

# The Embedded Muse 112

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

March 21, 2005

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
- Computer Magazines
- Debouncing
- Jobs!
- Joke for the Week
- About The Embedded Muse

## Editor's Notes

Want to increase your team's productivity? Reduce bugs? Meet deadlines? Take my one day Better Firmware Faster seminar. You'll learn how to estimate a schedule accurately, thwart schedule-killing bugs, manage reuse, build predictable real-time code, better ways to deal with uniquely embedded problems like reentrancy, heaps, stacks and hardware drivers, and much, much more.

I'm presenting this:

- Austin, TX on April 18
- Baltimore, MD on April 20

Want to be your company's embedded guru? Join us! There's more info at <http://www.ganssle.com/classes.htm>, including cheap flights to these cities from around the USA.

If your outfit has a dozen or more engineers who can benefit from this training I can present the seminar on-site.

The Embedded Systems Programming folks asked for articles appropriate for the upcoming April Fools' Day. I couldn't resist; see: <http://embedded.com/showArticle.jhtml?articleID=159901030> .

*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.*

## Computer Magazines

A reader asked me what magazines I recommend for someone doing firmware work. It's a good but loaded question; if you subscribe to all the publications I do you'll never get any work done!

This field moves awfully fast... but there are a surprising number of pockets where things never change at all. I see plenty of developers stuck in some sort of time warp, creating firmware the same way they did years ago. It's crucial that we programmers and engineers stay connected with the latest developments in the field.

Equally important is to keep abreast of commercial products. Tools change; new products come out that can accelerate development tremendously.

Here's my software-related reading list:

Circuit Cellar Ink – Though more or less an electronics hobbyist magazine, and though it really doesn't have much of a software focus, each issue includes interesting projects with code. I enjoy Tom Cantrell's column which often describes new embedded processors and technology, and find the projects interesting. See <http://www.circuitcellar.com>.

Communications of the ACM – As professionals we're supposed to act like professionals, which includes belonging to our professional organization. Communications of the ACM is the Association of Computer Machinery's primary organ. Though many or even most of the articles aren't relevant to a firmware person's needs, there's enough of interest to warrant reading it. The columns are interesting and entertaining. The ACM, like the IEEE, is expensive to join, and the publications can be frustratingly devoid of practical content. But for me, at least, the benefit outweighs the price. See <http://acm.org>.

Crosstalk – Though not well known in the embedded space, Crosstalk is an excellent publication aimed at developers building big systems. If you're working with the CMM, TSP, or PSP this is a must-read. Free. See <http://www.stsc.hill.af.mil/crosstalk/about.html>).

Dr Dobb's Journal – There's one or two columns a month about embedded systems, including the always interesting ruminations of Ed Nisley. The ads are useful. Most of the other articles are nearly orthogonal to the needs of embedded systems (C#, web pages, etc). See <http://ddj.com/>.

*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.*

EDN – EDN is truly a hardware-centric magazine aimed at electronics engineers, but the Design Ideas section often contains firmware algorithms. If you're an EE like me then the magazine is a must-read. Firmware lives at the intersection of hardware and software, so firmware developers should have some knowledge of chips and circuits. Free. See <http://edn.com> .

EE Times – A must-read for anyone in this industry. EE Times won't give you any insight into new techniques, but it does cover new products, businesses and emerging trends. You'll find articles about future technologies like quantum teleportation. (I can't wait to teleport rather than fly!) And it's the best source of info about employment trends and more. It's free and weekly. See <http://eetimes.com> .

Embedded Systems Programming – The one magazine that really caters to firmware developers. Every issue has at least one article I want to read. The ads are highly-targeted at us as well. Free. See <http://embedded.com> . (Note: an awful lot of people complain that they're subscription requests are ignored. I've repeatedly beaten the magazine people up about this, but nothing seems to change. You can read it on-line at <http://embedded.com> ).

IEEE Computer – A monthly publication that's interesting though usually not tremendously useful to the practicing engineer. I enjoy the columnists' perspective and one or two articles a month. See [www.computer.org](http://www.computer.org) .

IEEE Micro – This IEEE publication is targeted at the microprocessor industry. To my frustration I rarely find much of interest in the magazine but stay subscribed in the hope that will change. See <http://www.computer.org/micro> .

IEEE Transactions on Software Engineering – I really, really want to get something useful from this magazine so at least read the titles every issue. No doubt the academics find much of interest but I suspect few engineers would benefit much. See <http://www.computer.org/tse> .

MAKE – A new O'Reilly publication. This is a hobbyist's dream magazine, with articles about building all sorts of cool projects. The current issue even has an article about time management for geeks. I've only just subscribed so can't rate it yet, but do find the \$35/yr price scandalously high for a 4 issue publication. See <http://make.oreilly.com> .

Queue – Another ACM publication. If you're an ACM member this is free and worthwhile; I wouldn't join just for Queue, though. Occasional article of interest, but fun columns. See <http://www.acmqueue.org> .

*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.*

SD Times – Sort of an EE Times for software people, SD Times is aimed mostly at IT-type developers. But sometimes there will be an embedded article. The ads and product reviews are a useful way to keep up with tools. Free. See <http://sdtimes.com>.

Software Development Magazine - Nothing at all about embedded, but lots about software engineering. This is the magazine to get to stay up with the Agile community. Free. See <http://www.sdmagazine.com>.

Software Test and Performance – This new magazine by the SD Times people has yet to prove itself. I'm very interested in the testing issue, and hope the publication blossoms. Free. See <http://www.stpmag.com>.

It's pretty much impossible to actually \*read\* all of these each month. I browse through them all, ripping out occasional articles and ads of interest and toss those into a pile for airplane reading.

What software-related publications do you read?

## Debouncing

The code I described in <http://www.ganssle.com/debouncing.htm> debounces only one edge of the switch's closure, to be very responsive to the user. But Barrett Davis has another take on this, and sent in some code that will debounce both edges. It's very straightforward and includes a test harness:

```
// -----  
// debounce.cpp  
// Copyright (c) 2005 Barrett Davis, released into the public domain.  
// -----  
#include <iostream>  
#include "stdafx.h"  
  
#define MAX_CHECKS 5  
  
unsigned char RawKeyPressed( void );  
  
// -----  
// debounce() data:  
// -----  
const int MAX_DEBOUNCE_INDEX = MAX_CHECKS;  
unsigned char state[ MAX_DEBOUNCE_INDEX ]; // Debounce switch array  
unsigned char debounce_state; // Debounced state.  
unsigned char debounce_index; // Index into the array.
```

*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.*

```

void debounce( void )
{
    // -----
    // Debounce the switch array, using a modification of Jack Ganssle's
    // DebounceSwitch3() algorithm found at ww.ganssle.com/debouncing.pdf
    // -----
    if( debounce_index >= MAX_DEBOUNCE_INDEX ) {
        debounce_index = 0;           // Keep the index within bounds.
    }
    state[ debounce_index++ ] = RawKeyPressed(); // Read the raw key
    unsigned char on  = 0xFF;         // 1=switch closure, 0=don't care.
    unsigned char off = 0x00;         // 0=switch open, 1=don't care.
    for( int ii = 0; ii < MAX_DEBOUNCE_INDEX; ii++ ) {
        on  &= state[ ii ];           // Accumulate constant 1s.
        off |= state[ ii ];           // Accumulate constant 0s.
    }
    debounce_state |= on;             // Force 1s on in result.
    debounce_state &= off;           // Force 0s off in result.
    return;
}

const int MAX_KEY    = 60;
int      key_index   = 0;

unsigned char RawKeyPressed( void )
{
    // -----
    // Return a sample key value for testing.
    // -----
    static const unsigned char key[ MAX_KEY ] = { 0,0,1,1,0,1,0,0,1,0,
                                                    1,1,1,1,1,1,1,1,1,1,
                                                    1,1,1,1,1,1,0,0,1,0,
                                                    0,0,0,0,0,0,0,8,0,0,
                                                    0,1,0,0,3,0,0,0,5,1,
                                                    1,1,1,1,1,1,3,1,1,0};

    if( key_index >= MAX_KEY ) key_index = 0;
    return key[ key_index ];
}

void init( void )
{
    // -----
    // Initialize our test system.
    // -----
    debounce_state = 0;
    for(int i=0; i<sizeof(state);i++)state[i]=0;
    debounce_index = 0;
    return;
}

int main( void )

```

*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.*

```

{
// -----
// Unit test to compare switch debouncing techniques.
// -----
init();
printf("\nRaw Debounced");           // Column header
for( int ii = 0; ii < MAX_KEY; ii++ ) {
    debounce();
    printf("\n%i %i",RawKeyPressed() & 0xff, debounce_state & 0xff);
    key_index++;
}
return 0;
}

```

## 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.

Opto 22, a leading manufacturer of industrial control and remote monitoring solutions, is looking for qualified embedded systems engineers to help with product development and QA. We have great opportunities for engineers with the following skills and experience:

- Firmware development in C/C++
- TCP/IP knowledge
- Microprocessor based hardware checkout and design
- Strong communication skills

Opto 22 is a successful, established, and growing company that provides excellent benefits and learning opportunities. Send resumes to: [careers@opto22.com](mailto:careers@opto22.com)

TimeSys Corporation is searching for Linux Software Engineers to be based in Campbell, California, and Pittsburgh, Pennsylvania. This position is responsible for developing, packaging, maintaining and testing board support packages (BSP). For more information regarding this position, please, visit TimeSys' Web site at [www.timesys.com](http://www.timesys.com). Interested applicants should submit their resume to [jobs@timesys.com](mailto:jobs@timesys.com).

## Joke for the Week

Chuck Sommer sent this, sort of depressing, observation: If a programmer spends half his time "DEBUGGING" what do you call how he spends the other half of his time?

*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.*

"BUGGING"

## 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*".

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)**