RF TECHNICAL TRAINING

MAY 2017

PAOLO CHIARLONE
SENIOR SALES ENGINEER

TODAY WE WILL LEARN...

THE DIGI RF PRODUCTS!!!

PROFESSOR:
PAOLO CHIARLONE

HELLO… I AM…

AGENDA

- General introduction and info
- RF concepts, Physics
- Wireless Protocols and Topology
- ZigBee vs. DigiMesh
- XBee Modules
- ZigBee Protocol
- XBee Software
- XCTU
- Tests / Diagnostics
- Demos
- Programmable XBee
- Digi RF Gateways
- Examples of Applications
- Appendix (additional info)

BUSINESS SOLUTIONS

WHY ARE WE HERE?

THE MARKET NEEDS

BUILDING CONNECTED PRODUCTS FOR THE INTERNET OF THINGS

- Address the specific needs of customers, by building intelligent connected products

FOCUS ON CUSTOMER'S CORE COMPETENCY AND TIME-TO-MARKET

- BSP integration, wireless, device security, regulatory requirements, and applications designs are not core competencies of customers

SECURITY, PERFORMANCE AND REPUTATION

- Device manufacturers need to build connected products that are safe, secure, compliant, without facing any potential issues that will affect reputation

ANATOMY OF AN M2M SOLUTION

CREATE

Relentlessly reliable products that get you to market faster.

EMBEDDED MODULES ▸ RF MODULES ▸

WIRELESS DESIGN SERVICES ▸

DEPLOY

Build critical infrastructures with proven wireless technology.

CELLULAR ROUTERS & GATEWAYS ▸

WIRELESS MODEMS ▸

MANAGE

Secure and maintain widely deployed connected devices.

DIGI TRUSTFENCE ▸ REMOTE MANAGER ▸

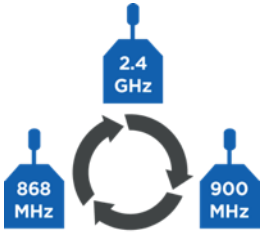
PROFESSIONAL SERVICES ▸

Digi covers every aspect of the entire M2M solution!

RF PRODUCT FAMILY

DIGI IS A PROVIDER OF INTEGRATED GATEWAY SOLUTIONS AND INDUSTRIAL RF RADIOS INCLUDING THE WORLD's #1 MOST SOLD MODULE

DIGI COMPLETE SOLUTION



MULTIPLE FREQUENCIES FOR WORLDWIDE DEPLOYMENT

868 MHz

900 MHz

2.4 GHz

MULTIPLE PROTOCOLS

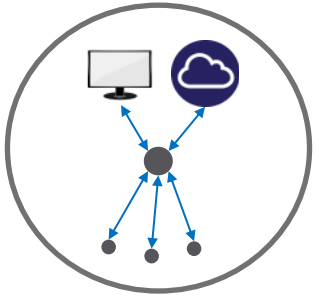
802.11 (Wi-Fi)

802.15.4

DigiMesh

ZigBee

Thread, more…



COMPLETE SOLUTION

Design Services

Hardware

Software

Certifications & Support

RF PHYSICS INTRO

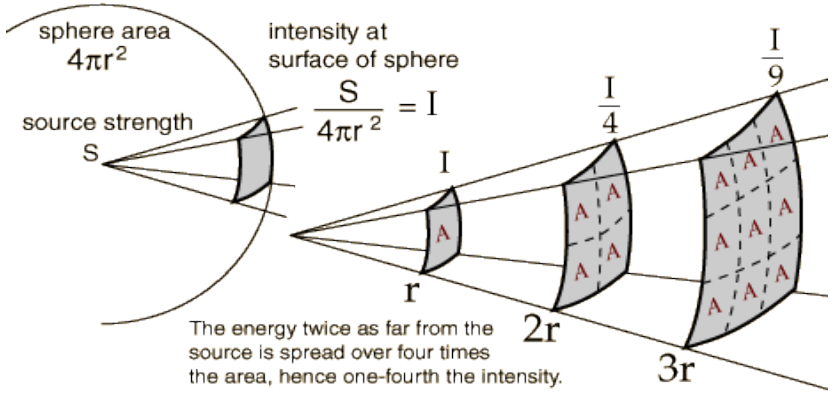
WIRE VS. WIRELESS

Over the air (Wireless) vs. Wire:

- There's only so much bandwidth in the air
  - Send lots of data, or send data to many things, but not both at the same time!!!
  - Wires – definable bandwidth – can always add another wire
- The lower the frequency the higher the range,
  but the smaller the amount of data
- Wireless has always higher latency than wire
- Wire is predictable and reliable
- Wireless is easier to deploy, moveable and cheaper in maintenance

INVERSE SQUARE LAW

- As a wave propagates out from the source, the total energy radiated from the source remains the same, but the strength of the wave decreases, as the distance from the source increases
- Hence, the power needs to increase exponentially with the distance: 3dBm is double power but 6dBm is double distance



SYSTEM GAIN VS. RANGE

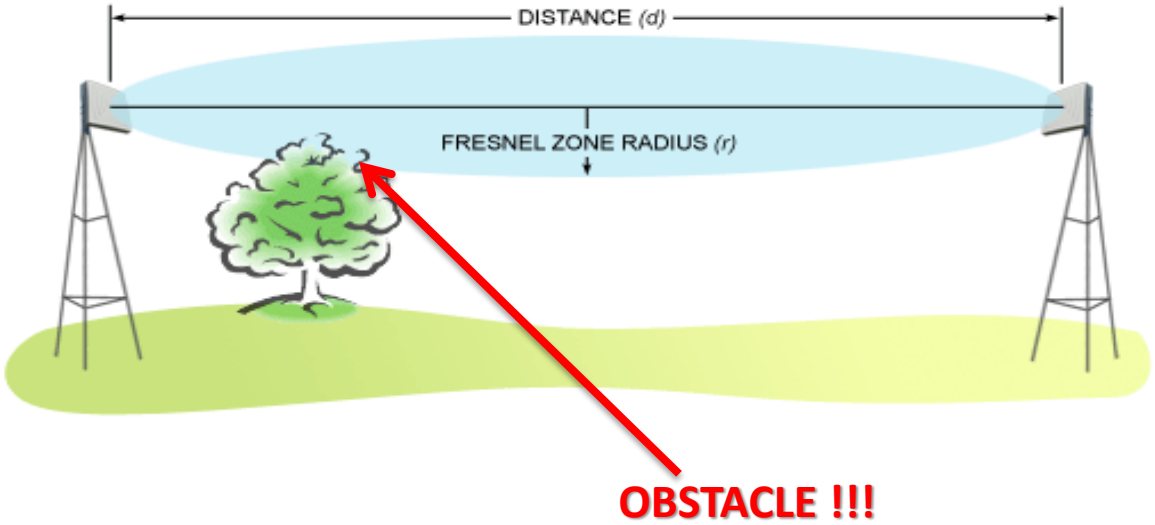
- **Rule of thumb**
  - ✓ 12dB of increased system gain is necessary to double the transmission distance in typical non RF line of sight conditions
  - ✓ 6dB of increased system gain is necessary to double the transmission distance in typical RF line of sight conditions
  - ✓ Same goes for the opposite - reduction!

FRESNEL ZONE

- American Football-shaped path
- Not a direct path – even if it is called LOS (Line-Of-Sight)
- Larger than you think!

PATH OBSTACLES

- Obstacles in the path will reduce both signal and range!
- Increase the antenna height to clear the path!

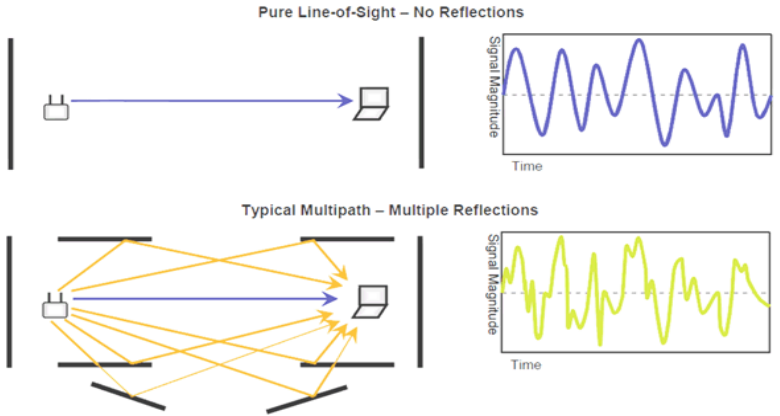


FRESNEL ZONE DIAMETERS

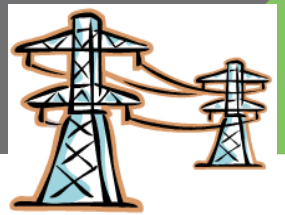
Range Distance	900 MHz Modems required Fresnel Zone diameter	2.4 GHz Modems required Fresnel Zone diameter
1000 ft. (300 m)	16 ft. (5 m)	11 ft. (3 m)
1 Mile (1.6 km)	32 ft. (10 m)	21 ft. (6 m)
5 Miles (8 km)	68 ft. (21 m)	43 ft. (13 m)
10 Miles (16 km)	95 ft. (29 m)	59 ft. (18 m)

MULTIPATH INTERFERENCE

- Obstacles not only reduce the signal if directly in the path, but they can also create interference if outside the direct path
- Interferences reduce the quality of the signal
- If the quality is lower, it is harder for the receiver to “hear”



CONDUCTED POWER VS. EIRP



- Conducted: The TX power of the RF module

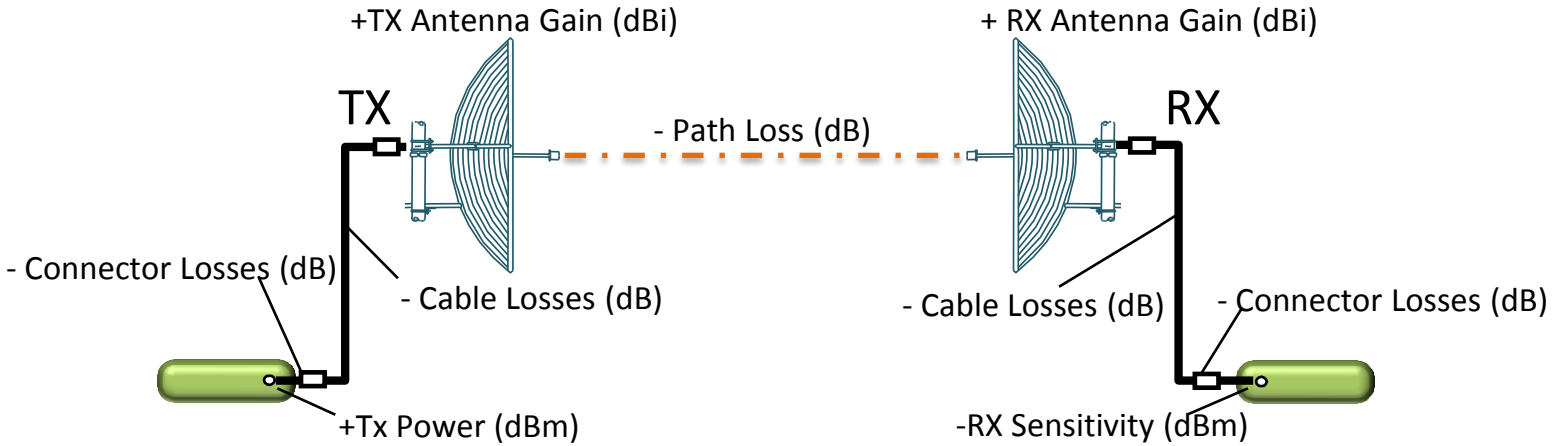
= Conducted

- EIRP: EIRP (Effective Isotropic Radiated Power) is the Conducted power of the radio + antenna – cable loss

–  +  = EIRP

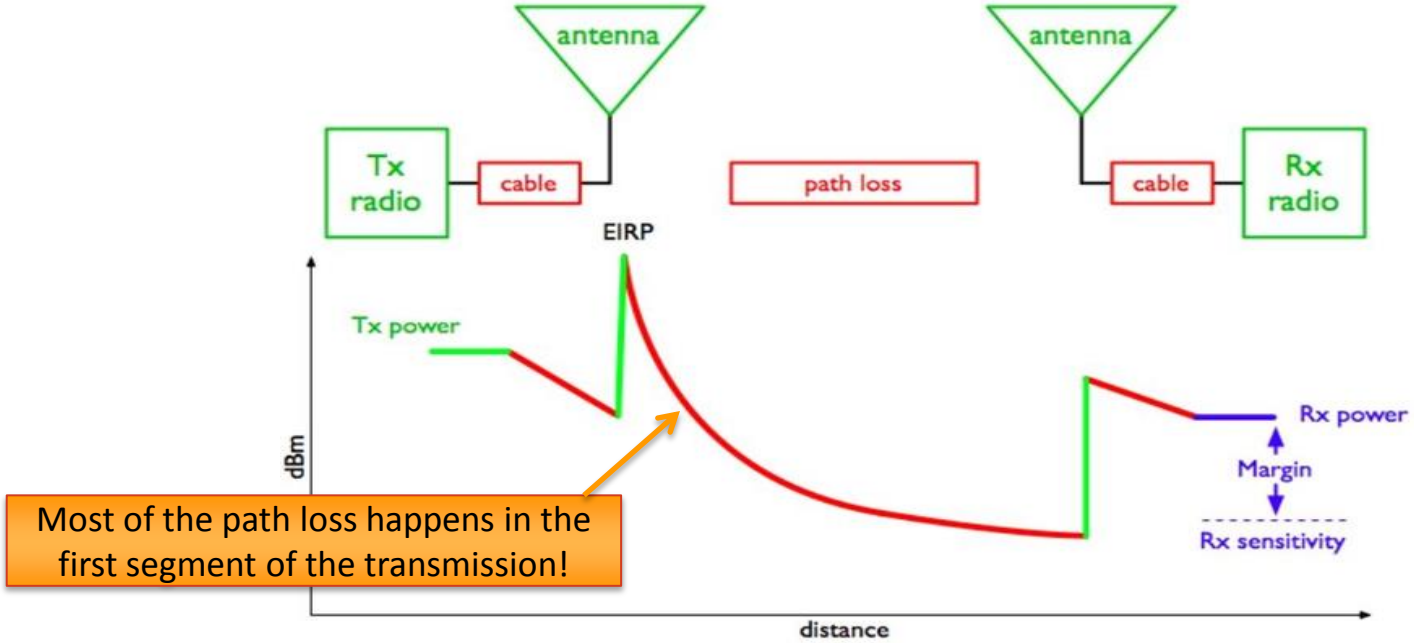
LINK BUDGET

The Link Budget is a way of quantifying the link performance, and determine the signal power



Link Budget (dB) = *TX Power + TX Antenna Gain - Path Loss + RX Antenna Gain - Total Connector and Cable Losses*

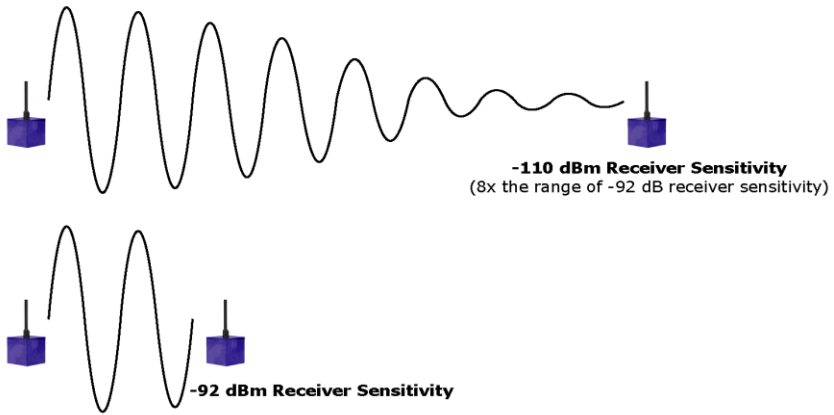
LINK BUDGET IN GRAPHICS



The difference between the Link budget and the RX sensitivity is called the **Link Margin**. If the Link Margin is negative, there is no communication!!!!

RX SENSITIVITY – RANGE ADVANTAGE

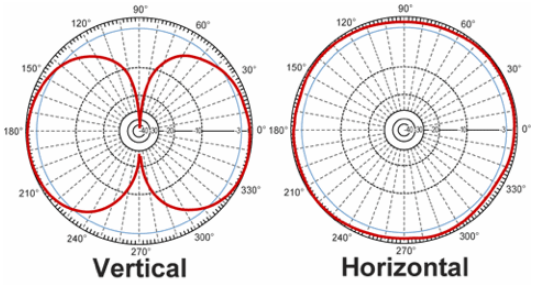
- Higher Receiver Sensitivity allows greater transmission range with the same Transmit Power
- This is because most of the RF signal is lost in the initial part



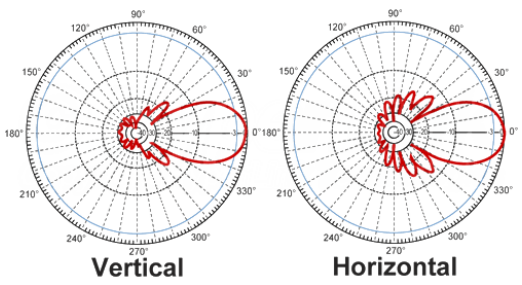
ANTENNAS

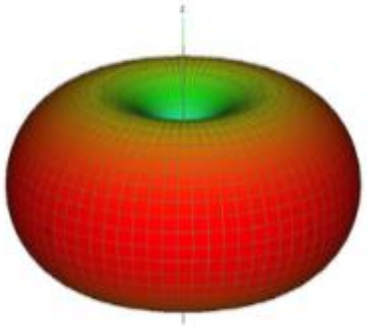
- An antenna is essentially a transducer that converts electrical currents into electromagnetic waves and vice versa
- The design of the transmitter's antenna determines the shape of the electromagnetic field (radio wave) delivered or propagated (typically doughnut-shaped)
- The field delivered from an antenna extends into the space surrounding it and its strength diminishes as the distance increases

Omnidirectional



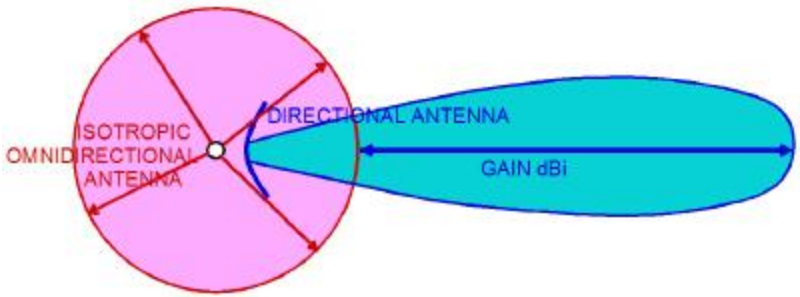
Directional

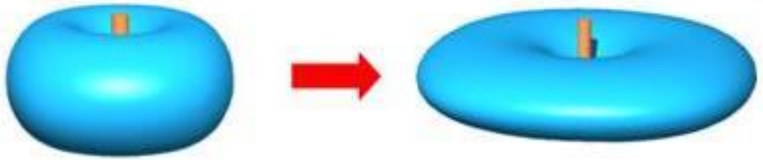




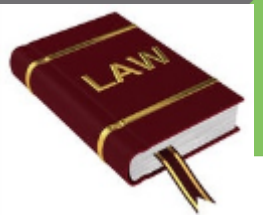
ANTENNA GAIN

- Omni-directional and directional antennas **FOCUS** energy
- **They DO NOT ADD energy!**
- Due to reciprocity, these two effects are equal
- So an antenna transmitting a 3x stronger signal, will also capture 3 times as much energy

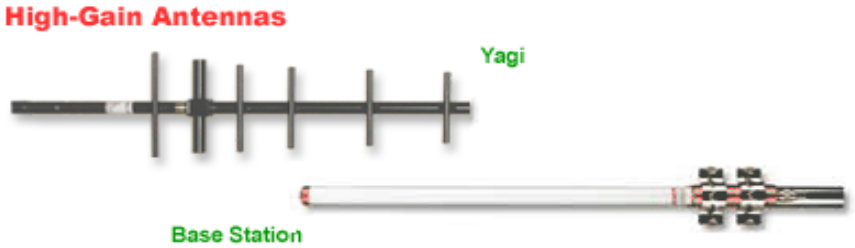




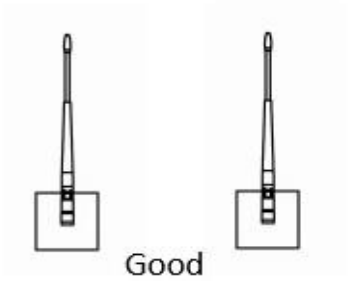
ANTENNA GAIN RESTRICTIONS

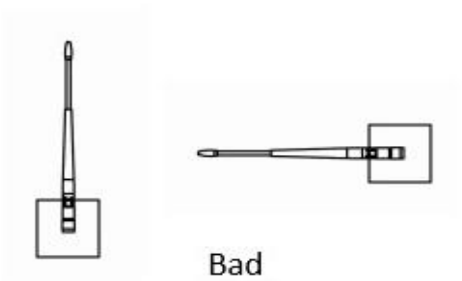


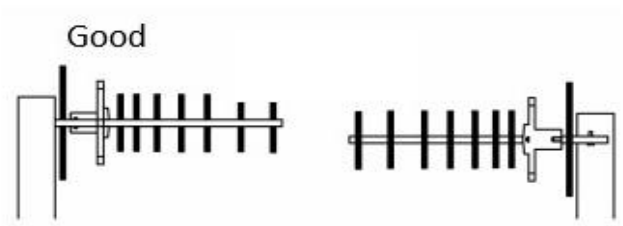
- The power restrictions are Government-imposed
- In the United States, FCC regulates the use of antennas via FCC Part 15.247
- The Equivalent Isotropic Radiated Power (EIRP) represents the total effective transmit power of the radio, including gains from the antenna and subtracting losses from cables, connectors, etc.
- FCC defines the maximum power of an ISM 2.4 GHz radio to be 1 watt (or 30 dBm) for conducted power and a maximum of 4 Watts (or 36 dBm) for EIRP power



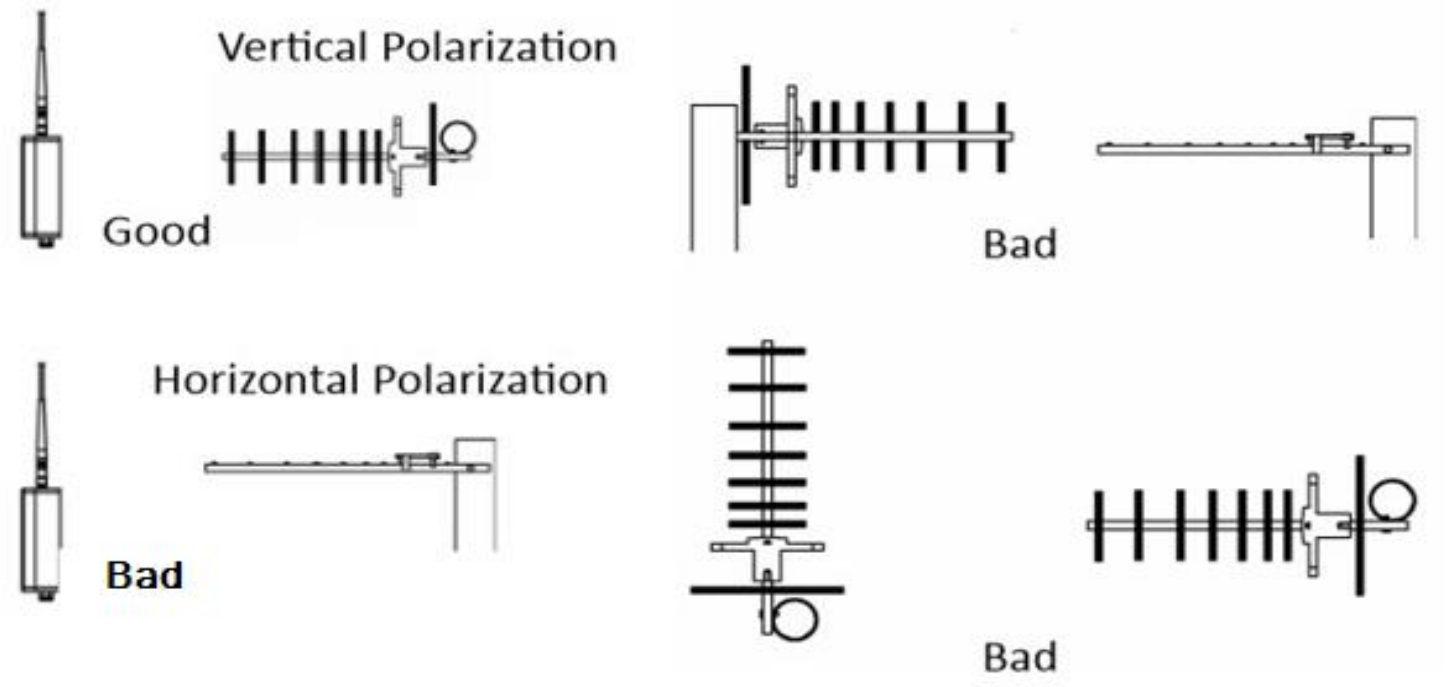
ANTENNA ORIENTATION







ANTENNA ORIENTATION



REGULATORY BODIES

__BEFORE__ using an RF device in the field, you must obtain an approval from the Regulatory Body, of the country of installation, such as:

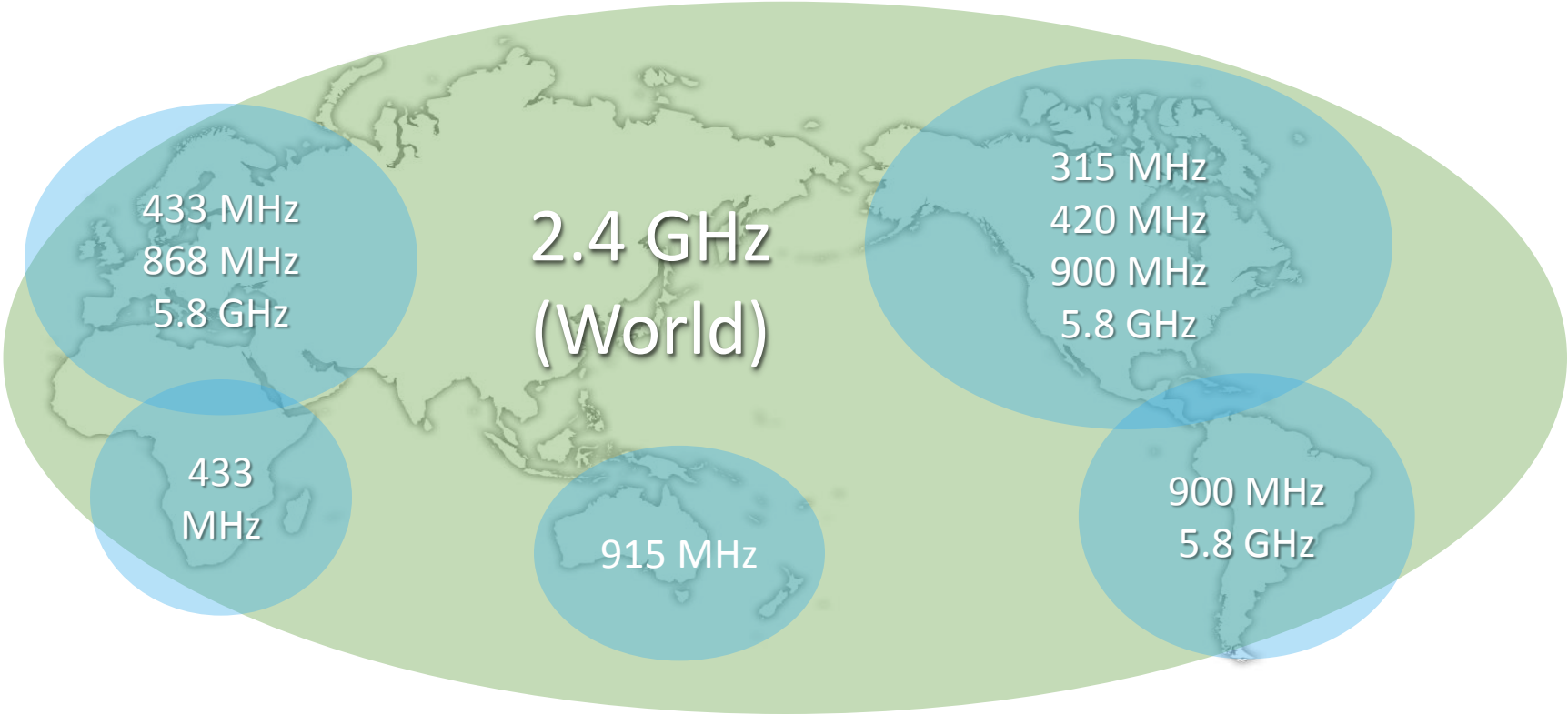
- Anatel (Brazil)
- C-Tick (Australia)
- COFETEL/IFETEL (Mexico)
- ETSI (Europe, some APAC)
- FCC (United States)
- IC (Canada)
- MTC (Peru)
- SubTel (Chile)
- Telec (Japan)

ISM FREQUENCIES – WHY?

- The Industrial, Scientific and Medical radio bands were originally reserved internationally for the use of RF energy for purposes other than communications
- The ISM bands are defined by the ITU-R in 5.138, 5.150, and 5.280 of the Radio Regulations
- They have no licensing cost! ☺

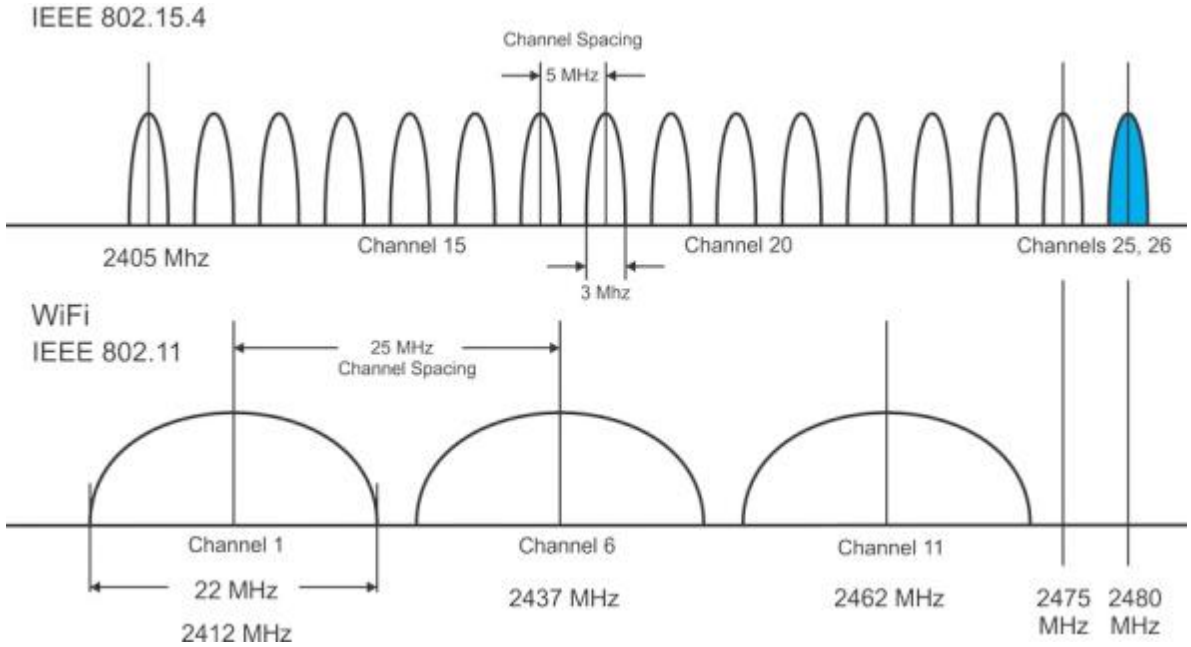
Center freq.	Availability
13.560 MHz	RFID
27.120 MHz	CB Radio Models
40.68 MHz	Radio Models
433 MHz	EU only and subject to local acceptance
868 MHz	EU only
915 MHz	ISM Americas only
2.450 GHz	ISM Worldwide
5.8 GHz	ISM Worldwide

LICENSE-FREE (ISM) BANDS



WiFi AND 802.15.4 COEXISTENCE

- It is possible to have both radios operating without interference
- Pick 802.15.4 channels that are not overlapping (15, 20, 25, 26)



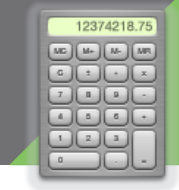
RADIO MODEM OPERATIONS

To increase the immunity to noise, use Spread Spectrum!
Two types of spread spectrum

- **FHSS - Frequency Hopping Spread Spectrum**
  - ✓ Transmitter rapidly hops from one channel to the next in a pseudo-random fashion, avoiding long-term interferences
  - ✓ Receiver follows transmitter
  - ✓ *Digi products: XTend, XBee 900HP*
  - ✓ Generally better in noisy machine environments

- **DSSS - Direct Sequence Spread Spectrum**
  - ✓ Each bit is represented by N, shorter segments, called chips
  - ✓ Increases over-the-air rate by a factor of N, widening the spectrum
  - ✓ Correlator inside receiver examines the chips and maps chips back to bits, while simultaneously spreading undesired signals
  - ✓ Digi products: XBee/XBeePRO 802.15.4 and ZigBee
  - ✓ Generally better in RF-rich environments

DBM TO mW CONVERSION



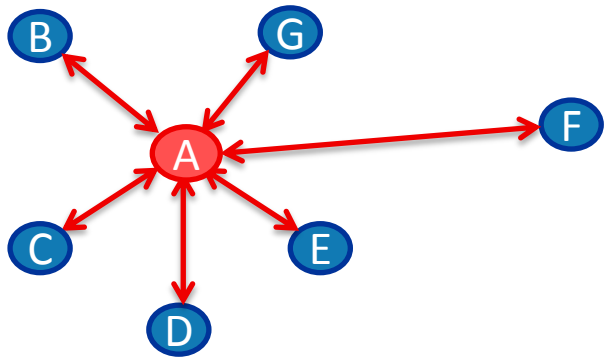
dBm	Watts	dBm	Watts	dBm	Watts
0	1.0 mW	16	40 mW	32	1.6 W
1	1.3 mW	17	50 mW	33	2.0 W
2	1.6 mW	18	63 mW	34	2.5 W
3	2.0 mW	19	79 mW	35	3.2 W
4	2.5 mW	20	100 mW	36	4.0 W
5	3.2 mW	21	126 mW	37	5.0 W
6	4 mW	22	158 mW	38	6.3 W
7	5 mW	23	200 mW	39	8.0 W
8	6 mW	24	250 mW	40	10 W
9	8 mW	25	316 mW	41	13 W
10	10 mW	26	398 mW	42	16 W
11	13 mW	27	500 mW	43	20 W
12	16 mW	28	630 mW	44	25 W
13	20 mW	29	800 mW	45	32 W
14	25 mW	30	1.0 W	46	40 W
15	32 mW	31	1.3 W	47	50 W

DIGI

WIRELESS ARCHITECTURES AND PROTOCOLS

ARCHITECTURE: P2MP (STAR) NETWORK

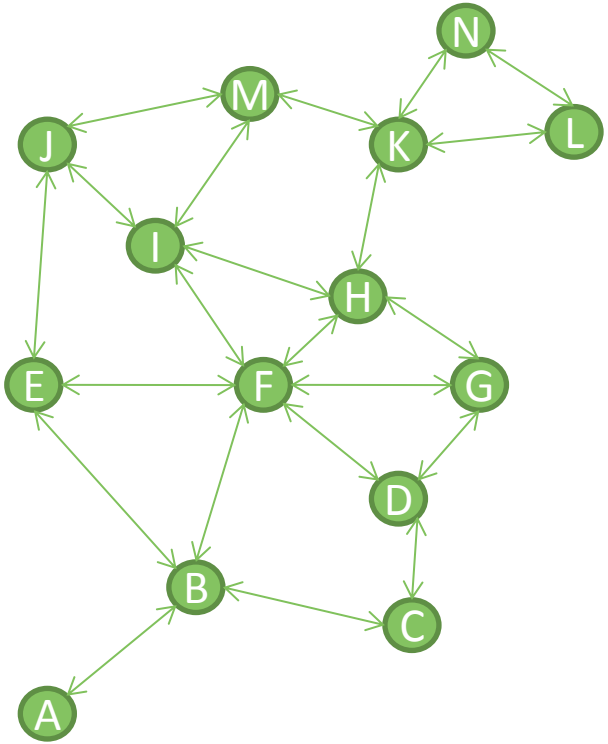
- Radios can only communicate with nodes which are in immediate RF range
- Cannot cover at great distances as mesh networks
- Generally faster network performance than a mesh network



ARCHITECTURE: MESH NETWORK

Any radio can communicate with any other radio in the network

If a destination radio is not in immediate range, the message can be forwarded by intermediate nodes

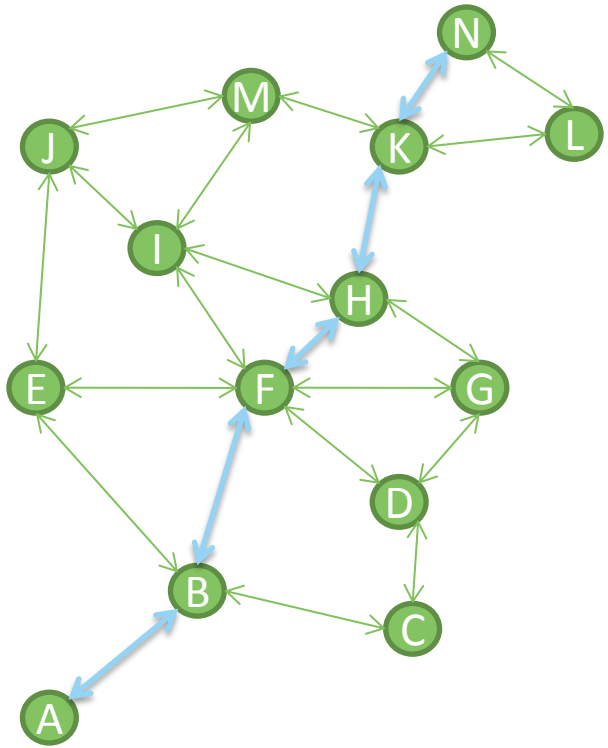


ARCHITECTURE: MESH NETWORK

Messages are automatically routed to their destination

Routes are discovered as needed

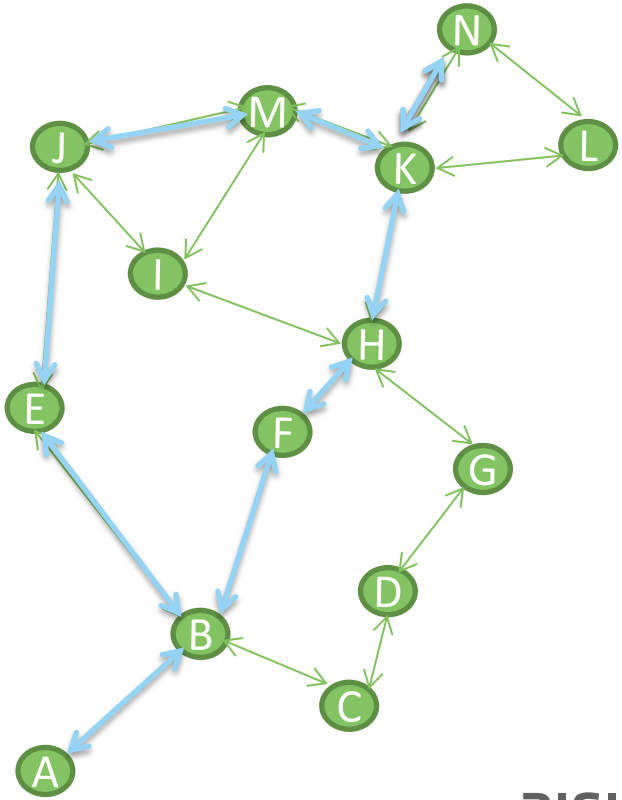
Redundant routes improve reliability



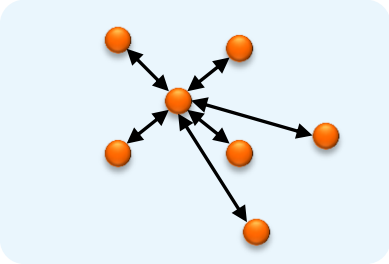
ARCHITECTURE: MESH NETWORK

If a route becomes unusable, a new one can be discovered (Route Discovery)

Generally have slower performance than star networks

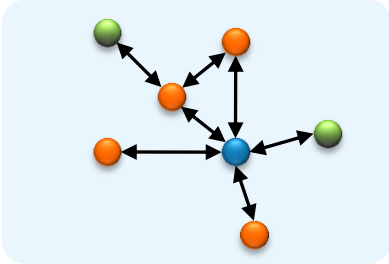


NETWORK TOPOLOGY



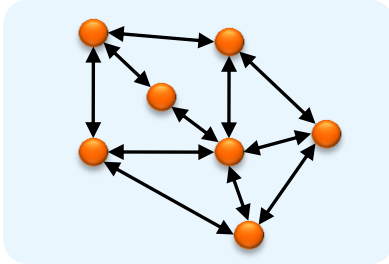
Multipoint

- No Mesh
- Fast Network
- One Node Type
- Sleeping Nodes (some exceptions)
- Simple Setup
- 900 MHz and 2.4 GHz
- Some Interoperability



ZigBee

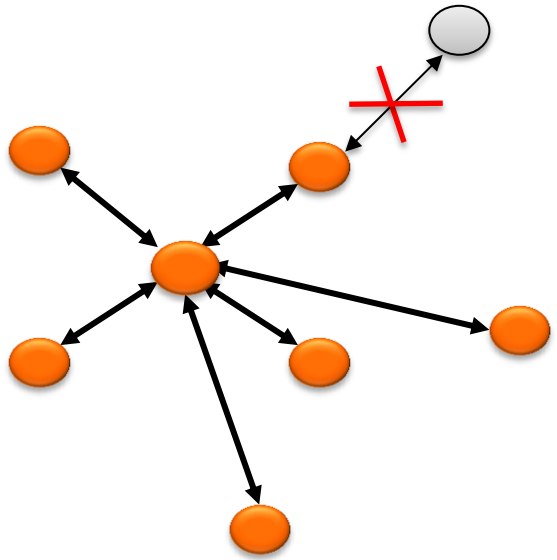
- Mesh
- Slower Network
- Multiple Node Types
- No Sleeping Routers
- Complex Setup
- 2.4 GHz only
- Interoperability
- Open Standard



DigiMesh

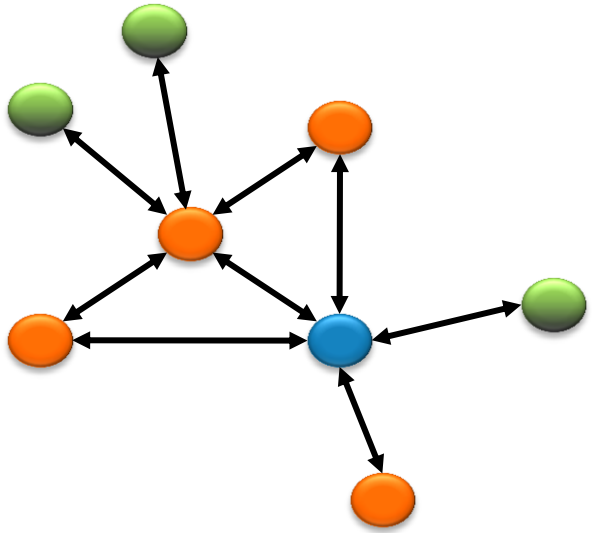
- Mesh
- Average Network
- One Node Type
- Support for Sleeping Routers
- Simple Setup
- 900 MHz and 2.4 GHz
- Proprietary Protocol (Customizable)

PROTOCOL: IEEE 802.15.4



- Requires direct communication between origin and destination
- Faster, lower latency
- Minimal design complexity, simple
- No hops, limited range
- Basis for mesh networks
- No ability to forward messages
- Multiple frequency bands available

PROTOCOL: ZIGBEE

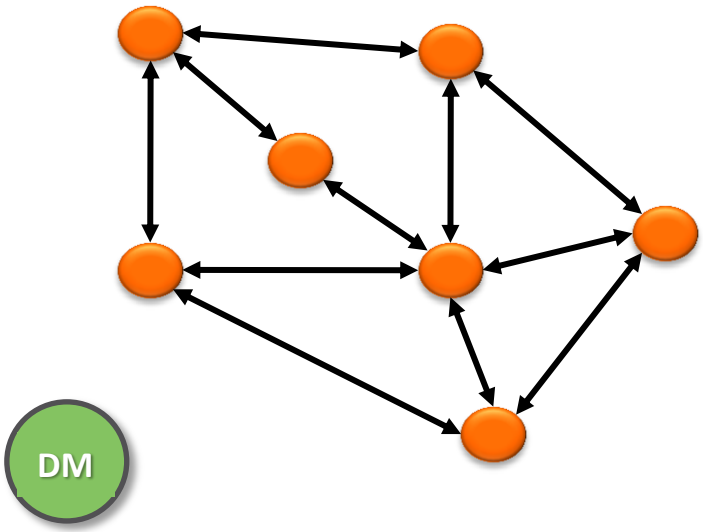


- Managed by the ZigBee Alliance
- Built on top of the 802.15.4 MAC/PHY
- Support for mesh networking
- Multiple node types
- Interoperability across vendors
- Automatic route discovery
- Self-healing
- Sleeping end devices (routers and coordinator cannot sleep)
- Limited to 2.4 GHz

ZigBee® Alliance

DIGI

PROTOCOL: DIGIMESH



- Digi proprietary
- Built on top of the 802.15.4 MAC/PHY
- Support for mesh networking
- Single node type
- Minimal design complexity, simple
- No parent-child relationships
- Automatic route discovery
- Self-healing
- All nodes can sleep
- Great for battery operations
- Available for multiple frequencies

ZigBee vs. DigiMesh:
http://www.digi.com/pdf/wp_zigbeevsdigimesh.pdf

ZIGBEE vs. DIGIMESH

COMPARISON

ZIGBEE NODES

Coordinator

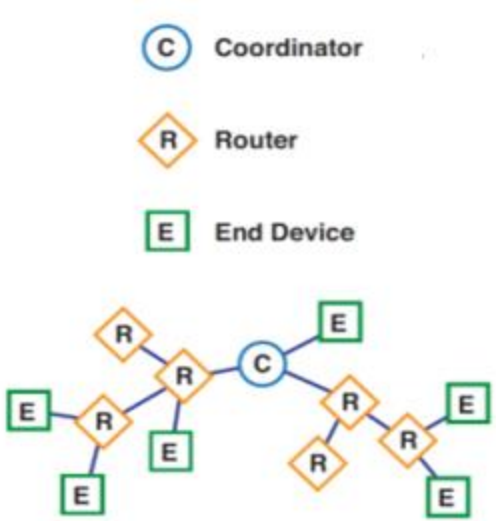
- Forms a network Personal Area Network (PAN) on a channel and PAN ID
- Necessary to initially form a network!
- Allows routers and end devices to join the PAN
- Trust center for security
- Cannot sleep!

Router

- Joins a network
- Routes data through the network
- Allows other routers and end-devices to join
- It has to be always awake or the branch could lose connectivity

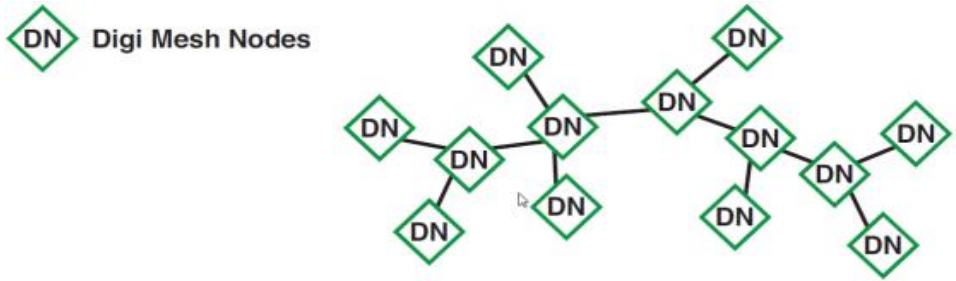
End device

- Joins a network
- Cannot route data, needs a parent to do so
- Supports low-power/sleep modes, cannot be always awake!



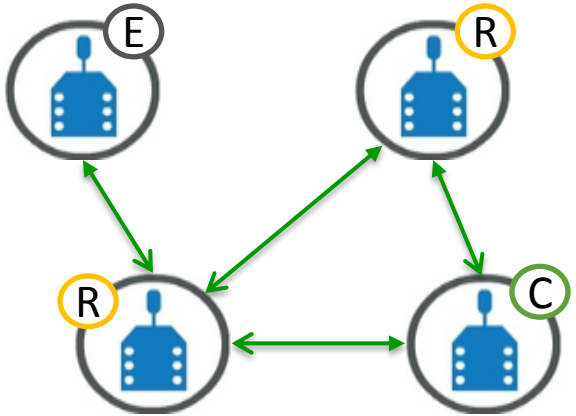
DIGIMESH NODES

- DigiMesh has only one node type of nodes!
- As a homogenous network, all nodes can route data and are interchangeable
- There are no parent-child relationships
- All nodes can be configured as sleep / battery-powered devices
- There is no 16-bit addressing (simpler) – only 64-bit



STARTING A ZIGBEE NETWORK

- Coordinator selects the Channel in within the mask
- Then it selects a (random) 16-bit PAN ID
- Routers and end devices join the network
- Routers can be added to extend network coverage



STARTING AN XBEE ZB NETWORK

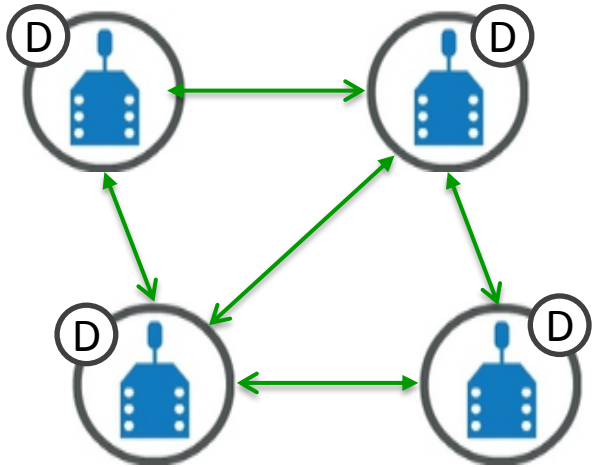
Notes:

- Set SC command (bitmask – channels to scan)
- Set ID command (extended PAN ID to select)
- By default, Routers and End Device scan the channels allowed by the SC bitmask and join any PAN ID
- Ensure a coordinator (or joined router) is running on a valid PAN ID and on a channel included in the SC bitmask
- Read the actual operating PAN ID and operating channel using the OI and CH commands
- Join failures can be diagnosed using the AI command

STARTING A DIGIMESH NETWORK

DM

- All node have to be pre-configured with the same Preamble ID (HP), Channel Mask (CM) and Network ID (ID)
- Nodes can be added later to extend network coverage
- Selectively, routing can be enabled or disabled on individual nodes (CE)
- Could use the Preamble ID to split networks in nearby locations



ADDRESSING NODES

- ZigBee devices support both 64-bit and 16-bit
- A node can be identified with either one
- DigiMesh only supports 64-bit addressing

Feature	ZigBee	DigiMesh
Supports 64-bit Address • Unique per device, the MAC address • Fixed – best way to absolutely address a device	✓	✓
Supports 16-bit Address • Assigned to devices when joining • Used in routing tables • Volatile – can change under certain conditions	✓	
Supports Broadcast Addressing (Note: slow with ZigBee)	✓	✓

DM SYNCHRONOUS SLEEP

- **AKA: Sleeping Routers**
- Every node can route data
- All nodes **synchronize** with each other
- **All nodes can sleep**
- All sleeping nodes share the same sleep timing
- Only available with **Digimesh**!

SECURITY

ZigBee

Two 128-bit AES Encryption methods:

1. Network Key – pre-shared by all devices
   Can be selected from a list but does not change
2. APS Key – shared only by the origin and destination
   Can be different per each message

Keys can be pre-loaded or establish (random)

Joining can also be disabled (NJ) for added security!

DigiMesh

Simpler, one Pre-Shared 128-bit AES encryption key

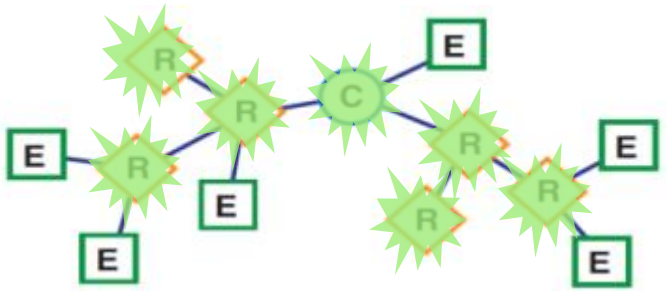
PROTOCOL BACKGROUND “CHATTING”

Noisy!

C Coordinator

R Router

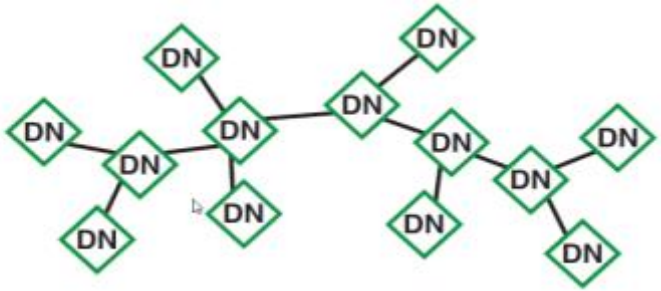
E End Device



Quiet

DM

DN Digi Mesh Nodes




LET'S TAKE
-A-
Coffee
BREAK

ZIGBEE vs. DIGIMESH RECAP

ZigBee

- Needs a sequence to get started
- Multiple node types
- More complex deployment
- 2.4 GHz only
- Shorter maximum range
- Interoperable with others
- Basic sleep modes available
- Only end-devices can sleep
- Very low sleep currents
- By default, nodes send data to the Coordinator

DigiMesh

- Very easy to get started
- Single node type
- Simple deployment strategy
- Multiple frequencies
- Long range options
- Proprietary
- Basic sleep modes available
- Sleeping router support
- Low sleep currents
- By default, nodes send data in broadcast mode

WHICH TECHNOLOGY IS RIGHT FOR ME?

ZigBee offers these advantages:

- Open standard with interoperability between vendors
- Third party integration (e.g. Home Automation or Smart Energy)
- Enhanced security is required

DigiMesh offers these advantages:

- Network setup is simpler
- Multiple frequencies available
- More flexibility to expand the network – No parent/child ratios to worry
- Quieter network allows larger packets and higher throughput
- Broadcast intensive programs such as Modbus

DIGI XBEE MODULES

THE NEW XBEE LOGO

SOMETIMES A LITTLE CHANGE HELPS…

New Logo, in line with the new Digi marketing

WHAT IS THE XBEE?

- XBee are small RF (radio frequency) mo
- Can transmit and receive data
- With the XBee, you c be installed, or to avoid the inst
- XBe ultiple protocols

odule in the World!!!

Guess how many XBee were sold in 2016???

Product page:
http://www.digi.com/lp/xbee

MAIN FEATURES OF THE XBEE

Serial-to-RF OEM module

Complete protocol loaded in FW

Application layer for easy operations (AT)

Full application protocol layer for complex operations (API)

Powerful feature set includes:

- Remote configuration
- OTA FW updates
- Network diagnostics
- Many I/O options

Low power / long battery life

Various Antenna Options: Whip, PCB, U.FL, RP-SMA

Complete solution, HW+FW - No design needed!

Fully certified / approved !!!

XBEE I/O

- 4-wire UART data interface (**LVTTL 3.3V, NO 5V!!!!**)
- 8+1 Digital I/O lines (4 muxed with ADC)
- 1 PWM (RSSI) **output**
- 4 ADC **inputs**, 10 bit resolution, 1KHz sample rate
- For analog sampling, VREF must be connected
- Digital I/O can bit-bang at 50Hz
- Supply voltage **monitoring** on some models
- PWM **output** on some models
- XBee modules have internal pull-up/pull-down (29-50 kOhm)

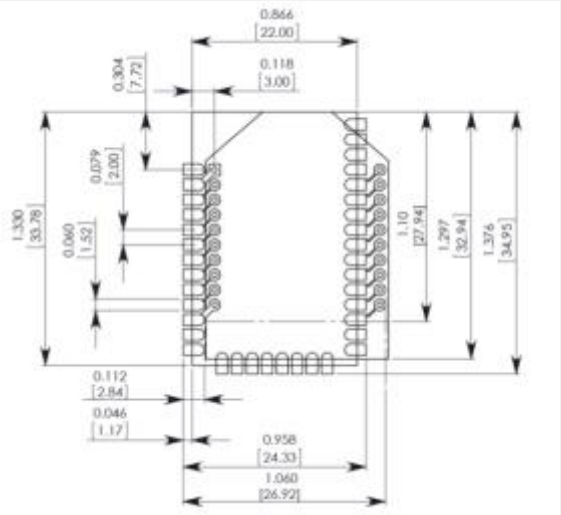
XBEE FORM FACTORS

THROUGH-HOLE

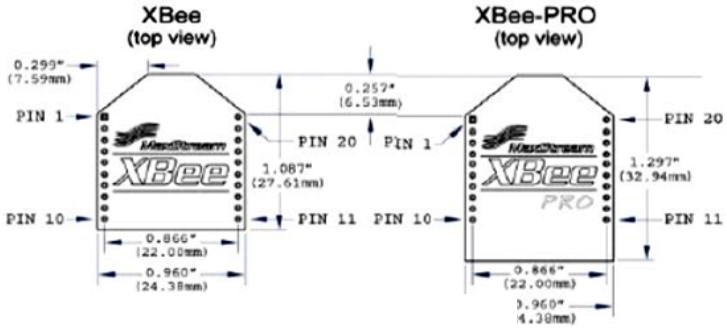
SURFACE-MOUNT

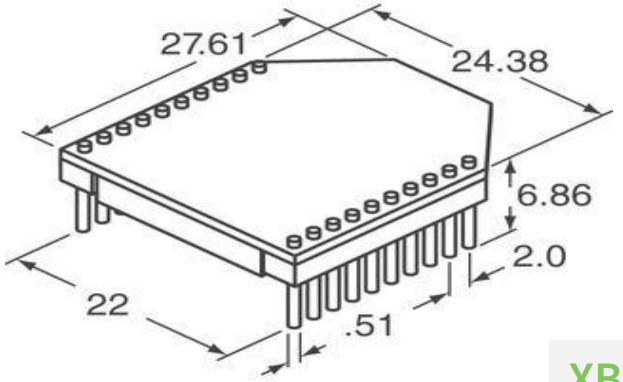
Through-hole: 20 pins
SMD:37 pins
TH PRO is slightly longer than Standard
GPIO, ADC, PWM
I/O is at 3.3v!!!
UART and SPI available

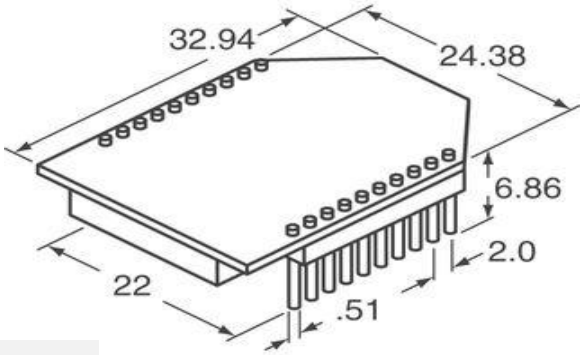
Both form factors can share a common footprint (slightly shifted)
Minimal redesign for existing through-hole XBee Customers
Main board can be designed for both



MECHANICAL DIMENSIONS (TH)

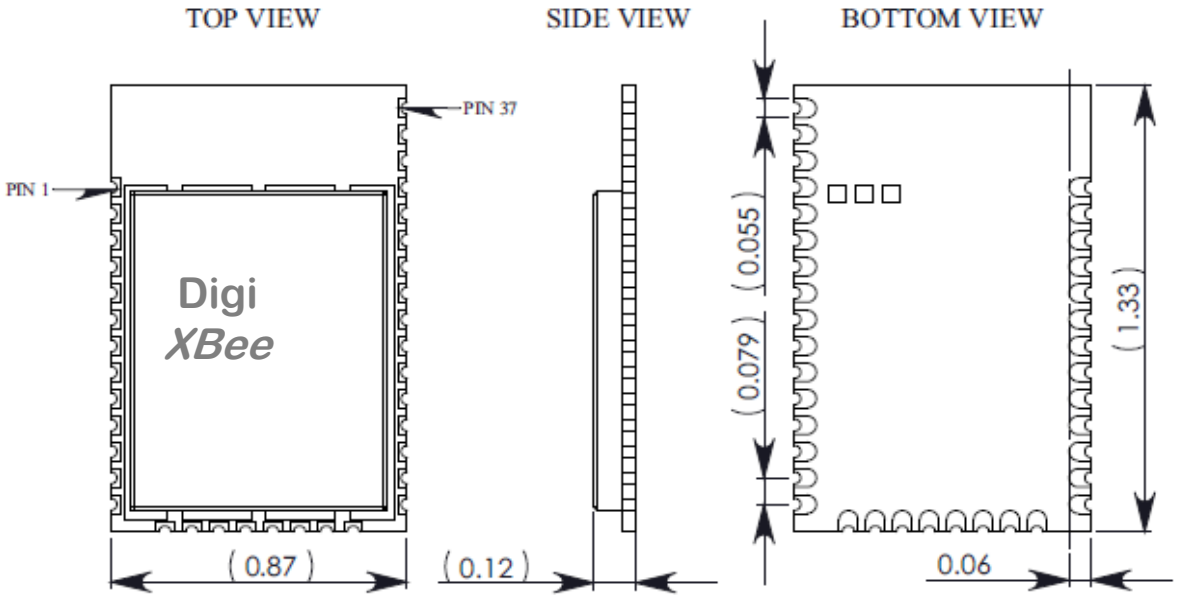






XBEE THROUGH HOLE

MECHANICAL DIMENSIONS (SMT)



XBee SMD

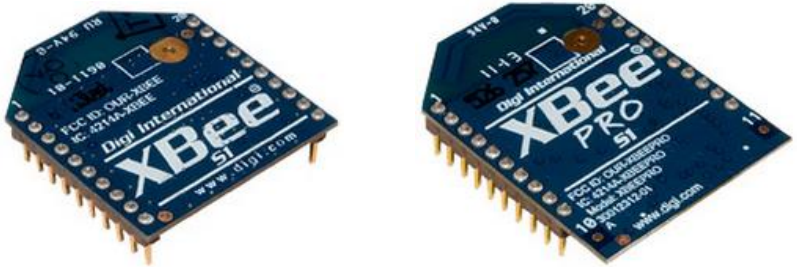
XBEE vs. XBEE-PRO

You can mix and match them in the same network!

- Lower Power
- Less Range
- Lower Priced
- Smaller Form Factor (TH only)

- Higher Power
- More Range
- Higher Priced
- Longer Form Factor (TH only)

**** SMT Non-PRO and PRO modules are the same size***



XBEE MODELS

XBEE 802.15.4

PROBLEM SOLVED

- Simplest, fast connectivity, point to multipoint, no mesh

PRODUCT DESCRIPTION

- Simple, out-of-the-box RF communications, no configuration needed
- Point-to-multipoint network topology
- 2.4 GHz globally deployable frequency
- Common XBee footprint for a variety of RF modules
- Industry leading sleep current of sub 1uA
- Highest bandwidth, lowest latency
- Firmware upgrades via UART, SPI or over the air
- Can migrate to DigiMesh and ZigBee protocols

XBEE DIGIMESH 2.4

PROBLEM SOLVED

- Simple, fast connectivity with mesh

PRODUCT DESCRIPTION

- Dynamic peer-to-peer DigiMesh protocol
- No configuration needed for out-of-the-box RF communications
- 2.4 GHz globally deployable frequency
- Advanced sleep modes for battery operations
- Higher bandwidth

XBEE ZIGBEE

PROBLEM SOLVED

- Industry standard, 3rd party connectivity with mesh

PRODUCT DESCRIPTION

- Through-Hole and Surface Mount form factors enable flexible design options
- High link budgets of 110 dB for Xbee and 119 dB for XBee-PRO
- LOS range of up to 2 miles
- 2.4 GHz globally deployable frequency
- Firmware upgrades via UART, SPI or over the air (OTA)
- Programmable versions available!

XBEE 900HP

PROBLEM SOLVED

- Long range connectivity with mesh

PRODUCT DESCRIPTION

- Dynamic peer-to-peer DigiMesh protocol
- LOS range of up to 28 miles with high-gain antenna
- Over-the-air firmware updates
- Software-selectable channel mask for interference immunity
- 900 MHz frequency
- Advanced sleep modes for battery operations

XBEE SX

NEW

PROBLEM SOLVED

- High power, very long range connectivity with mesh

PRODUCT DESCRIPTION

- Two models: SX-PRO, powerful 1-Watt for long range, and SX, 20 mW for lower consumption
- LOS range of up to 65 miles with high-gain antenna
- DigiMesh networking topology for redundancy and reliability, with self-healing and discovery
- 256-bit AES encryption for secure data communications
- XBee SMT form factor saves valuable PCB space
- Fully certified for use in unlicensed 900 MHz band

XBEE WiFi 802.11

PROBLEM SOLVED

- WLAN module for deployment in existing infrastructures

PRODUCT DESCRIPTION

- Build cloud-connected Wi-Fi prototypes in under an hour
- Standard 20-pin XBee TH and Surface-Mount footprints
- Ideal for Industrial Applications
- Easily connect to a smartphone or tablet for configuration or data transfer
- Fast 802.11 b/g/n provides up to 72 Mbps data rate

XBEE CELLULAR

NEW

PROBLEM SOLVED

- Cellular connectivity, no infrastructure required

PRODUCT DESCRIPTION

- Available for 4G LTE now (US), and 3G (World) in JUNE2017
- Convenient Digi data plans available for Cellular
- FCC certified and Carrier End-device certified
- Standard 20-pin XBee TH form factor
- Smallest end-device certified cellular modem
- XBee Transparent and API modes simplify SW design
- Low-power options for battery powered applications
- OTA firmware updates
- Security features including SSL/TLS v1.2 and others

XBEE CELLULAR MODELS

6 MONTHS OF FREE SERVICE

FIND THE PERFECT XBEE SOLUTION!

	Protocol	Frequency	Form Factor	Topology	Output Power	Range (Ideal LoS)
Digi XBee® ZigBee / Thread-Ready	ZigBee	2.4 GHz		Mesh	Up to 63mW / 18dBm	Up to 2 miles (3.2km)
Digi XBee® 802.15.4	802.15.4	2.4 GHz		Point to Multipoint	Up to 63mW / 18dBm	Up to 2 miles (3.2km)
Digi XBee® DigiMesh® 2.4	DigiMesh	2.4 GHz		Mesh	Up to 63mW / 18dBm	Up to 2 miles (3.2km)
Digi XBee® Wi-Fi	802.11 bgn	2.4 GHz		Point to Multipoint	Up to 40mW / 16dBm	Up to 300 feet
Digi XBee® Cellular	LTE Cat 1	Bands 4, 13		Point to Multipoint	Up to 200mW / 23dBm	*Varies based on tower location
Digi XBee® SX	DigiMesh	900 MHz		Mesh, Point to Multipoint	Up to 1W / 30dBm	Up to 65 miles
Digi XBee-PRO® 900HP	Proprietary	900 MHz		Mesh, Point to Multipoint	Up to 250mW / 24dBm	Up to 9 miles
Digi XBee® 868LP	DigiMesh	868 MHz		Mesh, Point to Multipoint	Up to 25mW / 14dBm	Up to 5 miles

XBEE / RF FAMILY COMPARISON

Family	Frequency	Protocol	Description	RF Line of Sight Range	Form Factor	Development Kit Part Numbers	RF Data Rate	Current Draw Tx/Rx	Hardware Reference # / Chipset(s)	Certified Regions
XBee® Cellular	Bands 4 and 13	LTE Cat 1	Pre-certified for LTE Cat 1, in an XBee form factor	Cellular Network Coverage	Through-Hole	XKC-V1T-U	10 Mbps Down / 5 Mbps up	860mA / 530mA	SiLabs EFM32GG395F1024 ARM M3 MCU	US
XBee® Wi-Fi	2.4 GHz	IEEE 802.11	Wi-Fi 802.11b/g/n with easy provisioning and point-to-multipoint device connectivity	N/A	Through-Hole	XKA2B-WFT-0	1 to 72 Mbps	309 mA / 100 mA	$4B SiLabs EFM32LG230 ARM M3 MCU, Atheros AR4100 Transceiver	US, CA, EU, AU, JP
XBee® DigiMesh® 2.4		DigiMesh®	DigiMesh networking, low-cost, low-power	4000 ft (1200 m)		XK-WDM	250 Kbps	33mA / 28mA	$3C SiLabs EM357 SoC	US, CA, EU, AUS/ NZ, BR, JP
XBee-PRO DigiMesh® 2.4			Extended-range DigiMesh	2 miles (3200 m)				120 mA / 31 mA		US, CA, AU, NZ, BR
XBee® 802.15.4		Proprietary 802.15.4	low-cost, low power point-to-multipoint device connectivity	4000 ft (1.2 km)	Surface Mount	XKB2-A2T-WWC		33mA / 28mA		US, CA, EU, AUS/ NZ, BR, JP
XBee-PRO® 802.15.4			Point to multipoint extended range version	2 miles (3.2 km)				120mA / 31 mA		US, CA, AU, NZ, BR
XBee® ZigBee		ZigBee® Pro	ZigBee mesh networking, low-cost, low-power	4000 ft / 1.2 km		XKB2-Z7T-WZM		33mA / 28mA		US, CA, EU, AUS/ NZ, BR, JP
XBee-PRO® ZigBee			Extended-range ZigBee	2 miles / 3.2 km		XKA2C-Z7T-U		120 mA / 31 mA		US, CA, AU, NZ, BR
XBee® ZigBee - Thread Ready		ZigBee® Pro Thread	ZigBee protocol (upgradable to Thread protocol) low cost, low power	4000 ft (1.2km)	Surface Mount	XKB2-Z7T-WTZM		33mA / 28mA	$3D SiLabs EM35R7 SoC	US, CA, EU
XBee-PRO® 900HP	900 MHz	Multipoint	Extended-range peer-to-peer mesh, sleeping routers	9 miles / 14.5 km	Through-Hole	XKB9-DMT-UHP (US/CA) XKB9-DMT-AHP (AU) XKB9-DMT-BHP (BR) XKB9-DMT-SHP (SGP)	10 Kbps or 200 Kbps	215 mA / 29 mA	$3B SiLabs EFM32GG390F128 ARM M3 MCU, Analog Devices ADF7023 Transceiver	US, CA, AU, BR
XBee® SX		DigiMesh®	20mW networking XBee module for mission critical applications	9 miles / 14 km	Surface Mount	XKBX-DMS-0	250 Kbps	55 mA / 40 mA	$1B SiLabs EFM32LG230F256G ARM M3 MCU, Analog Devices ADF7023 Transceiver, LNA/SAW (PRO version: PA+LNA/SAW)	US, CA, AU, NZ (BR Pending)
XBee-PRO® SX			1-Watt networking XBee module for mission critical applications	65 miles / 105 km				900 mA / 40 mA		US, CA, AU, (BR Pending)

Full table here: http://www.digi.com/pdf/chart_xbee_rf_features.pdf

CHOOSING THE RIGHT XBEE

Steps for choosing an XBee

1. Region: Country (or countries) of deployment
   - i.e. Frequency, Max Power?
2. Topology, Throughput?
   - i.e. How much data is being sent, how often, to who?
3. Protocol: Open Standard vs. Proprietary
   - i.e. Do you need to interface with other vendors or not?
4. Specs: Power consumption, Form factor, etc.
   - i.e. Do you need sleeping nodes? Battery Life? How much space?
5. Location: Range, Temperature, Obstacles
   - i.e. What are the requirements of your location?

MORE ABOUT THE XBEE

XBIB DEVELOPMENT BOARD

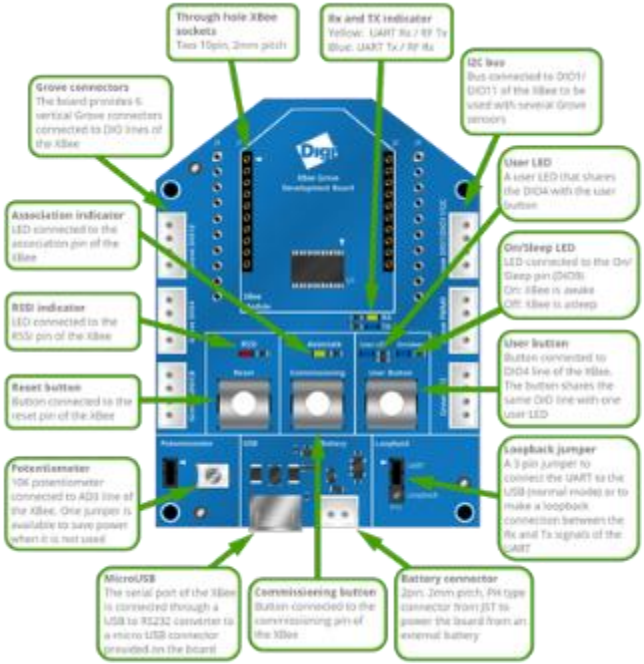
- The “traditional” XBee Development board is called XBIB
- Good for fast integration and connectivity
- There is a version for TH and a version for SMT
- Supports any XBee model
- Each model comes in two versions:
  - DB9 RS232 (requires power supply)
  - USB (self-powered)
- Connectors used on the XBIB:
  - Through-hole single-row receptacles
    Samtec P/N: MMS-110-01-L-SV
  - Through-hole single-row receptacles
    Mill-Max P/N: 831-43-0101-10-001000
  - Surface-mount double-row receptacles
    Century Interconnect P/N: CPRMSL20-D-0-1
  - Surface-mount single-row receptacles
    Samtec P/N: SMM-110-02-SM-S

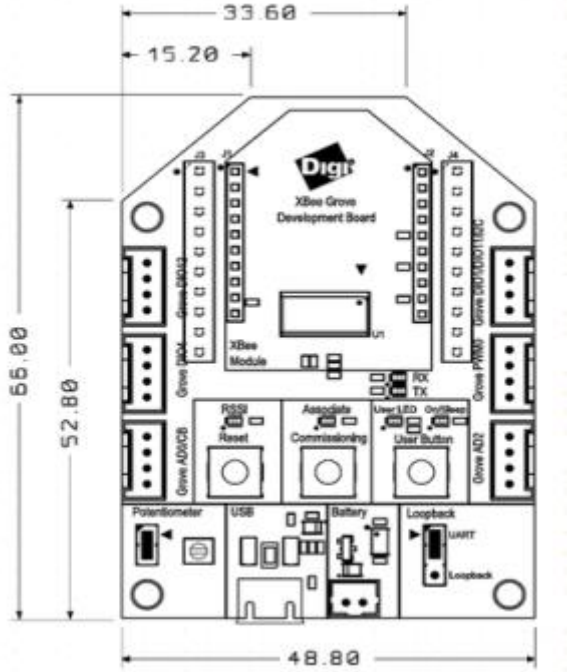
GROVE KIT

- New dev kit based on the Grove System
- Documentation:
  https://www.digi.com/resources/documentation/Digidocs/90001457-13/
- Compatible with ZigBee, DigiMesh, 802.15.4
- Allows plugging Grove extensions like I2C, analog or digital
- In addition, provides battery power, has a potentiometer, buttons, switches, LED, DC-DC transformer (3.3v and 5v)
- Powered via a standard USB cable or via battery
- Both versions: TH and SMT

Learn More on Seeed Studio:
http://www.seeed.cc/grove/
http://www.seeedstudio.com/document/pdf/Introduction%20to%20Grove.pdf

OVERVIEW OF THE GROVE KIT





XBEE DEVELOPMENT KITS

Name	XBee S2C DigiMesh Development Kit	ZigBee Mesh Kit	900HP Mesh Kit	Thread Ready S2D Kit
XBee Family	DigiMesh modules	ZigBee modules	DigiMesh & 900HP modules	Thread-ready modules
Overview	Intro to Mesh networking with DigiMesh	Intro to Mesh networking with ZigBee	Intro to DigiMesh and our 900HP modules	Intro to Thread with Mesh modules
Market	Worldwide	Worldwide	N. America, Australia, Brazil, Singapore	Worldwide
P/N	**XK-WDM**	**XKB2-Z7T-WZM**	**XKB9-DMT-xHP**	**XKB2-Z7T-WTZM**

XBEE DEVELOPMENT KITS

Name	Wireless S2C Connectivity Kit	Arduino Coding Kit	Digi XBee Cloud Kit
XBee Family	802.15.4 modules	802.15.4 modules	ZigBee modules
Overview	Basic Intro to 802.15.4	Intro to 802.15.4 with fun projects using Arduino-compatible HW/coding	Easy connectivity to Cloud
Market	Worldwide	Worldwide	Worldwide
P/N	**XKB2-A2T-WWC**	**XKB2-AT-WWC**	**XKA2C-Z7T-U**

CLOUD KIT

- Kit with a multi-functional Development Board for Cloud
- Documentation:
  http://ftp1.digi.com/support/documentation/html/90001399/90001399_A/Files/kit-getting-started.html
- Allows prototyping without additional external HW
- In addition, provides cloud connectivity via the XBee Gateway
- Dashboard via the Heroku website:
  https://xbeegateway.herokuapp.com/#/login
- Devboard has LED, buttons, switches, potentiometer, etc.
- Kit also includes spare components

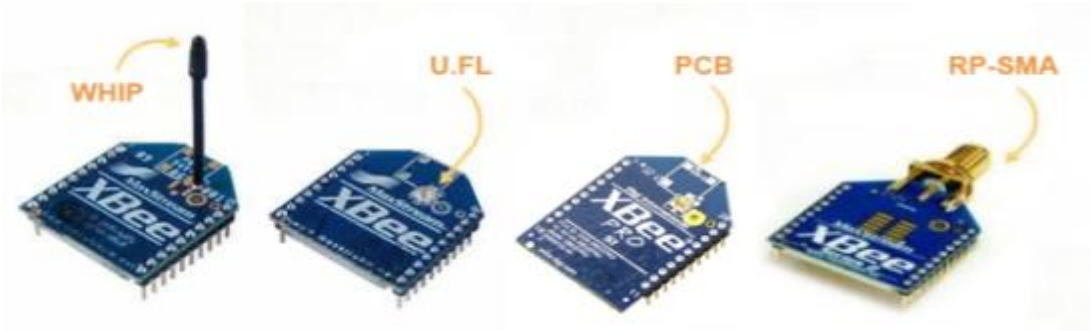
CONNECTIVITY OF THE CLOUD KIT

XBEE CLOUD KIT | XBEE GATEWAY | DEVICE CLOUD | HEROKU APP DASHBOARD

PART NUMBER	DESCRIPTION
XKA2C-Z7T-U	XBee ZigBee Cloud Kit – USA/Canada
XKA2C-Z7T-W	XBee ZigBee Cloud Kit – Worldwide

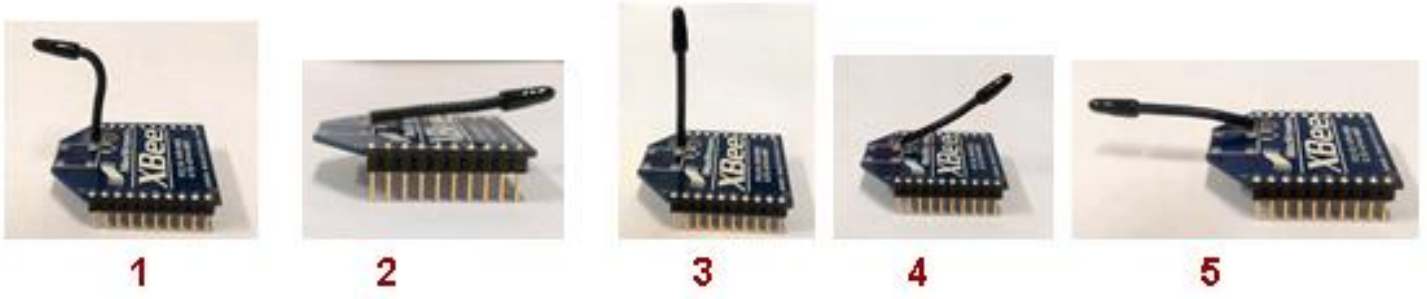
ANTENNA OPTIONS

- The XBee modules have different antenna options



- Variants of antennas that can be used are listed under the “Appendix B – Agency Certifications” of the Hardware Reference Manual Digi P/N 90000976

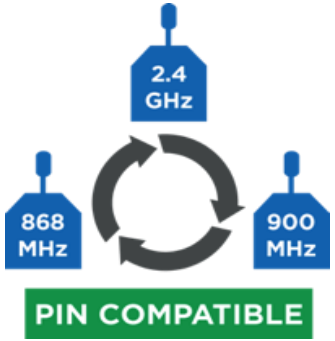
XBEE WIRE ORIENTATION

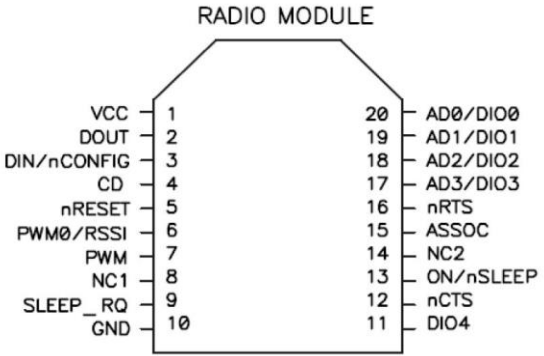


TEST Which is the optimal antenna position?

HW PINOUT

- The XBee modules have a common pinout
- That helps interchange and compatibility
- Not 100% of the functions are present in all the modules!





Learn More – White Papers:
http://www.digi.com/resources/white-papers

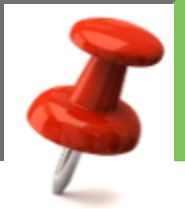
PIN ASSIGNMENTS – TH

Pin #	Name	Direction	Default State	Description
1	VCC	-	-	Power supply
2	DOUT	Output	Output	UART Data Out
3	DIN / **CONFIG**	Input	Input	UART Data In
4	DIO12	Both	Disabled	Digital I/O 12
5	**RESET**	Both	Open-Collector with pull-up	Module Reset (reset pulse must be at least 200 ns)
6	RSSI PWM / DIO10	Both	Output	RX Signal Strength Indicator / Digital IO
7	DIO11	Both	Input	Digital I/O 11
8	[reserved]	-	Disabled	Do not connect
9	DTR / SLEEP_RQ/ DIO8	Both	Input	Pin Sleep Control Line or Digital IO 8
10	GND	-	-	Ground
11	DIO4	Both	Disabled	Digital I/O 4
12	**CTS** / DIO7	Both	Output	Clear-to-Send Flow Control or Digital I/O 7. CTS, if enabled, is an output.
13	ON / **SLEEP**	Output	Output	Module Status Indicator or Digital I/O 9
14	VREF	Input	-	Not used for EM250. Used for programmable secondary processor. For compatibility with other XBEE modules, we recommend connecting this pin voltage reference if Analog sampling is desired. Otherwise, connect to GND.
15	Associate / DIO5	Both	Output	Associated Indicator, Digital I/O 5
16	**RTS** / DIO6	Both	Input	Request-to-Send Flow Control, Digital I/O 6. RTS, if enabled, is an input.
17	AD3 / DIO3	Both	Disabled	Analog Input 3 or Digital I/O 3
18	AD2 / DIO2	Both	Disabled	Analog Input 2 or Digital I/O 2
19	AD1 / DIO1	Both	Disabled	Analog Input 1 or Digital I/O 1
20	AD0 / DIO0 / Commissioning Button	Both	Disabled	Analog Input 0, Digital IO 0, or Commissioning Button

PIN ASSIGNMENTS – SMT

Pin #	Name	Direction	Default State	Description
1	GND	-	-	Ground
2	VCC	-	-	Power Supply
3	DIO13/DOUT	Both	Output	UART Data Out
4	DIO14/DIN/nCONFIG	Both	Input	UART Data In
5	DIO12	Both	Disabled	GPIO
6	nRESET	Input	Input	Module Reset
7	DIO10/PWM0	Both	Disabled	GPIO
8	DIO11/PWM1	Both	Disabled	GPIO
9	Reserved	-	-	Do Not Connect
10	DIO8/nDTR/SLEEP_RQ	Both	Input	GPIO
11	GND	-	-	Ground
12	DIO19/SPI_nATTN	Both	Output	GPIO/SPI Attention
13	GND	-	-	Ground
14	DIO18/SPI_CLK	Both	Input	GPIO/SPI Clock
15	DIO17/SPI_nSSEL	Both	Input	GPIO/SPI Slave Select
16	DIO16/SPI_SI	Both	Input	GPIO/SPI Slave In
17	DIO15/SPI_SO	Both	Output	GPIO/SPI Slave Out
18	Reserved	-	-	Do Not Connect
19	Reserved	-	-	Do Not Connect
20	Reserved	-	-	Do Not Connect
21	Reserved	-	-	Do Not Connect
22	GND	-	-	Ground
23	Reserved	-	-	Do Not Connect
24	DIO4	Both	Disabled	GPIO
25	DIO7/nCTS	Both	Output	Clear-to-Send Flow Control/ GPIO
26	DIO9/On_nSLEEP	Both	Output	Module Status Indicator/GPIO
27	VREF	-	-	Not connected
28	DIO5/ASSOC	Both	Output	Associate Indicator/GPIO
29	DIO6/nRTS	Both	Input	Request-to-Send Flow Control/ GPIO
30	DIO3/AD3	Both	Disabled	Analog Input/GPIO
31	DIO2/AD2	Both	Disabled	Analog Input/GPIO
32	DIO1/AD1	Both	Disabled	Analog Input/GPIO
33	DIO0/AD0	Both	Disabled	Analog Input/GPIO
34	Reserved	-	-	Do Not Connect
35	GND	-	-	Ground
36	RF	Both	-	RF IO for RF Pad Variant
37	Reserved	-	-	Do Not Connect

RECOMMENDED PIN CONNECTIONS



- The only required pin connections are VCC, GND, DOUT and DIN
- To support serial port firmware updates, connect:
  VCC, GND, DOUT, DIN ***PLUS*** RTS and DTR
- All unused pins should be left disconnected
- All inputs on the radio can be pulled high or low with 30k internal pull-up or pull-down resistors using the PR and PD commands
- No specific treatment is needed for unused outputs
- For applications that need to ensure the lowest sleep current, unconnected inputs should never be left floating: use internal or external pull-up or pull-down resistors, or set the unused I/O lines to outputs

DIO AND ADC PINS

The XBee/XBee-PRO RF Modules support ADC (Analog-to-digital conversion) and digital I/O line passing. The following pins support multiple functions:

Table 2-01. Pin functions and their associated pin numbers and commands
AD = Analog-to-Digital Converter, DIO = Digital Input/Output
Pin functions not applicable to this section are denoted within (parenthesis).

Pin Function	Pin#	AT Command
AD0 / DIO0	20	D0
AD1 / DIO1	19	D1
AD2 / DIO2	18	D2
AD3 / DIO3 / (COORD_SEL)	17	D3
AD4 / DIO4	11	D4
AD5 / DIO5 / (ASSOCIATE)	15	D5
DIO6 / (RTS)	16	D6
DIO7 / (CTS)	12	D7
DI8 / (DTR) / (Sleep_RQ)	9	D8

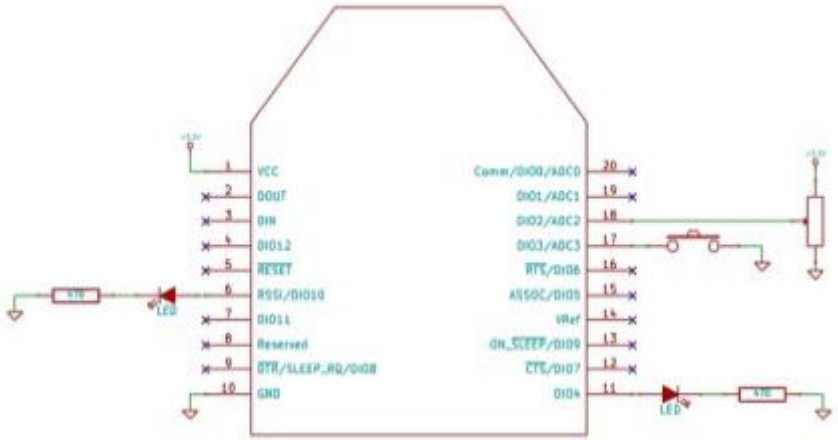
To enable ADC and DIO pin functions:

For ADC Support:	Set ATDn = 2
For Digital Input support:	Set ATDn = 3
For Digital Output Low support:	Set ATDn = 4
For Digital Output High support:	Set ATDn = 5

PIN NAMING CONVENTION

Module Pin Names	Module Pin Numbers	Configuration Command
CD/DIO12	4	P2
PWM0/RSSIM/DIO10	6	P0
PWM/DIO11	7	P1
DIO4	11	D4
CTS/DIO7	12	D7
ASSOC/DIO5	15	D5
RTS/DIO6	16	D6
AD3/DIO3	17	D3
AD2/DIO2	18	D2
AD1/DIO1	19	DI
AD0/DIO0	20	D0

EXAMPLE: HOW TO USE THE I/O PINS

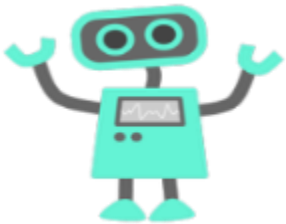


Line	Pin	Value
DIO2/ADC2	18	ADC (Analog Input)
DIO3/ADC3	17	Digital Input
RSSI/DIO10	6	PWM (Analog Output)
DIO4	11	Digital Output

WHAT DO I USE THE I/O PINS FOR?

- Turn on/off an LED (low power circuit)
- Turn on/off a transistor (higher power circuit)
- Control a relay (AC appliance like lamp, toaster, etc.)
- Control a motor via a H-bridge (water pump, etc.)
- Read an analog value from a sensor (light, temp, etc.)
- Read the value of a potentiometer (user controlled)

A lot of things in your life!!!!




ZB SLEEP MODE CONFIGURATION

Disabled

- SM=0
- Modules always awake

Pin Sleep

- SM=1
- Pin 9 (Sleep_RQ) HIGH = Sleep
- Pin 9 (Sleep_RQ) LOW = WakeUp

Cyclic

- SM=4
- SP=Sleep Period before wakeup (SN=multiplier)
- ST= Sleep time before sleep

Cyclic with Pin Wakeup

- SM=5
- Same as Cyclic mode, but if Pin 9 (Sleep_RQ) transitions from HIGH to LOW, the module will WakeUp prematurely

End Device **Routers**

NOTE: data is buffered on routers for a maximum of 30 seconds; after that, is lost!


LET'S TAKE
-A-
Coffee
BREAK

ZIGBEE PROTOCOL

DETAILED

ADDRESSING

- ZigBee devices use 64-bit and 16-bit addresses
  - 64-bit address
    - ✓ Unique per device (MAC)
    - ✓ Fixed – best way to address a device
  - 16-bit address
    - ✓ Assigned to devices when joining
    - ✓ Used in routing tables
    - ✓ Volatile – can change under certain conditions
- Any ZigBee transmission specifies:
  - Source and destination addresses
  - Source and destination endpoints
  - Cluster ID
  - Profile ID

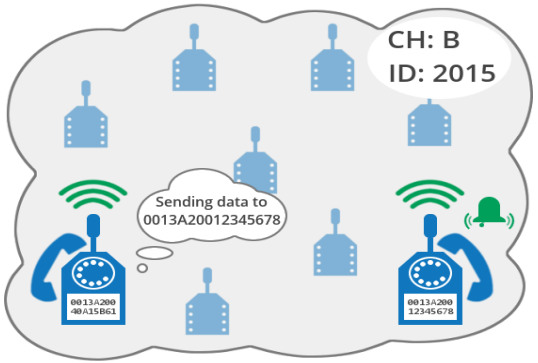
Sample Address Table

64-bit Address	16-bit Address
0013 A200 4000 0001	0x4414
0013 A200 400A 3568	0x1234
0013 A200 4004 1122	0xC200
0013 A200 4002 1123	0xFFFE (unknown)

XBEE SIMPLE FUNCTIONALITY OVERVIEW

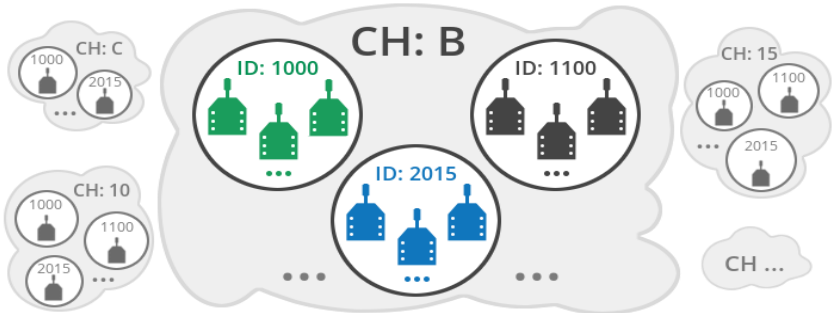
- Wireless data flow in a Point-to-Point communication:

- Addressing a node in a P2MP architecture:



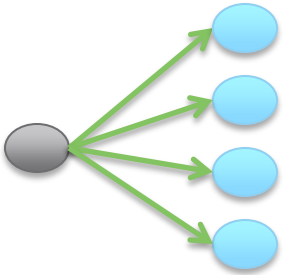
ABOUT CHANNELS AND NETWORK ID

- **Channel** (**CH**): frequency used to communicate (within the specific band)
- **Personal Area Network identifier** (**ID**): unique number that identifies the network (and differentiate it from others)
- An XBee will only be able to communicate with other radios within the **same network** and using the **same channel**
- Multiple networks can co-exist on different PAN ID or Channels

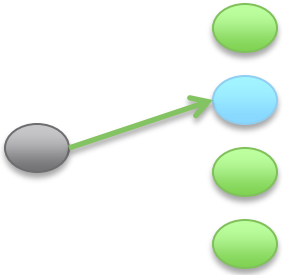


TX MODES: BROADCAST vs. UNICAST

- **Broadcasts**
  - Sent to all radios in the network



- **Unicasts**
  - Addressed to a single radio in the network



UNICASTS: NETWORK ACK / RETRIES

- Unicasts are called a “reliable” delivery method
- When a transmission is received correctly the receiver sends an acknowledgement to the sender
- If the acknowledgment is not received in the allocated time the transmission is attempted again (until all retries are used up)
- Two levels of acknowledgments/retries
  - **MAC**: for intermediate transmissions (between adjacent nodes along a route)
  - **NWK**: for transmissions along the entire route

PARENT-CHILD RELATIONSHIP

- ZigBee networks are subject to a parent-child relationship
- A coordinator/router has a limited number of end-nodes that can be directly addressed, called children
- Without a parent, a child cannot join a network!
- XBee S2B coordinator can support up to 10 end-devices and a router can support 12 end-devices
- XBee S2C (both coordinator and router) can support up to 20 end-devices
- The number of remaining slots in the child table can be queried using the **NC** parameter (number or remaining children)

PAYLOAD

- When the radio is in transparent (AT) mode, packet size is irrelevant as the data will be fragmented as needed
- However, when using API frames, it is important not to exceed the allowable payload size
- Maximum payload size varies depending on parameters on the radio
- **NP** parameter reports the maximum payload size
- If you are using a XBee ZigBee, **NP** will always return 0xFF:
  This is because the ZigBee protocol does support payload fragmentation (when needed)
- If you want to avoid fragmentation, the payload size should be max 84 bytes without encryption, and 66 bytes with encryption enabled
- If transmission is a broadcast, max payload is 92 bytes
- NOTE: payload depends on the XBee model!!

ZIGBEE SECURITY

ZigBee supports a high level of security

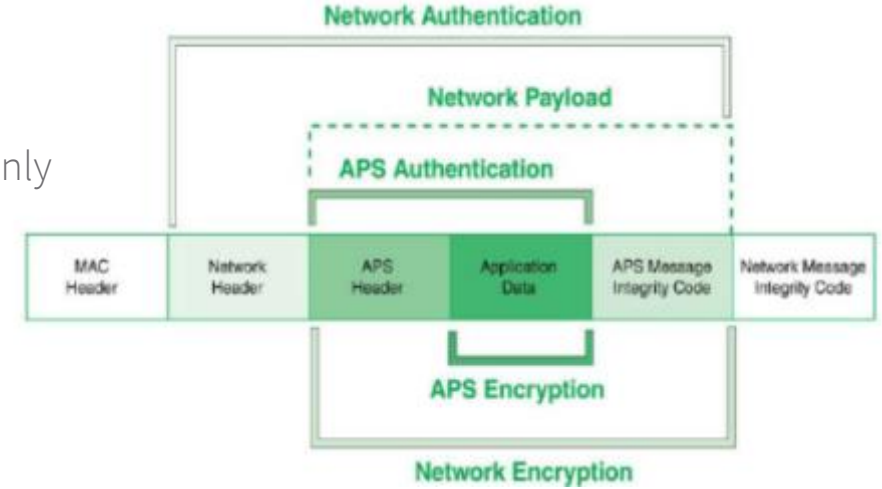
- 128-bit AES algorithm
- Preconfigured (fixed) or random key (generated)
- Trust center to manage the key
- Two types: Network Encryption or APS Encryption (*see next page*)
- Slower throughput!

Description of Settings:

- **EE** = if 1, encryption is enabled
- **KY** = Link Key, encrypts the link (and passwords!)
  if 0, the key is shared "*clear*" – **SECURITY HOLE!**
  if not 0, a "*trust center*" stores the key and new nodes can only join if they know the preconfigured key
- **NK** = key to be used (if 0, means random generated)
- **NJ** = permit joining – if 0xFF always allowed – **POSSIBLE SECURITY HOLE!**
  If 0x00 does not allow anymore joining – **DANGEROUS!**

ZIGBEE AUTH / ENCRYPTION

- **Network encryption**
  - Messages encrypted at a network level (payload + header)
  - Packets encrypted authenticated and decrypted at each hop
- **APS encryption**
  - Encrypts only the payload of the message
  - Each packet can have a different key
  - Key is known by originator and destination only



BANDWIDTH USAGE – XBEE S2B EXAMPLE

- The Network architecture has to be planned on the real performance and within the limits
- It is very easy to overview the real numbers!

Configuration	Data Throughput
1 hop, RR, SD	35kbps
1 hop, RR, SE	19kbps
1 hop, RE, SD	25kbps
1 hop, RE, SE	16kbps
1 hop, ER, SD	21kbps
1 hop, ER, SE	16kbps
4 hops, RR, SD	10kbps
4 hops, RR, SE	5kbps

RR = router to router,
RE = router to end device (non-sleeping),
ER = end device (non-sleeping) to router,
SD = security disabled,
SE = security enabled.
4 hops = 5 nodes total, 3 intermediate router nodes

Let's say we need to TX once every 10 seconds.

Each transmission needs to be 100 bytes long.

$$\left(16{,}000\,\frac{bits}{second}\right)\left(\frac{1\ byte}{8\ bits}\right) = 2{,}000\,\frac{bytes}{second}$$

$$\left(2{,}000\,\frac{bytes}{second}\right)\left(\frac{1 transmission}{100 bytes}\right) = 20\,\frac{transmissions}{second} * 10\ seconds$$

$$= 200\ transmissions\ (nodes)$$

But we really can't use 100% of the bandwidth.

Let's say 20% or 30% of that.

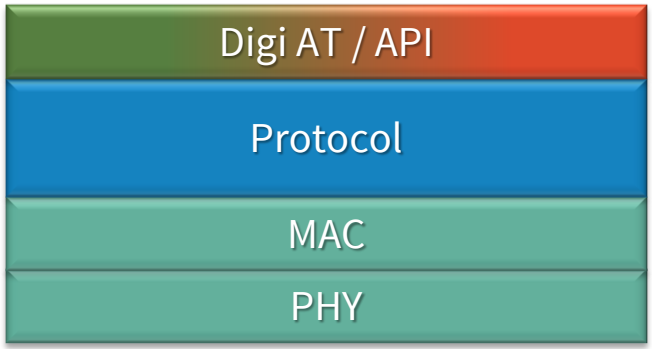
$$= 40\ to\ 60\ nodes$$

INTERFACING THE XBEE

AT AND API MODES

WHAT IS AT / API ?

- AT or API interface application layer on top of protocol layer:
  - 802.15.4
  - DigiMesh
  - ZigBee
- Powerful feature set includes:
  - Extended addressing
  - Remote configuration
  - I/O line set/read capabilities
  - Network diagnostics
- Method of communication to the radio from the application
- Common for both ZigBee and DigiMesh module



AT VS. API FEATURE POINTS

AT

ADVANTAGES

- Simple
- Transparent
- Best for streaming

DISADVANTAGES

- Switching destination is cumbersome
- Slow for commands
- Better for humans
- No additional information

AT

VS

API

API

ADVANTAGES

- Quick commands
- Additional information
- Over the air (OTA) configuration
- Direct addressing
- Supports Cluster-ID

DISADVANTAGES

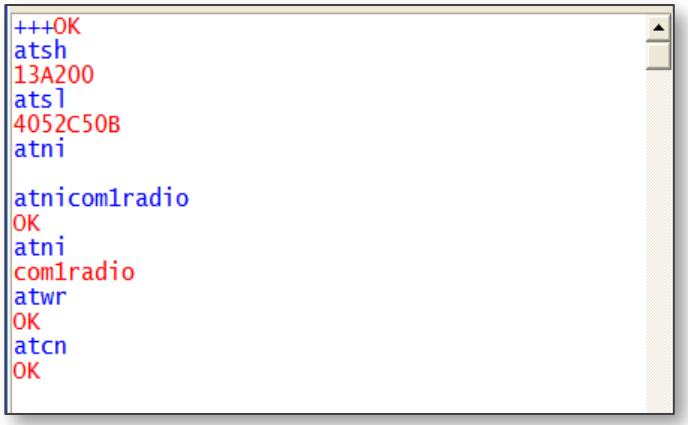
- More Complex
- Not transparent
- Requires “intelligence”

TRANSPARENT / AT MODE

- Originally developed for Hayes modems in the 1980s
- Transparent mode works “out of the box”
- Serial cable replacement
- Command mode (enter ‘+++’ default)
  needs 1 sec pre and post silence to work!!
- Simple configuration of network, addressing,
  and other advanced features
- Payload data gets encapsulated into a packet,
  so it is not interpreted as a command
- NO check is done on the packets, NO FrameID and
  NO error information is reported
- AT mode is “for humans” or for simple devices with no intelligence

AT MODE COMMANDS

- Method for configuring radio settings
- Read or write registers
- Can be set/queried in three different ways:
  - AT command mode
  - Local API commands
  - Remote API commands



USEFUL AT COMMANDS

- **Default for terminal is 9600, 8, N, 1**
- +++ (without ENTER) - switch to Command Mode
  - reverts to Data Mode after 1sec timeout (GT parameter)
  - or, ATCN to exit Command Mode immediately
- ATFR – software reset
- ATRE – factory defaults
- ATWR – saves the current settings
- ATID – sets the PAN ID
- ATND – node discovery
- ATNI – node identifier (mnemonic)
- ATDN – resolves dest node by name
- ATCH – operating channel
- ATOP – Operating PAN ID
- ATDH – destination high address
- ATDL – destination low address
- ATMP – report parent address
- ATNC – number of remaining child
- ATNJ – allowed time for nodes to join
- ATDJ1 – does not allow joining of the node until next reset
- ATNR0 –network reset for node
- ATNR1 – broadcast (full) network reset
- ATDD – device (product) type
- ATED (for 802.15.4) – ambient noise
- ATDB – signal info
- ATAI – association (join) indicator
- ATCB – commissioning button simulation
- ATVR – Firmware version
- ATHV – Hardware version
- ATEE – Enable/disable Security
- ATIS – sample digital or analog I/O

API MODE

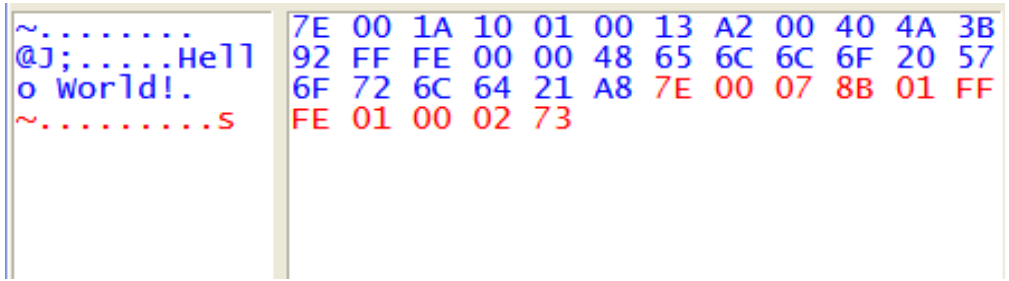
- Commands and data sent to the module through the API protocol (Application Programming Interface)
- API frames allows using:
  - Simple transmit and receive, using module addresses
  - Advanced transmit and receive frames expose ZigBee cluster IDs, endpoints

API

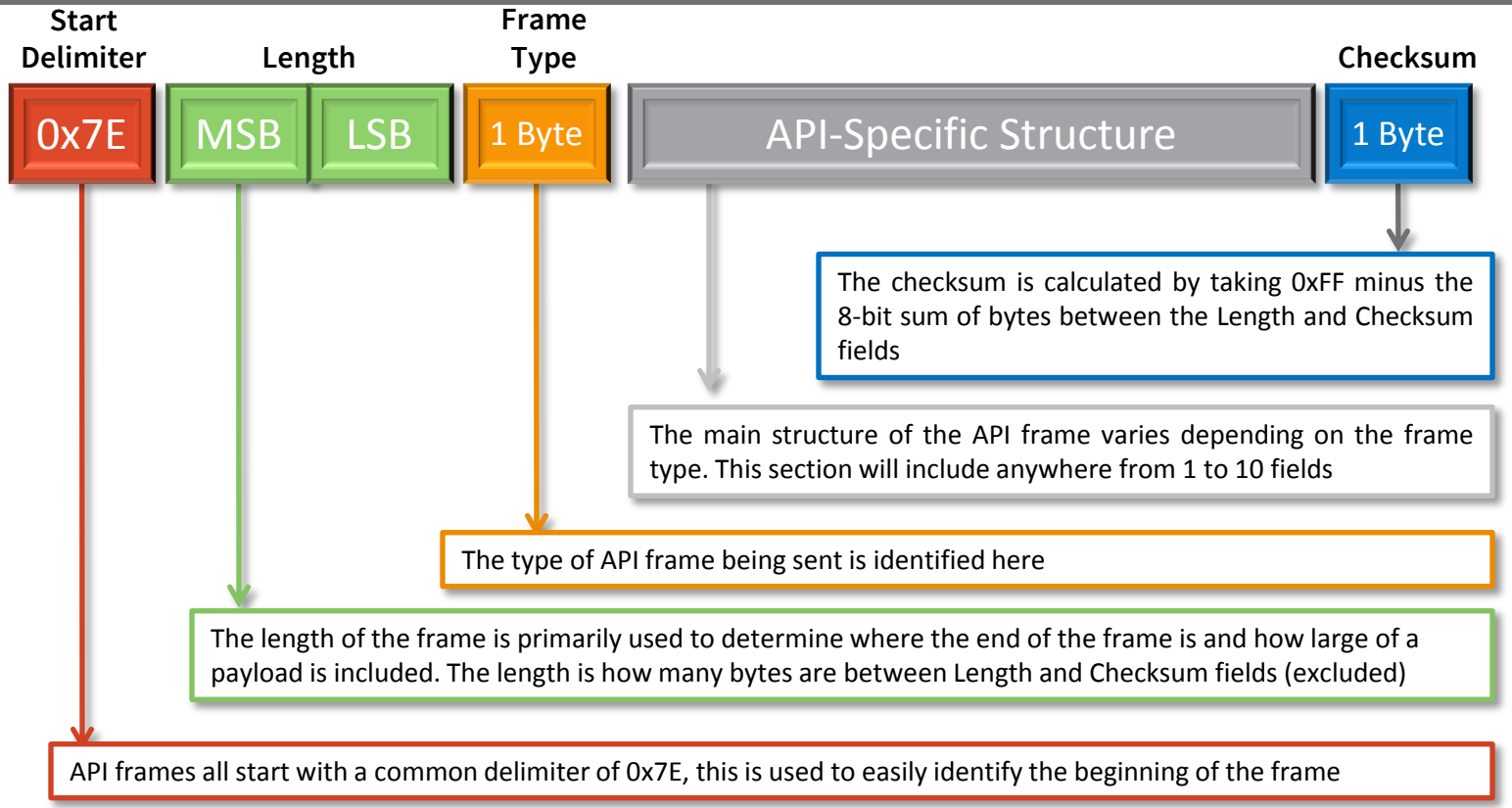
- API frames expose both simple and advanced ZigBee capabilities
- Status Frames indicates device and network-level behavior including join indications, reset notifications, etc…
- Header and CRC are checked before sending the packet out (and could be rejected)

API MODE COMMANDS

- Data transmitted and received is contained in frames
- Frame headers contain valuable information
- Frames get FrameID assigned to
- Fast way to address packets, change parameters, etc.



API FRAME STRUCTURE



API FRAMES CONSTRUCTION

Start Delimiter	Length		Frame Data							Checksum
			Frame type	Data						
1	2	3	4	5	6	7	8	9	...	n	n + 1
0x7E	MSB	LSB	API frame type	Frame-type-specific data						Single byte

MSB: most-significant byte, **LSB**: least-significant byte

- To manually build an API Frame, you can use XCTU under:
  Tools -> API frame Generator
- In alternative, Digi provides an online builder at:
  http://ftp1.digi.com/support/utilities/digi_apiframes2.htm
- The code running in your HW, has to build, send, receive, recognize these frames!

API FRAME TYPES

API Frame Name	Value
Modem Status	0x8A
AT Command	0x08
AT Command - Queue Parameter Value	0x09
AT Command Response	0x88
Remote Command Request	0x17
Remote Command Response	0x97
ZigBee Transmit Request 0x10	0x10
Explicit Addressing ZigBee Command Frame	0x11
ZigBee Transmit Status	0x8B
ZigBee Receive Packet (AO=0)	0x90
ZigBee Explicit Rx Indicator (AO=1)	0x91
ZigBee IO Data Sample Rx Indicator	(0x92)
XBee Sensor Read Indicator (AO=0)	0x94
Node Identification Indicator (AO=0)	0x95

Note: Digi may add new API frames to future versions of firmware, so please build into your software interface the ability to filter out additional API frames with unknown API identifiers

API – NETWORK/NODE DIAGNOSTICS

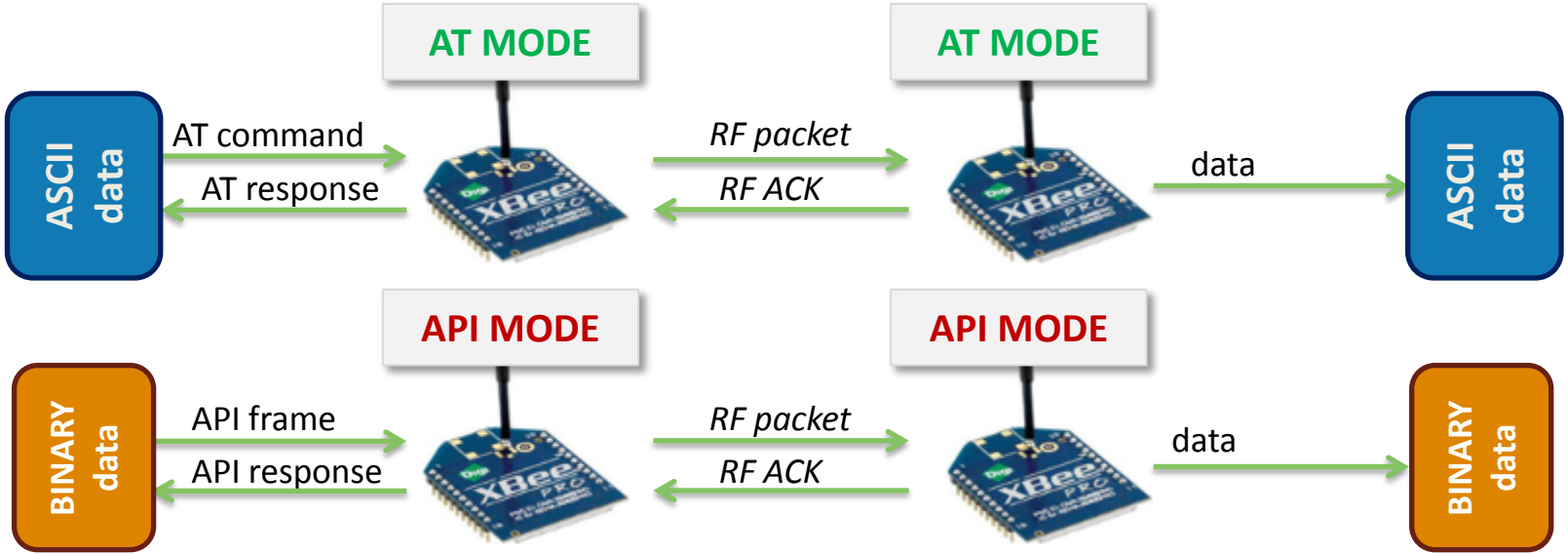
Through API mode it is possible to:

- Receive node status
  (frame 0x8a – modem status)
  (frame 0x08/0x09/0x88 – AT command and response)
- Receive network status messages
  (frame 0x95 – Node Identification Identifier)
  - New nodes joining
  - Commisioning Button Pressed
- Request network status
  (frame 0x17/0x97 – Remote AT command and response)
  - Node Discovery
  - Node queries

API - OVER THE AIR

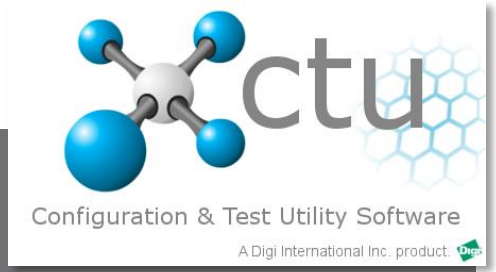
- Using API mode it is possible
  - Changing configuration over the air
  - Set/Reset and read I/O lines
  - Read A/D values
  - Load new FW into the modules!
- XBee also can send you automatic low battery warning (voltage threshold)

XBEE DATA TRANSMISSION OVERVIEW

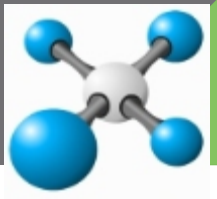


- There is no difference in sending data in AT or API mode, as the data in the air is always RF!!!
- It is possible to send data in API mode and receive in AT mode !! (and vice versa)

DIGI XCTU


xctu
Configuration & Test Utility Software
A Digi International Inc. product.

XCTU FUNCTIONALITIES



- Free Digi desktop application
- Java- based, supports Windows, Linux, MacOS
- Allows managing the XBee (and other RF) Digi modules
- Easy interface to create AT commands and even build more complex API frames
- Features:
  - ✓ Managing local modules
  - ✓ Managing remote modules (if in the same network)
  - ✓ Frame Generator and Frame Interpreter
  - ✓ Range Test
  - ✓ Module Recovery
  - ✓ Firmware update
  - ✓ Console – ASCII Terminal (AT mode) or Binary (API mode)
  - ✓ Network Graphical representation

Product page and download: http://www.digi.com/xctu
Walkthrough: http://docs.digi.com/display/XBeeArduinoCodingPlatform/XCTU+walkthrough

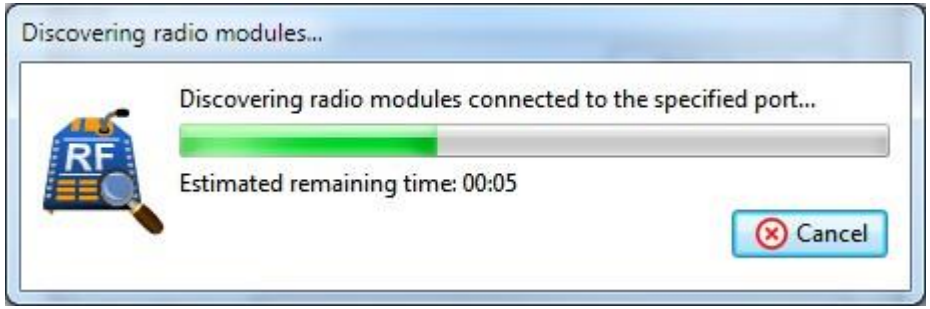
INSTALL THE DRIVERS!!!

- Before using the XBee in your computer, you need to install the drivers
- Generally, when you connect the XBee board for the first time, the drivers are **automatically** installed
- You don't need to **manually** install the USB drivers unless your operating system notifies you that automatic driver installation has failed
- In that case, you can download and install the USB drivers from the Digi Support Site on the link below:
  http://www.digi.com/support/productdetail?pid=4549
- Go under "drivers", choose your operating system, download the file, and follow the steps in the installation wizard

XCTU ADDING A DEVICE

- Click the “Add a Radio module” button from the toolbar

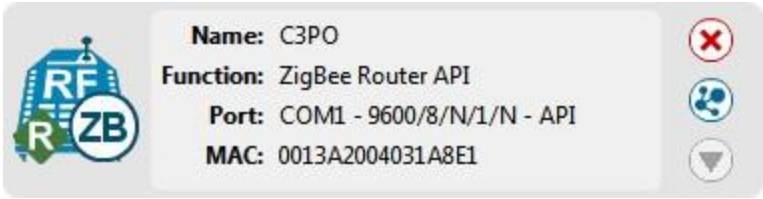
- Select the serial port to which the radio module is connected and configure the serial settings of the port
- When ready, click Finish to add the radio module to the list of modules



XCTU MANAGING DEVICE INFO

The module info bar will contain a circle at the right-bottom side with 2 letters that indicate the protocol of the radio module:

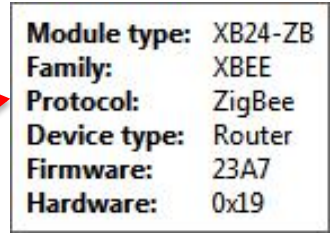
- **ZB**: ZigBee protocol
- **DM**: DigiMesh (Digi's proprietary protocol)
- **802**: 802.15.4 protocol
- **DP**: Point-to-multipoint protocol
- **SE**: Smart Energy protocol
- **ZN**: ZNet protocol
- **WF**: Wi-Fi protocol
- **XC**: XSC (XStream Compatibility) protocol
- **XT**: XTend native protocol
- **XLR**: XLR PRO
- **?**: Unknown protocol



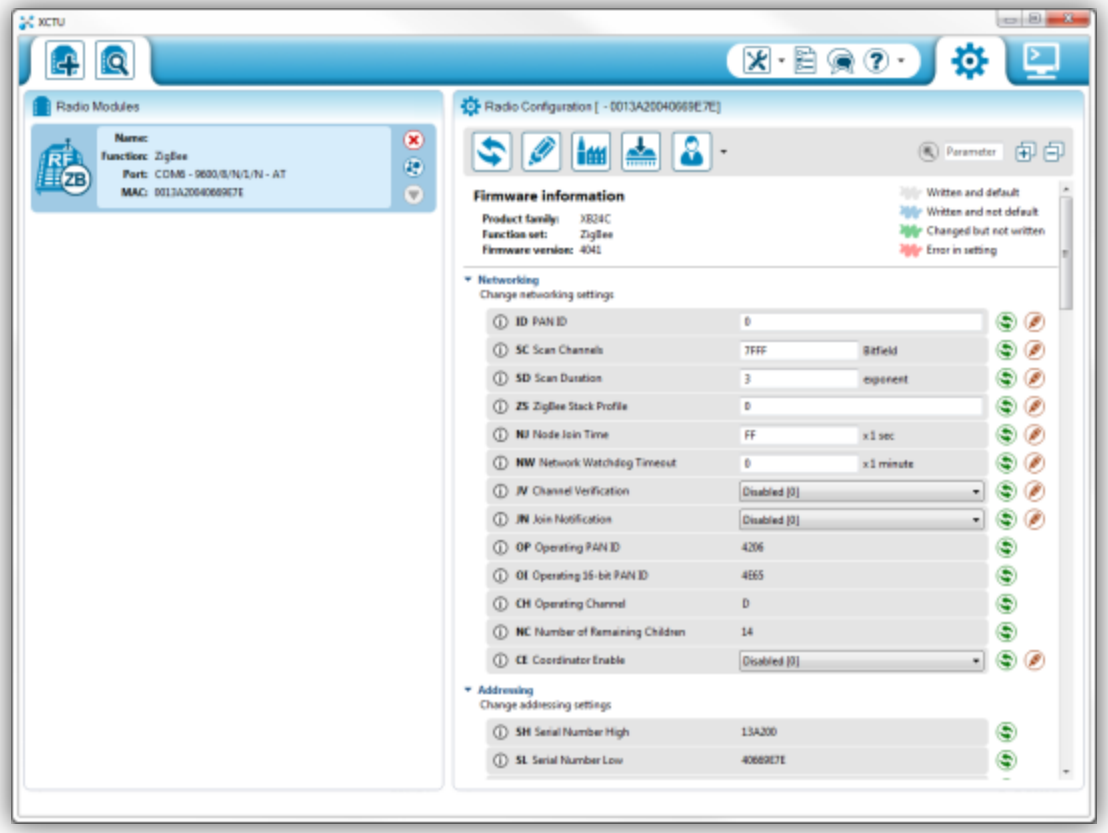
At the left-bottom side there will be also a little image with a letter which indicates the role of the module within its network:

- **C**: Coordinator
- **R**: Router
- **E**: End device

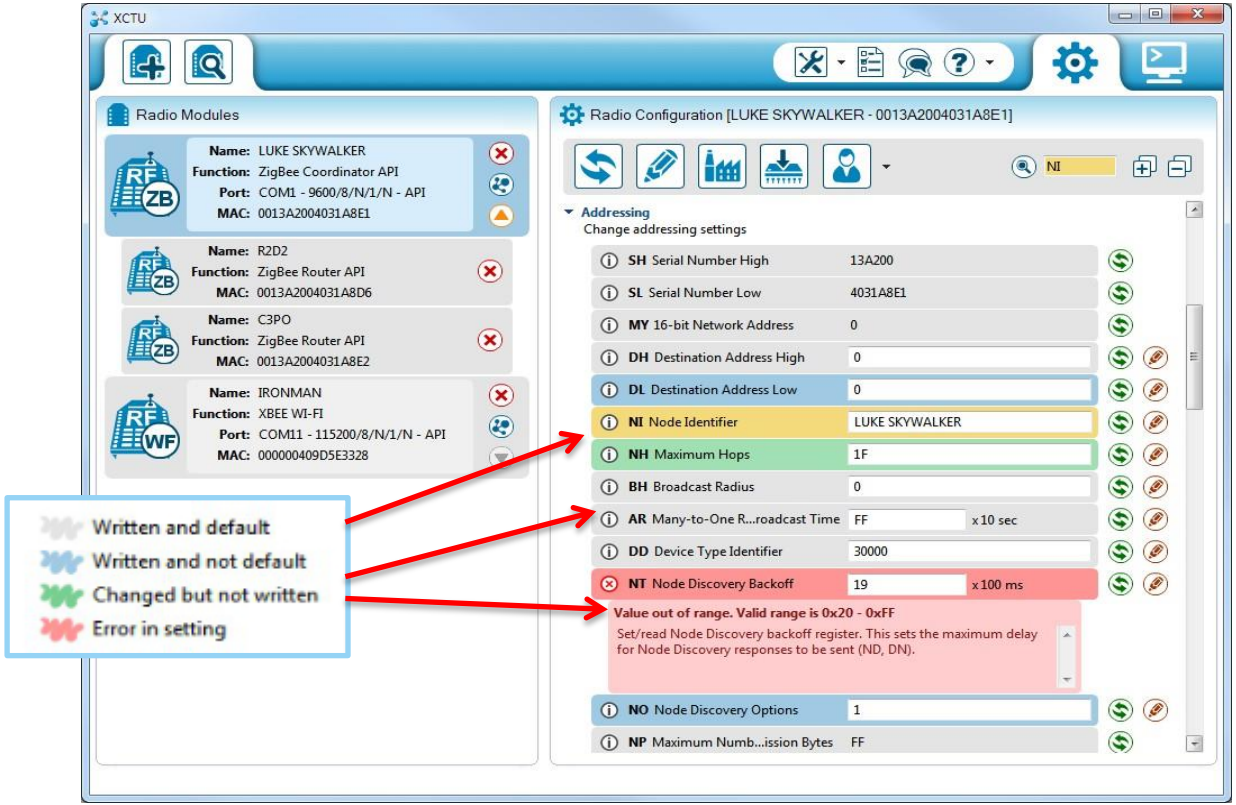
If you hover over the icon with the mouse, XCTU will display more information about the module



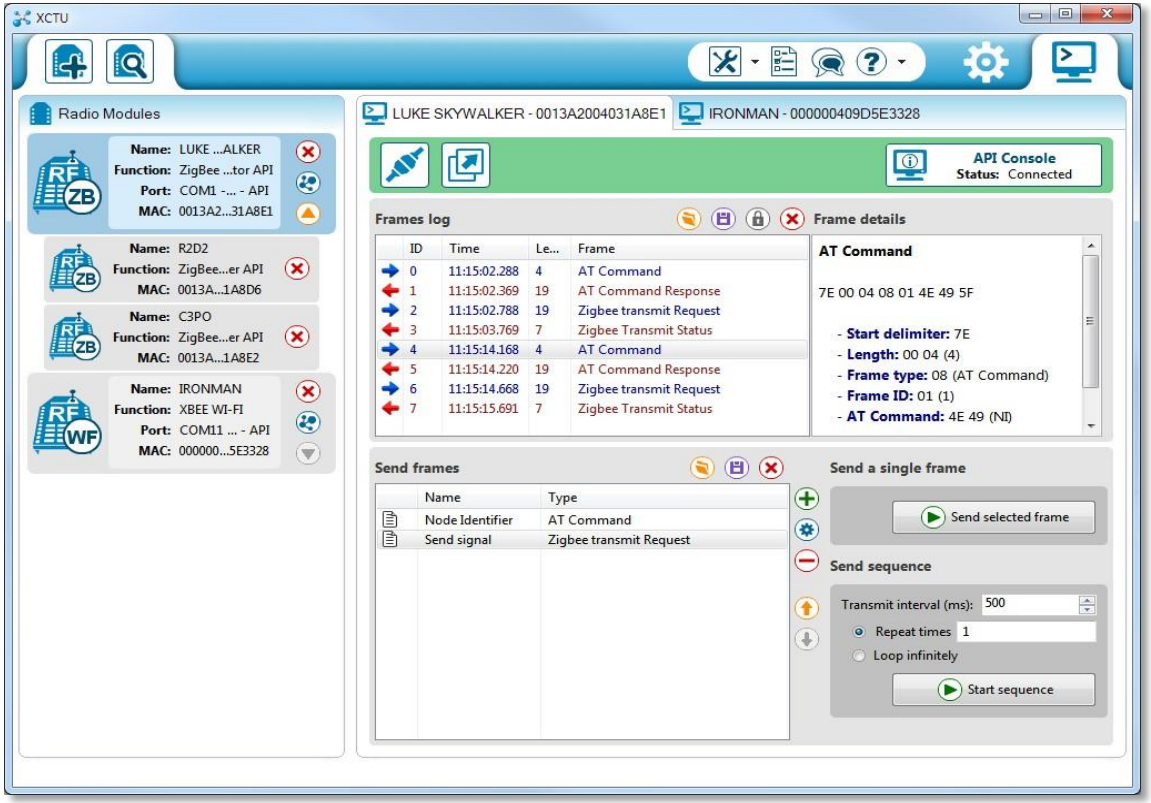
XCTU: FIRMWARE INFO AND SETTINGS



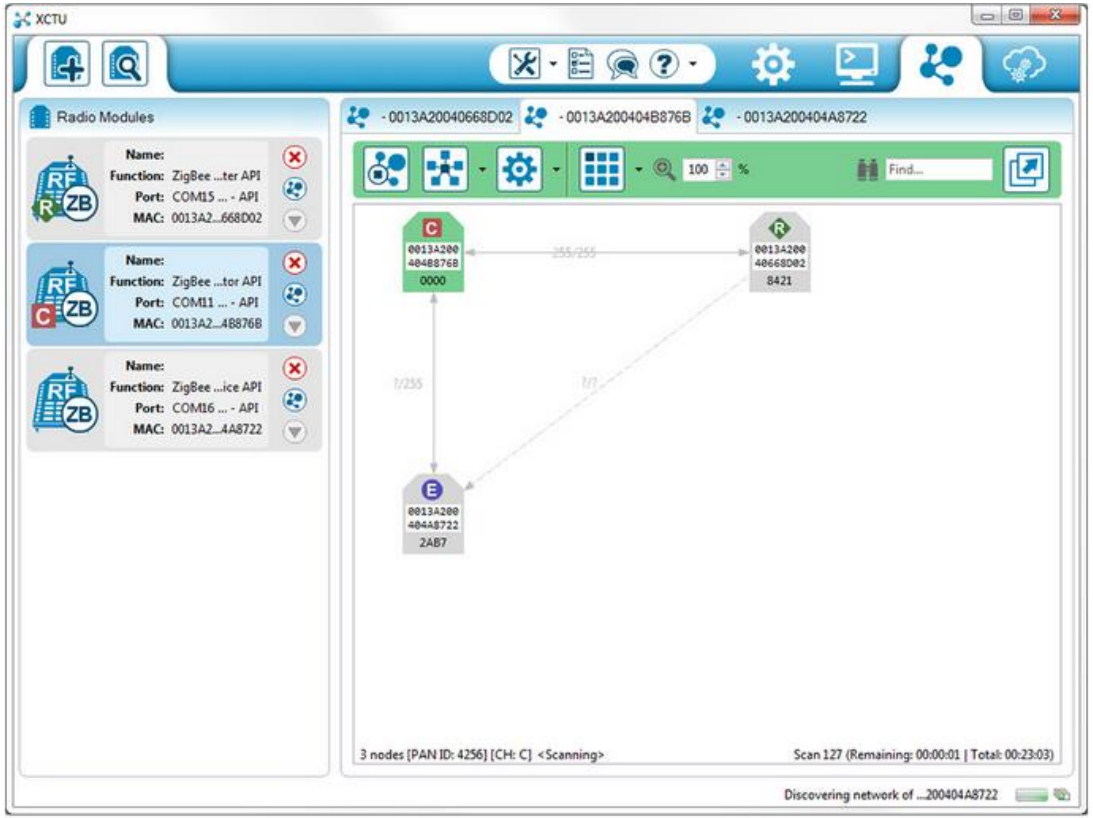
XCTU: CHANGING PARAMETERS



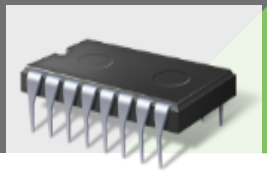
XCTU: NODE VIEW WITH API CONSOLE



XCTU: NETWORK GRAPHICAL VIEW

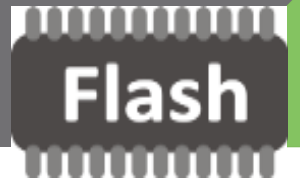


XBEE FIRMWARE

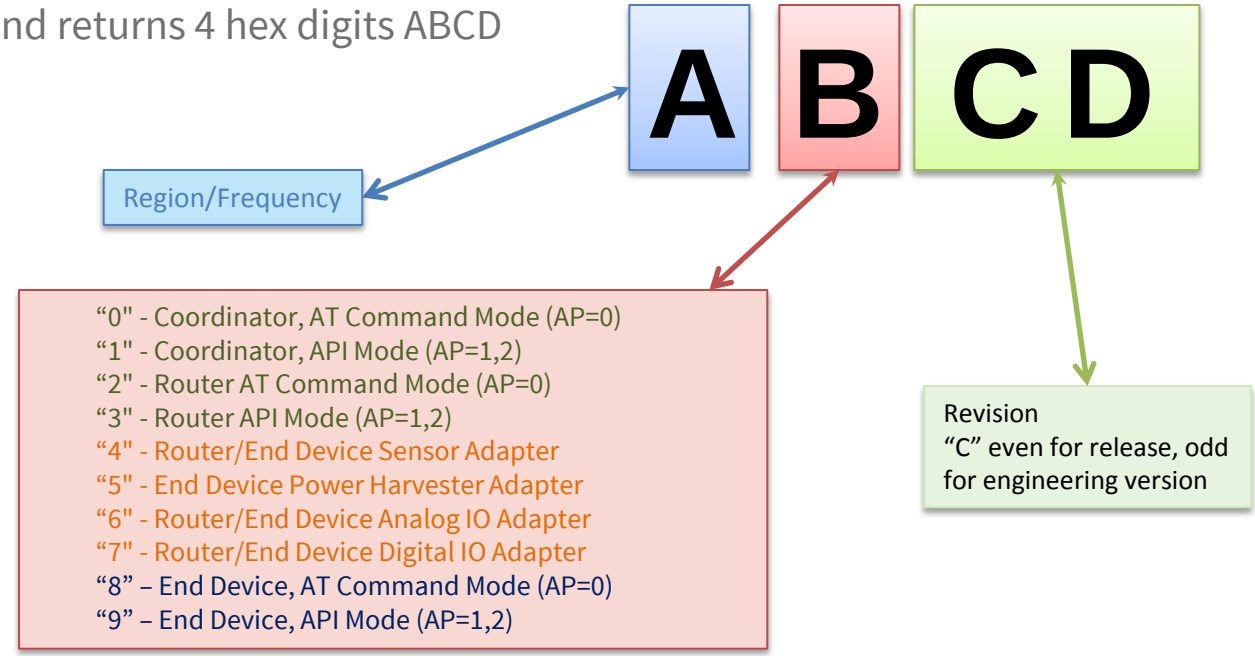


- Radio firmware is the program code stored in the XBee persistent (FLASH) memory
- The radio firmware can be changed to maintain the modules up-to-date with the new features
- XCTU allows you changing the firmware of local and remote XBee over the air (OTA) using the Update firmware tool
- XCTU maintains a library of firmware files to be used in your modules
- Remember to upgrade to the latest firmware:
  Help > Update the Radio Firmware Library

XBEE FIRMWARE VERSIONS

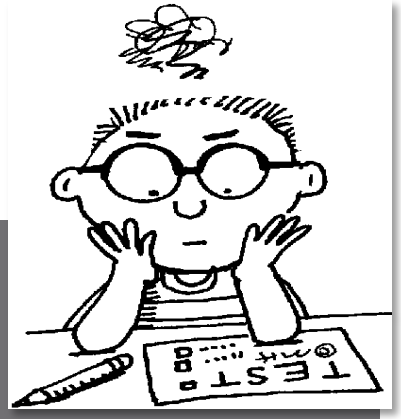


- XBee version numbers have 4 significant digits
- **ATVR** command returns 4 hex digits ABCD
  e.g. 21A7



DIGI

TESTS / DIAGNOSTICS



NETWORK DIAGNOSTICS

- Node Identifier command
- Network discovery
- Commissioning button/assoc. LED
- Neighbor polling*
- Loopback Testing (cluster 12)
- Trace routing*
- Adjacent link testing*
- NACK messages*

(unique features, available for DigiMesh Only)*

SIMPLE DIAGNOSTICS

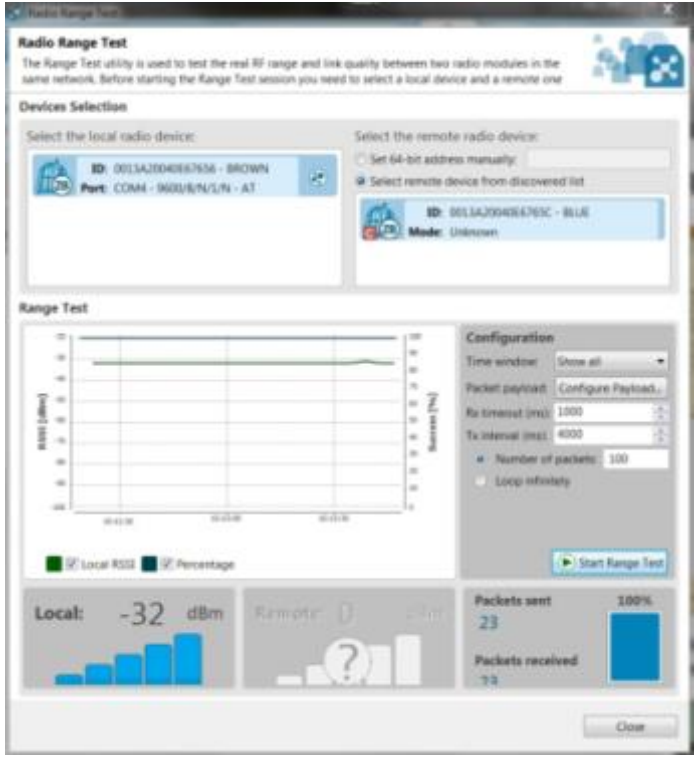
- Have the module in AT mode and launch XCTU
- Under “Console”, press “Open”
- Now you can type any of the following AT Commands

- VR – Prints the Firmware version loaded into the module
- HV – Prints the HW version of the module
- AI – Association Indication, if it is 0 it means it has associated with a Coordinator or a Router, otherwise it is not. To see the list of the values, check the Hardware Reference Manual P/N 90000976_P
- NI – Node Identifier, allows setting a mnemonic ID of the module, so it is easier to recognize it (not really a diagnostic but helps a lot!)
- MY – Shows the 16-bit address of the module (variable/volatile) <- ZB only
- SH and SL – Show the High and Low part of the 64-bit address of the module (MAC – unique/fixed)
- ND – Node Discover, lists all the modules found in the current PAN

NOTE: the module has to be in AT mode, also remember to type +++ first!!

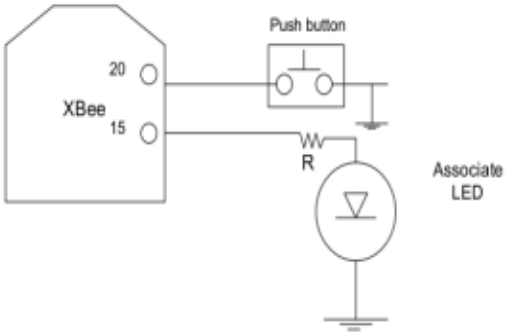
RANGE TEST

- The Range Test is designed to send data between two radios and verify the validity and time of response
- It allows verifying the signal intensity (RSSI) expressed in dBm
- The Range Test can be done in XCTU under: Tools -> Range Test
- It is possible to define the number of packets to use for the test
- If “Loop Infinitely” is chosen, then the test will run until you click: “Stop Range Test”
- The test results are visible in real time while the test is running
- In alternative, connect RX and TX on the remote node (pins 2, 3) and send some data (loopback mode)



COMMISS. BUTTON & ASSOCIATION LED

- Two important I/O lines that can be used for diagnostic
- Commissioning button:
  - 1 button press:
    - Wakes up device if it is sleeping
    - Broadcasts a node identification frame
  - 2 button press:
    - Causes a sleeping router node to become a sleep coordinator
  - 4 button press:
    - Node leaves the PAN and tries to associate to ANY PAN (same as ATRE)
- Associate LED
  - Various blink codes for sleep status
  - Blinks fast for one second when it receives a node ident frame



DM DIAGNOSTICS: NEIGHBOR POLLING

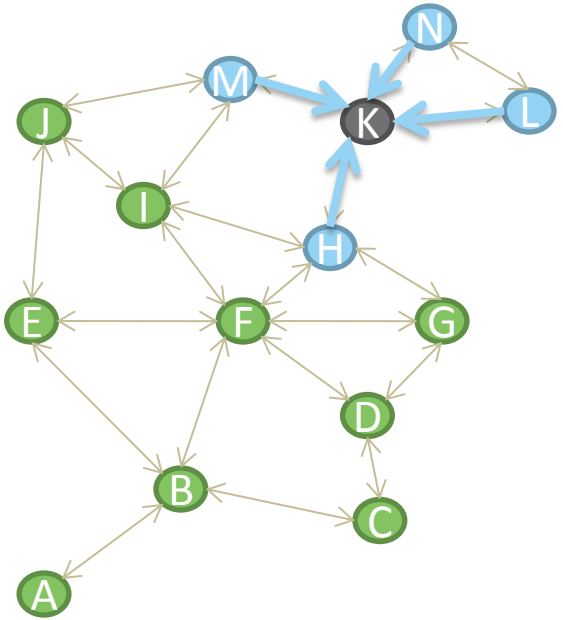
DM

- Find Neighbor command (FN)
- Variant of the Network Discovery (ND) command
- Only immediate neighbors will respond to the command, one message per neighbor
- This command can be remotely initiated
- Useful for mapping out network connections

HOW NEIGHBOR POLLING WORKS - 1

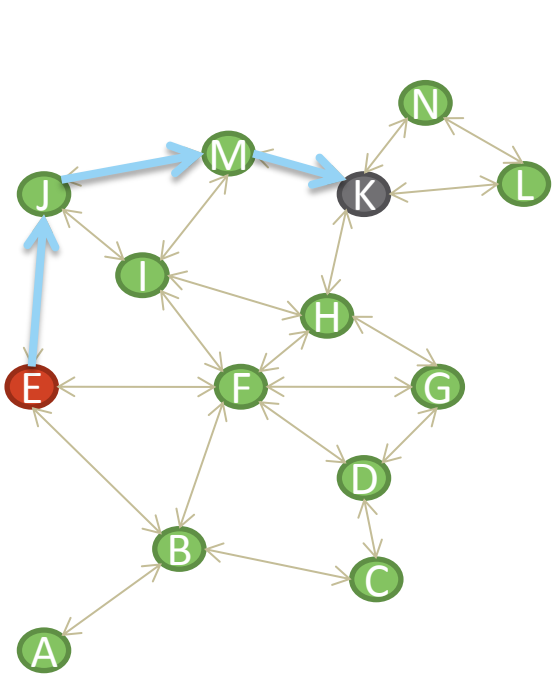
- If node “K” did a local Find Neighbor (FN) command then nodes “M,” “H,” “N” and “L” would respond by unicasting responses to “K”

- Locally in AT mode: “ATFN”



HOW NEIGHBOR POLLING WORKS - 2

- Command can be remotely initiated (“E” could ask “K” to find its neighbors)
- Create a Remote AT command with value of “FN”



DM DIAGNOSTICS: NACK OPTION

- Unicast API option
- Causes a route information packet to be sent to the unicast originator in the event that a link fails
- Can be enabled on some or all unicasts
- Bit 2 of Transmit Options (TO) for all messages
- Bit 2 of byte 16 of Transmit Request message

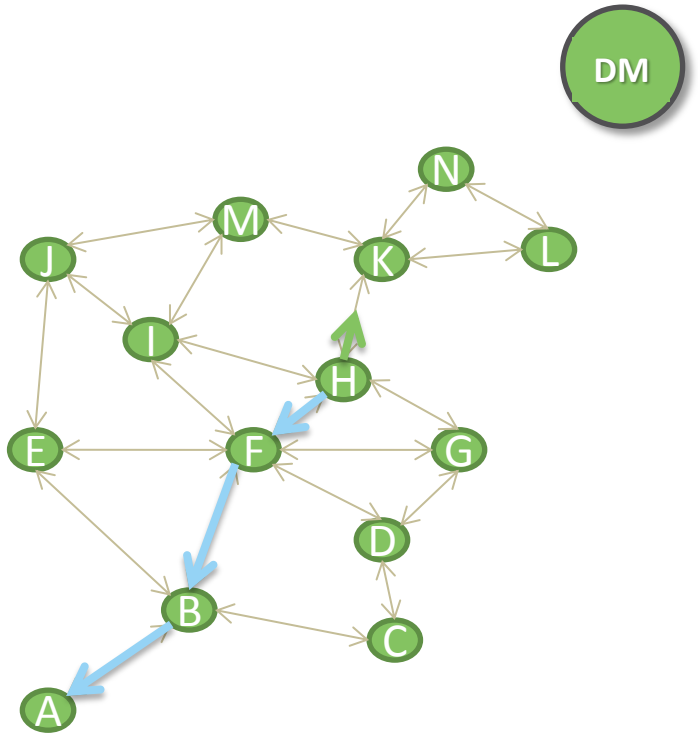
HOW NACK WORKS

- Example: “A” unicasts to “N”
- Packet has NACK enabled
- Packet fails between radios “H” and “K”
- Route information packet:

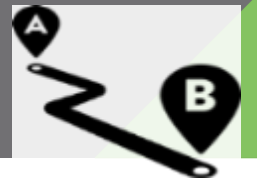
Route Source	Route Dest	Link Source	Link Dest	Retries Used
A	N	H	K	10

Unicast Packet

Route Information Packet



DM DIAGNOSTICS: TRACE ROUTE



Create a Trace route packet

- No explicit frame required
- Frame type: **0x10** (Transmit Request)
- Option: **0x08**
- Example: 7E 0012 **10** 01 0013A2004054AA9D FFFE 00 **08** **54 65 73 74** B9
- What is the meaning of the payload in green?
- Returns: a Route Information (0x8D) per hop and a standard Transmit Status

LOOPBACK (CLUSTER 0x12)

- By using the Cluster 0x12, any data transmitted to the remote device, will be returned to the sender
- The loopback will be done “internally” to the radio so it won’t test external failures
- To test external connections, it is necessary to use an external loopback (either physical connector or wires)
- Can be done both in AT and in API modes

DM DIAGNOSTICS: LOOPBACK TEST

Create an test a loopback TX request frame

- Frame type: 0x11 (Explicit Addressing Command Frame)
- Cluster: 0x0012
- Profile: 0xC105
- Example: 7E 002C 11 01 0013A20040506070 FFFE E8 E8 0012 C105 00 00 54 68 69 73 20 69 73 20 61 20 6C 6F 6F 70 62 61 63 6B 20 74 65 73 74 21 B2
- What is the meaning of the payload in green?
- Returns: a Receive Packet (0x90) containing the transmitted payload and a standard Transmit Status

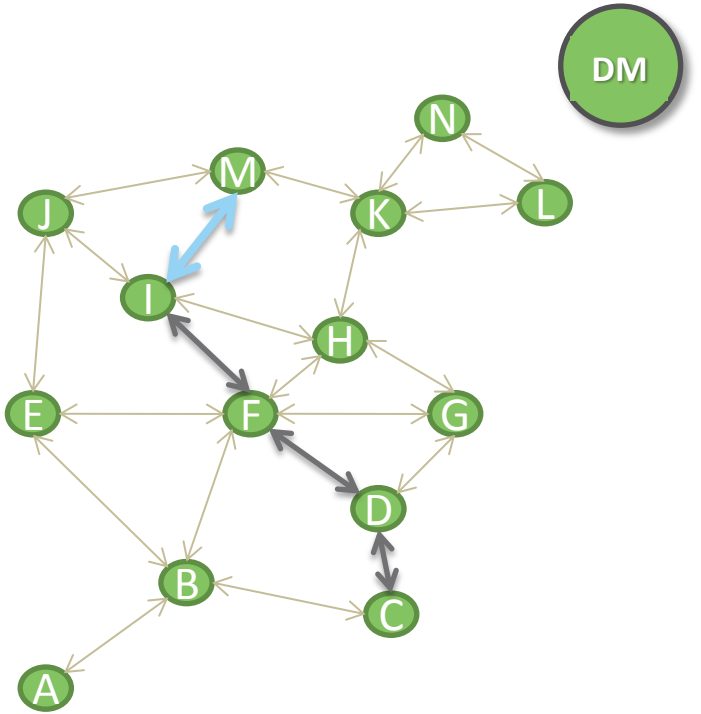
DM DIAGNOSTICS: ADJACENT LINK TEST

Create an Adjacent Link test (modules MUST be adjacent!)

- Frame type: 0x11 (Explicit Addressing Command Frame)
- Cluster: 0x0014
- Profile: 0xC105
- Packet 40 bytes (0x0028) repeated 1000 times (0x03E8)
- Example: 7E 0020 11 01 0013A20040521234 FFFE E6 E6 0014 C105 00 00 0013A2004052ABCD 0028 03E8 EB
- Returns:
  - Bytes 13-14: Success packets
  - Byte 15: 0x00=success 0x03=invalid parameter
  - Byte 18: max RSSI observed during the test
  - Byte 19: min RSSI reading observed during the test
  - Byte 20: average RSSI observed during the test

(REMOTE) ADJACENT LINK TESTING

- Tests of up to 4000 packets can be performed between nodes
- Can be remotely initiated
- In this example:
  - Radio “C” remotely requests that Radio “I” test its link with radio “M”
  - Statistics packet will be returned to radio “C”




LET'S TAKE
-A-
Coffee
BREAK

XBEE PROGRAMMABLE

LIMITATIONS OF THE STANDARD XBEE

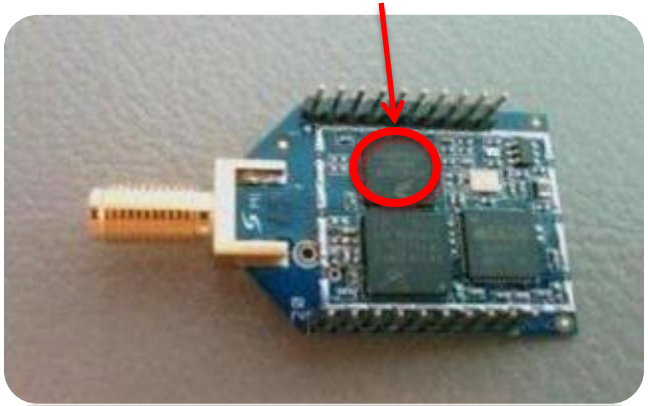
- No on-board intelligence
  - Only remote commands triggers XBee behavior
  - Preconfigured input sampling
  - No custom code running on the XBee
- No precise I/O timing
  - PWM output is for signal strength only
  - Unpredictable network delays interfere with precisely timed outputs

WHAT IS A PROGRAMMABLE XBEE?

XBee 865/868LP

XBee-PRO ZigBee

Freescale Microprocessor (MC9S08QE32)

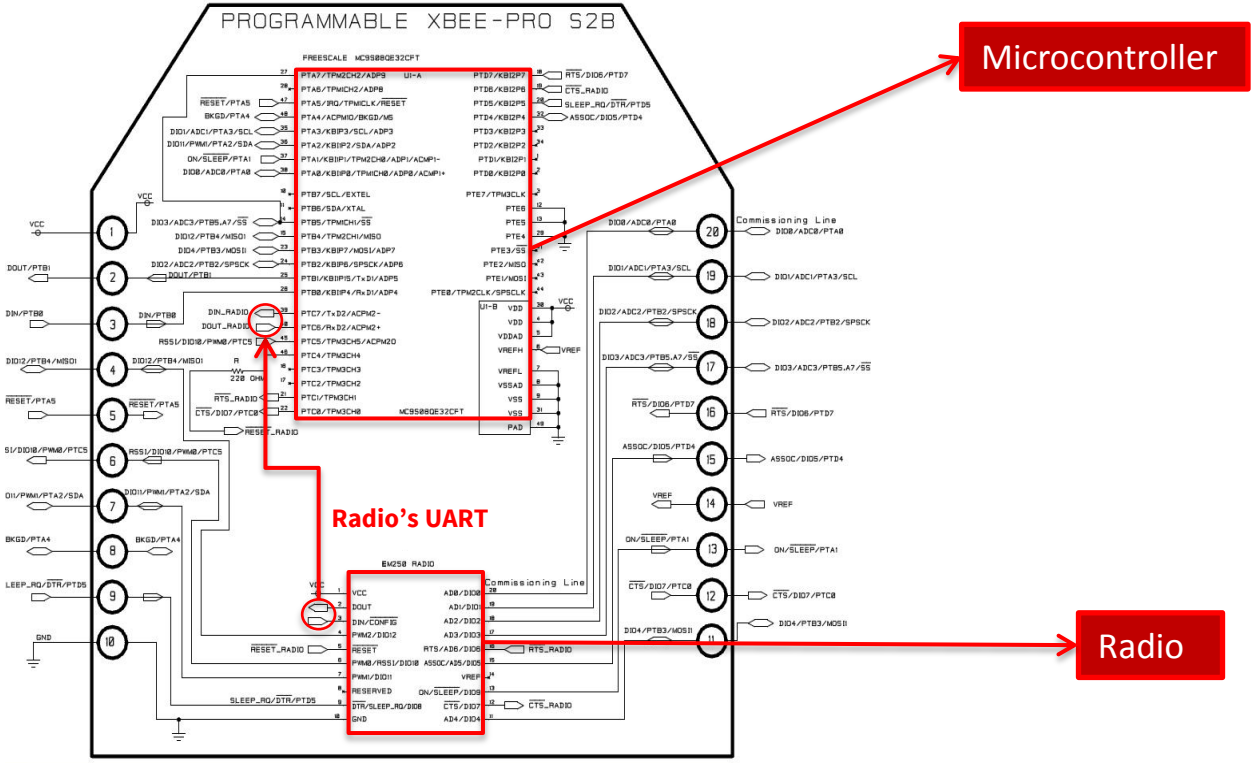


8 bit, HCS08 family, 50 MHz
32K FLASH 2K RAM

XBee-PRO 900HP

XBee ZigBee SMT

XBEE PROG BLOCK DIAGRAM (TH)



TWO INDEPENDENT SETS OF SIGNALS

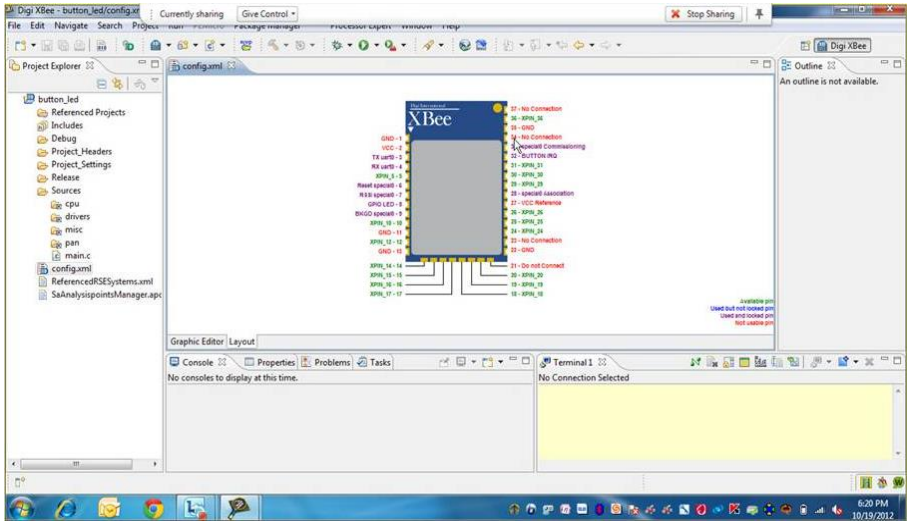
XBee Pin	XBee Pin Function	Radio Pin	Microprocessor Pin
1	VCC	1	4, 5, 30
2	DOUT/PTB1	No Direct Access	25
3	DIN/PTB0	No Direct Access	26
4	DIO12/PTB4/MIS01	4	15
5	!RESET/PTA5	No Direct Access	47
6	RSSI/DIO10/PWM0/PTC5	6	45
7	DIO11/PWM1/PTA2/SDA	7	36
8	BKGD/PTA4	No Connection	48
9	SLEEP_RQ/!DTR/PTD5	9	20
10	GND	10	7,8,9,31, 41
11	DIO4/PTB3/MOSI1	11	23
12	!CTS/DIO7/PTC0	No Direct Access	22
13	ON/!SLEEP/PTA1	No Direct Access	37
14	VREF	No Connection	6
15	ASSOC/DIO5/PTD4	15	32
16	!RTS/DIO6/PTD7	No Direct Access	18
17	DIO3/ADC3/PTB5.A7/!SS	17	14, 27
18	DIO2/ADC2/PTB2/SPSCK	18	24
19	DIO1/ADC1/PTA3/SCL	19	35
20	DIO0/ADC0/PTA0	20	38

BENEFITS OF THE XBEE PROG

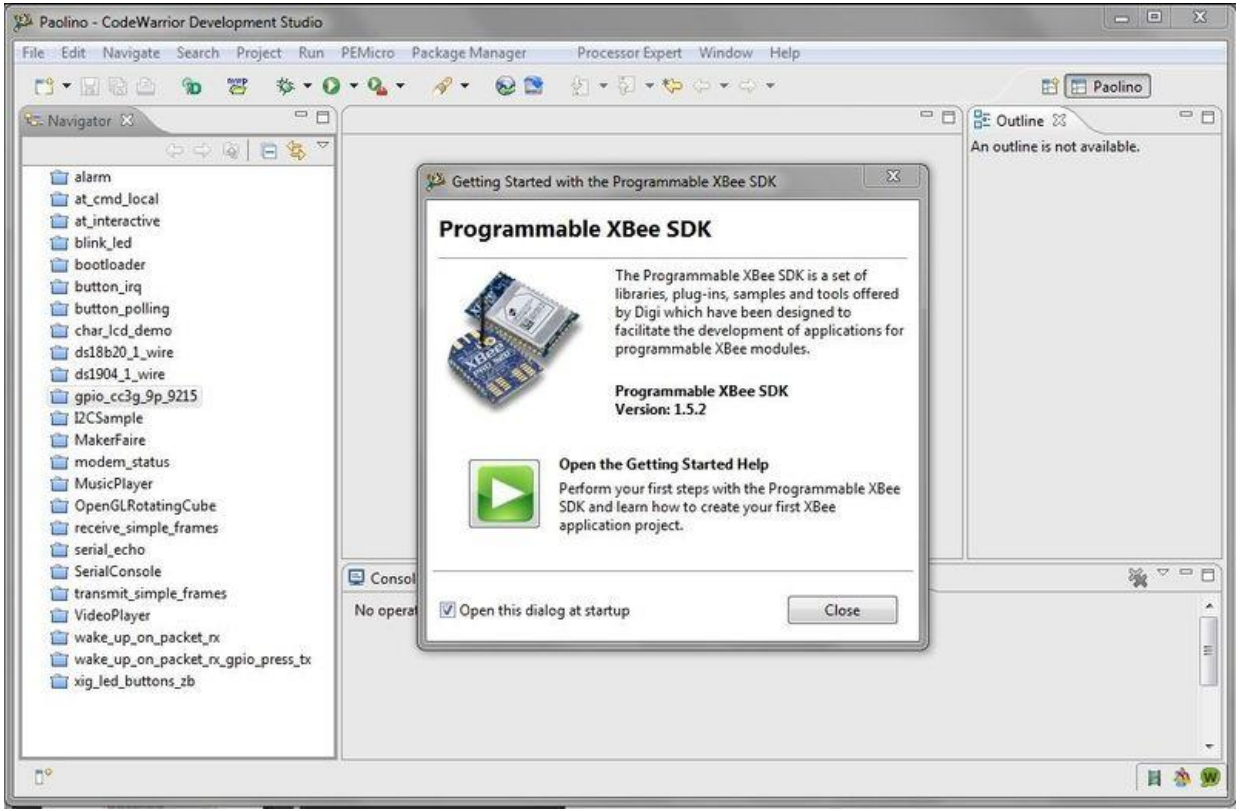
- It is like having an **XBee + Microcontroller** in one package!
- Ability to load a module with a customized **ANSI C code**
- Code can be changed over the air (see example in the SDK)
- Protected **Sandbox**: no risk of corrupting the radio firmware
- **Same certifications** will apply as standard XBee
- Radio code can be updated over the air (same as standard modules)
- Digi provides over **40 samples** in source code!!!
- Eliminates the need for **external host processor**
- Less HW complexity and reduced design time
- Supports for various XBee versions
- **Saving $$$** in the BOM
- Microcontroller is COMPLETELY independent from the radio!!!

XBEE PROGRAMMABLE SDK

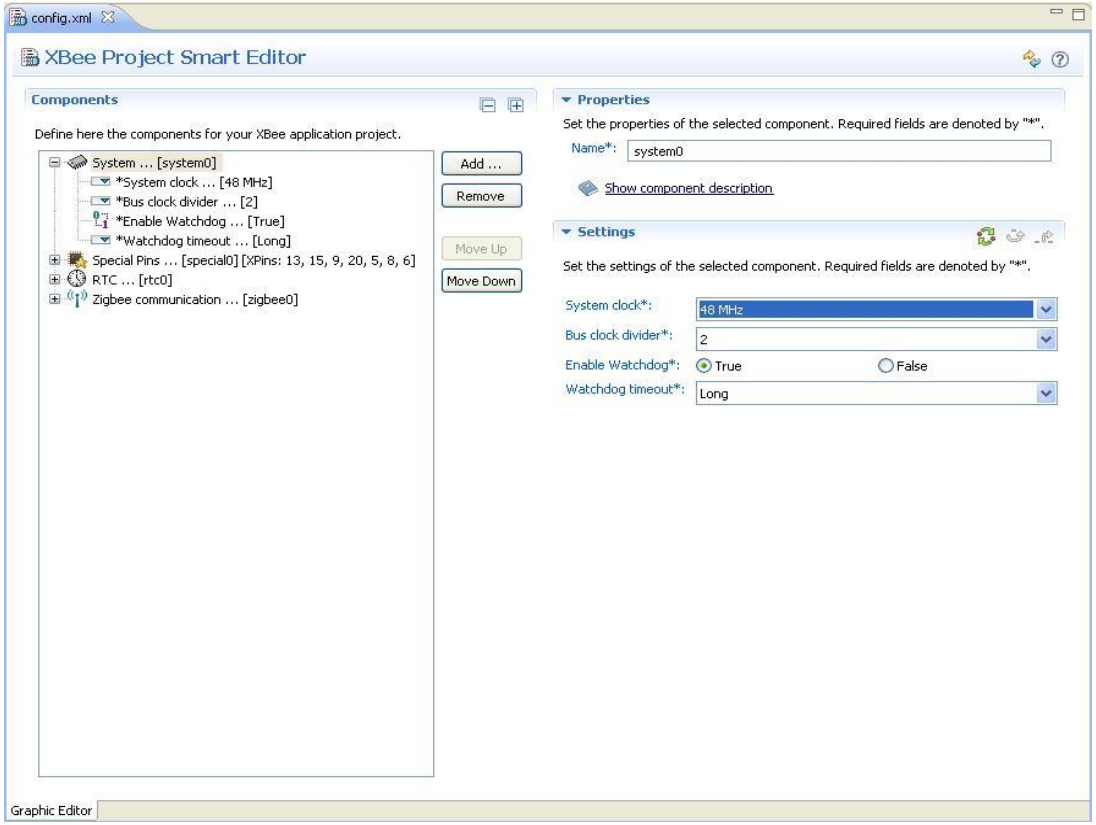
- Customized by Digi for XBee application development
- Based on Eclipse Framework, using CodeWarrior v10 compiler
- Full DevUI: edit, configure, build, and debug applications
- Debug using P&E Micro USB debugger
- Graphical Pin Configuration



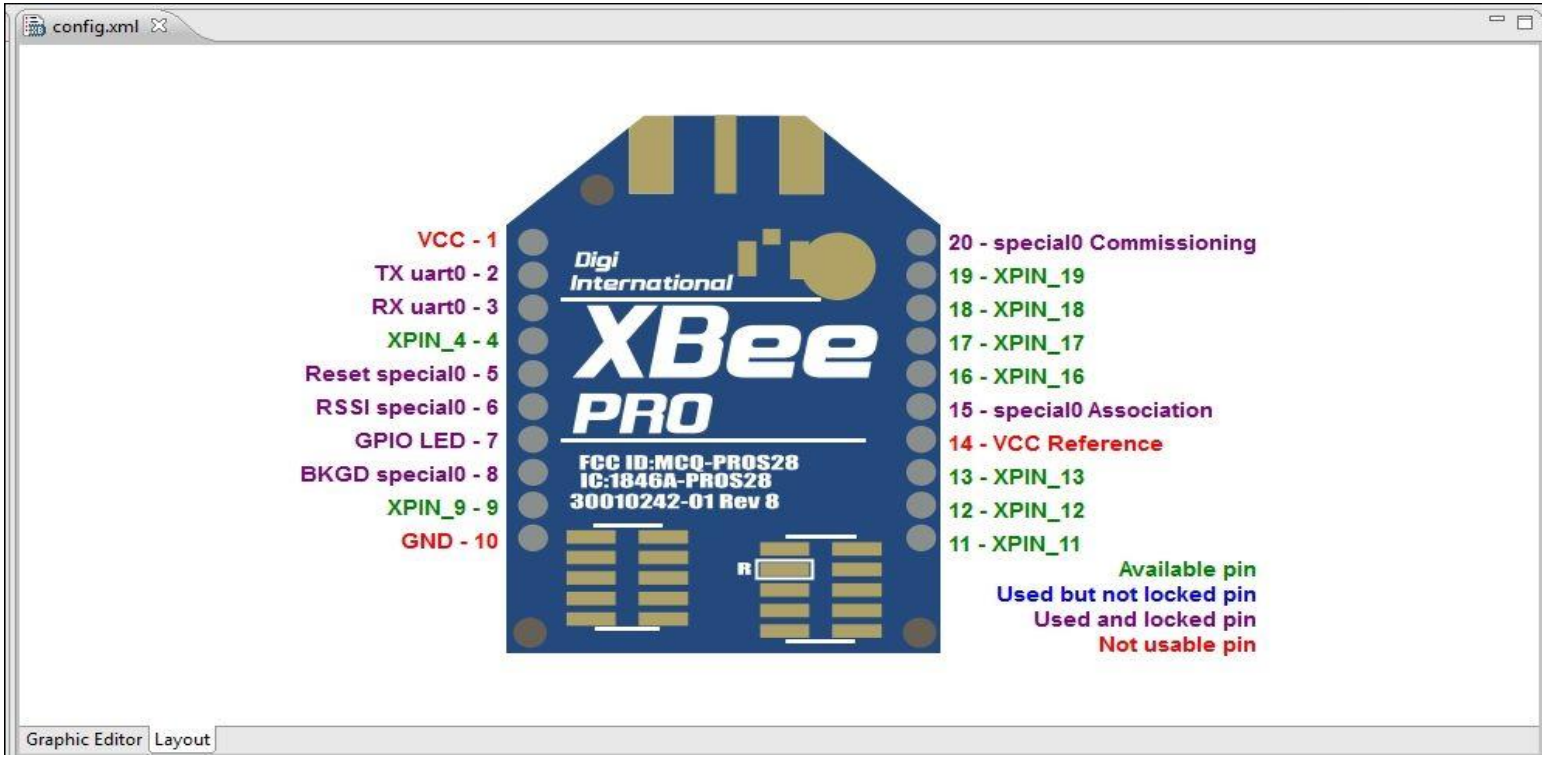
DEVELOPING WITH ECLIPSE



SMART EDITOR GUI



LAYOUT GRAPHIC VIEW



RF MODEMS, GATEWAYS AND ACCESSORIES

XBEE GATEWAY (FORMERLY X2e)

PROBLEM SOLVED

- IP connectivity to remote ZigBee networks

PRODUCT DESCRIPTION

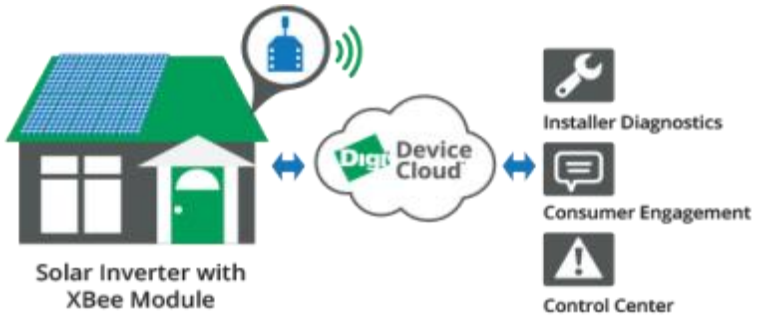
- FCC certified and Carrier End-device certified
- Secure WAN connectivity with SSL
- Cellular (using Telit 910), Wi-Fi and Ethernet WAN connectivity options
- Integration with Digi Device Cloud enables remote configuration and management of the complete network
- Easy local configuration via web-interface
- Based on a CC-Wi-i.MX28 running Linux OS
- Python programmable for custom application development

WHAT IS AN XBEE GATEWAY?

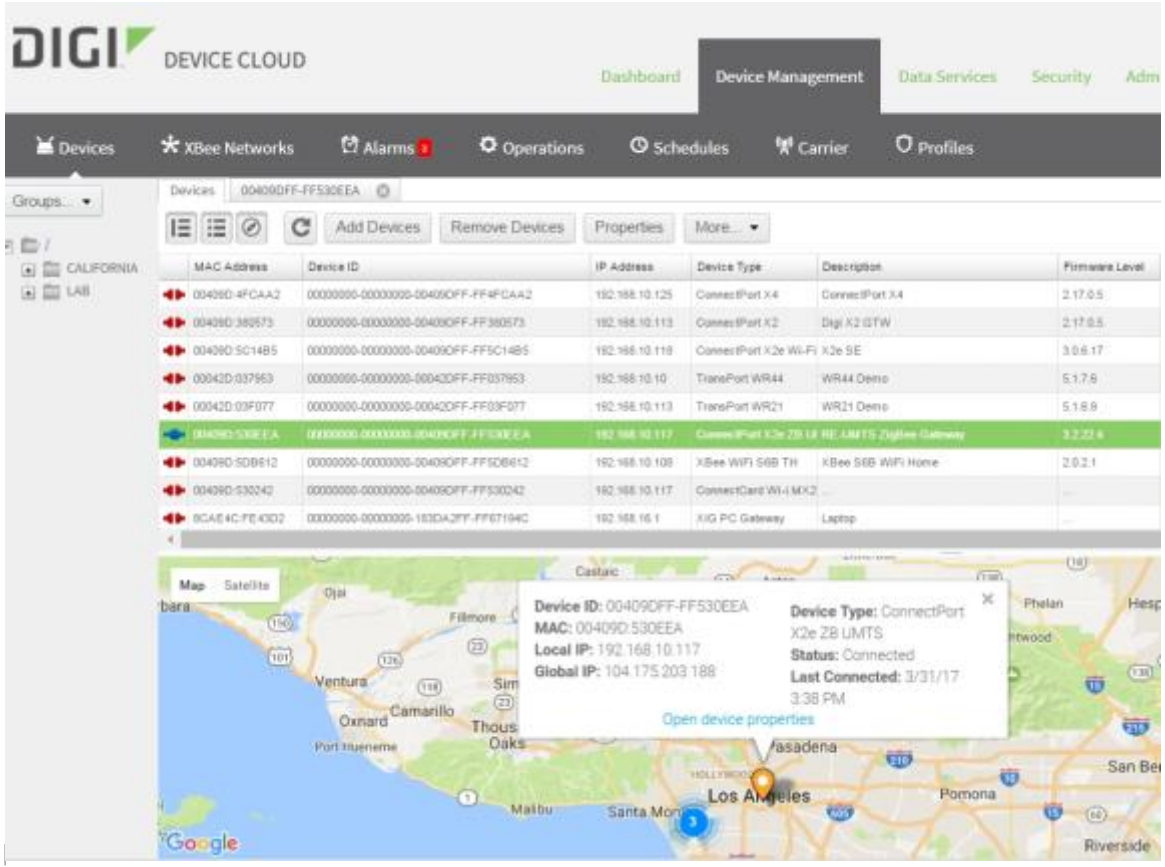
- A programmable ZigBee to IP gateway
- Is the coordinator of the network
- Ethernet + WiFi and Ethernet + Cellular 3G / UMTS models

EXTEND THE XBEE REACH!

- Once the XBee network is installed, there is usually the need to forward the data to the main network (IP)
- The best and easiest way to do so is by using a Digi Gateway
- The Digi Gateway converts and sends the XBee data into the IP network via Ethernet, WiFi or Cellular
- Once the data is in the IP network, it can be displayed, saved for history and easily managed (Cloud)
- Commands can also be sent back to the XBee modules!



XBEE GATEWAY AND DEVICE CLOUD



XBEE 900MHZ RF MODEMS

PROBLEM SOLVED

- High power, long range connectivity with mesh

PRODUCT DESCRIPTION

- ISM 900 MHz operating frequency
- Two models: compatibility with 900HP or with 900 SX
- Up to 65-mile range
- 256-bit AES Encryption
- Support for DigiMesh
- Enabled for Device Cloud
- Over-the-air firmware updates
- Various optional ports: analog, digital, RS232 and RS485
- Rugged metal enclosure and industrial temperature

DM

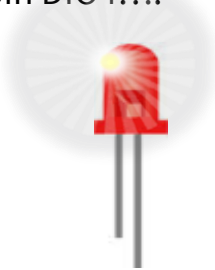
XBEE ADAPTERS AND EXTENDERS

- XBee adapters expand applications with full XBee connectivity and additional peripherals
- Many versions of adapters are available:
  - RS-232, RS-485, USB, XStick, GPIO, Analog I/O
  - AC SmartPlug, Temp, Light, Humidity, more…
- Battery option
- Routers - allowing extending the range
- End nodes – with/without sensors
- ZigBee and DigiMesh options

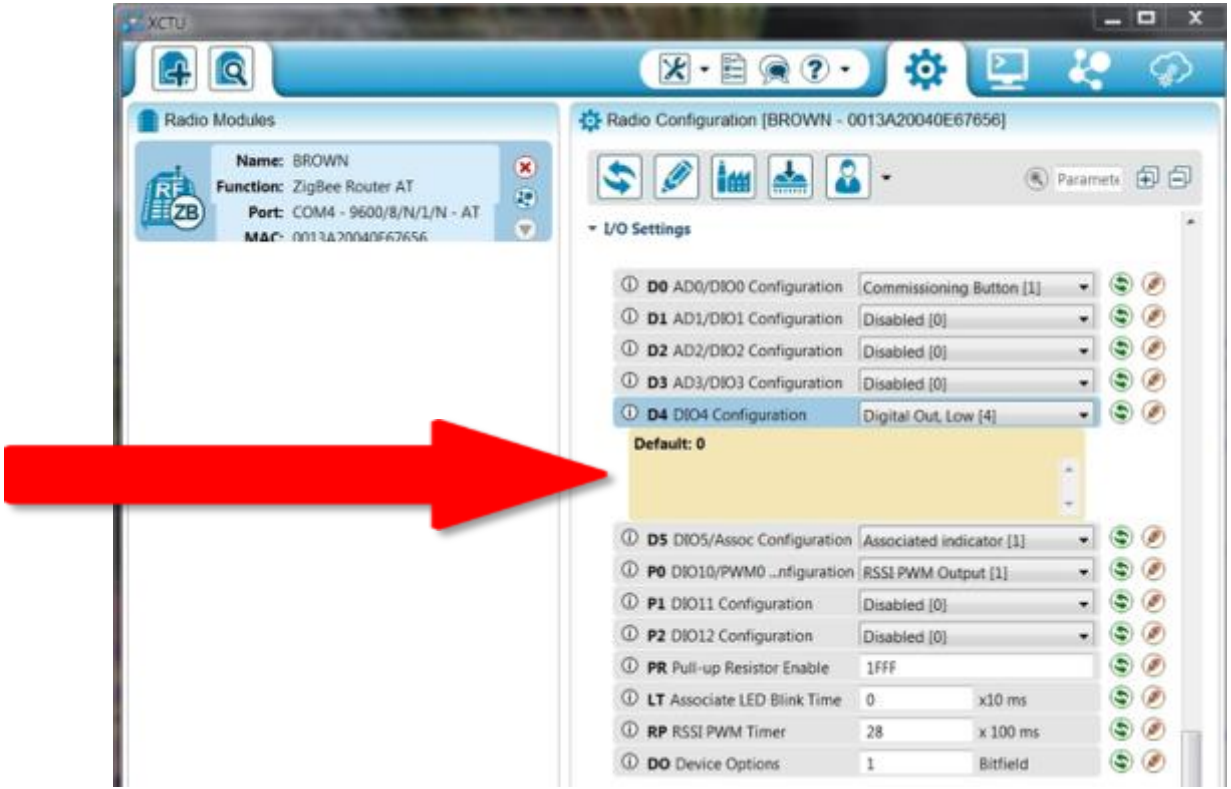
XBEE DEMOS

DEMO 1 – DIGITAL I/O

1) Connect the XBIB of an XBee module (can be any type) to the COM port (or USB port) on the computer
2) Launch XCTU
3) Click on the “Configuration” tab
4) Look for “I/O Settings”, then locate “DIO4”
5) Change the values and alternate from 0x04 (ON) to 0x05 (OFF)
6) Verify that the LED in the XBIB turns on and off
7) Also, can be done in the console via the commands:
   - ATD44
   - ATD45
8) Also, it is possible to send a **remote AT command**, via an API packet
   Example: 7E 00 10 **17** 01 0013A20040D9201E FF FE 02 **4434 05** 5F
8) Now think about what would happen if a light or a motor were connected to pin DIO4….



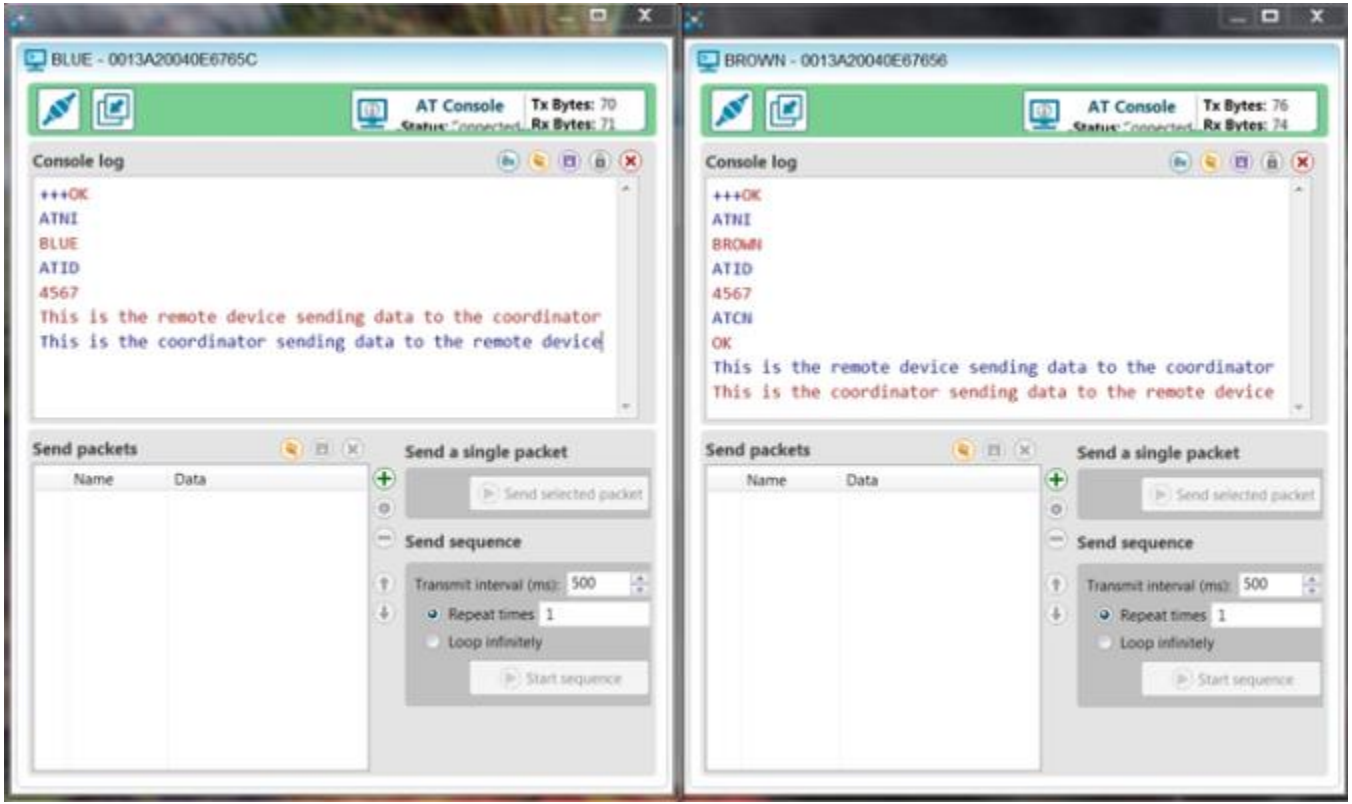
DEMO 1 - XCTU DIGITAL I/O



DEMO 2 - AT ECHO

1) Connect the XBIB of the Coordinator and Router to two COM ports (or USB ports) on different computers (or same one, it doesn't matter)
2) Launch 2 XCTU sessions, one per each COM port
3) Click "Configuration" tab
4) Located "Networking" (or "MAC/PHY" depends on the model) make sure the modules are associated to the same network
5) Under "Addressing", in the Coordinator, put the "Destination Address High" (DH) and "Low" (DL) of the other module
6) You don't need to change the DH and DL of the other module, as it will transmit to the Coordinator by default
   (in DM the data is transmitted in broadcast by default)
7) Go to the "Terminal" tab
8) Type some chars in both XCTU windows
9) Verify that what you're typing appears in the other window

DEMO 2 - AT ECHO



DEMO 3 – NETWORK MAP

1) Create a PAN with 3+ nodes: Coordinator, Router, Router (for DigiMesh, just 3+ nodes)
2) Launch XCTU
3) Make sure the module is in API mode
4) Go to the “Network” tab
5) Then click on “Scan”
6) Slowly the window will populate with the modules
7) The numbers show the signal (dBm) in each direction

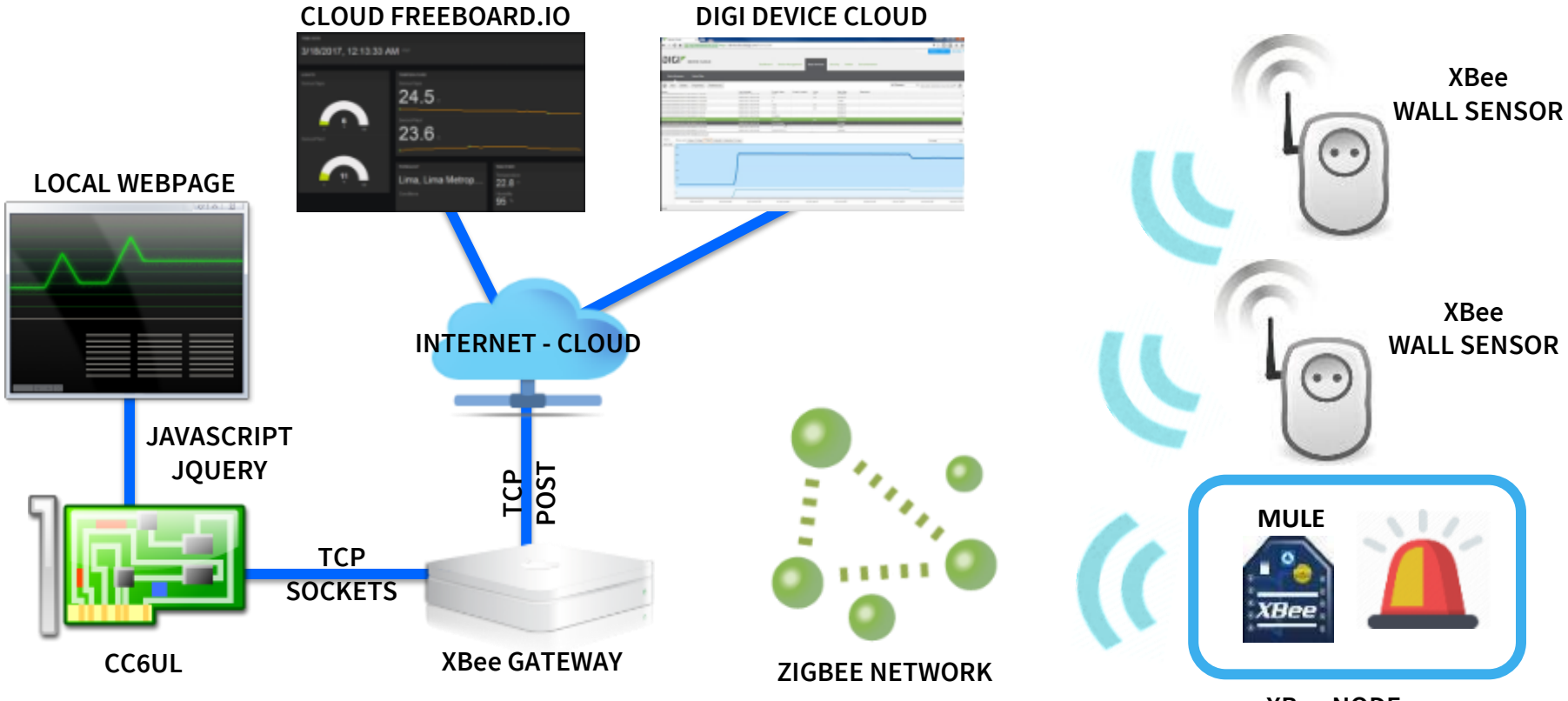
DEMO 4 – ZIGBEE SELF INSERTION

1) Create a PAN with 3 nodes: Coordinator, Router, Router
2) Launch XCTU
3) Go to the “Console” tab
4) Issue the command “ATND”
5) Verify that all modules are listed
6) Turn off one of the routers
7) Wait few secs
8) Issue again the command “ATND”
9) The Router shouldn’t be listed anymore
10) Turn the Router back on
11) Wait few secs
12) Issue again the command “ATND”
13) Verify that the Router is listed again

DEMO 5 – HOPPING

1) Create a PAN with 2 nodes: Coordinator, Router
2) Launch XCTU
3) Go to the “Terminal” tab
4) Issue the command “ATND”
5) Verify that all modules are listed
6) Move the Router far away from the Coordinator, until it loses the signal
7) Issue the command “ATND”
8) Now only the Coordinator is in the network
9) Add a new Router in the network
10) Position the new Router in the middle between the other two modules
11) Issue the command “ATND”
12) Verify that all the modules are listed again

DEMO 6 – GATEWAY + CLOUD + EMBEDDED



MORE DEMOS

- Many more demos are available online on the Digi Wiki:
  http://examples.digi.com/
  http://gallery.digi.com/
- Sparkfun has tutorials as well:
  https://learn.sparkfun.com/tutorials/exploring-xbees-and-xctu
- Arduino XBee tutorials:
  http://blog.arduino.cc/?s=xbee&btnG=search

EXAMPLE OF APPLICATIONS AND MARKETS

WHO IS THE DIGI CUSTOMER?

Small/Medium size businesses that lack in-house embedded RF Engineers

Requires quick time to market < 12 months

Plans to expand to other RF frequencies and protocols

APPLICATION EXAMPLES AND MARKETS

Precision Agriculture

Tank Monitoring

Oil & Gas

Digital Signage

Others: Lighting, UAV, Water monitoring and analisys

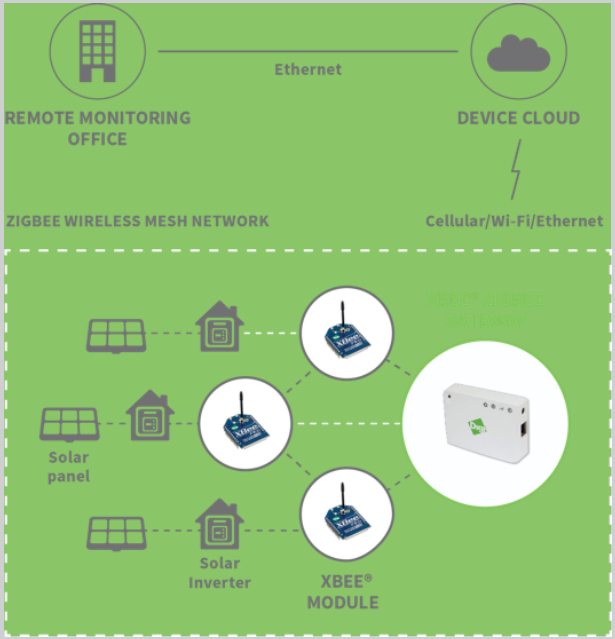
WHERE TO USE THE XBEE?

- XBee are very generic modules, can be employed in a multitude of different applications
- A/C power (i.e. XBee PRO) or battery powered (i.e. XBee)
- Some examples are:
  - **Industrial** Wire replacement (RS-232 or RS-485)
  - **Home Automation** temperature, humidity sensor
  - **Security** presence, door sensor
  - **Agriculture** humidity, irrigation sensor
  - **Tank Monitoring** water, oil levels metering
  - **Energy Generation** wind turbine, solar panel monitoring
  - **Energy Consumption** Energy meters
  - **Gambling** Lotto, Slot machine real time data acquisition
  - **Robots, Drones, RC Cars** Control and telemetry

EXAMPLE: SOLAR ENERGY

MONITOR SOLAR ENERGY PRODUCTION

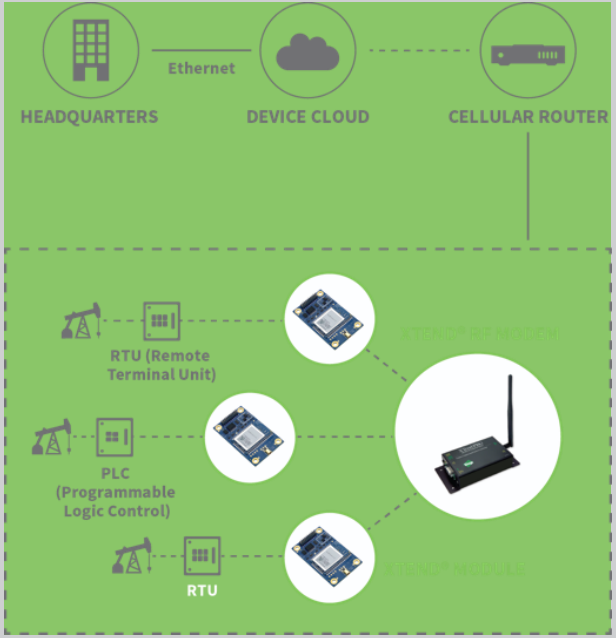
- Wirelessly enable remote monitoring and management
- Easy-to-install and replicate system allows for rapid expansion
- Reduced operational costs through predictive maintenance and reduced trucks movements
- Rapid integration and time-to-market



EXAMPLE: OIL FIELD

MANAGE ASSETS IN HARSH ENVIRONMENTS

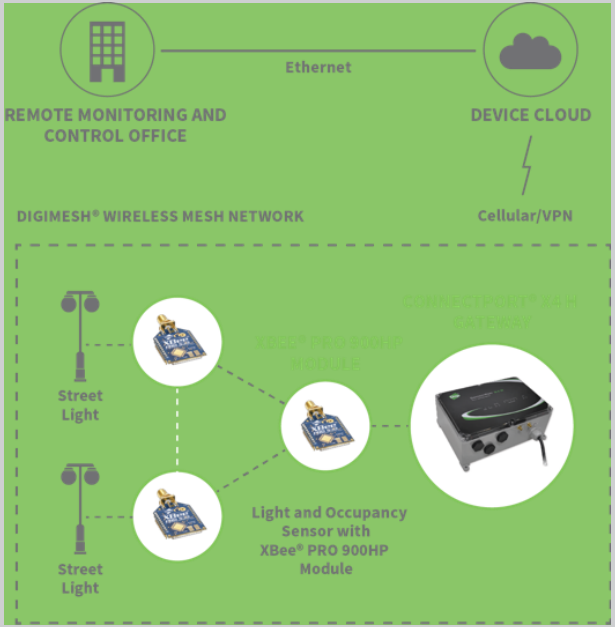
- Seamlessly connect assets, gather and integrate data
- Remotely manage and control the network devices for security compliance and system updating
- Reduced operational costs through predictive maintenance and reduced trucks movements
- Increase worker safety: no need to send humans



EXAMPLE: STREET LIGHTING

CONTROL AND MONITOR PUBLIC ASSETS

- Wirelessly connect existing infrastructure
- Flexible network options to meet global market requirements
- Proven wireless modules and gateways for complete remote connectivity to dispersed assets
- Lower energy and maintenance costs
- Lower CO2 emissions



EXAMPLE: AGRICULTURE

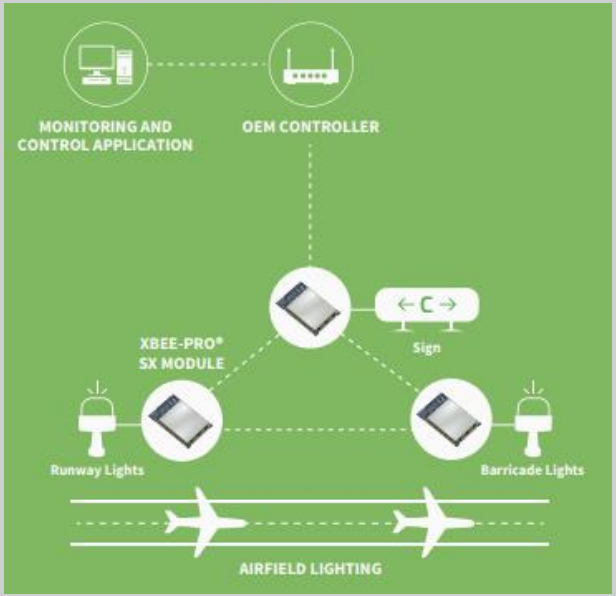
CONTROL AND MONITOR RESOURCES

- Temperature, level, humidity sensor networks
- Monitoring system provides real-time data
- Ensures maximum output and quality of stored goods, prevents costly losses
- Monitors tanks containing water, fertilizers, etc.
- Allows food management
- Reduces the trucks movement

EXAMPLE: AIRPORT LIGHTING

AIRFIELD LIGHTING SYSTEM

- Solar-powered lights and signs
- Distributed throughout military/civilian airfield
- Need to be monitored for failure
- Mission-critical: reliability of RF data is paramount



EXAMPLE: UAV

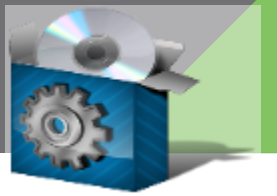
UNMANNED VEHICLES CONTROLS

- Wirelessly connect flying vehicles
- Civilian drones for hobbyists
- Commercial drones for delivery and safety services
- Long-Range, Reliable RF communications required for control and telemetry
- Gather data, statistics, telemetry
- Remote data transmission

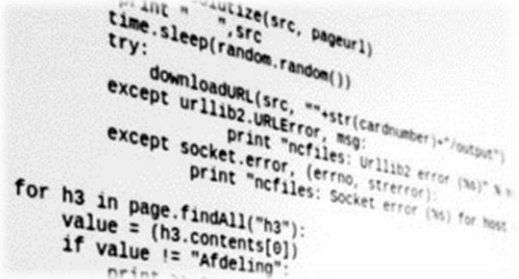
WHAT TO DO NEXT?

- There are many platforms that can be used as “intelligence” for the XBee family:
  - ✓ The XBee programmable
  - ✓ The Digi family (i.e. Connect ME, ConnectCard i.MX28 ConnectCore 6, etc.)
  - ✓ 3rd party: Arduino, etc…
- ALL of the HW platforms above come with **code libraries** to interface the XBee and read/write parameters, send data, receive data
- But ALL of the above require to write an **application** to manage the protocol and the data, and additional coding for **Cloud** connectivity
- **The other option is to use a Digi Gateway** (i.e. ZigBee Gateway)
- In this case the **protocol** is handled automatically by the Gateway and the **cloud connectivity** is automatic (Digi Remote Manager)

ADDITIONAL SOFTWARE LIBRARIES



- XBee have a great SW support for several HW platforms
- There are generic ANSI C libraries:
  https://github.com/digidotcom/xbee_ansic_library
- There are Java libraries:
  https://github.com/digidotcom/XBeeJavaLibrary
- There are Arduino-specific libraries (non-Digi):
  https://code.google.com/p/xbee-arduino/
- mBed
  https://developer.mbed.org/teams/Digi-International-Inc/code/XBeeLib/

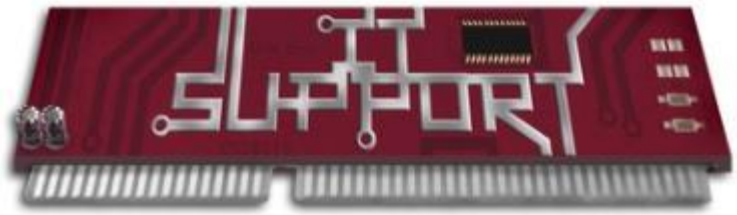


LIBRARIES FOR XBEE API PROTOCOL

open source

ARM mbed

python
node
ARDUINO
Java
ARM mbed
ANSI C



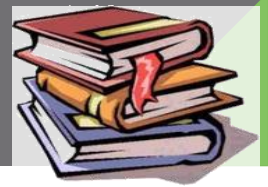
SUPPORT AND LEARNING

SUPPORT

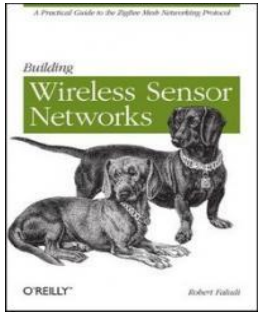
Tech Support by Digi:

- Support Ticket: http://www.digi.com/login?ReturnUrl=%2fsupport%2feservice%2flogin
- Product Support: http://www.digi.com/support/product-support
- Email: Rf-experts@Digi.com
- Chat: http://chat.digi.com/Chat/servlet/AppMain?__lFILE=TSChatForm.jsp
- Phone: +1-952-912-3444 or +1-801-765-9885

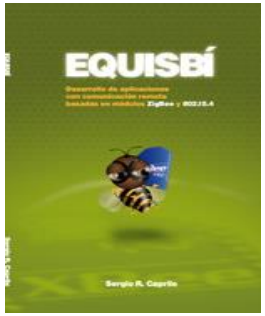
BOOKS



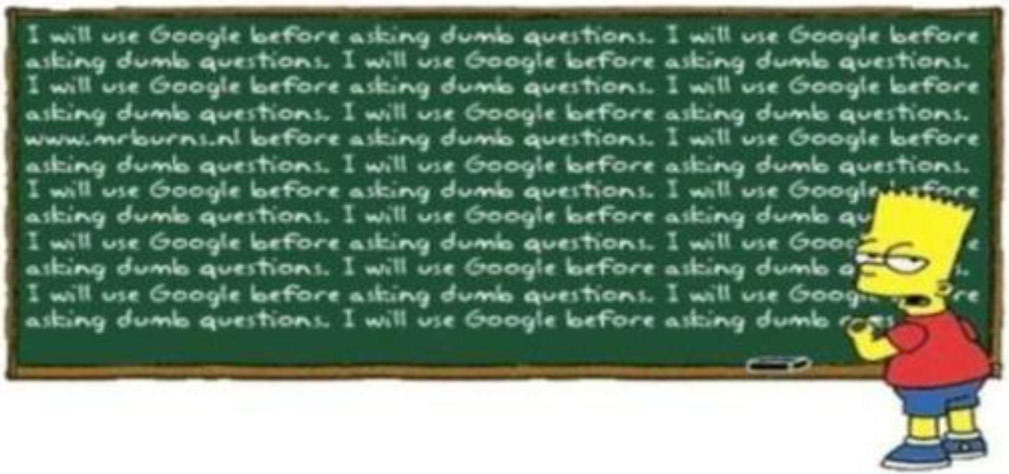
- **Building Wireless Sensor Networks**
  by Rob Faludi
  Publisher O'Reilly Media - Dec 2010
  ISBN 978-0-596-80773-3



- **Equisbí: Desarrollo de aplicaciones con comunicación remota basadas en módulos ZigBee y 802.15.4**
  by Sergio Caprile
  Published Sep 2009 - Spanish
  ISBN 978-987-1301-17-1




Questions?


I will use Google before asking dumb questions. I will use Google before
asking dumb questions. I will use Google before asking dumb questions.
I will use Google before asking dumb questions. I will use Google before
asking dumb questions. I will use Google before asking dumb questions.
www.mrburns.nl before asking dumb questions. I will use Google before
asking dumb questions. I will use Google before asking dumb questions.
I will use Google before asking dumb questions. I will use Google
asking dumb questions. I will use Google before asking dumb
I will use Google before asking dumb questions. I will use
asking dumb questions. I will use Google before asking dumb
I will use Google before asking dumb questions. I will use
asking dumb questions. I will use Google before asking dumb

CONGRATULATIONS!!!

- You are now a…

Digi XBee Graduate!