The Embedded Muse 150

Editor: Jack Ganssle   (jack@ganssle.com)   October 11, 2007

You may redistribute this newsletter for noncommercial purposes. For commercial use contact info@ganssle.com.

EDITOR: Jack Ganssle, jack@ganssle.com

CONTENTS:
- Editor's Notes
- Book Review
- Tools and Tips
- Open Spaces
- 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.

My colleague Bill Gatliff has just announced some new dates for his highly-regarded Embedded Linux Jumpstart class. Bill is a professional Linux developer and tenured instructor at the Embedded Systems Conference, where his classes consistently rank in the top 5%. He's arguably the most sought-after embedded Linux instructor around. See his website at http://billgatliff.com for more information.

The Ganssle Group, www.ganssle.com

Book Review

Only rarely do I review a non-technical book, but many of us are managers working hard to find ways to make our teams more effective. Management is an odd sort of job; most technical leaders are folks who have been promoted from the ranks of the engineers, usually with no real managerial experience and even less training. It's important that we get the skills needed, at the very least by reading broadly.

I found "First, Break All The Rules" by Marcus Buckingham and Curt Coffman to be a fascinating, important and easy-to-read book that every manager should study. The Gallup organization interviewed 80,000 managers and something like a million workers to find the essential qualities of leadership. This work distills the wisdom found in that study.

They found the best employees can positively answer twelve essential questions. The most important half-dozen are:
1) Do I know what is expected from me at work?
2) Do I have the materials and equipment I need to do my work right?
3) Do I have the opportunity to do what I do best every day?
4) In the last seven days, have I received recognition or praise for good work?
5) Does my supervisor, or someone at work, seem to care about me as a person?
6) Is there someone at work who encourages my development?

The most important four roles of the manager are:
1) Select people
2) Set expectations
3) Motivate the people
4) Develop the people

None of this is surprising or rocket science, but few managers make a concerted effort to practice these roles, and to ensure employees can answer all twelve questions affirmatively.

The strength of this book is it isn't infused with slogans; it's all based on real research which the authors translate into useful strategies for working managers. Don't look for cookbook recipes because there are none; every person is different. Do expect general approaches that we *must* practice to build world-class teams.

The book is a bit repetitious, and would be better if it were only two-thirds of its 271 pages. But it's a very fast and interesting read. Highly recommended. If you're not a manager, but suffering under someone who has poor leadership skills, slip a copy of the book under your boss's door.

Tools and Tips

Jim Donelson recommends the following sites for code for the Blowfish block cipher. He says they look like very promising methods for embedded projects:
- http://www.schneier.com/blowfish.html
- http://www.schneier.com/blowfish-download.html

Open Spaces

Jim Scandale has some experiences to relate about open spaces: "I work in an open lab at a university. We have anywhere from none to six people working simultaneously at any one time. These are almost all senior-level graduate students in Computer Science and Industrial Engineering. I have always been an advocate of the silent workspace since I am easily distracted but this experience (going on six years now) has been a real eye-opener. I have gotten some very complicated work done and except for the occasional lapse into "social hour", we have all been able to work.

"The key I believe is to keep the room silent. This is not an enforced rule and in fact is not a rule at all. But the students and workers here are all focused on what they are doing instead of a paycheck and that makes the difference. For a while this lab was also a meeting and conference room and that made things, literally impossible. But since we moved the conference room next door, the lab is an ideal workspace. I don't know how this would work in a corporate office or even an engineering one but I believe that the key to making it work is the quiet."

Sharon Foster wrote: "Sometimes partitions only give the illusion of privacy and productivity. The noisiest, most disruptive environment I ever worked in had 6 foot high partitions. Since no one could see them, people felt free to talk in a normal tone or louder, whether they were on the phone or talking to a visitor. Some people used their speaker-phone feature to play back messages. Others carried on conversations with people three cubicles away. I hated it, and it was one of many reasons that I left engineering. In one of the software engineering books that I sold away,--possibly one of the Steve McConnell books?--there are statistics that show that interruptions take a lot more time than responding to the actual question or input. There's also the time it takes to context switch from what you were doing to what the person who interrupted you wants to talk about. And there is the time it takes to context switch back to what you were doing. That can take as much as 20 minutes. Longer if, as in your example, you were working out a complex piece of code in your head and hadn't written anything down yet."

A reader who wishes to remain anonymous wrote: "In the latest issue of the Muse, you asked about people who have experience in working in an open office. Well, I have about 5.5 years worth of this "experience" in an open office while working for a large Japanese automotive OEM in North America. The original concept, of course, is to improve open communication. In Japan, this seems to work reasonably well. However, over there, children are taught from a very early age to get consensus on everything!!! For those of us who have not been trained like this, this takes some getting used to. Just to clarify, when I say I work in an open office, I mean it's completely open. There are no cubicles of any sort, no 1/2 wall dividers... nothing. Just a large open area filled with desks. When it does come time to get a group of people together to get consensus on something quickly, it makes it very easy to stand up at your desk, take one sweeping view of the office and decide if everyone that you need for the meeting is present. So, it does create communication, but to the opposite extreme where there is way to much communication that you don't need or want to know about...you know...they guy across from you arguing with his wife, the guys beside you discussing the latest changes and testing that needs to get done, the engineers behind from you discussing how the schedule just won't work, and of course, the endless interruptions from project managers who decide that every little schedule change needs to be personally broadcast as soon as it happens. Believe me, there are no doors to hide behind, no cubicle entrances to cover up, ...oh and headphones are not allowed since they prevent open-communication. For someone who appreciates a quiet environment when concentration is needed, it can be quite a challenge.   As far as data on this issue goes, it appears that most people have realized that if you want to get a lot done, relative to normal productivity, come in early or stay late or work on the weekends. Based on conversations I have had with co-workers, we've unscientifically determined that we can double productivity if we work during hours that no-one else is around!!!"

Steve deRosier offered this: "Joel Spolsky has a lot to say about productivity and private offices. Here's how he designed his office:
http://www.joelonsoftware.com/articles/BionicOffice.html

"On interruptions: http://www.joelonsoftware.com/articles/fog0000000068.html

"And on productivity of different programmers:
http://www.joelonsoftware.com/articles/HighNotes.html

"Oh and on getting things done (#5 applies to interruptions):
http://www.joelonsoftware.com/printerfriendly/articles/fog0000000332.html and
http://www.joelonsoftware.com/articles/fog0000000073.html (older
version, but the important item is at the top in this one "Smart and
Gets Things Done.")"

Rob wrote: "My manager is a good coder, and prefers to bounce his ideas off of me and my co-worker. While this is productive for him, it's absolutely destroying my ability to code. The biggest stretch I ever get is maybe 10 uninterrupted minutes, unless he's not in the office. *Walls* aren't the issue, it's distractions of any kind (interruptions, meetings, chit-chat).

"That said, there are definitely times where interaction is required: at the DESIGN phase.

"I should also clarify: I'm not talking about multiple people working on the same piece of code (extreme programming or agile), I'm talking about 2 or more people, each supposed to be working on different pieces (even if it's of the same system). The cards do collapse when you're interrupted. When I consulted from my home, my wife never 'got' that concept, and she interrupted me frequently throughout the day, then got upset when I had to work long hours just to get anything done.

"People who don't write code don't understand."

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.

My company, ILX Lightwave, has several open engineering positions in Bozeman, MT. Email
resumes to Adam Lamb (alamb@ilxlightwave.com). These jobs include:

Analog Design Engineer - This person will work on a dynamic R&D team in defining, designing and developing our next generation of test and measurement instruments. Responsibilities include the design and development of analog and mixed-signal control loops that utilize lasers, photo detectors, A/D and D/A converters, and thermoelectric coolers. BS in Electrical Engineering is required, MSEE preferred. Experience working with laser diodes and other electro-optical devices is a plus. A high energy, self-starter with excellent teamwork and interpersonal skills is desired. Strong oral and written communication skills are essential.

Engineering Group Manager - For this position we are looking for an exceptional individual to lead a small team of highly motivated professionals to meet schedule, performance, and cost objectives in the development of test and measurement instruments. In addition to project leadership responsibility, this position will also be a

primary resource for senior level analog, digital, or firmware design on the project team. This position requires at least 5 years of senior level design and at least 3 years of project management experience. A BS or MS in Electrical Engineering or related field is required. The successful candidate for this position will be a high energy self-starter with excellent technical and people management skills.

Firmware Engineer - This person will work on a dynamic R&D team in defining, designing and developing our next generation of test and measurement instruments. Responsibilities include the design and development of embedded firmware designed to interface digital control loops with analog and mixed-signal control circuitry. The designs are for the development of our benchtop test instruments that utilize lasers, photo detectors, A/D and D/A converters, and thermoelectric coolers.

Manufacturing Engineer - This individual will work with our production team to design high performance automated test systems, troubleshoot electronic instruments, and collaborate with development teams in the design and introduction of new instrumentation products. BSEE or BSCE required, advanced degree preferred.

Joke for the Week

.
"The process is called estimation, not exactimation." – Philip Armour

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.

To subscribe, send a message to 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 for more information.