

Henry Wiechman,
*Software Product Management,
Embedded Processing*

Kevin Chalmers,
*Director,
Linux™ Core Product Development*

Chase Maupin,
*Integration team manager,
Linux Core Product Development
Texas Instruments*

Executive summary

In today's fast-paced and hyper-competitive electronics markets, software developers must focus the majority of their time on delivering that next-generation innovative functionality that will set their products apart from the competition and less time on commodity software. These engaging and creative features can mean the difference between success and failure in a new product introduction. Enabling and empowering the rapid development of new functionality starts at the foundational level of the system's software environment – that is, at the level of the Linux™ kernel – and builds upward from there.

Unfortunately, having a robust, stable, high-quality and up-to-date kernel on which to base software development sometimes can be as time consuming as actually developing new functionality. Often, software development groups must spend a considerable amount of time evaluating, tracking and implementing kernel updates, maintaining and supporting code patches, stress testing new kernel additions and other activities which distract programmers from innovative feature development and put market releases at risk.

(continued)

Mainline Linux™ ensures stability and innovation

TI has removed many of these distractions by adopting a policy for incorporating recent stable kernels provided by the kernel.org open source community into the software development kits (SDKs) that support the rapid development of new functionality on TI's major Arm®-based embedded processors, including the Sitara™ and Jacinto™ processors.

Keeping up with the kernel

Because of the constantly evolving nature of open source projects, developers are often faced with a dilemma regarding when to upgrade the underlying Linux kernel to a more recent version. Determining the appropriateness of a migration to a new version of the kernel often includes a cost/benefit analysis. The new features, functionality, and bug fixes incorporated into a kernel must first be identified, defined and described. In addition, the value, if any, associated with each new feature must be determined and quantified. The value of the benefits of the new kernel must be carefully weighed against the costs of the migration. When locking onto an older kernel for a significant period of time, the costs incurred during a kernel migration can include resolving a large number of implementation issues. For example, a bug fix at a certain point in a new version of the kernel can affect other sections of the product's code base. Sometimes these effects are unexpected and can require considerable time and effort to overcome. A common example of this would be the need to migrate to new common frameworks that have been developed in the mainline kernel which could render a code base obsolete. Moreover, a code base may contain a number of work-arounds or patches for known bugs or feature gaps. Some patches may no longer be needed with the new kernel. However, the remaining patches must be moved with the code base into the new kernel. This can involve a laborious effort of tracking and evaluating each and every patch to determine whether it is still needed or can be discarded. In addition, if the patch is discarded, consideration must be given to any potential ramifications throughout the software environment. A large number of changes during a migration can make maintaining continuity in the code base difficult while trying to leverage the development investment previously made in the code. Through the use of mainline development methods TI is looking to reduce such costs of migrations and help users more easily take advantage of the benefits of the new kernel. The decision to migrate to a new kernel is never trivial because the mission-critical role of

Maintaining focus

a development group is to provide new and innovative features and functionality that will make products competitive in the marketplace. With this goal in mind, a stable, robust and high-quality operating system environment is a fundamental requirement.

TI's support of the mainline Linux kernel ensures an efficient development environment and avoids much of the disruption and distraction that can accompany a migration to a new kernel. TI provides evaluative and analytical information on the supported kernels to facilitate effective user decision making and to enable orderly migrations when necessary. All primary TI Arm-based embedded processors in the future will support the same mainline Linux kernel, simplifying the conversion of product code bases from one kernel environment to the next and across a system manufacturer's different product lines. Users will not need to worry about features disappearing or not being migrated from one kernel version to another as their presence in mainline ensures a smooth transition between kernel versions. Users will be able to focus on their differentiating features and not worry about the foundation on which they are based.

For a number of years, TI has been an ardent supporter of the open source community and, specifically, the **kernel.org** community organization. When new versions of the kernel were made available, the various product groups within TI would base their solutions on particular versions of the kernel. Today new features and functionality as well as TI's bug fixes are provided to the kernel.org community so these improvements can be incorporated into mainline Linux. In turn, TI has committed to fully evaluate, and thoroughly document, test and productize software development kits (SDKs) utilizing mainline Linux kernels for its major processors.

Through this process, users of TI technology will be able to take full advantage of support for the latest devices, new features and bug fixes in new versions of the kernel. And, these improvements will be made available at regular intervals so that over time developers will benefit from the incremental growth in stability, robustness and quality of the mainline Linux kernel. Such an environment offers the kind of continuity and continuance that simplifies user migrations from one kernel to a newer release. Because of the compatibilities inherent in each version of mainline Linux, users will not have to worry about whether features and functionality in their code base will transfer to the next version of the kernel. TI will cooperate and collaborate with the maintainers at kernel.org who perform stringent code reviews and analyses to ensure the long-term quality and continuity of mainline Linux. Before any new code is incorporated into a new kernel, the code itself and the ramifications it might have for the rest of the operating environment are constantly and exhaustively evaluated by the organization. See Figure 1 on the following page.

For its part, TI has committed to analyzing new versions of the kernel and thoroughly documenting the additions and changes to the kernel. For those versions of the mainline Linux kernel provided to its users, TI will compile easy-to-read change logs and well documented release notes so users can quickly decide when a migration is warranted. In addition, TI will perform comprehensive stress tests on new kernels, applying a wide range of use cases and application requirements to ensure the quality, stability and robustness of the development environment supported by TI devices. Advanced users will also be able to take the Linux kernel directly from kernel.org to gain access to new features or critical fixes ahead of TI releases.

Mainline helps increase quality and robustness

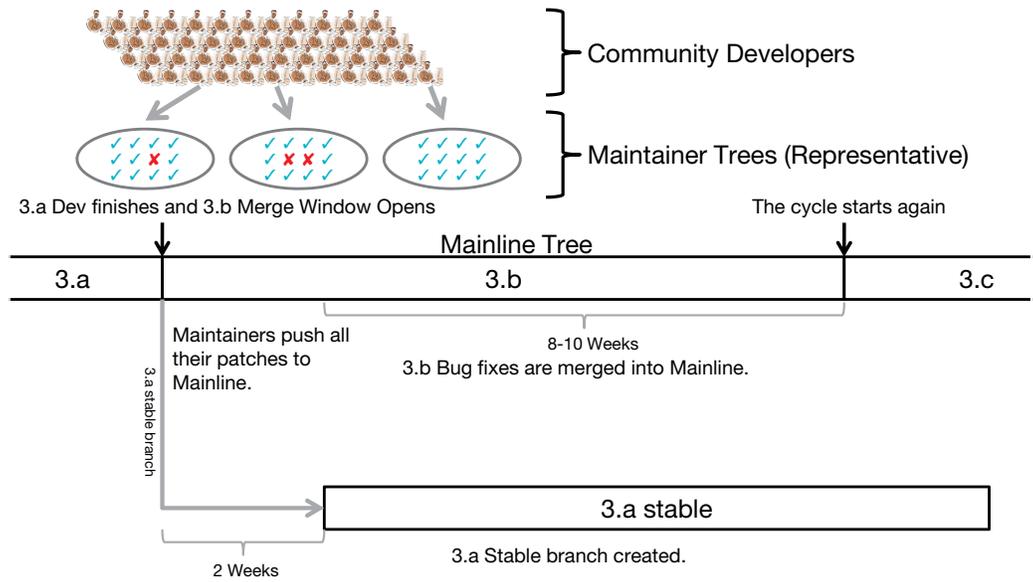


Figure 1: TI and other developers contribute to kernel.org to increase the quality and robustness of mainline Linux

Supporting mainline Linux

TI will support mainline Linux and distribute it to users through the SDKs that complement key TI Arm-based embedded processors, including the Sitara™ and Jacinto™ processors. In addition to supporting stable mainline kernels, TI's SDKs also include a number of tools and capabilities which accelerate code development, such as application examples, boot loaders, various software libraries, training resources and well-researched documentation. Figure 2 shows the Sitara Linux SDK.

The SDKs will be based on stable versions of the mainline Linux kernel. TI has committed to investing in enhancements to mainline Linux and collaborating with kernel.org with the intent of supporting the “long-term stable” (LTS) version of the kernel following this designation by kernel.org. LTS is that version of the

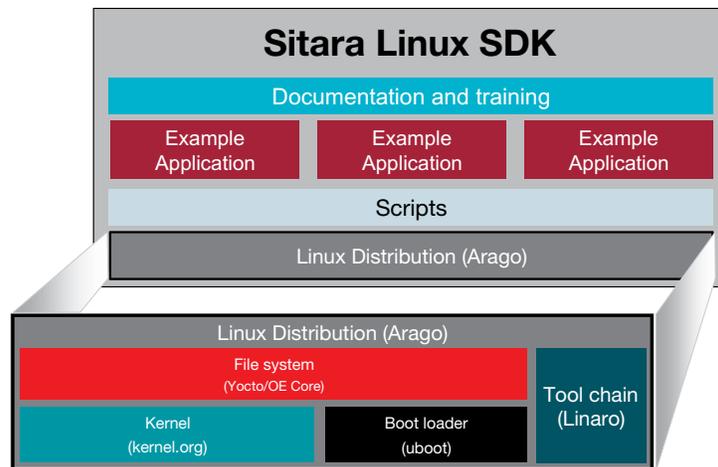


Figure 2: TI's Sitara™ Linux™ SDK and Arago Linux distribution

kernel which kernel.org has committed to supporting for a period of at least two years in order to cover a typical product life cycle while providing a common place to share both security and bug fixes. The kernel version currently available in the SDK will act as a stepping stone to the eventual implementation of the LTS version. The TI Linux SDK for a given processor based on the current LTS kernel is expected to be available after a necessary delay to facilitate testing and productization.

For more information For more information on TI's embedded development resources visit <https://www.ti.com/design-resources/embedded-development.html>

Important Notice: The products and services of Texas Instruments Incorporated and its subsidiaries described herein are sold subject to TI's standard terms and conditions of sale. Customers are advised to obtain the most current and complete information about TI products and services before placing orders. TI assumes no liability for applications assistance, customer's applications or product designs, software performance, or infringement of patents. The publication of information regarding any other company's products or services does not constitute TI's approval, warranty or endorsement thereof.

KeyStone and Sitara are trademarks of Texas Instruments Incorporated. All other trademarks are the property of their respective owners.

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES "AS IS" AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI's products are provided subject to TI's Terms of Sale (www.ti.com/legal/termsofsale.html) or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI's provision of these resources does not expand or otherwise alter TI's applicable warranties or warranty disclaimers for TI products.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2020, Texas Instruments Incorporated