



*>on a single chip". So how is it not the case?*

I was commenting the whole of his statement, not the on-board ROM bit.  
See the rest of the paragraph for full enlightenment.

*>> The word sizes are  
>> the vanilla ones, so that adding external ROM or RAM can be done using  
>> cheap, off the shelf, components (no point in having a cheap  
>> microcontroller if it needs expensive external chips).  
>  
> You may not be familiar with the 8051, but it's addressing modes were  
>quite unusual.*

Having programmed in 8051 assembly, I am reasonably familiar with it and its addressing modes. I wouldn't say that a word size of 8 bits is particularly odd...

*>(If you only programmed it in C, you might not have realized  
>just how odd it is.) As an example, how many bits do you think you need for  
>a general purpose pointer in an 8051? (Keep in mind it has code and data in  
>distinct address spaces.) And what about the external MOV instructions that  
>take an 8-bit address? Do you know where the other 8 bits of the address  
>comes from? (It's truly bizarre.)*

I'm failing to see the connection between these questions and the size of the 8051 word. In your humble opinion, why is it called an 8-bit microcontroller?

*>>>Low  
>>>power consumption and extremely low cost-per-chip often counts for more  
>>>than clock speed.  
>  
>> Low power consumption is often not an issue (no point in using a < 1 mW  
>> controller in a microwave oven)  
>  
> His claim was "often". And he didn't say it was the most important  
>thing, he said it counted more than clock speed. You aren't seriously going  
>to argue that clock speed is important in a microwave oven.*

If it's not important, then try to control one using a 1 Hz clock...  
But my point was that low power is not that often an overriding concern, while low cost per chip \*is\*.

If you'd read my sentences and paragraphs in their entirety instead of interrupting the process at arbitrary points, you might even be able to see my points by yourself...

*> So he said A is often more important than B and you came up with a case  
>where neither is particularly important. How do you think this refutes what  
>he said?  
>*

comp.unix.programmer: Re: [OT] Re: writing a dailer in c for a 8051 based system

>> *but low cost per chip is an overriding*  
>> *concern, otherwise the designer would opt for a traditional microprocessor*  
>> *+ ROM + RAM + I/O interfaces approach, which is often more comfortable to*  
>> *program on (the typical CPU in a microcontroller is less capable than an*  
>> *usual microprocessor, the program and the data often reside in different*  
>> *address spaces, and, for machine code compactness reasons, the program*  
>> *memory is often paged).*  
>  
> *Well that's the first thing you've said that's correct.*

Everything I said in my previous post was factually correct. Learn to read and you may eventually realise it.

Dan

--  
Dan Pop  
DESY Zeuthen, RZ group  
Email: Dan.Pop@ifh.de