CE Workgroup

Status of Embedded Linux

March 2018

Tim Bird

Architecture Group Chair

LF Core Embedded Linux Project

Nature of this talk…

- Quick overview of lots of embedded topics
- A springboard for further research
  - If you see something interesting, you have a link or something to search for

Outline

Kernel Versions
Technology Areas
CE Workgroup Projects
Other Stuff
Resources

Outline

Kernel Versions

Technology Areas

CE Workgroup Projects

Other Stuff

Resources

Kernel Versions

- Linux v4.11 – 30 Apr 2017 – 70 days
- Linux v4.12 – 2 Jul 2017 – 63 days
- Linux v4.13 – 3 Sep 2017 – 63 days
- Linux v4.14 – 12 Nov 2017 – 70 days
- Linux v4.15 – 28 Jan 2018 -- 77 days
  - I predicted: 21 Jan 2018 (70 days)
  - What happened? – Spectre/Meltdown
- We're on 4.16-rc3 now
  - Expect 4.16 on March 25 (but I'd prefer April 1!)

Linux 4.11

- New kernel refcount API
- TinyDRM subsystem added
- New statx() system call
  - https://lwn.net/Articles/707602/
  - 2038-safe time values
  - Mask of fields to obtain (for efficiency)
- Sched.h refactoring
  - Non-mainline code: watch out!

Linux 4.12

- BFQ and Kyber block I/O schedulers
- Mini-tty prep work
  - Not full mini-tty implementation yet
- Proper support for USB type-C connectors
- AnalyzeBoot tool
  - Reads dmesg (and possibly ftrace log) and produces html graph of boot events
  - Part of Intel pm-graph tools project
    - https://github.com/01org/pm-graph
  - See tools/power/pm-graph/analyze_boot.py

Linux 4.13

- TLS implementation in the kernel
  - Should help with HTTPS performance
  - See https://lwn.net/Articles/666509/
- Next-interrupt prediction
- F2FS support for disk quotas
- Kselftest transitioning to TAP13 protocol

Linux 4.14

- New kernel stack unwinder (ORC) for x86_64
  - Better unwinding via kernel-specific out-of-band structure (for every kernel PC address)
  - See https://lwn.net/Articles/728339/
- zstd compression for btrfs and squashfs
- Better cpufreq coordination with SMP

Linux 4.15

- Cramfs supports mapping persistent memory
  - Can use for XIP
- AMD display core system accepted
- Device tree compiler has support for overlays
- RISC-V support
- Spectre/Meltdown mitigations
  - KPTI
  - retpolines

Linux 4.16 – some stuff

- Initial support for the Jailhouse hypervisor
- eBPF support for functions
- arm64 mitigations for Spectre and Meltdown
- High resolution times now have two modes, to allow them to be run in software interrupt context
- More Spectre migitations
  - array_index_nospec()

Outline

Kernel Versions

Technology Areas

CE Workgroup Projects

Other Stuff

Resources

Bootup Time

- Analyze_boot tool – new in in 4.12
- Some good previous talks:
  - ELCE 2017 - *A Pragmatic Guide to Boot-Time Optimization* by Chris Simmonds
  - ELCE 2014 - *12 Lessons Learnt in Boot Time Reduction* by Andrew Murray
  - ELC 2015 - *Fastboot Tools and Techniques* by John Mehaffey
- Android boot time ideas
  - ELC 2017 – *Improving the bootup speed of AOSP* – Bernhard Rosenkranzer

Device Tree

- Device Tree validation
  - Schema for binding language, validator for bindings and for device tree data
  - New proposal for device tree validation by Pantellis and Grant Likely
- Updated Device Tree specification
  - Want to update material and make it more available
- Overlays
  - Device tree compiler has support for overlays

File Systems

- zstd compression for btrfs and squashfs (4.14)
  - Faster and smaller compression/decompression
    - https://clearlinux.org/blogs/linux-os-data-compression-options-comparing-behavior
  - How to use it (BTRFS):
    - https://btrfs.wiki.kernel.org/index.php/Compression
  - See https://www.phoronix.com/scan.php?page=news_item&px=Linux-4.14-Zstd-Pull
- F2FS support for disk quotes (4.13, 4.15)
  - Apparently used by Android
- UBIFS support for encryption (4.10)

Graphics

- TinyDRM
  - Provides graphic support for small simple displays (eg displays over i2C or SPI)
  - Hope to replace framebuffer drivers over time
  - See https://www.phoronix.com/scan.php?page=news_item&px=TinyDRM-Patches-Posted
- Presentation
  - ELC 2017 *What Can Vulkan do for You*? - by Jason Ekstrand
- Working on support for virtual reality
  - Keith Packard's talk at LCA 2018

GPU drivers

- Nvidia, Vivante and Broadcom GPUs have open drivers
  - Nouveau, Etnaviv, and VideoCore 4
- Qualcomm Adreno
  - Freedreno continues to be developed (June 2017)
    - See https://www.xda-developers.com/open-source-adreno-project-freedreno-receives-new-update/
- Imagination PowerVR – no public driver, although one was teased in 2015
  - Apple dropping Imagination (April 2017)
- ARM Mali – Some work (Lima project) on earlier chip versions
  - Status update: https://lwn.net/Articles/716600/
  - Some recent work:
    - https://github.com/yuq/linux-lima
    - https://notabug.org/cafe/chai

Networking

- Time Sensitive Networking
  - ELCE 2017 *Deterministic Networking for Real-Time Systems (Using TSN)* – by Henrik Austad
  - so_txtime option for high-resolution transmit time
  - IEEE deterministic networking (DetNet) working group
    - Lots of standards
- Bluetooth 5 – supported

Power Management

- Power-efficient workqueues
  - More efficient work scheduling
    - Results in about 15% better energy consumption
    - See https://lwn.net/Articles/731052/
- Better cpufreq coordination with SMP
  - Allows non-local CPU to adjust frequency
    - Good for when a non-local CPU schedules work on a CPU, and the work needs a frequency boost
  - See https://lwn.net/Articles/732740/

Real Time

- Realtime Summit (Oct 2017)
  - Realtime trouble, lessons learned
  - Using Coccinelle to detect and fix nested execution context violations
  - SCHED_DEADLINE: what's next?
  - Future of tracing
  - See https://lwn.net/Articles/738001/
- Status of Prempt-RT patch
  - Hotplug locking
  - Timer wheel rework
  - Big outstanding issue: dentry cache locking

Real Time (cont.)

- Presentations:
  - ELCE 2017 *Deterministic Networking for Real-Time Systems (Using TSN)* – by Henrik Austad
  - ELCE 2017 *Measuring the Impacts of the Preempt-RT Patch* – by Maxime Chevallier
  - ELC 2017 *Effectively Measure and Reduce Kernel Latencies for Real-time Constraints* – by Jim Huang
  - ELC 2017 *Real-Time Linux on Embedded Multicore Processors* – by Andres Ehmanns
- More stuff at ELC 2018

Security

- Spectre and Meltdown
  - Break security via side-channel timing attacks using speculative execution
  - Variants 1, 2 (Spectre), and 3 (Meltdown)
- Is a family of vulnerabilities related to speculative execution
  - Many modern processors vulnerable
    - Many embedded processors not affected
- Very severe problem:
  - Can read data you're not supposed to
  - Vulnerability has existed for 20 years!
  - Cannot be fixed with CPU firmware updates
  - Mitigations are expensive

How they work...

- Basic idea:
  - Make processor execute speculatively
  - Get data into cache based on that execution
  - Time the access to cache to determine data

Spectre/Meltdown analogy

- Cooking analogy:
  - Mom is making either a pie or a cake
    - The ingredients are secret
  - You aren't allowed to eat pie, but Mom doesn't know who the dessert is for when she starts
  - To save time, she makes both, and then throws the pie away (and gives you the cake)
  - She had to go to the store for ingredients
    - She leaves the ingredients in her pantry after using them
  - You ask Mom to make you something with the same ingredients (e.g. cookies)
    - If it takes her a short time, you can figure out what ingredients are in her pantry
  - (You do this a million times)

Spectre – Variant 1

- Variant 1 = bounds-check bypass
  - What is it?
    - Use speculative execution to detect data outside the bounds of an array
    - It does not cross security boundaries
  - What processors affected:
    - Any with speculative execution (ARM, Intel, AMD, ...)
  - Mitigations:
    - fence operation to prevent speculation
    - bounds-friendly mask
      - Prevents speculative code from accessing outside the array
    - array_index_nospec()

Spectre – Variant 2

- Variant 2 = branch target injection
  - What is it?
    - poisoning of the branch prediction buffer, to make speculative execution happen “incorrectly”
  - What processors affected: Many
  - Mitigations:
    - retpoline mechanism
      - fancy returns to avoid speculation
      - Needs compiler support for retpolines
    - RSB (return stack buffer)-stuffing
    - New processor flags by Intel

Meltdown – Variant 3

- Variant 3 = rogue data cache load
  - What is it?
    - Determine data in kernel address space through speculative execution
    - This crosses security boundaries !!
    - Data read prior to check of security privilege (on speculative execution)
      - Results are “retired” when security privilege is processed, but by that time, data is in cache and it’s value can be detected
  - What processors affected: Intel, ARM Cortex A75
  - Mitigations:
    - KPTI (Kernel Page Table Isolation)
      - Remove kernel address space from user process
      - Is very expensive, due to new overhead on every syscall

Security issue handling

- Lots of questions (and some complaints) about how Spectre/Meltdown were handled
- Flaws detected by multiple security researchers in similar time frame (summer 2017)
- All agreed to info. embargo until January
  - Embargo mostly held – news broke on Jan 2
  - Normal Linux security channels were not used
    - Specifically ‘security@kernel.org’
    - Complaints about kernel developers who could help not getting information soon enough
- Distros, and Tier 2 OSes and customers did not get enough notice

Status of mitigations

- Variant 1:
  - fence operations and bounds-masking are still being worked on (not in 4.15, some in 4.16-rc3)
  - Much more work expected
- Variant 2:
  - Some retpolines are in 4.15
  - Some new flags from Intel to turn off prediction, that the kernel supports
- Variant 3:
  - KPTI in 4.15 for Intel
  - KPTI in 4.16 for Arm64
- See https://lwn.net/Articles/746551/

Security

- Kernel hardening
  - http://kernsec.org/wiki/index.php/Kernel_Self_Protection_Project
    - Rare_write infrastructure
      - Keep some code and data read-only most of the time
      - https://lwn.net/Articles/724319/
    - GCC plugins for kernel security
      - Kernexec
        - Prevent kernel from executing user-space code
      - Structleak (mainlined in 4.11)
        - Zero out kernel structures passed to user space, under some conditions
      - See https://lwn.net/Articles/712161/
      - Randstruct
        - Randomize C structure layout
        - See https://lwn.net/Articles/722293/

Security Presentations

- ELC 2017 *Securing Embedded Linux Systems with TPM 2.0* – by Philip Tricca
- ELCE 2017 *Security Features for UBIFS* – by Richard Weinberger

System Size

- Initramfs compression method is selectable
- Nicolas Pitre work
  - Configurable POSIX timers – in v4.10
    - https://lwn.net/Articles/701095/
  - Mini TTY
    - Smaller implementation of TTY subsystem, for embedded
    - Saves about 38K
    - https://lwn.net/Articles/721074/
    - People wanted refactoring of full-size TTY instead of new small implementation, but Nicolas said that wasn't feasible

System Size (cont.)

- Shrinking the scheduler
  - Drops features and eliminates realtime and deadline scheduler classes
  - Saves about 20k
  - https://lwn.net/Articles/725376/
  - Lots of resistance to this
  - Code complexity increase is not worth saving 20k (according to Ingo Molnar)
  - Disagreement on whether Linux should support computers with sub-1MB memory

Size Presentations

- ELCE 2017 *Embedded Linux Size Reduction Techniques* – By Michael Opdenacker
  - Great overview of reduction techniques and status
    - Toybox and musl (smaller libc) are worth looking at
    - Long list of things that can be worked on
- Linaro Connect SFO 2017*: Internet of Tiny Linux (IoTL): Episode IV* – by Nicolas Pitre
  - http://connect.linaro.org/resource/sfo17/sfo17-100/
- LinuxCon North America: *Running Linux on Tiny Peripherals* – by Marcel Holtmann
  - Got Linux to around 1MB for IOT sensor project

Nicolas Pitre LWN.net articles

- Nicolas has a series of articles on shrinking the kernel
  - https://lwn.net/Articles/746780
- Covers lots of issues:
  - Link Time Optimization
  - CONFIG_TRIM_UNUSED_KSYMS
  - Removing sub-systems
- It's a 4-part series
  - One more part coming
  - Last part will requires subscription for first 2 weeks
    - Either subscribe, or wait 2 weeks

Testing

- Kselftest
- Fuego
- Kernelci.org
- LAVA V2
- Kernel regression tracking
- Plumbers session on testing

Kselftest

- Unit test system inside kernel source tree
- Recent work:
  - -silent option, to reduce output clutter
  - Support for O= option, to build outside source directory
  - Lots more regression tests (preferred place for syscall compatibility/regression tests (over LTP)
  - Converting to TAP (Test Anything Protocol) for test output (started in 4.13)
- See https://lwn.net/Articles/737893/

Fuego

- New Test Framework for collaborating on tests and test infrastructure for Linux
- V1.2 Oct 2017
  - Unified output format
    - Convert all test results to JSON, in a format compatible with Kernel CI
  - New pass criteria system
  - Test dependency system
    - Board dynamic variables
- Tests being added on a consistent basis
- Move documentation to reStructuredText

Kernelci.org

- Place to get free build/boot testing for your board
  - Builds 126 trees continuously, then reports any errors
- http://kernelci.org
- Presentations:
  - ELC and ELCE 2016 – by Kevin Hilman
  - Linaro Connect:
    - Kernelci and lava update - See https://lwn.net/Articles/716600/
- The most successful public, distributed build and test system for Linux, in the world!

LAVA

- Linaro Automation and Validation Architecture
- V2
  - Job files now use Jinja2 templates
    - Was previously hand-written JSON
  - Jobs are run asynchronously, without polling,
  - ZeroMQ is used for communications.
  - ReactOBus is used to run jobs from messages.
  - Requires more explicit board configuration

Other efforts

- Kernel regression tracking
  - Thorsten Leemhuis reported at kernel summit issues and difficulties doing regression tracking
    - Kernel developers don't like Bugzilla
    - Not enough people doing this work (no community effect)
    - Errors on specific hardware are hard to reproduce
    - Would be good to identify sub-systems with more regressions and target those for more testing
  - See https://lwn.net/Articles/737666/ and https://lwn.net/Articles/738216/
- Plumbers sessions on testing
  - See https://lwn.net/Articles/734016/ and https://lwn.net/Articles/735034/

Toolchains

- LLVM 4.0.0 is released
  - Some code size improvements from optimizations (GVNHoist)
  - Experimental support for LLVM coroutines
  - https://lwn.net/Articles/716979/
- Presentations:
  - ELC 2017 - *GCC/Clang Optimizations for Embedded Linux* – by Khem Raj
  - Plumbers 2017 *Building the kernel with Clang* – by Nick Desaulniers
    - https://lwn.net/Articles/734071/

Tracing

- Dynamic function tracing events
  - Ability to create a tracepoint for a function at runtime
  - Goal is to avoid having a tracepoint become part of kernel ABI
  - Is work-in-progress
  - See https://lwn.net/Articles/747256
- Presentations:
  - ELC 2017 *Dynamic Tracing Tools on ARM/AArch64 Platform: Updates and Challenges* - by Hiroyuki Ishii
    - Great overview of Linux tracing capabilities and programs

Miscellaneous

- Printk issues
- Year 2038 work
- Linus issues with Kconfig
- AGL making inroads
- Android mainlining status
- Linux in Supercomputers
- FreeRTOS switched to MIT license

Printk issues

- Discussion on kernel summit mailing list
  - Lots of issues with printk
    - It's not per-CPU, console lock held too long, it has complicated code paths, and lots more
  - See thread start at:
    - https://lists.linuxfoundation.org/pipermail/ksummit-discuss/2017-June/004358.html
- Recent discussions about KERN_CONT
  - KERN_CONT is unreliable for SMP kernels
  - Latest kernel puts '\n' between lines that don't have KERN_CONT
  - Eventual removal of KERN_CONT
    - Maybe use of seq_buf for outputting serialized date atomically
  - https://lwn.net/Articles/732420/

Year 2038 work

- 3 areas of work
  - Converting all 32-bit timestamps to 64-bit in the kernel
    - e.g. New statx() system call
    - Many patches are in-progress (vfs layer, v4l, device-mapper, input subsystem)
  - C libraries
    - Lots of work in glibc to make everything backwards compatible
      - Even programs built with 32-bit timestamps should work
  - Distribution builds – fixing up individual packages
- See https://lwn.net/Articles/717076/

Linus issues with Kconfig

- Discussion on kernel summit mailing list
  - Kconfig is too hard for end users
  - What can be done?
  - Linus' complaint:
    - https://lists.linuxfoundation.org/pipermail/ksummit-discuss/2017-June/004504.html
- Ideas:
  - Config fragments
  - Higher level options
  - Better dependencies
    - From distro feature to kernel config

AGL status

- First car in US with Entune (AGL-based infotainment OS) was 2018 Toyota Camry
  - Announced at Open Source Summit Japan by Toyota
- Mazda and Toyota collaborating on Entune
  - https://www.theregister.co.uk/2017/08/29/mazda_toyota_linux_entune_car_infotainment/

Android mainline status

- Lots of Android SoC support still out-of-tree
  - Vendors have mainlined some things, but it will take time (many years)
  - Android kernels for shipping devices are likely to remain 2-years behind mainline
    - LTS support expires at 2 years
    - Greg will maintain some LTS kernels for 6 years, but stop if vendors don't use it
  - There is interest in improving LTP
    - But mainline on Android devices would be better
  - See https://lwn.net/Articles/738225/ for report by Greg Kroah-Hartman

Linux in Supercomputers

- Linux now runs 100% of the top 500 supercomputers
  - As of November, 2017
  - Was 99.6% (498 out of 500) in June 2017
  - Most powerful machine, China's "Sunway TaihuLight" uses 650,000 processors!
  - See http://www.omgubuntu.co.uk/2017/11/linux-now-powers-100-worlds-top-500-supercomputers

FreeRTOS license change

- FreeRTOS switch to MIT license
  - Richard Barry started working for Amazon last year
  - Amazon released FreeRTOS version10 with MIT license
    - Removed GPL v2 (with extra clauses)
    - Added branding “fair use” clause to MIT
  - Is a pretty big deal, IMHO
  - See https://lwn.net/Articles/740372

Outline

Kernel Versions

Technology Areas

CE Workgroup Projects

Other Stuff

Resources

Projects and initiatives

- Shared Embedded Distribution
- LTSI
- Fuego
- eLinux wiki

Shared Embedded Distribution

- Goals
  - Create an industry-supported distribution of embedded Linux
    - Main goal is very long term support (15 years)
- Status
  - Working on building Debian with Yocto Project
  - 3 projects - meta-debian, isar and elbe wish to collaborate and combine their yocto recipes into a single layer.
- Next steps
  - Continued integration of Debian-based build and packaging systems

Long Term Support Initiative

- LTSI 4.9 is current LTSI kernel
  - Work is in progress on next release 4.14
- Most of industry is using LTS or LTSI
- Using upstream-first policy for patches
- Security fixes are very important
- Presentation:
  - ELCE 2017 *Using Long Term Stable Kernel for the Embedded Products* – by Tsugikazu Shibata

Fuego - Linux Test Framework

- Working on lots of issues
  - LTP improvements and updates
  - Improved visualization and robustness
  - Recently added support for hardware control
- Want to interoperate with board farm standards
  - First, have to create board farm standards
  - Probably plan a test automation track as Plumbers
- Presentation:
  - Japan Jamboree 63: *Fuego Status and Roadmap December 2017* – by Tim Bird

eLinux wiki

- http://elinux.org
  - Web site dedicated to information for embedded Linux developers
    - The wikipedia of embedded linux!
- Hundreds of pages covering numerous topic areas: bootup time, realtime, security, power management, flash filesystem, toolchain, editors
- Slides and Videos for 12 years of ELC!!
- Please use and add to site

Outline

Kernel Versions
Technology Areas
CE Workgroup Projects
Other Stuff
Resources

Trade Associations

- Linaro still doing lots of great work
  - Lava v2 and kernelci
  - Now promoting Zephyr
  - Linaro Connect consistently has useful material
- Linux Foundation
  - Continuing to grow
    - First event in China sold out in 2 weeks (1200 attendees)
  - Over 100 conferences, 67 projects
    - Not just Linux
  - More than 500 members

Conferences

- Embedded Linux Conference Europe
  - Lots of great sessions!
  - See https://elinux.org/ELC_Europe_2017_Presentations
- Embedded Linux Conference 2018
  - March 12-14, Portland, Oregon, USA
- Japan Jamborees
  - Continuing
- Open Source Summit Japan
  - June 20-22, Tokyo, Japan
- ELC Europe 2018
  - October 22-24, Edinburgh, Scotland

Legal Issues

- SPDX adopted by Linux kernel
  - Extensive review done of files without license identifiers
  - Lots of files were tagged with SPDX license IDs
  - See https://lwn.net/Articles/739183/
  - and kernel commit: ead751507
    - applied in 4.14-rc7!
    - https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git/commit/?id=ead751507de86d90fa250431e9990a8b881f713c
  - Some complaints about process used for patch

Community issues

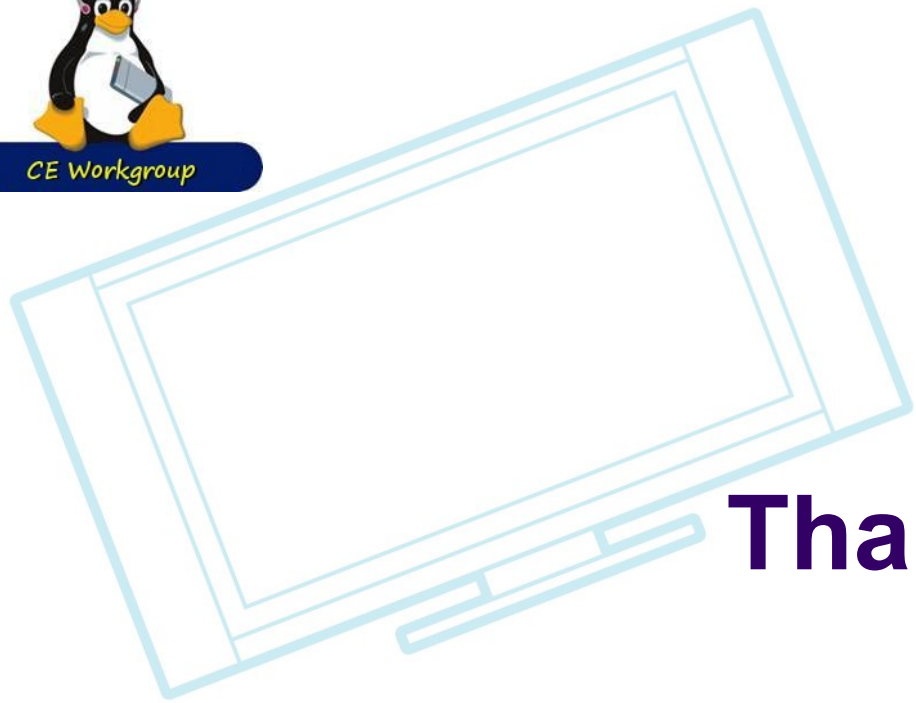
- Complaints about abusive maintainers in the Linux Community
  - Daniel Vetter gave a talk at LCA about the issue
    - See https://lwn.net/Articles/745817/
  - Other talks at same event describe how to get involved
- Linux Foundation TAB (Technical Advisory Board) is looking at issue
  - “code of conflict” was issued in 2015, but few issues have been brought to TAB
  - Currently discussing possible actions to improve community discourse

Outline

Kernel Versions
Technology Areas
CE Workgroup Projects
Other Stuff
Resources

Resources

- LWN.net
  - http://lwn.net/
  - If you are not subscribed, please do so
- Kernel Newbies
  - http://kernelnewbies.org/Linux_4.??
- eLinux wiki - http://elinux.org/
  - Especially http://elinux.org/Events for slides and videos
- Celinux-dev mailing list



Thanks!

Meltdown Analogy Observations

- You have to start with an empty pantry
  - Must clear the cache before attempting exploit
- The operation has to recover quickly, in order to read privileged data fast
  - If when you ask for a pie, you get put in jail, it's too slow to get useful data
    - Meltdown had tricky way to avoid fault on privileged read
- Must be able to ask for same ingredient quickly, before pantry gets "overwritten"
- Need a timer to figure out how long Mom took to access pantry
  - One mitigation for Spectre, in web browser, was to eliminate high-res timer access