

The Embedded Muse 127

Editor: Jack Ganssle (jack@ganssle.com)

March 21, 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
- Test Driven Development
- Yet More on Tools
- Jobs!
- Joke for the Week
- About The Embedded Muse

Editor's Notes

Ashley Holland sent a link with a story about how engineers discovered a potentially fatal flaw in the Titan-bound Huygens spacecraft. Though this story does promote (mildly) MATLAB, it's interesting and worth reading. Fact is, I'm a sucker for any space story, especially those where an engineer saves an important mission!

http://www.mathworks.com/company/newsletters/news_notes/jan06/model_recovery.html.

This Computerworld story

(<http://www.computerworld.com/hardwaretopics/hardware/story/0,10801,108568,00.html?source=x10>) is an interview with J. Presper Eckert about the birth of Eniac. Fascinating stuff for computer history buffs. The article's author is a professor at the University of the Virgin Islands... now that's a job I want!

Denver and Dallas – Learn to develop better firmware faster in my one-day seminar in Denver on April 26 and Dallas April 28. Details at: <http://www.ganssle.com/classes.htm>. Till March 27 there's a \$50 discount.

TV chef Emeril is always “kicking things up a notch” by adding garlic and other bits of 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>.

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.

Test Driven Development, by James Grenning

James Grenning of Object Mentor and one of the signers of the Agile Manifesto contributed the following about Test Driven Development (TDD) for this issue. I think you'll find it quite interesting:

Test Driven Development is a practice that could save embedded developers a lot of time and help to improve the quality of their work. A number of recent issues raised by Jack are nicely addressed by TDD. I'll give you a brief overview and a few links so you can look for more information.

In TDD automated unit tests and embedded application code are developed in parallel. Features are realized and the architecture is implemented.

Briefly here is what you do to use TDD for developing embedded software:

- 1) You write automated unit tests that plug into a unit test harness.
- 2) Tests and code are written in a tight feedback loop:
 - write a small test expressing what you want the code to do.
 - Write the code to make the test pass, repeat.
- 3) The unit tests can be run on either the development platform or the target.
- 4) You first get code running in the friendly confines of your development system.
- 5) Periodically you compile unit tests and application code and test in the target (if you have one).
- 6) Periodically you link the whole program and make sure it runs through its paces with the target

TDD has been around for a while. We first started using it in late 1999, and first applied it to embedded work in 2000. Although this practice was developed outside of the embedded world, it seems to me that it was invented for embedded.

In TDD the embedded software engineer writes test code that can be run on both the development system and the target environment. I've heard many embedded developers scoff at testing in the development system: "You need the real hardware to test this code". Sure there are hardware dependencies, but in my opinion the embedded developer can decouple from the hardware fairly easily and perform meaningful tests on the host platform. Not all code can be run on the host, but most code can.

The power of TDD lies in the feedback it provides. The practice of TDD can answer these two questions:

- Does the code I just wrote do what I want it to do?

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.

- Does the rest of the system still behave as it should?

Knowing the answers to these questions is very powerful. The first question is pretty easy to answer without automated unit tests; you just test your changes. But it is difficult to answer the second question economically unless you have automated unit tests. Too often software changes result in unwanted behavior, a.k.a. BUGS! If your product owners say “its OK to break existing features when you add new one” then you have no need for TDD.

Here is what a few simple unit tests look like for a stack.

The test framework calls `SetUp` before each TEST and calls `TearDown` after each TEST. Each test is independent and results in either success or failure.

```
static struct Stack* stack;

static void SetUp()
{
    stack = Stack_create();
}
static void TearDown()
{
    Stack_destroy(stack);
}

TEST(Stack, IsEmpty)
{
    CHECK(Stack_isEmpty(stack));
}

TEST(Stack, IsNotEmpty)
{
    Stack_push(stack, 25);
    CHECK(!Stack_isEmpty(stack));
}

TEST(Stack, PushPop)
{
    Stack_push(stack, 25);
    LONGS_EQUAL(25, Stack_pop(stack));
    CHECK(Stack_isEmpty(stack));
}
```

One embedded team I know using TDD reported that the end of their project lifecycle was spooky. They were not finding the multitude of pesky bugs their usual process produces. So what did they do about it? They used the time to find some system level bugs that might otherwise have made it to the field, been reported by the customer and then fixed at great expense.

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.

Another team I know switched target platforms in the middle of their development cycle. The tests helped them to confidently make this change and to find and fix the problems quickly.

I ported the open source test harness to the Hitachi SH3 and discovered that the `strstr()` function behaved differently than the other execution platforms: Coldfire, gcc, Visual C++ and Visual studio. The unit test harness had an odd failure. The tests pointed us in the right direction and quickly we discovered that `strstr("hello", "")` works differently on the SH3 as compared to the other environments. We made a quick change to the test harness and it again is portable among those environments. I would not have liked to have to chase that bug down in a live embedded system (actually that would probably have been fun, but luckily I did not have to find out).

This activity also results in meaningful data. For example: the number of unit tests passing (this number should go up every week), unit test coverage (this number should continually grow, maybe close to 100%).

There are several papers on TDD on our website. I'll also be doing a talk on TDD for embedded at ESC in April. Please come and hear more about it.

<http://www.objectmentor.com/resources/listArticles?key=topic&topic=Embedded%20Software>

<http://www.objectmentor.com/news/index#JwG%20ESC%20April%202006>

For an open source unit test harness in C++ (that can also be used to test C) take a look at CppTestTools.

<http://www.fitnessse.org/FitServers.CppFit.CppTestTools>

Yet More on Tools

Mike Teachman contributed the following tool information: "The mention of InfoSelect made me think of a free tool that I use a few times a day.

"It's Copernic desktop search. www.copernic.com I use it solely to search MS Outlook for emails and contacts. It indexes all your emails. It's blazingly fast and has an intuitive interface. I tried Google desktop search, but, for me Copernic was superior. Before writing this email I wondered "has Jack already mentioned Copernic in a newsletter". So, I opened Copernic and typed "ganssle copernic". nothing came up."

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.

Howard Speegle writes: "I don't have a job to offer, but I have several small jobs that need to be done in C for various microcontrollers. These range in size from an evening or two or three for a savvy guy, to maybe a couple of months of part-time work. Some are as simple as an application to implement a serial to parallel interface using a Silicon Labs C8051F330. Others involve porting some existing firmware to a Renesas M16C. I can provide excellent specs and flow charts and will do the debugging myself. I am not at all picky about the location of the programmer as long as we can communicate by e-mail. All work will be done at the programmer's end, so my location shouldn't really matter. If any of your readers have an interest in such small jobs, I would like to hear from them. <mailto:Howard@DivaAutomation.com>Howard@DivaAutomation.com"

Joke for the Week

Kinney Bacon sent a great link: <http://www.sysprog.net/quotmain.html>

Very good (and unfortunately true) quotes about software. For instance: "Always code as if the guy who ends up maintaining your code will be a violent psychopath who knows where you live."

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.

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.