



Wireless Communication

Introduction to Embedded System



Contents

- Introduction to Wi-Fi
- Wi-Fi Technology
- IEEE 802.11 Wireless LAN
- Network topologies
- Wi-Fi Security
- ESP32 Wi-Fi features
- ESP32 Wi-Fi Arduino functions



Contents

- Introduction to Bluetooth
- Bluetooth Class
- Bluetooth Version
- Bluetooth usage scenario
- Bluetooth Low Energy
- ESP32 Bluetooth
- ESP32 Bluetooth Classic and BLE

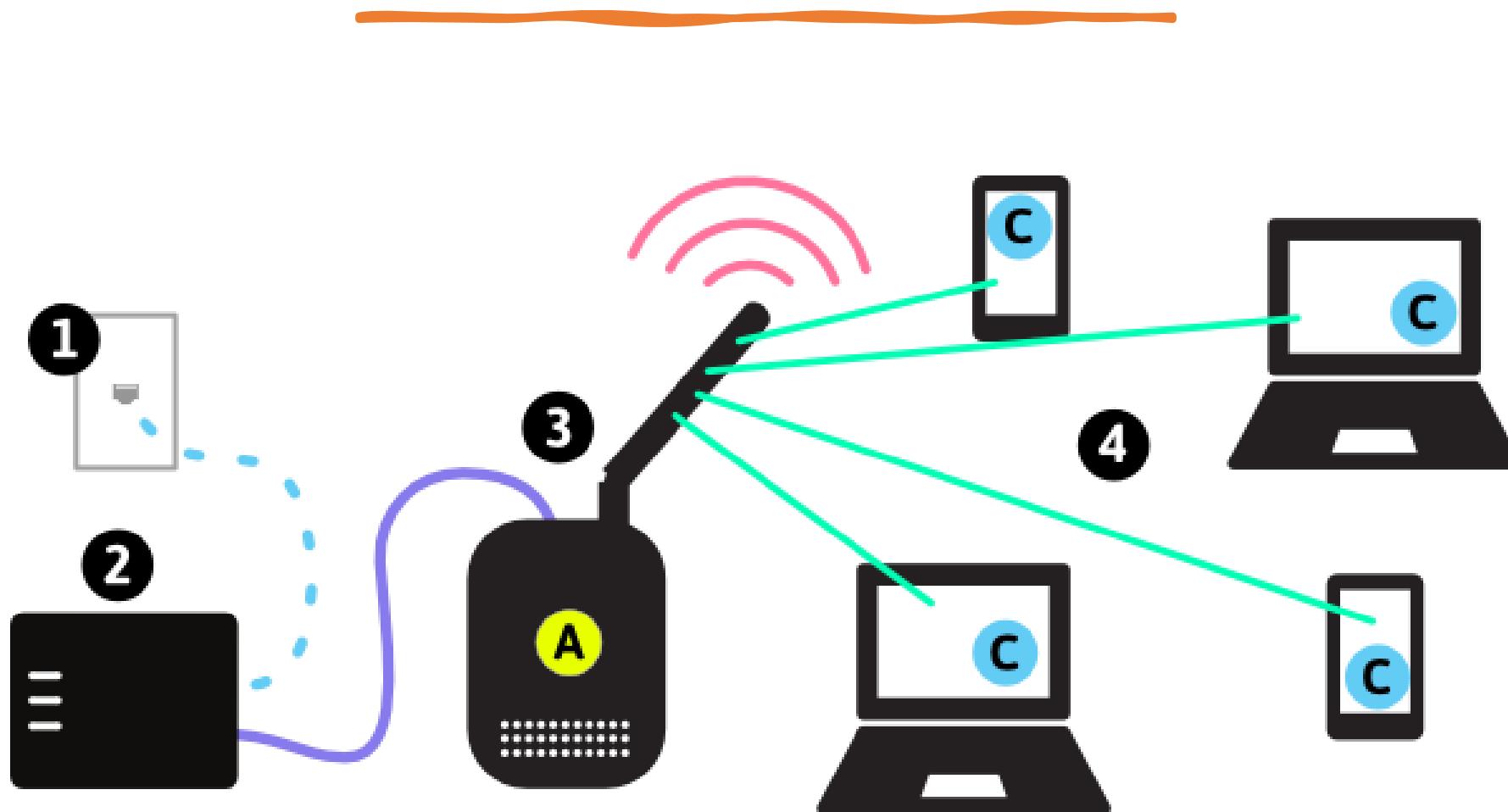
Introduction

- ในปัจจุบัน ความต้องการในการสื่อสารแบบไร้สาย เริ่มมีมากขึ้นมากกว่าแต่ก่อน โดยเฉพาะอย่างยิ่งกับอุปกรณ์พกพาขนาดเล็ก ทำให้การเชื่อมต่อโดยใช้สาย อาจไม่ใช่คำตอบที่เหมาะสมที่สุดอีกต่อไป
- หนึ่งในเทคโนโลยีการสื่อสารแบบไร้สายที่ได้รับความนิยมในปัจจุบัน คือ Wi-Fi ซึ่งเป็นการเชื่อมต่อคอมพิวเตอร์ ไปยังอุปกรณ์อื่นๆ ในเครือข่าย หรือใช้ในการเชื่อมต่อไปยังเครือข่ายอินเทอร์เน็ต
- การทำงานของ Wi-Fi จะอ้างอิงกับมาตรฐานการสื่อสาร IEEE 802.11 ซึ่งเป็นมาตรฐานเกี่ยวกับโครงข่ายสื่อสารแบบไร้สาย (Wireless Local Area Network: WLAN)

Purpose of WIFI

- วัตถุประสงค์ของการสื่อสารแบบไร้สายแบบ Wi-Fi คือ การลดความซับซ้อนลง โดยให้
- เป้าหมายของการสื่อสาร Wi-Fi มีดังนี้
 - ทำให้การเข้าถึงข้อมูลง่ายขึ้น
 - ลดการใช้สายสัญญาณ
 - ลดการใช้อุปกรณ์อื่นๆ ที่สินเปลือ เช่น สวิตช์, หัวแปลง, ขาเชื่อมต่อ เป็นต้น

WIFI = Wireless LAN



Wireless WAN (Wide Area Network)

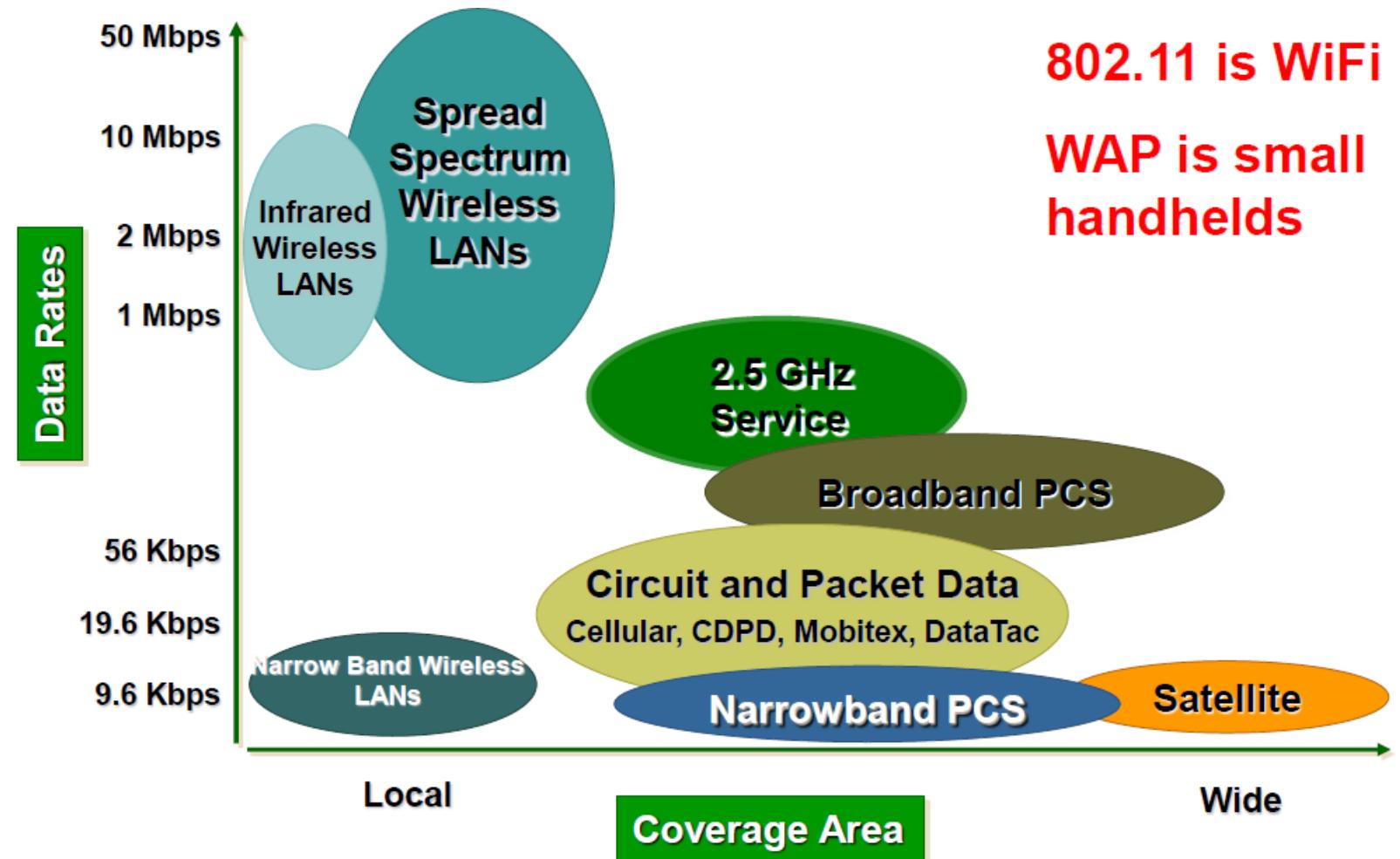


Wireless Landscape



Wireless Technology	Transmission Distance	Speed
Bluetooth	33 feet	1 Mbps
Satellite	Worldwide	290ms latency
1G Analog cellular	Nationwide	
2G digital cellular	Nationwide	14 Kbps
2.5G digital cellular	Nationwide	384 Kbps
3G digital cellular	Nationwide	2-10 Mbps
WLAN 802.11b	375 feet	11 Mbps
WLAN 802.11a, g	300 feet	54/128 Mbps
Fixed broadband Wireless (BWA)	35 miles	1 Gbps
WAP	Nationwide	384 Kbps
WiMAX 802.16, 4G	10 miles	75 Mbps

Wireless Landscape



Wi-Fi Technology

- การสื่อสารของ Wi-Fi จะใช้คลื่นวิทยุในแบบย่านความถี่ที่ถูกนิยามเพื่อให้ใช้งานทางด้าน อุตสาหกรรม วิทยาศาสตร์ และ การแพทย์ หรือที่เรียกว่า ISM (Industrial, Scientific, and Medical) band ซึ่งเป็นกลุ่มแบบความถี่ที่ไม่มีการกำหนดการใช้งานเฉพาะทาง อาจกล่าวได้ว่า เราสามารถใช้ได้โดยที่ไม่ต้องมีใบอนุญาต
 - อย่างไรก็ตาม คลื่นวิทยุในแบบความถี่เหล่านี้ อาจมีการกำหนดใช้งานเฉพาะที่แตกต่างกันในแต่ละประเทศ ผู้ใช้จะต้องศึกษาข้อมูลในส่วนนี้โดยละเอียด เพื่อป้องกันข้อพิพาททางกฎหมาย
- โดยปกติแล้ว คลื่นความถี่ย่าน ISM band สามารถแบ่งแบบย่านใช้งานได้ 3 แบบ คือ
 - ย่าน 900 MHz สามารถเลือกใช้ได้ตั้งแต่ 902 – 928 MHz
 - ย่าน 2.4 GHz สามารถเลือกใช้ได้ตั้งแต่ 2.4 – 2.4853 GHz
 - ย่าน 5.4 GHz โดยมีย่านใช้งานได้ตั้งแต่ 5.275 – 5.85 GHz
- เครือข่ายไร้สายแบบ Wi-Fi จะใช้คลื่นความถี่ย่าน 2.4 GHz และ 5.4 GHz เป็นหลัก (ยกเว้น 802.11ah ที่เป็นเครือข่ายไร้สายแบบ Wi-Fi ที่ใช้พลังงานต่ำ ซึ่งจะกล่าวถึงต่อไป)

Wi-Fi Technology

- เทคโนโลยีของ Wi-Fi ในปัจจุบัน สามารถแบ่งออกได้ตาม Version ต่างๆ ได้ดังตารางต่อไปนี้

Generation	IEEE Standard	Maximum Linkrate
Wi-Fi 6	802.11ax	600–9608 Mbit/s
Wi-Fi 5	802.11ac	433–6933 Mbit/s
Wi-Fi 4	802.11n	72–600 Mbit/s
Wi-Fi 3	802.11g	3–54 Mbit/s
Wi-Fi 2	802.11a	1.5 to 54 Mbit/s
Wi-Fi 1	802.11b	1 to 11 Mbit/s

11AX

THE PATH TO TRULY BRILLIANT WI-FI



4x

BETTER IN DENSE ENVIRONMENTS

Improve average throughput per user by at least four times in dense or congested environments



FASTER THROUGHPUT

Deliver up to 40 percent higher peak data rates for a single client device



INCREASE NETWORK EFFICIENCY

By more than four times



EXTEND BATTERY LIFE

Of client devices

802.11b (WIFI1)

- เป็น Wi-Fi มาตรฐานแรกสุดที่เปิดใช้งาน โดยเริ่มใช้งานครั้งแรกในปี 1999
- ทำงานที่คลื่นความถี่ป่า 2.4 GHz
- มีอัตราเร็วในการสื่อสารสูงสุดที่ 11 Mbps (ตามทฤษฎี) ภายในระยะ 30 เมตร
- ความเร็วในการใช้งานจริงโดยเฉลี่ยแล้ว จะอยู่ที่ประมาณ 4-6 Mbps
- มีระยะการสื่อสารที่ 100 ถึง 150 ฟุต
- ได้รับความนิยมในการใช้งานอย่างมากในเวลานั้น เนื่องจากมีต้นทุนไม่สูงมาก (ในขณะนั้น)
- มีข้อเสียในเรื่องของคลื่นที่ใช้งาน เนื่องจากอาจถูกรบกวนจากอุปกรณ์อื่นๆ ได้ เช่น โทรศัพท์มือถือ, อุปกรณ์ที่รองรับการเชื่อมต่อ Bluetooth ซึ่งมีผลต่อความเร็วสูงสุดในการสื่อสาร

802.11a (WIFI2)

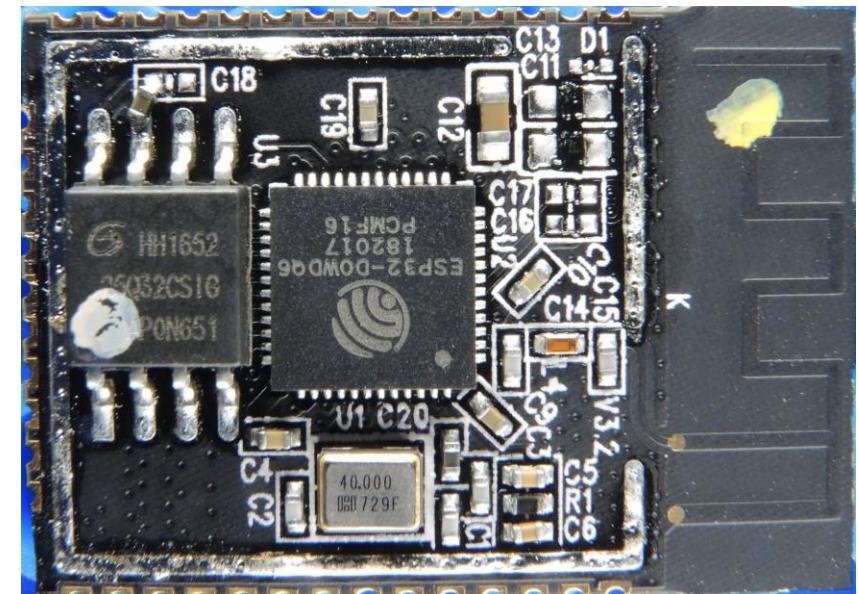
- เริ่มประการใช้ครั้งแรกในปี 2001
- ใช้คลื่อนความถี่ในย่าน 5 GHz ซึ่งแตกต่างจาก 802.11b ที่ใช้คลื่อนย่าน 2.4 GHz
- รองรับความเร็วในการสื่อสารสูงสุดที่ 54 Mbps
- มีระยะการสื่อสารสูงสุดที่ 50-75 พุต ซึ่งมีระยะที่สั้นกว่ากลุ่มที่ใช้คลื่น 2.4 GHz
- ไม่ได้รับความนิยมในการใช้งาน เนื่องจากมีค่าใช้จ่ายที่สูงกว่าในขณะนั้น และยังไม่รองรับการใช้งานร่วมกับ 802.11b อีกด้วย

802.11g (WIFI3)

- เป็นการนำข้อดีของมาตรฐาน 802.11b และ 802.11a มาใช้ร่วมกัน โดยประกาศใช้ครั้งแรกในปี 2003
- ใช้คลื่อความถี่ 2.4 GHz
- รองรับความเร็วสูงสุดที่ 54 Mbps
- ระยะการสื่อสารสูงสุดที่ 100-150 ฟุต
- รองรับการใช้งานร่วมกับ 802.11b

802.11n (WIFI4)

- เป็นมาตรฐาน Wi-Fi ตัวแรกที่รองรับการใช้เทคโนโลยี MIMO (Multiple-input and Multiple-output) ซึ่งก็คือ การใช้เส้าส่งสัญญาณพร้อมกันหลายคลื่น และเครื่องรับสามารถรับสัญญาณพร้อมกันได้หลายเครื่อง ช่วยให้สามารถสื่อสารได้เร็วขึ้น
- เปิดใช้งานครั้งแรกในปี 2009
- มีระยะการสื่อสารสูงสุดที่ 300 ฟุต
- รองรับความเร็วสูงสุดที่ 600 Mbps (ขึ้นอยู่กับอุปกรณ์ที่ใช้)
- รองรับการใช้คลื่นความถี่ที่ย่าน 2.4 GHz และ 5 GHz



802.11ac (WIFI5)

- ตีพิมพ์มาตรฐานหรือใช้งานเมื่อปี ค.ศ. 2013
- ใช้คลื่นความถี่ที่ 5 GHz มีอัตรา ความเร็วในการรับส่งข้อมูลระดับ Gbps โดยสามารถแบ่งเป็น 2 ประเภท คือ
 1. 802.11ac Wave 1 มีความเร็ว รับส่งข้อมูลสูงสุดถึงประมาณ 1.3 Gbps มีแบนด์วิธที่กว้างสูงสุด 80 Mhz โดยทำงานแบบเทคโนโลยี SUMIMO (Single User - MIMO) โดย access point จะมีความสามารถส่งข้อมูลให้ผู้ใช้งานได้ที่ละ 1 เครื่องเท่านั้น ซึ่งเมื่อมีผู้ใช้งานพร้อมกันมากขึ้นอัตราการรับส่งข้อมูล (Throughput) โดยรวมของเครือข่ายจะลดลงอย่างมาก
 2. 802.11ac Wave 2 จะมีความเร็วรับส่งข้อมูลสูงสุดถึงประมาณ 3.4 Gbps มีแบนด์วิธที่กว้างสูงสุดถึง 160 Mhz ใช้เทคโนโลยี MU-MIMO (Multi User - MIMO) ที่จะช่วยให้ access point สามารถส่งข้อมูลไปให้ผู้ใช้ได้หลายๆเครื่องพร้อมกัน

802.11ax (WIFI6)



802.11ax

Easing Wi-Fi congestion in dense environments



expands capacity

extends coverage

enhances experiences



802.11ax (WIFI6)

- Wi-Fi 6 นั้น มีเทคโนโลยีและฟีเจอร์ใหม่ ๆ ที่พัฒนาขึ้นจาก Wi-Fi 5 หลายอย่าง แต่ในส่วนของข้อมูลเชิงลึก หรือเชิงเทคนิคที่เข้าใจได้ยากนั้น จะไม่กล่าวถึงในบทความนี้ เพราะไม่มีประโยชน์กับผู้ใช้งานทั่วไป จึงขอยกแค่ ส่วนที่เป็นคีย์ฟีเจอร์หลัก ๆ มาอธิบาย ได้แก่
 - Orthogonal Frequency Division Multiple Access (OFDMA)
 - Multi-user MIMO (MU-MIMO)
 - Target wake time (TWT)
 - 1024 quadrature amplitude modulation mode (1024-QAM)

5G Latency 5ms

WIFI6 Latency 6ms



802.11ax (WIFI6)

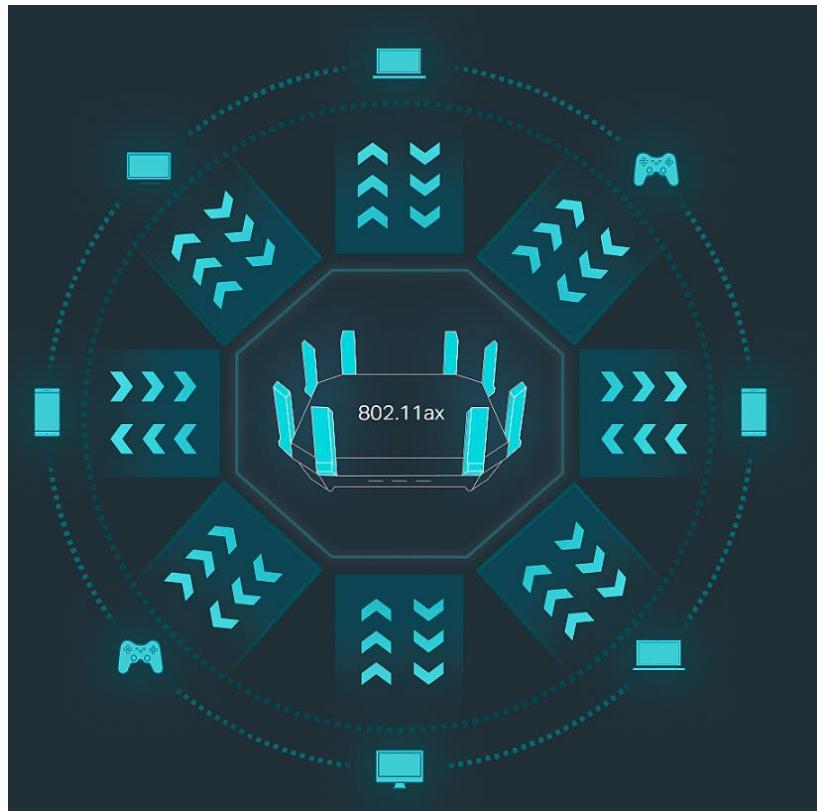
OFDMA



<https://www.tp-link.com/th/wifi6/>

802.11ax (WIFI6)

MU-MIMO



4x More Capacity for More Devices

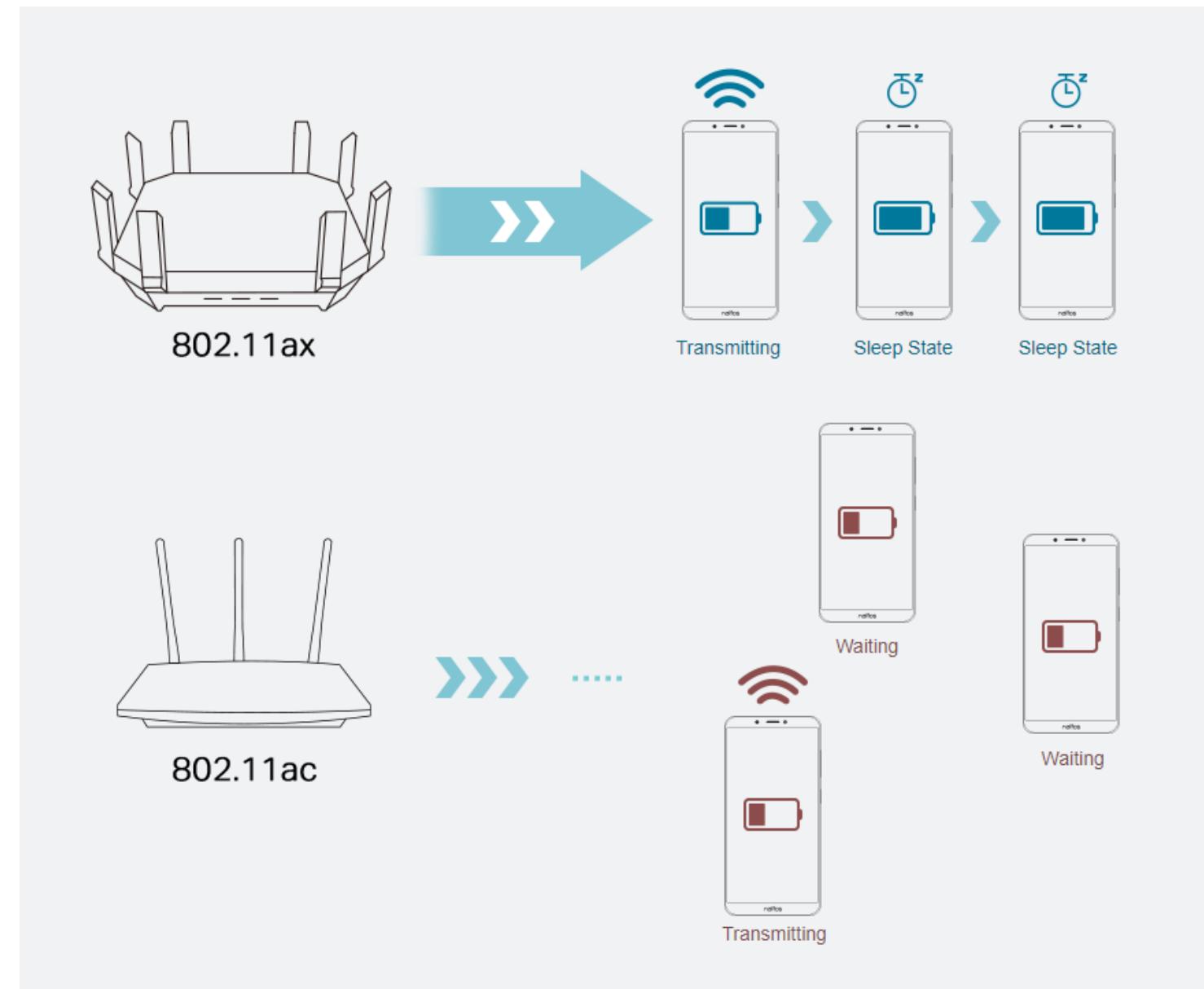
Ideal for Crowded Networks

802.11ax 8x8 DL/UL MU-MIMO OFDMA BSS Color 4X

802.11ac MU-MIMO

802.11ax (WIFI6)

Target wake time (TWT)



<https://www.tp-link.com/th/wifi6/>

802.11ax (WIFI 6E)

- WiFi 6E เป็นเทคโนโลยีต่อยอดจาก WiFi 6 โดยจะไม่ได้รองรับการเชื่อมต่อแค่สองคลื่นแล้ว คือ 2.4GHz+5GHz แต่จะเพิ่มขึ้นมาอีกหนึ่งช่องคลื่นสัญญาณคือ 6GHz ที่มีช่องสัญญาณกว้างขึ้น และรองรับการจำนวนการเชื่อมต่อมากกว่าเดิม ให้ผลลัพธ์โครงข่ายได้อย่างมีประสิทธิภาพทุกๆ สถานะแวดล้อม ประเด็นที่น่าสนใจคือทางกลุ่มได้ใช้尼尼ามว่า (very dense and congested environments.) นั่นหมายความว่า สภาพแวดล้อมที่หนาแน่นและแออัดมาก ซึ่งจากที่ยกตัวอย่างข้างต้น คือ สนามกีฬา, สถานที่สาธารณะต่าง ๆ อาทิ สถานีรถไฟ/รถไฟฟ้า, สนามบิน, สถานีขนส่งผู้โดยสาร, ห้างสรรพสินค้า, โกลด์ингเก็บสินค้า, โรงงาน อุตสาหกรรม เป็นต้น ก็จะให้ผลลัพธ์ที่ดีขึ้นกว่าเดิม และสมจริง (closer to reality) หรือโลกของ AR/VR นั่นเอง



802.11ax (WIFI 6E)



11ax networks

- 6th generation
- 2.4 GHz and 5 GHz
- 80 MHz channels
- 1 Gbps in phones



11ax **EXTENDED** to 6 GHz

- 6th generation **EXTENDED**
- 2.4 GHz, 5 GHz, and **6 GHz**
- **160 MHz** channels
- **2 Gbps** in phones

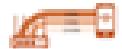
802.11be (WIFI 7)

Wi-Fi CERTIFIED™ 7: Advanced performance for next generation Wi-Fi®

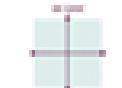
Features



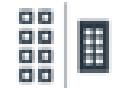
320 MHz channels



Multi-link Operation (MLO)



4K QAM



512 Compressed Block Ack



Multiple RUs to a single STA

Benefits

2X higher throughput

Deterministic latency, increased efficiency, greater reliability

20% higher transmission rates

Reduced transmission overhead

Enhanced spectral efficiency





802.11ah

- เป็นมาตรฐานสื่อสารแบบไร้สายที่พัฒนามาจาก 802.11 เดิม (WIFI)
- ออกแบบมาสำหรับงานที่ต้องใช้พลังงานต่ำ เช่น งานทางด้าน IoT เนื่องจาก 802.11 เดิมนั้น ถึงแม้ว่าจะเป็นมาตรฐานที่ใช้งานแพร่หลายที่สุด แต่มีเพรอมข้อมูลขนาดใหญ่ และใช้พลังงานมาก
- ใช้ย่านคลื่นสัญญาณที่ต่ำกว่า 1 GHz (SubGHz)
- สามารถสื่อสารได้ในระยะที่ไกลกว่า และมีคุณสมบัติทะลุทะลวงสิ่งกีดขวาง (เช่น กำแพง) ได้ดีกว่า ซึ่งเป็นมาจากการใช้งานคลื่นในย่านที่ต่ำกว่า



802.11ah



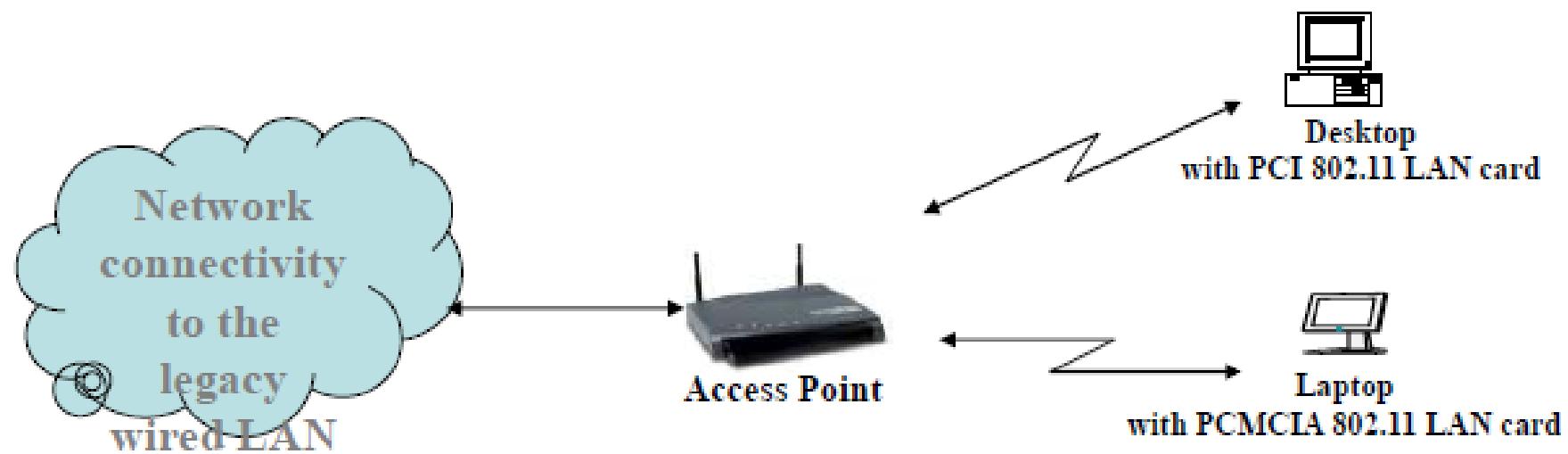
Power efficient, long range, scalable Wi-Fi

IEEE 802.11ah Spectrum Allocations

ISM ALLOCATIONS APPLICABLE FOR IEEE 802.11AH	
COUNTRY	BAND LIMITS (MHZ)
China	755 - 787
Europe	863 - 868
Japan	916.5 - 927.5
Korea	917.5 - 923.5
Singapore	866 - 869 & 920 - 925
USA	902 - 928

IEEE 802.11 Wireless LAN

- Access Point (AP) ทำหน้าที่เป็นสะพานเชื่อมต่อ เพื่อส่งข้อมูลระหว่างการสื่อสารแบบไร้สาย กับแบบมีสาย โดยมีเสาอากาศเพื่อกระจายสัญญาณให้อุปกรณ์อื่นๆ สามารถเชื่อมต่อแบบไร้สายได้



IEEE 802.11 Wireless LAN

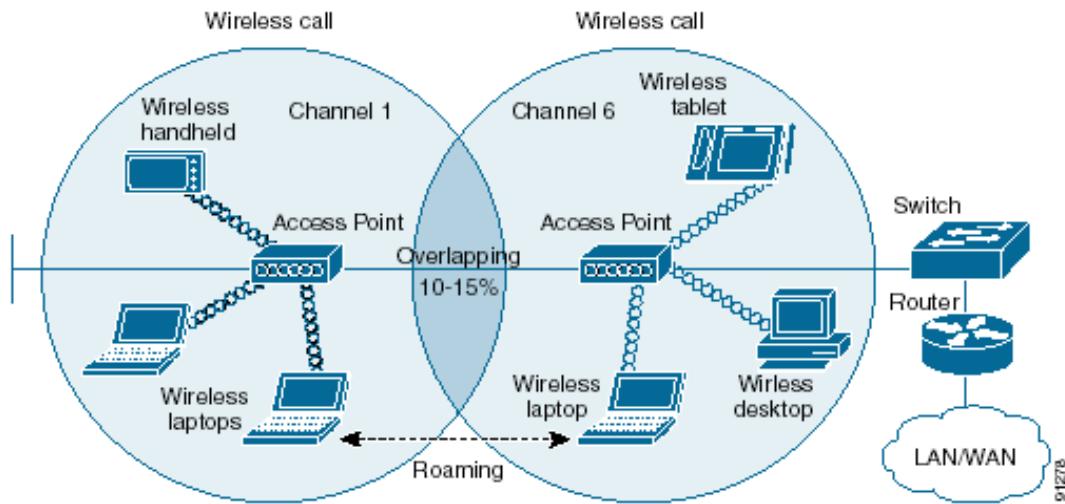
- ระยะการเชื่อมต่อของการสื่อสารแบบ WiFi จะขึ้นอยู่กับกำลังส่งของสัญญาณที่ Access Point และสิ่งกีดขวาง
- ในกรณีที่มีพื้นที่บริเวณกว้าง อาจจะต้องมีการติดตั้ง AP เป็นจำนวนหลายตัว โดยมีพื้นที่หับซ้อนของสัญญาณที่กระจายของ AP แต่ละตัวอยู่ประมาณ 20-30%
- ในกรณีที่ต้องสื่อสารระยะไกล อาจต้องมีการใช้อุปกรณ์ทวนสัญญาณ (Repeater) เข้ามาช่วย เพื่อเพิ่มระยะในการสื่อสารให้ใกล้ขึ้น (โดยอุปกรณ์ที่ใช้ อาจเป็นตัว Repeater เลย หรือ AP ที่รองรับการทำงานลักษณะนี้)
- Client สามารถเชื่อมตอกับ AP ได้เพียงตัวเดียวเท่านั้น หากมีการเคลื่อนที่ไปยัง AP ตัวอื่นที่มีระยะใกล้กว่า ก็จะเชื่อมตอกับ AP ตัวนั้น (Hand-off)

Wi-Fi Network Topologies

- การทำงานของโครงข่ายไร้สายตามมาตรฐาน IEEE 802.11 สามารถแบ่งออกได้ 3 แบบ ดังนี้
 1. AP-based topology (Infrastructure Topology)
 2. Peer-to-Peer topology (Ad-hoc Topology)
 3. Point-to-Multipoint bridge topology
 4. Mesh Topology

AP-based topology

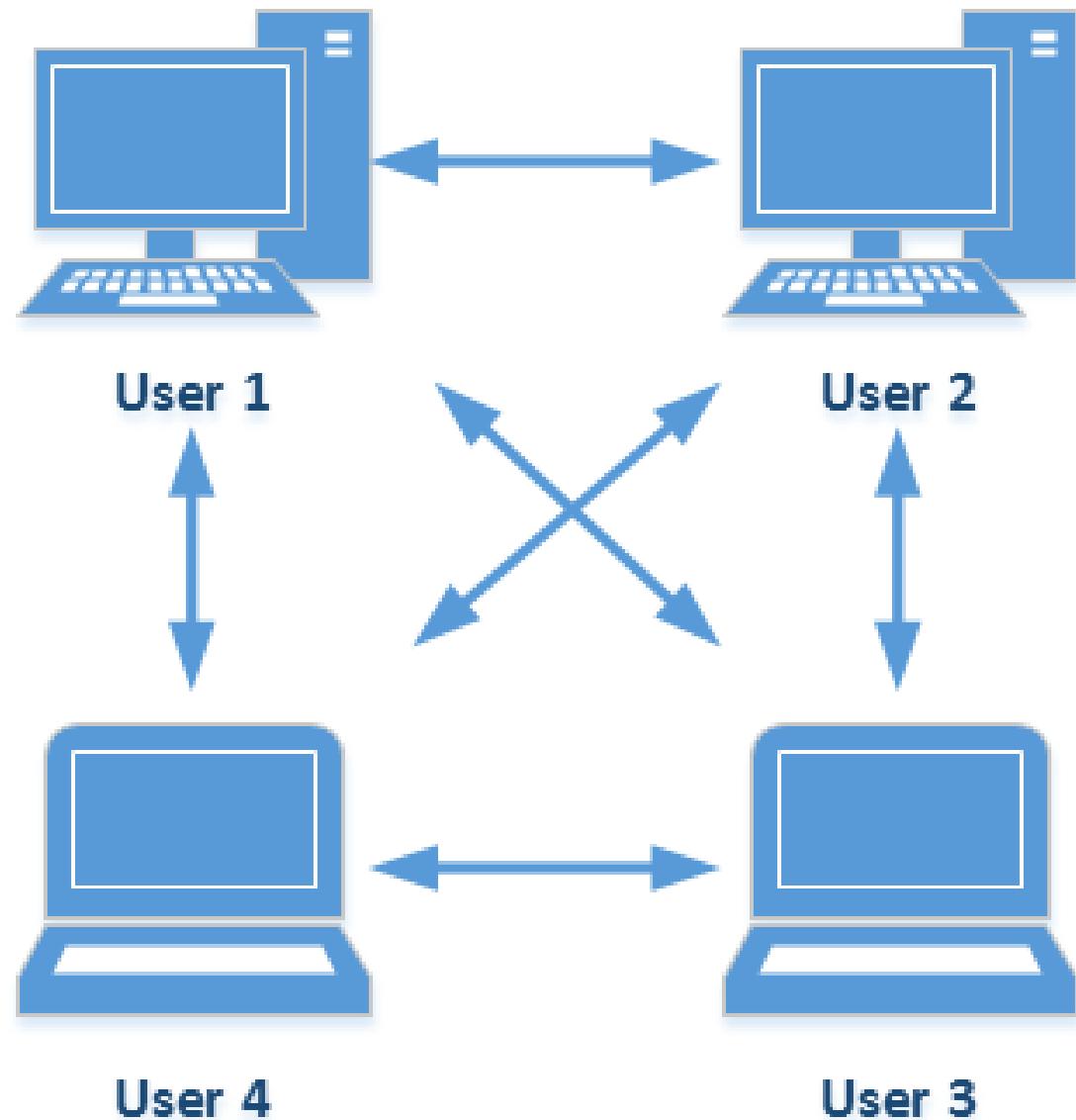
Figure 1-3 Typical WLAN



- เป็นรูปแบบที่ Client ทุกตัว จะต้องเชื่อมต่อเข้า กับอุปกรณ์ที่เรียกว่า Access point (AP) หรือ อุปกรณ์อื่นๆ ที่มีความสามารถในการเปิดการใช้ งาน Hotspot
- ในการสื่อสาร ข้อมูลจาก Client จะเดินทางผ่าน AP เพื่อทำการส่งต่อข้อมูลไปยังที่หมาย หรือส่ง ต่อไปยัง AP ตัวอื่นๆ ที่ต้องการ

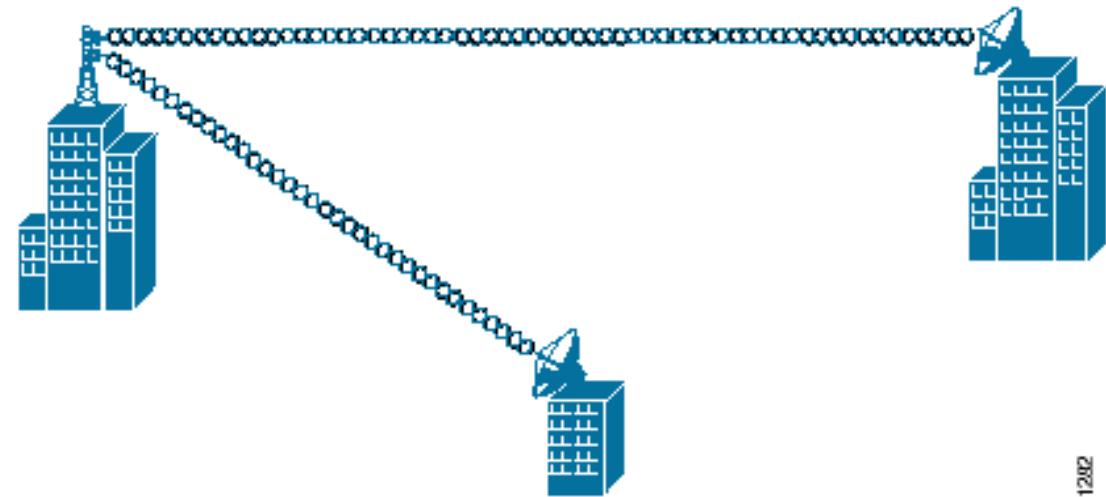
Peer-to-Peer Topology

- เป็นรูปแบบการสื่อสารที่อุปกรณ์ Client ไม่ต้องเชื่อมต่อเข้ากับ Access Point ตัวกลาง แต่ Client ทุกตัวที่อยู่ในระยะสามารถเชื่อมต่อเข้าหากัน และแลกเปลี่ยนข้อมูลกันเองได้



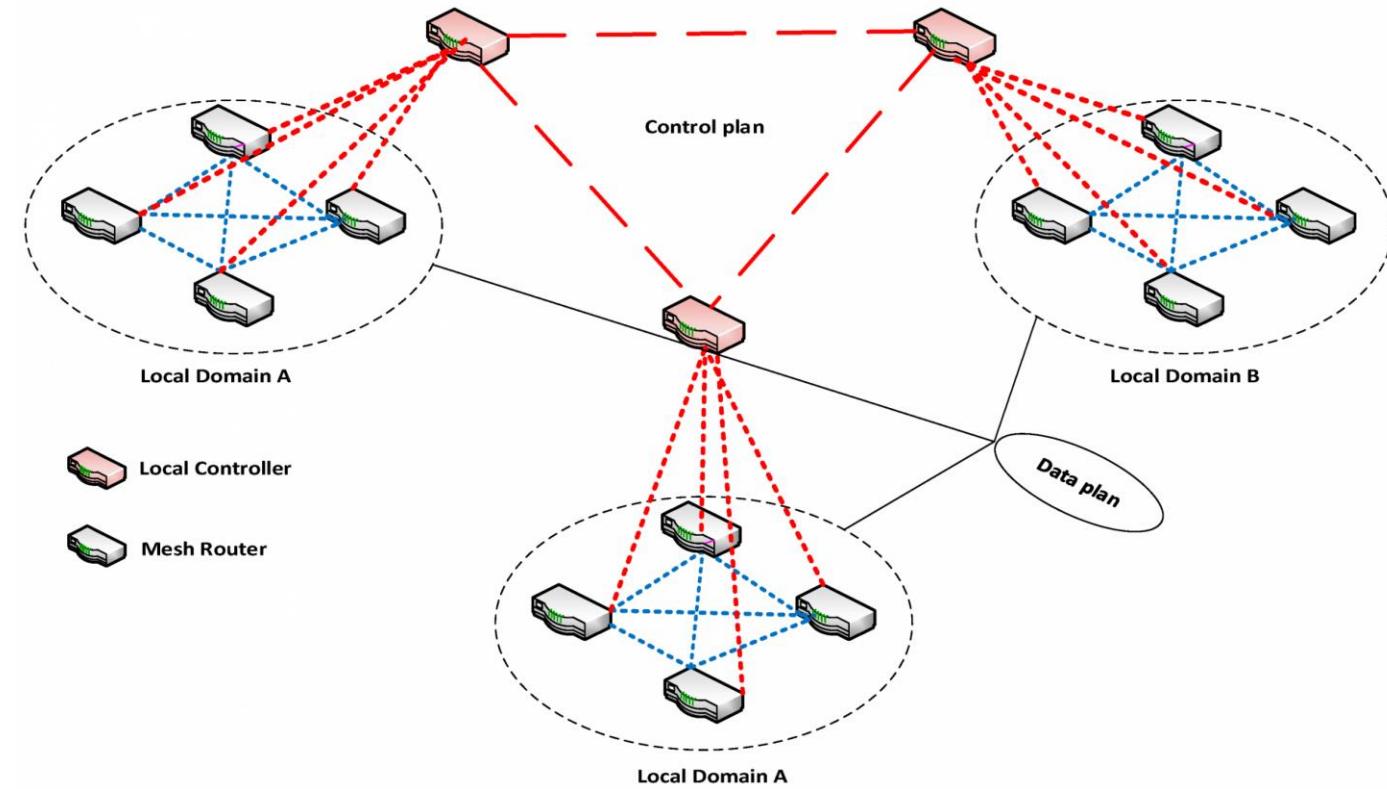
Point to Multipoint bridge topology

- เป็นรูปแบบที่ใช้สำหรับการเชื่อมต่อเครือข่ายท้องถิ่น (Local Area Network: LAN) ไปยังเครือข่าย LAN อีกกลุ่มหนึ่งเพื่อแลกเปลี่ยนข้อมูล ถึงแม้ว่าจะอยู่ระยะห่างกันก็ตาม เช่น การสื่อสารข้อมูลระหว่างอาคารหนึ่งไปยังอาคารอีกแห่งหนึ่ง ดังภาพ
- ระยะของการสื่อสาร จะขึ้นอยู่กับประเภทของเสาสัญญาณ, อุปกรณ์ที่ใช้, และสภาพแวดล้อมโดยรอบ เช่น อาคาร, สิ่งกีดขวางต่างๆ เป็นต้น



Mesh topology

- อุปกรณ์แต่ละตัวในเครือข่ายทำหน้าที่เป็น node และ เชื่อมต่อกันได้อย่างอิสระ ไม่มีศูนย์กลาง อุปกรณ์แต่ละตัว สามารถรับและส่งข้อมูลต่อไปยัง node อื่นๆ ได้
- เหมาะสมสำหรับพื้นที่ขนาดใหญ่หรือพื้นที่ที่ต้องการความ ครอบคลุมสัญญาณสูง เช่น โรงงาน สนามกีฬา หรือหมู่บ้าน
- สัญญาณครอบคลุมพื้นที่กว้าง รองรับการขยายเครือข่ายได้ ง่าย และสามารถใช้งานได้แม่บ้าง node จะหยุดทำงาน



Wi-Fi Security

- การรักษาความปลอดภัยโครงข่ายไร้สายแบบ Wi-Fi เช่น การควบคุมการเข้าใช้งาน มีองค์ประกอบที่เกี่ยวข้องดังนี้
 - Service Set Identifier (SSID)
 - Wired Equivalent Privacy (WEP)
 - Wireless Protected Access (WPA)
 - MAC Address Filtering
 - IEEE 802.11i
- การเข้ารหัสแบบ WEP และ WPA จะต้องเลือกที่อุปกรณ์ Router ในโครงข่าย และอาจต้องตรวจสอบอุปกรณ์ว่ารองรับการเข้ารหัสแบบใดบ้าง
- หมายเหตุ: WEP มีการใช้งานน้อยลง เนื่องจากถูกแทนที่ด้วย WPA ที่มีความปลอดภัยสูงกว่า
- การเข้าใช้งาน จะต้องทราบ SSID ของ AP ที่ต้องการ

ESP32 Wi-Fi

ESP32 Wi-Fi

- ESP32 รองรับเทคโนโลยี 802.11 b/g/n โดย 802.11n (WIFI4) ทำงานที่ความถี่ 2.4 GHz และความเร็วสูงสุดที่ 150 Mbps
- การใช้งาน Wi-Fi ใน ESP32 รองรับการใช้งาน 3 โหมด คือ
 1. AP (Access Point)
 - เป็นโหมดที่ ESP32 ทำหน้าที่เป็น Access Point โดยรองรับอุปกรณ์ที่เชื่อมต่อเพียง 1 ตัวเท่านั้น
 2. STA (Station)
 - เชื่อมต่อ ESP32 กับ Access Point หรือเราเตอร์ เช่นเดียวกับอุปกรณ์มือถือหรือคอมพิวเตอร์
 3. AP + STA
 - ทำงานได้ทั้งสองโหมดพร้อมกัน โดยเชื่อมต่อกับ Access Point ขณะเดียวกันก็เปิด Access Point ให้กับอุปกรณ์อื่น หมายความว่า ESP32 สามารถทำงานทั้งสองโหมดได้พร้อมกัน

ESP32 Wi-Fi

- ฟีเจอร์ Wi-Fi ของ ESP32
 - Wi-Fi Direct (P2P): สามารถเชื่อมต่อกับอุปกรณ์อื่นได้โดยตรงแบบ peer-to-peer โดยไม่ต้องมีเราเตอร์หรือ Access Point
 - SoftAP: ทำให้ ESP32 สร้าง Access Point แบบซอฟต์แวร์ขึ้นมาเพื่อให้อุปกรณ์อื่นเชื่อมต่อกับมันได้
 - ESP-NOW: โปรโตคอลเฉพาะของ ESP32 ที่อนุญาตให้ส่งข้อมูลแบบไร้สายระหว่าง ESP32 หลายตัวได้โดยไม่ต้องเชื่อมต่อกับ Access Point หรือเครือข่าย Wi-Fi
 - Over-the-Air (OTA) Updates: สามารถอัปเดตเฟิร์มแวร์ของ ESP32 ผ่าน Wi-Fi ได้โดยไม่ต้องต่อสาย USB

ESP32 Wi-Fi ArduinoIDE

ESP32 Wi-Fi Arduino Library

- การใช้งาน Wi-Fi ใน ESP32 จะต้องทำการเพิ่มไฟล์ส่วนใน Firmware ที่พัฒนาขึ้น โดยไฟล์ส่วนหัวสำหรับโมดูล Wi-Fi ของ ESP32 คือ WiFi.h (ตัวพิมพ์ใหญ่และพิมพ์เล็กต้องตรง)
 - กรณีใช้งานบน ESP8266 ไฟล์ส่วนหัวนี้จะชื่อว่า ESP8266WiFi.h

```
void WiFi.mode(wifi_mode_t mode);
```

- ใช้กำหนดโหมดการใช้งาน Wi-Fi ของ ESP32 โดยสามารถกำหนดได้ที่พารามิเตอร์ (wifi_mode_t) mode ดังนี้
 - WIFI_AP – ใช้งานในโหมด AP
 - WIFI_STA – ใช้งานในโหมด STA
 - WIFI_AP_STA – ใช้งานในโหมด AP+STA
 - WIFI_OFF – ปิดการใช้งาน WiFi
- ฟังก์ชันนี้ไม่มีการคืนค่ากลับ

ESP32 Wi-Fi Arduino Library (AP Mode)

```
void WiFi.softAP(const char* ssid, const char* passphrase = NULL, int channel = 1, int ssid_hidden = 0);
```

- เป็นฟังก์ชันสำหรับการกำหนดคุณสมบัติการใช้งานในโหมด Access Point โดยพารามิเตอร์แต่ละตัว มีรายละเอียดดังนี้
 - (const char*) ssid – ชื่อ SSID ที่ต้องการตั้ง เมื่อใช้อุปกรณ์อื่นค้นหาจะแสดงชื่อนี้
 - (const char*) passphrase – รหัสผ่านที่ต้องการกำหนด โดยจะต้องมีความยาวไม่ต่ำกว่า 8 ตัวอักษร ค่าเริ่มต้นคือ NULL (ไม่มีรหัสผ่าน)
 - (int) channel – ช่องสัญญาณ WiFi ที่ต้องการ มีค่าเริ่มต้นเป็นช่องที่ 1
 - (int) ssid_hidden – ซ่อนจากการค้นหา โดยอุปกรณ์ตัวอื่นจะมองไม่เห็น
- ฟังก์ชันนี้ไม่มีการคืนค่ากลับ

ESP32 Wi-Fi Arduino Library (AP Mode)

```
void WiFi.softAPConfig(IPAddress local_ip, IPAddress gateway, IPAddress subnet);
```

- เป็นฟังก์ชันกำหนดค่าหมายเลข IP (IP Address) ให้กับ ESP32 โดยพารามิเตอร์แต่ละตัว คือ
 - (IPAddress) local_ip – กำหนดค่า IP ให้กับ ESP32 เมื่อมีอุปกรณ์อื่น ๆ มีเชื่อมต่อกับ ESP32 จะเห็น IP ของ ESP32 ตามค่าพารามิเตอร์ที่กำหนดนี้
 - (IPAddress) gateway – กำหนด IP ของ Gateway ให้กับ ESP32
 - (IPAddress) subnet – กำหนด IP ของ Subnet ให้กับ ESP32
- หากไม่มีการกำหนดค่าในส่วนของ local_ip จะใช้ค่า default คือ 192.168.4.1
- ฟังก์ชันนี้ไม่มีการคืนค่ากลับ

ESP32 Wi-Fi Arduino Library (AP Mode)

IPAddress WiFi.softAPIP();

- ใช้สำหรับดูหมายเลข IP ของ ESP32 โดยฟังก์ชันนี้ไม่มีพารามิเตอร์ แต่มีการคืนค่าหมายเลข IP กลับมา

ESP32 Wi-Fi Arduino Library (STA Mode)

```
void WiFi.begin(char *ssid, char *passphrase = NULL);
```

- เริ่มต้นเชื่อมต่อ กับ Access Point ที่ต้องการ พารามิเตอร์ คือ
 - (char*) ssid – ชื่อของ Access Point ที่ต้องการให้ไปเชื่อมต่อด้วย
 - (char*) passphrase – รหัสผ่านของ SSID ที่ต้องการเชื่อมต่อ ค่าเริ่มต้นเป็น NULL หรือไม่มีรหัสผ่าน
- ฟังก์ชันนี้ไม่มีการคืนค่ากลับ

ESP32 Wi-Fi Arduino Library (STA Mode)

`wl_status_t WiFi.status();`

- ใช้ในการตรวจสอบสถานะการเชื่อมต่อของ Access Point
- ฟังก์ชันนี้ไม่มีพารามิเตอร์ แต่มีการคืนค่า ดังนี้
 - `WL_IDLE_STATUS` – กำลังเริ่มต้นการเชื่อมต่อ
 - `WL_NO_SSID_AVAIL` – ไม่พบชื่อ WiFi ที่ต้องการเชื่อมต่อในพื้นที่
 - `WL_SCAN_COMPLETED` – ค้นหา WiFi เสร็จสิ้นแล้ว (ใช้กรณีเรียกฟังก์ชันค้นหา WiFi)
 - `WL_CONNECTED` – เชื่อมต่อกับ Access Point สำเร็จ
 - `WL_CONNECT_FAILED` – เชื่อมต่อกับ Access Point ไม่สำเร็จ
 - `WL_CONNECTION_LOST` – หลุดจากการเชื่อมต่อกับ Access Point และจะมีการเชื่อมต่อใหม่อัตโนมัติ
 - `WL_DISCONNECTED` – ไม่ได้เชื่อมต่อกับ Access Point ใด ๆ

ESP32 Wi-Fi Arduino Library (STA Mode)

`bool WiFi.isConnected();`

- ใช้ในการตรวจสอบสถานะการเชื่อมต่อ เช่นเดียวกับ `wl_status_t WiFi.status();` แต่จะคืนค่ากลับเป็นชนิด `bool` เพื่อแจ้งสถานะการเชื่อมต่อ โดยคืนค่าเป็น `true` ถ้าเชื่อมต่อสำเร็จ และ `false` ถ้าเชื่อมต่อไม่สำเร็จ

`wl_status_t WiFi.waitForConnectResult();`

- เป็นฟังก์ชันตรวจสอบสถานะการเชื่อมต่อเช่นกัน แต่ฟังก์ชันนี้ จะรอเชื่อมต่อเพียง 10 วินาที หากเกินจากนี้ จะยกเลิกการเชื่อมต่อ และหลุดออกจากการทำงานของฟังก์ชัน
- ไม่มีพารามิเตอร์ แต่คืนค่าผลการรอการเชื่อมต่อ

ESP32 Wi-Fi Arduino Library (STA Mode)

```
void WiFi.reconnect();
```

- ใช้ในการเชื่อมต่อกับ SSID อีกครั้งในกรณีที่เชื่อมต่อไม่สำเร็จ
- ไม่มีพารามิเตอร์ และไม่มีการคืนค่ากลับ

```
void WiFi.disconnect(bool wifioff = false);
```

- ใช้เพื่อยกเลิกการเชื่อมต่อกับ Access Point ที่ต้องการ โดยมีพารามิเตอร์คือ wifioff เพื่อปิดการเชื่อมต่อ

```
IPAddress WiFi.localIP();
```

- ใช้ในการตรวจสอบหมายเลข IP ของ ESP32 ที่ได้จาก Access Point (ผู้ Access Point จะต้องเลือกโหมดการจ่ายหมายเลข เป็น DHCP)

ESP32 Wi-Fi Arduino Library (finding Wi-Fi)

```
int WiFi.scanNetworks();
```

- ใช้ในการค้นหา Access Point ที่อยู่ในระยะของการเชื่อมต่อ โดยคืนค่าเป็นจำนวน Access Point ที่ค้นหาได้

```
String WiFi.SSID(int networkItem);
```

- ใช้ในการแสดงข้อมูลรายละเอียดของ Access Point ที่ค้นพบ
- พารามิเตอร์ คือ networkItem ซึ่งเป็นลำดับของ Access Point ที่ค้นพบ
- มีการคืนค่าเป็นชื่อของ Access Point ที่ค้นพบ

ESP32 Wi-Fi Arduino Library (finding Wi-Fi)

`long WiFi.RSSI(int networkItem);`

- ใช้ในการตรวจสอบความเข้มของสัญญาณ (Received Signal Strength Indication: RSSI) ของ Access Point ที่ค้นพบ
- พารามิเตอร์ คือ networkitem ซึ่งเป็นลำดับของ Access Point ที่ค้นพบ
- มีการคืนค่าเป็นข้อมูลความเข้มของสัญญาณ โดยมีชนิดของข้อมูลเป็น long

ESP32 Wi-Fi Arduino Library (finding Wi-Fi)

```
wifi_auth_mode_t encryptionType(int networkItem);
```

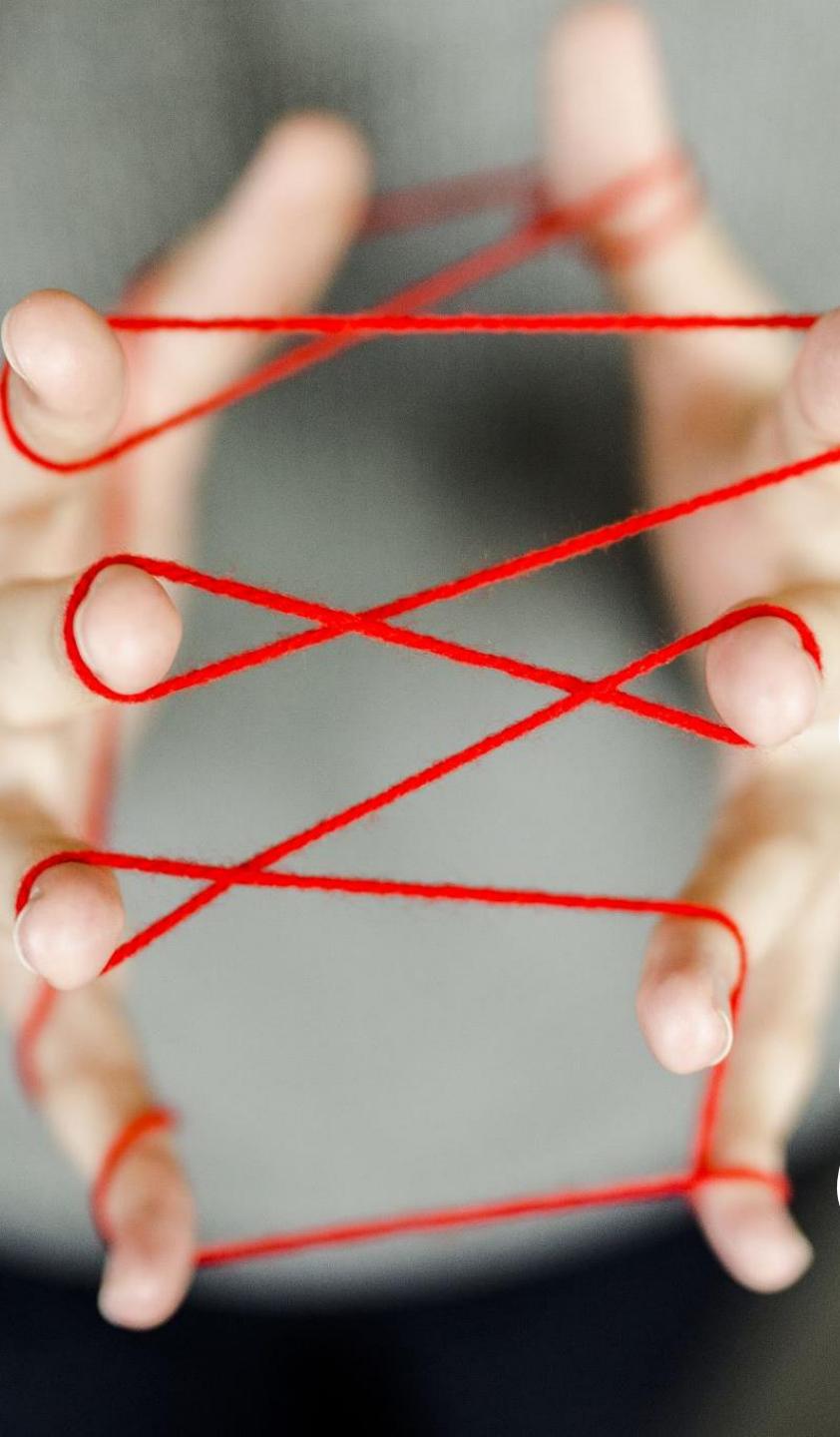
- ใช้ในการตรวจสอบ Access Point ปลายทางว่ามีรหัสผ่านหรือไม่ รวมถึงรูปแบบการเข้ารหัส
- พารามิเตอร์ คือ networkItem ซึ่งเป็นลำดับของ Access Point ที่ค้นพบ
- มีการคืนค่า ดังนี้
 - WIFI_AUTH_OPEN – ไม่มีรหัสผ่าน
 - WIFI_AUTH_WEP – มีรหัสผ่าน และเข้ารหัสแบบ WEP
 - WIFI_AUTH_WPA_PSK – มีรหัสผ่าน และเข้ารหัสแบบ WPA PSK
 - WIFI_AUTH_WPA2_PSK – มีรหัสผ่าน และเข้ารหัสแบบ WPA2 PSK
 - WIFI_AUTH_WPA_WPA2_PSK – มีรหัสผ่าน และเข้ารหัสแบบ WPA/WPA2 PSK
 - WIFI_AUTH_WPA2_ENTERPRISE – มีรหัสผ่าน และเข้ารหัสแบบ WEP

Ref

- The Official ESP32 Book, Dogan Ibrahim. Ahmet Ibrahim, Elektor Publication Inc.
- Kolban's Book on ESP32, Neil Kolban, Leanpub is copyright
- ESP32 Techinal Reference Manual, Espressif System Copyright 2020,

https://www.espressif.com/sites/default/files/documentation/esp32_technical_reference_manual_en.pdf

- https://th.wikipedia.org/wiki/IEEE_802.11
- <http://wise.swu.ac.th/Default.aspx?tabid=3440>
- <https://www.grandmetric.com/2018/07/06/ended-wpa3-wi-fi-security-evolution/>
- <https://www.enterpriseitpro.net/wpa3-wifi-security-standard/>
- <https://support.huawei.com/enterprise/es/doc/EDOC1100102755?idPath=24030814|21782164|21782201|22318535|21560861>



Introduction to Bluetooth

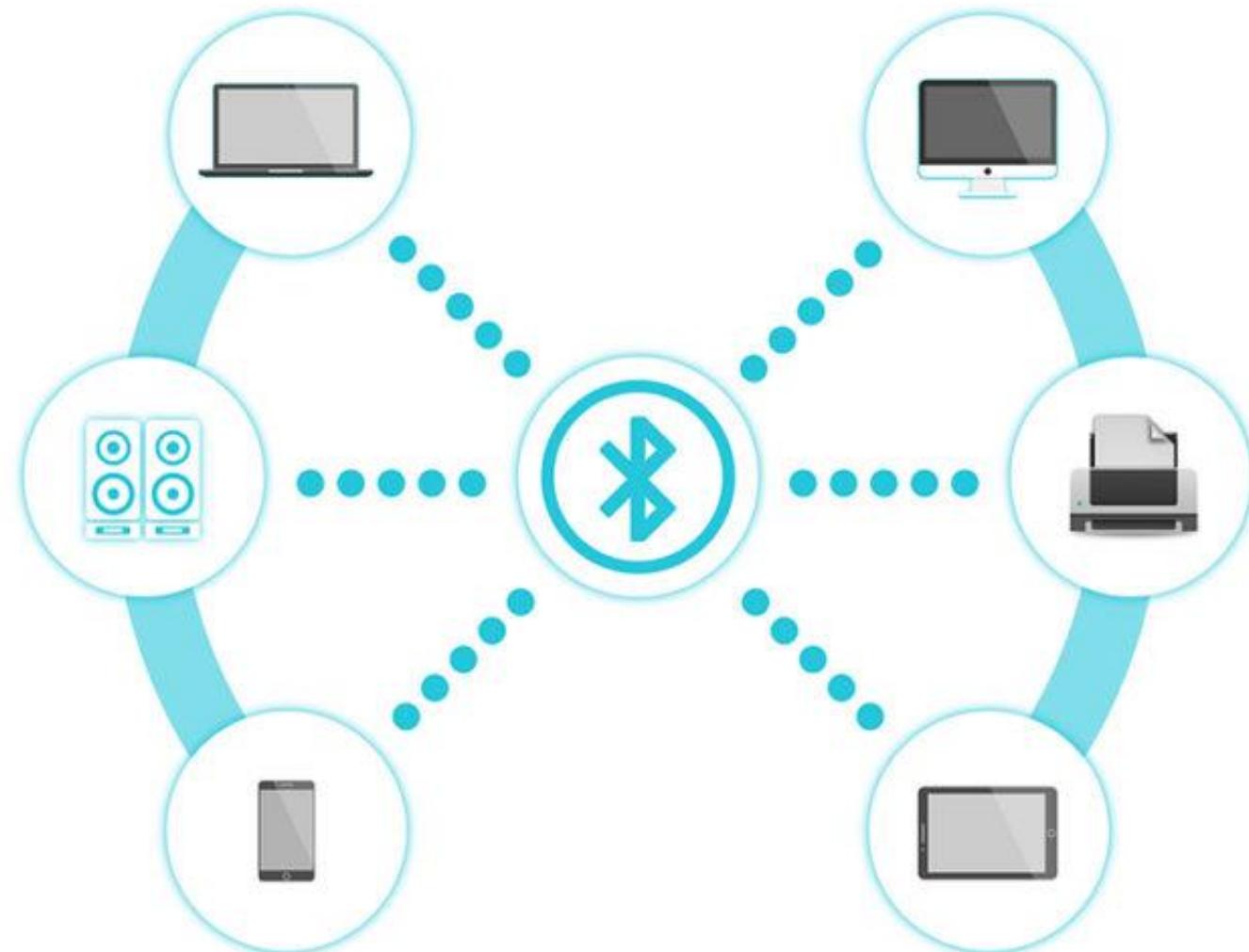


Contents

- Introduction to Bluetooth
- Bluetooth Class
- Bluetooth Version
- Bluetooth usage scenario
- Bluetooth Low Energy
- ESP32 Bluetooth
- ESP32 Bluetooth Classic and BLE

What is Bluetooth?

- Bluetooth คือชื่อของมาตรฐานการสื่อสารแบบเปิด (open standard) สำหรับเทคโนโลยีการสื่อสารไร้สายในระยะใกล้ โดยใช้คลื่นวิทยุ (Radio Frequency: RF)



What is Bluetooth?

- ถูกออกแบบมาภายใต้แนวคิดของการสร้างเครือข่ายไร้สาย (Wireless Personal Area Network: WPAN) ต้นทุนต่ำสำหรับอุปกรณ์พกพา เช่น เครื่องคอมพิวเตอร์พกพา (Laptop), โทรศัพท์มือถือ, อุปกรณ์สวมใส่ (Wearable) เป็นต้น
- ถูกพัฒนาขึ้นครั้งแรกโดยบริษัท Ericsson เมื่อปี พ.ศ.2537 โดยเริ่มจากการศึกษาความเป็นไปได้ของการใช้การสื่อสารด้วยคลื่นวิทยุที่มีต้นทุนต่ำ และใช้พลังงานต่ำ สำหรับเชื่อมต่อระหว่างโทรศัพท์มือถือกับอุปกรณ์ของทางบริษัท

First Bluetooth Phone
Ericsson T39

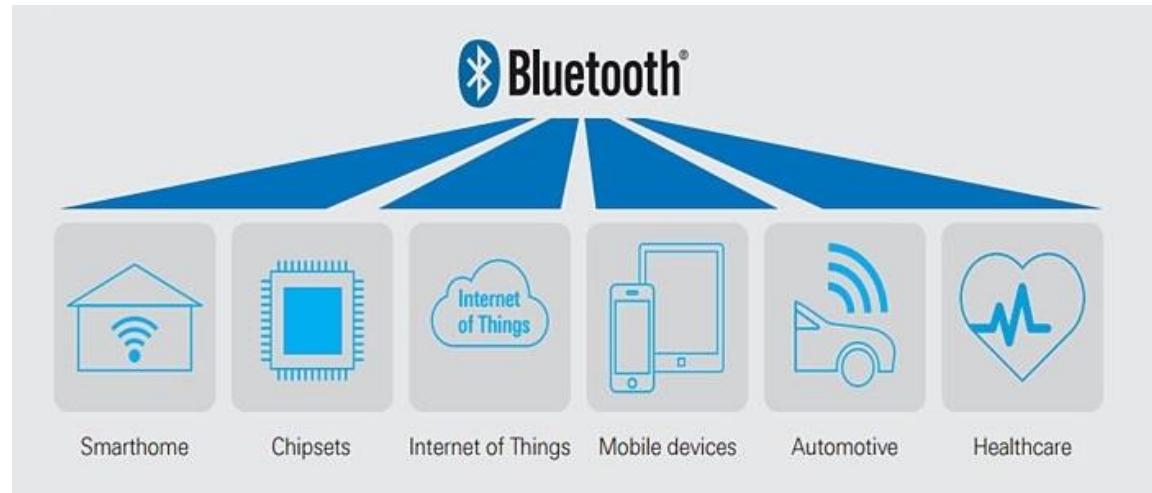


What is Bluetooth?

- แนวคิดเริ่มต้นของ Bluetooth เกิดจากความคิดที่ต้องการพัฒนาชิปสำหรับสื่อสารด้วยวิทยุที่มีราคาต่ำ (ประมาณการไว้ที่ \$5) ที่สามารถใช้งานร่วมกับคอมพิวเตอร์พกพา, โทรศัพท์มือถือ, เครื่องพิมพ์ (Printer) และอื่นๆ เพื่อใช้ในการแลกเปลี่ยนข้อมูลสื่อสารระหว่างอุปกรณ์ตัวยกัน
- นอกจากนี้ การออกแบบชิปที่ว่านี้ ยังต้องใช้พลังงานต่ำ เพื่อใช้งานร่วมกับอุปกรณ์ที่ต้องใช้ไฟเลี้ยงจากแบตเตอรี่ได้ด้วย
- จากแนวคิดนี้ จึงได้มีการจัดตั้งกลุ่ม Bluetooth Special Interest Group (SIG) ขึ้น เพื่อกำหนดมาตรฐานสำหรับ Bluetooth โดยตรง



What is Bluetooth?



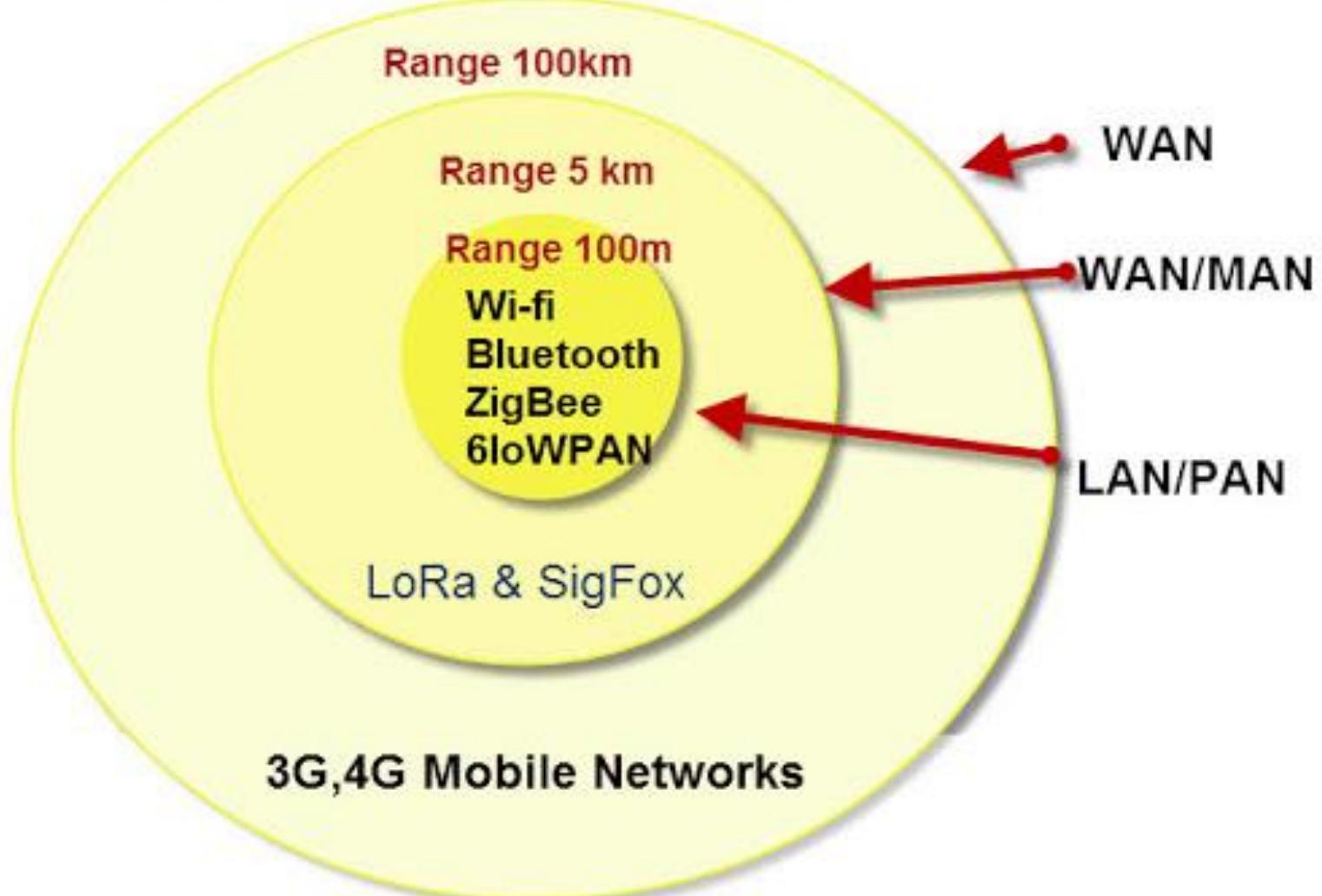
- ในความเป็นจริงแล้ว Bluetooth ไม่ได้มีแค่เป็นเทคโนโลยีใหม่แต่อย่างใด (ณ เวลาที่ใช้งานครั้งแรก) แต่ใช้คลื่นวิทยุและเทคโนโลยีเครือข่ายไร้สายที่มีอยู่แล้ว ซึ่งมีข้อดี คือ สามารถใช้ร่วมกับอุปกรณ์ที่รองรับการสื่อสารด้วยคลื่นวิทยุแบบเดิมได้เลย
- อย่างไรก็ตาม สิ่งที่ทำให้ Bluetooth มีเอกลักษณ์ที่แตกต่างจากการสื่อสารแบบไร้สายอื่นๆ ก็คือ การประยุกต์ใช่องค์ประกอบเฉพาะของตน และเทคโนโลยีที่มีอยู่แล้ว ในการนิยามการทำงานหลัก (core operation) และประเภทการใช้งาน (application profiles)



Wireless PAN

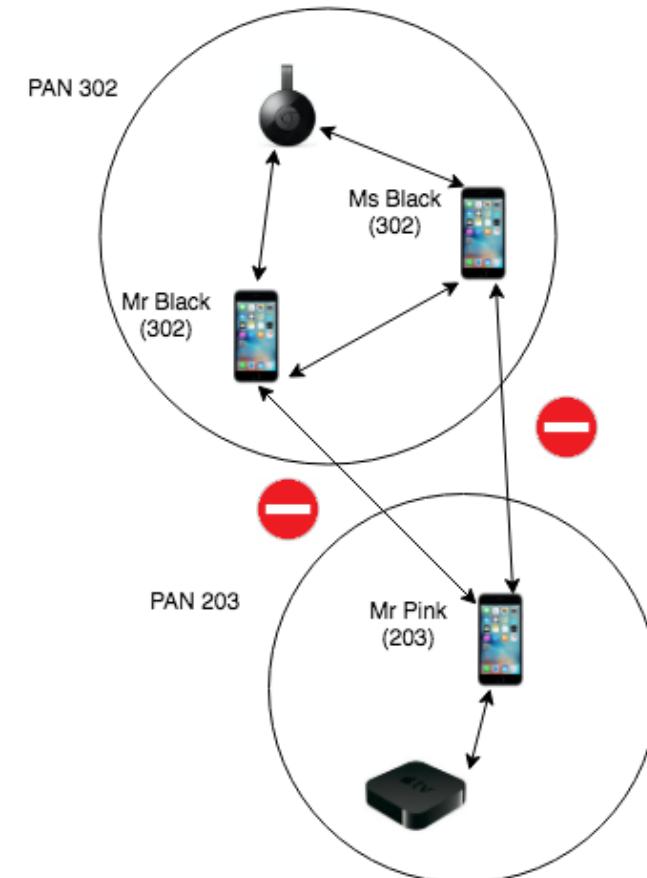
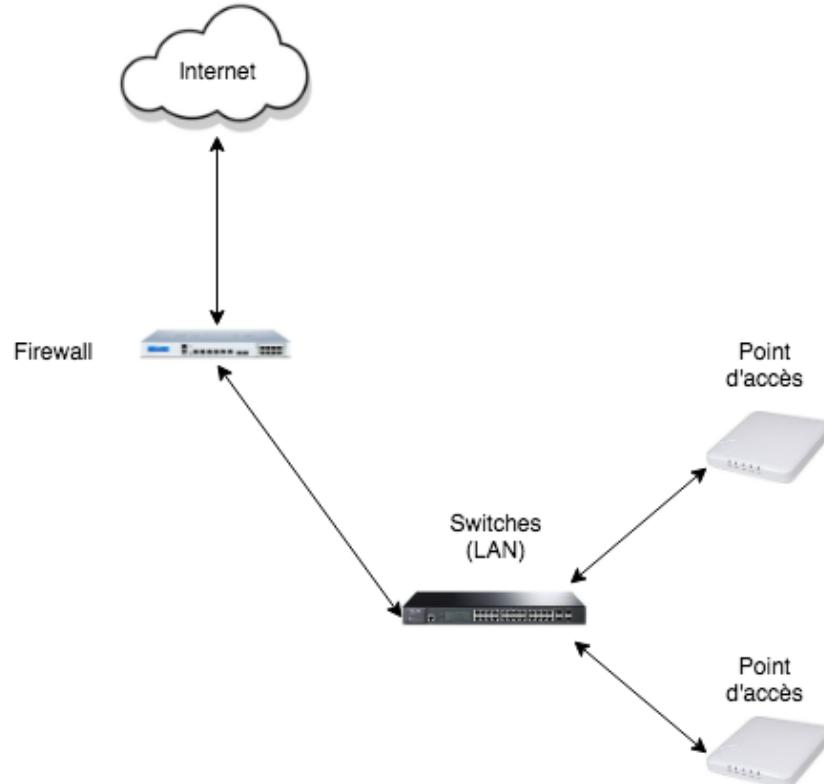
Wireless PAN

Wireless Networking Overview



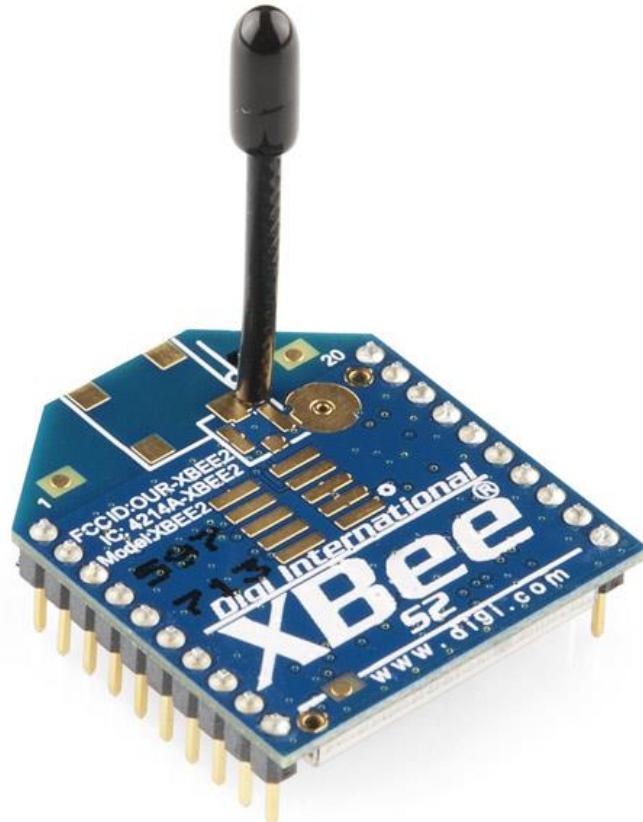
Wireless PAN

PAN = Personal Area Network



Wireless PAN

Zigbee 802.15.4



ZigBee Alliance เป็นมาตรฐานที่มีอัตราการรับส่งข้อมูลสูงสุดไม่เกิน 250 kbit/s ใช้พลังงานต่ำ

ความถี่ใช้งานตามมาตรฐานไว้ 3 ปีน

2.4 GHz (16 ช่อง)

915 MHz (10 ช่อง)

868 MHz (1 ช่อง)

ข้อเสีย Zigbee คือราคาแพง

Wireless PAN

Zigbee 802.15.4



รหัสสินค้า EWLM114

XBee ZigBee TH (Wire Antenna)
Series 2 (S2C)

1,250.00 บาท

⚠ สินค้าหมด



รหัสสินค้า EWLM115

XBee ZigBee TH (RPSMA)
Antenna) Series 2 (S2C)

1,250.00 บาท



รหัสสินค้า EWLM116

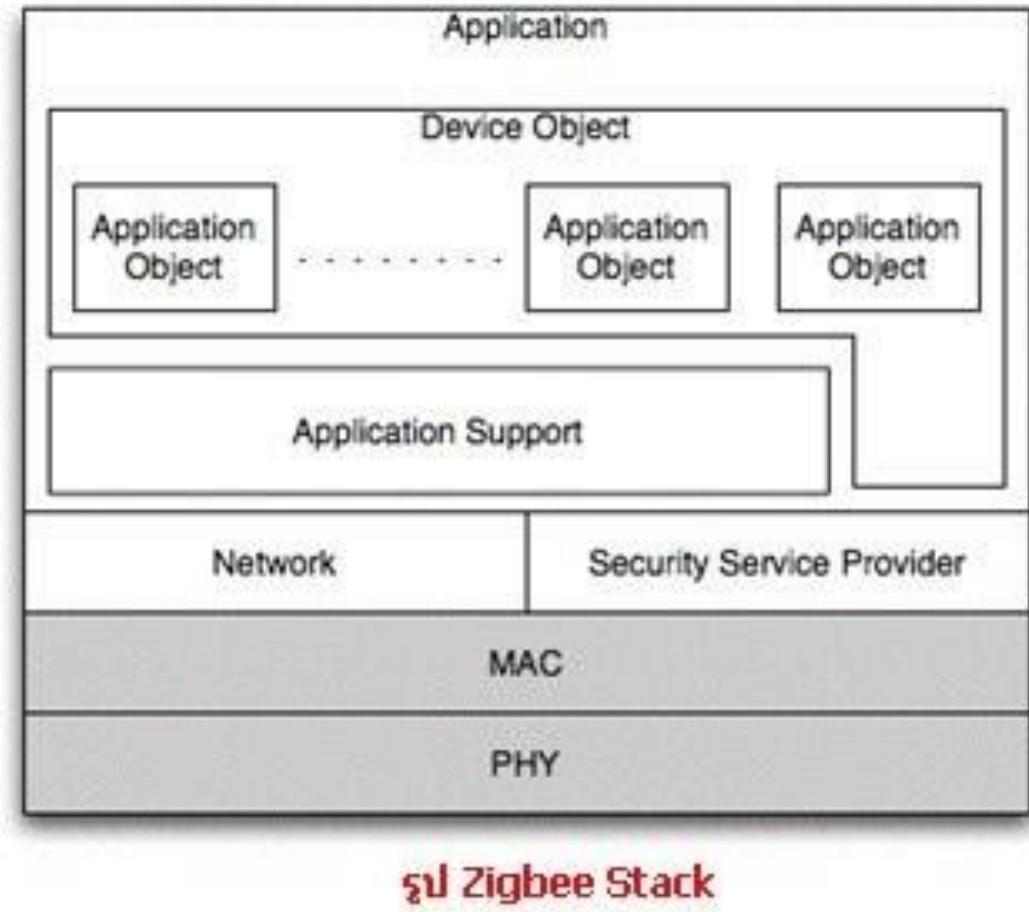
XBee-PRO ZigBee TH (RPSMA)
Antenna) Series 2 (S2C)

1,980.00 บาท

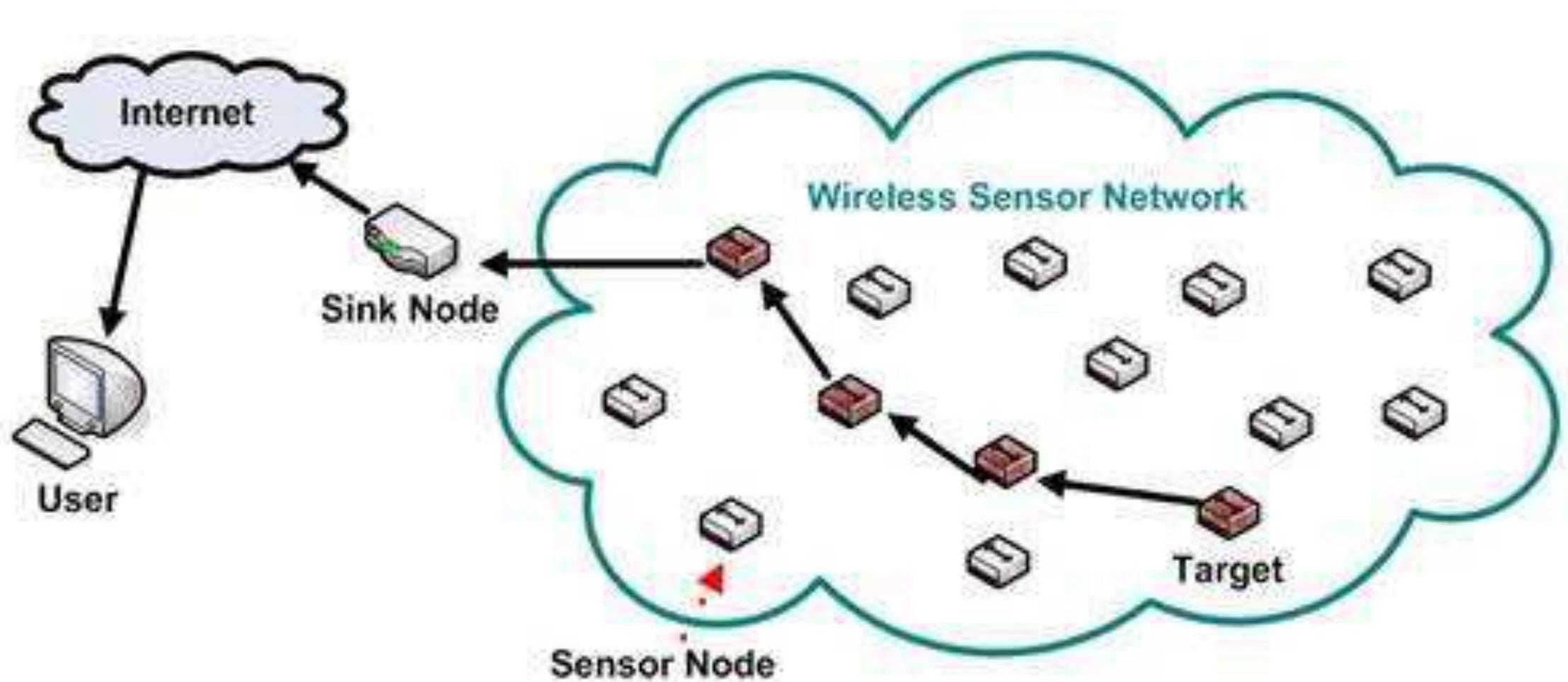
ข้อเสีย Zigbee คือราคาง่

Wireless PAN

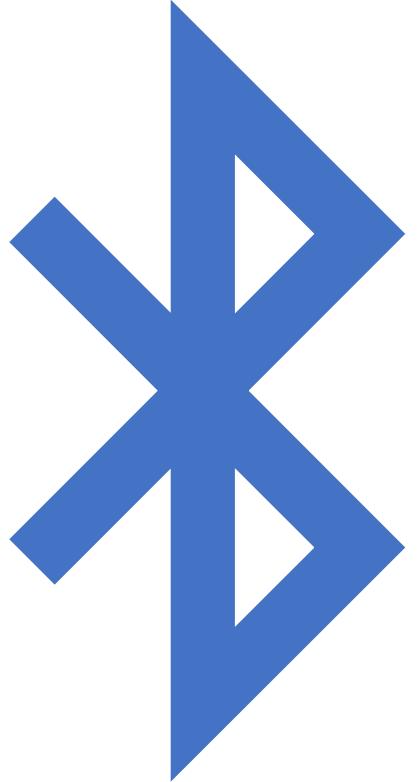
ZigBee นำ Physical Layer และ MAC Layer ของ IEEE 802.15.4 ซึ่งเป็นมาตรฐานการกำหนดการสื่อสารไร้สายแบบ WPAN (Wireless Personal Area Network) มาทำงานใน 2 Layer ล่างสุดในเรื่องของ ระดับกำลังสัญญาณ Link Quality Access Control Security และใน Layer ถัดขึ้นไปจะเป็นรูปแบบของ ZigBee



Wireless PAN



Wireless Networking Technologies						
Standard	ZigBee IEEE 802.15.4	Bluetooth IEEE 802.15.1	Z-Wave No International Standard	Wi-Fi IEEE 802.11a, b, g (n to be ratified)	LonWorks EIA 709.1, 2, 3	Proprietary Proprietary
Industry organizations	ZigBee Alliance	Bluetooth SIG	Z-Wave Alliance	Wi-Fi Alliance	LonMark Interoperability Association	N/A
Topology	Mesh, star, tree	Star	Mesh	Star	Medium-dependent	P2P, star, mesh
RF frequency	868/915 MHz, 2.4 GHz	2.4 GHz	Sub 1G	2.4 GHz, 5.8 GHz	N/A (wired technology)	433/868/900 MHz, 2/4 GHz
Data rate	250 kbits/s	723 kbits/s		11 to 105 Mbits/s	15 kbits/s to 10 Mbits/s	10 to 250 kbits/s
Range	10 to 300 m	10 m	50 m	10 to 100 m	Medium-dependent	10 to 70 m
Power	Very low	Low	Low	High	Wired	Very low to low
Battery operation (life)	Alkaline (months to years)	Rechargeable (days to weeks)		Rechargeable (hours)	N/A	Alkaline (months to years)
Nodes	65,000	8	230	32	32,000	100 to 1000



History of Bluetooth

History of Bluetooth

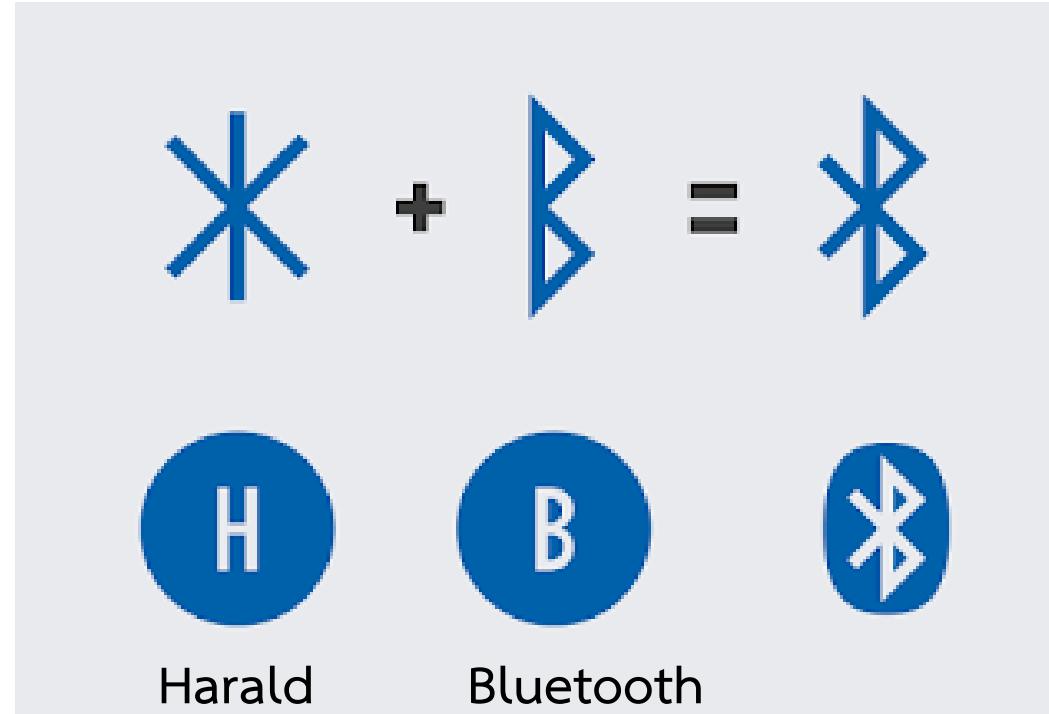
- คำว่า Bluetooth เป็นความจริงแล้วเป็นชื่อของกษัตริย์ประเทศเดนมาร์ก ที่มีชื่อว่า Harald Bluetooth ซึ่งในช่วงปี ค.ศ. 940-981 หรือประมาณ 1,000 กว่าปีก่อนหน้า



ชื่อ Bluetooth จึงรำลึกถึงกษัตริย์ Bluetooth ผู้ครองประเทศกลุ่มสแกนดิเนเวีย กลุ่มผู้นำในด้านการผลิตโทรศัพท์มือถือ

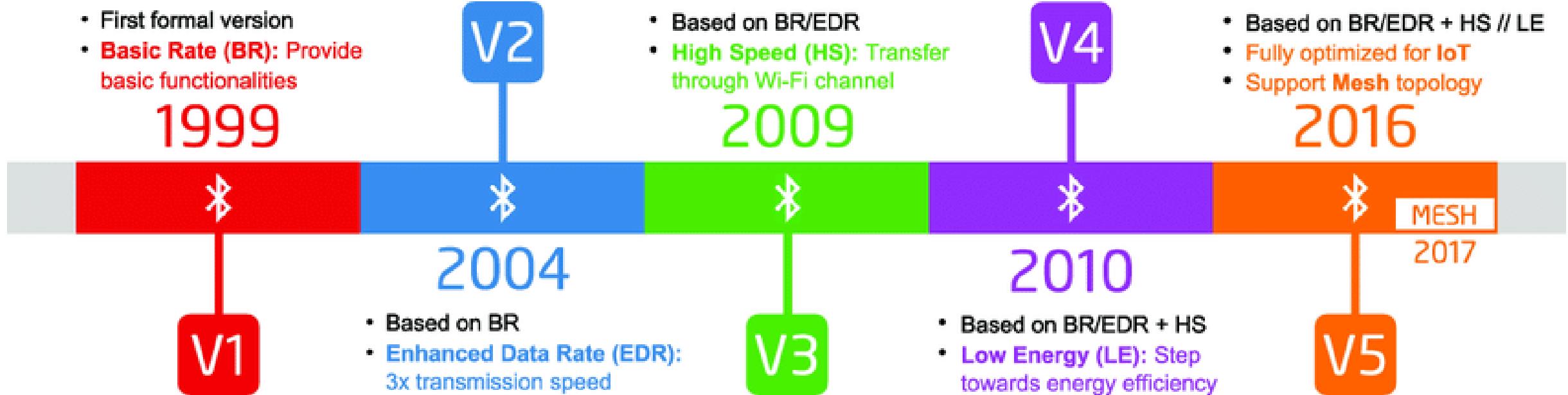
History of Bluetooth

- คำว่า Bluetooth เป็นความจริงแล้วเป็นชื่อของกษัตริย์ประเทศเดนมาร์ก ที่มีชื่อว่า Harald Bluetooth ซึ่งในช่วงปี ค.ศ. 940-981 หรือประมาณ 1,000 กว่าปีก่อนหน้า



ชื่อ Bluetooth จึงรำลึกถึงกษัตริย์ Bluetooth ผู้ครองประเทศกลุ่มสแกนดิเนเวีย กลุ่มผู้นำในด้านการผลิตโทรศัพท์มือถือ

History of Bluetooth



Bluetooth Class

Bluetooth Class

Bluetooth Class หมายถึงระดับความแรงของสัญญาณที่อุปกรณ์ Bluetooth สามารถส่งออกได้ ซึ่งมีผลโดยตรงต่อระยะทางในการเชื่อมต่อ สำหรับสัญญาณบลูทูธ หลักๆจะมีด้วยกันทั้งหมด 4 Class (บางที่อาจจะมีแค่ 3 Class เนื่องจาก Class 4 สัญญาณค่อนข้างอ่อนมากและไม่นิยมใช้) ประกอบไปด้วย

Bluetooth คลาส 1 (Class 1)

จะสามารถรับส่งข้อมูลได้ในระยะรัศมี 100 เมตร และจะใช้พลังงาน ประมาณ 100 mW

Bluetooth คลาส 2 (Class 2)

จะสามารถรับส่งข้อมูลได้ในระยะรัศมี 10 เมตร และจะใช้พลังงาน ประมาณ 2.5 mW

Bluetooth คลาส 3 (Class 3)

จะสามารถรับส่งข้อมูลได้ในระยะรัศมี 10 เซนติเมตรถึง 1 เมตร และจะใช้พลังงาน ประมาณ 1mW

Bluetooth คลาส 4 (Class 4 ไม่ค่อนนิยมเนื่องจากสัญญาณอ่อน)

จะสามารถรับส่งข้อมูลได้ในระยะรัศมี 10 เซนติเมตรถึง 0.5 เมตร และจะใช้พลังงาน ประมาณ 0.5 mW

Bluetooth Class

Type	Power	Max Power Level	Designed Operating Range	Sample Devices
Class 1	High	100 mW (20 dBm)	Up to 100 m (328 feet)	USB adapters, access points
Class 1.5 (low energy) ⁷	Med-High	10 mW (10 dBm)	Up to 30 m (100 feet), but typically 5 m (16 feet)	Beacons, wearable sensors
Class 2	Medium	2.5 mW (4 dBm)	Up to 10 m (33 feet)	Mobile devices, Bluetooth adapters, smart card readers
Class 3	Low	1 mW (0 dBm)	Up to 1 m (3 feet)	Bluetooth adapters

Bluetooth Version

Bluetooth Version

Bluetooth Version หมายถึงรุ่นของเทคโนโลยี Bluetooth ที่ถูกพัฒนาและอัปเดตมาในแต่ละยุคสมัย แต่ละเวอร์ชัน มีการปรับปรุงความเร็วในการส่งข้อมูล, ระยะการเชื่อมต่อ, ความเสถียร, และประยุกต์พลังงานมากขึ้น โดยหลักๆ สำหรับเวอร์ชันของบลูทูธนั้น มีตั้งแต่เวอร์ชัน 1.0 ไปจนถึง 5.0

- Bluetooth 2.0 = 1 Mbps
- Bluetooth 2.0+EDR = 3 Mbps
- Bluetooth 2.1+EDR = 3 Mbps
- Bluetooth 3.0+HS = 24 Mbps
- Bluetooth 4.0 = 24 Mbps
- Bluetooth 4.1 = 24 Mbps
- Bluetooth 5.0 = 48 Mbps

Bluetooth 1-3 Classic
Bluetooth 4-5 BLE

Year Introduced	Bluetooth Version	Feature
2004	2.0	Enhanced Data Rate
2007	2.1	Secure Simple Pairing
2009	3.0	High Speed with 802.11 Wi-Fi Radio
2010	4.0	Low-energy protocol
2013	4.1	Indirect IoT device connection
2014	4.2	IPv6 protocol for direct internet connection
2016	5.0	4x range, 2x speed, 8x message capacity + IoT

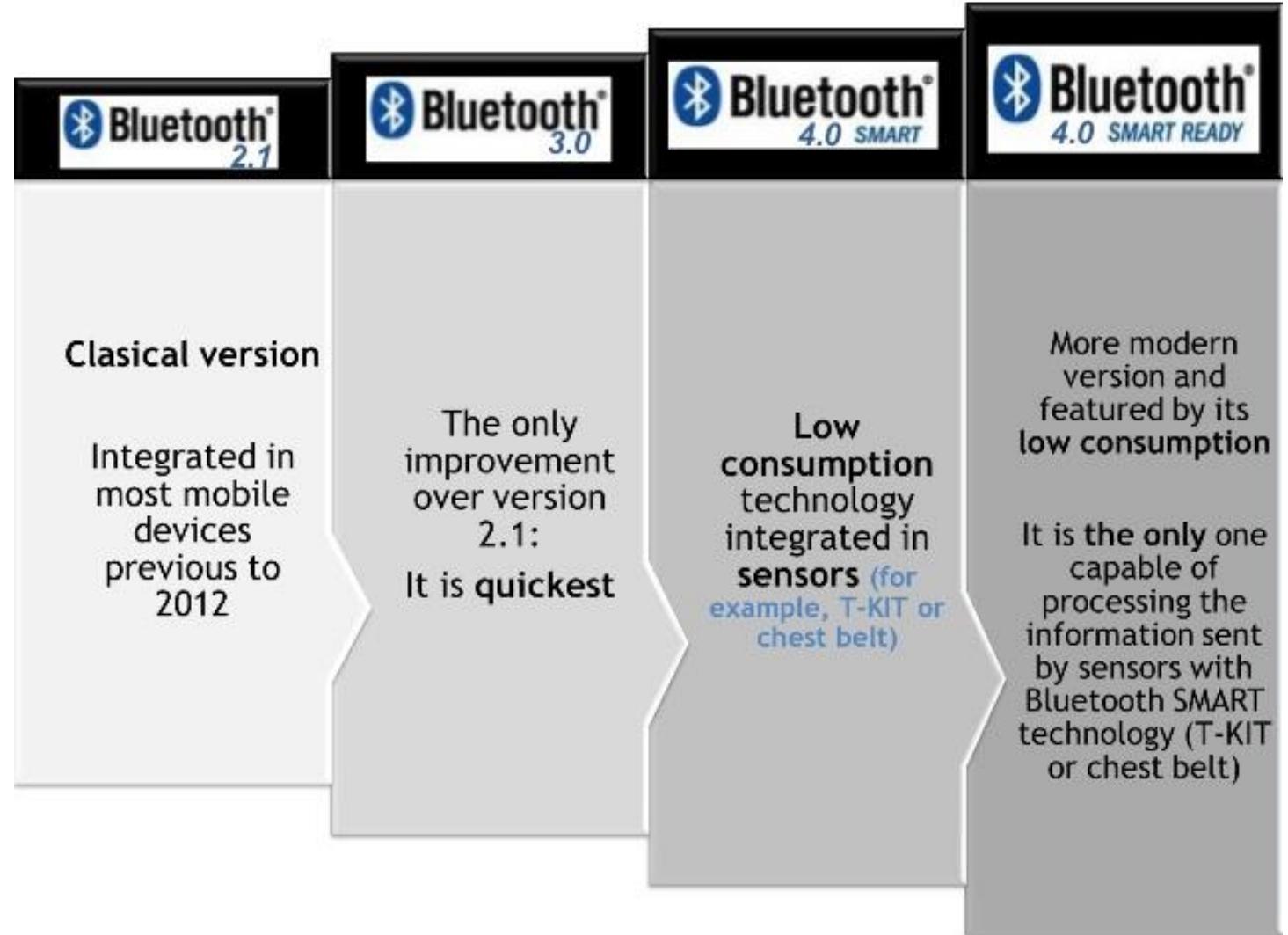
Bluetooth Version

Bluetooth Version Compatibility

Backward Compatibility Bluetooth

Ex. ໂທຣສັບທໍມີອົດືອງຮັບບລຸຖູ 4.0

ໃນຂະໜາດທີ່ຫຼັກພັດບລຸຖູຮອງຮັບແດ່ 2.0



Bluetooth usage scenario

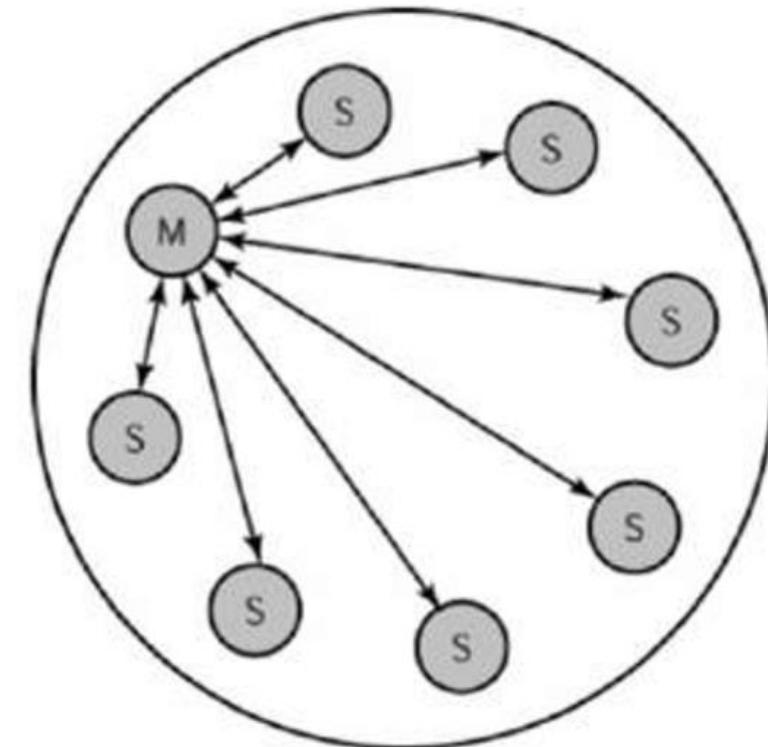
Bluetooth usage scenario

- รูปแบบการใช้งานของ Bluetooth สามารถแบ่งออกได้ตามรูปแบบการสื่อสาร ดังนี้
- Point-to-Point
 - เป็นการเชื่อมต่อแบบจุดต่อจุด โดยเป็นการสื่อสารระหว่างอุปกรณ์ 2 ตัว
 - การทำงานเป็นแบบ Master-Slave โดยที่อุปกรณ์สามารถเลือกได้ว่าจะทำงานแบบใด



Bluetooth usage scenario

- Piconet
 - เป็นโครงข่ายที่สร้างขึ้นโดย Master 1 ตัว และ Slave มากกว่า 1 ตัว (สูงสุด 7 ตัว)
 - ในเครือข่ายจะมีการแบ่งช่องสัญญาณสำหรับอุปกรณ์แต่ละตัวที่ทำการเชื่อมต่อ
 - รองรับความเร็วในการสื่อสารแต่ละตัว สูงสุดที่ 1 Mbps



Bluetooth usage scenario

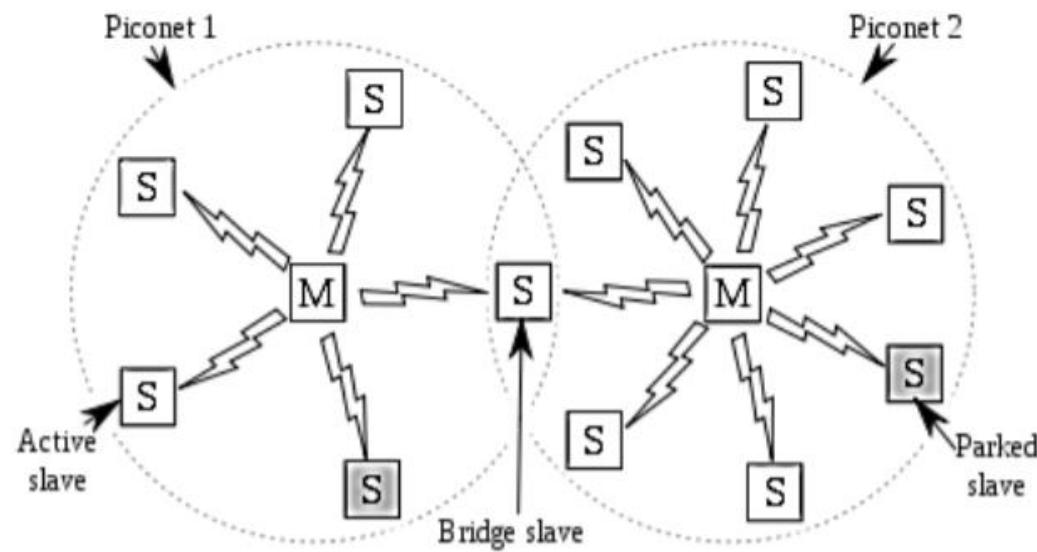
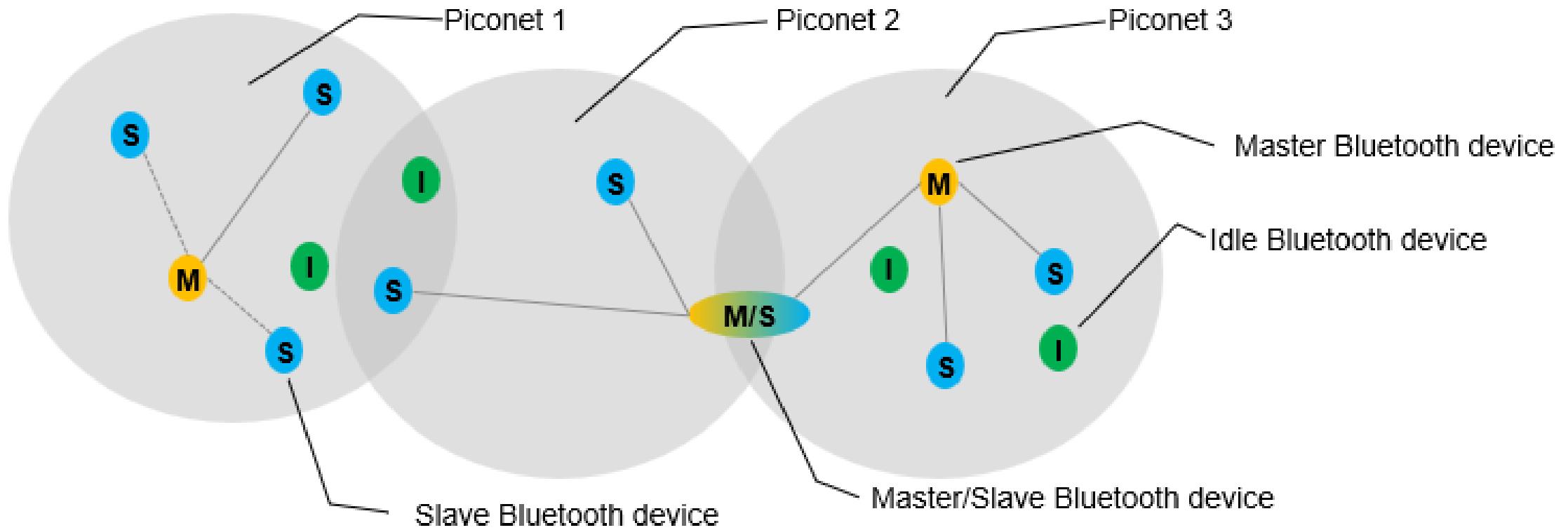


Fig. SCATTERNET

- Scatternet

- เกิดจากการที่เครือข่าย Piconet แต่ละวง เชื่อมต่อกัน และมีการสื่อสารถึงกัน
- รองรับจำนวน Piconet ได้สูงสุด 10 เครือข่าย

Bluetooth usage scenario



Summary of Wireless Network Configurations

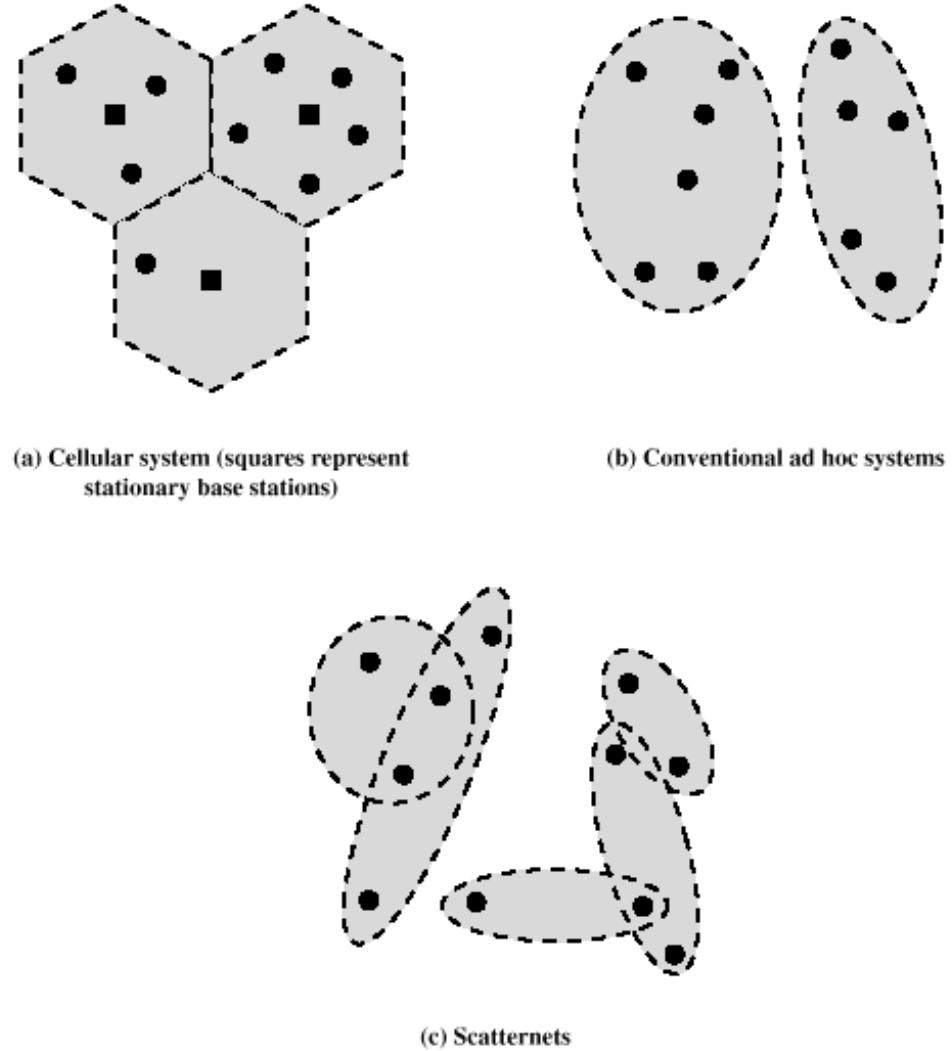
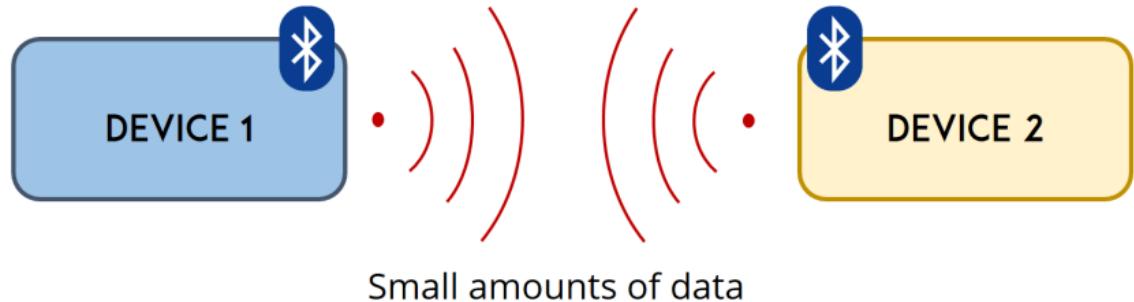


Figure 15.5 Wireless Network Configurations

Bluetooth Low Energy

- เป็น Bluetooth ที่ใช้เทคโนโลยีที่แตกต่างจากรุ่นก่อนๆ (V2.0, V3.0)
- มีระยะการส่งที่สั้นกว่า ใช้กำลังส่งต่ำกว่า (ในทางทฤษฎี โดยอ้างอิงจากระยะการส่งสูงสุด)
- มีคุณสมบัติทะลุทะลวงที่ด้อยกว่า เมื่อเทียบกับเวอร์ชันเดิม
- ประหยัดพลังงานกว่า
- สามารถค้นหาและเชื่อมต่อได้เร็วกว่า
- ราคาต่ำกว่า



Bluetooth Low Energy

- Bluetooth Low Energy modules, solutions and application presentation by Future Electronics
- <https://www.youtube.com/watch?v=AIHpSCYOQNI>

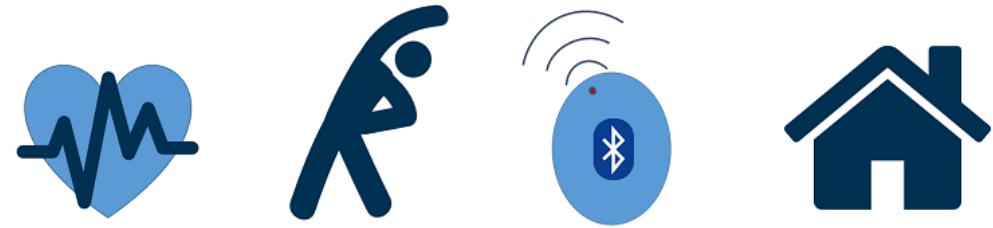
Bluetooth Classic vs BLE

As of DEC, 2016	 Bluetooth™		
Specifications	Classic Bluetooth	Bluetooth Low Energy (V 4.2)	Bluetooth 5
Range	100 m	Greater than 100 m	Greater than 400 m
Data Rate	1-3 Mbps	1 Mbps	2 Mbps
Application Throughput	0.7 -2.1 Mbps	0.27 Mbps	—
Frequency	2.4 GHz	2.4 GHz	—
Security	56/128-bit	128-bit AES with Counter Mode CBC-MAC	—
Robustness	Adaptive fast frequency hopping, FEC, fast ASK	24-bit CRC, 32-bit Message Integrity Check	—
Latency	100 ms	6 ms	—
Time Lag	100 ms	3 ms	—
Voice Capable	Yes	No	—
Network Topology	Star	Star	—
Power Consumption	1 W	0.01 to 0.5 W	—
Peak Current Consumption	less than 30 mA	less than 15 mA	—

Bluetooth Classic vs BLE

นอกจากนี้ BLE ไม่เพียงรองรับการสื่อสารแบบ point-to-point เท่านั้น แต่ยังรวมถึง broadcast mode และ mesh network

	Bluetooth Low Energy (LE)	Bluetooth Basic Rate/ Enhanced Data Rate (BR/EDR)
Optimized For...	Short burst data transmission	Continuous data streaming
Frequency Band	2.4GHz ISM Band (2.402 – 2.480 GHz Utilized)	2.4GHz ISM Band (2.402 – 2.480 GHz Utilized)
Channels	40 channels with 2 MHz spacing (3 advertising channels/37 data channels)	79 channels with 1 MHz spacing
Channel Usage	Frequency-Hopping Spread Spectrum (FHSS)	Frequency-Hopping Spread Spectrum (FHSS)
Modulation	GFSK	GFSK, π/4 DQPSK, 8DPSK
Power Consumption	~0.01x to 0.5x of reference (depending on use case)	1 (reference value)
Data Rate	LE 2M PHY: 2 Mb/s LE 1M PHY: 1 Mb/s LE Coded PHY (S=2): 500 Kb/s LE Coded PHY (S=8): 125 Kb/s	EDR PHY (8DPSK): 3 Mb/s EDR PHY (π/4 DQPSK): 2 Mb/s BR PHY (GFSK): 1 Mb/s
Max Tx Power*	Class 1: 100 mW (+20 dBm) Class 1.5: 10 mW (+10 dBm) Class 2: 2.5 mW (+4 dBm) Class 3: 1 mW (0 dBm)	Class 1: 100 mW (+20 dBm) Class 2: 2.5 mW (+4 dBm) Class 3: 1 mW (0 dBm)
Network Topologies	Point-to-Point (including piconet) Broadcast Mesh	Point-to-Point (including piconet)



เนื่องจากคุณสมบัติของมัน BLE จึงเหมาะสมสำหรับแอปพลิเคชันที่ต้องการแลกเปลี่ยนข้อมูลจำนวนเล็กน้อย

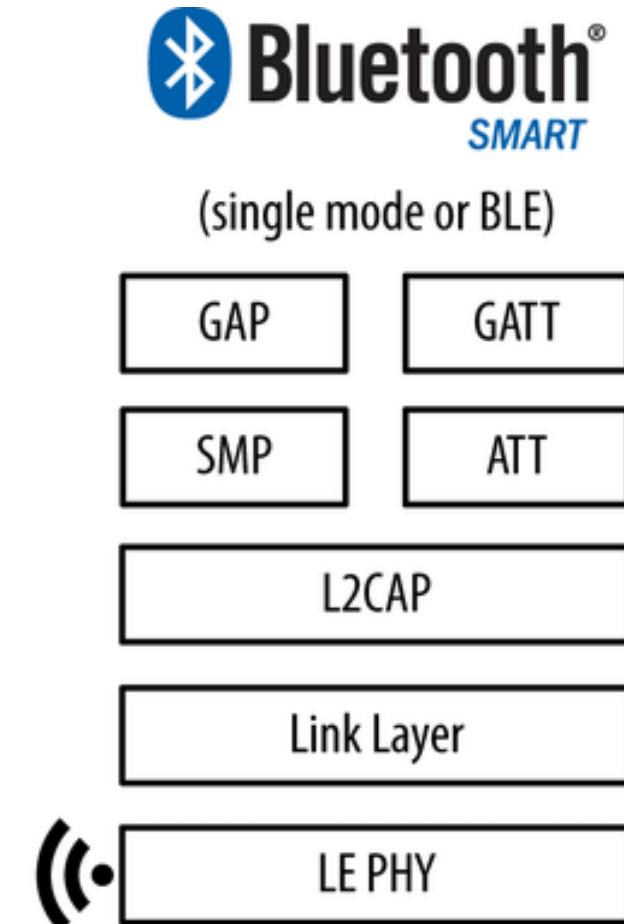
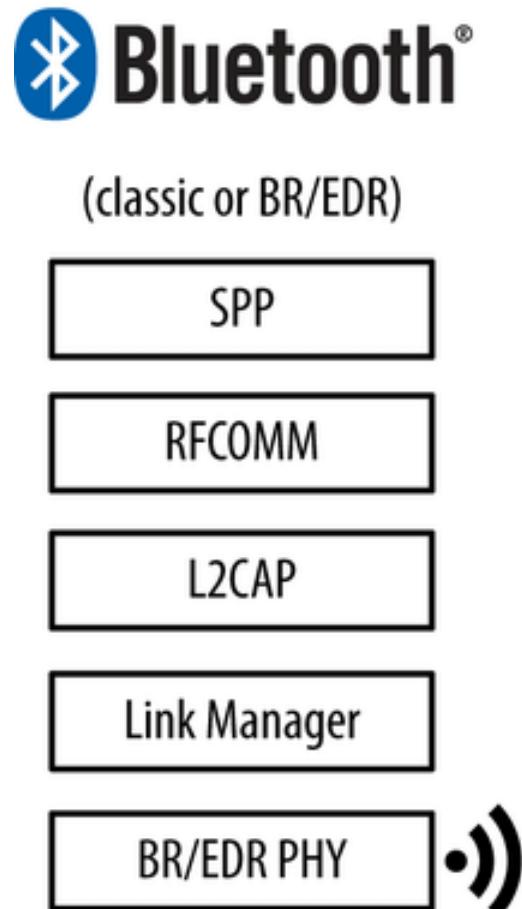
ดังที่ได้กล่าวไว้ก่อนหน้านี้ BLE ยังรองรับ broadcast mode และ mesh network:

Broadcast mode : เซิร์ฟเวอร์ส่งข้อมูลไปยังไคลเอนต์จำนวนมากที่เชื่อมต่อ

Mesh network : อุปกรณ์ทั้งหมดเชื่อมต่อในคือการเชื่อมต่อหลายต่อหลายอย่าง

Bluetooth BLE Protocol

Bluetooth BLE มีความแตกต่างจาก Bluetooth Classic ในส่วนของ Protocol



Bluetooth BLE Protocol

Bluetooth BLE มีความแตกต่างจาก Bluetooth Classic ในส่วนของ Protocol



(classic or BR/EDR)



(dual mode or BR/EDR/LE)



(single mode or BLE)



Data Specific Profile

Data processing

GAP configuration

Central
Unit
(Master)



Peripheral
Unit
(Slave)

GATT configuration

Client



Server

*Data
exported
as a
Service*

Service

Characteristic

Characteristic

Descriptor

Application: BT SIG or Proprietary Profile
(vendor specific)

Generic Attribute
Profile (GATT)

Generic Access
Profile (GAP)

Attribute Protocol

Security Manager

Logical Link Control and Adaptation Protocol

Host-Controller Interface

Link Layer

Direct Test Mode

Physical Layer

Bluetooth Low Energy Stack

BLE GAP

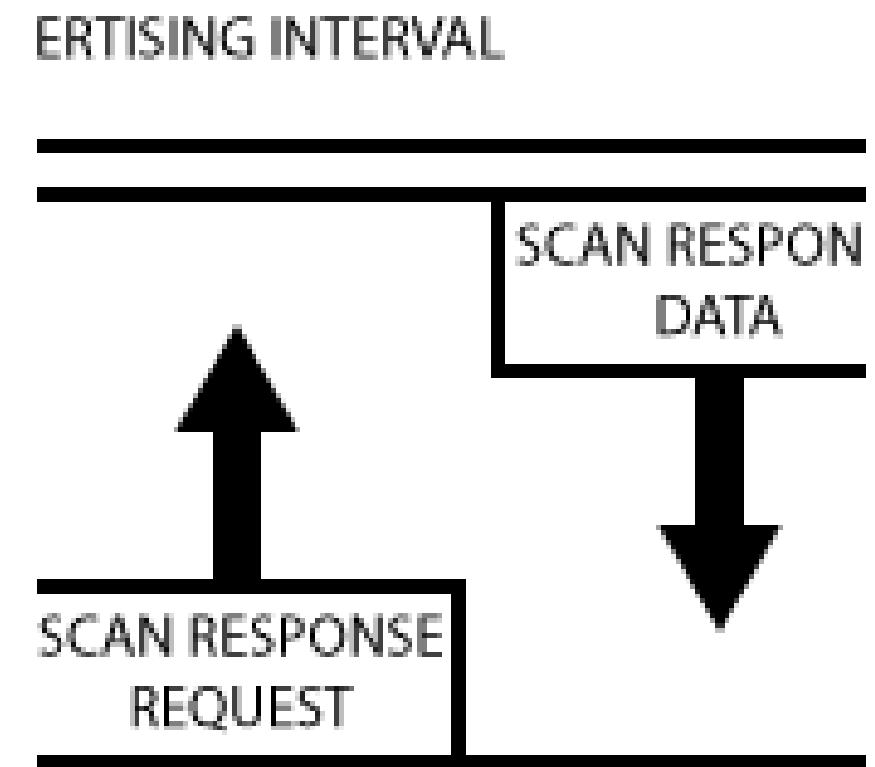
- ย่อมาจากคำว่า Generic Access Profile โดยเป็นการกำหนดรายละเอียดของการเชื่อมต่อ และการกระจายข้อมูลไปยังอุปกรณ์อื่นๆ
- ถือเป็นรูปแบบการสื่อสารที่ทำให้อุปกรณ์อื่นๆ สามารถมองเห็นได้ และยังกำหนดการสื่อสารสำหรับอุปกรณ์แต่ละตัวได้อิสระ
- รูปแบบการสื่อสารของ GAP สามารถแบ่งออกเป็น 2 รูปแบบหลักๆ ตามบทบาทของอุปกรณ์นั้นๆ คือ
 1. Peripheral Device
 - ใช้กับอุปกรณ์ที่มีขนาดเล็ก, พลังงานต่ำ, หรือมีทรัพยากรจำกัด เพื่อให้สามารถเชื่อมตอกับอุปกรณ์ที่มีประสิทธิภาพสูงกว่าได้ ซึ่งจะถูกใช้งานเป็น Central Device
 2. Central Device
 - เป็นตัวกลางที่ใช้ในการกระจายข้อมูลไปยังอุปกรณ์อื่นๆ โดยมักจะเป็นอุปกรณ์ที่มีประสิทธิภาพสูงกว่า Peripheral Device เช่น Smartphone, Tablet, BLE Gateway

Advertising and Response

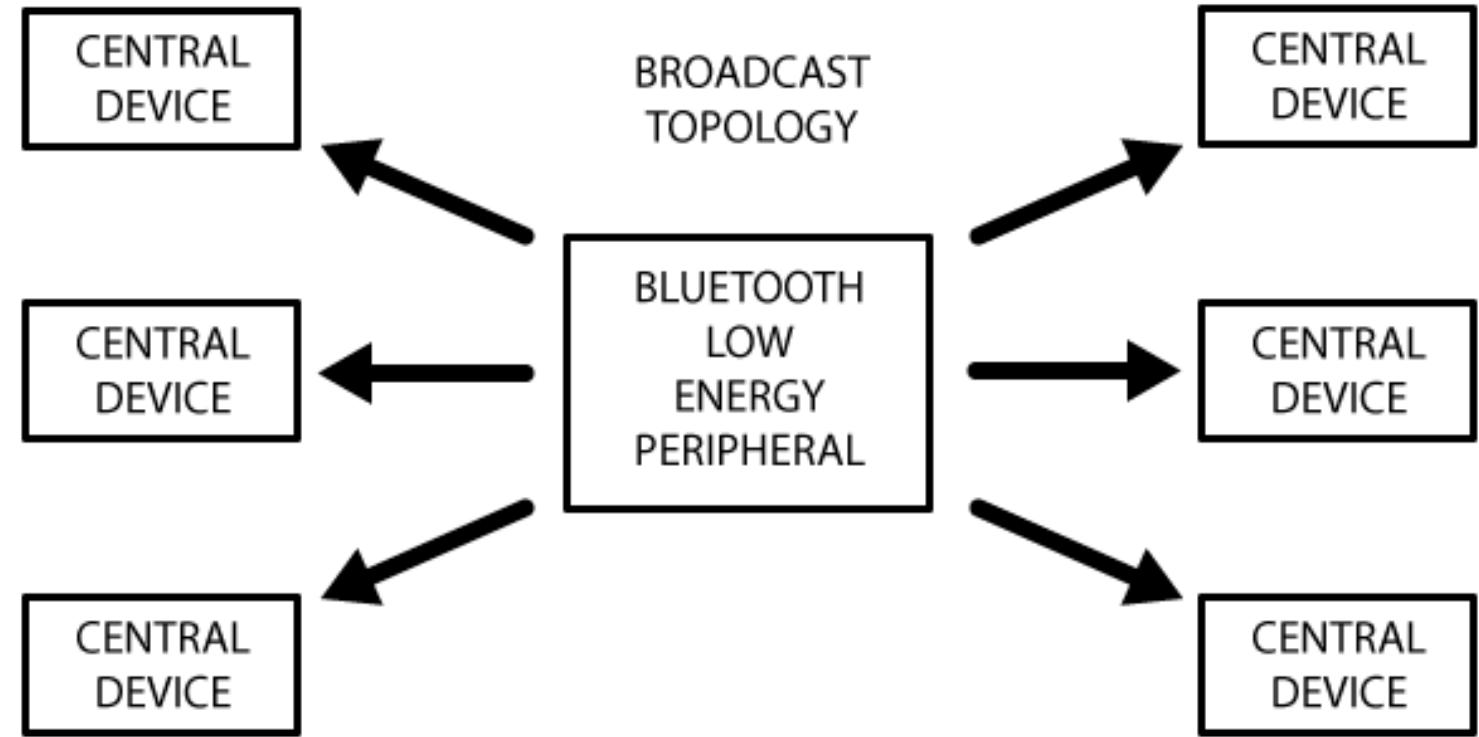
- การกระจายข้อมูลโดยใช้ GAP สามารถทำได้ 2 แบบ คือ Advertising Data และ Scan Response Payload
- ขนาดข้อมูลทั้ง 2 แบบ จะมีขนาดของข้อมูลสูงสุดที่ 31 Bytes โดยมีลักษณะเหมือนกัน แต่ Advertising Data จะต้องมีการใช้งาน เนื่องจากข้อมูลจะถูกส่งออกอยู่ตลอดเวลา เพื่อให้อุปกรณ์ที่ทำหน้าที่เป็นส่วนกลาง (Central Device) รับรู้ว่ามีการเชื่อมต่ออยู่
- กรณีของ Scan Response Payload จะเป็นกรณีที่อุปกรณ์ตัวกลาง มีการร้องขอข้อมูลจากอุปกรณ์ในลูกข่าย (Peripheral Device) และยังช่วยให้ผู้พัฒนาอุปกรณ์สามารถเพิ่มข้อมูลอื่นๆ เช่น ชื่ออุปกรณ์ลงใน Advertising payload ได้อีกด้วย

GAP Advertising Process

- อุปกรณ์ที่เป็น Peripheral Device จะส่ง Advertising Data ในช่วงระยะเวลาที่กำหนดไว้ โดยเมื่อหมดระยะเวลาที่wanี้แล้ว ก็จะทำการส่งข้อมูลใหม่อีกครั้ง
- การส่งข้อมูลนี้ หากมีการหน่วงเวลาการส่งข้อมูลออกไป จะช่วยในเรื่องของการประยัดพลังงานได้ แต่การตอบสนองก็จะช้าลงตามไปด้วย
- ในการนี้ที่อุปกรณ์ที่ทำหน้าที่เป็นตัวกลาง ต้องการที่จะมีการตรวจสอบการตอบสนอง ก็สามารถที่จะส่งข้อมูล Scan response payload เพื่อรับข้อมูลอื่นๆ จากอุปกรณ์ลูกข่ายได้

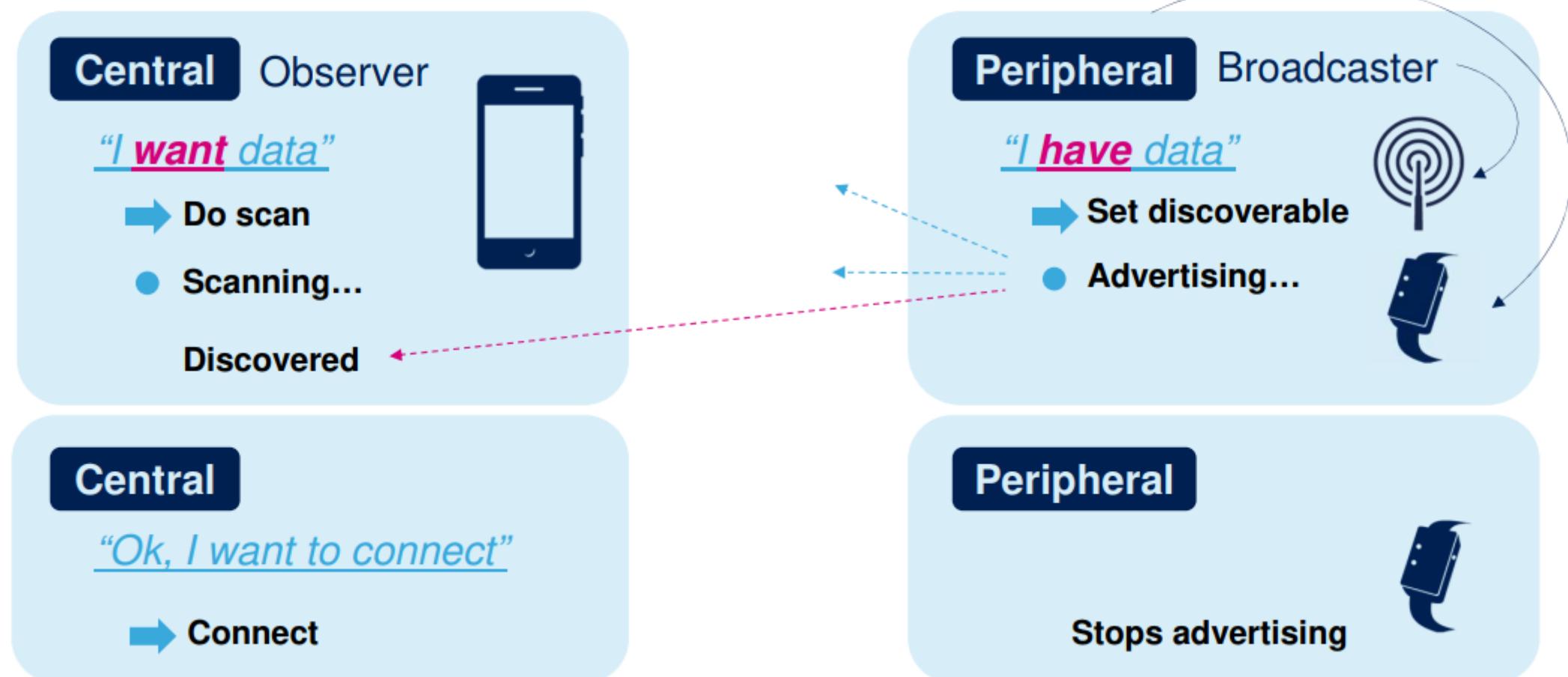


Broadcast Network Topology



GAP – Generic Access Profile

- GAP layer controls **advertising** and **connections**
(makes a device visible to the outside world)
- Also determines how two devices can interact with each other



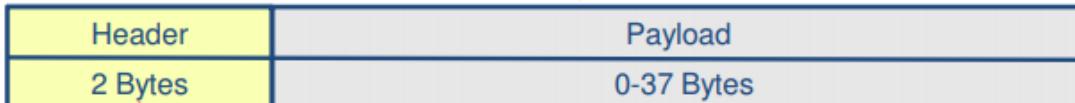
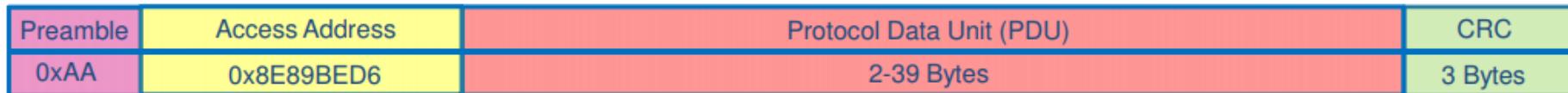
Advertising Packet Format

10101010 - used to sync receiver

0x8E89BED6 for ALL advertising packets

Time on air

$$1+4+2+6+3 = 128\text{bits} * 1\text{Mbps} = 128\mu\text{s min}$$

$$1+4+2+37+3 = 376\text{bits} * 1\text{Mbps} = 376\mu\text{s max}$$


ADV PDU type / Payload size

PDU Type	RFU	TxAdd	RxAdd	Length	RFU
4 bits	2 bits	1 bits	1 bits	6 bits	2 bits



Advertise capabilities (advertiser) or respond (scanner)

- "I am connectable" (Advertiser)
- "I would like to connect" (Initiator)
- "I am scannable but not connectable" (Advertiser)
- "I am requesting your scannable data" (Scanner)

48-bit Device Address

- Sometimes called "Bluetooth MAC Address"
- Only 31 bytes left for useful data!

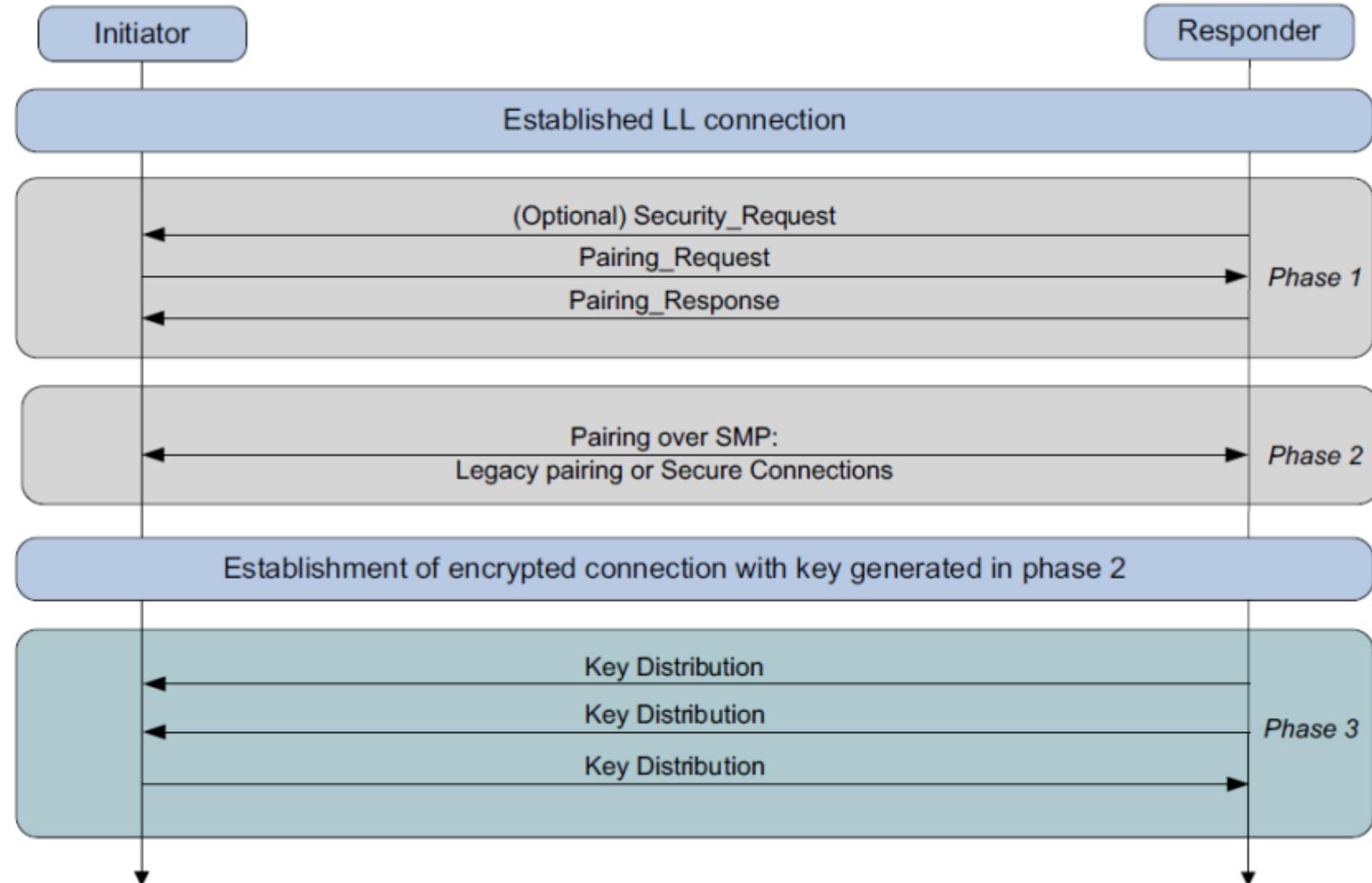
LE 5.0 support extended broadcast packet length of 255 bytes on other 0~36 channels

- Common Advertising types (**AD Type**):
- 0x01 Flags
 - 0x02 16 UUID – SIG assigned
 - 0x06 128-bit UUID – Vendor Specific
 - 0x09 Complete local name
 - 0x0A Tx Power level
 - 0xFF Mfr-specific data



Pairing is performed to establish keys which can then be used to encrypt a link.

- **Phase 1:** Pairing Feature Exchange
- **Phase 2 (LE legacy pairing):** Short Term Key (STK) Generation
- **Phase 2 (LE Secure Connections):** Long Term Key (LTK) Generation
- **Phase 3:** Transport Specific Key Distribution

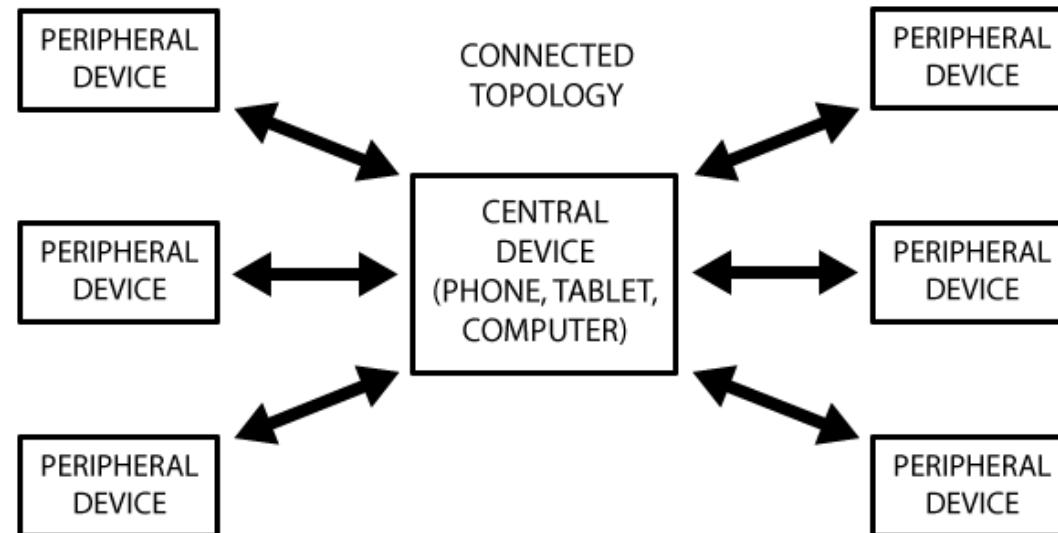


GATT

- ย่อมาจาก Generic Attribute Profile
- เป็นการกำหนดการสื่อสารระหว่างอุปกรณ์ที่ใช้ Bluetooth 2 ตัว โดยใช้แนวคิดของ Service และ Characteristic ซึ่งเก็บรายละเอียดของการให้บริการ, คุณสมบัติ, และข้อมูลอื่นๆ ที่เกี่ยวข้อง โดยอยู่ในรูปของเลขฐานสอง 16 บิต
- มีรูปแบบการสื่อสารที่แตกต่างจาก GAP โดยที่ GATT จะเป็นการสื่อสารระหว่างอุปกรณ์ 2 ตัว เท่านั้น โดยเมื่อมีการเชื่อมต่อแล้ว จะไม่มีการกระจายการสื่อสารไปยังอุปกรณ์อื่นๆ ในบริเวณใกล้เคียง และอุปกรณ์อื่นๆ จะไม่สามารถมองเห็นได้ จนกว่าจะมีการตัดการเชื่อมต่อกับอุปกรณ์ตัวนั้น
 - ตัวอย่างเช่น มีเพียงอุปกรณ์ BLE ที่เป็น Peripheral เพียงตัวเดียว ที่สามารถเชื่อมต่อกับอุปกรณ์ที่เป็น Central ได้ในช่วงเวลาหนึ่ง
- หากต้องมีการสื่อสารทั้ง 2 ทิศทาง จะต้องใช้ GATT เท่านั้น

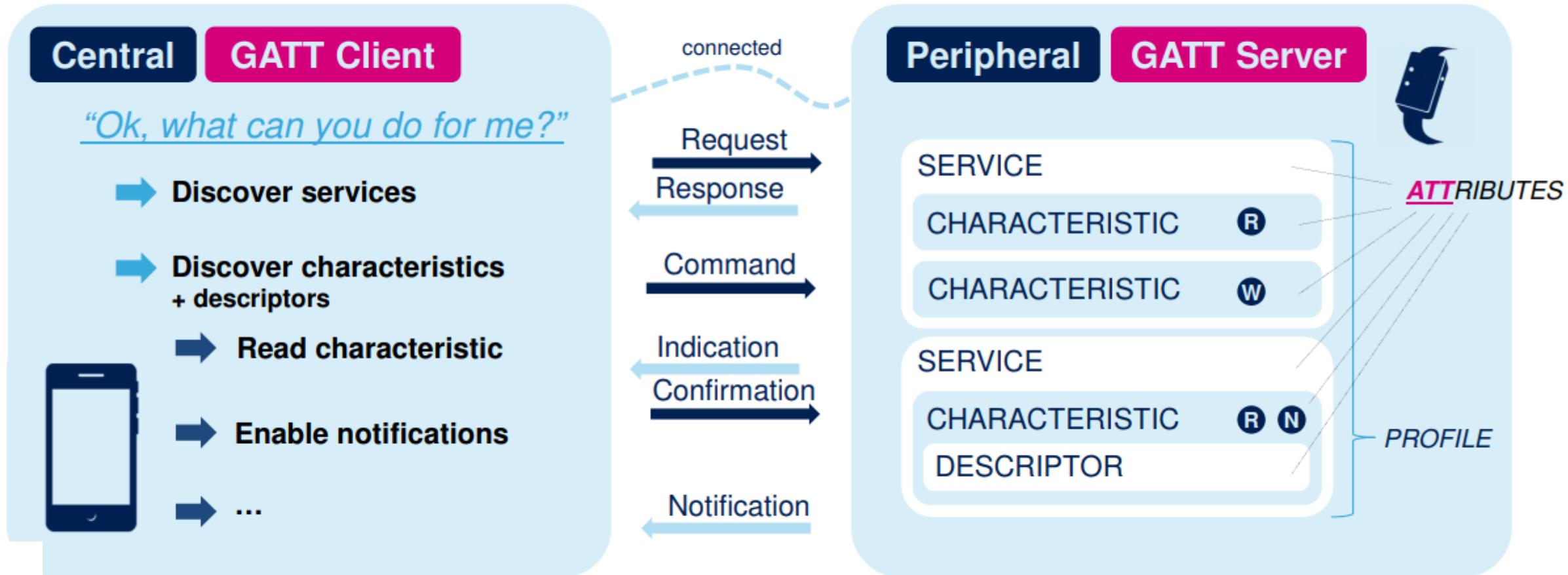
GATT Network Topology

- Central Device สามารถเชื่อมต่อกับอุปกรณ์ Peripheral ตัวอื่นๆ ได้ แต่อุปกรณ์ Peripheral จะต้องสื่อสารกับอุปกรณ์ Central ตัวเดียวเท่านั้น
- ถึงแม้จะเป็นการสื่อสารที่ต้องติดต่อกับอุปกรณ์เพียงตัวเดียว แต่การสื่อสารสามารถทำได้ทั้ง 2 ทาง (ได้ทั้งรับและส่งข้อมูล) ซึ่งแตกต่างจาก GAP



GATT – Generic Attribute Profile

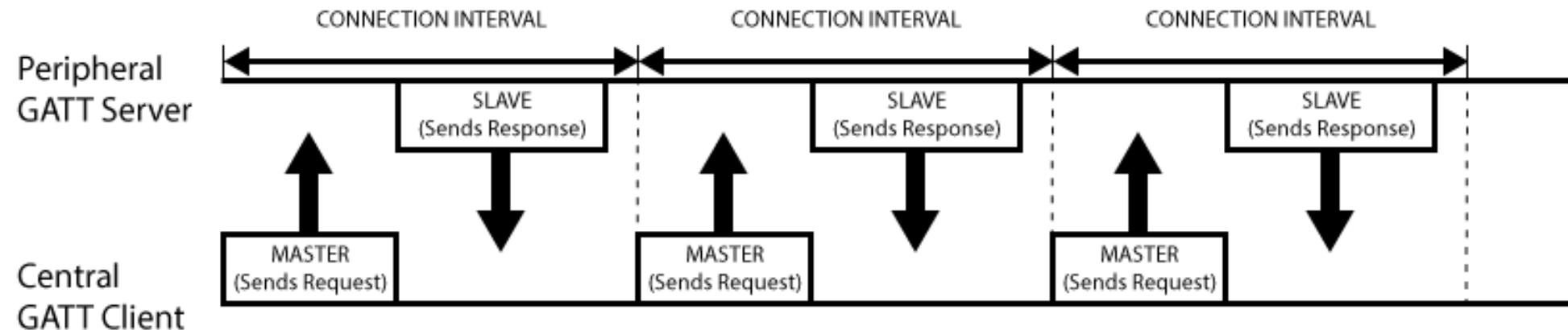
- Defines the way how two BLE devices exchange data



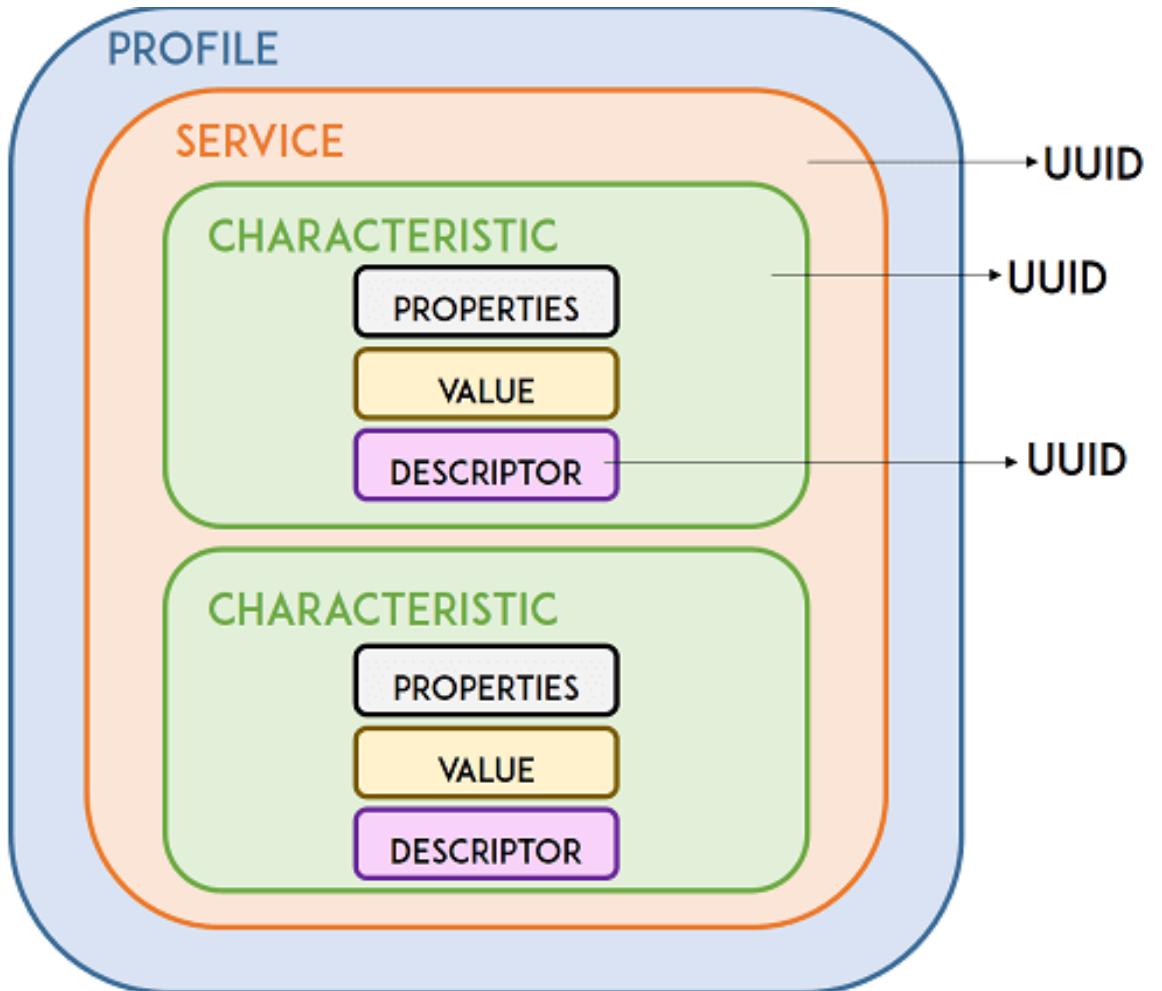
Communication attempt repeated with connection interval defined by central (suggested by peripheral)

GATT Transactions

- GATT มีลักษณะการสื่อสารแบบ Client/Server Model
- อุปกรณ์ Peripheral จะทำหน้าที่เป็น GATT Server ส่วน Central device จะทำหน้าที่เป็น Client
- ในการสื่อสาร จะเริ่มต้นจากอุปกรณ์ Master (GATT Client) ส่งข้อมูลร้องขอ และรอรับข้อมูลจากอุปกรณ์ Slave (GATT Server)
- เมื่อเชื่อมต่อกันได้แล้ว อุปกรณ์ที่เป็น Peripheral จะกำหนดช่วงเวลาในการสื่อสาร (Connection Interval) ให้กับอุปกรณ์ตัวกลาง และอุปกรณ์ตัวกลางก็จะทำการเชื่อมต่อใหม่ทุกรังสรรค์ตามช่วงเวลาที่กำหนด เพื่อตรวจสอบว่ามีข้อมูลใหม่ให้นำไปใช้งานหรือไม่



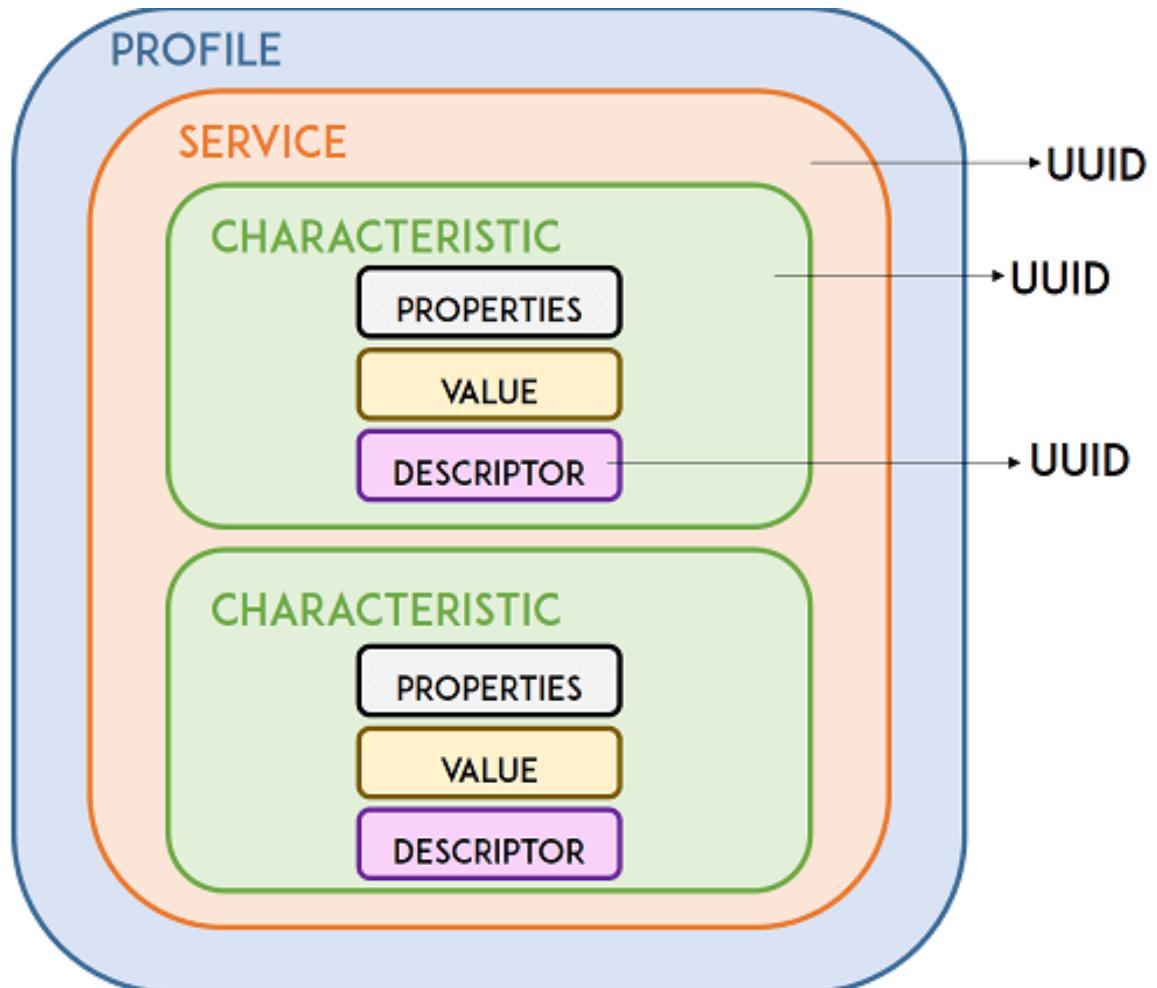
Services & Characteristic



Profiles

- ส่วนของโปรไฟล์สำหรับ Service ต่างๆ
- ถูกกำหนดโดยผู้ผลิตอุปกรณ์ หรือองค์กร Bluetooth SIG เช่น โปรไฟล์สำหรับการวัดอัตราการเต้นของหัวใจ (Heart Rate Profile)

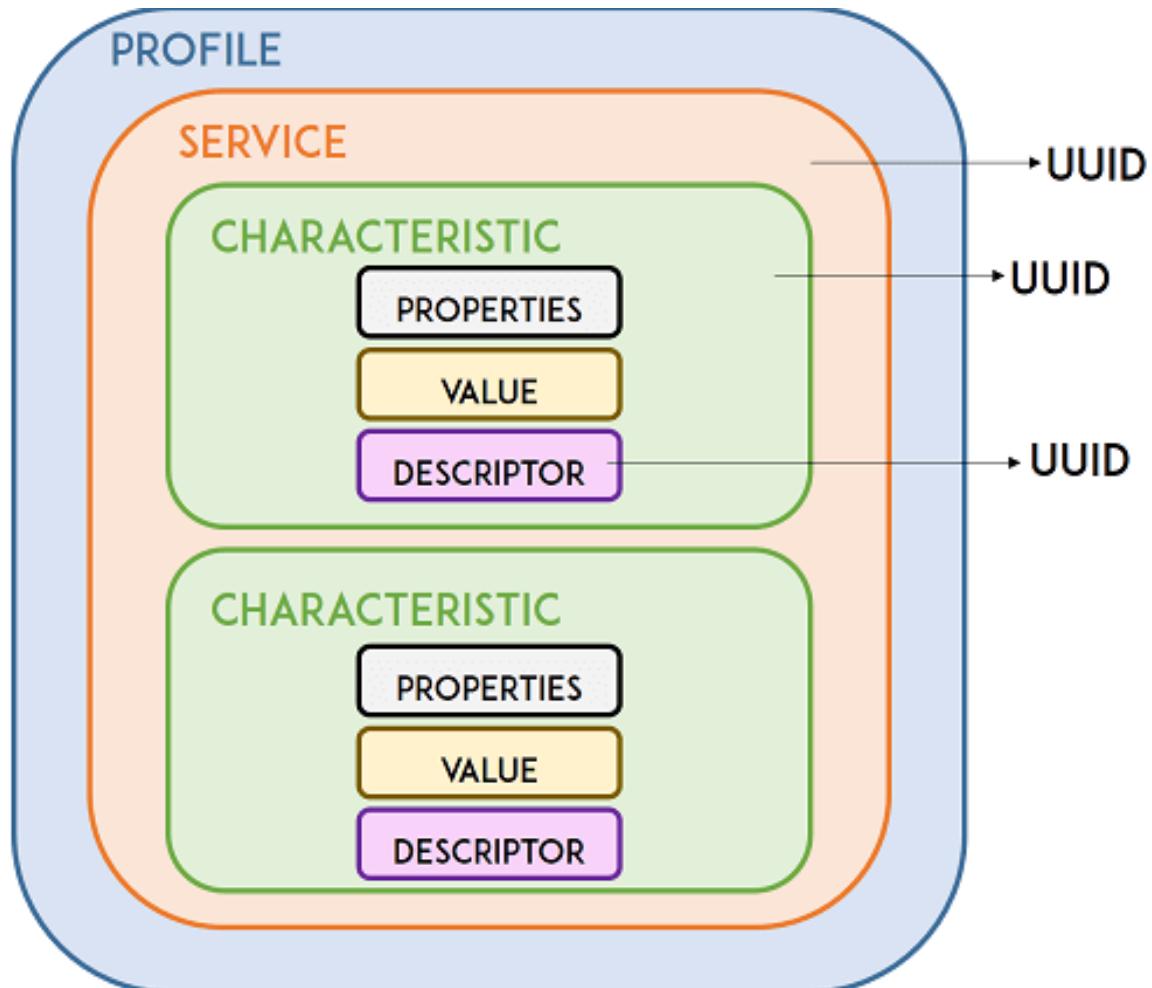
Services & Characteristic



Services

- เป็นส่วนของการกำหนดบริการต่างๆ ที่เกี่ยวข้องในแต่ละด้าน
- ประกอบด้วยข้อมูลคุณสมบัติต่างๆ ที่เกี่ยวข้อง (Characteristic)
- แต่ละ Service จะมีหมายเลขกำหนดให้เฉพาะ เพื่อแยกการให้บริการแต่ละแบบออกจากกัน เรียกหมายเลขนี้ว่า UUID
- หมายเลข UUID มีขนาด 16 บิต สำหรับบริการที่มีการกำหนดอย่างเป็นทางการจากองค์กรที่เกี่ยวข้อง หรือขนาด 128 บิต สำหรับบริการที่มีการสร้างขึ้นมาเองเพิ่มเติม (เช่นโดยผู้ผลิต)

Services & Characteristic



Characteristic

- เป็นกลุ่มข้อมูลที่กำหนดคุณสมบัติของการติดต่อสื่อสาร ถือว่าเป็นระดับล่างสุดของการกำหนดรูปแบบการสื่อสารของ BLE
- มีการกำหนดคุณสมบัติที่เกี่ยวข้องกับการใช้งานในแต่ละด้าน เช่น พิลด์ข้อมูลค่า X,Y,Z ของเซ็นเซอร์วัดความเร่ง 3 แกน
- กำหนดด้วยหมายเลขขนาด 16 บิต หรือ 128 บิต
- ถือเป็นองค์ประกอบสำคัญที่เกี่ยวข้องกับการสื่อสารระหว่างอุปกรณ์โดยตรง

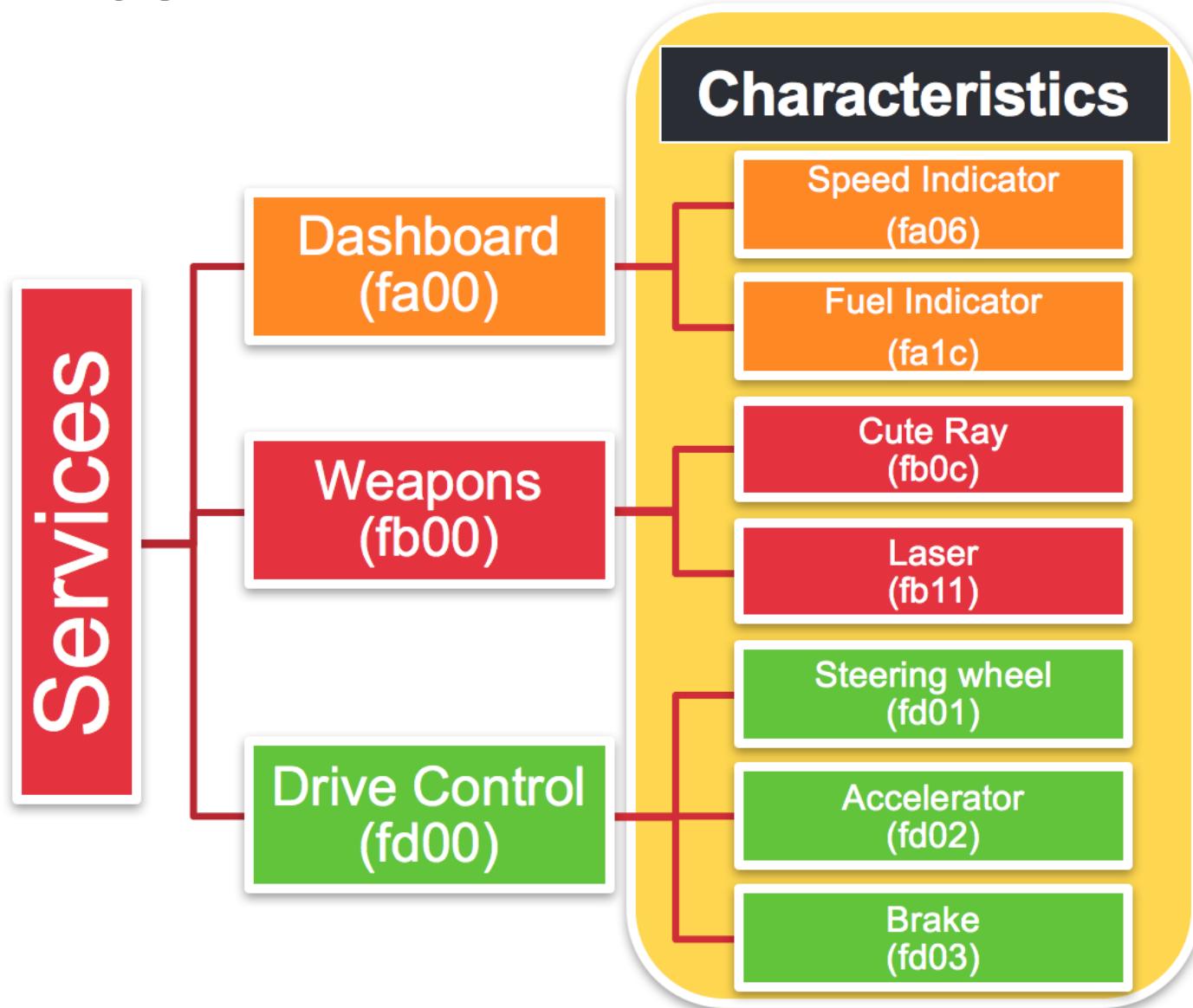
BLE Service

Name	Uniform Type Identifier	Assigned Number	Specification
Generic Access	org.bluetooth.service.generic_access	0x1800	GSS
Alert Notification Service	org.bluetooth.service.alert_notification	0x1811	GSS
Automation IO	org.bluetooth.service.automation_io	0x1815	GSS
Battery Service	org.bluetooth.service.battery_service	0x180F	GSS
Blood Pressure	org.bluetooth.service.blood_pressure	0x1810	GSS
Body Composition	org.bluetooth.service.body_composition	0x181B	GSS
Bond Management Service	org.bluetooth.service.bond_management	0x181E	GSS
Continuous Glucose Monitoring	org.bluetooth.service.continuous_glucose_monitoring	0x181F	GSS
Current Time Service	org.bluetooth.service.current_time	0x1805	GSS
Cycling Power	org.bluetooth.service.cycling_power	0x1818	GSS
Cycling Speed and Cadence	org.bluetooth.service.cycling_speed_and_cadence	0x1816	GSS
Device Information	org.bluetooth.service.device_information	0x180A	GSS
Environmental Sensing	org.bluetooth.service.environmental_sensing	0x181A	GSS
Fitness Machine	org.bluetooth.service.fitness_machine	0x1826	GSS
Generic Attribute	org.bluetooth.service.generic_attribute	0x1801	GSS

BLE Service

GATT Descriptor	0x2907	External Report Reference
GATT Descriptor	0x2908	Report Reference
GATT Descriptor	0x2909	Number of Digits
GATT Descriptor	0x290A	Value Trigger Setting
GATT Descriptor	0x290B	Environmental Sensing Configuration
GATT Descriptor	0x290C	Environmental Sensing Measurement
GATT Descriptor	0x290D	Environmental Sensing Trigger Setting
GATT Descriptor	0x290E	Time Trigger Setting
GATT Service	0x1800	Generic Access
GATT Service	0x1801	Generic Attribute
GATT Service	0x1802	Immediate Alert
GATT Service	0x1803	Link Loss
GATT Service	0x1804	Tx Power
GATT Service	0x1805	Current Time
GATT Service	0x1806	Reference Time Update
GATT Service	0x1807	Next DST Change
GATT Service	0x1808	Glucose
GATT Service	0x1809	Health Thermometer
GATT Service	0x180A	Device Information
GATT Service	0x180D	Heart Rate
GATT Service	0x180E	Phone Alert Status
GATT Service	0x180F	Battery
GATT Service	0x1810	Blood Pressure
GATT Service	0x1811	Alert Notification
GATT Service	0x1812	Human Interface Device

BLE Service



BLE Characteristic

คุณลักษณะนี้เป็นของบริการเสมอและเป็นที่ที่ข้อมูลจริงอยู่ในลำดับชั้น (value) คุณลักษณะมีสองคุณลักษณะเสมอ: การประมวลผลคุณลักษณะ (ที่ให้ข้อมูลเมตากiejaw กับข้อมูล) และค่าคุณลักษณะ

นอกจากนี้ค่าคุณลักษณะสามารถตามด้วยตัวบอกซึ่งขยายเพิ่มเติมเกี่ยวกับข้อมูลเมตากับที่มีอยู่ในการประมวลผลคุณลักษณะคุณสมบัติอธิบายว่าค่าคุณลักษณะสามารถต่อตัวกับได้อย่างไร โดยทั่วไปจะมีการดำเนินการและขั้นตอนที่สามารถใช้ได้กับคุณสมบัติ:

- Broadcast
- **Read**
- Write without response
- **Write**
- **Notify**
- Indicate
- Authenticated Signed Writes
- Extended Properties

BLE UUID

บริการคุณลักษณะ (characteristic) และตัวบ่งชี้ (descriptor) มี UUID (Universally Unique Identifier) UUID គឺ
หมายเลขเฉพาะ 128 បិត (16 បូព្យ) តាមរយៈរៀងចែន:

55072829-bc9e-4c53-938a-74a6d4c78776

UUIDs ដែលទទួលការគ្រប់គ្រងនៅក្នុង SIG ត្រូវបានបង្ហាញនៅក្នុងការពារនា
<https://www.bluetooth.com/specifications/gatt/services/>

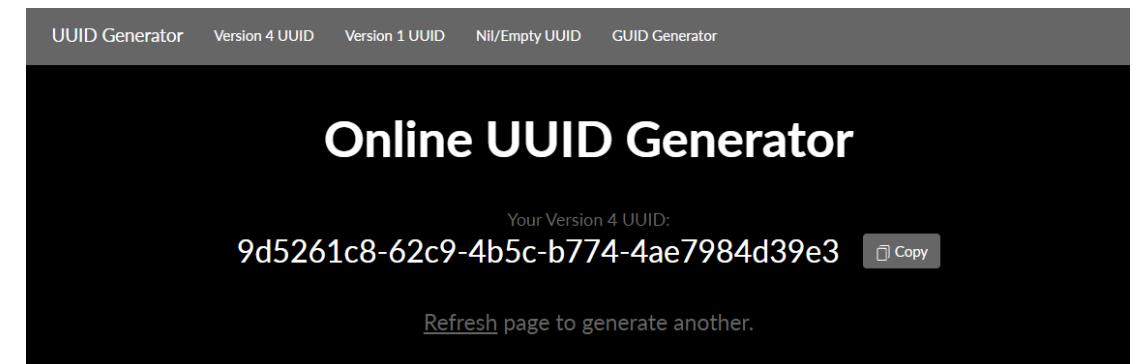
នៅក្នុងការពារនាអាជីវកម្ម នឹងបានបង្ហាញនៅក្នុងការពារនាអាជីវកម្ម ដោយប្រើប្រាស់ការបង្ហាញ
នៃ UUID នេះ
<https://www.uuidgenerator.net/>

ដើម្បីបង្ហាញ UUID នៃការពារនាអាជីវកម្ម ត្រូវបានបង្ហាញនៅក្នុងការពារនាអាជីវកម្ម ដោយប្រើប្រាស់ការបង្ហាញ

List of Bluetooth profiles

- รายละเอียดของ Bluetooth profile สำหรับใช้ด้านต่าง สามารถศึกษาได้ที่ URL นี้
- https://en.wikipedia.org/wiki/List_of_Bluetooth_profiles

- Generating UUIDs
- <https://www.uuidgenerator.net/>



The screenshot shows the 'UUID Generator' interface with two main sections: 'Version 1 UUID Generator' and 'Version 4 UUID Generator'.
The 'Version 1 UUID Generator' section contains a 'Generate a version 1 UUID' button. Below it is a 'Bulk Version 1 UUID Generation' form with fields for 'How Many?' (set to 'Max 500'), a 'Generate' button, and a 'Download to a file' link.
The 'Version 4 UUID Generator' section contains a 'Generate a version 4 UUID' button. Below it is a 'Bulk Version 4 UUID Generation' form with fields for 'How Many?' (set to 'Max 500'), a 'Generate' button, and a 'Download to a file' link.
Both sections include a 'What is a [Version] UUID?' link and a detailed description of what each type of UUID is.



GAP GATT

- GAP และ GATT ทำงานร่วมกันเพื่อให้การเชื่อมต่อ BLE เป็นไปอย่างราบรื่นและมีประสิทธิภาพ โดย GAP ดูแล การเชื่อมต่อและการตั้งค่าในขณะที่ GATT ดูแลการแลกเปลี่ยนข้อมูลที่เกิดขึ้นระหว่างการเชื่อมต่อ

ESP32

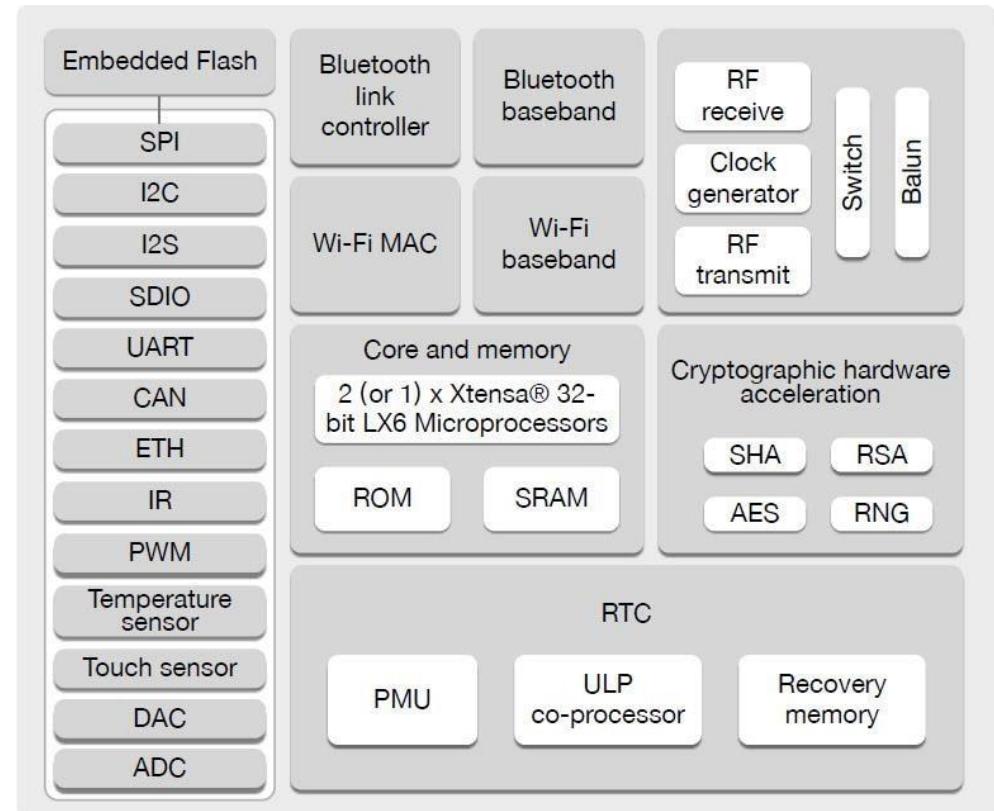
Bluetooth



ESP32 Bluetooth

โมดูลสื่อสาร Bluetooth ใน ESP32 มีคุณสมบัติเบื้องต้น

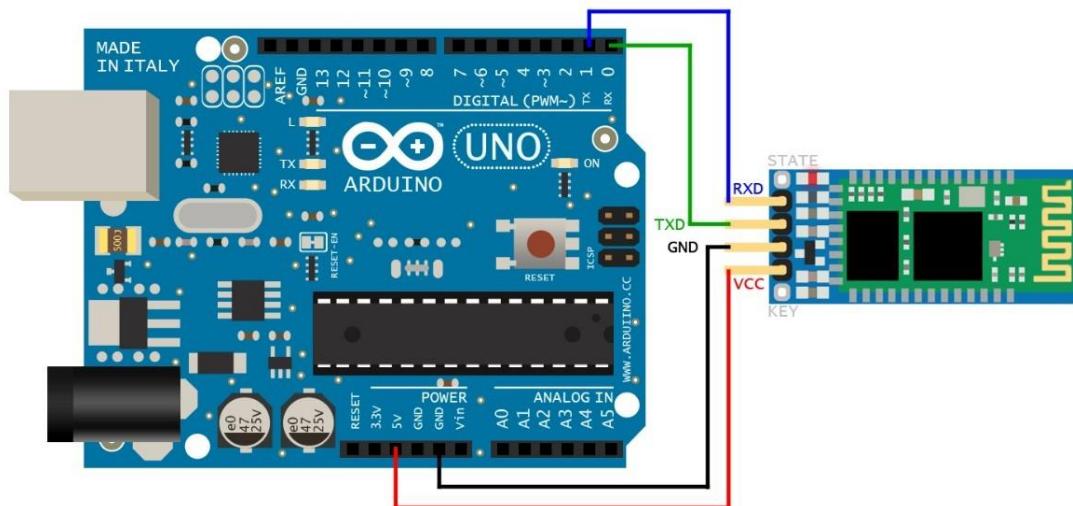
- รองรับการทำงานของ Bluetooth V4.2
- รองรับการส่งข้อมูลที่ Class 1, 2 และ 3
- กำลังส่งสูงสุดที่ +12 dBm
- รองรับการทำโครงข่าย Piconet และ Scatternet
- รองรับการเชื่อมต่อกับ Bluetooth รุ่นก่อนๆ (Classic BT) และ Bluetooth พลังงานต่ำ (BLE)
- สามารถทำการแพร่กระจาย (Advertising) และสแกน (Scanning) พร้อมกันได้



ESP32 Bluetooth

Bluetooth ใน ESP32 สามารถเลือกใช้งานได้ 2 แบบ คือ

- Classic Bluetooth (ก่อนหน้า V 4.0)
- Bluetooth Low Energy (ตั้งแต่ V 4.0 ขึ้นไป)

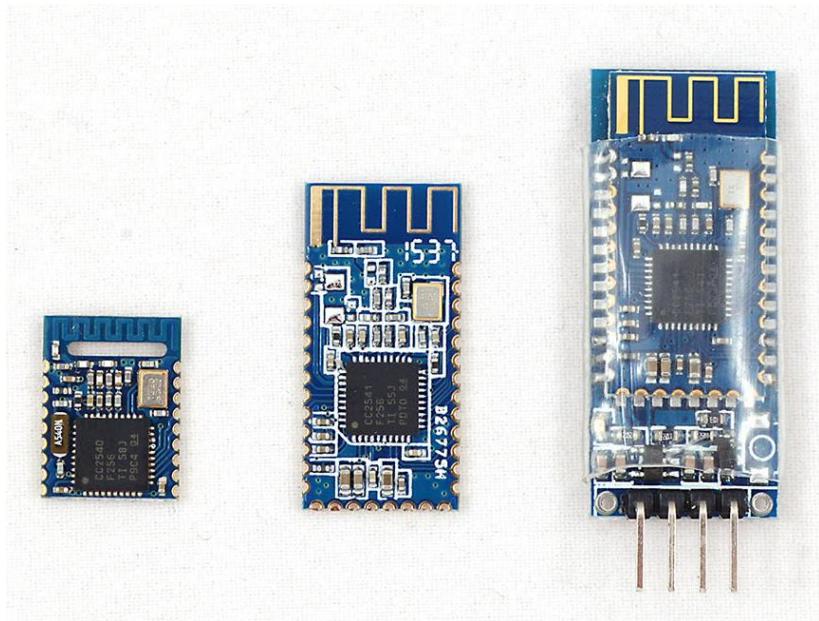


Classic Bluetooth

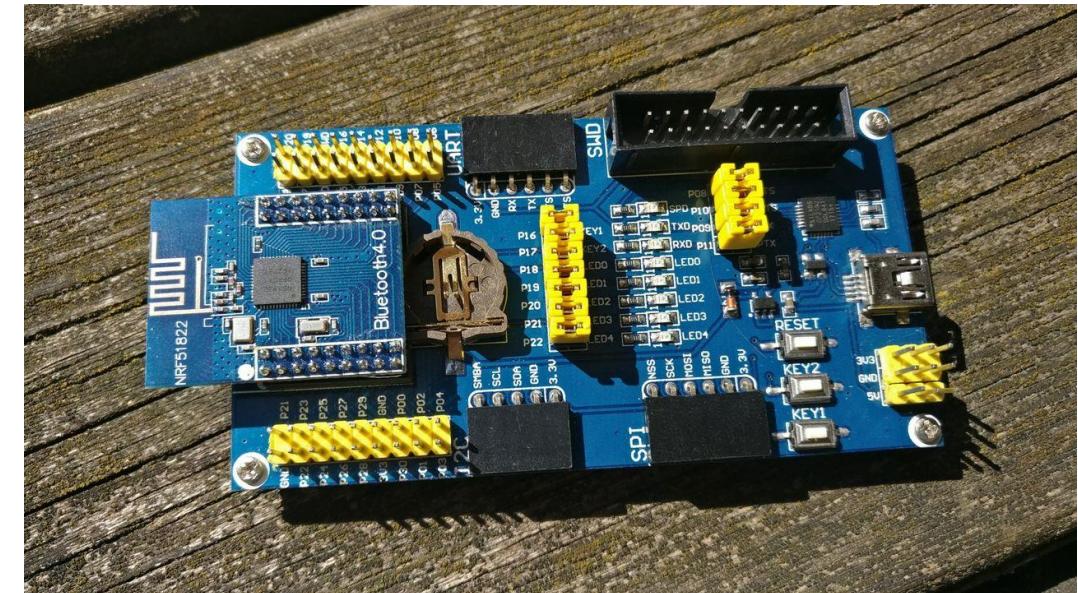
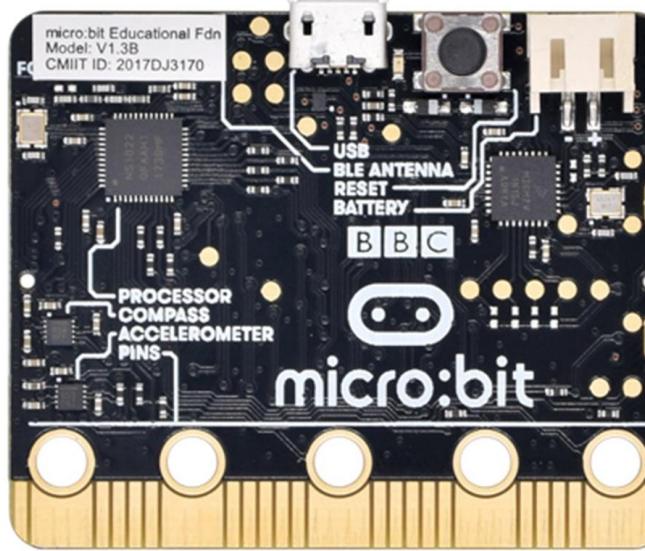
ESP32 Bluetooth

Bluetooth ใน ESP32 สามารถเลือกใช้งานได้ 2 แบบ คือ

- Classic Bluetooth (ก่อนหน้า V 4.0)
- Bluetooth Low Energy (ตั้งแต่ V 4.0 ขึ้นไป)



Bluetooth CC2541 (Texas Instruments)



Bluetooth nrf51822 (Nordic)

มีอะไรใหม่!

เปรียบเทียบจุดเด่นจุดด้อย
ของ Node32 Lite, CUCUMBER R



Espressif Module	ESP32-WROOM-32	ESP32-S2-WROVER
Technology	EPS32	EPS32-S2
MCU	Xtensa LX6	Xtensa LX7
Number of cores	2	1
Number of bits	32	32
Clock frequency	240MHz	240MHz
Processor Core	1 (ULP)	2 (ULP-RISCV, ULP-FSM)
WiFi	802.11 b/g/n	802.11 b/g/n
Time-of-Flight (TOF)	✗	✓
Bluetooth	BT 4.2 BR/EDR & BLE	✗
RAM	520 KB	320 KB
ROM	448 KB	128 KB
Internal Flash	4 MB	4 MB
PSRAM	✗	2 MB
RTC Memory	8 KB	16 KB
GPIO	31	38
USB OTG	✗	✓
SPI	4	4

มีอะไรใหม่!

เปรียบเทียบจุดเด่นจุดด้อย
ของ Node32 Lite, CUCUMBER R



I2C	2	2
I2S	2	1
UART	3	2
ADC	18	20
ADC resolution	12-bit	12-bit
DAC	2	2
DAC resolution	8-bit	8-bit
Software PWM	16	8
SDMMC interface	✓	✗
Temperature sensor	✓	✓
Touch sensor	10	14
LCD Interface	✗	✓
Camera Interface	✗	✓
Can	1	✗
Ethernet MAC	1	✗
Security	Secure boot Flash encryption	Secure boot Flash encryption 4096 - bit OTP
Crypto	AES, SHA-2, RSA, ECC, RNG	AES-128/192/256, SHA-2, RSA, RNG, HMAC, Digital signature

ESP32 Bluetooth Classic ArduinoIDE Library

ESP32 Bluetooth Serial Function

- กรณีต้องการใช้งาน Bluetooth ในการสื่อสารแบบอนุกรม จะต้องนำเข้าไลบรารีที่ชื่อว่า BluetoothSerial.h และจะต้องสร้างออปเจ็คจากคลาส BluetoothSerial เพื่อเข้าใช้งานไลบรารีดังกล่าว (ในที่นี่จะใช้ชื่อว่า SerialBT แต่สามารถตั้งชื่ออื่นได้)
- ตัวอย่างการนำเข้าไลบรารีและการสร้างออปเจ็ค แสดงได้ดังนี้

```
#include "BluetoothSerial.h"
```

```
BluetoothSerial SerialBT;
```

- ในการทดสอบเบื้องต้น อาจทดลองนำอุปกรณ์ที่รองรับ Bluetooth เชื่อมต่อเข้ากับ ESP32 โดยในกรณีที่ใช้ Smartphone สามารถดาวน์โหลดโปรแกรม Serial Bluetooth Terminal เพื่อทดสอบเบื้องต้นได้โดยสามารถใช้ได้ทั้งระบบปฏิบัติการ Android และ iOS

ESP32 Bluetooth Serial Function

- SerialBT.begin(name)
- เป็นฟังก์ชันสำหรับเริ่มต้นการทำงานของไลบรารี BluetoothSerial โดยจะต้องระบุชื่อของอุปกรณ์ซึ่งเป็นพารามิเตอร์ของฟังก์ชันนี้
- SerialBT.write()
- เป็นฟังก์ชันสำหรับการเขียนข้อมูลออกทาง BluetoothSerial

ESP32 Bluetooth Serial Function

- SerialBT.read()
- เป็นฟังก์ชันสำหรับอ่านข้อมูลเข้ามาทาง BluetoothSerial
- SerialBT.available()
- ใช้สำหรับตรวจสอบบัฟเฟอร์ BluetoothSerial ว่ามีข้อมูลหรือไม่

ESP32 Bluetooth Serial Function

Android

App > Serial Bluetooth Terminal

Link: https://play.google.com/store/apps/details?id=de.kai_morich.serial_bluetooth_terminal

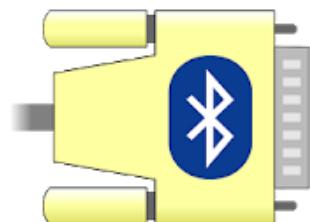
iOS

ไม่สามารถใช้งาน Bluetooth Classic ใน iOS ได้

Window

App > Bluetooth Serial Terminal

Link: <https://www.microsoft.com/th-th/p/bluetooth-serial-terminal/9wzdncrdfst8>



Serial Bluetooth Terminal

Kai Morich เครื่องมือ

★★★★★ 1,481

3+

เสนอการซื้อในแอป
แอปพลิเคชันนี้ทำงานร่วมกับอุปกรณ์ทั้งหมดของคุณได้

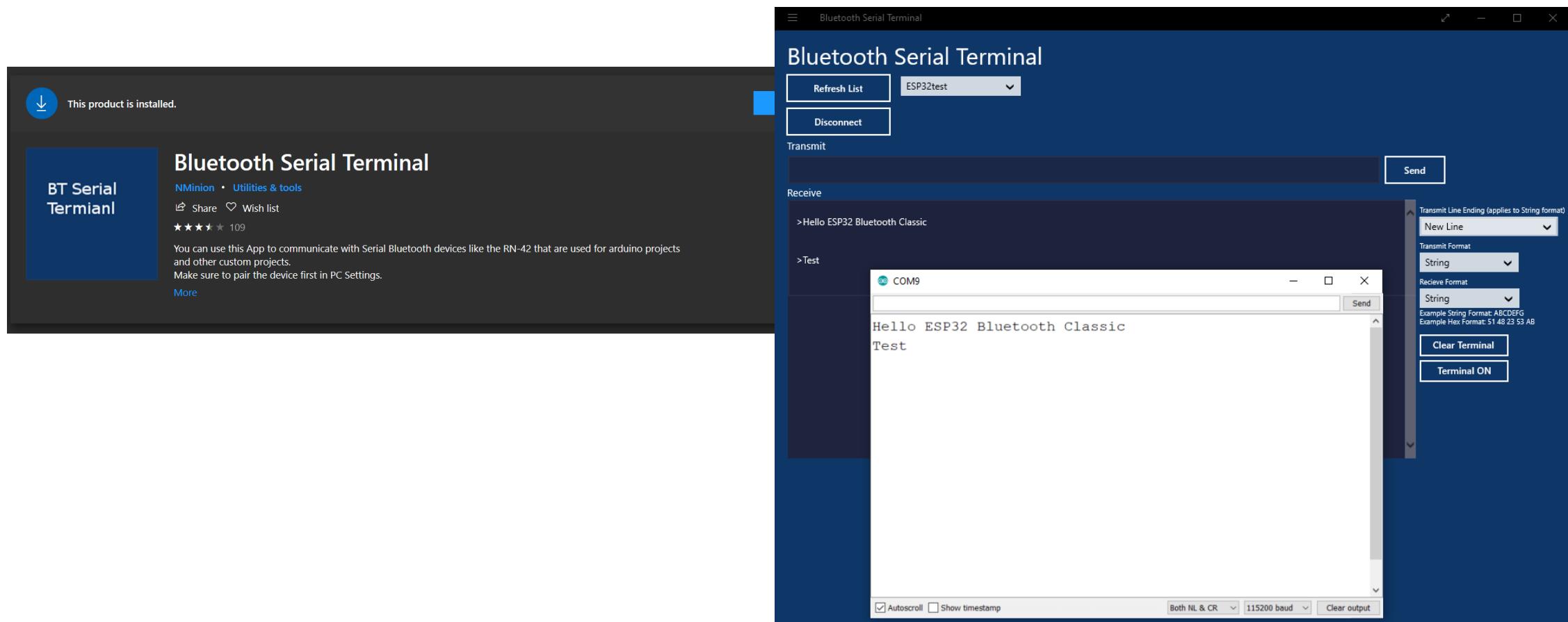
ติดตั้งแล้ว

ESP32 Bluetooth Serial Function

Window

App > Bluetooth Serial Terminal

Link: <https://www.microsoft.com/th-th/p/bluetooth-serial-terminal/9wzdncrdfst8>



ESP32 Bluetooth BLE ArduinolDE Library

ESP32 Bluetooth BLE

BLE Server

```
#include <BLEDevice.h>
#include <BLEUtils.h>
#include <BLESERVER.h>
#include <BLE2902.h>
```

BLE Scan

```
#include <BLEDevice.h>
#include <BLEUtils.h>
#include <BLEScan.h>
#include <BLEAdvertisedDevice.h>
```

BLE Server จะเรียกใช้ 7 Function

1. init()
2. createServer()
3. createService()
4. createCharacteristic()
5. setValue()
6. getAdvertising()
7. start()

ESP32 Bluetooth BLE

อธิบายการใช้งานแต่ละ Function ผ่านโค้ดตัวอย่าง Lab11.4

Android

App > nRF Connect

Link: <https://play.google.com/store/apps/details?id=no.nordicsemi.android.mcp>

iOS

App > nRF Connect

Link: <https://apps.apple.com/us/app/nrf-connect/id1054362403>

ESP32 Bluetooth BLE



nRF Connect for Mobile

Nordic Semiconductor ASA เครื่องมือ

★★★★★ 1,852 รายการ

3+

แอปพลิเคชันนี้ทำงานร่วมกับอุปกรณ์ทั้งหมดของคุณได้

เพิ่มในรายการสิ่งที่อยากรับ

ติดต่อ

Scanner

No Filter

ESP32_B01B1412CFA4

Services: 4FAFC201-1FB5-459E-8FCC-C5C9C331914B

Tx Power: 3 dBm

N/A N/A

N/A

Manufacturer Data: Microsoft <0x0006>

0x01-09-20-02-8A-08-

E1-5E-3E-38-8D-80-72-4D-3F-23-4C-4F-D0-19-D4-3B-

AA-08-D6-AF-03

N/A N/A

N/A

Manufacturer Data: Microsoft <0x0006>

0x01-09-20-02-9A-63-EF-42-DA-FD-6E-79-E6-D5-F0-12-

FD-02-4E-BA-01-AC-61-A5-F4-AD-6C

N/A N/A

N/A

Manufacturer Data: Microsoft <0x0006> 0x01-09-20-02-

EC-B5-6B-BB-BB-44-93-52-C6-61-57-54-E4-F5-28-12-

A7-B4-C8-6B-E5-CC-95

Scanner RSSI Graph Advertiser Settings

Scanner

No Filter

ESP32_B01B1412CFA4

Services: 4FAFC201-1FB5-459E-8FCC-C5C9C331914B

Tx Power: 3 dBm

N/A N/A

N/A

Manufacturer Data: Microsoft <0x0006>

0x01-09-20-02-9A-63-EF-42-DA-FD-6E-79-E6-D5-F0-12-

FD-02-4E-BA-01-AC-61-A5-F4-AD-6C

N/A N/A

N/A

Manufacturer Data: Microsoft <0x0006> 0x01-09-20-02-

EC-B5-6B-BB-BB-44-93-52-C6-61-57-54-E4-F5-28-12-

A7-B4-C8-6B-E5-CC-95

Connected

Scanner RSSI Graph Advertiser Settings

Close Services Disconnect

Advertised Services

Unknown Service

UUID: 4FAFC201-1FB5-459E-8FCC-C5C9C331914B

Attribute Table

Generic Access

UUID: 1800

PRIMARY SERVICE

Generic Attribute

UUID: 1801

PRIMARY SERVICE

Unknown Service

UUID: 4FAFC201-1FB5-459E-8FCC-C5C9C331914B

PRIMARY SERVICE

Unknown Characteristic

UUID: BEB5483E-36E1-4688-B7F5-EA07361B26A8

Properties: Read and Write

Value:

0x48-65-6C-6F-20-57-6F-72-6C-64-20-42-4C-45-20-45-53-

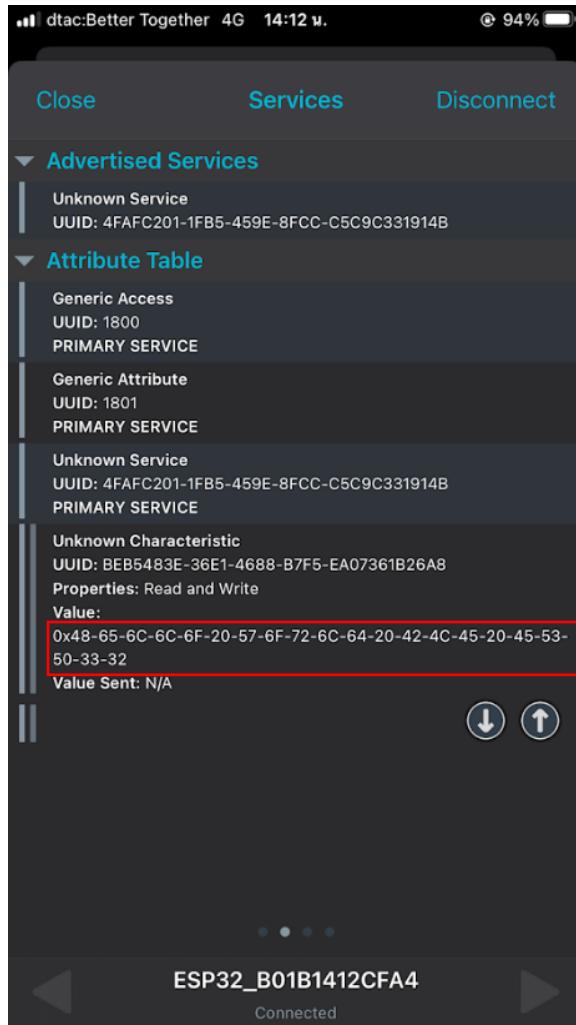
50-33-32

Value Sent: N/A

Connected

Scanner RSSI Graph Advertiser Settings

ESP32 Bluetooth BLE



nRF Connect for Mobile

Nordic Semiconductor ASA เครื่องมือ

★★★★★ 1,852

3+

แอปพลิเคชันนี้ทำงานร่วมกับอุปกรณ์ทั้งหมดของคุณได้

เพิ่มในรายการสั่งที่อยากรับ

ติดต่อ

สามารถแปลง Hex to String ได้จาก
<https://www.binaryhexconverter.com/hex-to-ascii-text-converter>

Hexadecimal Value

48656c6c6f20576f726c6420424c45204553503332

Convert

Ascii (String)

Hello World BLE ESP32

swap conversion: [Ascii Text To Hexadecimal Converter](#)