System Design Considerations for Medical Device Manufacturers

Jim McElroy
Director, Medical Industry Business Development, Green Hills Software

Mark Prowten
Director, OEM Embedded Wireless, Silex Technology America
The Medical Device Manufacturer’s Goal

Deliver innovative yet safe, effective and secure medical devices to market cost effectively
Device Development Challenges

- **Regulatory** - Increased oversight, regulation, and most important – Accountability
  - U.S. Medical Device Accountability Act most definitely will affect how software will be developed in the future
  - Submissions: FDA 510(k), PMA
  - Standards: IEC 62304, IEC 60601, ISO 14971, IEC 61508, IEC 62366

- **Technical** - Medical devices are becoming increasingly complex
  - More capable, more connectivity, requiring less power, smaller footprint, lower cost to produce
  - More security, reliability, and safety
  - Many of these forces work against each other (ex. security and safety vs. innovation and interoperability)
Client Applications – Connectivity on the Rise

- Anesthesia
- Automated External Defibrillator (AED)
- Blood Analyzers
- Ventilator
- Dialysis
- Infusion pumps
- Glucose monitoring
- Patient monitoring
- Respirator
- Surgical Navigation
- Sleep apnea
- Imaging
- Magnetic Resonance Imaging (MRI)

- Hospital beds
- Medical dispensing
- Radiography
- Computed Tomography Imaging (CT)
- Positron Emission Tomography Imaging (PET)
- Ultrasound Imaging
Technical Considerations

- Device Safety Classification – Class II, III
- Software Safety Classification under IEC 62304 – Class A, B, or C
- Security concerns
- Architecture of the device
- Leveraging SOUP in medical devices
- Power and mobility considerations
- Device Connectivity and Interoperability – Wired and/or Wireless
  - To other devices – PAN, LAN, WAN connectivity
    - USB PHDC, Bluetooth, Bluetooth LE, ZigBee, Wi-Fi, ...
  - To the hospital/clinician IT infrastructure
Methods for Mitigating Risk

- Increase your team’s **software development maturity**
- Adhere to domain-specific software development **standards (e.g., 62304)**
- **Reuse** existing validated software
- Leverage modern software **development tools** that provide optimized development, debugging, and testing solutions for medical devices
- Deploy **COTS hardware and software**
- Utilize **domain-specific**, trusted **software platforms** and operating systems
Safe, Secure, Reliable Platform for Medical Devices

- Separation Kernel - VASs
- Native guest OS and application execution
- SDK for “native” security and safety critical apps
- Communication across partitions is possible
- Safety BSP

Benefits
- Mitigates Risk – separate critical from non-critical tasks
- Leverage legacy investments
- Virtualization
Adding Wireless?

- Wireless is everywhere today it has become the quintessential form of communications and will continue to expand and evolve.

- Adding wireless to medical devices has many benefits to patients, caregivers, and medical facilities.

- Why 802.11 Wi-Fi?
  - Provides ample bandwidth for just about any medical device
  - Enables mobility by removing the tether of the network cable
  - Founded on industry standards and interoperability
  - Provides secure reliable connectivity for challenging environments
  - Is widely deployed in hospitals and medical offices
Understanding the Wireless Puzzle

Which Radio a/b/g/n?

Intelligent Module, Radio Card or SiP?

How is security and authentication handled?

Certifications and Compliance?

What about the Antenna?

In-house or outsource development?

Which RTOS?

Protocol Stack & User Interface?

Wireless Driver Development?
Wireless Implementation Considerations

- **Safety Impact**
  - Will adding wireless reduce or improve safety for...
    - ... the patients?
    - ... the patient records?
    - ... the medical facility?

- **Reliability**
  - Reliable device communication is critical.
  - Wireless has proven itself as a reliable means of voice and data communication.
  - However, reliability can be greatly diminished if the RTOS, wireless driver, security supplicant, and radio do not work seamlessly together.

- **Interoperability with Hospital Wireless Infrastructure**
  - Device Conformance with Enterprise Wireless Systems is critical
  - Security Supplicant integration and verification
Security Considerations

- **Security of Medical Devices**
  - Devices must be secure
  - Data transport must be secure over the air and over the LAN.
  - If not secure intruders could hack the medical device and potentially affect device function.

- **Security of Patient Information**
  - Security of patient data is critical, the use of Wireless can help protect data only if the proper security precautions are enabled.
  - Documented validation of security interoperability between the RTOS, the wireless driver, supplicant, wireless radio and Enterprise security system is paramount for the device manufacture.
  - If not secure patient data could become vulnerable.
Driver Development

- What you need to get a 802.11 radio working in a customer’s design

  - **BSP (SDIO stack)**
    - SDIO radios require a functional SDIO interface to the host processor
    - If the SDIO stack is not functional the customer or Silex will need to implement it before the radio will work.

  - **Driver**
    - The driver handles the communication and control of the wireless radio. Drivers need to be ported to specific operating systems and versions of operating systems ie. Linux 2.6.123 may be different then 2.6.124.

  - **Supplicant**
    - The wireless security supplicant handles the encryption needed to get on a secure network.

  - **Testing**
    - To ensure that the driver works as expected, the security supplicant will need to be tested prior to release to the field.
Integrating Wireless with INTEGRITY for Medical Devices

Solving Medical Application Development Challenges
Platform for Medical Devices

Integrated development tools

Commercial OTS safety-certified real-time operating system with extensive middleware

Supporting life cycle data

Complete set of analysis, development, and approval support services

Development Tools

- **Rhapsody**
  - SysML/UML, code generation
- **MULTI**
  - IDE, compiler, debugger
- **VectorCAST**
  - Unit test, coverage analysis

RTOS and Middleware

- **INTEGRITY RTOS**
- **µ-velOSity Microkernel**

Approval Support Data

- **INTEGRITY**
  - Guidelines for usage in medical devices
- **INTEGRITY**
  - Kernel life cycle data, source code
- **MULTI**
  - Confidence from use report

Support Services

- **Training**
  - Tools, standards, best practices
- **Analysis**
  - Product roadmaps, risk management
- **Development**
  - System, software integration, test
- **510(k)**
  - Clearance, Planning and Process Support

© 2011 Green Hills Software, Inc. and Silex Technology, Inc. Slide 14
Platform Productivity and Quality Technologies

- Static Code Analysis
- Compiler technology
- Debugger Technology
- Simulation
- Hardware Probes
Getting the User Interface Right - Early

- A significant number of MDRs (example: infusion pumps) are related to **human factor issues**
- Rapid prototyping and definition of the user interface reduces functional risk
- Early visibility of the interaction will identify potential “operator” related problems
Software Verification and Interoperability Testing

- Testing – Alone, not enough to guarantee a safe and effective device
- Automated Unit Testing
  - Lowest level software item in IEC 62304
  - Typically organized by a source file
  - Generation of stubs and drivers
- Automated Integration Testing
  - Testing an aggregation of SOFTWARE ITEMS
- Test Harness generation for UNITS and Aggregate SOFTWARE ITEMS
- Automated System Testing
  - Script-based, GUI driven, simulation
- Support for Regression Testing – host and target
Wireless and INTEGRITY - Device Manufacturers

- Reduced Development, Testing and Verification Time
  - Driver integration all ready done eliminates engineering development for driver
  - Engineering can focus on specific product features vs. communication functionality and interoperability
  - Driver / OS interoperability Testing and Verification greatly reduced or eliminated

- Faster Time to Market
  - Reduced wireless development time can greatly reduce the time to production
  - A driver development could take 4 – 6 weeks without verification testing
  - Tight integration with the RTOS reduces the development time considerably

- Net Result
  - Reduced Cost and Increased Revenue
Benefits of Wireless Integration with INTEGRITY

Benefits for the Patients Include

- **Improved mobility**
  - Wireless enabled Point of Care devices allow patients to be more mobile while continuing to monitor their condition.
  - Mobility for patients helps diagnosis, enables quicker response for caregivers resulting in quicker recovery
  - Faster and more accurate data collection

- **Accelerated Wellness**
  - Real time data facilitates improved medical decisions resulting in improved patient care and accelerates recovery
  - Allows patients to recovery quicker and return home sooner

- **Reduced Medical Costs**
  - Facilities expenses
  - Insurance costs
  - Patient cost
Benefits for the Medical Facility

- Improved Efficiencies for Caregivers and Facilities

- Increase caregivers efficiencies
  - Wireless enabled devices increase information flow from patients to caregivers.
  - Increased data enables more intelligent decisions.

  - Intelligent decisions not only resulting in improved patient care but also can facilitate improved processes, reduce unnecessary steps and help overall efficiencies.

- Medical facilities reap rewards through increased efficiencies and regulatory compliance
  - Improved patient care from real time data accelerates recovery which reduces patient stays resulting in reduced medical and facility expenses
  - Allows more patients to be treated with less resources while improving care
Contact Information

- Jim McElroy
  - Director, Medical Industry Business Development
  - Jim.mcelroy@ghs.com
  - Web: www.ghs.com

- Mark Prowten
  - Director, OEM Embedded Wireless
  - mprowten@silexamerica.com
Appendix
Green Hills Software

- Innovative Leader in Embedded Industry
  - 37 offices worldwide; HQ in Santa Barbara, CA
  - 28 years in the embedded industry
  - Diverse customer base including aerospace & defense, medical, networking, consumer, industrial, railway & automotive

- Safety, Security, Reliability Experts
  - Continua Health Alliance Member
  - Certified Operating Systems
  - Advanced Development Tools
  - Middleware, BSPs, device drivers
# Green Hills Software - Security and Safety Certifications

<table>
<thead>
<tr>
<th>Certifying Authority</th>
<th>Level Achieved</th>
<th>Applicability</th>
<th>Industry</th>
</tr>
</thead>
<tbody>
<tr>
<td>FAA</td>
<td>DO-178B Level A</td>
<td>Reliability, Safety</td>
<td>Avionics</td>
</tr>
<tr>
<td>EASA</td>
<td>DO-178B Level A</td>
<td>Reliability, Safety</td>
<td>Avionics</td>
</tr>
<tr>
<td>NSA</td>
<td>EAL6+, High Robustness, Type 1</td>
<td>Security</td>
<td>Defense</td>
</tr>
<tr>
<td>FDA</td>
<td>Class II, III</td>
<td>Reliability, Safety</td>
<td>Medical</td>
</tr>
<tr>
<td>TUV Nord, Exida</td>
<td>IEC 61508: SIL 3</td>
<td>Safety</td>
<td>Industrial Automation</td>
</tr>
<tr>
<td>TUV Nord, Exida</td>
<td>EN 50128: SWSIL 4</td>
<td>Safety</td>
<td>Rail, Transportation</td>
</tr>
<tr>
<td>Transdyne Corp.</td>
<td>SEI/CMMI Certified</td>
<td>Quality</td>
<td>All</td>
</tr>
<tr>
<td>IEEE and The Open Group</td>
<td>1003.1 IEEE POSIX Certified</td>
<td>Open, Interoperable</td>
<td>All</td>
</tr>
</tbody>
</table>
INTEGRITY MED – Approval Support Package

- INTEGRITY kernel specific
  - Quality Manual
  - Coding Standards
  - Software Configuration Management
  - Software Requirements Document Standards
  - Software Design Document Standards
  - Software Development Plan
  - Software Verification Plan
  - Software Requirements Document
  - Software Architecture and Design Document
  - Software Test Cases and Procedure
  - Software Problem Report Example
  - Software Change History Example
  - Requirements Traceability Report
  - Software Verification Results Report
  - API Usage Manual

- Change Management Documentation
- Confidence in-use report for MULTI IDE and compilers (optional)
Platform for Constrained Applications

- Bare metal application execution
- Typical for processors with no memory management unit
- High performance
- Fast boot
- Low cost
- Small footprint
- Integrated middleware

Application Software

Micro Kernel

Hardware
Best Practices in Medical Software Development

- IEC 62304 provides a strong domain-specific framework of processes, activities, and tasks
- Ensure safety, security, reliability
  - Certified operating system platform
  - High assurance development tools
- Leverage automation to improve productivity
  - Tools that speed up the development and debug cycle while ultimately improving product quality
- Utilize COTS “Approval” Documentation
  - Software of Unknown Provenance (SOUP)
  - Evidence: test plans, procedures, results