

ΑΣΚΗΣΗ 8

ΕΝΤΟΛΕΣ ΕΙΣΟΔΟΥ-ΕΞΟΔΟΥ ΚΑΙ ΧΡΗΣΗ ΤΟΥ BASIC COMPILER

Δήλωση μεταβλητών:

Dim <μεταβλητή> as <είδος>

- Boolean (1-byte, True or False)
- Short (1-byte integers in the range -128 to 127)
- Integer (2-byte integers in the range -32,768 to 32,767)
- Long (4-byte integers in the range -2,147,483,648 to 2,147,483,647)
- Single (4-byte floating point, 7 digits precision, IEEE 754)

- π.χ **Dim a As Integer**

Αριθμητικές πράξεις:

(+, -, *, /, MOD)

- π.χ **sum=a+b**

Είσοδος-Έξοδος:

<μεταβλητή> = **Get** (<port εισόδου σε Hex>)

- π.χ **a=Get(05)**

Put <port εξόδου σε Hex>, <μεταβλητή>

- π.χ **Put 08,A**

Print <port εξόδου σε Hex>, "<Κείμενο>", <μεταβλητή>, **CrLf**

- π.χ **Print 09, "To athroisma einai", sum, CrLf**

1. Άσκηση Υπολογισμού Αθροίσματος

α) Να γραφτεί ένα πρόγραμμα στον Basic Compiler του Z80 Simulator IDE, που θα υπολογίζει το άθροισμα 2 ακεραίων.

Το πρόγραμμα θα πρέπει:

- Να ζητά από τον Χρήστη να πληκτρολογήσει 2 ακεραίους αριθμούς.
- Να υπολογίζει το άθροισμα των 2 αριθμών
- Να εμφανίζει στο "Output Terminal" και σε μία LED Device το αποτέλεσμα.

Dim a As Integer	'Dilwsi metavlitwn Dim b As Integer
Dim b As Integer	
Dim sum As Integer	
Print 09, "Dwse 2 akeraious ", CrLf	'E3odos minimatos-dedomenwn stin port 09H
a = Get(05)	'Eisodos dedomenwn stin port 05H
b = Get(06)	
sum = a + b	Ypologismos athroismatos
Print 09, "To athroisma einai ", sum, CrLf	'E3odos tis timis tis metavlitis "sum", sto port 09H με analogo minima
Put 08, sum	'E3odos tis timis tis metavlitis "sum", sto port 08H

β) Στον Κώδικα Assembly, που δημιουργεί ο Simulator, προσπαθήστε να εντοπίσετε τις εντολές που έχετε μάθει.

γ) Παρακολουθείστε την εκτέλεση του προγράμματος, καταγράφοντας τις τιμές των καταχωρητών και τις θέσεις μνήμης που αλλάζουν.

Συνθήκες ελέγχου/Επαναλήψεων:

- **While** <Συνθήκη>
Μορφή: (While-Wend)
.....Εντολές
Wend

π.χ: While n>0
 m=m*n
 n=n-1 `gia ton elegxo tis "While"
Wend

- **If** <Συνθήκη> **Then**
Μορφή: (IF-THEN-ELSE-ENDIF)
.....Εντολές
Else
.....Εντολές
EndIf

π.χ: If mo>=5 Then
 Print 09, "Perase to mathima", mo, CrLf
Else
 Print 09, "Mesos oros askisewn, mikroteros tou 5", mo, CrLf
EndIf

- **FOR** <Αρχική τιμή μεταβλητής> **TO** <τελική τιμή> **STEP** <Βήμα>
Μορφή: (FOR-TO-STEP-NEXT)
.....Εντολές
Next <μεταβλητή>

π.χ: FOR A=0 TO 15	`Εαν den orizetai to STEP, enoeitai vima 1
sum=sum+A	
Next A	`Εποmeno A (A=A+1)

2. Άσκηση Υπολογισμού Παραγοντικού

α) Να γραφτεί ένα πρόγραμμα στον Basic Compiler του Z80 Simulator IDE, που θα υπολογίζει το παραγοντικό ενός ακεραίου αριθμού. Είναι γνωστό ότι το παραγοντικό ενός φυσικού αριθμού N (N!) είναι το γινόμενο $1*2*3*4*...*N$.

Το πρόγραμμα θα πρέπει:

- Να ζητά από τον χρήστη να πληκτρολογήσει ένα ακέραιο αριθμό.
- Να ελέγχει εάν ο αριθμός είναι μεγαλύτερος του μηδέν. Εάν είναι, θα συνεχίζει την εκτέλεση του προγράμματος, διαφορετικά θα ζητάει από τον χρήστη να δώσει καινούργιο αριθμό.
- Να υπολογίζει το παραγοντικό του αριθμού που δόθηκε, χρησιμοποιώντας τον βρόγχο ελέγχου While-Wend
- Να εμφανίζει στο "Terminal Output" το αποτέλεσμα.

Dim n As Long	'Dilwsi metavlitwn
Dim m As Long	
m = 1	'Orismos Arxikis timis tis metavlitis "m"
input:	'Orismos Label "Input:"
Print 09, "Dwse Thetiko Akerairo Arithmo ", CrLf	'E3odos dedomenwn apo tin port 09H
n = Get(05)	'Eisodos dedomenwn apo tin Port 05H
If n <= 0 Then	'Elegxos ean o arithmos einai thetikos akeraios
Print 09, "Lathos Arithmos! ", CrLf	
Goto input	'pigaine sto label "input:"
Else	
While n > 0	'ypologismos paragontikou
m = m * n	
n = n - 1	'Gia ton elegxo/leitourgia tis While
Wend	
Endif	
Print 09, "To paragontiko einai ", m, CrLf	

β) Στον κώδικα Assembly, που δημιουργεί ο Simulator, προσπαθήστε να εντοπίσετε τις εντολές που έχετε μάθει.

γ) Παρακολουθείστε την εκτέλεση του προγράμματος, καταγράφοντας τις τιμές των καταχωρητών και τις θέσεις μνήμης που αλλάζουν.

δ) Ξαναγράψτε το πρόγραμμα υπολογισμού παραγοντικού, χρησιμοποιώντας το βρόγχο ελέγχου FOR-TO-STEP-NEXT στη θέση της While-Wend.

3. Εντολές Εισόδου-Εξόδου

α) Να εκτελέσετε το παρακάτω πρόγραμμα εντολή προς εντολή και με βάση τα αποτελέσματα του εξομοιωτή να συμπληρώσετε τον πίνακα.

Assembly Language	Machine Language	Bytes	Cycles	States
START: IN A,(11H)	0000:			
OUT (22H),A				
LD C,33H				
IN B,(C)				
LD C,44H				
OUT (C),B				
LD C,55H				
LD D,53H				
OUT (C),D				
LD E,54H				
OUT (C),E				
LD A,45H				
OUT (55H),A				
LD A,50H				
OUT (55H),A				
END				

β) Να εκτελέσετε το παρακάτω πρόγραμμα εντολή προς εντολή και με βάση τα αποτελέσματα του εξομοιωτή να συμπληρώσετε τον πίνακα.

Assembly Language	Machine Language	Bytes	Cycles	States
START: LD BC,6311H	0000:			
LD DE,1230H				
LD SP,1225H				
PUSH BC				
PUSH DE				
LD A,00H				
LOOP: INC A				
DAA				
OUT (C),A				
LD (DE),A				
DJNZ LOOP				
POP HL				
POP BC				
LD C,55H				
OTIR				
END				