

The Embedded Muse 123

Editor: Jack Ganssle (jack@ganssle.com)

Jan. 26, 2006

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
- Yet More on Tools
- Tool for the Theory of Constraints?
- Jobs!
- Joke for the Week
- About The Embedded Muse

Editor's Notes

TV chef Emeril is always “kicking things up a notch” by adding garlic and other zestiness to his recipes. How are your firm's engineering recipes? Want to kick up your development processes by more than a notch? I can present my Better Firmware Faster class at your facility. See <http://www.ganssle.com/classes.htm> .

I'll be presenting this seminar publicly in Denver on April 26 and Dallas April 28. Details at: <http://www.ganssle.com/classes.htm>. With luck there might be some decent skiing in the Rockies that week!

Christine Frayda has started a Wikibook called “The Software Engineers' Handbook,” which you can find at http://en.wikibooks.org/wiki/Software_Engineers_Handbook . There's a lot of material on-line there already, but she's hoping for help from the developer community.

If you're not familiar with wikis, well, it's time to get up to speed. There's an overview here: http://en.wikibooks.org/wiki/Main_Page , but basically a wiki is an open-content web-based collection of knowledge that anyone can edit. Though there's plenty of controversy about the accuracy of the most well-known of all wikis (Wikipedia, a vast and tremendously-useful encyclopedia at http://en.wikipedia.org/wiki/Main_Page), most wikis serve niche markets and don't suffer from the malicious attacks and competing views that have been directed at Wikipedia.

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

Quite a few engineering teams deploy local wikis as their prime documentation engine for projects. It's a great way to keep documents instantly up-to-date and constantly available.

Yet More on Tools

The tool discussion that started a couple of issues ago is still generating a lot of response. Thanks, all!

Kevin Light wrote: "There is a relatively new open source, cross platform, IDE tool which is moving forward really fast. <http://www.codeblocks.org> While the RC2 release shows off some of the features, the Subversion HEAD really is unbelievable and is usually pretty stable. The wxSmith plug-in for developing wxWidgets applications is very good for being bleeding-edge."

Steve Litt contributed: "Another hierarchy organizer is VimOutliner. VimOutliner's claim to fame is it's set up so that the touch typist can input and organize in real time without losing his train of thought. It has most of the standard outliner features -- expand/collapse, promote/demote, executable lines (execute random commands right from the outline), interoutline linking (Linux version only), checkboxes, body text. See <http://www.vimoutliner.org> .

"I use it several times a day, and even use outlines to create other formats such as web pages. If you look at <http://www.troubleshooters.com/linux>, that web page is stored as an outline and I run a script to update the HTML."

Mark Swayne wrote: "I've really enjoyed the discussion of tools. There are so many great tools available and so little time to find them. I'm an enthusiastic Vim user, and I find this quick reference card very handy: <http://tnerual.eriogerg.free.fr/vim.html> .

"For PERT and Gantt charting, the open-source Gantt-Project is worth checking out. It does both Gantt and PERT charts and uses an XML file format, so you can easily parse your data files and integrate it with other tools. See <http://ganttproject.sourceforge.net/> .

"Freemind (<http://freemind.sourceforge.net/>) is a really great tool for brainstorming and quickly building up outlines of hierarchical information. It's a "mind mapping" tool, a concept I wasn't aware of until I found Freemind. The best way to understand what it does is to look at the website.

"Doxygen (<http://www.doxygen.org/>) is a great to for automatically extracting comments from your code and creating consistent documentation. If you combine it with

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

Graphviz (<http://www.research.att.com/sw/tools/graphviz/>), it will generate immensely useful call diagrams.

‘Finally, I'd like to mention Tiddlywiki (<http://www.tiddlywiki.com/>). Tiddlywiki is a complete wiki-web in one HTML file. It works with most browsers, including Firefox and IE. I find it to be a very useful journaling tool. I keep my development diaries in a Tiddlywiki, the search, tagging and linking functions are very useful.’

Rocco Iacchetti found a different sort of tool: ‘We would like to indicate a tool for C development that we have found very helpful. It is Ristancase Development Assistant for C and eventually for VHDL.

‘The tool contains an intelligent editor, with the possibility to jump to function definitions and implementations, to find every use of a variable or function and so on. It gives you a lot of syntax checks (similar to lint), also with the integration of MISRA checks.

‘Finally it gives you the ability to generate the code documentation, in particular it is able to write flow diagrams of every routine.

‘It integrates a lot of compiler for different CPUs: we have used the integration to Texas DSP Code Composer Studio and to IAR Systems Compiler for Atmel ATmega family.

‘The tool is very cheap, even if it is not free. You could have more information on <http://www.ristancase.com> .’

Tool for the Theory of Constraints?

Tjark van Dijk and I have been corresponding about the Theory of Constraints. Here are his thoughts: ‘I checked out the project management tools, as these are most important for me. I find it very disappointing that none of these packages provide any means of implementing project management by the Theory Of Constraints that is preached by Elyahum Goldratt.

‘Is there an affordable tool that helps implementing TOC project management?’

‘We are a small organisation and we use an Excel sheet to manage project progress. It works and we can adapt it to our needs, but a good project management tool can help so much more.

‘TOC explanation:

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

“In short the method states that the problem with classic project planning, a project will never finish early, so by definition it will finish late. Every task is planned with a certain buffer to cover for uncertainty. If to a task 5 working days are assigned and the task is finished in 2 days, at least 2 things will happen:

1. The engineer will start polishing up the design, as he has still 3 days left
2. The person in charge of the next task will not be ready to continue on this task for at least 3 days.

“By the TOC rules, when each task is planned, you plan 2 times.

1. The reasonable time to finish the task, if no trouble occurs at all
2. A buffer time to cover for Murphy, and any kind of technical problems. The buffer times of all tasks are put together and planned at the end. The working procedure is then:
 - the engineer works on a task and reports at the end of every day when the task is expected to complete. If the task is estimated to run late, the project manager can assign time from the buffer to this task or decide to take other actions (like hiring a specialist to solve a major problem).
 - when a task is 2 to 3 days to completion, the person that has to follow up on this task gets a wake-up call to stand by to take over the project when this task is in the critical chain of tasks.
 - next to the critical chain there are "supplier tasks". Tasks that can start at some point and must end before the result is needed in the critical chain. These tasks are started from the critical chain progress on an early warning basis. This must prevent that a "supplier task" will become part of the critical chain by delivering too late. The person working on a critical chain task must be protected and assisted. So no distractions, no disturbances and no other tasks during this period.

“The result is that the project buffer is managed by a project manager, problems are signaled early (because of the estimation of the end date of a task) so the manager can take action if necessary. Tasks can end early and task progress is measured by what is left to do instead of what is already done. By defining and guarding the critical chain, the critical project flow is guarded in total and projects are much less likely to be late.

‘For more information, I can recommend some books:

Elyahum Goldratt, The Goal: a ‘novel’ about production management using TOC

Elyahum Goldratt, The Race: a ‘novel’ about project management using TOC

Elyahum Goldrat, Theory Of Constraints: didn't read this book, but this should give more insight in TOC

Thomas B. McMullen, Jr, Project management in the fast lane: A practical approach for project management when managing multiple projects using TOC. “

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

Jobs!

Let me know if you're hiring firmware or embedded designers. No recruiters please.

Micrium, the uC/OS-II people (and more) is looking for an entry level Software/Computer Engineer (0-3 years experience max.). You will be porting embedded software modules onto various popular microprocessors (ARM, PowerPC, Nios-II, MicroBlaze and others) and DSPs. You will also be supporting customers either over the phone, e-mail and possibly in person.

You must have a very good knowledge of the C programming language and also be comfortable with assembly language.

You must also have good understanding of hardware: CPUs, DMA, Ethernet, UARTs, Timers, etc.

The candidate must have excellent ENGLISH written and oral skills. Windows development skills would be a plus.

The candidate must be neat, meticulous, organized, have a great attitude, willing to work with others and learn new things. Candidate must be autonomous and require minimal supervision.

e-mail: Jean.Labrosse@Micrium.com

Software Engineer/Electrical Engineer IV

NACCO Materials Handling Group, Inc. ("NMHG") has an opening for a Software Engineer/Electrical Engineer IV to work in our Greenville, North Carolina facility. Responsibilities:

- * Broad ranging assignments in electric lift truck control systems that include overall control

- system integration of AC and DC electric motors, hydraulics, operator interfaces, sensors and actuators

- * Specific activities include preparation of requirements definitions, development of control firmware, use/development of PC based tools, and Failure Modes Effects Analysis

- * Direct interfacing with technical publications, manufacturing and customers.

Qualifications:

- * BSEE or BSCS and 5+ years of design experience

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

- * Experience with integration of embedded control system firmware for hydraulic and/or motor control systems using C or C++.
- * Working knowledge of power electronics, AC and DC electric motors and controllers is required.
- * Experience with CANbus and distributed systems is preferred
- * Basic knowledge of electrical and electronic circuits is expected
- * Experience with Visual C/C++ or Visual Basic, and/or XML is a plus
- * Excellent organizational and communication skills

e-mail: rob.kemp@nmhg.com fax (503) 721-1301. EOE. www.nmhg.com

Project Engineer-Controls

NACCO Materials Handling Group, Inc. has an opening for a Project Engineer-Controls to work in our Portland, Oregon facility.

Responsibilities:

- * Develop software design specifications for embedded, micro-controller based control systems used on forklift trucks
- * Develop and execute project plans for the software development including coordination of design efforts with hardware engineers and vehicle platform teams
- * Develop test procedures for validation of all supplier software including hardware-in-loop (HIL) test system, coordinating the test validation and software signoff for ECN release
- * Developing and managing design schedules to meet overall platform launch schedules.

Qualifications/Requirements:

- * BSEE, BSEET or BSCS or equivalent experience required
- * 8-15 years of experience in the embedded controls software development field required
- * Knowledge of CAN bus systems (SAE-J1939 and CAN Open specifications) preferred
- * Knowledge of C, Matlab and Python preferred
- * Working knowledge of MS Project, Excel and Word
- * Excellent written and verbal communication skills
- * Must be able to work in a team environment

e-mail: rob.kemp@nmhg.com fax (503) 721-1301. EOE. www.nmhg.com

Symbol Technologies Inc. has opening for Software Engineer I. This position is for a junior-level Software Engineer performing work in product design, analysis, or manufacturing support for a combination of those areas. Engineers in this position will

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

work on a wide variety of tasks ranging in complexity from low to mid-level. The level of responsibility in this position is moderate.

2-5 Years experience in software design and analysis as applied to Embedded or Application Software Development. Experience in 8-bit, 16-bit microprocessors/microcontrollers is required. Knowledge of the C Programming language, some assembly languages, microprocessor development environments and debugging techniques required. Knowledge of real-time embedded systems design, coding and debugging required.

Send resume and salary req. to: <http://careers.symbol.com>, tracking code 42752. Symbol Technologies, Inc., One Symbol Plaza, Holtsville, New York 11742.

Joke for the Week

From Rich Ries:

TOP TEN THINGS ENGINEERING SCHOOL DIDN'T TEACH YOU

10. There are at least 10 types of capacitors.
9. Theory tells you how a circuit works, not why it does not work.
8. Not everything works according to the specs in the databook.
7. Anything practical you learn will be obsolete before you use it, except the complex math, which you will never use.
6. Always try to fix the hardware with software.
5. Engineering is like having an 8 a.m. class and a late afternoon lab every day for the rest of your life.
4. Overtime pay? What overtime pay?
3. Managers, not engineers, rule the world.
2. If you like junk food, caffeine and all-nighters, go into software.
1. Dilbert is not a comic strip, it's a documentary.

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*".

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

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.

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

The Ganssle Group, www.ganssle.com