



## HP Forum Archive 14

[ [Return to Index](#) | [Top of Index](#) ]

### Re: HP-85B internal information?

Message #1 Posted by [Vassilis Prevelakis](#) on 26 Nov 2003, 5:28 p.m.

Valentin Albillo wrote:

> Why, Vassilis, writing machine language programs is *\*never\** trivial ! :-) At least for us, puny humans !

I stand corrected!

I meant writing 85 *assembly* is non trivial.

a) I don't think I have ever written machine language. When I was writing 6502 code, I was lucky to own a BBC micro which included an assembler in its BASIC language. When I moved to the PC (sad day), I realized that I could use the MSDOS DEBUG program to convert 386 assembler code to machine code. So I was lucky.

b) I have the 86 assembler ROM manual (and will soon post it on the [www.series80.org](http://www.series80.org) site) and after going through the code and examples, my comment was "What the f\*\*\*?" But in order to get the ROM emulator working I will have to become familiar with this architecture. I'll post my impressions when I have had the time to really go through the manual.

In the mean time, I would like to get a program in 85 assembly that reads bytes from a program ROM (like the MEM command in the assembler ROM). I need this in order to be able to see the handshake on the bus with my logic analyser.

BTW we know that 87 and 85 binary programs are incompatible. Does anybody know whether its due to the assembly process or is the assembler source code different?

\*\*vp

### Re: HP-85B internal information?

Message #2 Posted by [Valentin Albillo](#) on 27 Nov 2003, 8:03 a.m.,

*in response to message #1 by Vassilis Prevelakis*

Hi, Vassilis:

Vassilis posted:

**"BTW we know that 87 and 85 binary programs are incompatible. Does anybody know whether its due to the assembly process or is the assembler source code different?"**

As far as I can remember, there are at least two reasons for the incompatibility:

- HP-86/87's operating system (and BASIC language) allowed for up to 5 binaries loaded in memory at a time, while HP-85's did allow only one. This means, among other things, that binaries' headings are quite different, so a machine can only understand and load (unload) properly binaries which have the correct heading for its operating system.
- All binaries made very extensive use of calls to the internal ROM operating system's routines, and their symbolic names, addresses, parameters, return values, and requirements in general are sure to be very different between the HP-85 and the HP-86/87.

In the best case, you'll need to use a new table of symbolic names <-> ROM addresses. In the typical case, you'd also need to change your assembler code to cater for the extra parameters, etc. In the worst case, the internal ROM routine does not exist and there's no equivalent, so you'll have to create your own. The same applies to the HP-75C.

So much regarding incompatibilities.

Best regards from V.

### **Re: HP-85B internal information?**

*Message #3 Posted by [Vassilis Prevelakis](#) on 28 Nov 2003, 12:28 a.m.,  
in response to message #2 by Valentin Albillo*

So let me see if I get this straight. If I have a (say) HP-85 binary program that I want to run on the HP87, then all I need to do is disassemble the binary and feed the assembler source into the HP87 assembler.

Does this sound right?

Can this be done with ROMs as well?

\*\*vp

**Re: HP-85B internal information?**

Message #4 Posted by [Valentin Albillo](#) on 28 Nov 2003, 4:38 a.m.,  
in response to message #3 by [Vassilis Prevelakis](#)

Hi, Vassilis:

Vassilis posted:

**"If I have a (say) HP-85 binary program that I want to run on the HP87, then all I need to do is disassemble the binary and feed the assembler source into the HP87 assembler. Does this sound right? Can this be done with ROMs as well?"**

As mentioned in my previous post, there's the problem that all system calls (there will be a lot in most any binary program) will have different addresses and some may have different syntax, parameters, and results, or even do not exist altogether. The resulting source code would most likely not compile, but even if it did, the resulting binary would be 99.9999% sure to crash the machine on execution.

Consider, for instance, the case for handling BASIC variables (perhaps passed as arguments to the binary or used to return results). HP-85's variables must have names which are either single letters (A..Z) or letter-digit (A0, Z7), disregarding capitalization, plus you can't have string arrays DIM A\$(10,10). On the other hand, the HP-86/87 does allow longer names, with capitalization (Cost = 23), and string arrays. It's obvious that all routines having to do something with variable names (parsing, etc) or string arrays, are going to be very different among both operating systems.

Also, any routines dealing with the CRT, either alpha text or graphics, will be vastly different as well, any printing operations intended for the HP-85 built-in printer will not automatically translate to their corresponding HP-IB printers counterparts, same for mass storage operations used with the built-in tape drive which won't translate to their Mass Storage ROM equivalents, etc.

In short, I don't think that you can feed HP-85 assembler source code to the HP-86/87 Assembler ROM and expect it to compile and/or run without extensive modifications.

There's also the problem of documentation: if I remember correctly, the HP-86/87 Assembler ROM Owner's Manual was dreadful, specially from the standpoint of \*completeness\*. Only a few dozens of the myriad of potentially callable internal system's routines were even mentioned in the manual or documented, and even that very scarcely. So you were on your own trying to accomplish anything, absolutely lacking most any necessary information. I don't know if this documentation problem was ever solved, either by further materials from HP or else through User Clubs.

Best regards from V.