

# 68K Mac Debugger 2.0b1

## *Release Notes*

Copyright Apple Computer Inc.  
1994-1995. All rights reserved.

---

---

### New in this Release

- The 68K registers window now shows only the condition registers, A0–A7, D0–D7, and the PC.
- Grayed out items in the General Preferences window are those that apply only to the Power Mac Debugger.

---

---

### Known Outstanding Bugs

**Important:** If you find any bugs other than those listed below, please report them as soon as possible to DEV.SUPPORT (Partners & Associates) or APPLE.BUGS (all others).

**Warning:** Structs that are not typedef'd do not evaluate correctly when compiled with the Prerelease version of SCpp, linked with the Pre-release ILink and run through ILinkToSym. The symptoms of this problem are that the fields of the struct being evaluated have names of fields of another struct, and the values displayed are not what is expected.

There are two workarounds to this problem. The phenomenon does not occur if structs are typedef'd, so the problem will not occur if you typedef your structs by placing the C keyword `typedef` before the declaration of the struct:

```

struct foo
{
    long    field1;
    long    field2;
    short   field3;
};

```

should be changed to:

```

typedef struct foo
{
    long    field1;
    long    field2;
    short   field3;
} foo;

```

The other workaround is to use the state file created by ILink 2.0 instead of using the .SYM file. The problem as described does not occur when using a state file; it occurs only after the conversion of the state file into a .SYM file by ILinkToSym.

- The current version of the 68K Mac Debugger cannot read state files (i.e. .NJ files) produced by ILink 2.1. The workaround is to run ILinkToSYM 2.1 , using the `-sym 3.2` option, on the state file, producing a SYM file which the debugger can read.
- The host can lose contact with the target during memory searches taking longer than 30 seconds.
- When using MetroWerks symbolic files, you cannot edit a C-string or Str255 in any variable display window (i.e., stack crawl, evaluate window, globals window).
- No value will be displayed when evaluating a variable of type enum if the symbolic file is a state file (extension “.NJ”) or a SYM file created using ILinkToSYM.
- When doing two-machine debugging, if the target CPU crashes you must quit and relaunch the debugger host application to reacquire the target.
- If an application is targeted via break on launch and the symbolic file is a state file (extension “.NJ”), an access fault will occur when Run is chosen.
- Conditional breakpoints fail intermittently.
- If you quit the debugger while the application is stopped, and select the Kill button in the Resume/Kill dialog, the machine may hang or crash. This will only occur during single - machine debugging.

- Currently there is a problem with the 68K Mac Debugger when code segments are repeatedly loaded and unloaded. This occurs in your application when you call the `UnloadSeg` toolbox trap. Symptoms of this problem include:
  - Program execution not stopping at breakpoints
  - Program execution stopping, but not at a breakpoint
  - Getting an “Unknown breakpoint” alert.

To avoid this problem, do not call `UnloadSeg` in your application. If you’re using `MacApp`, set the global variable `gUnloadAllSegs` to `false`. Additionally, you can set all your code resources to be locked and unpurgeable to avoid this problem. We realize this is a real problem and are working on a fix. We will be posting an update as soon as a solution is found.

- Do not attempt to link an application when that application’s state file is still open in the debugger. If you do this, `ILink` will complain about a magic byte sequence in the Bento database. This is a bogus error message. The message should state that the link could not continue because the state file is open in another application.

---

## Bug Fixes

---

### Bug Fixes in v. 2.0b1 (ETO/MPW Pro #17)

- The debugger fails to evaluate PStrings and CStrings for Metrowerks-generated code .
- There are some clipboard problems in this release. If you copy something in a window, then close that window, the clipboard reverts to its original contents. If you want to copy something from a window, then paste it elsewhere, keep the window open until you’ve pasted the data.
- If you have a register or a memory location selected in an active window, you may crash when you step or run and hit a breakpoint.
- An alert saying that the stack pointer has changed will appear when a conditional breakpoint is set.
- You cannot coerce to a basic type in the evaluator window .
- When stepping, the “View As...” options do not remain set in the Stack Window for variable fields.
- The valid breakpoint line icon does not always appear on all lines as it should .

- Under certain circumstances, when the debugger is stopped it is impossible to switch from source to assembly in the Browser window .
- The debugger crashes when viewing a variable of size greater than 256 as a CString .
- The “Find Code For” menu item does not work in a browser created from a state file
- It is not possible to modify the address of a pointer to be a negative value .
- The two standard file dialogs, one for identifying the symbolic file to be opened, the other to identify a source file, have an improperly labeled button: “Drive” when it should be “Desktop.” Also, the volume icon is not enabled
- It is possible, while using a state file, to set multiple breakpoints at the same location .
- Bit field modification does not work correctly .
- Crashes occur when clicking in the file Browser on a file that has many functions .
- The breakpoint view in the 68K instruction view disappears .
- When debugging two different processes and switching between them, the PC arrow does not always show up in the targeted process .
- SYM files created by Metrowerks do not have breakpoint markers .
- Evaluation of a struct that has a field of the same type as the struct causes a crash when closing the Browser window .
- The debugger may crash when scrolling a large distance in the 68K instruction view .

---

### **Bug Fixes in v. 2.0a3 (ETO/MPW Pro #16)**

- If the host crashes while debugging (using the 68K DebugServices Nub), you must restart both the host and target (nub) machines.
- The host crashes if you attempt to undo an edit in the register window .
- You will get multiple alerts telling you that the SYM file is out of date with the source if you target an application before opening the .SYM/.NJ file .
- Crashes occur after option-dragging off a copy of the currently displayed code in the Browser window. Crashes also occur when doing “Find Code For” to bring up a new window containing the code .
- When the symbolics file and the executables dates do not match and the out of date alert is brought up, the wrong alert appeared .
- Fixed a problem that kept globals from being evaluated properly in ILink files .
- Fixed a bug where the debugger would incorrectly display a variables value as zero, and then not be modifiable, when using the Metrowerks compiler .