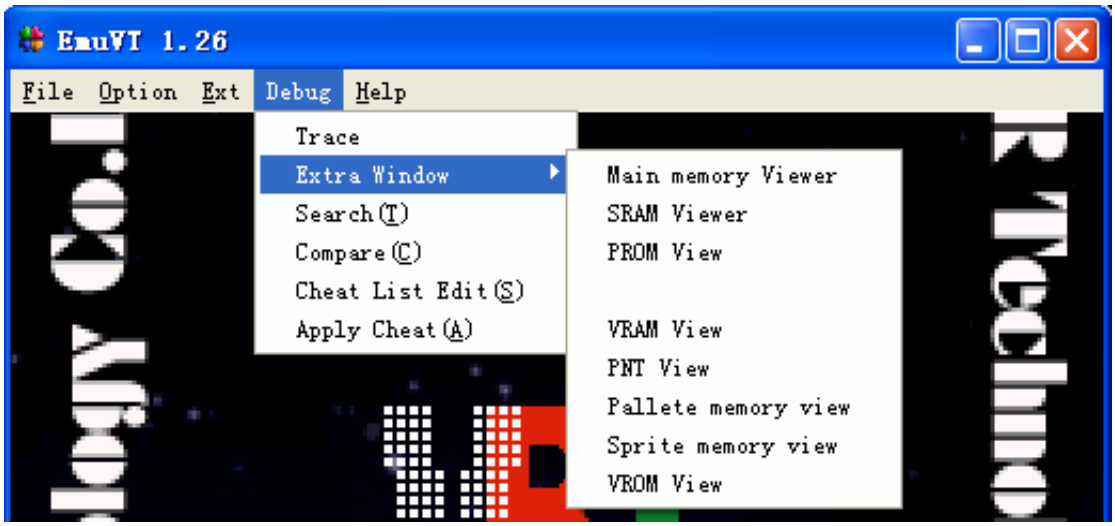
Debug Windows users note:

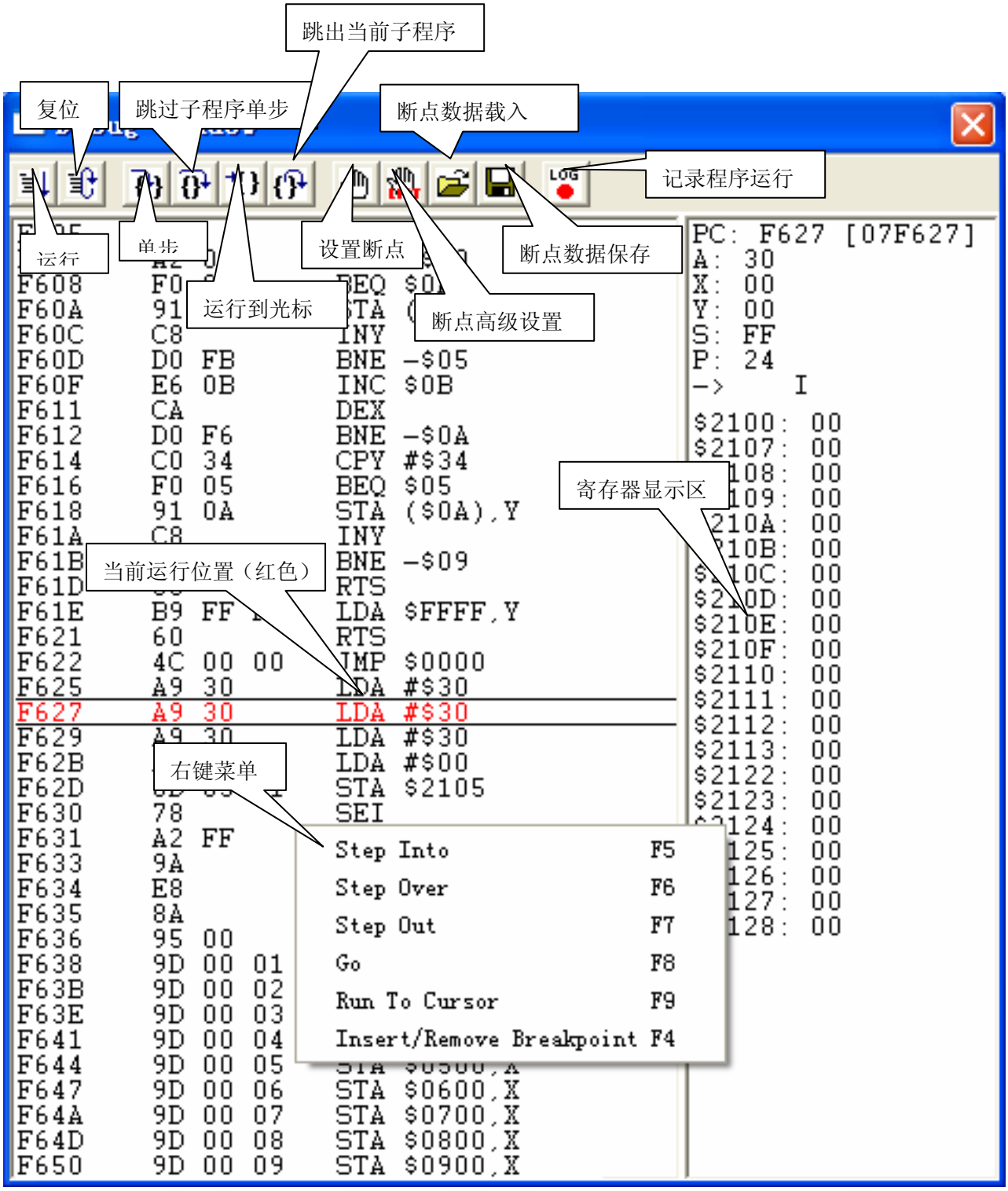
设置完之后，Trace Set 窗口可以关闭也可以不关闭，点运行那个按钮（Debug Windows 左起第一个按钮）才能跟踪，或者点复位那个按钮 （Debug Windows 左起第一个按钮）然后点运行那个按钮（Debug Windows 左起第二个按钮）。在 Debug 的过程中不要把 Debug Windows 关闭，否则不能跟踪。Trace Set 窗口中的“Enable”选项只是显示这个 break point 是否使能,而且是反的,没勾的时候是使能，有勾的时候是禁止。

Debug 菜单

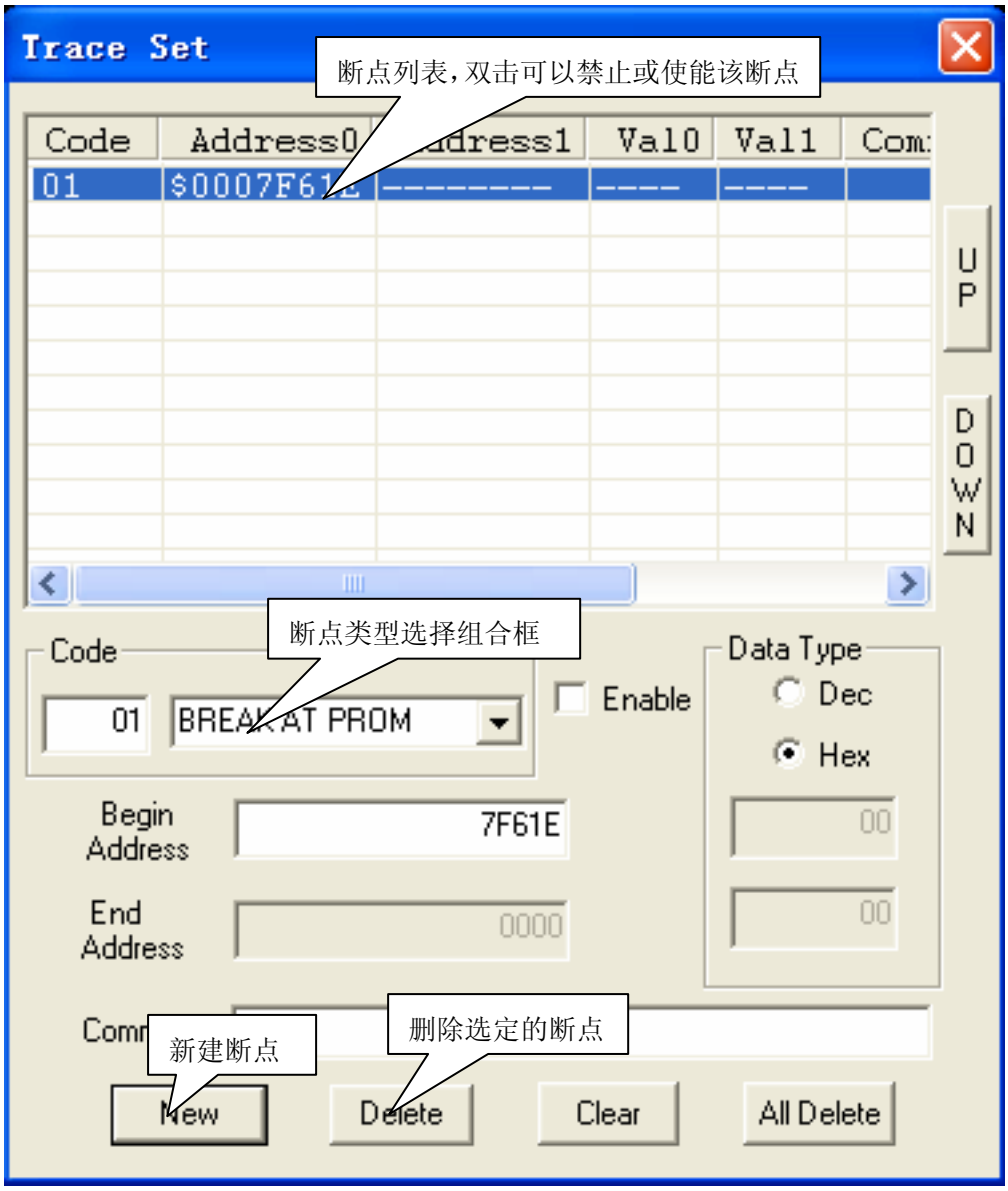


Trace	跟踪调试窗口
Extra Window	内存查看器
Search	主内存搜索查找
Compare	主内存比较
Cheat List Edit	金手指设置器
Apply Cheat	金手指使能

Trace 跟踪调试窗口



按下“断点高级设置”：



断点类型：

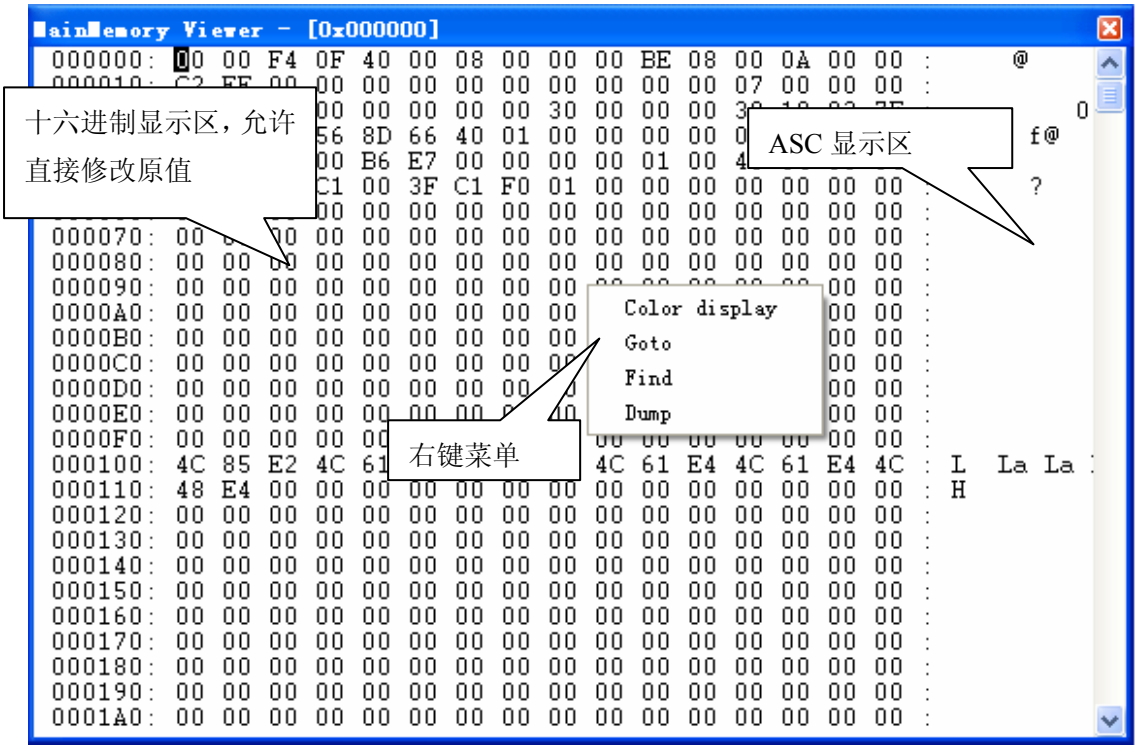
BREAK AT PROM	运行到物理地址中断
BREAK AT ADDRESS	运行到逻辑地址中断
BREAK IN ADDRESS	运行到逻辑地址范围中断
BREAK IN PROM	运行到物理地址范围中断
BREAK AT WRITE	写指定地址（寄存器）中断
BREAK IN WRITE	写指定地址（寄存器）范围中断
BREAK AT WRITE AT	以指定值写指定地址（寄存器）中断
BREAK IN WRITE AT	以指定值写指定地址（寄存器）范围中断
BREAK AT WRITE IN	以指定范围值写指定地址（寄存器）中断

BREAK IN WRITE IN	以指定范围值写指定地址（寄存器）范围中断
BREAK AT READ	读指定地址（寄存器）中断
BREAK IN READ	读指定地址（寄存器）范围中断
BREAK AT READ AT	读指定地址（寄存器）得到指定值中断
BREAK IN READ AT	读指定地址（寄存器）范围得到指定值中断
BREAK AT READ IN	读指定地址（寄存器）得到指定范围值中断
BREAK IN READ IN	读指定地址（寄存器）范围得到指定范围值中断
EXT IRQ BREAK	设定扩展 IRQ 中断断点
NMI BREAK	设定 NMI 中断断点
IRQ BREAK	设定 IRQ 中断断点
SCPU IRQ BREAK	设定 SCPU IRQ 中断断点
TIMER IRQ BREAK	设定 TIMER IRQ 中断断点

举例：如果欲设置这样一个断点，当对$200 到$20F 范围有写操作，且写入的值为$00 到$7F 之间的任意值时。首先我们点 New 按钮，然后在组合框选择“BREAK IN WRITE IN”，代码是 25，在 Begin Address 中输入 200，在 End Address 中输入 210，选择 Hex 模式，在 Date 靠上的一个输入框输入 00，靠下一个输入框输入 80，这样就设置好了一个断点。

Extra Window 内存查看器

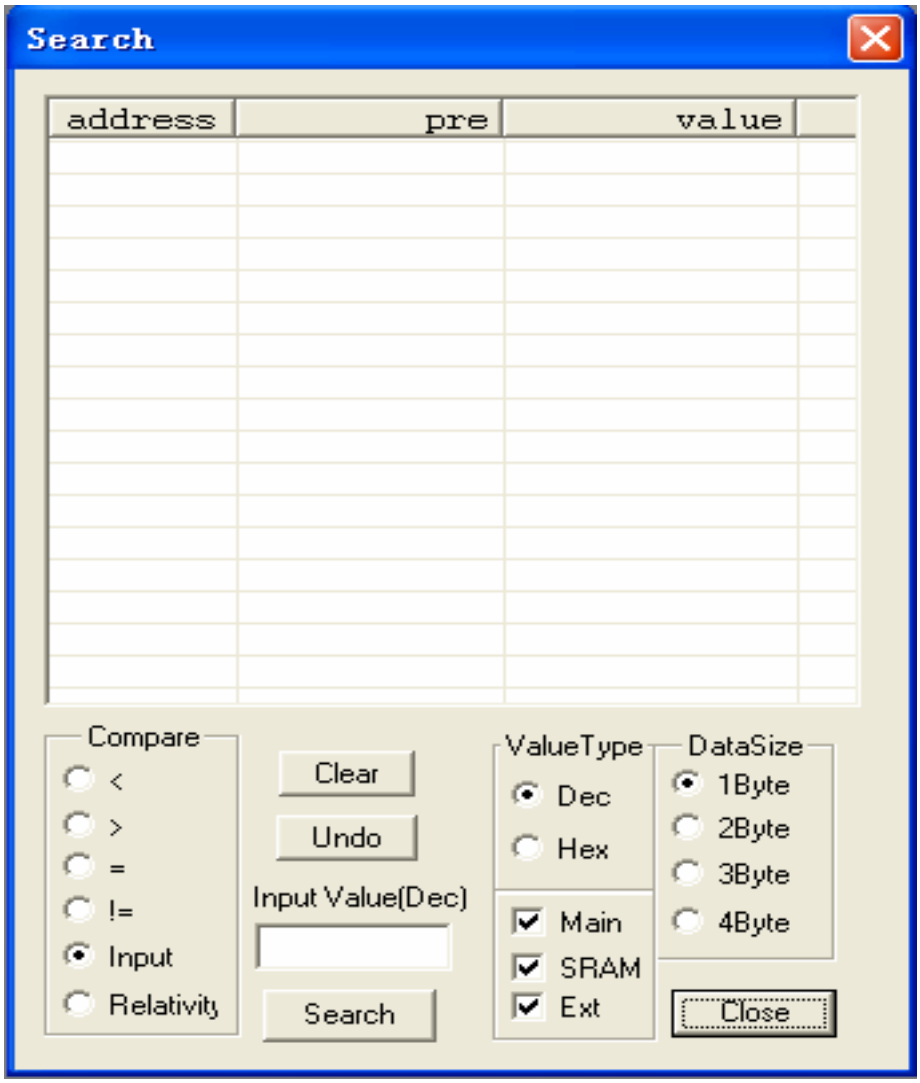
Main memory Viewer	主内存查看器
SRAM Viewer	扩充内存查看器
PROM Viewer	程序空间查看器
VRAM Viewer	PPU RAM 内存查看器
PNT Viewer	PPU PNT 内存查看器
Pallete memory Viewer	调色板内存查看器
Sprite memory Viewer	卡通内存查看器
VROM Viewer	PPU 点阵区查看


MainMemory Viewer - [0x000000]
十六进制显示区，允许直接修改原值
ASC 显示区
Color display
Goto
Find
Dump
右键菜单

右键菜单说明：

Color Display	以特殊颜色标示特殊的内容÷
Goto	跳到指定的内存偏移地址
Find	在当前内存区域查找内容
Dump	将当前内存区域内容以二进制或文本方式存到指定文件

Search 主内存搜索查找



这个功能类似于“金山游侠”或“FPE”之类的游戏工具，用来搜索找出特定内存单元的含义，比如做一些游戏的 HACK，我们需要找出游戏中“命”的数量是存在哪个内存单元，以便我们做出不死“命”的游戏版本来。