

# **1. Εγχειρίδιο χρήσης**

Το παρόν κεφάλαιο αποτελεί το εγχειρίδιο χρήσης του προγράμματος. Στοχεύει στην εξοικείωση του χρήστη με το πρόγραμμα Microlab Simulator – TSIK και στην αναλυτική επεξήγηση κάθε λειτουργίας αυτού.

Είναι χωρισμένο σε επτά ενότητες :

1. Τι είναι το Microlab Simulator – TSIK
2. Απαιτήσεις του προγράμματος
3. Εγκατάσταση του προγράμματος
4. Αρχεία και συντομεύσεις
5. Βασικές λειτουργίες
6. Εργαλεία επόπτευσης των καταχωρητών και της μνήμης
7. Άλλα βοηθητικά εργαλεία

## **1.1 Τι είναι το Microlab Simulator – TSIK**

Το πρόγραμμα Microlab Simulator – TSIK είναι ένας εξομοιωτής του εκπαιδευτικού μικροϋπολογιστικού συστήματος Microprocessor Laboratory® – HP 5036a® που είναι γνωστό με το όνομα microlab.

## **1.2 Απαιτήσεις του προγράμματος**

Προκειμένου να είναι δυνατή η εγκατάσταση και εκτέλεση του προγράμματος είναι απαραίτητο ο υπολογιστής να ικανοποιεί κάποιες προδιαγραφές υλικού (hardware specifications) και κάποιες προδιαγραφές λογισμικού (software specifications).

### **1.2.1. Προδιαγραφές υλικού**

- Επεξεργαστής Pentium® τουλάχιστον 233 MHz ή άλλος ισοδύναμος ή καλύτερος.
- Τουλάχιστον 128MB RAM
- Κάρτα ήχου που να υποστηρίζει DirectX
- Κάρτα οθόνης που να υποστηρίζει DirectX
- Αν δεν υπάρχει ήδη εγκατεστημένο το Microsoft® .NET Framework απαιτούνται 50 MB ελεύθερος χώρος στο σκληρό δίσκο, διαφορετικά μόνο 2MB.

### **1.2.2. Προδιαγραφές λογισμικού**

- Microsoft® .NET Framework 1.1.

Προκειμένου όμως να είναι δυνατή η εγκατάσταση του Microsoft .NET Framework 1.1 πρέπει να υπάρχουν στο σύστημα εγκατεστημένα τα εξής :

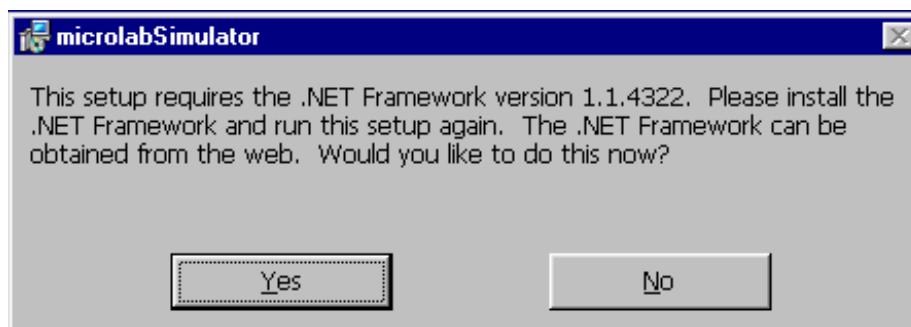
- Ένα από τα λειτουργικά συστήματα :
  - Microsoft Windows® 98
  - Microsoft Windows NT® 4.0 (Απαιτείται το Service Pack 6a)
  - Microsoft Windows Millennium Edition® (Windows Me®)
  - Microsoft Windows 2000® (Προτείνεται η εγκατάσταση του Service Pack 2)
  - Microsoft Windows XP® Professional
  - Microsoft Windows XP® Home Edition
- και ο Internet Explorer® 5.1 ή κάποια νεότερη έκδοση αυτού.

### **1.3 Εγκατάσταση του προγράμματος**

#### **1.3.1. Εγκατάσταση του Microlab Simulator – TSIK**

Για την εγκατάσταση του προγράμματος εκτελέστε το αρχείο με όνομα microlabSimulator.exe (βρίσκεται στο φάκελο setup/programSetup του CD).

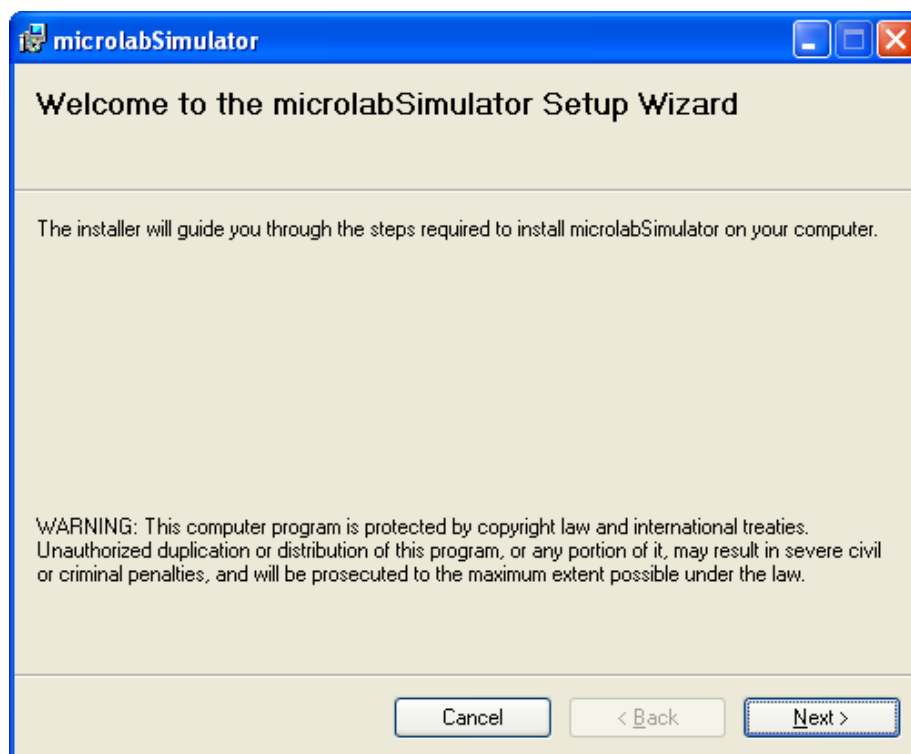
Στην περίπτωση που στο σύστημα δεν είναι εγκατεστημένο το Microsoft® .NET Framework 1.1 το πρόγραμμα εγκατάστασης θα παρουσιάσει μια οθόνη σαν αυτή που ακολουθεί.



Οθόνη 1-1 : Οθόνη που εμφανίζεται όταν δεν υπάρχει εγκατεστημένο το Microsoft® .NET Framework 1.1

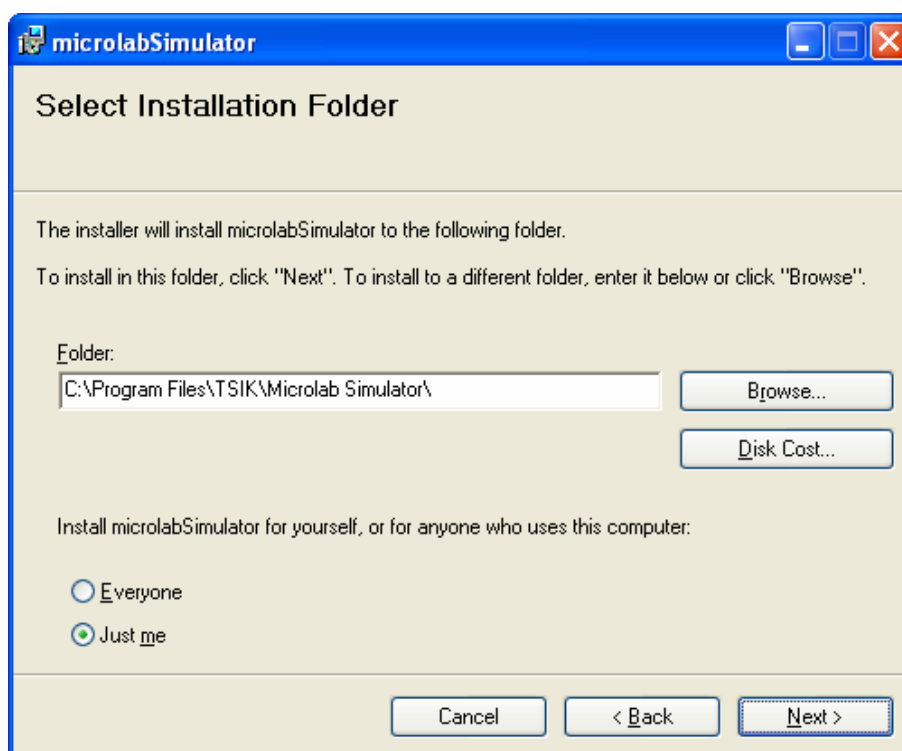
Για να εγκαταστήσετε το Microsoft® .NET Framework 1.1 ακολουθήστε την διαδικασία που περιγράφεται στην παράγραφο 5.3.2 (Εγκατάσταση του Microsoft® .NET Framework 1.1).

Αν το Microsoft® .NET Framework 1.1 είναι ήδη εγκατεστημένο θα εμφανιστεί η οθόνη που καλωσορίζει το χρήστη στο πρόγραμμα εγκατάστασης.



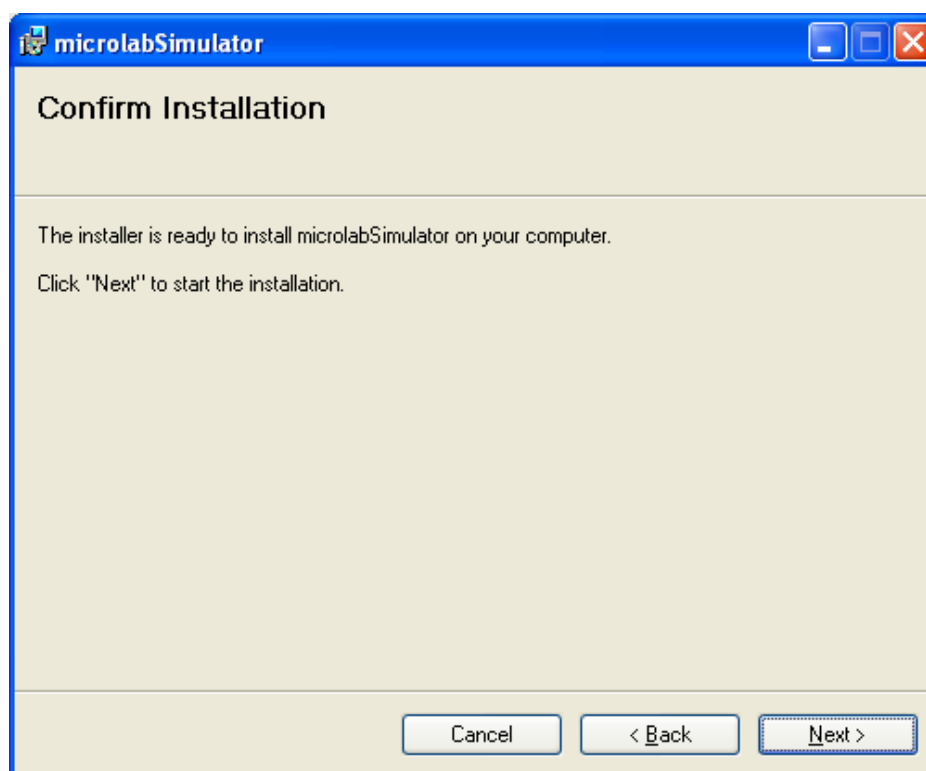
Οθόνη 1-2 : Πρώτη οθόνη εγκατάστασης του Microlab Simulator – TSIK

Για να συνεχίσετε πατήστε το πλήκτρο **“Next”** οπότε και θα εμφανιστεί μια οθόνη που σας επιτρέπει να καθορίσετε τις παραμέτρους της εγκατάστασης.



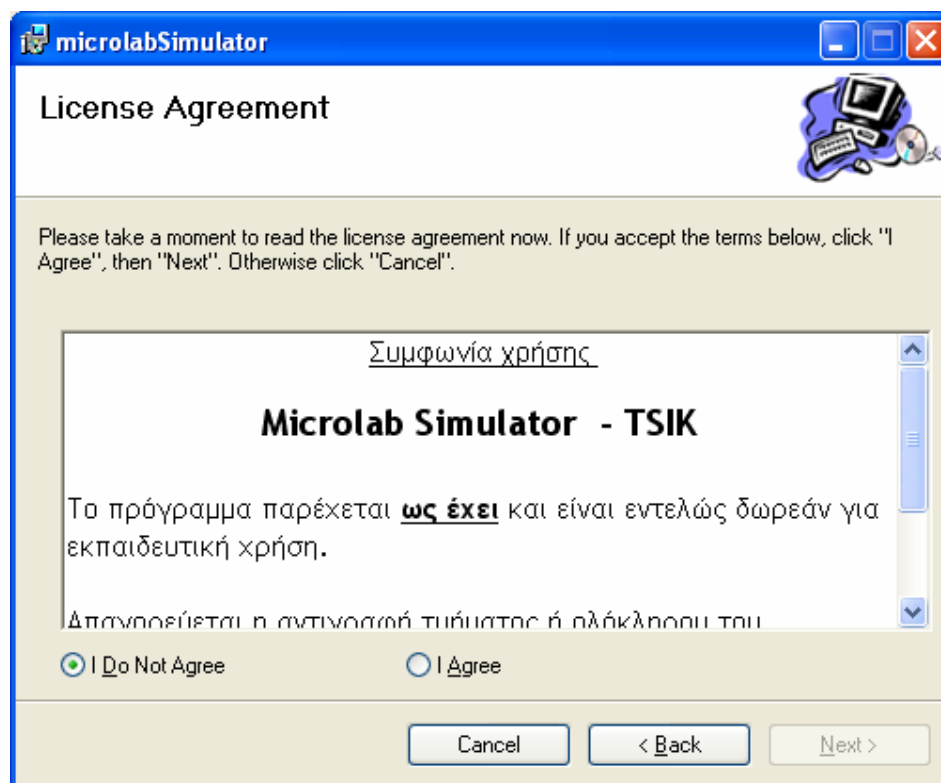
Οθόνη 1-3 : Καθορισμός παραμέτρων εγκατάστασης του Microlab Simulator – TSIK

Η επόμενη οθόνη σας ζητά να επιβεβαιώσετε τις προηγούμενες ρυθμίσεις και σας παρέχει τη δυνατότητα να τις αλλάξετε αν το επιθυμείτε πατώντας το πλήκτρο **“Back”**. Για να συνεχίσετε πατήστε το πλήκτρο **“Next”**.



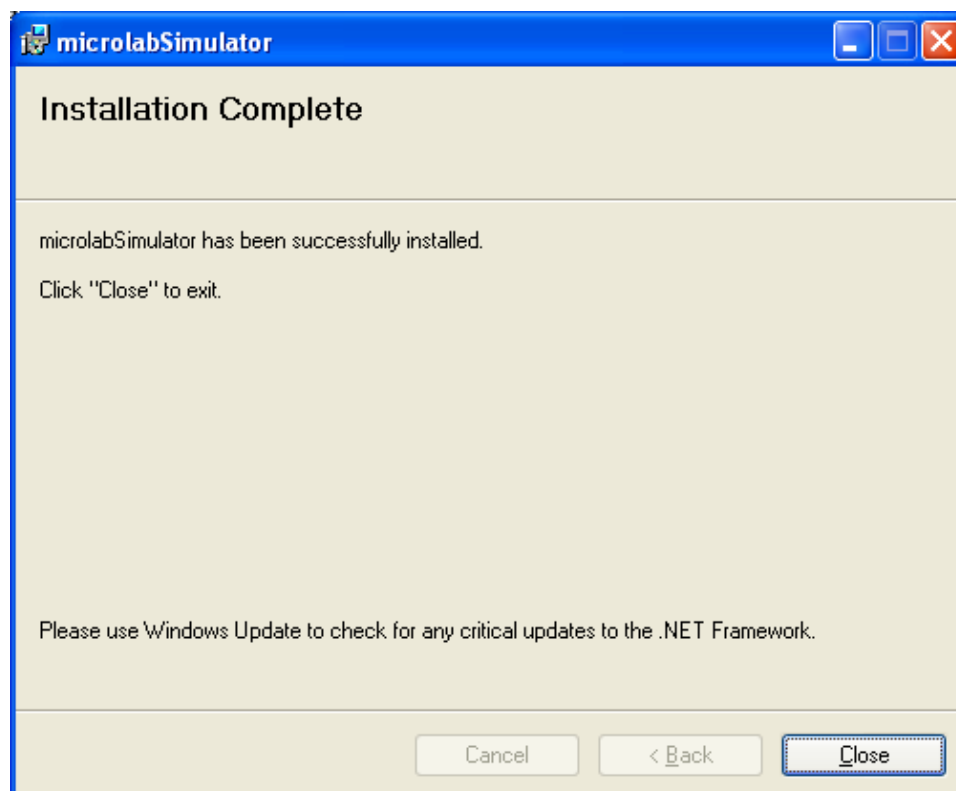
Οθόνη 1-4 : Επιβεβαίωση παραμέτρων εγκατάστασης του Microlab Simulator – TSIK

Στη συνέχεια η οθόνη που εμφανίζεται σας παρουσιάζει τη συμφωνία χρήσης. Προκειμένου να μπορέσετε να εγκαταστήσετε το πρόγραμμα πρέπει να αποδεχθείτε τους όρους χρήσης του. Αν συμφωνείτε με αυτούς πατήστε **“I Agree”** και στη συνέχεια το πλήκτρο **“Next”**.



Οθόνη 1-5 : Οθόνη συμφωνίας χρήσης του Microlab Simulator – TSIK

Το πρόγραμμα θα εγκατασταθεί τώρα στον υπολογιστή σας και στο τέλος της διαδικασίας θα δείτε την ακόλουθη οθόνη που σας ενημερώνει ότι η εγκατάσταση ήταν επιτυχής.



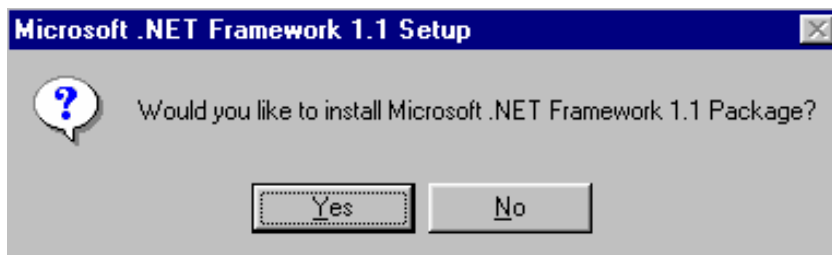
Οθόνη 1-6 : Ολοκλήρωση εγκατάστασης του Microlab Simulator – TSIK

### 1.3.2. Εγκατάσταση του Microsoft® .NET Framework 1.1

Εντοπίστε το αρχείο dotnetfx.exe που βρίσκεται στον φάκελο (directory) setup/prerequisites/framework του CD και εκτελέστε το. Αν δεν έχετε το CD τότε μπορείτε να ανακτήσετε (download) το αρχείο αυτό από την ιστοσελίδα (site) της Microsoft® ακολουθώντας το σύνδεσμο (link) :

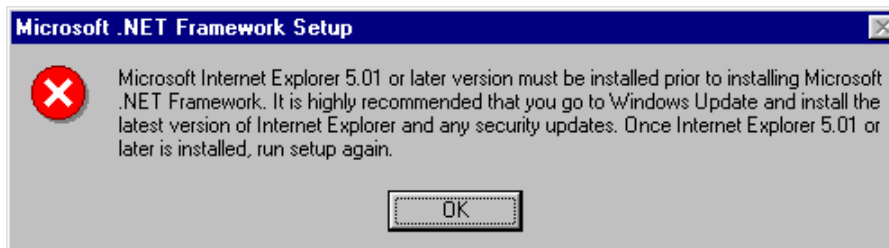
<http://www.download.microsoft.com/download/e/b/2/eb247c2a-e6b3-4694-98a2-b27111d233dd/dotnetredist.exe>

Όταν εκτελέστε το αρχείο dotnetfx.exe θα εμφανιστεί η οθόνη :



Οθόνη 1-7 : Εγκατάσταση του Microsoft® .NET Framework 1.1

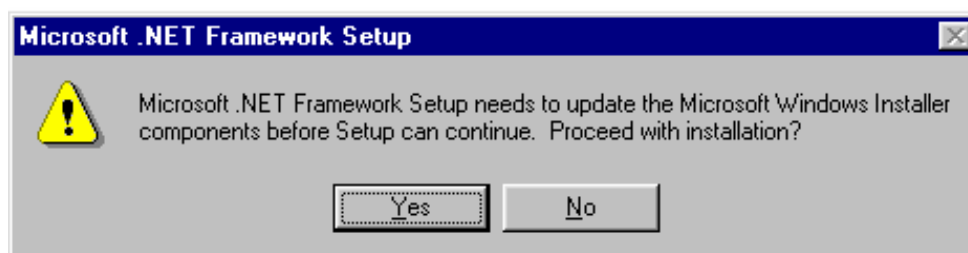
Πατήστε το “YES” προκειμένου να συνεχιστεί η εγκατάσταση. Αν στο σύστημα δεν υπάρχει εγκατεστημένος ο Internet Explorer® ή αυτός που είναι εγκατεστημένος είναι προγενέστερος του 5.01 το πρόγραμμα εγκατάστασης του Microsoft® .NET Framework 1.1 θα εμφανίσει την οθόνη :



Οθόνη 1-8 : Οθόνη που εμφανίζεται όταν δεν υπάρχει εγκατεστημένη η κατάλληλη έκδοση του Internet Explorer®

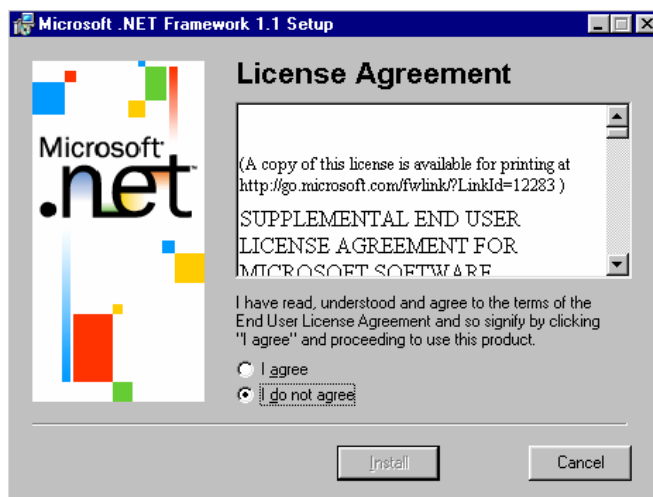
Προκειμένου να εγκαταστήσετε τον Internet Explorer® 6.0 δείτε την παράγραφο 5.3.3 που ακολουθεί.

Στην περίπτωση που υπάρχει εγκατεστημένη μια έκδοση του Internet Explorer® νεότερη της 5.01 το πρόγραμμα εγκατάστασης του Microsoft® .NET Framework 1.1 θα συνεχίσει κανονικά. Ενδέχεται (χωρίς αυτό να είναι βέβαιο) να εμφανιστεί η ακόλουθη οθόνη ζητώντας σας να επιτρέψετε την ανανέωση ορισμένων αρχείων του λειτουργικού σας συστήματος με νεότερες εκδόσεις τους. Για να μπορέσετε να συνεχίσετε επιτρέψετε την ανανέωση πατώντας το πλήκτρο “Yes”.



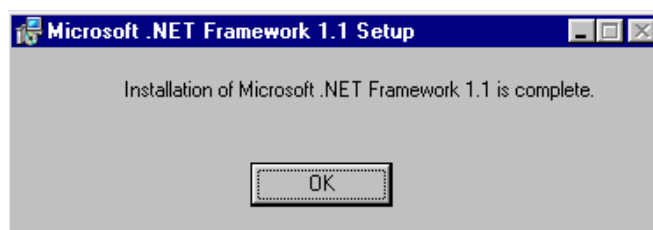
Οθόνη 1-9 : Οθόνη Update Microsoft Windows Installer Components

Στο σημείο αυτό θα παρουσιαστεί η οθόνη συμφωνίας δικαιωμάτων (License Agreement). Προκειμένου να συνεχίσει η εγκατάσταση πρέπει να πατήσετε το “I Agree” και στη συνέχεια το πλήκτρο “Install”.



Οθόνη 1-10 : License Agreement για το Microsoft® .NET Framework 1.1

Όταν η εγκατάσταση του Microsoft® .NET Framework 1.1 ολοκληρωθεί θα εμφανιστεί η ακόλουθη οθόνη.

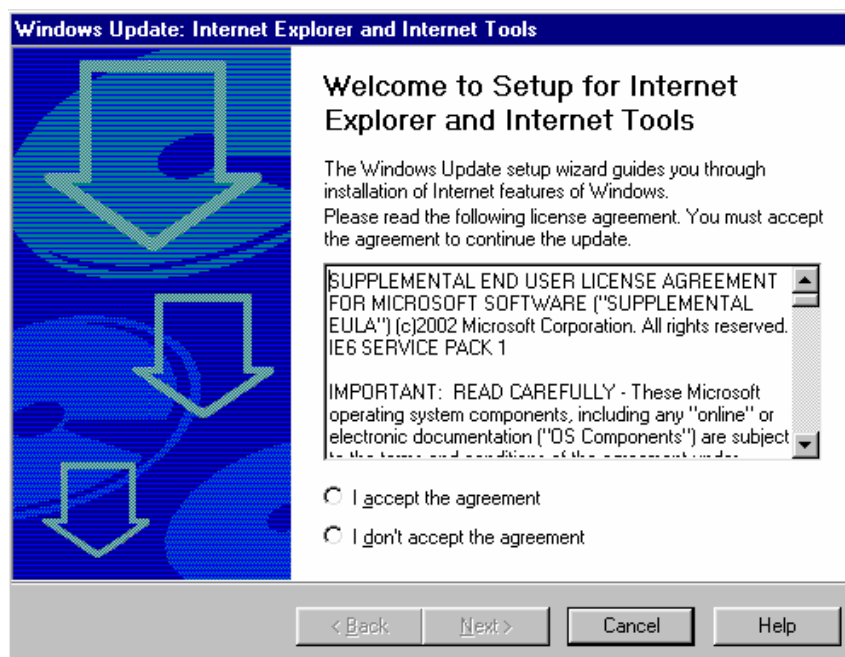


Οθόνη 1-11 : Οθόνη τέλους εγκατάστασης του Microsoft® .NET Framework 1.1

### **1.3.3. Εγκατάσταση του Internet Explorer® 6.0**

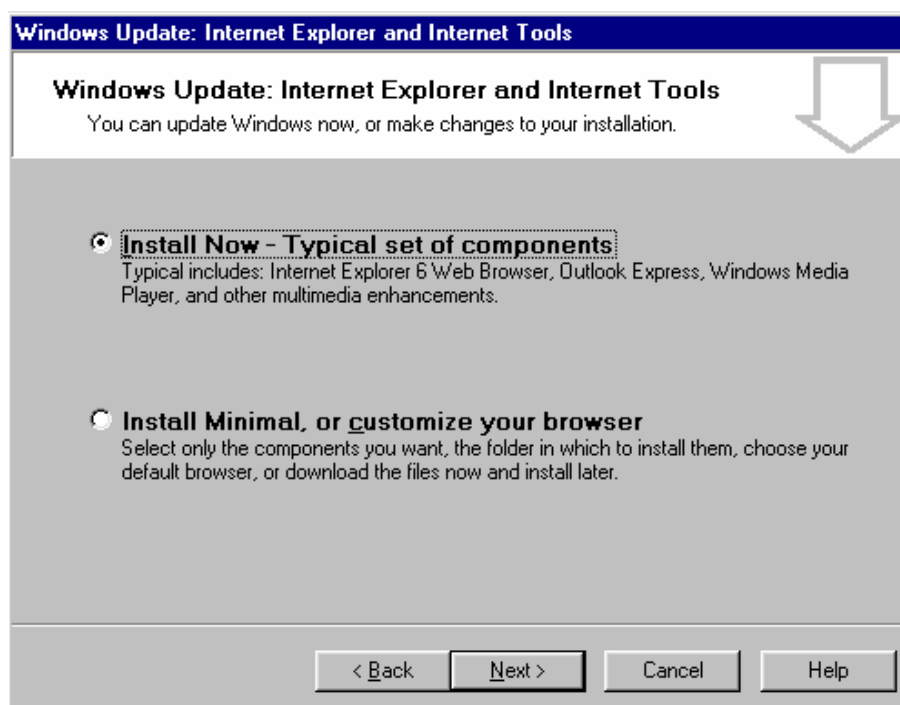
Το αρχείο εγκατάστασης του Internet Explorer® 6.0 (ie6setup.exe) βρίσκεται στο φάκελο setup/prerequisites/IE του CD.

Όταν εκτελέσετε το αρχείο ie6setup.exe θα εμφανιστεί η επόμενη οθόνη. Πατήστε το “**I accept the agreement**” και μετά το πλήκτρο “**Next**”.



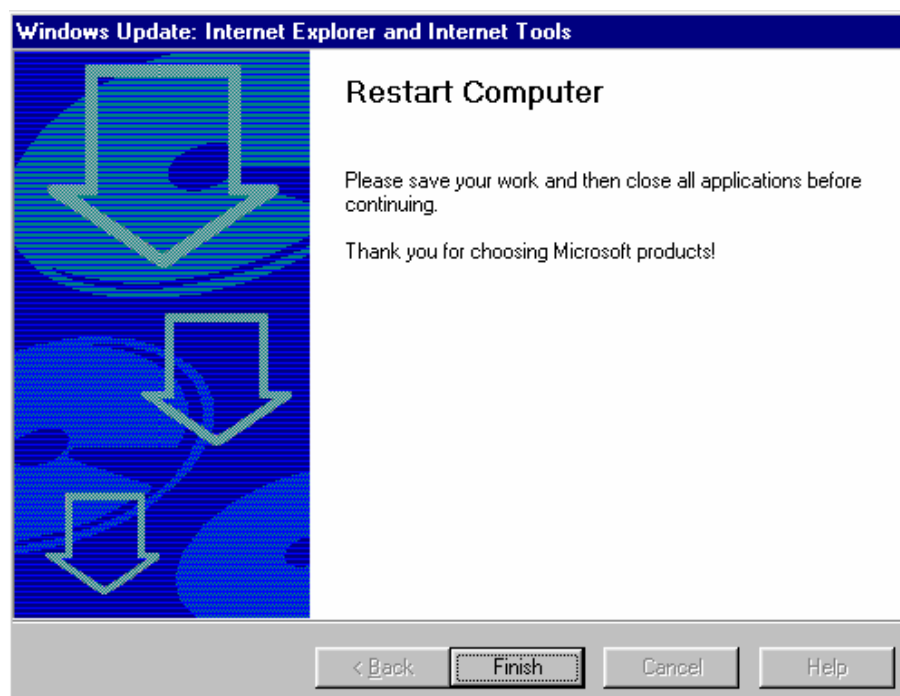
Οθόνη 1-12 : Πρώτη οθόνη εγκατάστασης του Internet Explorer® 6.0

Η οθόνη που ακολουθεί θα σας ζητήσει να επιλέξετε τον τύπο της εγκατάστασης. Κρατήστε την πρώτη επιλογή και πατήστε το πλήκτρο **“Next”**.



Οθόνη 1-13 : Επιλογή τύπου εγκατάστασης του Internet Explorer® 6.0

Στη συνέχεια θα σας ζητηθεί να επανεκκινήσετε (restart) τον υπολογιστή. Μόλις αυτό γίνει η εγκατάσταση του Internet Explorer® 6.0 θα έχει ολοκληρωθεί.



Οθόνη 1-14 : Για την ολοκλήρωση της εγκατάστασης του Internet Explorer® 6.0 απαιτείται επανεκκίνηση

## 1.4 Αρχεία και συντομεύσεις

Στην επιφάνεια εργασίας (desktop) θα δημιουργηθεί μια συντόμευση (shortcut) προς το πρόγραμμα η οποία θα έχει τη μορφή :



Εικόνα 1-1: Το εικονίδιο του προγράμματος

Επίσης όλα τα αρχεία με επέκταση (extension) .8085 θα συσχετιστούν (file association) με το πρόγραμμα. Αυτό σημαίνει ότι με διπλό κλικ (double click) σε οποιοδήποτε αρχείο με επέκταση .8085 θα ανοίγει το πρόγραμμα MicroLab Simulator – TSIK φορτώνοντας ταυτόχρονα το συγκεκριμένο αρχείο.

## 1.5 Βασικές Λειτουργίες

### 1.5.1. Κειμενογράφος (Text Editor)

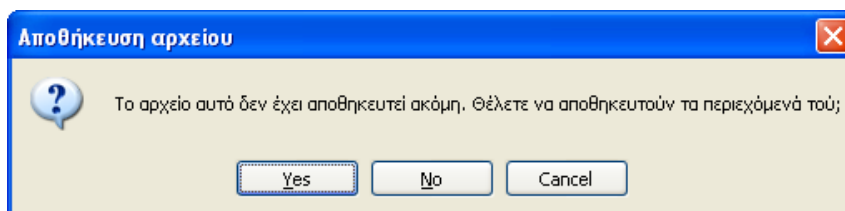
Ο κειμενογράφος του MicroLab Simulator – TSIK υποστηρίζει όλες τις συνήθεις εργασίες επεξεργασίας κειμένου, δηλαδή :

#### 1.5.1.1. Δημιουργία νέου αρχείου (New File)

Συντόμευση : Control + N

Εικονίδιο : 

Δημιουργεί ένα νέο αρχείο και καθαρίζει το χώρο κειμένου. Αν όταν πατηθεί το πλήκτρο αυτό στο χώρο κειμένου υπάρχει μη αποθηκευμένο κείμενο τότε ο χρήστης θα ρωτηθεί αν θέλει να το αποθηκεύσει πριν τη δημιουργία του νέου αρχείου. Η περίπτωση αυτή περιγράφεται από την οθόνη που ακολουθεί :



Οθόνη 1-15 : Πλήκτρο “Νέο αρχείο” με μη αποθηκευμένα δεδομένα στο χώρο κειμένου

- Αν ο χρήστης πατήσει **“Yes”** τότε θα του δοθεί η δυνατότητα να αποθηκεύσει το προηγούμενο κείμενο πριν δημιουργηθεί το νέο αρχείο.
- Αν πατήσει **“No”** τότε το νέο αρχείο θα δημιουργηθεί χωρίς να αποθηκευτεί το κείμενο που υπήρχε πριν στο χώρο κειμένου.
- Αν ο χρήστης πατήσει το **“Cancel”** τότε δεν θα εκτελεστεί η λειτουργία δημιουργίας νέου αρχείου.

#### 1.5.1.2. Άνοιγμα αρχείου (Open File)

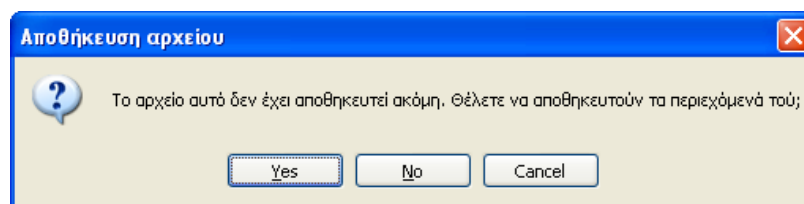
Συντόμευση : Control + O

Εικονίδιο : 

Φορτώνει στο χώρο κειμένου ένα ήδη υπάρχον αρχείο. Αν όταν πατηθεί το πλήκτρο αυτό στο χώρο κειμένου υπάρχει μη αποθηκευμένο κείμενο τότε ο χρήστης θα ρωτηθεί αν θέλει να το



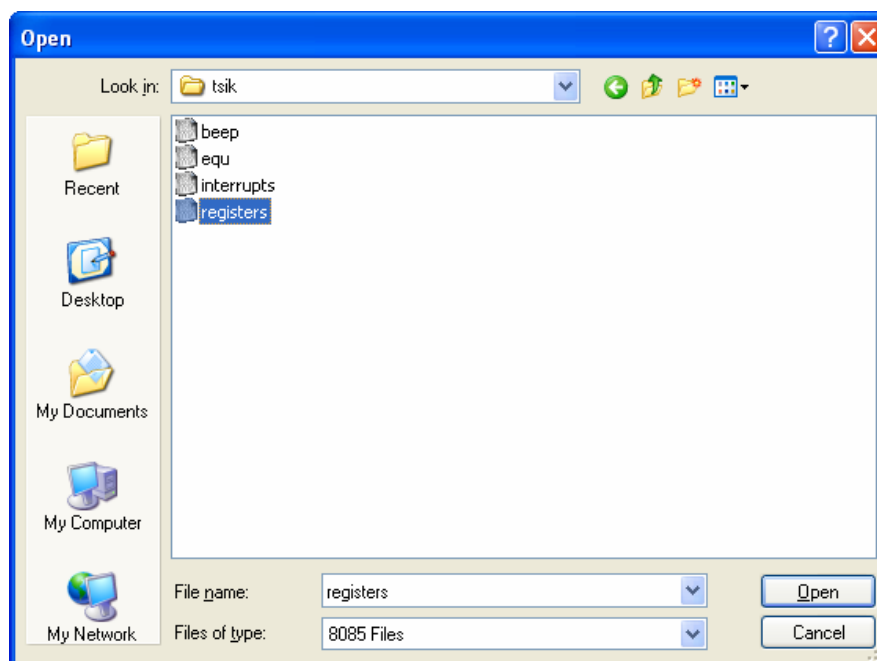
αποθηκεύσει πριν το άνοιγμα του νέου αρχείου. Η περίπτωση αυτή περιγράφεται από την οθόνη που ακολουθεί :



Οθόνη 1-16 : Πλήκτρο “Άνοιγμα αρχείου” με μη αποθηκευμένα δεδομένα στο χώρο κειμένου

- Αν ο χρήστης πατήσει **“Yes”** τότε θα του δοθεί η δυνατότητα να αποθηκεύσει το προηγούμενο κείμενο πριν φορτωθεί το νέο αρχείο.
- Αν πατήσει **“No”** τότε το νέο αρχείο θα φορτωθεί χωρίς να αποθηκευτεί το κείμενο που υπήρχε πριν στο χώρο κειμένου.
- Αν ο χρήστης πατήσει το **“Cancel”** τότε δεν θα εκτελεστεί η λειτουργία ανοίγματος αρχείου.

Σε αντίθετη περίπτωση, δηλαδή αν δεν υπάρχει μη αποθηκευμένο κείμενο στο χώρο κειμένου, θα ανοίξει ένας διάλογος που επιτρέπει στο χρήστη να επιλέξει το αρχείο που θα φορτωθεί στον χώρο κειμένου. Ο διάλογος αυτός επιτρέπει την επιλογή μόνο εκείνων των αρχείων που έχουν επέκταση (extension) .8085.



Οθόνη 1-17 : Διάλογος για το άνοιγμα αρχείου

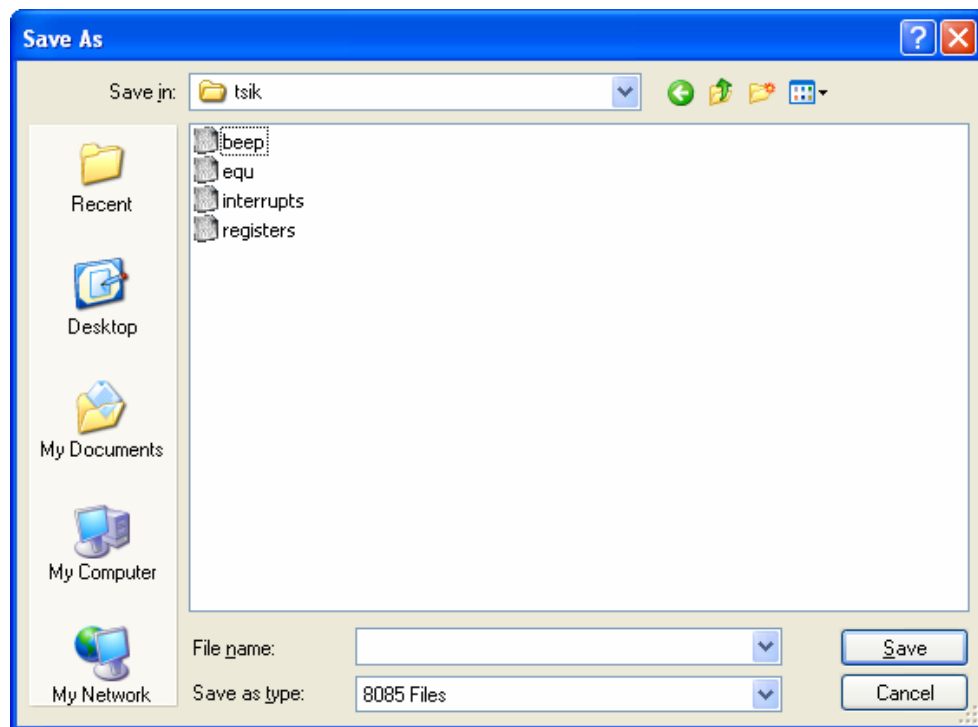
### 1.5.1.3. Αποθήκευση αρχείου (Save File)

Συντόμευση : Control + S

Εικονίδιο : 

Αποθηκεύει το περιεχόμενο του χώρου κειμένου σε ένα αρχείο. Αν το αρχείο είχε αποθηκευτεί προηγουμένως τότε απλά ανανεώνει το περιεχόμενο του αρχείου στο δίσκο με τα τρέχοντα περιεχόμενα του χώρου κειμένου.

Αν είναι η πρώτη φορά που θα αποθηκευτεί το αρχείο τότε ανοίγει ο διάλογος της επόμενης οθόνης που επιτρέπει στο χρήστη να επιλέξει το όνομα που θα του δώσει καθώς και το φάκελο (directory) στο οποίο θα το αποθηκεύσει.



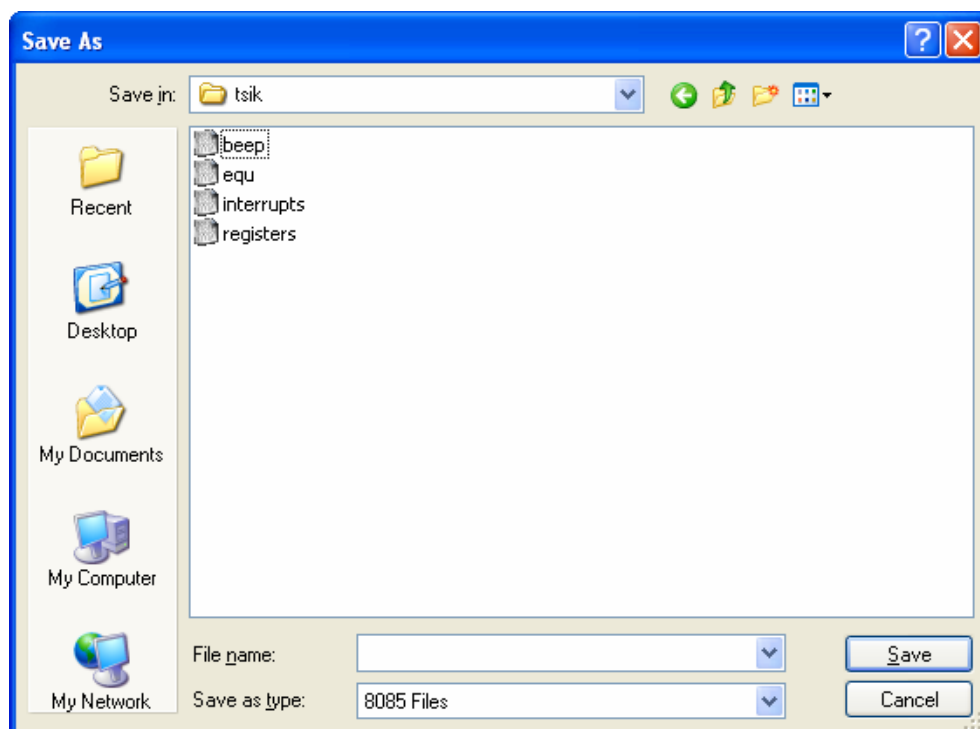
Οθόνη 1-18 : Διάλογος για αποθήκευση αρχείου

#### 1.5.1.4. Αποθήκευση αρχείου ως (Save File as)

Συντόμευση : Control + Shift + S

Εικονίδιο : 

Αποθηκεύει το περιεχόμενο του χώρου κειμένου σε ένα αρχείο. Ακόμη και αν το αρχείο είχε αποθηκευτεί προηγουμένως δίδεται η δυνατότητα στο χρήστη να αποθηκεύσει το περιεχόμενο του χώρου κειμένου σε ένα άλλο διαφορετικό αρχείο στο δίσκο.



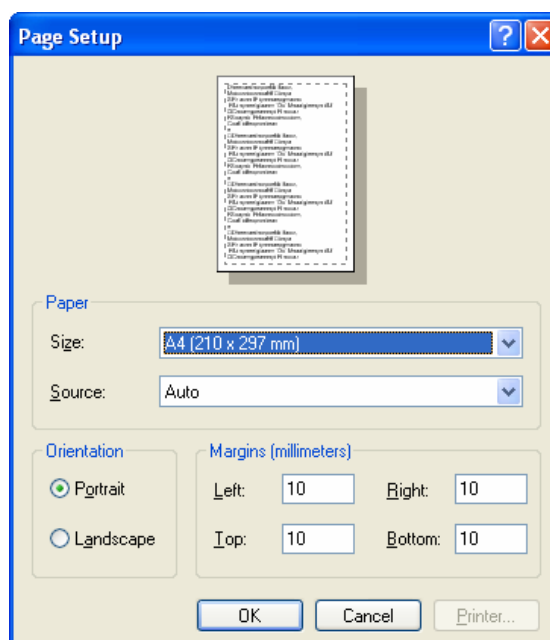
Οθόνη 1-19 : Διάλογος για αποθήκευση αρχείου ως

### 1.5.1.5. Ρύθμιση σελίδας εκτύπωσης (Page Setup)

Συντόμευση : Control + Shift + U

Εικονίδιο : 

Επιτρέπει τη ρύθμιση των περιθωρίων και της θέσης (orientation) της σελίδας. Οι ρυθμίσεις αυτές θα χρησιμοποιηθούν για την εκτύπωση των περιεχομένων του χώρου κειμένου.



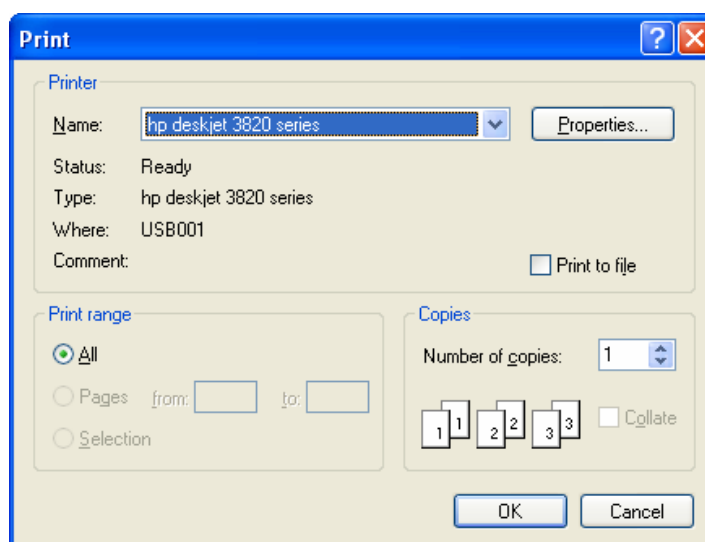
Οθόνη 1-20 : Ρύθμιση σελίδας εκτύπωσης

### 1.5.1.6. Εκτύπωση (Print)

Συντόμευση : Control + P

Εικονίδιο : 

Εκτυπώνει το περιεχόμενο του χώρου κειμένου. Κατά την εκτέλεση της εντολής αυτής εμφανίζεται ο διάλογος της παρακάτω οθόνης.



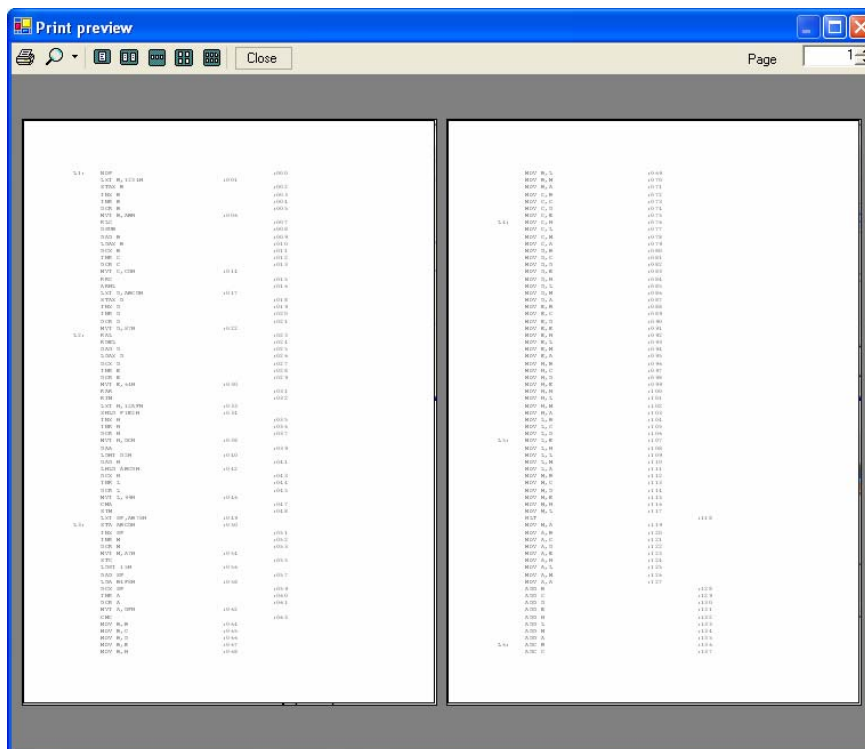
Οθόνη 1-21 : Διάλογος Εκτύπωσης

### 1.5.1.7. Επισκόπηση Εκτύπωσης (Print Preview)

Συντόμευση : Control + Shift + P

Εικονίδιο : 

Εμφανίζει την μορφή του κειμένου που θα εκτυπωθεί στην οθόνη, λαμβάνοντας υπόψη τις ρυθμίσεις σελίδας εκτύπωσης.



Οθόνη 1-22 : Παράδειγμα οθόνης επισκόπησης εκτύπωσης

### 1.5.1.8. Αποκοπή (Cut)

Συντόμευση : Control + X

Εικονίδιο : 

Αποκόπτει το επιλεγμένο τμήμα κειμένου από το χώρο κειμένου και το τοποθετεί στο δοχείο αποκομμάτων (clipboard).

### 1.5.1.9. Αντιγραφή (Copy)

Συντόμευση : Control + C

Εικονίδιο : 

Αντιγράφει το επιλεγμένο τμήμα κειμένου από το χώρο κειμένου στο δοχείο αποκομμάτων (clipboard).

### 1.5.1.10. Επικόλληση (Paste)

Συντόμευση : Control + V

Εικονίδιο : 

Αντιγράφει το τμήμα κειμένου που βρίσκεται στο δοχείο αποκομμάτων (clipboard) στο χώρο κειμένου, στο σημείο που βρίσκεται ο δρομέας (cursor).

**1.5.1.11. Αναιρέση (Undo)**

Συντόμευση : Control + Z

Εικονίδιο : 

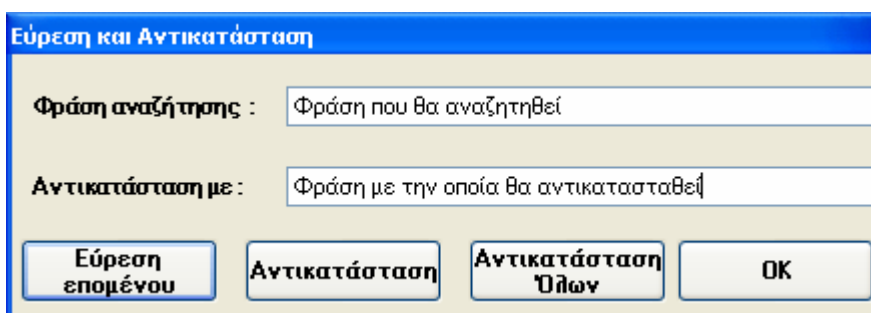
Αναιρεί την τελευταία δραστηριότητα που έγινε στο χώρο κειμένου.

**1.5.1.12. Εύρεση και αντικατάσταση (Find and replace)**

Συντόμευση : Control + F

Εικονίδιο : 

Επιτρέπει στο χρήστη να εντοπίσει μια συμβολοσειρά (string) μέσα στο χώρο κειμένου και αν επιθυμεί να την αντικαταστήσει με μια άλλη. Συγκεκριμένα παρουσιάζεται ο διάλογος που φαίνεται στην επόμενη οθόνη.



Οθόνη 1-23 : Οθόνης εύρεσης και αντικατάστασης

- Πλήκτρο **“Εύρεση επομένου”** : Βρίσκει την επόμενη θέση της φράσης αναζήτησης αρχίζοντας το ψάξιμο από την τρέχουσα θέση του δρομέα. Αν η φράση δε βρεθεί ενημερώνει το χρήστη με κατάλληλο μήνυμα.
- Πλήκτρο **“Αντικατάσταση”** : Αντικαθιστά την πρώτη παρουσία της φράσης αναζήτησης με την φράση αντικατάστασης.
- Πλήκτρο **“Αντικατάσταση όλων”** : Αντικαθιστά όλες τις παρουσίες της φράσης αναζήτησης με την φράση αντικατάστασης.
- Πλήκτρο **“OK”** : Κλείνει το παράθυρο εύρεσης και αντικατάστασης.

**1.5.1.13. Μεγέθυνση γραμματοσειράς**


Συντόμευση : Control + ]

Εικονίδιο : 

Αυξάνει το μέγεθος της γραμματοσειράς του χώρου κειμένου.

**1.5.1.14. Σμίκρυνση γραμματοσειράς**

Συντόμευση : Control + [

Εικονίδιο : 

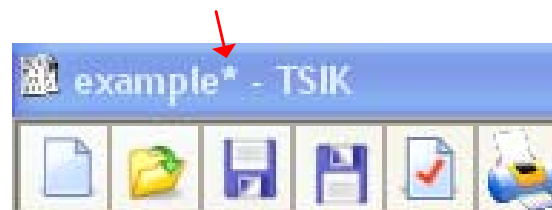
Μειώνει το μέγεθος της γραμματοσειράς του χώρου κειμένου.

**1.5.1.15. Ένδειξη μη αποθήκευσης**

Το πρόγραμμα δείχνει ανά πάσα στιγμή στο χρήστη κατά πόσον το περιεχόμενο του χώρου κειμένου είναι αποθηκευμένο. Συγκεκριμένα όταν το κείμενο στο χώρο κειμένου δεν έχει ακόμη αποθηκευτεί, ή αν έχει τροποποιηθεί από την προηγούμενη φορά που έχει αποθηκευτεί ένας αστερίσκος '\*' θα φαίνεται δίπλα από το όνομα αρχείου.



Δεν υπάρχει αστερίσκος οπότε το περιεχόμενο του χώρου κειμένου είναι αποθηκευμένο.



Υπάρχει αστερίσκος οπότε το περιεχόμενο του χώρου κειμένου δεν είναι αποθηκευμένο.

Οθόνη 1-24 : Οθόνης ένδειξης μη αποθήκευσης

**1.5.2. Βοηθητικά εργαλεία****1.5.2.1. Μετατροπéας μεταξύ συστημάτων**

Συντόμευση : F3

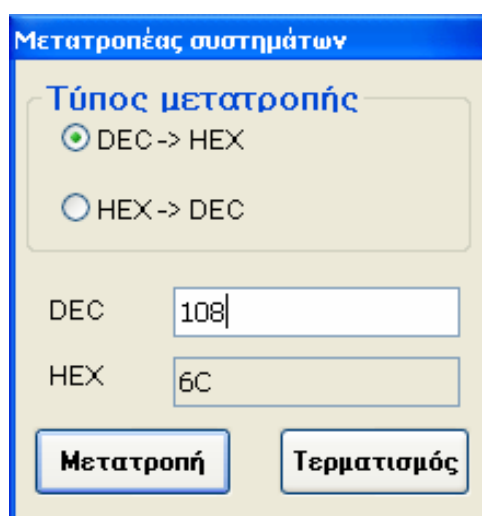
Εικονίδιο : 

Πρόκειται για ένα εργαλείο που επιτρέπει τη μετατροπή αριθμών μεταξύ του δεκαδικού και του δεκαεξαδικού συστήματος.

Ο τύπος μετατροπής ορίζεται ως εξής :

- DEC -> HEX : Για μετατροπή από το δεκαδικό στο δεκαεξαδικό και
- HEX -> DEC : Για μετατροπή από το δεκαεξαδικό στο δεκαδικό

Το πλήκτρο “**Μετατροπή**” εκτελεί τη μετατροπή και το πλήκτρο “**Τερματισμός**” κλείνει το παράθυρο του μετατροπέα.



Οθόνη 1-25 : Οθόνη μετατροπέα συστημάτων

**1.5.2.2. Περί (About)**

Συντόμευση : -

Εικονίδιο : 

Δίνει πληροφορίες για τον κατασκευαστή του προγράμματος καθώς και τη δυνατότητα επικοινωνίας μαζί του με αποστολή ηλεκτρονικού ταχυδρομείου (email).



Οθόνη 1-26 : Οθόνη περί (About)

Όταν ο χρήστης κάνει κλικ με το ποντίκι πάνω στην ηλεκτρονική διεύθυνση **tsik@microlab.ntua.gr** τότε αυτόματα ανοίγει το προεπιλεγμένο πρόγραμμα αποστολής ηλεκτρονικού ταχυδρομείου του συστήματός του (default email program). Ως παραλήπτης του ηλεκτρονικού μηνύματος είναι ο κατασκευαστής του προγράμματος Microlab Simulator - TSIK. Ο χρήστης μπορεί να γράψει τα σχόλια, τις παρατηρήσεις του για το πρόγραμμα.

**Μηνύματα με λάθη του προγράμματος  
και παρατηρήσεις είναι ευπρόσδεκτα.**

Το πλήκτρο “Έξοδος” κλείνει τη συγκεκριμένη οθόνη.

**1.5.3. Μεταφραστής (assembler)**

Όπως αναφέρθηκε πολλές φορές σε προηγούμενα κεφάλαια το πρόγραμμα μέσω του μεταφραστή που διαθέτει προσφέρει στο χρήστη την δυνατότητα αυτόματης μετατροπής μεταξύ συμβολικής γλώσσας (assembly) και γλώσσας μηχανής.

Ο μεταφραστής υποστηρίζει όλες τις εντολές του 8085 (ακόμη και τις κρυφές) και τις οδηγίες :

- EQU : Δήλωση σταθεράς
- ORG : Θέση μετάφρασης
- END : Τέλος μετάφρασης
- DB : Τοποθέτηση δεδομένων σε συγκεκριμένη θέση μνήμης

Δίνει επίσης την δυνατότητα στο χρήστη να γράψει σχόλια (comments) στο πρόγραμμά του.

**1.5.3.1. Μετάφραση του προγράμματος**

Η μετάφραση του προγράμματος που βρίσκεται στο χώρο κειμένου μπορεί να γίνει πατώντας το πλήκτρο F5 ή το εικονίδιο που φαίνεται στο επόμενο πλαίσιο.

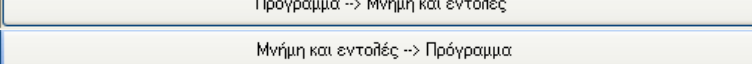
Συντόμευση : F5

Εικονίδιο : 

**1.5.3.2. Μνήμη και εντολές**

Το πρόγραμμα παρέχει τη δυνατότητα στο χρήστη να δει το μεταφρασμένο πρόγραμμα όπως αυτό έχει τοποθετηθεί στη μνήμη του συστήματος. Δίπλα από κάθε θέση μνήμης φαίνεται η εντολή που υπάρχει εκεί.

Συντόμευση : Control + ~

Εικονίδιο : 

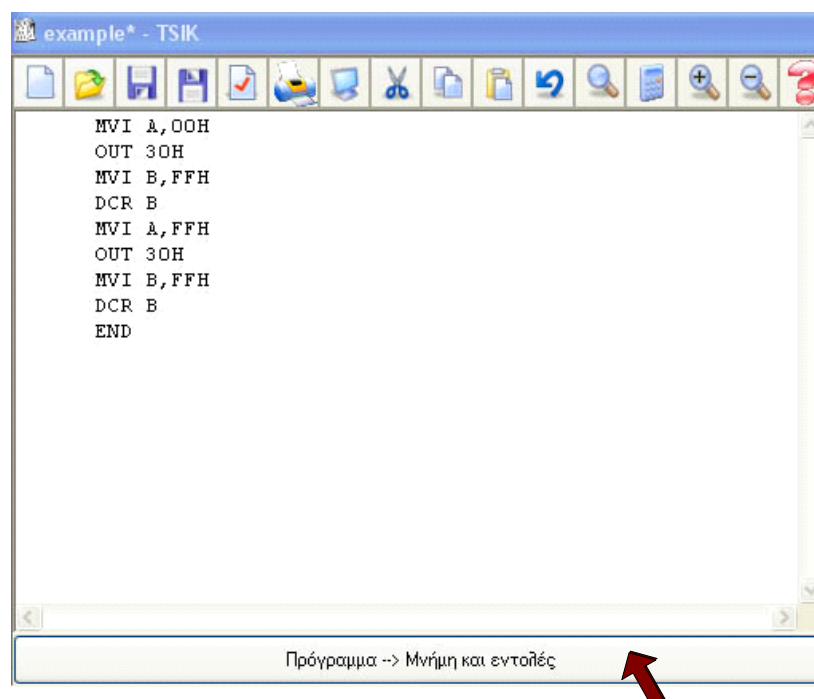
Έστω για παράδειγμα το πρόγραμμα :

```
MVI A,00H
OUT 30H
MVI B,FFH
DCR B
MVI A,FFH
OUT 30H
MVI B,FFH
DCR B

END
```

*Πρόγραμμα 1-1: Παράδειγμα προγράμματος*

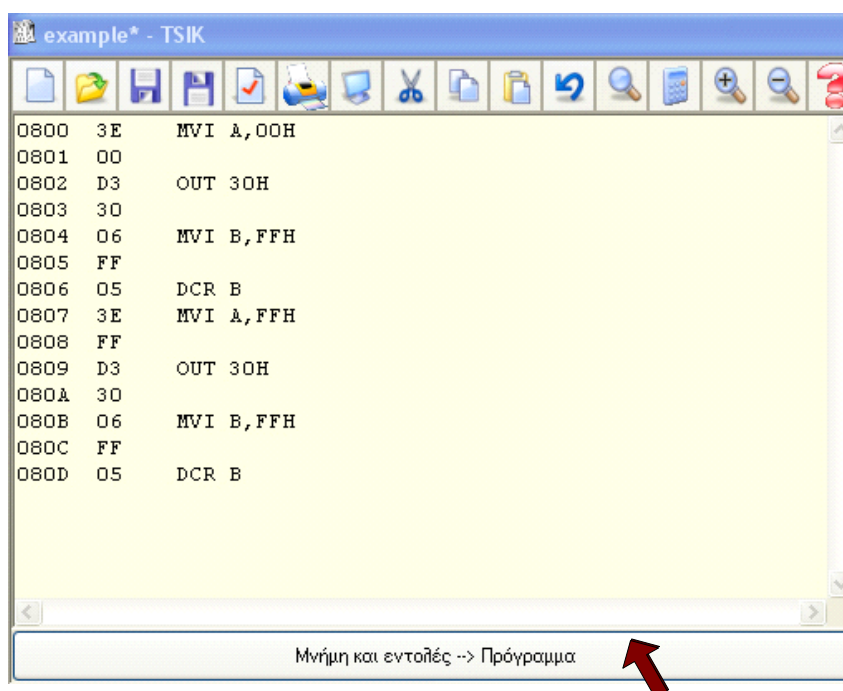
Για να φανεί το μεταφρασμένο πρόγραμμα μπορεί να πατηθεί η συντόμευση Control + ~ ή να πατηθεί το πλήκτρο “**Πρόγραμμα → Μνήμη και εντολές**” όπως φαίνεται στην επόμενη οθόνη.



*Οθόνη 1-27 : Πρόγραμμα → Μνήμη και εντολές (1)*



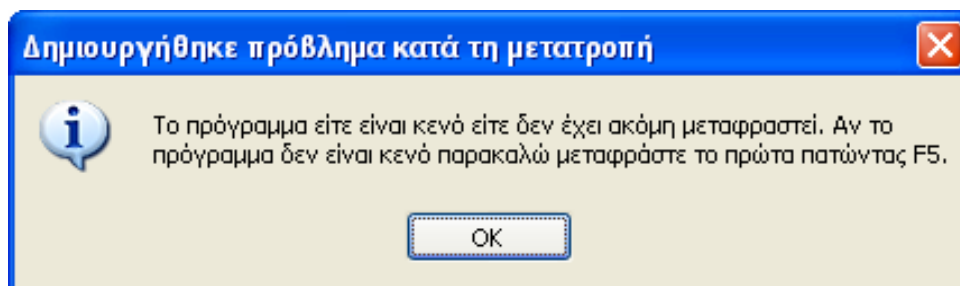
Έτσι θα φανεί το ίδιο πρόγραμμα μεταφρασμένο :



Οθόνη 1-28 : Πρόγραμμα → Μνήμη και εντολές (2)

Χρησιμοποιώντας την ίδια συντόμευση (Control + ~) ή πατώντας το πλήκτρο **“Μνήμη και εντολές → Πρόγραμμα”** θα γίνει μετάβαση στην προηγούμενη κατάσταση.

Στην περίπτωση που το πρόγραμμα που γράφηκε στο χώρο κειμένου είναι κενό, ή δεν έχει ακόμη μεταφραστεί θα παραχθεί ένα μήνυμα λάθους.



Οθόνη 1-29 : Μήνυμα λάθους κατά τη μετατροπή

Στον πίνακα που ακολουθεί στην επόμενη σελίδα φαίνονται όλες οι εντολές και οι οδηγίες που υποστηρίζονται από το μεταφραστή με τον ενδεδειγμένο τρόπο σύνταξής τους.

Εντολές 8085				
ACI byte	DCR A	MOV A,C	MOV L,M	RST 4
ADC A	DCR B	MOV A,D	MOV M,A	RST 5
ADC B	DCR C	MOV A,E	MOV M,B	RST 6
ADC C	DCR D	MOV A,H	MOV M,C	RST 7
ADC D	DCR E	MOV A,L	MOV M,D	RSTV
ADC E	DCR H	MOV A,M	MOV M,E	RZ
ADC H	DCR L	MOV B,A	MOV M,H	RET
ADC L	DCR M	MOV B,B	MOV M,L	SBB A
ADC M	DCX B	MOV B,C	MVI A,byte	SBB B
ADD A	DCX D	MOV B,D	MVI B,byte	SBB C
ADD B	DCX H	MOV B,E	MVI C,byte	SBB D
ADD C	DCX SP	MOV B,H	MVI D,byte	SBB E
ADD D	DI	MOV B,L	MVI E,byte	SBB H
ADD E	DSUB	MOV B,M	MVI H,byte	SBB L
ADD H	EI	MOV C,A	MVI L,byte	SBB M
ADD L	HLT	MOV C,B	MVI M,byte	SBI byte
ADD M	IN byte	MOV C,C	NOP	SHLD doubleByte
ADI byte	INR A	MOV C,D	ORA A	SHLX
ANA A	INR B	MOV C,E	ORA B	SIM
ANA B	INR C	MOV C,H	ORA C	SPHL
ANA C	INR D	MOV C,L	ORA D	STA doubleByte
ANA D	INR E	MOV C,M	ORA E	STAX B
ANA E	INR H	MOV D,A	ORA H	STAX D
ANA H	INR L	MOV D,B	ORA L	STC
ANA L	INR M	MOV D,C	ORA M	SUB A
ANA M	INX B	MOV D,D	ORI byte	SUB B
ANI	INX D	MOV D,E	OUT	SUB C
ARHL	INX H	MOV D,H	PCHL	SUB D
CALL label	INX SP	MOV D,L	POP B	SUB E
CC label	JC label	MOV D,M	POP D	SUB H
CM label	JM label	MOV E,A	POP H	SUB L
CMA	JMP label	MOV E,B	POP PSW	SUB M
CMC	JNC label	MOV E,C	PUSH B	SUI byte
CMP A	JNX5 label	MOV E,D	PUSH D	XCHG
CMP B	JNZ label	MOV E,E	PUSH H	XRA A
CMP C	JP label	MOV E,H	PUSH PSW	XRA B
CMP D	JPE label	MOV E,L	RAL	XRA C
CMP E	JPO label	MOV E,M	RAR	XRA D
CMP H	JX5 label	MOV H,A	RC	XRA E
CMP L	JZ label	MOV H,B	RDEL	XRA H
CMP M	LDA doubleByte	MOV H,C	RIM	XRA L
CNC label	LDAX B	MOV H,D	RLC	XRA M
CNZ label	LDAX D	MOV H,E	RM	XRI byte
CP label	LDHI byte	MOV H,H	RNC	XTHL
CPE labell	LDSI byte	MOV H,L	RNZ	
CPI byte	LHLD doubleByte	MOV H,M	RP	Οδηγίες προς τον assembler
CPO label	LHLX	MOV L,A	RPE	
CZ label	LXI B,doubleByte	MOV L,B	RPO	
DAA	LXI D,doubleByte	MOV L,C	RRC	
DAD B	LXI H,doubleByte	MOV L,D	RST 0	DB byte
DAD D	LXI SP,doubleByte	MOV L,E	RST 1	ORG doubleByte
DAD H	MOV A,A	MOV L,H	RST 2	EQU name,byte
DAD SP	MOV A,B	MOV L,L	RST 3	EQU name,doubleByte

Πίνακας 1-1: Έγκυρες εντολές για τον μεταφραστή (assembler)

<b>byte</b>	Είναι ένας δεκαεξαδικός αριθμός <u>δύο</u> ψηφίων που τελειώνει με το χαρακτήρα 'H'. Για παράδειγμα τα 01H ή ABH είναι έγκυρα ενώ τα S2H ή 28 ή 2H είναι άκυρα
<b>doubleByte</b>	Είναι ένας δεκαεξαδικός αριθμός <u>τεσσάρων</u> ψηφίων που τελειώνει με το χαρακτήρα 'H'. Για παράδειγμα τα 0124H ή A25BH είναι έγκυρα ενώ τα S215H ή 228H ή 215A είναι άκυρα
<b>label</b>	Είναι μια ετικέτα. Για να είναι έγκυρη μια εντολή που ακολουθείται από μια ετικέτα πρέπει η να υπάρχει κάπου στον πρόγραμμα το όνομα της ετικέτας αυτής ακολουθούμενο από το χαρακτήρα ':'
<b>name</b>	Είναι μια συμβολοσειρά με τουλάχιστον πέντε αλφαριθμητικούς χαρακτήρες που πρέπει να αρχίζει με χαρακτήρα και όχι αριθμό.

Πίνακας 1-2: Εξήγηση των ορισμών byte – doubleByte – label και name)

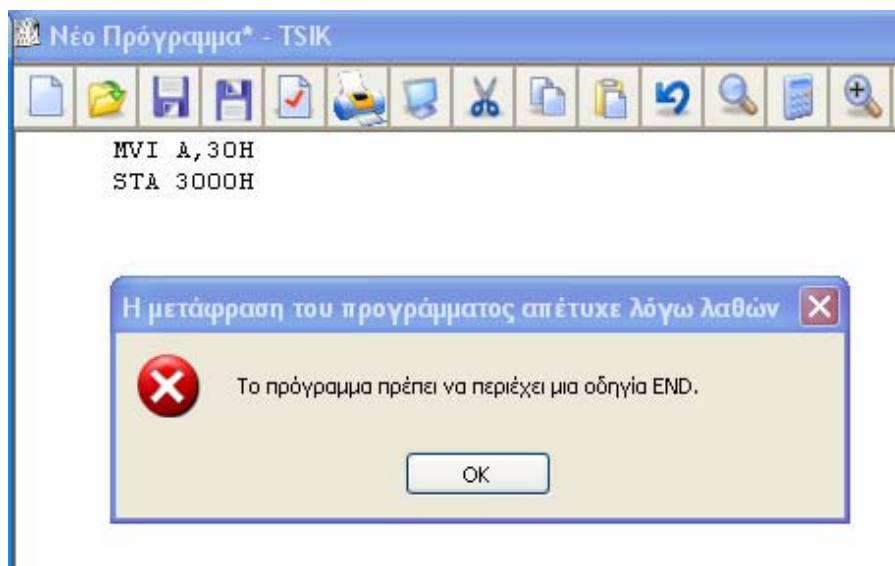
### 1.5.3.3. Οδηγία END

Κάθε μη κενό πρόγραμμα πρέπει να περιέχει μία τουλάχιστον οδηγία `END`. Για παράδειγμα ο μεταφραστής θα θεωρήσει ότι το πρόγραμμα που ακολουθεί δεν έχει ορθή σύνταξη.

```
MVI A, 30H
STA 3000H
```

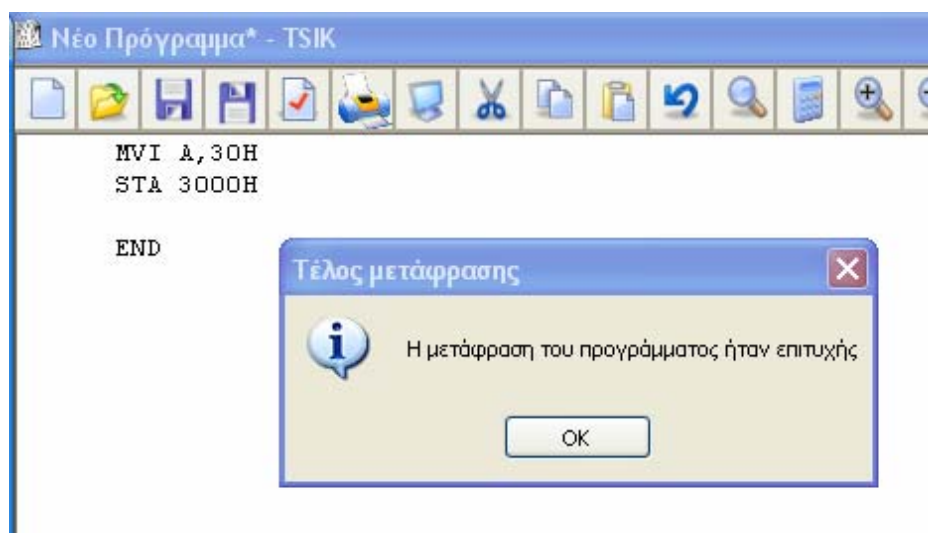
Πρόγραμμα 1-2: Κάθε πρόγραμμα πρέπει να περιέχει τουλάχιστον μια οδηγία `END` (1)

Στην επόμενη οθόνη φαίνεται το μήνυμα που παράγεται :



Οθόνη 1-30 : Κάθε πρόγραμμα πρέπει να περιέχει τουλάχιστον μια οδηγία `END` (1)

Αν τώρα προστεθεί μια οδηγία `END` οπουδήποτε μέσα στο πρόγραμμα τότε αυτό θα μεταφραστεί επιτυχώς.



Οθόνη 1-31 : Κάθε πρόγραμμα πρέπει να περιέχει τουλάχιστον μια οδηγία `END` (2)

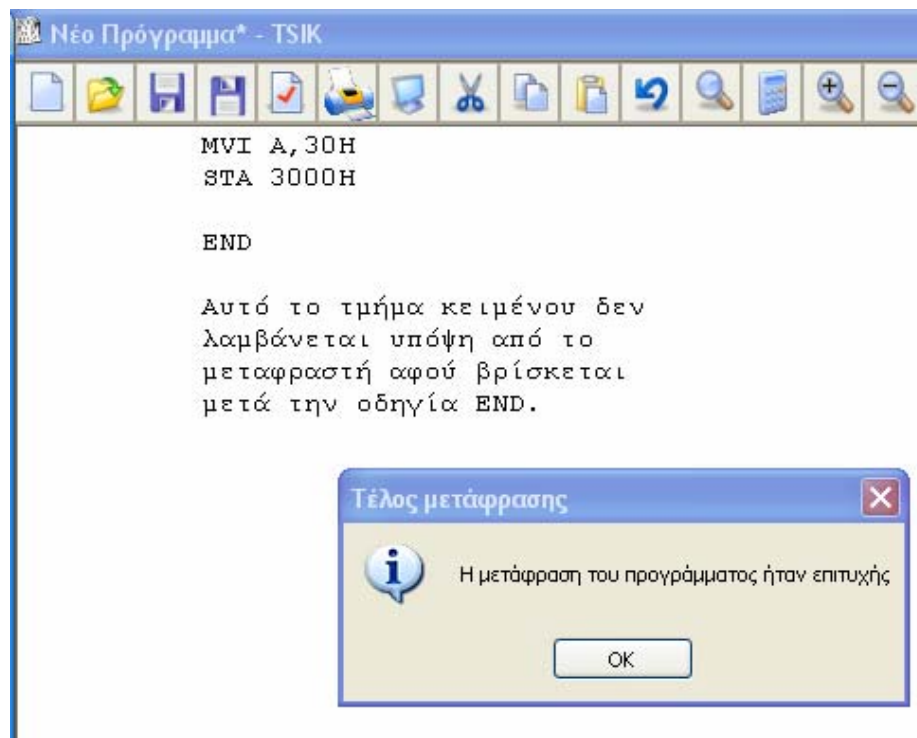
Πρέπει εδώ να σημειωθεί το τμήμα κειμένου που έπεται της οδηγίας `END` δεν λαμβάνεται υπόψη από το μεταφραστή. Για παράδειγμα το πρόγραμμα που ακολουθεί θα μεταφραστεί επιτυχώς.

```
MVI A, 30H
STA 3000H
```

```
END
```

Αυτό το τμήμα κειμένου δεν λαμβάνεται υπόψη από το μεταφραστή αφού βρίσκεται μετά την οδηγία END.

Πρόγραμμα 1-3: Οτιδήποτε βρίσκεται μετά την οδηγία END δεν λαμβάνεται υπόψη κατά τη μετάφραση (2)



Οθόνη 1-32 : Οτιδήποτε βρίσκεται μετά την οδηγία END δεν λαμβάνεται υπόψη κατά τη μετάφραση

#### 1.5.3.4. Οδηγία DB

Ο οδηγία DB τοποθετεί στην τρέχουσα θέση μνήμης, στη θέση μνήμης δηλαδή που βρίσκεται ο μεταφραστής όταν βρίσκει την οδηγία DB, το byte που ακολουθεί την οδηγία.

Για παράδειγμα το πρόγραμμα :

```
MVI A, 52H
DB 44H
INR A

END
```

Πρόγραμμα 1-4 : Παράδειγμα χρήσης της οδηγίας DB

μεταφράζεται ως :

0800	3E	MVI A, 52H
0801	52	
<b>0802</b>	<b>44</b>	<b>MOV B, H</b>
0803	3C	INR A

Πρόγραμμα 1-5 : Μετάφραση παραδείγματος χρήσης της οδηγίας DB

Όπως βλέπουμε εμφανίστηκε η εντολή `MOV B, H` στη θέση μνήμης 0802 χωρίς να έχει γραφεί στο πρόγραμμα. Αυτό οφείλεται στην οδηγία `DB` που καθόρισε την τοποθέτηση του byte 44 στη θέση μνήμης 0802. Ο δεκαεξαδικός αριθμός 44 αντιστοιχεί στην εντολή `MOV B, H`.

### 1.5.3.5. Οδηγία EQU

Η οδηγία `EQU` δηλώνει μια σταθερά. Παραδείγματα σύνταξης αυτής φαίνονται στη συνέχεια :

```
EQU NAME1, ABH
EQU NAME2, 3000H

END
```

Πρόγραμμα 1-6 : Παραδείγματα σύνταξης της οδηγίας EQU

όπου `name1` ή `name2` είναι μια συμβολοσειρά με τουλάχιστον πέντε αλφαριθμητικούς χαρακτήρες που πρέπει να αρχίζει με χαρακτήρα και όχι αριθμό.

Οπουδήποτε στο πρόγραμμα εμφανίζεται το `name1` θα είναι σαν να υπάρχει εκεί η τιμή `ABH` και οπουδήποτε εμφανίζεται το `name2` είναι σαν να υπάρχει εκεί το `3000H`.

Η τιμή με την οποία μπορεί να εξισωθεί μια σταθερά μπορεί να αποτελείται από 2 ή 4 bytes. Έτσι πρέπει να αποτελείται από 2 ή 4 δεκαεξαδικά ψηφία αντίστοιχα, που να έχουν στο τέλος τον χαρακτήρα 'H'.

Για παράδειγμα το πρόγραμμα :

```
MVI A, ALLON
STA OUTPORT

EQU ALLON, 00H
EQU OUTPORT, 3000H

END
```

Πρόγραμμα 1-7 : Παράδειγμα χρήσης της οδηγίας EQU

μεταφράζεται ως :

0800	3E	MVI A, 00H
0801	00	
0802	32	STA 3000H
0803	00	
0804	30	

Πρόγραμμα 1-8 : Μετάφραση παραδείγματος χρήσης της οδηγίας EQU

βλέπουμε ότι κατά τη μετάφραση του προγράμματος το `ALLON` που βρίσκεται στην πρώτη εντολή (`MVI A, ALLON`) αντικαταστάθηκε με την τιμή `00H`. Αντίστοιχα το `OUTPORT` (`STA OUTPORT`) αντικαταστάθηκε με το `3000H`.

**1.5.3.6. Οδηγία ORG**

Η οδηγία `ORG` ορίζει τη θέση μνήμης από την οποία αρχίζει η μετάφραση του προγράμματος.

Έστω για παράδειγμα το πρόγραμμα :

```
MVI A, 80H
STA 3000H
END
```

*Πρόγραμμα 1-9 : Πρόγραμμα χωρίς οδηγία ORG*

Στην περίπτωση αυτή το μεταφρασμένο πρόγραμμα θα είναι :

0800	3E	MVI A, 80H
0801	80	
0802	32	STA 3000H
0803	00	
0804	30	

*Πρόγραμμα 1-10: Μεταφρασμένο πρόγραμμα χωρίς οδηγία ORG*

Όπως φαίνεται από το προηγούμενο πρόγραμμα όταν δεν υπάρχει οδηγία `ORG` η μετάφραση αρχίζει από τη θέση μνήμης 0800 HEX.

Για να αρχίσει η μετάφραση του προγράμματος από τη θέση μνήμης 0810 HEX πρέπει να γράψουμε :

```
ORG 0810H
MVI A, 80H
STA 3000H

END
```

*Πρόγραμμα 1-11 : Οδηγία ORG*

Όπως φαίνεται από το επόμενο πλαίσιο οι θέσεις μνήμης από το 0800 μέχρι και την 080F γεμίζουν με εντολές `NOP`, δηλαδή με μηδενικά.

0800	00	NOP
0801	00	NOP
0802	00	NOP
0803	00	NOP
0804	00	NOP
0805	00	NOP
0806	00	NOP
0807	00	NOP
0808	00	NOP
0809	00	NOP
080A	00	NOP
080B	00	NOP
080C	00	NOP
080D	00	NOP
080E	00	NOP
080F	00	NOP
0810	3E	MVI A, 08H
0811	08	

*Πρόγραμμα 1-12 : Μεταφρασμένο πρόγραμμα με οδηγία ORG*

Η οδηγία `ORG` δεν μπορεί να χρησιμοποιηθεί για μετάφραση σε θέση μνήμης προηγούμενη από την τρέχουσα. Προκειμένου να γίνει αυτό κατανοητό παραθέτουμε το επόμενο παράδειγμα.

```
MVI A, 80H
ORG 0800H
STA 3000H

END
```

Πρόγραμμα 1-13 : Περίπτωση λάθους με την οδηγία `ORG`

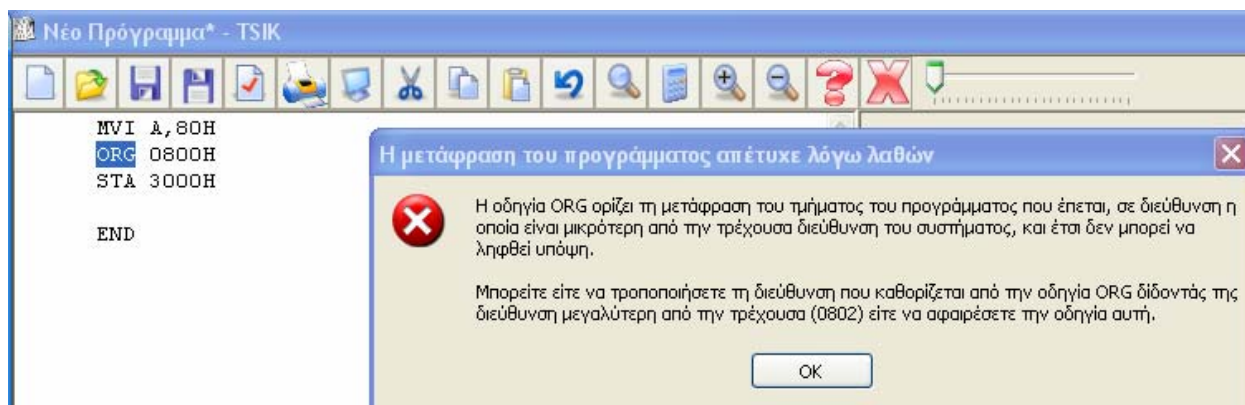
Η μετάφραση της εντολής `MVI A, 80H` είναι :

```
0800 3E MVI A, 80H
0801 80
```

Πρόγραμμα 1-14 : Μετάφραση περίπτωσης λάθους με την οδηγία `ORG`

Στη συνέχεια η οδηγία `ORG` ζητά από το μεταφραστή να μεταφράσει την εντολή `STA 3000H` στη θέση μνήμης 0800H. Αν αυτό ήταν επιτρεπτό θα σήμαινε ότι τα περιεχόμενα της μνήμης στις θέσεις μνήμης 0800 και 0801 όπου είναι μεταφρασμένη η εντολή `MVI A, 80H` θα χάνονταν.

Το μήνυμα λάθους στην περίπτωση αυτή φαίνεται στην επόμενη οθόνη.



Οθόνη 1-33 : Μήνυμα λάθους για την οδηγία `ORG`

### 1.5.3.7. Σχόλια

Ο χρήστης μπορεί να γράψει σχόλια για το πρόγραμμά του τα οποία δεν θα λαμβάνονται υπόψη κατά τη μετάφραση.

Ένα σχόλιο αρχίζει με ένα χαρακτήρα `;` και ισχύει μέχρι την αρχή της επόμενης γραμμής. Ένα παράδειγμα προγράμματος με σχόλια φαίνεται στη συνέχεια.

```
;Το πρόγραμμα αυτό ανάβει τα τρία δεξιότερα ;LED
και μετά τερματίζει

;Τα LED's είναι αρνητικής λογικής
;οπότε στον A τοποθετείται η τιμή
MVI A, F8H ;11111000

STA 3000H ;Αναμνα των LED's

END ;Τέλος προγράμματος
```

Πρόγραμμα 1-15: Χρήση σχολίων (Comments)

**1.5.3.8. Δεκαεξαδικές τιμές**

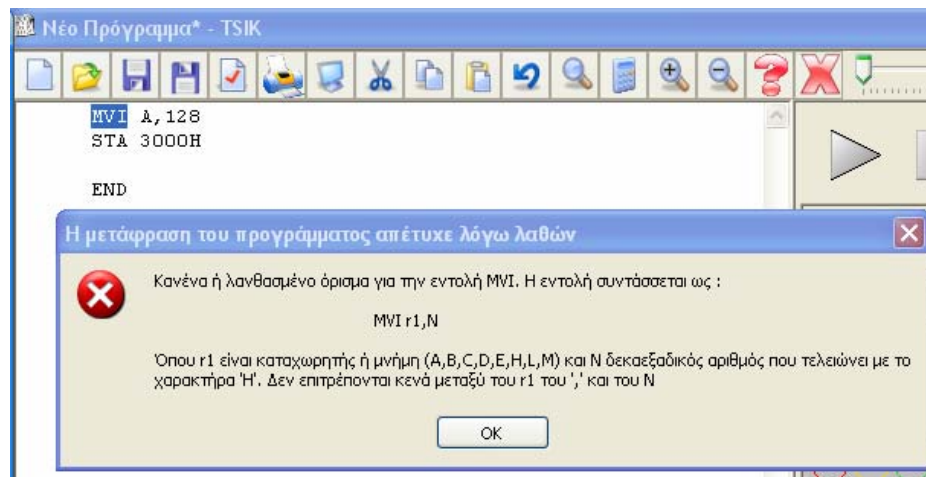
Ο μεταφραστής αναγνωρίζει μόνο δεκαεξαδικές τιμές, οι οποίες μάλιστα για να είναι έγκυρες πρέπει να τελειώνουν με το χαρακτήρα 'H'. Για να γίνει αυτό σαφές παρατίθεται το ακόλουθο παράδειγμα.

```
MVI A,128
STA 3000H

END
```

Πρόγραμμα 1-16: Ο μεταφραστής ΔΕΝ αντιλαμβάνεται δεκαδικές τιμές (1)

Γράφουμε το πρόγραμμα αυτό στο χώρο κειμένου και πατούμε το πλήκτρο F5 προκειμένου να γίνει η μετάφραση. Το αποτέλεσμα φαίνεται στην οθόνη που έπεται :



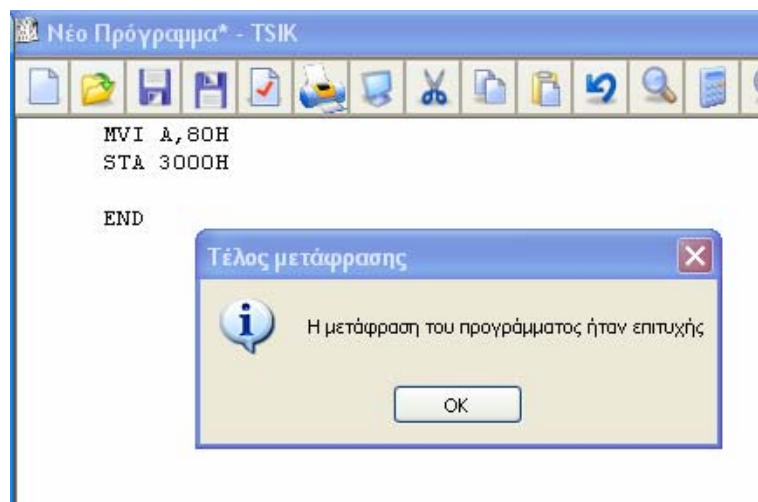
Οθόνη 1-34 : Ο μεταφραστής ΔΕΝ αντιλαμβάνεται δεκαδικές τιμές (1)

Για να μεταφραστεί σωστά το πρόγραμμα πρέπει να μετατρέψουμε την δεκαδική τιμή 128 στην αντίστοιχη δεκαεξαδική, που είναι το 80H. Έτσι τώρα το πρόγραμμα θα έχει τη μορφή :

```
MVI A,80H
STA 3000H

END
```

Πρόγραμμα 1-17: Ο μεταφραστής ΔΕΝ αντιλαμβάνεται δεκαδικές τιμές (2)



Οθόνη 1-35 : Ο μεταφραστής ΔΕΝ αντιλαμβάνεται δεκαδικές τιμές (2)



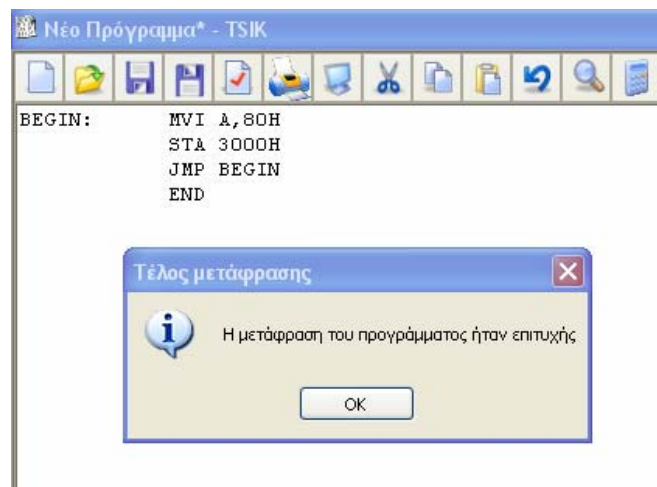
### 1.5.3.9. Εντολές άλματος

Για τη μετάφραση των εντολών άλματος πρέπει να γίνει χρήση ετικετών (labels). Για να είναι έγκυρη μια εντολή άλματος πρέπει να ακολουθείται από μια ετικέτα η οποία να είναι δηλωμένη σε κάποιο σημείο του προγράμματος ακολουθούμενη από ένα χαρακτήρα ' : '.

Ένα παράδειγμα έγκυρου προγράμματος που χρησιμοποιεί την εντολή άλματος **JMP** φαίνεται στη συνέχεια.

```
BEGIN: MVI A, 80H
        STA 3000H
        JMP BEGIN
        END
```

Πρόγραμμα 1-18: Έγκυρο πρόγραμμα με εντολή άλματος



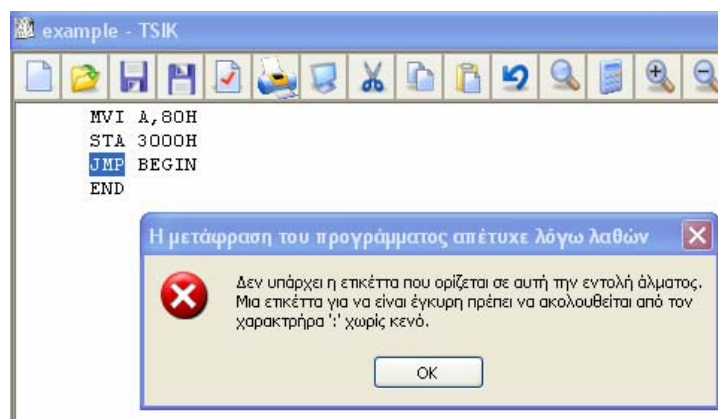
Οθόνη 1-36 : Οθόνη ορθού προγράμματος με εντολή άλματος

Στην περίπτωση που μια εντολή άλματος αναφέρεται σε μια ετικέτα που δεν υπάρχει πουθενά η μετάφραση θα αποτύχει.

```
MVI A, 80H
STA 3000H
JMP BEGIN
END
```

Πρόγραμμα 1-19: Η ετικέτα της εντολής άλματος δεν υπάρχει

Το πρόγραμμα παράγει ένα μήνυμα λάθους της μορφής :



Οθόνη 1-37 : Η ετικέτα της εντολής άλματος δεν υπάρχει

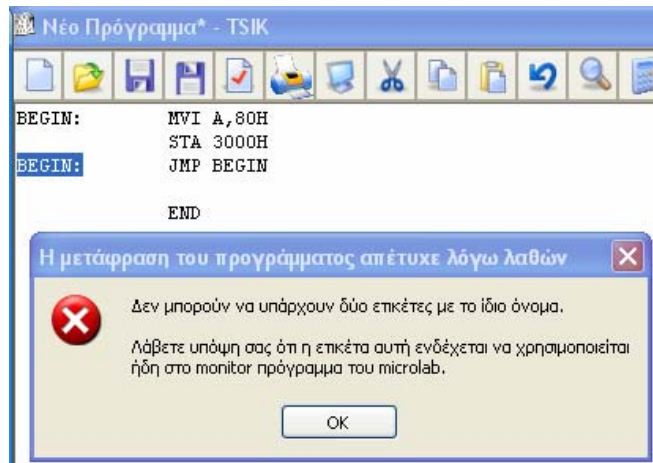
Μια ετικέτα δεν μπορεί να είναι δηλωμένη περισσότερες από μία φορά σε ένα πρόγραμμα.

```
BEGIN: MVI A, 80H
        STA 3000H
BEGIN:   JMP BEGIN

        END
```

*Πρόγραμμα 1-20: Δεν μπορούν να υπάρχουν δύο ετικέτες με το ίδιο όνομα*

Στο προηγούμενο πρόγραμμα η ετικέτα BEGIN: είναι δηλωμένη δύο φορές κάτι που δεν είναι επιτρεπτό. Η μετάφραση του προγράμματος αυτού παράγει ένα μήνυμα λάθους της μορφής :



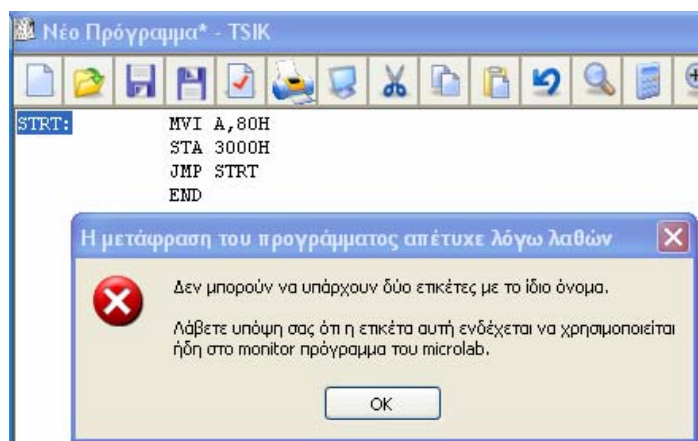
*Οθόνη 1-38 : Δεν μπορούν να υπάρχουν δύο ετικέτες με το ίδιο όνομα*

Όπως αναφέρει και το μήνυμα λάθους της προηγούμενης οθόνης είναι δυνατόν μια ετικέτα που χρησιμοποιεί ο χρήστης στο πρόγραμμά του να χρησιμοποιείται ήδη στο πρόγραμμα επόπτη (monitor program). Για παράδειγμα ας δούμε το ακόλουθο πρόγραμμα.

```
STRT:   MVI A, 80H
        STA 3000H
        JMP STRT
        END
```

*Πρόγραμμα 1-21 : Μια ετικέτα ενδέχεται να χρησιμοποιείται ήδη στο πρόγραμμα επόπτη (monitor program)*

Παρόλο που η ετικέτα με όνομα "STRT" είναι μία μόνο φορά δηλωμένη η μετάφραση του προγράμματος δεν είναι επιτυχής. Συγκεκριμένα η οθόνη που παρουσιάζεται είναι :



*Οθόνη 1-39 : Μια ετικέτα ενδέχεται να χρησιμοποιείται ήδη στο πρόγραμμα επόπτη (monitor program)*

Αυτό οφείλεται στο ότι η συγκεκριμένη ετικέτα χρησιμοποιείται από το πρόγραμμα επόπτη (monitor program). Οι ετικέτες που χρησιμοποιούνται στο πρόγραμμα επόπτη (monitor program) φαίνονται στον πίνακα που ακολουθεί.

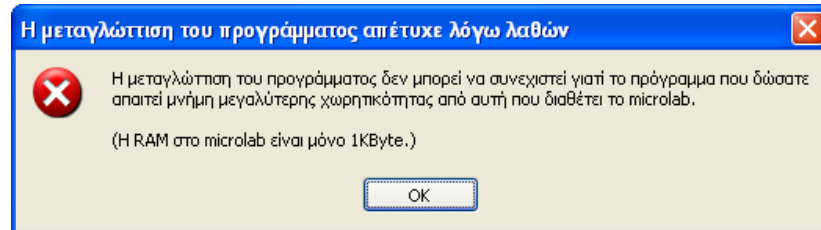
Ταξινομημένες κατά όνομα						Ταξινομημένες κατά διεύθυνση					
Label	Adr.	Label	Adr.	Label	Adr.	Adr.	Label	Adr.	Label	Adr.	Label
ALL	0281	IM	027D	RUN1	03C5	0000	RESET	0279	PCL	04F8	CONV
ANDGT	04E0	INSS	03F1	RUN2	03E0	0008	RS1	027D	IM	04FF	LOOP
BEEP	0010	KIND1	014D	SATL1	048D	0010	BEEP	0281	ALL	0514	MOVE
BEEP1	0012	KIND2	0156	SATL2	04A1	0012	BEEP1	0287	ICX	0518	MLP
BEEP2	0447	KIND3	0164	SATL3	04A7	0018	STDM	028D	PPM	0531	STONE
BEEP3	044E	KIND4	0177	SATL4	04AF	0020	RS4	0291	BLNKM	053E	WTM
BEEP4	0460	KIT	01B0	SDM	0235	0024	TRAP	0297	BLNK	0541	WTM1
BEEP5	0464	KPU	0185	SDM1	0237	0028	RS5	029F	CHDSS	055A	SQRL
BEEP6	046F	KPU1	0190	SDS	01C8	002C	RS55	02A4	CINSS	0560	SQRL1
BEEP7	047E	KRD	019A	SDS1	01D0	0030	RS6	02A9	CRUN	057B	SQRL2
BEEP8	0480	KRD1	01A3	SERV	06C8	0034	RS65	02AE	CSTRM	0586	SQRL3
BEEP9	0488	LLOSE	06E4	SERV1	06EB	0038	RS7	02B8	CFETA	0599	ORGAN
BLNK	0297	LOOP	04FF	SERV2	0700	003C	RS75	02BD	CFETP	059B	ORGAN1
BLNKM	0291	LSM	07D4	SHIFT	05CC	0040	STRT	02C2	CFETR	05AE	DLY
CDCRR	02CD	MA	024D	SNAKE	06C2	004C	STRT1	02C8	CSTRR	05B5	SPKR
CFETA	02B8	MB	0255	SPH	026D	0064	STRT2	02CD	CDCRR	05BB	CODE
CFETP	02BD	MC	0259	SPKR	05B5	0073	STRT3	02D5	FETR	05C0	READ
CFETR	02C2	MD	025D	SPL	0271	008A	STRT4	02DC	FETR1	05CC	SHIFT
CHDSS	029F	ME	0261	SQRL	055A	009C	STRT5	02DE	FETR2	05D8	NOKEY
CHXSN	07FF	MERR	00DC	SQRL1	0560	00A6	STRT6	02F1	FETR3	05E2	TABLE
CINSS	02A4	MERR1	00E5	SQRL2	057B	00B7	STRT7	02FE	FETR4	05F9	ROCT
CODE	05BB	MERR2	00ED	SQRL3	0586	00C8	PPER	0300	FETR5	0601	ROCT1
CONV	04F8	MH	0265	STDM	0018	00DC	MERR	0310	FETR6	0603	ROCT2
CRUN	02A9	ML	0269	STRM	03AF	00E5	MERR1	0319	DPS	0616	ROCT3
CSTRM	02AE	MLP	0518	STRR	0413	00ED	MERR2	0328	FETA	0639	ROCT4
CSTRR	02C8	MOVE	0514	STRT	0040	00F3	TRP	0329	FETAR	0655	ROCT5
DCC	0218	NCTL	034A	STRT1	004C	00FA	TRP1	032F	FETA1	065C	ROCM
DCD	01E9	NOKEY	05D8	STRT2	0064	00FF	TRP2	034A	NCTL	0662	STW
DCD1	01F3	ON	04F0	STRT3	0073	0111	TRP3	034D	FETA2	0669	STW1
DCD2	0210	ORGAN	0599	STRT4	008A	0115	TRP4	0364	FETA3	0672	STWL
DCRM	03AA	ORGAN1	059B	STRT5	009C	011B	TRP5	0381	FETA4	0693	STW2
DCRR	03FD	PADL	07A2	STRT6	00A6	0124	TRP6	0383	FETA5	069B	STW3
DEL1	0431	PASS	0738	STRT7	00B7	0141	TRP7	0393	FETA6	06A4	STW4
DEL2	0434	PASS1	073F	STW	0662	014D	KIND1	039C	FETA7	06A8	STW5
DEL3	0436	PCH	0275	STW1	0669	0156	KIND2	03AA	DCRM	06B7	STW6
DELA	0429	PCL	0279	STW2	0693	0164	KIND3	03AF	STRM	06BC	STWM
DELB	0430	PLAY	075A	STW3	069B	0177	KIND4	03C2	RUN	06C2	SNAKE
DLY	05AE	PLAY1	0767	STW4	06A4	0185	KPU	03C5	RUN1	06C8	SERV
DMT	0241	PPER	00C8	STW5	06A8	0190	KPU1	03E0	RUN2	06E4	LLOSE
DPS	0319	PPM	028D	STW6	06B7	019A	KRD	03F1	INSS	06EB	SERV1
ECHO	04D7	RBND	070C	STWL	0672	01A3	KRD1	03F7	HDSS	06F7	ENDGM
ENDGM	06F7	RBND1	0727	STWM	06BC	01B0	KIT	03FD	DCRR	0700	SERV2
FETA	0328	READ	05C0	TABLE	05E2	01C8	SDS	0413	STRR	070C	RBND
FETA1	032F	RESET	0000	TIME1	07B0	01D0	SDS1	0429	DELA	0719	RSRV
FETA2	034D	ROCM	065C	TIMER	07AD	01E9	DCD	0430	DELB	0727	RBND1
FETA3	0364	ROCT	05F9	STONE	0531	01F3	DCD1	0431	DEL1	0738	PASS
FETA4	0381	ROCT1	0601	TRAP	0024	0210	DCD2	0434	DEL2	073F	PASS1
FETA5	0383	ROCT2	0603	TRIL	07B9	0218	DCC	0436	DEL3	075A	PLAY
FETA6	0393	ROCT3	0616	TRIL1	07BF	0235	SDM	0447	BEEP2	0767	PLAY1
FETA7	039C	ROCT4	0639	TRIL2	07CD	0237	SDM1	044E	BEEP3	0782	FSTR
FETAR	0329	ROCT5	0655	TRP	00F3	0241	DMT	0460	BEEP4	0799	RSTGM
FETCH	0247	RS1	0008	TRP1	00FA	0247	FETCH	0464	BEEP5	07A2	PADL
FETR	02D5	RS4	0020	TRP2	00FF	024D	MA	046F	BEEP6	07AD	TIMER
FETR1	02DC	RS5	0028	TRP3	0111	0251	FLG	047E	BEEP7	07B0	TIME1
FETR2	02DE	RS55	002C	TRP4	0115	0255	MB	0480	BEEP8	07B9	TRIL
FETR3	02F1	RS6	0030	TRP5	011B	0259	MC	0488	BEEP9	07BF	TRIL1
FETR4	02FE	RS65	0034	TRP6	0124	025D	MD	048D	SATL1	07CD	TRIL2
FETR5	0300	RS7	0038	TRP7	0141	0261	ME	04A1	SATL2	07D4	LSM
FETR6	0310	RS75	003C	WTM	053E	0265	MH	04A7	SATL3	07E2	RSM
FLG	0251	RSM	07E2	WTM1	0541	0269	ML	04AF	SATL4	07F0	XSTART
FSTR	0782	RSRV	0719	XLOOP	07F2	026D	SPH	04D7	ECHO	07F2	XLOOP
HDSS	03F7	RSTGM	0799	XSTART	07F0	0271	SPL	04E0	ANDGT	07FF	CHXSN
ICX	0287	RUN	03C2			0275	PCH	04F0	ON		

Πίνακας 1-3: Ετικέτες που χρησιμοποιούνται από το πρόγραμμα επόπτη (monitor program)

### 1.5.3.10. Το πρόγραμμα δεν μπορεί να έχει αυθαίρετο μέγεθος

Όπως αναφέρθηκε και σε προηγούμενη παράγραφο το microlab διαθέτει μόνο 1 Kbytes πραγματικής RAM. Αυτό σημαίνει ότι το πρόγραμμα του χρήστη δεν πρέπει να καταλαμβάνει χώρο μεγαλύτερο από αυτόν.

Στην περίπτωση που το πρόγραμμα είναι τόσο μεγάλο που δεν υπάρχει χώρος για να μεταφραστεί το μήνυμα λάθους που θα παραχθεί θα έχει τη μορφή :



Οθόνη 1-40 : Μήνυμα λάθους όταν το πρόγραμμα του χρήστη δεν χωρά στη RAM

Αν δοκιμάσουμε να μεταφράσουμε ένα πρόγραμμα που αποτελείται μόνο από εντολές STA 3000H θα διαπιστώσουμε ότι μπορούν να χωρέσουν το πολύ 341 εντολές. Αυτό γιατί  $3 \cdot 341 = 1023$  ενώ  $3 \cdot 342 = 1026 > 1024$  που είναι η χωρητικότητα της μνήμης RAM του microlab.

```

STA 3000H
STA 3000H
STA 3000H
STA 3000H
STA 3000H
STA 3000H
.....
STA 3000H

END

```

Πρόγραμμα 1-22 : Το πρόγραμμα του χρήστη δεν μπορεί να είναι αυθαίρετα μεγάλο (Το πολύ 341 εντολές STA)

## 1.6 Ο Εξομοιωτής

Στις επόμενες παραγράφους θα εξηγηθούν οι διάφορες λειτουργίες που σχετίζονται με την εξομοίωση του microlab.

Σκοπός αυτών δεν είναι να διδάξουν στο χρήστη τον τρόπο λειτουργίας του microlab αλλά μόνο να τον βοηθήσουν να αντιληφθεί τον τρόπο χειρισμού του εξομοιωτή. Ο χρήστης του εξομοιωτή υποτίθεται ότι είναι εξοικειωμένος με το microlab.

### 1.6.1. Πλήκτρα χειρισμού του microlab

#### 1.6.1.1. Έναρξη της λειτουργίας του microlab

Συντόμευση : F6

Εικονίδιο : 

Όταν πατηθεί το προηγούμενο πλήκτρο ή η συντόμευση F6 μεταφράζεται το πρόγραμμα που είναι γραμμένο στο χώρο κειμένου και ξεκινά το microlab. Στη μνήμη τυχαίας προσπέλασης (RAM) του συστήματος τοποθετείται αυτόματα το μεταφρασμένο πρόγραμμα. Αν το τμήμα κειμένου είναι κενό τότε το microlab ξεκινά με τη μνήμη τυχαίας να είναι κενή. Και στις δύο περιπτώσεις εκτελείται το πρόγραμμα επόπτη (monitor program).

Στην περίπτωση που η μετάφραση του προγράμματος αποτύχει το microlab δεν θα ξεκινήσει και ο χρήστης θα ενημερωθεί με κατάλληλο μήνυμα.

#### **1.6.1.2. Τερματισμός λειτουργίας του microlab**

Συντόμευση : F6

Εικονίδιο : 

Όταν πατηθεί το προηγούμενο πλήκτρο ή η συντόμευση F6 τερματίζεται η λειτουργία του microlab και ανανεώνεται η εικόνα των καταχωρητών και της μνήμης (η λειτουργία της ανανέωσης εξετάζεται σε επόμενη παράγραφο). Τα δεδομένα των καταχωρητών και της μνήμης παραμένουν αλλά μόνο για επόπτευση. Όταν το microlab ξαναξεκινήσει οι τιμές αυτές δεν θα ισχύουν πλέον.

#### **1.6.1.3. Παύση λειτουργίας του microlab**

Συντόμευση : F7

Εικονίδιο : 

Όταν πατηθεί το προηγούμενο πλήκτρο ή η συντόμευση F7 διακόπτεται προσωρινά η λειτουργία του microlab. Η διαφορά της λειτουργίας παύσης από τη λειτουργία τερματισμού έγκειται στο ότι η λειτουργία τερματισμού δεν επιτρέπει τη συνέχιση της λειτουργίας του microlab από το σημείο στο οποίο αυτή διακόπηκε. Αντίθετα στη λειτουργία παύσης / συνέχισης όταν ξαναπατηθεί η συντόμευση F7 το microlab συνεχίζει τη λειτουργία του από το σημείο από το οποίο αυτή διακόπηκε προσωρινά.

Μια πρόσθετη λειτουργία του πλήκτρου αυτού είναι η ανανέωση της τιμής των καταχωρητών και της μνήμης. Αυτό σημαίνει δηλαδή ότι κάθε φορά που το πρόγραμμα διακόπτεται προσωρινά οι καταχωρητές και η μνήμη θα παίρνουν τις τιμές των αντίστοιχων συσκευών την στιγμή εκείνη.

#### **1.6.1.4. Συνέχιση λειτουργίας του microlab**

Συντόμευση : F7

Εικονίδιο : 

Είναι η δυαδική λειτουργία της παύσης, δηλαδή συνεχίζει την εκτέλεση του microlab από το σημείο που βρισκόταν όταν πατήθηκε το πλήκτρο “Παύση”.

#### **1.6.1.5. Καθαρισμός καταχωρητών και μνήμης**

Συντόμευση : F8

Εικονίδιο : 

Καθαρίζει (μηδενίζει) όλους τους καταχωρητές και τη μνήμη του συστήματος. Η λειτουργία είναι χρήσιμη μεταξύ διαδοχικών εκκινήσεων και τερματισμών του microlab αφού έτσι είναι βέβαιο ότι το σύστημα είναι “εντελώς καθαρό”.

#### **1.6.1.6. Ρύθμιση ταχύτητας εκτέλεσης**

Συντόμευση : -

Εικονίδιο : 

Οι ταχύτητες των υπολογιστών ποικίλλουν και έτσι η πραγματική ταχύτητα εξομοίωσης δεν θα είναι ίδια σε κάθε υπολογιστή. Αυτό έχει ως αποτέλεσμα οι ρουτίνες καθυστέρησης του προγράμματος επόπτη να μην αντιστοιχούν σε πραγματικούς χρόνους.

Σύροντας τον δείκτη αυτό προς τα δεξιά προστίθεται με τεχνητή καθυστέρηση στον εξομοιωτή. Έτσι είναι δυνατόν το εξομοιωμένο microlab να τρέχει με την ίδια ταχύτητα σε υπολογιστές διαφορετικών δυνατοτήτων.

### 1.6.2. Συσκευές εισόδου / εξόδου

Το τμήμα της οθόνης του προγράμματος που αναπαριστά τις συσκευές εισόδου εξόδου του microlab φαίνεται στο σχήμα που ακολουθεί.



Σχήμα 1-1 : Συσκευές εισόδου / εξόδου του microlab μέσα από τον εξομοιωτή

Ο χειρισμός των συσκευών αυτών δεν διαφέρει σε τίποτε από το πραγματικό microlab. Επειδή όπως είπαμε και πριν θεωρούμε ότι ο χρήστης του εξομοιωτή είναι εξοικειωμένος με το microlab δεν κρίνεται σκόπιμη οποιαδήποτε άλλη παρατήρηση πέραν της αναφοράς των συντομεύσεων πληκτρολογίου.

Πλήκτρο	Συντόμευση	Εναλλακτική Συντόμευση	Πλήκτρο	Συντόμευση	Εναλλακτική Συντόμευση
RESET	S	Escape	0	0	Num 0
RUN	N	Control + Num Enter	1	1	Num 1
HARDWARE STEP	W		2	2	Num 2
INSTRUCTION STEP	T	Num /	3	3	Num 3
INTERRUPT	I	Num . (dot)	4	4	Num 4
FETCH PC	P	Control + Num -	5	5	Num 5
FETCH ADDRESS	H	Control + Num +	6	6	Num 6
FETCH REGISTER	G	Num *	7	7	Num 7
DECREASE	R	Num -	8	8	Num 8
STORE / INCREMENT	O	Num +	9	9	Num 9
Διακόπτης 7 (Αριστερά)	Alt + Num 7		A	A	Control + Num 1
Διακόπτης 6	Alt + Num 6		B	B	Control + Num 2
Διακόπτης 5	Alt + Num 5		C	C	Control + Num 3
Διακόπτης 4	Alt + Num 4		D	D	Control + Num 4
Διακόπτης 3	Alt + Num 3		E	E	Control + Num 5
Διακόπτης 2	Alt + Num 2		F	F	Control + Num 6
Διακόπτης 1	Alt + Num 1				
Διακόπτης 0 (Δεξιά)	Alt + Num 0				

Πίνακας 1-4: Πίνακας συντομεύσεων πληκτρολογίου για τις συσκευές εισόδου του εξομοιωμένου microlab

Όπως προκύπτει από τον προηγούμενο πίνακα για τα πλήκτρα ελέγχου το microlab (Reset, Run κλπ) οι συντομεύσεις πληκτρολογίου είναι τα γράμματα που φαίνονται υπογραμμισμένα σε κάθε πλήκτρο.

### 1.6.3. Εργαλεία επόπτευσης των καταχωρητών και της μνήμης

Όπως αναφέρθηκε σε προηγούμενες παραγράφους ο εξομοιωτής Microlab Simulator – TSIK προσφέρει εργαλεία επόπτευσης της κατάστασης του συστήματος προς διευκόλυνση του χρήστη.

#### 1.6.3.1. Επόπτευση της μνήμης

Για να δει ο χρήστης του πραγματικού microlab την τιμή μιας θέσης μνήμης (έστω της 0815) πρέπει να πατήσει διαδοχικά τα πλήκτρα :

```

FETCH ADDRESS
0
8
1
5

```

Προκειμένου να δει τις τιμές πολλών θέσεων μνήμης απαιτείται αρκετός χρόνος. Ο εξομοιωτής Microlab Simulator – TSIK παρέχει ένα σημαντικά ευκολότερο τρόπο επόπτευσης της μνήμης, τον χώρο απεικόνισης μνήμης τμήμα του οποίου φαίνεται στην επόμενη οθόνη :

0800	00	0801	01	0802	34	0803	12	0804	02	0805	03	0806	04	0807	05
0808	06	0809	AB	080A	07	080B	08	080C	09	080D	0A	080E	0B	080F	0C
0810	0D	0811	0E	0812	CD	0813	0F	0814	10	0815	11	0816	CD	0817	AB
0818	12	0819	13	081A	14	081B	15	081C	16	081D	87	081E	17	081F	18
0820	19	0821	1A	0822	1B	0823	1C	0824	1D	0825	1E	0826	64	0827	1F
0828	20	0829	21	082A	AF	082B	12	082C	22	082D	E2	082E	F1	082F	23
0830	24	0831	25	0832	26	0833	DC	0834	27	0835	28	0836	D2	0837	29
0838	2A	0839	CD	083A	AB	083B	2B	083C	2C	083D	2D	083E	2E	083F	99
0840	2F	0841	30	0842	31	0843	78	0844	AB	0845	32	0846	CD	0847	AB
0848	33	0849	34	084A	35	084B	36	084C	A7	084D	37	084E	38	084F	15
0850	39	0851	3A	0852	F8	0853	B4	0854	3B	0855	3C	0856	3D	0857	3E
0858	DF	0859	3F	085A	40	085B	41	085C	42	085D	43	085E	44	085F	45
0860	46	0861	47	0862	48	0863	49	0864	4A	0865	4B	0866	4C	0867	4D
0868	4E	0869	4F	086A	50	086B	51	086C	52	086D	53	086E	54	086F	55
0870	56	0871	57	0872	58	0873	59	0874	5A	0875	5B	0876	5C	0877	5D

Οθόνη 1-41 : Χώρος απεικόνισης μνήμης

Στον χώρο αυτό απεικονίζονται όλες οι θέσεις της μνήμης τυχαίας προσπέλασης (RAM) του microlab. Προκειμένου να ανανεωθεί ο χώρος αυτός αρκεί ο χρήστης να πατήσει το πλήκτρο “Παύση” ή τη συντόμευση F7. Στη συνέχεια μπορεί να ξαναπατήσει το πλήκτρο αυτό να συνεχίσει την εκτέλεση του microlab από το σημείο στο οποίο αυτή διακόπηκε όταν πατήθηκε το πλήκτρο “Παύση”.

#### 1.6.3.2. Επόπτευση των καταχωρητών

Όμοια υπάρχει και ο χώρος απεικόνισης των καταχωρητών. Η ανανέωση του χώρου αυτού γίνεται με τον ίδιο τρόπο που γίνεται η ανανέωση του χώρου απεικόνισης της μνήμης.

A	I	B	C	D	E	H	L
00	0B	00	00	00	00	00	00
S	Z	X5	AC	P	V	CY	PC
0	0	0	0	0	0	0800	0B80

Οθόνη 1-42 : Χώρος απεικόνισης καταχωρητών



Προσοχή χρειάζεται όμως ως προς το εξής : Οι καταχωρητές αυτοί δεν δείχνουν τις τιμές των πραγματικών καταχωρητών του επεξεργαστή Intel 8085® αλλά τις εικονικές τους τιμές που αφορούν στο εκτελούμενο πρόγραμμα χρήστη μέσω του προγράμματος επόπτη (monitor program).

## **1.7 Άλλα βοηθητικά εργαλεία**

### **1.7.1. Βηματική εκτέλεση του προγράμματος χρήστη**

Όπως είναι γνωστό ο χρήστης μπορεί να εκτελέσει το πρόγραμμά του εντολή προς εντολή χρησιμοποιώντας το πλήκτρο “Instruction Step”. Ο εξομοιωτής Microlab Simulator – TSIK παρέχει ορισμένες πρόσθετες λειτουργίες κατά τη βηματική εκτέλεση.

Προκειμένου να δειχθούν αυτές οι λειτουργίες γράφουμε στο χώρο κειμένου το πρόγραμμα που ακολουθεί :

```

MVI A, 00H

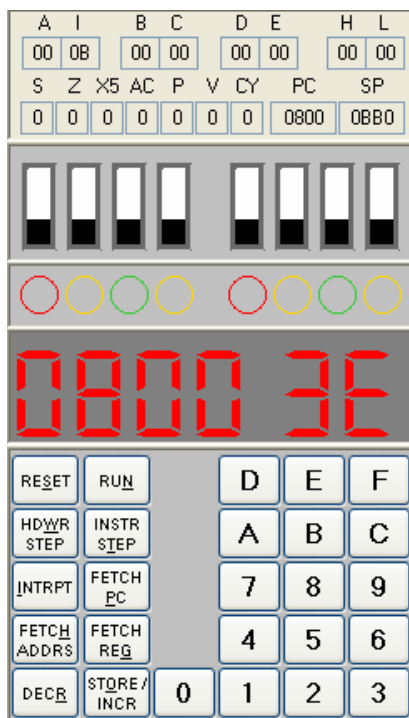
START: DCR A
      STA 3000H
      JMP START

      END

```

Πρόγραμμα 1-23 : Πρόσθετες λειτουργίες κατά το instruction step

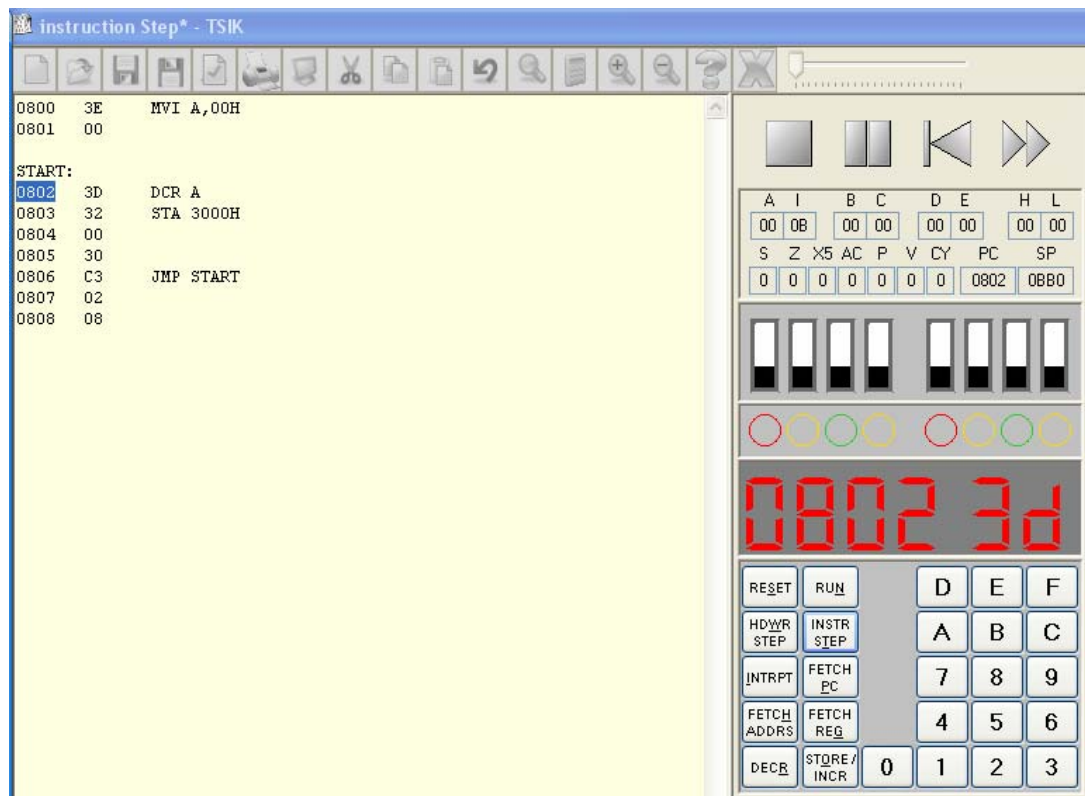
Στη συνέχεια πατούμε το πλήκτρο F6 για να αρχίσει η εκτέλεση του microlab και μετά το πλήκτρο “Fetch PC”. Η κατάσταση στο τμήμα απεικόνισης των θυρών εισόδου εξόδου του microlab πρέπει να είναι η εξής



Οθόνη 1-43 : Πρόσθετες λειτουργίες στο Instruction Step (1)

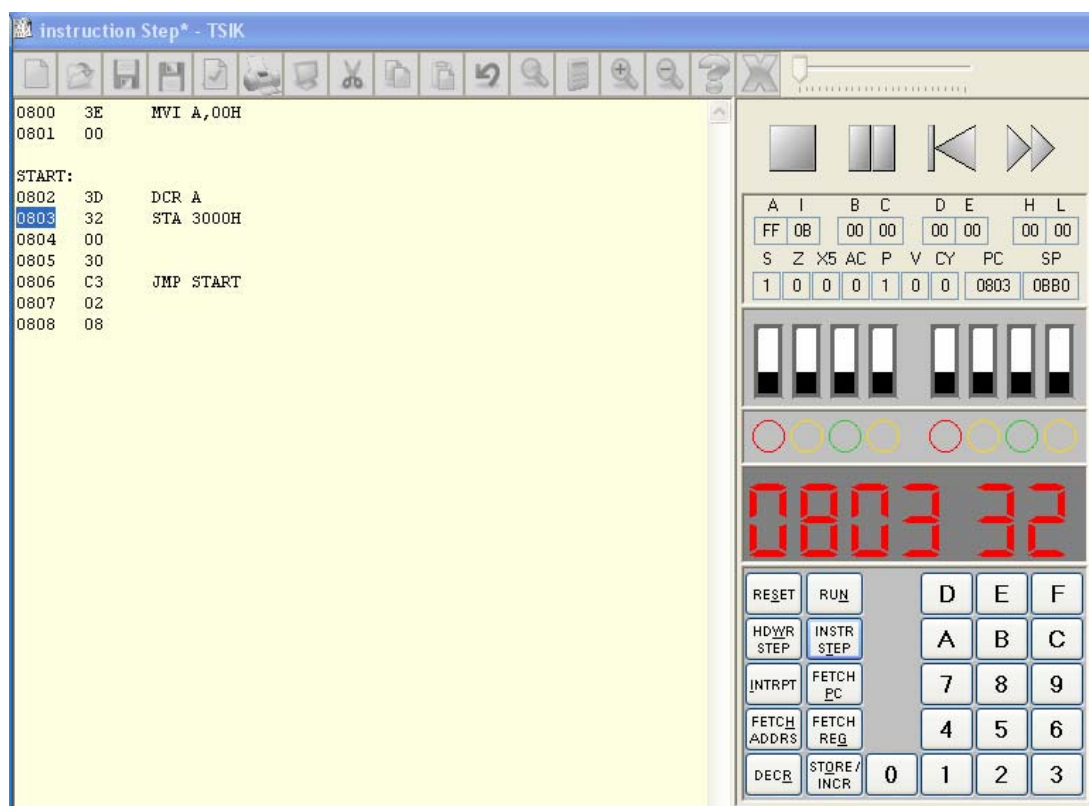
Στη συνέχεια πατούμε το πλήκτρο “Instruction Step”. Βλέπουμε ότι το πρόγραμμα υπογραμμίζει την επόμενη προς εκτέλεση εντολή και ανανεώνει τον χώρο απεικόνισης καταχωρητών. Όπως βλέπουμε στην οθόνη που έπεται η υπογραμμισμένη εντολή είναι η DCR A, που είναι η επόμενη που θα εκτελεστεί. Επίσης φαίνεται και η τιμή του καταχωρητή A είναι ίση με το 0.





Οθόνη 1-44 : Πρόσθετες λειτουργίες στο Instruction Step (2)

Ξαναπατούμε το πλήκτρο “Instruction Step”. Η οθόνη του εξομοιωτή θα είναι τώρα η εξής :



Οθόνη 1-45 : Πρόσθετες λειτουργίες στο Instruction Step (3)

Βλέπουμε ότι η τιμή του καταχωρητή A άλλαξε και ότι τώρα η επόμενη προς εκτέλεση εντολή είναι τώρα η STA 3000H.

### 1.7.2. Διακοπή RST 6.5

Είναι γνωστό ότι το πλήκτρο INTRP του microlab είναι συνδεδεμένο στην διακοπή υλικού (hardware Interrupt) RST 6.5.

Το δύσκολο σημείο με τις διακοπές στο microlab είναι ότι η ρουτίνα εξυπηρέτησης της διακοπής RST 6.5 πρέπει να ξεκινά από τη θέση μνήμης 0AFC. Συχνά οι προγραμματιστές του microlab τοποθετούν εκεί μια εντολή άλματος χωρίς συνθήκη JMP σε κάποια άλλη θέση μνήμης.

Ο εξομοιωτής παρέχει μια λειτουργία που βοηθά σημαντικά τον προγραμματιστή που χρησιμοποιεί διακοπές. Συγκεκριμένα αν ονομάσει μια ετικέτα του προγράμματός του **INTR\_ROUTINE** τότε αυτόματα η ρουτίνα αυτή γίνεται ρουτίνα εξυπηρέτησης της συγκεκριμένης διακοπής.

Ο εξομοιωτής δεν αγνοεί τον τρόπο λειτουργίας του προγράμματος επόπτη (monitor program) του microlab, απλά τοποθετεί στη θέση μνήμης 0AFC μια εντολή άλματος JMP στη θέση μνήμης που βρίσκεται η ετικέτα INTR\_ROUTINE.

```

        MVI A,0DH      ;Ενεργοποίηση της
                        ;διακοπής 6,5
        SIM
        EI             ;Ενεργοποίηση του
                        ;συστήματος διακοπών

        MVI D,FFH
KKL:    MOV A,D
        OUT 30H
        JMP KKL

;Ρουτίνα εξυπηρέτησης διακοπής
INTR_ROUTINE:
        EI
        DCR D
        RET

        END

```

Πρόγραμμα 1-24 : Πρόσθετη λειτουργία για τη διακοπή RST 6.5

Το μεταφρασμένο πρόγραμμα έχει ως εξής :

```

0800  3E      MVI A,0DH
0801  0D
0802  30      SIM
0803  FB      EI
0804  16      MVI D,FFH
0805  FF

KKL:
0806  7A      MOV A,D
0807  D3      OUT 30H
0808  30
0809  C3      JMP KKL
080A  06
080B  08

INTR_ROUTINE:
080C  FB      EI
080D  15      DCR D
080E  C9      RET

```

Πρόγραμμα 1-25 : Μεταφρασμένο πρόγραμμα πρόσθετης λειτουργία για τη διακοπή RST 6.5

Από το προηγούμενο πλαίσιο βλέπουμε ότι η θέση μνήμης της ετικέτας `INTR_ROUTINE` είναι η 080C. Κοιτάζοντας στο χώρο απεικόνισης μνήμης και συγκεκριμένα στη θέση μνήμης 0AFC θα δούμε τις τιμές που φαίνονται στο πλαίσιο που ακολουθεί που αντιστοιχούν στην εντολή `JMP INTR_ROUTINE`.

```
0AFC  C3
0AFD  0C
0AFE  08
```

## 1.8 Σύνοψη συντομεύσεων πληκτρολογίου

### 1.8.1. Εργασίες εξομοιωτή

Εργασία	Συντόμευση	Εικονίδιο	Εργασία	Συντόμευση	Εικονίδιο
Νέο αρχείο	Control + N		Εύρεση και αντικατάσταση	Control + F	
Άνοιγμα αρχείου	Control + O		Μεγέθυνση γραμματοσειράς	Control + ]	
Αποθήκευση αρχείου	Control + S		Σμίκρυνση γραμματοσειράς	Control + [	
Αποθήκευση αρχείου ως	Control + Shift + S		Μετατροπές συστημάτων	F3	
Ρύθμιση σελίδας εκτύπωσης	Control + Shift + U		Περί (About)	-	
Εκτύπωση	Control + P		Μετάφραση	F5	
Επισκόπηση εκτύπωσης	Control + Shift + P		Έναρξη λειτουργίας	F6	
Αποκοπή	Control + X		Τερματισμός λειτουργίας	F6	
Αντιγραφή	Control + C		Παύση λειτουργίας	F7	
Επικόλληση	Control + V		Συνέχιση λειτουργίας	F7	
Αναίρεση	Control + Z		Καθαρισμός καταχωρητών & μνήμης	F8	

### 1.8.2. Εργασίες συστήματος εισόδου / εξόδου του microlab

Εργασία	Συντόμευση 1	Συντόμευση 2	Εργασία	Συντόμευση 1	Συντόμευση 2
RESET	S	Escape	0	0	Num 0
RUN	N	Control + Num Enter	1	1	Num 1
HARDWARE STEP	W		2	2	Num 2
INSTRUCTION STEP	T	Num /	3	3	Num 3
INTERRUPT	I	Num . (dot)	4	4	Num 4
FETCH PC	P	Control + Num -	5	5	Num 5
FETCH ADDRESS	H	Control + Num +	6	6	Num 6
FETCH REGISTER	G	Num *	7	7	Num 7
DECREASE	R	Num -	8	8	Num 8
STORE/ INCREMENT	O	Num +	9	9	Num 9
Διακόπτης 7 (αρ/τερος)	Alt + Num 7	-	A	A	Control + Num 1
Διακόπτης 6	Alt + Num 6	-	B	B	Control + Num 2
Διακόπτης 5	Alt + Num 5	-	C	C	Control + Num 3
Διακόπτης 4	Alt + Num 4	-	D	D	Control + Num 4
Διακόπτης 3	Alt + Num 3	-	E	E	Control + Num 5
Διακόπτης 2	Alt + Num 2	-	F	F	Control + Num 6
Διακόπτης 1	Alt + Num 1	-			
Διακόπτης 0 (Δεξ/τερος)	Alt + Num 0	-			