6LoWPAN Implementation

Harshvardhan Mundada [22ucs089]
Saket Samarth [22ucs173]

Supervisor:
Dr. Rajbir kaur

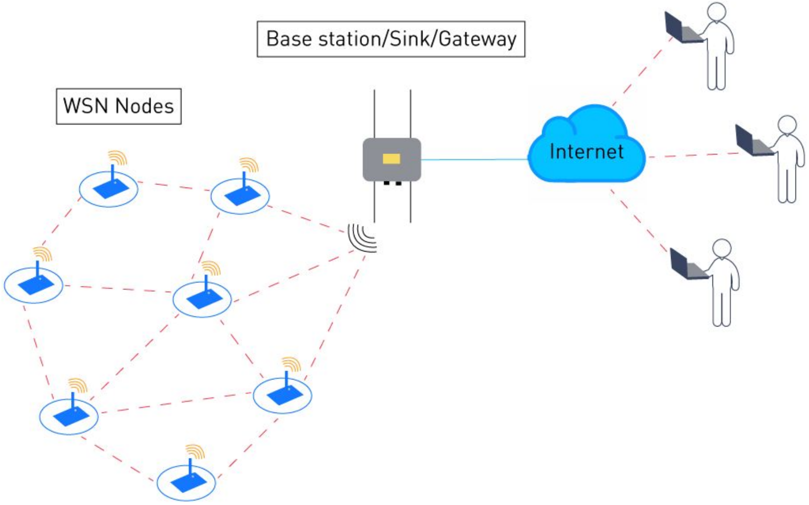
Power-Efficient IoT Meets the Internet: 6LoWPAN & IPv6 in Action

- Design and build 6LoWPAN testbed.
- Analyze and explore 6LoWPAN communication.
- Enable efficient low-power wireless networking.
- Prototype for extended development work.

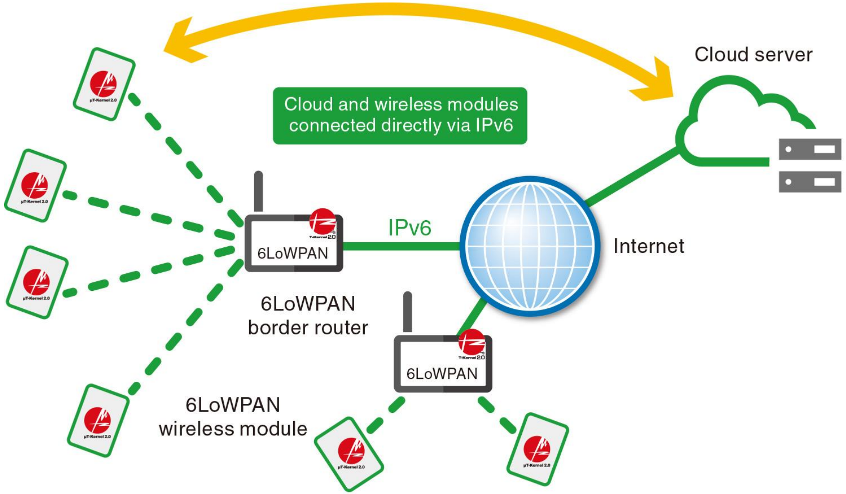
Objective and Workflow

- Establish a 6LoWPAN network testbed.
- Configure devices with 802.15.4 radio modules.
- Deploy suitable software for 6LoWPAN communication.
- Set up roles: Border Router and End Device.
- Validate low-power IPv6 communication between nodes.

Concepts of 6LoWPAN and need for its Exploration



WSN Network



6LoWPAN Network

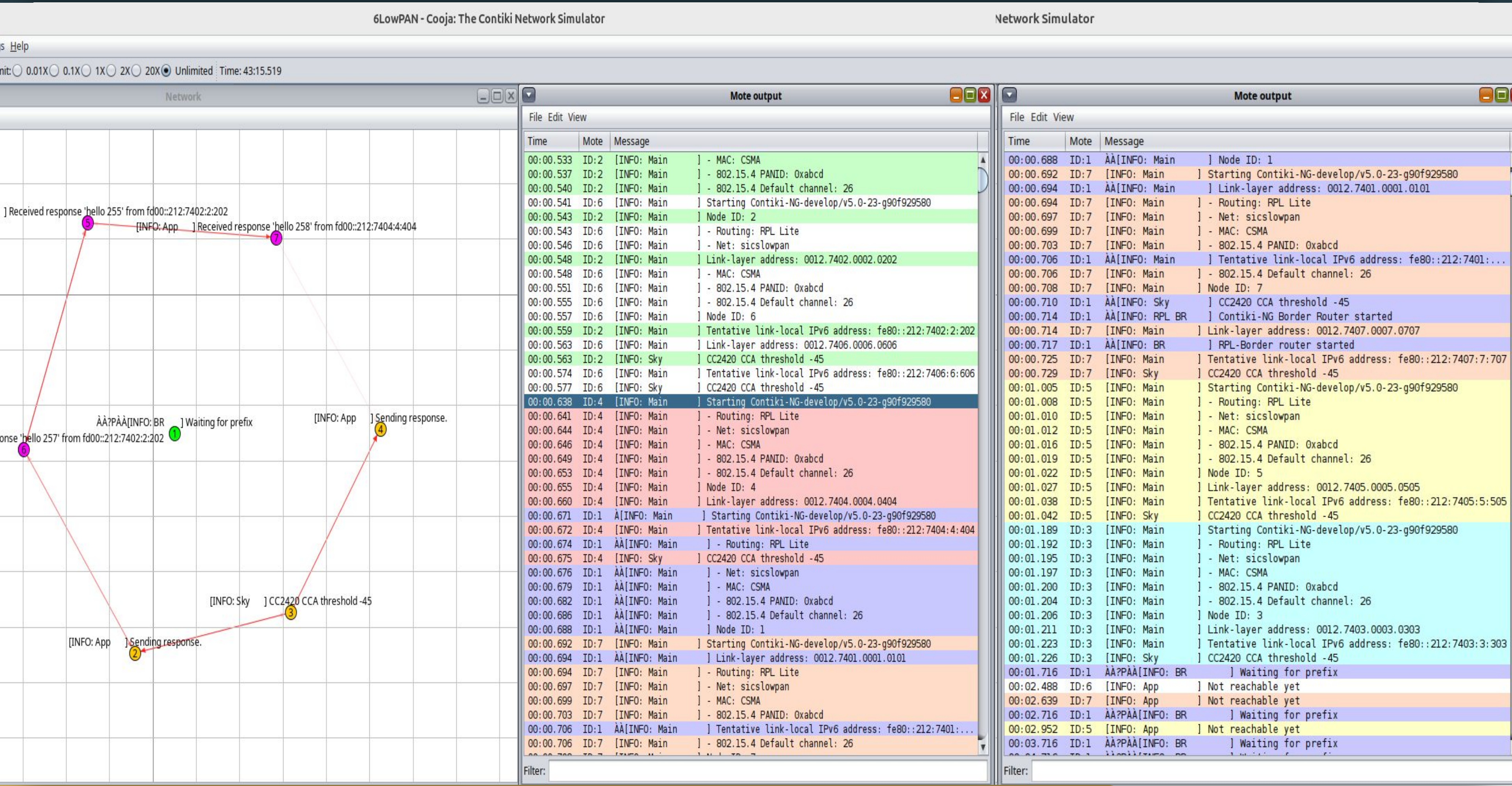
	Wireless Sensor Network	6LoWPAN
1]	Proprietary protocols limit interoperability	Uses IPv6 for universal addressing
2]	Lacks direct IP connectivity	Seamless Internet integration for devices
3]	Harder integration with the Internet	Low-power communication efficiency
4]	Limited scalability in large networks	Standardized, scalable network architecture
5]	Difficult multi-vendor device communication	Future-ready for smart tech expansion

Concepts of 6LoWPAN and need for its Exploration

- Works at network layer.
- Built on IEEE 802.15.4 MAC/PHY layer.
- Enables direct IPv6 communication over radio.
- Assigns unique IPs, no translation needed.
- Uses header compression, fragmentation for efficiency.
- Designed for scalable, low-power IoT networks.

Setting the Foundation – Simulating 6LoWPAN in Cooja

- Cooja simulates IoT and 6LoWPAN networks.
- Configured Border Router and End Devices.
- Tested network topologies and communications.
- Validated design through real-time simulation


6LowPAN - Cooja: The Contiki Network Simulator
Network Simulator
Help
0.01X 0.1X 1X 2X 20X Unlimited Time: 43:15.519
Network
] Received response 'hello 255' from fd00::212:7402:2:202
[INFO: App] Received response 'hello 258' from fd00::212:7404:4:404
ÀÀ?PÀÀ[INFO: BR] Waiting for prefix
[INFO: App] Sending response.
onse 'hello 257' from fd00::212:7402:2:202
[INFO: Sky] CC2420 CCA threshold -45
[INFO: App] Sending response.
Mote output
File Edit View
Time Mote Message
00:00.533 ID:2 [INFO: Main] - MAC: CSMA
00:00.537 ID:2 [INFO: Main] - 802.15.4 PANID: 0xabcd
00:00.540 ID:2 [INFO: Main] - 802.15.4 Default channel: 26
00:00.541 ID:6 [INFO: Main] Starting Contiki-NG-develop/v5.0-23-g90f929580
00:00.543 ID:2 [INFO: Main] Node ID: 2
00:00.543 ID:6 [INFO: Main] - Routing: RPL Lite
00:00.546 ID:6 [INFO: Main] - Net: sicslowpan
00:00.548 ID:2 [INFO: Main] Link-layer address: 0012.7402.0002.0202
00:00.548 ID:6 [INFO: Main] - MAC: CSMA
00:00.551 ID:6 [INFO: Main] - 802.15.4 PANID: 0xabcd
00:00.555 ID:6 [INFO: Main] - 802.15.4 Default channel: 26
00:00.557 ID:6 [INFO: Main] Node ID: 6
00:00.559 ID:2 [INFO: Main] Tentative link-local IPv6 address: fe80::212:7402:2:202
00:00.563 ID:6 [INFO: Main] Link-layer address: 0012.7406.0006.0606
00:00.563 ID:2 [INFO: Sky] CC2420 CCA threshold -45
00:00.574 ID:6 [INFO: Main] Tentative link-local IPv6 address: fe80::212:7406:6:606
00:00.577 ID:6 [INFO: Sky] CC2420 CCA threshold -45
00:00.638 ID:4 [INFO: Main] Starting Contiki-NG-develop/v5.0-23-g90f929580
00:00.641 ID:4 [INFO: Main] - Routing: RPL Lite
00:00.644 ID:4 [INFO: Main] - Net: sicslowpan
00:00.646 ID:4 [INFO: Main] - MAC: CSMA
00:00.649 ID:4 [INFO: Main] - 802.15.4 PANID: 0xabcd
00:00.653 ID:4 [INFO: Main] - 802.15.4 Default channel: 26
00:00.655 ID:4 [INFO: Main] Node ID: 4
00:00.660 ID:4 [INFO: Main] Link-layer address: 0012.7404.0004.0404
00:00.671 ID:1 À[INFO: Main] Starting Contiki-NG-develop/v5.0-23-g90f929580
00:00.672 ID:4 [INFO: Main] Tentative link-local IPv6 address: fe80::212:7404:4:404
00:00.674 ID:1 ÀÀ[INFO: Main] - Routing: RPL Lite
00:00.675 ID:4 [INFO: Sky] CC2420 CCA threshold -45
00:00.676 ID:1 ÀÀ[INFO: Main] - Net: sicslowpan
00:00.679 ID:1 ÀÀ[INFO: Main] - MAC: CSMA
00:00.682 ID:1 ÀÀ[INFO: Main] - 802.15.4 PANID: 0xabcd
00:00.686 ID:1 ÀÀ[INFO: Main] - 802.15.4 Default channel: 26
00:00.688 ID:1 ÀÀ[INFO: Main] Node ID: 1
00:00.692 ID:7 [INFO: Main] Starting Contiki-NG-develop/v5.0-23-g90f929580
00:00.694 ID:1 ÀÀ[INFO: Main] Link-layer address: 0012.7401.0001.0101
00:00.694 ID:7 [INFO: Main] - Routing: RPL Lite
00:00.697 ID:7 [INFO: Main] - Net: sicslowpan
00:00.699 ID:7 [INFO: Main] - MAC: CSMA
00:00.703 ID:7 [INFO: Main] - 802.15.4 PANID: 0xabcd
00:00.706 ID:1 ÀÀ[INFO: Main] Tentative link-local IPv6 address: fe80::212:7401:...
00:00.706 ID:7 [INFO: Main] - 802.15.4 Default channel: 26
Filter:
Mote output
File Edit View
Time Mote Message
00:00.688 ID:1 ÀÀ[INFO: Main] Node ID: 1
00:00.692 ID:7 [INFO: Main] Starting Contiki-NG-develop/v5.0-23-g90f929580
00:00.694 ID:1 ÀÀ[INFO: Main] Link-layer address: 0012.7401.0001.0101
00:00.694 ID:7 [INFO: Main] - Routing: RPL Lite
00:00.697 ID:7 [INFO: Main] - Net: sicslowpan
00:00.699 ID:7 [INFO: Main] - MAC: CSMA
00:00.703 ID:7 [INFO: Main] - 802.15.4 PANID: 0xabcd
00:00.706 ID:1 ÀÀ[INFO: Main] Tentative link-local IPv6 address: fe80::212:7401:...
00:00.706 ID:7 [INFO: Main] - 802.15.4 Default channel: 26
00:00.708 ID:7 [INFO: Main] Node ID: 7
00:00.710 ID:1 ÀÀ[INFO: Sky] CC2420 CCA threshold -45
00:00.714 ID:1 ÀÀ[INFO: RPL BR] Contiki-NG Border Router started
00:00.714 ID:7 [INFO: Main] Link-layer address: 0012.7407.0007.0707
00:00.717 ID:1 ÀÀ[INFO: BR] RPL-Border router started
00:00.725 ID:7 [INFO: Main] Tentative link-local IPv6 address: fe80::212:7407:7:707
00:00.729 ID:7 [INFO: Sky] CC2420 CCA threshold -45
00:01.005 ID:5 [INFO: Main] Starting Contiki-NG-develop/v5.0-23-g90f929580
00:01.008 ID:5 [INFO: Main] - Routing: RPL Lite
00:01.010 ID:5 [INFO: Main] - Net: sicslowpan
00:01.012 ID:5 [INFO: Main] - MAC: CSMA
00:01.016 ID:5 [INFO: Main] - 802.15.4 PANID: 0xabcd
00:01.019 ID:5 [INFO: Main] - 802.15.4 Default channel: 26
00:01.022 ID:5 [INFO: Main] Node ID: 5
00:01.027 ID:5 [INFO: Main] Link-layer address: 0012.7405.0005.0505
00:01.038 ID:5 [INFO: Main] Tentative link-local IPv6 address: fe80::212:7405:5:505
00:01.042 ID:5 [INFO: Sky] CC2420 CCA threshold -45
00:01.189 ID:3 [INFO: Main] Starting Contiki-NG-develop/v5.0-23-g90f929580
00:01.192 ID:3 [INFO: Main] - Routing: RPL Lite
00:01.195 ID:3 [INFO: Main] - Net: sicslowpan
00:01.197 ID:3 [INFO: Main] - MAC: CSMA
00:01.200 ID:3 [INFO: Main] - 802.15.4 PANID: 0xabcd
00:01.204 ID:3 [INFO: Main] - 802.15.4 Default channel: 26
00:01.206 ID:3 [INFO: Main] Node ID: 3
00:01.211 ID:3 [INFO: Main] Link-layer address: 0012.7403.0003.0303
00:01.223 ID:3 [INFO: Main] Tentative link-local IPv6 address: fe80::212:7403:3:303
00:01.226 ID:3 [INFO: Sky] CC2420 CCA threshold -45
00:01.716 ID:1 ÀÀ?PÀÀ[INFO: BR] Waiting for prefix
00:02.488 ID:6 [INFO: App] Not reachable yet
00:02.639 ID:7 [INFO: App] Not reachable yet
00:02.716 ID:1 ÀÀ?PÀÀ[INFO: BR] Waiting for prefix
00:02.952 ID:5 [INFO: App] Not reachable yet
00:03.716 ID:1 ÀÀ?PÀÀ[INFO: BR] Waiting for prefix
Filter:

Research for Optimal Solutions : Hardware

Devices:	nRF52840	AT86RF233	CC2652R
RF performance & Processor	1 Mbps @2.4 GHz, up to 100m range [High data rate], 32-bit Cortex-M4 processor	250 kbps @ 2.4 GHz, ~30–50m range, ATmega2560-based processor	1 Mbps @ 2.4 GHz, up to 100m range [Longe Range, Robust Signal], 32-bit ARM Cortex-M4
Integration Ease	USB, GPIO, QSPI, I2C, UART — plug-and-play	SPI interface, moderate setup [Need to build PCB]	GPIO, UART, I2C — needs TI toolchain
Power Efficiency	4.6 µA (sleep), 5.3 mA (TX @ 0 dBm) [Ultra Low Power]	~9 mA (TX), ~2 µA (sleep) [Low Power, suitable for battery use]	6.9 mA (TX @ 0 dBm), ~1 µA (sleep) [low Power, optimized for IoT]
Scalability	Up to 1000+ nodes in mesh (Thread)	≤50 nodes practical in 802.15.4	100–250 nodes in Zigbee network
Multiprotocol Support	BLE 5, Zigbee, Thread, ANT	IEEE 802.15.4 only	BLE 5, Zigbee 3.0, Thread
Cost	800-1000 Rs.	350-550 Rs.	1450-1800 Rs.
Software Ecosystem	RIOT, Zephyr, OpenThread	RIOT, Contiki, Arduino	SimpleLink SDK, TI-RTOS, Contiki, RIOT
Ease of access [Availability]	Widely available via Mouser, Digikey, Robu.in, Amazon	Easily available via Robu.in, Amazon, local shops	Moderate availability, mostly via TI distributors or specialty stores

Hardware Selection: nRF52840

After extensive research we concluded to go with nRF52840:

- Supports IEEE 802.15.4 for 6LoWPAN.
- Powered by 32-bit Cortex-M4 processor.
- Reliable 2.4 GHz multiprotocol radio.
- Compatible with OpenThread, Zephyr stacks

Research for Optimal Solutions : Software

Softwares:	OpenThread	RIOT OS	Contiki-ng	Zephyr
Compatibility with nRF52840	All features and peripherals work seamlessly with no configuration issues.	Most features are functional with minimal configuration needed.	Many features unsupported; requires heavy customization or workarounds.	All features and peripherals work seamlessly with no configuration issues.
Ease of Configuration	Very Easy setup with CLI tools and GUI tools	Easy Setup	Complex because the whole stack needs config	Easy Setup
Available Literature	Recently updated and in ample quantity	Recently updated but not sufficient	Well researched,extensive but outdated	Recently updated but not sufficient
Functionality	Thread protocol, CLI, CLI UART, border router	Multi-threading, network stacks	6LoWPAN, RPL, IPv6 routing	Full RTOS: scheduler, drivers, TCP/IP
Community Support	Highly Active and Large [Open Sourced by Google]	Moderately Active	Focused but Smaller	Established and Strong

Software Solution: Open Thread

After extensive research we concluded to go with OpenThread:

- Implements Thread using 6LoWPAN protocol
- Built on IEEE 802.15.4 standard
- Optimized for low-power mesh networks
- Backed by Google, open-source, well-supported
- Seamless integration with nRF52840 SoC

Constructing the System

Full Guide : [Construct 6LoWPAN with Rpi and nrF dongle]

Requirements: 2 Raspberry pi 3B+, 2 nRF52840 Dongles

- Inserted dongles into Raspberry Pi devices
- Built firmware using official nRF libraries in python i.e. nrfutil and OT modules [Git Repo: https://github.com/openthread/openthread]
- Flashed and logged into OT dongles
- Configured devices, channels, and addresses
- Established 6LoWPAN over IEEE 802.15.4
- Successfully sent UDP packet across network

Configuration and Testing

We successfully configured our devices by building and flashing required firmwares. Then assigned required network details. And then tested network by sending UDP packet from router to end device and vice versa

- Configuration:

  Initialize network dataset
  dataset init new

  Set channel and network key
  dataset channel 11
  dataset network key 00112233445566778899aabbccddeeff

  Start network and bring interface up
  dataset commit active
  ifconfig up
  thread start

Insights of Configuration and Testing - Border Router

```
Routes:
::/0 s med e400
Services:
44970 01 50000500000e10 s e400 0
44970 5d fd0c27129c14cbf3dff77eeeadfbba7ed122 s e400 1
Contexts:
fd4d:1f64:e58b:1::/64 1 sc
Commissioning:
22847 - - -
Done
> ifconfig up
\Done
> thread start
Error 35: InvalidCommand
 \thread start
Done
> thread start
Done
> ipaddr
fd0c:2712:9c14:cbf3:0:ff:fe00:fc11
fd4d:1f64:e58b:1:cc92:1971:e96e:4fb
fd0c:2712:9c14:cbf3:0:ff:fe00:fc10
fd0c:2712:9c14:cbf3:0:ff:fe00:e400
fd0c:2712:9c14:cbf3:dff7:7eee:adfb:ba7e
fe80:0:0:0:9c98:6354:5bd7:5821
Done
> dataset active -x
0e08000000000000100004a03000016350600040001fffe00208b373a8805891d4e70708fd0c27129c14cbf3030f4f70656e5468726561642d666537650102fe7e0410935e6c41976627f08e436ad7
1fd6745c0c0402a0f7f8000300000b05100011223344556677889 9aabbccddeeff
Done
> state
leader
Done
> udp open
Done
> udp bind :: 1234
Done
> udp senfe80:0:0:0:bc3f:4634:181d:7521 1234 hello_device
Done
> 12 bytes from fe80:0:0:0:bc3f:4634:181d:7521 1234 hello_router
```

Show desktop

Insights of Configuration and Testing - Child Node

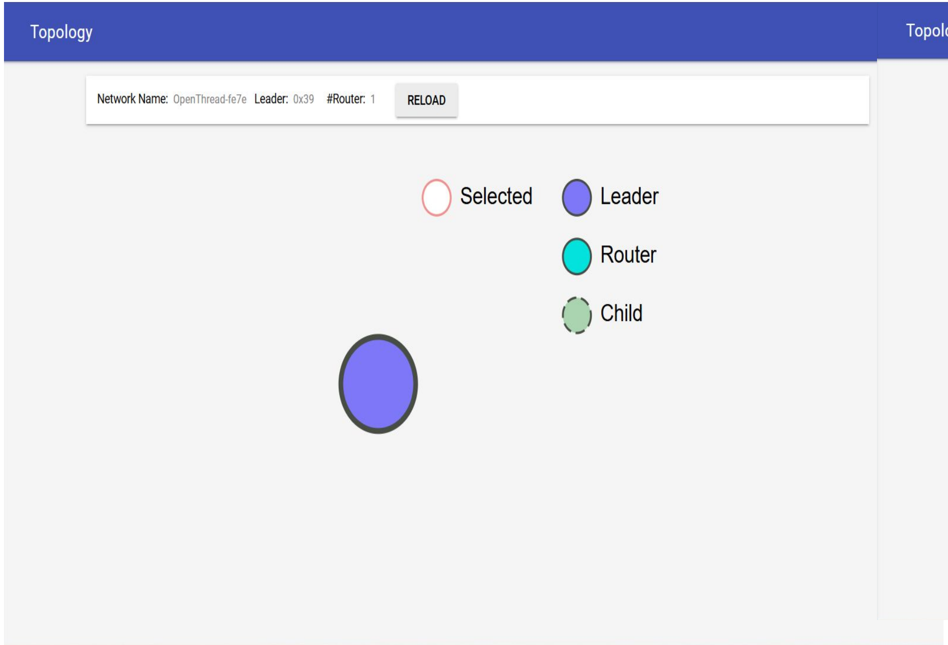
```
Routes:
::/0 s med e400
Services:
44970 01 51000500000e10 s e400 0
44970 5d fd0c27129c14cbf3dff77eeeadfbba7ed123 s e400 1
Contexts:
fd4d:1f64:e58b:1::/64 1 sc
Commissioning:
22848 - - -
Done
> ipaddr
fd4d:1f64:e58b:1:5def:1ee5:8d50:eb71
fd0c:2712:9c14:cbf3:0:ff:fe00:e401
fd0c:2712:9c14:cbf3:2994:8fd2:c37:8b06
fe80:0:0:0:bc3f:4634:181d:7521
Done
> udp open
Done
> socket bind
Error 35: InvalidCommand
>
>
>
> udp socket bind 1234
Error 35: InvalidCommand
> udp bind 1234
Error 6: Parse
>udp bind::1234
Error 35: InvalidCommand
> udp bind::1234
Error 35: InvalidCommand
> udp bind::1234
Error 35: InvalidCommand
>
Error 35: InvalidCommand
> udp bind :: 1234
Done
> 12 bytes from fe80:0:0:0:9c98:6354:5bd7:5821 1234 hello_device
udp send fe80:0:0:0:9c98:6354:5bd7:5821 1234 hello_router
Done
> |
```

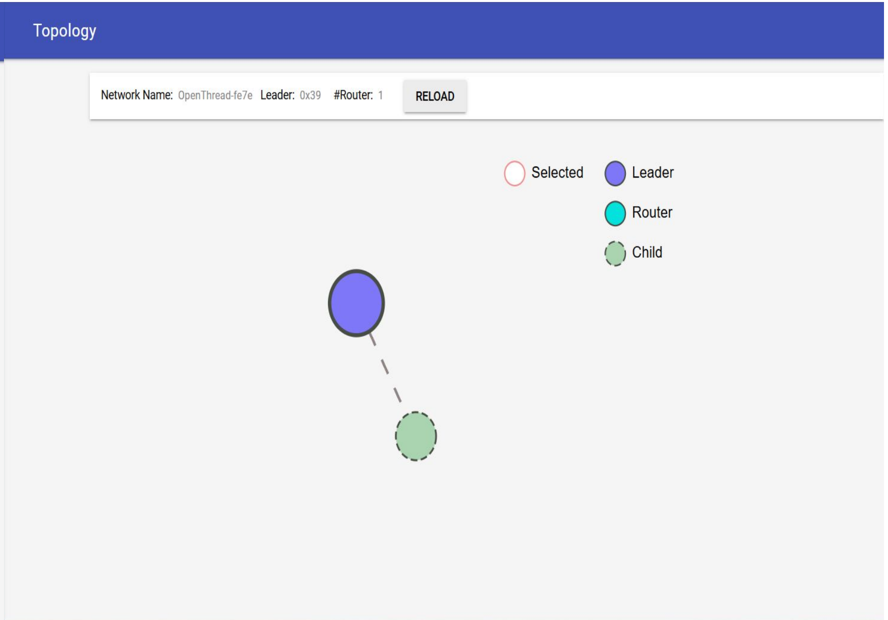
Show desktop

Physical Deployment of System

Network Topology

Given are two states of network. When there is only one node i.e. Border Router in system and one when there is End Device i.e. child also the animation





Future Work

We have identified these things to improve in the future:

1. Expand the network size to 5.
2. Add sensors to the devices on the network and send the sensor data to the border router.
3. Test OpenThread sleepy devices.
4. Test the routing protocol and routes with sleep cycling devices.

References

1] C. Ellmer, *OpenThread vs. Contiki IPv6: An Experimental Evaluation*, Master's Thesis, Uppsala University, 2017.

[2] H.-S. Kim, S. Kumar, and D. E. Culler, “Thread/OpenThread: A Compromise in Low-Power Wireless Multihop Network Architecture for the Internet of Things,” *IEEE Communications Magazine*, vol. 57, no. 7, pp. 55–61, 2019. doi:10.1109/MCOM.2019.1800788.

[3] “Raspberry Pi and 6LoWPAN,” netidee, 2018. [Online]. Available: https://www.netidee.at/comatrix/raspberry-pi-and-6lowpan

[4] “Is 6LoWPAN the best for future of IoT Technology?” Smowcode Blog, 2024. [Online]. Available: https://blog.smowcode.com/is-6lowpan-the-best-for-future-of-iot-technology