

---

Subject: ANN: HXA V0.12

Posted by [Anton Treuenfels](#) on Mon, 22 Aug 2005 04:11:27 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Third version of the HXA cross-assembler:

[www.home.earthlink.net/~hxa](http://www.home.earthlink.net/~hxa)

Fixed a couple of bugs. Added true (if rudimentary) segmented source code, ability to output Intel hexadecimal object files and a couple of other minor things. Couple of new proof-of-concept demos show complete 8080/85 and Z80 instruction sets implemented as macros.

- Anton Treuenfels

---

---

Subject: Re: ANN: HXA V0.12

Posted by [Anonymous](#) on Tue, 23 Aug 2005 16:58:00 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Originally posted by: josephoswaldgg@hotmail.com

Anton Treuenfels wrote:

> Third version of the HXA cross-assembler:

>

> [www.home.earthlink.net/~hxa](http://www.home.earthlink.net/~hxa)

>

> Fixed a couple of bugs. Added true (if rudimentary) segmented source

> code, ability to output Intel hexadecimal object files and a couple of

> other minor things. Couple of new proof-of-concept demos show complete

> 8080/85 and Z80 instruction sets implemented as macros.

>

> - Anton Treuenfels

Any chance this could get ported to a non-proprietary and still available AWK or Perl implementation?

What obstacles do you see in such a port?

---

---

Subject: Re: ANN: HXA V0.12

Posted by [Anton Treuenfels](#) on Wed, 24 Aug 2005 01:30:53 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

"josephoswaldgg@hotmail.com" <josephoswald@gmail.com> wrote in message

> Any chance this could get ported to a non-proprietary and still

- > available AWK or Perl implementation?
- >
- > What obstacles do you see in such a port?

If the question is "Can an assembler be written in another dialect of AWK?" the answer is yes. "The AWK Programming Language" by the original AWK creators includes a simple assembler as one of its examples (a very clear example, too!). "Amazing AWK Assembler" by Henry Spencer is still floating around out there, although it is sometimes cited as an example of how *\*not\** to do things (I've looked at it but can't really understand it).

If the question is "Can HXA be ported to another version of AWK?" the answer is...maybe. Probably not directly. TAWK has a number of very useful extensions that simply don't exist in versions such as gAWK. Among them:

1) TAWK is compiled, not interpreted (very quickly, I might add). Not in itself a deal-breaker, but any other version would likely run much more slowly.

2) TAWK's user-defined functions are much closer to the way functions work in most major languages. The kludge used for "local" variables in most AWKs (ie., defining extra arguments in the parameter list and then not initializing them when calling the function) is completely unnecessary in TAWK. I suppose most functions could be re-written in this style, perhaps along with a few global variables used exclusively for loop indices and such.

3) TAWK has true multi-dimensional arrays, rather than the concatenated indices "kludge" of other AWKs. This is tougher to get around, as HXA uses multi-dimensional arrays quite often and nearly as often is interested in the value of the second or later index independently of the previous indices. A port might be better off trying to implement "equivlant functionality" rather than attempting to duplicate the TAWK functionality directly.

4) TAWK array indices are by default sorted in increasing order (a property this version of HXA relies on even more heavily than previous versions), while other AWKs make no order guarantee (the current maintainer of gAWK has been known to wonder how Pat Thompson did it). Many times when HXA uses this property to visit array elements in order the indices are numeric, and so these could often be changed to FOR loops. But sometimes even if they are numeric there are large gaps (on the order of  $2^{20}$  or more) between consecutive index values, so it could take a while for this kind of loop to complete.

In sum, to port to another version of AWK it might be easiest to try to figure out what the purpose of anything HXA does is, and if it uses some special feature of TAWK to accomplish it, try to figure out how to provide

equivalent functionality in that version of AWK.

Me, when I think about porting to a different language I usually consider what I could do with C and a good regular expression library :). I'd also probably make a lot of use of structures, which don't exist in TAWK (I fake them sometimes with those multi-dimensional arrays). I'd very much miss the automatic memory management of all the AWK variants, though.

I probably wouldn't make a lot of changes to the overall design of HXA. I'd do the "equivalent functionality" thing. With any luck it's modularized heavily enough that that shouldn't be too exceeding difficult (just match the function inputs and outputs and internally do whatever).

- Anton Treuenfels

---

Subject: Re: ANN: HXA V0.12

Posted by [Andy McFadden](#) on Wed, 24 Aug 2005 06:31:07 GMT

[View Forum Message](#) <> [Reply to Message](#)

In comp.sys.apple2.programmer Anton Treuenfels <atreuenfels@earthlink.net> wrote:

> Me, when I think about porting to a different language I usually consider  
> what I could do with C and a good regular expression library :). I'd also  
> probably make a lot of use of structures, which don't exist in TAWK (I fake  
> them sometimes with those multi-dimensional arrays).  
> I'd very much miss the automatic memory management of all the AWK variants,  
> though.

Have you looked at Python?

--

Send mail to [fadden@fadden.com](mailto:fadden@fadden.com) (Andy McFadden) - <http://www.fadden.com/>

CD-Recordable FAQ - <http://www.cdrfaq.org/>

CiderPress Apple II archive utility for Windows - <http://www.faddensoft.com/>

Fight Internet Spam - <http://spam.abuse.net/spam/> & <http://spamcop.net/>

---

Subject: Re: ANN: HXA V0.12

Posted by [Anton Treuenfels](#) on Thu, 25 Aug 2005 03:53:41 GMT

[View Forum Message](#) <> [Reply to Message](#)

"Andy McFadden" <[fadden@fadden.com](mailto:fadden@fadden.com)> wrote in message  
news:LuUOe.10377\$p%3.39738@typhoon.sonic.net...

> In comp.sys.apple2.programmer Anton Treuenfels <atreuenfels@earthlink.net>  
wrote:

>> Me, when I think about porting to a different language I usually consider  
>> what I could do with C and a good regular expression library :). I'd also  
>> probably make a lot of use of structures, which don't exist in TAWK (I fake  
>> them sometimes with those multi-dimensional arrays).  
>> I'd very much miss the automatic memory management of all the AWK variants,  
>> though.  
>  
> Have you looked at Python?

Can't say that I have. I don't know much about Python or Perl or Ruby. When I came across an old (really old) version of gAWK I thought "Here is something I could really use! - if it wasn't so broken!".

When I found TAWK I thought "This has everything I thought was missing!". So now if the problem looks like a text file the solution looks like TAWK to me :) Well worth every penny I paid for it - I just wish I'd upgraded when I had the chance.

But let's not get ahead of ourselves. I don't have any immediate plans myself to convert HXA to another language. If anyone wants to do that themselves, well, the sources are there and it's GPL'd - have at it!.

If someone is just interested in adding a new processor and is willing to write the single replacement module that would be needed (plus the necessary test files showing that it handles correct/incorrect input and the documentation, of course), I'd be willing to compile it and add it to the download.

- Anton Treuenfels

---