



In the embedded market, the size, cost, memory consumption, CPU usage, and power consumption of a device are critical to determining its commercial success.

The BLUEmagic 3.0 protocol stack can be optimized for an individual application by means of the BLUEmagic 3.0 software development kit (SDK), ensuring that the resource consumption of the embedded Bluetooth device is minimized while device performance is maximized. The approach taken by Open Interface North America contrasts with that taken by many of its competitors, whose current protocol stack products were originally designed for use in personal computers, in which space, memory, CPU, and power resources are not as limited as in embedded devices.

In a market that is moving away from the desktop and toward distributed, application-specific, resource-constrained embedded systems, the smart choice is BLUEmagic 3.0 software, the solution that was engineered from the start with embedded Bluetooth devices in mind.

The BLUEmagic 3.0 embedded Bluetooth protocol stack is a thorough implementation of a fully-configurable, resource-efficient protocol stack targeted at embedded Bluetooth devices and applications. The BLUEmagic 3.0 protocol stack is scalable, modular, and portable, with an innovative, high-performance, and compact architecture targeting embedded devices.

#### OUR GOAL

The goal of Open Interface North America is to provide application developers, silicon vendors, and system integrators with the tools necessary to rapidly develop devices using embedded Bluetooth wireless technology without burdening the developers with learning the Bluetooth protocol or bloating the product with functionality that it will never use.

The BLUEmagic 3.0 embedded Bluetooth software development kit is easy to learn and easy to use, giving full access to all configurable aspects of the BLUEmagic 3.0 protocol stack and enabling the production of robust application software.

All of these features result in users of BLUEmagic 3.0 software producing better, smaller, more easily maintained embedded Bluetooth products that reach the market faster and at lower development cost.

Open Interface North America has extensive experience developing wireless connectivity software, including development of the BlueMagic 2.0 protocol stack, the first Bluetooth protocol stack to be included in a commercially available product. This experience has given Open Interface North America the perspective necessary to build the BLUEmagic 3.0 protocol stack from the ground up, finely tuned to the embedded Bluetooth device market.

With so many combinations of processors and operating systems already supported, BLUEmagic software is uniquely adaptable to whatever the future of Bluetooth technology might bring.

#### Supported specifications and guidelines

- Bluetooth v1.1, v1.2, and v2.0 + EDR
- Security White Paper Security Modes 1, 2, and 3
- Car - Communication - Application - Promotion (CCAP)
- Home Printing with Mobile Terminals by MIPC

#### Available Bluetooth profiles and additional components

- GAP
- SPP
- SDAP
- Synch
- FTP
- Fax
- DUN
- PAN
- BNEP
- HID
- HCRP
- PBAP
- BPP
- BIP
- VDP
- Cordless Telephony
- Intercom
- GenOBEX
- Object Push
- Hands-Free (up to v1.5)
- Headset
- SIM Access
- TCS Binary
- A/V Control Transport
- A/V Distribution Transport
- A/V Remote Control
- Advanced Audio Distribution
- Generic A/V Distribution
- SBC and eSBC
- CSS and XML parsers

#### BLUEmagic 3.0 core stack

- Policy Manager
- Device Manager
- Security Manager
- Memory Manager
- CThru™ Dispatcher
- SDP Client
- SDP Server
- L2CAP
- RFCOMM
- HCI

#### Supported transports

- USB
- UART H4, H5, and BCSP

#### Examples of embedded target environments supported

- embedded Linux
- embedded Windows®
- µITRON
- ATI Nucleus™ Plus
- Express Logic ThreadX®
- Red Hat eCos™
- µCOS
- Green Hills INTEGRITY®
- Zeevo BlueOS™
- GCT Vincent OS™
- QNX®

Support is available for other real-time operating systems upon request.



OPEN INTERFACE

## Competitive advantages

### Scalable

BLUEmagic 3.0 software is built around a streamlined Bluetooth protocol stack and support modules. This compact kernel consists of RFCOMM, SDP Client, SDP Server, L2CAP, and HCI protocol layers; Policy Manager, Device Manager, Security Manager, and Memory Manager modules; and the CThru Dispatcher callback manager. Implementations of this kernel consume approximately 64 kilobytes of ROM when configured for certain common Bluetooth applications.

More important, even, than code size in determining the cost of an embedded device is RAM usage. BLUEmagic 3.0 software uses a unique pooled dynamic memory management system that prevents memory fragmentation and greatly diminishes an application's physical RAM requirements. One common memory heap is used for all of an application's RAM needs, except for the function call stack. Application and protocol stack memory are kept separate, resulting in more predictable behavior from the protocol stack and allowing the application to use a memory scheme separate from that used by the stack. The application developer can measure the RAM usage of a particular application and gauge precisely how much RAM to place onboard, tuning the size of the allocated memory heap in software by means of a single parameter. No operating system memory management support is required.

Using BLUEmagic 3.0 software, Open Interface North America has successfully built and tested Bluetooth applications that require less than 4 kilobytes of RAM for operation. The size of internal data structures, the number of connections and groups, and the number of sessions and links needed by the application can be configured by means of a configuration parameter file without recompiling the application source code or the protocol stack library.

### Modular

The BLUEmagic 3.0 protocol stack is built from small modules separated by well-defined interfaces and inter-module communication. This modularity is based on principles of object-oriented design and data encapsulation.

The BLUEmagic 3.0 SDK allows the application developer to link in only the library files for the modules needed by a specific application instead of wasting valuable memory with code that would be excess baggage for a particular application.

BLUEmagic 3.0 profile modules even have the ability to automatically load and unload themselves as appropriate to conserve memory.

This modularity greatly improves ease of maintenance, since a developer need be concerned only with altering the modules directly relevant to a particular maintenance effort, not with unintended consequences that this might otherwise have in other modules.

Modularity adds to the value of a protocol stack at the edges of the HCI transport layer. By carefully isolating the HCI transport layer, the BLUEmagic 3.0 protocol stack provides for quick and simple substitution of drivers and integration with Link Manager APIs.

### Portable

The BLUEmagic 3.0 protocol stack is coded entirely in ANSI C; the API exposed by the BLUEmagic 3.0 SDK therefore allows a programmer to work entirely in C for application development. This also ensures that BLUEmagic 3.0 software is portable to various 16-, 32-, and 64-bit environments with very little effort.

The only platform-specific functions that exist within the BLUEmagic 3.0 protocol stack are simple functions for reading the system clock and low-level system I/O. Altering these functions for a particular platform is a trivial task. Porting the BLUEmagic 3.0 software to a new platform is estimated to take days rather than months; your 16-, 32-, or 64-bit target environment should be no exception. As mentioned above, isolation of the HCI transport layer provides for quick and simple substitution of drivers and integration with Link Manager APIs.

The BLUEmagic 3.0 protocol stack may even be run on a bare board, with no operating system.

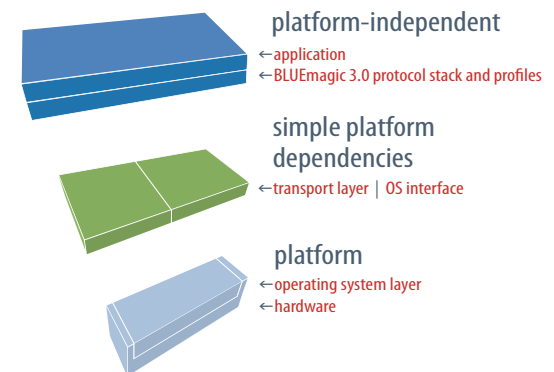
#### Out-of-the-box application development

The BLUEmagic 3.0 SDK comes complete with a set of application prototyping tools, including a set of binary library files allowing the BLUEmagic 3.0 protocol stack to operate on a development system running Linux, Microsoft Windows 2000, or Microsoft Windows XP. This tool set gives the developer a head start on designing and prototyping an application before the BLUEmagic 3.0 protocol stack has been ported to the specific target hardware, or even before the hardware itself has been completed.

#### Debugging and quality assurance

A guiding principle in the design of BLUEmagic 3.0 software was "design for testability".

The BLUEmagic 3.0 SDK provides debugging tools that enable runtime argument checks, memory corruption checks, and memory leak checks. All runtime checks can be disabled in the production build of an application to ensure minimal footprint and optimal performance. During application development and debugging, however, such checks are invaluable to developers.



### Easy to learn, easy to use, easy to maintain

The BLUEmagic 3.0 SDK is easy to learn and easy to use, reducing time to market for embedded Bluetooth devices built with the BLUEmagic 3.0 protocol stack.

Application-specific configuration of the BLUEmagic 3.0 protocol stack is accomplished through a well-documented application programming interface (API) consisting of header files coded in ANSI C. The BLUEmagic 3.0 low-level API provides full configuration access to virtually all elements of the protocol stack, allowing developers of deeply embedded Bluetooth applications to make efficient use of memory and processor power for devices with tight resource

constraints. Application developers programming for host processors without such tight constraints may choose to use the optional BLUEmagic Host Application Programming Interface (BHAPI) software, which abstracts common Bluetooth profile functionality out to the level of simple interactions with Bluetooth services. Very little knowledge of the Bluetooth protocol is necessary to develop an embedded Bluetooth application with BHAPI software.

Neither the BLUEmagic 3.0 protocol stack libraries nor the profile libraries need to be recompiled to change the configuration of the protocol stack and profiles.

## Product features

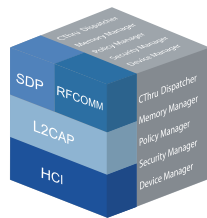
### Core stack

The BLUEmagic 3.0 software kernel is compliant with the Generic Access Profile and Serial Port Profile specifications and has been certified as meeting the requirements of the Bluetooth Qualification Process.

The Device Manager, Security Manager, Policy Manager, CThru Dispatcher, and Memory Manager are support modules, providing services accessible by the SDP, RFCOMM, L2CAP, and HCI modules of the core protocol stack and considered part of the BLUEmagic 3.0 software kernel.

The Device Manager is a module that provides a higher level of abstraction above the Host Controller Interface layer and collects many device configuration functions into one module. The Device Manager API provides interfaces for performing device discovery, managing connections, reading and writing device configuration, and informing applications of certain asynchronous HCI events that may be of interest. The Security Manager is a module that implements a security architecture similar to that outlined in the Bluetooth® Security White Paper. Security modes 1, 2, and 3 are supported.

Memory space and CPU cycles are conserved by the core protocol stack's implementation of a zero-copy scheme that prevents unnecessary copying of data as it moves up and down the protocol stack.



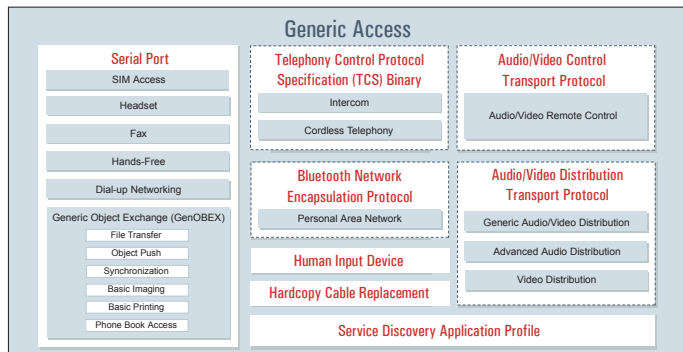
#### BLUEmagic 3.0 core stack

- RFCOMM
- SDP Client
- SDP Server
- L2CAP
- HCI
- Policy Manager
- Device Manager
- Security Manager
- CThru Dispatcher
- Memory Manager

### Profiles and optional protocols

BLUEmagic 3.0 is the world's first Bluetooth protocol stack qualified for Bluetooth Specification v2.0 + EDR and has also received Bluetooth qualification for all standard Bluetooth v1.1 and v1.2 profiles and protocols. Reference designs demonstrating these profiles are available upon request.

Bluetooth profiles and optional protocols



Profiles are depicted as being contained within profiles and protocols on which they depend. Protocols are distinguished by dashed-line boundaries.

### Single-threaded CThru™ architecture

In attempting to design the best possible embedded Bluetooth protocol stack, Open Interface North America considered not only which features were needed but also which features of past Bluetooth protocol stacks could be discarded. Since the BLUEmagic 3.0 protocol stack is targeted at single-threaded embedded Bluetooth systems, dependence on external system services for threading and memory management was eliminated.

In a synchronous architecture, when a calling function issues a call to a function, the calling function can be "blocked", meaning that it waits for a return value from the called function before proceeding. Significant delays can occur when a calling function is waiting for a called function

to execute a lengthy sequence of operations and return a value, especially when the called function may fail to ever complete its execution.

The BLUEmagic 3.0 protocol stack is built around the asynchronous, single-threaded, command/event-oriented CThru architecture, which is implemented consistently throughout the protocol stack. API functions "call through" the protocol stack layers without blocking, relying on the CThru Dispatcher to manage the sending of events back up the protocol stack. The CThru Dispatcher effectively acts as a task scheduler for all layers of the protocol stack. The overhead for this architecture implementation and the CThru Dispatcher is small, with low memory and CPU requirements.

This innovative architecture eliminates the memory and performance overhead caused by context switching, allowing the BLUEmagic 3.0 protocol stack to operate with less physical memory and at greater speed, which means that devices using the BLUEmagic 3.0 protocol stack can be smaller, faster, and cheaper than they would be otherwise.

Application developers may still use multiple threads in their designs, but the CThru architecture and the low-level BLUEmagic 3.0 API make it possible to avoid doing so when desired.

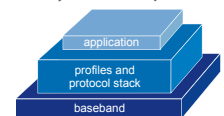
Application developers programming multi-threaded applications for capable host devices may choose to take another approach, using the simple, synchronous, thread-safe BLUEmagic Host Application Programming Interface (BHAPI) software, an optional service-level wrapper around the profile interfaces.

### Various system architectures supported

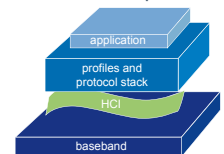
For reasons as varied as development cost, production timelines, developer expertise, available software, hardware choices, technology maturity, and product form factor, different device manufacturers will choose different system architectures for enabling their products with Bluetooth technology. BLUEmagic 3.0 software can accommodate them all.

The system-on-chip architecture is the ultimate in Bluetooth integration, wherein the incremental cost of adding Bluetooth functionality drops to its lowest, since no additional processor is needed. The Bluetooth software runs together with the application on one chip. This architecture requires the Bluetooth software to be very compact. The BLUEmagic 3.0 protocol stack has already been integrated into system-on-chip devices.

System on chip (SOC)

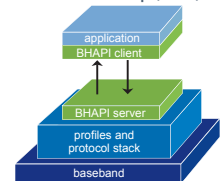


Host controller implementation



The host-controller architecture has been quite common during these early days of Bluetooth technology adoption. This architecture relies on a well-implemented HCI layer and a variety of drivers for the transport across the HCI interface. BLUEmagic 3.0 software comes with a complete set of drivers for USB and UART (BCSP, H4, H5, 3-wire) serial transport.

Stack on chip (StOC)



The stack-on-chip architecture places the Bluetooth protocol stack on a separate processor from the application. This diminishes the burden on the host processor. The BLUEmagic 3.0 stack-on-chip solution places all Bluetooth software on the baseband chip and is capable of full data rates, unlike competing host solutions that burden the host with the profile code and are capable of only diminished data rates.

## Services and product components

### Testing, service, porting, and support

Unlike other providers of Bluetooth wireless technology software, Open Interface North America provides more than just a software drop. Open Interface North America is committed to providing documentation, training, support, testing, application customization, codecs, porting, and other services to its customers. This full range of services allows customers of Open Interface North America to concentrate on doing what they do best, whether it is application development or system integration.

Open Interface North America provides users of BLUEmagic 3.0 software with anything they need to develop better, smaller, more easily maintained embedded Bluetooth products that reach the market faster and at lower development cost.

### BLUEmagic 3.0 software has already been ported to the following operating systems, processors, and development tools:

CPUs: x86, Motorola DragonBall 68k, Hitachi SH-4, Hitachi SH-3, Hitachi SH-2, ARM7, ARM9, StrongARM, ARC, MIPS, GCT GDM1202, Renesas M32C, PowerPC, National Semiconductor CR16, and other proprietary processors

Operating systems: Linux, embedded Linux, Microsoft Windows 2000 and Windows XP, Microsoft Windows CE 3.0, embedded Windows,  $\mu$ TRON,  $\mu$ COS, Green Hills Integrity, Express Logic ThreadX, ATI Nucleus, Red Hat eCos, GCT Vincent OS, Zeevo BlueOS, OSE, bare board (no operating system), and other proprietary operating systems

Development tools: GCC, Microsoft Visual C++ (version 6.0 with Service Pack 4, .NET or .NET 2003), Embedded Visual Tools 3.0, Hitachi Embedded Workshop (HEW), Green Hills MULTI, Texas Instruments, MetaWare, IAR Embedded Workbench, ARM Developer Suite (ADS), Renesas NC308, and other proprietary development tools.

### BLUEmagic Host Application Programming Interface (BHAPI) software

The BLUEmagic 3.0 SDK or source code package may be ordered with or without the BLUEmagic Host Application Programming Interface (BHAPI) software package.

### Components of the BLUEmagic 3.0 SDK

The BLUEmagic 3.0 embedded Bluetooth software development kit includes the following components:

- installation guides for BLUEmagic 3.0 software running under Linux, Microsoft Windows 2000/XP, and any embedded target platforms that were ordered separately
- user's guide and reference documentation for the BLUEmagic 3.0 protocol stack and SDK
- USB and UART Bluetooth device driver library and installation files for Linux and Windows 2000/XP

- BLUEmagic 3.0 Bluetooth protocol stack and profile library files for Windows 2000/XP and Linux
- BLUEmagic 3.0 Bluetooth protocol stack and profile library files for embedded platforms ordered separately
- API and configuration files as fully commented C header files
- make utility and Makefile, project files, and other files necessary to build sample applications for specific platforms
- sample implementations of platform specific files
- executable files and platform-independent source code for sample applications

The FTS for Bluetooth virtual sniffer from Frontline Test Equipment, Inc. is incorporated into the BLUEmagic 3.0 SDK. This protocol analyzer can be used on a Windows XP/2000 development system to debug and develop applications written using BLUEmagic 3.0 software.

The baseline BLUEmagic 3.0 SDK includes the components of the SDK for application prototyping on a development system running Linux or Microsoft Windows 2000/XP. Additional target platform support is included with a Platform-Support Package (PSP), which is bundled with the baseline SDK per customer request.

The BLUEmagic 3.0 Evaluation SDK includes only the components of the SDK for application prototyping on a development system running Linux, Microsoft Windows 2000, or Windows XP. The Evaluation SDK is distributed with software that limits the run time of applications built with BLUEmagic 3.0 software.

### Components of the BLUEmagic 3.0 source code distribution

The BLUEmagic 3.0 source code distribution package includes:

- source code for all elements of the BLUEmagic 3.0 embedded Bluetooth protocol stack, transport modules, and profiles
- makefile files, project files, other files necessary to build binary library files, and executable files
- additional source code reference documentation

The BLUEmagic 3.0 source code distribution package also includes the entirety of the BLUEmagic 3.0 SDK, with the exception of some pre-built binary files.

### Additional tools supporting development with BLUEmagic 3.0 software

The BLUEmagic 3.0 SDK and source code packages are distributed with an accompanying collection of software tools, which may be of use to Open Interface North America customers in their development efforts. These tools may be subject to license agreements other than those imposed by Open Interface North America.

## Open Interface North America, Inc.

506 Second Avenue, Suite 420, Seattle, WA 98104 USA

Tel: +1-206-315-5570

Fax: +1-206-315-5580

[www.oi-us.com](http://www.oi-us.com)

[www.oi-direct.com](http://www.oi-direct.com)

[info@openinterface.com](mailto:info@openinterface.com)

Open Interface North America, the double circle device, BLUEmagic, BLUEmagic Software, BLUEmagic SPP, BLUEsleuth, BLUEtusk, BHAPI, CThru, SOUNDabout, and The Magic of Connection are trademarks or registered trademarks of Open Interface, Inc. or Open Interface North America, Inc. Bluetooth and the Bluetooth design are trademarks of Bluetooth SIG, Inc. and are used under license. All other product and company names are used for identification purposes only and may be trademarks or registered trademarks of their respective owners.