

Re: Computer programmers' habits in electronics

Source: <http://sci.tech-archive.net/Archive/sci.electronics.design/2005-12/msg03685.html>

- *From:* "Frithiof Andreas Jensen" <frithiof.jensen@xx>
 - *Date:* Wed, 21 Dec 2005 12:24:57 +0100
-

"Tim Wescott" <tim@xxxxxxxxxxxxxxxxxxx> wrote in message
news:DISdnc8M46OsyjXenZ2dnUVZ_sGdnZ2d@xxxxxxxxxxxxxxxxxxx

>> As a circuit designer I've always liked "block" diagramming of a
>> system before I begin, so I don't create redundant (or useless)
>> circuit chunks.
>>
>> So I find it hard to fathom how you can write software without some
>> similar organizing scheme.

That is indeed how it is done in real life – we use large whiteboards ...
and coloured pens ;-)

>> I once took a course at the community college in Pascal (that will
>> date me :-)
>>
>> The instructor insisted on using "outlining" which, to me, was trying
>> to write raw code without any sense of direction.
>>
>> When I kept using block diagramming she got pissed at me and started
>> giving me F's on the assignments, in spite of the resulting code being
>> quite compact.

When arguing with academics you must always call upon Higher Beings – err.
references – such as Jourdon f.ex. who has written USD 150 book about how to
draw block diagrams and state machines when designing software. Just arguing
that something works is plebeian. ;-)

> Interestingly enough folks have been trying to find structured ways to
> do this with software for decades, with moderate to good success. The
> current methodology is called "UML", for "Universal Modeling Language".
> While the "Universal" is more of a comment on the narrow-mindedness of
> the software engineering world than on the applicability of UML for
> anything but software, it is a very handy conceptual tool.
>
> It provides, among other things, a very powerful block diagramming

Re: Computer programmers' habits in electronics

> language tailored to designing object-oriented software applications.

Bleth, Ack, Blaaaarghhhh <cough, cough>

UML is yet-another-attempt at turning a task that is hard to understand and execute into coloured Lego bricks that can be drawn by managers and handed over to disposable code-monkeys in India!!

The UML is, in most cases, something that is done by a reverse engineering tool *after* the design is implemented because the *customer* wants it, not because the *designers* needed it (if they did need it, there was probably too many layers of redirection and obfuscation* in the way of The Job anyway ;-)

Discalimer:

Of course Rose Realtime can be used to advantage if one *must* use a large amount of C++/Java to avoid too much exposure to the underlying language.

UML sort of works, but like most computer based tools, the notation and the insistence on all the details up front actually gets in the way of *thinking* which is the most important task.

*Abstraction – O.O. term for inserting soo many interface layers and glue code that one forgets what the task was ;-)

>

> So you were right all along, and had a dippy and vindictive instructor.

>

Yep!!

> --

>

> Tim Wescott

> Wescott Design Services

> <http://www.wescottdesign.com>

.

• **References:**

◆ **Re: Computer programmers' habits in electronics**

◇ From: Jim Thompson

◆ **Re: Computer programmers' habits in electronics**

◇ From: Tim Wescott

• Prev by Date: **Re: Computer programmers' habits in electronics**

Re: Computer programmers' habits in electronics

- Next by Date: ***Re: PTC (positive Thermo coefficient) device temp ratings.***
- Previous by thread: ***Re: Computer programmers' habits in electronics***
- Next by thread: ***Re: Computer programmers' habits in electronics***
- Index(es):
 - ◆ ***Date***
 - ◆ ***Thread***