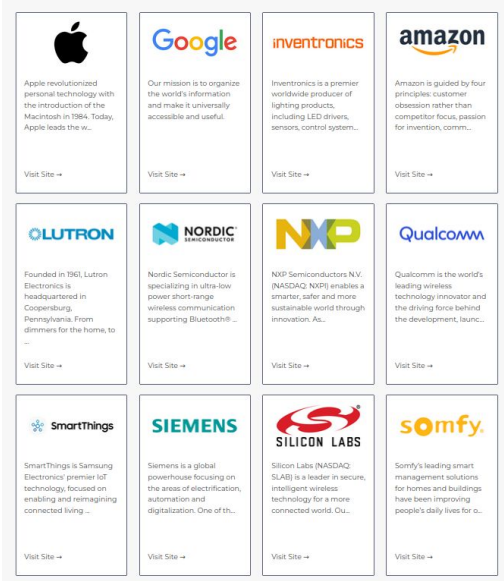
Tock and Thread Networking

Tyler Potyondy, UC San Diego
tpotyondy@ucsd.edu

What is Thread Networking?

Thread is a low power, secure and future-proof mesh networking technology for IoT products.



BUILT FOR IOT
Low power, secure and robust wireless mesh built on IP

CONVERGENCE & COEXISTANCE
IP as a point of convergence

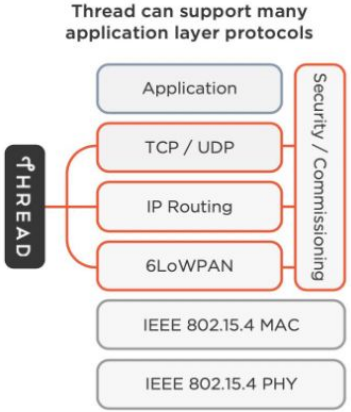
GLOBAL SOLUTION
Open standard for smart homes and buildings

FLEXIBLE & FUTURE PROOF
Enabling interoperability

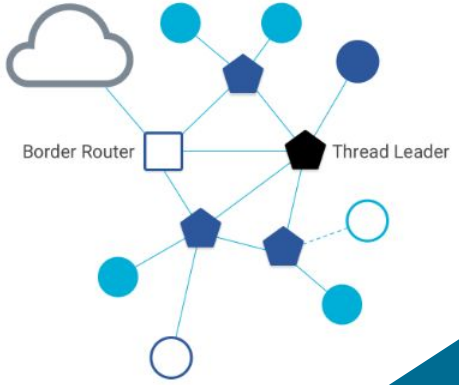
Thread Group - Thread 1.3.0 Webinar; 9-29-22;
https://www.threadgroup.org/

What is Thread Networking?

- Thread builds a networking layer on top of 802.15.4
- Utilizes IPv6 routing for each node, allowing for interoperability with existing networks
- Combines star and mesh topologies to provide a robust network



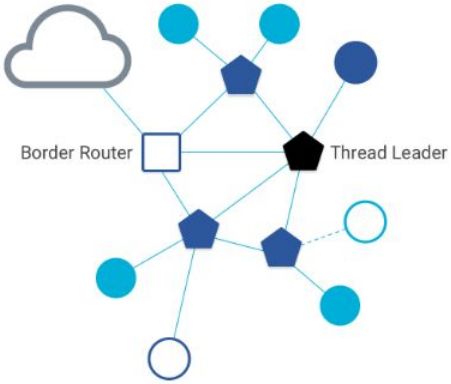
[1]



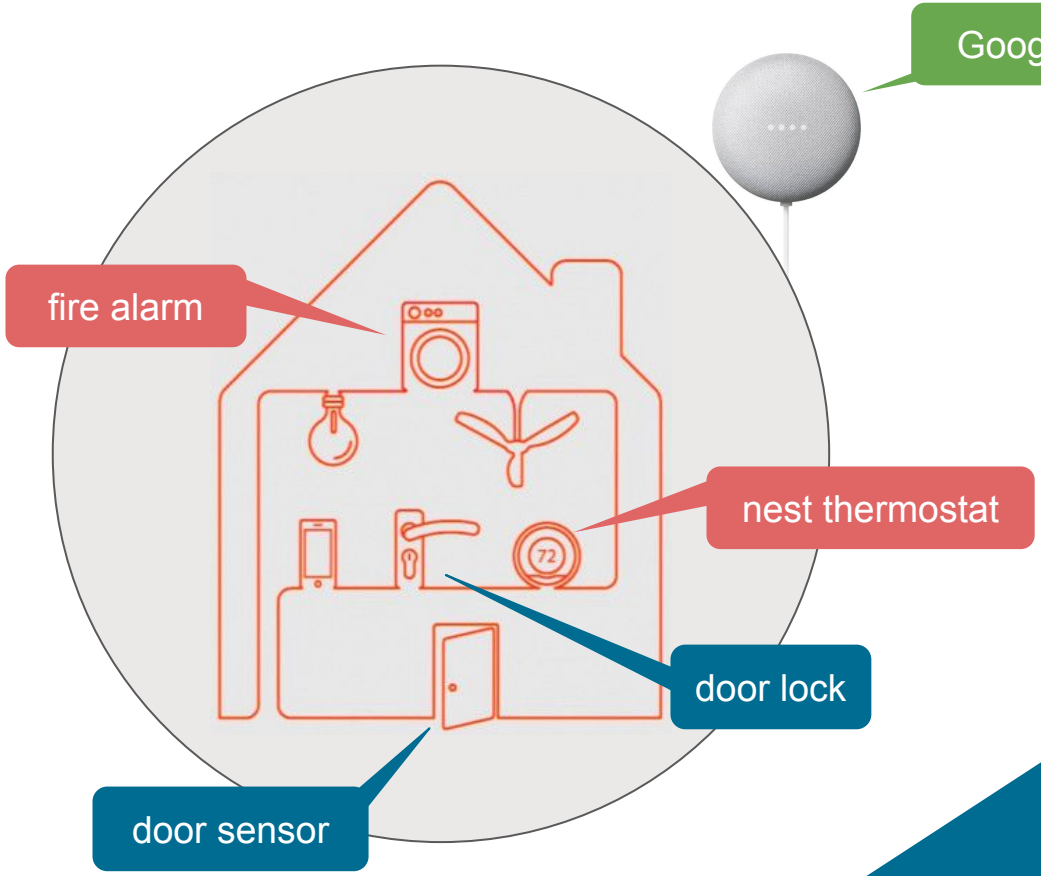
[1] Thread Group - Thread 1.3.0 Webinar; 9-29-22

Thread Networking - Node Types

- Border Router
  - Node providing bridge from Thread network to external internet
- Router
  - Responsible for maintaining network topology
  - Forwards packets for network devices
  - Mesh communication with other routers
  - Power intensive node
- Child
  - Communicates with one router
  - Does not forward packets
  - Can disable transceiver to enter low power state (sleepy end device)



Example of Real World Thread Network



Border Router

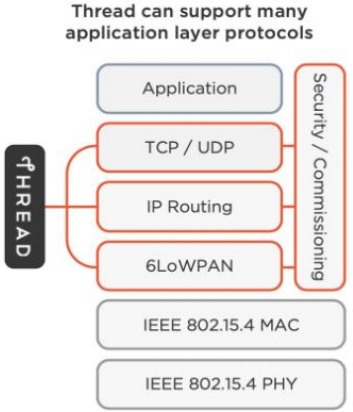
Router Node

Child Node

(Image of House) https://www.the-ambient.com

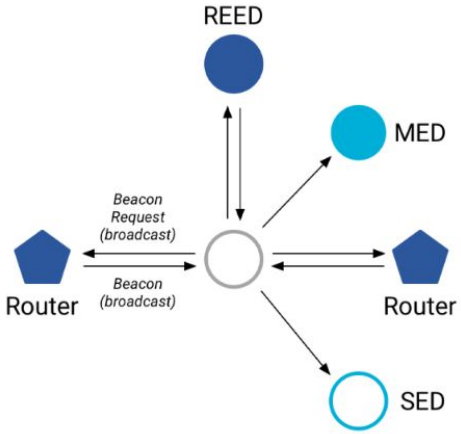
Thread Networking and Tock

- Tock currently supports each networking layer theoretically required for deploying a Thread network
- Tock implements the AES128-CCM encryption needed to secure Thread payloads
- Can we add kernel support for Thread networking?



[1] Thread Group - Thread 1.3.0 Webinar; 9-29-22

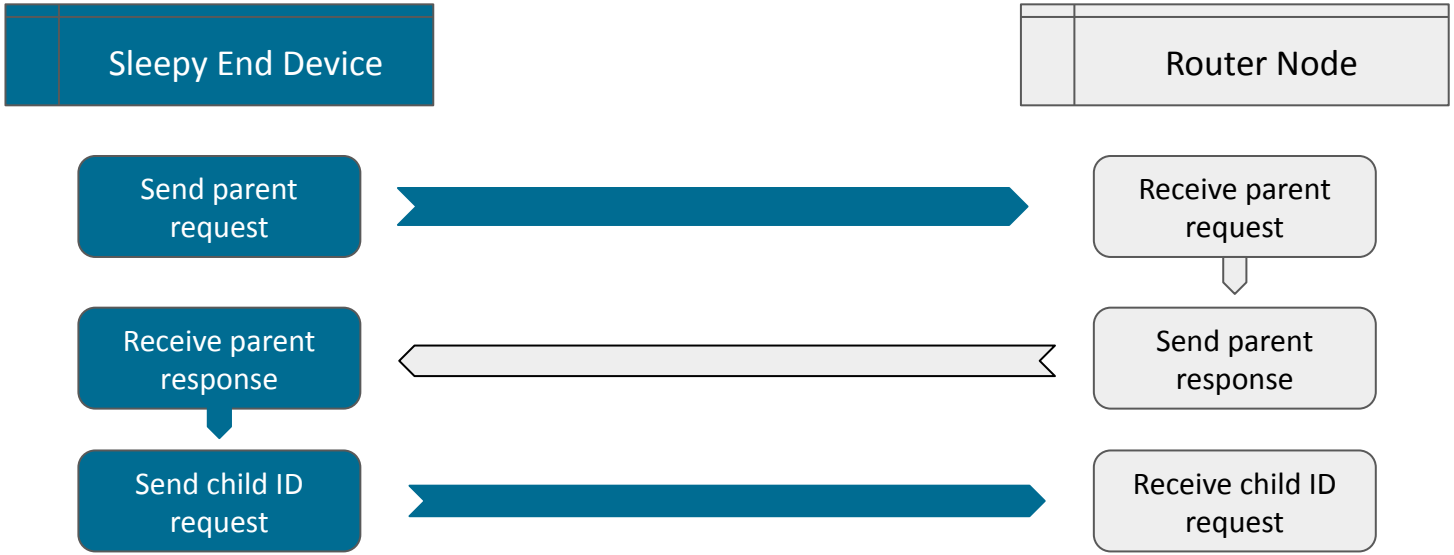
Tock – Sleepy End Device



- A sleepy end device (SED) is the simplest of Thread node types
- OpenThread is Google's open source implementation of the Thread networking specification
- **Goal:** Tock kernel level support to successfully join an OpenThread network

OPENTHREAD
released by Google

Tock – Sleepy End Device



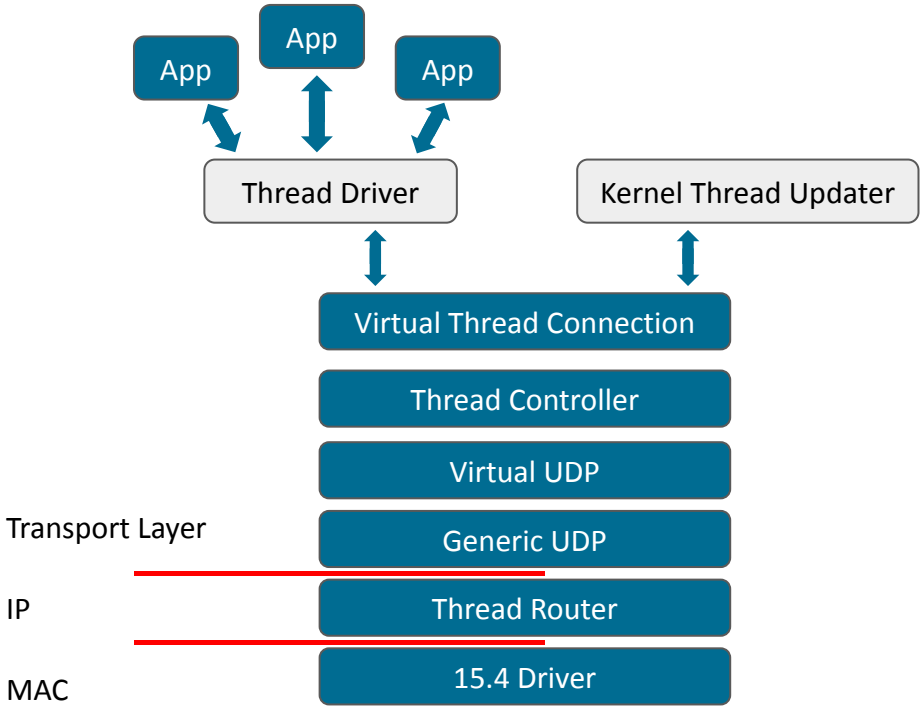
Tock – Sleepy End Device

Successfully performed “handshakes” required for OpenThread to recognize a Tock SED as a node in the Thread network

Successfully proved that the Tock networking stack and AES128-CCM encryption can support a Thread network

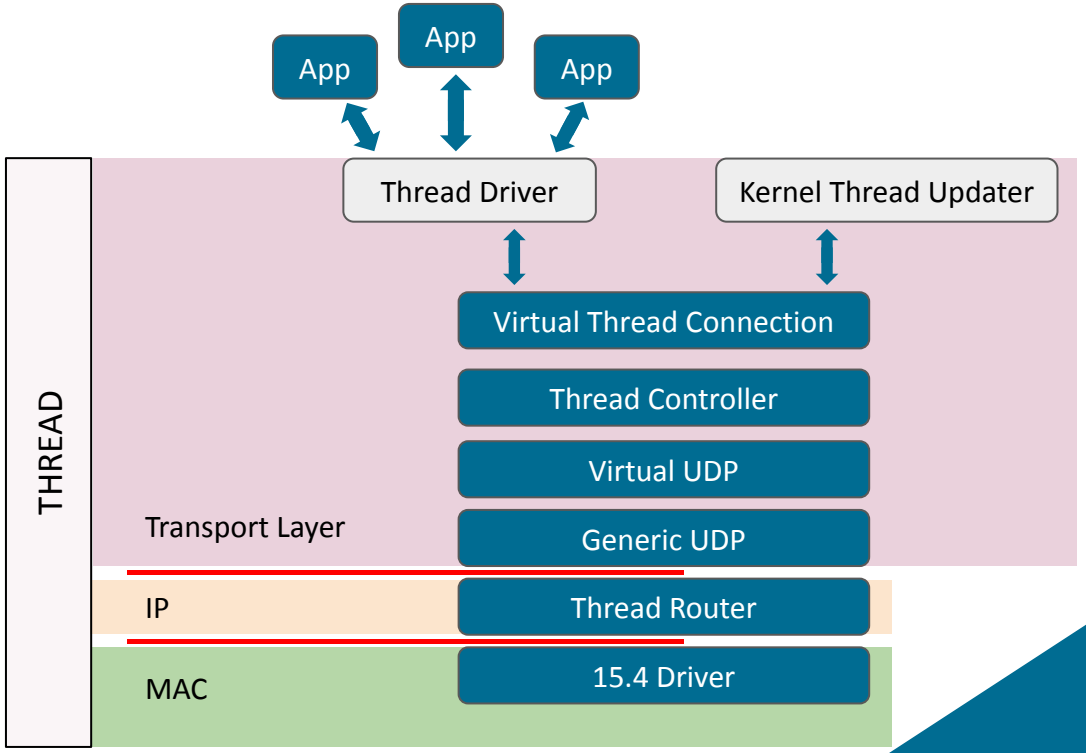
Open Questions/Current Work

How to best design the Thread Networking Capsule to be generalizable?



Open Questions/Current Work

Unique Challenge:
Thread spans 3 layer of the networking stack



Open Questions/Current Work

How to best track the device's Thread network state?

Will the kernel's sharing of cryptographic resources lead to starvation and subsequently dropped packets?

Should each device support multiple distinct Thread networks?

How to best incorporate Thread across the networking stack?

Should all apps be allowed to access the Thread Network?
Should we possess a means to differentiate access?

Questions?