

# The Embedded Muse 149

Editor: Jack Ganssle ([jack@ganssle.com](mailto:jack@ganssle.com))

September 21, 2007

You may redistribute this newsletter for noncommercial purposes. For commercial use contact [info@ganssle.com](mailto:info@ganssle.com).

EDITOR: Jack Ganssle, [jack@ganssle.com](mailto:jack@ganssle.com)

## CONTENTS:

- Editor's Notes
- Agile 2007
- More on Test Driven Development
- Multitasking
- Jobs!
- Joke for the Week
- About The Embedded Muse

## **Editor's Notes**

Did you know it IS possible to create accurate schedules? Or that most projects consume 50% of the development time in debug and test, and that it's not hard to slash that number drastically? Or that we know how to manage the quantitative relationship between complexity and bugs? Learn this and far more at my Better Firmware Faster class, presented at your facility. See <http://www.ganssle.com/classes.htm> .

Are you in the Seattle or Minneapolis area? I'll present the Better Firmware Faster class in Minneapolis December 5, and Seattle December 7. Registration and other info here: <http://www.ganssle.com/classes.htm> . You'll earn 0.7 Continuing Education Units, learn a lot, and hopefully have a bit of fun, too.

I'm just back from the Embedded Systems Conference in Boston. The topic of code coverage came up in one class. Some standards, such as various levels of DO-178B, require it. Do you use code coverage tools like Bullseye or Semantic Designs? If so, are you pleased with the results? Drop me an email and I'll share the results in a future Muse.

\*\*\*\*\*

This issue of the Muse is sponsored by Gary Stringham:

Have you been stuck trying to integrate hardware and firmware?

*Copyright 2003 by The Ganssle Group. All Rights Reserved. You may distribute this for non-commercial purposes. Contact us at [info@ganssle.com](mailto:info@ganssle.com) for more information.*

Is your firmware tasked with fixing hardware defects?  
Have you felt blind trying to see what your chip is doing?  
Click here for your free copy of ten test and debug hooks that  
reduce troubleshooting time and expense in your embedded system.  
[www.garystingham.com/museoffer.shtml](http://www.garystingham.com/museoffer.shtml)

\*\*\*\*\*

## **Agile 2007**

I attended the Agile 2007 conference in Washington DC last month. It's a small but highly-focused convention that offered classes mostly targeted to PC/IT developers, though a few sessions were specifically aimed at the embedded crowd. One comment that resonated was this: There's no partial credit. Done means done. Done doesn't mean it's partly tested. It doesn't mean the documents aren't finished.

The best developer I ever had didn't get "done." He'd tell me the code was done when he had it all figured out in his mind. This person was so good that he was like Mozart taking nearly-perfect dictation once the music/code existed in his head. Though this was a thing of beauty to watch, the code wasn't done and served no useful purpose till it was compiled, linked, integrated and tested. Done. "Done," used improperly, is just another ugly four letter word.

I also enjoyed a series of talks about open office environments. The idea is to tear down walls to increase communication. Some of the presenters had removed all of the offices and cubicles, grouping everyone into one bull pen. Others created a mix of open spaces and cubes, sometimes with a few private areas available for use on an as-needed basis.

Great communication is indeed a hallmark of any sort of effective behavior, whether in marriage or the engineering lab. But privacy has its role, too. DeMarco and Lister showed long ago that interruptions are productivity killers. The human brain can juggle five to nine things at once. When you've created that beautifully-intricate function in your head – when it's designed but not "done" – and someone interrupts, wondering where the coffee filters are stored, the entire structure collapses. You've got to rebuild it at some cost in efficiency.

To my knowledge open office spaces haven't been studied very much. One experiment I monitored at a government agency saw a 70% productivity drop, though the spaces offered no private retreats for those doing complicated thinking.

I'm interested in any data or experiences people have had in open office spaces, and will pass along comments.

*Copyright 2003 by The Ganssle Group. All Rights Reserved. You may distribute this for non-commercial purposes. Contact us at [info@ganssle.com](mailto:info@ganssle.com) for more information.*

## More on Test Driven Development

The May/June 2007 issue of IEEE Computer is devoted to TDD, and has a couple of worthwhile articles. One that scales directly to firmware developers describes using TDD to build a PID Controller.

Another, by Robert Martin of Object Mentor starts off: “Professional software developers ship clean, flexible code that works – on time. It seems to me that this statement is the minimum standard for professional behavior for software development. Yet, in my travels as a software consultant, I’ve met many software developers who don’t set the bar this high and instead ship late, buggy, messy, and bloated code.” He then goes on to slightly contradict himself by saying that the developers \*want\* those qualities, but just can’t believe these goals are achievable. That mirrors my experiences traveling around the firmware industry, though I’d add the caveat that sometimes unmitigated pressures from on-high make developers despair.

The guest editors (Ron Jeffries and Grigori Melnik) include a table summarizing 19 experiments in using TDD. Four of those did not measure productivity effects. 11 had an increase in effort – a decrease in productivity – ranging from 5 to 100%. Sounds like a loser, except that quality often improved dramatically. Seven of the studies showed no quality effect, one resulted in worse code (note that case was one that didn’t measure productivity), and 11 yielded quality gains ranging from 16 to an impressive 267%.

For the studies that measured both productivity and quality, the results were:

Decrease in productivity	Improvement in quality
19	40
25-35	62
15	76
5 to 6	30
17	36
16	18
60 to 100	38 to 267
22	0
90	45
16	16

Sometimes studies cloud rather than clarify issues, but the overall picture, painted impressionistically by these widely-varying numbers, says that TDD users deliver later but with much better code. Why? One could speculate that they are delivering more code, embodied in a complete test suite.

*Copyright 2003 by The Ganssle Group. All Rights Reserved. You may distribute this for non-commercial purposes. Contact us at [info@ganssle.com](mailto:info@ganssle.com) for more information.*

## **Multitasking**

No, by multitasking I'm not talking about implementing an RTOS. This is about juggling multiple activities more or less at the same time. You work on project A, move to B, then C, and back to A.

And give up huge amounts of productivity.

In "The New Brain: How the Modern Age Is Rewiring Your Mind," author and neurologist Richard Restak claims that on our brain works most efficiently when working on a single task for a sustained – not intermittent – period of time.

In Quality Software Management Gerald Weinberg claims that multitasking between three projects incurs a 40% context switching overhead. So in that 40 hour workweek, if switching frequently between three activities, you're putting in 24 useful hours of work. To look at this another way, over 800 hours a year – 20 work weeks – go to context switching.

Sometimes I envy the younger generation as they somehow manage to juggle a half-dozen electronic connections at the same time. Their brains must be more attuned to simultaneously managing many activities, since for their entire lives they've been immersed in the electronic maelstrom.

Or maybe not. Research at the Institute for the Future of the Mind used groups of people between 18 and 21, and between 35 to 39, to study these effects. All were required to do a simple translation scheme. The younger folks were 10% more productive, when not interrupted. But a short IM, cell call, or text message eliminated the difference. The oldsters and young 'uns did equally poorly.

## **Jobs!**

Let me know if you're hiring firmware or embedded designers. No recruiters please, and I reserve the right to edit ads to fit the format and intents of this newsletter.

Jeff Dombach ([jeff@jldsystems.com](mailto:jeff@jldsystems.com)) in Lancaster, PA is looking for an embedded/electronic engineer. Will work with embedded software and hardware. Implementing interfaces, building, testing prototypes/small runs. Various applications in the measurement, video, audio and test equipment industries.

Key Job Responsibilities:

*Copyright 2003 by The Ganssle Group. All Rights Reserved. You may distribute this for non-commercial purposes. Contact us at [info@ganssle.com](mailto:info@ganssle.com) for more information.*

1. Embedded implementation, C and assembly language.
2. Embedded development tools.
3. Software design and development.
4. Hardware design and development.
5. Schematic Capture and Board Layout.
6. Real time software and control.
7. Interrupt Drivers.
8. Serial/Ethernet Interfaces.

See <http://www.jldsystems.com> .

Vorne Industries is seeking a Senior Hardware Engineer and a C++ Software Engineer to help us develop next-generation products that clearly stand above the crowd. We are looking strictly for the best engineers that are energized by accomplishment and who are expert at their craft. As we are a small (but growing) team, a diverse and broad-ranging skill set is absolutely essential.

Here's the essence as a Hardware Engineer, you must truly enjoy creating embedded products or as a Software Engineer, you must truly enjoy creating exceptional software and for both positions, you must be passionate about excellence. When it comes to quality we have two standards great and insanely great!

Our positions will challenge you, encourage professional growth, and add value to your career. Your major responsibilities will include microprocessor-based high-speed digital design, analog design, programmable logic design, and system architecture.

Hardware Engineer:

A broad and versatile skill set with significant experience in digital (100MHz+ bus speeds), analog (e.g. switching converters), and programmable logic design.

Outstanding aptitude and proven achievements in system design and architecture.

Excellent writing skills and a talent for getting your ideas across on paper.

An intense drive to continually learn and relentlessly improve.

A proclivity and talent for bringing order (a coherent and well-conceived design) from chaos (product feature ideas and requests).

Enthusiasm, enthusiasm, enthusiasm! We thrive on a positive, success-oriented culture.

Exceptional attention to detail. Thoroughness and accuracy are necessary in every facet of this job.

A BSEE or MSEE from an accredited college or university and at least three years of related experience.

A willingness to demonstrate all of the above we believe that a thorough application

*Copyright 2003 by The Ganssle Group. All Rights Reserved. You may distribute this for non-commercial purposes. Contact us at [info@ganssle.com](mailto:info@ganssle.com) for more information.*

process will help both of us make a good decision.

Software Engineer:

Expert C++ skills and a thorough understanding of object-oriented programming, analysis, and design.

An outstanding aptitude for algorithms, architecture, and clean, efficient coding.

Excellent writing skills and a talent for getting your ideas across on paper.

An intense drive to continually learn and relentlessly improve.

A proclivity and talent for bringing order (a coherent and well-architected design) from chaos (product feature ideas and requests).

Enthusiasm, enthusiasm, enthusiasm! We thrive on a positive, success-oriented culture.

Exceptional attention to detail. Thoroughness and accuracy are necessary in every facet of this job.

A Bachelors Degree in Computer Science or Computer Engineering from an accredited college or university.

A willingness to demonstrate all of the above we believe that a thorough application process will help both of us make a good decision.

Visit [www.vorne.com/careers](http://www.vorne.com/careers) for more information. If you like what you see, complete our 5-minute on-line profile and submit your resume and salary history to [hr@vorne.com](mailto:hr@vorne.com). We look forward to hearing from you!

DEKA is filled with over 250 people who are driven to be among the very best in their respective fields. Consequently, the atmosphere is rewarding and incredibly intense. As a DEKA employee, you will have the opportunity to work on projects that have a positive, enduring impact on millions of people worldwide. These programs range in scope from medical, energy, consumer, public health and defense.

DEKA is located in the historic millyard district of Manchester, NH a vibrant, picturesque New England city within an hour drive of Boston, the White Mountains and the seacoast.

Embedded Software Engineer

This position involves real-time embedded software development for FDA-regulated medical devices. You'll develop real-time embedded software for FDA-regulated medical equipment. Develop and document software requirements and design. Design and implement systems performing real-time electro-mechanical control. Develop and execute testing to prove that the designs meet the requirements.

Required Experience

2-7 years of C/C++ development

*Copyright 2003 by The Ganssle Group. All Rights Reserved. You may distribute this for non-commercial purposes. Contact us at [info@ganssle.com](mailto:info@ganssle.com) for more information.*

BSEE/CS or equivalent

Familiarity with hardware-level interfaces (e.g. serial ports, A/D converters)

Preferred Experience

Implementation for real-time embedded systems

OOAD using UML or similar.

Development under an RTOS (e.g. Windows CE, Nucleus, VxWorks, QNX, MicroC/OS-II)

Microcontroller/DSP development (e.g. Microchip, 8051, TMS320)

Interprocessor/chip Communications (e.g. RS232, I2C)

User Interface implementation

Motor Control

Assembly coding

Experience with FDA/FAA or similarly regulated systems

Please send resume to: [careers@dekaresearch.com](mailto:careers@dekaresearch.com), subject "ISW"

Control System Engineer

Engineer will have a broad range of responsibilities developing control system for new and medical and non-medical devices. Work may include developing algorithms, writing embedded code, specifying sensors and investigating plant dynamics. Work is very hands-on and controls engineers frequently act as "test pilots" when developing new controllers. Must have experience in the development of electromechanical control systems. Knowledge of Matlab/C++ is required.

Education

MS in Mechanical or Electrical Engineering with emphasis on control system development, or a BSME or BSEE and equivalent work experience.

Please send resume to: [careers@dekaresearch.com](mailto:careers@dekaresearch.com), subject "Controls"

MadgeTech, Inc., a leading manufacturer of dataloggers, is seeking an Electrical Engineer for the R&D department. The qualified candidate will have experience with microcontroller programming, design experience with a variety of sensors, low-power analog and digital design experience, PCB layout experience and knowledgeable of various battery technologies. Desirable skills include Microchip programming and Windows programming. Madgetech is located in Warner, NH just off Route 89. Submit your resume and cover letter to: [jobpostings@madgetech.com](mailto:jobpostings@madgetech.com) or consult our web site at [www.MadgeTech.com](http://www.MadgeTech.com) for more information.

## **Joke for the Week**

.

*Copyright 2003 by The Ganssle Group. All Rights Reserved. You may distribute this for non-commercial purposes. Contact us at [info@ganssle.com](mailto:info@ganssle.com) for more information.*

Andy Kunz sent this link to a site that compares languages to cars. It's pretty funny:  
<http://www.cs.caltech.edu/~mvanier/hacking/rants/cars.html> .

## **About The Embedded Muse**

The Embedded Muse is an occasional newsletter sent via email by Jack Ganssle. Send complaints, comments, and contributions to him at [jack@ganssle.com](mailto:jack@ganssle.com).

To subscribe, send a message to [majordomo@ganssle.com](mailto:majordomo@ganssle.com), with the words "subscribe embedded *your-email-address*" in the body. To unsubscribe, change the message to "unsubscribe embedded *your-email-address*". ". BUT - please use YOUR email address in place of "email-address".

The Embedded Muse is supported by The Ganssle Group, whose mission is to help embedded folks get better products to market faster. We offer seminars at your site offering hard-hitting ideas - and action - you can take now to ***improve firmware quality and decrease development time***. Contact us at [info@ganssle.com](mailto:info@ganssle.com) for more information.

*Copyright 2003 by The Ganssle Group. All Rights Reserved. You may distribute this for non-commercial purposes. Contact us at [info@ganssle.com](mailto:info@ganssle.com) for more information.*

**The Ganssle Group, [www.ganssle.com](http://www.ganssle.com)**