Ushering in a New Era of Internet Connectivity

Thread Networking Protocol Simplifies Connecting "Things" in the Home and Beyond

Introduction

Thread is the future of wireless mesh networking, and it is poised to become a pivotal technology for connecting "things" to the Internet. "Things" can be loosely defined as power-sensitive, space-constrained devices that incorporate microcontrollers (MCUs), sensors, RF technology and software. Wireless connectivity bridges these devices to the Internet, which then can be remotely controlled and automated. Multiple wireless technologies exist in the market that support such interconnectivity, but Thread uniquely serves the specific needs of the Connected Home. It fills a critical technology gap in the Internet of Things (IoT) by providing a low-power, secure and scalable IP-based wireless mesh networking protocol based on proven standards.

The Internet of Things

The analyst firm IHS Technology predicts that the IoT will grow to 75.4 billion connected devices by 2025. Such an unprecedented level of connectivity is the result of remarkable advances in technology tracing back to the early 1970s when Intel released the first commercially available microprocessor, the 4004. Ten years later, desktop

computing reached the state of the art, ushering in an era of unprecedented personal productivity.

Laptop computing emerged in the 1990s and, coincident with expansion of the Internet, marked the beginning of connected mobility. However, it wasn't until the advent of the smartphone in the mid-2000s did we truly realize an untethered Internet experience. Smartphones put the power of personal computing combined with Internet connectivity in the palms of our hands. Advances in low-power computing, sensing and communication have led to a new class of connected devices. These small, resource-constrained, battery-powered "things" no longer just focus on personal productivity, but rather serve to sense, interact and automate our surroundings at home, in the workplace and throughout our urban environments in radically new and useful ways. Figure 1 visualizes these changes over the evolution of connected devices.
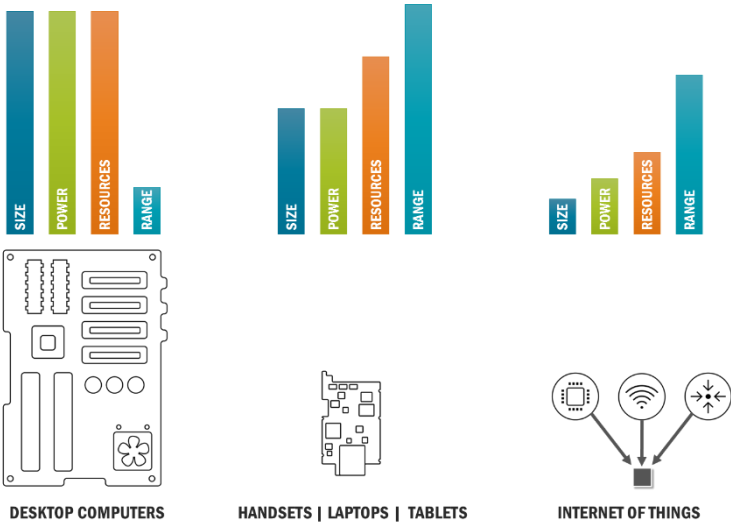

Figure 1: Emergence of the Internet of Things

Why IP Matters

Internet Protocol (IP) is the principal communications protocol of the Internet. IP provides the core mechanism for relaying datagrams across IP networks, and its routing capabilities enable internetworking. After decades of rapid expansion of the Internet, IP has become ubiquitous.

Networking technologies that do not natively support IP must first be adapted to IP at a gateway. This process involves mapping of the local network addresses and repackaging of the network-level payload into an IP datagram. As you can see in Figure 2, local packets that are encrypted must be decrypted at the gateway and then re-secured in the IP datagram.
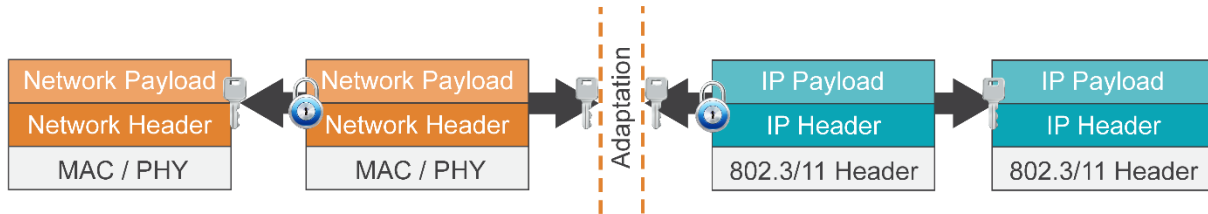

Figure 1: Networks without Native IP Support

In contrast, local networks that natively support IP such as Thread and Wi-Fi can forward and route application payloads without intervention. Packets encrypted in the local network can remain secured end to end, as shown in Figure 3.
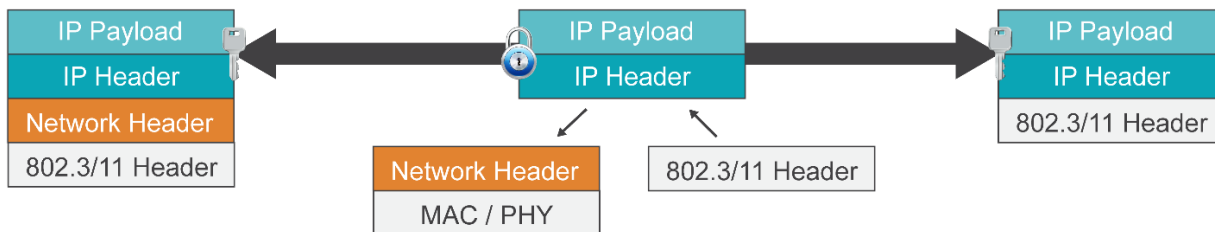

Figure 2: Networks with Native IP Support

IPv6 versus IPv4

IPv4 supports 32-bit addressing. It became evident in the 1990s that an address space of 4.3 billion (2^{32} = 4 294 967 296) was going to be insufficient to support the anticipated growth in Internet connectivity. IPv6 increases the address size to 128-bits and the resulting address space to approximately 240 undecillion! Local devices with IPv6 Globally Unique Addresses (GUAs) can be addressed directly by the wide area network (WAN).

6LoWPAN

6LoWPAN is an acronym for IPv6 over Low Power Wireless Personal Area Networks. It was designed to enable the efficient transmission of IPv6 datagrams over 802.15.4 links and thereby enable resource-constrained devices such as "things" to be able to natively participate in the IoT. It supports three key functions:

Packet Size Adaptation

The smallest IPv6 MTU (Maximum Transmission Unit) supported is 1280 bytes. A typical application payload size in an 802.15.4 packet using UDP (User Datagram Protocol) and DTLS (Datagram Transport Layer Security) is 63 bytes. 6LoWPAN provides a fragmentation and reassembly mechanisms to adapt IPv6 datagrams to these smaller 802.15.4 payloads. IPv6 packets that do not fit are split into fragments and sent over the air via 802.15.4. Not all fragments may be received in the correct order. However, 6LoWPAN only requires that all fragments are received and will reorder fragments during reassembly if needed.

Header compression

To minimize the overhead of sending IPv6 messages in 802.15.4 frames, 6LoWPAN provides stateless compression mechanisms for both IPv6 and transport headers that take advantage of cross-layer redundancies between protocols such as source and destination addressing, payload length, traffic class and flow labels. . Thread utilizes IPHC (Improved Header Compression) and

NHC (Next Header Compression). IPHC is used to compress the IPv6 header. NHC is used to compress the UDP header.

Layer Two Forwarding

Thread uses IP routing to forward packets. The IP routing table is maintained with each destination and the next hop to it. The 6LoWPAN mesh header is used to do link level next hop forwarding based on the IP routing table information.

Getting Connected

A diverse set of wireless technologies exists in the networking marketplace. For the long term we see three primary solutions emerge as leading protocols in the connected home and IoT: Wi-Fi, Bluetooth Smart and 802.15.4 mesh networking protocols.

Wi-Fi is a wireless powerhouse that is adept at transporting audio, video and data throughout the home. The high data rates and output power that it supports, however, demand a lot of power. As such its utility for battery-power applications is limited.

Bluetooth® Smart is ideal for connecting point solutions to smartphones and for bulk data transfers on battery-power devices. Although existing solutions do not natively support IP, the introduction of Bluetooth Core Specification 4.2 lays the groundwork by adding support for IPv6 and 6LoWPAN.

Mesh technologies such as ZigBee PRO and Thread form the backbone of a connected home. These self-healing networks have been optimized to support low bandwidth control and automation applications where multi-year battery life is critical.

ZigBee PRO has been the dominant 802.15.4 mesh networking solution for well over 10 years. It offers a robust, secure and scalable mesh network supporting more than 250 nodes as well as extensive support for sleepy end devices for battery-backed operation.

Thread builds upon the foundation of ZigBee® with advances in security, routing and fault tolerance. It also addresses the long-awaited pull in the market for mesh

networking technology that supports IPv6 and 6LoWPAN, bringing much-needed IP connectivity to each individual end node in the IoT. Table 1 lays out the features of the various networking protocols.

| | Wi-Fi | BT Smart | 802.15.4 | |
			ZigBee PRO	Thread
Bandwidth	150Mbps+	1Mbps	250kbps	250kbps
Low Power Consumption	✘	✔	✔	✔
Native IP addressability	✔	✘	✘	✔
Simple IP bridging	✔	✘	✘	✔
Mesh networking	✘	✘	✔	✔
Practical network size limit	32	10	250+	250+
Security Support	AES-128/256	AES-128	AES-128	AES-128, ECC
No single point of failure	✘	✘	✘	✔

Table 1: Wireless Technologies for the Internet of Things

Why Thread, Why Now?

The Thread Group was formed on July 15, 2014, to collaborate, pool expertise and develop a solution to meet these needs. One year later, the Thread Group released the Thread specification. The founding members of the organization include Silicon Labs, Nest Labs, Yale Security, Samsung Electronics, Freescale Semiconductor, Big Ass Fans and ARM.

Thread's focus on low-power consumption and native support for IP delivers on the promise of seamless connectivity between "Things" and the Internet. It affords not only efficient communication between low-cost, battery-backed devices but also a simplified interface to the cloud and mobile devices. Thread offers a compelling and new wireless mesh networking solution designed to meet the needs of the connected home:

Standards-Based Protocol

Open standards-based protocols reap the benefits of technical peer reviews, cross-vendor interoperability and competition. Thread is based on a foundation of existing IEEE and IETF standards including IEEE 802.15.4. The official Thread Specification was published by the Thread Group on July 13, 2015.

Simplified Configuration

Thread does away with some of the complexities of prior mesh networking standards. There are only two different node types: Router Eligible or End Device.

Router Eligible nodes become routers if needed to support the mesh. The first Router Eligible node to form the network will be autonomously designated a router as well as the Leader. A Leader performs additional network management tasks and makes decisions on behalf of the network. Other Router Eligible nodes in the network can autonomously assume the role of a Leader but there is only one Leader per network at a given time.

Nodes that join as **End Devices** do not support any routing capabilities. Instead, they send messages to a router designated as its "parent" and the parent performs routing operations on behalf of its "child." End Devices route communication through parents and can be programmed to be "sleepy" to reduce power consumption. End Devices that are unable to communicate with their parent after multiple attempts will autonomously search and attach to a new parent. Figure 4 shows a Thread node network with Router Eligible End Devices (REEDs), a Leader, and Thread Routers.
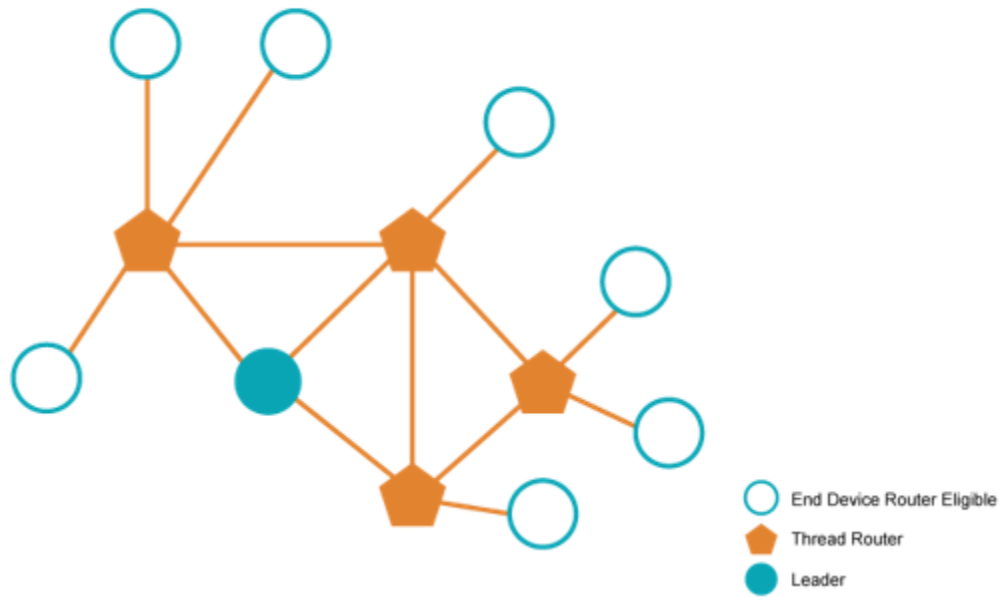
Figure 3: Thread Node Types

IP-Based Messaging

Thread provides native support for IP. All devices in a Thread network have an IPv6 address and can be accessed directly by local devices on the Home Area Network (HAN) or off-network using Thread-capable IP routers called Border Routers. Figure 5 illustrates a typical Thread networking setup.
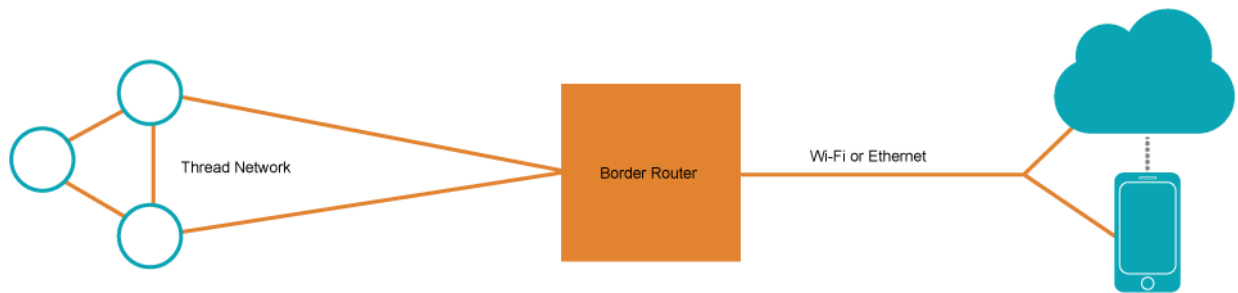

Figure 4: Thread Network Connectivity

Nodes on the network form IPv6 global addresses from prefixes assigned by border routers, or locally by a self-assigned prefix to form a Unique Local Address (ULA). Routing IDs for use in the network are assigned by the Leader. Thread leverages UDP (User Datagram Protocol) for messaging instead of its heavier weight alternative, TCP (Transmission Control Protocol). Unlike TCP, UDP is a connectionless protocol that forgoes TCP features such as error checking, packet sequencing and retransmissions in exchange for faster and more efficient transmission. This efficiency equates to less overhead which is extremely important for battery-backed, resource-constrained devices. Thread utilizes the Constrained Application Protocol (CoAP) in conjunction with UDP to remain lightweight but overcome some of the limitations of UDP. CoAP reincorporates support for retries and packet sequencing. CoAP further provides support for lightweight versions of GET, POST, PUT and DELETE methods that can be easily translated to HTTP for simplified integration with the web. Utilizing off-network CoAP to HTTP proxies, Thread makes it possible to query IoT devices directly from a browser!

Low-Power Operation

Thread provides extensive support for low-power operation using sleepy end nodes. As the name would suggest, these sleepy devices are meant to spend the majority of their time in a low-power sleep state. With Thread, sleepy devices are not required to check in with the network, enabling prolonged battery life. Messages for sleepy devices are buffered by their parents. Message transmission is triggered only after a sleepy device wakes up and polls its parent for any inbound messages.

Scalable and Resilient

Thread was architected with the specific goal of supporting networks of 250+ nodes. With this constraint, features and capabilities were adapted to maximize throughput and minimize resource requirements. In a Thread network, the

maximum number of active routers is 32. A direct and important consequence of this is that routing information can be efficiently distributed across the network and all routers are able to maintain visibility of all routes within the network. As nodes are added to the network and the topology changes, the network adapts by exchanging Mesh Link Establishment (MLE) messages. Additional routers can be autonomously added if there are Router Eligible devices operating as end devices. Such Router Eligible end devices or REEDs, listen to routing messages and will petition the Leader of the network to become a router if determined necessary to improve the overall performance of the network.

Thread is a robust, self-healing mesh network that offers no single point of failure. If a router fails, the network will dynamically re-route traffic around the failed node. If a Leader fails, another router on the network will be autonomously elected the new Leader. Multiple border routers can be used to provide fail-safe redundancy for off-network communication.

Interoperable

The Thread Group has contracted a standard test harness to be used for certification of all Thread stacks as well as final Thread products. This test harness will be provided to all Thread member companies to allow high quality development and testing of software prior to Thread certification testing.

All Thread components (ICs, software stacks or modules) must be certified as Thread compliant before being used in an end product by a customer. All end product devices that want to carry the Thread logo must submit their devices for laboratory certification at an approved test lab.

Secure and User-Friendly

Thread networks are built to be simple to use yet extremely secure. Thread uses AES-128 to protect all networking transactions at the MAC level and utilizes a combination of ECC and J-PAKE algorithms to securely add devices to a network

(i.e. commissioning). Applications have the option to utilize standards-based IP security protocols (e.g., DTLS) to additionally secure application payloads.

The Thread network assumes that there is a commissioning device available for a user to designate to add new devices to the Thread network. Thread makes it possible to designate both off-network devices such as smart phones and computers and on-network Thread devices to become Commissioners. Off-network devices must first petition with a Thread Border Router to become a registered Commissioner using a secure DTLS authentication handshake specified by Thread. Once registered, the Border Router then performs arbitration with the Leader on behalf of the Commissioner candidate to become an authorized Commissioner. On-network devices do not require registration, but may need to perform arbitration with the **Leader** to be authorized. (The Leader ensures that only one commissioner is active on the network at a time.) Once authorized, the active Commissioner will be made known throughout the Thread network.

To join, a user instructs the commissioning device that a new devices is to be added and inputs a unique passphrase associated with the joining device. This passphrase is used to establishing a secure DTLS session to authenticate and authorize the joining device. Only then is the device provided access to the Thread network. Once the device is added to the network, the commissioning device is no longer active.

Accelerating Time to Market

With the rapid proliferation of devices for the Internet of Things, it's ever more critical to get products from the idea stage onto store shelves as quickly as possible. Here are two ways Thread and Silicon Labs help accelerate time to market.

Application Development

Thread does not specify an application layer but instead provides basic messaging services such as unicast and multicast message services using UDP and CoAP. Similar to Wi-Fi, Thread focuses on the secure and reliable transport of information but rather by using a low-power mesh network instead of a high-powered star network. Tools such as Silicon Labs AppBuilder simplify the use of these IP messaging services by abstracting away stack level details through the use of easy-to-use callbacks and plug-ins. Application code development in this framework is highly portable and can be reused across supported 802.15.4 devices. AppBuilder providers a simple GUI to configure devices and networking parameters such as device type, commands, network forming and joining behaviors as well as security modes and operation. Silicon Labs provides a sample application for Thread to get started quickly that can be easily customized and extended.

Networking Debug

Understanding, adapting and tuning network level behavior is an important development phase in developing applications for mesh networks. Traditional solutions use wireless sniffers to remotely capture over-the-air networking transactions. Silicon Labs networking ICs build this capability in using a unique two-wire interface called Packet Trace that outputs every packet sent and received by the device appended with link quality information. This information can be aggregated across all nodes in the network into a single view using a tool called Desktop Network Analyzer, providing a complete and comprehensive view of all networking activity. Desktop Network Analyzer correlates all traffic into events and can decode Thread messages. Network activity can be logged over extended periods of time to identify issues that may be infrequent or occur intermittently in the network.

Conclusion

We are entering a new era of Internet connectivity. The Internet of Things will eclipse the historic peaks of desktop computers, laptops and smartphones combined. Small, resource constrained devices will drive this growth in an incredibly wide array of applications. From home automation to building security sensors, from farming irrigation to industrial process automation and from healthcare to personal hygiene. It will touch multiple aspects of our lives and change how we interact with our environment. Most of these devices will be wirelessly connected. Multiple wireless technologies exist that can provide IP connectivity, but we expect three solutions to dominate the connected home: Wi-Fi, Bluetooth Smart and 802.15.4 mesh networking protocols. Thread is the future of mesh networking. It delivers on the promise of an IP-based mesh networking solution that is secure, reliable, scalable and optimized for low power operation.

For more information on Silicon Labs' Thread networking solutions, visit www.silabs.com/thread.

Making Electronics Smart, Connected, and Energy Friendly

Silicon Labs (NASDAQ: SLAB) is a leading provider of silicon, software and system solutions for the Internet of Things, Internet infrastructure, industrial control, consumer and automotive markets. We solve the electronics industry's toughest problems, providing customers with significant advantages in performance, energy savings, connectivity and design simplicity. Backed by our world-class engineering teams with unsurpassed software and mixed-signal design expertise, Silicon Labs empowers

developers with the tools and technologies they need to advance quickly and easily from initial idea to final product.