

# DOING MEDICAL RIGHT WITH NUCLEUS

October 2008

## ABSTRACT

The design and development of electronic devices is no trivial matter - especially when the devices serve the medical industry. As the expense of developing electronic devices continues to rise, engineering teams are turning to commercial off-the-shelf (COTS) software to address the cost issue. The question is, while COTS software is used in arenas such as consumer electronics, mobile phones, etc., can it also be used in the production of safety critical devices? This paper discusses the similarities and illustrates the benefits gained by using the Nucleus OS approach when developing today's medical devices.

Author:  
Todd Brian  
Product Marketing Manager, Nucleus OS Platform  
todd\_brian@mentor.com

Mentor Graphics Corporation  
8005 SW Boeckman Road  
Wilsonville, OR 97070 USA

Designing, developing, and differentiating an electronic device is no trivial matter. It's expensive and time consuming; and not surprisingly, it is becoming more so. The expense of software in comparison to the entire cost of a product's development is almost fifty percent of that device. In some markets, software costs greatly exceed this percentage. Driven by consumer demand for expanded features, such as connectivity and portability, the cost of developing software in every market is increasing.

Many engineering groups, particularly those who are developing non-safety critical devices, have turned to commercial off-the-shelf (COTS) software to deter the rising development costs. There are a number of benefits to using this strategy. In any business, concentrating on core competencies pays off. COTS developers are able to focus on a particular area of expertise and become experts. They also have greater opportunity to amortize their development costs over millions, if not billions, of devices as opposed to a single product line or product family. Only the very largest engineering departments can maintain a competitive advantage over those who use a dedicated COTS vendor. The purpose of this white paper is to illustrate the benefits and advantages to be gained by using the Nucleus OS approach in the development of medical devices.

There are many similarities between developing software for medical devices and producing software for consumer electronics devices.

The following list describes some of these challenges:

- Dealing with new standards and complex connectivity protocols
- Unrealistic schedules
- More code, more bugs
- More tasks, fewer people
- Real-time performance
- Storage of critical control data
- Network communication
- Wireless connectivity
- Human interface and controls

## THE ADDED RIGOR OF MEDICAL DEVICES

However, developers of safety critical devices such as those used in medical, avionics, and transportation face an additional set of hurdles that must also be addressed. These include:

- Regulatory burdens
- Protection from unexpected events
- Client privacy and data integrity

## NUCLEUS OS FOR MEDICAL DEVICES

Nucleus OS is deployed as a series of libraries. These libraries are grouped in six different categories of OS services, as displayed in Figure 1.

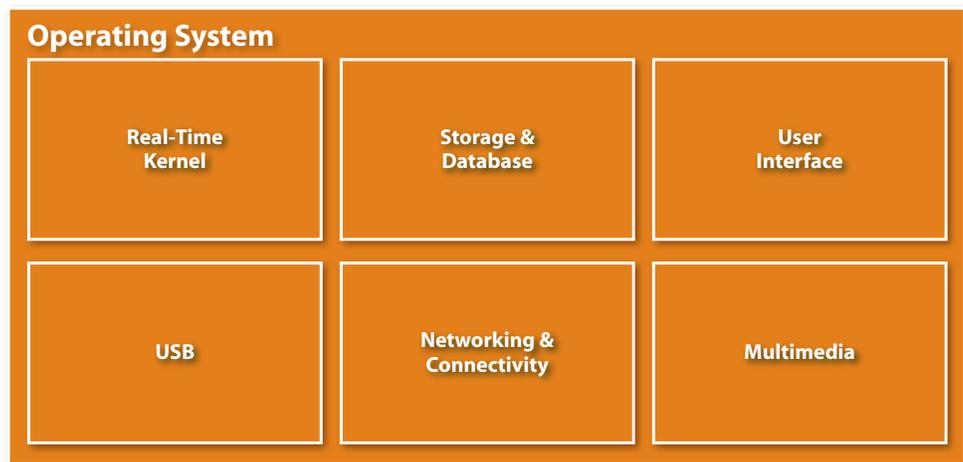


Figure 1: Nucleus OS can be deployed as a series of libraries.

---

### **Real-time Kernel Services**

The Nucleus OS Kernel is an integrated collection of deterministic, real-time kernel services, kernel extensions, and APIs. Nucleus OS Kernel Services forms the foundation of the Nucleus OS and has been deployed in a wide range of products (well over a billion devices have shipped to date) including consumer electronic products, and in particular, mobile phones.

### **Networking and Connectivity Services**

Nucleus OS Networking and Connectivity Services offer extensive networking protocols, drivers, and utilities – all designed specifically for embedded systems. 53 networking protocols are included, covering a vast array of RFCs and IEEE specifications. Nucleus OS offers a familiar BSD sockets interface, high performance, and validation against external test suites including ANVL and TAHI. A C++ interface is also available.

### **Storage and Database Services**

Nucleus OS File System Services provide data management and storage solutions in a real-time, multi-tasking application. Multiple file formats are supported. These include FAT, ISO9660, and a fault tolerant Flash format. All formats are accessible via a simple, extensible API. A wide range of media types are supported including IDE, Flash, SATA, USB, CDROM, PCMCIA, Compact Flash, and SD.

### **User Interface Services**

UI Services simplifies the development of high-quality graphical user interfaces by providing low-end rendering services and a full GUI control framework for constructing traditional windowing interfaces such as dialogs, buttons, edit boxes, and so forth. Nucleus OS also provides a menu-driven 3D interface engine with drag and drop design tools, which facilitates the creation

of the next generation in consumer devices. Flexible input management options enable any GUI to be controlled via touch panel, mouse, keypad, and keyboard.

### **Multimedia Services**

Nucleus OS incorporates a multimedia framework that implements the OpenMAX/IL industry standard. This facilitates the rapid integration of multimedia codecs from disparate sources, including hardware accelerated codecs for playback and the recording of audio/visual media.

### **USB Services**

Nucleus OS USB Services offer a complete out-of-the-box USB software solution for any project requiring connectivity via the standard USB bus, including host and function stacks, class drivers, hardware drivers, and multimedia management. The software is USB 2.0 certified and supports USB On-The-Go (OTG). The range of class drivers provided is among the widest in the industry covering a vast range of peripheral types. Vast arrays of controller types are also supported, from the standards-based host OHCI and EHCI to manufacturer-specific controllers.

Beyond the USB specifications, several multimedia transports take advantage of USB. Nucleus OS supports MTP, which is used to manipulate multimedia content for example, synchronizing songs between a PC and an MP3 player. Nucleus OS also supports PictBridge, which is used for direct image printing from a specified device, such as a digital camera.

## **REGULATORY COMMITMENTS**

There are several reasons why Nucleus OS is the preferred solution for use in medical devices. Among the listed below, Nucleus OS also reduces the regulatory burden for the designer.

---

### **Proven Track Record in Medical Devices**

Companies such as Bayer, Aspect Medical, Metran America, Analogic, ZOLL Medical Corporation, Plexar, and many others have all chosen to use Nucleus as the software foundation for their devices. Proven and reliable, Nucleus has been deployed in over 90 FDA Class II and Class III devices. These devices include X-Ray and ultrasound machines, portable heart defibrillators, blood analyzers, homodynamic monitors, infusion pumps, digital endoscopes, heart monitors, CT scanners, portable organ preservation equipment, and ophthalmic photo coagulators.

### **Delivered with Complete Source Code**

For over 15 years, Nucleus OS been deployed with complete source code. The code is well documented and is easy to use and read. Mentor Graphics is proud of its history of delivering high-quality source code which is provided at no additional cost. Also, Mentor Graphics provides an ever growing list of sample code, demos, and examples that minimize the time required to deploy a new product.

### **Completely Scaleable**

Nucleus OS is designed so that it is completely scaleable. If you don't wish to utilize a particular service or feature, the un-needed code is not contained in the executable image. This dramatically lowers the cost of certifying the device.

### **Utilizes MMU Enabled Hardware**

No code is perfect. Having an OS that takes advantage of the MMU hardware serves two purposes. One, it allows developers to catch common programming errors as early in the process as possible. And two, it provides a way of handling unplanned exceptions in the field.

## **CONNECTIVITY PRIVACY AND SECURITY**

As in other embedded application areas, medical device connectivity, particularly wireless, is growing at a staggering rate. The conundrum for medical device manufacturers is how to add connectivity, but also keep the transmitted data secure. The Health Insurance Portability Accountability Act (HIPAA) regulation, while mainly applying to the protection of millions of American workers, improves portability and continuity of health insurance coverage. HIPAA has huge ramifications for device manufacturers and their customers.

An individual's privacy is paramount under HIPAA. Accidental disclosure of one's data has fines that start in the thousands of dollars range and dramatically rise from there. Contrast that with the desire to keep patients out of the hospitals, computerized calling, and wireless connectivity. Device manufacturers need to build products that meet their customer's ease of use requirements, while at the same time ensure that a patient's data is not improperly disclosed.

With support for TCP/IP (ANVL and TAHI tested) and over 50 networking protocols to choose from, Nucleus OS offers a number of protocols that specifically provide the means for secure communication:

- SecSH (Secure Shell)
- L2TP
- 802.11x (a – i)
- IPsec
- SSL
- IPv6 (IPv6 Ready)

Nucleus provides the entire connectivity infrastructure a device requires for web enabled tethered and untethered communications.

## PROTECTION FROM UNEXPECTED EVENTS

As mentioned earlier, use of hardware MMU detects bugs early in the development cycle. This is the most cost effective time to find and fix bugs as opposed to later in the development life cycle. It also aids in handling unexpected conditions in the field, prevents a costly recall or, more importantly, prevents bodily harm or death to a patient if the bug is not fixed. This is not the only way a device can fail to perform correctly.

Power failure, device failure, or other events can all cause equipment malfunctions. For some devices, it's essential that its state at the point of failure is known. A **fault tolerant file system** is considered a critical component for many medical devices. One example might be an infusion pump used for disease therapy. Many drugs are extremely toxic and must be delivered in well measured doses over a very period of time. If the pump's batteries are exhausted during the therapy and the batteries must be replaced, using a fault tolerant file system allows the unit to maintain its settings so that the therapy can continue without interruption.

## NUCLEUS OS FAULT TOLERANT FILE SYSTEM

The Nucleus OS fault tolerant file system uses non-volatile Flash memory to store its device data. It is also compatible with the Nucleus OS FAT and CD-ROM based file systems. Combining fault tolerant and FAT based file systems is easy because they both utilize the same API and are easily configured via settings, so that each storage device is accessible through the same API.

Additionally, the FAT file system can utilize a variety of storage media including: Flash, USB, mass storage, IDE accessible drives, CF, and SD/MMC cards. This flexibility is perfect for those who are looking for PC file system compatibility with removable media. Diagnostics, off-site storage, and post mortem analysis are easily performed by removing, for example, a USB stick and downloading the PC compatible file into the PC for review.

## CONCLUSION

Nucleus OS has been employed in thousands of designs and deployed in billions of units worldwide. With over 90 design wins in medical applications alone, Nucleus is recognized by the FDA and other agencies around the globe as a solid candidate for use in medical devices ranked class II and class III.

Mentor Graphics strives to provide solid development tools and operating system software. Mentor is committed to developing emerging technologies such as HIPAA compliant connectivity, fault tolerant file systems, and fault recovery software, as well as a host of OS services that enables developers to focus more attention on added value software rather than foundation type software such as an operating system.

**For additional product information please visit [www.mentor.com/embedded](http://www.mentor.com/embedded).**

Copyright 2008 Mentor Graphics Corporation. This document contains information that is proprietary to Mentor Graphics Corporation and may be duplicated in whole or in part by the original recipient for internal business purposes only, provided that this entire notice appears in all copies. In accepting this document, the recipient agrees to make every reasonable effort to prevent unauthorized use of this information.

**Corporate Headquarters**  
Mentor Graphics Corporation  
8005 SW Boeckman Road  
Wilsonville, OR 97070-7777  
Phone: 503.685.7000  
Fax: 503.685.1204  
**Sales and Product Information**  
Phone: 800.547.3000

**Silicon Valley**  
Mentor Graphics Corporation  
1001 Ridder Park Drive  
San Jose, California 95131 USA  
Phone: 408.436.1500  
Fax: 408.436.1501  
**North American Support Center**  
Phone: 800.547.3003

**Europe**  
Mentor Graphics  
Deutschland GmbH  
Arnulfstrasse 201  
80634 Munich  
Germany  
Phone: +49.89.57096.0  
Fax: +49.89.57096.400

**Pacific Rim**  
Mentor Graphics (Taiwan)  
Room 1001, 10F  
International Trade Building  
No. 333, Section 1, Keelung Road  
Taipei, Taiwan, ROC  
Phone: 886.2.87252000  
Fax: 886.2.27576027

**Japan**  
Mentor Graphics Japan Co., Ltd.  
Gotenyama Garden  
7-35, Kita-Shinagawa 4-chome  
Shinagawa-Ku, Tokyo 140-0001  
Japan  
Phone: 81.3.5488.3033  
Fax: 81.3.5488.3004

