The Embedded Muse 107

Editor: Jack Ganssle   (jack@ganssle.com)   Dec. 6, 2004

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
- Salary Survey
- Testing!
- Temperature
- Ham Radio for Dummies
- Jobs!
- Joke for the Week
- About The Embedded Muse

Editor's Notes

I'll hold the year's last public Better Firmware Faster seminar this Friday, December 10th in Las Vegas. This is the only non-vendor class that shows practical, hard-hitting ways to get your products out much faster with fewer bugs. See http://www.ganssle.com/classes.htm for more details including cheap fly-in options, and information about free hotel rooms.

I often do this seminar on-site, for companies with a dozen or more embedded folks who'd like to learn more efficient ways to build firmware. See http://www.ganssle.com/onsite.htm.

Testing!

<rant mode on>
There is no single short term goal more important to a firmware developer than completing a module.

Completing means writing, testing and documenting the code.

Writing means producing a body of code that accomplishes its design goals, and that follows the lessons of modern software design. Cutting through all of the hype around

OOP, one critical lesson we all need to follow now is to totally encapsulate our code and data. The days of global variables should be long gone.

Documenting means different things to different people, depending on contractual requirements and corporate standards. Surely, though, any code whose operation isn't clear from the comments is not documented.

Testing means a complete functional test. When you are done, that module works - FOR SURE. Testing is where nearly every firmware project breaks down.

Testing hasn't the glamour of creating new code, yet is at least as important.

Testing avoids embarrassment. When you pass that function off to a colleague, you'll know that it really works over the entire domain of input conditions.

Testing gives you confidence. God knows we've all created mountains of poorly tested code. Adding another function is like adding to a house of cards. We know, deep in our souls, that the foundation is unstable and will create no end of problems... in the very near future. Sufficient testing creates trust in the code.

Though Demming and other quality gurus tell us we cannot test quality into a product, surely the software world has taught us that inadequate testing is a clear path to failure.

<rant off> - Inspired by seeing yet another large project collapse when testing was (as it often is) deferred till the end of the project. The "Ship It" mantra from management compressed the schedule to the point where much of the testing got lopped off and discarded. The product didn't work well in the customer's real application. Another $3m down the drain. Testing is not the only tool to create great code, but it's one that's important, and that all too often is poorly done.

Here's a resource for test plans:
http://irmc.state.nc.us/documents/approvals/reporting/TESTPLAN.pdf

One for safety-critical software can be obtained by sending a snail-mail letter requesting ITOP Number 1-1-057 to:

ATTN: CSTE-DTC-TT-M
HQ, U.S. Army Developmental Test Command
Aberdeen Proving Ground, MD 21005-5055
DSN: 298-1073

(However, this is a controlled document which may not be available outside of the US, UK, France and Germany).

Temperature

I'll never forget the call. Steve, one of our salesmen at the place I worked at the time, called from the road, bitterly complaining about our "lousy grain analyzers", that didn't work at all until they had warmed up for an hour or so.

Warmed up? Shades of the vacuum tube days! But all of our gear used ICs, mostly medium scale integration parts linked to an 8 bit controller. Admittedly the analog section was drifty at times, but Steve complained that the instruments didn't even start up until power had been on for a while.

He was demoing equipment from the company van across America's Great Plains. Somehow I managed to ask the right question: "Steve, where are you now?" "Montana" was the reply, "and get me an answer fast; I'm shivering!"

A few more queries determined that the temperature this fine January morning in Montana was 20 below zero. Fahrenheit. Suddenly it all made sense.

Our unit was designed for lab conditions, and so used standard plastic-packaged logic. 7400-series stuff, reliable as the day is long, but limited in its application to much more benign environments. In fact, most of these plastic parts are rated for 0-70 degree C operation only, a rating far exceed by Steve's winter Montana sojourn.

An awful lot of embedded equipment is used in quite harsh environments; too many designers fail to consider the temperature ratings of their components, and many are shocked to see how narrowly many components are speced. Consider the radio in your car - it must work immediately from temperatures far below zero to those resulting from a blazing summer sun baking dark sheet metal. Surely, this simply application must exceed military specs!

The moral is to check specs carefully, especially when your creation must run outside of a room temperature environment.

Ham Radio for Dummies

Ward Silver's Ham Radio for Dummies appeared in my in-box recently. Published in 2004 by Wiley it's a moderately hefty 360 page introduction to the world of Amateur Radio (aka "ham radio.")

For those not in the know, Amateur Radio is a means of communicating world-wide with surprisingly sophisticated equipment using a vast array of frequencies. It's internationally regulated; all hams must have a license which comes only after passing a test.

Ham radio is sort of out of the purview of embedded systems, but this hobby pushed many of us into the world of electronics and computers. I've had a license for many decades; as a teenager building (vacuum tube!) radios I learned an awful lot about electronics. For me designing and building equipment was more fun than chatting with other hams… but that's ham radio's appeal. There are many different facets to the avocation.

First I have to admit that the "For Dummies" books irritate me. I've spent a lifetime studying many subjects and may be uneducated on some, but never consider myself a "dummy." A title like "For Novices" or "An Introduction To" is a bit more seemly, yet for some reason these dummy books have a wide appeal.

This is a book for rank novices – not dummies, but for people who are interested in the hobby but just don't know where to go to learn more. Though the ARRL, the ham radio advocacy group, (http://www.arrl.org) does offer lots of useful information, this book packages the data in a more convenient form than any other publication I know of. The author does a superb job of describing what the hobby is all about. In fact, perhaps half the book discusses different aspects of ham radio. Did you know you can run your own TV station? Mr. Silver shows how. How about radioteletype, moonbounce, or other operating modes? This book gives an overview of each, with good links for more information.

It's peppered with amusing anecdotes and cartoons. The writing is lively and non-technical, easy enough for anyone to grasp.

You can't operate as a ham without a license and Mr. Silver clearly describes the testing process, as well as the different kinds of licenses available. This is not, however, a test preparation manual. You'll need other books, such as those at http://www.arrl.org/catalog/lm/ . Thus the book is totally tech-free.

What's the test like? In the US there are 35 or 50 multiple choice questions. Get 75% and you pass. Questions are both technical (electronics) and regulatory (the operating rules). Trust me… it's not hard to pass, especially using the aforementioned study material.

Though there is a license that doesn't need Morse code, any serious operator will want a license with more privileges. That requires passing a code test at 5 words per minute, 25 characters a minute or about two seconds per character. This requirement has been substantially downgraded from the 13 or 20 word per minute test of just a few years ago. With a little study and practice 5 WPM is a breeze.

One strength of the book is that Mr. Silver clearly explains actual operating procedures in a fashion that's more engaging than the ARRL publications.

Even after 35 years as a ham I didn't know about beacons used to check radio propagation (covered on page 101). And he discusses the digital modes which are all the rage today, with which I have no experience and therefore learned a few things.

Ham radio exists in a very different environment than when I first became interested in the hobby. It was relatively easy to build a rig when radios had only a handful of vacuum tubes. Today's multimode transceivers are packed full of surface-mounted ICs. It's harder to build this sort of equipment in a typical home shop. Yet there are still sources for kids and equipment, and a surprising number of hams build their own gear, especially "QRP" (very low power) gear. This book has a list of companies that sell kits.

Appendix B, a list of links and other sources, is invaluable.

The "bible" of ham radio is the ARRL Handbook for Radio Communication, which has a mediocre introduction to the hobby, but is fantastically complete in electronics and radio theory, coupled with plenty of build-it projects. Ward Silver's Ham Radio for Dummies fills the introductory niche left blank by the Handbook.

Jobs!

Let me know if you're hiring firmware or embedded designers. I'll continue to run notices for embedded developers as long as the job situation stays in the dumper. No recruiters please.

Scientific-Atlanta, a global communications innovator at the forefront of today's telecommunications revolution, currently has several openings for Embedded Software (including OS Kernel and VoIP) Developers, and Test Engineers. These positions are located in Lawrenceville, Georgia, and in our Cupertino, California office. We are creating the next generation of interactive multimedia networks for global video, voice and data transmissions. You can review and apply directly to our jobs by visiting us at http://www.scientificatlanta.com/contactus/employment.htm

Plantronics, the world leader in communications headsets, is seeking "Embedded Software Engineers" for our Core Technology Development group who have Firmware development and Wireless, Bluetooth, DECT, or 802.11 experience.

To apply go to Plantronics website at http://www.plantronics.com/careers. Click on "Embedded Software Engineers" to complete a short questionnaire that takes 2-5 minutes. Plantronics is an AA/EEO Employer.

Joke for the Week

Last issue I mused about how we, like Rodney Dangerfield, "can't get no respect." Steve Litt of Troubleshooters.com responded with this:

You think you get no respect -- try being a Linux guy. They bombard you with all these buzzwords you don't know and don't care about: Media Player, Internet Explorer, Outlook (they just call it "email"). The smarter ones discuss "the registry", "active X", and "C sharp".

All of them discuss magic incantations (start_button->accessories->utilities->doublespace), and look at you like you're an idiot when you don't equate the incantation with a program.

So you use the Linux guy's motto: Don't get mad, get even! You convert the kids to Linux -- no more Windows license fees. Even on the one remaining Windows (98 -- why upgrade to forced registration?), you have your daughter use OpenOffice to edit MS Office documents, Mozilla Composer instead of Frontpage Express, and Mozilla instead of (virus inviting) Internet Explorer.

And you make up your own incantation:

1. Put your box on a DHCP enabled subnet
2. Boot the Knoppix CD
3. Make sure network, video and sound are right
4. knoppix-installer
5. Enjoy

Put it another way:

* Bandwidth and blank CD to create Knoppix installation disk with browser, email, office suite, the world-famous Gimp drawing program, sound programs, video manipulation programs, several web editing programs, finance programs, database, web server, file and print server, and firewall: $1.00

* Not paying Bill Gates one red cent: Priceless.

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*".

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.