



Whitepaper

The Advantages of Pre-programmed Bluetooth & Wi-Fi® Modules for IoT Developers

Parker Dorris

Senior Product Manager, Silicon Labs





The Advantages of Pre-programmed Bluetooth & Wi-Fi® Modules for IoT Developers

By Parker Dorris, Senior Product Manager, Silicon Labs.

The most successful IoT products make Bluetooth and Wi-Fi connectivity easy for end customers to setup and use but delivering that ease-of-use presents significant design challenges for developers of smart home appliances, sensors, electric vehicles, and other connected products.

The lack of time and resources in companies with limited in-house wireless design expertise can lead to slipped delivery schedules and multiple product re-designs. The four typical stages of the wireless development process are software/firmware development (including embedded coding and debugging), hardware design and certification, mobile app development, and cloud connectivity. The development process can take up to a year to complete, and each stage has its own hurdles to overcome. This whitepaper discusses the unique challenges presented in each development stage and highlights how leveraging pre-programmed, pre-certified wireless modules drastically streamlines the product development cycle for IoT developers.

Developers today want to add rich Wi-Fi and Bluetooth connectivity to their products to enhance their applications and end-customer experience, but they face many challenges that can slow down the process and delay time to market. Gaining expertise in wireless connectivity requires careful study of specifications, trial-and-error prototype builds, and team coordination; the time in research and development cycles could otherwise be spent making a product truly differentiated in its market. For instance, Bluetooth design requires expertise in several areas of development including hardware, firmware, mobile, and cloud. With the increase in data-security issues today, there are more rigorous testing and certification requirements to ensure safe and secure connections (authentication, encryption, pairing, etc.). Form factor issues, such as antenna placement, are also critical for connectivity development because developers have limited space as devices get smaller and smaller.

Once a hardware design has taken shape, the required certification process can be time consuming and expensive and in some countries not even an available option. In addition, the lack of technical support during the development process often adds stress, vendor costs, and delays if developers need to contact various vendors within the Bluetooth stack. Overcoming these and other challenges results in a typical development cycle that can take 12 months or more. Many companies can't staff all the related, but distinct, areas of expertise that go into achieving reliable, robust, and certified connectivity inhouse and need to outsource these efforts. This adds contractor coordination and costs to the already growing list of development challenges to manage.

Index

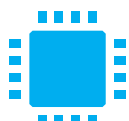
- Wireless development stages and common challenges
- Benefits of pre-programmed wireless modules
- Value of Gecko OS
- Silicon Labs Bluetooth Xpress Pre-Programmed Modules
- Silicon Labs Wi-Fi Xpress Pre-Programmed Modules
- Getting Started with Wireless Xpress Solutions
- Wireless Xpress Bluetooth Starter Kit
- Wireless Xpress Wi-Fi starter kit
- Summary

Wireless development stages and common challenges

Wireless development has many inherent challenges and each stage has its own obstacles. Use-case specific issues, evolving standards, and ‘future proofing’ concerns mean that developers must design proactively to ensure their connected products will function properly. A typical development timeframe for adding wireless connectivity can be approximately 3-6 months for each of the four typical stages of the process:



Firmware Development



Hardware Design



Mobile app Development



Cloud Connectivity

Firmware development

In the firmware development stage, developers using unprogrammed modules must become experts in Bluetooth communication or Wi-Fi protocols and vendor-specific software stacks. In traditional Bluetooth and Wi-Fi design, developers must create an embedded host + network co-processor design with a communication link that operates at a low level controlling the network co-processor. Half of the development work includes writing firmware code and the other half is spent on testing. Testing itself has its own challenges as developers need to try to predict use cases that would affect functionality, test many scenarios to see if the application functions properly, and then troubleshoot and fix bugs which can take up valuable time and resources.

Hardware design

Choosing the right hardware is a critical piece of wireless functionality and the integrity of system design. Using unprogrammed modules to add wireless connectivity to their products poses a variety of problems including potential delays, antenna design issues, and RF certification hurdles. FCC certification alone can cost thousands of dollars and take months of testing and validation. Good RF performance is a critical design challenge and Table 1 shows a multitude of hardware components/factors that can potentially impact RF performance.

RF Performance Factor	Technology
Antenna type, supplier, and placement	Antenna placement, type, material composition, manufacturer, and cost can change signal gain to the matching network resulting in mismatch and poor performance.
Antenna trace shape and length	Minor variations in shape and length can change the expected signal energy and therefore the recommended matching network
Board manufacturer	Differing distances or insulation material between layers, PCB via materials, trace widths, screw holes, etc. can have effects.
Component suppliers	Different suppliers’ components can behave differently and result in different performance. This can result when designers use the ones they have ‘on the shelf’ versus the recommended supplier or try to save a few pennies with a cheaper alternative.
Component types	Different component technologies can affect received power and voltage (e.g., wire-wound resistor vs. thin-film).
Plastics and screw location	Screw placement can have coupling effects for both radiated and received energy.
Battery location	Battery location and technology can affect signal power. A charging battery can also be an unknown player.
Display location	Like batteries, displays can create interference on the antenna.

Table 1: Hardware component impacts on RF performance

Antenna integration and optimization to minimize interference and coupling are two of the most formidable challenges to delivering a successful design, especially for companies without in-house RF expertise. Selecting the best antenna for the application/device can be complicated and then configuration and layout can add a lot of time delays due to trial and error cycles before developers optimize the antenna functionality. A PCB will need many tweaks to optimize antenna performance and also help from a PCB manufacturer. For SoC design, antenna-optimization work requires specialized RF engineer experts with an RF certification, special lab equipment, and an RF-isolated testing environment. Even RF engineers can spend weeks optimizing antennas for Rx/Tx performance and low BOM costs due to keep out zones, component selection, placement, etc. Reducing unpredictable RF performance and reducing the development time frame are key considerations from a hardware perspective. Selecting a wireless module that's already certified solves most of these hardware-development challenges and saves developers many months of prototyping work.

Mobile applications

The mobile app development stage is often the most challenging for companies since many don't have developers in house with mobile application experience. For this development stage, developers must become experts in both Android and iOS development, which means more APIs to understand. They often outsource to vendors who build the mobile infrastructure, perform testing, etc. which can be very time consuming and costly due to the difficulty in finding subject matter experts in both iOS and Android development. This is the most desired and most expensive skill-set today. Finding developers who know how to properly code and interface with low level Bluetooth and Wi-Fi APIs in both iOS and Android platforms is rare, so it's costly. Companies might also need to outsource to separate vendors who are experts in each platform, which adds more up-front costs. There are always additional costs and extra time it takes to contact vendors and conduct the bidding process/RFPs, etc. which are affected by company budgets and purchasing processes. It can take several months, at least, for developers to perfect the mobile application experience.

Cloud connectivity

This stage of product development is a critical one and can be challenging and prone to errors and potential launch delays that affect the success of IoT applications. Getting and maintaining reliable cloud connectivity and properly collecting data are huge concerns in IoT applications today. For example, if developers want to do online firmware or pull health metrics from personal health devices in the field (e.g., heart monitors, CPAP machines, connected inhalers), there are various risks to harnessing this data that could have dire outcomes if unsuccessful.

It's almost impossible for companies, especially small ones, to develop a cloud-connection framework/infrastructure from scratch. Developers also often have problems with unreliable links and connectivity and they might also be restricted by the MCU. Ultimately they want rich connectivity that is stable and reliable to enhance their customer/user experience. Having reliable connectivity links is a critical piece of product longevity and customer satisfaction. Firmware updates are also an important part of product maintenance and are usually outsourced by companies. This is very costly as they have to pay monthly (or yearly) to a subscription-based service to manage these updates. Using an integrated solution that already has the infrastructure for adding cloud connectivity can save developers several months of framework development.

Benefits of pre-programmed wireless modules

IoT developers today want robust functionality in the smallest footprint possible and they want solutions that support easy Wi-Fi and Bluetooth connectivity without the long learning curve. Leveraging integrated modules that already include pre-programmed firmware, pre-certified RF and hardware, easy mobile app framework, and cloud connectivity, streamlines the entire development process and takes the guesswork out of successful connectivity.

A product that utilizes a certified RF module, as opposed to a discrete circuit design, can leverage the certification of the off-the-shelf module and avoid the Intentional Radiation (IR) testing portion of the FCC certification process. Faster time to market, risk mitigation, and lower Non-Recurring Engineering (NRE) costs of product development makes leveraging a pre-certified RF module the optimal strategy for many companies and developers.

Selecting a product with functional and tested system firmware that's pre-programmed at manufacturing provides:

- a low-level wireless stack abstracted by the pre-programmed module's interface, lowering the barrier to entry for developers
- fully functional embedded application firmware, circumventing the need to develop in-house
- testing infrastructure built and deployed to test the firmware by the module vendor, which limits the scope of any infrastructure that needs to be built by the developer
- built-in support for firmware updates and bootloading, thus future-proofing for security updates, bug fixes, and robustness improvements

Using a module instead of designing with an SoC provides:

- integrated antenna options to minimize form factor and limit the need for in-house RF expertise
- pre-certified RF to avoid lengthy and costly RF certification efforts
- acceptable RF transmission out-of-box

Starting mobile app development with a wireless interfacing library that's been optimized for ease of use and thoroughly tested provides:

- an abstracted interface to core Bluetooth APIs across multiple mobile operating systems
- specialized APIs designed for optimal control of the interfacing embedded module
- a cloud-to-module route for firmware updates

Connecting to cloud infrastructure already in place as part of an integrated product provides:

- elimination of setup time and costs required to build cloud infrastructure from scratch
- a tested and operational firmware update deployment solution
- minimized risk of many of the security-related mistakes made by those new to cloud-connectivity design

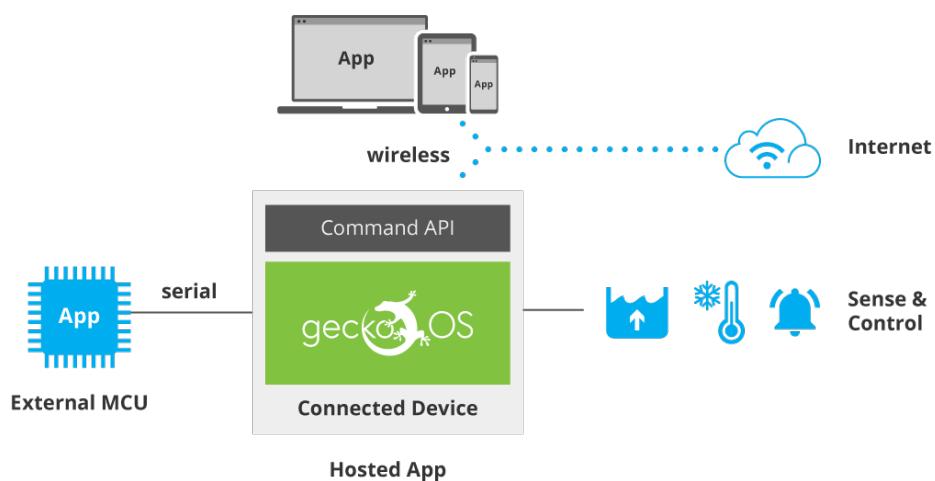
Silicon Labs Wireless Xpress products, powered by Gecko OS, and application firmware running on pre-certified Silicon Labs modules, combines these product development cycle stage optimizations to provide a streamlined embedded-to-phone and embedded-to-cloud connectivity.

Value of Gecko OS

Gecko OS is a highly-optimized IoT operating system designed specifically to power hardware platforms with secure Wi-Fi networking capability and is the best choice for resource-constrained devices. Hardware running Gecko OS provides products with a powerful and secure wireless connection to a mobile device or the cloud. Developers can build external MCU applications that communicate with Gecko OS using the Xpress Command API via serial interface, HTTP REST, or a remote terminal. The [Gecko OS API](#) is a huge benefit to IoT developers because it provides a common software foundation across multiple product lines and includes the firmware, boot-loader, command APIs, and wireless stacks. The API provides direct access to peripherals connected to Gecko OS hardware via GPIO.

In these types of Network Co-Processor (NCP) topologies, the external MCU must manage the wireless stack with low-level maintenance APIs, store large buffers of data in RAM, and devote CPU resources to interface monitoring. These responsibilities can be burdensome for MCUs without sufficient capabilities, and the resulting performance constraints force developers to choose the host MCU that best suits the NCP, not the best MCU for the application.

Gecko OS products relieve developers of this burden by maintaining much of the wireless interface without external MCU intervention, only exposing critical variables and commands for external MCU control. Gecko OS products such as Wireless Xpress frees customers to choose from the full spectrum of MCUs on the market, from functionally dense ARM-core processors to cost-effective 8-bit MCUs.



Silicon Labs Bluetooth Xpress Pre-Programmed Modules

Bluetooth Xpress products combine optimizations in the four-stage development cycle to deliver a BLE 5-enabled module that can get a design from concept to prototype in a day. Powered by Gecko OS, the Bluetooth Xpress product includes the Xpress framework for mobile application development, which greatly reduces the development time required to add Bluetooth connectivity to mobile apps.

The key features of Silicon Labs Bluetooth Xpress modules (BGX13P and BGX13S) include:

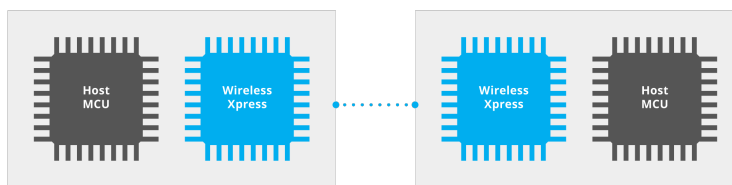
- an out-of-box, pre-programmed, pre-certified Bluetooth solution supporting Bluetooth 5 features including 2M and long range PHY
- on-board wireless stacks controlled through a high-level Gecko OS Xpress Command API for setup and control
- requires only modest resources from a host processor, enabling developers to add wireless connectivity to any microcontroller (MCU)
- baud rate, flow control, GPIO, and Bluetooth performance parameters are all configurable for application-specific requirements
- available in both PCB and SiP modules
- one point of contact for design support

Bluetooth Xpress Use Cases

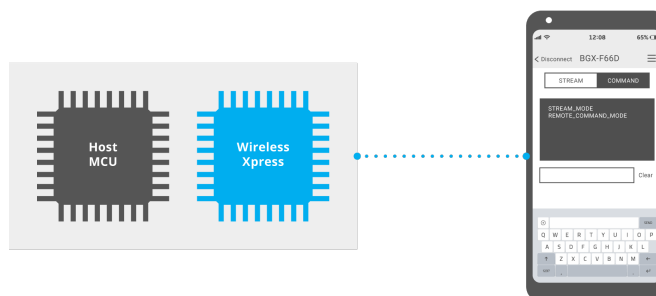
Bluetooth Xpress modules function in two use cases:

1. **BGX13 to smart phone** – In this use case, the BGX acts as a peripheral, which is discoverable and connectable by a smart phone acting as the central.
2. **BGX13 to BGX13** – In this use case, one BGX acts as a discoverable, connectable peripheral, and a second BGX acts as the central, which scans and connects. When connected, a BGX13 can function in either a streaming mode, where data bytes are transmitted between the two points in a no-overhead data stream, or in command mode, where Gecko OS commands can be executed remotely.

In both defined use cases, flow control signals from the BGX indicate buffer status and can be used to indicate whether the embedded host can transmit data.



BGX to BGX Use Case



BGX to Phone Use Case

[Learn more about Bluetooth Xpress modules](#)

Silicon Labs Wi-Fi Xpress Pre-Programmed Modules

Similar to Bluetooth Xpress, Wi-Fi Xpress uses the Gecko OS interface to provide streamlined control, pre-certified modules, integrated Wi-Fi stacks, and built-in mobile and cloud connectivity advantages. Developers can get reliable W-Fi-connectivity quickly for their applications/devices and access to design tools, demos, and support documentation such as application notes and examples.

Other key features of Silicon Labs Wi-Fi Xpress modules:

- TCP and UDP client and server functionality
- 802.11b/g/n, 2.4GHz bands
- +20 dBm output power
- file system for logging and host/module OTA
- web server functionality

Wi-Fi Xpress Use Cases

Wi-Fi Xpress modules function in a wide variety of smart home, commercial, and retail use cases. Wi-Fi Xpress provides a low-development cost path from silicon to cloud with applications from enabled home-security systems and asset-tracking systems, to cloud-connected point of sale (POS) systems.

[Learn more about Wi-Fi Xpress modules](#)

Getting Started with Wireless Xpress Solutions

Silicon Labs starter kits already come with the wireless modules for either Bluetooth or Wi-Fi connectivity and are the easiest and fastest way to get a product connected and start prototyping.

Wireless Xpress Bluetooth Starter Kit

The Wireless Xpress BGX13P Starter Kit is the fastest way to get products Bluetooth-connected. The kit includes everything you need and operates in a stand-alone mode when a USB is attached, enabling Bluetooth connection and communication through the serial-to-USB bridge to BGX13's Xpress command interface, and in an STK mode when an EFM8 or EFM32 STK is attached to the board's expansion header.

The Wireless Xpress BGX13P starter kit includes a BGX13P expansion board, USB micro cable, and a BGX getting started card.

Other key features of the starter kit include:

- Wireless Xpress BGX13P module
- CP2102N serial to USB bridge
- 2 x mechanical buttons connected to BGX13P I/O
- 2 x LEDs connected to BGX13P I/O
- expansion header and full pin access enables easy interface off-board
- [Simplicity Studio](#) Xpress configurator
- Simplicity Studio interface demo

[Learn more about getting started with Bluetooth Xpress](#)

Wireless Xpress Wi-Fi starter kit

The Wireless Xpress Wi-Fi starter kit ([AMW007-E04](#)) is the easiest way to start evaluation and development in any new Wi-Fi-enabled device. The base starter kit operates standalone and can be used with a simple terminal emulator via the USB port. Developers can also supplement their Wireless Xpress expansion board with a microcontroller development kit such as the [EFM8UB1](#) or [EFM8UB2](#).

The base starter kit includes:

- Wireless Xpress AMW007-E04 expansion board
 - AMW007 Wi-Fi module
 - CP2102N USB to UART bridge
 - 3 x mechanical pushbuttons (2 user definable & 1 reset)
 - LEDs (2 user definable, 2 Tx/Rx Data, power)
 - expansion header
- USB micro cable

[Learn more about getting started with Wi-Fi Xpress](#)

Summary

This paper discussed the four stages of IoT wireless development, the benefits of pre-programmed wireless modules, and how Silicon Labs pre-programmed Bluetooth Xpress and Wi-Fi Xpress modules help designers reduce complexity and streamline the development process dramatically. Several application use cases were also provided. There are a number of key benefits that developers gain when using pre-programmed modules that overcome development challenges – configurable/customizable, instant prototype, flexible to use with any MCU, and if the module vendor also makes the SoC, there is deep support for all module components. With Silicon Labs modules, developers get robust, in-house development support because Silicon Labs owns all the components including the silicon/wireless stack with no outsourcing needed. Developers can get started quickly by leveraging the Bluetooth Xpress and Wi-Fi Xpress starter kits as they include the wireless module plus additional components that make the set-up and wireless connection process simple and fast.

Silicon Labs modules provide developers a drop-in connectivity prototype that gives them a head start in their application launch, which in turn means more revenue. In addition, developers can save countless months of development work and get products to market faster by leveraging Silicon Lab's pre-programmed, pre-certified wireless modules-no programming necessary and no risk of failing FCC certifications. Find more details on Silicon Labs [Wireless Xpress solutions](#).

References

[Six Hidden Costs in a 99 Cent SoC Whitepaper](#)

[Xpress Streaming Service](#)

<https://www.silabs.com/products/development-tools/wireless/wi-fi>

<https://www.silabs.com/support/getting-started/wifi/xpress>

<https://www.silabs.com/support/getting-started/bluetooth/xpress>