

Re: Just what makes an architecture "C Friendly"?

Re: Just what makes an architecture "C Friendly"?

Source: <http://coding.derkeiler.com/Archive/General/comp.arch.embedded/2006-06/msg00226.html>

- *From:* "Bryan Hackney" <bbhack@xxxxxxxxx>
 - *Date:* 4 Jun 2006 20:54:13 -0700
-

Bryan Hackney wrote:

Steve at fivetrees wrote:

[...]

And even if you are just generating a small bit of embedded code for that environment by running Linux and GCC quite happily on a PC you probably were not thinking of running GCC out of a dozen words of target ROM.

Huh?

Unless I'm misunderstanding something, we're not talking about the platform that the C compiler runs on. That's of no interest (to me). We're talking about the target it compiles for.

As for "it is becoming an option more often to take the library and file system and OS from development platform to the embedded target with the minimal effort" – if you're talking about embedded PCs, fine. Again that's of no interest to me, and I don't see how it's relevant to this thread.

I've been using a 32 bit processor with MMU running Linux as a fairly deeply embedded processor for years. The source is 100% compilable with the PC (development system) as a target, and about 80% functionally portable between the development system and the actual embedded target. I think that's what he means.

And yes, the development system for the processor was the starting point. Axis provides a stable starting point, royalty free tools and

Re: Just what makes an architecture "C Friendly"?

Linux and GCC ports.

The target filesystem and minimal user space (mostly BusyBox) on the embedded target would not be suitable for a PC, and I don't think that was an argument.

Oops. It's late. I used the term "development system" to mean different things. The first instance should have read "host system" and the second instance should have read "development board and associated s/w".