

## Re: what's a callback?

**Source:** <http://sci.tech-archive.net/Archive/sci.electronics.design/2004-12/4036.html>

---

**From:** John Larkin ([john\\_at\\_spamless.usa](mailto:john_at_spamless.usa))

**Date:** 12/20/04

Date: Mon, 20 Dec 2004 08:17:30 -0800

On Mon, 20 Dec 2004 10:12:58 GMT, "Anthony Fremont"  
<[spam@anywhere.com](mailto:spam@anywhere.com)> wrote:

>John Larkin wrote:

>> On Sun, 19 Dec 2004 21:50:21 GMT, "Anthony Fremont"

>> <[spam@anywhere.com](mailto:spam@anywhere.com)> wrote:

>>

>>>

>>> "Fred Bloggs" <[nospam@nospam.com](mailto:nospam@nospam.com)> wrote in message

>>>

>>>> No- that is called "setting a flag to remind you to finish some  
>>>> chores" and is an example of \*re-entrant code\*:-) Why in the world  
>>>> would that be called "callback" unless you collect useless jargon?

>>>

>>> That's certainly the most unique definition of re-entrancy that I've  
>>> seen.

>>>

>>> When I think of re-entrant code, I think of code that has no local  
>>> variable storage associated to it. I also think of code that can  
>>> call itself recursively or be executed in several threads across  
>>> multiple processors concurrently with only one copy in memory.

>>

>>

>> AKA "pure" code.

>

>Well.....I guess if it were to be really pure code, all addresses would  
>have to be relative to the instruction counter. ;- ) We used to call  
>this floatable code as it could be just plunked into memory anywhere and  
>executed.

My (perhaps non-professional-programmer) definition of that is  
"relocatable" or "PIC" (position independent) code.

"Pure" code is code that has no associated statics and if of course  
not self-modifying, so that it can be executed by multiple threads  
without hassle.

John

Re: what's a callback?