

# Re: Microcontroller Project

---

*Source:* <http://sci.tech-archive.net/Archive/sci.electronics.design/2006-03/msg03912.html>

---

- *From:* "David L. Jones" <[altzone@xxxxxxxx](mailto:altzone@xxxxxxxx)>
  - *Date:* 27 Mar 2006 14:33:03 -0800
- 

Keith wrote:

In article <1143494228.117311.162780@xx>, altzone@xxxxxxxx says...

Keith wrote:

In article <1143447041.276029.172680@xx>, altzone@xxxxxxxx says...

Keith wrote:

On Sun, 26 Mar 2006  
17:28:03 -0800, David L.  
Jones wrote:

akshaychander@xxxxxxxx  
wrote:

Sorry  
for  
the  
late  
reply.

Yes,  
I  
have  
to  
do  
the  
project  
in  
8051.  
I  
have  
studied

Re: Microcontroller Project

the  
theory  
of  
8051,  
but  
have  
very  
little  
practical  
experience.

As  
far  
as  
languages  
are  
concerned,  
there  
is  
no  
restriction.  
I  
am  
planning  
on  
using  
C.

In that case  
a good C  
compiler  
will take  
care of most  
of the low  
level  
8051 stuff  
for you.

I disagree! A C compiler  
will just mask the nonsense  
underneath. It  
\*will\* come back to bite  
you (think "stack").

Ok, I'm not an 8051 guy, so I'll take your  
word there is a potential  
issue here.

## Re: Microcontroller Project

I looked into various compilers when I did a major (~25kloc) 8051 project a decade ago. The only one that didn't mask the limitations (and even advantages) of the 8051 was PL/M51.

I'd then suggest you take a look at some more modern compilers perhaps. Countless people use C on the 8051, I'm sure it's not that bad.

The point is that one is *\*necessarily\** doing bit-banging on the hardware (the whole point of a uC). C doesn't give much advantage and a *\_lot\_* of headaches.

*\*snip\**

I see no advantage in using C. Portability isn't an issue since hardware is the name of the game with uCs. Any abstraction will lead to resource conflicts. No, actually, I haven't been bitten by stack issues because I've used assembler. I don't let the compiler assign *\*anything\**; not even what byte the bits go in.

*\*snip\**

I disagree. C has no advantage over a half-decent macro assembler and a MAKE utility. Abstraction has a *\*lot\** of disadvantages when you're severely hardware constrained. Sure, people use C on 8051s, but I suspect for many C is the only tool in their shed.

Ok, it looks like this one could never end.

You are pro-assembler, I'm pro-C.

I used to do all my work in assembler unless I started using C and then never looked back. I still do some inline assembler where it's called for, but that's it.

There are *\*massive\** advantages to using C (or another high level language) on a micro – speed of development, platform portability (which is a BIG requirement for some people), ease of peer review, ease of maintenance, easier code-reuse, easier model migration, better code visibility to name a few. This is why C compilers are immensely popular these days.

Yes, there are advantages to assembler as well (and disadvantages to C), that's why I like many others use the best of both worlds. Making a

Re: Microcontroller Project

blanket statement like there are \*no\* advantages to using C is just plain rubbish.

Dave :)

.