# EECE416 :Microcomputer Fundamentals and Design

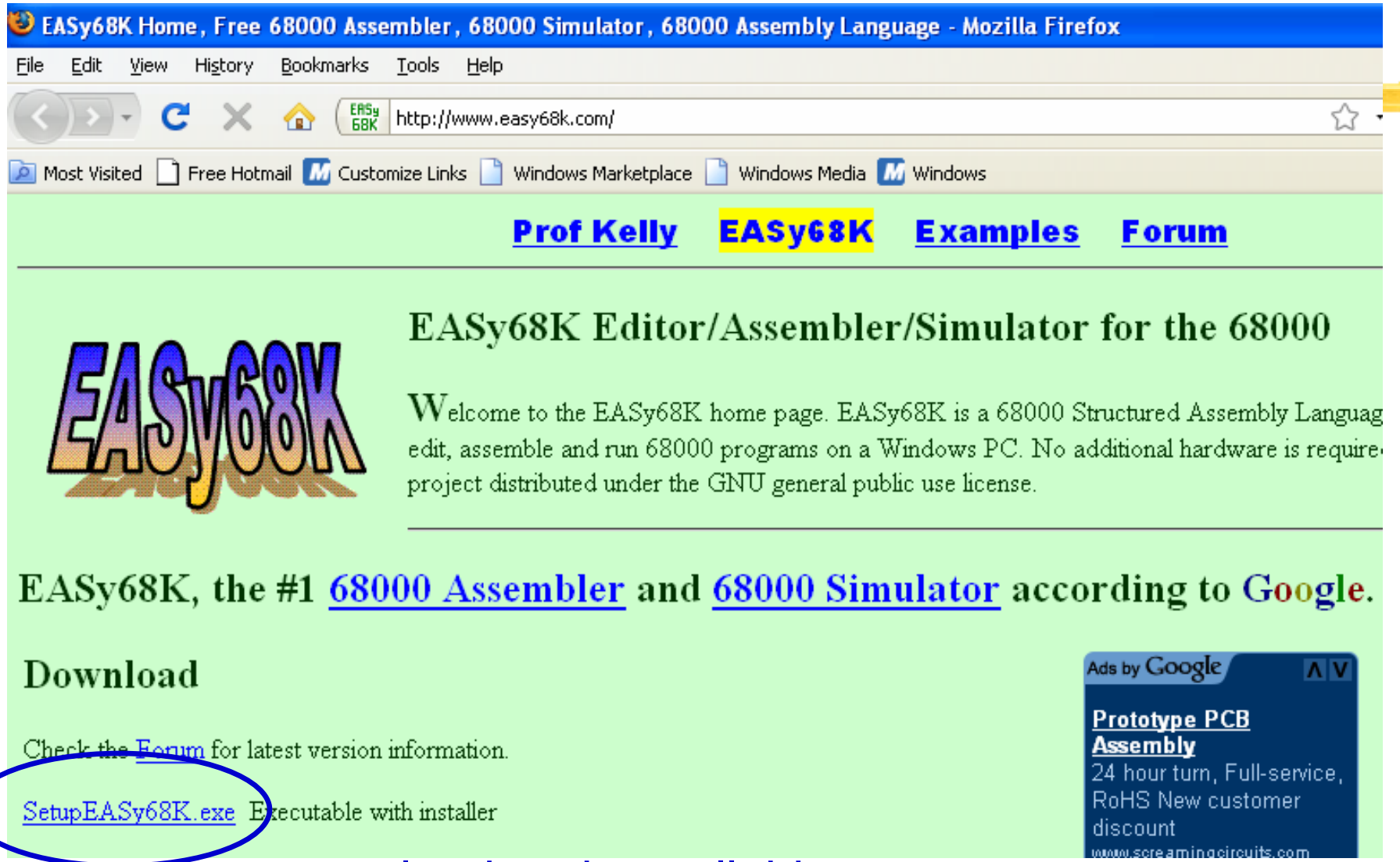
Easy68K

– Editor, Assembler, and Simulator

**TUTORIAL**

Charles Kim

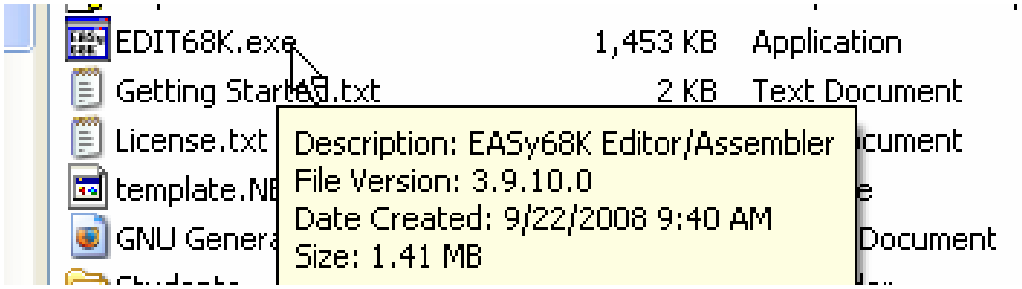
# Easy68K and Download



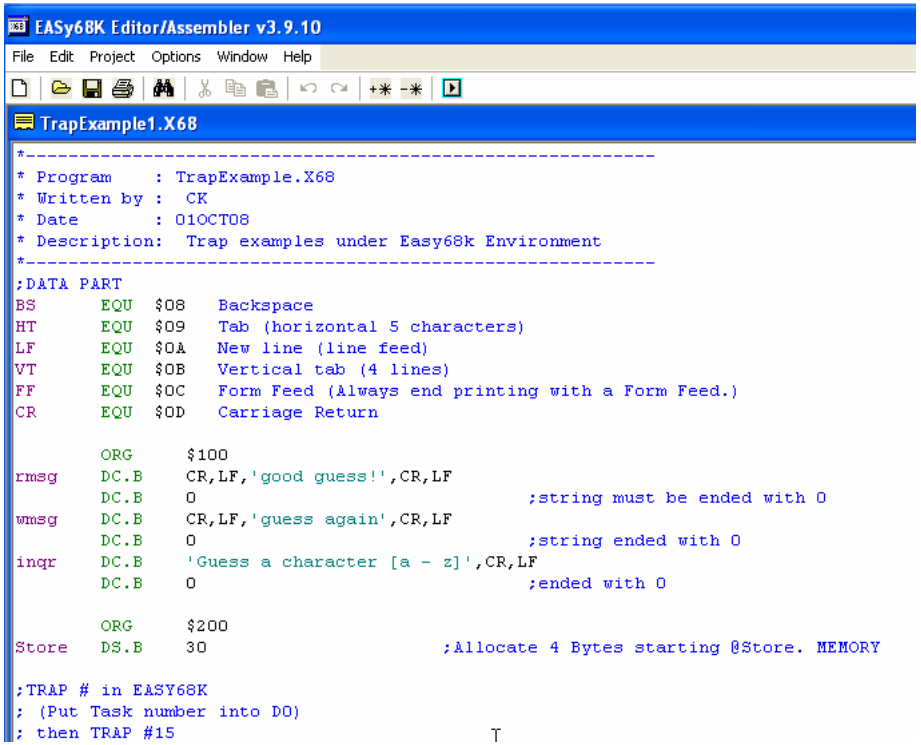

Download and Install this!

# EDIT68K

- EDIT68K.EXE
  - For Coding and Assembling

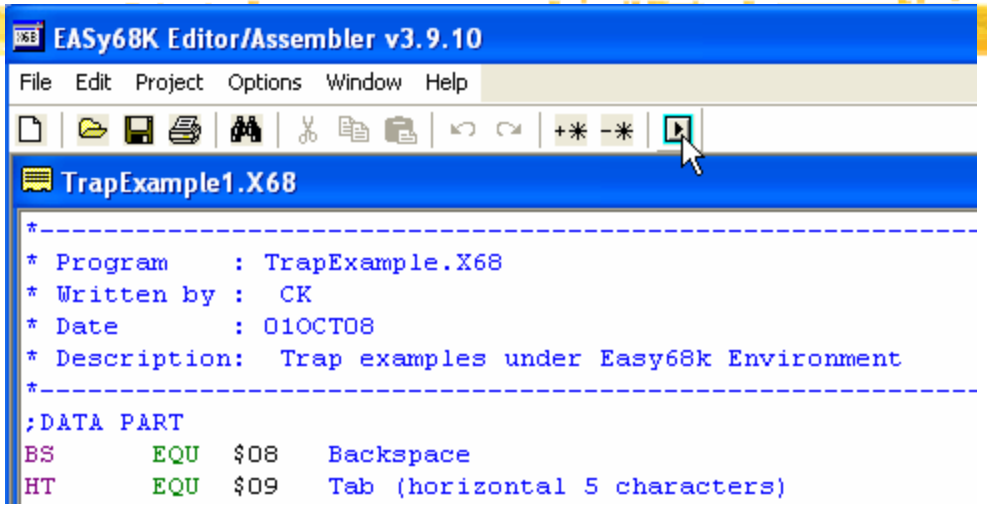



- Coding
- Save

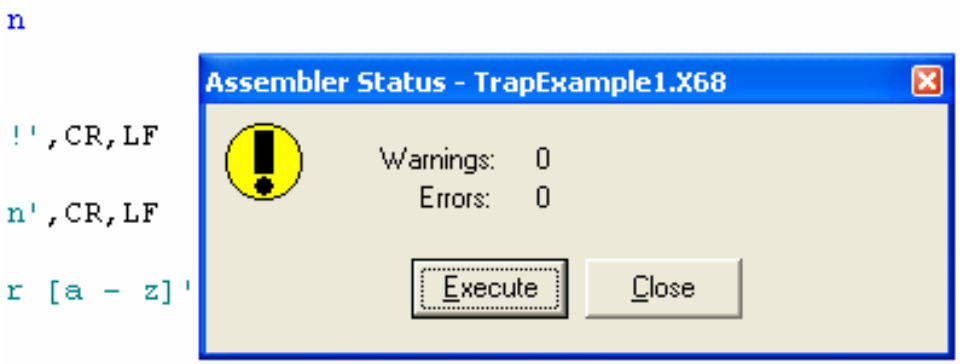


# Assembling

- Click the “Assemble Source Code” button OR F9 Key

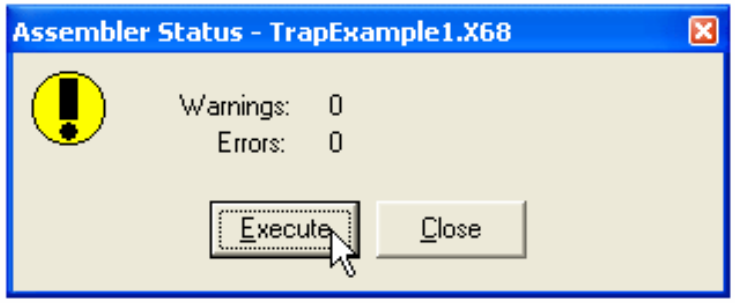



- When there is no Error, the following window pops up.

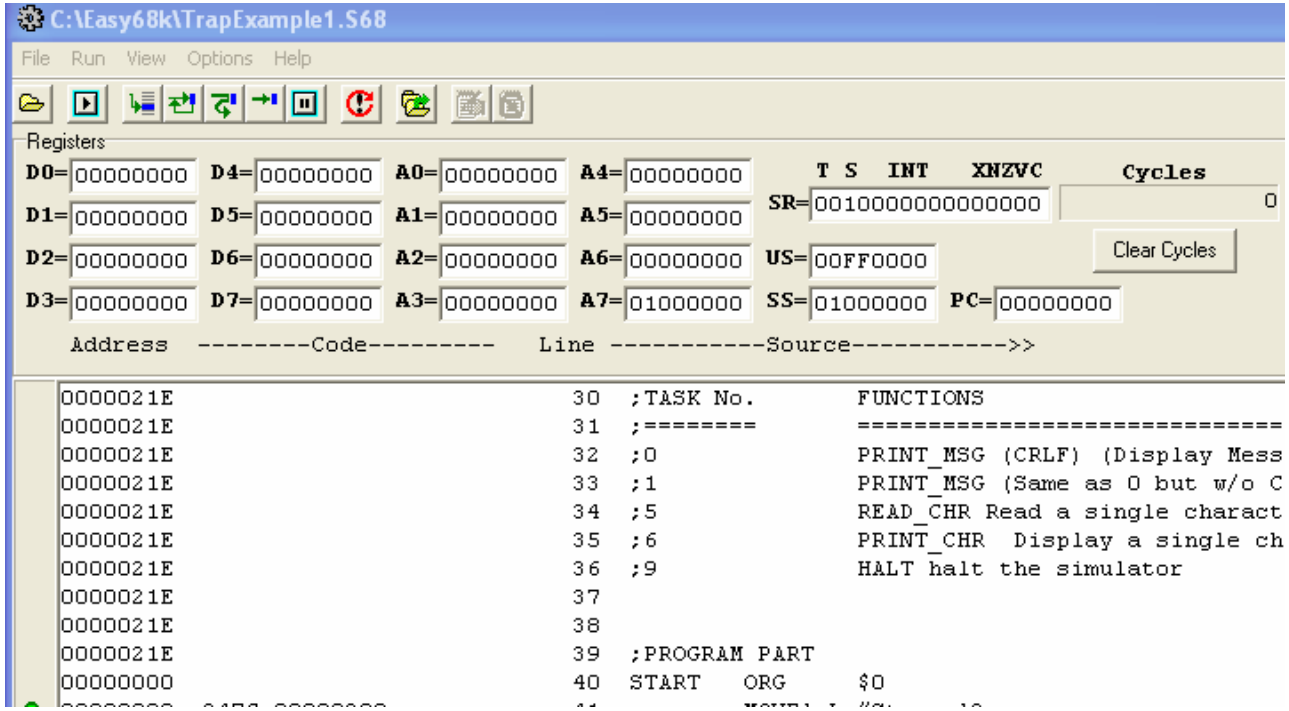


# Simulation

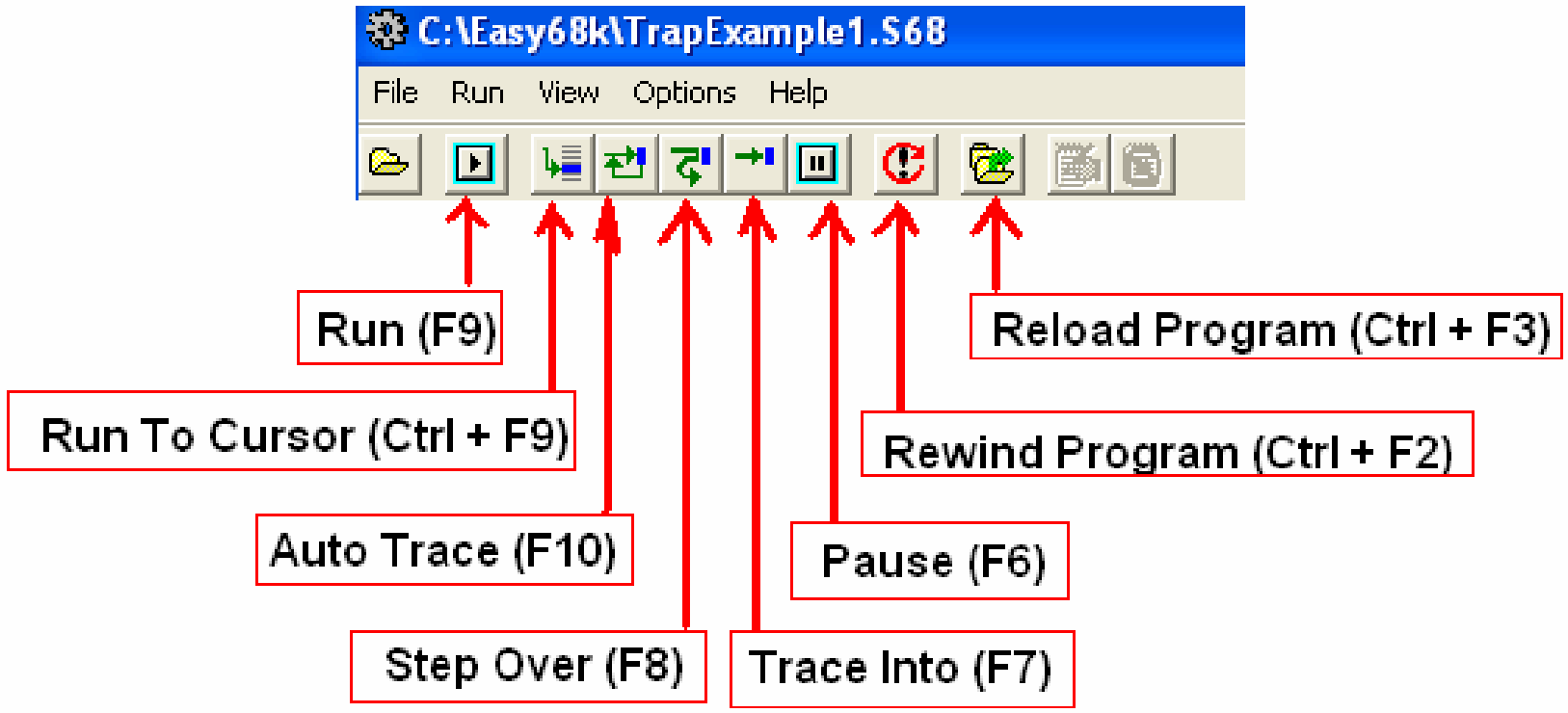
- Click, “Execute” button.




- Initial Simulation Screen
-

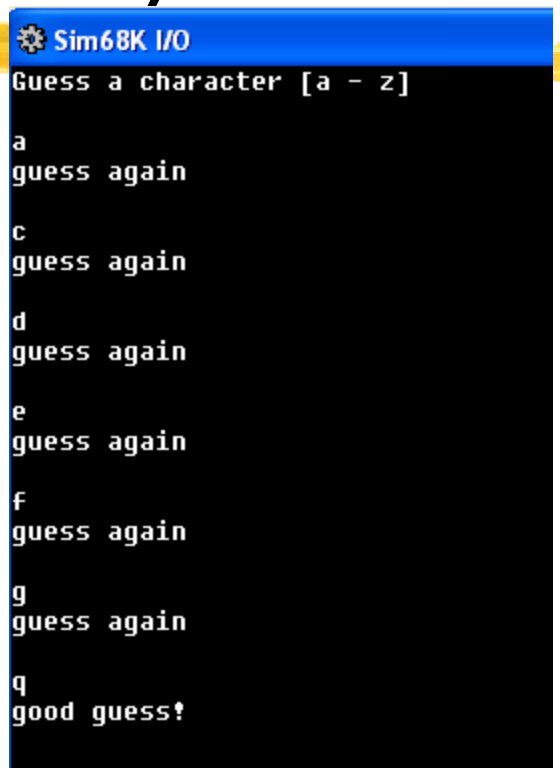


# Simulation Buttons

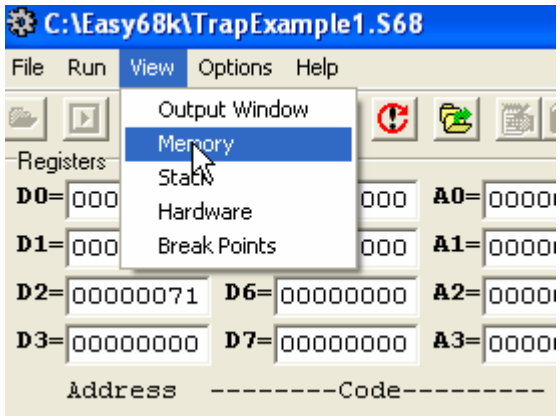


# Memory View

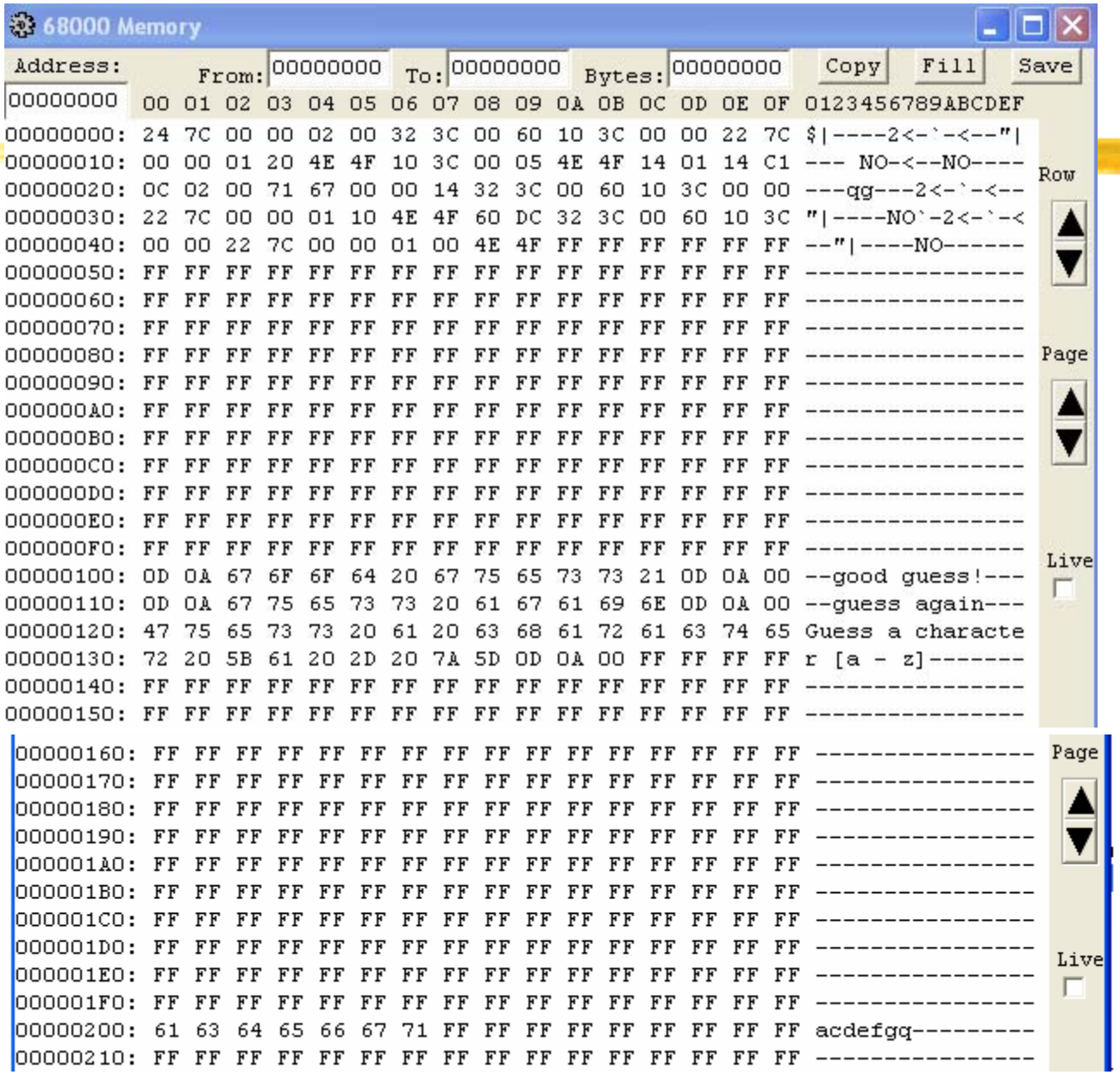
- **After Auto Trace or Run (with Cmd screen)**




- **Click “View” and Select “Memory”**



# Memory View


68000 Memory
Address:
From: 00000000
To: 00000000
Bytes: 00000000
Copy
Fill
Save
00000000 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 0123456789ABCDEF
00000000: 24 7C 00 00 02 00 32 3C 00 60 10 3C 00 00 22 7C $|----2<-`-<--"|
00000010: 00 00 01 20 4E 4F 10 3C 00 05 4E 4F 14 01 14 C1 --- NO-<--NO----
00000020: 0C 02 00 71 67 00 00 14 32 3C 00 60 10 3C 00 00 ---qg---2<-`-<--
00000030: 22 7C 00 00 01 10 4E 4F 60 DC 32 3C 00 60 10 3C "|----NO`-2<-`-<
00000040: 00 00 22 7C 00 00 01 00 4E 4F FF FF FF FF FF FF --"|----NO------
00000050: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ----------------
00000060: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ----------------
00000070: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ----------------
00000080: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ----------------
00000090: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ----------------
000000A0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ----------------
000000B0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ----------------
000000C0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ----------------
000000D0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ----------------
000000E0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ----------------
000000F0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ----------------
00000100: 0D 0A 67 6F 6F 64 20 67 75 65 73 73 21 0D 0A 00 --good guess!---
00000110: 0D 0A 67 75 65 73 73 20 61 67 61 69 6E 0D 0A 00 --guess again---
00000120: 47 75 65 73 73 20 61 20 63 68 61 72 61 63 74 65 Guess a characte
00000130: 72 20 5B 61 20 2D 20 7A 5D 0D 0A 00 FF FF FF FF r [a - z]-------
00000140: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ----------------
00000150: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ----------------
Row
Page
Live
00000160: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ----------------
00000170: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ----------------
00000180: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ----------------
00000190: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ----------------
000001A0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ----------------
000001B0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ----------------
000001C0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ----------------
000001D0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ----------------
000001E0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ----------------
000001F0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ----------------
00000200: 61 63 64 65 66 67 71 FF FF FF FF FF FF FF FF FF acdefgq---------
00000210: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ----------------
Page
Live

# TRAPs in EASY68K

- TRAP
  - Interruption of Execution
  - Respond Programmer's Task Command
- Number of Tasks in Easy68K
  - 22 Tasks
  - Examples
    - Read Message (from Keyboard)
    - Print Message (to Computer Monitor)
    - Read a Character
    - Print a Character
    - Read a Number
    - Print a Number
    - Etc
- **Traditional TRAP tasks**
    - **Print Message (to Computer Monitor) of address at A1**
      - **--- Task # 0 (with CR.LF)**
      - **--- Task #1 (without CR.LF)**
    - **Read a Character and store it at D1.B --- Task #5**
    - **Print a Character stored at D1.B --- Task #6**
    - **Key Echo On on or off --- Task #12 (OFF– D1.B==0, ON --- D1.B== Non zero)**
- How is a TRAP called/executed?
  - Put task number to A1 or D0
  - Put address into A1 (for "print message" case)
  - Then "Trap #15" instruction line

```
MOVE.B   #5,D0          Read a single character
TRAP     #15            Store into D1
```

```
MOVE.B   #0,D0
MOVEA.L  #rmsg, A1
TRAP     #15
```