

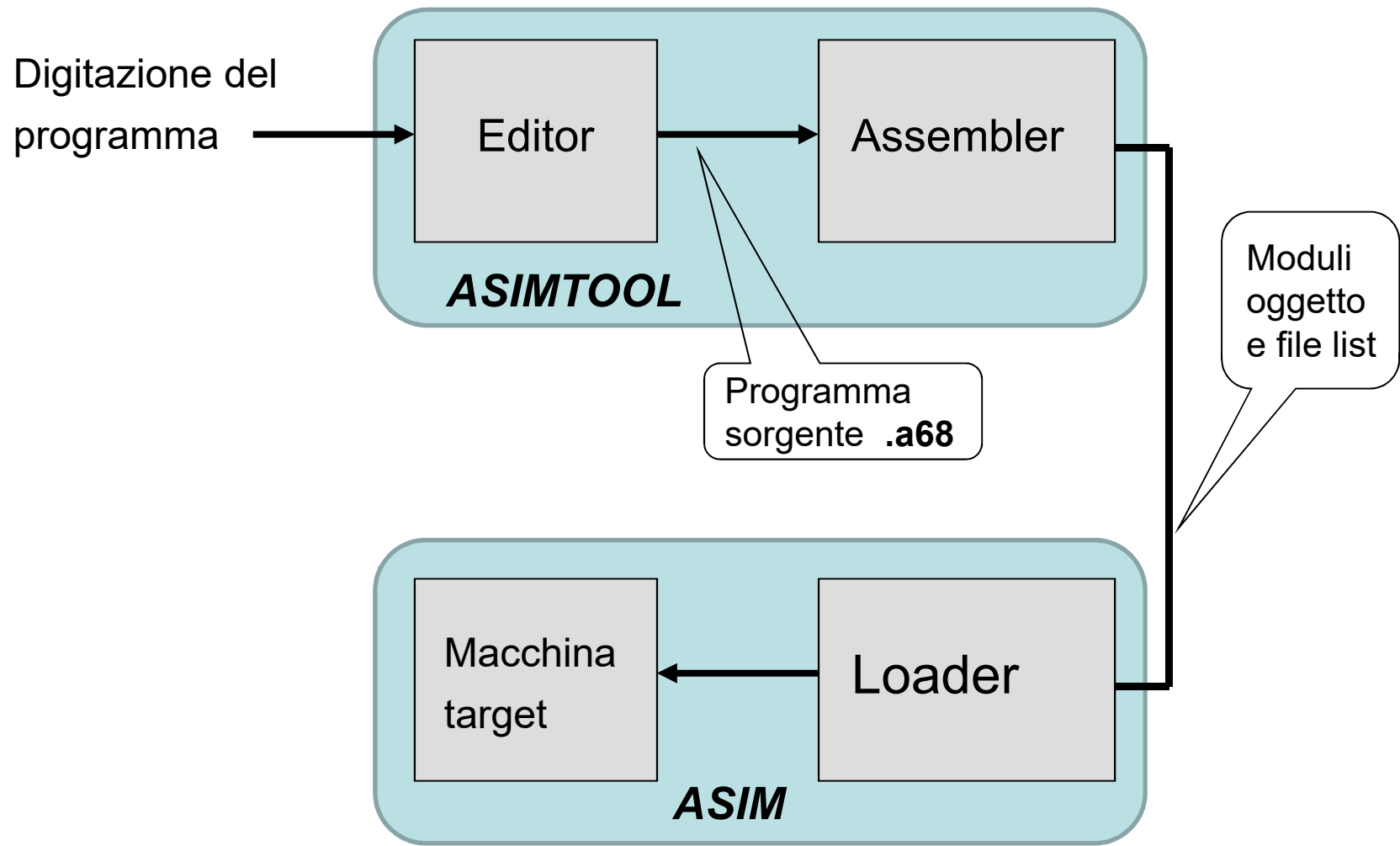
Corso di Calcolatori Elettronici I

**Guida all'installazione
ed all'uso di ASIM**

ing. Alessandro Cilaro

Corso di Laurea in Ingegneria Biomedica

Ciclo di sviluppo semplificato di programmi assembly MC68000 nel sistema didattico ASIM



AsimTool ed ASIM

- **AsimTool**

- è un “**assemblatore**”: riceve un file scritto in linguaggio assembly 68000 e genera un file “oggetto” che rappresenta il programma (istruzioni/dati) così come verrà posto nella memoria del calcolatore

- **Asim**

- è un **simulatore**: riproduce la struttura del calcolatore (fatta almeno da una CPU ed un modulo di memoria), il funzionamento dettagliato del processore, i suoi registri interni, la sequenza di caricamento ed esecuzione delle istruzioni, etc.

AsimTool ed ASIM

- **AsimTool**

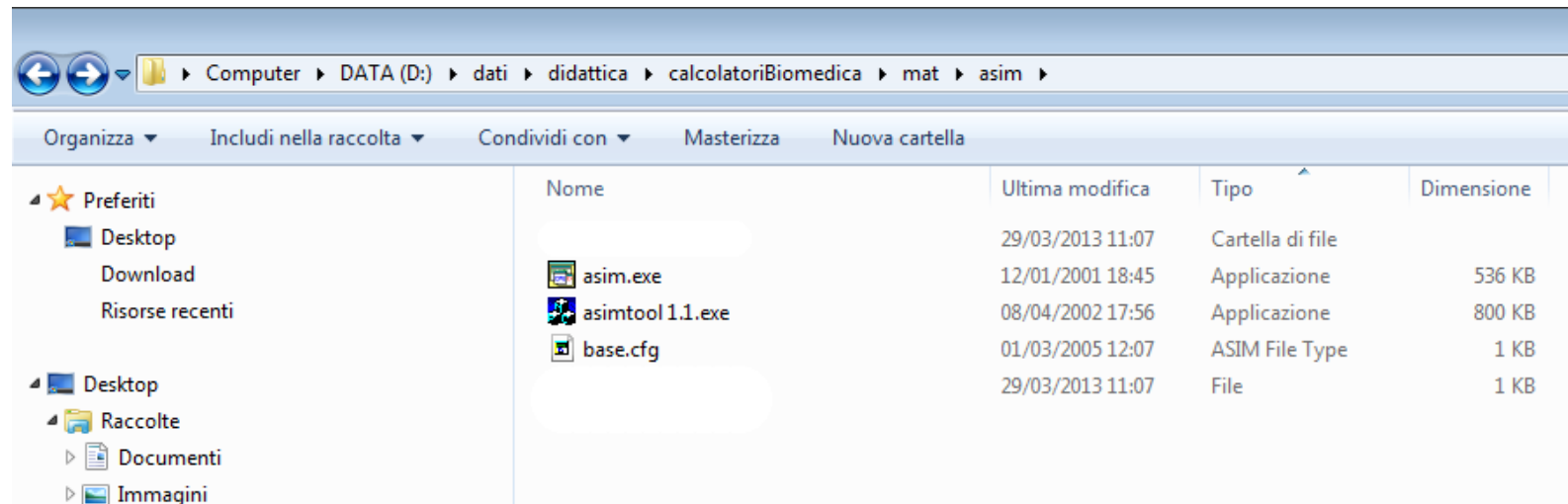
- fornisce un *editor* per scrivere/modificare i programmi assembler (estensione **.a68**)
- permette di assemblare tramite un apposito comando
- produce un file **.H68** (il file oggetto vero e proprio) insieme ad un file **.LIS**, che descrive in forma testuale il contenuto della memoria

- **Asim**

- richiede un file **.cfg** che descrive la struttura del sistema da simulare
- useremo solo il file **base.cfg**, che descrive un semplice sistema con un processore ed una memoria

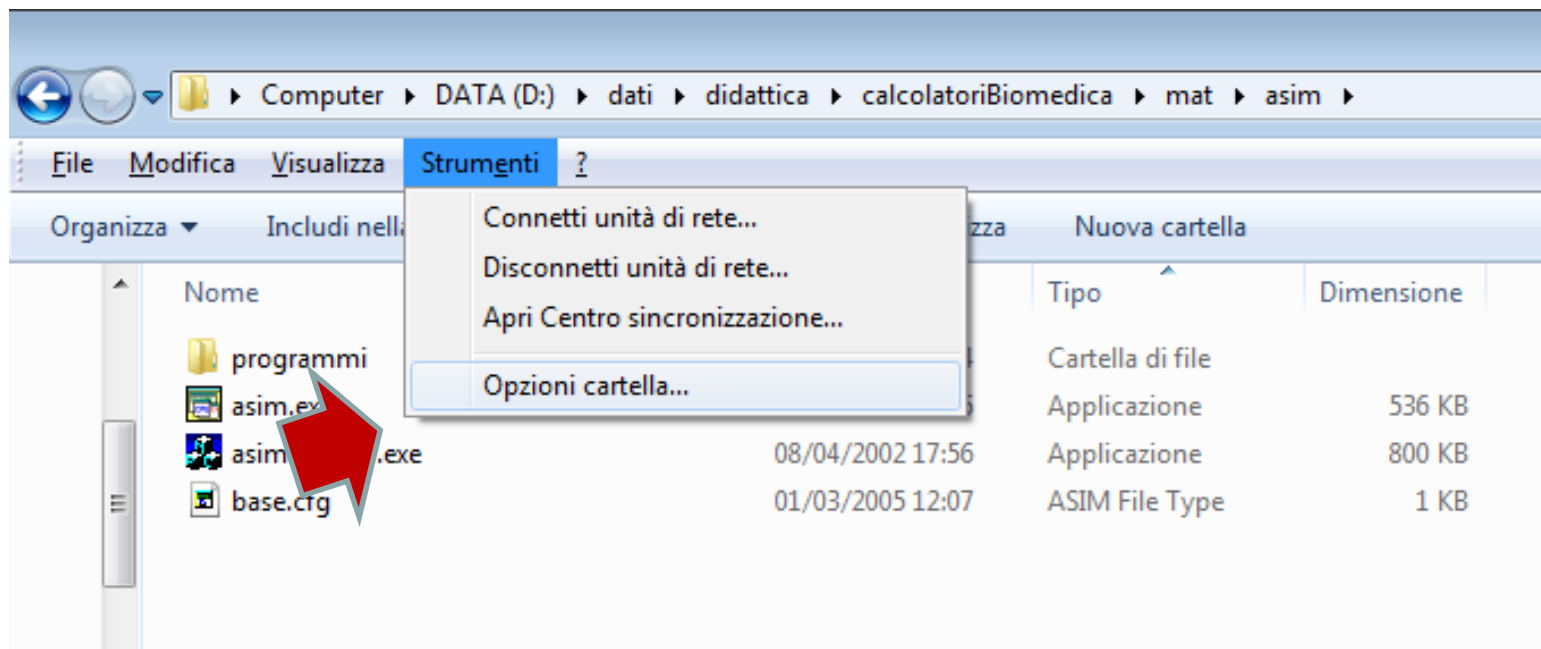
Installazione

- E' sufficiente copiare i tre file forniti (`asim.exe`, `asimtool1.1.exe` e `base.cfg`) in un qualsiasi percorso nel proprio PC
- E' consigliabile evitare che i nomi delle cartelle nel percorso contengano degli spazi



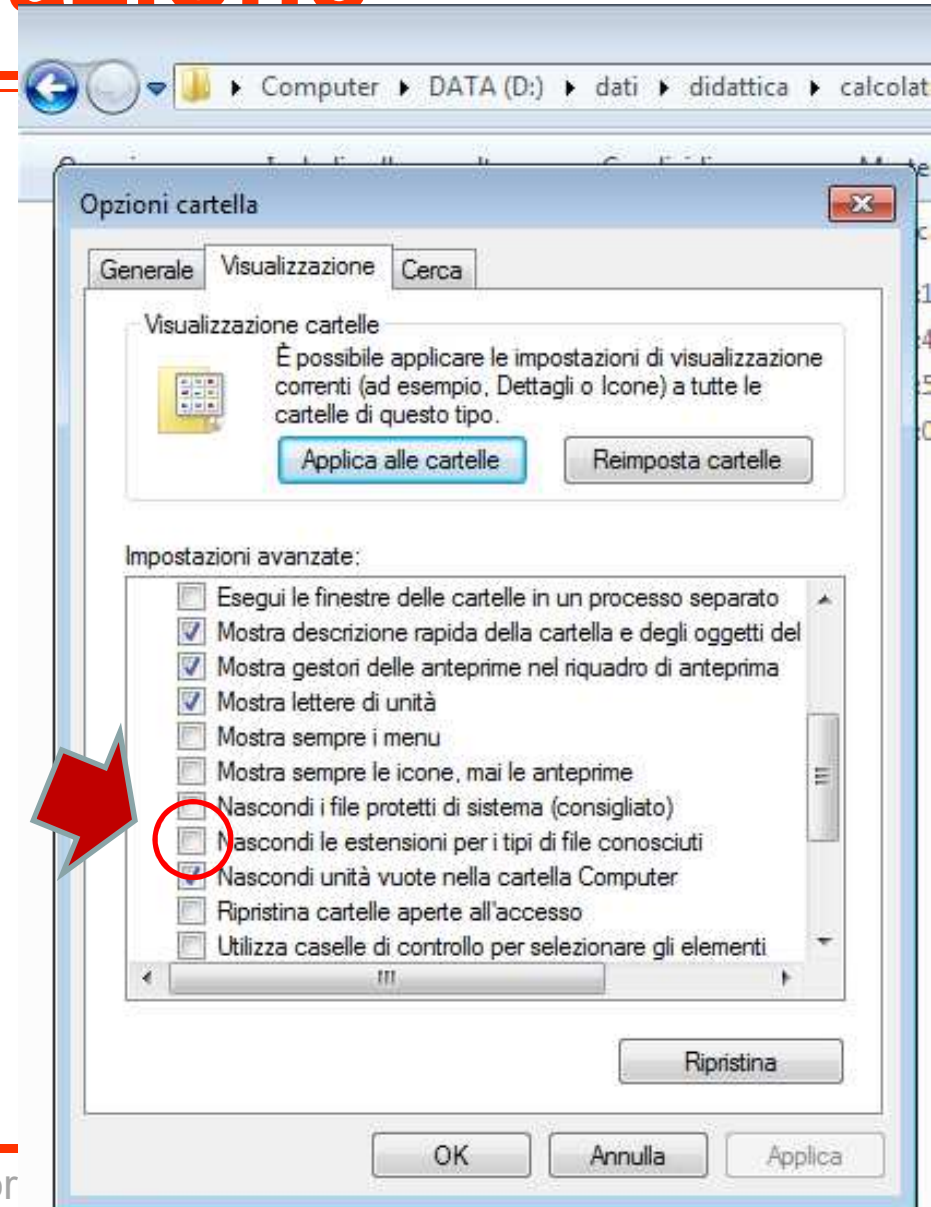
Installazione

- Per evitare problemi con il riconoscimento dei file, è necessario andare in *Esplora Risorse* → Menù *Strumenti* → Cliccare su *Opzioni Cartella*



Installazione

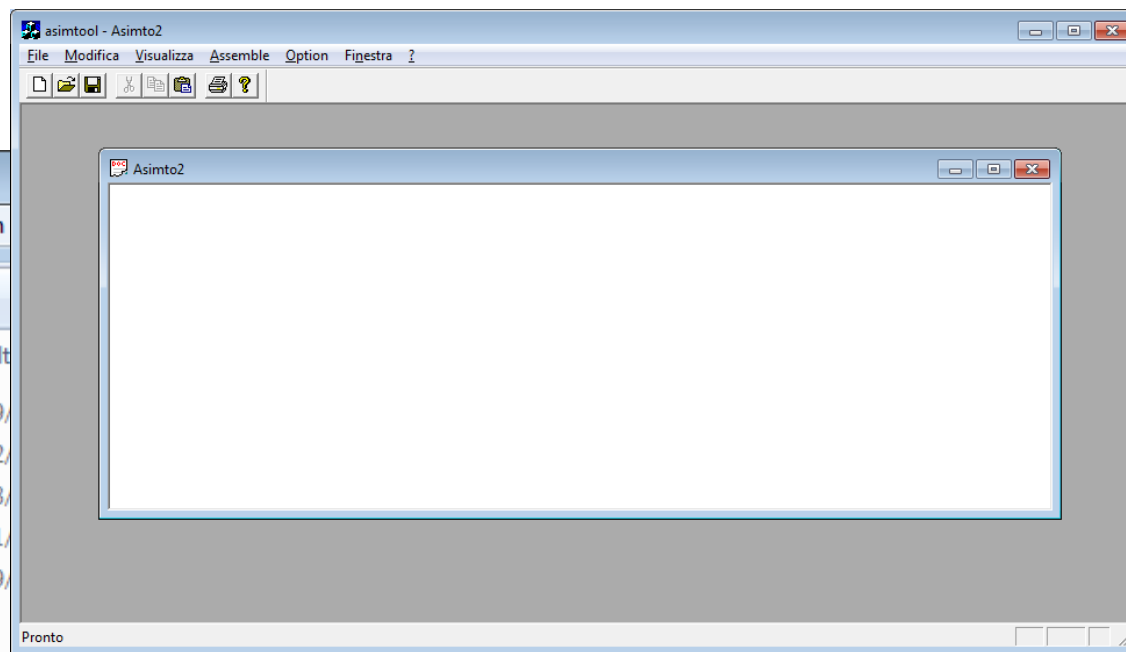
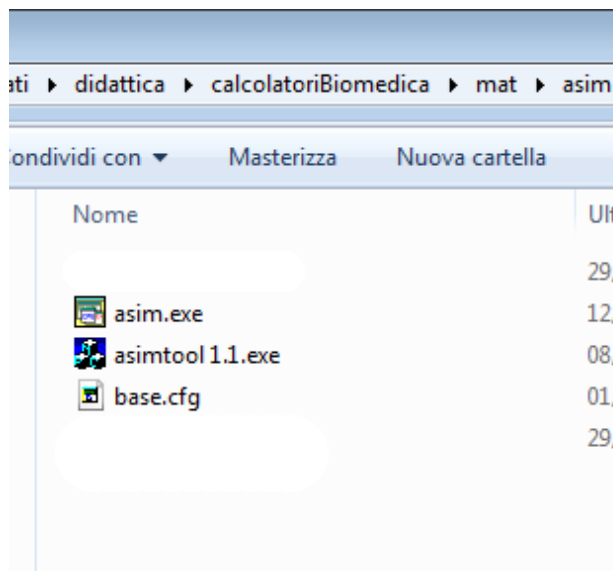
- ... poi cliccare sul tab *Visualizzazione* e scorrere l'elenco fino ad individuare l'opzione *Nascondi le estensioni per i tipi di file conosciuti*
- L'opzione deve essere **NON SPUNTATA** come in figura



AsimTool

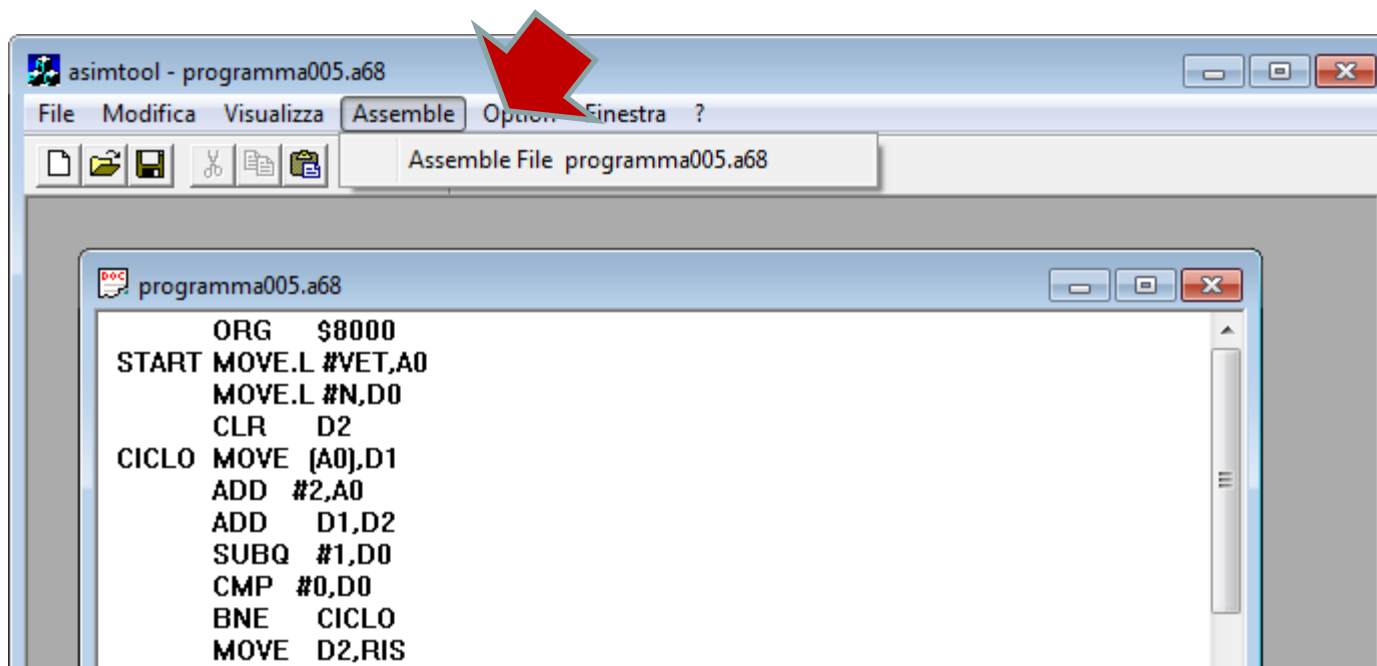
- Permette di creare, aprire, modificare e salvare file `.a68` contenente le istruzioni macchina in forma testuale (linguaggio assembly)

per lanciarlo, doppio click su `asimtool1.1.exe`



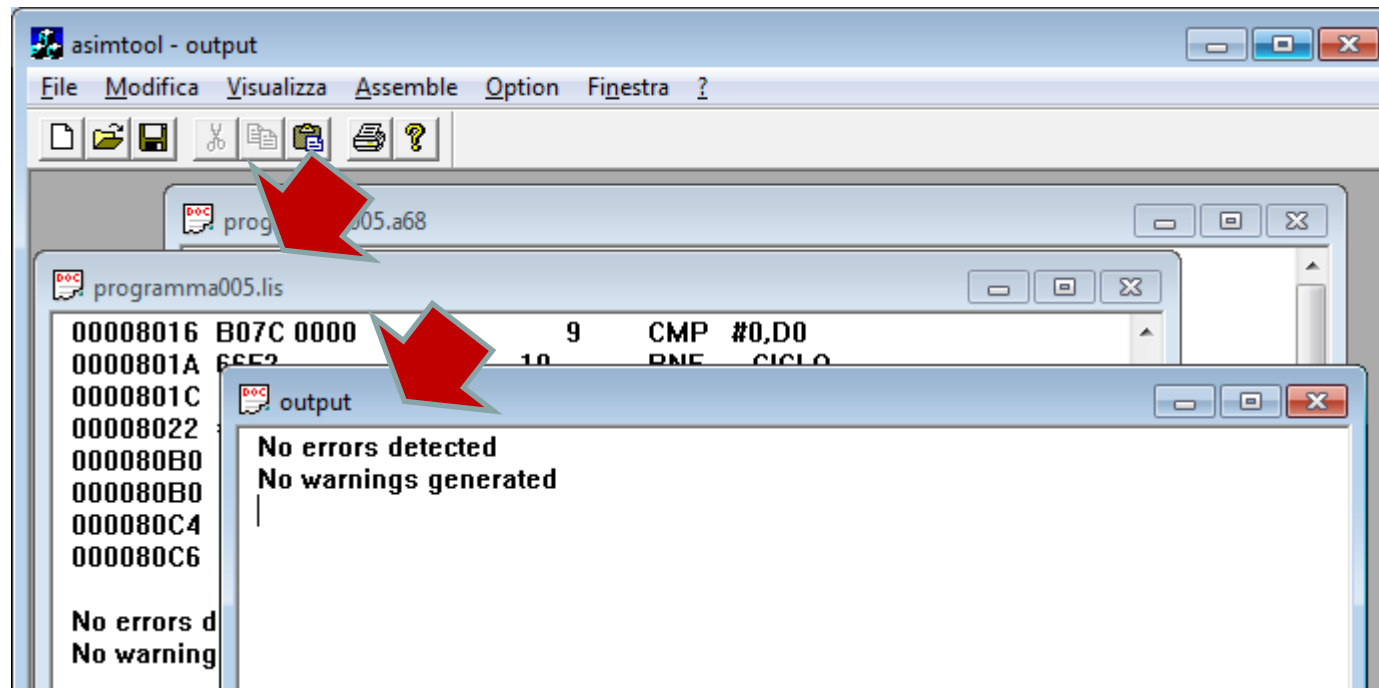
AsimTool

- Per assemblare il file, basta andare sul menu **Assemble** e cliccare sulla relativa voce



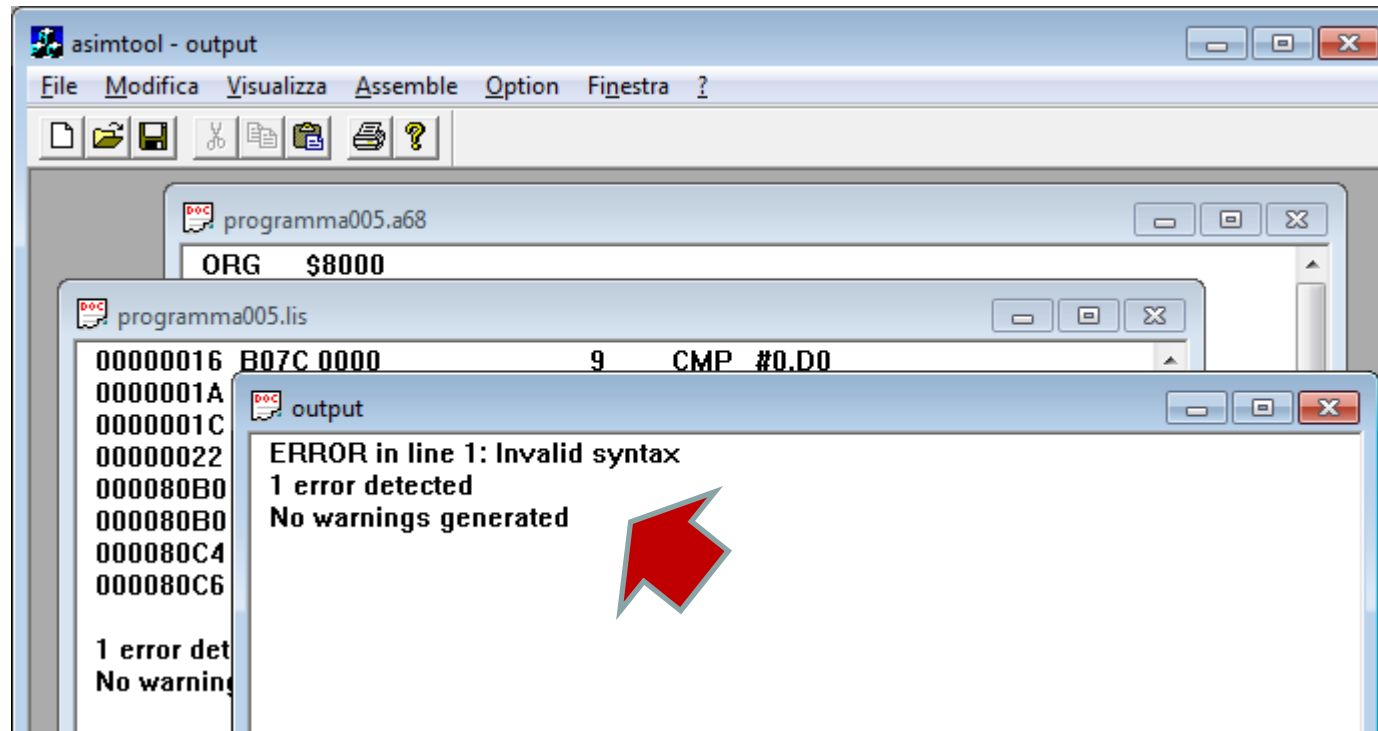
AsimTool

- Se non ci sono errori di sintassi, vengono mostrati il file di report ed il file LIS



AsimTool

- Altrimenti, viene mostrata la lista degli errori, con indicazione della riga (*line*) alla quale è stato riscontrato l'errore



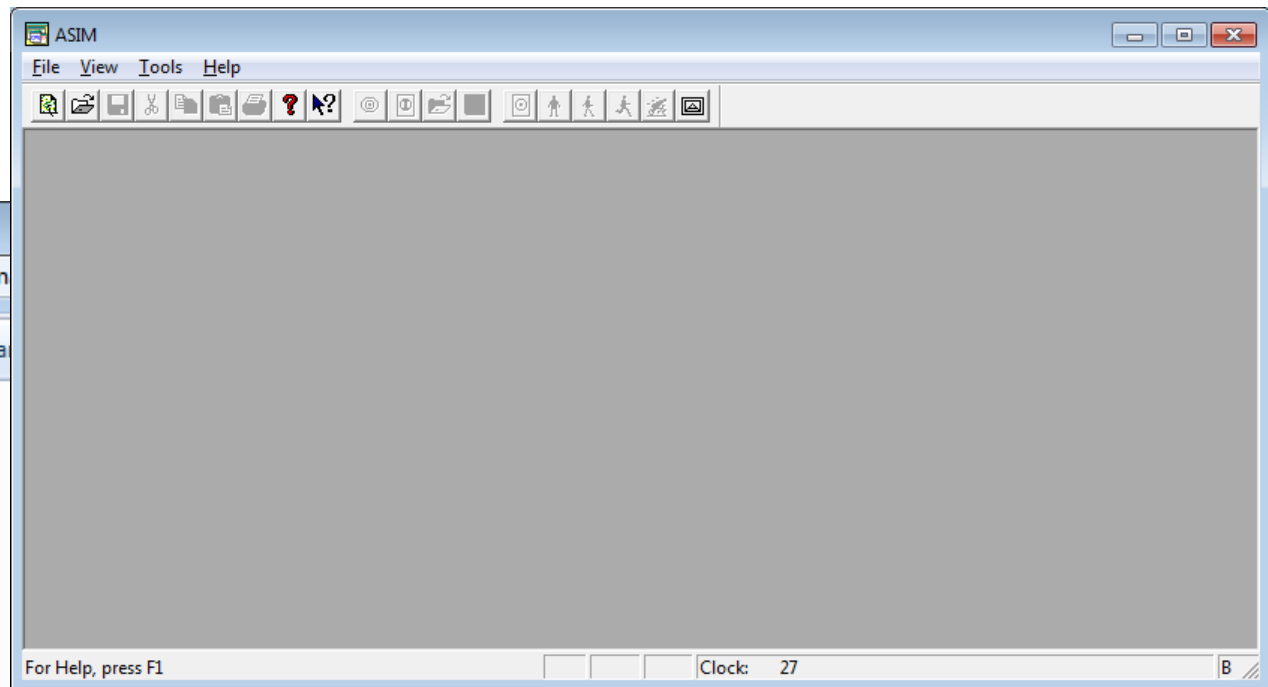
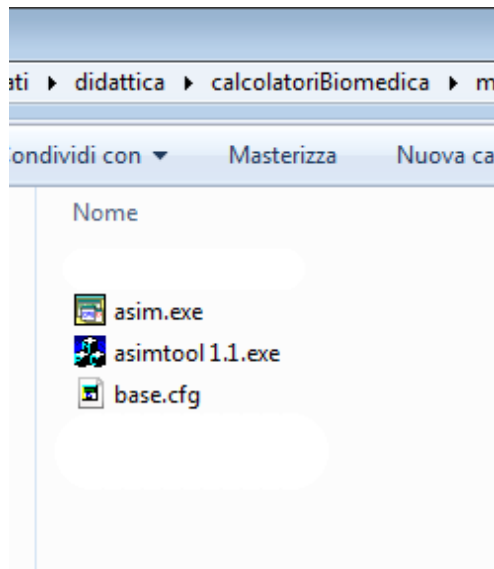
AsimTool

- Gli errori in fase di compilazione possono avere molteplici cause.
- Bisogna avere ben chiara la struttura del file assembler...
 - ad esempio, ogni riga deve sempre cominciare con un'etichetta o con uno spazio/TAB
- ...e la sintassi di ciascuna istruzione
 - non tutte le istruzioni usano gli stessi modi di indirizzamento
 - l'ordine ed il tipo di operandi è spesso importante!
 - etc. etc.

Asim

- Permette di simulare calcolatori costituiti da almeno una CPU ed una memoria, la cui architettura è descritta in un file `.cfg`

per lanciare ASIM, doppio click su `asim.exe`



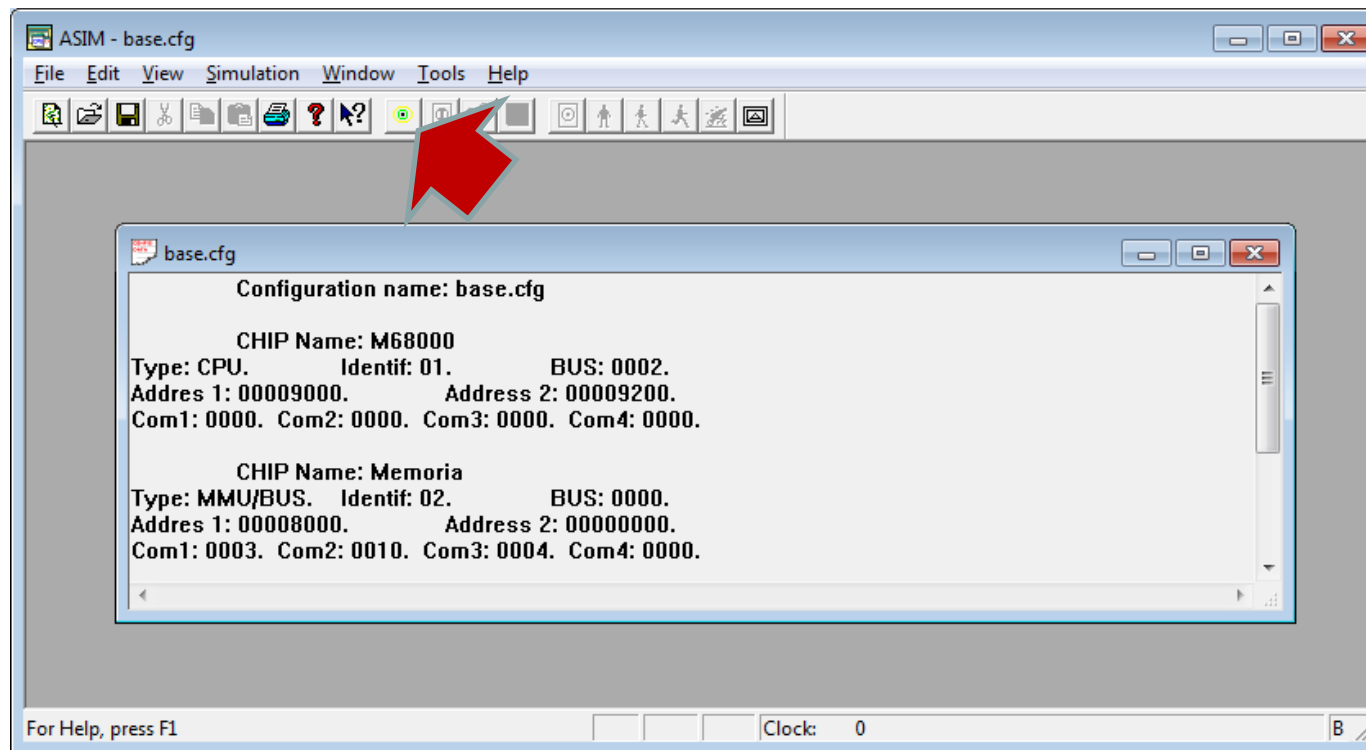
Asim

- Simuleremo solo un sistema base fatto da una CPU ed una memoria (descritto nel file `base.cfg` fornito con ASIM)
- Menu **F**ile → **o**pen. Localizzare il file `base.cfg` ed aprirlo



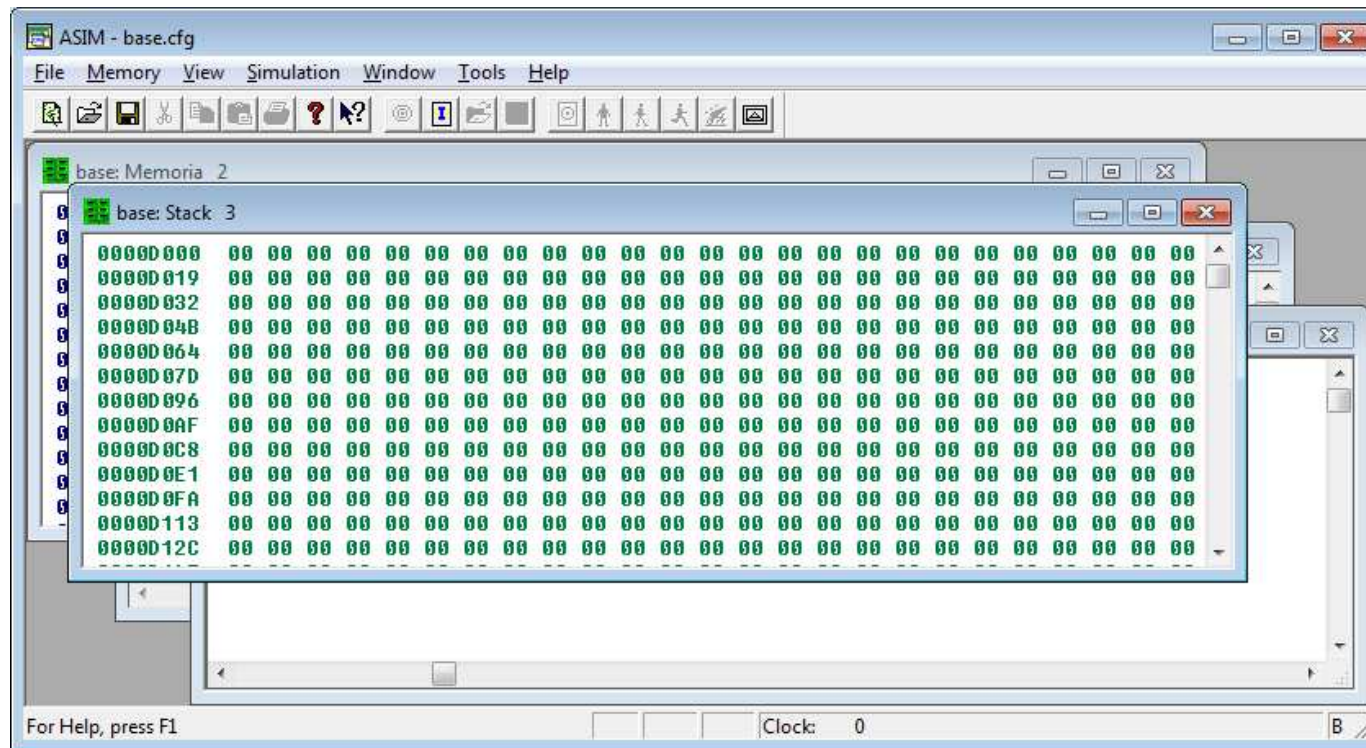
Asim

- Viene visualizzata la struttura del sistema indicata dal file. Cliccare sul bottone indicato sotto per “costruire” il sistema.



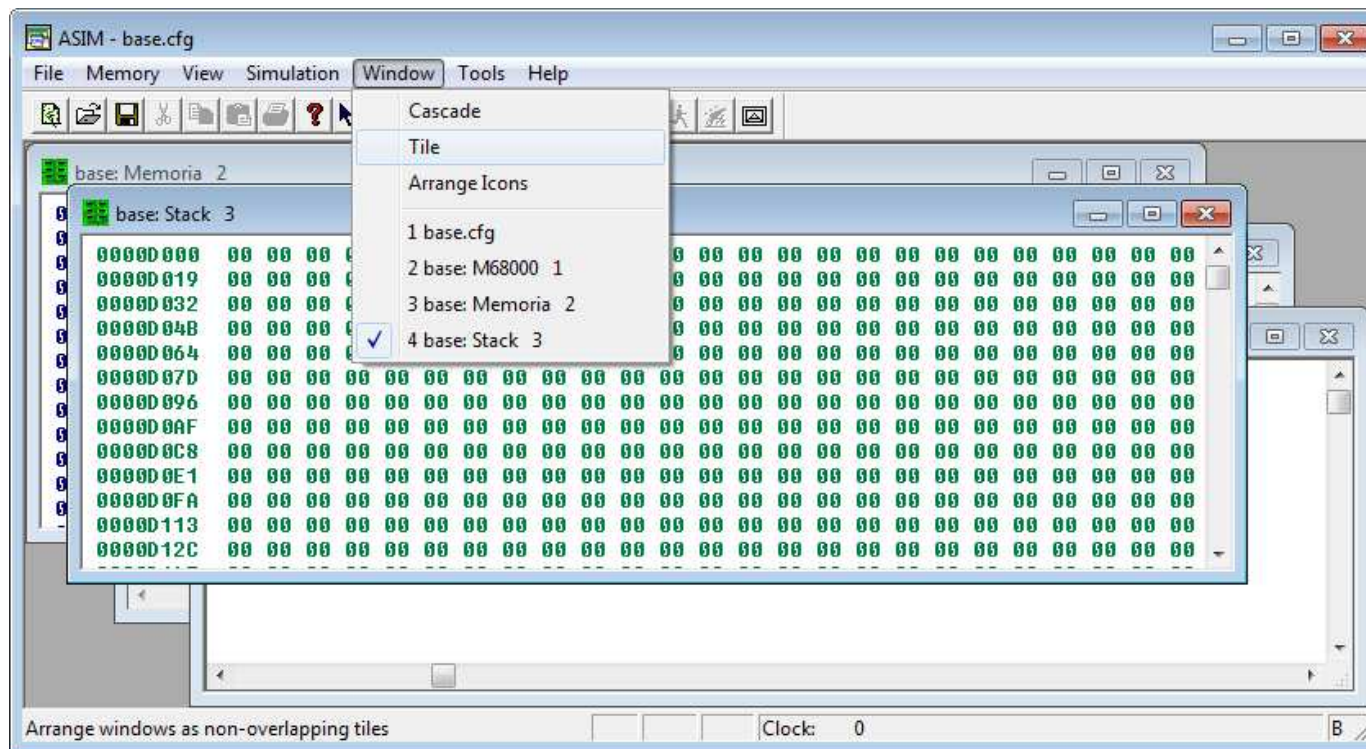
Asim

- Vengono create diverse finestre, una per ogni componente del sistema (memoria, CPU, etc)



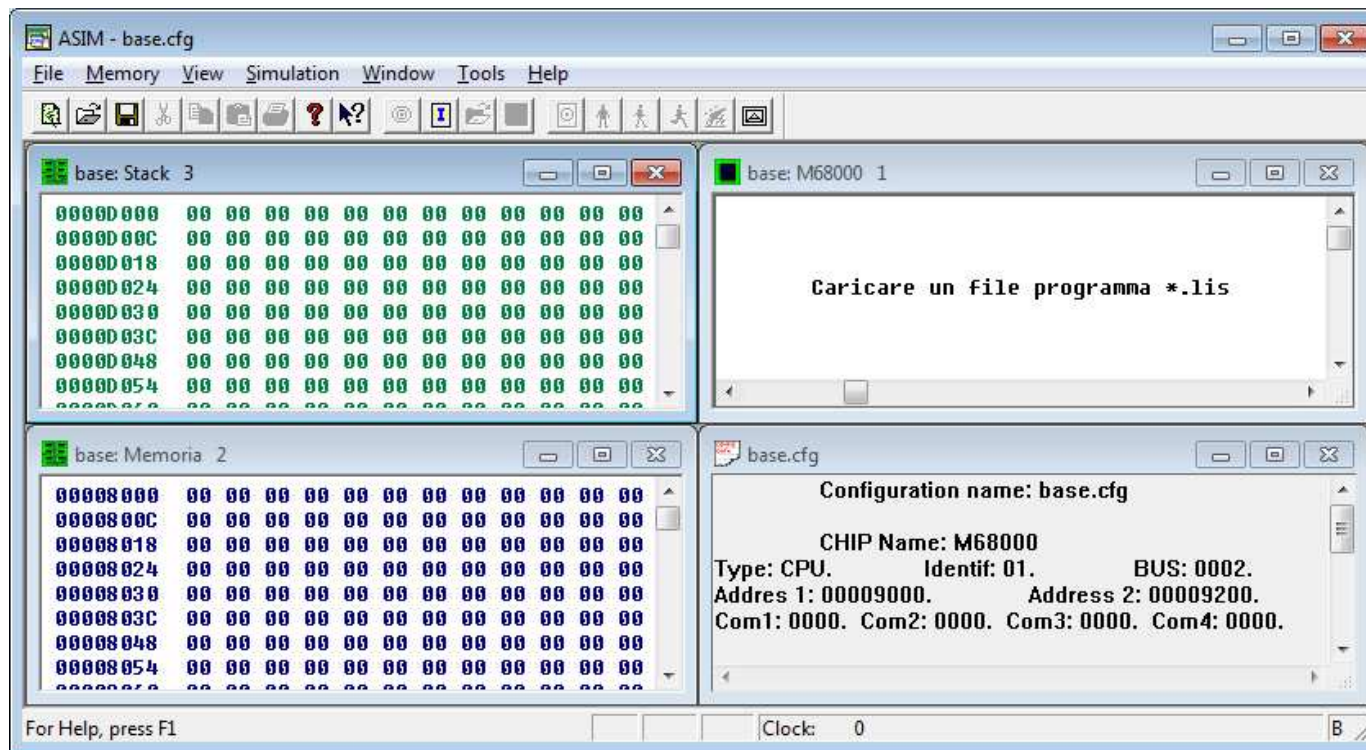
Asim

- Per organizzare meglio le finestre, andare nel menù **Window** → **Tile**



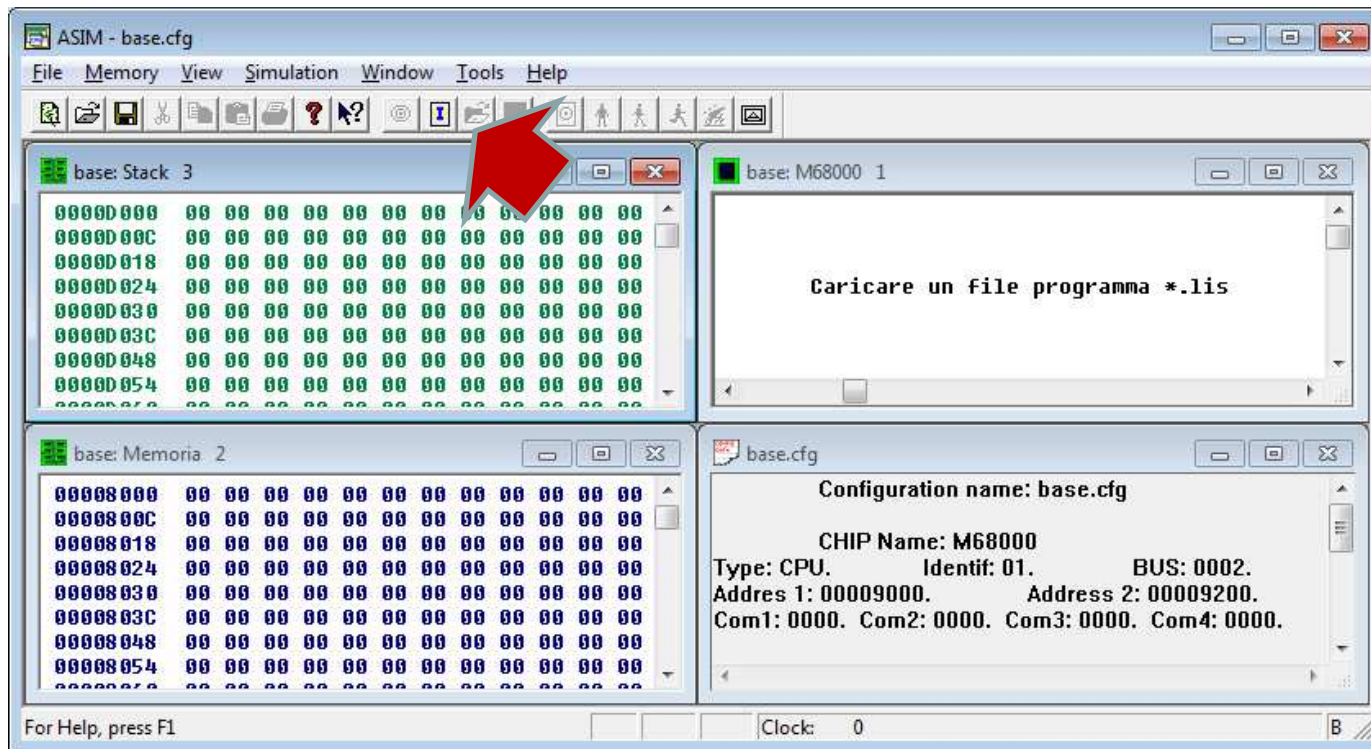
Asim

- Le finestre vengono affiancate. Quella relativa al processore reca il titolo: “**base: M68000 1**”



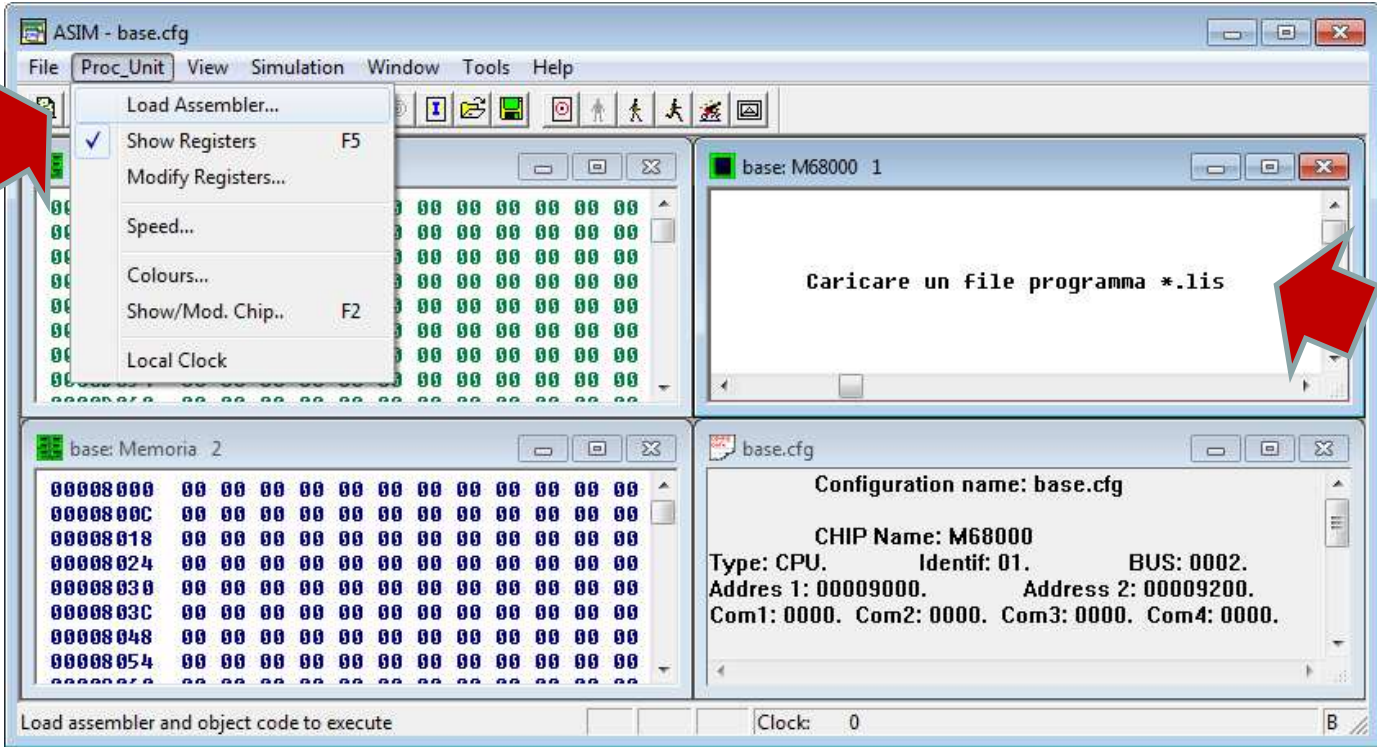
Asim

- Inizializzare il sistema cliccando sul bottone indicato in figura



Asim

- Per caricare un programma, cliccare sulla finestra “M68000”. Quindi, menù Proc_Unit → Load Assembler



The screenshot shows the ASIM software interface with the following components:

- Proc_Unit Menu:** A red arrow labeled (2) points to the 'Load Assembler...' option in the Proc_Unit menu.
- M68000 Window:** A red arrow labeled (1) points to the 'Caricare un file programma *.lis' text in the 'base: M68000 1' window.
- Memory Window:** A window titled 'base: Memoria 2' showing a memory dump with addresses and hex values.
- Configuration Window:** A window titled 'base.cfg' showing configuration details for the M68000 chip.

Configuration details from the 'base.cfg' window:

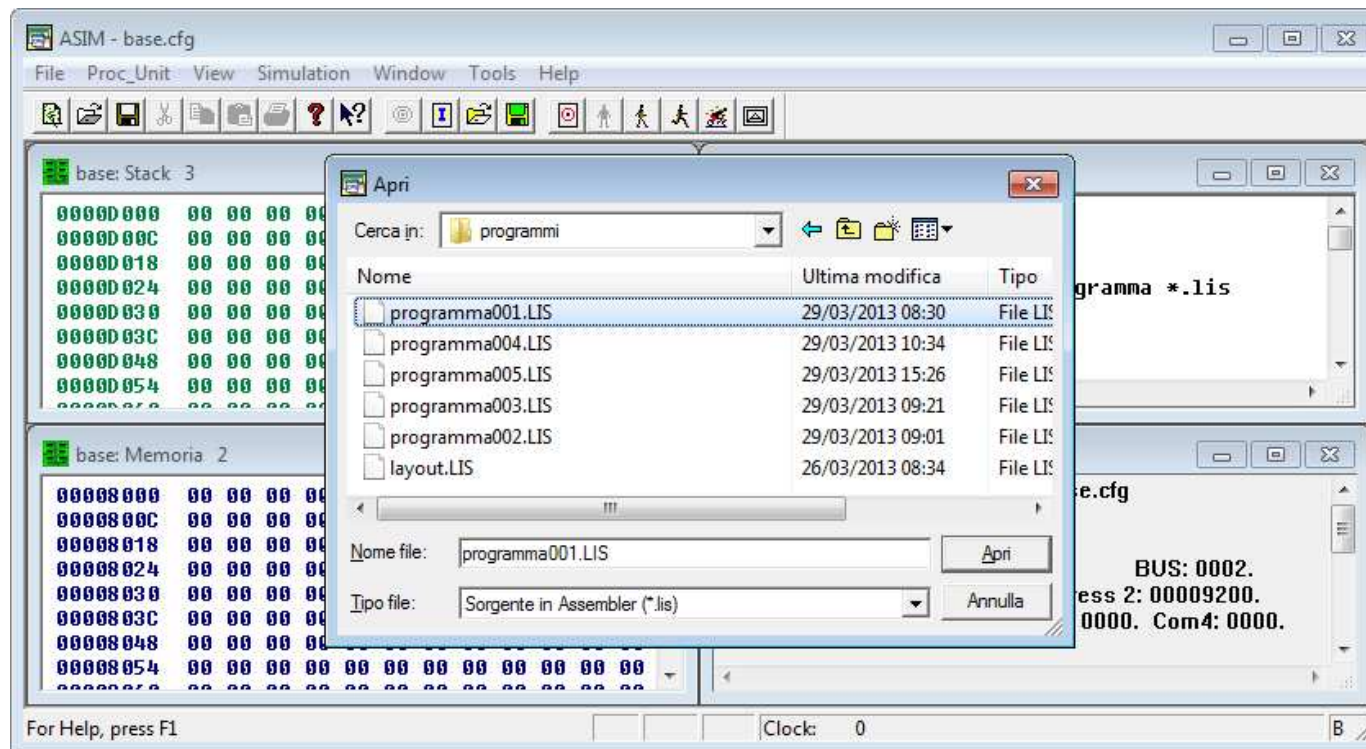
```
Configuration name: base.cfg
CHIP Name: M68000
Type: CPU.          Identif: 01.          BUS: 0002.
Address 1: 00009000. Address 2: 00009200.
Com1: 0000. Com2: 0000. Com3: 0000. Com4: 0000.
```

Proc_Unit
→
Load Assembler

(1)
Cliccare sulla finestra M68000

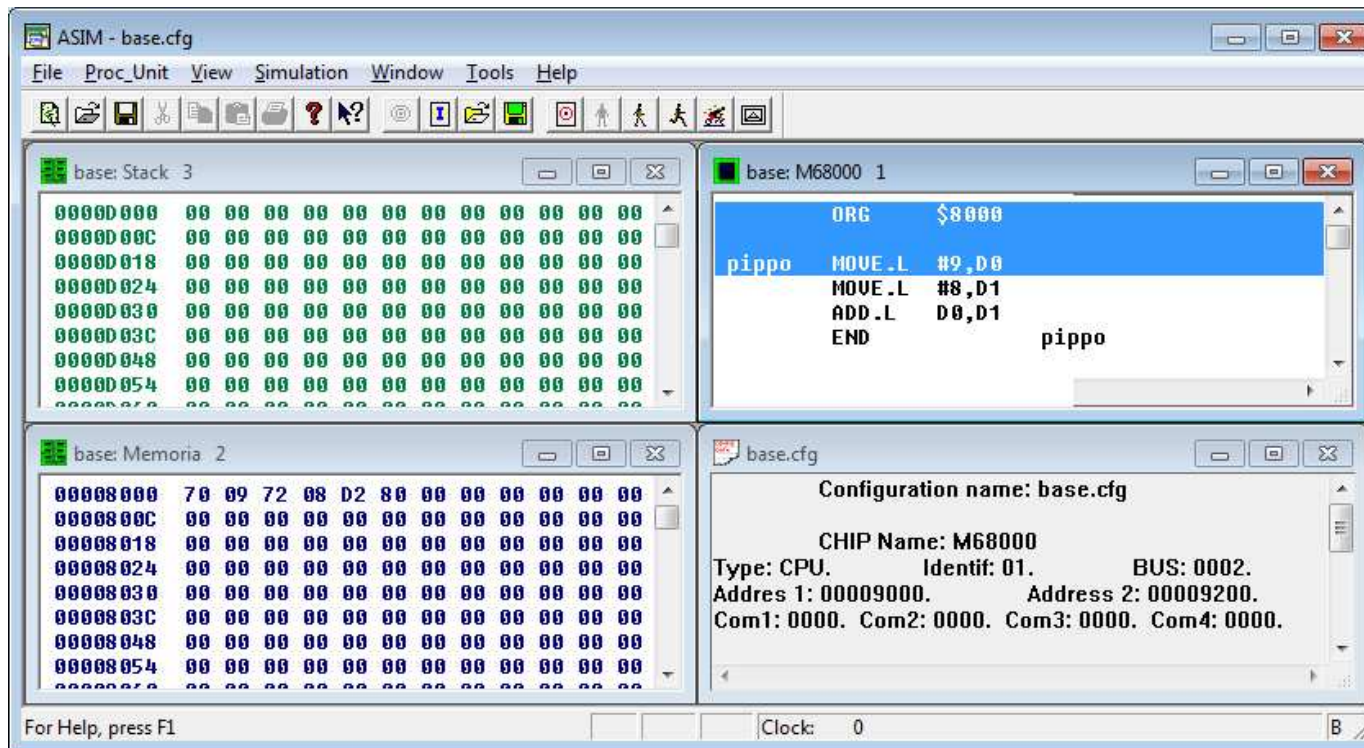
Asim

- Localizzare il file .LIS ed aprirlo



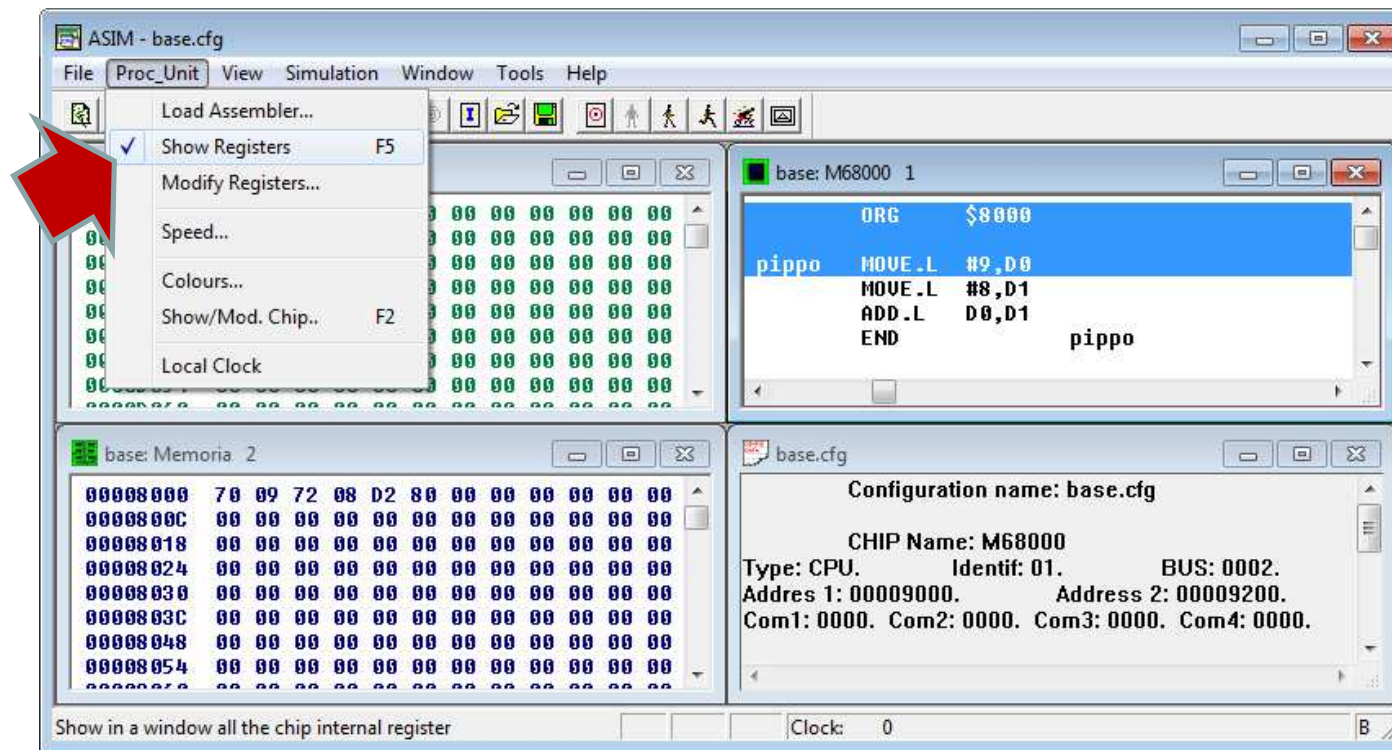
Asim

- Il programma appare nella finestra “M68000”. La prossima istruzione da simulare è evidenziata in blu



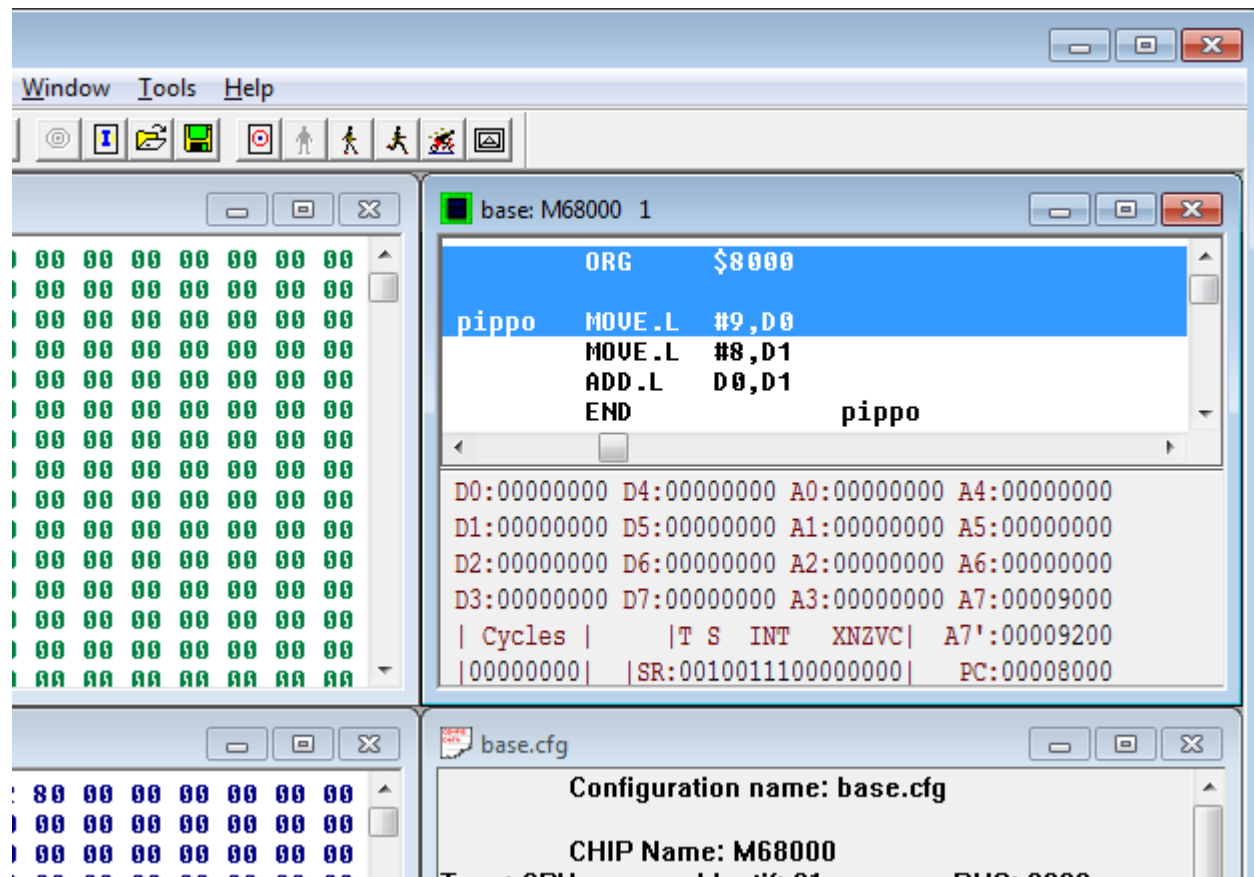
Asim

- Per visualizzare i registri del processore, dopo aver cliccato sulla finestra “M68000”, cliccare sul menù Proc_Unit, quindi sul comando Show Registers



Asim

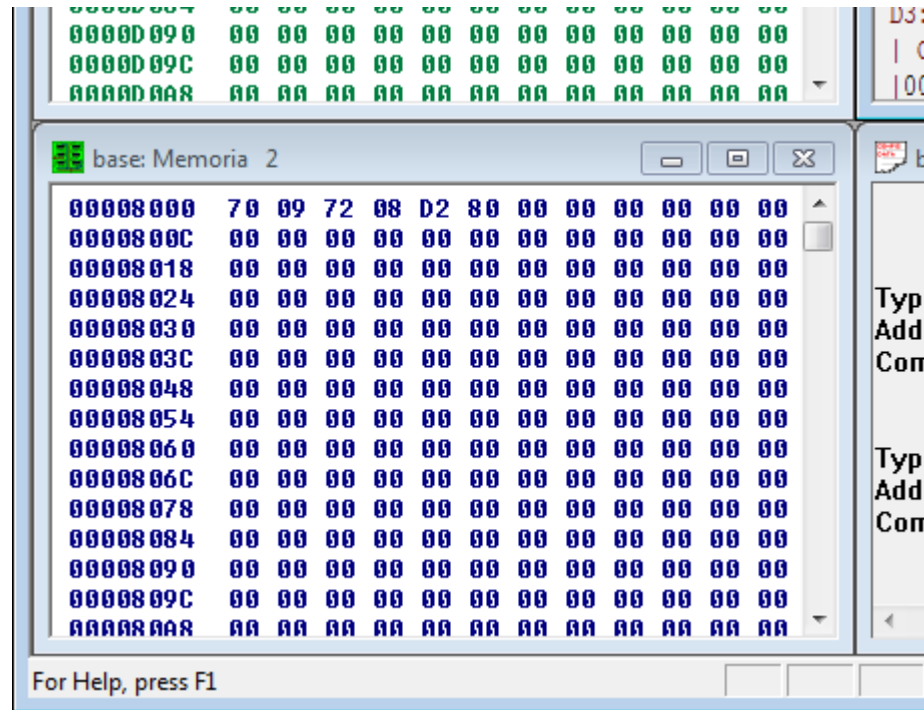
- I registri appaiono all'interno della finestra relativa al processore, sotto le istruzioni del programma



Asim

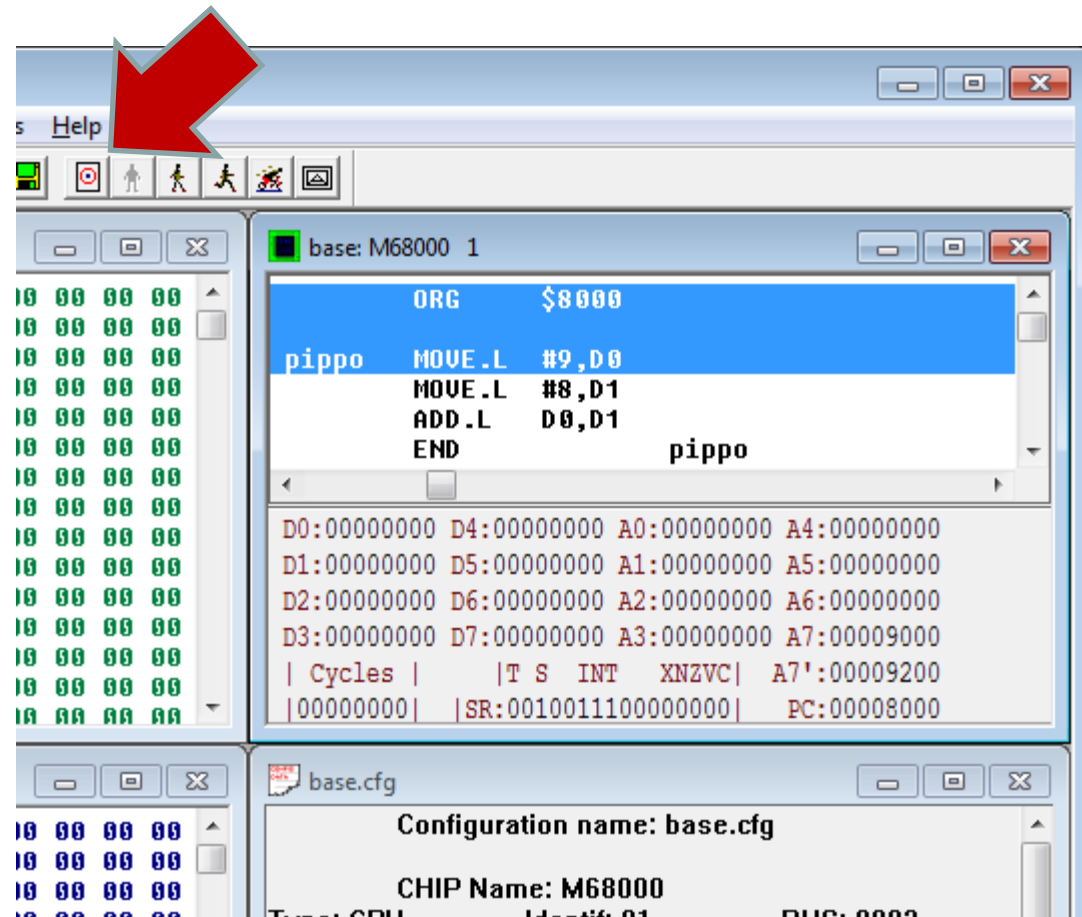
- E' inoltre possibile verificare il contenuto della memoria, byte per byte, esaminando la finestra dal titolo **“base: Memoria 2”**

- Ogni riga mostra una sequenza di byte contenuta in memoria, riportando sulla sinistra l'indirizzo del primo byte visualizzato



Asim

- Dare un comando di *reset* al sistema, in modo da portarlo nello stato iniziale, cliccando sul pulsante indicato in figura



Asim

- E' infine possibile simulare il programma
- I due pulsanti indicati permettono di eseguire il programma un'istruzione alla volta (passo-passo), oppure l'intero programma senza interruzioni
- I registri e la memoria vengono aggiornati durante la simulazione in base all'effetto delle istruzioni eseguite

esecuzione
passo-passo

esecuzione veloce
senza interruzioni

