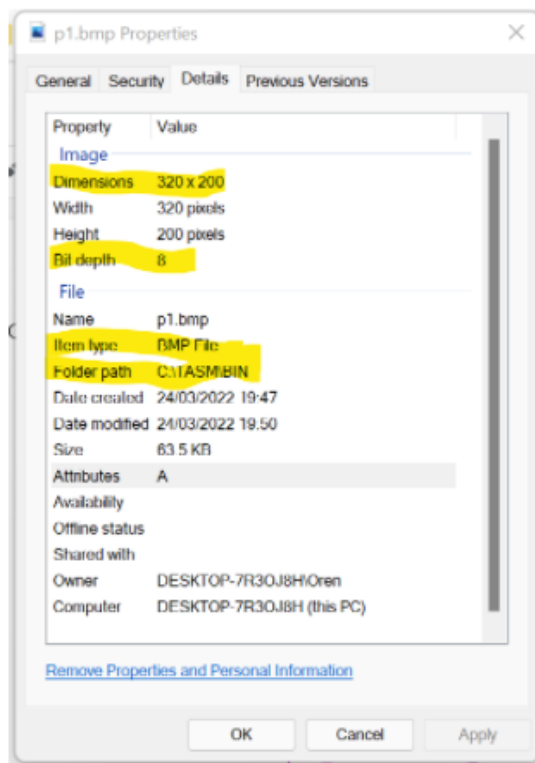
נצייר בצייר בתוך הריבוע השחור עיגול צבעוני. ⇨ pc.bmp
ונשמור אותו לתיקיה של tasm / bin.

התאמת תמונה לפורמט BMP 256

תמונות מתאימות לנו הם בפורמט **BMP** עם **BIT DEPTH** של 8 (256 צבעים). :

התאמת תמונה לפורמט BMP 256

תמונות מתאימות לנו הם בפורמט **BMP** עם **BIT DEPTH** של 8 (256 צבעים) :

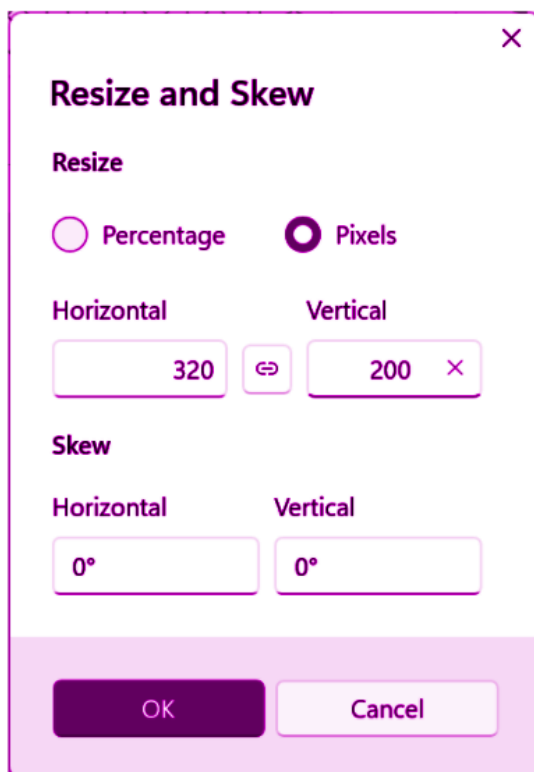


כדי להפוך תמונה כל שהיא לתמונה שתתאים לנו אז :

- פותחים אותה בצייר של חלונות:

RIGHT CLICK ON FILE -> OPEN WITH -> PAINT

- עושים שינוי גודל **RESIZE** לגודל של 320 על 200 פיקסלים :



- שומרים את הקובץ , בתיקייה TASM/BIN , כקובץ BMP - כאשר ה TYPE שלו הוא 256 צבעים :

