Designing Embedded Devices for Bluetooth Compatibility

Using Bluetooth qualified components during the design process will help reduce cost and time-to-market.

By Dan Jerrestam and Staffan Bernström, Enea Realtime and Eva Skoglund, OSE Systems

A systematic approach is needed when building products with Bluetooth capabilities, as a number of obstacles must be overcome. The product has to live up to the Bluetooth standard and pass the pertinent mandatory test cases. In addition, the Bluetooth components must be integrated onto a selected platform consisting of hardware and software. During product development it must be possible to verify the Bluetooth functionality within the platform and in the peer-to-peer communication found between the layers of the Bluetooth stack.

So what is a good strategy for devising Bluetooth enabled products? If Time-To-Market (TTM) is important for the product, then the Bluetooth components should be purchased from a third-party vendor. Today most Bluetooth components are available on the market as pre-qualified products. A pre-qualified component does not need to be tested again with the exception of re-testing some parameters in the embedded solution, e.g. as spurious emission. However, any application-specific behavior must be developed for the Bluetooth enabled end-consumer product, and the Bluetooth standard compatible components must be integrated with the platform.

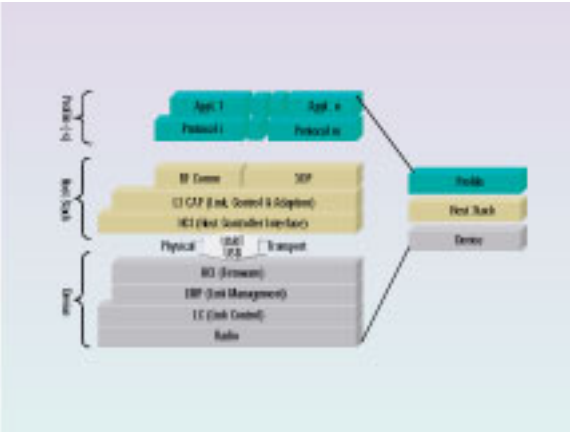

Figure 1. (top) A complete Bluetooth stack is shown with the different layers needed for a Bluetooth implementation in a 2-CPU system. The bottom level, depicted in gray, shows the Bluetooth device, which resides in one of the CPUs of a 2-CPU system. This level has Bluetooth packets and handles links between nodes in the Bluetooth network. The intermediate layer contains the host stack and is responsible for links and formatting data into dedicated Protocol Data Units (PDU) multiplexed on

top of the link setup by the Bluetooth device. In addition, information on the services used (referred to as profiles using the Bluetooth terminology) are exchanged using Service Discovery Protocol (SDP). The top layer depicted in green contains the applications and protocol components that either use or offer services as defined by the Bluetooth standard.

The Development of the Product

Product architecture

After the design team has selected a platform consisting of hardware and software (including the operating system), the Bluetooth components must be integrated onto this platform. Figure 2 shows some of the basic components that must be included for creating the final product.

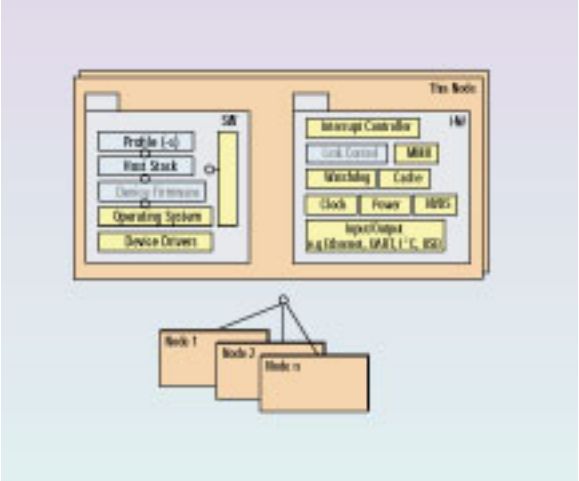

Figure 2.

The packages needed on the Bluetooth-enabled node, as viewed from a software developer's point of view, are visualized within the figure. These packages are divided into software and hardware components. The Bluetooth components are marked in light blue. Note the dashed components, *Device Firmware and Link Control*, are only needed if a 1-CPU solution is devised. The other boxes indicate other nodes to which *this node* has established Bluetooth links. The circles denote Bluetooth-related interfaces that are of great interest for optimal product development.

The selected hardware interfaces of the hardware components must all be considered for obtaining the services needed. The *Interrupt Controller* provides support for all events in the hardware, including an interrupt controlled *Link Control* of Bluetooth. Soft interrupts are also handled here, e.g. timer activities. The Memory Management Unit (MMU) coordinates basic handling of memory as well as more advanced memory management, e.g. partitioning of memory. The *Watchdog* allows the system to be restarted if a fatal error occurs. To minimize the power consumption, a *Power* module is needed for changing between different power states. Configuration data including Bluetooth parameters need to be stored in a persistent storage, Non Volatile Data Storage (*NVDS*). Finally, in most cases there is a need for *Input/Output* support. For example, it is convenient for the LAN Access Profile (in the gateway role) and the Personal Area Networking (PAN) Profile (acting as Network Access Point) to have Ethernet capabilities to provide a connection to external Ethernet-based networks. Such capabilities also provide a mechanism for the physical transport of HCI packets between the *host stack* and the Bluetooth *device*. UART or the RS232 port is often used for configuration purposes.

The software consists of Bluetooth components, components provided by the operating system vendor (the *Operating System* and in most cases the *Device Drivers* and possibly some of the *Services*), and possibly components that are developed by the device-designer's organization. Some operating system vendors also provide the Bluetooth software components. For example, OSE Systems license a Bluetooth Host Stack from Ericsson and provides this integrated to the real time operating system OSE. If the product is a 1-CPU solution, then at least one of the device drivers must be allocated for the link control and the *Device Firmware* must contain the functionality of the Link Manager Protocol as specified in the Bluetooth standard. The *Services* component may contain adaptations for integration of the *Host stack* on the system or other general services such as Internet protocols or an embedded file system or configuration data management.

Figure 3. Once the BQB has approved the product it may be marked with the Bluetooth logo.

From this overview it is obvious that the complexity and the possibility for failure of the development cycle is greatly reduced when a large number of components are purchased from third-party vendors.

Development Support

Once product development begins, a number of tools are needed to aid in the delivery of a stable and bug-free system. Categories include, design tools, compilers, linker, debuggers, logging, system monitor and tools that monitor externally used protocols. Examples of design tools are Rational Rose&#153 using UML and SDL-based tools. A decent system monitor tool, such as OSE's Illuminator&#153, will observe processes, the state of a process, the memory usage and active messages within the system.

Bluetooth-related tools fall mainly within the categories of logging and tools that monitor the Bluetooth protocols over air. It is suggested that the Bluetooth components shall contain logging possibilities at the following levels:

&#149From/to upper components relative to the host stack

&#149From/to lower components relative to the host stack

&#149If a 1-CPU Bluetooth solution is used then logging of what occurs between *Device Firmware* and *Link Control* is important, i.e. how the software handles the baseband for Bluetooth

&#149There are also interactions between the Bluetooth components and the *Services* component that need to be verified

Monitoring packets over air allows verification of the inter-working between different Bluetooth nodes. But just a complete log of all packets exchanged within members of a Bluetooth piconet is not practical. The numbers of packets that need to be analyzed are just too many. Additional requirements on such a monitor may be:

&#149PDU must be possible to visualize on different protocol levels, e.g. L2CAP and

SDP
&#149Simple filtering on packet types must be possible
&#149Triggering on a selected PDU
&#149Scripting for automated analysis of the outcome of test cases

Verification of the Bluetooth related functionality

Once the different Bluetooth-related interfaces for logging and monitoring are implemented in the product, the process for verification for compatibility with the Bluetooth standards may begin. If there are Bluetooth components that are not pre-qualified within the product, then it must pass through the applicable mandatory test cases. Test cases are created to verify the component by logging it onto relevant interfaces.

If only the profiles are to be tested, then the pertinent interoperability tests need to be verified. Before these tests commence, the integration of the third-party Bluetooth components must be verified. That may be done by logging some supported use-cases that match that of the final product, e.g. configuration, connecting Bluetooth devices, inserting service descriptions, searching for services and attributes of services, connecting over L2CAP or RFCOMM or OBEX and so on. After completing the tests at your own site, the Bluetooth qualification process must be performed.

Qualified for Bluetooth

The Bluetooth qualification process ensures that products comply with the Bluetooth specification. Only products that pass the Bluetooth qualification process are entitled to use the free license to the patents required to implement Bluetooth wireless technology, and to use the Bluetooth brand. By ensuring that products comply with the specification, the qualification process helps to make certain that Bluetooth products will work reliably, and that they will work with other Bluetooth products.

The Bluetooth Qualification Body (BQB) checks a Member's product declarations and reviews test plan/test reports filed in the Compliance Folder. The BQB has the authority to list products without further review. A list of approved BQBs is provided under &quotdeveloper information/ qualification program&quot at www.bluetooth.com.

The development cycle and financial commitments can be reduced in the qualification procedure by involving a BQB well before the planned product launch. This ensures that interoperability testing is scheduled, any remaining end-product conformance tests are completed, and the necessary documentation is filed before the product is ready for launch. However, in addition to the qualification tests, a product must fulfill the regulatory requirements in the countries where it will be sold.

The Bluetooth qualification testing is based on the concept of testing against a validated reference test system (conformance testing), and functional testing against another operational Bluetooth product (interoperability testing). In the future, the conformance testing of the protocol stack will be performed using a validated reference test system. Until then, conformance testing is performed by other means. Members are free to select appropriate test equipment to run the required tests. To strengthen confidence in the lower layer interoperability, there will be mandatory protocol interoperability tests executed with the aid of Blue Units

(BU). BUs are designated protocol test products used in absence of a validated test system. Testing using BUs will be phased out as the validated protocol conformance test capability becomes available.
The Bluetooth test specifications and BU test specification lists 575 active test cases with all mandatory and optional functionality supported (see Table 1).

When the supported profiles are chosen for a specific product, the functionality of the profiles must be decided. To evaluate a particular implementation, it is necessary to possess a statement of capabilities and profiles. This statement is called an Implementation Conformance Statement (ICS). The number of test cases performed depends on the supported optional and mandatory functionality shown in the ICS and determined by a test case mapping.

Bluetooth Device

A Bluetooth device consists of the Bluetooth radio hardware and the lower layers of the protocol stack, Baseband (BB) and Link Manager (LM), up to HCI.
Complete pre-qualified Bluetooth devices are available as qualified components on the Bluetooth Qualification Web site. This alternative minimizes the tests that need to be performed. It is important to select a component which fulfills the requirements of the end product, e.g. supported features and temperature range.
It is also possible to build a Bluetooth device of a qualified radio chip or chipset in a reference design. The radio parameters are affected by external components and implementation, such as the oscillator, antenna, capacitors and PCB material. The radio conformance tests must be performed, but access to and analysis of the test results of the qualified component may reduce the Bluetooth qualification testing, such as some tests under extreme temperature and voltage variations. The pertinent tests to be performed are certified by the BQB. Any hardware change in the reference design will require complete re-testing. When selecting a qualified software component for the Device, it is convenient to select a software component qualified in conjunction with the specific radio that will be used.
The radio link conformance testing can be performed using a combination of standard test equipment and special Bluetooth test equipment. There are 16 active tests cases on the radio, 168 for the lower layers protocol stack, BB and LM and 14 additional test cases defined for the BU (each test must be performed against two different BUs).

Bluetooth Host Stack

Bluetooth host stacks are available as qualified components on the Bluetooth Qualification Web site. When selecting a qualified host stack it is important to ensure that the stack is designed to run in the specific operating system (OS) and hardware platform that will be used. In the case of the OSE real-time operating system, the Bluetooth Host Stack interface to the OS is directly designed for the OSE interface, which ensures a highly reliable integration. If a portable host stack is selected, some general requirements must be fulfilled, such as OS independence, hardware independence (endian independent, alignment independent), and compiler independence.

The number of test cases depends on the supported functionality of the protocol but in all there are 88 active tests defined. The Bluetooth Network Encapsulation Protocol (BNEP) with adherent profile PAN is one of the latest contributions.

Two profiles require conformance testing, Generic Access Profile (GAP) and Serial Port Profile (SPP). The number of test cases depends on the supported functionality of the profile and currently 54 test cases exist.

Profiles

In addition to testing the protocol stack, the implementation of the profiles has to be tested to ensure that products implementing the same profile will interoperate. Profile interoperability testing is a means to increase confidence for having interoperable Bluetooth products in the field. In order to ensure that interoperability tests are repeatable and consistently performed, a set of listed products, Designated Profile Interoperability Testers (DPIT), will be identified for use during interoperability testing. The objective is to build a pool of DPITs that will be used for testing each of the Bluetooth Profiles. Initially the requirements for profile interoperability testing depend on availability of DPITs for the profiles implemented by the product. In this situation the member will be required to provide engineering evidence of having met the criteria of the defined interoperability tests. Some products may require engineering evidence for profiles implemented where no sufficient number of DPITs exists and testing against DPITs for those profiles where sufficient DPITs do exist.

The number of test cases to be performed depends on chosen profile(s) and the supported functionality. In total there are 235 profile interoperability active test cases defined.

All interoperability test cases must be performed at the end product level even when a pre-qualified profile is chosen.

Summary

Figure 4 provides an example on a product that includes the LAN profile. Depending on whether the components in the Host stack and the Device are pre-qualified or not, the resulting maximal number of test cases are summarized. The example assumes that all features in the LAN profile are supported, thus 10 interoperability test cases need to be performed.

Figure 4. An example on a product that includes the LAN profile.
The upper test case shows the number of test cases when all pertinent functionality is supported.
It is assumed for the case in the middle that only the protocol layers SDP and L2CAP are pre-qualified. In the Device the lower layers protocol is pre-qualified in conjunction with the physical layer, which is made of a qualified radio chip in a reference design.
The number of re-tests for the case with only pre-qualified components (bottom) depends on the integrated components. The BQB decides upon the re-tests to be performed. When Host stack and Device are selected to minimize the re-tests for some applications, then only the test case &quotRadiated Spurious Emission&quot has to be performed. This test case will be re-used for the regulatory requirements in the US and EU.
It is clear from this example that the Bluetooth qualification process may be very costly if Bluetooth qualified components are not used. It is strongly recommended that the designer or device developer/manufacturer contact a BQB for guidance early in the development phase. It is also advised that the Bluetooth-related functionality is verified before the conformance and interoperability tests commence. Manufacturers with requirements on a short TTM should use qualified components as much as possible to reduce the development time and decrease the number of test cases that must be passed. This can be achieved by the inclusion of some well-placed test points within the product that will decrease costs and time spent for conformance and interoperability testing.

Source URL (retrieved on *03/09/2014 - 4:46pm*):
http://www.wirelessdesignmag.com/product-releases/2002/01/designing-embedded-devices-bluetooth-compatibility?qt-blogs=0